User's Manual

32

RENESAS

SH7455 Group, SH7456 Group

User's Manual: Hardware

Renesas 32-Bit RISC Microcomputer
SuperH^{TM} RISC engine Family

SH74552	R5F74552KBG
SH74562	R5F74562KBG

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.1.10 Sep 2011

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of 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. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system 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. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

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

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins
   Handle unused pins in accord 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 LSI, an associated shoot-through current flows internally, and malfunctions may occur due to the false recognition of the pin state as an input signal. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.
5. Differences between Products
   Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.
   — The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

This page is blank for reasons of layout.

Revision History	SH7455 Group, SH7456 Group User's Manual: Hardware

Rev.	**Date**	**Page of Previous Edition**	**Description**
Rev.1.10	Sep. 22, 2011	—	First edition issued.

This page is blank for reasons of layout.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the hardware functions and electrical characteristics of the MCU. It is intended for users designing application systems incorporating the MCU. A basic understanding of electrical circuits, logic circuits, and microcontrollers is required to use this manual.

This manual includes an overview of the product; descriptions of the CPU, system control functions, peripheral functions, and the electrical characteristics; and usage notes.

Particular attention must be paid to the precautionary notes included in this manual. These notes appear in the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the main places that have been revised and added since the first edition. It does not list all revisions. Refer to the text of the manual for details.

Renesas provides the following documents for the SH7455 Group and SH7456 Group. Be sure to refer to the latest versions of these documents. The most recent versions of these documents can be obtained from the Renesas Electronics web site.

Document Type	Description	Document Title	Document No.
Datasheet	Hardware overview and electrical characteristics	—	—
User's manual for hardware	Hardware specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, and timing charts) and operation descriptions	SH7455 Group, SH7456 Group User's Manual: Hardware	This user's manual
User's manual for software	Descriptions of the CPU and instruction set	SH-4A Extended Function Software Manual	REJ09B0224
Application note	Sample application programs and other materials	—	
Renesas technical update	Product specifications, the latest updates on documents, and other information		

2. Notation of Numbers and Symbols

The notational conventions for register names, bit names, numbers, and symbols used in this manual are described below.

(1) Register Names, Bit Names, and Pin Names

Registers, bits, and pins are referred to in the text by symbols. These symbols are followed by the word "register", "bit", or "pin" to distinguish the three categories.

Examples: the CS# pin, the Vcc pin,
the ICR0 register

(2) Notation of Numbers

Binary numbers are notated as B'nnnn (However, the "B" may be omitted when it is clear that the number is binary), hexadecimal numbers are notated as H'nnnn, and decimal numbers are notated as nnnn.

Examples - Binary: B'11 or 11
Hexadecimal: H'EFA0
Decimal: 1234

(3) Notation for "L" Active (Active-Low) Signals

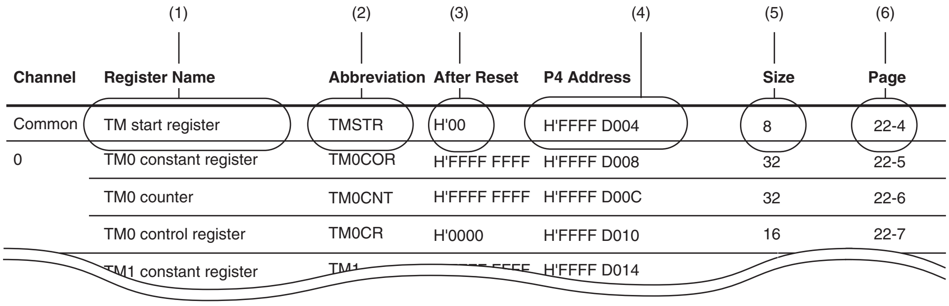
A sharp sign (#) is appended to the names of signals and pins which are "L" active.

Example: the CS0# pin

3. Register Configuration

Each section in this manual provides a table listing all the registers used by the corresponding module before the register descriptions in the section. The symbols and terms used in these tables are described below.

[Register Table]



Channel	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Common	TM start register	TMSTR	H'00	H'FFFF D004	8	22-4
0	TM0 constant register	TM0COR	H'FFFF FFFF	H'FFFF D008	32	22-5
	TM0 counter	TM0CNT	H'FFFF FFFF	H'FFFF D00C	32	22-6
	TM0 control register	TM0CR	H'0000	H'FFFF D010	16	22-7
	TM1 constant register	TM1		H'FFFF D014		

Note: • The bit names and text in the figure above are examples that are unrelated to the content of this manual.

(1) Register Name
The register name are shown for each register.

(2) Abbreviation
Gives the name of the register.

(3) After Reset
Indicates the values of each bit after a hardware reset in hexadecimal.

(4) P4 Address
Indicates the P4 address of each register.
The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

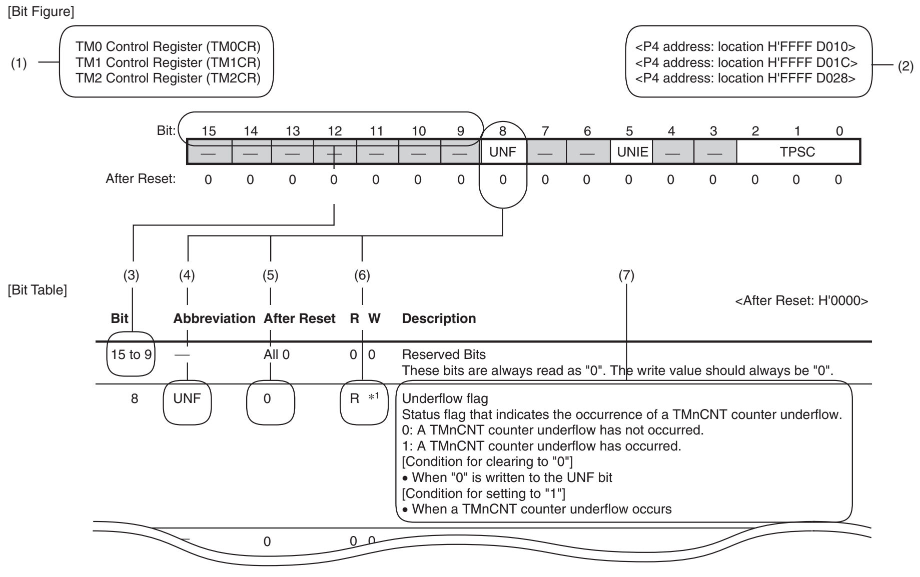
(5) Access Size
8-bit access is indicated as "8", 16-bit access as "16", and 32-bit access as "32", respectively.
For registers that allow multiple accesses, each access size is indicated with a slash (/).
If an access size is indicated without a slash (/), only the indicated size is allowed.

- For 32-bit registers that can be accessed using 32-bit and 16-bit accesses
  The access size is indicated as "16/32".
- For 8-bit registers that can be accessed using 8-bit access, and also using 16-bit access at the same time with the next aligned 8-bit register
  The access size is indicated as "8/16".

(6) Page
Indicates the page on which the functions and bit settings of the register are described. If the description runs to multiple pages, only the first page is indicated. For example, if the description starts on page 12-9 and continues to page 12-10, page 12-9 is indicated.

4. Register Notation

Each register description includes both a bit figure that shows the bit sequence and a bit table that describes the content set with each bit. The symbols and terms used are described below.



[Bit Figure]

(1)
TM0 Control Register (TM0CR)
TM1 Control Register (TM1CR)
TM2 Control Register (TM2CR)

(2)
<P4 address: location H'FFFF D010>
<P4 address: location H'FFFF D01C>
<P4 address: location H'FFFF D028>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UNF	—	—	UNIE	—	—	TPSC		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Bit Table]

(3) (4) (5) (6) (7)

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 9	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
8	UNF	0	R	*1	Underflow flag Status flag that indicates the occurrence of a TMnCNT counter underflow. 0: A TMnCNT counter underflow has not occurred. 1: A TMnCNT counter underflow has occurred. [Condition for clearing to "0"] • When "0" is written to the UNF bit [Condition for setting to "1"] • When a TMnCNT counter underflow occurs
	—	0	0	0	

Note: • The bit names and text in the figure above are examples that are unrelated to the content of this manual.

(1) Register Name

Indicates the name and abbreviation of each register.

(2) Register Address

Indicates the P4 address of each register.
The P4 address applies when the P4 area of the virtual address space is used.
When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address.
This address is used when accessing area 7 of the physical address space using the TLB.

(3) Bit

Indicates the bit number.
The bits are shown in order from 31 to 0 for 32-bit registers and from 15 to 0 for 32-bit registers.

(4) Abbreviation

Indicates the name of the bit or field.
Reserved bits are indicated with a dash ("-") and are shaded in the bit figure.
Note that there are cases, such as time counters, where no bit names are indicated and the field is left blank.

(5) After Reset

Indicates the value of each bit after a hardware reset. The value is shown in binary unless specified otherwise.
0: Indicates that the value is "0" after a reset.
1: Indicates that the value is "1" after a reset.
Undefined: Indicates that the value is undefined after a reset.

(6) R and W

Indicates whether the bit(s) can be read, written to, or neither read nor written to.
The notations used are listed below. When a register contains bits with the notation N (write prohibited) in the write (W) column, the entire register is read-only, regardless of the notations for the other bits.

- R column notations

R: Readable
?: Value undefined when read
0: Value "0" when read
1: Value "1" when read
*1: Other (Refers to a note below the bit description table.)

- W column notations

W: Writable
0: Always write "0"
1: Always write "1"
—: Write prohibited
*1: Other (Refers to a note below the bit description table.)

(7) Description

Describes the function of the bit or field.

5. Abbreviations and Symbols

The table below describes the abbreviations and symbols used in this document.

Abbreviation	Full Name
FPU	Floating-Point Unit
MMU	Memory Management Unit
SHwyRAM	Super Hyway RAM
CPG	Clock Pulse Generator
INTC	Interrupt Controller
WDT	Watchdog Timer
BSC	Bus State Controller
DMAC	Direct Memory Access Controller
ATU-IIIS	Advanced Timer Unit IIIS
TMU	Timer Unit
SCIF	Serial Communication Interface
RSPI	Renesas Serial Peripheral Interface
IIC3	I2C Bus Interface 3
CAN	Controller Area Network
ADC	A/D Converter
DRI	Direct RAM Input Interface
DRO	Direct RAM Output Interface
PDAC	Parallel DAC Controller
PSEL	Parallel Selector
UBC	User Break Controller
AUDR	AUD RAM Monitor
H-UDI	User Debugging Interface

6. Trademarks and Registered Trademarks

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

Contents

Section 21 Advanced Timer Unit IIIS (ATU-IIIS) 21-1

Section 22 Timer Unit (TMU)..22-1

Section 23 Serial Communication Interface with FIFO (SCIF)..23-1

Section 1 Overview

1.1 Features

The SH7455 Group and SH7456 Group are a single-chip RISC (reduced instruction set computer) microcontroller based on a Renesas original RISC CPU core. It provides an extensive set of peripheral functions necessary for implementing application systems integrated on the same chip.

The CPU used in the SH7455 Group and SH7456 Group features a RISC instruction set and adopts a superscalar architecture to radically increase instruction execution speed. It uses the Renesas SH-4A as the CPU core and can implement high-performance/high-functionality systems at low cost, even for real-time control and other applications that require high-speed performance and could not be implemented with earlier microcontrollers.

The SH7455 Group and SH7456 Group include 32 Kbytes of instruction cache and 32 Kbytes of operand cache. Either copy-back or write-through can be selected for the operand cache. It furthermore includes a memory management unit (MMU) that can access a 4 GB address space. It includes a 4-entry fully associative TLB for instructions and a 64-entry fully associative TLB for both instruction and operands.

The SH7455 Group and SH7456 Group also include 8 Kbytes of IL memory, 16 Kbytes of OL memory and 256 Kbytes of SuperHyway RAM (SHwyRAM) as internal SRAM. The IL and OL memory units are capable of high-speed access and can be used as system stack required high performance.

The SH7455 Group and SH7456 Group provide a direct RAM interface function (DRI) that transfers parallel data directly to internal SHwyRAM and can transfer data input from, for example, an image sensor, to internal SHwyRAM.

The SH7455 Group and SH7456 Group integrate on the same chip a wide range of peripheral functions necessary for system construction. These include a floating point unit (FPU), large-capacity ROM and RAM blocks, a direct memory access controller (DMAC), several types of timer, the Renesas serial peripheral interface (RSPI), a user break controller (UBC), a RAM monitor function, a serial communication interface with FIFO (SCIF), a controller area network (CAN), A/D converters (ADC), a DAC interface function, an interrupt controller (INTC), and I/O ports.

Programs of a ROM (F-ZTAT™ version flash memory) can be downloaded or erased either using a ROM programmer or with software. This means that the user can reload software with the SH7455 Group and SH7456 Group mounted on a board.

Note: • F-ZTAT is a trademark of Renesas Electronics Corp.

1.1.1 Applications

Automobile equipment control (driver-assist systems, etc.) and industrial equipment system control.

1.1.2 Specifications Overview

Table 1.1 lists an overview of the SH7455 Group and SH7456 Group specifications.

Table 1.1 Specifications Overview

Item	Description
CPU	• Renesas original Super-H architecture • Compatibility at the object code level between the SH-1, SH-2, SH-3, and SH-4. • 32-bit internal data bus • General-purpose register file 16 32-bit general-purpose registers (and 8 32-bit shadow registers) 7 32-bit control registers 4 32-bit system registers Register banks for rapid interrupt response • RISC type instruction set (upwards compatible with the SH series) Instruction length: fixed 16-bit length for improved code efficiency Load/store architecture Delayed branch instructions Conditional execution Instruction set based on the C programming language • Two-instruction simultaneous execution superscalar architecture, including FPU • Instruction execution time: a maximum of two instructions per cycle • Address space: 4 GB • Address space identifier (ASID): 8 bits for 256 virtual address spaces • Built-in multiplier • Eight stage pipeline • Harvard architecture

Item	Description
FPU	• Built-in floating point coprocessor (FPU) • Single precision (32 bits) and double precision (64 bits) support • Supports IEEE 754 standard data formats and exceptions • Rounding mode: round to nearest and round toward zero • Handling unnormalized numbers: truncating towards zero, generating an interrupt to conform to the IEEE 754 standard • Floating point registers: 32 bits × 16 registers × 2 banks (16 standard precision or 8 double precision) × 2 banks • 32-bit CPU-FPU floating point communication register (FPUL) • Supports an FMAC (multiply and accumulate) instruction • Supports the FDIV (division) and FSQRT (square root) instructions • Supports the FLDI0/FLDI1 (load the constants 0 or 1) instructions • Instruction execution times Latency (FADD/FSUB): 3 cycles (single precision), 5 cycles (double precision) Latency (FMAC/FMUL): 5 cycles (single precision), 7 cycles (double precision) Pitch (FADD/FSUB): 1 cycle (single precision), 1 cycle (double precision) Pitch (FMAC/FMUL): 1 cycle (single precision), 3 cycles (double precision) Note: • FMAC is only supported for single precision. • 3D graphics instructions (single precision only) Four-dimensional vector transformation and matrix operation (FTRV): 4 cycles (pitch), 8 cycles (latency) Four-dimensional vector inner product (FIPR): 1 cycle (pitch), 5 cycles (latency) • 11-stage pipeline
Memory management unit (MMU)	• 4 GB address space, 256 address space identifiers (ASID: 8 bits) • Single virtual memory mode and multiple virtual memory mode • Supports multiple page sizes: 1 Kbyte, 4 Kbytes, 8 Kbytes, 64 Kbytes, 256 Kbytes, 1 Mbyte, 4 Mbytes, and 64 Mbytes. • Four-entry fully associative TLB for instructions • 64-entry fully associative TLB for instructions and operands • Supports both software based replacement methods and random counter based replacement algorithms • The TLB content can be accessed directly with address mapping • Access rights check function
Cache memory	• Instruction cache (IC) 32 Kbytes, 4-way set associative 256 entries/way, 32-byte block length Low-power function (way prediction structure) • Operand cache (OC) 32 Kbytes, 4-way set associative 256 entries/way, 32-byte block length • Single stage copy-back buffer, single stage write-through buffer Store queue (32 bytes × 2 entries)

Item	Description
IL memory	• 8-Kbyte high-speed access RAM • Two-page structure • Allows read/write access from following three ports. SuperHyway bus Cache/RAM internal bus Instruction bus • Allows CPU access to 8, 16, 32, and 64-bit operands • Allows accesses in 8, 16, 32, and 64-bit, as well as 16 and 32-byte, units by external requests
OL memory	• 16-Kbyte high-speed access RAM • 4-page structure • Allows read/write access from following three ports. SuperHyway bus Cache/RAM internal bus Operand bus • Allows CPU access to 8, 16, 32, and 64-bit operands • Allows accesses in 8, 16, 32, and 64-bit, as well as 16 and 32-byte, units by external requests
ROM	• 1-Mbyte flash memory
RAM	• 256-Kbyte SRAM
Operating modes	• Operating mode Single-chip mode • On-board programming modes User mode Boot mode User boot mode • Processing states Reset state Instruction execution state
User break controller (UBC)	• Supports debugging with a user break interrupt • Two break channels • The address, data value, access type, and data size can all be used as break conditions. • Supports a sequential break function
Clock generator (CPG)	• Internal clocks CPU clock (Ick): 160 MHz maximum SHwy clock (SHck): 80 MHz maximum Peripheral clock (Pck): 40 MHz maximum Peripheral A clock (PAck): 80 MHz maximum FlexRay clock (FRck): 80 MHz maximum • Input clock frequency: 20 MHz maximum
Bus state controller (BSC)	• The SH7455 Group and SH7456 Group do not include a BSC function

Item	Description
Watchdog timer (WDT)	• One channel • In watchdog timer mode, a reset is issued internally by a counter overflow and the WDTOVF# signal is output. • In interval timer mode, the interval timer interrupt is generated by a counter overflow.
Interrupt controller (INTC)	• Interrupt priority IRQ interrupt (IRQ0 to IRQ2 and IRQ5 to IRQ7): 15 levels On-chip peripheral module interrupt: 30 levels
Direct memory access controller (DMAC)	• 6 channels (DMA0 to DMA5) + 6 channels (DMA6 to DMA11) • Transfer data size: 1 byte, 2 bytes (word), 4 bytes (long word), 16 bytes, 32 bytes • Maximum number of transfers: 16,777,216 • Transfer address method: dual address • Transfer modes: cycle stealing mode 1, cycle stealing mode 2, or burst mode • Transfer request sources: automatic request (software request), on-chip peripheral module request (SCIF, RSPI, IIC3, ATU-IIIS, ADC, DRI), and external request (DMA0, DMA2, and DMA3 only) • Priority between modules The priority between the DMAC0 module (DMA0 to DMA5) and the DMAC1 module (DMA6 to DMA11) is round robin. • Channel priority within a module Either a fixed priority (DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5) or round robin can be selected for DMA0 to DMA5. Either a fixed priority (DMA6 > DMA7 > DMA8 > DMA9 > DMA10 > DMA11) or round robin can be selected for DMA6 to DMA11.
Advanced timer unit IIIS (ATU-IIIS)	• 59 channels • Provides timer A (5 channels × 2 systems), timer F (3 channels), timer G (6 channels), and timer TOU (8 channels × 5 systems)
Timer unit (TMU)	• Three auto-reload 32-bit timer channels • Each channel can select one of five counter input clocks: one of 5 peripheral clocks (Pck/4, Pck/16, Pck/64, Pck/256, or Pck/1024) (Note: Pck is the peripheral clock)
Serial communication interface with FIFO (SCIF)	• Four channels • Transmit and receive FIFOs each with 16 bytes • Supports both asynchronous and synchronous modes • Supports full-duplex communication • The transfer clock can be selected from either an internal clock from a baud rate generator or an external clock input to the SCK pin • Maximum transfer rate in clock synchronous mode: 3.3 Mbps • Maximum transfer rate in asynchronous mode: 5 Mbps
Renesas serial peripheral interface (RSPI)	• Three channels • Synchronous serial communication • Supports both master and slave modes • Programmable bit length, clock polarity, and clock phase • Supports sequential iterative execution of transfer operations • Supports both MSB first and LSB first transfer • Maximum transfer rate: 10 Mbps

Item	Description
I^{2}C interface (IIC)	• One channel • Supports the Philips proposed I^{2}C bus (Inter IC Bus) interface standard • Master and slave functions
Controller area network (CAN)	• Four channels • 64 mailboxes
A/D converter (ADC)	• Two modules • 12 bits, 16 channels (AD0: 12 channels, AD1: 4 channels) • Provides three conversion modes Continuous scan mode Single cycle scan mode A/D conversion value summation mode (performs an A/D conversion of the same channel 2 to 4 times and adds the converted values) • Conversion times When AVcc = 5 V: 1.25 μs When AVcc = 3.3 V: 1.25 μs • Absolute error When AVcc = 5 V and high-speed conversion: ±16LSB When AVcc = 3.3 V and high-speed conversion: ±32LSB
Parallel DAC control (PDAC)	• One channel • 10-bit parallel output • This is a parallel DAC control circuit that controls a 10-bit D/A converter • Generates modulation A, modulation B, and modulation C output waveforms.
Direct RAM input interface (DRI)	• Three channels • Two pin groups • Acquisition timing adjustment function • Decimation control function • Minimum acquisition period: 25 ns (special mode enabled)
Direct RAM output interface (DRO)	• One channel • Reads SHwyRAM and outputs parallel data to off-chip circuits • Data width: 8 or 16 bits • Maximum transfer speed: 20 Mbytes/s (when 16 bits is selected and, a 10 MHz of DRO transfer clock is specified by a register setting)
Parallel selector (PSEL)	• One channel • This is a parallel selector circuit that periodically changes the external selector outputs • This module is activated by an activation event and stopped by either a stop command or a termination event • 4-bit selector output • Two clock output and one clear signal output
FlexRay	• Two channels: SH7455 Group • None: SH7456 Group
AUD RAM monitor function (AUDR)	• Functions for reading/writing memory mapped modules connected to an internal or external bus • Parallel 4-bit data input and output • Transfer frequency: 12.5 MHz maximum

Item	Description
I/O ports	• Number of ports: 108 • Built-in input threshold value switching function (0.35, 0.5, or 0.7 × Vcc)
Module stop function	• Supports the module stop function for the PDAC, PSEL, DRI, and DRO modules.
User debugging interface	• H-UDI (User Debugging Interface)
Supply voltage	• Internal logic voltage: 1.5 V +0.15 V, -0.1 V • I/O voltage: 3.3 V ±0.3 V or 5.0 V ±0.5 V
Operating temperature	• Ta = -40 to +125°C
Package	• PRBG0176GA-A (0.8 mm pitch)

1.2 Product Line Overview

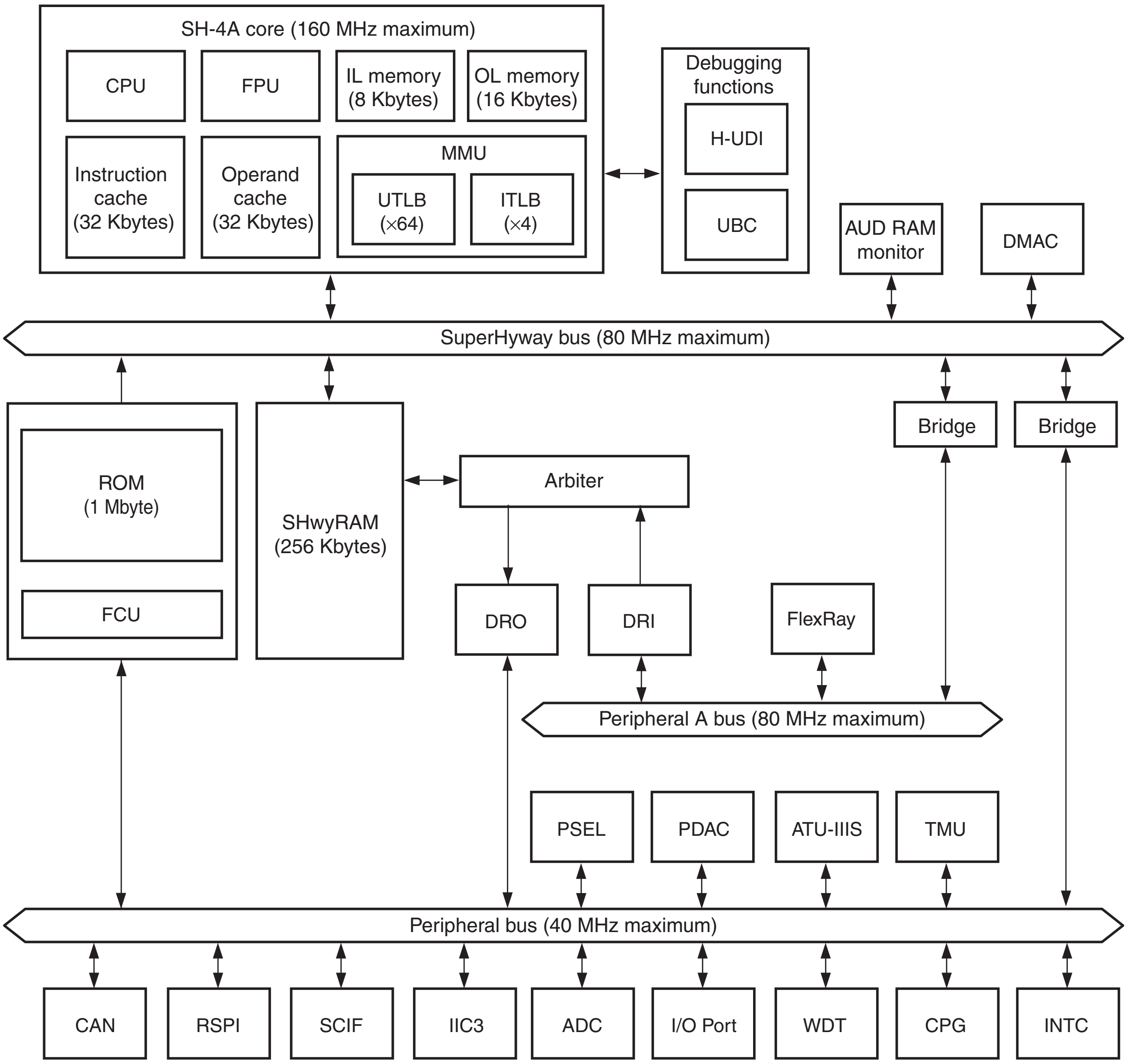
Table 1.2 lists the products.

Table 1.2 Products

Product	Model	ROM Capacity	RAM Capacity	Package	FlexRay
SH74552	R5F74552KBG	1 Mbyte	IL memory: 8 Kbytes, OL memory: 16 Kbytes, and SHwyRAM: 256 Kbytes	PRBG0176GA-A	Yes
SH74562	R5F74562KBG				No

1.3 Block Diagram

Figure 1.1 shows the block diagram.



Legend:

CPU:	Central processing unit
FPU:	Floating point unit
MMU:	Memory management unit
UTLB:	Shared TLB
ITLB:	Instruction TLB
H-UDI:	User debugging interface
UBC:	User break controller
DMAC:	Direct memory access controller
DRI:	Direct RAM input interface
DRO:	Direct RAM output interface
FCU:	ROM write sequencer
CAN:	Controller area network
TMU:	Timer unit
ATU-IIIS:	Advanced timer unit - IIIS
PDAC:	Parallel DAC controller
PSEL:	Parallel selector
ADC:	A/D converter
RSPI:	Renesas serial peripheral interface
SCIF:	Serial communication interface with FIFO
IIC3:	I²C bus interface 3
WDT:	Watchdog timer
CPG:	Clock generator
INTC:	Interrupt controller

Figure 1.1 Block Diagram

1.4 Pin Arrangement

Figure 1.2 shows the pin arrangement.

Position of pin A1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
A	Vss (N.C.)	PG0/MOSI0/TO40	PF5/SCL/(CTX3)	PF1/CTX0	DET3OR5	Vcc	PL8/TIA14/IRQ7/DREQ3	PL6/TIA12/(TIF1A)	PH15/DROD7/TO37/DDC15	PH13/DROD5/(TO35)/DDC13	PH9/DROD1/(TO31)/DDC09/CTS2#	PH5/DROD13/TO25/DDC05/TIA01	PH2/DROD10/TO22/DDC02/TIF1A	PH0/DROD8/TO20/DDC00/TIF0A	Vss (N.C.)	A
B	PG1/MISO0/TO41	PG2/RSPCK0/TO42	PG3/TO43/SSL00/(IRQ7)	PF4/SDA/(CRX3)	PF0/CRX0	ASEBRK#/BRKACK	PL9/TIA15/AUDREVT#	PL5/TIA11/(TIF0B)	PL2/DROWR	PH12/DROD4/TO34/DDC12	PH8/DROD0/(TO30)/DDC08/RTS2#	PH4/DROD12/TO24/DDC04/TIA00	PH1/DROD9/TO21/DDC01/TIF0B	PH3/DROD11/TO23/DDC03/TIF1B	PK14/AUDRSYN#	B
C	PG4/IRQ2/TO44/SSL01	Vss	WDTOVF#	Vdd	Vdd	Vdd	PL4/TIA10/(TIF0A)	Vss	Vcc	PH14/DROD6/(TO36)/DDC14/IRQ1	PH10/DROD2/(TO32)/DDC10	PH6/DROD14/TO26/DDC06/TIA02	PK12/AUDRD3	PK13/AUDRCLK	PK11/AUDRD2	C
D	FWE	RESET#	Vss	Vss	Vdd	Vdd	PL3/IRQ6	Vss	Vcc	PH11/DROD3/(TO33)/DDC11	PH7/DROD15/(TO27)/DDC07/TIA03	PK8/DREQ2	PK9/AUDRD0/RTS3#	PK10/AUDRD1/CTS3#	PK6/TXD3	D
E	MD1	NMI	Vss	Vss								Vss	PK0/IRQ5/SSL10	PK5/DINC4/RXD3	PJ14/TXD1/MOSI1	E
F	XTAL	EXTAL	Vss	Vss								Vcc	PJ10/RXD0/PWMOFF4/AD0TRG#	PJ15/SCK1/PSPCK1	PJ13/RXD1/MISO1	F
G	PLLVss	PLLVcc	MD0	MPMD								PJ1/(CTX0)/FTXA	PJ7/CTX3/TIF2B/TXD2	PJ12/SCK0/TCLKB/(IRQ0)	PJ11/TXD0/AD0END	G
H	TCK	TMS	MD2	TRST#								PJ0/(CRX0)/FRXA	PJ4/CRX2/FTXENA/CTS0#	PJ6/CRX3/TIF2A/RXD2/TIA04	PJ5/CTX2/FTXENB/SCK2	H
J	PD1/PDIDATA1	TDO	TDI	Vss								PN1/AD1IN1	PN0/AD1IN0	PJ3/CTX1/FTXB/RTS0#	PJ2/CRX1/FRXB	J
K	PD4/PDIDATA4	PD3/PDIDATA3	Vss	Vss								PN4/AD1IN4	PN5/AD1IN5	AVss	AVcc	K
L	PD8/PDIDATA8	PD7/PDIDATA7	Vcc	Vcc								PM0/AD0IN0	AVss	AVREFL	AVREFH	L
M	PD9/PDIDATA9	PD6/PDIDATA6	PD0/PDIDATA0	Vss	Vss	Vss	Vdd	Vdd	PC6/CLKOUT/TO36	Vcc	Vss	AVss	PM4/AD0IN4	AVREFL	AVREFH	M
N	PD10/PDIWR	PD5/PDIDATA5	PA4/TO04/DDB04	PA7/TO07/DDB07	PA10/TO12/DDB10/PSLDATA0	PA11/TO13/DDB11/PSLDATA1	Vdd	Vdd	PC1/TO31/MISO2	Vcc	Vss	PM2/AD0IN2	PM6/AD0IN6	PM9/AD0IN9	AVss	N
P	PD2/PDIDATA2	PA3/TO03/DDB03	PA0/TO00/DDB00	PA2/TO02/DDB02	PA6/TO06/DDB06	PA9/TO11/DDB09/PSLCLKA	PA13/TO15/DDB13/PSLDATA3	PB1/PWMOFF1/DINB1	PC0/TO30/MOSI2/(IRQ6)	PC3/TO33/SSL20/IRQ0	PM15/AD0IN15	PM13/AD0IN13	PM11/AD0IN11	PM8/AD0IN8	AVcc	P
R	Vss (N.C.)	PE15/TO27/PSLCLR	PA1/TO01/DDB01	PA5/TO05/DDB05	PA8/TO10/DDB08/PSLCLKB	PA12/TO14/DDB12/PSLDATA2	PB0/PWMOFF0/DINB0	PB3/PWMOFF3/DINB3	PC2/TO32/RSPCK2/DREQ0	PC5/TO35	PC14	PM14/AD0IN14	PM12/AD0IN12	PM10/AD0IN10	AVcc (N.C.)	R
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	

Figure 1.2 Pin Arrangement (Top Transparent View)

1.5 Pin Functions

Table 1.3 lists the pin functions.

Table 1.3 Pin Functions

		User Pin									Pin State after a Reset			
Pin No.	Pin Name	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Power Supply Name in Circuit	Input Buffer Type	Switching Output Driving Ability	Function	I/O	Pin State*1	Input Enabled/ Disabled
G4	MPMD							Vcc	Schmitt	—	MPMD	Input	Hiz (pull-up)	Enabled
H3	MD2							Vcc	Schmitt	—	MD2	Input	Hiz (pull-down)	Enabled
E1	MD1							Vcc	Schmitt	—	MD1	Input	Hiz (pull-down)	Enabled
G3	MD0							Vcc	Schmitt	—	MD0	Input	Hiz (pull-down)	Enabled
D1	FWE							Vcc	Schmitt	—	FWE	Input	Hiz (pull-down)	Enabled
D2	RESET#							Vcc	Schmitt	—	RESET#	Input	Hiz	Enabled
E2	NMI							Vcc	Schmitt	—	NMI	Input	Hiz	Enabled
F2	EXTAL							Vcc	CMOS	—	EXTAL	Input	—	Enabled
F1	XTAL							Vcc		—	XTAL	Output	XTAL	—
C3	WDTOVF#							Vcc		No	WDTOVF#	Output	WDTOVF#	—
H4	TRST#							Vcc	Schmitt	—	TRST#	Input	Hiz	Enabled
H1	TCK							Vcc	TTL	—	TCK	Input	Hiz	Enabled
H2	TMS							Vcc	Schmitt	—	TMS	Input	Hiz	Enabled
J3	TDI							Vcc	Schmitt	—	TDI	Input	Hiz	Enabled
J2	TDO							Vcc		—	TDO	Output	Hiz	—
B6	ASEBRK#/BRKACK							Vcc	Schmitt	No	ASEBRK#/ BRKACK	Input	Hiz	Enabled
A5	DET3OR5							Vcc	Schmitt	—	DET3OR5	Input	Hiz (pull-up)	Enabled
P3	PA0/TO00/DDB00	PA0		TO00	DDB00			Vcc	Threshold value switching	Yes	PA0	Input	Hiz	Disabled
R3	PA1/TO01/DDB01	PA1		TO01	DDB01			Vcc	Threshold value switching	Yes	PA1	Input	Hiz	Disabled
P4	PA2/TO02/DDB02	PA2		TO02	DDB02			Vcc	Threshold value switching	Yes	PA2	Input	Hiz	Disabled
P2	PA3/TO03/DDB03	PA3		TO03	DDB03			Vcc	Threshold value switching	Yes	PA3	Input	Hiz	Disabled
N3	PA4/TO04/DDB04	PA4		TO04	DDB04			Vcc	Threshold value switching	Yes	PA4	Input	Hiz	Disabled
R4	PA5/TO05/DDB05	PA5		TO05	DDB05			Vcc	Threshold value switching	Yes	PA5	Input	Hiz	Disabled
P5	PA6/TO06/DDB06	PA6		TO06	DDB06			Vcc	Threshold value switching	Yes	PA6	Input	Hiz	Disabled
N4	PA7/TO07/DDB07	PA7		TO07	DDB07			Vcc	Threshold value switching	Yes	PA7	Input	Hiz	Disabled
R5	PA8/TO10/DDB08/PSLCLKB	PA8		TO10	DDB08	PSLCLKB		Vcc	Threshold value switching	Yes	PA8	Input	Hiz	Disabled
P6	PA9/TO11/DDB09/PSLCLKA	PA9		TO11	DDB09	PSLCLKA		Vcc	Threshold value switching	Yes	PA9	Input	Hiz	Disabled

		User Pin									Pin State after a Reset			
Pin No.	Pin Name	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Power Supply Name in Circuit	Input Buffer Type	Switching Output Driving Ability	Function	I/O	Pin State*1	Input Enabled/ Disabled
N5	PA10/TO12/DDB10/PSLDATA0	PA10		TO12	DDB10	PSLDATA0		Vcc	Threshold value switching	Yes	PA10	Input	Hiz	Disabled
N6	PA11/TO13/DDB11/PSLDATA1	PA11		TO13	DDB11	PSLDATA1		Vcc	Threshold value switching	Yes	PA11	Input	Hiz	Disabled
R6	PA12/TO14/DDB12/PSLDATA2	PA12		TO14	DDB12	PSLDATA2		Vcc	Threshold value switching	Yes	PA12	Input	Hiz	Disabled
P7	PA13/TO15/DDB13/PSLDATA3	PA13		TO15	DDB13	PSLDATA3		Vcc	Threshold value switching	Yes	PA13	Input	Hiz	Disabled
R7	PB0/PWMOFF0/DINB0	PB0		PWMOFF0	DINB0			Vcc	Threshold value switching	Yes	PB0	Input	Hiz	Disabled
P8	PB1/PWMOFF1/DINB1	PB1		PWMOFF1	DINB1			Vcc	Threshold value switching	Yes	PB1	Input	Hiz	Disabled
R8	PB3/PWMOFF3/DINB3	PB3		PWMOFF3	DINB3			Vcc	Threshold value switching	Yes	PB3	Input	Hiz	Disabled
P9	PC0/TO30/MOSI2/(IRQ6)	PC0		TO30		MOSI2	(IRQ6)	Vcc	Threshold value switching	Yes	PC0	Input	Hiz	Disabled
N9	PC1/TO31/MISO2	PC1		TO31		MISO2		Vcc	Threshold value switching	Yes	PC1	Input	Hiz	Disabled
R9	PC2/TO32/RSPCK2/DREQ0	PC2		TO32		RSPCK2	DREQ0	Vcc	Threshold value switching	Yes	PC2	Input	Hiz	Disabled
P10	PC3/TO33/SSL20/IRQ0	PC3		TO33		SSL20	IRQ0	Vcc	Threshold value switching	Yes	PC3	Input	Hiz	Disabled
R10	PC5/TO35	PC5		TO35				Vcc	Threshold value switching	Yes	PC5	Input	Hiz	Disabled
M9	PC6/CLKOUT/TO36	PC6	CLKOUT	TO36				Vcc	Threshold value switching	Yes	PC6	Input	Hiz	Disabled
R11	PC14	PC14						Vcc	Threshold value switching	Yes	PC14	Input	Hiz	Disabled
M3	PD0/PDIDATA0	PD0		PDIDATA0				Vcc	Threshold value switching	Yes	PD0	Input	Hiz	Disabled
J1	PD1/PDIDATA1	PD1		PDIDATA1				Vcc	Threshold value switching	Yes	PD1	Input	Hiz	Disabled
P1	PD2/PDIDATA2	PD2		PDIDATA2				Vcc	Threshold value switching	Yes	PD2	Input	Hiz	Disabled
K2	PD3/PDIDATA3	PD3		PDIDATA3				Vcc	Threshold value switching	Yes	PD3	Input	Hiz	Disabled
K1	PD4/PDIDATA4	PD4		PDIDATA4				Vcc	Threshold value switching	Yes	PD4	Input	Hiz	Disabled
N2	PD5/PDIDATA5	PD5		PDIDATA5				Vcc	Threshold value switching	Yes	PD5	Input	Hiz	Disabled
M2	PD6/PDIDATA6	PD6		PDIDATA6				Vcc	Threshold value switching	Yes	PD6	Input	Hiz	Disabled
L2	PD7/PDIDATA7	PD7		PDIDATA7				Vcc	Threshold value switching	Yes	PD7	Input	Hiz	Disabled
L1	PD8/PDIDATA8	PD8		PDIDATA8				Vcc	Threshold value switching	Yes	PD8	Input	Hiz	Disabled

		User Pin								Switching	Pin State after a Reset			
								Power Supply		Output				Input
Pin								Name in		Driving				Enabled/
No.	Pin Name	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Circuit	Input Buffer Type	Ability	Function	I/O	Pin State*1	Disabled
M1	PD9/PDIDATA9	PD9		PDIDATA9				Vcc	Threshold value switching	Yes	PD9	Input	Hiz	Disabled
N1	PD10/PDIWR	PD10		PDIWR				Vcc	Threshold value switching	Yes	PD10	Input	Hiz	Disabled
R2	PE15/TO27/PSLCLR	PE15		TO27		PSLCLR		Vcc	Threshold value switching	Yes	PE15	Input	Hiz	Disabled
B5	PF0/CRX0	PF0	CRX0					Vcc	Threshold value switching	Yes	PF0	Input	Hiz	Disabled
A4	PF1/CTX0	PF1	CTX0					Vcc	Threshold value switching	Yes	PF1	Input	Hiz	Disabled
B4	PF4/SDA/(CRX3)	PF4	SDA				(CRX3)	Vcc	Threshold value switching	Yes	PF4	Input	Hiz	Disabled
A3	PF5/SCL/(CTX3)	PF5	SCL				(CTX3)	Vcc	Threshold value switching	Yes	PF5	Input	Hiz	Disabled
A2	PG0/MOSI0/TO40	PG0	MOSI0	TO40				Vcc	CMOS	Yes	PG0	Input	Hiz	Enabled
B1	PG1/MISO0/TO41	PG1	MISO0	TO41				Vcc	CMOS	Yes	PG1	Input	Hiz	Enabled
B2	PG2/RSPCK0/TO42	PG2	RSPCK0	TO42				Vcc	CMOS	Yes	PG2	Input	Hiz	Enabled
B3	PG3/TO43/SSL00/(IRQ7)	PG3		TO43	SSL00		(IRQ7)	Vcc	CMOS	Yes	PG3	Input	Hiz	Enabled
C1	PG4/IRQ2/TO44/SSL01	PG4	IRQ2	TO44	SSL01			Vcc	Threshold value switching	Yes	PG4	Input	Hiz	Disabled
A14	PH0/DROD8/TO20/DDC00/TIF0A	PH0	DROD8	TO20	DDC00	TIF0A		Vcc	Threshold value switching	Yes	PH0	Input	Hiz	Disabled
B13	PH1/DROD9/TO21/DDC01/TIF0B	PH1	DROD9	TO21	DDC01	TIF0B		Vcc	Threshold value switching	Yes	PH1	Input	Hiz	Disabled
A13	PH2/DROD10/TO22/DDC02/TIF1A	PH2	DROD10	TO22	DDC02	TIF1A		Vcc	Threshold value switching	Yes	PH2	Input	Hiz	Disabled
B14	PH3/DROD11/TO23/DDC03/TIF1B	PH3	DROD11	TO23	DDC03	TIF1B		Vcc	Threshold value switching	Yes	PH3	Input	Hiz	Disabled
B12	PH4/DROD12/TO24/DDC04/TIA00	PH4	DROD12	TO24	DDC04	TIA00		Vcc	Threshold value switching	Yes	PH4	Input	Hiz	Disabled
A12	PH5/DROD13/TO25/DDC05/TIA01	PH5	DROD13	TO25	DDC05	TIA01		Vcc	Threshold value switching	Yes	PH5	Input	Hiz	Disabled
C12	PH6/DROD14/TO26/DDC06/TIA02	PH6	DROD14	TO26	DDC06	TIA02		Vcc	Threshold value switching	Yes	PH6	Input	Hiz	Disabled
D11	PH7/DROD15/(TO27)/DDC07/TIA03	PH7	DROD15	(TO27)	DDC07	TIA03		Vcc	Threshold value switching	Yes	PH7	Input	Hiz	Disabled
B11	PH8/DROD0/(TO30)/DDC08/RTS2#	PH8	DROD0	(TO30)	DDC08	RTS2#		Vcc	Threshold value switching	Yes	PH8	Input	Hiz	Disabled
A11	PH9/DROD1/(TO31)/DDC09/CTS2#	PH9	DROD1	(TO31)	DDC09	CTS2#		Vcc	Threshold value switching	Yes	PH9	Input	Hiz	Disabled
C11	PH10/DROD2/(TO32)/DDC10	PH10	DROD2	(TO32)	DDC10			Vcc	Threshold value switching	Yes	PH10	Input	Hiz	Disabled
D11	PH11/DROD3/(TO33)/DDC11	PH11	DROD3	(TO33)	DDC11			Vcc	Threshold value switching	Yes	PH11	Input	Hiz	Disabled
B10	PH12/DROD4/TO34/DDC12	PH12	DROD4	TO34	DDC12			Vcc	Threshold value switching	Yes	PH12	Input	Hiz	Disabled

		User Pin								Switching	Pin State after a Reset			
Pin No.	Pin Name	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Power Supply Name in Circuit	Input Buffer Type	Output Driving Ability	Function	I/O	Pin State*1	Input Enabled/ Disabled
A10	PH13/DROD5/(TO35)/DDC13	PH13	DROD5	(TO35)	DDC13			Vcc	Threshold value switching	Yes	PH13	Input	Hiz	Disabled
C10	PH14/DROD6/(TO36)/DDC14/IRQ1	PH14	DROD6	(TO36)	DDC14		IRQ1	Vcc	Threshold value switching	Yes	PH14	Input	Hiz	Disabled
A9	PH15/DROD7/TO37/DDC15	PH15	DROD7	TO37	DDC15			Vcc	Threshold value switching	Yes	PH15	Input	Hiz	Disabled
H12	PJ0/(CRX0)/FRXA	PJ0	(CRX0)	FRXA				Vcc	Threshold value switching	Yes	PJ0	Input	Hiz	Disabled
G12	PJ1/(CTX0)/FTXA	PJ1	(CTX0)	FTXA				Vcc	CMOS	Yes	PJ1	Input	Hiz	Enabled
J15	PJ2/CRX1/FRXB	PJ2	CRX1	FRXB				Vcc	Threshold value switching	Yes	PJ2	Input	Hiz	Disabled
J14	PJ3/CTX1/FTXB/RTS0#	PJ3	CTX1	FTXB		RTS0#		Vcc	CMOS	Yes	PJ3	Input	Hiz	Enabled
H13	PJ4/CRX2/FTXENA/CTS0#	PJ4	CRX2	FTXENA		CTS0#		Vcc	CMOS	Yes	PJ4	Input	Hiz	Enabled
H15	PJ5/CTX2/FTXENB/SCK2	PJ5	CTX2	FTXENB		SCK2		Vcc	CMOS	Yes	PJ5	Input	Hiz	Enabled
H14	PJ6/CRX3/TIF2A/RXD2/TIA04	PJ6	CRX3	TIF2A		RXD2	TIA04	Vcc	Threshold value switching	Yes	PJ6	Input	Hiz	Disabled
G13	PJ7/CTX3/TIF2B/TXD2	PJ7	CTX3	TIF2B		TXD2		Vcc	Threshold value switching	Yes	PJ7	Input	Hiz	Disabled
F13	PJ10/RXD0/PWMOFF4/AD0TRG#	PJ10	RXD0	PWMOFF4		AD0TRG#		Vcc	Threshold value switching	Yes	PJ10	Input	Hiz	Disabled
G15	PJ11/TXD0/AD0END	PJ11	TXD0			AD0END		Vcc	Threshold value switching	Yes	PJ11	Input	Hiz	Disabled
G14	PJ12/SCK0/TCLKB/(IRQ0)	PJ12	SCK0	TCLKB			(IRQ0)	Vcc	Threshold value switching	Yes	PJ12	Input	Hiz	Disabled
F15	PJ13/RXD1/MISO1	PJ13	RXD1	MISO1				Vcc	Threshold value switching	Yes	PJ13	Input	Hiz	Disabled
E15	PJ14/TXD1/MOSI1	PJ14	TXD1	MOSI1				Vcc	Threshold value switching	Yes	PJ14	Input	Hiz	Disabled
F14	PJ15/SCK1/RSPCK1	PJ15	SCK1	RSPCK1				Vcc	Threshold value switching	Yes	PJ15	Input	Hiz	Disabled
E13	PK0/IRQ5/SSL10	PK0	IRQ5	SSL10				Vcc	Threshold value switching	Yes	PK0	Input	Hiz	Enabled
E14	PK5/DINC4/RXD3	PK5			DINC4	RXD3		Vcc	Threshold value switching	Yes	PK5	Input	Hiz	Enabled
D15	PK6/TXD3	PK6				TXD3		Vcc	Threshold value switching	Yes	PK6	Input	Hiz	Enabled
D12	PK8/DREQ2	PK8	DREQ2					Vcc	Threshold value switching	Yes	PK8	Input	Hiz	Enabled
D13	PK9/AUDRD0/RTS3#	PK9	AUDRD0			RTS3#		Vcc	Threshold value switching	Yes	PK9	Input	Hiz	Enabled
D14	PK10/AUDRD1/CTS3#	PK10	AUDRD1			CTS3#		Vcc	Threshold value switching	Yes	PK10	Input	Hiz	Enabled
C15	PK11/AUDRD2	PK11	AUDRD2					Vcc	Threshold value switching	Yes	PK11	Input	Hiz	Enabled
C13	PK12/AUDRD3	PK12	AUDRD3					Vcc	Threshold value switching	Yes	PK12	Input	Hiz	Enabled

		User Pin								Switching	Pin State after a Reset			
Pin No.	Pin Name	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Power Supply Name in Circuit	Input Buffer Type	Output Driving Ability	Function	I/O	Pin State*1	Input Enabled/ Disabled
C14	PK13/AUDRCLK	PK13	AUDRCLK					Vcc	Threshold value switching	Yes	PK13	Input	Hiz	Enabled
B15	PK14/AUDRSYN#	PK14	AUDRSYN#					Vcc	Threshold value switching	Yes	PK14	Input	Hiz	Enabled
B9	PL2/DROWR	PL2	DROWR					Vcc	Threshold value switching	Yes	PL2	Input	Hiz	Disabled
D7	PL3/IRQ6	PL3		IRQ6				Vcc	Threshold value switching	Yes	PL3	Input	Hiz	Disabled
C7	PL4/TIA10/(TIF0A)	PL4	TIA10	(TIF0A)				Vcc	Threshold value switching	Yes	PL4	Input	Hiz	Disabled
B8	PL5/TIA11/(TIF0B)	PL5	TIA11	(TIF0B)				Vcc	Threshold value switching	Yes	PL5	Input	Hiz	Disabled
A8	PL6/TIA12/(TIF1A)	PL6	TIA12	(TIF1A)				Vcc	Threshold value switching	Yes	PL6	Input	Hiz	Disabled
A7	PL8/TIA14/IRQ7/DREQ3	PL8	TIA14	IRQ7		DREQ3		Vcc	Threshold value switching	Yes	PL8	Input	Hiz	Disabled
B7	PL9/TIA15/AUDREVT#	PL9	TIA15				AUDREVT#	Vcc	Threshold value switching	Yes	PL9	Input	Hiz	Disabled
L12	PM0/AD0IN0	PM0	AD0IN0					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN0	Input	Hiz	Enabled
N12	PM2/AD0IN2	PM2	AD0IN2					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN2	Input	Hiz	Enabled
M13	PM4/AD0IN4	PM4	AD0IN4					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN4	Input	Hiz	Enabled
N13	PM6/AD0IN6	PM6	AD0IN6					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN6	Input	Hiz	Enabled
P14	PM8/AD0IN8	PM8	AD0IN8					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN8	Input	Hiz	Enabled
N14	PM9/AD0IN9	PM9	AD0IN9					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN9	Input	Hiz	Enabled
R14	PM10/AD0IN10	PM10	AD0IN10					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN10	Input	Hiz	Enabled
P13	PM11/AD0IN11	PM11	AD0IN11					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN11	Input	Hiz	Enabled
R13	PM12/AD0IN12	PM12	AD0IN12					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN12	Input	Hiz	Enabled
P12	PM13/AD0IN13	PM13	AD0IN13					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN13	Input	Hiz	Enabled
R12	PM14/AD0IN14	PM14	AD0IN14					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN14	Input	Hiz	Enabled
P11	PM15/AD0IN15	PM15	AD0IN15					AVcc	CMOS (function 1) Analog (function 2)	—	AD0IN15	Input	Hiz	Enabled
J13	PN0/AD1IN0	PN0	AD1IN0					AVcc	CMOS (function 1) Analog (function 2)	—	AD1IN0	Input	Hiz	Enabled
J12	PN1/AD1IN1	PN1	AD1IN1					AVcc	CMOS (function 1) Analog (function 2)	—	AD1IN1	Input	Hiz	Enabled

		User Pin									Pin State after a Reset			
Pin No.	Pin Name	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Power Supply Name in Circuit	Input Buffer Type	Switching Output Driving Ability	Function	I/O	Pin State*1	Input Enabled/ Disabled
K12	PN4/AD1IN4	PN4	AD1IN4					AVcc	CMOS (function 1) Analog (function 2)	—	AD1IN4	Input	Hiz	Enabled
K13	PN5/AD1IN5	PN5	AD1IN5					AVcc	CMOS (function 1) Analog (function 2)	—	AD1IN5	Input	Hiz	Enabled
P15	AVcc									—	AVcc	—	—	—
N15	AVss									—	AVss	—	—	—
K15	AVcc									—	AVcc	—	—	—
K14	AVss									—	AVss	—	—	—
M15	AVREFH									—	AVREFH	—	—	—
M14	AVREFL									—	AVREFL	—	—	—
L15	AVREFH									—	AVREFH	—	—	—
L14	AVREFL									—	AVREFL	—	—	—
G2	PLLVcc									—	PLLVcc	—	—	—
G1	PLLVss									—	PLLVss	—	—	—
C4	Vdd									—	Vdd	—	—	—
C5	Vdd									—	Vdd	—	—	—
C6	Vdd									—	Vdd	—	—	—
D5	Vdd									—	Vdd	—	—	—
D6	Vdd									—	Vdd	—	—	—
M7	Vdd									—	Vdd	—	—	—
M8	Vdd									—	Vdd	—	—	—
N7	Vdd									—	Vdd	—	—	—
N8	Vdd									—	Vdd	—	—	—
J4	Vss									—	Vss	—	—	—
F4	Vss									—	Vss	—	—	—
C2	Vss									—	Vss	—	—	—
D3	Vss									—	Vss	—	—	—
D4	Vss									—	Vss	—	—	—
E3	Vss									—	Vss	—	—	—
E4	Vss									—	Vss	—	—	—
F3	Vss									—	Vss	—	—	—
M4	Vss									—	Vss	—	—	—
M5	Vss									—	Vss	—	—	—
M6	Vss									—	Vss	—	—	—
A6	Vcc									—	Vcc	—	—	—
L3	Vcc									—	Vcc	—	—	—
L4	Vcc									—	Vcc	—	—	—
M10	Vcc									—	Vcc	—	—	—
N10	Vcc									—	Vcc	—	—	—
K3	Vss									—	Vss	—	—	—

		User Pin									Pin State after a Reset			
Pin No.	Pin Name	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Power Supply Name in Circuit	Input Buffer Type	Switching Output Driving Ability	Function	I/O	Pin State*1	Input Enabled/ Disabled
K4	Vss									—	Vss	—	—	—
M11	Vss									—	Vss	—	—	—
N11	Vss									—	Vss	—	—	—
C9	Vcc									—	Vcc	—	—	—
D9	Vcc									—	Vcc	—	—	—
F12	Vcc									—	Vcc	—	—	—
C8	Vss									—	Vss	—	—	—
D8	Vss									—	Vss	—	—	—
E12	Vss									—	Vss	—	—	—
A1	Vss(N.C.)									—	Vss	—	—	—
A15	Vss(N.C.)									—	Vss	—	—	—
R1	Vss(N.C.)									—	Vss	—	—	—
R15	AVcc(N.C.)									—	AVcc	—	—	—
L13	AVss									—	AVss	—	—	—
M12	AVss									—	AVss	—	—	—

Notes: *1 The meanings of the notations "pull up" and "pull down" are as follows. See section 38, Electrical Characteristics, for the MOS pull-up and pull-down transistor current drains.

Pull up: A pull-up function is provided for the pin.

Pull down: A pull-down function is provided for the pin.

- There is no FlexRay module (FRXA, FTXA, FRXB, FTXB, FTXENA, and FTXENB) in the SH7456 Group.
- All Vss and Vss (N.C.) pins are connected.
- AVcc (N.C.) [the R15 pin] and AVcc [the P15 pin] are connected to AVcc of the AD0 module.
- AVss [the N15 pin] and AVss [the M12 and L13 pins] are connected to AVss of the AD0 module.
- AVREFH [the M15 pin] is connected to AVREFH of the AD0 module.
- AVREFH [the L15 pin] is connected to AVREFH of the AD1 module.
- AVREFL [the M14 pin] is connected to AVREFL of the AD0 module.
- AVREFL [the L14 pin] is connected to AVREFL of the AD1 module.
- Although the power supply pins marked "(N.C.)" may be left open without affecting microcontroller operation, we recommend connecting them for power supply stabilization.
- When the same pin function is assigned to two pin locations, descriptions are separately given to the pin location not represented in parentheses and the one represented in parentheses. Here, the pin function of the two pin locations is identical. For details on the pin function assignment, refer to table 18.15.

1.6 Descriptions of Pin Functions

Table 1.4 lists the descriptions of pin functions.

Table 1.4 Descriptions of Pin Functions

Classification	Pins Name	I/O	Functions
Power supply	Vcc	I	System and I/O port power supply. All these pins must be connected.
	Vss	I	Ground pins. All these pins must be connected to ground (GND).
	Vdd	I	IC internal logic circuit power supply. All these pins must be connected.
	DET3OR5	I	Vcc voltage level specification pin.
	PLLVcc	I	PLL frequency multiplier power supply
	PLLVss	I	PLL frequency multiplier ground
Clock	EXTAL	I	Crystal resonator or external clock input
	XTAL	O	Crystal resonator connection
	CLKOUT	O	System clock output pin
System control	MD0 to MD2	I	Operating mode setting pins. Do not change the signal levels on these pins during operation.
	FWE	I	This pin enables or disables ROM programming.
	MPMD	I	Operating mode setting pins. Do not change the signal levels on these pins during operation.
	RESET#	I	Reset input pin. This MCU goes to the hardware reset state when the RESET# pin is set "L" level.
Interrupt	NMI	I	Nonmaskable interrupt request signal pin.
	IRQ0 to IRQ2, IRQ5 to IRQ7	I	External interrupt request signal input pin.
Watchdog timer (WDT)	WDTOVF#	O	Counter overflow output pin.
I/O ports	PA0 to PA13	I/O	14-pin general input/output ports.
	PB0, PB1, PB3	I/O	3-pin general input/output ports.
	PC0 to PC3, PC5, PC6, PC14	I/O	7-pin general input/output ports.
	PD0 to PD10	I/O	11-pin general input/output ports.
	PE15	I/O	1-pin general input/output ports.
	PF0, PF1, PF4, PF5	I/O	4-pin general input/output ports.
	PG0 to PG4	I/O	5-pin general input/output ports.
	PH0 to PH15	I/O	16-pin general input/output ports.
	PJ0 to PJ7, PJ10 to PJ15	I/O	14-pin general input/output ports.
	PK0, PK5, PK6, PK8 to PK14	I/O	10-pin general input/output ports.
	PL2 to PL6, PL8, PL9	I/O	7-pin general input/output ports.

Classification	Pins Name	I/O	Functions
I/O ports	PM0, PM2, PM4, PM6, PM8 to PM15	I	12-pin general input ports.
	PN0, PN1, PN4, PN5	I	4-pin general input ports.
Direct memory access controller (DMAC)	DREQ0, DREQ2, DREQ3	I	DMA transfer request input pins.
Advanced timer unit IIIS (ATU-IIIS)	TCLKB	I	External clock input to clock bus 5.
	TIA00 to TIA04, TIA10 to TIA12, TIA14, TIA15	I	Input-capture triggers for timer A channels.
	TIF0A, TIF0B, TIF1A, TIF1B, TIF2A, TIF2B	I	Event inputs for timer F channels.
	PWMOFF0, PWMOFF1, PWMOFF3, PWMOFF4	I	Timer TOU PWM output-prohibit control signal inputs.
	TO00 to TO07, TO10 to TO15, TO20 to TO27, TO30 to TO37, TO40 to TO44	O	Pulse/PWM outputs for timer TOU channels.
Serial communication interface with FIFO (SCIF)	SCK0 to SCK2	I/O	Clock input/output pins.
	RXD0 to RXD3	I	Receive data input pins.
	TXD0 to TXD3	O	Transmit data output pins.
	RTS0#, RTS2#, RTS3#	I/O	Request to send.
	CTS0#, CTS2#, CTS3#	I/O	Clear to send.
Renesas serial peripheral interface (RSPI)	RSPCK0 to RSPCK2	I/O	RSPI0 to RSPI2 clock input/output pins.
	MOSI0, MOSI1, MOSI2	I/O	RSPI0 to RSPI2 master transmit data input/output pins.
	MISO0, MISO1, MISO2	I/O	RSPI0 to RSPI2 slave transmit data input/output pins.
	SSL00, SSL10, SSL20	I/O	RSPI0 to RSPI2 slave select data input/output pins.
	SSL01	O	RSPI0 slave select data output pin.
I^{2}C bus interface 3 (IIC3)	SCL	I/O	I^{2}C serial clock input/output pin.
	SDA	I/O	I^{2}C serial data input/output pin.
CAN module	CRX0 to CRX3	I	Pins for receiving data.
	CTX0 to CTX3	O	Pins for transmitting data.
A/D converter (ADC)	AVcc	I	A/D converter power supply. All these pins must be connected.
	AVss	I	Analog ground pin. All these pins must be connected.
	AVREFL	I	Input pin for analog reference voltage. All these pins must be connected.
	AVREFH	I	Input pin for analog reference voltage. All these pins must be connected.
	AD0IN0, AD0IN2, AD0IN4, AD0IN6, AD0IN8 to AD0IN15	I	Analog input pins for AD0 module.
	AD1IN0, AD1IN1, AD1IN4, AD1IN5	I	Analog input pins for AD1 module.
	AD0TRG#	I	Input pin for scan conversion trigger of A/D module.
	AD0END	O	Output pin for monitoring AD0 module conversion timing of AD0IN0.

Classification	Pins Name	I/O	Functions
Direct RAM input interface (DRI)	DDB00 to DDB13	I	Input pins of DRI input data.
	DDC00 to DDC15	I	Input pins of DRI input data.
	DINB0, DINB1, DINB3	I	Input pins of DRI event input.
	DINC4	I	Input pin of DRI event input.
Direct RAM output interface (DRO)	DROD0 to DROD15	O	DRO output data bus.
	DROWR	O	DRO output data strobe.
Parallel DAC controller (PDAC)	PDIDATA0 to PDIDATA9	O	Setting data output to the D/A converter.
	PDIWR	O	Write signal output to the D/A converter.
Parallel selector (PSEL)	PSLCLKA	O	PSEL clock A output pin.
	PSLCLKB	O	PSEL clock B output pin.
	PSLDATA0 to PSLDATA3	O	PSEL select data output pins.
	PSLCLR	O	PSEL clear pulse output pin.
FlexRay module	FRXA	I	Channel A receive data input pin.
	FTXA	O	Channel A transmit data output pin.
	FTXENA	O	Channel A transmit enable pin. Data transmission is disabled when this pin is "H" level. Data transmission is enabled when this pin is "L" level.
	FRXB	I	Channel B receive data input pin.
	FTXB	O	Channel B transmit data output pin.
	FTXENB	O	Channel B transmit enable pin. Data transmission is disabled when this pin is "H" level. Data transmission is enabled when this pin is "L" level.
AUD RAM monitor (AUDR)	AUDRCLK	I	Synchronization clock input pin. Input the clock to be used for debugging to this pin. Frequencies up to 12.5 MHz can be used.
	AUDRSYN#	I	Data start position recognition signal input.
	AUDRD0 to AUDRD3	I/O	Command, address, and data input/output pins.
	AUDREVT#	O	Event output pin.
User debugging interface (H-UDI)	TCK	I	JTAG serial clock input pin.
	TMS	I	Mode select input pin.
	TRST#	I	H-UDI reset input pin.
	TDI	I	Data input pin.
	TDO	O	Data output pin.
	ASEBRK#/BRKACK	I/O	Pins for an emulator.

This page is blank for reasons of layout.

Section 2 Programming Model

The programming model of the SH-4A is explained in this section. The SH-4A has registers and data formats as shown below.

2.1 Data Formats

The data formats supported by the SH-4A are shown in figure 2.1.

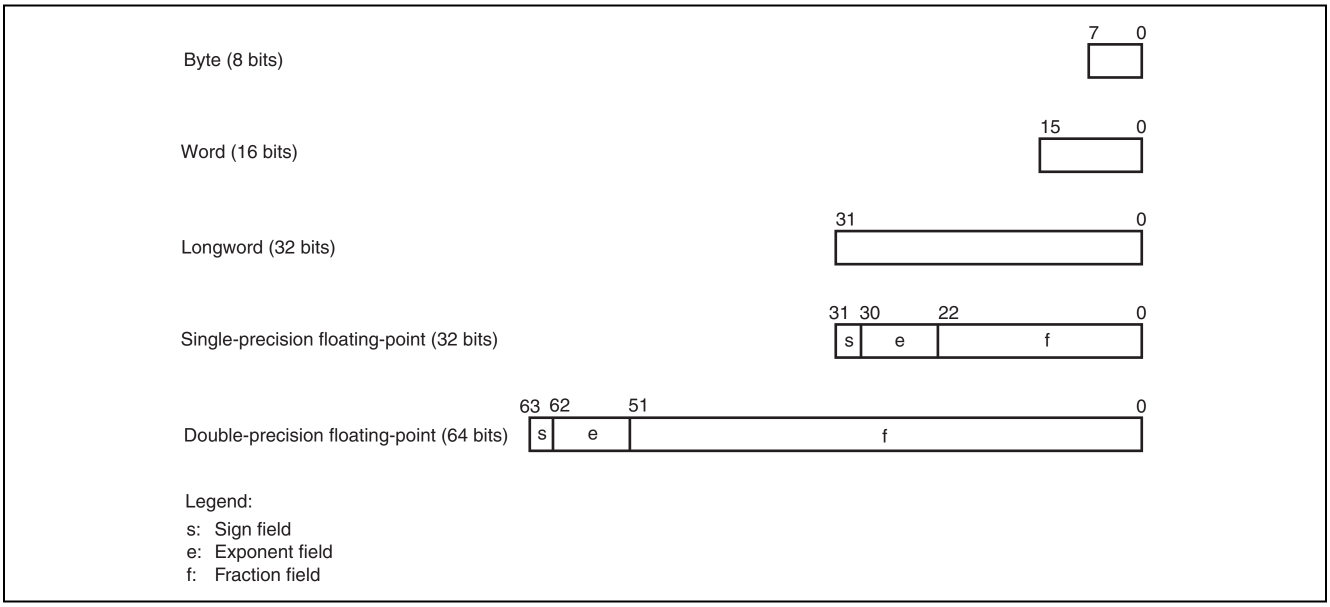


Figure 2.1 Data Formats

2.2 Register Descriptions

2.2.1 Privileged Mode and Banks

(1) Processing Modes

The SH-4A has two processing modes, user mode, and privileged mode. The SH-4A normally operates in user mode, and switches to privileged mode when an exception occurs or an interrupt is accepted. There are four kinds of registers—general registers, system registers, control registers, and floating-point registers—and the registers that can be accessed differ in the two processing modes.

(2) General Registers

There are 16 general registers, designated R0 to R15. General registers R0 to R7 are banked registers which are switched by a processing mode change.

- Privileged mode

  In privileged mode, the register bank bit (RB) in the status register (SR) defines which banked register set is accessed as general registers, and which set is accessed only through the load control register (LDC) and store control register (STC) instructions.

  When the RB bit is "1" (that is, when bank 1 is selected), the 16 registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. In this case, the eight registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 are accessed by the LDC/STC instructions.

  When the RB bit is "0" (that is, when bank 0 is selected), the 16 registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. In this case, the eight registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 are accessed by the LDC/STC instructions.
- User mode

  In user mode, the 16 registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. The eight registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 cannot be accessed.

(3) Control Registers

Control registers comprise the global base register (GBR) and status register (SR), which can be accessed in both processing modes, and the saved status register (SSR), saved program counter (SPC), vector base register (VBR), saved general register 15 (SGR), and debug base register (DBR), which can only be accessed in privileged mode. Some bits of the status register (such as the RB bit) can only be accessed in privileged mode.

(4) System Registers

System registers comprise the multiply-and-accumulate registers (MACH/MACL), the procedure register (PR), and the program counter (PC). Access to these registers does not depend on the processing mode.

(5) Floating-Point Registers and System Registers Related to FPU

There are thirty-two floating-point registers, FR0–FR15 and XF0–XF15. FR0–FR15 and XF0–XF15 can be assigned to either of two banks (FPR0_BANK0–FPR15_BANK0 or FPR0_BANK1–FPR15_BANK1).

FR0–FR15 can be used as the eight registers DR0/2/4/6/8/10/12/14 (double-precision floating-point registers, or pair registers) or the four registers FV0/4/8/12 (register vectors), while XF0–XF15 can be used as the eight registers XD0/2/4/6/8/10/12/14 (register pairs) or register matrix XMTRX.

System registers related to the FPU comprise the floating-point communication register (FPUL) and the floating-point status/control register (FPSCR). These registers are used for communication between the FPU and the CPU, and the exception handling setting.

Register values after a reset are shown in table 2.1.

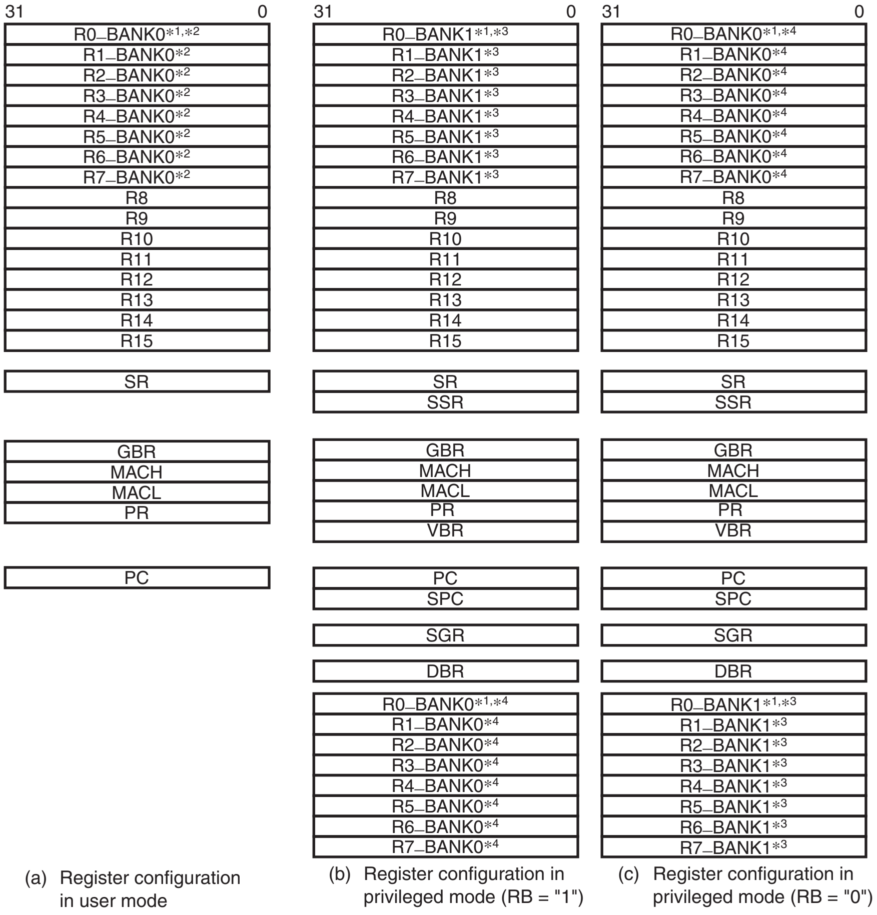
Table 2.1 Initial Register Values

Type	Registers	After Reset*[1]
General registers	R0_BANK0 to R7_BANK0, R0_BANK1 to R7_BANK1, R8 to R15	Undefined
Control registers	SR	MD bit = "1", RB bit = "1", BL bit = "1", IMASK = "B'1111", others including reserved bits = "0"
	GBR, SSR, SPC, SGR, DBR	Undefined
	VBR	H'0000 0000
System registers	MACH, MACL, PR	Undefined
	PC	H'A000 0000
Floating-point registers	FR0 to FR15, XF0 to XF15, FPUL	Undefined
	FPSCR	H'0004 0001

Note: *1 Initialized by a hardware reset.

The CPU register configuration in each processing mode is shown in figure 2.2.

User mode and privileged mode are switched by the processing mode bit (MD) in the status register.



Notes: *1 R0 is used as the index register in indexed register-indirect addressing mode and indexed GBR indirect addressing mode.

*2 Banked registers

*3 Banked registers
Accessed as general registers when the RB bit is set to "1" in SR. Accessed only by LDC/STC instructions when the RB bit is cleared to "0".

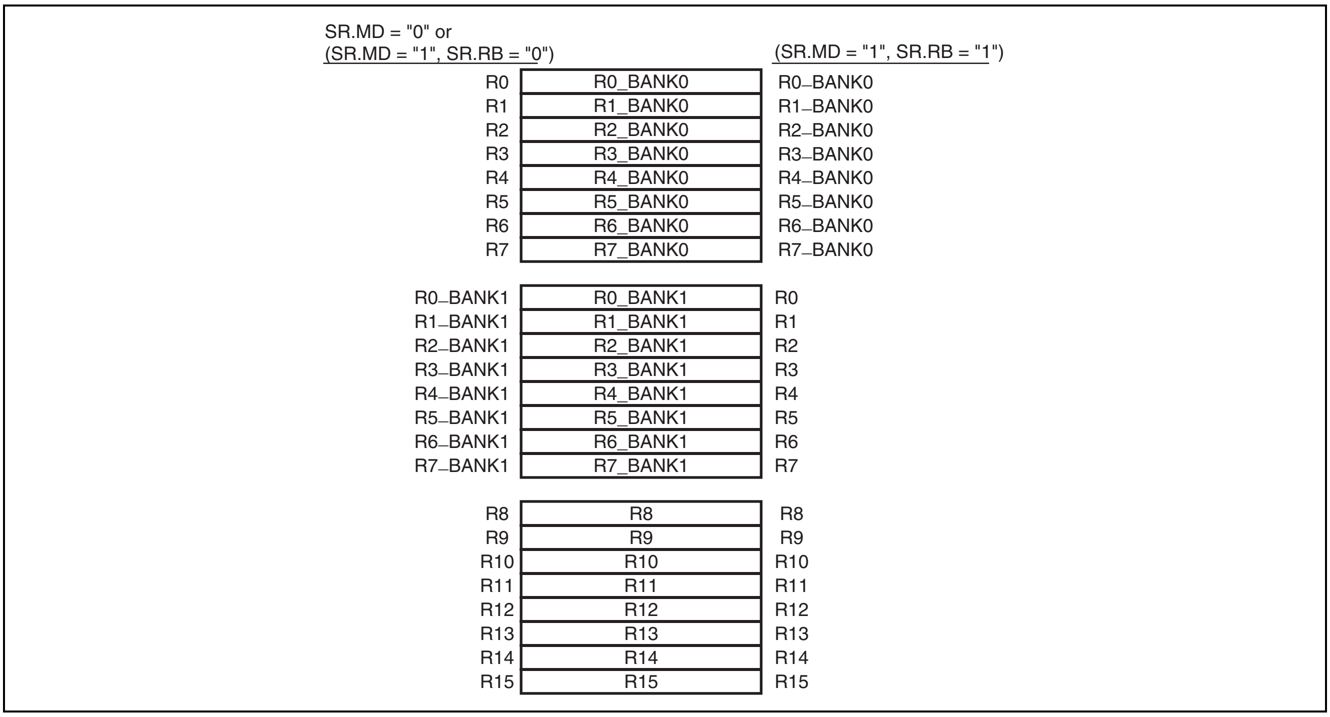
*4 Banked registers
Accessed as general registers when the RB bit is cleared to "0" in SR. Accessed only by LDC/STC instructions when the RB bit is set to "1".

Figure 2.2 CPU Register Configuration in Each Processing Mode

2.2.2 General Registers

Figure 2.3 shows the relationship between the processing modes and general registers. The SH-4A has twenty-four 32-bit general registers (R0_BANK0 to R7_BANK0, R0_BANK1 to R7_BANK1, and R8 to R15). However, only 16 of these can be accessed as general registers R0 to R15 in one processing mode. The SH-4A has two processing modes, user mode and privileged mode.

- R0_BANK0 to R7_BANK0
  Allocated to R0 to R7 in user mode (SR.MD = "0")
  Allocated to R0 to R7 when SR.RB = "0" in privileged mode (SR.MD = "1").
- R0_BANK1 to R7_BANK1
  Cannot be accessed in user mode.
  Allocated to R0 to R7 when SR.RB = "1" in privileged mode.



SR.MD = "0" or (SR.MD = "1", SR.RB = "0")		(SR.MD = "1", SR.RB = "1")
R0	R0_BANK0	R0_BANK0
R1	R1_BANK0	R1_BANK0
R2	R2_BANK0	R2_BANK0
R3	R3_BANK0	R3_BANK0
R4	R4_BANK0	R4_BANK0
R5	R5_BANK0	R5_BANK0
R6	R6_BANK0	R6_BANK0
R7	R7_BANK0	R7_BANK0
R0_BANK1	R0_BANK1	R0
R1_BANK1	R1_BANK1	R1
R2_BANK1	R2_BANK1	R2
R3_BANK1	R3_BANK1	R3
R4_BANK1	R4_BANK1	R4
R5_BANK1	R5_BANK1	R5
R6_BANK1	R6_BANK1	R6
R7_BANK1	R7_BANK1	R7
R8	R8	R8
R9	R9	R9
R10	R10	R10
R11	R11	R11
R12	R12	R12
R13	R13	R13
R14	R14	R14
R15	R15	R15

Figure 2.3 General Registers

Note on Programming: As the user's R0 to R7 are assigned to R0_BANK0 to R7_BANK0, and after an exception or interrupt R0 to R7 are assigned to R0_BANK1 to R7_BANK1, it is not necessary for the interrupt handler to save and restore the user's R0 to R7 (R0_BANK0 to R7_BANK0).

2.2.3 Floating-Point Registers

Figure 2.4 shows the floating-point register configuration. There are thirty-two 32-bit floating-point registers, FPR0_BANK0 to FPR15_BANK0, AND FPR0_BANK1 to FPR15_BANK1, comprising two banks. These registers are referenced as FR0 to FR15, DR0/2/4/6/8/10/12/14, FV0/4/8/12, XF0 to XF15, XD0/2/4/6/8/10/12/14, or XMTRX. Reference names of each register are defined depending on the state of the FR bit in FPSCR (see figure 2.4).

1. Floating-point registers, FPRn_BANKi (32 registers)
   FPR0_BANK0, FPR l_BANK0, FPR2_BANK0, FPR3_BANK0,
   FPR4_BANK0, FPR5_BANK0, FPR6_BANK0, FPR7_BANK0,
   FPR8_BANK0, FPR9_BANK0, FPR10_BANK0, FPR11_BANK0,
   FPR12_BANK0, FPR13_BANK0, FPR14_BANK0, FPR15_BANK0
   FPR0_BANK1, FPR1_BANK1, FPR2_BANK1, FPR3_BANK1,
   FPR4_BANK1, FPR5_BANK1, FPR6_BANK1, FPR7_BANK1,
   FPR8_BANK1, FPR9_BANK1, FPR10_BANK1, FPR11_BANK1,
   FPR12_BANK1, FPR13_BANK1, FPR14_BANK1, FPR15_BANK1

2. Single-precision floating-point registers, FRi (16 registers)
   When FPSCR.FR = "0", FR0 to FR15 are assigned to FPR0_BANK0 to FPR15_BANK0;
   when FPSCR.FR = "1", FR0 to FR15 are assigned to FPR0_BANK1 to FPR15_BANK1.

3. Double-precision floating-point registers or single-precision floating-point registers, DRi (8 registers): A DR register comprises two FR registers.
   DR0 = {FR0, FR1}, DR2 = {FR2, FR3}, DR4 = {FR4, FR5}, DR6 = {FR6, FR7},
   DR8 = {FR8, FR9}, DR10 = {FR10, FR11}, DR12 = {FR12, FR13}, DR14 = {FR14, FR15}

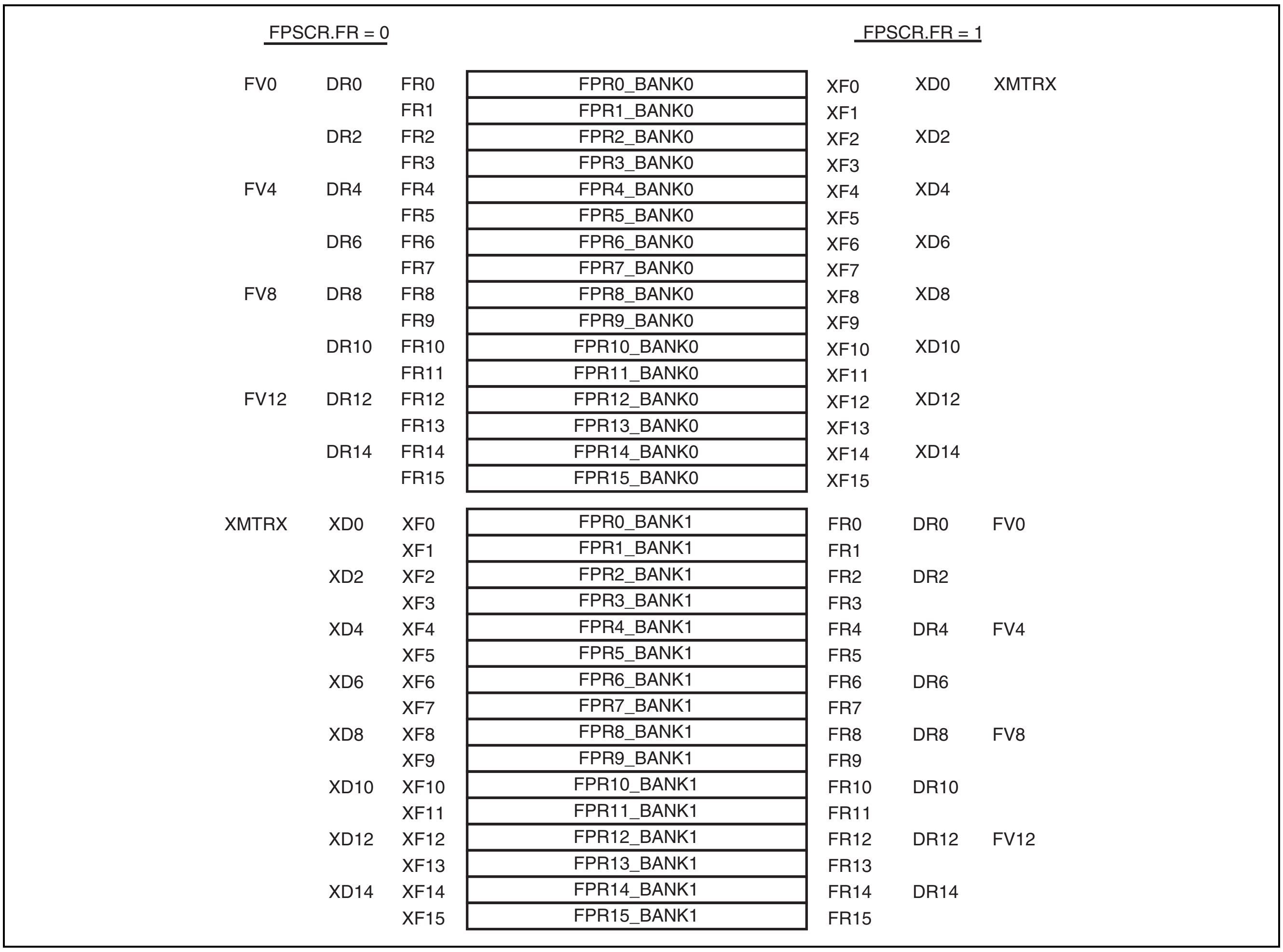
4. Single-precision floating-point vector registers, FVi (4 registers): An FV register comprises four FR registers.
   FV0 = {FR0, FR1, FR2, FR3}, FV4 = {FR4, FR5, FR6, FR7},
   FV8 = {FR8, FR9, FR10, FR11}, FV12 = {FR12, FR13, FR14, FR15}

5. Single-precision floating-point extended registers, XFi (16 registers)
   When FPSCR.FR = "0", XF0 to XF15 are assigned to FPR0_BANK1 to FPR15_BANK1;
   when FPSCR.FR = "1", XF0 to XF15 are assigned to FPR0_BANK0 to FPR15_BANK0.

6. Double-precision floating-point extended registers, XDi (8 registers): An XD register comprises two XF registers.
   XD0 = {XF0, XF1}, XD2 = {XF2, XF3}, XD4 = {XF4, XF5}, XD6 = {XF6, XF7},
   XD8 = {XF8, XF9}, XD10 = {XF10, XF11}, XD12 = {XF12, XF13}, XD14 = {XF14, XF15}

7. Single-precision floating-point extended register matrix, XMTRX: XMTRX comprises all 16 XF registers.

$$\mathrm{XMTRX} = \begin{bmatrix} \mathrm{XF0} & \mathrm{XF4} & \mathrm{XF8} & \mathrm{XF12} \\ \mathrm{XF1} & \mathrm{XF5} & \mathrm{XF9} & \mathrm{XF13} \\ \mathrm{XF2} & \mathrm{XF6} & \mathrm{XF10} & \mathrm{XF14} \\ \mathrm{XF3} & \mathrm{XF7} & \mathrm{XF11} & \mathrm{XF15} \end{bmatrix}$$



FPSCR.FR = 0				FPSCR.FR = 1		
FV0	DR0	FR0	FPR0_BANK0	XF0	XD0	XMTRX
		FR1	FPR1_BANK0	XF1		
	DR2	FR2	FPR2_BANK0	XF2	XD2	
		FR3	FPR3_BANK0	XF3		
FV4	DR4	FR4	FPR4_BANK0	XF4	XD4	
		FR5	FPR5_BANK0	XF5		
	DR6	FR6	FPR6_BANK0	XF6	XD6	
		FR7	FPR7_BANK0	XF7		
FV8	DR8	FR8	FPR8_BANK0	XF8	XD8	
		FR9	FPR9_BANK0	XF9		
	DR10	FR10	FPR10_BANK0	XF10	XD10	
		FR11	FPR11_BANK0	XF11		
FV12	DR12	FR12	FPR12_BANK0	XF12	XD12	
		FR13	FPR13_BANK0	XF13		
	DR14	FR14	FPR14_BANK0	XF14	XD14	
		FR15	FPR15_BANK0	XF15		
XMTRX	XD0	XF0	FPR0_BANK1	FR0	DR0	FV0
		XF1	FPR1_BANK1	FR1		
	XD2	XF2	FPR2_BANK1	FR2	DR2	
		XF3	FPR3_BANK1	FR3		
	XD4	XF4	FPR4_BANK1	FR4	DR4	FV4
		XF5	FPR5_BANK1	FR5		
	XD6	XF6	FPR6_BANK1	FR6	DR6	
		XF7	FPR7_BANK1	FR7		
	XD8	XF8	FPR8_BANK1	FR8	DR8	FV8
		XF9	FPR9_BANK1	FR9		
	XD10	XF10	FPR10_BANK1	FR10	DR10	
		XF11	FPR11_BANK1	FR11		
	XD12	XF12	FPR12_BANK1	FR12	DR12	FV12
		XF13	FPR13_BANK1	FR13		
	XD14	XF14	FPR14_BANK1	FR14	DR14	
		XF15	FPR15_BANK1	FR15		

Figure 2.4 Floating-Point Registers

2.2.4 Control Registers

(1) Status Register (SR)

Startus Register (SR)

BIt:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MD	RB	BL	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

BIt:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FD	—	—	—	—	—	M	Q	IMASK				—	—	S	T
After Reset:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0

<After Reset: H'7000 00F0>

Bit	Abbreviation	After Reset	R	W	Description
31	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
30	MD	1	R	W	Processing Mode Selects the processing mode. 0: User mode (Some instructions cannot be executed and some resources cannot be accessed.) 1: Privileged mode This bit is set to "1" by an exception or interrupt.
29	RB	1	R	W	Privileged Mode General Register Bank Specification Bit 0: R0_BANK0 to R7_BANK0 are accessed as general registers R0 to R7 and R0_BANK1 to R7_BANK1 can be accessed using LDC/STC instructions 1: R0_BANK1 to R7_BANK1 are accessed as general registers R0 to R7 and R0_BANK0–R7_BANK0 can be accessed using LDC/STC instructions This bit is set to "1" by an exception or interrupt.
28	BL	1	R	W	Exception/Interrupt Block Bit This bit is set to "1" by a reset, a general exception, or an interrupt. While this bit is set to "1", an interrupt request is masked. In this case, this processor enters the reset state when a general exception other than a user break occurs.
27 to 16	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
15	FD	0	R	W	FPU Disable Bit When this bit is set to "1" and an FPU instruction is not in a delay slot, a general FPU disable exception occurs. When this bit is set to "1" and an FPU instruction is in a delay slot, a slot FPU disable exception occurs. (FPU instructions: H'F*** instructions and LDS (.L)/STS(.L) instructions using FPUL/FPSCR)
14 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9	M	0	R	W	M Bit Used by the DIV0S, DIV0U, and DIV1 instructions.
8	Q	0	R	W	Q Bit Used by the DIV0S, DIV0U, and DIV1 instructions.

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	IMASK	All 1	R	W	Interrupt Mask Level Bits An interrupt whose priority is equal to or less than the value of the IMASK bits is masked. It can be chosen by CPU operation mode register (CPUOPM) whether the level of IMASK is changed to accept an interrupt or not when an interrupt is occurred. For details, see appendix A, CPU Operation Mode Register (CPUOPM).
3, 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	S	0	R	W	S Bit Used by the MAC instruction.
0	T	0	R	W	T Bit Indicates true/false condition, carry/borrow, or overflow/underflow. For details, see section 3, Instruction Set.

(2) Saved Status Register (SSR) (32 bits, Privileged Mode, After Reset = Undefined)

The contents of SR are saved to SSR in the event of an exception or interrupt.

(3) Saved Program Counter (SPC) (32 bits, Privileged Mode, After Reset = Undefined)

The address of an instruction at which an interrupt or exception occurs is saved to SPC.

(4) Global Base Register (GBR) (32 bits, After Reset = Undefined)

GBR is referenced as the base address of addressing @(disp,GBR) and @(R0,GBR).

(5) Vector Base Register (VBR) (32 bits, Privileged Mode, After Reset = H'0000 0000)

VBR is referenced as the branch destination base address in the event of an exception or interrupt. For details, see section 5, Exception Handling.

(6) Saved General Register 15 (SGR) (32 bits, Privileged Mode, After Reset = Undefined)

The contents of R15 are saved to SGR in the event of an exception or interrupt.

(7) Debug Base Register (DBR) (32 bits, Privileged Mode, After Reset = Undefined)

When the user break debugging function is enabled (CBCR.UBDE = 1), DBR is referenced as the branch destination address of the user break handler instead of VBR.

2.2.5 System Registers

(1) Multiply-and-Accumulate Registers (MACH and MACL) (32 bits, After Reset = Undefined)

MACH and MACL are used for the added value in a MAC instruction, and to store the operation result of a MAC or MUL instruction.

(2) Procedure Register (PR) (32 bits, After Reset = Undefined)

The return address is stored in PR in a subroutine call using a BSR, BSRF, or JSR instruction. PR is referenced by the subroutine return instruction (RTS).

(3) Program Counter (PC) (32 bits, After Reset = H'A000 0000)

PC indicates the address of the instruction currently being executed.

(4) Floating-Point Status/Control Register (FPSCR)

Floating-Point Status/Control Register (FPSCR)

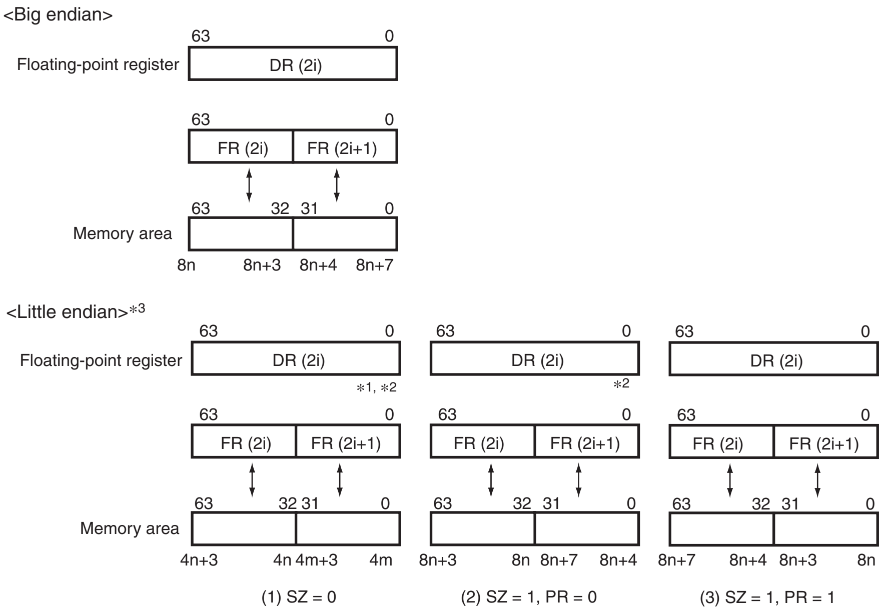
BIt:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	FR	SZ	PR	DN	Cause	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

BIt:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable (EN)					Flag					RM	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0004 0001>

Bit	Abbreviation	After Reset	R	W	Description
31 to 22	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
21	FR	0	R	W	Floating-Point Register Bank 0: FPR0_BANK0 to FPR15_BANK0 are assigned to FR0 to FR15 and FPR0_BANK1 to FPR15_BANK1 are assigned to XF0 to XF15 1: FPR0_BANK0 to FPR15_BANK0 are assigned to XF0 to XF15 and FPR0_BANK1 to FPR15_BANK1 are assigned to FR0 to FR15
20	SZ	0	R	W	Transfer Size Mode 0: Data size of FMOV instruction is 32-bits 1: Data size of FMOV instruction is a 32-bit register pair (64 bits) For relationship between the SZ bit, PR bit, and endian, see figure 2.5.
19	PR	0	R	W	Precision Mode 0: Floating-point instructions are executed as single-precision operations 1: Floating-point instructions are executed as double-precision operations (graphics support instructions are undefined) For relationship between the SZ bit, PR bit, and endian, see figure 2.5.
18	DN	1	R	W	Denormalization Mode 0: Denormalized number is treated as such 1: Denormalized number is treated as zero

Bit	Abbreviation	After Reset	R	W	Description
17 to 12	Cause	All 0	R	W	FPU Exception Cause Field
11 to 7	Enable (EN)	All 0	R	W	FPU Exception Enable Field
6 to 2	Flag	All 0	R	W	FPU Exception Flag Field Each time an FPU operation instruction is executed, the FPU exception cause field is cleared to "0". When an FPU exception occurs, the bits corresponding to FPU exception cause field and flag field are set to "1". The FPU exception flag field remains set to "1" until it is cleared to "0" by software. For bit allocations of each field, see table 2.2.
1, 0	RM	01	R	W	Rounding Mode These bits select the rounding mode. 00: Round to Nearest 01: Round to Zero 10: Reserved 11: Reserved



Notes: *1 In the case of SZ = 0 and PR = 0, DR register can not be used.
*2 The bit-location of DR register is used for double precision format when PR = 1.
(In the case of (2), it is used when PR is changed from 0 to 1.)
*3 Little endian cannot be selected in this MCU.

Figure 2.5 Relationship between SZ Bit and Endian

Table 2.2 Bit Allocation for FPU Exception Handling

Field Name		FPU Error (E)	Invalid Operation (V)	Division by Zero (Z)	Overflow (O)	Underflow (U)	Inexact (I)
Cause	FPU exception cause field	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12
Enable	FPU exception enable field	None	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7
Flag	FPU exception flag field	None	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2

(5) Floating-Point Communication Register (FPUL) (32 bits, After Reset = Undefined)

Information is transferred between the FPU and CPU via FPUL.

2.3 Memory-Mapped Registers

Some control registers are mapped to the following memory areas. Each of the mapped registers has two addresses.

H'1C00 0000 to H'1FFF FFFF
H'FC00 0000 to H'FFFF FFFF

These two areas are used as follows.

- H'1C00 0000 to H'1FFF FFFF

  This area must be accessed using the address translation function of the MMU.

  Setting the page number of this area to the corresponding field of the TLB enables access to a memory-mapped register.

  The operation of an access to this area without using the address translation function of the MMU is not guaranteed.
- H'FC00 0000 to H'FFFF FFFF

  Access to area H'FC00 0000 to H'FFFF FFFF in user mode will cause an address error. Memory-mapped registers can be referenced in user mode by means of access that involves address translation.

Note: • Do not access addresses to which registers are not mapped in either area. The operation of an access to an address with no register mapped is undefined. Also, memory-mapped registers must be accessed using a fixed data size. The operation of an access using an invalid data size is undefined.

2.4 Data Formats in Registers

Register operands are always longwords (32 bits). When a memory operand is only a byte (8 bits) or a word (16 bits), it is sign-extended into a longword when loaded into a register.

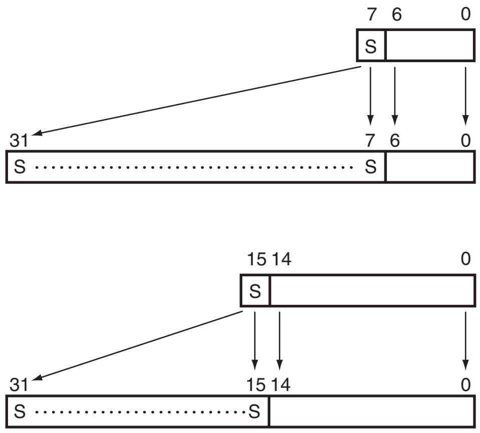


Figure 2.6 Formats of Byte Data and Word Data in Register

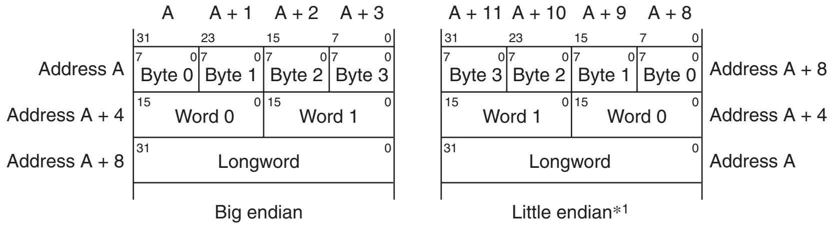
2.5 Data Formats in Memory

Memory data formats are classified into bytes, words, and longwords. Memory can be accessed in an 8-bit byte, 16-bit word, or 32-bit longword form. A memory operand less than 32 bits in length is sign-extended before being loaded into a register.

A word operand must be accessed starting from a word boundary (even address of a 2-byte unit: address 2n), and a longword operand starting from a longword boundary (even address of a 4-byte unit: address 4n). An address error will result if this rule is not observed. A byte operand can be accessed from any address.

Big endian can be selected for the data format. Bit positions are numbered left to right from most-significant to least-significant. Thus, in a 32-bit longword, the leftmost bit, bit 31, is the most significant bit and the rightmost bit, bit 0, is the least significant bit.

The data format in memory is shown in figure 2.7.



Note: *1 Little endian cannot be selected in this MCU.

Figure 2.7 Data Formats in Memory

For the 64-bit data format, see figure 2.5.

2.6 Processing States

The SH-4A has major two processing states: the reset state and instruction execution state.

(1) Reset State

In this state the CPU is reset.
The internal state of the CPU and the on-chip peripheral module registers are initialized. For details, see register descriptions for each section.

(2) Instruction Execution State

In this state, the CPU executes program instructions in sequence. The instruction execution state has the normal program execution state and the exception handling state.

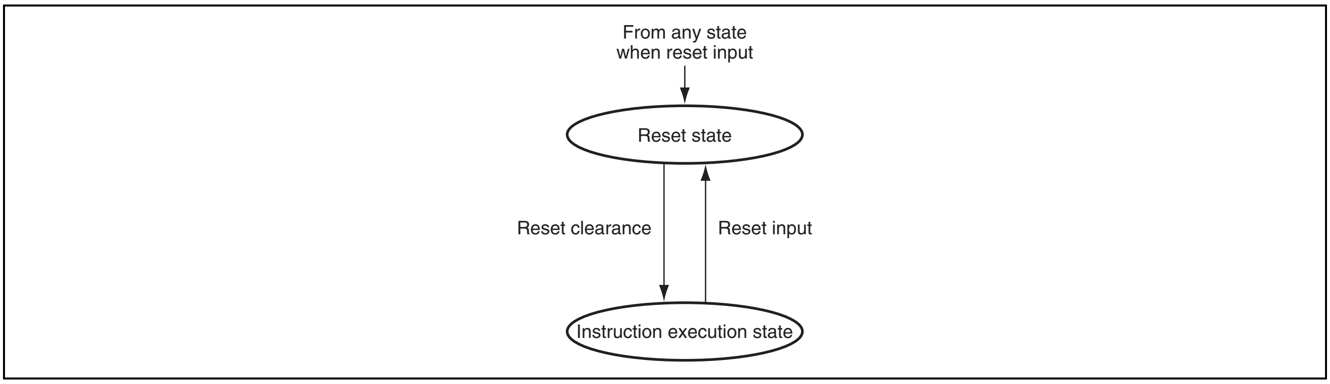


Figure 2.8 Processing State Transitions

2.7 Usage Notes

2.7.1 Notes on Self-Modifying Code

To accelerate the processing speed, the instruction prefetching capability of the SH-4A has been significantly enhanced from that of the SH-4. Therefore, in the case when a code in memory is rewritten and attempted to be executed immediately, there is increased possibility that the code before being modified, which has already been prefetched, is executed.

To ensure execution of the modified code, one of the following sequence of instructions should be executed between the code rewriting instruction and execution of the modified code.

(1) When the Codes to be Modified are in Non-Cacheable Area

```
SYNCO
ICBI @Rn
```

The target for the ICBI instruction can be any address within the range where no address error exception occurs.

(2) When the Codes to be Modified are in Cacheable Area (Write-Through)

```
SYNCO
ICBI @Rn
```

All instruction cache areas corresponding to the modified codes should be invalidated by the ICBI instruction. The ICBI instruction should be issued to each cache line. One cache line is 32 bytes.

(3) When the Codes to be Modified are in Cacheable Area (Copy-Back)

```
OCBP @Rm or OCBWB @Rm
SYNCO
ICBI @Rn
```

All operand cache areas corresponding to the modified codes should be written back to the main memory by the OCBP or OCBWB instruction. Then all instruction cache areas corresponding to the modified codes should be invalidated by the ICBI instruction. The OCBP, OCBWB, and ICBI instruction should be issued to each cache line. One cache line is 32 bytes.

Note: • Self-modifying code is code that dynamically rewrites itself while the instructions that make up the code are being executed.

This page is blank for reasons of layout.

Section 3 Instruction Set

The SH-4A's instruction set is implemented with 16-bit fixed-length instructions. The SH-4A can use byte (8-bit), word (16-bit), longword (32-bit), and quadword (64-bit) data sizes for memory access. Single-precision floating-point data (32 bits) can be moved to and from memory using longword or quadword size. Double-precision floating-point data (64 bits) can be moved to and from memory using longword size. When the SH-4A moves byte-size or word-size data from memory to a register, the data is sign-extended.

3.1 Execution Environment

(1) PC

At the start of instruction execution, the PC indicates the address of the instruction itself.

(2) Load-Store Architecture

The SH-4A has a load-store architecture in which operations are basically executed using registers. Except for bit-manipulation operations such as logical AND that are executed directly in memory, operands in an operation that requires memory access are loaded into registers and the operation is executed between the registers.

(3) Delayed Branches

Except for the two branch instructions BF and BT, the SH-4A's branch instructions and RTE are delayed branches. In a delayed branch, the instruction following the branch is executed before the branch destination instruction.

(4) Delay Slot

This execution slot following a delayed branch is called a delay slot. For example, the BRA execution sequence is as follows:

Table 3.1 Execution Order of Delayed Branch Instructions

Instructions				Execution Order
	BRA	TARGET	(Delayed branch instruction)	BRA
	ADD		(Delay slot)	↓
	:			ADD
	:			↓
TARGET	target-inst		(Branch destination instruction)	target-inst

A slot illegal instruction exception may occur when a specific instruction is executed in a delay slot. For details, see section 5, Exception Handling. The instruction following BF/S or BT/S for which the branch is not taken is also a delay slot instruction.

(5) T Bit

The T bit in SR is used to show the result of a compare operation, and is referenced by a conditional branch instruction. An example of the use of a conditional branch instruction is shown below.

```
ADD      #1, R0     ; T bit is not changed by ADD operation
CMP/EQ   R1, R0     ; If R0 = R1, T bit is set to "1"
BT       TARGET     ; Branches to TARGET if T bit = 1 (R0 = R1)
```

In an RTE delay slot, the SR bits are referenced as follows. In instruction access, the MD bit is used before modification, and in data access, the MD bit is accessed after modification. The other bits—S, T, M, Q, FD, BL, and RB—after modification are used for delay slot instruction execution. The STC and STC.L SR instructions access all SR bits after modification.

(6) Constant Values

An 8-bit constant value can be specified by the instruction code and an immediate value. 16-bit and 32-bit constant values can be defined as literal constant values in memory, and can be referenced by a PC-relative load instruction.

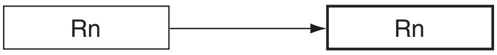
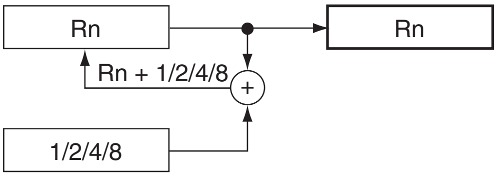
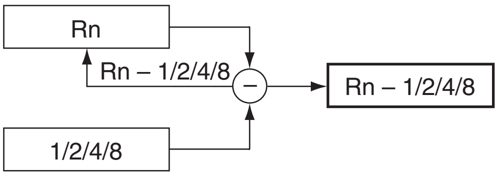
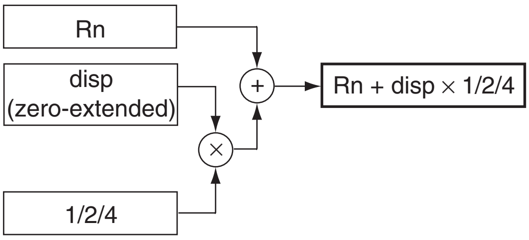
```
MOV.W    @(disp, PC), Rn
MOV.L    @(disp, PC), Rn
```

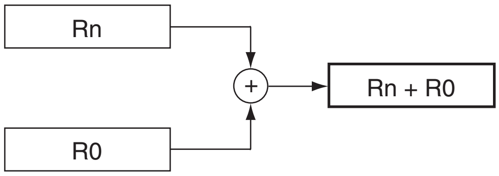
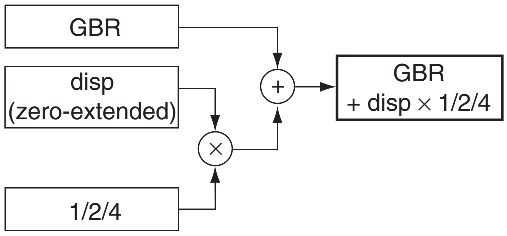
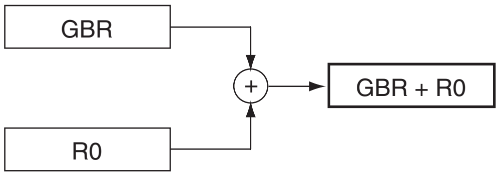
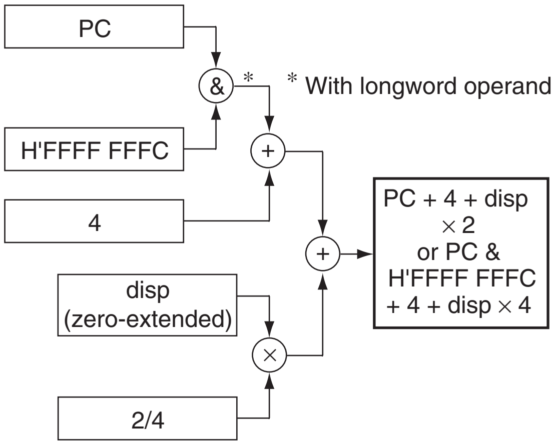
There are no PC-relative load instructions for floating-point operations. However, it is possible to set 0.0 or 1.0 by using the FLDI0 or FLDI1 instruction on a single-precision floating-point register.

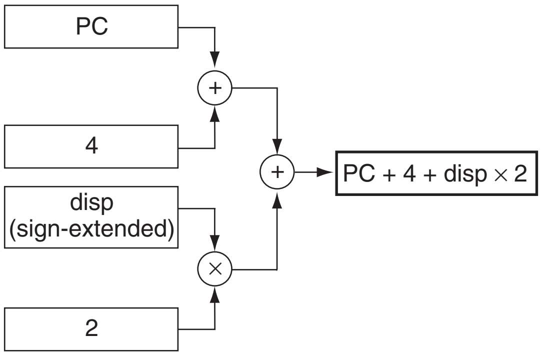
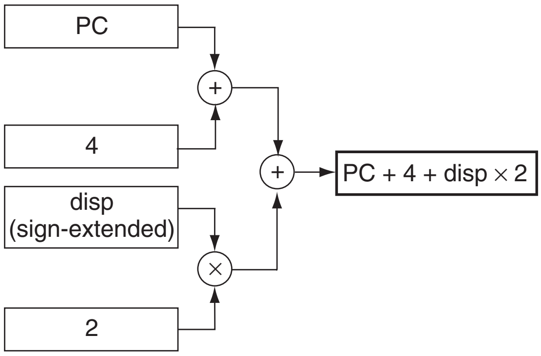
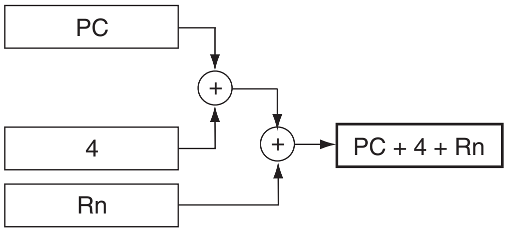
3.2 Addressing Modes

Addressing modes and effective address calculation methods are shown in table 3.2. When a location in virtual memory space is accessed (AT in MMUCR = 1), the effective address is translated into a physical memory address. If multiple virtual memory space systems are selected (SV in MMUCR = 0), the least significant bit of PTEH is also referenced as the access ASID. For details, see section 7, Memory Management Unit (MMU).

Table 3.2 Addressing Modes and Effective Addresses

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Register direct	Rn	Effective address is register Rn. (Operand is register Rn contents.)	—
Register indirect	@Rn	Effective address is register Rn contents. 	Rn → EA (EA: effective address)
Register indirect with post-increment	@Rn+	Effective address is register Rn contents. A constant is added to Rn after instruction execution: 1 for a byte operand, 2 for a word operand, 4 for a longword operand, 8 for a quadword operand. 	Rn → EA After instruction execution Byte: Rn + 1 → Rn Word: Rn + 2 → Rn Longword: Rn + 4 → Rn Quadword: Rn + 8 → Rn
Register indirect with pre-decrement	@–Rn	Effective address is register Rn contents, decremented by a constant beforehand: 1 for a byte operand, 2 for a word operand, 4 for a longword operand, 8 for a quadword operand. 	Byte: Rn – 1 → Rn Word: Rn – 2 → Rn Longword: Rn – 4 → Rn Quadword: Rn – 8 → Rn Rn → EA (Instruction executed with Rn after calculation)
Register indirect with displacement	@(disp:4, Rn)	Effective address is register Rn contents with 4-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size.  Rn disp (zero-extended) + Rn + disp × 1/2/4 × 1/2/4	Byte: Rn + disp → EA Word: Rn + disp × 2 → EA Longword: Rn + disp × 4 → EA

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Indexed register indirect	@(R0, Rn)	Effective address is sum of register Rn and R0 contents. 	Rn + R0 → EA
GBR indirect with displacement	@(disp:8, GBR)	Effective address is register GBR contents with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size.  GBR disp (zero-extended) 1/2/4 × + GBR + disp × 1/2/4	Byte: GBR + disp → EA Word: GBR + disp × 2 → EA Longword: GBR + disp × 4 → EA
Indexed GBR indirect	@(R0, GBR)	Effective address is sum of register GBR and R0 contents. 	GBR + R0 → EA
PC-relative with displacement	@(disp:8, PC)	Effective address is PC + 4 with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 2 (word), or 4 (longword), according to the operand size. With a longword operand, the lower 2 bits of PC are masked.  PC & * * With longword operand H'FFFF FFFC 4 + disp (zero-extended) 2/4 × + PC + 4 + disp × 2 or PC & H'FFFF FFFC + 4 + disp × 4	Word: PC + 4 + disp × 2 → EA Longword: PC & H'FFFF FFFC + 4 + disp × 4 → EA

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
PC-relative	disp:8	Effective address is PC + 4 with 8-bit displacement disp added after being sign-extended and multiplied by 2.  PC + 4 + PC + 4 + disp × 2 disp (sign-extended) × 2	PC + 4 + disp × 2 → Branch-Target
	disp:12	Effective address is PC + 4 with 12-bit displacement disp added after being sign-extended and multiplied by 2.  PC + 4 + PC + 4 + disp × 2 disp (sign-extended) × 2	PC + 4 + disp × 2 → Branch-Target
	Rn	Effective address is sum of PC + 4 and Rn.  	PC + 4 + Rn → Branch-Target
Immediate	#imm:8	8-bit immediate data imm of TST, AND, OR, or XOR instruction is zero-extended.	—
	#imm:8	8-bit immediate data imm of MOV, ADD, or CMP/EQ instruction is sign-extended.	—
	#imm:8	8-bit immediate data imm of TRAPA instruction is zero-extended and multiplied by 4.	—

Note: • For the addressing modes below that use a displacement (disp), the assembler descriptions in this manual show the value before scaling (×1, ×2, or ×4) is performed according to the operand size. This is done to clarify the operation of the MCU. Refer to the relevant assembler notation rules for the actual assembler descriptions.

@ (disp:4, Rn)	; Register indirect with displacement
@ (disp:8, GBR)	; GBR indirect with displacement
@ (disp:8, PC)	; PC-relative with displacement
disp:8, disp:12	; PC-relative

3.3 Instruction Set

Table 3.3 shows the notation used in the SH instruction lists shown in tables 3.4 to 3.13.

Table 3.3 Notation Used in Instruction List

Item	Format	Description	
Instruction mnemonic	OP.Sz SRC, DEST	OP:	Operation code
		Sz:	Size
		SRC:	Source operand
		DEST:	Source and/or destination operand
		Rm:	Source register
		Rn:	Destination register
		imm:	Immediate data
		disp:	Displacement
Operation notation		→, ←	Transfer direction
		(xx)	Memory operand
		M/Q/T	SR flag bits
		&	Logical AND of individual bits
		|	Logical OR of individual bits
		^	Logical exclusive-OR of individual bits
		~	Logical NOT of individual bits
		<<n, >>n	n-bit shift
Instruction code	MSB ↔ LSB	mmmm:	Register number (Rm, FRm)
		nnnn:	Register number (Rn, FRn)
		0000:	R0, FR0
		0001:	R1, FR1
		:	
		1111:	R15, FR15
		mmm:	Register number (DRm, XDm, Rm_BANK)
		nnn:	Register number (DRn, XDn, Rn_BANK)
		000:	DR0, XD0, R0_BANK
		001:	DR2, XD2, R1_BANK
		:	
		111:	DR14, XD14, R7_BANK
		mm:	Register number (FVm)
		nn:	Register number (FVn)
		00:	FV0
		01:	FV4
		10:	FV8
		11:	FV12
		iiii:	Immediate data
		dddd:	Displacement
Privileged mode	—	"Privileged" means the instruction can only be executed in privileged mode.	
T bit	Value of T bit after instruction execution	—: No change	
New	—	"New" means that the instruction was newly added to the SH-4A version in which the value of the VER field in the processor version register (PVR) is H'20.	

Note: • Scaling (×1, ×2, ×4, or ×8) is executed according to the size of the instruction operand.

Table 3.4 Fixed-Point Transfer Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MOV	#imm,Rn	imm → sign extension → Rn	1110nnnniiiiiiii	—	—	—
MOV.W	@(disp*[1],PC), Rn	(disp × 2 + PC + 4) → sign extension → Rn	1001nnnndddddddd	—	—	—
MOV.L	@(disp*[1],PC), Rn	(disp × 4 + PC & H'FFFF FFFC + 4) → Rn	1101nnnndddddddd	—	—	—
MOV	Rm,Rn	Rm → Rn	0110nnnnmmmm0011	—	—	—
MOV.B	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0000	—	—	—
MOV.W	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0001	—	—	—
MOV.L	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0010	—	—	—
MOV.B	@Rm,Rn	(Rm) → sign extension → Rn	0110nnnnmmmm0000	—	—	—
MOV.W	@Rm,Rn	(Rm) → sign extension → Rn	0110nnnnmmmm0001	—	—	—
MOV.L	@Rm,Rn	(Rm) → Rn	0110nnnnmmmm0010	—	—	—
MOV.B	Rm,@-Rn	Rn-1 → Rn, Rm → (Rn)	0010nnnnmmmm0100	—	—	—
MOV.W	Rm,@-Rn	Rn-2 → Rn, Rm → (Rn)	0010nnnnmmmm0101	—	—	—
MOV.L	Rm,@-Rn	Rn-4 → Rn, Rm → (Rn)	0010nnnnmmmm0110	—	—	—
MOV.B	@Rm+,Rn	(Rm)→ sign extension → Rn, Rm + 1 → Rm	0110nnnnmmmm0100	—	—	—
MOV.W	@Rm+,Rn	(Rm) → sign extension → Rn, Rm + 2 → Rm	0110nnnnmmmm0101	—	—	—
MOV.L	@Rm+,Rn	(Rm) → Rn, Rm + 4 → Rm	0110nnnnmmmm0110	—	—	—
MOV.B	R0,@(disp*[1],Rn)	R0 → (disp + Rn)	10000000nnnndddd	—	—	—
MOV.W	R0,@(disp*[1],Rn)	R0 → (disp × 2 + Rn)	10000001nnnndddd	—	—	—
MOV.L	Rm,@(disp*[1],Rn)	Rm → (disp × 4 + Rn)	0001nnnnmmmmdddd	—	—	—
MOV.B	@(disp*[1],Rm),R0	(disp + Rm) → sign extension → R0	10000100mmmmdddd	—	—	—
MOV.W	@(disp*[1],Rm),R0	(disp × 2 + Rm) → sign extension → R0	10000101mmmmdddd	—	—	—
MOV.L	@(disp*[1],Rm),Rn	(disp × 4 + Rm) → Rn	0101nnnnmmmmdddd	—	—	—
MOV.B	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0100	—	—	—
MOV.W	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0101	—	—	—
MOV.L	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0110	—	—	—
MOV.B	@(R0,Rm),Rn	(R0 + Rm) → sign extension → Rn	0000nnnnmmmm1100	—	—	—
MOV.W	@(R0,Rm),Rn	(R0 + Rm) → sign extension → Rn	0000nnnnmmmm1101	—	—	—
MOV.L	@(R0,Rm),Rn	(R0 + Rm) → Rn	0000nnnnmmmm1110	—	—	—
MOV.B	R0,@(disp*[1],GBR)	R0 → (disp + GBR)	11000000dddddddd	—	—	—
MOV.W	R0,@(disp*[1],GBR)	R0 → (disp × 2 + GBR)	11000001dddddddd	—	—	—
MOV.L	R0,@(disp*[1],GBR)	R0 → (disp × 4 + GBR)	11000010dddddddd	—	—	—
MOV.B	@(disp*[1],GBR),R0	(disp + GBR) → sign extension → R0	11000100dddddddd	—	—	—
MOV.W	@(disp*[1],GBR),R0	(disp × 2 + GBR) → sign extension → R0	11000101dddddddd	—	—	—
MOV.L	@(disp*[1],GBR),R0	(disp × 4 + GBR) → R0	11000110dddddddd	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MOVA	@(disp*1,PC),R0	disp × 4 + PC & H'FFFF FFFC + 4 → R0	11000111dddddddd	—	—	—
MOVCO.L	R0,@Rn	LDST → T If (T == 1) R0 → (Rn) 0 → LDST	0000nnnn01110011	—	LDST	New
MOVLI.L	@Rm,R0	1 → LDST (Rm) → R0 When interrupt/exception occurred 0 → LDST	0000mmmm01100011	—	—	New
MOVUA.L	@Rm,R0	(Rm) → R0 Load non-boundary alignment data	0100mmmm10101001	—	—	New
MOVUA.L	@Rm+,R0	(Rm) → R0, Rm + 4 → Rm Load non-boundary alignment data	0100mmmm11101001	—	—	New
MOVT	Rn	T → Rn	0000nnnn00101001	—	—	—
SWAP.B	Rm,Rn	Rm → swap lower 2 bytes → Rn	0110nnnnmmmm1000	—	—	—
SWAP.W	Rm,Rn	Rm → swap upper/lower words → Rn	0110nnnnmmmm1001	—	—	—
XTRCT	Rm,Rn	Rm:Rn middle 32 bits → Rn	0010nnnnmmmm1101	—	—	—

Note: *1 The assembler of Renesas uses the value after scaling (×1, ×2, or ×4) as the displacement (disp).

Table 3.5 Arithmetic Operation Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
ADD	Rm,Rn	Rn + Rm → Rn	0011nnnnmmmm1100	—	—	—
ADD	#imm,Rn	Rn + imm → Rn	0111nnnniiiiiiii	—	—	—
ADDC	Rm,Rn	Rn + Rm + T → Rn, carry → T	0011nnnnmmmm1110	—	Carry	—
ADDV	Rm,Rn	Rn + Rm → Rn, overflow → T	0011nnnnmmmm1111	—	Overflow	—
CMP/EQ	#imm,R0	When R0 = imm, 1 → T Otherwise, 0 → T	10001000iiiiiiii	—	Comparison result	—
CMP/EQ	Rm,Rn	When Rn = Rm, 1 → T Otherwise, 0 → T	0011nnnnmmmm0000	—	Comparison result	—
CMP/HS	Rm,Rn	When Rn ≥ Rm (unsigned), 1 → T Otherwise, 0 → T	0011nnnnmmmm0010	—	Comparison result	—
CMP/GE	Rm,Rn	When Rn ≥ Rm (signed), 1 → T Otherwise, 0 → T	0011nnnnmmmm0011	—	Comparison result	—
CMP/HI	Rm,Rn	When Rn > Rm (unsigned), 1 → T Otherwise, 0 → T	0011nnnnmmmm0110	—	Comparison result	—
CMP/GT	Rm,Rn	When Rn > Rm (signed), 1 → T Otherwise, 0 → T	0011nnnnmmmm0111	—	Comparison result	—
CMP/PZ	Rn	When Rn ≥ 0, 1 → T Otherwise, 0 → T	0100nnnn00010001	—	Comparison result	—
CMP/PL	Rn	When Rn > 0, 1 → T Otherwise, 0 → T	0100nnnn00010101	—	Comparison result	—
CMP/STR	Rm,Rn	When any bytes are equal, 1 → T Otherwise, 0 → T	0010nnnnmmmm1100	—	Comparison result	—
DIV1	Rm,Rn	1-step division (Rn ÷ Rm)	0011nnnnmmmm0100	—	Calculation result	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
DIV0S	Rm,Rn	MSB of Rn → Q, MSB of Rm → M, M^Q → T	0010nnnnmmmm0111	—	Calculation result	—
DIV0U		0 → M/Q/T	0000000000011001	—	0	—
DMULS.L	Rm,Rn	Signed, Rn × Rm → MAC, 32 × 32 → 64 bits	0011nnnnmmmm1101	—	—	—
DMULU.L	Rm,Rn	Unsigned, Rn × Rm → MAC, 32 × 32 → 64 bits	0011nnnnmmmm0101	—	—	—
DT	Rn	Rn – 1 → Rn; when Rn = 0, 1 → T When Rn ≠ 0, 0 → T	0100nnnn00010000	—	Comparison result	—
EXTS.B	Rm,Rn	Rm sign-extended from byte → Rn	0110nnnnmmmm1110	—	—	—
EXTS.W	Rm,Rn	Rm sign-extended from word → Rn	0110nnnnmmmm1111	—	—	—
EXTU.B	Rm,Rn	Rm zero-extended from byte → Rn	0110nnnnmmmm1100	—	—	—
EXTU.W	Rm,Rn	Rm zero-extended from word → Rn	0110nnnnmmmm1101	—	—	—
MAC.L	@Rm+,@Rn+	Signed, (Rn) × (Rm) + MAC → MAC Rn + 4 → Rn, Rm + 4 → Rm 32 × 32 + 64 → 64 bits	0000nnnnmmmm1111	—	—	—
MAC.W	@Rm+,@Rn+	Signed, (Rn) × (Rm) + MAC → MAC Rn + 2 → Rn, Rm + 2 → Rm 16 × 16 + 64 → 64 bits	0100nnnnmmmm1111	—	—	—
MUL.L	Rm,Rn	Rn × Rm → MACL 32 × 32 → 32 bits	0000nnnnmmmm0111	—	—	—
MULS.W	Rm,Rn	Signed, Rn × Rm → MACL 16 × 16 → 32 bits	0010nnnnmmmm1111	—	—	—
MULU.W	Rm,Rn	Unsigned, Rn × Rm → MACL 16 × 16 → 32 bits	0010nnnnmmmm1110	—	—	—
NEG	Rm,Rn	0 – Rm → Rn	0110nnnnmmmm1011	—	—	—
NEGC	Rm,Rn	0 – Rm – T → Rn, borrow → T	0110nnnnmmmm1010	—	Borrow	—
SUB	Rm,Rn	Rn – Rm → Rn	0011nnnnmmmm1000	—	—	—
SUBC	Rm,Rn	Rn – Rm – T → Rn, borrow → T	0011nnnnmmmm1010	—	Borrow	—
SUBV	Rm,Rn	Rn – Rm → Rn, underflow → T	0011nnnnmmmm1011	—	Underflow	—

Table 3.6 Logic Operation Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
AND	Rm,Rn	Rn & Rm → Rn	0010nnnnmmmm1001	—	—	—
AND	#imm,R0	R0 & imm → R0	11001001iiiiiiii	—	—	—
AND.B	#imm, @(R0,GBR)	(R0 + GBR) & imm → (R0 + GBR)	11001101iiiiiiii	—	—	—
NOT	Rm,Rn	~Rm → Rn	0110nnnnmmmm0111	—	—	—
OR	Rm,Rn	Rn | Rm → Rn	0010nnnnmmmm1011	—	—	—
OR	#imm,R0	R0 | imm → R0	11001011iiiiiiii	—	—	—
OR.B	#imm, @(R0,GBR)	(R0 + GBR) | imm → (R0 + GBR)	11001111iiiiiiii	—	—	—
TAS.B	@Rn	When (Rn) = 0, 1 → T Otherwise, 0 → T In both cases, 1 → MSB of (Rn)	0100nnnn00011011	—	Test result	—
TST	Rm,Rn	Rn & Rm; when result = 0, 1 → T Otherwise, 0 → T	0010nnnnmmmm1000	—	Test result	—
TST	#imm,R0	R0 & imm; when result = 0, 1 → T Otherwise, 0 → T	11001000iiiiiiii	—	Test result	—
TST.B	#imm, @(R0,GBR)	(R0 + GBR) & imm; when result = 0, 1 → T Otherwise, 0 → T	11001100iiiiiiii	—	Test result	—
XOR	Rm,Rn	Rn ∧ Rm → Rn	0010nnnnmmmm1010	—	—	—
XOR	#imm,R0	R0 ∧ imm → R0	11001010iiiiiiii	—	—	—
XOR.B	#imm, @(R0,GBR)	(R0 + GBR) ∧ imm → (R0 + GBR)	11001110iiiiiiii	—	—	—

Table 3.7 Shift Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
ROTL	Rn	T ← Rn ← MSB	`0100nnnn00000100`	—	MSB	—
ROTR	Rn	LSB → Rn → T	`0100nnnn00000101`	—	LSB	—
ROTCL	Rn	T ← Rn ← T	`0100nnnn00100100`	—	MSB	—
ROTCR	Rn	T → Rn → T	`0100nnnn00100101`	—	LSB	—
SHAD	Rm,Rn	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> Rm → [MSB → Rn]	`0100nnnnmmmm1100`	—	—	—
SHAL	Rn	T ← Rn ← 0	`0100nnnn00100000`	—	MSB	—
SHAR	Rn	MSB → Rn → T	`0100nnnn00100001`	—	LSB	—
SHLD	Rm,Rn	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> Rm → [0 → Rn]	`0100nnnnmmmm1101`	—	—	—
SHLL	Rn	T ← Rn ← 0	`0100nnnn00000000`	—	MSB	—
SHLR	Rn	0 → Rn → T	`0100nnnn00000001`	—	LSB	—
SHLL2	Rn	Rn << 2 → Rn	`0100nnnn00001000`	—	—	—
SHLR2	Rn	Rn >> 2 → Rn	`0100nnnn00001001`	—	—	—
SHLL8	Rn	Rn << 8 → Rn	`0100nnnn00011000`	—	—	—
SHLR8	Rn	Rn >> 8 → Rn	`0100nnnn00011001`	—	—	—
SHLL16	Rn	Rn << 16 → Rn	`0100nnnn00101000`	—	—	—
SHLR16	Rn	Rn >> 16 → Rn	`0100nnnn00101001`	—	—	—

Table 3.8 Branch Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
BF	label	When T = 0, disp × 2 + PC + 4 → PC When T = 1, nop	`10001011dddddddd`	—	—	—
BF/S	label	Delayed branch; when T = 0, disp × 2 + PC + 4 → PC When T = 1, nop	`10001111dddddddd`	—	—	—
BT	label	When T = 1, disp × 2 + PC + 4 → PC When T = 0, nop	`10001001dddddddd`	—	—	—
BT/S	label	Delayed branch; when T = 1, disp × 2 + PC + 4 → PC When T = 0, nop	`10001101dddddddd`	—	—	—
BRA	label	Delayed branch, disp × 2 + PC + 4 → PC	`1010dddddddddddd`	—	—	—
BRAF	Rn	Delayed branch, Rn + PC + 4 → PC	`0000nnnn00100011`	—	—	—
BSR	label	Delayed branch, PC + 4 → PR, disp × 2 + PC + 4 → PC	`1011dddddddddddd`	—	—	—
BSRF	Rn	Delayed branch, PC + 4 → PR, Rn + PC + 4 → PC	`0000nnnn00000011`	—	—	—
JMP	@Rn	Delayed branch, Rn → PC	`0100nnnn00101011`	—	—	—
JSR	@Rn	Delayed branch, PC + 4 → PR, Rn → PC	`0100nnnn00001011`	—	—	—
RTS		Delayed branch, PR → PC	`0000000000001011`	—	—	—

Table 3.9 System Control Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
CLRMAC		0 → MACH, MACL	0000000000101000	—	—	—
CLRS		0 → S	0000000001001000	—	—	—
CLRT		0 → T	0000000000001000	—	0	—
ICBI	@Rn	Invalidates instruction cache block	0000nnnn11100011	—	—	New
LDC	Rm,SR	Rm → SR	0100mmmm00001110	Privileged	LSB	—
LDC	Rm,GBR	Rm → GBR	0100mmmm00011110	—	—	—
LDC	Rm,VBR	Rm → VBR	0100mmmm00101110	Privileged	—	—
LDC	Rm,SGR	Rm → SGR	0100mmmm00111010	Privileged	—	New
LDC	Rm,SSR	Rm → SSR	0100mmmm00111110	Privileged	—	—
LDC	Rm,SPC	Rm → SPC	0100mmmm01001110	Privileged	—	—
LDC	Rm,DBR	Rm → DBR	0100mmmm11111010	Privileged	—	—
LDC	Rm,Rn_BANK	Rm → Rn_BANK (n = 0 to 7)	0100mmmm1nnn1110	Privileged	—	—
LDC.L	@Rm+,SR	(Rm) → SR, Rm + 4 → Rm	0100mmmm00000111	Privileged	LSB	—
LDC.L	@Rm+,GBR	(Rm) → GBR, Rm + 4 → Rm	0100mmmm00010111	—	—	—
LDC.L	@Rm+,VBR	(Rm) → VBR, Rm + 4 → Rm	0100mmmm00100111	Privileged	—	—
LDC.L	@Rm+,SGR	(Rm) → SGR, Rm + 4 → Rm	0100mmmm00110110	Privileged	—	New
LDC.L	@Rm+,SSR	(Rm) → SSR, Rm + 4 → Rm	0100mmmm00110111	Privileged	—	—
LDC.L	@Rm+,SPC	(Rm) → SPC, Rm + 4 → Rm	0100mmmm01000111	Privileged	—	—
LDC.L	@Rm+,DBR	(Rm) → DBR, Rm + 4 → Rm	0100mmmm11110110	Privileged	—	—
LDC.L	@Rm+,Rn_ BANK	(Rm) → Rn_BANK, Rm + 4 → Rm	0100mmmm1nnn0111	Privileged	—	—
LDS	Rm,MACH	Rm → MACH	0100mmmm00001010	—	—	—
LDS	Rm,MACL	Rm → MACL	0100mmmm00011010	—	—	—
LDS	Rm,PR	Rm → PR	0100mmmm00101010	—	—	—
LDS.L	@Rm+,MACH	(Rm) → MACH, Rm + 4 → Rm	0100mmmm00000110	—	—	—
LDS.L	@Rm+,MACL	(Rm) → MACL, Rm + 4 → Rm	0100mmmm00010110	—	—	—
LDS.L	@Rm+,PR	(Rm) → PR, Rm + 4 → Rm	0100mmmm00100110	—	—	—
LDTLB		PTEH/PTEL → TLB	0000000000111000	Privileged	—	—
MOVCA.L	R0,@Rn	R0 → (Rn) (without fetching cache block)	0000nnnn11000011	—	—	—
NOP		No operation	0000000000001001	—	—	—
OCBI	@Rn	Invalidates operand cache block	0000nnnn10010011	—	—	—
OCBP	@Rn	Writes back and invalidates operand cache block	0000nnnn10100011	—	—	—
OCBWB	@Rn	Writes back operand cache block	0000nnnn10110011	—	—	—
PREF	@Rn	(Rn) → operand cache	0000nnnn10000011	—	—	—
PREFI	@Rn	Reads 32-byte instruction block into instruction cache	0000nnnn11010011	—	—	New
RTE		Delayed branch, SSR/SPC → SR/PC	0000000000101011	Privileged	—	—
SETS		1 → S	0000000001011000	—	—	—
SETT		1 → T	0000000000011000	—	1	—
SLEEP*[1]		Sleep	0000000000011011	Privileged	—	—
STC	SR,Rn	SR → Rn	0000nnnn00000010	Privileged	—	—
STC	GBR,Rn	GBR → Rn	0000nnnn00010010	—	—	—
STC	VBR,Rn	VBR → Rn	0000nnnn00100010	Privileged	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
STC	SSR,Rn	SSR → Rn	0000nnnn00110010	Privileged	—	—
STC	SPC,Rn	SPC → Rn	0000nnnn01000010	Privileged	—	—
STC	SGR,Rn	SGR → Rn	0000nnnn00111010	Privileged	—	—
STC	DBR,Rn	DBR → Rn	0000nnnn11111010	Privileged	—	—
STC	Rm_BANK,Rn	Rm_BANK → Rn (m = 0 to 7)	0000nnnn1mmm0010	Privileged	—	—
STC.L	SR,@-Rn	Rn – 4 → Rn, SR → (Rn)	0100nnnn00000011	Privileged	—	—
STC.L	GBR,@-Rn	Rn – 4 → Rn, GBR → (Rn)	0100nnnn00010011	—	—	—
STC.L	VBR,@-Rn	Rn – 4 → Rn, VBR → (Rn)	0100nnnn00100011	Privileged	—	—
STC.L	SSR,@-Rn	Rn – 4 → Rn, SSR → (Rn)	0100nnnn00110011	Privileged	—	—
STC.L	SPC,@-Rn	Rn – 4 → Rn, SPC → (Rn)	0100nnnn01000011	Privileged	—	—
STC.L	SGR,@-Rn	Rn – 4 → Rn, SGR → (Rn)	0100nnnn00110010	Privileged	—	—
STC.L	DBR,@-Rn	Rn – 4 → Rn, DBR → (Rn)	0100nnnn11110010	Privileged	—	—
STC.L	Rm_BANK,@-Rn	Rn – 4 → Rn, Rm_BANK → (Rn) (m = 0 to 7)	0100nnnn1mmm0011	Privileged	—	—
STS	MACH,Rn	MACH → Rn	0000nnnn00001010	—	—	—
STS	MACL,Rn	MACL → Rn	0000nnnn00011010	—	—	—
STS	PR,Rn	PR → Rn	0000nnnn00101010	—	—	—
STS.L	MACH,@-Rn	Rn – 4 → Rn, MACH → (Rn)	0100nnnn00000010	—	—	—
STS.L	MACL,@-Rn	Rn – 4 → Rn, MACL → (Rn)	0100nnnn00010010	—	—	—
STS.L	PR,@-Rn	Rn – 4 → Rn, PR → (Rn)	0100nnnn00100010	—	—	—
SYNCO		Data accesses invoked by the following instructions are not executed until execution of data accesses which precede this instruction has been completed.	0000000010101011	—	—	New
TRAPA	#imm	#imm << 2 → TRA, PC + 2 → SPC, SR → SSR, R15 → SGR, 1 → SR.MD/BL/RB, H'160 → EXPEVT, VBR + H'0100 → PC	11000011iiiiiiii	—	—	—

Note: *1 Do not use the SLEEP instruction because sleep mode is not available in this MCU.

Table 3.10 Floating-Point Single-Precision Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FLDI0	FRn	H'0000 0000 → FRn	1111nnnn10001101	—	—	—
FLDI1	FRn	H'3F80 0000 → FRn	1111nnnn10011101	—	—	—
FMOV	FRm,FRn	FRm → FRn	1111nnnnmmmm1100	—	—	—
FMOV.S	@Rm,FRn	(Rm) → FRn	1111nnnnmmmm1000	—	—	—
FMOV.S	@(R0,Rm),FRn	(R0 + Rm) → FRn	1111nnnnmmmm0110	—	—	—
FMOV.S	@Rm+,FRn	(Rm) → FRn, Rm + 4 → Rm	1111nnnnmmmm1001	—	—	—
FMOV.S	FRm,@Rn	FRm → (Rn)	1111nnnnmmmm1010	—	—	—
FMOV.S	FRm,@-Rn	Rn-4 → Rn, FRm → (Rn)	1111nnnnmmmm1011	—	—	—
FMOV.S	FRm,@(R0,Rn)	FRm → (R0 + Rn)	1111nnnnmmmm0111	—	—	—
FMOV	DRm,DRn	DRm → DRn	1111nnn0mmm01100	—	—	—
FMOV	@Rm,DRn	(Rm) → DRn	1111nnn0mmmm1000	—	—	—
FMOV	@(R0,Rm),DRn	(R0 + Rm) → DRn	1111nnn0mmmm0110	—	—	—
FMOV	@Rm+,DRn	(Rm) → DRn, Rm + 8 → Rm	1111nnn0mmmm1001	—	—	—
FMOV	DRm,@Rn	DRm → (Rn)	1111nnnnmmm01010	—	—	—
FMOV	DRm,@-Rn	Rn-8 → Rn, DRm → (Rn)	1111nnnnmmm01011	—	—	—
FMOV	DRm,@(R0,Rn)	DRm → (R0 + Rn)	1111nnnnmmm00111	—	—	—
FLDS	FRm,FPUL	FRm → FPUL	1111mmmm00011101	—	—	—
FSTS	FPUL,FRn	FPUL → FRn	1111nnnn00001101	—	—	—
FABS	FRn	FRn & H'7FFF FFFF → FRn	1111nnnn01011101	—	—	—
FADD	FRm,FRn	FRn + FRm → FRn	1111nnnnmmmm0000	—	—	—
FCMP/EQ	FRm,FRn	When FRn = FRm, 1 → T Otherwise, 0 → T	1111nnnnmmmm0100	—	Comparison result	—
FCMP/GT	FRm,FRn	When FRn > FRm, 1 → T Otherwise, 0 → T	1111nnnnmmmm0101	—	Comparison result	—
FDIV	FRm,FRn	FRn/FRm → FRn	1111nnnnmmmm0011	—	—	—
FLOAT	FPUL,FRn	(float) FPUL → FRn	1111nnnn00101101	—	—	—
FMAC	FR0,FRm,FRn	FR0 × FRm + FRn → FRn	1111nnnnmmmm1110	—	—	—
FMUL	FRm,FRn	FRn × FRm → FRn	1111nnnnmmmm0010	—	—	—
FNEG	FRn	FRn ∧ H'8000 0000 → FRn	1111nnnn01001101	—	—	—
FSQRT	FRn	sqrt (FRn) → FRn*1	1111nnnn01101101	—	—	—
FSUB	FRm,FRn	FRn – FRm → FRn	1111nnnnmmmm0001	—	—	—
FTRC	FRm,FPUL	(long) FRm → FPUL	1111mmmm00111101	—	—	—

Note: *1 sqrt (FRn) is the square root of FRn.

Table 3.11 Floating-Point Double-Precision Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FABS	DRn	DRn & H'7FFF FFFF FFFF FFFF → DRn	1111nnn001011101	—	—	—
FADD	DRm,DRn	DRn + DRm → DRn	1111nnn0mmm00000	—	—	—
FCMP/EQ	DRm,DRn	When DRn = DRm, 1 → T Otherwise, 0 → T	1111nnn0mmm00100	—	Comparison result	—
FCMP/GT	DRm,DRn	When DRn > DRm, 1 → T Otherwise, 0 → T	1111nnn0mmm00101	—	Comparison result	—
FDIV	DRm,DRn	DRn /DRm → DRn	1111nnn0mmm00011	—	—	—
FCNVDS	DRm,FPUL	double_to_ float (DRm) → FPUL	1111mmm010111101	—	—	—
FCNVSD	FPUL,DRn	float_to_ double (FPUL) → DRn	1111nnn010101101	—	—	—
FLOAT	FPUL,DRn	(float)FPUL → DRn	1111nnn000101101	—	—	—
FMUL	DRm,DRn	DRn × DRm → DRn	1111nnn0mmm00010	—	—	—
FNEG	DRn	DRn ^ H'8000 0000 0000 0000 → DRn	1111nnn001001101	—	—	—
FSQRT	DRn	sqrt (FRn) → FRn*[1]	1111nnn001101101	—	—	—
FSUB	DRm,DRn	DRn – DRm → DRn	1111nnn0mmm00001	—	—	—
FTRC	DRm,FPUL	(long) DRm → FPUL	1111mmm000111101	—	—	—

Note: *1 sqrt (FRn) is the square root of FRn.

Table 3.12 Floating-Point Control Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
LDS	Rm,FPSCR	Rm → FPSCR	0100mmmm01101010	—	—	—
LDS	Rm,FPUL	Rm → FPUL	0100mmmm01011010	—	—	—
LDS.L	@Rm+,FPSCR	(Rm) → FPSCR, Rm+4 → Rm	0100mmmm01100110	—	—	—
LDS.L	@Rm+,FPUL	(Rm) → FPUL, Rm+4 → Rm	0100mmmm01010110	—	—	—
STS	FPSCR,Rn	FPSCR → Rn	0000nnnn01101010	—	—	—
STS	FPUL,Rn	FPUL → Rn	0000nnnn01011010	—	—	—
STS.L	FPSCR,@-Rn	Rn – 4 → Rn, FPSCR → (Rn)	0100nnnn01100010	—	—	—
STS.L	FPUL,@-Rn	Rn – 4 → Rn, FPUL → (Rn)	0100nnnn01010010	—	—	—

Table 3.13 Floating-Point Graphics Acceleration Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FMOV	DRm,XDn	DRm → XDn	1111nnn1mmm01100	—	—	—
FMOV	XDm,DRn	XDm → DRn	1111nnn0mmm11100	—	—	—
FMOV	XDm,XDn	XDm → XDn	1111nnn1mmm11100	—	—	—
FMOV	@Rm,XDn	(Rm) → XDn	1111nnn1mmmm1000	—	—	—
FMOV	@Rm+,XDn	(Rm) → XDn, Rm + 8 → Rm	1111nnn1mmmm1001	—	—	—
FMOV	@(R0,Rm),XDn	(R0 + Rm) → XDn	1111nnn1mmmm0110	—	—	—
FMOV	XDm,@Rn	XDm → (Rn)	1111nnnnmmm11010	—	—	—
FMOV	XDm,@-Rn	Rn – 8 → Rn, XDm → (Rn)	1111nnnnmmm11011	—	—	—
FMOV	XDm,@(R0,Rn)	XDm → (R0 + Rn)	1111nnnnmmm10111	—	—	—
FIPR	FVm,FVn	inner_product (FVm, FVn) → FR[n+3]	1111nnmm11101101	—	—	—
FTRV	XMTRX,FVn	transform_vector (XMTRX, FVn) → FVn	1111nn0111111101	—	—	—
FRCHG		~FPSCR.FR → FPSCR.FR	1111101111111101	—	—	—
FSCHG		~FPSCR.SZ → FPSCR.SZ	1111001111111101	—	—	—
FPCHG		~FPSCR.PR → FPSCR.PR	1111011111111101	—	—	New
FSRRA	FRn	1/sqrt(FRn) → FRn*[1]	1111nnnn01111101	—	—	New
FSCA	FPUL,DRn	sin(FPUL) → FRn cos(FPUL) → FR[n + 1]	1111nnn011111101	—	—	New

Note: *1 sqrt(FRn) is the square root of FRn.

3.4 Usage Notes

Do not use the SLEEP instruction because sleep mode is not available in this MCU.

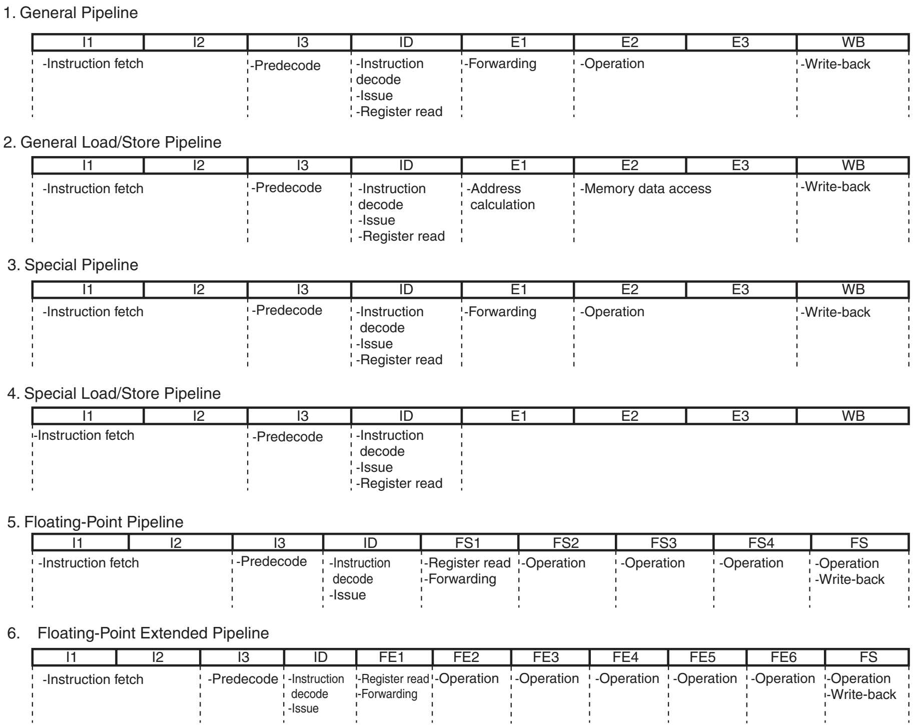
This page is blank for reasons of layout.

Section 4 Pipelining

The SH-4A is a 2-ILP (instruction-level-parallelism) superscalar pipelining microprocessor. Instruction execution is pipelined, and two instructions can be executed in parallel.

4.1 Pipelines

Figure 4.1 shows the basic pipelines. Normally, a pipeline consists of eight stages: instruction fetch (I1/I2/I3), decode and register read (ID), execution (E1/E2/E3), and write-back (WB). An instruction is executed as a combination of basic pipelines.



1. General Pipeline

I1	I2	I3	ID	E1	E2	E3	WB
-Instruction fetch		-Predecode	-Instruction decode -Issue -Register read	-Forwarding	-Operation		-Write-back

2. General Load/Store Pipeline

I1	I2	I3	ID	E1	E2	E3	WB
-Instruction fetch		-Predecode	-Instruction decode -Issue -Register read	-Address calculation	-Memory data access		-Write-back

3. Special Pipeline

I1	I2	I3	ID	E1	E2	E3	WB
-Instruction fetch		-Predecode	-Instruction decode -Issue -Register read	-Forwarding	-Operation		-Write-back

4. Special Load/Store Pipeline

I1	I2	I3	ID	E1	E2	E3	WB
-Instruction fetch		-Predecode	-Instruction decode -Issue -Register read				

5. Floating-Point Pipeline

I1	I2	I3	ID	FS1	FS2	FS3	FS4	FS
-Instruction fetch		-Predecode	-Instruction decode -Issue	-Register read -Forwarding	-Operation	-Operation	-Operation	-Operation -Write-back

6. Floating-Point Extended Pipeline

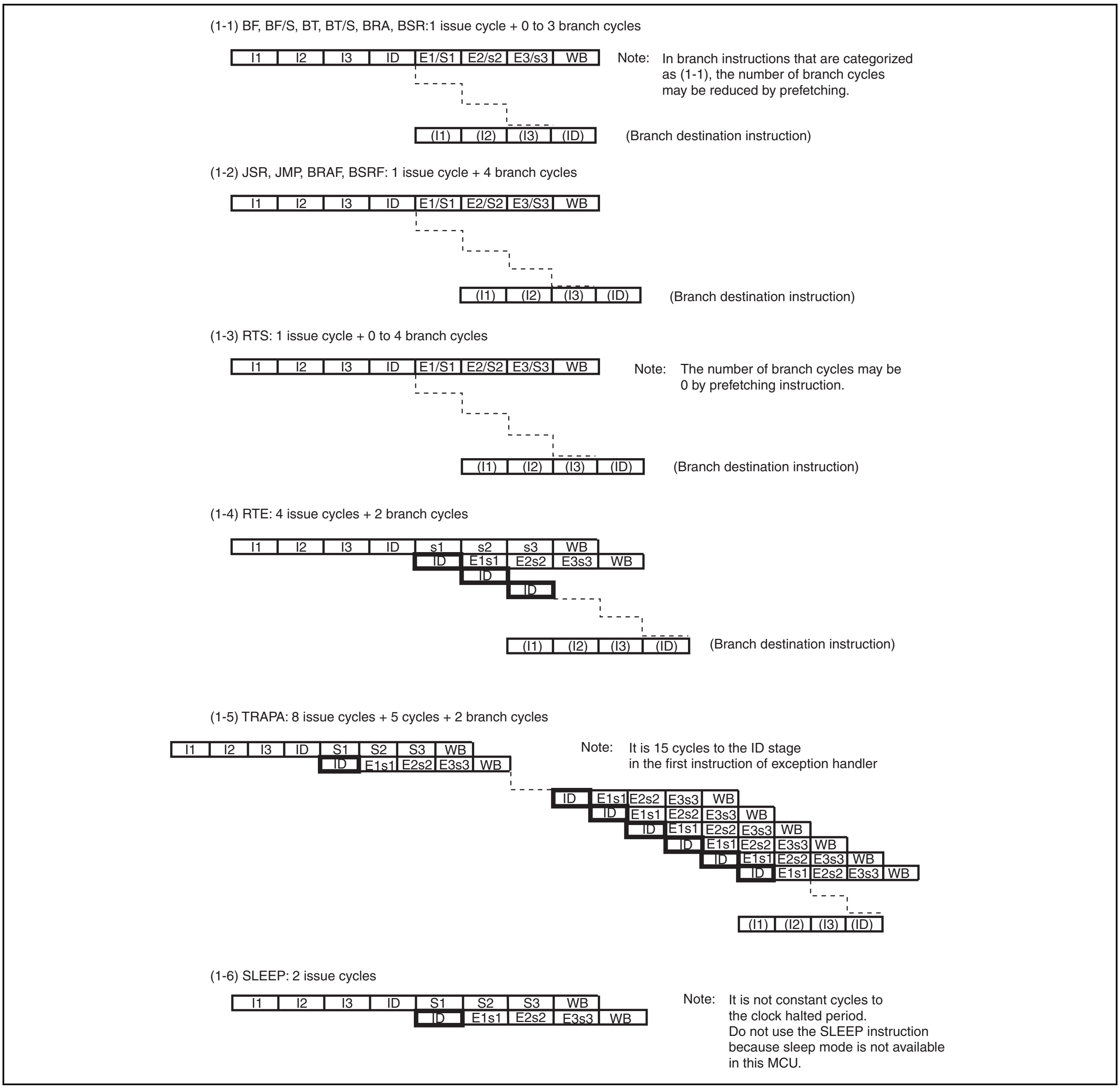
I1	I2	I3	ID	FE1	FE2	FE3	FE4	FE5	FE6	FS
-Instruction fetch		-Predecode	-Instruction decode -Issue	-Register read -Forwarding	-Operation	-Operation	-Operation	-Operation	-Operation	-Operation -Write-back

Figure 4.1 Basic Pipelines

Figure 4.2 shows the instruction execution patterns. Representations in figure 4.2 and their descriptions are listed in table 4.1.

Table 4.1 Representations of Instruction Execution Patterns

Representation	Description
E1 E2 E3 WB	CPU EX pipe is occupied
S1 S2 S3 WB	CPU LS pipe is occupied (with memory access)
s1 s2 s3 WB	CPU LS pipe is occupied (without memory access)
E1/S1	Either CPU EX pipe or CPU LS pipe is occupied
E1S1 , E1s1	Both CPU EX pipe and CPU LS pipe are occupied
M2 M3 MS	CPU MULT operation unit is occupied
FE1 FE2 FE3 FE4 FE5 FE6 FS	FPU-EX pipe is occupied
FS1 FS2 FS3 FS4 FS	FPU-LS pipe is occupied
ID	ID stage is locked
└──	Both CPU and FPU pipes are occupied



(1-1) BF, BF/S, BT, BT/S, BRA, BSR:1 issue cycle + 0 to 3 branch cycles

Note: In branch instructions that are categorized as (1-1), the number of branch cycles may be reduced by prefetching.

(Branch destination instruction)

(1-2) JSR, JMP, BRAF, BSRF: 1 issue cycle + 4 branch cycles

(Branch destination instruction)

(1-3) RTS: 1 issue cycle + 0 to 4 branch cycles

Note: The number of branch cycles may be 0 by prefetching instruction.

(Branch destination instruction)

(1-4) RTE: 4 issue cycles + 2 branch cycles

(Branch destination instruction)

(1-5) TRAPA: 8 issue cycles + 5 cycles + 2 branch cycles

Note: It is 15 cycles to the ID stage in the first instruction of exception handler

(1-6) SLEEP: 2 issue cycles

Note: It is not constant cycles to the clock halted period. Do not use the SLEEP instruction because sleep mode is not available in this MCU.

Figure 4.2 Instruction Execution Patterns (1)

(2-1) 1-step operation (EX type): 1 issue cycle

EXT[SU].[BW], MOVT, SWAP, XTRCT, ADD*, CMP*, DIV*, DT, NEG*, SUB*, AND, AND#, NOT, OR, OR#, TST, TST#, XOR, XOR#, ROT*, SHA*, SHL*, CLRS, CLRT, SETS, SETT

Note: Except for AND#, OR#, TST#, and XOR# instructions that use GBR relative addressing mode.

I1	I2	I3	ID	E1	E2	E3	WB

(2-2) 1-step operation (LS type): 1 issue cycle

MOVA

I1	I2	I3	ID	s1	s2	s3	WB

(2-3) 1-step operation (MT type): 1 issue cycle

MOV#, NOP

I1	I2	I3	ID	E1/S1	E2/s2	E3/s3	WB

(2-4) MOV (MT type): 1 issue cycle

MOV

I1	I2	I3	ID	E1/s1	E2/s2	E3/S3	WB

Figure 4.2 Instruction Execution Patterns (2)

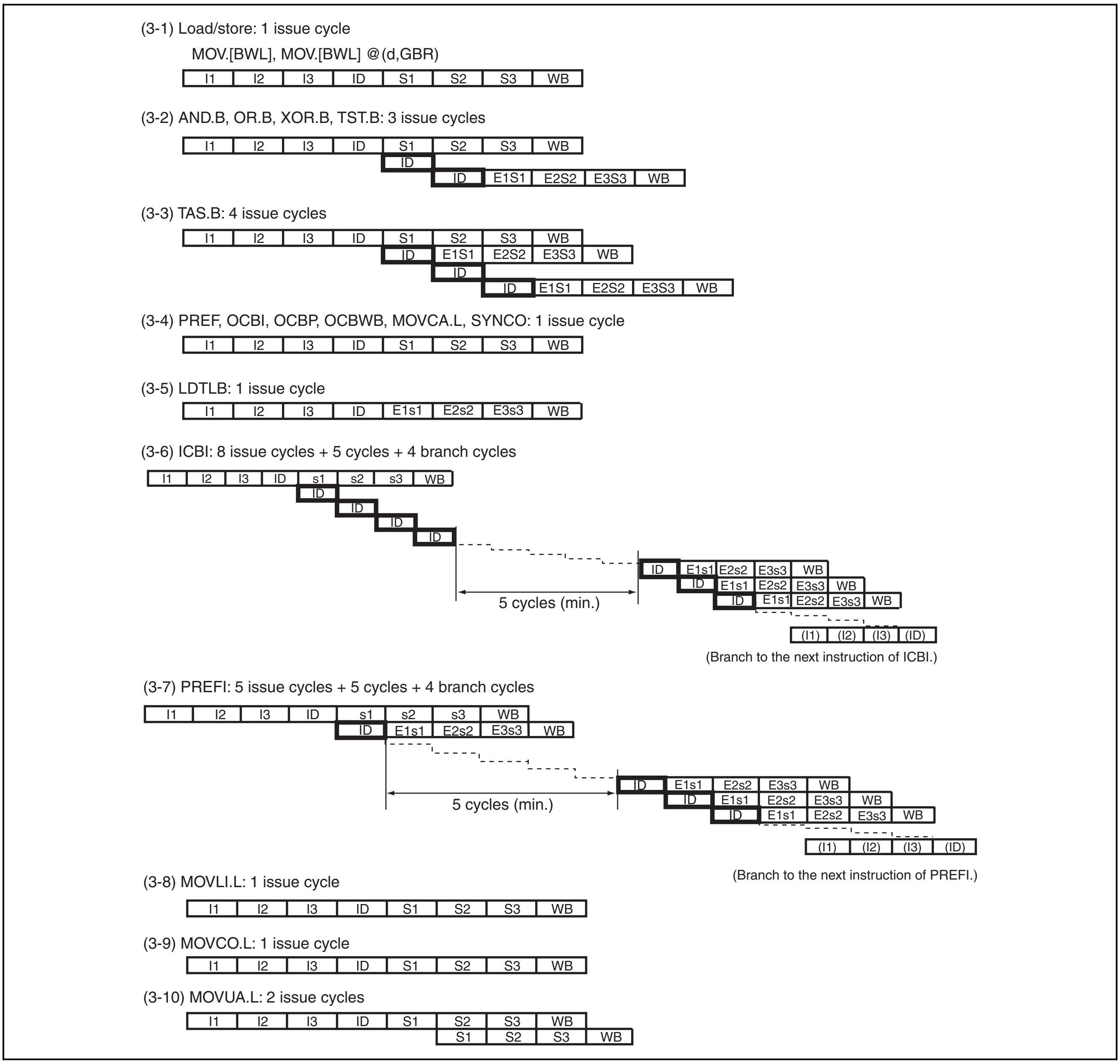
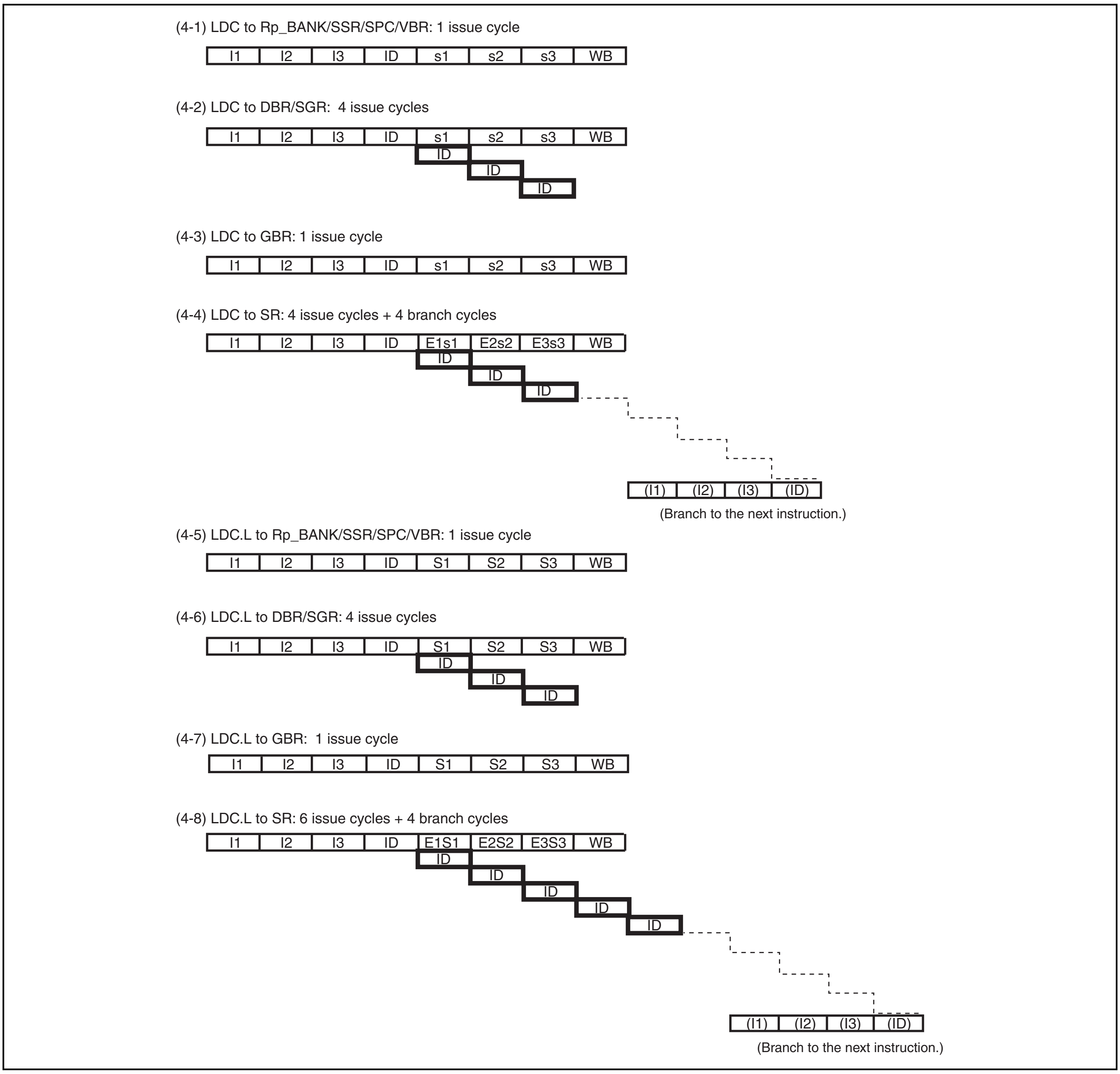

(3-1) Load/store: 1 issue cycle
MOV.[BWL], MOV.[BWL] @(d,GBR)
I1 I2 I3 ID S1 S2 S3 WB
(3-2) AND.B, OR.B, XOR.B, TST.B: 3 issue cycles
I1 I2 I3 ID S1 S2 S3 WB
ID
ID E1S1 E2S2 E3S3 WB
(3-3) TAS.B: 4 issue cycles
I1 I2 I3 ID S1 S2 S3 WB
ID E1S1 E2S2 E3S3 WB
ID
ID E1S1 E2S2 E3S3 WB
(3-4) PREF, OCBI, OCBP, OCBWB, MOVCA.L, SYNCO: 1 issue cycle
I1 I2 I3 ID S1 S2 S3 WB
(3-5) LDTLB: 1 issue cycle
I1 I2 I3 ID E1s1 E2s2 E3s3 WB
(3-6) ICBI: 8 issue cycles + 5 cycles + 4 branch cycles
I1 I2 I3 ID s1 s2 s3 WB
ID
ID
ID
ID
ID E1s1 E2s2 E3s3 WB
ID E1s1 E2s2 E3s3 WB
ID E1s1 E2s2 E3s3 WB
5 cycles (min.)
(I1) (I2) (I3) (ID)
(Branch to the next instruction of ICBI.)
(3-7) PREFI: 5 issue cycles + 5 cycles + 4 branch cycles
I1 I2 I3 ID s1 s2 s3 WB
ID E1s1 E2s2 E3s3 WB
ID E1s1 E2s2 E3s3 WB
ID E1s1 E2s2 E3s3 WB
ID E1s1 E2s2 E3s3 WB
5 cycles (min.)
(I1) (I2) (I3) (ID)
(Branch to the next instruction of PREFI.)
(3-8) MOVLI.L: 1 issue cycle
I1 I2 I3 ID S1 S2 S3 WB
(3-9) MOVCO.L: 1 issue cycle
I1 I2 I3 ID S1 S2 S3 WB
(3-10) MOVUA.L: 2 issue cycles
I1 I2 I3 ID S1 S2 S3 WB
S1 S2 S3 WB

Figure 4.2 Instruction Execution Patterns (3)



(4-1) LDC to Rp_BANK/SSR/SPC/VBR: 1 issue cycle

(4-2) LDC to DBR/SGR: 4 issue cycles

(4-3) LDC to GBR: 1 issue cycle

(4-4) LDC to SR: 4 issue cycles + 4 branch cycles

(4-5) LDC.L to Rp_BANK/SSR/SPC/VBR: 1 issue cycle

(4-6) LDC.L to DBR/SGR: 4 issue cycles

(4-7) LDC.L to GBR: 1 issue cycle

(4-8) LDC.L to SR: 6 issue cycles + 4 branch cycles

Figure 4.2 Instruction Execution Patterns (4)

(4-9) STC from DBR/GBR/Rp_BANK/SSR/SPC/VBR/SGR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-10) STC from SR: 1 issue cycle

I1	I2	I3	ID	E1s1	E2s2	E3s3	WB

(4-11) STC.L from DBR/GBR/Rp_BANK/SSR/SPC/VBR/SGR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-12) STC.L from SR: 1 issue cycle

I1	I2	I3	ID	E1S1	E2S2	E3S3	WB

(4-13) LDS to PR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-14) LDS.L to PR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-15) STS from PR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-16) STS.L from PR: 1 issue cycle

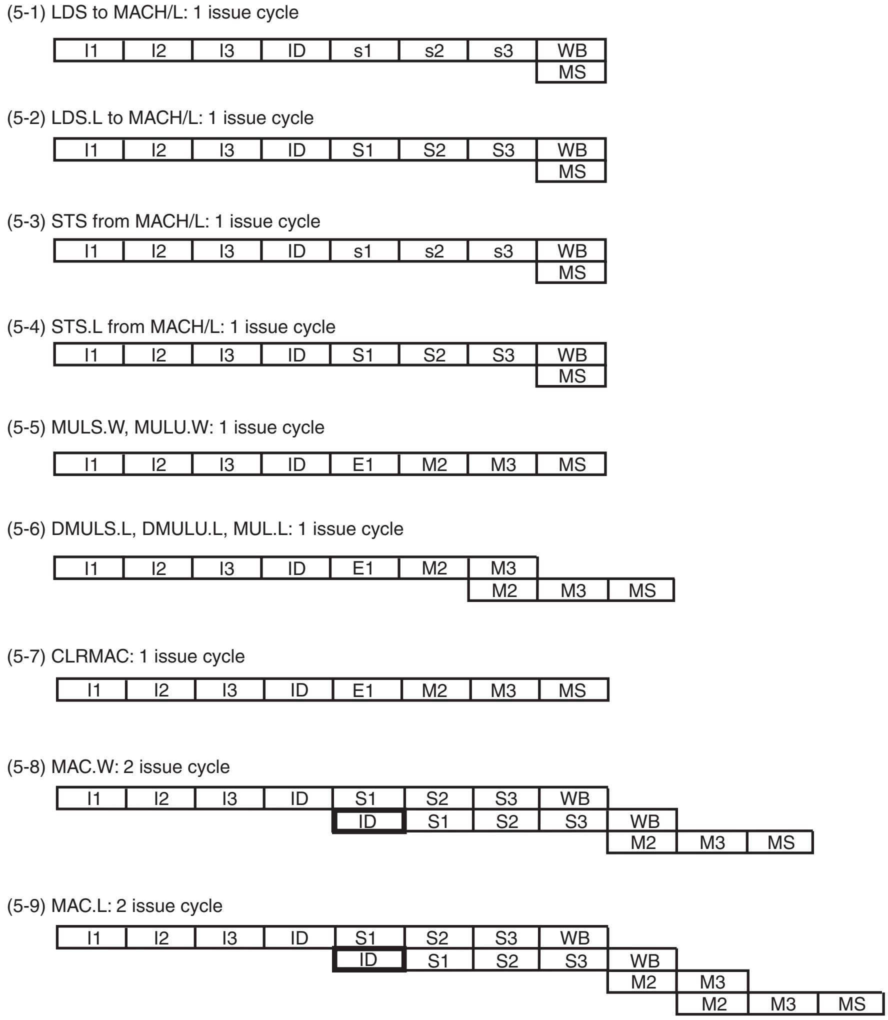
I1	I2	I3	ID	S1	S2	S3	WB

(4-17) BSRF, BSR, JSR delay slot instructions (PR set): 0 issue cycle

(I1)	(I2)	(I3)	(ID)	(??1)	(??2)	(??3)	(WB)

Notes: The value of PR is changed in the E3 stage of delay slot instruction.
When the STS and STS.L instructions from PR are used as delay slot instructions, changed PR value is used.

Figure 4.2 Instruction Execution Patterns (5)



(5-1) LDS to MACH/L: 1 issue cycle

(5-2) LDS.L to MACH/L: 1 issue cycle

(5-3) STS from MACH/L: 1 issue cycle

(5-4) STS.L from MACH/L: 1 issue cycle

(5-5) MULS.W, MULU.W: 1 issue cycle

(5-6) DMULS.L, DMULU.L, MUL.L: 1 issue cycle

(5-7) CLRMAC: 1 issue cycle

(5-8) MAC.W: 2 issue cycle

(5-9) MAC.L: 2 issue cycle

Figure 4.2 Instruction Execution Patterns (6)

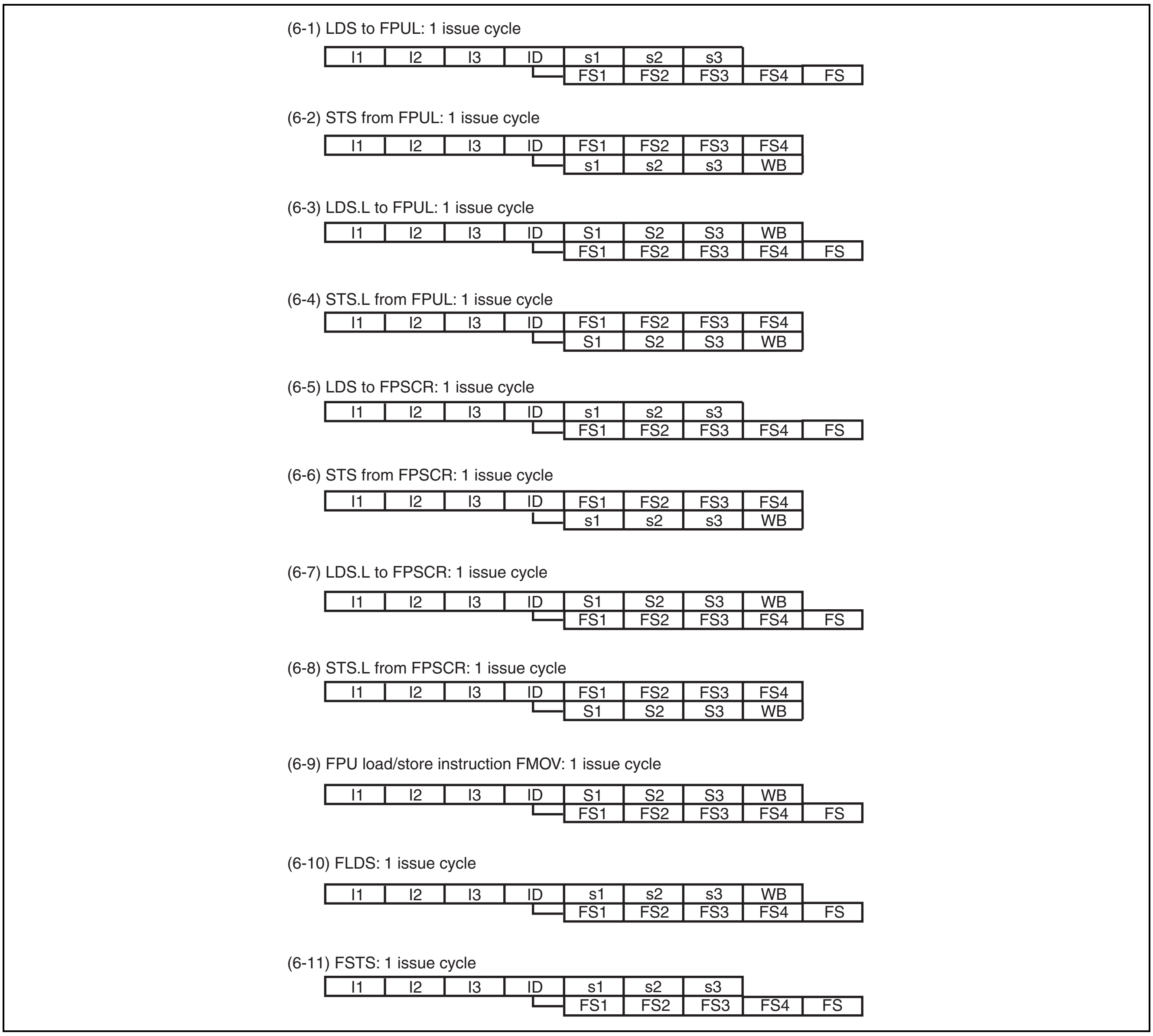
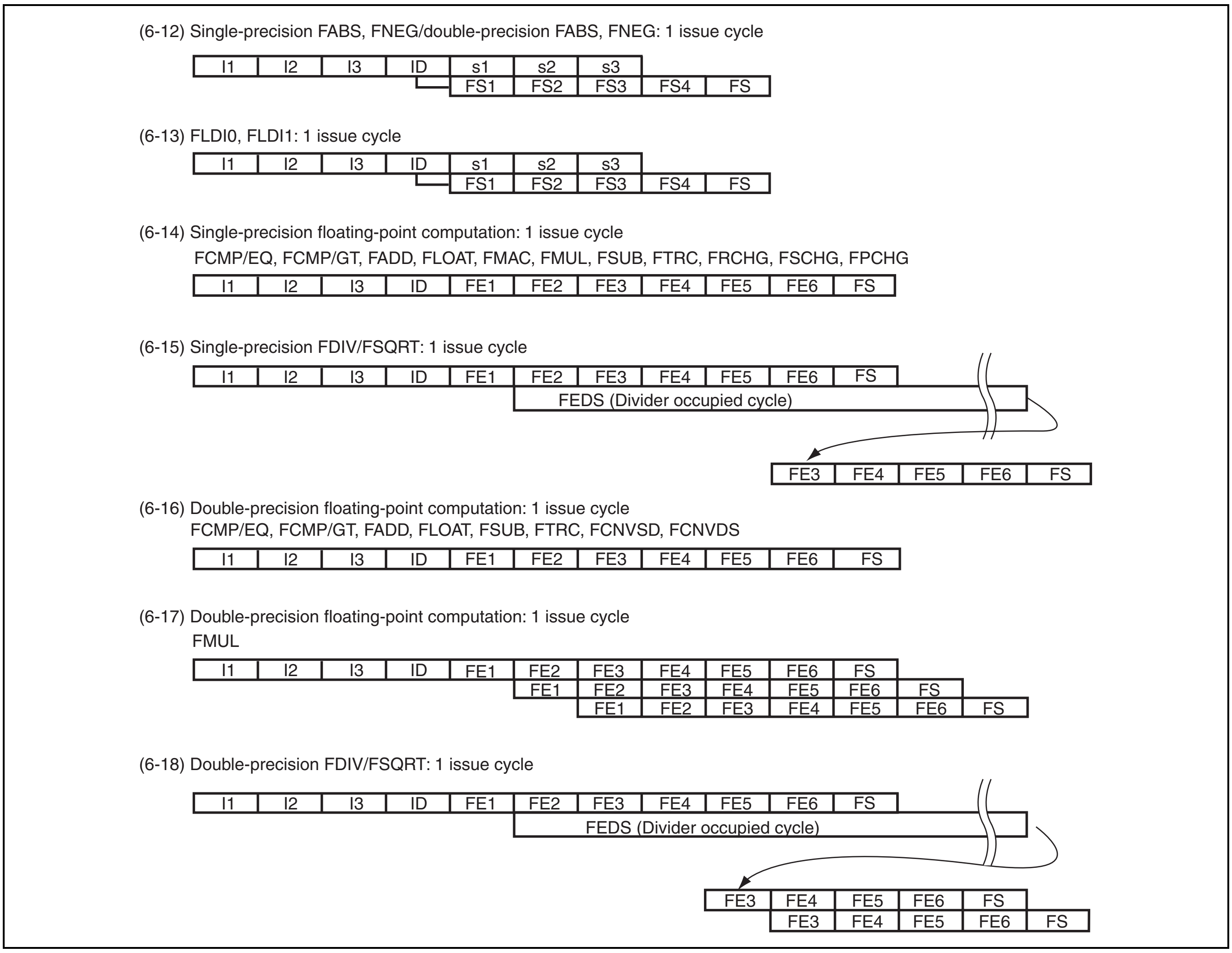

(6-1) LDS to FPUL: 1 issue cycle
I1 I2 I3 ID s1 s2 s3
FS1 FS2 FS3 FS4 FS
(6-2) STS from FPUL: 1 issue cycle
I1 I2 I3 ID FS1 FS2 FS3 FS4
s1 s2 s3 WB
(6-3) LDS.L to FPUL: 1 issue cycle
I1 I2 I3 ID S1 S2 S3 WB
FS1 FS2 FS3 FS4 FS
(6-4) STS.L from FPUL: 1 issue cycle
I1 I2 I3 ID FS1 FS2 FS3 FS4
S1 S2 S3 WB
(6-5) LDS to FPSCR: 1 issue cycle
I1 I2 I3 ID s1 s2 s3
FS1 FS2 FS3 FS4 FS
(6-6) STS from FPSCR: 1 issue cycle
I1 I2 I3 ID FS1 FS2 FS3 FS4
s1 s2 s3 WB
(6-7) LDS.L to FPSCR: 1 issue cycle
I1 I2 I3 ID S1 S2 S3 WB
FS1 FS2 FS3 FS4 FS
(6-8) STS.L from FPSCR: 1 issue cycle
I1 I2 I3 ID FS1 FS2 FS3 FS4
S1 S2 S3 WB
(6-9) FPU load/store instruction FMOV: 1 issue cycle
I1 I2 I3 ID S1 S2 S3 WB
FS1 FS2 FS3 FS4 FS
(6-10) FLDS: 1 issue cycle
I1 I2 I3 ID s1 s2 s3 WB
FS1 FS2 FS3 FS4 FS
(6-11) FSTS: 1 issue cycle
I1 I2 I3 ID s1 s2 s3
FS1 FS2 FS3 FS4 FS

Figure 4.2 Instruction Execution Patterns (7)


(6-12) Single-precision FABS, FNEG/double-precision FABS, FNEG: 1 issue cycle

I1 I2 I3 ID s1 s2 s3
FS1 FS2 FS3 FS4 FS

(6-13) FLDI0, FLDI1: 1 issue cycle

I1 I2 I3 ID s1 s2 s3
FS1 FS2 FS3 FS4 FS

(6-14) Single-precision floating-point computation: 1 issue cycle
FCMP/EQ, FCMP/GT, FADD, FLOAT, FMAC, FMUL, FSUB, FTRC, FRCHG, FSCHG, FPCHG

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS

(6-15) Single-precision FDIV/FSQRT: 1 issue cycle

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS
FEDS (Divider occupied cycle)
FE3 FE4 FE5 FE6 FS

(6-16) Double-precision floating-point computation: 1 issue cycle
FCMP/EQ, FCMP/GT, FADD, FLOAT, FSUB, FTRC, FCNVSD, FCNVDS

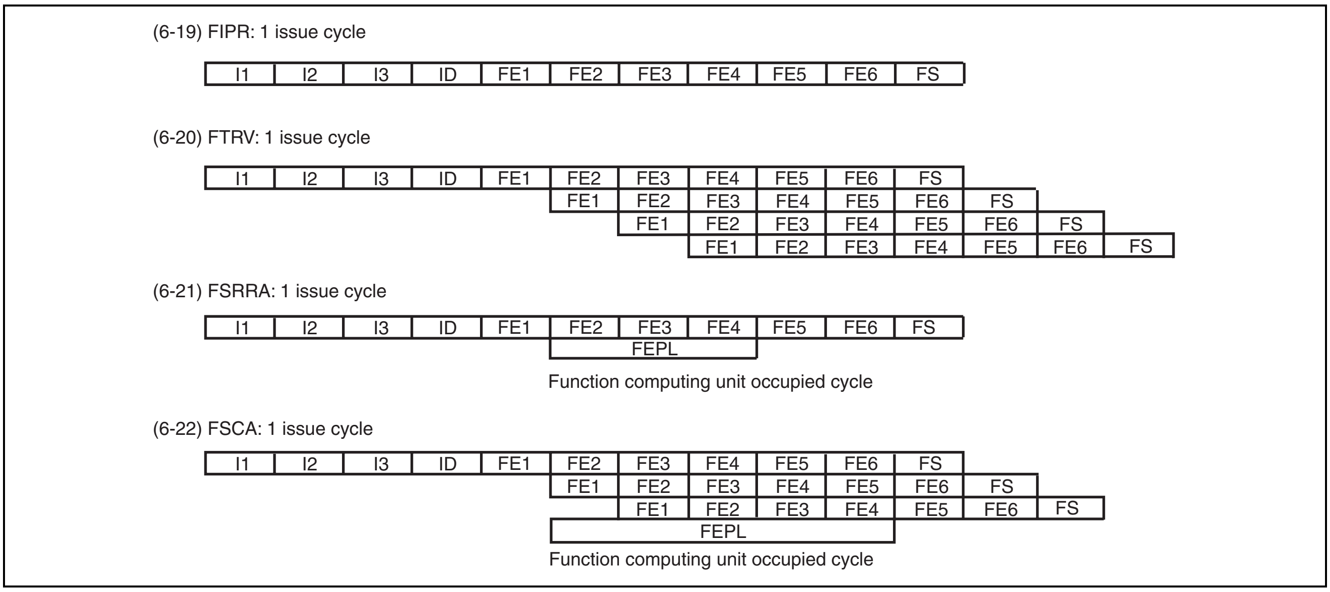
(6-17) Double-precision floating-point computation: 1 issue cycle
FMUL

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS

(6-18) Double-precision FDIV/FSQRT: 1 issue cycle

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS
FEDS (Divider occupied cycle)
FE3 FE4 FE5 FE6 FS
FE3 FE4 FE5 FE6 FS

Figure 4.2 Instruction Execution Patterns (8)


(6-19) FIPR: 1 issue cycle

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS

(6-20) FTRV: 1 issue cycle

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS

(6-21) FSRRA: 1 issue cycle

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS
FEPL
Function computing unit occupied cycle

(6-22) FSCA: 1 issue cycle

I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS
FEPL
Function computing unit occupied cycle

Figure 4.2 Instruction Execution Patterns (9)

4.2 Parallel-Executability

Instructions are categorized into six groups according to the internal function blocks used, as shown in table 4.2. Table 4.3 shows the parallel-executability of pairs of instructions in terms of groups. For example, ADD in the EX group and BRA in the BR group can be executed in parallel.

Table 4.2 Instruction Groups

Instruction Group	Instruction			
EX	ADD	DT	ROTL	SHLR8
	ADDC	EXTS	ROTR	SHLR16
	ADDV	EXTU	SETS	SUB
	AND #imm,R0	MOVT	SETT	SUBC
	AND Rm,Rn	MUL.L	SHAD	SUBV
	CLRMAC	MULS.W	SHAL	SWAP
	CLRS	MULU.W	SHAR	TST #imm,R0
	CLRT	NEG	SHLD	TST Rm,Rn
	CMP	NEGC	SHLL	XOR #imm,R0
	DIV0S	NOT	SHLL2	XOR Rm,Rn
	DIV0U	OR #imm,R0	SHLL8	XTRCT
	DIV1	OR Rm,Rn	SHLL16	
	DMUS.L	ROTCL	SHLR	
	DMULU.L	ROTCR	SHLR2	
MT	MOV #imm,Rn	MOV Rm,Rn	NOP	
BR	BF	BRAF	BT	JSR
	BF/S	BSR	BT/S	RTS
	BRA	BSRF	JMP	
LS	FABS	FMOV.S FR,@adr	MOV.[BWL] @adr,R	STC CR2,Rn
	FNEG	FSTS	MOV.[BWL] R,@adr	STC.L CR2,@-Rn
	FLDI0	LDC Rm,CR1	MOVA	STS SR2,Rn
	FLDI1	LDC.L @Rm+,CR1	MOVCA.L	STS.L SR2,@-Rn
	FLDS	LDS Rm,SR1	MOVUA	STS SR1,Rn
	FMOV @adr,FR	LDS Rm,SR2	OCBI	STS.L SR1,@-Rn
	FMOV FR,@adr	LDS.L @adr,SR2	OCBP	
	FMOV FR,FR	LDS.L @Rm+,SR1	OCBWB	
	FMOV.S @adr,FR	LDS.L @Rm+,SR2	PREF	
FE	FADD	FDIV	FRCHG	FSCA
	FSUB	FIPR	FSCHG	FSRRA
	FCMP (S/D)	FLOAT	FSQRT	FPCHG
	FCNVDS	FMAC	FTRC	
	FCNVSD	FMUL	FTRV	
CO	AND.B #imm,@(R0,GBR)	LDC.L @Rm+,SR	PREFI	TRAPA
	ICBI	LDTLB	RTE	TST.B #imm,@(R0,GBR)
	LDC Rm,DBR	MAC.L	SLEEP*[1]	XOR.B #imm,@(R0,GBR)
	LDC Rm, SGR	MAC.W	STC SR,Rn	
	LDC Rm,SR	MOVCO	STC.L SR,@-Rn	
	LDC.L @Rm+,DBR	MOVLI	SYNCO	
	LDC.L @Rm+,SGR	OR.B #imm,@(R0,GBR)	TAS.B	

Note: *1 Do not use the SLEEP instruction because sleep mode is not available in this MCU.

Legend:

R:	Rm/Rn
@adr:	Address
SR1:	MACH/MACL/PR
SR2:	FPUL/FPSCR
CR1:	GBR/Rp_BANK/SPC/SSR/VBR
CR2:	CR1/DBR/SGR
FR:	FRm/FRn/DRm/DRn/XDm/XDn

The parallel execution of two instructions can be carried out under following conditions.

1. Both addr (preceding instruction) and addr+2 (following instruction) are specified within the minimum page size (1 Kbyte).
2. The execution of these two instructions is supported in table 4.3, Combination of Preceding and Following Instructions.
3. Data used by an instruction of addr does not conflict with data used by a previous instruction.
4. Data used by an instruction of addr+2 does not conflict with data used by a previous instruction.
5. Both instructions are valid.

Table 4.3 Combination of Preceding and Following Instructions

		Preceding Instruction (addr)					
		EX	**MT**	**BR**	**LS**	**FE**	**CO**
Following Instruction (addr+2)	EX	No	Yes	Yes	Yes	Yes	
	MT	Yes	Yes	Yes	Yes	Yes	
	BR	Yes	Yes	No	Yes	Yes	
	LS	Yes	Yes	Yes	No	Yes	
	FE	Yes	Yes	Yes	Yes	No	
	CO						No

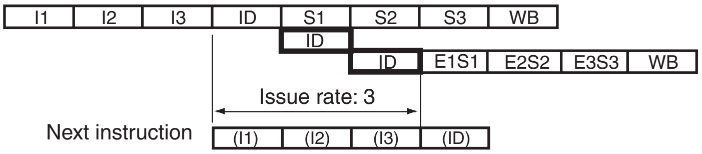
4.3 Issue Rates and Execution Cycles

Instruction execution cycles are summarized in table 4.4. Instruction Group in the table 4.4 corresponds to the category in the table 4.2. Penalty cycles due to a pipeline stall are not considered in the issue rates and execution cycles in this section.

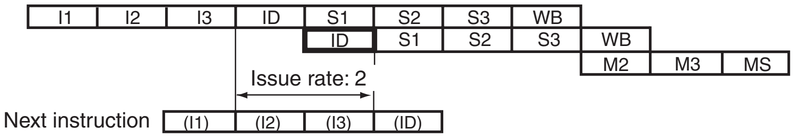
1. Issue Rate

Issue rates indicates the issue period between one instruction and next instruction.

E.g. AND.B instruction



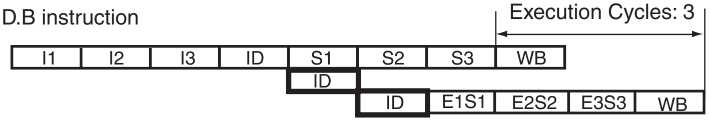
E.g. MAC.W instruction



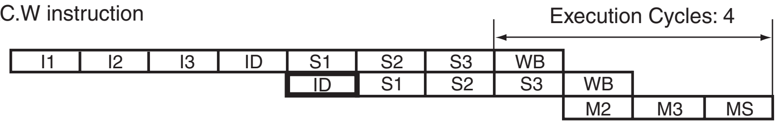
2. Execution Cycles

Execution cycles indicates the cycle counts an instruction occupied the pipeline based on the next rules.

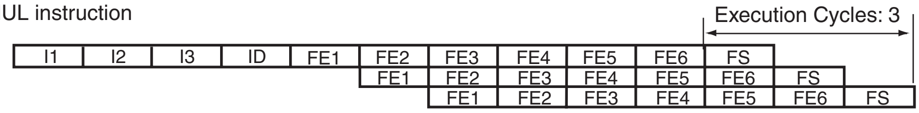
CPU instruction
E.g. AND.B instruction



E.g. MAC.W instruction



FPU instruction
E.g. FMUL instruction


Execution Cycles: 3
I1 I2 I3 ID FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS
FE1 FE2 FE3 FE4 FE5 FE6 FS

E.g. FDIV instruction

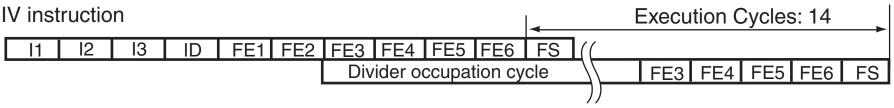


Table 4.4 Issue Rates and Execution Cycles

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Data transfer instructions	1	EXTS.B	Rm,Rn	EX	1	1	2-1
	2	EXTS.W	Rm,Rn	EX	1	1	2-1
	3	EXTU.B	Rm,Rn	EX	1	1	2-1
	4	EXTU.W	Rm,Rn	EX	1	1	2-1
	5	MOV	Rm,Rn	MT	1	1	2-4
	6	MOV	#imm,Rn	MT	1	1	2-3
	7	MOVA	@(disp,PC),R0	LS	1	1	2-2
	8	MOV.W	@(disp,PC),Rn	LS	1	1	3-1
	9	MOV.L	@(disp,PC),Rn	LS	1	1	3-1
	10	MOV.B	@Rm,Rn	LS	1	1	3-1
	11	MOV.W	@Rm,Rn	LS	1	1	3-1
	12	MOV.L	@Rm,Rn	LS	1	1	3-1
	13	MOV.B	@Rm+,Rn	LS	1	1	3-1
	14	MOV.W	@Rm+,Rn	LS	1	1	3-1
	15	MOV.L	@Rm+,Rn	LS	1	1	3-1
	16	MOV.B	@(disp,Rm),R0	LS	1	1	3-1
	17	MOV.W	@(disp,Rm),R0	LS	1	1	3-1
	18	MOV.L	@(disp,Rm),Rn	LS	1	1	3-1
	19	MOV.B	@(R0,Rm),Rn	LS	1	1	3-1
	20	MOV.W	@(R0,Rm),Rn	LS	1	1	3-1
	21	MOV.L	@(R0,Rm),Rn	LS	1	1	3-1
	22	MOV.B	@(disp,GBR),R0	LS	1	1	3-1
	23	MOV.W	@(disp, GBR),R0	LS	1	1	3-1
	24	MOV.L	@(disp, GBR),R0	LS	1	1	3-1
	25	MOV.B	Rm,@Rn	LS	1	1	3-1
	26	MOV.W	Rm,@Rn	LS	1	1	3-1
	27	MOV.L	Rm,@Rn	LS	1	1	3-1
	28	MOV.B	Rm,@-Rn	LS	1	1	3-1
	29	MOV.W	Rm,@-Rn	LS	1	1	3-1
	30	MOV.L	Rm,@-Rn	LS	1	1	3-1
	31	MOV.B	R0,@(disp,Rn)	LS	1	1	3-1
	32	MOV.W	R0,@(disp,Rn)	LS	1	1	3-1
	33	MOV.L	Rm,@(disp,Rn)	LS	1	1	3-1
	34	MOV.B	Rm,@(R0,Rn)	LS	1	1	3-1
	35	MOV.W	Rm,@(R0,Rn)	LS	1	1	3-1
	36	MOV.L	Rm,@(R0,Rn)	LS	1	1	3-1
	37	MOV.B	R0,@(disp,GBR)	LS	1	1	3-1
	38	MOV.W	R0,@(disp,GBR)	LS	1	1	3-1
	39	MOV.L	R0,@(disp,GBR)	LS	1	1	3-1
	40	MOVCA.L	R0,@Rn	LS	1	1	3-4

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Data transfer instructions	41	MOVCO.L	R0,@Rn	CO	1	1	3-9
	42	MOVLI.L	@Rm,R0	CO	1	1	3-8
	43	MOVUA.L	@Rm,R0	LS	2	2	3-10
	44	MOVUA.L	@Rm+,R0	LS	2	2	3-10
	45	MOVT	Rn	EX	1	1	2-1
	46	OCBI	@Rn	LS	1	1	3-4
	47	OCBP	@Rn	LS	1	1	3-4
	48	OCBWB	@Rn	LS	1	1	3-4
	49	PREF	@Rn	LS	1	1	3-4
	50	SWAP.B	Rm,Rn	EX	1	1	2-1
	51	SWAP.W	Rm,Rn	EX	1	1	2-1
	52	XTRCT	Rm,Rn	EX	1	1	2-1
Fixed-point arithmetic instructions	53	ADD	Rm,Rn	EX	1	1	2-1
	54	ADD	#imm,Rn	EX	1	1	2-1
	55	ADDC	Rm,Rn	EX	1	1	2-1
	56	ADDV	Rm,Rn	EX	1	1	2-1
	57	CMP/EQ	#imm,R0	EX	1	1	2-1
	58	CMP/EQ	Rm,Rn	EX	1	1	2-1
	59	CMP/GE	Rm,Rn	EX	1	1	2-1
	60	CMP/GT	Rm,Rn	EX	1	1	2-1
	61	CMP/HI	Rm,Rn	EX	1	1	2-1
	62	CMP/HS	Rm,Rn	EX	1	1	2-1
	63	CMP/PL	Rn	EX	1	1	2-1
	64	CMP/PZ	Rn	EX	1	1	2-1
	65	CMP/STR	Rm,Rn	EX	1	1	2-1
	66	DIV0S	Rm,Rn	EX	1	1	2-1
	67	DIV0U		EX	1	1	2-1
	68	DIV1	Rm,Rn	EX	1	1	2-1
	69	DMULS.L	Rm,Rn	EX	1	2	5-6
	70	DMULU.L	Rm,Rn	EX	1	2	5-6
	71	DT	Rn	EX	1	1	2-1
	72	MAC.L	@Rm+,@Rn+	CO	2	5	5-9
	73	MAC.W	@Rm+,@Rn+	CO	2	4	5-8
	74	MUL.L	Rm,Rn	EX	1	2	5-6
	75	MULS.W	Rm,Rn	EX	1	1	5-5
	76	MULU.W	Rm,Rn	EX	1	1	5-5
	77	NEG	Rm,Rn	EX	1	1	2-1
	78	NEGC	Rm,Rn	EX	1	1	2-1
	79	SUB	Rm,Rn	EX	1	1	2-1
	80	SUBC	Rm,Rn	EX	1	1	2-1
	81	SUBV	Rm,Rn	EX	1	1	2-1

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Logical instructions	82	AND	Rm,Rn	EX	1	1	2-1
	83	AND	#imm,R0	EX	1	1	2-1
	84	AND.B	#imm,@(R0,GBR)	CO	3	3	3-2
	85	NOT	Rm,Rn	EX	1	1	2-1
	86	OR	Rm,Rn	EX	1	1	2-1
	87	OR	#imm,R0	EX	1	1	2-1
	88	OR.B	#imm,@(R0,GBR)	CO	3	3	3-2
	89	TAS.B	@Rn	CO	4	4	3-3
	90	TST	Rm,Rn	EX	1	1	2-1
	91	TST	#imm,R0	EX	1	1	2-1
	92	TST.B	#imm,@(R0,GBR)	CO	3	3	3-2
	93	XOR	Rm,Rn	EX	1	1	2-1
	94	XOR	#imm,R0	EX	1	1	2-1
	95	XOR.B	#imm,@(R0,GBR)	CO	3	3	3-2
Shift instructions	96	ROTL	Rn	EX	1	1	2-1
	97	ROTR	Rn	EX	1	1	2-1
	98	ROTCL	Rn	EX	1	1	2-1
	99	ROTCR	Rn	EX	1	1	2-1
	100	SHAD	Rm,Rn	EX	1	1	2-1
	101	SHAL	Rn	EX	1	1	2-1
	102	SHAR	Rn	EX	1	1	2-1
	103	SHLD	Rm,Rn	EX	1	1	2-1
	104	SHLL	Rn	EX	1	1	2-1
	105	SHLL2	Rn	EX	1	1	2-1
	106	SHLL8	Rn	EX	1	1	2-1
	107	SHLL16	Rn	EX	1	1	2-1
	108	SHLR	Rn	EX	1	1	2-1
	109	SHLR2	Rn	EX	1	1	2-1
	110	SHLR8	Rn	EX	1	1	2-1
	111	SHLR16	Rn	EX	1	1	2-1
Branch instructions	112	BF	disp	BR	1+0 to 2	1	1-1
	113	BF/S	disp	BR	1+0 to 2	1	1-1
	114	BT	disp	BR	1+0 to 2	1	1-1
	115	BT/S	disp	BR	1+0 to 2	1	1-1
	116	BRA	disp	BR	1+0 to 2	1	1-1
	117	BRAF	Rm	BR	1+3	1	1-2
	118	BSR	disp	BR	1+0 to 2	1	1-1
	119	BSRF	Rm	BR	1+3	1	1-2
	120	JMP	@Rn	BR	1+3	1	1-2
	121	JSR	@Rn	BR	1+3	1	1-2
	122	RTS		BR	1+0 to 3	1	1-3

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
System control instructions	123	NOP		MT	1	1	2-3
	124	CLRMAC		EX	1	1	5-7
	125	CLRS		EX	1	1	2-1
	126	CLRT		EX	1	1	2-1
	127	ICBI	@Rn	CO	8+5+3	13	3-6
	128	SETS		EX	1	1	2-1
	129	SETT		EX	1	1	2-1
	130	PREFI		CO	5+5+3	10	3-7
	131	SYNCO	@Rn	CO	Undefined	Undefined	3-4
	132	TRAPA	#imm	CO	8+5+1	13	1-5
	133	RTE		CO	4+1	4	1-4
	134	SLEEP*[1]		CO	Undefined	Undefined	1-6
	135	LDTLB		CO	1	1	3-5
	136	LDC	Rm,DBR	CO	4	4	4-2
	137	LDC	Rm,SGR	CO	4	4	4-2
	138	LDC	Rm,GBR	LS	1	1	4-3
	139	LDC	Rm,Rp_BANK	LS	1	1	4-1
	140	LDC	Rm,SR	CO	4+3	4	4-4
	141	LDC	Rm,SSR	LS	1	1	4-1
	142	LDC	Rm,SPC	LS	1	1	4-1
	143	LDC	Rm,VBR	LS	1	1	4-1
	144	LDC.L	@Rm+,DBR	CO	4	4	4-6
	145	LDC.L	@Rm+,SGR	CO	4	4	4-6
	146	LDC.L	@Rm+,GBR	LS	1	1	4-7
	147	LDC.L	@Rm+,Rp_BANK	LS	1	1	4-5
	148	LDC.L	@Rm+,SR	CO	6+3	4	4-8
	149	LDC.L	@Rm+,SSR	LS	1	1	4-5
	150	LDC.L	@Rm+,SPC	LS	1	1	4-5
	151	LDC.L	@Rm+,VBR	LS	1	1	4-5
	152	LDS	Rm,MACH	LS	1	1	5-1
	153	LDS	Rm,MACL	LS	1	1	5-1
	154	LDS	Rm,PR	LS	1	1	4-13
	155	LDS.L	@Rm+,MACH	LS	1	1	5-2
	156	LDS.L	@Rm+,MACL	LS	1	1	5-2
	157	LDS.L	@Rm+,PR	LS	1	1	4-14
	158	STC	DBR,Rn	LS	1	1	4-9
	159	STC	SGR,Rn	LS	1	1	4-9
	160	STC	GBR,Rn	LS	1	1	4-9
	161	STC	Rp_BANK,Rn	LS	1	1	4-9
	162	STC	SR,Rn	CO	1	1	4-10
	163	STC	SSR,Rn	LS	1	1	4-9

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
System control instructions	164	STC	SPC,Rn	LS	1	1	4-9
	165	STC	VBR,Rn	LS	1	1	4-9
	166	STC.L	DBR,@-Rn	LS	1	1	4-11
	167	STC.L	SGR,@-Rn	LS	1	1	4-11
	168	STC.L	GBR,@-Rn	LS	1	1	4-11
	169	STC.L	Rp_BANK,@-Rn	LS	1	1	4-11
	170	STC.L	SR,@-Rn	CO	1	1	4-12
	171	STC.L	SSR,@-Rn	LS	1	1	4-11
	172	STC.L	SPC,@-Rn	LS	1	1	4-11
	173	STC.L	VBR,@-Rn	LS	1	1	4-11
	174	STS	MACH,Rn	LS	1	1	5-3
	175	STS	MACL,Rn	LS	1	1	5-3
	176	STS	PR,Rn	LS	1	1	4-15
	177	STS.L	MACH,@-Rn	LS	1	1	5-4
	178	STS.L	MACL,@-Rn	LS	1	1	5-4
	179	STS.L	PR,@-Rn	LS	1	1	4-16
Single-precision floating-point instructions	180	FLDI0	FRn	LS	1	1	6-13
	181	FLDI1	FRn	LS	1	1	6-13
	182	FMOV	FRm,FRn	LS	1	1	6-9
	183	FMOV.S	@Rm,FRn	LS	1	1	6-9
	184	FMOV.S	@Rm+,FRn	LS	1	1	6-9
	185	FMOV.S	@(R0,Rm),FRn	LS	1	1	6-9
	186	FMOV.S	FRm,@Rn	LS	1	1	6-9
	187	FMOV.S	FRm,@-Rn	LS	1	1	6-9
	188	FMOV.S	FRm,@(R0,Rn)	LS	1	1	6-9
	189	FLDS	FRm,FPUL	LS	1	1	6-10
	190	FSTS	FPUL,FRn	LS	1	1	6-11
	191	FABS	FRn	LS	1	1	6-12
	192	FADD	FRm,FRn	FE	1	1	6-14
	193	FCMP/EQ	FRm,FRn	FE	1	1	6-14
	194	FCMP/GT	FRm,FRn	FE	1	1	6-14
	195	FDIV	FRm,FRn	FE	1	14	6-15
	196	FLOAT	FPUL,FRn	FE	1	1	6-14
	197	FMAC	FR0,FRm,FRn	FE	1	1	6-14
	198	FMUL	FRm,FRn	FE	1	1	6-14
	199	FNEG	FRn	LS	1	1	6-12
	200	FSQRT	FRn	FE	1	14	6-15
	201	FSUB	FRm,FRn	FE	1	1	6-14
	202	FTRC	FRm,FPUL	FE	1	1	6-14
	203	FMOV	DRm,DRn	LS	1	1	6-9
	204	FMOV	@Rm,DRn	LS	1	1	6-9

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Single-precision floating-point instructions	205	FMOV	@Rm+,DRn	LS	1	1	6-9
	206	FMOV	@(R0,Rm),DRn	LS	1	1	6-9
	207	FMOV	DRm,@Rn	LS	1	1	6-9
	208	FMOV	DRm,@-Rn	LS	1	1	6-9
	209	FMOV	DRm,@(R0,Rn)	LS	1	1	6-9
Double-precision floating-point instructions	210	FABS	DRn	LS	1	1	6-12
	211	FADD	DRm,DRn	FE	1	1	6-16
	212	FCMP/EQ	DRm,DRn	FE	1	1	6-16
	213	FCMP/GT	DRm,DRn	FE	1	1	6-16
	214	FCNVDS	DRm,FPUL	FE	1	1	6-16
	215	FCNVSD	FPUL,DRn	FE	1	1	6-16
	216	FDIV	DRm,DRn	FE	1	30	6-18
	217	FLOAT	FPUL,DRn	FE	1	1	6-16
	218	FMUL	DRm,DRn	FE	1	3	6-17
	219	FNEG	DRn	LS	1	1	6-12
	220	FSQRT	DRn	FE	1	30	6-18
	221	FSUB	DRm,DRn	FE	1	1	6-16
	222	FTRC	DRm,FPUL	FE	1	1	6-16
FPU system control instructions	223	LDS	Rm,FPUL	LS	1	1	6-1
	224	LDS	Rm,FPSCR	LS	1	1	6-5
	225	LDS.L	@Rm+,FPUL	LS	1	1	6-3
	226	LDS.L	@Rm+,FPSCR	LS	1	1	6-7
	227	STS	FPUL,Rn	LS	1	1	6-2
	228	STS	FPSCR,Rn	LS	1	1	6-6
	229	STS.L	FPUL,@-Rn	LS	1	1	6-4
	230	STS.L	FPSCR,@-Rn	LS	1	1	6-8
Graphics acceleration instructions	231	FMOV	DRm,XDn	LS	1	1	6-9
	232	FMOV	XDm,DRn	LS	1	1	6-9
	233	FMOV	XDm,XDn	LS	1	1	6-9
	234	FMOV	@Rm,XDn	LS	1	1	6-9
	235	FMOV	@Rm+,XDn	LS	1	1	6-9
	236	FMOV	@(R0,Rm),XDn	LS	1	1	6-9
	237	FMOV	XDm,@Rn	LS	1	1	6-9
	238	FMOV	XDm,@-Rn	LS	1	1	6-9
	239	FMOV	XDm,@(R0,Rn)	LS	1	1	6-9
	240	FIPR	FVm,FVn	FE	1	1	6-19
	241	FRCHG		FE	1	1	6-14
	242	FSCHG		FE	1	1	6-14
	243	FPCHG		FE	1	1	6-14

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Graphics acceleration instructions	244	FSRRA	FRn	FE	1	1	6-21
	245	FSCA	FPUL,DRn	FE	1	3	6-22
	246	FTRV	XMTRX,FVn	FE	1	4	6-20

Note: *1 Do not use the SLEEP instruction because sleep mode is not available in this MCU.

Section 5 Exception Handling

5.1 Overview

Exception handling processing is handled by a special routine which is executed by a reset, general exception handling, or interrupt. For example, if the executing instruction ends abnormally, appropriate action must be taken in order to return to the original program sequence, or report the abnormality before terminating the processing. The process of generating an exception handling request in response to abnormal termination, and passing control to a user-written exception handling routine, in order to support such functions, is given the generic name of exception handling.

The exception handling in the SH-4A is of three kinds: resets, general exceptions, and interrupts.

5.2 Register Descriptions

Table 5.1 lists the configuration of registers related to exception handling.

Table 5.1 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
TRAPA exception register	TRA	Undefined	H'FF00 0020	32	5-2
Exception event register	EXPEVT	H'0000 0000	H'FF00 0024	32	5-2
Interrupt event register	INTEVT	Undefined	H'FF00 0028	32	5-3
Unsupported function detection exception register	EXPMASK	H'0000 0013	H'FF2F 0004	32	5-3

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

5.2.1 TRAPA Exception Register (TRA)

The TRA register consists of 8-bit immediate data (imm) for the TRAPA instruction. TRA is set automatically by hardware when a TRAPA instruction is executed. TRA can also be modified by software.

TRAPA Exception Register (TRA) <P4 address: location H'FF00 0020>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRACODE								—	—
After Reset:	0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 2	TRACODE	Undefined	R	W	TRAPA Code 8-bit immediate data of TRAPA instruction is set
1, 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

5.2.2 Exception Event Register (EXPEVT)

The EXPEVT register consists of a 12-bit exception code. The exception code set in EXPEVT is that for a reset or general exception event. The exception code is set automatically by hardware when an exception occurs. EXPEVT can also be modified by software.

Exception Event Register (EXPEVT) <P4 address: location H'FF00 0024>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	EXPCODE											
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11 to 0	EXPCODE	All 0	R	W	Exception Code The exception code for a reset or general exception is set. For details, see table 5.2.

5.2.3 Interrupt Event Register (INTEVT)

The INTEVT register consists of a 14-bit exception code. The exception code is set automatically by hardware when an exception occurs. INTEVT can also be modified by software.

Interrupt Event Register (INTEVT) <P4 address: location H'FF00 0028>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	INTCODE													
After Reset:	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13 to 0	INTCODE	Undefined	R	W	Exception Code The exception code for an interrupt is set. For details, see table 5.2.

5.2.4 Unsupported Function Detection Exception Register (EXPMASK)

The EXPMASK register is used to enable or disable the generation of exceptions in response to the use of any of functions 1 and 2 listed below. The functions of 1 and 2 are planned not to be supported in the future SuperH-family products. The exception generation functions of EXPMASK can be used in advance of execution; the detection function then checks for the use of these functions in the software. This will ease the transfer of software to the future SuperH-family products that do not support the respective functions.

1. Handling of an instruction other than the NOP instruction in the delay slot of the RTE instruction.
2. Performance of IC/OC memory-mapped associative write operations.

According to the value of EXPMASK, functions 1 can generate a slot illegal instruction exception, and 2 can generate a data address error exception.

Generation of each exception can be disabled by writing "1" to the corresponding bit in EXPMASK. However, it is recommended that the above functions should not be used when making a program to maintain the compatibility with the future products.

Use the MOV instruction of the CPU to update EXPMASK. After updating the register and then reading the register once, execute either of the following instructions. Executing either instruction guarantees the operation with the updated register value.

- Execute the RTE instruction.
- Execute the ICBI instruction for any address (including non-cacheable area).

Unsupported Function Detection Exception Register (EXPMASK) <P4 address: location H'FF2F 0004>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	MM CAW	—	—	—	RTE DS
After Reset:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1

<After Reset: H'0000 0013>

Bit	Abbreviation	After Reset	R	W	Description
31 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	MMCAW	1	R	W	Memory-Mapped Cache Associative Write 0: Memory-mapped cache associative write is disabled. (A data address error exception will occur.) 1: Memory-mapped cache associative write is enabled.
3, 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
0	RTEDS	1	R	W	RTE Delay Slot 0: An instruction other than the NOP instruction in the delay slot of the RTE instruction is disabled. (An instruction other than the NOP instruction is taken as a slot illegal instruction). 1: An instruction other than the NOP instruction in the delay slot of the RTE instruction is enabled.

5.3 Exception Handling Functions

5.3.1 Exception Handling Flow

In exception handling, the contents of the program counter (PC), status register (SR), and R15 are saved in the saved program counter (SPC), saved status register (SSR), and saved general register15 (SGR), and the CPU starts execution of the appropriate exception handling routine according to the vector address. An exception handling routine is a program written by the user to handle a specific exception. The exception handling routine is terminated and control returned to the original program by executing a return-from-exception instruction (RTE). This instruction restores the PC and SR contents and returns control to the normal processing routine at the point at which the exception occurred. The SGR contents are not written back to R15 with an RTE instruction.

The basic processing flow is as follows. For the meaning of the SR bits, see section 2, Programming Model.

1. The PC, SR, and R15 contents are saved in SPC, SSR, and SGR, respectively.
2. The block bit (BL) in SR is set to "1".
3. The mode bit (MD) in SR is set to "1".
4. The register bank bit (RB) in SR is set to "1".
5. In a reset, the FPU disable bit (FD) in SR is cleared to "0".
6. The exception code is written to bits 13 to 0 of the exception event register (EXPEVT) or interrupt event register (INTEVT).
7. The CPU branches to the determined exception handling vector address, and the exception handling routine begins.

5.3.2 Exception Handling Vector Addresses

The reset vector address is fixed at H'A000 0000. Exception and interrupt vector addresses are determined by adding the offset for the specific event to the vector base address, which is set by software in the vector base register (VBR). In the case of the TLB miss exception, for example, the offset is H'0000 0400, so if H'9C08 0000 is set in VBR, the exception handling vector address will be H'9C08 0400. If a further exception occurs at the exception handling vector address, a duplicate exception will result, and recovery will be difficult; therefore, addresses that are not to be converted (in P1 and P2 areas) should be specified for vector addresses using the MMU.

5.4 Exception Types and Priorities

Table 5.2 shows the types of exceptions, with their relative priorities, vector addresses, and exception/interrupt codes.

Table 5.2 Exceptions

					Exception Transition Direction		
Exception Category	**Execution Mode**	**Exception**	**Priority Level**	**Priority Order**	**Vector Address**	**Offset**	**Exception Code**
Reset	Abort type	Hardware reset	1	1	H'A000 0000	—	H'000
		H-UDI reset	1	1	H'A000 0000	—	H'000
		Instruction TLB multiple-hit exception	1	3	H'A000 0000	—	H'000
		Data TLB multiple-hit exception	1	4	H'A000 0000	—	H'000
General exception*[2]	Re-execution type	User break before instruction execution*[1]	2	0	(VBR/DBR)	H'100/—	H'1E0
		Instruction address error	2	1	(VBR)	H'100	H'0E0
		Instruction TLB miss exception	2	2	(VBR)	H'400	H'040
		Instruction TLB protection violation exception	2	3	(VBR)	H'100	H'0A0
		General illegal instruction exception	2	4	(VBR)	H'100	H'180
		Slot illegal instruction exception	2	4	(VBR)	H'100	H'1A0
		General FPU disable exception	2	4	(VBR)	H'100	H'800
		Slot FPU disable exception	2	4	(VBR)	H'100	H'820
		Data address error (read)	2	5	(VBR)	H'100	H'0E0
		Data address error (write)	2	5	(VBR)	H'100	H'100
		Data TLB miss exception (read)	2	6	(VBR)	H'400	H'040
		Data TLB miss exception (write)	2	6	(VBR)	H'400	H'060
		Data TLB protection violation exception (read)	2	7	(VBR)	H'100	H'0A0
		Data TLB protection violation exception (write)	2	7	(VBR)	H'100	H'0C0
		FPU exception	2	8	(VBR)	H'100	H'120
		Initial page write exception	2	9	(VBR)	H'100	H'080
	Completion type	Unconditional trap (TRAPA)	2	4	(VBR)	H'100	H'160
		User break after instruction execution*[1]	2	10	(VBR/DBR)	H'100/—	H'1E0
Interrupt	Completion type	Nonmaskable interrupt	3	—	(VBR)	H'600	H'1C0
		General interrupt request	4	—	(VBR)	H'600	—

Notes: *1 When UBDE in CBCR = "1", PC = DBR. In other cases, PC = VBR + H'100.

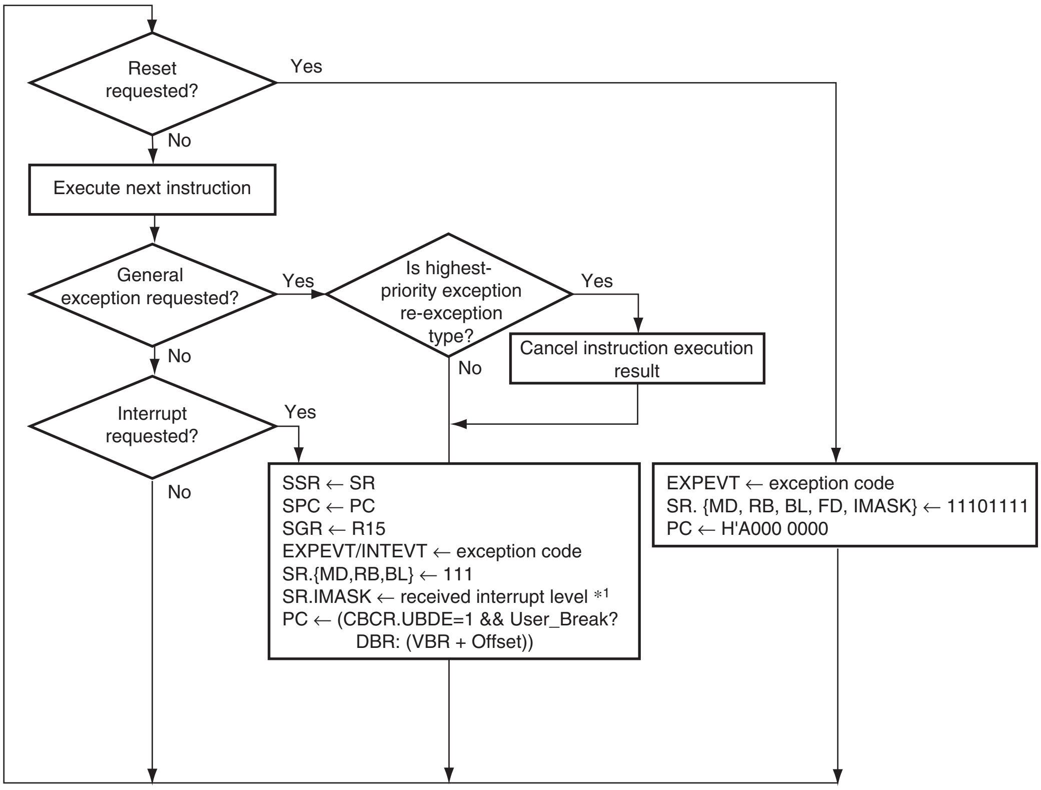
*2 A reset occurs when a general exception other than a user break is generated while SR.BL = "1".

- Priority is first assigned by priority level, then by priority order within each level (the lowest number represents the highest priority).
- Control passes to H'A000 0000 in a reset, and to [VBR + offset] in other cases.
- Stored in EXPEVT for a reset or general exception, and in INTEVT for an interrupt.

5.5 Exception Flow

5.5.1 Exception Flow

Figure 5.1 shows an outline flowchart of the basic operations in instruction execution and exception handling. For the sake of clarity, the following description assumes that instructions are executed sequentially, one by one. Figure 5.1 shows the relative priority order of the different kinds of exceptions (reset, general exception, and interrupt). Register settings in the event of an exception are shown only for SSR, SPC, SGR, EXPEVT/INTEVT, SR, and PC. However, other registers may be set automatically by hardware, depending on the exception. For details, see section 5.6, Description of Exceptions. Also, see section 5.6.4, Priority Order with Multiple Exceptions, for exception handling during execution of a delayed branch instruction and a delay slot instruction, or in the case of instructions in which two data accesses are performed.


Reset requested?
Yes
No
Execute next instruction
General exception requested?
Yes
Is highest-priority exception re-exception type?
Yes
No
Cancel instruction execution result
No
Interrupt requested?
Yes
No
SSR ← SR
SPC ← PC
SGR ← R15
EXPEVT/INTEVT ← exception code
SR.{MD,RB,BL} ← 111
SR.IMASK ← received interrupt level *1
PC ← (CBCR.UBDE=1 && User_Break? DBR: (VBR + Offset))
EXPEVT ← exception code
SR. {MD, RB, BL, FD, IMASK} ← 11101111
PC ← H'A000 0000

Note: *1 When the exception of the highest priority is an interrupt. Whether IMASK is updated or not can be set by software.

Figure 5.1 Instruction Execution and Exception Handling

5.5.2 Exception Source Acceptance

A priority ranking is provided for all exceptions for use in determining which of two or more simultaneously generated exceptions should be accepted. Five of the general exceptions—general illegal instruction exception, slot illegal instruction exception, general FPU disable exception, slot FPU disable exception, and unconditional trap exception—are detected in the process of instruction decoding, and do not occur simultaneously in the instruction pipeline. These exceptions therefore all have the same priority. General exceptions are detected in the order of instruction execution. However, exception handling is performed in the order of instruction flow (program order). Thus, an exception for an earlier instruction is accepted before that for a later instruction. An example of the order of acceptance for general exceptions is shown in figure 5.2.

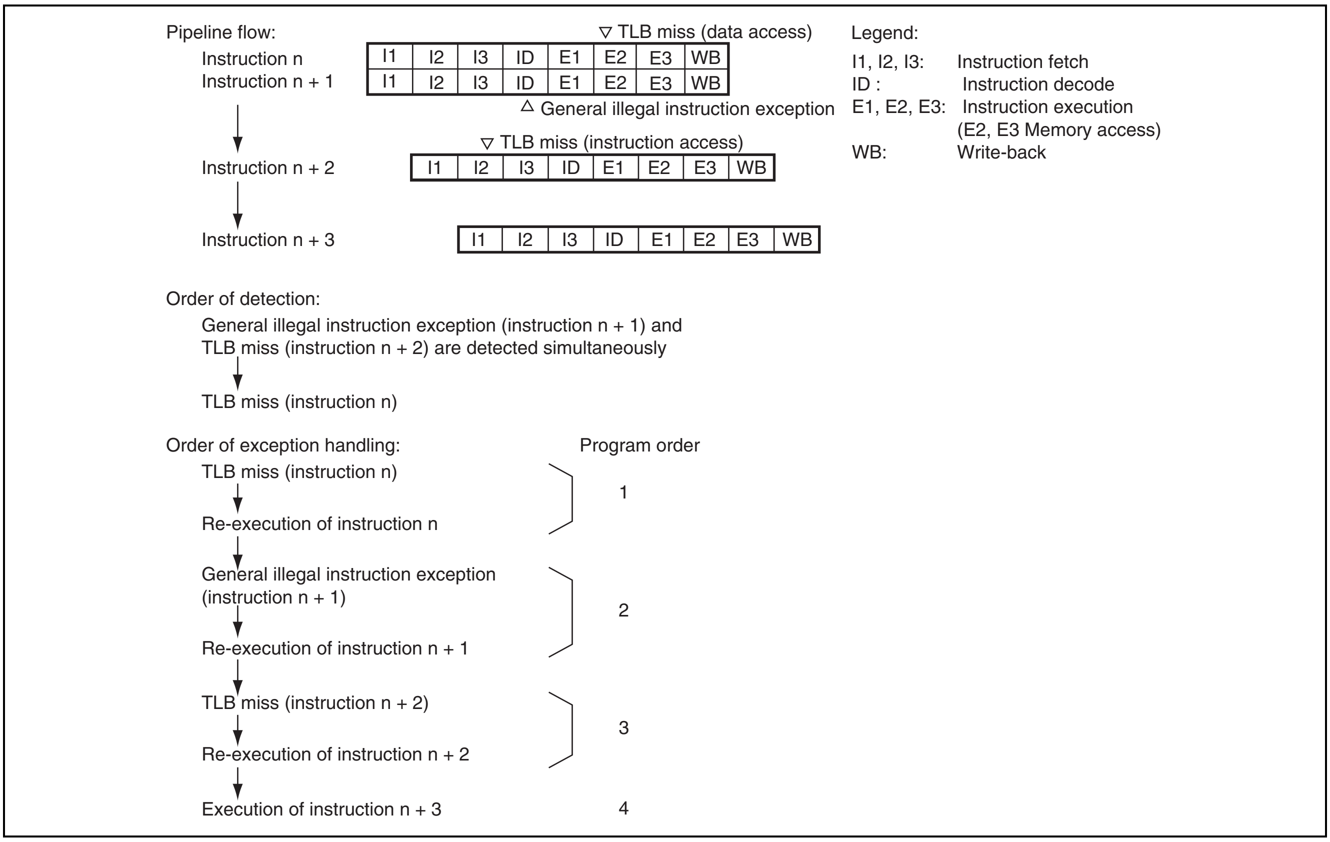

Pipeline flow:
▽ TLB miss (data access)
Instruction n
I1 I2 I3 ID E1 E2 E3 WB
Instruction n + 1
I1 I2 I3 ID E1 E2 E3 WB
△ General illegal instruction exception
▽ TLB miss (instruction access)
Instruction n + 2
I1 I2 I3 ID E1 E2 E3 WB
Instruction n + 3
I1 I2 I3 ID E1 E2 E3 WB
Legend:
I1, I2, I3: Instruction fetch
ID : Instruction decode
E1, E2, E3: Instruction execution
(E2, E3 Memory access)
WB: Write-back
Order of detection:
General illegal instruction exception (instruction n + 1) and
TLB miss (instruction n + 2) are detected simultaneously
TLB miss (instruction n)
Order of exception handling:
Program order
TLB miss (instruction n)
Re-execution of instruction n
1
General illegal instruction exception
(instruction n + 1)
Re-execution of instruction n + 1
2
TLB miss (instruction n + 2)
Re-execution of instruction n + 2
3
Execution of instruction n + 3
4

Figure 5.2 Example of General Exception Acceptance Order

5.5.3 Exception Requests and BL Bit

When the BL bit in SR is "0", general exceptions and interrupts are accepted.

When the BL bit in SR is "1" and an general exception other than a user break is generated, the CPU's internal registers and the registers of the other modules are set to their states following a hardware reset, and the CPU branches to the same address as in a reset (H'A000 0000). For the operation in the event of a user break, see section 35, User Break Controller (UBC). If an ordinary interrupt occurs, the interrupt request is held pending and is accepted after the BL bit has been cleared to "0" by software. If a nonmaskable interrupt (NMI) occurs, it can be held pending or accepted according to the setting made by software. For further details, refer to the hardware manual of the product.

Thus, normally, SPC and SSR are saved and then the BL bit in SR is cleared to "0", to enable multiple exception state acceptance.

5.5.4 Return from Exception Handling

The RTE instruction is used to return from exception handling. When the RTE instruction is executed, the SPC contents are restored to PC and the SSR contents to SR, and the CPU returns from the exception handling routine by branching to the SPC address. If SPC and SSR were saved to external memory, set the BL bit in SR to "1" before restoring the SPC and SSR contents and issuing the RTE instruction.

5.6 Description of Exceptions

The various exception handling operations explained here are exception sources, transition address on the occurrence of exception, and processor operation when a transition is made.

5.6.1 Resets

(1) Hardware Reset

- Condition:
  Hardware reset request
- Operations:
  Exception code H'000 is set in EXPEVT, initialization of the CPU and on-chip peripheral module is carried out, and then a branch is made to the reset vector (H'A000 0000). For details, see the register descriptions in the relevant sections. A hardware reset should be executed when power is supplied.

(2) H-UDI Reset

- Source: SDIR.TI[7:4] = B'0110 (negation) or B'0111 (assertion)
- Transition address: H'A000 0000
- Transition operations:
  Exception code H'000 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A000 0000.

  CPU and on-chip peripheral module initialization is performed. For details, see the register descriptions in the relevant sections.

(3) Instruction TLB Multiple Hit Exception

- Source: Multiple ITLB address matches
- Transition address: H'A000 0000
- Transition operations:
  Exception code H'000 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A000 0000.

  CPU and on-chip peripheral module initialization is performed. For details, see the register descriptions in the relevant sections.

(4) Data TLB Multiple-Hit Exception

- Source: Multiple UTLB address matches
- Transition address: H'A000 0000
- Transition operations:
  Exception code H'000 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A000 0000.

  CPU and on-chip peripheral module initialization. For details, see the register descriptions in the relevant sections.

5.6.2 General Exceptions

(1) Data TLB Miss Exception

- Source: Address mismatch in UTLB address comparison
- Transition address: VBR + H'0000 0400
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'040 (for a read access) or H'060 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0400.

  To speed up TLB miss processing, the offset is separate from that of other exceptions.

```
Data_TLB_miss_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access ? H'0000 0040 : H'0000 0060;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0400;
}
```

(2) Instruction TLB Miss Exception

- Source: Address mismatch in ITLB address comparison
- Transition address: VBR + H'0000 0400
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'40 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0400.

  To speed up TLB miss processing, the offset is separate from that of other exceptions.

```
ITLB_miss_exception()
{
   TEA = EXCEPTION_ADDRESS;
   PTEH.VPN = PAGE_NUMBER;
   SPC = PC;
   SSR = SR;
   SGR = R15;
   EXPEVT = H'0000 0040;
   SR.MD = 1;
   SR.RB = 1;
   SR.BL = 1;
   PC = VBR + H'0000 0400;
}
```

(3) Initial Page Write Exception

- Source: TLB is hit in a store access, but dirty bit D = 0
- Transition address: VBR + H'0000 0100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'080 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100.

```
Initial_write_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0080;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(4) Data TLB Protection Violation Exception

- Source: The access does not accord with the UTLB protection information (PR bits or EPR bits) shown in table 5.3 and table 5.4.

Table 5.3 UTLB Protection Information (TLB Compatible Mode)

PR	Privileged Mode	User Mode
00	Only read access possible	Access not possible
01	Read/write access possible	Access not possible
10	Only read access possible	Only read access possible
11	Read/write access possible	Read/write access possible

Table 5.4 UTLB Protection Information (TLB Extended Mode)

EPR [5]	Read Permission in Privileged Mode
0	Read access possible
1	Read access not possible

EPR [4]	Write Permission in Privileged Mode
0	Write access possible
1	Write access not possible

EPR [2]	Read Permission in User Mode
0	Read access possible
1	Read access not possible

EPR [1]	Write Permission in User Mode
0	Write access possible
1	Write access not possible

- Transition address: VBR + H'0000 0100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0A0 (for a read access) or H'0C0 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100.

```
Data_TLB_protection_violation_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access ? H'0000 00A0 : H'0000 00C0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(5) Instruction TLB Protection Violation Exception

- Source: The access does not accord with the ITLB protection information (PR bits or EPR bits) shown in table 5.5 and table 5.6.

Table 5.5 ITLB Protection Information (TLB Compatible Mode)

PR	Privileged Mode	User Mode
0	Access possible	Access not possible
1	Access possible	Access possible

Table 5.6 ITLB Protection Information (TLB Extended Mode)

EPR [5], EPR [3]	Execution Permission in Privileged Mode
11, 01	Possible
10	Instruction fetch not possible, Rn access of ICBI instruction possible
00	Not possible

EPR [2], EPR [0]	Execution Permission in User Mode
11, 01	Possible
10	Instruction fetch not possible, Rn access of ICBI instruction possible
00	Not possible

- Transition address: VBR + H'0000 0100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0A0 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100.

```
ITLB_protection_violation_exception()
{
   TEA = EXCEPTION_ADDRESS;
   PTEH.VPN = PAGE_NUMBER;
   SPC = PC;
   SSR = SR;
   SGR = R15;
   EXPEVT = H'0000 00A0;
   SR.MD = 1;
   SR.RB = 1;
   SR.BL = 1;
   PC = VBR + H'0000 0100;
}
```

(6) Data Address Error

- Sources:
  - Word data access from other than a word boundary (2n +1)
  - Longword data access from other than a longword data boundary (4n +1, 4n + 2, or 4n +3)
  - Quadword data access from other than a quadword data boundary (8n +1, 8n + 2, 8n +3, 8n + 4, 8n + 5, 8n + 6, or 8n + 7)
  - Access to area H'8000 0000 to H'FFFF FFFF in user mode

    Areas H'E000 0000 to H'E3FF FFFF and H'E500 0000 to H'E5FF FFFF can be accessed in user mode. For details, see section 7, Memory Management Unit (MMU) and section 9, IL Memory/OL Memory.
  - The MMCAW bit in EXPMASK is "0", and the IC/OC memory mapped associative write is performed.
- Transition address: VBR + H'0000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0E0 (for a read access) or H'100 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100. For details, see section 7, Memory Management Unit (MMU).

```
Data_address_error()
{
   TEA = EXCEPTION_ADDRESS;
   PTEH.VPN = PAGE_NUMBER;
   SPC = PC;
   SSR = SR;
   SGR = R15;
   EXPEVT = read_access? H'0000 00E0: H'0000 0100;
   SR.MD = 1;
   SR.RB = 1;
   SR.BL = 1;
   PC = VBR + H'0000 0100;
}
```

(7) Instruction Address Error

- Sources:
  - — Instruction fetch from other than a word boundary (2n +1)
  - — Instruction fetch from area H'8000 0000 to H'FFFF FFFF in user mode

    Area H'E500 0000 to H'E5FF FFFF can be accessed in user mode. For details, see section 9, IL Memory/OL Memory.
- Transition address: VBR + H'0000 0100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in the SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0E0 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100. For details, see section 7, Memory Management Unit (MMU).

```
Instruction_address_error()
{
   TEA = EXCEPTION_ADDRESS;
   PTEH.VPN = PAGE_NUMBER;
   SPC = PC;
   SSR = SR;
   SGR = R15;
   EXPEVT = H'0000 00E0;
   SR.MD = 1;
   SR.RB = 1;
   SR.BL = 1;
   PC = VBR + H'0000 0100;
}
```

(8) Unconditional Trap

- Source: Execution of TRAPA instruction
- Transition address: VBR + H'0000 0100
- Transition operations:

  As this is a processing-completion-type exception, the PC contents for the instruction following the TRAPA instruction are saved in SPC. The value of SR and R15 when the TRAPA instruction is executed are saved in SSR and SGR. The 8-bit immediate value in the TRAPA instruction is multiplied by 4, and the result is set in TRA [9:0]. Exception code H'160 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100.

```
TRAPA_exception()
{
    SPC = PC + 2;
    SSR = SR;
    SGR = R15;
    TRA = imm << 2;
    EXPEVT = H'0000 0160;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(9) General Illegal Instruction Exception

- Sources:
  - Decoding of an undefined instruction not in a delay slot
    Delayed branch instructions: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT/S, BF/S
    Undefined instruction: H'FFFD
  - Decoding in user mode of a privileged instruction not in a delay slot
    Privileged instructions: LDC, STC, RTE, LDTLB, but excluding LDC/STC instructions that access GBR
- Transition address: VBR + H'0000 0100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'180 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100. Operation is not guaranteed if an undefined code other than H'FFFD is decoded.

```
General_illegal_instruction_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0180;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(10) Slot Illegal Instruction Exception

- Sources:
  - Decoding of an undefined instruction in a delay slot
    Delayed branch instructions: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT/S, BF/S
    Undefined instruction: H'FFFD
  - Decoding of an instruction that modifies PC in a delay slot
    Instructions that modify PC: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT, BF, BT/S, BF/S, TRAPA, LDC Rm,SR, LDC.L @Rm+,SR, ICBI, PREFI
  - Decoding in user mode of a privileged instruction in a delay slot
    Privileged instructions: LDC, STC, RTE, LDTLB, but excluding LDC/STC instructions that access GBR
  - Decoding of a PC-relative MOV instruction or MOVA instruction in a delay slot
  - The RTEDS bit in EXPMASK is "0", and an instruction other than the NOP instruction in the delay slot is executed.
- Transition address: VBR + H'000 0100
- Transition operations:

  The PC contents for the preceding delayed branch instruction are saved in SPC. The SR and R15 contents when this exception occurred are saved in SSR and SGR.

  Exception code H'1A0 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100. Operation is not guaranteed if an undefined code other than H'FFFD is decoded.

```
Slot_illegal_instruction_exception()
{
   SPC = PC - 2;
   SSR = SR;
   SGR = R15;
   EXPEVT = H'0000 01A0;
   SR.MD = 1;
   SR.RB = 1;
   SR.BL = 1;
   PC = VBR + H'0000 0100;
}
```

(11) General FPU Disable Exception

- Source: Decoding of an FPU instruction*[1] not in a delay slot with SR.FD = "1"
- Transition address: VBR + H'0000 0100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'800 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100.

```
General_fpu_disable_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0800;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

Note: *1 FPU instructions are instructions in which the first 4 bits of the instruction code are F (but excluding undefined instruction H'FFFD), and the LDS, STS, LDS.L, and STS.L instructions corresponding to FPUL and FPSCR.

(12) Slot FPU Disable Exception

- Source: Decoding of an FPU instruction in a delay slot with SR.FD = "1"
- Transition address: VBR + H'0000 0100
- Transition operations:

  The PC contents for the preceding delayed branch instruction are saved in SPC. The SR and R15 contents when this exception occurred are saved in SSR and SGR.

  Exception code H'820 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100.

```
Slot_fpu_disable_exception()
{
    SPC = PC - 2;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0820;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(13) Pre-Execution User Break/Post-Execution User Break

- Source: Fulfilling of a break condition set in the user break controller
- Transition address: VBR + H'0000 0100, or DBR
- Transition operations:

  In the case of a post-execution break, the PC contents for the instruction following the instruction at which the breakpoint is set are set in SPC. In the case of a pre-execution break, the PC contents for the instruction at which the breakpoint is set are set in SPC.

  The SR and R15 contents when the break occurred are saved in SSR and SGR. Exception code H'1E0 is set in EXPEVT.

  The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100. It is also possible to branch to PC = DBR.

  For details of PC, etc., when a data break is set, see section 35, User Break Controller (UBC).

```
User_break_exception()
{
    SPC = (pre_execution break? PC : PC + 2);
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 01E0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = (BRCR.UBDE==1 ? DBR : VBR + H′0000 0100);
}
```

(14) FPU Exception

- Source: Exception due to execution of a floating-point operation
- Transition address: VBR + H'0000 0100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR . The R15 contents at this time are saved in SGR. Exception code H'120 is set in EXPEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0100.

```
FPU_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0120;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

5.6.3 Interrupts

(1) NMI (Nonmaskable Interrupt)

- Source: NMI pin edge detection
- Transition address: VBR + H'0000 0600
- Transition operations:

  The PC and SR contents for the instruction immediately after this exception is accepted are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'1C0 is set in INTEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to PC = VBR + H'0600. When the BL bit in SR is "0", this interrupt is not masked by the interrupt mask bits in SR, and is accepted at the highest priority level. When the BL bit in SR is "1", a software setting can specify whether this interrupt is to be masked or accepted. For details, see section 15, Interrupt Controller (INTC).

```
NMI()
{
   SPC = PC;
   SSR = SR;
   SGR = R15;
   INTEVT = H'0000 01C0;
   SR.MD = 1;
   SR.RB = 1;
   SR.BL = 1;
   If (cond) SR.IMASK = B'1111;
   PC = VBR + H'0000 0600;
}
```

(2) General Interrupt Request

- Source: The interrupt mask level bits setting in SR is smaller than the interrupt level of interrupt request, and the BL bit in SR is "0" (accepted at instruction boundary).
- Transition address: VBR + H'0000 0600
- Transition operations:

  The PC contents immediately after the instruction at which the interrupt is accepted are set in SPC. The SR and R15 contents at the time of acceptance are set in SSR and SGR.

  The code corresponding to the each interrupt source is set in INTEVT. The BL, MD, and RB bits are set to "1" in SR, and a branch is made to VBR + H'0600. For details, see section 15, Interrupt Controller (INTC).

```
Module_interruption()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    INTEVT = H'0000 0400 ~ H'0000 3FE0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    if (cond) SR.IMASK = level_of accepted_interrupt ();
    PC = VBR + H'0000 0600;
}
```

5.6.4 Priority Order with Multiple Exceptions

With some instructions, such as instructions that make two accesses to memory, and the indivisible pair comprising a delayed branch instruction and delay slot instruction, multiple exceptions occur. Care is required in these cases, as the exception priority order differs from the normal order.

(1) Instructions that Make Two Accesses to Memory

With MAC instructions, memory-to-memory arithmetic/logic instructions, TAS instructions, and MOVUA instructions, two data transfers are performed by a single instruction, and an exception will be detected for each of these data transfers. In these cases, therefore, the following order is used to determine priority.

1. Data address error in first data transfer
2. TLB miss in first data transfer
3. TLB protection violation in first data transfer
4. Initial page write exception in first data transfer
5. Data address error in second data transfer
6. TLB miss in second data transfer
7. TLB protection violation in second data transfer
8. Initial page write exception in second data transfer

(2) Indivisible Delayed Branch Instruction and Delay Slot Instruction

As a delayed branch instruction and its associated delay slot instruction are indivisible, they are treated as a single instruction. Consequently, the priority order for exceptions that occur in these instructions differs from the usual priority order. The priority order shown below is for the case where the delay slot instruction has only one data transfer.

1. A check is performed for the interrupt type and re-execution type exceptions of priority levels 1 and 2 in the delayed branch instruction.
2. A check is performed for the interrupt type and re-execution type exceptions of priority levels 1 and 2 in the delay slot instruction.
3. A check is performed for the completion type exception of priority level 2 in the delayed branch instruction.
4. A check is performed for the completion type exception of priority level 2 in the delay slot instruction.
5. A check is performed for priority level 3 in the delayed branch instruction and priority level 3 in the delay slot instruction. (There is no priority ranking between these two.)
6. A check is performed for priority level 4 in the delayed branch instruction and priority level 4 in the delay slot instruction. (There is no priority ranking between these two.)

If the delay slot instruction has a second data transfer, two checks are performed in step 2, as in the above case (Instructions that make two accesses to memory).

If the accepted exception (the highest-priority exception) is a delay slot instruction re-execution type exception, the branch instruction PR register write operation (PC → PR operation performed in a BSR, BSRF, or JSR instruction) is not disabled. Note that in this case, the contents of PR register are not guaranteed.

5.7 Usage Notes

(1) Return from Exception Handling

A. Check the BL bit in SR with software. If SPC and SSR have been saved to memory, set the BL bit in SR to "1" before restoring them.

B. Issue an RTE instruction. When RTE is executed, the SPC contents are saved in PC, the SSR contents are saved in SR, and branch is made to the SPC address to return from the exception handling routine.

(2) If a General Exception or Interrupt Occurs When BL Bit in SR = 1

A. General exception

When an exception other than a user break occurs, the PC value for the instruction at which the exception occurs is copied to SPC and a hardware reset is generated. The value in EXPEVT at this time is H'0000 0000; the SSR contents are undefined.

B. Interrupt

If an ordinary interrupt occurs, the interrupt request is held pending and is accepted after the BL bit in SR has been cleared to "0" by software. If a nonmaskable interrupt (NMI) occurs, it can be held pending or accepted according to the setting made by software.

(3) SPC when an Exception Occurs

A. Re-execution type general exception

The PC value for the instruction at which the exception occurred is set in SPC, and the instruction is re-executed after returning from the exception handling routine. If an exception occurs in a delay slot instruction, however, the PC value for the delayed branch instruction is saved in SPC regardless of whether or not the preceding delay slot instruction condition is satisfied.

B. Completion type general exception or interrupt

The PC value for the instruction following that at which the exception occurred is set in SPC. If an exception occurs in a branch instruction with delay slot, however, the PC value for the branch destination is saved in SPC.

(4) RTE Instruction Delay Slot

A. The instruction in the delay slot of the RTE instruction is executed only after the value saved in SSR has been restored to SR. The acceptance of the exception related to the instruction access is determined depending on SR before restoring, while the acceptance of other exceptions is determined depending on the processing mode by SR after restoring or the BL bit. The completion type exception is accepted before branching to the destination of RTE instruction. However, if the re-execution type exception is occurred, the operation cannot be guaranteed.

B. The user break is not accepted by the instruction in the delay slot of the RTE instruction.

(5) Changing the SR Register Value and Accepting Exception

A. When the MD or BL bit in the SR register is changed by the LDC instruction, the acceptance of the exception is determined by the changed SR value, starting from the next instruction.*[1] In the completion type exception, an exception is accepted after the next instruction has been executed. However, an interrupt of completion type exception is accepted before the next instruction is executed.

Note: *1 When the LDC instruction for SR is executed, following instructions are fetched again and the instruction fetch exception is evaluated again by the changed SR.

Section 6 Floating-Point Unit (FPU)

6.1 Overview

The FPU has the following features.

- Conforms to IEEE754 standard
- 32 single-precision floating-point registers (can also be referenced as 16 double-precision registers)
- Two rounding modes: Round to Nearest and Round to Zero
- Two denormalization modes: Flush to Zero and Treat Denormalized Number
- Six exception sources: FPU Error, Invalid Operation, Divide By Zero, Overflow, Underflow, and Inexact
- Comprehensive instructions: Single-precision, double-precision, graphics support, and system control
- In the SH-4A, the following three instructions are added on to the instruction set of the SH-4
  FSRRA, FSCA, and FPCHG

When the FD bit in SR is set to "1", the FPU cannot be used, and an attempt to execute an FPU instruction will cause an FPU disable exception (general FPU disable exception or slot FPU disable exception).

6.2 Data Formats

6.2.1 Floating-Point Format

A floating-point number consists of the following three fields:

- Sign bit (s)
- Exponent field (e)
- Fraction field (f)

The SH-4A can handle single-precision and double-precision floating-point numbers, using the formats shown in figures 6.1 and 6.2.

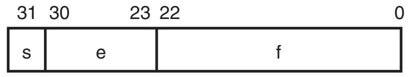


Figure 6.1 Format of Single-Precision Floating-Point Number

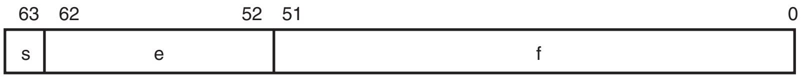


Figure 6.2 Format of Double-Precision Floating-Point Number

The exponent is expressed in biased form, as follows:

e = E + bias

The range of unbiased exponent E is E_{min} – 1 to E_{max} + 1. The two values E_{min} – 1 and E_{max} + 1 are distinguished as follows. E_{min} – 1 indicates zero (both positive and negative sign) and a denormalized number, and E_{max} + 1 indicates positive or negative infinity or a non-number (NaN). Table 6.1 shows floating-point formats and parameters.

Table 6.1 Floating-Point Number Formats and Parameters

Parameter	Single-Precision	Double-Precision
Total bit width	32 bits	64 bits
Sign bit	1 bit	1 bit
Exponent field	8 bits	11 bits
Fraction field	23 bits	52 bits
Precision	24 bits	53 bits
Bias	+127	+1023
E_{max}	+127	+1023
E_{min}	–126	–1022

Floating-point number value v is determined as follows:

If $E = E_{max} + 1$ and $f \neq 0$, v is a non-number (NaN) irrespective of sign s
If $E = E_{max} + 1$ and $f = 0$, $v = (-1)^s$ (infinity) [positive or negative infinity]
If $E_{min} \leq E \leq E_{max}$, $v = (-1)^s 2^E$ (1.f) [normalized number]
If $E = E_{min} - 1$ and $f \neq 0$, $v = (-1)^s 2^{Emin}$ (0.f) [denormalized number]
If $E = E_{min} - 1$ and $f = 0$, $v = (-1)^s 0$ [positive or negative zero]

Table 6.2 shows the ranges of the various numbers in hexadecimal notation. For the signaling non-number and quiet non-number, see section 6.2.2, Non-Numbers (NaN). For the denormalized number, see section 6.2.3, Denormalized Numbers.

Table 6.2 Floating-Point Ranges

Type	Single-Precision	Double-Precision
Signaling non-number	H'7FFF FFFF to H'7FC0 0000	H'7FFF FFFF FFFF FFFF to H'7FF8 0000 0000 0000
Quiet non-number	H'7FBF FFFF to H'7F80 0001	H'7FF7 FFFF FFFF FFFF to H'7FF0 0000 0000 0001
Positive infinity	H'7F80 0000	H'7FF0 0000 0000 0000
Positive normalized number	H'7F7F FFFF to H'0080 0000	H'7FEF FFFF FFFF FFFF to H'0010 0000 0000 0000
Positive denormalized number	H'007F FFFF to H'0000 0001	H'000F FFFF FFFF FFFF to H'0000 0000 0000 0001
Positive zero	H'0000 0000	H'0000 0000 0000 0000
Negative zero	H'8000 0000	H'8000 0000 0000 0000
Negative denormalized number	H'8000 0001 to H'807F FFFF	H'8000 0000 0000 0001 to H'800F FFFF FFFF FFFF
Negative normalized number	H'8080 0000 to H'FF7F FFFF	H'8010 0000 0000 0000 to H'FFEF FFFF FFFF FFFF
Negative infinity	H'FF80 0000	H'FFF0 0000 0000 0000
Quiet non-number	H'FF80 0001 to H'FFBF FFFF	H'FFF0 0000 0000 0001 to H'FFF7 FFFF FFFF FFFF
Signaling non-number	H'FFC0 0000 to H'FFFF FFFF	H'FFF8 0000 0000 0000 to H'FFFF FFFF FFFF FFFF

6.2.2 Non-Numbers (NaN)

Figure 6.3 shows the bit pattern of a non-number (NaN). A value is NaN in the following case:

- Sign bit: Don't care
- Exponent field: All bits are "1"
- Fraction field: At least one bit is "1"

The NaN is a signaling NaN (sNaN) if the MSB of the fraction field is "1", and a quiet NaN (qNaN) if the MSB is "0".

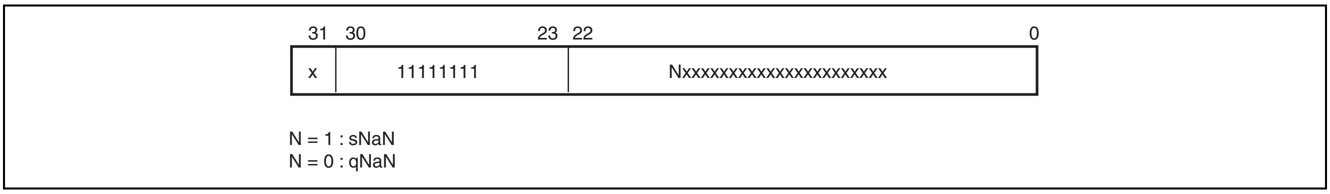


Figure 6.3 Single-Precision NaN Bit Pattern

An sNaN is assumed to be the input data in an operation, except the transfer instructions between registers, FABS, and FNEG, that generates a floating-point value.

- When the EN.V bit in FPSCR is "0", the operation result (output) is a qNaN.
- When the EN.V bit in FPSCR is "1", an invalid operation exception will be generated. In this case, the contents of the operation destination register are unchanged.

Following three instructions are used as transfer instructions between registers.

- FMOV FRm,FRn
- FLDS FRm,FPUL
- FSTS FPUL,FRn

If a qNaN is input in an operation that generates a floating-point value, and an sNaN has not been input in that operation, the output will always be a qNaN irrespective of the setting of the EN.V bit in FPSCR. An exception will not be generated in this case.

The qNAN values as operation results are as follows:

- Single-precision qNaN: H'7FBF FFFF
- Double-precision qNaN: H'7FF7 FFFF FFFF FFFF

See section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual for details of floating-point operations when a non-number (NaN) is input.

6.2.3 Denormalized Numbers

For a denormalized number floating-point value, the exponent field is expressed as "0", and the fraction field as a non-zero value.

When the DN bit in FPSCR of the FPU is "1", a denormalized number (source operand or operation result) is always positive or negative zero in a floating-point operation that generates a value (an operation other than transfer instructions between registers, FNEG, or FABS).

When the DN bit in FPSCR is "0", a denormalized number (source operand or operation result) is processed as it is. See section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual for details of floating-point operations when a denormalized number is input.

6.3 Register Descriptions

6.3.1 Floating-Point Registers

Figure 6.4 shows the floating-point register configuration. There are thirty-two 32-bit floating-point registers comprised with two banks: FPR0_BANK0 to FPR15_BANK0, and FPR0_BANK1 to FPR15_BANK1. These thirty-two registers are referenced as FR0 to FR15, DR0/2/4/6/8/10/12/14, FV0/4/8/12, XF0 to XF15, XD0/2/4/6/8/10/12/14, and XMTRX. Corresponding registers to FPR0_BANK0 to FPR15_BANK0, and FPR0_BANK1 to FPR15_BANK1 are determined according to the FR bit of FPSCR.

1. Floating-point registers, FPRn_BANKi (32 registers)
   FPR0_BANK0 to FPR15_BANK0
   FPR0_BANK1 to FPR15_BANK1

2. Single-precision floating-point registers, FRi (16 registers)
   When FPSCR.FR = 0, FR0 to FR15 are allocated to FPR0_BANK0 to FPR15_BANK0;
   when FPSCR.FR = 1, FR0 to FR15 are allocated to FPR0_BANK1 to FPR15_BANK1.

3. Double-precision floating-point registers, DRi (8 registers): A DR register comprises two FR registers.
   DR0 = {FR0, FR1}, DR2 = {FR2, FR3}, DR4 = {FR4, FR5}, DR6 = {FR6, FR7},
   DR8 = {FR8, FR9}, DR10 = {FR10, FR11}, DR12 = {FR12, FR13}, DR14 = {FR14, FR15}

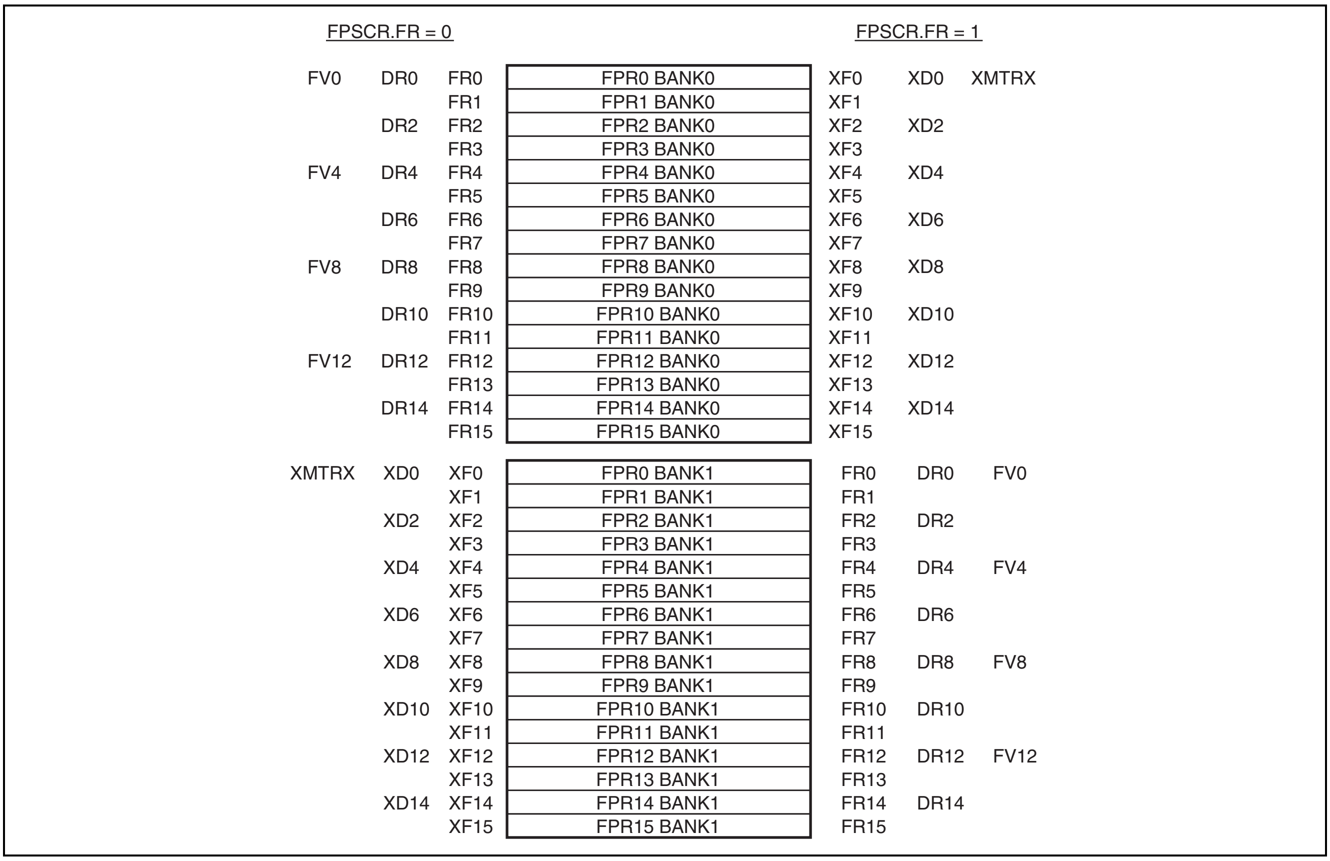
4. Single-precision floating-point vector registers, FVi (4 registers): An FV register comprises four FR registers.
   FV0 = {FR0, FR1, FR2, FR3}, FV4 = {FR4, FR5, FR6, FR7},
   FV8 = {FR8, FR9, FR10, FR11}, FV12 = {FR12, FR13, FR14, FR15}

5. Single-precision floating-point extended registers, XFi (16 registers)
   When FPSCR.FR = 0, XF0 to XF15 are allocated to FPR0_BANK1 to FPR15_BANK1;
   when FPSCR.FR = 1, XF0 to XF15 are allocated to FPR0_BANK0 to FPR15_BANK0.

6. Double-precision floating-point extended registers, XDi (8 registers): An XD register comprises two XF registers.
   XD0 = {XF0, XF1}, XD2 = {XF2, XF3}, XD4 = {XF4, XF5}, XD6 = {XF6, XF7},
   XD8 = {XF8, XF9}, XD10 = {XF10, XF11}, XD12 = {XF12, XF13}, XD14 = {XF14, XF15}

7. Single-precision floating-point extended register matrix, XMTRX: XMTRX comprises all 16 XF registers.

$$\mathrm{XMTRX} = \begin{bmatrix} \mathrm{XF0} & \mathrm{XF4} & \mathrm{XF8} & \mathrm{XF12} \\ \mathrm{XF1} & \mathrm{XF5} & \mathrm{XF9} & \mathrm{XF13} \\ \mathrm{XF2} & \mathrm{XF6} & \mathrm{XF10} & \mathrm{XF14} \\ \mathrm{XF3} & \mathrm{XF7} & \mathrm{XF11} & \mathrm{XF15} \end{bmatrix}$$



FPSCR.FR = 0				FPSCR.FR = 1		
FV0	DR0	FR0	FPR0 BANK0	XF0	XD0	XMTRX
		FR1	FPR1 BANK0	XF1		
	DR2	FR2	FPR2 BANK0	XF2	XD2	
		FR3	FPR3 BANK0	XF3		
FV4	DR4	FR4	FPR4 BANK0	XF4	XD4	
		FR5	FPR5 BANK0	XF5		
	DR6	FR6	FPR6 BANK0	XF6	XD6	
		FR7	FPR7 BANK0	XF7		
FV8	DR8	FR8	FPR8 BANK0	XF8	XD8	
		FR9	FPR9 BANK0	XF9		
	DR10	FR10	FPR10 BANK0	XF10	XD10	
		FR11	FPR11 BANK0	XF11		
FV12	DR12	FR12	FPR12 BANK0	XF12	XD12	
		FR13	FPR13 BANK0	XF13		
	DR14	FR14	FPR14 BANK0	XF14	XD14	
		FR15	FPR15 BANK0	XF15		
XMTRX	XD0	XF0	FPR0 BANK1	FR0	DR0	FV0
		XF1	FPR1 BANK1	FR1		
	XD2	XF2	FPR2 BANK1	FR2	DR2	
		XF3	FPR3 BANK1	FR3		
	XD4	XF4	FPR4 BANK1	FR4	DR4	FV4
		XF5	FPR5 BANK1	FR5		
	XD6	XF6	FPR6 BANK1	FR6	DR6	
		XF7	FPR7 BANK1	FR7		
	XD8	XF8	FPR8 BANK1	FR8	DR8	FV8
		XF9	FPR9 BANK1	FR9		
	XD10	XF10	FPR10 BANK1	FR10	DR10	
		XF11	FPR11 BANK1	FR11		
	XD12	XF12	FPR12 BANK1	FR12	DR12	FV12
		XF13	FPR13 BANK1	FR13		
	XD14	XF14	FPR14 BANK1	FR14	DR14	
		XF15	FPR15 BANK1	FR15		

Figure 6.4 Floating-Point Registers

6.3.2 Floating-Point Status/Control Register (FPSCR)

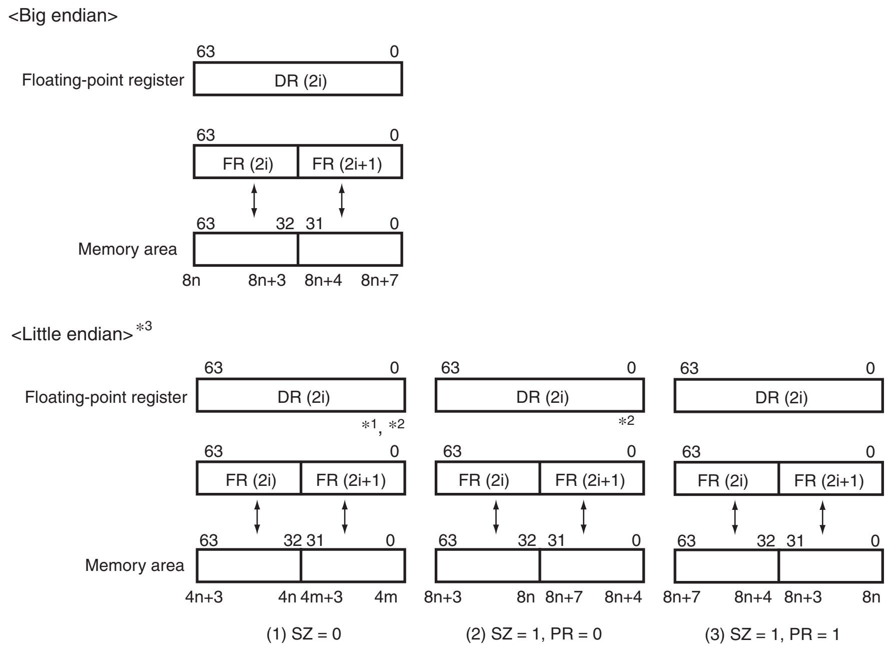
Floating-Point Status/Control Register (FPSCR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	FR	SZ	PR	DN	Cause	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable (EN)					Flag					RM	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0004 0001>

Bit	Abbreviation	After Reset	R	W	Description
31 to 22	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
21	FR	0	R	W	Floating-Point Register Bank 0: FPR0_BANK0 to FPR15_BANK0 are assigned to FR0 to FR15 and FPR0_BANK1 to FPR15_BANK1 are assigned to XF0 to XF15 1: FPR0_BANK0 to FPR15_BANK0 are assigned to XF0 to XF15 and FPR0_BANK1 to FPR15_BANK1 are assigned to FR0 to FR15
20	SZ	0	R	W	Transfer Size Mode 0: Data size of FMOV instruction is 32-bits 1: Data size of FMOV instruction is a 32-bit register pair (64 bits) For relations between endian and the SZ and PR bits, see figure 6.5.
19	PR	0	R	W	Precision Mode 0: Floating-point instructions are executed as single-precision operations 1: Floating-point instructions are executed as double-precision operations (graphics support instructions are undefined) For relations between endian and the SZ and PR bits, see figure 6.5.
18	DN	1	R	W	Denormalization Mode 0: Denormalized number is treated as such 1: Denormalized number is treated as zero
17 to 12	Cause	All 0	R	W	FPU Exception Cause Field
11 to 7	Enable	All 0	R	W	FPU Exception Enable Field
6 to 2	Flag	All 0	R	W	FPU Exception Flag Field Each time an FPU operation instruction is executed, the FPU exception cause field is cleared to "0". When an FPU exception occurs, the bits corresponding to FPU exception cause field and flag field are set to "1". The FPU exception flag field remains set to "1" until it is cleared to "0" by software. For bit allocations of each field, see table 6.3.
1, 0	RM	01	R	W	Rounding Mode These bits select the rounding mode. 00: Round to Nearest 01: Round to Zero 10: Reserved (setting prohibited) 11: Reserved (setting prohibited)



Notes: *1 In the case of SZ = 0 and PR = 0, DR register can not be used.

*2 The bit-location of DR register is used for double precision format when PR = 1. (In the case of (2), it is used when PR is changed from 0 to 1.)

*3 Little endian cannot be selected in this MCU.

Figure 6.5 Relation between SZ Bit and Endian

Table 6.3 Bit Allocation for FPU Exception Handling

Field Name		FPU Error (E)	Invalid Operation (V)	Division by Zero (Z)	Overflow (O)	Underflow (U)	Inexact (I)
Cause	FPU exception cause field	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12
Enable	FPU exception enable field	None	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7
Flag	FPU exception flag field	None	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2

6.3.3 Floating-Point Communication Register (FPUL)

Information is transferred between the FPU and CPU via FPUL. FPUL is a 32-bit system register that is accessed from the CPU side by means of LDS and STS instructions. For example, to convert the integer stored in general register R1 to a single-precision floating-point number, the processing flow is as follows:

R1 → (LDS instruction) → FPUL → (single-precision FLOAT instruction) → FR1

6.4 Rounding

In a floating-point instruction, rounding is performed when generating the final operation result from the intermediate result. Therefore, the result of combination instructions such as FMAC, FTRV, and FIPR will differ from the result when using a basic instruction such as FADD, FSUB, or FMUL. Rounding is performed once in FMAC, but twice in FADD, FSUB, and FMUL.

Which of the two rounding methods is to be used is determined by the RM bits in FPSCR.

RM = "00": Round to Nearest
RM = "01": Round to Zero

(1) Round to Nearest

The operation result is rounded to the nearest expressible value. If there are two nearest expressible values, the one with an LSB of 0 is selected.

If the unrounded value is $2^{Emax} (2 - 2^{-P})$ or more, the result will be infinity with the same sign as the unrounded value. The values of Emax and P, respectively, are 127 and 24 for single-precision, and 1023 and 53 for double-precision.

(2) Round to Zero

The digits below the round bit of the unrounded value are discarded.

If the unrounded value is larger than the maximum expressible absolute value, the value will become the maximum expressible absolute value with the same sign as unrounded value.

6.5 Floating-Point Exceptions

The FPU-related exceptions are described below.

(1) General FPU Disable Exceptions and Slot FPU Disable Exceptions

FPU-related exceptions are occurred when an FPU instruction is executed with SR.FD set to "1". When the FPU instruction is in other than delayed slot, the general FPU disable exception is occurred. When the FPU instruction is in the delay slot, the slot FPU disable exception is occurred.

(2) FPU Exception Sources

The exception sources are as follows:

- FPU error (E): When FPSCR.DN = "0" and a denormalized number is input
- Invalid operation (V): In case of an invalid operation, such as NaN input
- Division by zero (Z): Division with a zero divisor
- Overflow (O): When the operation result overflows
- Underflow (U): When the operation result underflows
- Inexact exception (I): When overflow, underflow, or rounding occurs

The FPU exception cause field in FPSCR contains bits corresponding to all of above sources E, V, Z, O, U, and I, and the FPU exception flag and enable fields in FPSCR contain bits corresponding to sources V, Z, O, U, and I, but not E. Thus, FPU errors cannot be disabled.

When an FPU exception occurs, the corresponding bit in the FPU exception cause field is set to "1", and "1" is added to the corresponding bit in the FPU exception flag field. When an FPU exception does not occur, the corresponding bit in the FPU exception cause field is cleared to "0", but the corresponding bit in the FPU exception flag field remains unchanged.

(3) FPU Exception Handling

FPU exception handling is initiated in the following cases:

- FPU error (E): FPSCR.DN = 0 and a denormalized number is input
- Invalid operation (V): FPSCR.Enable.V = 1 and (instruction = FTRV or invalid operation)
- Division by zero (Z): FPSCR.Enable.Z = 1 and division with a zero divisor or the input of FSRRA is zero
- Overflow (O): FPSCR.Enable.O = 1 and possibility of operation result overflow
- Underflow (U): FPSCR.Enable.U = 1 and possibility of operation result underflow
- Inexact exception (I): FPSCR.Enable.I = 1 and instruction with possibility of inexact operation result

Please refer section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual about the FPU exception case in detail.

All exception events that originate in the FPU are assigned as the same exception event. The meaning of an exception is determined by software by reading from FPSCR and interpreting the information it contains. Also, the destination register is not changed by any FPU exception handling operation.

If the FPU exception sources except for above are generated, the bit corresponding to source V, Z, O, U, or I is set to "1", and a default value is generated as the operation result.

- Invalid operation (V): qNaN is generated as the result.
- Division by zero (Z): Infinity with the same sign as the unrounded value is generated.

- Overflow (O):
  When rounding mode = RZ, the maximum normalized number, with the same sign as the unrounded value, is generated.
  When rounding mode = RN, infinity with the same sign as the unrounded value is generated.
- Underflow (U):
  When FPSCR.DN = 0, a denormalized number with the same sign as the unrounded value, or zero with the same sign as the unrounded value, is generated.
  When FPSCR.DN = 1, zero with the same sign as the unrounded value, is generated.
- Inexact exception (I): An inexact result is generated.

6.6 Graphics Support Functions

The SH-4A supports two kinds of graphics functions: new instructions for geometric operations, and pair single-precision transfer instructions that enable high-speed data transfer.

6.6.1 Geometric Operation Instructions

Geometric operation instructions perform approximate-value computations. To enable high-speed computation with a minimum of hardware, the SH-4A ignores comparatively small values in the partial computation results of four multiplications. Consequently, the error shown below is produced in the result of the computation:

$$\text{Maximum error} = \text{MAX (individual multiplication result} \times 2^{-\text{MIN (number of multiplier significant digits}-1\text{, number of multiplicand significant digits}-1)}) + \text{MAX (result value} \times 2^{-23}, 2^{-149})$$

The number of significant digits is 24 for a normalized number and 23 for a denormalized number (number of leading zeros in the fractional part).

In a future version of the SH Series, the above error is guaranteed, but the same result between different processor cores is not guaranteed.

(1) FIPR FVm, FVn (m, n: 0, 4, 8, 12)

This instruction is basically used for the following purposes:

- Inner product (m ≠ n):
  This operation is generally used for surface/rear surface determination for polygon surfaces.
- Sum of square of elements (m = n):
  This operation is generally used to find the length of a vector.

Since an inexact exception is not detected by an FIPR instruction, the inexact exception (I) bit in both the FPU exception cause field and flag field are always set to "1" when an FIPR instruction is executed. Therefore, if the I bit is set in the FPU exception enable field, FPU exception handling will be executed.

(2) FTRV XMTRX, FVn (n: 0, 4, 8, 12)

This instruction is basically used for the following purposes:

- Matrix (4 × 4) · vector (4):
  This operation is generally used for viewpoint changes, angle changes, or movements called vector transformations (4-dimensional). Since affine transformation processing for angle + parallel movement basically requires a 4 × 4 matrix, the SH-4A supports 4-dimensional operations.
- Matrix (4 × 4) × matrix (4 × 4):
  This operation requires the execution of four FTRV instructions.

Since an inexact exception is not detected by an FIRV instruction, the inexact exception (I) bit in both the FPU exception cause field and flag field are always set to "1" when an FTRV instruction is executed. Therefore, if the I bit is set in the FPU exception enable field, FPU exception handling will be executed. It is not possible to check all data types in the registers beforehand when executing an FTRV instruction. If the V bit is set in the FPU exception enable field, FPU exception handling will be executed.

(3) FRCHG

This instruction modifies banked registers. For example, when the FTRV instruction is executed, matrix elements must be set in an array in the background bank. However, to create the actual elements of a translation matrix, it is easier to use registers in the foreground bank. When the LDS instruction is used on FPSCR, this instruction takes four to five cycles in order to maintain the FPU state. With the FRCHG instruction, the FR bit in FPSCR can be changed in one cycle.

6.6.2 Pair Single-Precision Data Transfer

In addition to the powerful new geometric operation instructions, the SH-4A also supports high-speed data transfer instructions.

When the SZ bit is "1", the SH-4A can perform data transfer by means of pair single-precision data transfer instructions.

- FMOV DRm/XDm, DRn/XDRn (m, n: 0, 2, 4, 6, 8, 10, 12, 14)
- FMOV DRm/XDm, @Rn (m: 0, 2, 4, 6, 8, 10, 12, 14; n: 0 to 15)

These instructions enable two single-precision (2 × 32-bit) data items to be transferred; that is, the transfer performance of these instructions is doubled.

- FSCHG

This instruction changes the value of the SZ bit in FPSCR, enabling fast switching between use and non-use of pair single-precision data transfer.

This page is blank for reasons of layout.

Section 7 Memory Management Unit (MMU)

The SH-4A supports an 8-bit address space identifier, a 32-bit virtual address space, and a 29-bit physical address space. Address translation from virtual addresses to physical addresses is enabled by the memory management unit (MMU) in the SH-4A. The MMU performs high-speed address translation by caching user-created address translation table information in an address translation buffer (translation lookaside buffer: TLB).

The SH-4A has four instruction TLB (ITLB) entries and 64 unified TLB (UTLB) entries. UTLB copies are stored in the ITLB by hardware. A paging system is used for address translation. It is possible to set the virtual address space access right and implement memory protection independently for privileged mode and user mode.

In the SH-4A MMU, both TLB compatible mode (four paging sizes with four protection bits) and TLB extended mode (eight paging sizes with six protection bits) are provided for the MMU flag function.

Selection between TLB compatible mode and TLB extended mode is made by setting the relevant control register (bit ME in the MMUCR register) in software.

The flag functions of the MMU are described together for both TLB compatible mode and TLB extended mode.

7.1 Overview

The MMU was conceived as a means of making efficient use of physical memory. As shown in (0) in figure 7.1, when a process is smaller in size than the physical memory, the entire process can be mapped onto physical memory, but if the process increases in size to the point where it does not fit into physical memory, it becomes necessary to divide the process into smaller parts, and map the parts requiring execution onto physical memory as occasion arises ((1) in figure 7.1). Having this mapping onto physical memory executed consciously by the process itself imposes a heavy burden on the process. The virtual memory system was devised as a means of handling all physical memory mapping to reduce this burden ((2) in figure 7.1). With a virtual memory system, the size of the available virtual memory is much larger than the actual physical memory, and processes are mapped onto this virtual memory. Thus processes only have to consider their operation in virtual memory, and mapping from virtual memory to physical memory is handled by the MMU. The MMU is normally managed by the OS, and physical memory switching is carried out so as to enable the virtual memory required by a process to be mapped smoothly onto physical memory. Physical memory switching is performed via secondary storage, etc.

The virtual memory system that came into being in this way works to best effect in a time sharing system (TSS) that allows a number of processes to run simultaneously ((3) in figure 7.1). Running a number of processes in a TSS did not increase efficiency since each process had to take account of physical memory mapping. Efficiency is improved and the load on each process reduced by the use of a virtual memory system ((4) in figure 7.1). In this virtual memory system, virtual memory is allocated to each process. The task of the MMU is to map a number of virtual memory areas onto physical memory in an efficient manner. It is also provided with memory protection functions to prevent a process from inadvertently accessing another process's physical memory.

When address translation from virtual memory to physical memory is performed using the MMU, it may happen that the translation information has not been recorded in the MMU, or the virtual memory of a different process is accessed by mistake. In such cases, the MMU will generate an exception, change the physical memory mapping, and record the new address translation information.

Although the functions of the MMU could be implemented by software alone, having address translation performed by software each time a process accessed physical memory would be very inefficient. For this reason, a buffer for address translation (the translation lookaside buffer: TLB) is provided by hardware, and frequently used address translation information is placed here. The TLB can be described as a cache for address translation information. However, unlike a cache, if address translation fails—that is, if an exception occurs—switching of the address translation information is normally performed by software. Thus memory management can be performed in a flexible manner by software.

There are two methods by which the MMU can perform mapping from virtual memory to physical memory: the paging method, using fixed-length address translation, and the segment method, using variable-length address translation. With the paging method, the unit of translation is a fixed-size address space called a page.

In the following descriptions, the address space in virtual memory in the SH-4A is referred to as virtual address space, and the address space in physical memory as physical address space.

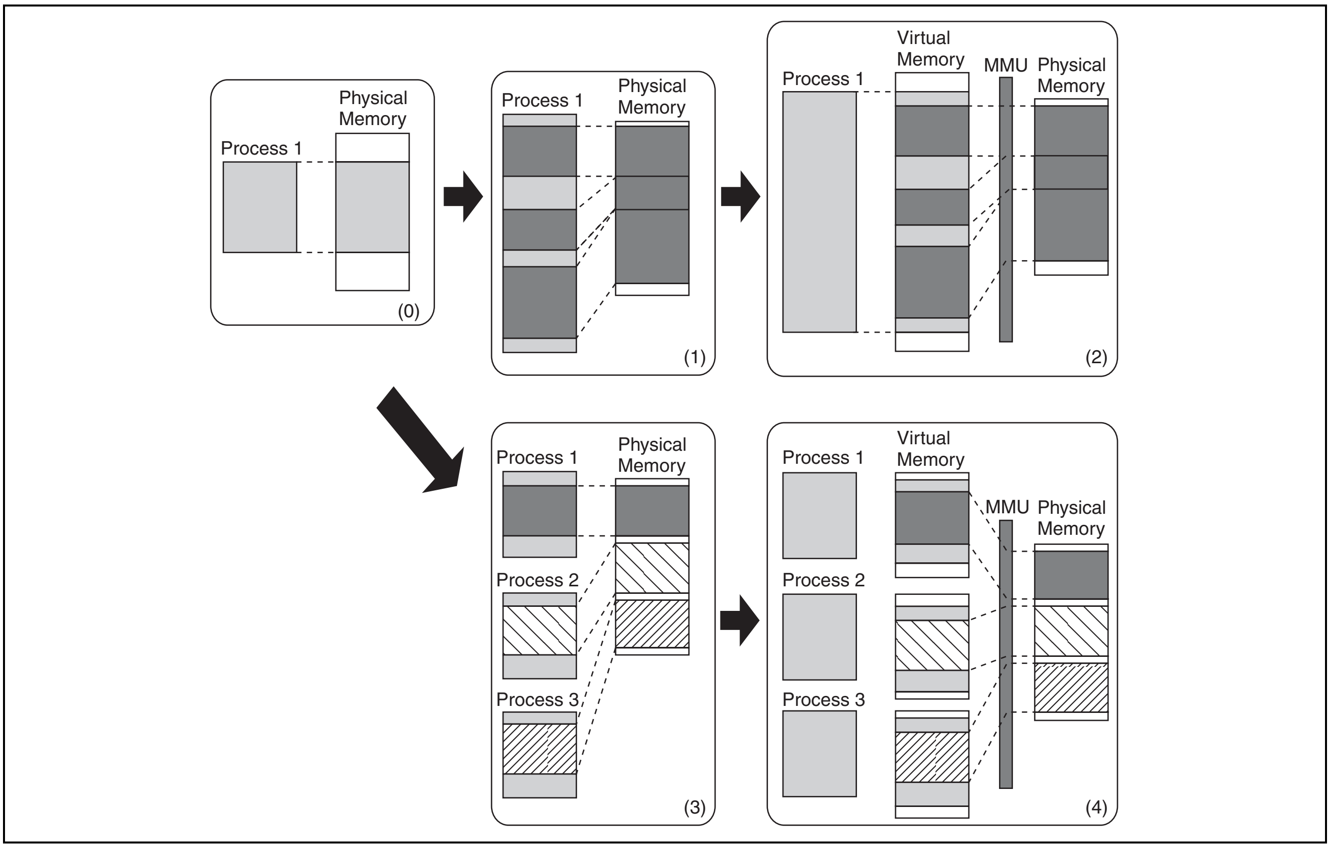


Figure 7.1 Role of MMU

7.1.1 Address Spaces

(1) Virtual Address Space

The SH-4A supports a 32-bit virtual address space, and can access a 4-Gbyte address space. The virtual address space is divided into a number of areas, as shown in figures 7.2 and 7.3. In privileged mode, the 4-Gbyte space from the P0 area to the P4 area can be accessed. In user mode, a 2-Gbyte space in the U0 area can be accessed. When the SQMD bit in the MMU control register (MMUCR) is "0", a 64-Mbyte space in the store queue area can be accessed. When the RMD bit in the on-chip memory control register (RAMCR) is "1", a 16-Mbyte space in on-chip memory area can be accessed. Accessing areas other than the U0 area, store queue area, and on-chip memory area in user mode will cause an address error.

When the AT bit in MMUCR is set to "1" and the MMU is enabled, the P0, P3, and U0 areas can be mapped onto any physical address space in 1-, 4-, 64-Kbyte, or 1-Mbyte page units in TLB compatible mode and in 1-, 4-, 8-, 64-, 256-Kbyte, 1-, 4-, or 64-Mbyte page units in TLB extended mode. By using an 8-bit address space identifier, the P0, P3, and U0 areas can be increased to a maximum of 256. Mapping from the virtual address space to the 29-bit physical address space is carried out using the TLB.

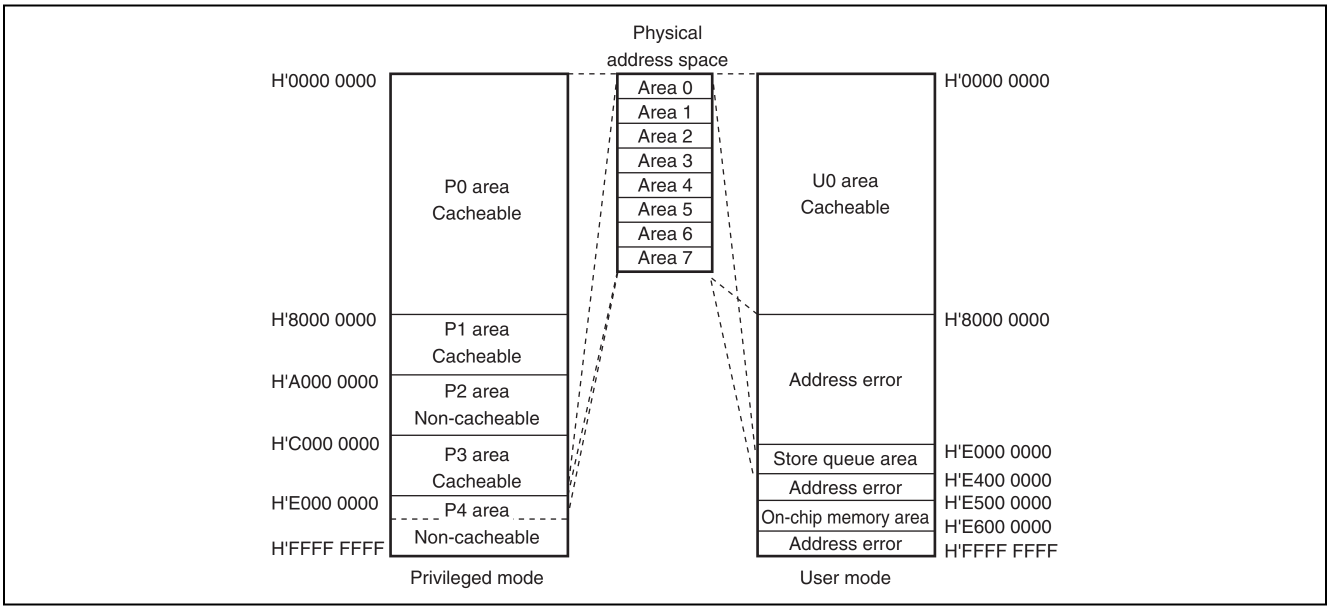


Figure 7.2 Virtual Address Space (AT in MMUCR = 0)

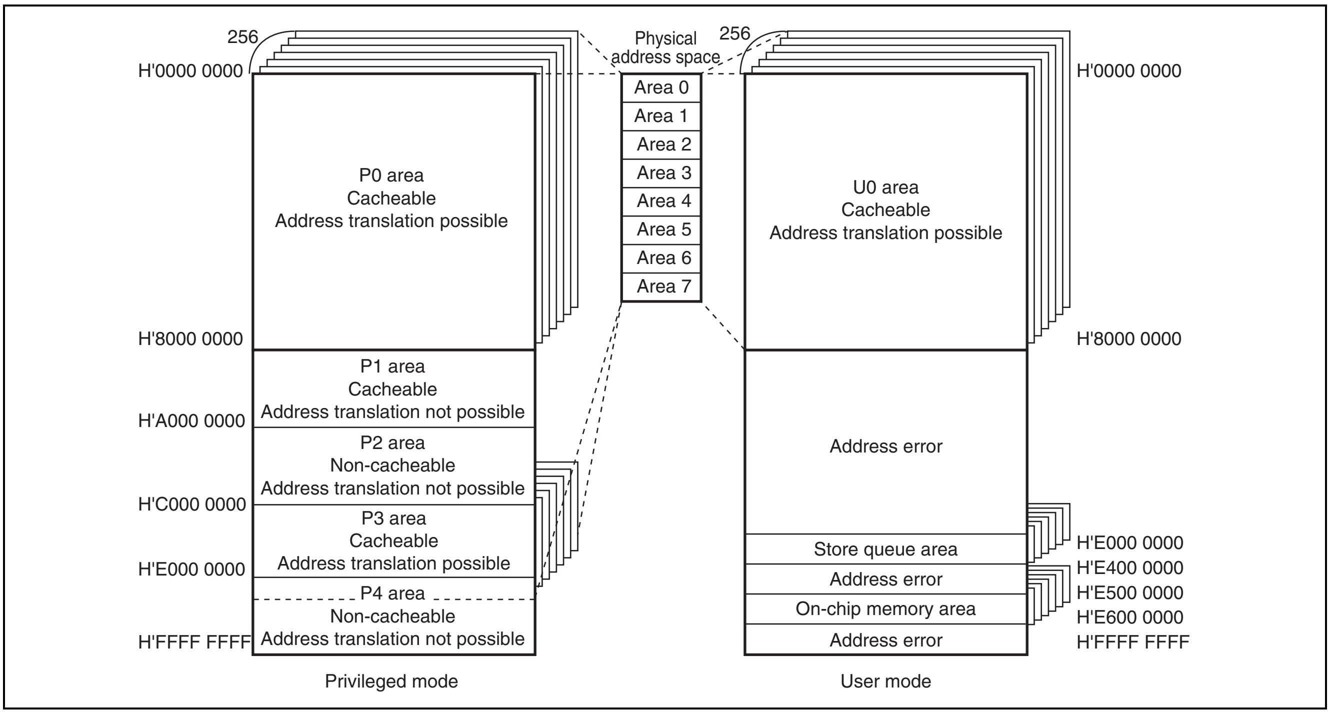


Figure 7.3 Virtual Address Space (AT in MMUCR = 1)

(a) P0, P3, and U0 Areas

The P0, P3, and U0 areas allow address translation using the TLB and access using the cache.

When the MMU is disabled, replacing the upper 3 bits of an address with 0s gives the corresponding physical address. Whether or not the cache is used is determined by the CCR setting. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the WT bit in CCR.

When the MMU is enabled, these areas can be mapped onto any physical address space in 1-, 4-, 64-Kbyte, or 1-Mbyte page units in TLB compatible mode and in 1-, 4-, 8-, 64, 256-Kbyte, 1-, 4-, or 64-Mbyte page units in TLB extended mode using the TLB. When CCR is in the cache enabled state and the C bit for the corresponding page of the TLB entry is "1", accesses can be performed using the cache. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the WT bit of the TLB entry.

When the P0, P3, and U0 areas are mapped onto the control register area which is allocated in the area 7 in physical address space by means of the TLB, the C bit for the corresponding page must be cleared to "0".

(b) P1 Area

The P1 area does not allow address translation using the TLB but can be accessed using the cache.

Regardless of whether the MMU is enabled or disabled, clearing the upper 3 bits of an address to "0" gives the corresponding physical address. Whether or not the cache is used is determined by the CCR setting. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the CB bit in CCR.

(c) P2 Area

The P2 area does not allow address translation using the TLB and access using the cache.

Regardless of whether the MMU is enabled or disabled, clearing the upper 3 bits of an address to "0" gives the corresponding physical address.

(d) P4 Area

The P4 area is mapped onto the internal resource of the SH-4A. This area except the store queue and on-chip memory areas does not allow address translation using the TLB. This area cannot be accessed using the cache. The P4 area is shown in detail in figure 7.4.

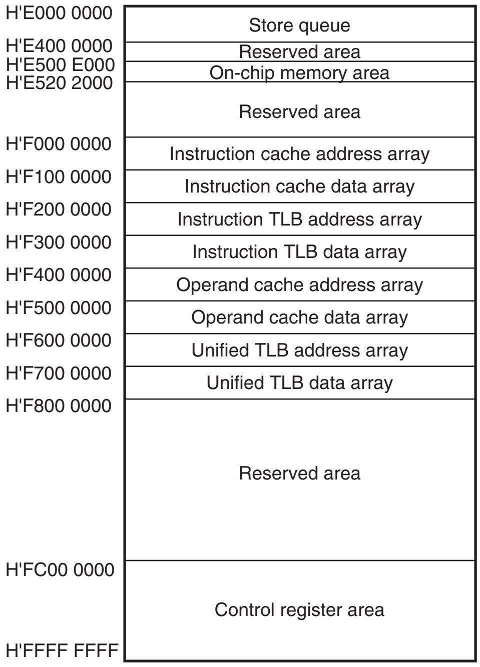


Figure 7.4 P4 Area

The area from H'E000 0000 to H'E3FF FFFF comprises addresses for accessing the store queues (SQs). In user mode, the access right is specified by the SQMD bit in MMUCR. For details, see section 8.7, Store Queues.

The area from H'E500 E000 to H'E520 1FFF comprises addresses for accessing the on-chip memory. In user mode, the access right is specified by the RMD bit in RAMCR. For details, see section 9, IL Memory/OL Memory.

The area from H'F000 0000 to H'F0FF FFFF is used for direct access to the instruction cache address array. For details, see section 8.6.1, IC Address Array.

The area from H'F100 0000 to H'F1FF FFFF is used for direct access to the instruction cache data array. For details, see section 8.6.2, IC Data Array.

The area from H'F200 0000 to H'F2FF FFFF is used for direct access to the instruction TLB address array. For details, see section 7.7.1, ITLB Address Array.

The area from H'F300 0000 to H'F3FF FFFF is used for direct access to instruction TLB data array. For details, see section 7.7.2, ITLB Data Array (TLB Compatible Mode) and section 7.7.3, ITLB Data Array (TLB Extended Mode).

The area from H'F400 0000 to H'F4FF FFFF is used for direct access to the operand cache address array. For details, see section 8.6.3, OC Address Array.

The area from H'F500 0000 to H'F5FF FFFF is used for direct access to the operand cache data array. For details, see section 8.6.4, OC Data Array.

The area from H'F600 0000 to H'F6FF FFFF is used for direct access to the unified TLB address array. For details, see section 7.7.4, UTLB Address Array.

The area from H'F700 0000 to H'F7FF FFFF is used for direct access to unified TLB data array. For details, see section 7.7.5, UTLB Data Array (TLB Compatible Mode) and section 7.7.6, UTLB Data Array (TLB Extended Mode).

The area from H'FC00 0000 to H'FFFF FFFF is the on-chip peripheral module control register area. For details, see register descriptions in each section of the hardware manual of the product.

(2) Physical Address Space

The SH-4A supports a 29-bit physical address space. The physical address space is divided into eight areas as shown in figure 7.5. Area 7 is a reserved area. For details, see section 11, Address Space of the hardware manual of the product.

Only when area 7 in the physical address space is accessed using the TLB, addresses H'1C00 0000 to H'1FFF FFFF of area 7 are not designated as a reserved area, but are equivalent to the control register area in the P4 area, in the virtual address space.

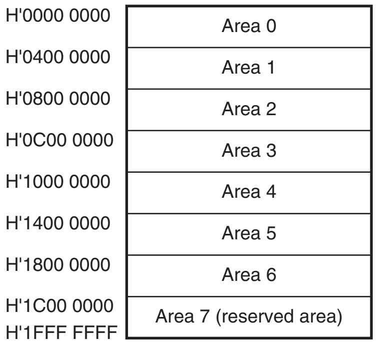


Figure 7.5 Physical Address Space

(3) Address Translation

When the MMU is used, the virtual address space is divided into units called pages, and translation to physical addresses is carried out in these page units. The address translation table in external memory contains the physical addresses corresponding to virtual addresses and additional information such as memory protection codes. Fast address translation is achieved by caching the contents of the address translation table located in external memory into the TLB. In the SH-4A, basically, the ITLB is used for instruction accesses and the UTLB for data accesses. In the event of an access to an area other than the P4 area, the accessed virtual address is translated to a physical address. If the virtual address belongs to the P1 or P2 area, the physical address is uniquely determined without accessing the TLB. If the virtual address belongs to the P0, U0, or P3 area, the TLB is searched using the virtual address, and if the virtual address is recorded in the TLB, a TLB hit is made and the corresponding physical address is read from the TLB. If the accessed virtual address is not recorded in the TLB, a TLB miss exception is generated and processing switches to the TLB miss exception handling routine. In the TLB miss exception handling routine, the address translation table in external memory is searched, and the corresponding physical address and page management information are recorded in the TLB. After the return from the exception handling routine, the instruction which caused the TLB miss exception is re-executed.

(4) Single Virtual Memory Mode and Multiple Virtual Memory Mode

There are two virtual memory systems, single virtual memory and multiple virtual memory, either of which can be selected with the SV bit in MMUCR. In the single virtual memory system, a number of processes run simultaneously, using virtual address space on an exclusive basis, and the physical address corresponding to a particular virtual address is uniquely determined. In the multiple virtual memory system, a number of processes run while sharing the virtual address space, and particular virtual addresses may be translated into different physical addresses depending on the process. The only difference between the single virtual memory and multiple virtual memory systems in terms of operation is in the TLB address comparison method (see section 7.3.3, Address Translation Method).

(5) Address Space Identifier (ASID)

In multiple virtual memory mode, an 8-bit address space identifier (ASID) is used to distinguish between multiple processes running simultaneously while sharing the virtual address space. Software can set the 8-bit ASID of the currently executing process in PTEH in the MMU. The TLB does not have to be purged when processes are switched by means of ASID.

In single virtual memory mode, ASID is used to provide memory protection for multiple processes running simultaneously while using the virtual address space on an exclusive basis.

Note: • Two or more entries with the same virtual page number (VPN) but different ASID must not be set in the TLB simultaneously in single virtual memory mode.

7.2 Register Descriptions

The following registers are related to MMU processing.

Table 7.1 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Page table entry high register	PTEH	Undefined	H'FF00 0000	32	7-9
Page table entry low register	PTEL	Undefined	H'FF00 0004	32	7-10
Translation table base register	TTB	Undefined	H'FF00 0008	32	7-11
TLB exception address register	TEA	Undefined	H'FF00 000C	32	7-11
MMU control register	MMUCR	H'0000 0000	H'FF00 0010	32	7-11
Page table entry assistance register	PTEA	Undefined	H'FF00 0034	32	7-14
Physical address space control register	PASCR	H'0000 0000	H'FF00 0070	32	7-15
Instruction re-fetch inhibit control register	IRMCR	H'0000 0000	H'FF00 0078	32	7-15

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

7.2.1 Page Table Entry High Register (PTEH)

The PTEH register consists of the virtual page number (VPN) and address space identifier (ASID). When an MMU exception or address error exception occurs, the VPN of the virtual address at which the exception occurred is set in the VPN bit by hardware. VPN varies according to the page size, but the VPN set by hardware when an exception occurs consists of the upper 22 bits of the virtual address which caused the exception. VPN setting can also be carried out by software. The number of the currently executing process is set in the ASID bit by software. ASID is not updated by hardware. VPN and ASID are recorded in the UTLB by means of the LDTLB instruction.

After the ASID field in PTEH has been updated, execute one of the following three methods before an access (including an instruction fetch) to the P0, P3, or U0 area that uses the updated ASID value is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the P0, P3, or U0 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is set to "0" (the value after a reset) before updating the PTEH register, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after the ASID field has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

Page Table Entry High Register (PTEH) <P4 address: location H'FF00 0000>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VPN															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPN						—	—	ASID							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 10	VPN	Undefined	R	W	Virtual Page Number
9, 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7 to 0	ASID	Undefined	R	W	Address Space Identifier

7.2.2 Page Table Entry Low Register (PTEL)

The PTEL register is used to hold the physical page number and page management information to be recorded in the UTLB by means of the LDTLB instruction. The contents of this register are not changed unless a software directive is issued.

Page Table Entry Low Register (PTEL) <P4 address: location H'FF00 0004>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	PPN												
After Reset:	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PPN						—	V	SZ1	PR1	PR0	SZ0	C	D	SH	WT
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 10	PPN	Undefined	R	W	Physical Page Number
9	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
8	V	Undefined	R	W	Page Management Information The meaning of each bit is same as that of corresponding bit in Common TLB (UTLB). For details, see section 7.3, TLB Functions (TLB Compatible Mode) and section 7.4, TLB Functions (TLB Extended Mode). Note: • SZ1, PR1, SZ0, and PR0 bits are valid only in TLB compatible mode.
7	SZ1	Undefined	R	W	
6	PR1	Undefined	R	W	
5	PR0	Undefined	R	W	
4	SZ0	Undefined	R	W	
3	C	Undefined	R	W	
2	D	Undefined	R	W	
1	SH	Undefined	R	W	
0	WT	Undefined	R	W	

7.2.3 Translation Table Base Register (TTB)

The TTB register is used to store the base address of the currently used page table, and so on. The contents of the TTB register are not changed unless a software directive is issued. This register can be used freely by software.

Translation Table Base Register (TTB) <P4 address: location H'FF00 0008>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TTB															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTB															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TTB	Undefined	R	W	TTB Bits These bits are used to store values such as the base address of the page table entry currently in use.

7.2.4 TLB Exception Address Register (TEA)

After an MMU exception or address error exception occurs, the virtual address at which the exception occurred is stored. The contents of this register can be changed by software.

TLB Exception Address Register (TEA) <P4 address: location H'FF00 000C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEA															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEA															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TEA	Undefined	R	W	TEA Bits These bits are used to store the virtual address that triggered an MMU exception or address error.

7.2.5 MMU Control Register (MMUCR)

The individual bits perform MMU settings as shown below. Therefore, MMUCR register rewriting should be performed by a program in the P1 or P2 area.

After MMUCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the P0, P3, U0, or store queue area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the P0, P3, or U0 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is "0" (the value after a reset) before updating MMUCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

The MMUCR register contents can be changed by software. However, the LRUI and URC bits may also be updated by hardware.

MMU Control Register (MMUCR) <P4 address: location H'FF00 0010>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LRUI						—	—	URB						—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	URC						SQMD	SV	ME	—	—	—	—	TI	—	AT
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 26	LRUI	All 0	R	W	Least Recently Used ITLB Bits These bits indicate the ITLB entry to be replaced. The LRU (least recently used) method is used to decide the ITLB entry to be replaced in the event of an ITLB miss. The entry to be purged from the ITLB can be confirmed using the LRUI bits. LRUI is updated by means of the algorithm shown below. x means that updating is not performed. 000xxx: ITLB entry 0 is used 1xx00x: ITLB entry 1 is used x1x1x0: ITLB entry 2 is used xx1x11: ITLB entry 3 is used xxxxxx: Other than above When the LRUI bit settings are as shown below, the corresponding ITLB entry is updated by an ITLB miss. Ensure that values for which "Setting prohibited" is indicated below are not set at the discretion of software. After a hardware reset, the LRUI bits are initialized to "0", and therefore a prohibited setting is never made by a hardware update. x means "don't care". 111xxx: ITLB entry 0 is updated 0xx11x: ITLB entry 1 is updated x0x0x1: ITLB entry 2 is updated xx0x00: ITLB entry 3 is updated Other than above: Setting prohibited
25, 24	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
23 to 18	URB	All 0	R	W	UTLB Replace Boundary Bits These bits indicate the UTLB entry boundary at which replacement is to be performed. Valid only when URB ≠ 0.
17, 16	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	URC	All 0	R	W	UTLB Replace Counter These bits serve as a random counter for indicating the UTLB entry for which replacement is to be performed with an LDTLB instruction. This bit is incremented each time the UTLB is accessed. If URB > 0, URC is cleared to "0" when the condition URC = URB is satisfied. Also note that if a value is written to URC by software which results in the condition of URC > URB, incrementing is first performed in excess of URB until URC = H'3F. URC is not incremented by an LDTLB instruction.
9	SQMD	0	R	W	Store Queue Mode Bit Specifies the right of access to the store queues. 0: User/privileged access possible 1: Privileged access possible (address error exception in case of user access)
8	SV	0	R	W	Single Virtual Memory Mode/Multiple Virtual Memory Mode Switching Bit When this bit is changed, ensure that 1 is also written to the TI bit. 0: Multiple virtual memory mode 1: Single virtual memory mode
7	ME	0	R	W	TLB Extended Mode Switching Bit 0: TLB compatible mode 1: TLB extended mode When modifying the ME bit value, always set the TI bit to "1" to invalidate the contents of ITLB and UTLB.
6 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	TI	0	0	W	TLB Invalidate Bit Writing "1" to this bit invalidates (clears to "0") all valid UTLB/ITLB bits. This bit is always read as "0".
1	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
0	AT	0	R	W	Address Translation Enable Bit These bits enable or disable the MMU. 0: MMU disabled 1: MMU enabled MMU exceptions are not generated when the AT bit is "0". In the case of software that does not use the MMU, the AT bit should be cleared to "0".

7.2.6 Page Table Entry Assistance Register (PTEA)

Page Table Entry Assistance Register (PTEA) <P4 address: location H'FF00 0034>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	EPR						ESZ				—	—	—	—
After Reset:	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0	0	0

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13 to 8	EPR	Undefined	R	W	Page Control Information Each bit has the same function as the corresponding bit of the unified TLB (UTLB). For details, see section 7.4, TLB Functions (TLB Extended Mode).
7 to 4	ESZ	Undefined	R	W	
3 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

7.2.7 Physical Address Space Control Register (PASCR)

The PASCR register controls the operation in the physical address space.

Physical Address Space Control Register (PASCR) <P4 address: location H'FF00 0070>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UB							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7 to 0	UB	All 0	R	W	Buffered Write Control for Each Area (64 Mbytes) When writing is performed without using the cache or in the cache write-through mode, these bits specify whether the next bus access from the CPU waits for the end of writing for each area. 0 : Buffered write (The CPU does not wait for the end of writing bus access and starts the next bus access) 1 : Unbuffered write (The CPU waits for the end of writing bus access and starts the next bus access) UB[7]: Corresponding to the control register area UB[6]: Corresponding to area 6 UB[5]: Corresponding to area 5 UB[4]: Corresponding to area 4 UB[3]: Corresponding to area 3 UB[2]: Corresponding to area 2 UB[1]: Corresponding to area 1 UB[0]: Corresponding to area 0

7.2.8 Instruction Re-Fetch Inhibit Control Register (IRMCR)

When the specific resource is changed, the IRMCR register controls whether the instruction fetch is performed again for the next instruction. The specific resource means the part of control registers, TLB, and cache.

In the initial state, the instruction fetch is performed again for the next instruction after changing the resource. However, the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction every time the resource is changed. Therefore, it is recommended that each bit in IRMCR is set to "1" and the specific instruction should be executed after all necessary resources have been changed prior to execution of the program which uses changed resources.

For details on the specific sequence, see descriptions in each resource.

Instruction Re-Fetch Inhibit Control Register (IRMCR) <P4 address: location H'FF00 0078>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	R2	R1	LT	MT	MC
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	R2	0	R	W	Re-Fetch Inhibit 2 after Register Change When MMUCR, PASCR, CCR, PTEH, or RAMCR is changed, this bit controls whether re-fetch is performed for the next instruction. 0: Re-fetch is performed 1: Re-fetch is not performed
3	R1	0	R	W	Re-Fetch Inhibit 1 after Register Change When a register allocated in addresses H'FF20 0000 to H'FF2F FFFF is changed, this bit controls whether re-fetch is performed for the next instruction. 0: Re-fetch is performed 1: Re-fetch is not performed
2	LT	0	R	W	Re-Fetch Inhibit after LDTLB Execution This bit controls whether re-fetch is performed for the next instruction after the LDTLB instruction has been executed. 0: Re-fetch is performed 1: Re-fetch is not performed
1	MT	0	R	W	Re-Fetch Inhibit after Writing Memory-Mapped TLB This bit controls whether re-fetch is performed for the next instruction after writing memory-mapped ITLB/UTLB while the AT bit in MMUCR is set to "1". 0: Re-fetch is performed 1: Re-fetch is not performed
0	MC	0	R	W	Re-Fetch Inhibit after Writing Memory-Mapped IC This bit controls whether re-fetch is performed for the next instruction after writing memory-mapped IC while the ICE bit in CCR is set to "1". 0: Re-fetch is performed 1: Re-fetch is not performed

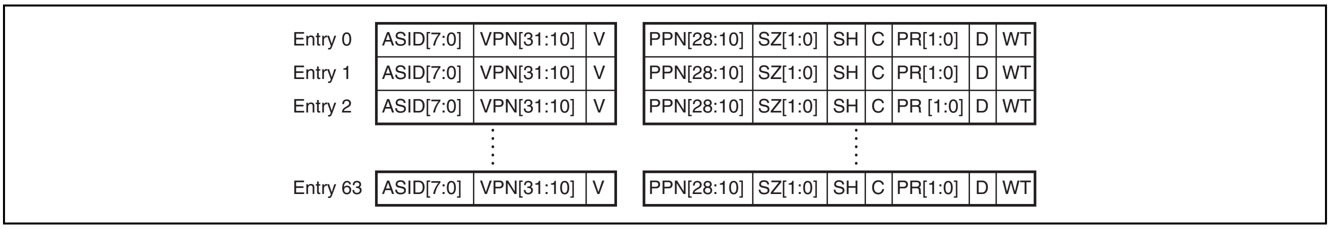
7.3 TLB Functions (TLB Compatible Mode)

7.3.1 Unified TLB (UTLB) Configuration

The UTLB is used for the following two purposes:

1. To translate a virtual address to a physical address in a data access
2. As a table of address translation information to be recorded in the ITLB in the event of an ITLB miss

The UTLB is so called because of its use for the above two purposes. Information in the address translation table located in external memory is cached into the UTLB. The address translation table contains virtual page numbers and address space identifiers, and corresponding physical page numbers and page management information. Figure 7.6 shows the UTLB configuration. The UTLB consists of 64 fully-associative type entries. Figure 7.7 shows the relationship between the page size and address format.



Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR[1:0]	D	WT
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR[1:0]	D	WT
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR [1:0]	D	WT
	⋮					⋮				
Entry 63	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR[1:0]	D	WT

Figure 7.6 UTLB Configuration (TLB Compatible Mode)

Legend:

- ASID: Address space identifier

  Indicates the process that can access a virtual page.

  In single virtual memory mode and user mode, or in multiple virtual memory mode, if the SH bit is "0", this identifier is compared with the ASID in PTEH when address comparison is performed.
- VPN: Virtual page number

  For 1-Kbyte page: Upper 22 bits of virtual address

  For 4-Kbyte page: Upper 20 bits of virtual address

  For 64-Kbyte page: Upper 16 bits of virtual address

  For 1-Mbyte page: Upper 12 bits of virtual address
- V: Validity bit

  Indicates whether the entry is valid.

  0: Invalid

  1: Valid

  Cleared to "0" by a hardware reset.
- PPN: Physical page number

  Upper 22 bits of the physical address of the physical page number.

  With a 1-Kbyte page, PPN[28:10] are valid.

  With a 4-Kbyte page, PPN[28:12] are valid.

  With a 64-Kbyte page, PPN[28:16] are valid.

  With a 1-Mbyte page, PPN[28:20] are valid.

  The synonym problem must be taken into account when setting the PPN (see section 7.5.5, Avoiding Synonym Problems).

- SZ[1:0]: Page size bits
  Specify the page size.
  00: 1-Kbyte page
  01: 4-Kbyte page
  10: 64-Kbyte page
  11: 1-Mbyte page
- SH: Share status bit
  When 0, pages are not shared by processes.
  When 1, pages are shared by processes.
- C: Cacheability bit
  Indicates whether a page is cacheable.
  0: Not cacheable
  1: Cacheable
  When the control register area is mapped, this bit must be cleared to "0".
- PR[1:0]: Protection key data
  2-bit data expressing the page access right as a code.
  00: Can be read from only in privileged mode
  01: Can be read from and written to in privileged mode
  10: Can be read from only in privileged or user mode
  11: Can be read from and written to in privileged mode or user mode
- D: Dirty bit
  Indicates whether a write has been performed to a page.
  0: Write has not been performed
  1: Write has been performed
- WT: Write-through bit
  Specifies the cache write mode.
  0: Copy-back mode
  1: Write-through mode

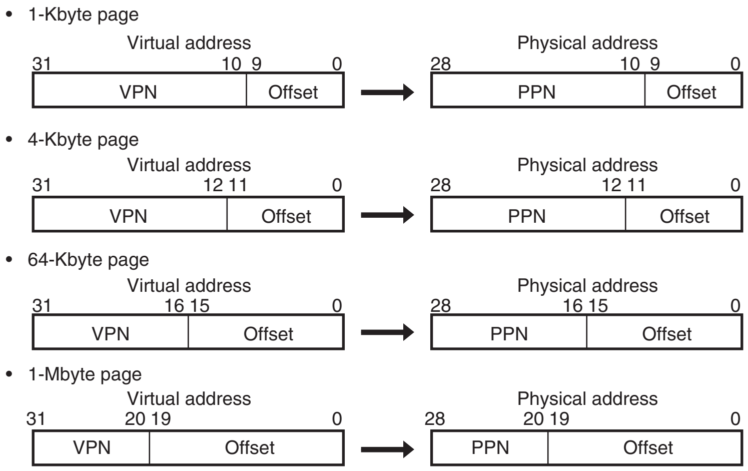


Figure 7.7 Relationship between Page Size and Address Format (TLB Compatible Mode)

7.3.2 Instruction TLB (ITLB) Configuration

The ITLB is used to translate a virtual address to a physical address in an instruction access. Information in the address translation table located in the UTLB is cached into the ITLB. Figure 7.8 shows the ITLB configuration. The ITLB consists of four fully-associative type entries.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 3	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR

Notes:
- The D and WT bits are not supported.
- There is only one PR bit, corresponding to the upper bit of the PR bits in the UTLB.

Figure 7.8 ITLB Configuration (TLB Compatible Mode)

7.3.3 Address Translation Method

Figure 7.9 shows a flowchart of a memory access using the UTLB.

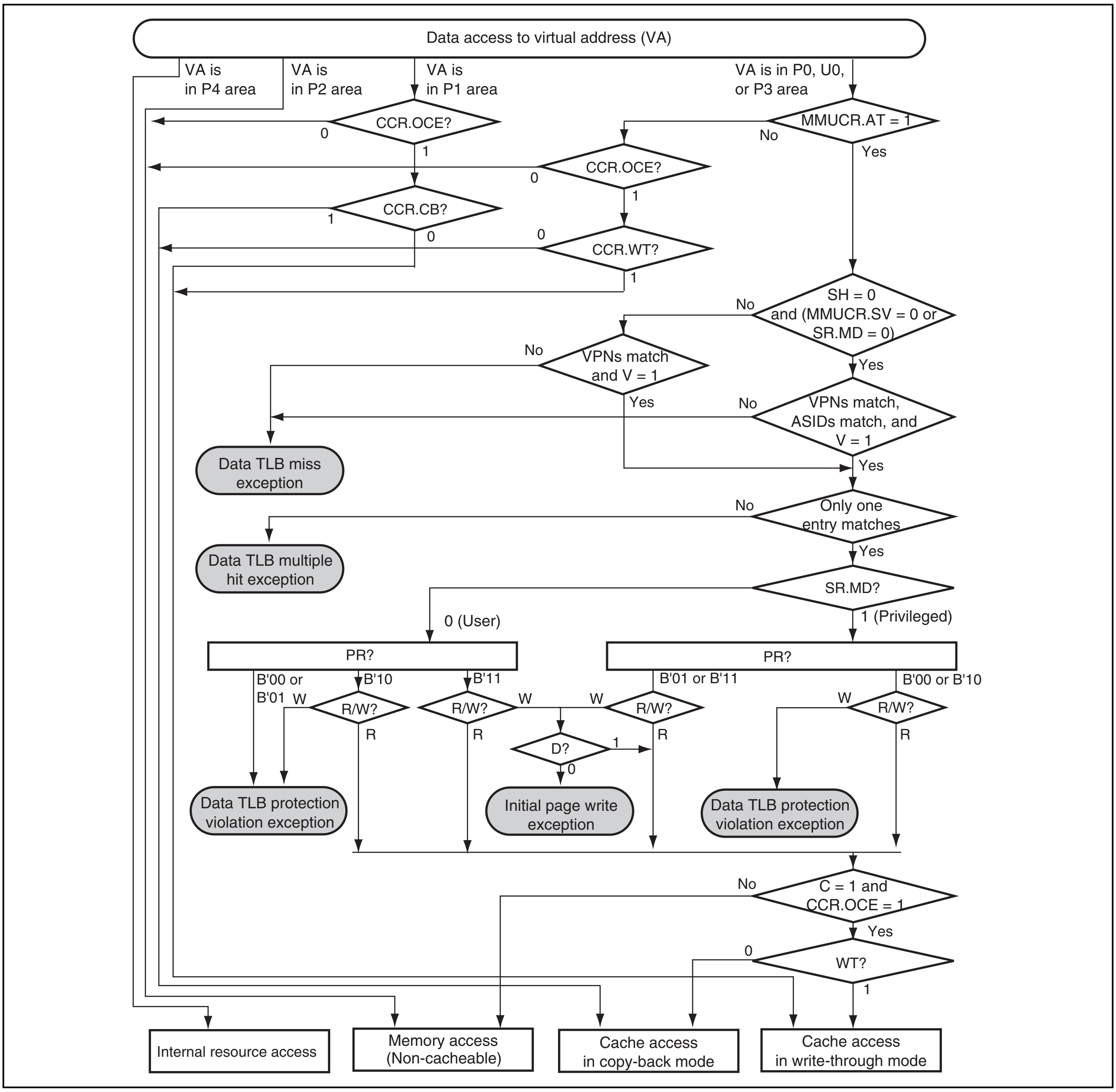

Data access to virtual address (VA)
VA is in P4 area
VA is in P2 area
VA is in P1 area
VA is in P0, U0, or P3 area
CCR.OCE?
0
1
MMUCR.AT = 1
No
Yes
CCR.OCE?
0
1
CCR.CB?
1
0
CCR.WT?
0
1
SH = 0 and (MMUCR.SV = 0 or SR.MD = 0)
No
Yes
VPNs match and V = 1
No
Yes
VPNs match, ASIDs match, and V = 1
No
Yes
Data TLB miss exception
Only one entry matches
No
Yes
Data TLB multiple hit exception
SR.MD?
0 (User)
1 (Privileged)
PR?
B'00 or B'01
B'10
B'11
W
R/W?
R
R/W?
W
R
D?
1
0
PR?
B'01 or B'11
B'00 or B'10
W
R/W?
R
W
R/W?
R
Data TLB protection violation exception
Initial page write exception
Data TLB protection violation exception
C = 1 and CCR.OCE = 1
No
Yes
WT?
0
1
Internal resource access
Memory access (Non-cacheable)
Cache access in copy-back mode
Cache access in write-through mode

Figure 7.9 Flowchart of Memory Access Using UTLB (TLB Compatible Mode)

Figure 7.10 shows a flowchart of a memory access using the ITLB.

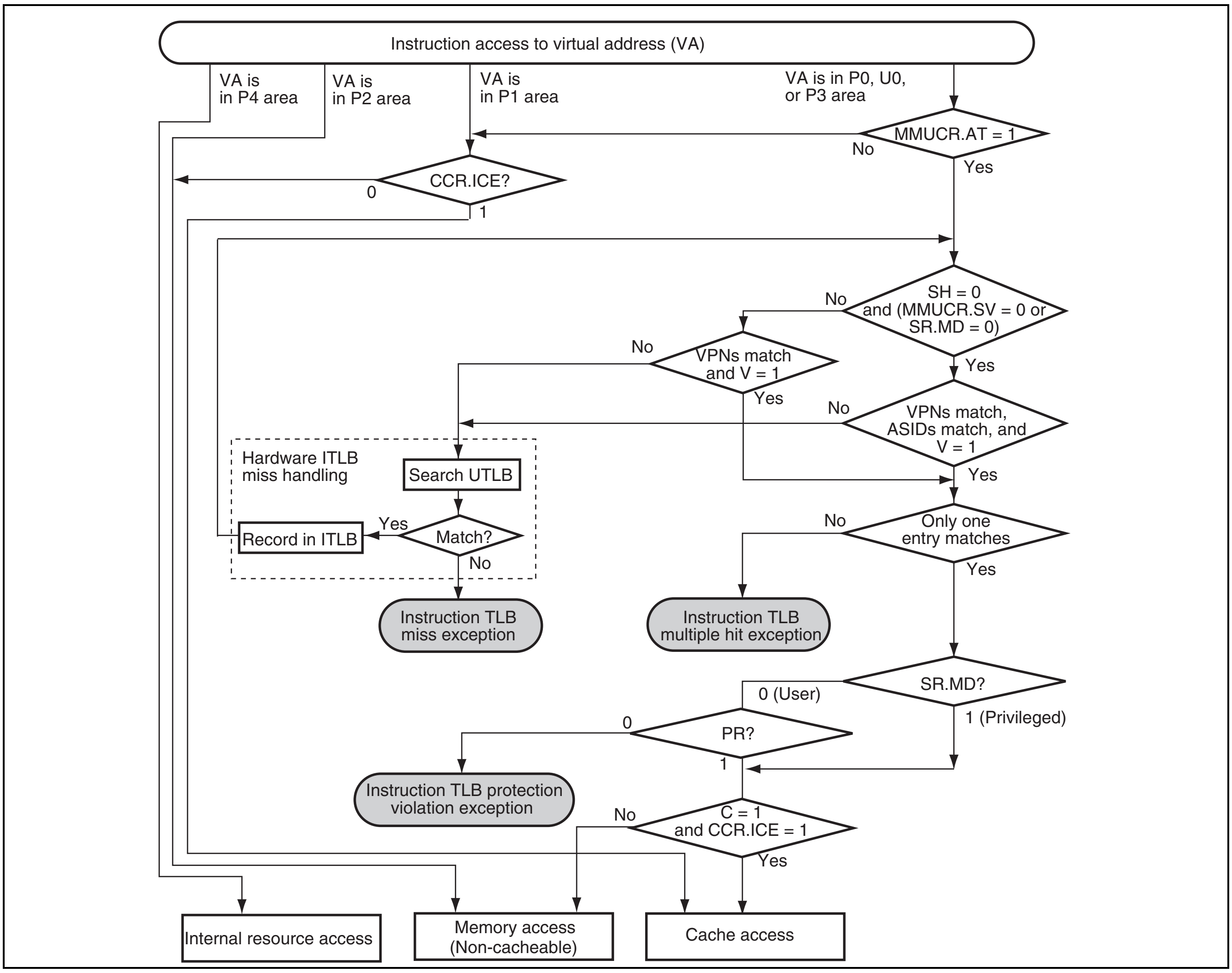

Instruction access to virtual address (VA)
VA is in P4 area
VA is in P2 area
VA is in P1 area
VA is in P0, U0, or P3 area
MMUCR.AT = 1
No
Yes
CCR.ICE?
0
1
SH = 0 and (MMUCR.SV = 0 or SR.MD = 0)
No
Yes
VPNs match and V = 1
No
Yes
VPNs match, ASIDs match, and V = 1
No
Yes
Hardware ITLB miss handling
Search UTLB
Match?
Yes
No
Record in ITLB
Only one entry matches
No
Yes
Instruction TLB miss exception
Instruction TLB multiple hit exception
SR.MD?
0 (User)
1 (Privileged)
PR?
0
1
Instruction TLB protection violation exception
C = 1 and CCR.ICE = 1
No
Yes
Internal resource access
Memory access (Non-cacheable)
Cache access

Figure 7.10 Flowchart of Memory Access Using ITLB (TLB Compatible Mode)

7.4 TLB Functions (TLB Extended Mode)

7.4.1 Unified TLB (UTLB) Configuration

Figure 7.11 shows the configuration of the UTLB in TLB extended mode. Figure 7.12 shows the relationship between the page size and address format.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
		⋮				⋮				
Entry 63	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT

Figure 7.11 UTLB Configuration (TLB Extended Mode)

Legend:

- ASID: Address space identifier

  Indicates the process that can access a virtual page.

  In single virtual memory mode and user mode, or in multiple virtual memory mode, if the SH bit is "0", this identifier is compared with the ASID in PTEH when address comparison is performed.

- VPN: Virtual page number

  For 1-Kbyte page: Upper 22 bits of virtual address

  For 4-Kbyte page: Upper 20 bits of virtual address

  For 8-Kbyte page: Upper 19 bits of virtual address

  For 64-Kbyte page: Upper 16 bits of virtual address

  For 256-Kbyte page: Upper 14 bits of virtual address

  For 1-Mbyte page: Upper 12 bits of virtual address

  For 4-Mbyte page: Upper 10 bits of virtual address

  For 64-Mbyte page: Upper 6 bits of virtual address

- V: Validity bit

  Indicates whether the entry is valid.

  0: Invalid

  1: Valid

  Cleared to "0" by a hardware reset.

- PPN: Physical page number

  Upper 19 bits of the physical address.

  With a 1-Kbyte page, PPN[28:10] are valid.

  With a 4-Kbyte page, PPN[28:12] are valid.

  With a 8-Kbyte page, PPN[28:13] are valid.

  With a 64-Kbyte page, PPN[28:16] are valid.

  With a 256-Kbyte page, PPN[28:18] are valid.

  With a 1-Mbyte page, PPN[28:20] are valid.

  With a 4-Mbyte page, PPN[28:22] are valid.

  With a 64-Mbyte page, PPN[28:26] are valid.

  The synonym problem must be taken into account when setting the PPN (see section 7.5.5, Avoiding Synonym Problems).

- ESZ: Page size bits

  Specify the page size.

  0000: 1-Kbyte page

  0001: 4-Kbyte page

  0010: 8-Kbyte page

  0100: 64-Kbyte page

  0101: 256-Kbyte page

  0111: 1-Mbyte page

  1000: 4-Mbyte page

  1100: 64-Mbyte page

Note: • When a value other than those listed above is recorded, operation is not guaranteed.

- SH: Share status bit

  When 0, pages are not shared by processes.

  When 1, pages are shared by processes.

- C: Cacheability bit

  Indicates whether a page is cacheable.

  0: Not cacheable

  1: Cacheable

  When the control register area is mapped, this bit must be cleared to "0".

- EPR: Protection key data

  6-bit data expressing the page access right as a code.

  Reading, writing, and execution (instruction fetch) in privileged mode and reading, writing, and execution (instruction fetch) in user mode can be set independently. Each bit is disabled by "0" and enabled by "1".

  EPR[5]: Reading in privileged mode

  EPR[4]: Writing in privileged mode

  EPR[3]: Execution in privileged mode (instruction fetch)

  EPR[2]: Reading in user mode

  EPR[1]: Writing in user mode

  EPR[0]: Execution in user mode (instruction fetch)

- D: Dirty bit

  Indicates whether a write has been performed to a page.

  0: Write has not been performed.

  1: Write has been performed.

- WT: Write-through bit

  Specifies the cache write mode.

  0: Copy-back mode

  1: Write-through mode

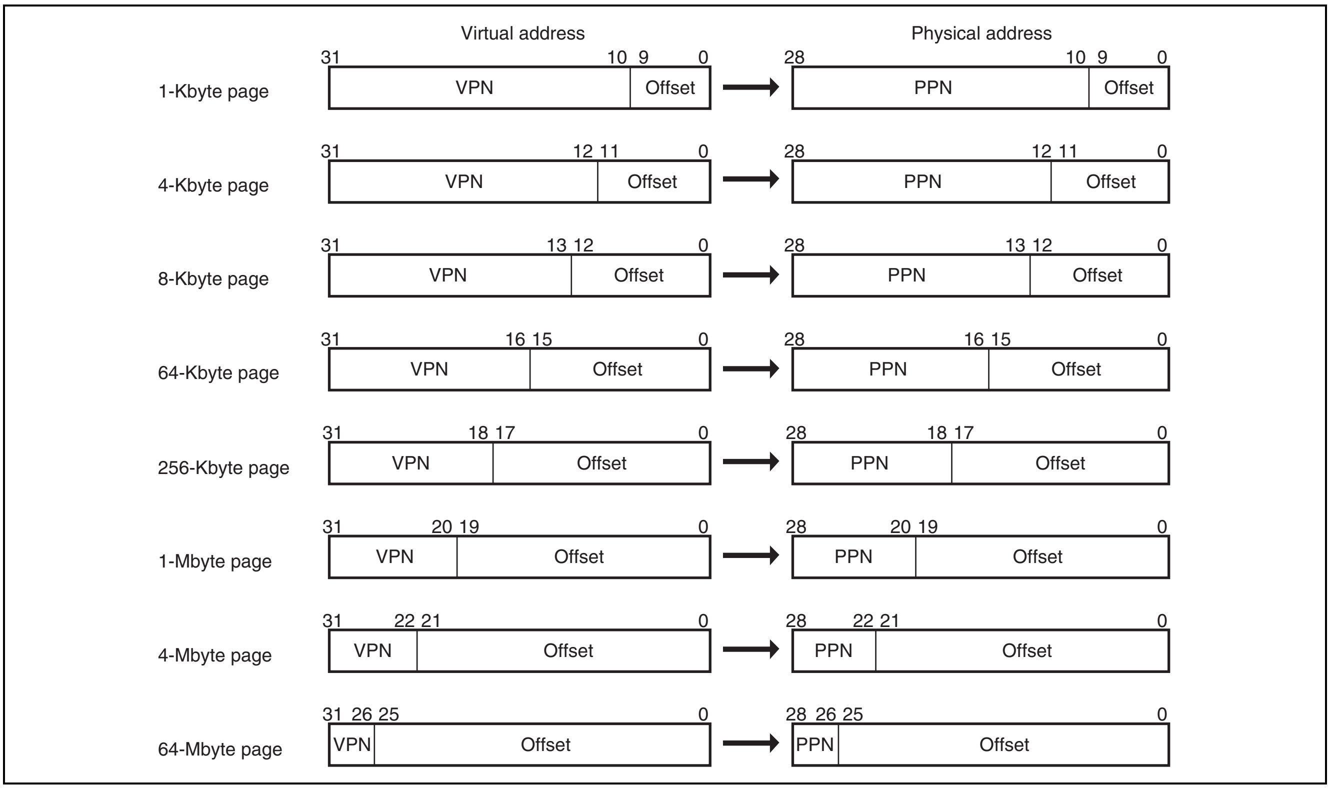


Figure 7.12 Relationship between Page Size and Address Format (TLB Extended Mode)

7.4.2 Instruction TLB (ITLB) Configuration

Figure 7.13 shows the configuration of the ITLB in TLB extended mode.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 3	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]

Note: • Bits EPR[4], EPR[1], D, and WT are not supported.

Figure 7.13 ITLB Configuration (TLB Extended Mode)

7.4.3 Address Translation Method

Figure 7.14 is a flowchart of memory access using the UTLB in TLB extended mode.

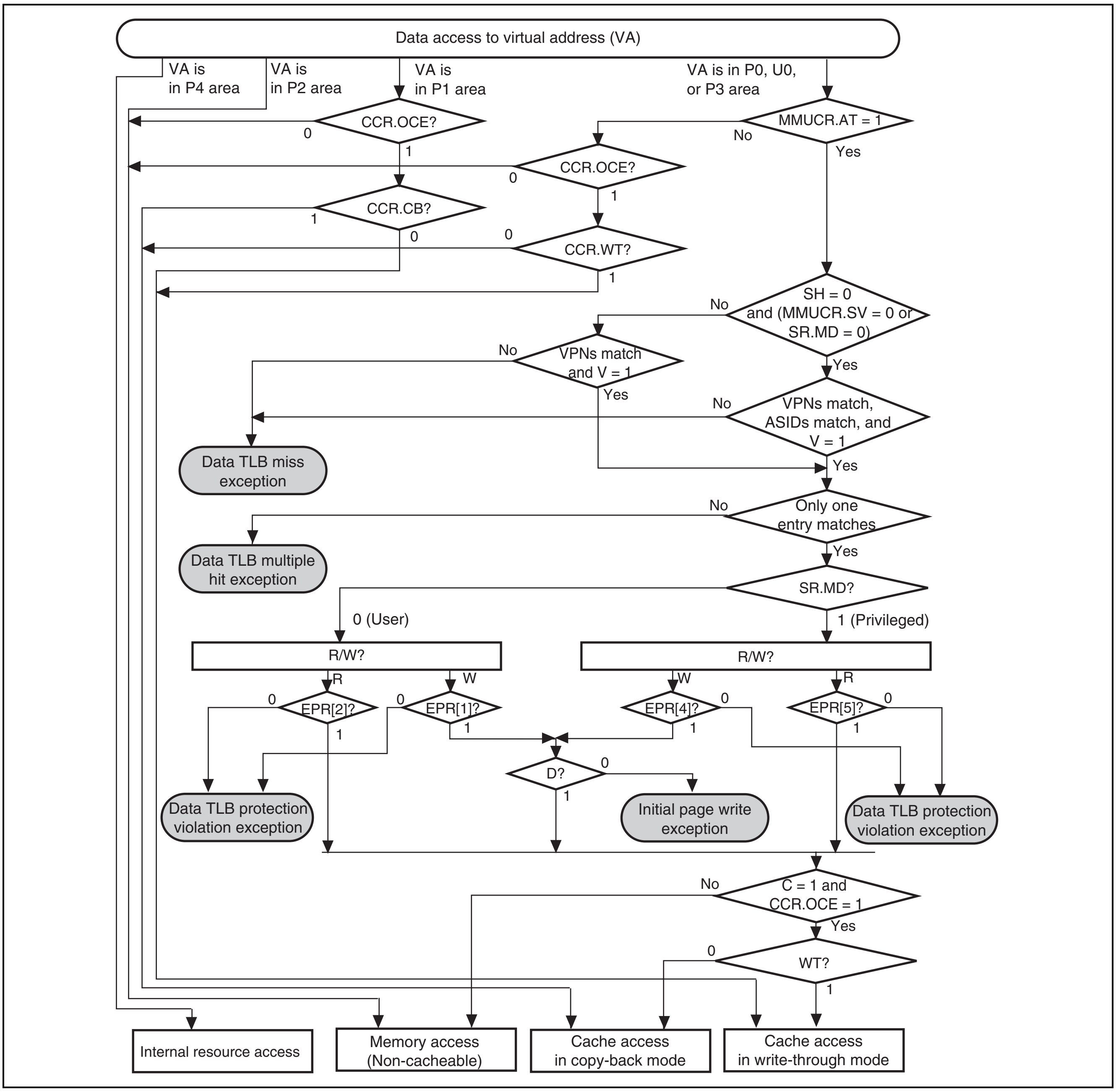

Data access to virtual address (VA)
VA is in P4 area
VA is in P2 area
VA is in P1 area
VA is in P0, U0, or P3 area
CCR.OCE?
0
1
MMUCR.AT = 1
No
Yes
CCR.OCE?
0
1
CCR.CB?
1
0
CCR.WT?
0
1
SH = 0 and (MMUCR.SV = 0 or SR.MD = 0)
No
Yes
VPNs match and V = 1
No
Yes
VPNs match, ASIDs match, and V = 1
No
Yes
Data TLB miss exception
Only one entry matches
No
Yes
Data TLB multiple hit exception
SR.MD?
0 (User)
1 (Privileged)
R/W?
R
W
R/W?
W
R
EPR[2]?
0
1
EPR[1]?
0
1
EPR[4]?
0
1
EPR[5]?
0
1
D?
0
1
Data TLB protection violation exception
Initial page write exception
Data TLB protection violation exception
C = 1 and CCR.OCE = 1
No
Yes
WT?
0
1
Internal resource access
Memory access (Non-cacheable)
Cache access in copy-back mode
Cache access in write-through mode

Figure 7.14 Flowchart of Memory Access Using UTLB (TLB Extended Mode)

Figure 7.15 is a flowchart of memory access using the ITLB in TLB extended mode.

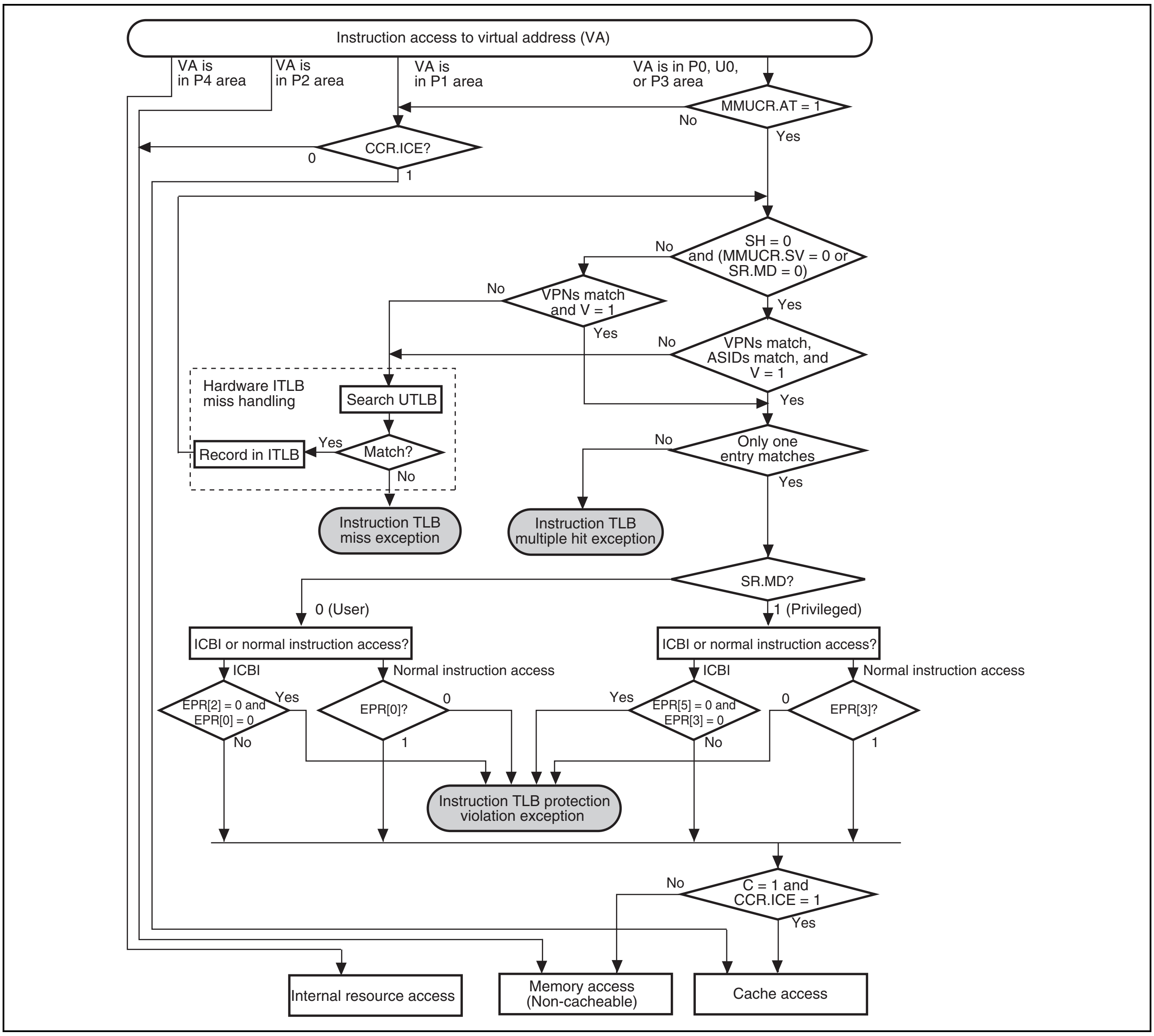

Instruction access to virtual address (VA)
VA is in P4 area
VA is in P2 area
VA is in P1 area
VA is in P0, U0, or P3 area
MMUCR.AT = 1
No
Yes
CCR.ICE?
0
1
SH = 0 and (MMUCR.SV = 0 or SR.MD = 0)
No
Yes
VPNs match and V = 1
No
Yes
VPNs match, ASIDs match, and V = 1
No
Yes
Hardware ITLB miss handling
Search UTLB
Match?
Yes
No
Record in ITLB
Instruction TLB miss exception
Only one entry matches
No
Yes
Instruction TLB multiple hit exception
SR.MD?
0 (User)
1 (Privileged)
ICBI or normal instruction access?
ICBI
Normal instruction access
EPR[2] = 0 and EPR[0] = 0
Yes
No
EPR[0]?
0
1
ICBI or normal instruction access?
ICBI
Normal instruction access
EPR[5] = 0 and EPR[3] = 0
Yes
No
EPR[3]?
0
1
Instruction TLB protection violation exception
C = 1 and CCR.ICE = 1
No
Yes
Internal resource access
Memory access (Non-cacheable)
Cache access

Figure 7.15 Flowchart of Memory Access Using ITLB (TLB Extended Mode)

7.5 MMU Functions

7.5.1 MMU Hardware Management

The SH-4A supports the following MMU functions.

1. The MMU decodes the virtual address to be accessed by software, and performs address translation by controlling the UTLB/ITLB in accordance with the MMUCR settings.
2. The MMU determines the cache access status on the basis of the page management information read during address translation (C and WT bits).
3. If address translation cannot be performed normally in a data access or instruction access, the MMU notifies software by means of an MMU exception.
4. If address translation information is not recorded in the ITLB in an instruction access, the MMU searches the UTLB. If the necessary address translation information is recorded in the UTLB, the MMU copies this information into the ITLB in accordance with the LRUI bit setting in MMUCR.

7.5.2 MMU Software Management

Software processing for the MMU consists of the following:

1. Setting of MMU-related registers. Some registers are also partially updated by hardware automatically.
2. Recording, deletion, and reading of TLB entries. There are two methods of recording UTLB entries: by using the LDTLB instruction, or by writing directly to the memory-mapped UTLB. ITLB entries can only be recorded by writing directly to the memory-mapped ITLB. Deleting or reading UTLB/ITLB entries is enabled by accessing the memory-mapped UTLB/ITLB.
3. MMU exception handling. When an MMU exception occurs, processing is performed based on information set by hardware.

7.5.3 MMU Instruction (LDTLB)

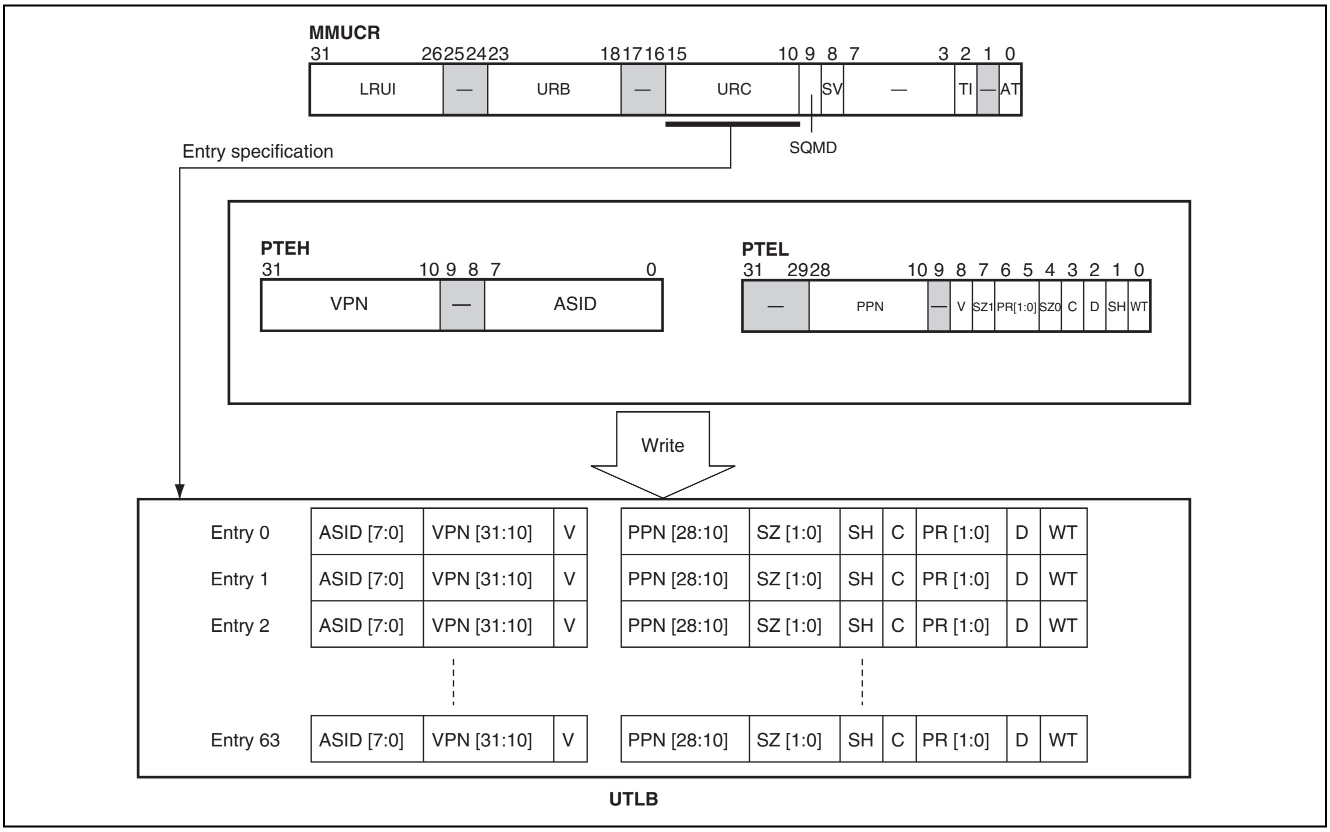
A TLB load instruction (LDTLB) is provided for recording UTLB entries. When an LDTLB instruction is issued, the SH-4A copies the contents of the PTEH and PTEL registers (as well as the contents of PTEA in TLB extended mode) to the UTLB entry indicated by the URC bit in MMUCR. ITLB entries are not updated by the LDTLB instruction, and therefore address translation information purged from the UTLB entry may still remain in the ITLB entry. As the LDTLB instruction changes address translation information, ensure that it is issued by a program in the P1 or P2 area.

After the LDTLB instruction has been executed, execute one of the following three methods before an access (include an instruction fetch) the area where TLB is used to translate the address is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the area where TLB is used to translate the address.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the LT bit in IRMCR is "0" (the value after a reset) before executing the LDTLB instruction, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

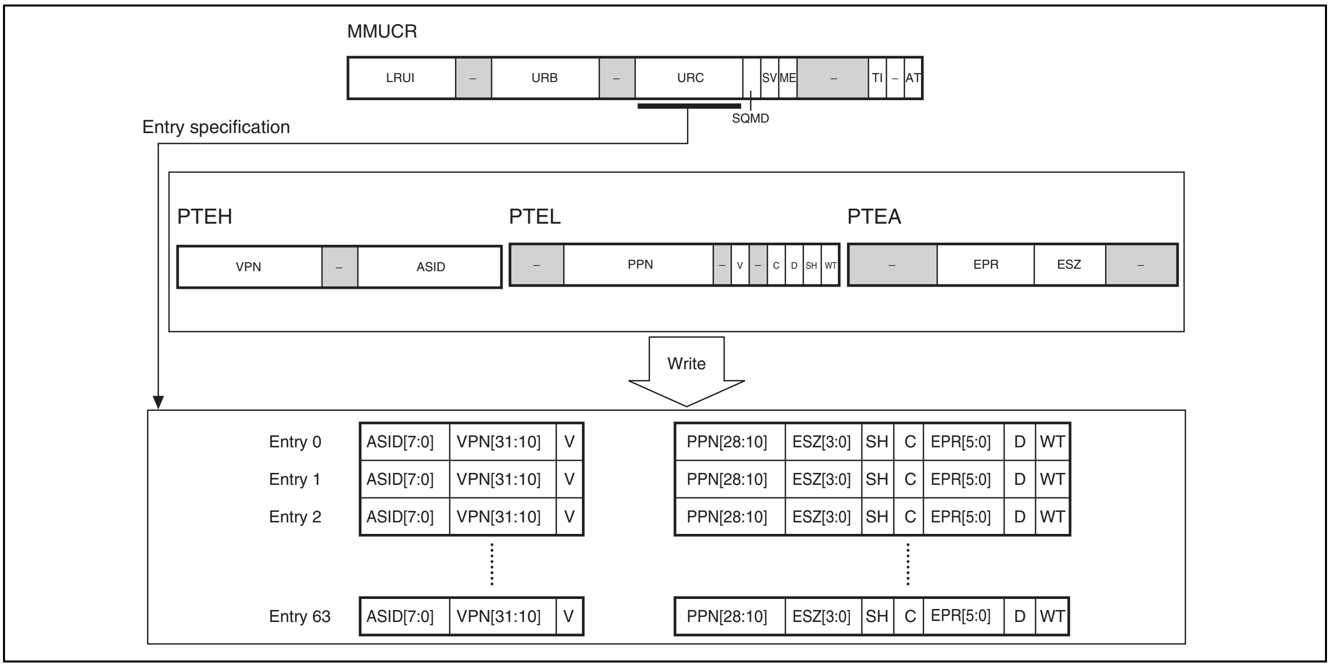
Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

The operation of the LDTLB instruction is shown in figures 7.16 and 7.17.



Entry 0	ASID [7:0]	VPN [31:10]	V	PPN [28:10]	SZ [1:0]	SH	C	PR [1:0]	D	WT
Entry 1	ASID [7:0]	VPN [31:10]	V	PPN [28:10]	SZ [1:0]	SH	C	PR [1:0]	D	WT
Entry 2	ASID [7:0]	VPN [31:10]	V	PPN [28:10]	SZ [1:0]	SH	C	PR [1:0]	D	WT
Entry 63	ASID [7:0]	VPN [31:10]	V	PPN [28:10]	SZ [1:0]	SH	C	PR [1:0]	D	WT

Figure 7.16 Operation of LDTLB Instruction (TLB Compatible Mode)



Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
Entry 63	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT

Figure 7.17 Operation of LDTLB Instruction (TLB Extended Mode)

7.5.4 Hardware ITLB Miss Handling

In an instruction access, the SH-4A searches the ITLB. If it cannot find the necessary address translation information (ITLB miss occurred), the UTLB is searched by hardware, and if the necessary address translation information is present, it is recorded in the ITLB. This procedure is known as hardware ITLB miss handling. If the necessary address translation information is not found in the UTLB search, an instruction TLB miss exception is generated and processing passes to software.

7.5.5 Avoiding Synonym Problems

When information on 1- or 4-Kbyte pages is written as TLB entries, a synonym problem may arise. The problem is that, when a number of virtual addresses are mapped onto a single physical address, the same physical address data is written to a number of cache entries, and it becomes impossible to guarantee data integrity. This problem does not occur with the instruction TLB and instruction cache because only data is read in these cases. In the SH-4A, entry specification is performed using bits 12 to 5 of the virtual address in order to achieve fast operand cache operation. However, bits 12 to 10 of the virtual address in the case of a 1-Kbyte page, and bit 12 of the virtual address in the case of a 4-Kbyte page, are subject to address translation. As a result, bits 12 to 10 of the physical address after translation may differ from bits 12 to 10 of the virtual address.

Consequently, the following restrictions apply to the writing of address translation information as UTLB entries.

1. When address translation information whereby a number of 1-Kbyte page UTLB entries are translated into the same physical address is written to the UTLB, ensure that the VPN[12:10] values are the same.
2. When address translation information whereby a number of 4-Kbyte page UTLB entries are translated into the same physical address is written to the UTLB, ensure that the VPN[12] value is the same.
3. Do not use 1-Kbyte page UTLB entry physical addresses with UTLB entries of a different page size.
4. Do not use 4-Kbyte page UTLB entry physical addresses with UTLB entries of a different page size.

The above restrictions apply only when performing accesses using the cache.

When the cache size is changed from 32 Kbytes to some other size, both the page sizes for which the synonym problem can occur and the VPN bit positions that are required to be made the same will differ from the above description. Table 7.2 lists the page sizes for which the synonym problem may occur for cache sizes from 8 Kbytes to 64 Kbytes.

Table 7.2 Cache Size and Synonym Problem Avoidance Measures

Cache Size	Page Sizes for which the Synonym Problem may Occur	VPN Bit Positions that are Recorded so as to be Equal
8 Kbytes	1-Kbyte pages	VPN[10]
16 Kbytes	1-Kbyte pages	VPN[11:10]
32 Kbytes	1-Kbyte pages	VPN[12:10]
	4-Kbyte pages	VPN[12]
64 Kbytes	1-Kbyte pages	VPN[13:10]
	4-Kbyte pages	VPN[13:12]

Note: • For the future expansion of the SuperH RISC engine family, when the same physical memory is used for the address translation information of a number of addresses, ensure that VPN[20:10] values are the same. Do not use the same physical address with the address translation information of a different page size.

7.6 MMU Exceptions

There are seven MMU exceptions: instruction TLB multiple hit exception, instruction TLB miss exception, instruction TLB protection violation exception, data TLB multiple hit exception, data TLB miss exception, data TLB protection violation exception, and initial page write exception. Refer to figures 7.9, 7.10, 7.14, 7.15, and section 5, Exception Handling for the conditions under which each of these exceptions occurs.

7.6.1 Instruction TLB Multiple Hit Exception

An instruction TLB multiple hit exception occurs when more than one ITLB entry matches the virtual address to which an instruction access has been made. If multiple hits occur when the UTLB is searched by hardware in hardware ITLB miss handling, an instruction TLB multiple hit exception will result.

When an instruction TLB multiple hit exception occurs, a reset is executed and cache coherency is not guaranteed.

(1) Hardware Processing

In the event of an instruction TLB multiple hit exception, hardware carries out the following processing:

1. Sets the virtual address at which the exception occurred in TEA.
2. Sets exception code H'000 in EXPEVT.
3. Branches to the reset handling routine (H'A000 0000).

(2) Software Processing (Reset Routine)

The ITLB entries which caused the multiple hit exception are checked in the reset handling routine. This exception is intended for use in program debugging, and should not normally be generated.

7.6.2 Instruction TLB Miss Exception

An instruction TLB miss exception occurs when address translation information for the virtual address to which an instruction access is made is not found in the UTLB entries by the hardware ITLB miss handling routine. The instruction TLB miss exception processing carried out by hardware and software is shown below. This is the same as the processing for a data TLB miss exception.

(1) Hardware Processing

In the event of an instruction TLB miss exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'040 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to "1", and switches to privileged mode.
7. Sets the BL bit in SR to "1", and masks subsequent exception requests.
8. Sets the RB bit in SR to "1".
9. Branches to the address obtained by adding offset H'0000 0400 to the contents of VBR, and starts the instruction TLB miss exception handling routine.

(2) Software Processing (Instruction TLB Miss Exception Handling Routine)

Software is responsible for searching the external memory page table and assigning the necessary page table entry. Software should carry out the following processing in order to find and assign the necessary page table entry.

1. In TLB compatible mode, write to the PTEL register the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory. In TLB extended mode, write to the PTEL and PTEA registers the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
2. When the entry to be replaced in entry replacement is specified by software, write the value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.
3. In TLB compatible mode, execute the LDTLB instruction and write the contents of the PTEH and PTEL registers to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of the PTEH, PTEL, and PTEA registers to the UTLB.
4. Finally, execute the exception handling return instruction (RTE) to terminate the exception handling routine and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.6.3 Instruction TLB Protection Violation Exception

An instruction TLB protection violation exception occurs when, even though an ITLB entry contains address translation information matching the virtual address to which an instruction access is made, the actual access type is not permitted by the access right specified by the PR or EPR bit. The instruction TLB protection violation exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of an instruction TLB protection violation exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'0A0 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to "1", and switches to privileged mode.
7. Sets the BL bit in SR to "1", and masks subsequent exception requests.
8. Sets the RB bit in SR to "1".
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the instruction TLB protection violation exception handling routine.

(2) Software Processing (Instruction TLB Protection Violation Exception Handling Routine)

Resolve the instruction TLB protection violation, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.6.4 Data TLB Multiple Hit Exception

A data TLB multiple hit exception occurs when more than one UTLB entry matches the virtual address to which a data access has been made.

When a data TLB multiple hit exception occurs, a reset is executed, and cache coherency is not guaranteed. The contents of PPN in the UTLB prior to the exception may also be corrupted.

(1) Hardware Processing

In the event of a data TLB multiple hit exception, hardware carries out the following processing:

1. Sets the virtual address at which the exception occurred in TEA.
2. Sets exception code H'000 in EXPEVT.
3. Branches to the reset handling routine (H'A000 0000).

(2) Software Processing (Reset Routine)

The UTLB entries which caused the multiple hit exception are checked in the reset handling routine. This exception is intended for use in program debugging, and should not normally be generated.

7.6.5 Data TLB Miss Exception

A data TLB miss exception occurs when address translation information for the virtual address to which a data access is made is not found in the UTLB entries. The data TLB miss exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of a data TLB miss exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'040 in the case of a read, or H'060 in the case of a write in EXPEVT (OCBP, OCBWB: read; OCBI, MOVCA.L: write).
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to "1", and switches to privileged mode.
7. Sets the BL bit in SR to "1", and masks subsequent exception requests.
8. Sets the RB bit in SR to "1".
9. Branches to the address obtained by adding offset H'0000 0400 to the contents of VBR, and starts the data TLB miss exception handling routine.

(2) Software Processing (Data TLB Miss Exception Handling Routine)

Software is responsible for searching the external memory page table and assigning the necessary page table entry. Software should carry out the following processing in order to find and assign the necessary page table entry.

1. In TLB compatible mode, write to the PTEL register the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to the PTEL and PTEA registers the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
2. When the entry to be replaced in entry replacement is specified by software, write the value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.
3. In TLB compatible mode, execute the LDTLB instruction and write the contents of the PTEH and PTEL registers to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of the PTEH, PTEL, and PTEA registers to the UTLB.
4. Finally, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.6.6 Data TLB Protection Violation Exception

A data TLB protection violation exception occurs when, even though a UTLB entry contains address translation information matching the virtual address to which a data access is made, the actual access type is not permitted by the access right specified by the PR or EPR bit. The data TLB protection violation exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of a data TLB protection violation exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'0A0 in the case of a read, or H'0C0 in the case of a write in EXPEVT (OCBP, OCBWB: read; OCBI, MOVCA.L: write).
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to "1", and switches to privileged mode.
7. Sets the BL bit in SR to "1", and masks subsequent exception requests.
8. Sets the RB bit in SR to "1".
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the data TLB protection violation exception handling routine.

(2) Software Processing (Data TLB Protection Violation Exception Handling Routine)

Resolve the data TLB protection violation, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.6.7 Initial Page Write Exception

An initial page write exception occurs when the D bit is "0" even though a UTLB entry contains address translation information matching the virtual address to which a data access (write) is made, and the access is permitted. The initial page write exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of an initial page write exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'080 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to "1", and switches to privileged mode.
7. Sets the BL bit in SR to "1", and masks subsequent exception requests.
8. Sets the RB bit in SR to "1".
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the initial page write exception handling routine.

(2) Software Processing (Initial Page Write Exception Handling Routine)

Software is responsible for the following processing:

1. Retrieve the necessary page table entry from external memory.
2. Write "1" to the D bit in the external memory page table entry.
3. In TLB compatible mode, write to the PTEL register the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to the PTEL and PTEA registers the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
4. When the entry to be replaced in entry replacement is specified by software, write that value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.
5. In TLB compatible mode, execute the LDTLB instruction and write the contents of the PTEH and PTEL registers to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of the PTEH, PTEL, and PTEA registers to the UTLB.
6. Finally, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.7 Memory-Mapped TLB Configuration

To enable the ITLB and UTLB to be managed by software, their contents are allowed to be read from and written to by a program in the P1/P2 area with a MOV instruction in privileged mode. Operation is not guaranteed if access is made from a program in another area.

After the memory-mapped TLB has been accessed, execute one of the following three methods before an access (including an instruction fetch) to an area other than the P1/P2 area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be an area other than the P1/P2 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the MT bit in IRMCR is "0" (the value after a reset) before accessing the memory-mapped TLB, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

The ITLB and UTLB are allocated to the P4 area in the virtual address space.

In TLB compatible mode, VPN, V, and ASID in the ITLB can be accessed as an address array, PPN, V, SZ, PR, C, and SH as a data array. VPN, D, V, and ASID in the UTLB can be accessed as an address array, PPN, V, SZ, PR, C, D, WT, and SH as a data array. V and D can be accessed from both the address array side and the data array side.

In TLB extended mode, VPN, V, and ASID in the ITLB can be accessed as an address array, PPN, V, ESZ, EPR, C, and SH as a data array. VPN, D, V, and ASID in the UTLB can be accessed as an address array, PPN, V, ESZ, EPR, C, D, WT, and SH as a data array. V and D can be accessed from both the address array side and the data array side.

In both TLB compatible mode and TLB extended mode, only longword access is possible. Instruction fetches cannot be performed in these areas. For reserved bits, a write value of "0" should be specified; their read value is undefined.

7.7.1 ITLB Address Array

The ITLB address array is allocated to addresses H'F200 0000 to H'F2FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and VPN, V, and ASID to be written to the address array are specified in the data field.

In the address field, bits [31:24] have the value H'F2 indicating the ITLB address array and the entry is specified by bits [9:8]. As only longword access is used, "0" should be specified for address field bits [1:0].

In the data field, bits [31:10] indicate VPN, bit [8] indicates V, and bits [7:0] indicate ASID.

The following two kinds of operation can be used on the ITLB address array:

1. ITLB address array read

   VPN, V, and ASID are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB address array write

   VPN, V, and ASID specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

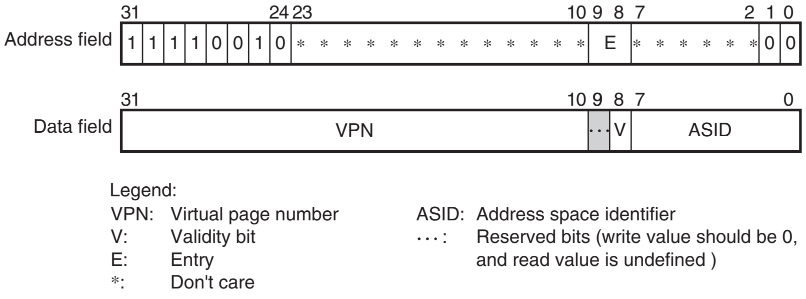


Figure 7.18 Memory-Mapped ITLB Address Array

7.7.2 ITLB Data Array (TLB Compatible Mode)

The ITLB data array is allocated to addresses H'F300 0000 to H'F37F FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and PPN, V, SZ, PR, C, and SH to be written to the data array are specified in the data field.

In the address field, bits [31:23] have the value H'F30 indicating ITLB data array and the entry is specified by bits [9:8].

In the data field, bits [28:10] indicate PPN, bit [8] indicates V, bits [7] and [4] indicate SZ, bit [6] indicates PR, bit [3] indicates C, and bit [1] indicates SH.

The following two kinds of operation can be used on ITLB data array:

1. ITLB data array read

   PPN, V, SZ, PR, C, and SH are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB data array write

   PPN, V, SZ, PR, C, and SH specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

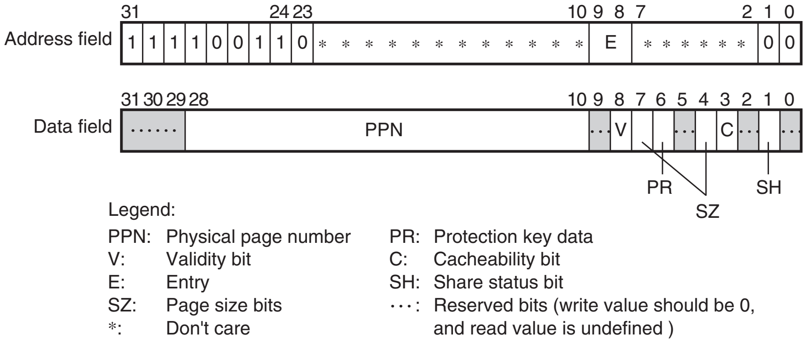


Figure 7.19 Memory-Mapped ITLB Data Array (TLB Compatible Mode)

7.7.3 ITLB Data Array (TLB Extended Mode)

In TLB extended mode the names of the data arrays have been changed from ITLB data array to ITLB data array 1, ITLB data array 2 is added, and the EPR and ESZ bits are accessible. In TLB extended mode, the PR and SZ bits of ITLB data array 1 are reserved and 0 should be specified as the write value for these bits. In addition, after a write to ITLB data array 1 is performed, a write to ITLB data array 2 of the same entry should always be performed.

In TLB compatible mode (MMUCR.ME = 0), ITLB data array 2 cannot be accessed. Operation if it is accessed is not guaranteed.

(1) ITLB Data Array 1

In TLB extended mode, bits 7, 6, and 4 in the data field, which correspond to the PR and SZ bits in compatible mode, are reserved. Specify 0 as the write value for these bits.

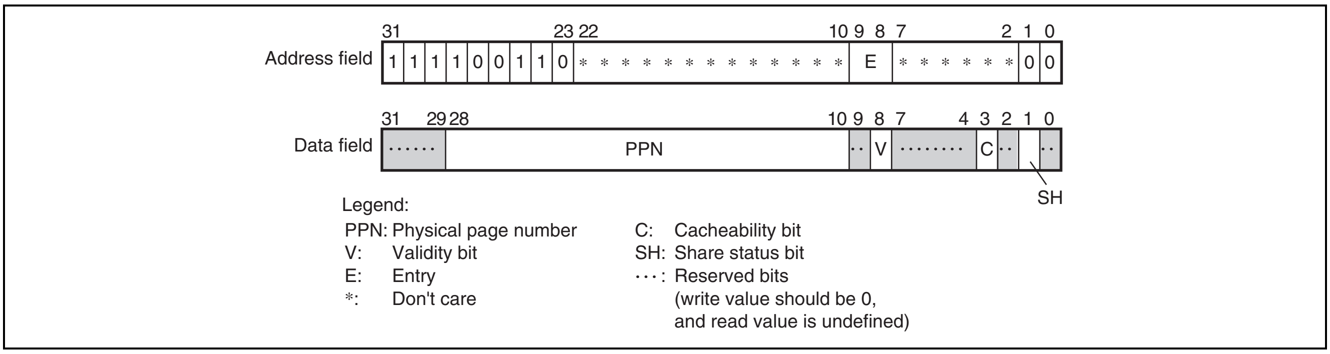

Address field
31
23 22
10 9 8 7
2 1 0
1 1 1 1 0 0 1 1 0 * * * * * * * * * * * * * E * * * * * * 0 0
Data field
31 29 28
10 9 8 7
4 3 2 1 0
PPN
V
C
SH
Legend:
PPN: Physical page number
V: Validity bit
E: Entry
*: Don't care
C: Cacheability bit
SH: Share status bit
···: Reserved bits (write value should be 0, and read value is undefined)

Figure 7.20 Memory-Mapped ITLB Data Array 1 (TLB Extended Mode)

(2) ITLB Data Array 2

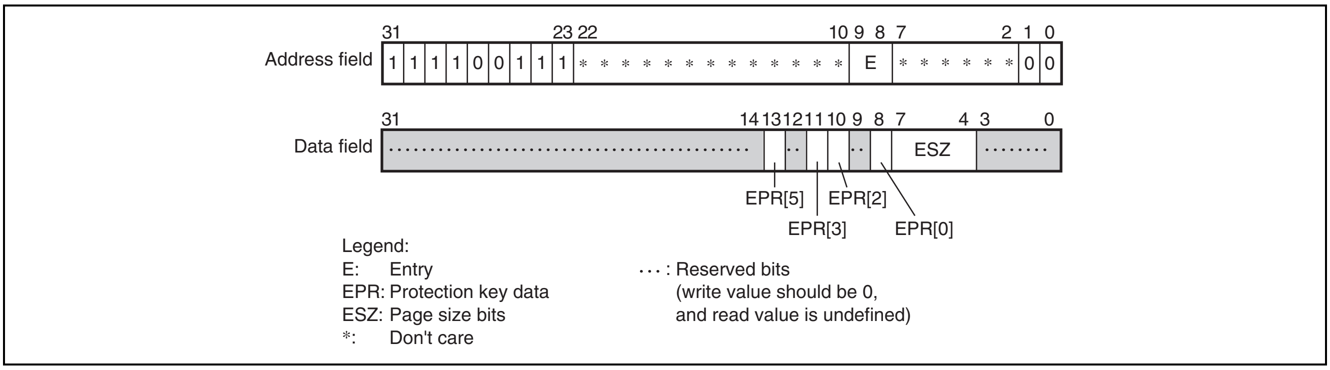
The ITLB data array is allocated to addresses H'F380 0000 to H'F3FF FFFF in the P4 area. Access to data array 2 requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and EPR and ESZ to be written to data array 2 are specified in the data field.

In the address field, bits [31:23] have the value H'F38 indicating ITLB data array 2 and the entry is specified by bits [9:8].

In the data field, bits [13], [11], [10], and [8] indicate EPR[5], [3], [2], and [0], and bits [7:4] indicate ESZ, respectively.

The following two kinds of operation can be applied to ITLB data array 2:

1. ITLB data array 2 read
   EPR and ESZ are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB data array 2 write
   EPR and ESZ specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.



Legend:
E: Entry
EPR: Protection key data
ESZ: Page size bits
*: Don't care
· · · : Reserved bits (write value should be 0, and read value is undefined)

Figure 7.21 Memory-Mapped ITLB Data Array 2 (TLB Extended Mode)

7.7.4 UTLB Address Array

The UTLB address array is allocated to addresses H'F600 0000 to H'F6FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and VPN, D, V, and ASID to be written to the address array are specified in the data field.

In the address field, bits [31:20] have the value H'F60 indicating the UTLB address array and the entry is specified by bits [13:8]. Bit [7] that is the association bit (A bit) in the address field specifies whether address comparison is performed in a write to the UTLB address array.

In the data field, bits [31:10] indicate VPN, bit [9] indicates D, bit [8] indicates V, and bits [7:0] indicate ASID.

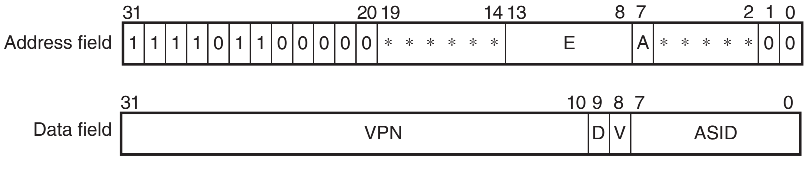
The following three kinds of operation can be used on the UTLB address array:

1. UTLB address array read

   VPN, D, V, and ASID are read into the data field from the UTLB entry corresponding to the entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is "1" or "0".
2. UTLB address array write (non-associative)

   VPN, D, V, and ASID specified in the data field are written to the UTLB entry corresponding to the entry set in the address field. The A bit in the address field should be cleared to "0".
3. UTLB address array write (associative)

   When a write is performed with the A bit in the address field set to "1", comparison of all the UTLB entries is carried out using the VPN specified in the data field and ASID in PTEH. The usual address comparison rules are followed, but if a UTLB miss occurs, the result is no operation, and an exception is not generated. If the comparison identifies a UTLB entry corresponding to the VPN specified in the data field, D and V specified in the data field are written to that entry. This associative operation is simultaneously carried out on the ITLB, and if a matching entry is found in the ITLB, V is written to that entry. Even if the UTLB comparison results in no operation, a write to the ITLB is performed as long as a matching entry is found in the ITLB. If there is a match in both the UTLB and ITLB, the UTLB information is also written to the ITLB.



Legend:

VPN:	Virtual page number	ASID:	Address space identifier
V:	Validity bit	A:	Association bit
E:	Entry	⋯ :	Reserved bits (write value should be 0, and read value is undefined)
D:	Dirty bit		
*:	Don't care		

Figure 7.22 Memory-Mapped UTLB Address Array

7.7.5 UTLB Data Array (TLB Compatible Mode)

The UTLB data array is allocated to addresses H'F700 0000 to H'F7FF FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and PPN, V, SZ, PR, C, D, SH, and WT to be written to data array are specified in the data field.

In the address field, bits [31:20] have the value H'F70 indicating UTLB data array and the entry is specified by bits [13:8].

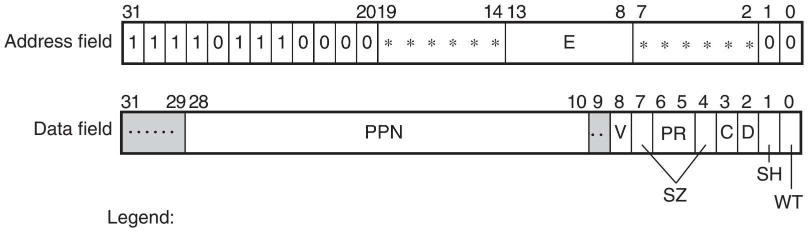
In the data field, bits [28:10] indicate PPN, bit [8] indicates V, bits [7] and [4] indicate SZ, bits [6:5] indicate PR, bit [3] indicates C, bit [2] indicates D, bit [1] indicates SH, and bit [0] indicates WT.

The following two kinds of operation can be used on UTLB data array:

1. UTLB data array read

   PPN, V, SZ, PR, C, D, SH, and WT are read into the data field from the UTLB entry corresponding to the entry set in the address field.
2. UTLB data array write

   PPN, V, SZ, PR, C, D, SH, and WT specified in the data field are written to the UTLB entry corresponding to the entry set in the address field.



Legend:

PPN:	Physical page number	PR:	Protection key data
V:	Validity bit	C:	Cacheability bit
E:	Entry	SH:	Share status bit
SZ:	Page size bits	WT:	Write-through bit
D:	Dirty bit	⋯ :	Reserved bits (write value should be 0, and read value is undefined)
*:	Don't care		

Figure 7.23 Memory-Mapped UTLB Data Array (TLB Compatible Mode)

7.7.6 UTLB Data Array (TLB Extended Mode)

In TLB extended mode, the names of the data arrays have been changed from UTLB data array to UTLB data array 1, UTLB data array 2 is added, and the EPR and ESZ bits are accessible. In TLB extended mode, the PR and SZ bits of UTLB data array 1 are reserved and 0 should be specified as the write value for these bits. In addition, when a write to UTLB data array 1 is performed, a write to UTLB data array 2 of the same entry should always be performed after that.

In TLB compatible mode (MMUCR.ME = 0), UTLB data array 2 cannot be accessed. Operation if they are accessed is not guaranteed.

(1) UTLB Data Array 1

In TLB extended mode, bits 7 to 4 in the data field, which correspond to the PR and SZ bits in compatible mode, are reserved. Specify 0 as the write value for these bits.

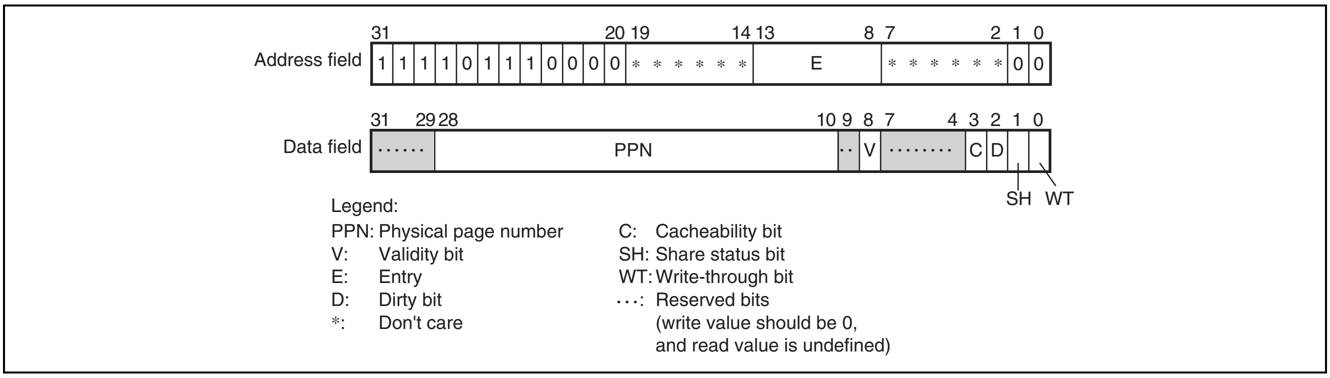


Figure 7.24 Memory-Mapped UTLB Data Array 1 (TLB Extended Mode)

(2) UTLB Data Array 2

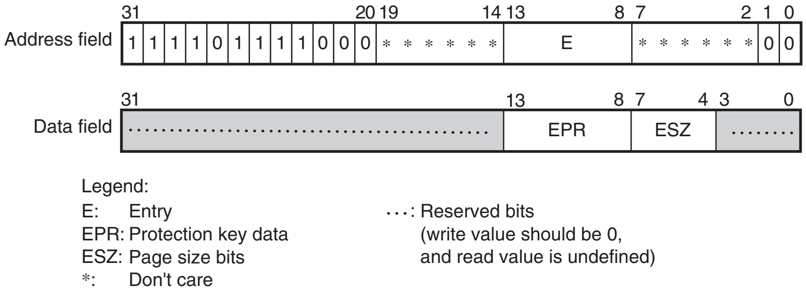
The UTLB data array is allocated to addresses H'F780 0000 to H'F7FF FFFF in the P4 area. Access to data array 2 requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and EPR and ESZ to be written to data array 2 are specified in the data field.

In the address field, bits [31:20] have the value H'F78 indicating UTLB data array 2 and the entry is specified by bits [13:8].

In the data field, bits [13:8] indicate EPR, and bits [7:4] indicate ESZ, respectively.

The following two kinds of operation can be applied to UTLB data array 2:

1. UTLB data array 2 read
   EPR and ESZ are read into the data field from the UTLB entry corresponding to the entry set in the address field.
2. UTLB data array 2 write
   EPR and ESZ specified in the data field are written to the UTLB entry corresponding to the entry set in the address field.



Legend:
E: Entry
EPR: Protection key data
ESZ: Page size bits
*: Don't care
. . . : Reserved bits
(write value should be 0,
and read value is undefined)

Figure 7.25 Memory-Mapped UTLB Data Array 2 (TLB Extended Mode)

This page is blank for reasons of layout.

Section 8 Caches

The SH-4A has an on-chip 32-Kbyte instruction cache (IC) for instructions and an on-chip 32-Kbyte operand cache (OC) for data.

8.1 Overview

Table 8.1 shows the overview of caches.

The SH-4A supports two 32-byte store queues (SQs) to perform high-speed writes to external memory. The overview of the store queues is given in table 8.2.

Table 8.1 Overview of Cache

Item	Instruction Cache	Operand Cache
Capacity	32-Kbyte cache	32-Kbyte cache
Type	4-way set-associative, virtual address index/physical address tag	4-way set-associative, virtual address index/physical address tag
Line size	32 bytes	32 bytes
Entries	256 entries/way	256 entries/way
Write method	—	Copy-back/write-through selectable
Replacement method	LRU (least-recently-used) algorithm	LRU (least-recently-used) algorithm

Table 8.2 Overview of Store Queue

Item	Store Queues
Capacity	32 bytes × 2
Addresses	H'E000 0000 to H'E3FF FFFF
Write	Store instruction (1-cycle write)
Write-back	Prefetch instruction (PREF instruction)
Access right	When MMU is disabled: Determined by SQMD bit in MMUCR When MMU is enabled: Determined by PR for each page

The operand cache of the SH-4A is 4-way set associative, each may comprising 256 cache lines. Figure 8.1 shows the configuration of the operand cache.

The instruction cache is 4-way set-associative, each way comprising 256 cache lines. Figure 8.2 shows the configuration of the instruction cache.

The SH-4A has an IC way prediction scheme to reduce power consumption. In addition, memory-mapped associative writing, which is detectable as an exception, can be enabled by using the unsupported function detection exception register (EXPMASK). For details, see section 5, Exception Handling.

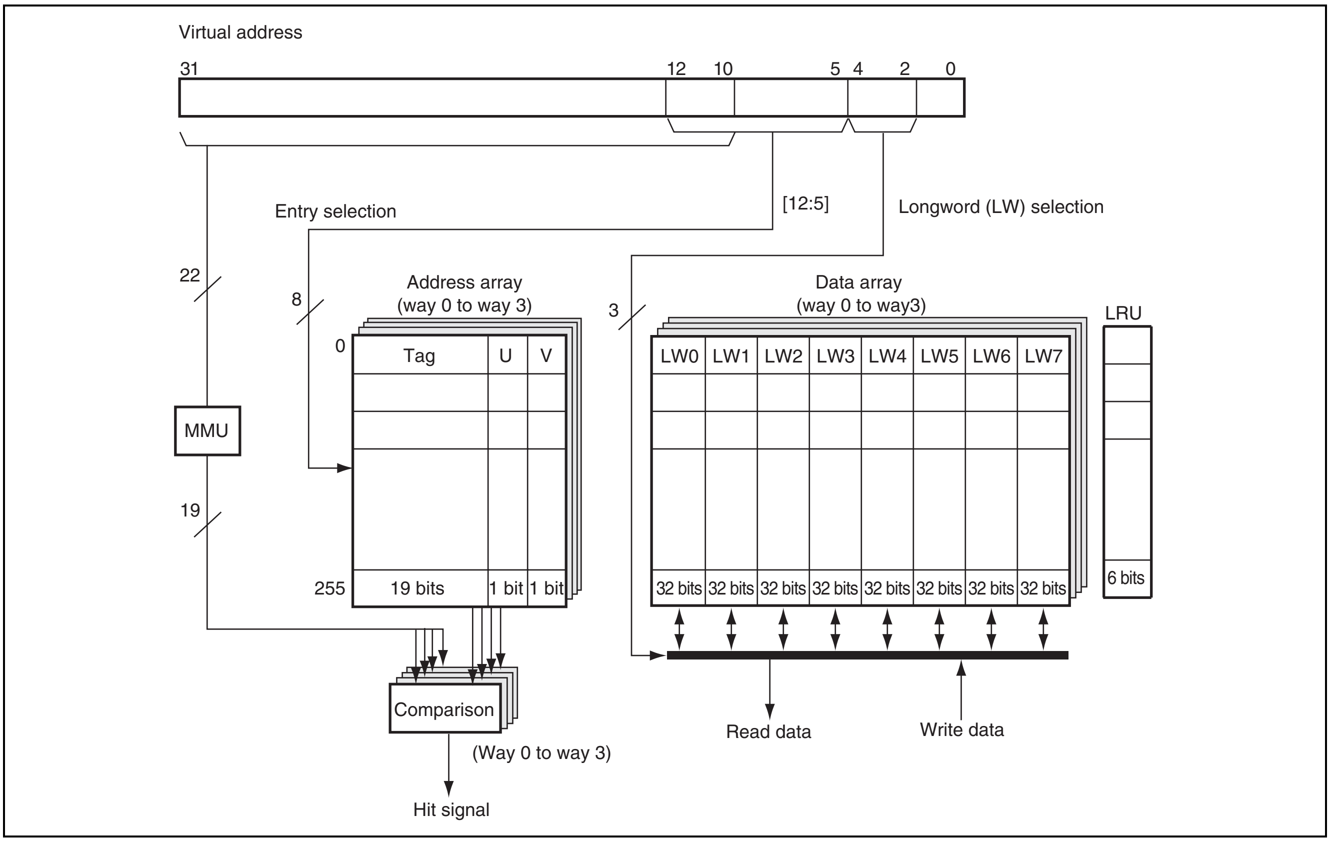


Figure 8.1 Configuration of Operand Cache (Cache size = 32 Kbytes)

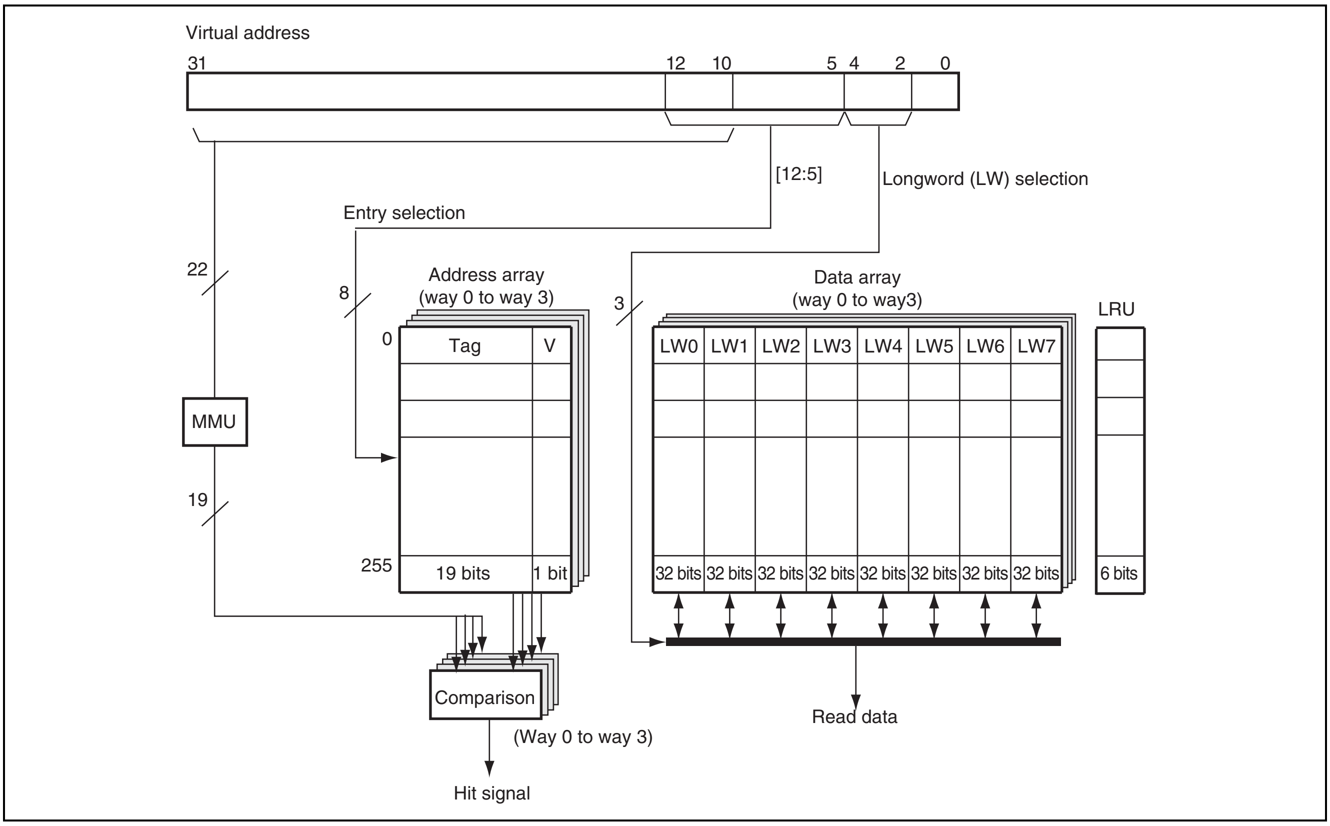


Figure 8.2 Configuration of Instruction Cache (Cache size = 32 Kbytes)

(1) Tag

Stores the upper 19 bits of the 29-bit physical address of the data line to be cached. The tag is not initialized by a hardware reset.

(2) V bit (validity bit)

Indicates that valid data is stored in the cache line. When this bit is "1", the cache line data is valid. The V bit is initialized to "0" by a hardware reset, but retains its value in a manual reset.

(3) U bit (dirty bit)

The U bit is set to "1" if data is written to the cache line while the cache is being used in copy-back mode. That is, the U bit indicates a mismatch between the data in the cache line and the data in external memory. The U bit is never set to "1" while the cache is being used in write-through mode, unless it is modified by accessing the memory-mapped cache (see section 8.6, Memory-Mapped Cache Configuration). The U bit is initialized to "0" by a hardware reset.

(4) Data array

The data field holds 32 bytes (256 bits) of data per cache line. The data array is not initialized by a hardware reset.

(5) LRU

In a 4-way set-associative method, up to 4 items of data can be registered in the cache at each entry address. When an entry is registered, the LRU bit indicates which of the 4 ways it is to be registered in. The LRU mechanism uses 6 bits of each entry, and its usage is controlled by hardware. The LRU (least-recently-used) algorithm is used for way selection, and selects the less recently accessed way. The LRU bits are initialized to "0" by a hardware reset. The LRU bits cannot be read from or written to by software.

8.2 Register Descriptions

The following registers are related to cache.

Table 8.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Cache control register	CCR	H'0000 0000	H'FF00 001C	32	8-4
Queue address control register 0	QACR0	Undefined	H'FF00 0038	32	8-6
Queue address control register 1	QACR1	Undefined	H'FF00 003C	32	8-6
On-chip memory control register	RAMCR	H'0000 0000	H'FF00 0074	32	8-7

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

8.2.1 Cache Control Register (CCR)

The CCR register controls the cache operating mode, the cache write mode, and invalidation of all cache entries.

CCR modifications must only be made by a program in the non-cacheable P2 area or IL memory. After CCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the cacheable area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the cacheable area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is "0" (the value after a reset) before updating CCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after CCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

Cache Control Register (CCR) <P4 address: location H'FF00 001C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ICI	—	—	ICE	—	—	—	—	OCI	CB	WT	OCE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	ICI	0	0	W	IC Invalidation Bit When "1" is written to this bit, the V bits of all IC entries are cleared to "0". This bit is always read as "0".
10, 9	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
8	ICE	0	R	W	IC Enable Bit Selects whether the IC is used. Note however when address translation is performed, the IC cannot be used unless the C bit in the page management information is also "1". 0: IC not used 1: IC used
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3	OCI	0	0	W	OC Invalidation Bit When "1" is written to this bit, the V and U bits of all OC entries are cleared to "0". This bit is always read as "0".
2	CB	0	R	W	Copy-Back Bit Indicates the P1 area cache write mode. 0: Write-through mode 1: Copy-back mode
1	WT	0	R	W	Write-Through Mode Indicates the P0, U0, and P3 area cache write mode. When address translation is performed, the value of the WT bit in the page management information has priority. 0: Copy-back mode 1: Write-through mode
0	OCE	0	R	W	OC Enable Bit Selects whether the OC is used. Note however when address translation is performed, the OC cannot be used unless the C bit in the page management information is also "1". 0: OC not used 1: OC used

8.2.2 Queue Address Control Register 0 (QACR0)

The QACR0 register specifies the area onto which store queue 0 (SQ0) is mapped when the MMU is disabled.

Queue Address Control Register 0 (QACR0) <P4 address: location H'FF00 0038>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AREA0			—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	Undefined	Undefined	Undefined	0	0

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4 to 2	AREA0	Undefined	R	W	When the MMU is disabled, these bits generate physical address bits [28:26] for SQ0.
1, 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

8.2.3 Queue Address Control Register 1 (QACR1)

The QACR1 register specifies the area onto which store queue 1 (SQ1) is mapped when the MMU is disabled.

Queue Address Control Register 1 (QACR1) <P4 address: location H'FF00 003C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AREA1			—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	Undefined	Undefined	Undefined	0	0

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4 to 2	AREA1	Undefined	R	W	When the MMU is disabled, these bits generate physical address bits [28:26] for SQ1.
1, 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

8.2.4 On-Chip Memory Control Register (RAMCR)

The RAMCR register controls the number of ways in the IC and OC and prediction of the IC way.

The RAMCR register modifications must only be made by a program in the non-cacheable P2 area. After RAMCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the cacheable area, the IL memory area or the OL memory area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the cacheable area, the IL memory area or the OL memory area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is "0" (the value after a reset) before updating RAMCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after RAMCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

On-Chip Memory Control Register (RAMCR) <P4 address: location H'FF00 0074>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMD	RP	IC2W	OC2W	ICWPW	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9	RMD	0	R	W	On-Chip Memory Access Mode Bit For details, see section 9.3.5, On-Chip Memory Protection Functions.
8	RP	0	R	W	On-Chip Memory Protection Enable Bit For details, see section 9.3.5, On-Chip Memory Protection Functions.
7	IC2W	0	R	W	IC Two-Way Mode Bit 0: IC is a four-way operation 1: IC is a two-way operation For details, see section 8.4.3, IC Two-Way Mode.
6	OC2W	0	R	W	OC Two-Way Mode Bit 0: OC is a four-way operation 1: OC is a two-way operation For details, see section 8.3.6, OC Two-Way Mode.
5	ICWPD	0	R	W	IC Way Prediction Stop Bit Selects whether the IC way prediction is used. 0: Instruction cache performs way prediction 1: Instruction cache does not perform way prediction
4 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

8.3 Operand Cache Operation

8.3.1 Read Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is read from a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tags read from the each way is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:
   - If there is a way whose tag matches and its V bit is "1", see No. 3.
   - If there is no way whose tag matches and its V bit is "1" and the U bit of the way which is selected to replace using the LRU bits is "0", see No. 4.
   - If there is no way whose tag matches and its V bit is "1" and the U bit of the way which is selected to replace using the LRU bits is "1", see No. 5.
3. Cache hit

   The data indexed by virtual address bits [4:0] is read from the data field of the cache line on the hitted way in accordance with the access size. Then the LRU bits are updated to indicate the hitted way is the latest one.
4. Cache miss (no write-back)

   Data is read into the cache line on the way, which is selected to replace, from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. When the corresponding data arrives in the cache, the read data is returned to the CPU. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, "1" is written to the V bit and "0" is written to the U bit on the way. Then the LRU bit is updated to indicate the way is latest one.
5. Cache miss (with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then data is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data, and when the corresponding data arrives in the cache, the read data is returned to the CPU. While the remaining one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, "1" is written to the V bit, and "0" to the U bit. And the LRU bits are updated to indicate the way is latest one. The data in the write-back buffer is then written back to external memory.

8.3.2 Prefetch Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is prefetched from a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:
   - If there is a way whose tag matches and its V bit is "1", see No. 3.
   - If there is no way whose tag matches and the V bit is "1", and the U bit of the way which is selected to replace using the LRU bits is "0", see No. 4.
   - If there is no way whose tag matches and the V bit is "1", and the U bit of the way which is selected to replace using the LRU bits is "1", see No. 5.

3. Cache hit

   Then the LRU bits are updated to indicate the hitted way is the latest one.

4. Cache miss (no write-back)

   Data is read into the cache line on the way, which is selected to replace, from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation the CPU doesn't wait the data arrives. While the one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, "1" is written to the V bit and "0" is written to the U bit on the way. And the LRU bit is updated to indicate the way is latest one.

5. Cache miss (with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then data is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation the CPU doesn't wait the data arrives. While the one cache line of data is being read, the CPU can execute the next processing. And the LRU bits are updated to indicate the way is latest one. The data in the write-back buffer is then written back to external memory.

8.3.3 Write Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is written to a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

   — If there is a way whose tag matches and its V bit is "1", see No. 3 for copy-back and No. 4 for write-through.

   — I If there is no way whose tag matches and its V bit is "1" and the U bit of the way which is selected to replace using the LRU bits is "0", see No. 5 for copy-back and No. 7 for write-through.

   — If there is no way whose tag matches and its V bit is "1" and the U bit of the way which is selected to replace using the LRU bits is "1", see No. 6 for copy-back and No. 7 for write-through.

3. Cache hit (copy-back)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then "1" is written to the U bit. The LRU bits are updated to indicate the way is the latest one.

4. Cache hit (write-through)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. A write is also performed to external memory corresponding to the virtual address. Then the LRU bits are updated to indicate the way is the latest one. In this case, the U bit isn't updated.

5. Cache miss (copy-back, no write-back)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then, the data, excluding the cache-missed data which is written already, is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address.

   Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, "1" is written to the V bit and the U bit on the way. Then the LRU bit is updated to indicate the way is latest one.

6. Cache miss (copy-back, with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then a data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then, the data, excluding the cache-missed data which is written already, is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, "1" is written to the V bit and the U bit on the way. Then the LRU bit is updated to indicate the way is latest one. Then the data in the write-back buffer is then written back to external memory.

7. Cache miss (write-through)

   A write of the specified access size is performed to the external memory corresponding to the virtual address. In this case, a write to cache is not performed.

8.3.4 Write-Back Buffer

In order to give priority to data reads to the cache and improve performance, the SH-4A has a write-back buffer which holds the relevant cache entry when it becomes necessary to purge a dirty cache entry into external memory as the result of a cache miss. The write-back buffer contains one cache line of data and the physical address of the purge destination.

Physical address bits [28:5]	LW0	LW1	LW2	LW3	LW4	LW5	LW6	LW7

Figure 8.3 Configuration of Write-Back Buffer

8.3.5 Write-Through Buffer

The SH-4A has a 64-bit buffer for holding write data when writing data in write-through mode or writing to a non-cacheable area. This allows the CPU to proceed to the next operation as soon as the write to the write-through buffer is completed, without waiting for completion of the write to external memory.

Physical address bits[28:0]	LW0	LW1

Figure 8.4 Configuration of Write-Through Buffer

8.3.6 OC Two-Way Mode

When the OC2W bit in RAMCR is set to "1", OC two-way mode which only uses way 0 and way 1 in the OC is entered. Thus, power consumption can be reduced. In this mode, only way 0 and way 1 are used even if a memory-mapped OC access is made.

The OC2W bit should be modified by a program in the P2 area. At that time, if the valid line has already been recorded in the OC, data should be written back by software, if necessary, "1" should be written to the OCI bit in CCR, and all entries in the OC should be invalid before modifying the OC2W bit.

8.4 Instruction Cache Operation

8.4.1 Read Operation

When the IC is enabled (ICE = 1 in CCR) and instruction fetches are performed from a cacheable area, the instruction cache operates as follows:

1. The tag, V bit, U bit and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:
   — If there is a way whose tag matches and the V bit is "1", see No. 3.
   — If there is no way whose tag matches and the V bit is "1", see No. 4.
3. Cache hit
   The data indexed by virtual address bits [4:2] is read as an instruction from the data field on the hit way. The LRU bits are updated to indicate the way is the latest one.
4. Cache miss
   Data is read into the cache line on the way which selected using LRU bits to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data, and when the corresponding data arrives in the cache, the read data is returned to the CPU as an instruction. While the remaining one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, and "1" is written to the V bit, the LRU bits are updated to indicate the way is the latest one.

8.4.2 Prefetch Operation

When the IC is enabled (ICE = 1 in CCR) and instruction prefetches are performed from a cacheable area, the instruction cache operates as follows:

1. The tag, V bit, Ubit and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:
   — If there is a way whose tag matches and the V bit is "1", see No. 3.
   — If there is no way whose tag matches and the V bit is "1", see No. 4.
3. Cache hit
   The LRU bits is updated to indicate the way is the latest one.
4. Cache miss
   Data is read into the cache line on a way which selected using the LRU bits to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation, the CPU doesn't wait the data arrived. While the one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, and "1" is written to the V bit, the LRU bits is updated to indicate the way is the latest one.

8.4.3 IC Two-Way Mode

When the IC2W bit in RAMCR is set to "1", IC two-way mode which only uses way 0 and way 1 in the IC is entered. Thus, power consumption can be reduced. In this mode, only way 0 and way 1 are used even if a memory-mapped IC access is made.

The IC2W bit should be modified by a program in the P2 area. At that time, if the valid line has already been recorded in the IC, "1" should be written to the ICI bit in CCR and all entries in the IC should be invalid before modifying the IC2W bit.

8.4.4 Instruction Cache Way Prediction Operation

The SH-4A incorporates an instruction cache (IC) way prediction scheme to reduce power consumption. This is achieved by activating only the data array that corresponds to a predicted way. When way prediction misses occur, data must be re-read from the correct way, which may lead to lower performance in instruction fetching. Setting the ICWPD bit to "1" disables the IC way prediction scheme. Since way prediction misses do not occur in this mode, there is no loss of performance in instruction fetching but the IC consumes more power. The ICWPD bit should be modified by a program in the non-cacheable P2 area. If a valid line has already been recorded in the IC at this time, invalidate all entries in the IC by writing "1" to the ICI bit in the CCR register before modifying the ICWPD bit.

8.5 Cache Operation Instruction

8.5.1 Coherency between Cache and External Memory

(1) Cache Operation Instruction

Coherency between cache and external memory should be assured by software. In the SH-4A, the following six instructions are supported for cache operations. Details of these instructions are given in section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual.

- Operand cache invalidate instruction: OCBI @Rn
  Operand cache invalidation (no write-back)

- Operand cache purge instruction: OCBP @Rn
  Operand cache invalidation (with write-back)

- Operand cache write-back instruction: OCBWB @Rn
  Operand cache write-back

- Operand cache allocate instruction: MOVCA.L R0,@Rn
  Operand cache allocation

- Instruction cache invalidate instruction: ICBI @Rn
  Instruction cache invalidation

- Operand access synchronization instruction: SYNCO
  Wait for data transfer completion

(2) Changes in Instruction Specifications Regarding Coherency Control

Of the operand cache operating instructions, the coherency control-related specifications of OCBI, OCBP, and OCBWB have been changed from those of the SH-4A with H'20-valued VER bits in the processor version register (PVR).

- Changes in the invalidate instruction OCBI@Rn

  When Rn is designating an address in a non-cacheable area, this instruction is executed as NOP in the SH-4A with H'20-valued VER bits in the processor version register (PVR). In versions of the SH-4A with extended functions, this instruction invalidates the operand cache line designated by way = Rn[14:13] and entry = Rn[12:5] provided that Rn[31:24] = H'F4 (OC address array area). In this process, writing back of the line does not take place even if the line to be invalidated is dirty. This operation is only executable in privileged mode, and an address error exception occurs in user mode. TLB-related exceptions do not occur.

  Do not execute this instruction for the memory-mapped array areas and control register areas for which Rn[31:24] is not H'F4, and their reserved areas (H'F0 to H'F3, H'F5 to H'FF).

- Changes in the purge instruction OCBP@Rn

  When Rn is designating an address in a non-cacheable area, this instruction is executed as NOP in the SH-4A with H'20-valued VER bits in the processor version register (PVR). In versions of the SH-4A with extended functions, this instruction invalidates the operand cache line designated by way = Rn[14:13] and entry = Rn[12:5] provided that Rn[31:24] = H'F4 (OC address array area). In this process, writing back of the line takes place when the line to be invalidated is dirty. This operation is only executable in privileged mode, and an address error exception occurs in user mode. TLB-related exceptions do not occur.

  Do not execute this instruction for the memory-mapped array areas and control register areas for which Rn[31:24] is not H'F4, and their reserved areas (H'F0 to H'F3, H'F5 to H'FF).

- Changes in the write-back instruction OCBWB@Rn

  When Rn is designating an address in a non-cacheable area, this instruction is executed as NOP in the SH-4A with H'20-valued VER bits in the processor version register (PVR). In versions of the SH-4A with extended functions, provided that Rn[31:24] = H'F4 (OC address array area), this instruction writes back the operand cache line designated by way = Rn[14:13] and entry = Rn[12:5] if it is dirty and clears the dirty bit to "0". This operation is only executable in privileged mode, and an address error exception occurs in user mode. TLB-related exceptions do not occur.

  Do not execute this instruction for the memory-mapped array areas and control register areas for which Rn[31:24] is not H'F4, and their reserved areas (H'F0 to H'F3, H'F5 to H'FF).

8.5.2 Prefetch Operation

The SH-4A supports a prefetch instruction to reduce the cache fill penalty incurred as the result of a cache miss. If it is known that a cache miss will result from a read or write operation, it is possible to fill the cache with data beforehand by means of the prefetch instruction to prevent a cache miss due to the read or write operation, and so improve software performance. If a prefetch instruction is executed for data already held in the cache, or if the prefetch address results in a UTLB miss or a protection violation, the result is no operation, and an exception is not generated. Details of the prefetch instruction are given in section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual.

- Prefetch instruction (OC) : PREF @Rn
- Prefetch instruction (IC) : PREFI @Rn

8.6 Memory-Mapped Cache Configuration

The IC and OC can be managed by software. The contents of IC data array can be read from or written to by a program in the P2 area by means of a MOV instruction in privileged mode. The contents of IC address array can also be read from or written to in privileged mode by a program in the P2 area or the IL memory area by means of a MOV instruction. Operation is not guaranteed if access is made from a program in another area. In this case, execute one of the following three methods for executing a branch to the P0, U0, P1, or P3 area.

1. Execute a branch using the RTE instruction.
2. Execute a branch to the P0, U0, P1, or P3 area after executing the ICBI instruction for any address (including non-cacheable area).
3. If the MC bit in IRMCR is "0" (the value after a reset) before making an access to the memory-mapped IC, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after making an access to the memory-mapped IC.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

In privileged mode, the OC contents can be read from or written to by a program in the P1 or P2 area by means of a MOV instruction. Operation is not guaranteed if access is made from a program in another area. The IC and OC are allocated to the P4 area in the virtual address space. Only data accesses can be used on both the IC address array and data array and the OC address array and data array, and accesses are always longword-size. Instruction fetches cannot be performed in these areas. For reserved bits, a write value of "0" should be specified and the read value is undefined.

8.6.1 IC Address Array

The IC address array is allocated to addresses H'F000 0000 to H'F0FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the write tag and V bit are specified in the data field.

In the address field, bits [31:24] have the value H'F0 indicating the IC address array, and the way is specified by bits [14:13] and the entry by bits [12:5]. The association bit (A bit) [3] in the address field specifies whether or not association is performed when writing to the IC address array. As only longword access is used, "0" should be specified for address field bits [1:0].

In the data field, the tag is indicated by bits [31:10], and the V bit by bit [0]. As the IC address array tag is 19 bits in length, data field bits [31:29] are not used in the case of a write in which association is not performed. Data field bits [31:29] are used for the virtual address specification only in the case of a write in which association is performed.

The following three kinds of operation can be used on the IC address array:

1. IC address array read

   The tag and V bit are read into the data field from the IC entry corresponding to the way and entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is "1" or "0".
2. IC address array write (non-associative)

   The tag and V bit specified in the data field are written to the IC entry corresponding to the way and entry set in the address field. The A bit in the address field should be cleared to "0".
3. IC address array write (associative)

   When a write is performed with the A bit in the address field set to "1", the tag in each way stored in the entry specified in the address field is compared with the tag specified in the data field. The way numbers of bits [14:13] in the address field are not used. If the MMU is enabled at this time, comparison is performed after the virtual address specified by data field bits [31:10] has been translated to a physical address using the ITLB. If the addresses match and the V bit in the way is "1", the V bit specified in the data field is written into the IC entry. In other cases, no operation is performed. This operation is used to invalidate a specific IC entry. If an ITLB miss occurs during address translation, or the comparison shows a mismatch, an exception is not generated, no operation is performed, and the write is not executed.

Note: • IC address array associative writing function may not be supported in the future SuperH Series. Therefore, it is recommended that the ICBI instruction should be used to operate the IC definitely by handling ITLB miss and reporting ITLB miss exception.

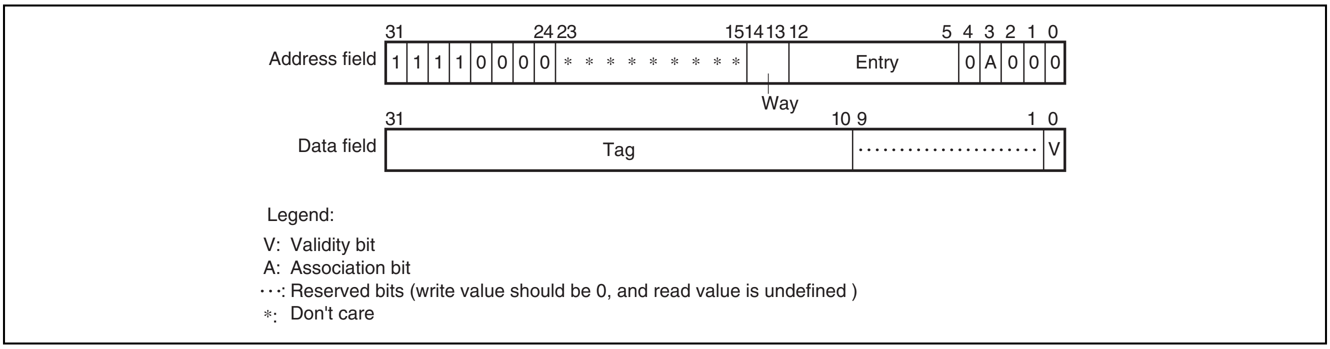


Figure 8.5 Memory-Mapped IC Address Array (Cache size = 32 Kbytes)

8.6.2 IC Data Array

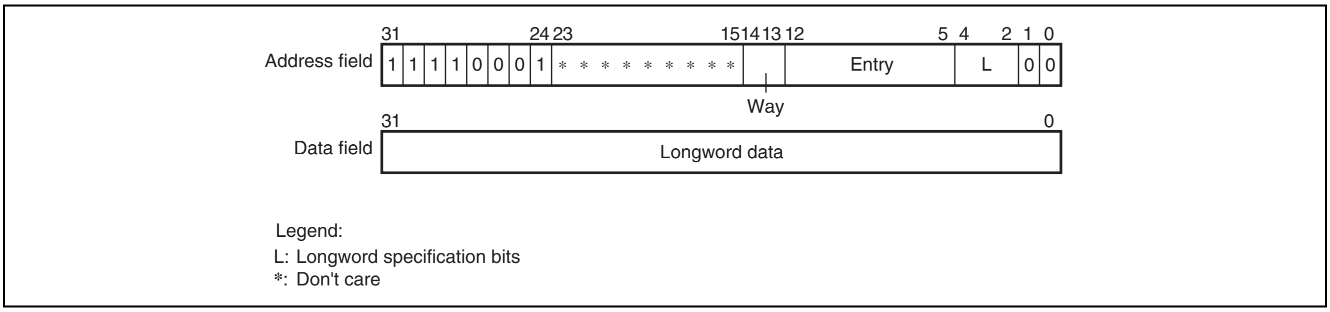
The IC data array is allocated to addresses H'F100 0000 to H'F1FF FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the longword data to be written is specified in the data field.

In the address field, bits [31:24] have the value H'F1 indicating the IC data array, and the way is specified by bits [14:13] and the entry by bits [12:5]. Address field bits [4:2] are used for the longword data specification in the entry. As only longword access is used, "0" should be specified for address field bits [1:0].

The data field is used for the longword data specification.

The following two kinds of operation can be used on the IC data array:

1. IC data array read
   Longword data is read into the data field from the data specified by the longword specification bits in the address field in the IC entry corresponding to the way and entry set in the address field.
2. IC data array write
   The longword data specified in the data field is written for the data specified by the longword specification bits in the address field in the IC entry corresponding to the way and entry set in the address field.



Legend:
L: Longword specification bits
*: Don't care

Figure 8.6 Memory-Mapped IC Data Array (Cache size = 32 Kbytes)

8.6.3 OC Address Array

The OC address array is allocated to addresses H'F400 0000 to H'F4FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the write tag, U bit, and V bit are specified in the data field.

In the address field, bits [31:24] have the value H'F4 indicating the OC address array, and the way is specified by bits [14:13] and the entry by bits [12:5]. The association bit (A bit) [3] in the address field specifies whether or not association is performed when writing to the OC address array. As only longword access is used, "0" should be specified for address field bits [1:0].

In the data field, the tag is indicated by bits [31:10], the U bit by bit [1], and the V bit by bit [0]. As the OC address array tag is 19 bits in length, data field bits [31:29] are not used in the case of a write in which association is not performed. Data field bits [31:29] are used for the virtual address specification only in the case of a write in which association is performed.

The following three kinds of operation can be used on the OC address array:

1. OC address array read

   The tag, U bit, and V bit are read into the data field from the OC entry corresponding to the way and entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is "1" or "0".
2. OC address array write (non-associative)

   The tag, U bit, and V bit specified in the data field are written to the OC entry corresponding to the way and entry set in the address field. The A bit in the address field should be cleared to "0".

   When a write is performed to a cache line for which the U bit and V bit are both "1", after write-back of that cache line, the tag, U bit, and V bit specified in the data field are written.
3. OC address array write (associative)

   When a write is performed with the A bit in the address field set to "1", the tag in each way stored in the entry specified in the address field is compared with the tag specified in the data field. The way numbers of bits [14:13] in the address field are not used. If the MMU is enabled at this time, comparison is performed after the virtual address specified by data field bits [31:10] has been translated to a physical address using the UTLB. If the addresses match and the V bit in the way is "1", the U bit and V bit specified in the data field are written into the OC entry. In other cases, no operation is performed. This operation is used to invalidate a specific OC entry. If the OC entry U bit is "1", and "0" is written to the V bit or to the U bit, write-back is performed. If a UTLB miss occurs during address translation, or the comparison shows a mismatch, an exception is not generated, no operation is performed, and the write is not executed.

Note: • OC address array associative writing function may not be supported in the future SuperH Series. Therefore, it is recommended that the OCBI, OCBP, or OCBWB instruction should be used to operate the OC definitely by reporting data TLB miss exception.

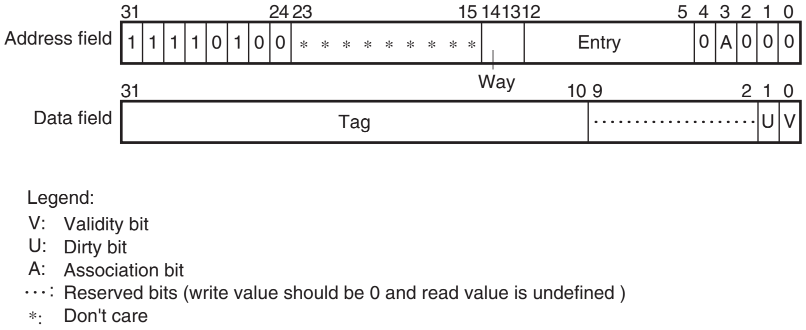


Figure 8.7 Memory-Mapped OC Address Array (Cache size = 32 Kbytes)

8.6.4 OC Data Array

The OC data array is allocated to addresses H'F500 0000 to H'F5FF FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the longword data to be written is specified in the data field.

In the address field, bits [31:24] have the value H'F5 indicating the OC data array, and the way is specified by bits [14:13] and the entry by bits [12:5]. Address field bits [4:2] are used for the longword data specification in the entry. As only longword access is used, "0" should be specified for address field bits [1:0].

The data field is used for the longword data specification.

The following two kinds of operation can be used on the OC data array:

1. OC data array read

   Longword data is read into the data field from the data specified by the longword specification bits in the address field in the OC entry corresponding to the way and entry set in the address field.
2. OC data array write

   The longword data specified in the data field is written for the data specified by the longword specification bits in the address field in the OC entry corresponding to the way and entry set in the address field. This write does not set the U bit to "1" on the address array side.

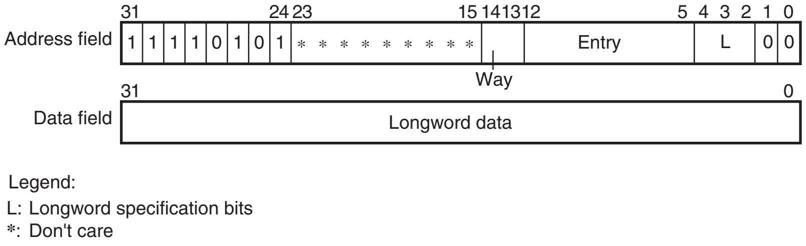


Figure 8.8 Memory-Mapped OC Data Array (Cache size = 32 Kbytes)

8.6.5 Memory Allocation Associative Write Operation

Associative writing to the IC and OC address arrays may not be supported in future SuperH-family products. The use of instructions ICBI, OCBI, OCBP, and OCBWB is recommended. These instructions handle ITLB misses, and notify instruction TLB miss exceptions and data TLB miss exceptions, thus providing a sure way of controlling the IC and OC. As a transitional measure, the SH-4A generates address errors when this function is used. If compatibility with previous products is a crucial consideration, on the other hand, the MMCAW bit in EXPMASK (H'FF2F 0004) can be set to "1" to enable this function. However, instructions ICBI, OCBI, OCBP, and OCBWB should be used to guarantee compatibility with future SuperH-family products.

8.7 Store Queues

The SH-4A supports two 32-byte store queues (SQs) to perform high-speed writes to external memory.

8.7.1 SQ Configuration

There are two 32-byte store queues, SQ0 and SQ1, as shown in figure 8.9. These two store queues can be set independently.

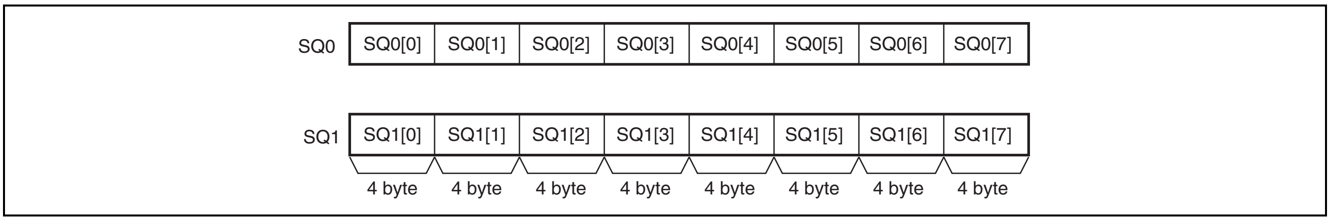


Figure 8.9 Store Queue Configuration

8.7.2 Writing to SQ

A write to the SQs can be performed using a store instruction for addresses H'E000 0000 to H'E3FF FFFC in the P4 area. A longword or quadword access size can be used. The meanings of the address bits are as follows:

[31:26]	: 111000	Store queue specification
[25:6]	: Don't care	Used for external memory transfer/access right
[5]	: 0/1	0: SQ0 specification 1: SQ1 specification
[4:2]	: LW specification	Specifies longword position in SQ0/SQ1
[1:0]	: 00	Fixed at "0"

8.7.3 Transfer to External Memory

Transfer from the SQs to external memory can be performed with a prefetch instruction (PREF). Issuing a PREF instruction for addresses H'E000 0000 to H'E3FF FFFC in the P4 area starts a transfer from the SQs to external memory. The transfer length is fixed at 32 bytes, and the start address is always at a 32-byte boundary. While the contents of one SQ are being transferred to external memory, the other SQ can be written to without a penalty cycle. However, writing to the SQ involved in the transfer to external memory is kept waiting until the transfer is completed.

The physical address bits [28:0] of the SQ transfer destination are specified as shown below, according to whether the MMU is enabled or disabled.

- When MMU is enabled (AT = 1 in MMUCR)

  The SQ area (H'E000 0000 to H'E3FF FFFF) is set in VPN of the UTLB, and the transfer destination physical address in PPN. The ASID, V, SZ, SH, PR, and D bits have the same meaning as for normal address translation, but the C and WT bits have no meaning with regard to this page. When a prefetch instruction is issued for the SQ area, address translation is performed and physical address bits [28:10] are generated in accordance with the SZ bit specification. For physical address bits [9:5], the address prior to address translation is generated in the same way as when the MMU is disabled. Physical address bits [4:0] are fixed at "0". Transfer from the SQs to external memory is performed to this address.

- When MMU is disabled (AT = 0 in MMUCR)

  The SQ area (H'E000 0000 to H'E3FF FFFF) is specified as the address at which a PREF instruction is issued. The meanings of address bits [31:0] are as follows:

[31:26]	: 111000	Store queue specification
[25:6]	: Address	Transfer destination physical address bits [25:6]
[5]	: 0/1	0: SQ0 specification 1: SQ1 specification and transfer destination physical address bit [5]
[4:2]	: Don't care	No meaning in a prefetch
[1:0]	: 00	Fixed at "0"

  Physical address bits [28:26], which cannot be generated from the above address, are generated from QACR0 and QACR1.

QACR0[4:2]	: Physical address bits [28:26] corresponding to SQ0
QACR1[4:2]	: Physical address bits [28:26] corresponding to SQ1

  Physical address bits [4:0] are always fixed at "0" since burst transfer starts at a 32-byte boundary.

8.7.4 Determination of SQ Access Exception

Determination of an exception in a write to an SQ or transfer to external memory (PREF instruction) is performed as follows according to whether the MMU is enabled or disabled. If an exception occurs during a write to an SQ, the SQ contents before the write are retained. If an exception occurs in a data transfer from an SQ to external memory, the transfer to external memory will be aborted.

- When MMU is enabled (AT = 1 in MMUCR)

  Operation is in accordance with the address translation information recorded in the UTLB, and the SQMD bit in MMUCR. Write type exception judgment is performed for writes to the SQs, and read type exception judgment for transfer from the SQs to external memory (using a PREF instruction). As a result, a TLB miss exception or protection violation exception is generated as required. However, if SQ access is enabled in privileged mode only by the SQMD bit in MMUCR, an address error will occur even if address translation is successful in user mode.

- When MMU is disabled (AT = 0 in MMUCR)

  Operation is in accordance with the SQMD bit in MMUCR.

  0: Privileged/user mode access possible

  1: Privileged mode access possible

  If the SQ area is accessed in user mode when the SQMD bit in MMUCR is set to "1", an address error will occur.

8.7.5 Reading from SQ

In privileged mode in the SH-4A, reading the contents of the SQs may be performed by means of a load instruction for addresses H'FF00 1000 to H'FF00 103C in the P4 area. Only longword access is possible.

[31:6]	: H'FF00 1000	Store queue specification
[5]	: 0/1	0: SQ0 specification 1: SQ1 specification
[4:2]	: LW specification	Specifies longword position in SQ0/SQ1
[1:0]	: 00	Fixed at "0"

This page is blank for reasons of layout.

Section 9 IL Memory/OL Memory

The SH-4A includes two types of memory modules for storage of instructions and data: OL memory and IL memory. The OL memory is suitable for data storage while the IL memory is suitable for instruction storage.

9.1 Overview

(1) OL Memory

- Capacity

  The OL memory in the SH-4A is 16 Kbytes.
- Page

  The OL memory is divided into four pages (pages 0A, 0B, 1A and 1B).
- Memory map

  The OL memory is allocated in the addresses shown in table 9.1 in both the virtual address space and the physical address space.

Table 9.1 OL memory Addresses

Page	Address
Page 0A	H'E500 E000 to H'E500 EFFF
Page 0B	H'E500 F000 to H'E500 FFFF
Page 1A	H'E501 0000 to H'E501 0FFF
Page 1B	H'E501 1000 to H'E501 1FFF

- Ports

  Each page has three independent read/write ports and is connected to the SuperHyway bus, the cache/RAM internal bus, and operand bus. The operand bus is used when the OL memory is accessed through operand access. The cache/RAM internal bus is used when the OL memory is accessed through instruction fetch. The SuperHyway bus is used for OL memory access from the SuperHyway bus master module.
- Priority

  In the event of simultaneous accesses to the same page from different buses, the access requests are processed according to priority. The priority order is: SuperHyway bus > Cache/RAM internal bus > operand bus.

(2) IL Memory

- Capacity

  The IL memory in the SH-4A is 8 Kbytes.
- Page

  The IL memory is divided into two pages (pages 0 and 1).
- Memory map

  The IL memory is allocated to the addresses shown in table 9.2 in both the virtual address space and the physical address space.

Table 9.2 IL Memory Addresses

Page	Address
Page 0	H'E520 0000 to H'E520 0FFF
Page 1	H'E520 1000 to H'E520 1FFF

- Ports

  The page has three independent read/write ports and is connected to the SuperHyway bus, the cache/RAM internal bus, and the instruction bus. The instruction bus is used when the IL memory is accessed through instruction fetch. The cache/RAM internal bus is used when the IL memory is accessed through operand access. The SuperHyway bus is used for IL memory access from the SuperHyway bus master module.
- Priority

  In the event of simultaneous accesses to the same page from different buses, the access requests are processed according to priority. The priority order is: SuperHyway bus > cache/RAM internal bus > instruction bus.

9.2 Register Descriptions

The following registers are related to the on-chip memory.

Table 9.3 Register Configuration

Name	Abbreviation	After Reset	P4 Address	Size	Page
On-chip memory control register	RAMCR	H'0000 0000	H'FF00 0074	32	9-4
OL memory transfer source address register 0	LSA0	Undefined	H'FF00 0050	32	9-5
OL memory transfer source address register 1	LSA1	Undefined	H'FF00 0054	32	9-6
OL memory transfer destination address register 0	LDA0	Undefined	H'FF00 0058	32	9-7
OL memory transfer destination address register 1	LDA1	Undefined	H'FF00 005C	32	9-8

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

9.2.1 On-Chip Memory Control Register (RAMCR)

The RAMCR register controls the protection functions in the on-chip memory.

On-Chip Memory Control Register (RAMCR) <P4 address: location H'FF00 0074>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMD	RP	IC2W	OC2W	ICWPD	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31to10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9	RMD	0	R	W	On-Chip Memory Access Mode Bit Specifies the right of access to the on-chip memory from the virtual address space. 0: An access in privileged mode is allowed (An address error exception occurs in user mode) 1: An access in user/ privileged mode is allowed
8	RP	0	R	W	On-Chip Memory Protection Enable Bit Selects whether or not to use the protection functions using ITLB and UTLB for accessing the on-chip memory from the virtual address space. 0: Protection functions are not used 1: Protection functions are used For further details, refer to section 9.3.5, On-Chip Memory Protection Functions.
7	IC2W	0	R	W	IC Two-Way Mode Bit For further details, refer to section 8.4.3, IC Two-Way Mode.
6	OC2W	0	R	W	OC Two-Way Mode Bit For further details, refer to section 8.3.6, OC Two-Way Mode.
5	ICWPD	0	R	W	IC Way Prediction Disable Bit For further details, refer to section 8.4.4, Instruction Cache Way Prediction Operation.
4 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

9.2.2 OL memory Transfer Source Address Register 0 (LSA0)

When MMUCR.AT = "0" or RAMCR.RP = "0", the LSA0 register specifies the transfer source physical address for block transfer to page 0A or 0B of the OL memory.

OL memory Transfer Source Address Register 0 (LSA0) <P4 address: location H'FF00 0050>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	L0SADR												
After Reset:	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L0SADR						—	—	—	—	L0SSZ					
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 10	L0SADR	Undefined	R	W	OL memory Page 0 Block Transfer Source Address Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify the transfer source physical address for block transfer to page 0A or 0B in the OL memory.
9 to 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5 to 0	L0SSZ	Undefined	R	W	OL memory Page 0 Block Transfer Source Address Select Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L0SADR values are used as bits 15 to 10 of the transfer source physical address for block transfer to page 0A or 0B in the OL memory. L0SSZ[5:0] correspond to the transfer source physical addresses [15:10]. 0: The operand address is used as the transfer source physical address 1: The L0SADR value is used as the transfer source physical address Settable values: 111111: Transfer source physical address is specified in 1-Kbyte units 111110: Transfer source physical address is specified in 2-Kbyte units 111100: Transfer source physical address is specified in 4-Kbyte units 111000: Transfer source physical address is specified in 8-Kbyte units 110000: Transfer source physical address is specified in 16-Kbyte units 100000: Transfer source physical address is specified in 32-Kbyte units 000000: Transfer source physical address is specified in 64-Kbyte units Settings other than the ones given above are prohibited

9.2.3 OL memory Transfer Source Address Register 1 (LSA1)

When MMUCR.AT = "0" or RAMCR.RP = "0", the LSA1 register specifies the transfer source physical address for block transfer to page 1A or 1B in the OL memory.

OL memory Transfer Source Address Register 1 (LSA1) <P4 address: location H'FF00 0054>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	L1SADR												
After Reset:	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L1SADR						—	—	—	—	L1SSZ					
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 10	L1SADR	Undefined	R	W	OL memory Page 1 Block Transfer Source Address Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify transfer source physical address for block transfer to page 1A or 1B in the OL memory.
9 to 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5 to 0	L1SSZ	Undefined	R	W	OL memory Page 1 Block Transfer Source Address Select Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L1SADR values are used as bits 15 to 10 of the transfer source physical address for block transfer to page 1A or 1B in the OL memory. L1SSZ bits [5:0] correspond to the transfer source physical addresses [15:10]. 0: The operand address is used as the transfer source physical address 1: The L1SADR value is used as the transfer source physical address Settable values: 111111: Transfer source physical address is specified in 1-Kbyte units 111110: Transfer source physical address is specified in 2-Kbyte units 111100: Transfer source physical address is specified in 4-Kbyte units 111000: Transfer source physical address is specified in 8-Kbyte units 110000: Transfer source physical address is specified in 16-Kbyte units 100000: Transfer source physical address is specified in 32-Kbyte units 000000: Transfer source physical address is specified in 64-Kbyte units Settings other than the ones given above are prohibited

9.2.4 OL memory Transfer Destination Address Register 0 (LDA0)

When MMUCR.AT = "0" or RAMCR.RP = "0", the LDA0 register specifies the transfer destination physical address for block transfer to page 0A or 0B of the OL memory.

OL memory Transfer Destination Address Register 0 (LDA0) <P4 address: location H'FF00 0058>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	L0DADR												
After Reset:	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L0DADR						—	—	—	—	L0DSZ					
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 10	L0DADR	Undefined	R	W	OL memory Page 0 Block Transfer Destination Address Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify transfer destination physical address for block transfer to page 0A or 0B in the OL memory.
9 to 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5 to 0	L0DSZ	Undefined	R	W	OL memory Page 0 Block Transfer Destination Address Select Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L0DADR values are used as bits 15 to 10 of the transfer destination physical address for block transfer to page 0A or 0B in the OL memory. L0DSZ bits [5:0] correspond to the transfer destination physical address bits [15:10]. 0: The operand address is used as the transfer destination physical address 1: The L0DADR value is used as the transfer destination physical address Settable values: 111111: Transfer destination physical address is specified in 1-Kbyte units 111110: Transfer destination physical address is specified in 2-Kbyte units 111100: Transfer destination physical address is specified in 4-Kbyte units 111000: Transfer destination physical address is specified in 8-Kbyte units 110000: Transfer destination physical address is specified in 16-Kbyte units 100000: Transfer destination physical address is specified in 32-Kbyte units 000000: Transfer destination physical address is specified in 64-Kbyte units Settings other than the ones given above are prohibited.

9.2.5 OL memory Transfer Destination Address Register 1 (LDA1)

When MMUCR.AT = "0" or RAMCR.RP = "0", the LDA1 register specifies the transfer destination physical address for block transfer to page 1A or 1B in the OL memory.

OL memory Transfer Destination Address Register 1 (LDA1) <P4 address: location H'FF00 005C>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	L1DADR												
After Reset:	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L1DADR						—	—	—	—	L1DSZ					
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 10	L1DADR	Undefined	R	W	OL memory Page 1 Block Transfer Destination Address Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify transfer destination physical address for block transfer to page 1A or 1B in the OL memory.
9 to 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5 to 0	L1DSZ	Undefined	R	W	OL memory Page 1 Block Transfer Destination Address Select Bits When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L1DADR values are used as bits 15 to 10 of the transfer destination physical address for block transfer to page 1A or 1B in the OL memory. L1DSZ bits [5:0] correspond to the transfer destination physical addresses [15:10]. 0: The operand address is used as the transfer destination physical address 1: The L1DADR value is used as the transfer destination physical address Settable values: 111111: Transfer destination physical address is specified in 1-Kbyte units 111110: Transfer destination physical address is specified in 2-Kbyte units 111100: Transfer destination physical address is specified in 4-Kbyte units 111000: Transfer destination physical address is specified in 8-Kbyte units 110000: Transfer destination physical address is specified in 16-Kbyte units 100000: Transfer destination physical address is specified in 32-Kbyte units 000000: Transfer destination physical address is specified in 64-Kbyte units Settings other than the ones given above are prohibited.

9.3 Operation

9.3.1 Instruction Fetch Access from the CPU

(1) OL Memory

Instruction fetch access from the CPU is performed via the cache/RAM internal bus. This access takes more than one cycle.

(2) IL Memory

Instruction fetch access from the CPU is performed as a direct access from the instruction bus by a virtual address. In the case of successive accesses to the same page of IL memory and as long as no page conflict occurs, the access takes one cycle.

9.3.2 Operand Access from the CPU and Access from the FPU

(1) OL Memory

Access from the CPU or FPU is performed via the operand bus for a given virtual address. Read access from the operand bus by virtual address takes one cycle if the access is made successively to the same page of OL memory and as long as no page conflict occurs. Write access from the operand bus by virtual address takes one cycle as long as no page conflict occurs.

(2) IL Memory

Operand access from the CPU and access from the FPU are performed via the cache/RAM internal bus. Access via the cache/RAM internal bus takes more than one cycle.

9.3.3 Access from the SuperHyway Bus Master Module

On-chip memory is always accessed by the SuperHyway bus master module, such as DMAC, via the SuperHyway bus which is a physical address bus. The same addresses as for the virtual addresses must be used.

9.3.4 OL Memory Block Transfer

High-speed data transfer can be performed through block transfer between the OL memory and external memory without cache utilization.

Data can be transferred from the external memory to the OL memory through a prefetch instruction (PREF). Block transfer from the external memory to the OL memory begins when the PREF instruction is issued to the address in the OL memory area in the virtual address space.

Data can be transferred from the OL memory to the external memory through a write-back instruction (OCBWB). Block transfer from the OL memory to the external memory begins when the OCBWB instruction is issued to the address in the OL memory area in the virtual address space.

In either case, transfer size is fixed to 32 bytes. Since the start address is always limited to a 32-byte boundary, the lower five bits of the address indicated by Rn are ignored, and are always dealt with as all 0s. In either case, other pages and cache can be accessed during block transfer. If the page is accessed during data transfer, the CPU will stall until block transfer ends.

The physical addresses [28:0] of the external memory performing data transfers with the OL memory are specified as follows according to whether the MMU is enabled or disabled.

(1) When MMU is Enabled (MMUCR.AT = "1") and RAMCR.RP = "1"

An address of the OL memory area is specified to the UTLB VPN field, and to the physical address of the transfer source (in the case of the PREF instruction) or the transfer destination (in the case of the OCBWB instruction) to the PPN field. The ASID, V, SZ, SH, PR, and D bits have the same meaning as normal address conversion; however, the C and WT bits have no meaning in this page.

When the PREF instruction is issued to the OL memory area, address conversion is performed in order to generate the physical address bits [28:10] in accordance with the SZ bit specification. The physical address bits [9:5] are generated from the virtual address prior to address conversion. The physical address bits [4:0] are fixed to "0". Block transfer is performed to the OL memory from the external memory which is specified by these physical addresses.

When the OCBWB instruction is issued to the OL memory area, address conversion is performed in order to generate the physical address bits [28:10] in accordance with the SZ bit specification. The physical address bits [9:5] are generated from the virtual address prior to address conversion. The physical address bits [4:0] are fixed to "0". Block transfer is performed from the OL memory to the external memory specified by these physical addresses.

In PREF or OCBWB instruction execution, an MMU exception is checked as read type. After the MMU execution check, a TLB miss exception or protection error exception occurs if necessary. If an exception occurs, the block transfer is inhibited.

(2) When MMU is Disabled (MMUCR.AT = "0") or RAMCR.RP = "0"

The transfer source physical address in block transfer to page 0A or 0B in the OL memory is set in the L0SADR bits of the LSA0 register. And the L0SSZ bits in the LSA0 register choose either the virtual addresses specified through the PRFF instruction or the L0SADR values as bits 15 to 10 of the transfer source physical address. In other words, the transfer source area can be specified in units of 1 Kbyte to 64 Kbytes.

The transfer destination physical address in block transfer from page 0A or 0B in the OL memory is set in the L0DADR bits of the LDA0 register. And the L0DSZ bits in the LDA0 register choose either the virtual addresses specified through the OCBWB instruction or the L0DADR values as bits 15 to 10 of the transfer destination physical address. In other words, the transfer source area can be specified in units of 1 Kbyte to 64 Kbytes.

Block transfer to page 1A or 1B in the OL memory is set to LSA1 and LDA1 as with page 0A or 0B in the OL memory.

When the PREF instruction is issued to the OL memory area, the physical address bits [28:10] are generated in accordance with the LSA0 or LSA1 specification. The physical address bits [9:5] are generated from the virtual address. The physical address bits [4:0] are fixed to "0". Block transfer is performed from the external memory specified by these physical addresses to the OL memory.

When the OCBWB instruction is issued to the OL memory area, the physical address bits [28:10] are generated in accordance with the LDA0 or LDA1 specification. The physical address bits [9:5] are generated from the virtual address. The physical address bits [4:0] are fixed to "0". Block transfer is performed from the OL memory to the external memory specified by these physical addresses.

9.3.5 On-Chip Memory Protection Functions

The SH-4A implements the following protection functions to the on-chip memory by using the on-chip memory access mode bit (RMD) and the on-chip memory protection enable bit (RP) in the on-chip memory control register (RAMCR).

- Protection functions for access from the CPU and FPU

  When RAMCR.RMD = 0, and the on-chip memory is accessed in user mode, it is determined to be an address error exception.

  When MMUCR.AT = 1 and RAMCR.RP = 1, MMU exception and address error exception are checked in the on-chip memory area which is a part of area P4 as with the area P0/P3/U0.

The above descriptions are summarized in table 9.4.

Table 9.4 Protection Function Exceptions to Access On-Chip Memory

MMUCR.AT	RAMCR.RP	SR.MD	RAMCR. RMD	Always Occurring Exceptions	Possibly Occurring Exceptions
0	x	0	0	Address error exception	—
			1	—	—
		1	x	—	—
1	0	0	0	Address error exception	—
			1	—	—
		1	x	—	—
	1	0	0	Address error exception	—
			1	—	MMU exception
		1	x	—	MMU exception

Legend:

x: Don't care

9.4 Usage Notes

9.4.1 Page Conflict

In the event of simultaneous access to the same page from different buses, page conflict occurs. Although each access is completed correctly, this kind of conflict tends to lower OL memory accessibility. Therefore it is advisable to provide all possible preventative software measures. For example, conflicts will not occur if each bus accesses different pages.

9.4.2 Access Across Different Pages

(1) OL Memory

Read access from the operand bus is performed in one cycle when the access is made successively to the same page but takes multiple cycles (a maximum of two wait cycles may be required) when the access is made across pages or the previous access was made to memory other than OL memory. For this reason, from the viewpoint of performance optimization, it is recommended to design the software such that the page corresponding to the address for read access from the operand bus does not change so often.

(2) IL Memory

Access from the instruction bus is performed in one cycle when the access is made successively to the same page but takes multiple cycles (a maximum of two wait cycles may be required) when the access is made across pages or the previous access was made to memory other than IL memory. For this reason, from the viewpoint of performance optimization, it is recommended to design the software such that the target page does not change so often in access from the instruction bus. For example, allocating a separate program to each page will deliver better efficiency.

9.4.3 IL Memory Coherency

In order to allocate instructions in the IL memory, write an instruction to the IL memory, execute the following sequence, then branch to the rewritten instruction.

- SYNCO
- ICBI @Rn

In this case, the target for the ICBI instruction can be any address (IL memory address may be possible) within the range where no address error exception occurs, and cache hit/miss is possible.

Section 10 Operating Modes

10.1 Operating Modes

This MCU provides a total of six operation modes (modes 0 to 5). The operating mode is set with the MPMD, MD0 to MD2, and FWE pins. Do not change the states of these operating mode pins during the MCU operation.

10.1.1 MCU Operating Modes

(1) Normal operating modes (modes 0 to 2) and emulation support modes (modes 3 to 5)

The MPMD pin selects between the normal operating mode and the emulation support mode. This MCU operates by inputting a "L" level to the MPMD pin.

The emulation support modes are used when an emulator is connected to the MCU pins. Operation is not guaranteed if an emulation support mode is selected when no emulator is connected. The normal operating mode is used when no emulator is connected to the MCU pins. In the normal operating mode, this MCU operates by inputting a "H" level to the MPMD pin. When operating this device with no emulator connected, a "H" level must be applied to the MPMD pin at reset to select a normal operating mode.

Note that in this manual, operation in a normal operating mode is assumed unless explicitly stated otherwise.

(2) Single chip modes (modes 0 to 2 and 3 to 5)

This MCU operates in the single chip mode by inputting "L" level to the MD0 pin.

In single chip mode, only the internal ROM and RAM is used.

10.1.2 On-Board Programming Modes

The MD1, MD2, and FWE pins select the modes in which programs can be written to ROM. There are three such modes: user mode, boot mode, and user boot mode.

(1) User modes (modes 0 and 3)

These are program modes in which user MAT data and programs can be rewritten using an arbitrary interface. The user MAT can be rewritten when a "H" level is applied to the FWE pin. When a "L" level is applied to the FWE pin, only readout is possible from ROM.

(2) Boot modes (modes 1 and 4)

These are program modes in which the user MAT and the user boot MAT data and programs can be rewritten using the SCIF. The SCIF communication bit rate between the host and this MCU can be adjusted automatically.

(3) User boot modes (modes 2 and 5)

These are program modes in which the user MAT data and programs can be rewritten using an arbitrary interface. A hardware reset restart is required to transition to user boot mode.

Table 10.1 lists the operating mode selections and table 10.2 lists the operating mode pin settings. Combinations of values not listed in table 10.2 (modes other than modes 0 to 5) are prohibited.

Table 10.1 Selection of Operating Modes

Operating mode number	Mode Name			Emulation Function	External Bus	ROM program
	MCU Operating Mode		On-Board Programming Mode			
Mode 0	Normal operating mode	Single-chip mode	User mode	Disabled	Disabled	Possible
Mode 1			Boot mode	Disabled	Disabled	Possible
Mode 2			User boot mode	Disabled	Disabled	Possible
Mode 3	Emulation support mode*[1]	Single-chip mode	User mode	Enabled	Disabled	Possible
Mode 4			Boot mode	Enabled	Disabled	Possible
Mode 5			User boot mode	Enabled	Disabled	Possible

Note: *1 Operation cannot be guaranteed if the MCU is set to emulation support mode when no emulator is connected.

Table 10.2 Operating Mode Pin Settings

Operating mode number	Mode Name			Pin Setting				
	MCU Operating Mode		On-Board Programming Mode	MPMD	MD2	MD1	MD0	FWE
Mode 0	Normal operating mode	Single-chip mode	User mode	"H"	"L"	"L"	"L"	"L/H"
Mode 1			Boot mode	"H"	"L"	"H"	"L"	"H"
Mode 2			User boot mode	"H"	"H"	"L"	"L"	"H"
Mode 3	Emulation support mode	Single-chip mode	User mode	"L"	"L"	"L"	"L"	"L/H"
Mode 4			Boot mode	"L"	"L"	"H"	"L"	"H"
Mode 5			User boot mode	"L"	"H"	"L"	"L"	"H"

10.2 Register Descriptions

Table 10.3 lists the register configuration.

Table 10.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Mode control register	MDCR	Undefined	H'FFFF 2001	8	10-3

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

10.2.1 Mode Control Register (MDCR)

The MDCR register is provided for verifying the states of the mode pins (MD0 to MD2, MPMD, and FWE) and the DET3OR5 pin after a reset has been cleared.

The MDCR register acquires the states of the MPDMD, MD0 to MD2, and FWE pins during the period when the RESET# pin is asserted with the timing of the negation of the RESET# pin. Note, however, that for the DET3OR5 pin, the value of the pin after a reset is cleared is not latched, and that therefore the level on the DET3OR5 pin can be read out at any time.

Mode Control Register (MDCR) <P4 address: location H'FFFF 2001>

Bit:	7	6	5	4	3	2	1	0
	DET3OR5	—	FWE	MPMD	—	MD2	MD1	MD0
After Reset:	Undefined	0	Undefined*1	Undefined*1	0	Undefined*1	Undefined*1	Undefined*1

<After Reset: Undefined>

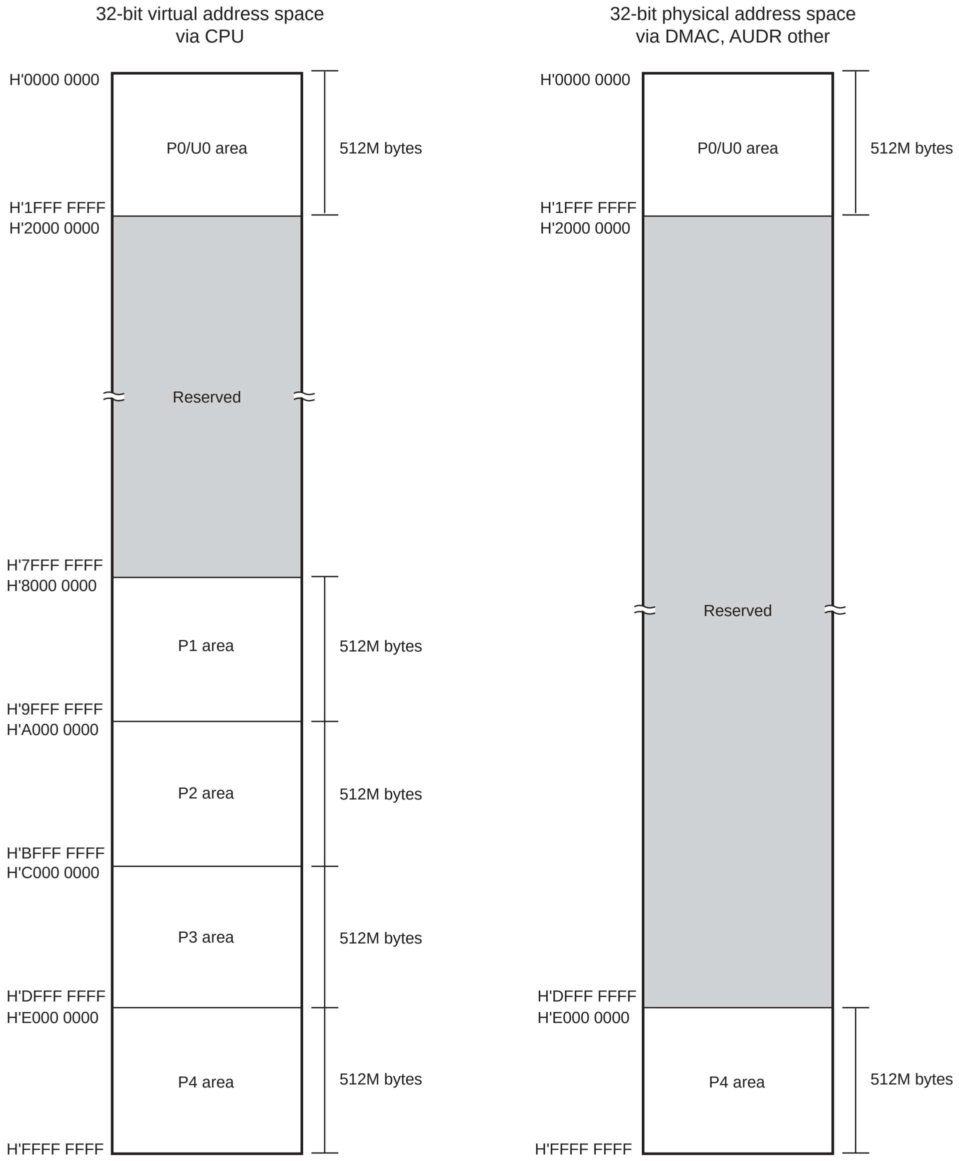
Bit	Abbreviation	After Reset	R	W	Description
7	DET3OR5	Undefined	R	—	DET3OR5 Input Level Bit Indicates the level input to the DET3OR5 pin.
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5	FWE	Undefined*1	R	—	FWE Bit Indicates the level input to the FWE pin during the period when the last reset was cleared. Note: • Verify the value of the FPMON register FWE bit for the ROM write and erase control state (disabled/enabled).
4	MPMD	Undefined*1	R	—	MPMD Bit Indicates the level input to the MPMD pin when the last reset was cleared.
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2	MD2	Undefined*1	R	—	MD2 Bit Indicates the level input to the MD2 pin during the period when the last reset was cleared.
1	MD1	Undefined*1	R	—	MD1 Bit Indicates the level input to the MD1 pin during the period when the last reset was cleared.

Bit	Abbreviation	After Reset	R	W	Description
0	MD0	Undefined*[1]	R	—	MD0 Bit Indicates the level input to the MD0 pin during the period when the last reset was cleared.

Note: *1 The value is determined from the state of the pin when the reset is cleared.

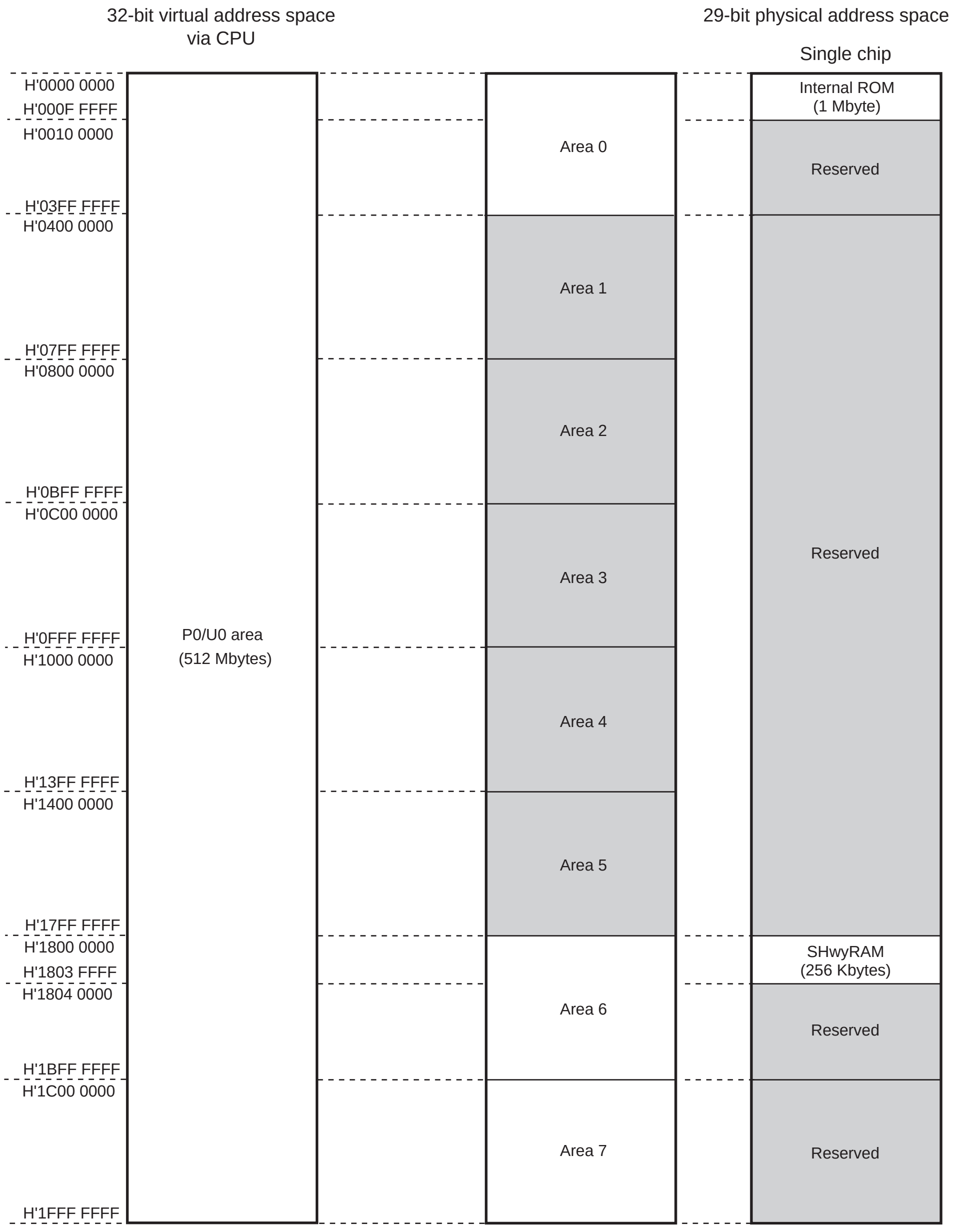
Section 11 Address Space

Figures 11.1 to 11.6 show this MCU address space. This MCU has a 32-bit (4-Gbyte) physical address space. Internal ROM and internal RAM (SHwyRAM) are mapped onto the lowest 512 Mbytes (H'0000 0000 to H'1FFF FFFF). IL memory, OL memory, and other internal resources are mapped onto the highest 512 Mbytes (H'E000 0000 to H'FFFF FFFF). The CPU can handle the 29-bit physical address space from the 32-bit virtual address space. For details, see section 7, Memory Management Unit (MMU). Specify the 32-bit physical address for the DMAC and AUDR modules. For details on the P0/U0 area to the P4 area, see figures 11.2 to 11.6.



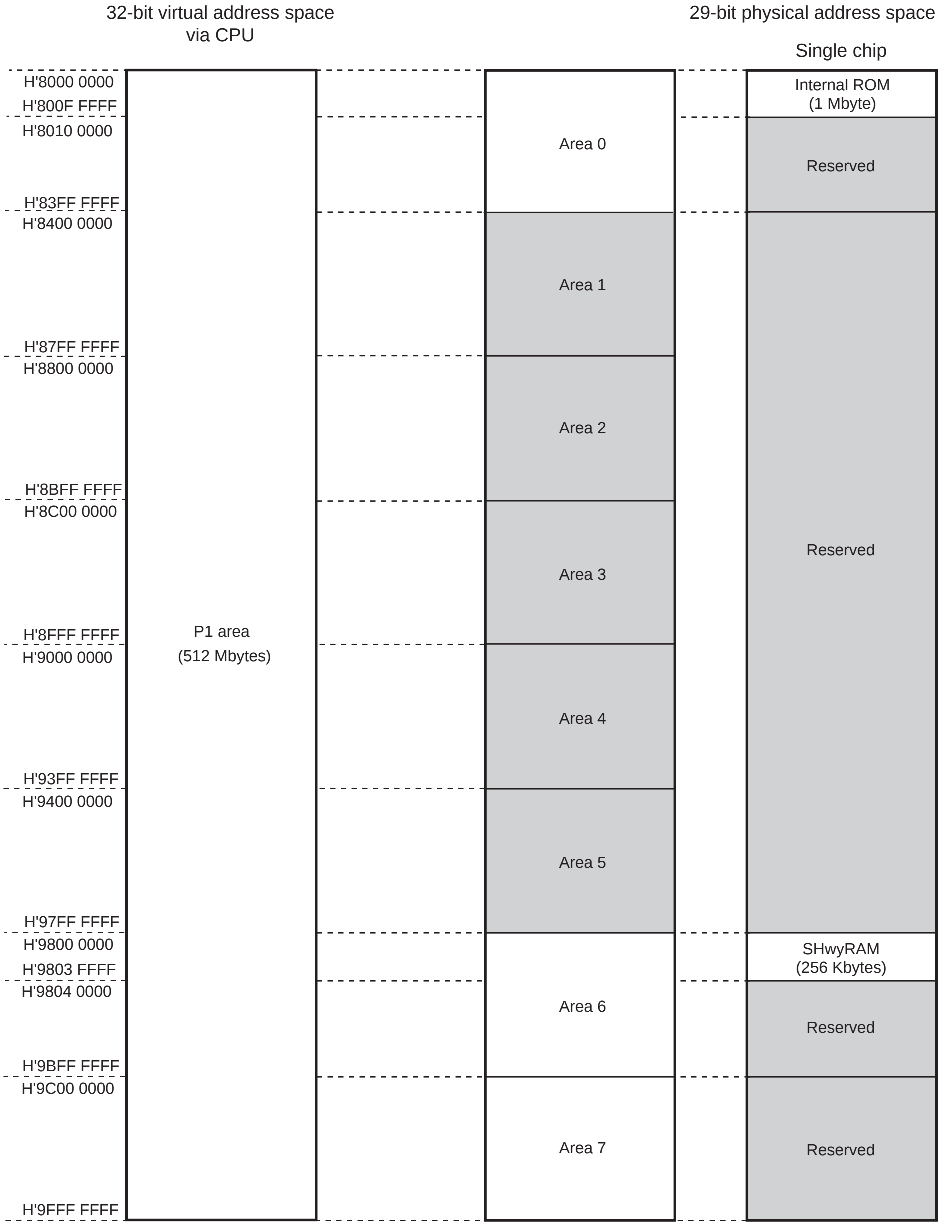
Note: The CPU, DMAC, AUDR, and other modules cannot access a reserved area.

Figure 11.1 Address Space



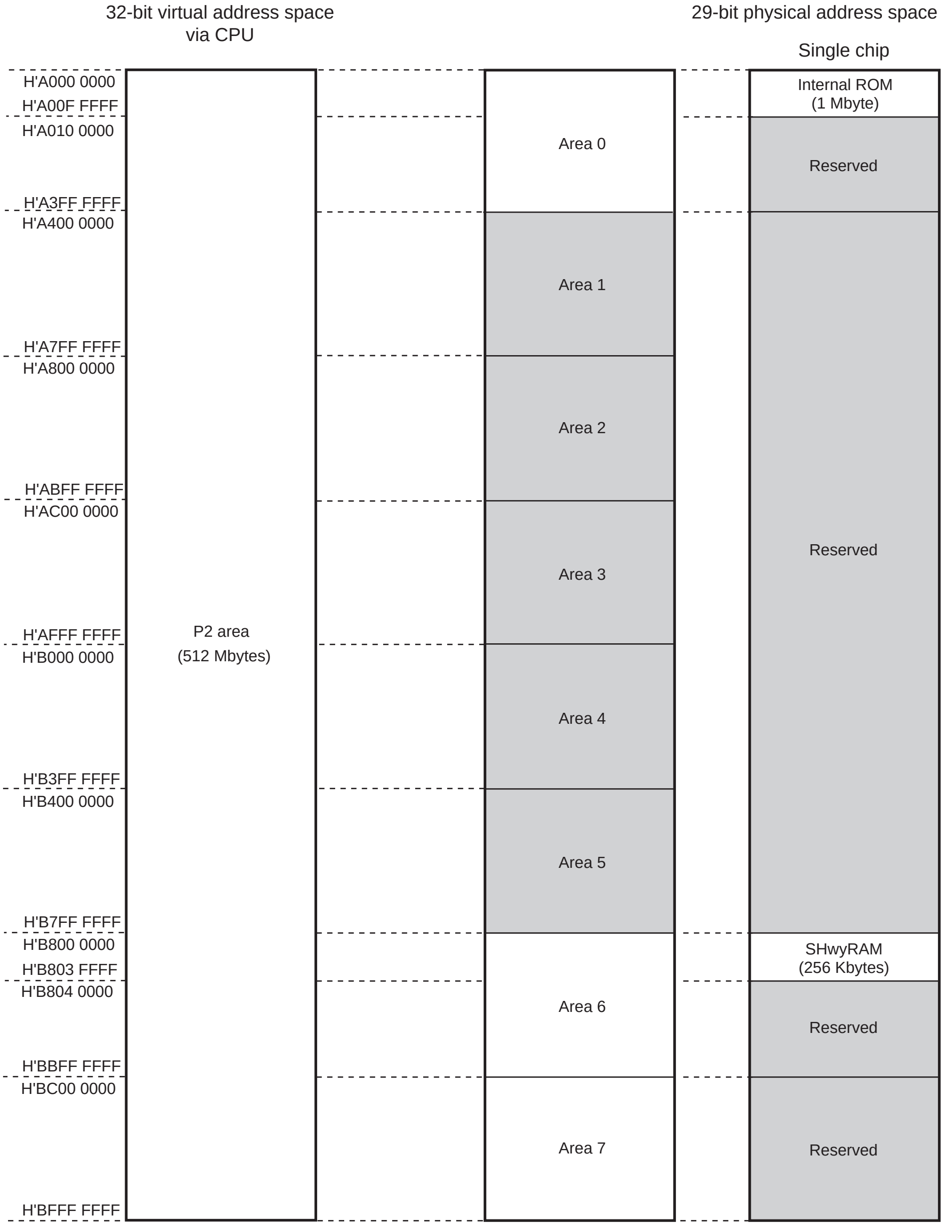
Note: The CPU, DMAC, AUDR, and other modules cannot access a reserved area.
But the CPU can access area 7 as a control register area using the MMU.

Figure 11.2 Address Space (P0/U0 Area)



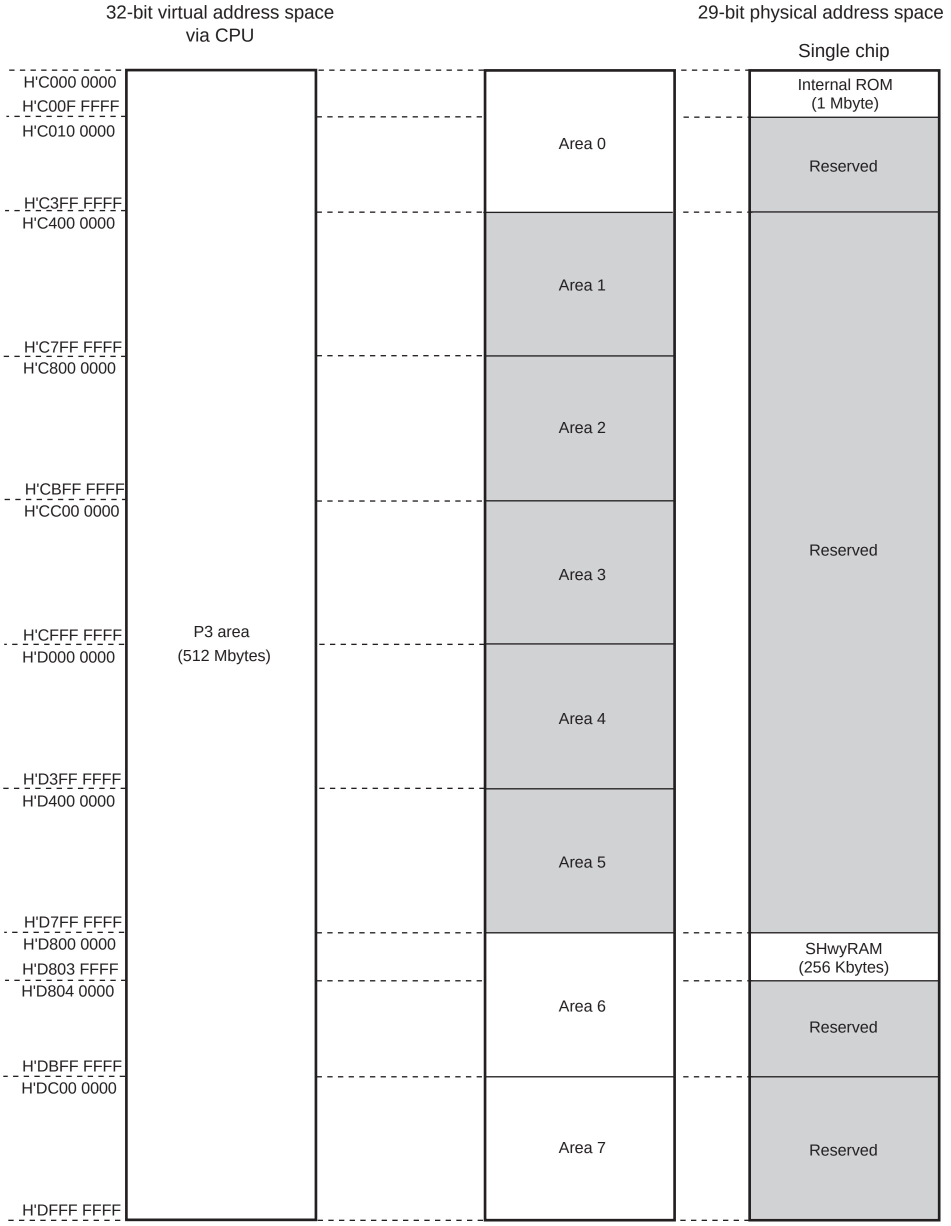
Note: The CPU, DMAC, AUDR, and other modules cannot access a reserved area.

Figure 11.3 Address Space (P1 Area)



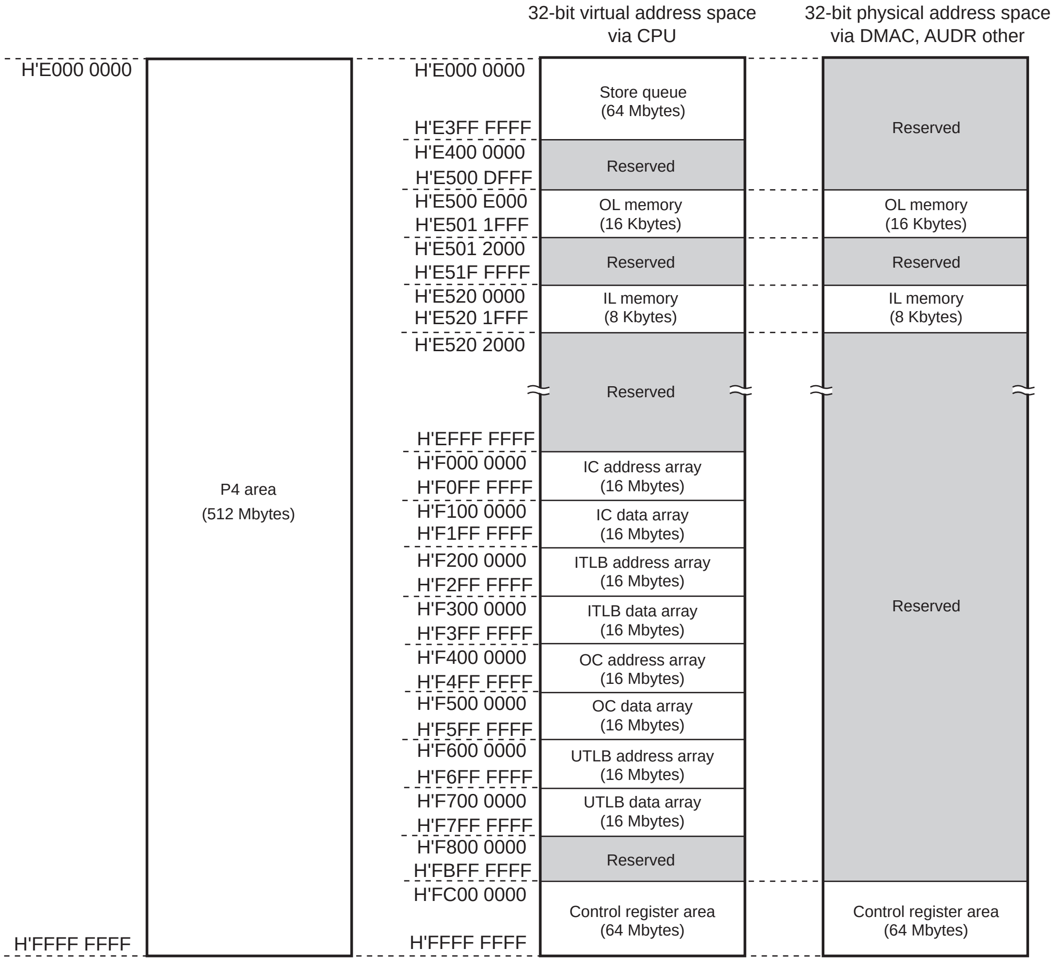
Note: The CPU, DMAC, AUDR, and other modules cannot access a reserved area.

Figure 11.4 Address Space (P2 Area)



Note: The CPU, DMAC, AUDR, and other modules cannot access a reserved area.
But the CPU can access area 7 as a control register area using the MMU.

Figure 11.5 Address Space (P3 Area)



Note: The CPU, DMAC, AUDR, and other modules cannot access a reserved area.

Figure 11.6 Address Space (P4 Area)

Section 12 ROM

The SH74552 and SH74562 each have 1 MB of on-chip flash memory (ROM) for storing program code.

12.1 Overview

- Two types of flash-memory MATs

  The ROM has two types of memory areas (hereafter referred to as memory MATs) in the same address space. These two MATs can be switched by the start-up mode or bank switching through the control register. When the user boot MAT is selected, the area from H'0000 8000 to H'000F FFFF has an undefined value when read and both writing and erasing are disabled.

  User MAT: 1 Mbyte

  User boot MAT: 32 Kbytes

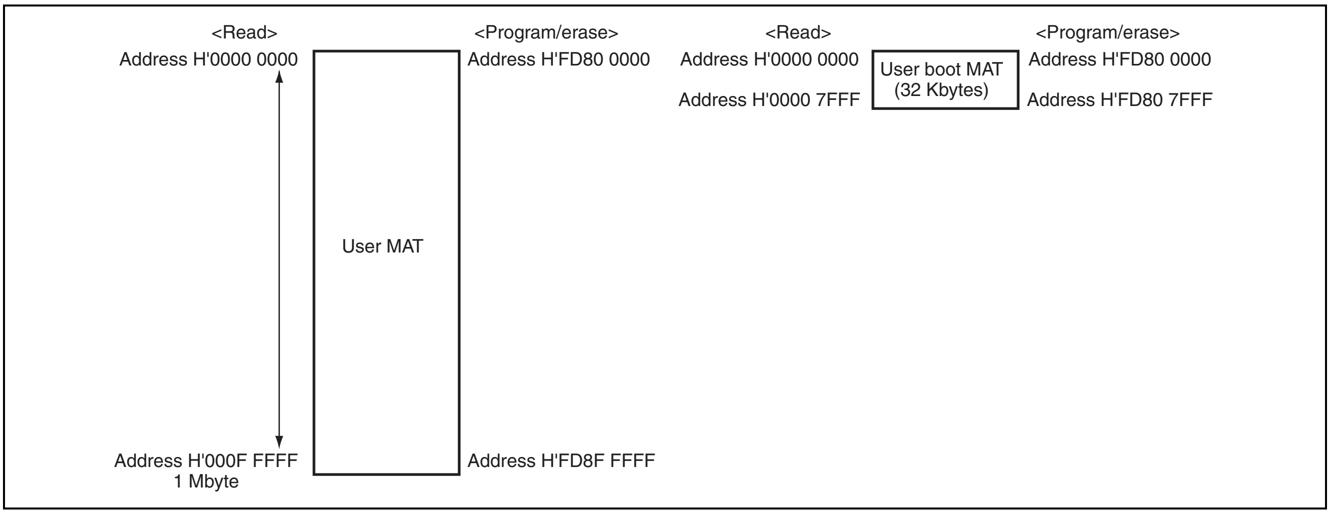


Figure 12.1 Memory MAT Configuration in ROM

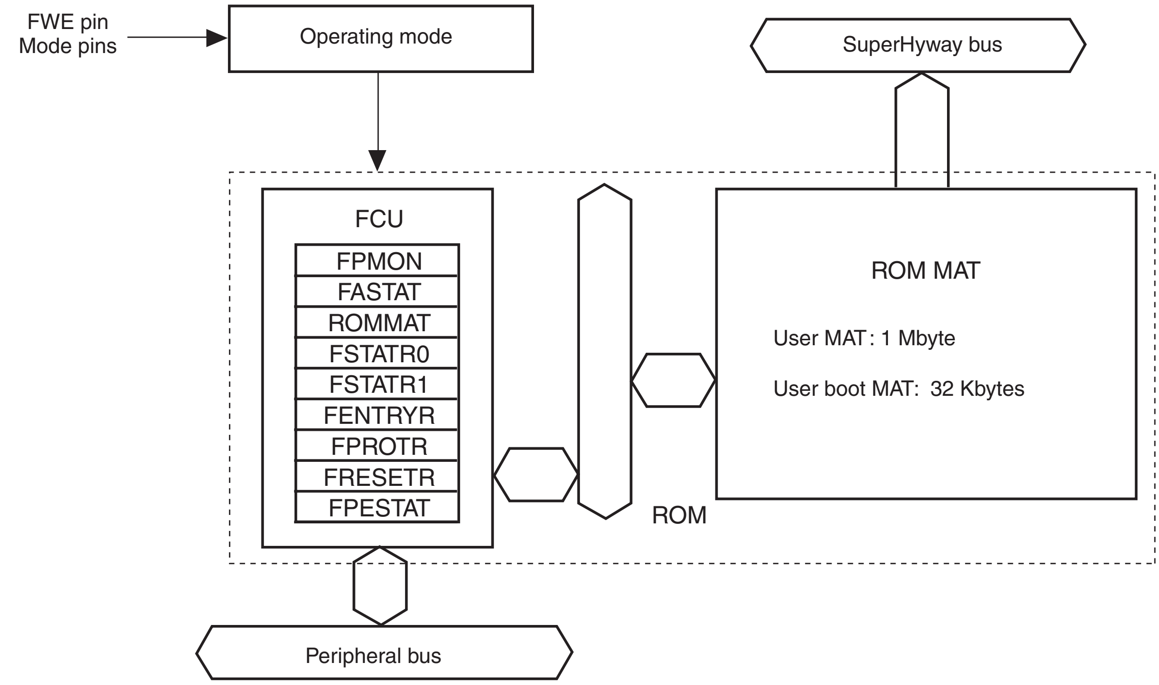
- High-speed readout over the SuperHyway bus

  Both the user MAT and user boot MAT can be read at high speed over the SuperHyway bus.

- Programming and erasing methods

  The ROM (flash memory) can be programmed or erased by issuing commands to the flash control unit (FCU) over the peripheral bus. The CPU can execute programs stored in areas other than ROM while the FCU is programming or erasing ROM.

Figure 12.2 shows the ROM block diagram.



Legend:

FCU:	Flash control unit
FPMON:	Flash pin monitor register
FASTAT:	Flash access status register
ROMMAT:	ROM MAT select register
FSTATR0, 1:	Flash status registers 0 and 1
FENTRYR:	Flash P/E mode entry register
FPROTR:	Flash protect register
FRESETR:	Flash reset register
FPESTAT:	Falsh P/E status register

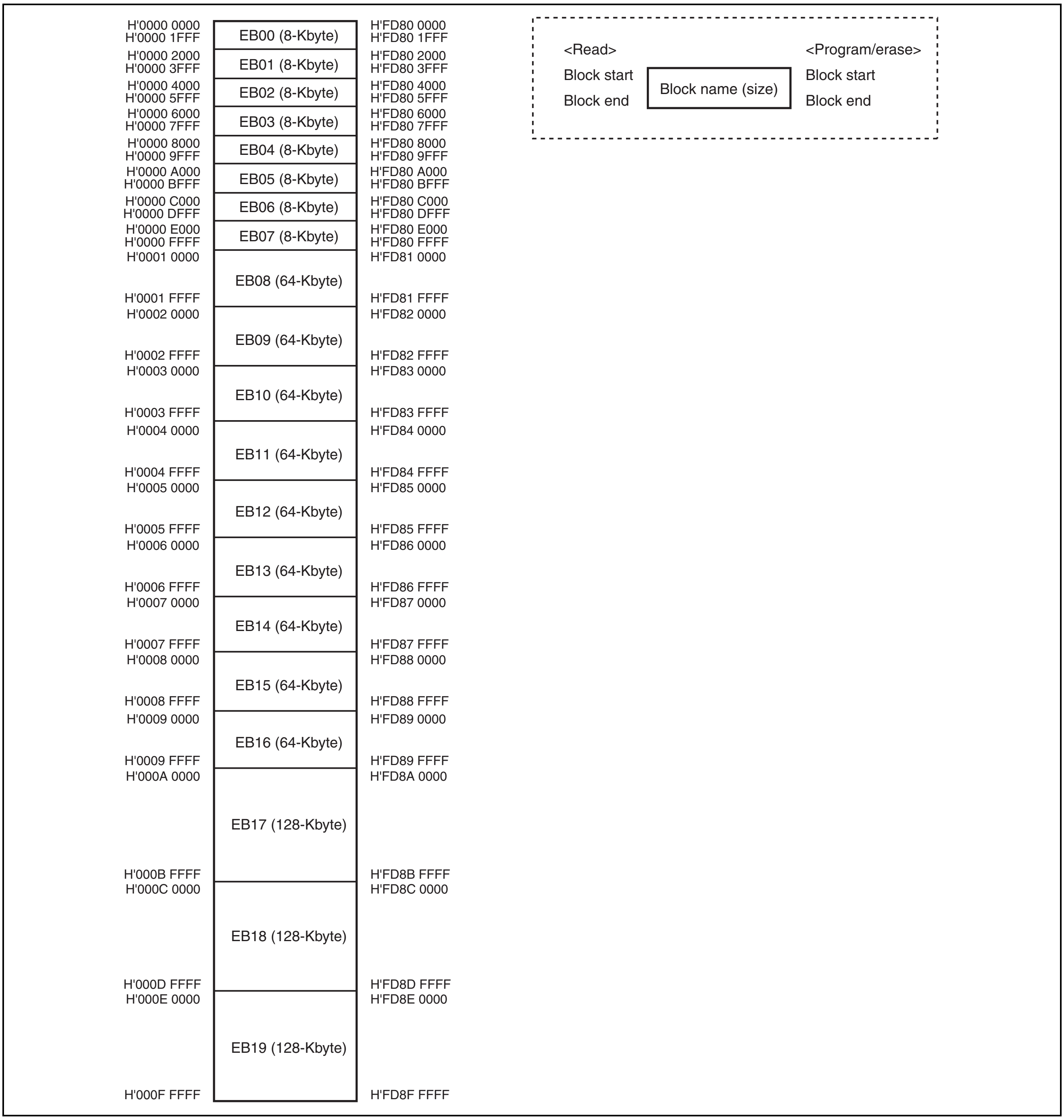
Figure 12.2 Block Diagram of ROM

- Programming/erasing unit

  The user MAT and user boot MAT are programmed in 256-byte units. The entire area of the user boot MAT is always erased at one time. The user MAT can be erased in block units.

  Figure12.3 shows the block allocation of the user MAT.

  SH74552/SH74562: 8-Kbyte (eight blocks), 64-Kbyte (nine blocks), 128-Kbyte (three blocks)



<Read> <Program/erase>

Block start	Block end	Block name (size)	Block start	Block end
H'0000 0000	H'0000 1FFF	EB00 (8-Kbyte)	H'FD80 0000	H'FD80 1FFF
H'0000 2000	H'0000 3FFF	EB01 (8-Kbyte)	H'FD80 2000	H'FD80 3FFF
H'0000 4000	H'0000 5FFF	EB02 (8-Kbyte)	H'FD80 4000	H'FD80 5FFF
H'0000 6000	H'0000 7FFF	EB03 (8-Kbyte)	H'FD80 6000	H'FD80 7FFF
H'0000 8000	H'0000 9FFF	EB04 (8-Kbyte)	H'FD80 8000	H'FD80 9FFF
H'0000 A000	H'0000 BFFF	EB05 (8-Kbyte)	H'FD80 A000	H'FD80 BFFF
H'0000 C000	H'0000 DFFF	EB06 (8-Kbyte)	H'FD80 C000	H'FD80 DFFF
H'0000 E000	H'0000 FFFF	EB07 (8-Kbyte)	H'FD80 E000	H'FD80 FFFF
H'0001 0000	H'0001 FFFF	EB08 (64-Kbyte)	H'FD81 0000	H'FD81 FFFF
H'0002 0000	H'0002 FFFF	EB09 (64-Kbyte)	H'FD82 0000	H'FD82 FFFF
H'0003 0000	H'0003 FFFF	EB10 (64-Kbyte)	H'FD83 0000	H'FD83 FFFF
H'0004 0000	H'0004 FFFF	EB11 (64-Kbyte)	H'FD84 0000	H'FD84 FFFF
H'0005 0000	H'0005 FFFF	EB12 (64-Kbyte)	H'FD85 0000	H'FD85 FFFF
H'0006 0000	H'0006 FFFF	EB13 (64-Kbyte)	H'FD86 0000	H'FD86 FFFF
H'0007 0000	H'0007 FFFF	EB14 (64-Kbyte)	H'FD87 0000	H'FD87 FFFF
H'0008 0000	H'0008 FFFF	EB15 (64-Kbyte)	H'FD88 0000	H'FD88 FFFF
H'0009 0000	H'0009 FFFF	EB16 (64-Kbyte)	H'FD89 0000	H'FD89 FFFF
H'000A 0000	H'000B FFFF	EB17 (128-Kbyte)	H'FD8A 0000	H'FD8B FFFF
H'000C 0000	H'000D FFFF	EB18 (128-Kbyte)	H'FD8C 0000	H'FD8D FFFF
H'000E 0000	H'000F FFFF	EB19 (128-Kbyte)	H'FD8E 0000	H'FD8F FFFF

Figure 12.3 User MAT and Block Allocation

- Three types of on-board programming modes
  - Boot mode

    The user MAT and user boot MAT can be programmed using the SCIF. The bit rate for SCIF communications between the host and this MCU can be automatically adjusted.
  - User mode

    The user MAT can be programmed with a desired interface. The user MAT can be rewritten by setting the FWE pin to the "H" level in normal user mode.
  - User boot mode

    The user MAT can be programmed with a desired interface. To make a transition to this mode, a hardware reset start operation is needed.
- Protection modes

  This MCU supports two modes to protect memory against programming or erasure: hardware protection by the levels on the FWE and mode pins and software protection by the FENTRY0 bit or lock bit settings. The FENTRY0 bit enables or disables ROM programming or erasure by the FCU. A lock bit is included in each erasure block of the user MAT to protect memory against programming or erasure.
- Programming and erasing, number of times for reprogramming and erasing

  Refer to section 38, Electrical Characteristics.

12.2 Input/Output Pins

Table 12.1 shows the input/output pins used for the ROM. The combination of the MD0 to MD2 pin levels and the FWE pin level determines the ROM programming mode (see section 12.4, Overview of ROM-Related Modes). In boot mode, the ROM can be programmed or erased by the host connected via the PJ10/RXD0 and PJ11/TXD0 pins (see section 12.5, Boot Mode).

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 12.1 Pin Configuration

Pin Name	I/O	Function
RESET#	Input	This MCU goes to the hardware reset state when this pin is set "L".
MD0 to MD2	Input	These pins specify the operating mode.
FWE	Input	This pin enables or disables ROM programming.
PJ10/RXD0	Input	SCIF0 receive data (for host transmission)
PJ11/TXD0	Output	SCIF0 transmit data (for host transmission)

12.3 Register Descriptions

Table 12.2 shows the ROM-related registers. The ROM-related registers are initialized by a hardware reset.

Table 12.2 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Flash pin monitor register	FPMON	H'00 H'80	H'FDFF A800	8	12-6
Flash access status register	FASTAT	H'00	H'FDFF A810	8	12-7
ROM MAT select register	ROMMAT	H'0000 H'0001	H'FDFF A820	8, 16	12-8
Flash status register 0	FSTATR0	H'80*[1]	H'FDFF A900	8, 16	12-9
Flash status register 1	FSTATR1	Undefined*[1]	H'FDFF A901	8, 16	12-11
Flash P/E mode entry register	FENTRYR	H'0000*[1]	H'FDFF A902	8, 16	12-12
Flash protect register	FPROTR	H'0000*[1]	H'FDFF A904	8, 16	12-14
Flash reset register	FRESETR	H'0000	H'FDFF A906	8, 16	12-15
Flash P/E status register	FPESTAT	H'0000*[1]	H'FDFF A91C	8, 16	12-16

Notes: *1 Can be initialized either by a hardware reset or by writing "1" to the FRESET bit in the FRESETR register.

- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

12.3.1 Flash Pin Monitor Register (FPMON)

The FPMON register monitors the FWE pin state.

Flash Pin Monitor Register (FPMON) <P4 address: location H'FDFF A800>

Bit:	7	6	5	4	3	2	1	0
	FWE	—	—	—	—	—	—	—
After Reset:	1/0	0	0	0	0	0	0	0

<After Reset: H'00, H'80>

Bit	Abbreviation	After Reset	R	W	Description
7	FWE	1/0	R	—	Flash Write Enable Bit This bit is provided to monitor the value of the FWE pin. Thus the initial value changes depending on the FWE pin input level when the chip is powered on. When the chip is powered on with the FWE pin at "L" level, the FWE bit can be set to "1" by driving the FWE pin to "H" level during the MCU operation. The FWE bit can be then set to "0" by driving the FWE pin to "L" level. When the chip is activated with the FWE pin at "H" level, the FWE bit remains as "1" and will not be affected by the FWE pin. To enable protection after reprogramming, drive the FWE pin to "L" level and power on the chip again. 0: Disables ROM programming and erasure 1: Enables ROM programming and erasure
6 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

12.3.2 Flash Access Status Register (FASTAT)

The FASTAT register indicates the access error status for the ROM. If any bit in the FASTAT register is set to "1", the FCU enters command-locked state (see section 12.8.3, Error Protection). To cancel a command-locked state, set the FASTAT register to H'10, and then issue a status-clear command to the FCU.

Flash Access Status Register (FASTAT) <P4 address: location H'FDFF A810>

Bit:	7	6	5	4	3	2	1	0
	ROM AE	—	—	CMD LK	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	ROMAE	0	R	*[1]	Access Error Bit Indicates whether or not a ROM access error has been generated. If this bit becomes "1", the ILGLERR bit in the FSTATR0 register is set to "1" and the FCU enters a command-locked state. 0: No ROM access error has occurred. 1: A ROM access error has occurred. [Condition for clearing to "0"] • When "0" is written after reading out ROMAE with the value "1". [Conditions for setting to "1"] • A read access command is issued to ROM program/erase addresses H'FD80 0000 to H'FD8F FFFF while the FENTRY0 bit in the FENTRYR register is "1" in ROM P/E normal mode. • An access command is issued to ROM program/erase addresses H'FD80 0000 to H'FD8F FFFF while the FENTRY0 bit in the FENTRYR register is "0". • A read access command is issued to ROM read addresses H'0000 0000 to H'000F FFFF while the FENTRYR register value is not H'0000. • A block erase, program, or lock bit program command is issued to ROM when the user boot MAT is selected. • An access command is issued to an address other than ROM program/erase addresses H'FD80 0000 to H'FD8F FFFF when the user boot MAT is selected.
6, 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
4	CMDLK	0	R	—	FCU Command Lock Bit Indicates whether the FCU is in command-locked state (see section 12.8.3, Error Protection). 0: The FCU is not in a command-locked state 1: The FCU is in a command-locked state [Condition for clearing to "0"] • The FCU completes the status-clear command processing while the FASTAT register is H'10. [Condition for setting to "1"] • The FCU detects an error and enters command-locked state.
3 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Note: *1 Writing a "0" after reading a "1" is only allowed in order to clear the flag.

12.3.3 ROM MAT Select Register (ROMMAT)

The ROMMAT register switches memory MATs in the ROM. Writing to the ROMMAT register is enabled only when a specified value is written to the high-order byte in word access.

ROM MAT Select Register (ROMMAT) <P4 address: location H'FDFF A820>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KEY								—	—	—	—	—	—	—	ROM SEL
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1

<After Reset: H'0000, H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	KEY	All 0	0	W	ROMMAT Register Write Key Code Bits These bits enable or disable ROMSEL bit modification. The data written to these bits are not retained. These bits are always read as "0". H'3B: Enable ROMSEL bit modification. Other than H'3B: Disable ROMSEL bit modification.
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	ROMSEL	0/1	R	W	ROM MAT Select Bit Selects a memory MAT in the ROM. The initial value is "1" when the MCU is started in user boot mode; otherwise, the initial value is "0". Writing to this bit is enabled only when this register is accessed in word size and H'3B is written to the KEY bits. 0: Selects the user MAT 1: Selects the user boot MAT

12.3.4 Flash Status Register 0 (FSTATR0)

The FSTATR0 register indicates the FCU status. The FRTATR0 register is initialized by a hardware reset, or setting the FRESET bit of the FRESETR register to "1".

Flash Status Register 0 (FSTATR0) <P4 address: location H'FDFF A900>

Bit:	7	6	5	4	3	2	1	0
	FRDY	ILGL ERR	ERS ERR	PRG ERR	FCUSQ	—	—	—
After Reset:	1	0	0	0	0	0	0	0

<After Reset: H'80>

Bit	Abbreviation	After Reset	R	W	Description
7	FRDY	1	R	—	Flash Ready Bit Indicates the processing state in the FCU. 0: Programming or erasure processing, or lock bit read command processing. 1: None of the above is in progress.
6	ILGLERR	0	R	—	Illegal Command Error Bit Indicates that the FCU has detected an illegal command or illegal ROM access. When this bit is "1", the FCU is in command-locked state (see section 12.8.3, Error Protection). 0: The FCU has not detected any illegal command or illegal ROM access 1: The FCU has detected an illegal command or illegal ROM access [Condition for clearing to "0"] • The FCU completes the status-clear command processing while the FASTAT register is H'10. [Conditions for setting to "1"] • The FCU has detected an illegal command. • The FCU has detected an illegal ROM access (the ROMAE bit in the FASTAT register is "1"). • The FENTRYR register setting is illegal.
5	ERSERR	0	R	—	Erasure Error Bit Indicates the result of ROM erasure by the FCU. When this bit is "1", the FCU is in command-locked state (see section 12.8.3, Error Protection). 0: Erasure processing has been completed successfully 1: An error has occurred during erasure [Condition for clearing to "0"] • The FCU completes the status-clear command processing. [Conditions for setting to "1"] • An error has occurred during erasure. • A block erase command has been issued for the area protected by a lock bit.

Bit	Abbreviation	After Reset	R	W	Description
4	PRGERR	0	R	—	Programming Error Bit Indicates the result of ROM programming by the FCU. When this bit is "1", the FCU is in command-locked state (see section 12.8.3, Error Protection). 0: Programming has been completed successfully 1: An error has occurred during programming [Condition for clearing to "0"] • The FCU completes the status-clear command processing. [Conditions for setting to "1"] • An error has occurred during programming. • A programming command has been issued for the area protected by a lock bit.
3	FCUSQ	0	R	—	Sequence Operation Bit Indicates the programming/erasure sequence has been entered. This bit is set to "1" during programming/erasure. During that period, do not specify the FCUSQ bit but only the FRDY bit for polling, and check if the process is completed. 0: Sequence stopped 1: Sequence is in progress
2 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

12.3.5 Flash Status Register 1 (FSTATR1)

The FSTATR1 register indicates the FCU status. The FSTATR1 register is initialized by a hardware reset, or setting the FRESET bit of the FRESETR register to "1".

Flash Status Register 1 (FSTATR1) <P4 address: location H'FDFF A901>

Bit:	7	6	5	4	3	2	1	0
	FCU ERR	—	—	FLO CKST	—	—	FRD TCT	—
After Reset:	0	0	0	0	0	0	0	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7	FCUERR	0	R	—	FCU Error Bit Indicates an error has occurred in the FCU. If the FCUERR bit is set to "1", initialize this MCU by applying a hardware reset. 0: No error has occurred in the FCU 1: An error has occurred in the FCU [Condition for clearing to "0"] • The FRESET bit in the FRESETR register is set to "1".
6, 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	FLOCKST	0	R	—	Lock Bit Status Bit Reflects the lock bit data read through lock bit read command execution. When the FRDY bit becomes "1" after the lock bit read command is issued, valid data is stored in this bit. This bit value is retained until the next lock bit read command is completed. 0: Protected state 1: Non-protected state
3, 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	FRDTCT	0	R	—	FCU Data Access Error Bit Indicates that an error has occurred during an FCU internal data access. If the FRDTCT bit is set to "1", initialize this MCU by applying a hardware reset. 0: No FCU data access error has occurred. 1: An FCU data access error has occurred.
0	—	Undefined	R	—	Reserved Bit The value read from this bit is undefined. Applications should always mask this bit when reading the FSTATR1 register.

12.3.6 Flash P/E Mode Entry Register (FENTRYR)

The FENTRYR register specifies the P/E mode for the ROM. Writing to the FENTRYR register is enabled only when a specified value is written to the high-order byte. Writing any other value initializes this register. To specify the P/E mode for the ROM so that the FCU can accept commands, set the FENTRY0 bit to "1". Note that if this register is set to a value other than H'0001 or H'0002, the ILGLERR bit in the FSTATR0 register will be set to "1" and the FCU will switch to the command-locked state.

The FENTRYR register can be initialized by a hardware reset, or setting the FRESET bit in the FRESETR register to "1".

Flash P/E Mode Entry Register (FENTRYR) <P4 address: location H'FDFF A902>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FEKEY								—	—	—	—	—	—	FENTRY1	FENTRY0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	FEKEY	All 0	0	W	FENTRYR Register Write Key Code Bits These bits enable or disable FENTRY0 bit modification. The data written to these bits are not retained. These bits are always read as "0". H'AA: Enable FENTRY0 bit modification. Other than H'AA: Disable FENTRY0 bit modification.
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	FENTRY1	0	0	0	ROM P/E Mode Entry Bit 1 This bit is not supported by the MCU. Always write "0" to FENTRY1.

Bit	Abbreviation	After Reset	R	W	Description
0	FENTRY0	0	R	W	ROM P/E Mode Entry Bit 0 These bits specify the P/E mode for the EB00 to EB19 blocks of ROM (read addresses: H'0000 0000 to H'000F FFFF; program/erase addresses: H'FD80 0000 to H'FD8F FFFF). 0: The block of ROM from EB00 to EB19 (1 Mbyte) is in read mode 1: The block of ROM from EB00 to EB19 (1 Mbyte) is in P/E mode Programming is enabled when the following conditions are all satisfied: • The FWE bit in the FPMON register is "1". • The FRDY bit in the FSTATR0 register is "1". • H'AA is written to the FEKEY bit in word access. [Conditions for clearing to "0"] • The FRDY bit in the FSTATR0 register becomes "1" and the FWE bit in the FPMON register becomes "0". • This register is written to in byte access. • A value other than H'AA is written to the FEKEY bit in word access. • "0" is written to the FENTRY0 bit while the write enabling conditions are satisfied. • The FENTRYR register is written to while the FENTRYR register is not H'0000 and the write enabling conditions are satisfied. [Condition for setting to "1"] • "1" is written to FENTRY0 while the write enabling conditions are satisfied and the FENTRYR register is H'0000.

12.3.7 Flash Protect Register (FPROTR)

The FPROTR register enables or disables the protection function through the lock bits against programming and erasure. Writing to the FPROTR register is enabled only when a specified value is written to the high-order byte in word access. Writing any other value initializes this register.

The FPROTR register is initialized by a hardware reset, or setting the FRESET bit in the FRESETR register to "1".

Flash Protect Register (FPROTR) <P4 address: location H'FDFF A904>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FPKEY								—	—	—	—	—	—	—	FPROTCN
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	FPKEY	All 0	0	W	FPROTR Register Write Key Code Bits These bits enable or disable FPROTCN bit modification. The data written to these bits are not retained. These bits are always read as "0". H'55: Enable FPROTCN bit modification. Other than H'55: Disable FPROTCN bit modification.
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	FPROTCN	0	R	W	Lock Bit Protect Cancel Bit Enables or disables protection through the lock bits against programming and erasure. 0: Enables protection through the lock bits 1: Disables protection through the lock bits [Conditions for clearing to "0"] • This register is written to in byte access. • A value other than H'55 is written to the FPKEY bit in word access. • H'55 is written to the FPKEY bit and "0" is written to the FPROTCN bit in word access. • The FENTRYR register value is H'0000. [Condition for setting to "1"] • H'55 is written to FPKEY and "1" is written to the FPROTCN bit in word access while the FENTRYR register value is not H'0000.

12.3.8 Flash Reset Register (FRESETR)

The FRESETR register is used for the initialization of the FCU and ROM. Writing to the FRESETR register is enabled only when a specified value is written to the high-order byte in word access.

Flash Reset Register (FRESETR) <P4 address: location H'FDFF A906>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FRKEY								—	—	—	—	—	—	—	FRESET
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	FRKEY	All 0	0	W	FRESETR Register Write Key Code Bits These bits enable or disable FRESET bit modification. The data written to these bits are not retained. These bits are always read as "0". H'CC: Enable FRESET bit modification. Other than H'CC: Disable FRESET bit modification.
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	FRESET	0	R	W	Flash Reset Bit Setting this bit to "1" forcibly terminates programming/erasure of ROM and initializes the FCU. A high voltage is applied to the ROM memory units during programming and erasure. To ensure sufficient time for the voltage applied to the memory unit to drop, keep the value of the FRESET bit at "1" for a period of t_{RESW2} (see section 38, Electrical Characteristics) when the FCU is initialized. Do not read from the ROM units while the value of the FRESET bit is kept at "1". The FCU commands are unavailable for use while the FRESET bit is set to "1", since this initializes the FENTRYR register. This bit can be written only when H'CC is written to the FRKEY bit in word access. 0: Issue no reset to the FCU. 1: Issues a reset to the FCU.

12.3.9 Flash P/E Status Register (FPESTAT)

The FPESTAT register indicates the result of programming/erasure of the ROM. The FPESTAT register is initialized by a hardware reset, or setting the FRESET bit in the FRESETR register to "1".

Flash P/E Status Register (FPESTAT) <P4 address: location H'FDFF A91C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PEERRST							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7 to 0	PEERRST	H'00	R	—	P/E Error Status Bits Indicates the source of an error that occurs during programming/erasure of the ROM. This bit value is only valid if the PRGERR or ERSERR bit value in the FSTATR0 register is "1"; otherwise the bit retains the value to indicate the source of an error that previously occurred. H'01: A write attempt made to an area protected by the lock bits H'02: A write error caused by other source than the above H'11: An erase attempt made to an area protected by the lock bits H'12: An erase error caused by other source than the above Other than above: Reserved

12.4 Overview of ROM-Related Modes

Figure 12.4 shows the ROM-related mode transition in this MCU. For the relationship between the MCU operating modes and the MD0 to MD2 and FWE pin settings, refer to section 10, Operating Modes.

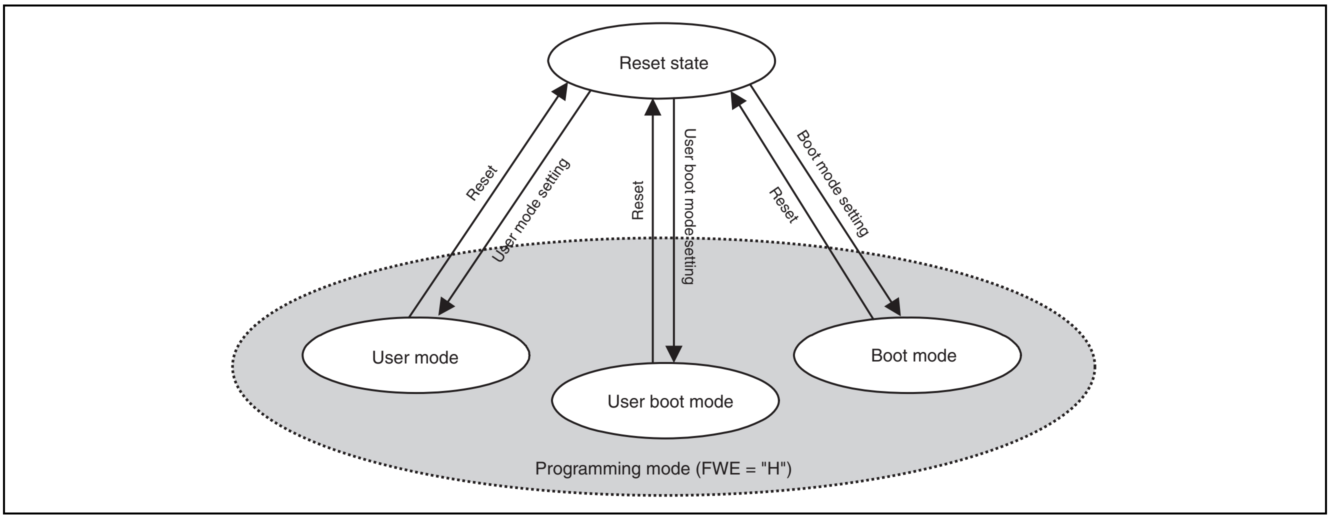


Figure 12.4 ROM-Related Mode Transition

- Although ROM can be read when the FWE pin is at the "L" level, the program (write) and erase operations cannot be used.
- ROM can be read, written, and erased if the FWE pin is at the "H" level.

Table 12.3 compares programming- and erasure-related items for the boot mode, user mode, and user boot mode.

Table 12.3 Comparison of Programming Modes

Item	Boot Mode	User Mode	User Boot Mode
Programming/erasure environment	On-board programming		
Programming/erasure enabled MAT	User MAT and user boot MAT	User MAT	User MAT
Programming/erasure control	Host	FCU	FCU
Entire area erasure	Available	Available	Available
Block erasure	Available	Available	Available
Programming data transfer	From host via SCIF	From any device via RAM	From any device via RAM
Reset-start MAT	Embedded program stored MAT	User MAT	User boot MAT

- The user boot MAT can be programmed or erased only in boot mode.
- In boot mode, the user MAT and user boot MAT can be programmed from the host via the SCIF after the key code is authenticated.
- In user boot mode, a boot operation with a desired interface can be implemented through mode pin settings different from those in boot mode.

12.5 Boot Mode

12.5.1 System Configuration

To program or erase the user MAT and user boot MAT in boot mode, send control commands and programming data from the host. The on-chip SCIF of this MCU is used in asynchronous mode for communications between the host and this MCU. The tool for sending control commands and programming data must be prepared in the host. When this MCU is started in boot mode, the program in the embedded program stored MAT is executed. This program automatically adjusts the SCIF bit rate and performs communications between the host and this MCU by means of the control command method.

Figure 12.5 shows the system configuration in boot mode. The NMI, IRQ7 to IRQ5, and IRQ2 to IRQ0 interrupts are ignored in this mode, but these pins must be fixed to non-active state.

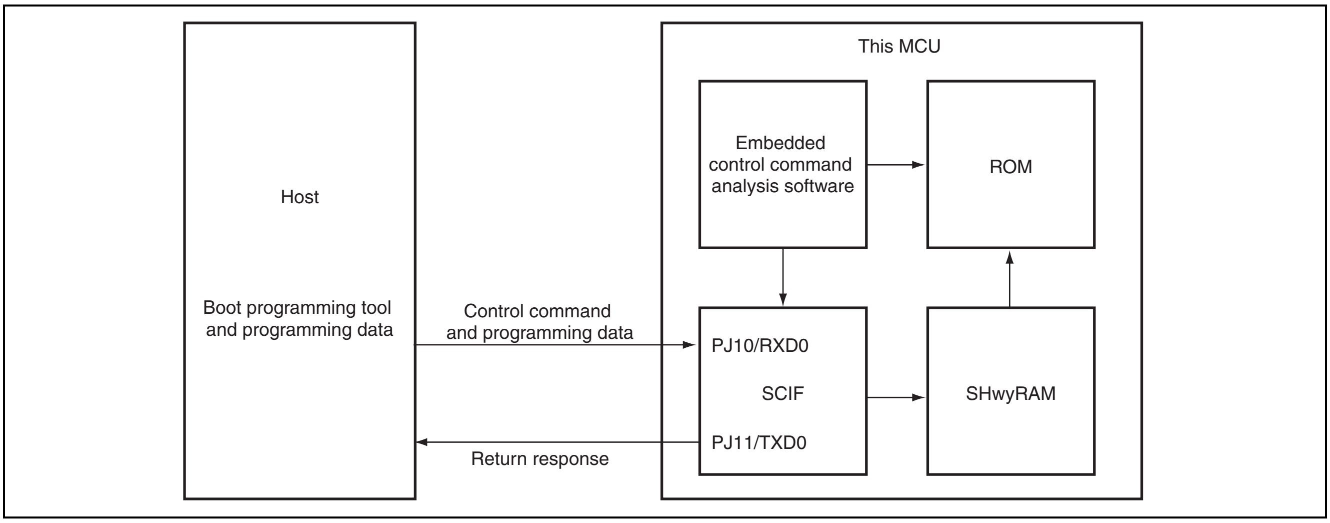

This MCU
Host
Boot programming tool and programming data
Control command and programming data
Return response
Embedded control command analysis software
ROM
PJ10/RXD0
SCIF
PJ11/TXD0
SHwyRAM

Figure 12.5 System Configuration in Boot Mode

12.6 User Mode and User Boot Mode

12.6.1 FCU Command List

To program or erase the user MAT in user mode and user boot mode, issue FCU commands to the FCU. Table 12.4 is a list of FCU commands for ROM programming and erasure.

Table 12.4 FCU Command List (ROM-Related Commands)

Command	Function
Program	Programs ROM (in 256-byte units)
Block erase	Erases ROM (in block units; erasing the lock bit)
Status register clear	Clears bits ILGLERR, ERSERR, and PRGERR in the FSTATR0 register and cancels the command-locked state
Lock bit read	Reads the lock bit of a specified erasure block (updates the FLOCKST bit in the FSTATR1 register to reflect the lock bit state)
Lock bit program	Writes to the lock bit of a specified erasure block

To issue a command to the FCU, write to a ROM program/erase address through the peripheral bus. Table 12.5 shows the FCU command format. Performing peripheral-bus write access as shown in table 12.5 under specified conditions starts each command processing in the FCU. For the conditions for FCU command acceptance, refer to section 12.6.2, Conditions for FCU Command Acceptance. For details of each FCU command, refer to section 12.6.3, FCU Command Usage.

Table 12.5 FCU Command Format

Command	Number of Command Cycles*[1]	First Cycle		Second Cycle		Third Cycle		Fourth to 130th Cycles		131st Cycle	
		Address	Data	Address	Data	Address	Data	Address	Data	Address	Data
Program	131	RA	H'E8	RA	H'80	WA	WD1	RA	WDn	RA	H'D0
Block erase	2	RA	H'20	BA	H'D0	—	—	—	—	—	—
Status register clear	1	RA	H'50	—	—	—	—	—	—	—	—
Lock bit read	2	RA	H'71	BA	H'D0	—	—	—	—	—	—
Lock bit program	2	RA	H'77	BA	H'D0	—	—	—	—	—	—

Note: *1 The number of command cycles is the number of issued times of peripheral bus write access to the program/erasure address.

Legend:

RA: ROM program/erase address
When the FENTRY0 bit is "1": An address in the range from H'FD80 0000 to H'FD8F FFFF

WA: ROM program address
Start address of 256-byte programming data

BA: ROM erasure block address
An address in the target erasure block (specified by the ROM program/erase address)

WDn: n-th word of programming data (n = 2 to 128)

12.6.2 Conditions for FCU Command Acceptance

The FCU determines whether to accept a command depending on the FCU mode or status. Figure 12.6 is an FCU mode transition diagram.

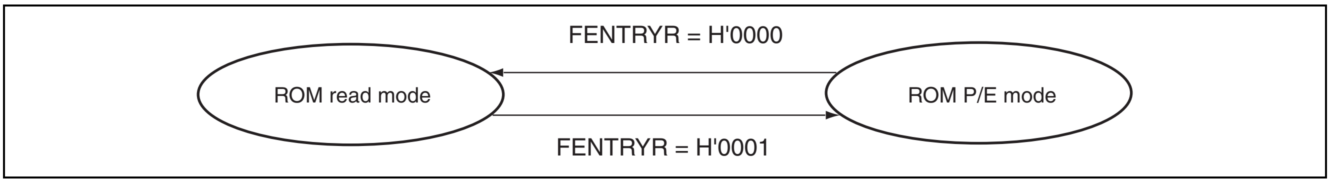


Figure 12.6 FCU Mode Transition Diagram (ROM-Related Modes)

(1) ROM read mode

In this mode, ROM can be read out at high speed over the SHwy bus. FCU commands are not accepted. This MCU switches to this mode when the FENTRY0 bit in the FENTRYR register is set to "0".

(2) ROM P/E mode

The FCU enters this mode when the FENTRY0 bit is set to "1". Table 12.6 lists the commands that the FCU accepts. The high-speed ROM readout operation cannot be used in this mode. Although read access to locations H'FD80 0000 to H'FD8F FFFF is illegal, undefined values will be returned. To read the ROM data, the FCU must switched to ROM read mode. If a peripheral-bus read access to a location from H'FD80 0000 to H'FD8F FFFF is issued in the state where the FENTRY0 bit is "1", a ROM access error will occur and the FCU will switch to the command-locked state. (See section 12.8.3, Error Protection.)

The FCU switches to the command-locked state whenever a command that cannot be accepted is issued. (See section 12.8.3, Error Protection.)

To assure that the FCU accepts a command, applications must switch the FCU to a mode in which the command can be accepted and, after verifying the values of bits FRDY, ILGLERR, ESERR, and PRGERR in the FSTATR0 register and the values of bits FCUERR and FRDTCT in the FSTATR1 register, only then issue the FCU command. Note that the value of the CMDLK bit in the FASTAT register is the logical OR of the values of bits ILGLERR, ERSERR, and PRGERR in the FSTATR0 register and the FCUERR bit in the FSTATR1 register. Therefore, the FCU error occurrence status can also be verified by checking the CMDLK bit. In table 12.6, the CMDLK bit is used as a bit that indicates the error occurrence status. The FRDY bit in the FSTATR0 register will be "0" during program/erase processing and during lock bit read processing.

Table 12.6 FCU Modes/States and Acceptable Commands

Item	Other State	Programming/ Erasure Processing	Programming/ Erasure Suspension Processing	Lock Bit Read Processing	Command-Locked
FRDY bit in FSTATR0	1	0	0	0	0/1
CMDLK bit in FASTAT	0	0	0	0	1
Program	○	×	×	×	×
Block erase	○	×	×	×	×
Status register clear	○	×	×	×	○
Lock bit read	○	×	×	×	×
Lock bit program	○	×	×	×	×

Legend:

○: Acceptable

×: Not acceptable

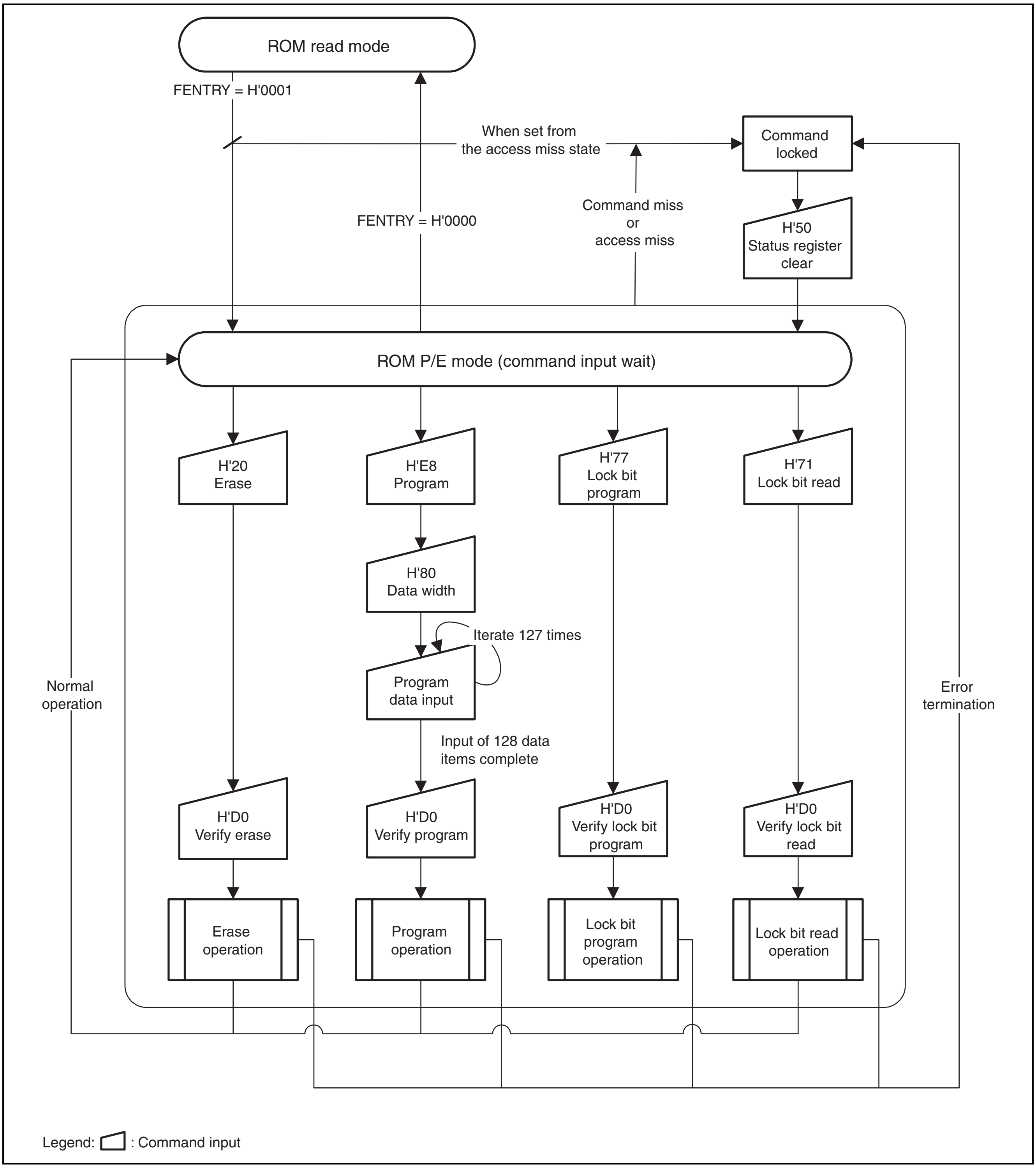


Figure 12.7 Command State Transitions in ROM Read Mode and P/E Mode

12.6.3 FCU Command Usage

This section presents an example of the user processing when issuing FCU commands.

The procedure presented here uses bits FRDY, ILGLERR, ERSERR, and PRGERR in the FSTATR0 register and bits FCUERR and FRDTCT in the FSTATR1 register to verify the FCU command processing status and the error occurrence status. Since registers FSTATR0 and FSTATR1 can be read at the same time with a word access, the FCU status can be verified with a single register access. When the method that verifies the FCU status with the FRDY bit in the FSTATR0 register and the CMDLK bit in the FASTAT register is used, while two register access are required, whether the error is occurred or not can be confirmed with just the CMDLK bit.

If, during FCU command processing, the FCU switches to the command-locked state due to the FCUERR bit or the FRDTCT bit being set to "1", the FRDY bit will retain the "0" value. Since FCU command processing is stopped in the command-locked state, the FRDY bit will not be set from "0" to "1". If the FRDY bit is held in the "0" state for a period longer than the program/erase time (see section 38, Electrical Characteristics), it is possible that abnormal operation such as FCU processing stopping in the command-locked state may occur. Thus the FCU should be reinitialized if that occurs. If FCU command processing completes and the FRDY bit is set to "1", the FCUERR bit and the FRDTCT bit will, in all cases, be in the "0" state. Therefore, the error occurrence status after command processing termination can be determined from the states of bits ILGLERR, ERSERR, and PRGERR.

(1) Methods for switching to ROM P/E mode

For an application to execute ROM related FCU commands, it is necessary to set the FCU to ROM P/E mode by setting the FENTRY0 bit in the FENTRYR register. (See section 12.6.2, Conditions for FCU Command Acceptance.) To use ROM related FCU commands, set the FENTRY0 bit to "1". See section 12.3.6, Flash P/E Mode Entry Register (FENTRYR) for the conditions for setting the FENTRY0 bit.

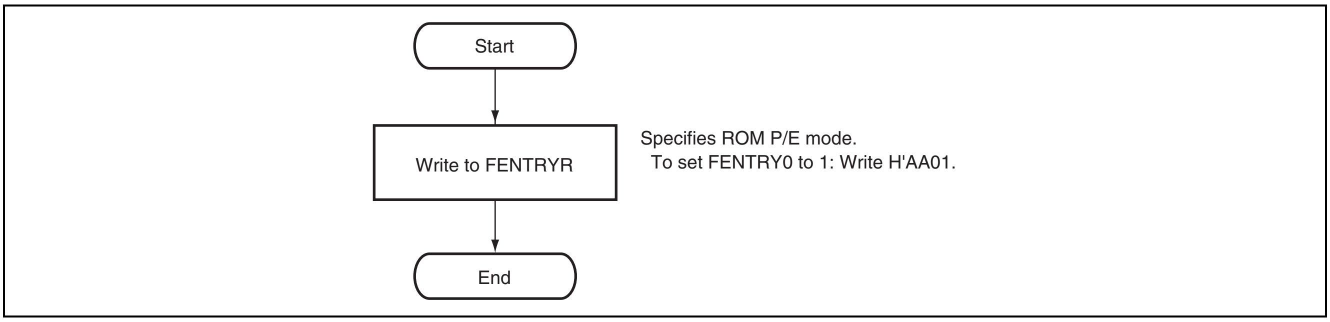
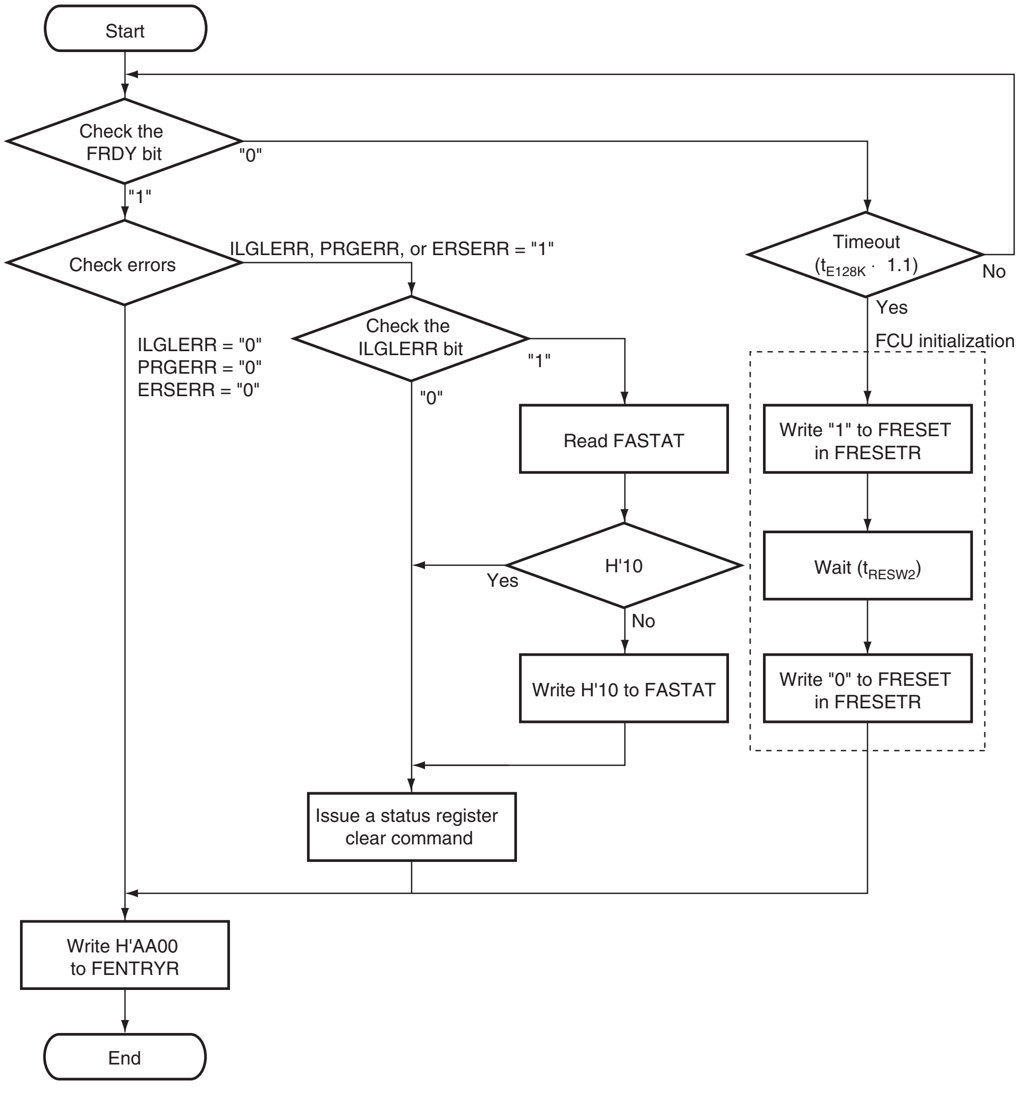


Figure 12.8 Procedure for Transition to ROM P/E Mode

(2) Entering ROM Read Mode

To enable high-speed ROM read access over the SuperHyway bus, it is necessary to set the FCU to ROM read mode by clearing the FENTRY0 bit in the FENTRYR register. (See section 12.6.2, Conditions for FCU Command Acceptance.) The transition from ROM P/E mode to ROM read mode should only be executed in the state where FCU command processing has completed and the FCU has not detected any errors.



Notes: t_{E128K} : Time required to erase a 128-kbyte block (see section 38, Electrical Characteristics).
t_{RESW2}: Reset pulse width during programming and erasure (see section 38, Electrical Characteristics).

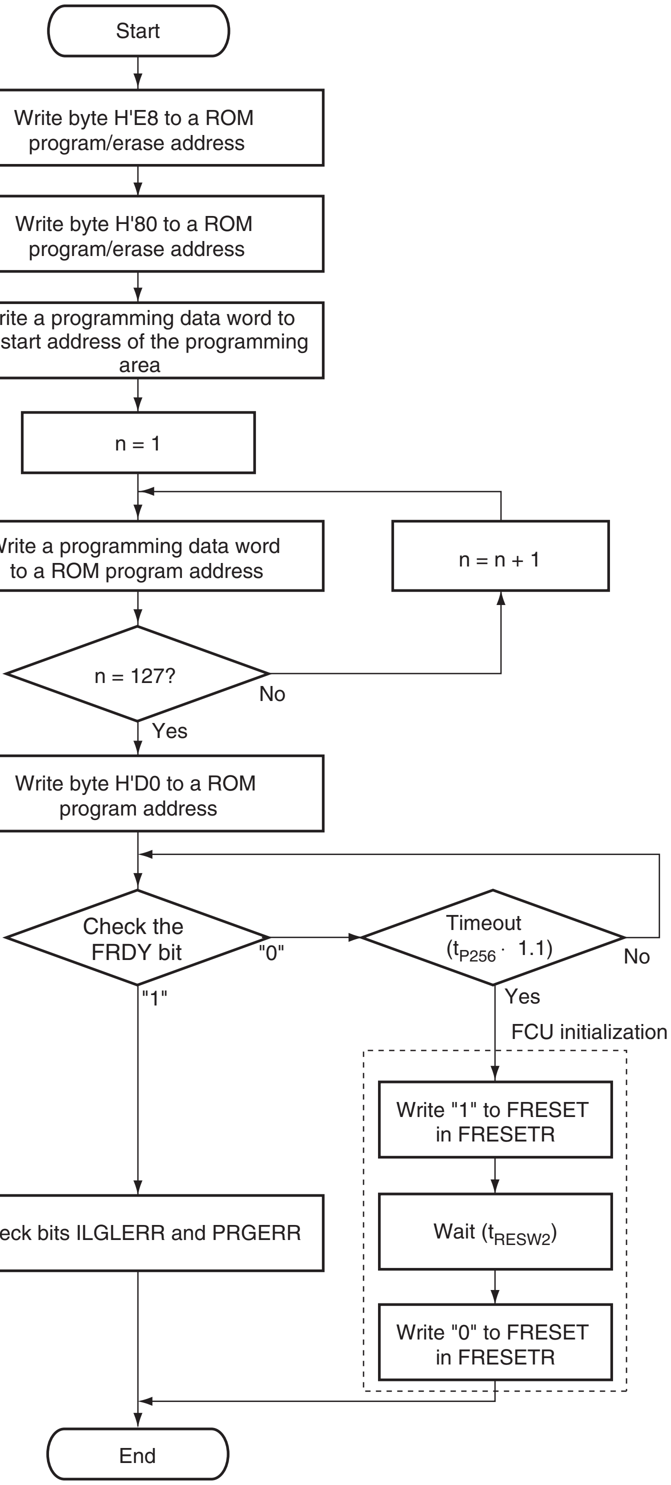
Figure 12.9 Procedure for Transition to ROM Read Mode

(3) Programming

To program the ROM, use the program command. Write byte H'E8 to a ROM program/erase address in the first cycle of the program command and byte H'80 in the second cycle. Access the peripheral bus in words from the third to 130th cycles of the command. In the third cycle, write the programming data to the start address of the target programming area. Here, the start address must be a 256-byte boundary address. After writing words to ROM program/erase addresses 127 times, write byte H'D0 to a ROM program/erase address in the 131st cycle; the FCU then starts ROM programming. Read the FRDY bit in the FSTATR0 register to confirm that ROM programming is completed.

The addresses that can be specified in the first to 131st cycles depend on the setting of the FENTRY0 bit in the FENTRYR register. An address in the range from H'FD80 0000 to H'FD8F FFFF is can be specified when the FENTRY0 bit is set to "1". If a command is issued while an illegal combination of the FENTRY0 bit value and addresses is specified, the FCU detects an error and enters command-locked state (see section 12.8.3, Error Protection).

If the area accessed in the third to 130th cycles includes addresses that do not need to be written, use H'FFFF as the data to be written to those addresses. To disable the lock bit protection during writing, set the FPROTCN bit in the FPROTR register before performing the write operation.


Start
Write byte H'E8 to a ROM program/erase address
Write byte H'80 to a ROM program/erase address
Write a programming data word to the start address of the programming area
n = 1
Write a programming data word to a ROM program address
n = n + 1
n = 127?
No
Yes
Write byte H'D0 to a ROM program address
Check the FRDY bit
"0"
"1"
Timeout (tP256 · 1.1)
No
Yes
FCU initialization
Write "1" to FRESET in FRESETR
Wait (tRESW2)
Write "0" to FRESET in FRESETR
Check bits ILGLERR and PRGERR
End

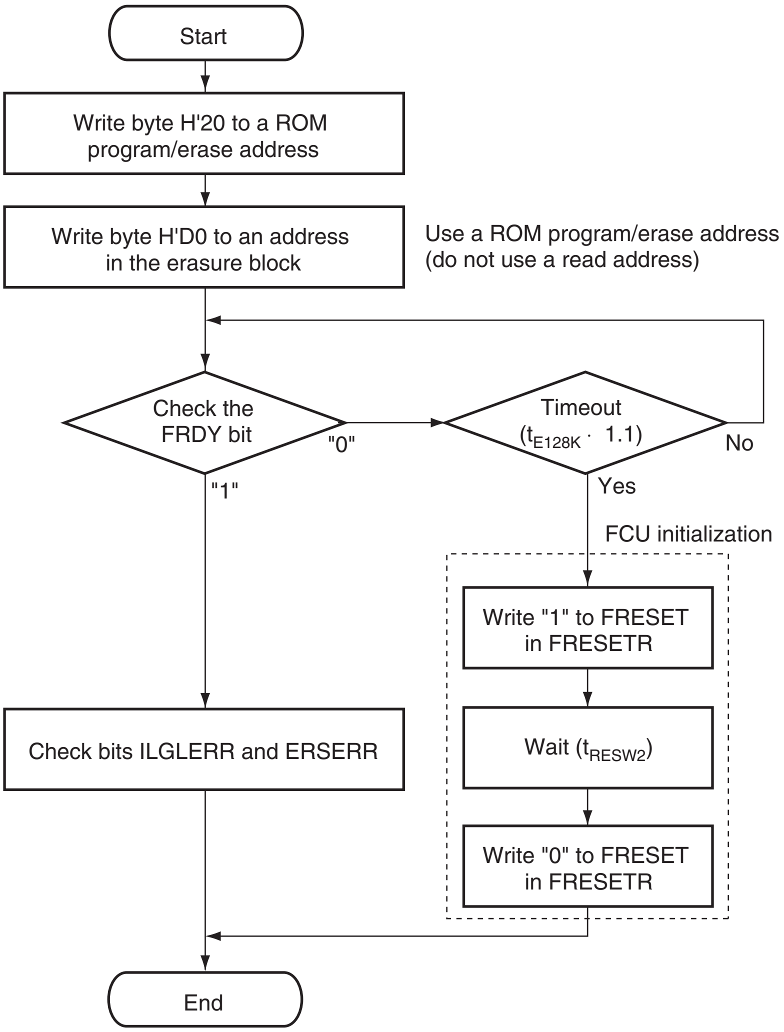
Notes: t_{P256}: Time required to write 256-byte data (see section 38, Electrical Characteristics).
t_{RESW2}: Reset pulse width during programming and erasure (see section 38, Electrical Characteristics).

Figure 12.10 Procedure for ROM Programming

(4) Erasure

To erase the ROM, use the block erase command. Write byte H'20 to a ROM program/erase address in the first cycle of the block erase command. Write byte H'D0 to an address in the target erasure block in the second cycle; the FCU then starts ROM erasure. Read the FRDY bit in the FSTATR0 register to confirm that ROM erasure is completed.

To ignore the protection provided by the lock bit during erasure, set the FPROTCN bit in the FPROTR register to 1 before starting erasure.


Start
Write byte H'20 to a ROM program/erase address
Write byte H'D0 to an address in the erasure block
Use a ROM program/erase address (do not use a read address)
Check the FRDY bit
"0"
"1"
Timeout (t_{E128K} · 1.1)
No
Yes
FCU initialization
Write "1" to FRESET in FRESETR
Wait (t_{RESW2})
Write "0" to FRESET in FRESETR
Check bits ILGLERR and ERSERR
End

Notes: t_{E128K} : Time required to erase a 128-kbyte block (see section 38, Electrical Characteristics).
t_{RESW2}: Reset pulse width during programming and erasure (see section 38, Electrical Characteristics).

Figure 12.11 Procedure for ROM Erasure

(5) Clearing Status Register 0 (FSTATR0)

To clear bits ILGLERR, PRGERR, and ERSERR in the FSTATR0 register, use the status register clear command. When any one of bits ILGLERR, PRGER, and ERSERR is "1", the FCU is in command-locked state, in which the FCU only accepts the status register clear command and does not accept other commands. When the ILGLERR bit is "1", check also the value of the ROMAE bit in the FASTAT register. If a status register clear command is issued without clearing these bits, the ILGLERR bit is not cleared.

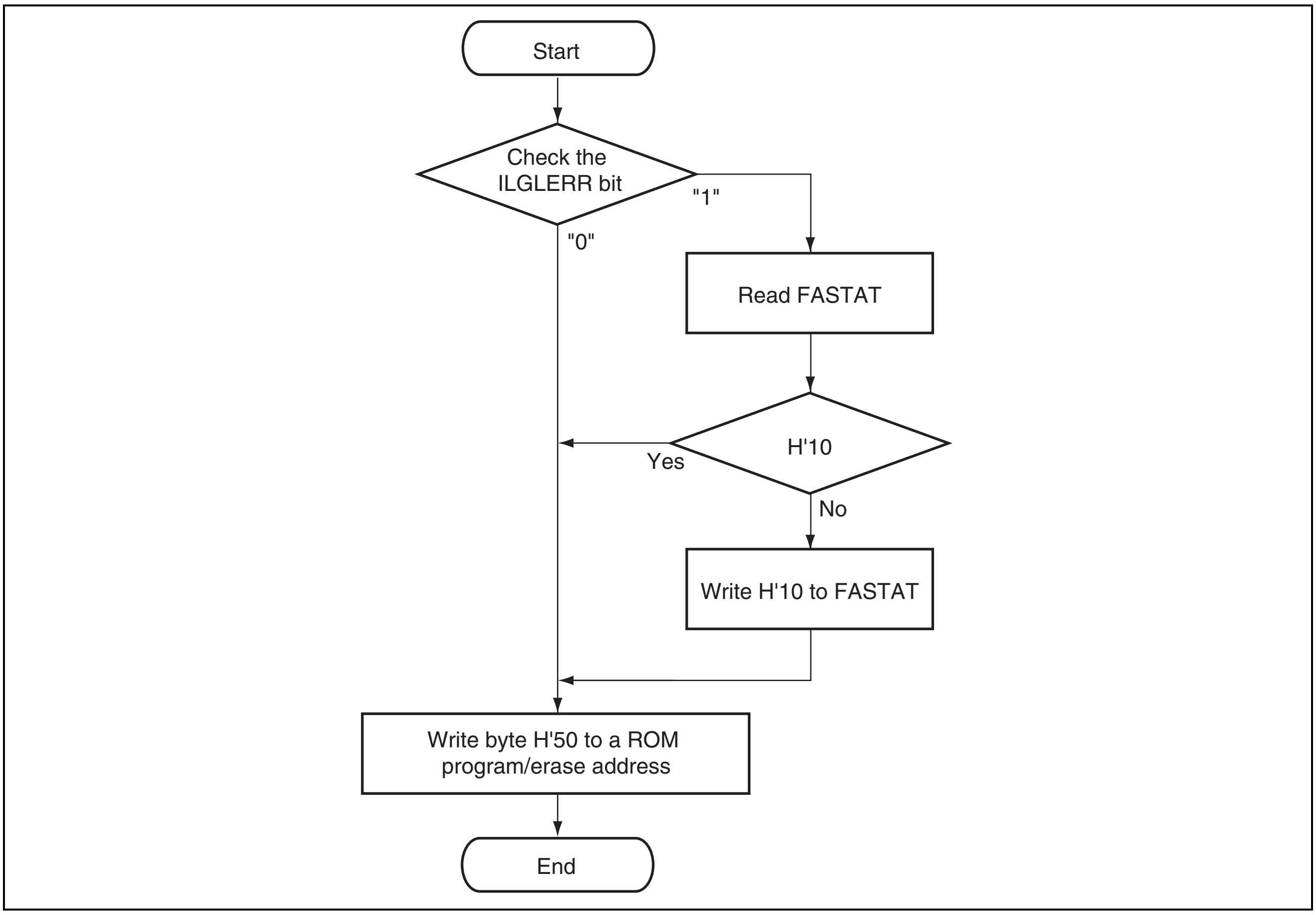
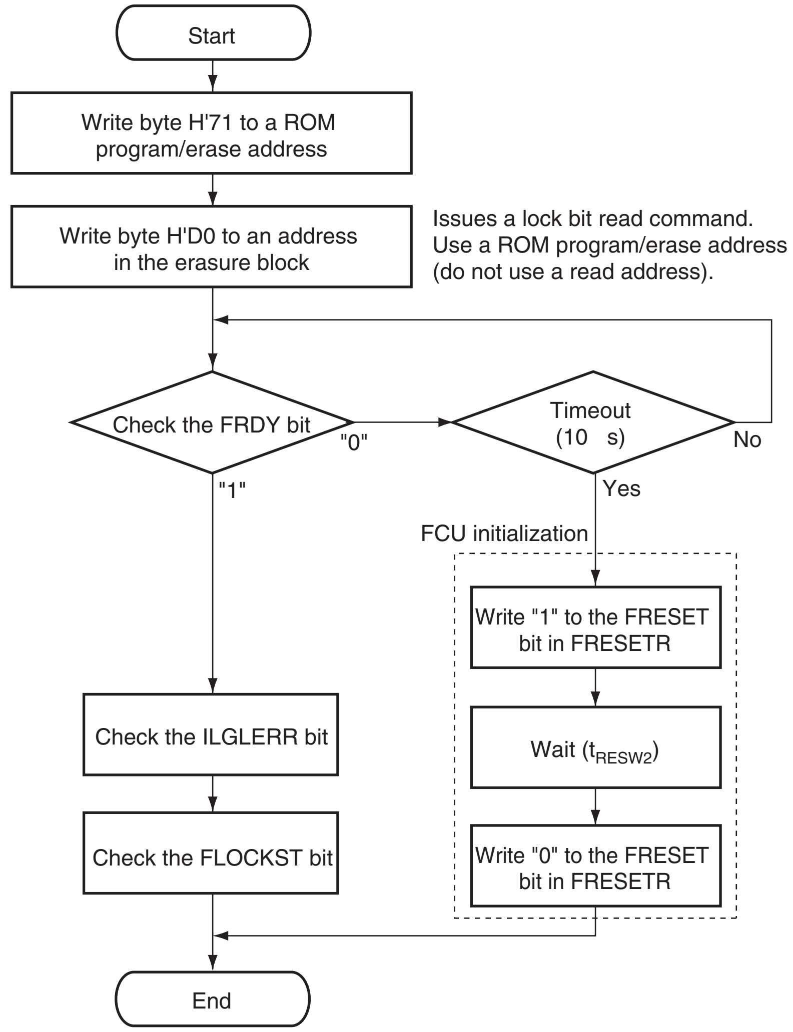


Figure 12.12 Procedure for Clearing Status Register 0

(6) Reading Lock Bit

Each erasure block in the user MAT has a lock bit. While the FPROTCN bit in the FPROTR register is "0", the erasure block whose lock bit is set to "0" cannot be programmed or erased.

To check the lock bit status, issue a lock bit read command. The specified erase block lock bit will be indicated by the FLOCKST bit in the FSTATR1 register.

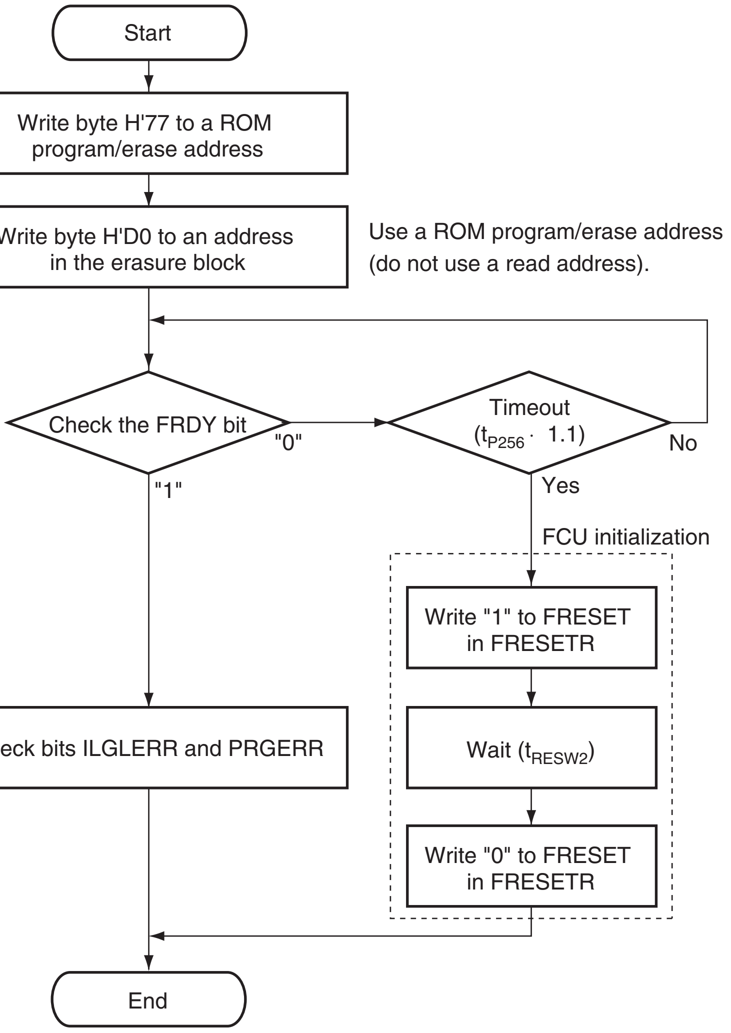

Start
Write byte H'71 to a ROM program/erase address
Write byte H'D0 to an address in the erasure block
Issues a lock bit read command.
Use a ROM program/erase address
(do not use a read address).
Check the FRDY bit
"0"
"1"
Timeout
(10   s)
No
Yes
FCU initialization
Write "1" to the FRESET bit in FRESETR
Wait (t_{RESW2})
Write "0" to the FRESET bit in FRESETR
Check the ILGLERR bit
Check the FLOCKST bit
End

Note: t_{RESW2} : Reset pulse width during programming and erasure (see section 38, Electrical Characteristics).

Figure 12.13 Procedure for Reading Lock Bit in Register Read Mode

(7) Writing to Lock Bit

Each erasure block in the user MAT has a lock bit. To write to a lock bit, use the lock bit program command. Write byte H'77 to a ROM program/erase address in the first cycle of the lock bit program command. Write byte H'D0 to an address in the target erasure block whose lock bit is to be written to in the second cycle; the FCU then starts writing to the lock bit. Read the FRDY bit in the FSTATR0 register to confirm that writing is completed.


Start
Write byte H'77 to a ROM program/erase address
Write byte H'D0 to an address in the erasure block
Use a ROM program/erase address (do not use a read address).
Check the FRDY bit
"0"
"1"
Timeout (t_{P256} · 1.1)
No
Yes
FCU initialization
Write "1" to FRESET in FRESETR
Wait (t_{RESW2})
Write "0" to FRESET in FRESETR
Check bits ILGLERR and PRGERR
End

Notes: t_{P256}: Time required to write 256-byte data (see section 38, Electrical Characteristics).
t_{RESW2}: Reset pulse width during programming and erasure (see section 38, Electrical Characteristics).

Figure 12.14 Procedure for Writing to the Lock Bit

To erase a lock bit, use the block erase command. While the FPROTCN bit in the FPROTR register is "0", the erasure block whose lock bit is set to "0" cannot be erased. Set the FPROTCN bit to "1", and then issue a block erase command to erase a lock bit. The block erase command erases all data in the specified erasure block; it is not possible to erase only the lock bit.

12.7 User Boot Mode

To program or erase the user MAT in user boot mode, issue FCU commands to the FCU. A custom boot environment can be implemented by writing a routine for programming/erasing ROM using the desired communications interface and then starting in user boot mode. Note that the user boot MAT must be written in boot mode.

12.7.1 Switching between User MAT and User Boot MAT

Although this MCU starts up from the user boot MAT in user boot mode, since the user MAT is also allocated to the same address area, it is necessary to switch from the user boot MAT to the user MAT to write the user MAT. Note that it will not be possible to access the user boot MAT after switching to the user MAT. Furthermore, if the cache function is enabled, since there will still be pre-switching data stored in the cache after switching MATs, it is possible that a cache hit will occur when accessing different MATs with the same address. The following processing is necessary to avoid these problems.

(1) Mask all interrupts

To prevent accesses to the ROM area due to interrupts after MAT switching, mask all interrupts. Since it is not possible to mask NMI interrupts, configure the system so that NMI interrupts will not occur during MAT switching.

(2) Copy the ROM writing program to internal RAM

Copy the programs that perform MAT switching, acquisition of the data to be written, the ROM programming itself, and other functions.

(3) Jump to RAM

After all the programs have been copied to RAM, jump to the program in RAM.

(4) Set the ROMMAT register

Execute the MAT switching processing in the internal RAM area so that CPU instruction fetches to the ROM area do not occur during MAT switching.

(5) Read the ROMMAT register

Perform a dummy read of the ROMMAT register after writing to the ROMMAT register to switch MATs. This is necessary to confirm that the register value changed.

(6) Invalidate the cache

Invalidate all cache lines after switching MATs. (See section 8, Caches.)

12.7.2 Programming the User MAT

After switching MATs, run a ROM write program created according to section 12.6, User Mode and User Boot Mode. This will acquire the write data sequentially using the desired communication protocol.

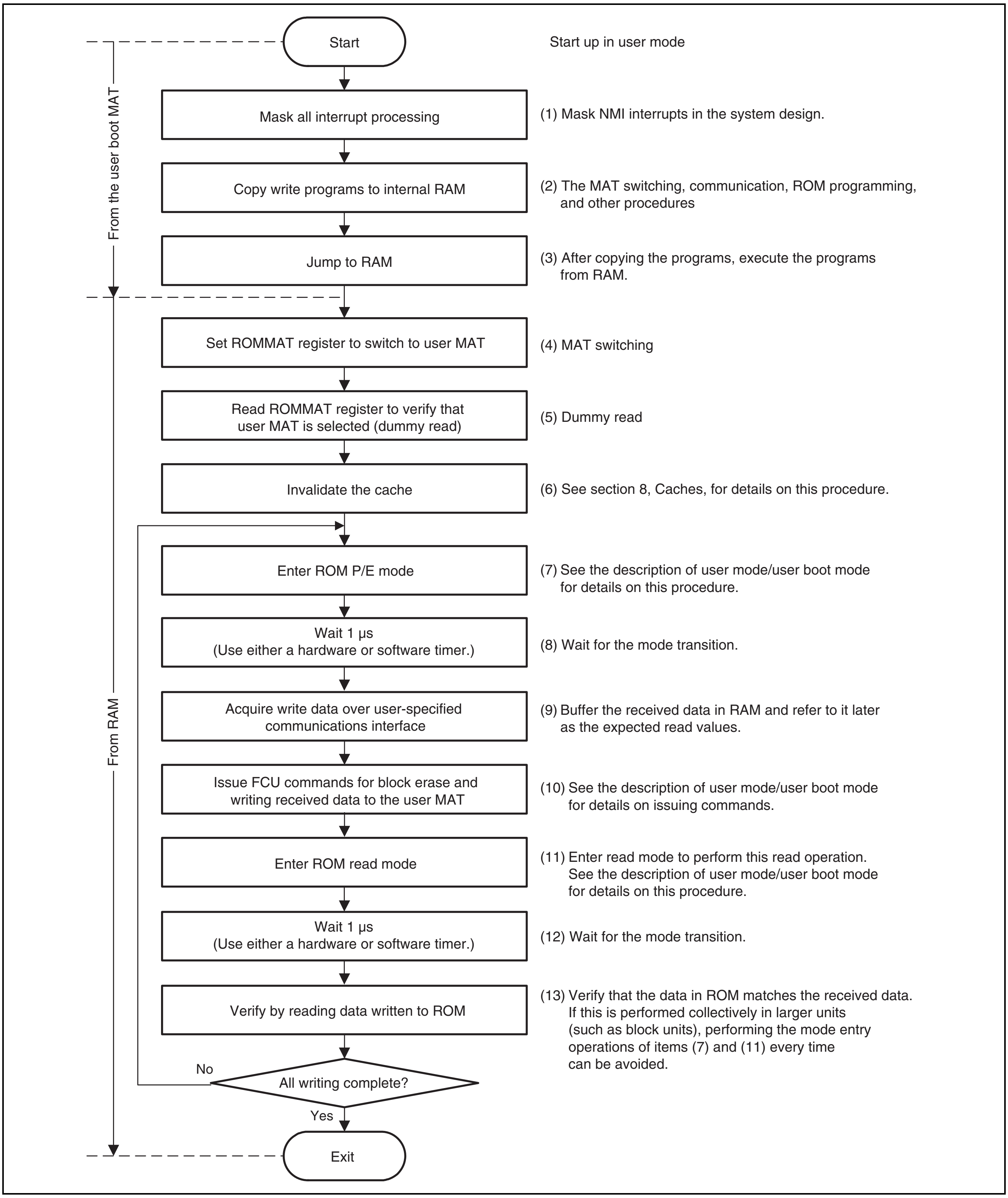

Start
Start up in user mode
From the user boot MAT
Mask all interrupt processing
(1) Mask NMI interrupts in the system design.
Copy write programs to internal RAM
(2) The MAT switching, communication, ROM programming, and other procedures
Jump to RAM
(3) After copying the programs, execute the programs from RAM.
From RAM
Set ROMMAT register to switch to user MAT
(4) MAT switching
Read ROMMAT register to verify that user MAT is selected (dummy read)
(5) Dummy read
Invalidate the cache
(6) See section 8, Caches, for details on this procedure.
Enter ROM P/E mode
(7) See the description of user mode/user boot mode for details on this procedure.
Wait 1 μs (Use either a hardware or software timer.)
(8) Wait for the mode transition.
Acquire write data over user-specified communications interface
(9) Buffer the received data in RAM and refer to it later as the expected read values.
Issue FCU commands for block erase and writing received data to the user MAT
(10) See the description of user mode/user boot mode for details on issuing commands.
Enter ROM read mode
(11) Enter read mode to perform this read operation. See the description of user mode/user boot mode for details on this procedure.
Wait 1 μs (Use either a hardware or software timer.)
(12) Wait for the mode transition.
Verify by reading data written to ROM
(13) Verify that the data in ROM matches the received data. If this is performed collectively in larger units (such as block units), performing the mode entry operations of items (7) and (11) every time can be avoided.
All writing complete?
No
Yes
Exit

Figure 12.15 User MAT Programming Example

12.8 Protection

There are three types of ROM programming/erasure protection: hardware, software, and error protection.

12.8.1 Hardware Protection

The hardware protection function disables ROM programming and erasure according to the MCU pin settings.

(1) Protection through FWE Pin

When an "L" level signal is being input on the FWE pin, the FWE bit in the FPMON register becomes "0". In this state, "1" cannot be written to the FENTRY0 bit in the FENTRYR register; that is, ROM P/E mode cannot be entered, which prevents the ROM from being programmed or erased.

When the FRDY bit is set to "1" and the FWE pin is "L" level, the FCU clears the FENTRY0 bit to disable ROM programming and erasure.

If an attempt is made to issue a programming or erasing command to the ROM against the protection through the FWE pin, the FCU detects an error and enters command-locked state.

(2) Protection through Mode Pins

For the operating modes set through the mode pins of this MCU, refer to section 10, Operating Modes. In user boot mode or user mode, the user boot MAT cannot be programmed or erased.

12.8.2 Software Protection

The software protection function disables ROM programming and erasure according to the control register settings or the lock bit settings in the user MAT. If an attempt is made to issue a programming or erasing command to the ROM against software protection, the FCU detects an error and enters command-locked state.

(1) FENTRYR protection

When the FENTRY0 bit is "0", the EB00 to EB19 blocks of ROM (read addresses: H'0000 0000 to H'000F FFFF, program/erase addresses: H'FD80 0000 to H'FD8F FFFF) goes to ROM read mode. Since FCU commands are not accepted in ROM read mode, ROM goes to the program/erase disabled mode. If an FCU command is issued in ROM read mode, the FCU will detect an illegal command error and go to the command-locked state. (See section 12.8.3, Error Protection.)

(2) Protection through Lock Bits

Each erasure block in the user MAT has a lock bit. When the FPROTCN bit in the FPROTR register is "0", the erasure block whose lock bit is set to "0" cannot be programmed or erased. To program or erase the erasure block whose lock bit is "0", set the FPROTCN bit to "1". If an attempt is made to issue a programming or erasing command against protection by lock bits, the FCU detects an programming/erasure error and enters command-locked state (see section 12.8.3, Error Protection).

12.8.3 Error Protection

Error protection is a state (the command-locked state) in which an FCU command error, an illegal access, or incorrect FCU operation was detected and FCU command acceptance is disabled. Setting the FCU to the command-locked state disables ROM program/erase operations. To clear the command-locked state, a status register clear command must be issued in the state where the FASTAT register is H'10.

Table 12.7 shows the relationship between the ROM related error protection types and the post-error detection values of the status bits (bits ILGLERR, ERSERR, and PRGERR in the FSTATR0 register, bits FCUERR and FRDTCT in the FSTATR1 register, and the ROMAE bit in the FASTST register).

Table 12.7 Error Protection Types

Error	Description	ILGLERR bit	ERSERR bit	PRGERR bit	FCUERR bit	FRDTCT bit	ROMAE bit
FENTRYR setting error	The key code (H'AA) has been supplied as the upper 8 bits of the FENTRYR register but the value of the lower 8 bits is other than H'01 or H'02.	1	0	0	0	0	0
Illegal command error	An undefined code has been specified in the first cycle of an FCU command.	1	0	0	0	0	0
	The value specified in the last of the multiple cycles of an FCU command is not H'D0.	1	0	0	0	0	0
	The value specified in the second cycle of a program command is not H'80.	1	0	0	0	0	0
	A command has been issued in command-locked state.	1	0/1	0/1	0/1	0/1	0/1
Erasure error	An error has occurred during erasure processing.	0	1	0	0	0	0
	A block erase command has been issued for the erasure block whose lock bit is set to "0" while the FPROTCN bit in the FPROTR register is "0".	0	1	0	0	0	0
Programming error	An error has occurred during programming processing.	0	0	1	0	0	0
	A program or lock bit program command has been issued for the erasure block whose lock bit is set to "0" while the FPROTCN bit in the FPROTR register is "0".	0	0	1	0	0	0
FCU error	An error has occurred during CPU processing in the FCU.	0	0	0	1	0	0
ROM access error	A read access command has been issued to addresses H'FD80 0000 to H'FD8F FFFF while FENTRY0 = "1" in ROM P/E normal mode.	1	0	0	0	0	1
	An access command has been issued to addresses H'FD90 0000 to H'FD9F FFFF while FENTRY1 = "0".	1	0	0	0	0	1
	An access command has been issued to addresses H'FD80 0000 to H'FD8F FFFF while FENTRY0 = "0".	1	0	0	0	0	1
	A read access command has been issued to addresses H'0000 0000 to H'000F FFFF while the FENTRYR register value is not H'0000.	1	0	0	0	0	1
	A ROM programming or erasing command (program, lock bit program, or block erase command) has been issued while the user boot MAT is selected.	1	0	0	0	0	1
	An access command has been issued to an address other than the addresses for ROM programming/erasure H'FD80 0000 to H'FD80 7FFF while the user boot MAT is selected.	1	0	0	0	0	1

12.9 Usage Notes

12.9.1 Key Code Stored Area

Addresses H'0000 00F0 to H'0000 00F7 in the user MAT store the key code for debugging function authentication to be used with the on-chip debugger. To restrict the debugging functions, write a key code in this area. After a key code is specified through the debugger, the code is stored in this area, which should be noted during checksum verification.

Note that key codes beginning with a byte B9 are prohibited.

12.9.2 Compatibility with Programming/Erasing Program of Conventional F-ZTAT SH Microcomputers (MCUs)

The flash memory programming/erasing program used for conventional F-ZTAT SH MCUs does not work with this MCU.

12.9.3 FWE Pin State

To program or erase ROM by driving the FWE pin to "H" level after powering on the chip with the FWE pin at "L" level, ensure that the FWE pin value does not change during reprogramming. If the FWE pin value changes during reprogramming, the FCU will detect a protection violation, and enters command-locked state after forcibly stopping the reprogram operation. As stopping reprogramming forcibly can cause a malfunction, ensure that the FWE pin does not change during reprogramming.

12.9.4 Reset during Programming or Erasure

To reset the FCU by setting the FRESET bit in the FRESETR register during programming or erasure, hold the FCU in the reset state for a period of t_{RESW2} (see section 38, Electrical Characteristics). Since a high voltage is applied to the ROM during programming and erasure, the FCU has to be held in the reset state long enough to ensure that the voltage applied to the memory unit has dropped. Do not read from the ROM while the FCU is in the reset state.

When a hardware reset is triggered by inputting an "L" level signal to the RESET# pin during programming or erasure of the flash memory, hold the reset state for a period of t_{RESW} (see section 38, Electrical Characteristics). In a hardware reset, not only does the voltage applied to the memory unit have to drop, but the power supply for the ROM and its internal circuitry also have to be initialized. Thus, the reset state must be maintained over a longer period than in the case of resetting the FCU.

While programming or erasure is performed, do not generate an internal reset caused by WDT counter overflow. A reset caused by WDT cannot ensure a sufficient time required for voltage drop for the memory unit, initialization of the power supply for the ROM, or initialization of its internal circuit.

12.9.5 Prohibition of Additional Programming

One area cannot be programmed twice in succession. To program an area that has already been programmed, be sure to erase the area before reprogramming.

12.9.6 Power Supply Control during Reprogramming

Ensure that the supply voltage is provided stably during ROM reprogramming. If the supply voltage is provided unstably, ROM may not be rewritten correctly or data may not be read correctly. An instantaneous interruption in the supply voltage during reprogramming will add an unintended stress to the MCU, causing malfunctions. Provide the supply voltage carefully.

12.9.7 Accessing ROM-Related Registers

It is not necessary to make settings to the ROM-related registers after a reset is canceled for normal program execution (reading data from ROM). When programming or erasing ROM, however, it is necessary to access the ROM-related registers. Since programming or erasing of ROM must be performed by a program executed from outside the ROM area, write to the ROM-related registers by means of a program executed from the SHwyRAM or IL memory.

Section 13 SuperHyway RAM (SHwyRAM)

13.1 Overview

The SuperHyway RAM (SHwyRAM) is connected to the SuperHyway bus and to the DRI/DRO dedicated bus, and its different areas are 64-Kbytes units (pages 0 to 3) that can be accessed independently from both. Figure 13.1 is a block diagram of the SHwyRAM module.

As shown in figure 13.2, the SHwyRAM is allocated to the upper 256 Kbytes of area 6 (H'1800 0000 to H'1803 FFFF in the 29-bit physical address space).

Note that the SHwyRAM is located in an area that can be cached and to which address translation can be applied (MMU), so it can be accessed from the P0/U0, P1, P2, and P3 areas.

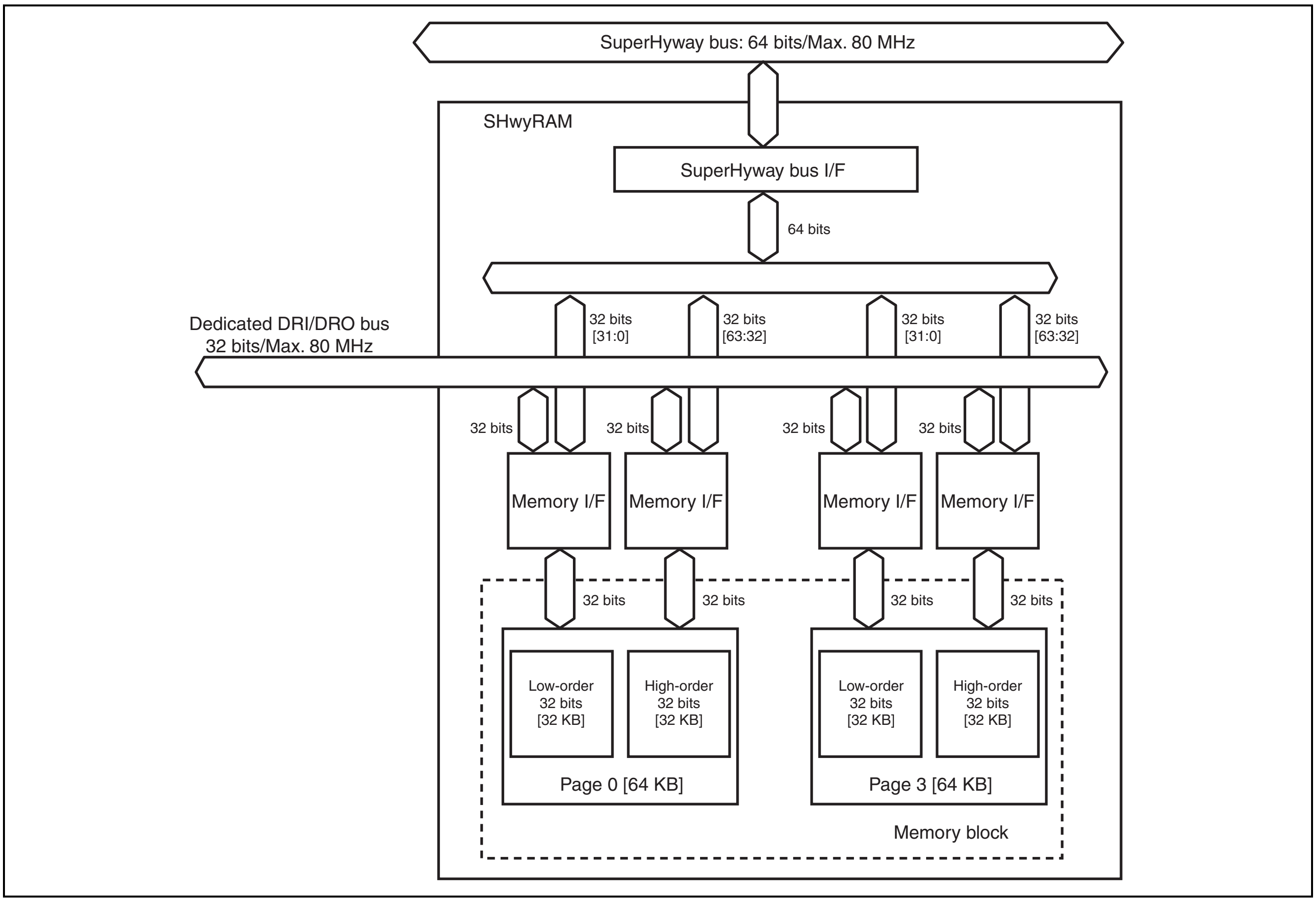
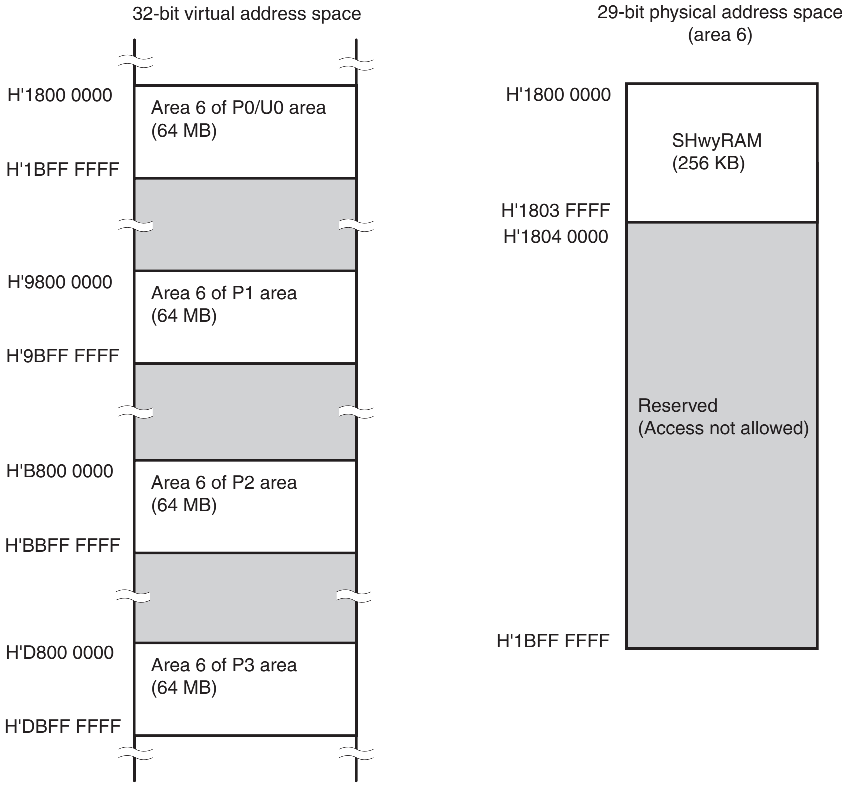


Figure 13.1 Block Diagram of SHwyRAM



Page	Address (29-bit physical address)
Page 0	H'1800 0000 to H'1800 FFFF
Page 1	H'1801 0000 to H'1801 FFFF
Page 2	H'1802 0000 to H'1802 FFFF
Page 3	H'1803 0000 to H'1803 FFFF

Figure 13.2 Address Space

(1) Access

The CPU and DMAC can access SHwyRAM in 8, 16, 32 or 64-bit units. The H-UDI and AUDR can access SHwyRAM in 8, 16 or 32-bit units. The DRI/DRO can access SHwyRAM in 32-bit units.

SHwyRAM is appropriate for program areas that require high-speed access as well as for stack and data areas.

(2) Ports

Each page has two read/write ports that are connected to the SuperHyway bus and the DRI/DRO dedicated bus.

(3) Priority

If access requests to the same page occur at the same time from different busses, the accesses are processed in priority order. The SuperHyway bus has higher priority than the DRI/DRO dedicated bus.

This page is blank for reasons of layout.

Section 14 Clock Generator (CPG)

14.1 Overview

The clock generator module (CPG) supplies clock pulses to internal and external devices in this MCU. The CPG module consists of an oscillator circuit and a PLL frequency multiplier circuit. The CPG module can be used to generate clock signals in one of two ways: with a crystal resonator connected or with an external clock input.

The oscillator circuit oscillates at the same frequency as the input clock.

The CPG module supplies the following five clock signals to this MCU internal circuits: the CPU clock (Ick), the SHwy clock (SHck), the peripheral clock (Pck), the peripheral A clock (PAck), and the FlexRay clock (FRck).

The CPU clock (Ick) is supplied to the CPU, the FPU, the cache, and other modules. The SHwy clock (SHck) is supplied to the SHwyRAM, ROM, and other modules. The CPU clock (Ick) frequency is eight times the frequency input to the EXTAL pin.

Table 14.1 lists the relation between input frequency and input clock.

Table 14.1 Relation between Input Frequency and Input Clock

Input frequency (MHz)	PLL frequency multiplier (input to CPU)	CPU clock (MHz)	SHwy clock (MHz)	Peripheral clock (MHz)	Peripheral A clock (MHz)	FlexRay clock (MHz)
20	×8	160	80	40	80	80

The peripheral clock (Pck) is supplied mainly to the internal peripheral modules, and has a frequency 2 times the frequency input to the EXTAL pin. The peripheral clock (Pck) is also output as the system clock from the CLKOUT pin. The peripheral A clock (PAck) is supplied to the direct RAM input interface (DRI). The FlexRay clock (FRck) is supplied to the FlexRay module.

Figure 14.1 shows the CPG module block diagram.

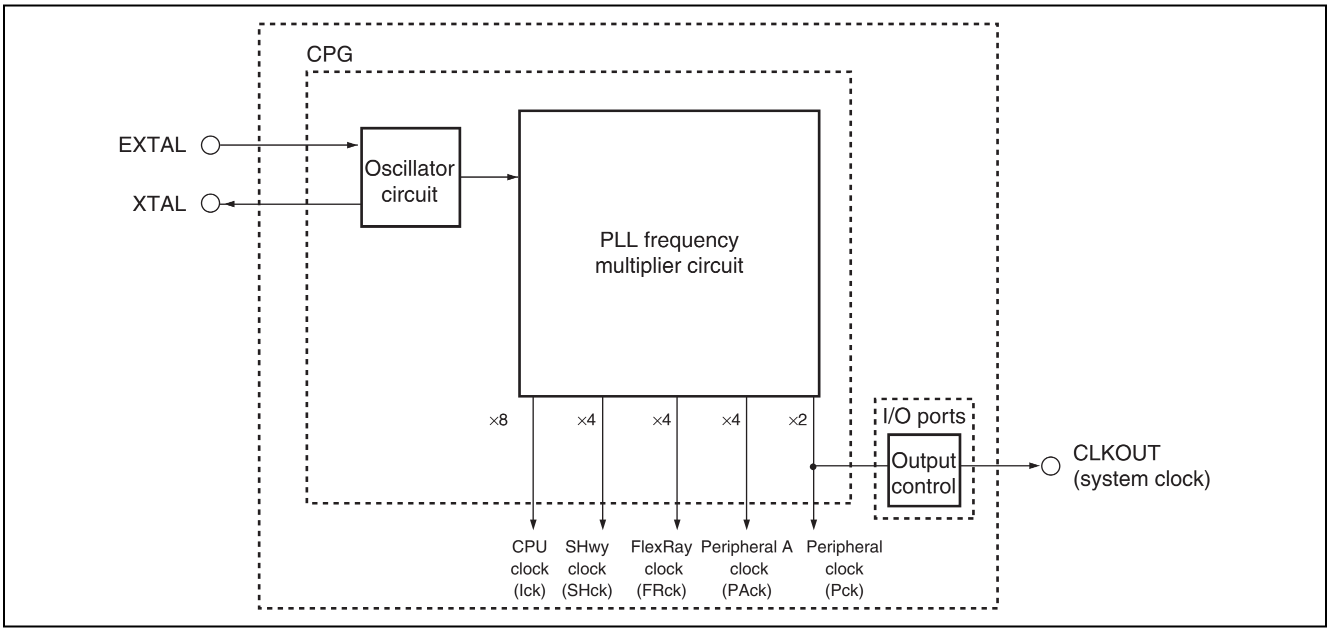


Figure 14.1 Block Diagram of CPG

14.2 Input/Output Pins

Table 14.2 lists the CPG module related pins.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 14.2 Pin Configuration

Pin Name	I/O	Function
EXTAL	Input	Crystal resonator or external clock input
XTAL	Output	Crystal resonator connection
CLKOUT	Output	System clock output
PLLVcc	Input	PLL frequency multiplier power supply
PLLVss	Input	PLL frequency multiplier ground

14.3 Register Descriptions

Table 14.3 lists the CPG module registers.

Table 14.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Oscillator status register	OSCSR	H'00	H'FFFF 2810	8	14-3
Oscillator control register	OSCCR	H'00	H'FFFF 2814	8	14-3

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

14.3.1 Oscillator Status Register (OSCSR)

The OSCSR register is a read-only register that holds the oscillator stop detection flag.

Oscillator status register (OSCSR) <P4 address: location H'FFFF 2810>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSC STOP
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	N	Reserved Bits These bits are always read as "0".
0	OSCSTOP	0	R	N	Oscillator Stop Detection Flag OSCSTOP is a read-only bit and cannot be written. Once the OSCSTOP bit is set to "1" it retains its value thereafter. It can only be cleared to "0" by a hardware reset. 0: The oscillator is in the normal operating state 1: Oscillator stop detected/internal oscillator circuit clock supply*[1] [Clearing condition] • Hardware reset [Setting condition] • When the oscillator circuit output is stuck in the "1" or "0" state when INOSCE = "1".

Note: *1 When EXTAL input is stopped, the frequency unique to the PLL circuit is used for oscillation.

14.3.2 Oscillator Control Register (OSCCR)

The OSCCR register controls the enabled/disabled state of the oscillator stop detection function.

Oscillator control register (OSCCR) <P4 address: location H'FFFF 2814>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	INOSCE
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	INOSCE	0	R	W	Oscillator Stop Detection Function Enable Bit 0: Disables the oscillator stop detection function 1: Enables the oscillator stop detection function

14.4 Clock Sources

Applications can select either a crystal resonator or an external clock as the clock source.

14.4.1 Crystal Resonator Connection

(1) Circuit structure

Figure 14.2 shows the method for connecting a crystal resonator. For the crystal resonator, use an AT-cut fundamental frequency crystal resonator. Note that the load capacitors (CL1 and CL2) shown in figure 14.2 must be used.

The clock pulse signal generated by the crystal resonator and the internal oscillator is multiplied by the PLL frequency multiplier circuit, and supplied to this MCU internal and external devices.

Consult with the crystal resonator's manufacturer regarding its compatibility with this MCU.

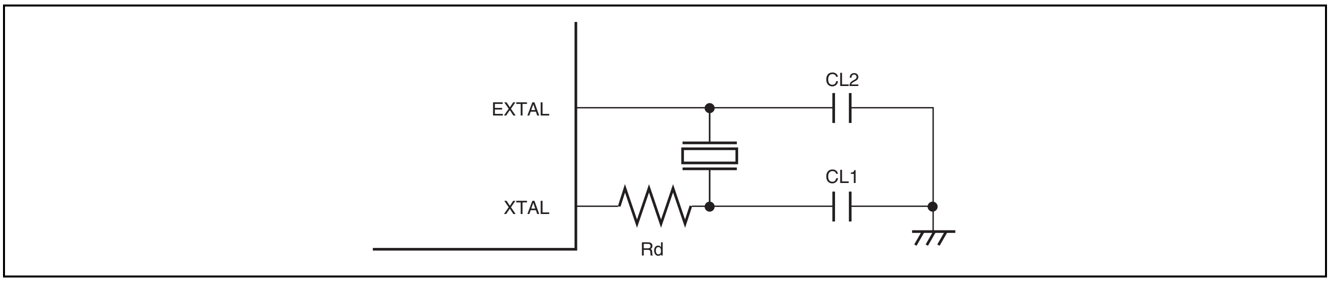


Figure 14.2 Crystal Resonator Connection Example

(2) Crystal resonator

Figure 14.3 shows the equivalent circuit for the crystal resonator.

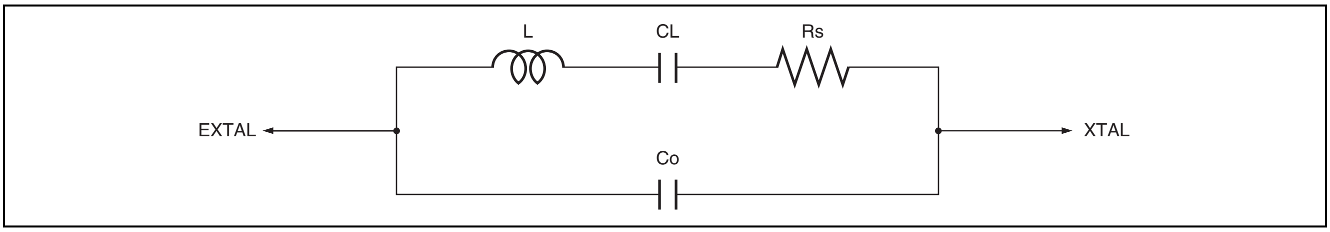


Figure 14.3 Crystal Resonator Equivalent Circuit

14.4.2 External Clock Input

Figure 14.4 shows an external clock input connection example.

Even when providing an external clock input, applications must wait the oscillator stabilization time after power is first applied to assure that the PLL circuit has time to stabilize.

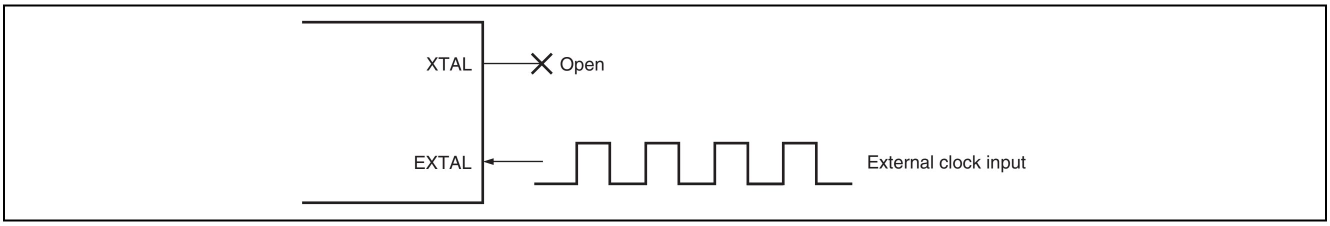


Figure 14.4 External Clock Input

14.5 Usage Notes

14.5.1 Board Design Notes

Locate the crystal resonator and the load capacitors as close as possible to the EXTAL and XTAL pins. To assure that the circuit is not influenced by noise and operates correctly, do not allow the EXTAL pin and XTAL pin lines to cross any other signal lines.

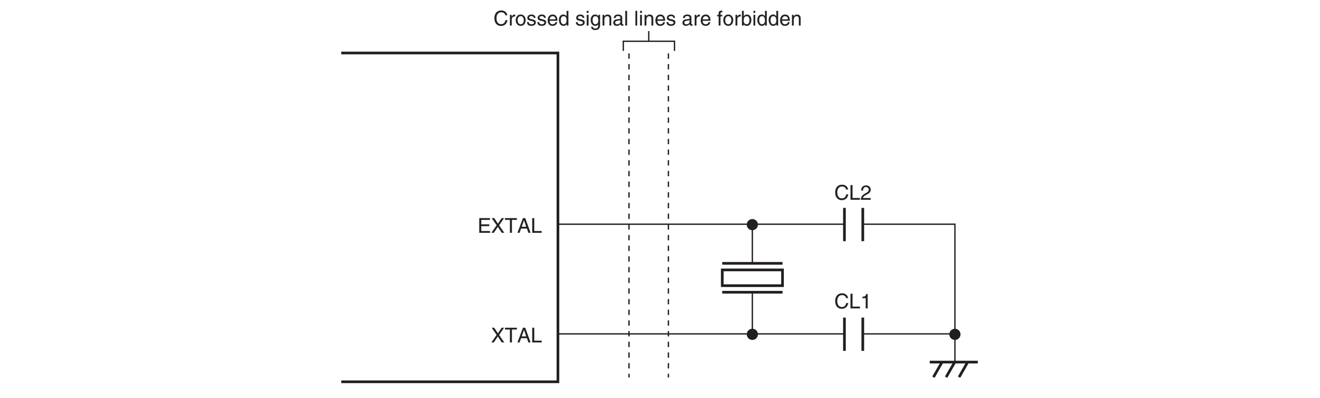


Figure 14.5 Board Design Notes

14.5.2 PLL Frequency Multiplier Circuit Power Supply Connection Notes

The PLLVcc and PLLVss pins must be isolated from other Vcc and Vss pins from the board power supply source. Also, the bypass capacitors C_{PB} and C_B must be inserted as close as possible to the pins.

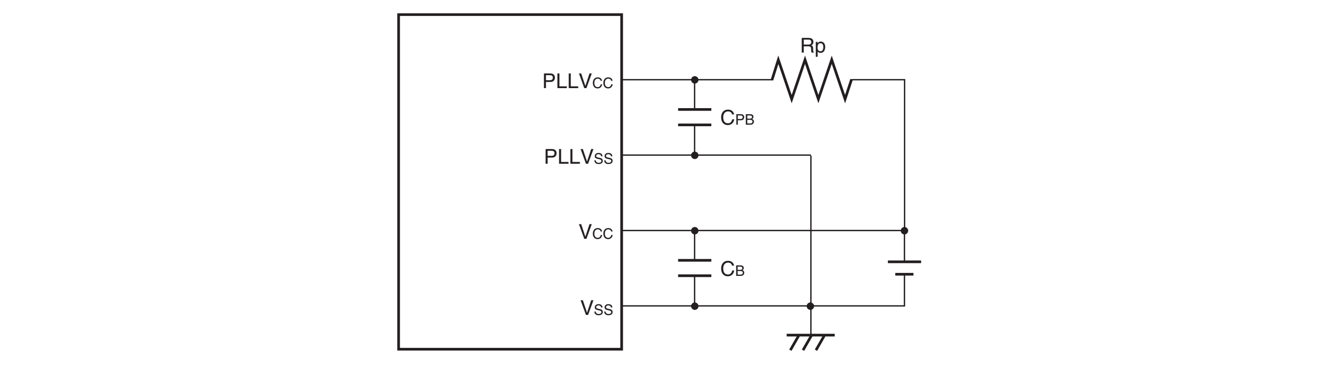


Figure 14.6 PLL Frequency Multiplier Circuit Power Supply Connection Notes

This page is blank for reasons of layout.

Section 15 Interrupt Controller (INTC)

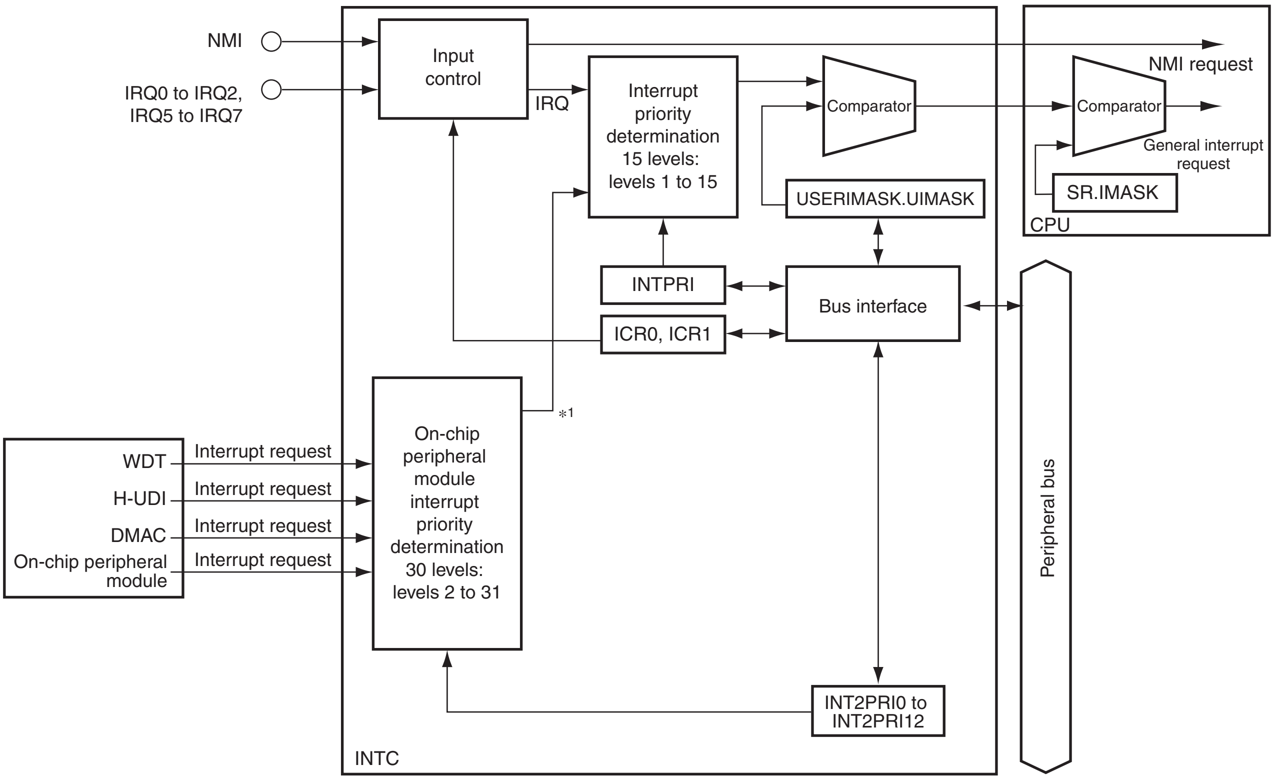
15.1 Overview

The interrupt controller (INTC) determines the priority of interrupt sources and controls interrupt requests to the CPU. The INTC has a register that sets the priority of each interrupt and interrupt requests are processed according to the priority set in this register by the user. Table 15.1 lists the overview of the INTC.

Table 15.1 Overview of INTC

Item	Description
Interrupt priority	• IRQ interrupt (IRQ0 to IRQ2 and IRQ5 to IRQ7): 15 levels • On-chip peripheral module interrupt: 30 levels
NMI request hold function	• Whether to hold NMI requests can be selected when the BL bit in the SR register is set to "1".
NMI pin input-level monitor function	• An NMI level bit indicates the the NMI pin state. By reading this bit in the interrupt exception handling routine, the pin state can be checked, enabling it to be used as a noise canceller.
NMI detection	• Rising or falling edge can be selected.
IRQn detection	• "H" or "L" level and rising or falling edge can be selected.
IMASK update selection function	• Automatically updates the IMASK bits in the SR register according to the accepted interrupt level

Figure 15.1 shows a block diagram of the INTC.



Note: *1 For notification of the priority of on-chip peripheral module interrupts, the lowest bit is discarded and the value is converted to 4 bits.

- The following modules csn issue on-chip peripheral module interrupts: TMU, ATU-IIIS, SCIF, RSPI, IIC, DRI, CAN, PDAC, ADC, DRO, FlexRay

Legend:

ICR0, ICR1:	Interrupt Control Register 0,1
INTPRI:	Interrupt Priority Level Setting Register
INT2PRI0 to 12:	Interrupt Priority Regiter
SR.IMASK:	Status Register. IMASK bits
USERIMASK.UIMASK:	User interrupt mask level register UIMASK bits

Figure 15.1 Block Diagram of INTC

15.1.1 Interrupt Request Sources in INTC

The INTC manages non-maskable interrupt (NMI interrupt) and general interrupt (IRQ interrupt and on-chip peripheral module interrupt) requests in exceptional handling. Tables 15.2 and 15.3 lists interrupt request sources, respectively.

Table 15.2 Interrupt Request Sources (NMI and IRQ Interrupts)

Module Name	Source Settable with Software-Configurable Interrupt Priority*1	Interrupt Request Source	INTEVT (Exception Code)	Hardware-Configurable Priority*2
INTC (NMI Interrupt)	—	NMIFL	H'1C0	High
INTC (IRQ Interrupt)	IRQ0	IRQ0	H'240	
	IRQ1	IRQ1	H'280	
	IRQ2	IRQ2	H'2C0	
	Reserved	Reserved	H'300	
	Reserved	Reserved	H'340	
	IRQ5	IRQ5	H'380	
	IRQ6	IRQ6	H'3C0	
	IRQ7	IRQ7	H'200	Low

Notes *1 By setting the INTPRI register, the priority can be selected from 0 to 15 level (setting 0 disables interrupts).
*2 The hardware-configurable priority is: NMI interrupt > IRQ interrupt > on-chip peripheral module interrupt.

Table 15.3 Interrupt Request Sources (On-Chip Peripheral Module Interrupts)

Module Name	Source Settable with Software-Configurable Interrupt Priority*1	Interrupt Request Sources*2	INTEVT (Exception Code)	Hardware-Configurable Priority*3
WDT	WDT	WDT	H'560	High
TMU	TUNI0	TUNI0	H'580	
	TUNI1	TUNI1	H'5A0	
	TUNI2	TUNI2	H'5C0	
H-UDI	HUDI	HUDI	H'600	
DMAC	DMAC0T3	DMINT0	H'700	
		DMINT1	H'720	
		DMINT2	H'740	
		DMINT3	H'760	
		DMAE0 (DMA0 to 5)	H'780	
		DMAE1 (DMA6 to 11)	H'7A0	
	DMAC4T5	DMINT4	H'7C0	
		DMINT5	H'840	
	DMAC6T11	DMINT6	H'860	
		DMINT7	H'880	
		DMINT8	H'8A0	
		DMINT9	H'8C0	
		DMINT10	H'8E0	Low

Module Name	Source Settable with Software-Configurable Interrupt Priority*1	Interrupt Request Sources*2	INTEVT (Exception Code)	Hardware-Configurable Priority*3
DMAC	DMAC6T11	DMINT11	H'900	High
ATU-IIIS Timer A	TA	ICIA00	H'A00	↑
		ICIA01	H'A20	
		ICIA02	H'A40	
		ICIA03	H'A60	
		ICIA04	H'A80	
		Reserved	H'AA0	
		OVIA0	H'AC0	
ATU-IIIS Timer F	TF	ICIF0	H'B00	
		ICIF1	H'B20	
		ICIF2	H'B40	
		Reserved	H'B60	
		OVIF0	H'B80	
		OVIF1	H'BA0	
		OVIF2	H'BC0	
		Reserved	H'BE0	
ATU-IIIS Timer G	CMIG0	CMIG0	H'C00	
	CMIG1	CMIG1	H'C20	
	CMIG2	CMIG2	H'C40	
	CMIG3	CMIG3	H'C60	
	CMIG4	CMIG4	H'C80	
	CMIG5	CMIG5	H'CA0	
ATU-IIIS Timer TOU	TOU00	TOU00UDF	H'D00	
		TOU01UDF	H'D20	
		TOU02UDF	H'D40	
		TOU03UDF	H'D60	
	TOU04	TOU04UDF	H'D80	
		TOU05UDF	H'DA0	
		TOU06UDF	H'DC0	
		TOU07UDF	H'DE0	
	TOU10	TOU10UDF	H'E00	
		TOU11UDF	H'E20	
		TOU12UDF	H'E40	
		TOU13UDF	H'E60	
	TOU14	TOU14UDF	H'E80	
		TOU15UDF	H'EA0	
		TOU16UDF	H'EC0	↓
		TOU17UDF	H'EE0	Low

Module Name	Source Settable with Software-Configurable Interrupt Priority*[1]	Interrupt Request Sources*[2]	INTEVT (Exception Code)	Hardware-Configurable Priority*[3]
ATU-IIIS Timer TOU	TOU20	TOU20UDF	H'F00	High
		TOU21UDF	H'F20	↑
		TOU22UDF	H'F40	
		TOU23UDF	H'F60	
	TOU24	TOU24UDF	H'F80	
		TOU25UDF	H'FA0	
		TOU26UDF	H'FC0	
		TOU27UDF	H'FE0	
	TOU30	TOU30UDF	H'1000	
		TOU31UDF	H'1020	
		TOU32UDF	H'1040	
		TOU33UDF	H'1060	
	TOU34	TOU34UDF	H'1080	
		TOU35UDF	H'10A0	
		TOU36UDF	H'10C0	
		TOU37UDF	H'10E0	
	TOU40	TOU40UDF	H'1100	
		TOU41UDF	H'1120	
		TOU42UDF	H'1140	
		TOU43UDF	H'1160	
	TOU44	TOU44UDF	H'1180	
		TOU45UDF	H'11A0	
		TOU46UDF	H'11C0	
		TOU47UDF	H'11E0	
SCIF	SCIF0	ERI0	H'1200	
		RXI0	H'1220	
		BRI0	H'1240	
		TXI0	H'1260	
	SCIF1	ERI1	H'1280	
		RXI1	H'12A0	
		BRI1	H'12C0	
		TXI1	H'12E0	
	SCIF2	ERI2	H'1300	
		RXI2	H'1320	
		BRI2	H'1340	
		TXI2	H'1360	
	SCIF3	ERI3	H'1380	
		RXI3	H'13A0	
		BRI3	H'13C0	↓
		TXI3	H'13E0	Low

Module Name	Source Settable with Software-Configurable Interrupt Priority*1	Interrupt Request Sources*2	INTEVT (Exception Code)	Hardware-Configurable Priority*3
RSPI	RSPI0	SPEI0	H'1400	High
		SPRI0	H'1420	
		SPTI0	H'1440	
	RSPI1	SPEI1	H'1460	
		SPRI1	H'1480	
		SPTI1	H'14A0	
	RSPI2	SPEI2	H'14C0	
		SPRI2	H'14E0	
		SPTI2	H'1500	
IIC3	IICI	IICI	H'1600	
DRI	DRI0	DRI0EVENT	H'1700	
		DRI0DEC	H'1720	
		DRI0TRM	H'1740	
	DRI1	DRI1EVENT	H'1760	
		DRI1DEC	H'1780	
		DRI1TRM	H'17A0	
	DRI2	DRI2EVENT	H'17C0	
		DRI2DEC	H'17E0	
		DRI2TRM	H'1800	
DRO	DRO	DRO	H'1820	
FlexRay*4	FRINT	FRINT0	H'1900	
		FRINT1	H'1920	
	FRTINT	FRTINT0	H'1940	
		FRTINT1	H'1960	
CAN	CAN0	ERS0	H'1A00	
		RXF0	H'1A20	
		TXF0	H'1A40	
		RXM00	H'1A80	
		RXM10	H'1AA0	
		TXM0	H'1AC0	
	CAN1	ERS1	H'1B00	
		RXF1	H'1B20	
		TXF1	H'1B40	
		RXM01	H'1B80	
		RXM11	H'1BA0	
		TXM1	H'1BC0	Low

Module Name	Source Settable with Software-Configurable Interrupt Priority*1	Interrupt Request Sources*2	INTEVT (Exception Code)	Hardware-Configurable Priority*3
CAN	CAN2	ERS2	H'1C00	High
		RXF2	H'1C20	
		TXF2	H'1C40	
		RXM02	H'1C80	
		RXM12	H'1CA0	
		TXM2	H'1CC0	
	CAN3	ERS3	H'1D00	
		RXF3	H'1D20	
		TXF3	H'1D40	
		RXM03	H'1D80	
		RXM13	H'1DA0	
		TXM3	H'1DC0	
ADC	ADC	AD0I	H'1E00	
		AD1I	H'1E20	
		AD0ID0	H'1E40	
		Reserved	H'1E60	
		AD0ID2	H'1E80	
		Reserved	H'1EA0	
		AD0ID4	H'1EC0	
		Reserved	H'1EE0	
		AD0ID6	H'1F00	
		Reserved	H'1F20	
		AD0ID8	H'1F40	
		AD0ID9	H'1F60	
		AD0ID10	H'1F80	
		AD0ID11	H'1FA0	
		AD0ID12	H'1FC0	
		AD0ID13	H'1FE0	
		AD0ID14	H'2000	
		AD0ID15	H'2020	
		AD1ID0	H'2040	
		AD1ID1	H'2060	
		Reserved	H'2080	
		Reserved	H'20A0	
		AD1ID4	H'20C0	
		AD1ID5	H'20E0	
		Reserved	H'2100	
		Reserved	H'2120	
PDAC	PDAC	PDIINT	H'2200	Low

Module Name	Source Settable with Software-Configurable Interrupt Priority*[1]	Interrupt Request Sources*[2]	INTEVT (Exception Code)	Hardware-Configurable Priority*[3]
ATU-IIIS Timer A	TA	ICIA10	H'2300	High
		ICIA11	H'2320	↑
		ICIA12	H'2340	
		Reserved	H'2360	
		ICIA14	H'2380	
		ICIA15	H'23A0	↓
		OVIA1	H'23C0	Low

Notes *1 By setting the INT2PRIn register (n = 0 to 12), the priorities of on-chip peripheral module interrupts can be selected from 0 to 31 levels (setting 0 or 1 disables interrupts).

*2 By setting the INT2Bm register (m = 0 to 10 and 12), the interrupt request sources can be checked.

*3 The hardware-configurable priority is: NMI interrupt > IRQ interrupt > on-chip peripheral module interrupt. In addition, the lower INTEVT (exception code) value has the higher hardware-configurable priority.

*4 The SH7456 Group does not include the FlexRay module.

15.2 Input/Output Pins

Table 15.4 shows the INTC pin configuration.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 15.4 Pin Configuration

Pin Name	I/O	Function
NMI	Input	Nonmaskable interrupt request signal input
IRQ0 to IRQ2 and IRQ5 to IRQ7	Input	External interrupt request signal input

15.3 Register Descriptions

Table 15.5 shows the INTC register configuration.

Table 15.5 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Interrupt control register 0	ICR0	Undefined	H'FFFF F000	32	15-11
Interrupt control register 1	ICR1	H'0000 0000	H'FFFF F01C	32	15-13
Interrupt priority register	INTPRI	H'0000 0000	H'FFFF F010	32	15-14
Interrupt source register	INTREQ	H'0000 0000	H'FFFF F024	32	15-15
Interrupt mask register	INTMSK	H'FF00 0000	H'FFFF F044	32	15-16
Interrupt mask clear register	INTMSKCLR	H'0000 0000	H'FFFF F064	32	15-17
NMI flag control register	NMIFCR	Undefined	H'FFFF F0C0	32	15-18
User interrupt mask level register	USERIMASK	H'0000 0000	H'FFFF F300	32	15-19
Interrupt priority setting register 0	INT2PRI0	H'0000 0000	H'FFFF F400	32	15-21
Interrupt priority setting register 1	INT2PRI1	H'0000 0000	H'FFFF F404	32	15-21
Interrupt priority setting register 2	INT2PRI2	H'0000 0000	H'FFFF F408	32	15-21
Interrupt priority setting register 3	INT2PRI3	H'0000 0000	H'FFFF F40C	32	15-21
Interrupt priority setting register 4	INT2PRI4	H'0000 0000	H'FFFF F410	32	15-21
Interrupt priority setting register 5	INT2PRI5	H'0000 0000	H'FFFF F414	32	15-21
Interrupt priority setting register 6	INT2PRI6	H'0000 0000	H'FFFF F418	32	15-21
Interrupt priority setting register 7	INT2PRI7	H'0000 0000	H'FFFF F41C	32	15-21
Interrupt priority setting register 8	INT2PRI8	H'0000 0000	H'FFFF F4A0	32	15-21
Interrupt priority setting register 9	INT2PRI9	H'0000 0000	H'FFFF F4A4	32	15-21
Interrupt priority setting register 10	INT2PRI10	H'0000 0000	H'FFFF F4A8	32	15-21
Interrupt priority setting register 11	INT2PRI11	H'0000 0000	H'FFFF F4AC	32	15-21
Interrupt priority setting register 12	INT2PRI12	H'0000 0000	H'FFFF F4B0	32	15-21
Interrupt source register 00 (mask state is not affected)	INT2A00	H'0000 0000	H'FFFF F430	32	15-23
Interrupt source register 01 (mask state is not affected)	INT2A01	H'0000 0000	H'FFFF F4C0	32	15-25
Interrupt source register 10 (mask state is affected)	INT2A10	H'0000 0000	H'FFFF F434	32	15-27
Interrupt source register 11 (mask state is affected)	INT2A11	H'0000 0000	H'FFFF F4C4	32	15-29

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Interrupt mask register 0	INT2MSKR	H'FFFF FFFF	H'FFFF F438	32	15-30
Interrupt mask register 1	INT2MSKR1	H'FFFF FFFF	H'FFFF F4D0	32	15-32
Interrupt mask clear register 0	INT2MSKCR	H'0000 0000	H'FFFF F43C	32	15-33
Interrupt mask clear register 1	INT2MSKCR1	H'0000 0000	H'FFFF F4D4	32	15-35
Per-module interrupt source register 0	INT2B0	H'0000 0000	H'FFFF F440	32	15-36
Per-module interrupt source register 1	INT2B1	H'0000 0000	H'FFFF F444	32	15-37
Per-module interrupt source register 2	INT2B2	H'0000 0000	H'FFFF F448	32	15-38
Per-module interrupt source register 3	INT2B3	H'0000 0000	H'FFFF F44C	32	15-39
Per-module interrupt source register 4	INT2B4	H'0000 0000	H'FFFF F450	32	15-41
Per-module interrupt source register 5	INT2B5	H'0000 0000	H'FFFF F454	32	15-42
Per-module interrupt source register 6	INT2B6	H'0000 0000	H'FFFF F458	32	15-43
Per-module interrupt source register 7	INT2B7	H'0000 0000	H'FFFF F45C	32	15-44
Per-module interrupt source register 8	INT2B8	H'0000 0000	H'FFFF F460	32	15-45
Per-module interrupt source register 10	INT2B10	H'0000 0000	H'FFFF F468	32	15-47
Per-module interrupt source register 11	INT2B11	H'0000 0000	H'FFFF F46C	32	15-49
Per-module interrupt source register 12	INT2B12	H'0000 0000	H'FFFF F494	32	15-50

Note:
- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

15.3.1 Interrupt Control Register 0 (ICR0)

The ICR0 register sets the input signal detection mode of NMI pin, and indicates the input level to the NMI pin.

Interrupt Control Register 0 (ICR0) <P4 address: location H'FFFF F000>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NMIL	MAI	—	—	—	—	NMIB	NMIE	—	—	—	—	—	—	—	—
After Reset:	Undefined	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31	NMIL	Undefined	R	—	NMI Input Level Bit This bit is set to the signal level input to the NMI pin. Applications can determine the NMI pin signal level by reading the NMIL bit. 0: "L" level is input to the NMI pin 1: "H" level is input to the NMI pin
30	MAI	0	R	W	NMI Interrupt Mask Bit CPU Specifies whether all interrupts are masked during the "L" level period of the NMI pin level regardless of the SR/BL bit. For details, see table 15.6. 0: Interrupts are enabled even if the NMI pin goes "L" level 1: Interrupts are disabled if the NMI pin goes "L" level
29 to 26	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
25	NMIB	0	R	W	NMI Block Mode Bit CPU Specifies whether an NMI interrupt is held until the BL bit is cleared to "0" or detected immediately when the SR/BL bit is set to "1". 0: NMI interrupts will be deferred when BL is "1" 1: NMI interrupts will not be deferred when BL is "1" Note: If interrupts are accepted when BL is "1", the previously saved exception information (SSR, SPC, SGR, and INTEVT) will be lost.
24	NMIE	0	R	W	NMI Edge Select Bit Selects whether an interrupt request signal input to the NMI pin is detected at the rising or the falling edge. When this bit is modified, the NMI interrupt is not detected for a period of up to 6 Pck cycles after the value of the bit is changed. 0: An interrupt request is detected at the falling edge of NMI input 1: An interrupt request is detected at the rising edge of NMI input
23	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1"
22 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Table 15.6 Combinations of BL Bit and MAI Bit Settings

BL Bit in SR	MAI Bit in ICR0	NMI Pin Level	Interrupt Enabled/Disabled
"0"	"0"	"L"	Enabled
		"H"	Enabled
	"1"	"L"	Disabled
		"H"	Enabled
"1"	"0"	"L"	Disabled
		"H"	Disabled
	"1"	"L"	Disabled
		"H"	Disabled

15.3.2 Interrupt Control Register 1 (ICR1)

The ICR1 register specifies the detection mode for IRQ interrupts (IRQ0 to IRQ2 and IRQ5 to IRQ7).

Interrupt Control Register 1 (ICR1) <P4 address: location H'FFFF F01C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ0S		IRQ1S		IRQ2S		—	—	—	—	IRQ5S		IRQ6S		IRQ7S	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31, 30	IRQ0S	All 0	R	W	IRQn Sense Select Bits These bits select whether an interrupt request signal input to the corresponding IRQ pin (IRQ0 to IRQ2 and IRQ5 to IRQ7) is detected at the rising edge, falling edge, "L" level, or "H" level. The IRQ0S bits correspond to the IRQ0 pin and the IRQ7S bits correspond to the IRQ7. 00: Interrupt requests are detected on an IRQn falling edge. 01: Interrupt requests are detected on an IRQn rising edge. 10: Interrupt requests are detected on an IRQn "L" level. 11: Interrupt requests are detected on an IRQn "H" level.
29, 28	IRQ1S	All 0	R	W	
27, 26	IRQ2S	All 0	R	W	
25 to 22	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
21, 20	IRQ5S	All 0	R	W	IRQn Sense Select Bits These bits select whether an interrupt request signal input to the corresponding IRQ pin (IRQ0 to IRQ2 and IRQ5 to IRQ7) is detected at the rising edge, falling edge, "L" level, or "H" level. The IRQ0S bits correspond to the IRQ0 pin and the IRQ7S bits correspond to the IRQ7. 00: Interrupt requests are detected on an IRQn falling edge. 01: Interrupt requests are detected on an IRQn rising edge. 10: Interrupt requests are detected on an IRQn "L" level. 11: Interrupt requests are detected on an IRQn "H" level.
19, 18	IRQ6S	All 0	R	W	
17, 16	IRQ7S	All 0	R	W	
15 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: n = 0 to 2, 5 to 7

15.3.3 Interrupt Priority Register (INTPRI)

The INTPRI register sets the IRQ interrupt (IRQ0 to IRQ2 and IRQ5 to IRQ7) priorities (levels 15 to 0).

Interrupt Priority Register (INTPRI) <P4 address: location H'FFFF F010>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ0				IRQ1				IRQ2				—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	IRQ5				IRQ6				IRQ7			
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 28	IRQ0	All 0	R	W	IRQn Interrupt Priority Level Bits These bits are used to set the interrupt priority of the corresponding IRQn to a value from H'F to H'1 (level 15 and level 1). The higher the value, the higher the priority. Setting a field to a value of H'0 (level 0) causes the corresponding interrupt to be masked.
27 to 24	IRQ1	All 0	R	W	
23 to 20	IRQ2	All 0	R	W	
19 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11 to 8	IRQ5	All 0	R	W	IRQn Interrupt Priority Level Bits These bits are used to set the interrupt priority of the corresponding IRQn to a value from H'F to H'1 (level 15 and level 1). The higher the value, the higher the priority. Setting a field to a value of H'0 (level 0) causes the corresponding interrupt to be masked.
7 to 4	IRQ6	All 0	R	W	
3 to 0	IRQ7	All 0	R	W	

Legend: n = 0 to 2, 5 to 7

15.3.4 Interrupt Source Register (INTREQ)

The INTREQ register indicates which of the IRQ0 to IRQ2 and IRQ5 to IRQ7 interrupts has been requested to the INTC. The bits in this register are not influenced by interrupt masking with the INTPRI and INTMSK registers.

Interrupt Source Register (INTREQ) <P4 address: location H'FFFF F024>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ0	IRQ1	IRQ2	—	—	IRIRQ	IRQ6	IRQ7	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	IRQ0	0	R	*1	IRQn Interrupt Source Bits Each bit indicates whether or not an interrupt request has been input to the pin corresponding to IRQn. • Edge detection (IRQnS in ICR1 set to "00" or "01") • Level detection (IRQnS in ICR1 set to "10" or "11") Writing to these bits has no effect. For the method of clearing these bits, see section 15.7.2, To Clear IRQ Interrupt Requests When Level Detection is Selected. 0: No interrupt request detected 1: Interrupt request detected
30	IRQ1	0	R	*1	
29	IRQ2	0	R	*1	
28, 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26	IRQ5	0	R	*1	IRQn Interrupt Source Bits Each bit indicates whether or not an interrupt request has been input to the pin corresponding to IRQn. • Edge detection (IRQnS in ICR1 set to "00" or "01") • Level detection (IRQnS in ICR1 set to "10" or "11") Writing to these bits has no effect. For the method of clearing these bits, see section 15.7.2, To Clear IRQ Interrupt Requests When Level Detection Is Selected. 0: No interrupt request detected 1: Interrupt request detected
25	IRQ6	0	R	*1	
24	IRQ7	0	R	*1	
23 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Note: *1 To clear a flag when edge detection is enabled, read the register and then write "0" to the flag bit only if its value was read as "1". Write "1" to the bits whose value was read as "0". Always write "1" to flag bits not to clear.

Legend: n = 0 to 2, 5 to 7

15.3.5 Interrupt Mask Register (INTMSK)

The INTMSK register indicates whether or not each of the IRQ0 to IRQ2 and IRQ5 to IRQ7 interrupt requests are masked. To clear the interrupt masking, write "1" to the corresponding bit in the INTMASKCLR register. Writing "0" to each bit in this register does not change the value.

Interrupt Mask Register (INTMSK) <P4 address: location H'FFFF F044>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ0	IRQ1	IRQ2	—	—	IRQ5	IRQ6	IRQ7	—	—	—	—	—	—	—	—
After Reset:	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'FF00 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	IRQ0	1	R	W	IRQn Interrupt Request Mask Setting Bits Each bit specifies whether or not interrupt requests corresponding to IRQn are masked. Writing "0" to these bits has no effect. For the method of clearing these bits, see section 15.3.6, Interrupt Mask Clear Register (INTMSKCLR). 0: Masking disabled 1: Masking enabled
30	IRQ1	1	R	W	
29	IRQ2	1	R	W	
28, 27	—	All 1	1	1	Reserved Bits These bits are always read as "1". The write value should always be "1".
26	IRQ5	1	R	W	IRQn Interrupt Request Mask Setting Bits Each bit specifies whether or not interrupt requests corresponding to IRQn are masked. Writing "0" to these bits has no effect. For the method of clearing these bits, see section 15.3.6, Interrupt Mask Clear Register (INTMSKCLR). 0: Masking disabled 1: Masking enabled
25	IRQ6	1	R	W	
24	IRQ7	1	R	W	
23 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: n = 0 to 2, 5 to 7

15.3.6 Interrupt Mask Clear Register (INTMSKCLR)

The INTMSKCLR register is used to clear masking of IRQ0 to IRQ2 and IRQ5 to IRQ7 interrupt requests set in the INTMSK register. Setting a bit in the INTMSKCLR register to "1" clears masking of the corresponding interrupt source. The value of the bits when read is undefined.

Interrupt Mask Clear Register (INTMSKCLR) <P4 address: location H'FFFF F064>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ0	IRQ1	IRQ2	—	—	IRQ5	IRQ6	IRQ7	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	IRQ0	0	?	W	IRQn Interrupt Request Mask Clear Bits Each bit specifies whether or not the interrupt request mask setting of the corresponding IRQn is cleared. These bits return an undefined value when read. Writing "0" to these bits has no effect. 0: Mask setting not cleared 1: Mask setting cleared
30	IRQ1	0	?	W	
29	IRQ2	0	?	W	
28, 27	—	All 0	?	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26	IRQ5	0	?	W	IRQn Interrupt Request Mask Clear Bits Each bit specifies whether or not the interrupt request mask setting of the corresponding IRQn is cleared. These bits return an undefined value when read. Writing "0" to these bits has no effect. 0: Mask setting not cleared 1: Mask setting cleared
25	IRQ6	0	?	W	
24	IRQ7	0	?	W	
23 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: n = 0 to 2, 5 to 7

15.3.7 NMI Flag Control Register (NMIFCR)

The NMIFCR register indicates whether or not an NMI interrupt has been detected by the INTC. The NMIFL bit is automatically set to "1" by hardware when an NMI interrupt is detected by the INTC. To clear the NMIFL bit, write "0" to the bit by software.

The NMIFL bit value does not affect NMI acceptance by the CPU. Although the NMI request detected by the INTC is cleared by CPU acceptance, the NMIFL bit is not cleared automatically. Even if "0" is written to the NMIFL bit before the NMI request is accepted by the CPU, the NMI request is not canceled.

NMI Flag Control Register (NMIFCR) <P4 address: location H'FFFF F0C0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NMIL	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NMIFL
After Reset:	Undefined	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31	NMIL	Undefined	R	—	NMI Input Level Bit This bit indicates the level of input to the NMI pin. 0: "L" level is input to the NMI pin 1: "H" level is input to the NMI pin
30 to 17	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
16	NMIFL	0	R	W	NMI Flag (NMI interrupt request signal detection) Bit This bit indicates whether or not an NMI interrupt has been detected by the INTC. Always write "0" to this bit to clear. Writing "1" to this bit has no effect. 0: No NMI interrupt detected 1: NMI interrupt request detected
15 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

15.3.8 User Interrupt Mask Level Register (USERIMASK)

The USERIMASK register sets the interrupt level.

Interrupts whose priority levels are lower than the level set in the UIMASK bits are masked. If the value of "H'F" is set to the UIMASK bits, all interrupts other than the NMI are masked.

The CPU only accepts interrupts with an interrupt level setting higher than the setting values of the UIMASK bits and the IMASK bits in the SR register.

Even if interrupts are accepted by the CPU, the UIMASK bit value is not changed.

This register is initialized to H'0000 0000 (all interrupts are enabled) when returning from a hardware reset.

To prevent incorrect writing, this register can only be written to with USERIMASKKEY bit (bits 31 to 24) set to H'A5.

User Interrupt Mask Level Register (USERIMASK)   <P4 address: location H'FFFF F300>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	USERIMASKKEY								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UIMASK				—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	USERIMASKKEY	All 0	0	W	USERINASK Register Write Key Code Bits These bits enable or disable UIMASK bit modification. The data written to these bits are not retained. These bits are always read as "0". H'A5: Enable UIMASK bit modification. Other than H'A5: Disable UIMASK bit modification.
23 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7 to 4	UIMASK	All 0	R	W	Interrupt Mask Level Bits Masks interrupts whose priority levels are lower than the level set in the UIMASK bits.
3 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(1) Procedure for Using User Interrupt Mask Level Register

When accessing this register in user mode, translate the address through the MMU. In the system that uses a multitasking OS, processes that can access the USERIMASK register must be controlled by using memory protection functions of the MMU. When terminating the task or switching to another task, be sure to clear the UIMASK bits to "H'0" before quitting the task. If the UIMASK bits are left set to a non-zero value, interrupts which are not higher in priority than the UIMASK level are held disabled, and correct operation may not be performed (for example, the OS cannot switch tasks).

An example of the usage procedure is shown below.

1. Classify interrupts to A and B as described below and set the A priority higher than the B priority.
   A. Interrupts to be accepted in the device driver (interrupts to be used by the operating system: a timer interrupt etc.)
   B. Interrupts to be disabled in the device driver
2. Make the MMU settings so that the address space including the USERIMASK register can only be accessed by the device driver in which interrupts should be disabled.
3. Branch to the device driver.
4. Set the UIMASK bits to mask B interrupts in the device driver that is operating in user mode.
5. Process interrupts with high priority in the device driver.
6. Clear the UIMASK bits to "H'0" to return from processing in the device driver.

15.3.9 Interrupt Priority Setting Registers 0 to 12 (INT2PRI0 to INT2PRI12)

The INT2PRI0 to INT2PRI12 registers set the priorities (as levels 0 to 31) of the on-chip peripheral module interrupts. The higher the setting value, the higher the priority. Each interrupt source can be given one of 30 priority levels by assigning one of 32 5-bit values. (Specifying a value of H’00 or H’01 has the same effect as masking interrupt requests for the corresponding source)

Interrupt Priority Setting Register 0 (INT2PRI0)	<P4 address: location H'FFFF F400>
Interrupt Priority Setting Register 1 (INT2PRI1)	<P4 address: location H'FFFF F404>
Interrupt Priority Setting Register 2 (INT2PRI2)	<P4 address: location H'FFFF F408>
Interrupt Priority Setting Register 3 (INT2PRI3)	<P4 address: location H'FFFF F40C>
Interrupt Priority Setting Register 4 (INT2PRI4)	<P4 address: location H'FFFF F410>
Interrupt Priority Setting Register 5 (INT2PRI5)	<P4 address: location H'FFFF F414>
Interrupt Priority Setting Register 6 (INT2PRI6)	<P4 address: location H'FFFF F418>
Interrupt Priority Setting Register 7 (INT2PRI7)	<P4 address: location H'FFFF F41C>
Interrupt Priority Setting Register 8 (INT2PRI8)	<P4 address: location H'FFFF F4A0>
Interrupt Priority Setting Register 9 (INT2PRI9)	<P4 address: location H'FFFF F4A4>
Interrupt Priority Setting Register 10 (INT2PRI10)	<P4 address: location H'FFFF F4A8>
Interrupt Priority Setting Register 11 (INT2PRI11)	<P4 address: location H'FFFF F4AC>
Interrupt Priority Setting Register 12 (INT2PRI12)	<P4 address: location H'FFFF F4B0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	See table 15.7					—	—	—	See table 15.7				
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	See table 15.7					—	—	—	See table 15.7				
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 24	See table 15.7	All 0	R	W	These bits are allocated to the module that is the interrupt request origin for this on-chip peripheral module interrupt. See table 15.7 for the correspondence between these registers and the interrupt request origins.
23 to 21	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
20 to 16	See table 15.7	All 0	R	W	These bits are allocated to the module that is the interrupt request origin for this on-chip peripheral module interrupt. See table 15.7 for the correspondence between these registers and the interrupt request origins.
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12 to 8	See table 15.7	All 0	R	W	These bits are allocated to the module that is the interrupt request origin for this on-chip peripheral module interrupt. See table 15.7 for the correspondence between these registers and the interrupt request origins.
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
4 to 0	See table 15.7	All 0	R	W	These bits are allocated to the module that is the interrupt request origin for this on-chip peripheral module interrupt. See table 15.7 for the correspondence between these registers and the interrupt request origins.

Table 15.7 shows the correspondence between interrupt request sources and bits in the INT2PRI0 to INT2PRI12 registers.

Table 15.7 Interrupt Request Sources and INT2PRI0 to INT2PRI12 Registers

	Bit			
Register	28 to 24	20 to 16	12 to 8	4 to 0
INT2PRI0	TUNI0 (TMU)	TUNI1 (TMU)	TUNI2 (TMU)	Reserved
INT2PRI1	WDT (WDT)	DMAC0T3 (DMAC)	DMAC4T5 (DMAC)	DMAC6T11 (DMAC)
INT2PRI2	SCIF0 (SCIF)	SCIF1 (SCIF)	SCIF2 (SCIF)	SCIF3 (SCIF)
INT2PRI3	RSPI0 (RSPI)	RSPI1 (RSPI)	RSPI2 (RSPI)	HUDI (H-UDI)
INT2PRI4	DRI0 (DRI)	DRI1 (DRI)	DRI2 (DRI)	DRO (DRO)
INT2PRI5	IICI (IIC3)	ADC (ADC)	TA (ATU-IIIS timer A)	TF (ATU-IIIS timer F)
INT2PRI6	CMIG0 (ATU-IIIS timer G)	CMIG1 (ATU-IIIS timer G)	CMIG2 (ATU-IIIS timer G)	CMIG3 (ATU-IIIS timer G)
INT2PRI7	CMIG4 (ATU-IIIS timer G)	CMIG5 (ATU-IIIS timer G)	TOU00 (ATU-IIIS timer TOU)	TOU04 (ATU-IIIS timer TOU)
INT2PRI8	TOU10 (ATU-IIIS timer TOU)	TOU14 (ATU-IIIS timer TOU)	TOU20 (ATU-IIIS timer TOU)	TOU24 (ATU-IIIS timer TOU)
INT2PRI9	TOU30 (ATU-IIIS timer TOU)	TOU34 (ATU-IIIS timer TOU)	TOU40 (ATU-IIIS timer TOU)	TOU44 (ATU-IIIS timer TOU)
INT2PRI10	CAN0 (CAN)	CAN1 (CAN)	CAN2 (CAN)	CAN3 (CAN)
INT2PRI11	Reserved	Reserved	FRINT*[1] (FlexRay)	FRTINT*[1] (FlexRay)
INT2PRI12	PDAC (PDAC)	Reserved	Reserved	Reserved

Note: *1 Reserved bits in the SH7456 Group. These bits are always read as "0". The write value should always be "0".

15.3.10 Interrupt Source Register 00 (INT2A00) (Mask State is not affected)

The INT2A00 register indicates interrupt sources generated by peripheral modules. Even if an interrupt is masked by the INT2MSKR register, the INT2A00 register still indicates the source by setting the corresponding bit. For interrupt source indications according to the setting of the INT2MSKR register (such that sources are not indicated for masked interrupts), use the INT2A10 register. Clearing of the interrupt source matching a particular bit must be performed in the corresponding peripheral module. It is also possible to identify interrupt sources by directly reading the INTEVT (exception code) sent to the CPU.

Interrupt Source Register 00 (INT2A00) (Mask State is not affected) <P4 address: location H'FFFF F430>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMIG5	CMIG4	CMIG3	CMIG2	CMIG1	CMIG0	TF	TA	ADC	IICI	DRO	DRI2	DRI1	DRI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HUDI	RSPI2	RSPI1	RSPI0	SCIF3	SCIF2	SCIF1	SCIF0	DMAC 6T11	DMAC 4T5	DMAC 0T3	WDT	—	TUNI2	TUNI1	TUNI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31, 30	—	All 0	0	—	Reserved Bits These bits are always read as "0".	
29	CMIG5	0	R	—	Timer G5 interrupt source indication bit	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
28	CMIG4	0	R	—	Timer G4 interrupt source indication bit	
27	CMIG3	0	R	—	Timer G3 interrupt source indication bit	
26	CMIG2	0	R	—	Timer G2 interrupt source indication bit	
25	CMIG1	0	R	—	Timer G1 interrupt source indication bit	
24	CMIG0	0	R	—	Timer G0 interrupt source indication bit	
23	TF	0	R	—	Timer F interrupt source indication bit	
22	TA	0	R	—	Timer A interrupt source indication bit	
21	ADC	0	R	—	ADC interrupt source indication bit	
20	IICI	0	R	—	IIC3 interrupt source indication bit	
19	DRO	0	R	—	DRO interrupt source indication bit	
18	DRI2	0	R	—	DRI2 interrupt source indication bit	
17	DRI1	0	R	—	DRI1 interrupt source indication bit	
16	DRI0	0	R	—	DRI0 interrupt source indication bit	
15	HUDI	0	R	—	H-UDI interrupt source indication bit	
14	RSPI2	0	R	—	RSPI2 interrupt source indication bit	
13	RSPI1	0	R	—	RSPI1 interrupt source indication bit	
12	RSPI0	0	R	—	RSPI0 interrupt source indication bit	
11	SCIF3	0	R	—	SCIF3 interrupt source indication bit	
10	SCIF2	0	R	—	SCIF2 interrupt source indication bit	
9	SCIF1	0	R	—	SCIF1 interrupt source indication bit	
8	SCIF0	0	R	—	SCIF0 interrupt source indication bit	
7	DMAC6T11	0	R	—	DMA6 to DMA11 interrupt source indication bit	

Bit	Abbreviation	After Reset	R	W	Description	
6	DMAC4T5	0	R	—	DMA4 to DMA4 interrupt source indication bit	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
5	DMAC0T3	0	R	—	DMA0 to DMA3 interrupt source indication bit	
4	WDT	0	R	—	WDT interrupt source indication bit	
3	—	0	0	—	Reserved Bit This bit is always read as "0".	
2	TUNI2	0	R	—	TMU2 interrupt source indication bit	
1	TUNI1	0	R	—	TMU1 interrupt source indication bit	
0	TUNI0	0	R	—	TMU0 interrupt source indication bit	

15.3.11 Interrupt Source Register 01 (INT2A01) (Mask State is not affected)

The INT2A01 register indicates interrupt sources generated by on-chip peripheral modules. Even if an interrupt is masked by the INT2MSKR1 register, the INT2A01 register still indicates the source by setting the corresponding bit. For interrupt source indications according to the setting of the INT2MSKR1 register (such that sources are not indicated for masked interrupts), use the INT2A11 register. Clearing of the interrupt source matching a particular bit must be performed in the corresponding on-chip peripheral module. It is also possible to identify interrupt sources by directly reading the INTEVT (exception code) sent to the CPU.

Interrupt Source Register 01 (INT2A01) (Mask State is not affected) <P4 address: location H'FFFF F4C0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	PDAC	FRTINT	FRINT
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CAN3	CAN2	CAN1	CAN0	TOU44	TOU40	TOU34	TOU30	TOU24	TOU20	TOU14	TOU10	TOU04	TOU00
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 19	—	All 0	0	—	Reserved Bits These bits are always read as "0".	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
18	PDAC	0	R	—	PDAC interrupt source indication bit	
17	FRTINT	0	R	—	Indicates FlexRay timer interrupt source indication bit Reserved bit in the SH7456 Group. This bit is always read as "0".	
16	FRINT	0	R	—	Indicates FlexRay interrupt source indication bit Reserved bit in the SH7456 Group. This bit is always read as "0".	
15, 14	—	All 0	0	—	Reserved Bits These bits are always read as "0".	
13	CAN3	0	R	—	CAN3 interrupt source indication bit	
12	CAN2	0	R	—	CAN2 interrupt source indication bit	
11	CAN1	0	R	—	CAN1 interrupt source indication bit	
10	CAN0	0	R	—	CAN0 interrupt source indication bit	
9	TOU44	0	R	—	Timer TOU4_4 to TOU4_7 interrupt source indication bit	
8	TOU40	0	R	—	Timer TOU4_0 to TOU4_3 interrupt source indication bit	
7	TOU34	0	R	—	Timer TOU3_4 to TOU3_7 interrupt source indication bit	
6	TOU30	0	R	—	Timer TOU3_0 to TOU3_3 interrupt source indication bit	
5	TOU24	0	R	—	Timer TOU2_4 to TOU2_7 interrupt source indication bit	

Bit	Abbreviation	After Reset	R	W	Description	
4	TOU20	0	R	—	Timer TOU2_0 to TOU2_3 interrupt source indication bit	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
3	TOU14	0	R	—	Timer TOU1_4 to TOU1_7 interrupt source indication bit	
2	TOU10	0	R	—	Timer TOU1_0 to TOU1_3 interrupt source indication bit	
1	TOU04	0	R	—	Timer TOU0_4 to TOU0_7 interrupt source indication bit	
0	TOU00	0	R	—	Timer TOU0_0 to TOU0_3 interrupt source indication bit	

15.3.12 Interrupt Source Register 10 (INT2A10) (Mask State is affected)

The INT2A10 register indicates interrupt sources generated by peripheral modules. However, if an interrupt is masked by the INT2MSKR register, the INT2A10 register does not indicate the source (the corresponding bit in this register is not set to "1"). To confirm interrupt occurrence with indications not affected by the setting of the INT2MSKR register, use the INT2A00 register. Clearing of the interrupt source matching a particular bit must be performed in the corresponding peripheral module. It is also possible to identify interrupt sources by directly reading the INTEVT (exception code) sent to the CPU.

Interrupt Source Register 10 (INT2A10) (Mask State is affected) <P4 address: location H'FFFF F434>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMIG5	CMIG4	CMIG3	CMIG2	CMIG1	CMIG0	TF	TA	ADC	IICI	DRO	DRI2	DRI1	DRI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HUDI	RSPI2	RSPI1	RSPI0	SCIF3	SCIF2	SCIF1	SCIF0	DMAC 6T11	DMAC 4T5	DMAC 0T3	WDT	—	TUNI2	TUNI1	TUNI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31, 30	—	All 0	0	—	Reserved Bits These bits are always read as "0".	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
29	CMIG5	0	R	—	Timer G5 interrupt source indication bit	
28	CMIG4	0	R	—	Timer G4 interrupt source indication bit	
27	CMIG3	0	R	—	Timer G3 interrupt source indication bit	
26	CMIG2	0	R	—	Timer G2 interrupt source indication bit	
25	CMIG1	0	R	—	Timer G1 interrupt source indication bit	
24	CMIG0	0	R	—	Timer G0 interrupt source indication bit	
23	TF	0	R	—	Timer F interrupt source indication bit	
22	TA	0	R	—	Timer A interrupt source indication bit	
21	ADC	0	R	—	ADC interrupt source indication bit	
20	IICI	0	R	—	IIC3 interrupt source indication bit	
19	DRO	0	R	—	DRO interrupt source indication bit	
18	DRI2	0	R	—	DRI2 interrupt source indication bit	
17	DRI1	0	R	—	DRI1 interrupt source indication bit	
16	DRI0	0	R	—	DRI0 interrupt source indication bit	
15	HUDI	0	R	—	H-UDI interrupt source indication bit	
14	RSPI2	0	R	—	RSPI2 interrupt source indication bit	
13	RSPI1	0	R	—	RSPI1 interrupt source indication bit	
12	RSPI0	0	R	—	RSPI0 interrupt source indication bit	
11	SCIF3	0	R	—	SCIF3 interrupt source indication bit	
10	SCIF2	0	R	—	SCIF2 interrupt source indication bit	
9	SCIF1	0	R	—	SCIF1 interrupt source indication bit	

Bit	Abbreviation	After Reset	R	W	Description	
8	SCIF0	0	R	—	SCIF0 interrupt source indication bit	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
7	DMAC6T11	0	R	—	DMA6 to DMA11 interrupt source indication bit	
6	DMAC4T5	0	R	—	DMA4 to DMA4 interrupt source indication bit	
5	DMAC0T3	0	R	—	DMA0 to DMA3 interrupt source indication bit	
4	WDT	0	R	—	Indicates WDT interrupt source indication bit	
3	—	0	0	—	Reserved Bit This bit is always read as "0".	
2	TUNI2	0	R	—	TMU2 interrupt source indication bit	
1	TUNI1	0	R	—	TMU1 interrupt source indication bit	
0	TUNI0	0	R	—	TMU0 interrupt source indication bit	

15.3.13 Interrupt Source Register 11 (INT2A11) (Mask State is affected)

The INT2A11 register indicates interrupt sources generated by on-chip peripheral modules. However, if an interrupt is masked by the INT2MSKR1 register, the INT2A11 register does not indicate the source (by setting the corresponding bit to "1"). To confirm interrupt occurrence with indications not affected by the setting of the INT2MSKR1 register, use the INT2A01 register. Clearing of the interrupt source matching a particular bit must be performed in the corresponding on-chip peripheral module. It is also possible to identify interrupt sources by directly reading the INTEVT (exception code) sent to the CPU.

Interrupt Source Register 11 (INT2A11) (Mask State is affected) <P4 address: location H'FFFF F4C4>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	PDAC	FRTINT	FRINT
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CAN3	CAN2	CAN1	CAN0	TOU44	TOU40	TOU34	TOU30	TOU24	TOU20	TOU14	TOU10	TOU04	TOU00
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 19	—	All 0	0	—	Reserved Bits These bits are always read as "0".	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
18	PDAC	0	R	—	PDAC interrupt source indication bit	
17	FRTINT	0	R	—	FlexRay timer interrupt source indication bit Reserved bit in the SH7456 Group. This bit is always read as "0".	
16	FRINT	0	R	—	FlexRay interrupt source indication bit Reserved bit in the SH7456 Group. This bit is always read as "0".	
15, 14	—	All 0	0	—	Reserved Bits These bits are always read as "0".	
13	CAN3	0	R	—	CAN3 interrupt source indication bit	
12	CAN2	0	R	—	CAN2 interrupt source indication bit	
11	CAN1	0	R	—	CAN1 interrupt source indication bit	
10	CAN0	0	R	—	CAN0 interrupt source indication bit	
9	TOU44	0	R	—	Timer TOU4_4 to TOU4_7 interrupt source indication bit	
8	TOU40	0	R	—	Timer TOU4_0 to TOU4_3 interrupt source indication bit	
7	TOU34	0	R	—	Timer TOU3_4 to TOU3_7 interrupt source indication bit	
6	TOU30	0	R	—	Timer TOU3_0 to TOU3_3 interrupt source indication bit	
5	TOU24	0	R	—	Timer TOU2_4 to TOU2_7 interrupt source indication bit	

Bit	Abbreviation	After Reset	R	W	Description	
4	TOU20	0	R	—	Timer TOU2_0 to TOU2_3 interrupt source indication bit	Indicates interrupt sources for each on-chip peripheral module. 0: No interrupts 1: Interrupts are generated
3	TOU14	0	R	—	Timer TOU1_4 to TOU1_7 interrupt source indication bit	
2	TOU10	0	R	—	Timer TOU1_0 to TOU1_3 interrupt source indication bit	
1	TOU04	0	R	—	Timer TOU0_4 to TOU0_7 interrupt source indication bit	
0	TOU00	0	R	—	Timer TOU0_0 to TOU0_3 interrupt source indication bit	

15.3.14 Interrupt Mask Register 0 (INT2MSKR)

The INT2MSKR register sets masking for each source indicated in the INT2A10 register. Interrupts whose corresponding bits in INT2MSKR register are set to "1" are not notified to the CPU.

Interrupt Mask Register 0 (INT2MSKR0) <P4 address: location H'FFFF F438>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMIG5	CMIG4	CMIG3	CMIG2	CMIG1	CMIG0	TF	TA	ADC	IICI	DRO	DRI2	DRI1	DRI0
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HUDI	RSPI2	RSPI1	RSPI0	SCIF3	SCIF2	SCIF1	SCIF0	DMAC 6T11	DMAC 4T5	DMAC 0T3	WDT	—	TUNI2	TUNI1	TUNI0
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

<After Reset: H'FFFF FFFF>

Bit	Abbreviation	After Reset	R	W	Description	
31, 30	—	All 1	1	1	Reserved Bits These bits are always read as "1". The write value should always be "1".	Masks interrupts for each on-chip peripheral module. Writing to this bit is invalid. 0: No mask setting 1: Mask setting
29	CMIG5	1	R	W	Timer G5 interrupts mask setting bit	
28	CMIG4	1	R	W	Timer G4 interrupts mask setting bit	
27	CMIG3	1	R	W	Timer G3 interrupts mask setting bit	
26	CMIG2	1	R	W	Timer G2 interrupts mask setting bit	
25	CMIG1	1	R	W	Timer G1 interrupts mask setting bit	
24	CMIG0	1	R	W	Timer G0 interrupts mask setting bit	
23	TF	1	R	W	Timer F interrupts mask setting bit	
22	TA	1	R	W	Timer A interrupts mask setting bit	
21	ADC	1	R	W	ADC interrupts mask setting bit	
20	IICI	1	R	W	IIC3 interrupts mask setting bit	
19	DRO	1	R	W	DRO interrupts mask setting bit	
18	DRI2	1	R	W	DRI2 interrupts mask setting bit	
17	DRI1	1	R	W	DRI1 interrupts mask setting bit	
16	DRI0	1	R	W	DRI0 interrupts mask setting bit	
15	HUDI	1	R	W	H-UDI interrupts mask setting bit	
14	RSPI2	1	R	W	RSPI2 interrupts mask setting bit	
13	RSPI1	1	R	W	RSPI1 interrupts mask setting bit	
12	RSPI0	1	R	W	RSPI0 interrupts mask setting bit	
11	SCIF3	1	R	W	SCIF3 interrupts mask setting bit	
10	SCIF2	1	R	W	SCIF2 interrupts mask setting bit	
9	SCIF1	1	R	W	SCIF1 interrupts mask setting bit	
8	SCIF0	1	R	W	SCIF0 interrupts mask setting bit	
7	DMAC6T11	1	R	W	DMA6 to DMA11 interrupts mask setting bit	
6	DMAC4T5	1	R	W	DMA4 to DMA4 interrupts mask setting bit	
5	DMAC0T3	1	R	W	DMA0 to DMA3 interrupts mask setting bit	
4	WDT	1	R	W	WDT interrupts mask setting bit	
3	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".	
2	TUNI2	1	R	W	TMU2 interrupts mask setting bit	
1	TUNI1	1	R	W	TMU1 interrupts mask setting bit	
0	TUNI0	1	R	W	TMU0 interrupts mask setting bit	

15.3.15 Interrupt Mask Register 1 (INT2MSKR1)

The INT2MSKR1 register sets masking for each source indicated in the INT2A11 register. Interrupts whose corresponding bits in INT2MSKR1 are set to 1 are not notified to the CPU.

Interrupt Mask Register 1 (INT2MSKR1) <P4 address: location H'FFFF F4D0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	PDAC	FRTINT	FRINT
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CAN3	CAN2	CAN1	CAN0	TOU44	TOU40	TOU34	TOU30	TOU24	TOU20	TOU14	TOU10	TOU04	TOU00
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

<After Reset: H'FFFF FFFF>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 19	—	All 1	1	1	Reserved Bits These bits are always read as "1". The write value should always be "1".	Masks interrupts for each on-chip peripheral module. Writing to this bit is invalid. 0: No mask setting 1: Mask setting
18	PDAC	1	R	W	PDAC interrupts mask setting bit	
17	FRTINT	1	R	W	FlexRay timer interrupt mask setting bit Reserved bit in the SH7456 Group. This bit is always read as "1". The write value should always be "1".	
16	FRINT	1	R	W	FlexRay interrupt mask setting bit Reserved bit in the SH7456 Group. This bit is always read as "1". The write value should always be "1".	
15, 14	—	All 1	1	1	Reserved Bits These bits are always read as "1". The write value should always be "1".	
13	CAN3	1	R	W	CAN3 interrupts mask setting bit	
12	CAN2	1	R	W	CAN2 interrupts mask setting bit	
11	CAN1	1	R	W	CAN1 interrupts mask setting bit	
10	CAN0	1	R	W	CAN0 interrupts mask setting bit	
9	TOU44	1	R	W	Timer TOU4_4 to TOU4_7 interrupts mask setting bit	
8	TOU40	1	R	W	Timer TOU4_0 to TOU4_3 interrupts mask setting bit	
7	TOU34	1	R	W	Timer TOU3_4 to TOU3_7 interrupts mask setting bit	
6	TOU30	1	R	W	Timer TOU3_0 to TOU3_3 interrupts mask setting bit	
5	TOU24	1	R	W	Timer TOU2_4 to TOU2_7 interrupts mask setting bit	

Bit	Abbreviation	After Reset	R	W	Description	
4	TOU20	1	R	W	Timer TOU2_0 to TOU2_3 interrupts mask setting bit	Masks interrupts for each on-chip peripheral module. Writing to this bit is invalid. 0: No mask setting 1: Mask setting
3	TOU14	1	R	W	Timer TOU1_4 to TOU1_7 interrupts mask setting bit	
2	TOU10	1	R	W	Timer TOU1_0 to TOU1_3 interrupts mask setting bit	
1	TOU04	1	R	W	Timer TOU0_4 to TOU0_7 interrupts mask setting bit	
0	TOU00	1	R	W	Timer TOU0_0 to TOU0_3 interrupts mask setting bit	

15.3.16 Interrupt Mask Clear Register 0 (INT2MSKCR)

The INT2MSKCR register can clear any masking set in the INT2MSKR register. Setting bits in this register to "1" clears the masking of the corresponding interrupt sources. Reading bits in this register is always "0".

Interrupt Mask Clear Register 0 (INT2MSKCR) <P4 address: location H'FFFF F43C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMIG5	CMIG4	CMIG3	CMIG2	CMIG1	CMIG0	TF	TA	ADC	IICI	DRO	DRI2	DRI1	DRI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HUDI	RSPI2	RSPI1	RSPI0	SCIF3	SCIF2	SCIF1	SCIF0	DMAC 6T11	DMAC 4T5	DMAC 0T3	WDT	—	TUNI2	TUNI1	TUNI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31, 30	—	All 0	0	0	Reserved Bits The write value should always be "0".	Each of these bits clears interrupt masking for the corresponding on-chip peripheral module. These bits are always read as “0”. Writing “0” to these bits has no effect. 0: Mask setting not cleared 1: Mask setting cleared
29	CMIG5	0	0	W	Timer G5 interrupt mask clear setting bit	
28	CMIG4	0	0	W	Timer G4 interrupt mask clear setting bit	
27	CMIG3	0	0	W	Timer G3 interrupt mask clear setting bit	
26	CMIG2	0	0	W	Timer G2 interrupt mask clear setting bit	
25	CMIG1	0	0	W	Timer G1 interrupt mask clear setting bit	
24	CMIG0	0	0	W	Timer G0 interrupt mask clear setting bit	
23	TF	0	0	W	Timer F interrupt mask clear setting bit	
22	TA	0	0	W	Timer A interrupt mask clear setting bit	
21	ADC	0	0	W	ADC interrupt mask clear setting bit	
20	IICI	0	0	W	IIC3 interrupt mask clear setting bit	
19	DRO	0	0	W	DRO interrupt mask clear setting bit	
18	DRI2	0	0	W	DRI2 interrupt mask clear setting bit	
17	DRI1	0	0	W	DRI1 interrupt mask clear setting bit	
16	DRI0	0	0	W	DRI0 interrupt mask clear setting bit	
15	HUDI	0	0	W	H-UDI interrupt mask clear setting bit	
14	RSPI2	0	0	W	RSPI2 interrupt mask clear setting bit	
13	RSPI1	0	0	W	RSPI1 interrupt mask clear setting bit	
12	RSPI0	0	0	W	RSPI0 interrupt mask clear setting bit	
11	SCIF3	0	0	W	SCIF3 interrupt mask clear setting bit	
10	SCIF2	0	0	W	SCIF2 interrupt mask clear setting bit	
9	SCIF1	0	0	W	SCIF1 interrupt mask clear setting bit	
8	SCIF0	0	0	W	SCIF0 interrupt mask clear setting bit	
7	DMAC6T11	0	0	W	DMA6 to DMA11 interrupt mask clear setting bit	
6	DMAC4T5	0	0	W	DMA4 to DMA4 interrupt mask clear setting bit	
5	DMAC0T3	0	0	W	DMA0 to DMA3 interrupt mask clear setting bit	
4	WDT	0	0	W	WDT interrupt mask clear setting bit	
3	—	0	0	0	Reserved Bit The write value should always be "0".	
2	TUNI2	0	0	W	TMU2 interrupt mask clear setting bit	
1	TUNI1	0	0	W	TMU1 interrupt mask clear setting bit	
0	TUNI0	0	0	W	TMU0 interrupt mask clear setting bit	

15.3.17 Interrupt Mask Clear Register 1 (INT2MSKCR1)

The INT2MSKCR1 register can clear any masking set in the INT2MSKR1 register. Setting bits in this register to "1" clears the masking of the corresponding interrupt sources. Reading bits in this register is always "0".

Interrupt Mask Clear Register 1 (INT2MSKCR1) <P4 address: location H'FFFF F4D4>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	PDAC	FRTINT	FRINT
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CAN3	CAN2	CAN1	CAN0	TOU44	TOU40	TOU34	TOU30	TOU24	TOU20	TOU14	TOU10	TOU04	TOU00
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 19	—	All 0	0	0	Reserved Bits The write value should always be "0".	Each of these bits clears interrupt masking for the corresponding on-chip peripheral module. These bits are always read as “0”. Writing “0” to these bits has no effect. 0: Mask setting not cleared 1: Mask setting cleared
18	PDAC	0	0	W	PDAC interrupt mask clear setting bit	
17	FRTINT	0	0	W	FlexRay timer interrupt mask clear setting bit Reserved bit in the SH7456 Group. The write value should always be "0".	
16	FRINT	0	0	W	FlexRay interrupt mask clear setting bit Reserved bit in the SH7456 Group. The write value should always be "0".	
15, 14	—	All 0	0	0	Reserved Bits The write value should always be "0"	
13	CAN3	0	0	W	CAN3 interrupt mask clear setting bit	
12	CAN2	0	0	W	CAN2 interrupt mask clear setting bit	
11	CAN1	0	0	W	CAN1 interrupt mask clear setting bit	
10	CAN0	0	0	W	CAN0 interrupt mask clear setting bit	
9	TOU44	0	0	W	Timer TOU4_4 to TOU4_7 interrupt mask clear setting bit	
8	TOU40	0	0	W	Timer TOU4_0 to TOU4_3 interrupt mask clear setting bit	
7	TOU34	0	0	W	Timer TOU3_4 to TOU3_7 interrupt mask clear setting bit	
6	TOU30	0	0	W	Timer TOU3_0 to TOU3_3 interrupt mask clear setting bit	
5	TOU24	0	0	W	Timer TOU2_4 to TOU2_7 interrupt mask clear setting bit	
4	TOU20	0	0	W	Timer TOU2_0 to TOU2_3 interrupt mask clear setting bit	
3	TOU14	0	0	W	Timer TOU1_4 to TOU1_7 interrupt mask clear setting bit	

Bit	Abbreviation	After Reset	R	W	Description	
2	TOU10	0	0	W	Timer TOU1_0 to TOU1_3 interrupt mask clear setting bit	Each of these bits clears interrupt masking for the corresponding on-chip peripheral module. These bits are always read as "0". Writing "0" to these bits has no effect. 0: Mask setting not cleared 1: Mask setting cleared
1	TOU04	0	0	W	Timer TOU0_4 to TOU0_7 interrupt mask clear setting bit	
0	TOU00	0	0	W	Timer TOU0_0 to TOU0_3 interrupt mask clear setting bit	

15.3.18 Per-Module Interrupt Source Registers 0 to 12 (INT2B0 to INT2B12)

In contrast to registers INT2A00, INT2A01, INT2A10, and INT2A11, which indicate interrupt sources by on-chip peripheral module, registers INT2B0 to INT2B12 indicate specific individual interrupt sources within their corresponding modules. Registers INT2B0 to INT2B12 are not affected by the mask settings of registers INT2MSKR and INT2MSKR1. To mask these specific individual interrupt sources, it is necessary to make masking settings in the corresponding on-chip peripheral module.

(1) INT2B0 register: TMU module interrupt detailed sources

Per-module interrupt source register 0 (INT2B0) <P4 address: location H'FFFF F440>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TUNI2	TUNI1	TUNI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 3	—	All 0	R	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding TMU interrupt source. The indications provided by this register are not cleared even if the TMU module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
2	TUNI2	0	R	—	TMU2 underflow interrupt	
1	TUNI1	0	R	—	TMU1 underflow interrupt	
0	TUNI0	0	R	—	TMU0 underflow interrupt	

(2) INT2B1 register: DMAC module detailed sources

Per-module interrupt source register 1 (INT2B1) <P4 address: location H'FFFF F444>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DMINT 11	DMINT 10	DMINT 9	DMINT 8	DMINT 7	DMINT 6	DMINT 5	DMINT 4	DMAE 1	DMAE 0	DMINT 3	DMINT 2	DMINT 1	DMINT 0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 14	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding DMA interrupt source. The indications provided by this register are not cleared even if the DMAC module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
13	DMINT11	0	R	—	DMA11 transfer complete interrupt	
12	DMINT10	0	R	—	DMA10 transfer complete interrupt	
11	DMINT9	0	R	—	DMA9 transfer complete/half end interrupt	
10	DMINT8	0	R	—	DMA8 transfer complete/half end interrupt	
9	DMINT7	0	R	—	DMA7 transfer complete/half end interrupt	
8	DMINT6	0	R	—	DMA6 transfer complete/half end interrupt	
7	DMINT5	0	R	—	DMA5 transfer complete interrupt	
6	DMINT4	0	R	—	DMA4 transfer complete interrupt	
5	DMAE1	0	R	—	DMAC1 (DMA6 to DMA11) address error interrupt	
4	DMAE0	0	R	—	DMAC0 (DMA0 to DMA5) address error interrupt	
3	DMINT3	0	R	—	DMA3 transfer complete/half end interrupt	
2	DMINT2	0	R	—	DMA2 transfer complete/half end interrupt	
1	DMINT1	0	R	—	DMA1 transfer complete/half end interrupt	
0	DMINT0	0	R	—	DMA0 transfer complete/half end interrupt	

(3) INT2B2 register: ATU-IIIS module detailed sources

Per-module interrupt source register 2 (INT2B2) <P4 address: location H'FFFF F448>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMIG5	CMIG4	CMIG3	CMIG2	CMIG1	CMIG0	—	OVIF2	OVIF1	OVIF0	—	ICIF2	ICIF1	ICIF0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	OVIA1	ICIA15	ICIA14	—	ICIA12	ICIA11	ICIA10	—	OVIA0	—	ICIA04	ICIA03	ICIA02	ICIA01	ICIA00
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31, 30	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding ATU-IIIS interrupt source. The indications provided by this register are not cleared even if the ATU-IIIS module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
29	CMIG5	0	R	—	Timer G5 compare-match interrupt	
28	CMIG4	0	R	—	Timer G4 compare-match interrupt	
27	CMIG3	0	R	—	Timer G3 compare-match interrupt	
26	CMIG2	0	R	—	Timer G2 compare-match interrupt	
25	CMIG1	0	R	—	Timer G1 compare-match interrupt	
24	CMIG0	0	R	—	Timer G0 compare-match interrupt	
23	—	0	0	—	Reserved Bit This bit is always read as "0".	
22	OVIF2	0	R	—	Timer F2 overflow interrupt	
21	OVIF1	0	R	—	Timer F1 overflow interrupt	
20	OVIF0	0	R	—	Timer F0 overflow interrupt	
19	—	0	0	—	Reserved Bit This bit is always read as "0".	
18	ICIF2	0	R	—	Timer F2 input capture interrupt	
17	ICIF1	0	R	—	Timer F1 input capture interrupt	
16	ICIF0	0	R	—	Timer F0 input capture interrupt	
15	—	0	0	—	Reserved Bit This bit is always read as "0".	
14	OVIA1	0	R	—	Timer A1 overflow interrupt	
13	ICIA15	0	R	—	Timer A1 channel 5 capture interrupt	
12	ICIA14	0	R	—	Timer A1 channel 4 capture interrupt	
11	—	0	0	—	Reserved Bit This bit is always read as "0".	
10	ICIA12	0	R	—	Timer A1 channel 2 capture interrupt	
9	ICIA11	0	R	—	Timer A1 channel 1 capture interrupt	
8	ICIA10	0	R	—	Timer A1 channel 0 capture interrupt	

Bit	Abbreviation	After Reset	R	W	Description	
7	—	0	0	—	Reserved Bit This bit is always read as "0".	These bits indicate the state of the corresponding ATU-IIIS interrupt source. The indications provided by this register are not cleared even if the ATU-IIIS module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
6	OVIA0	0	R	—	Timer A0 overflow interrupt	
5	—	0	0	—	Reserved Bit This bit is always read as "0".	
4	ICIA04	0	R	—	Timer A0 channel 4 capture interrupt	
3	ICIA03	0	R	—	Timer A0 channel 3 capture interrupt	
2	ICIA02	0	R	—	Timer A0 channel 2 capture interrupt	
1	ICIA01	0	R	—	Timer A0 channel 1 capture interrupt	
0	ICIA00	0	R	—	Timer A0 channel 0 capture interrupt	

(4) INT2B3 register: ATU-IIIS module detailed sources

Per-module interrupt source register 3 (INT2B3) <P4 address: location H'FFFF F44C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TOU27 UDF	TOU26 UDF	TOU25 UDF	TOU24 UDF	TOU23 UDF	TOU22 UDF	TOU21 UDF	TOU20 UDF
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TOU17 UDF	TOU16 UDF	TOU15 UDF	TOU14 UDF	TOU13 UDF	TOU12 UDF	TOU11 UDF	TOU10 UDF	TOU7 UDF	TOU6 UDF	TOU5 UDF	TOU4 UDF	TOU3 UDF	TOU2 UDF	TOU1 UDF	TOU0 UDF
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 24	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding ATU-IIIS interrupt source. The indications provided by this register are not cleared even if the ATU-IIIS module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
23	TOU27UDF	0	R	—	Timer TOU2_7 counter underflow interrupt	
22	TOU26UDF	0	R	—	Timer TOU2_6 counter underflow interrupt	
21	TOU25UDF	0	R	—	Timer TOU2_5 counter underflow interrupt	
20	TOU24UDF	0	R	—	Timer TOU2_4 counter underflow interrupt	
19	TOU23UDF	0	R	—	Timer TOU2_3 counter underflow interrupt	
18	TOU22UDF	0	R	—	Timer TOU2_2 counter underflow interrupt	
17	TOU21UDF	0	R	—	Timer TOU2_1 counter underflow interrupt	
16	TOU20UDF	0	R	—	Timer TOU2_0 counter underflow interrupt	
15	TOU17UDF	0	R	—	Timer TOU1_7 counter underflow interrupt	

Bit	Abbreviation	After Reset	R	W	Description	
14	TOU16UDF	0	R	—	Timer TOU1_6 counter underflow interrupt	These bits indicate the state of the corresponding ATU-IIIS interrupt source. The indications provided by this register are not cleared even if the ATU-IIIS module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
13	TOU15UDF	0	R	—	Timer TOU1_5 counter underflow interrupt	
12	TOU14UDF	0	R	—	Timer TOU1_4 counter underflow interrupt	
11	TOU13UDF	0	R	—	Timer TOU1_3 counter underflow interrupt	
10	TOU12UDF	0	R	—	Timer TOU1_2 counter underflow interrupt	
9	TOU11UDF	0	R	—	Timer TOU1_1 counter underflow interrupt	
8	TOU10UDF	0	R	—	Timer TOU1_0 counter underflow interrupt	
7	TOU07UDF	0	R	—	Timer TOU0_7 counter underflow interrupt	
6	TOU06UDF	0	R	—	Timer TOU0_6 counter underflow interrupt	
5	TOU05UDF	0	R	—	Timer TOU0_5 counter underflow interrupt	
4	TOU04UDF	0	R	—	Timer TOU0_4 counter underflow interrupt	
3	TOU03UDF	0	R	—	Timer TOU0_3 counter underflow interrupt	
2	TOU02UDF	0	R	—	Timer TOU0_2 counter underflow interrupt	
1	TOU01UDF	0	R	—	Timer TOU0_1 counter underflow interrupt	
0	TOU00UDF	0	R	—	Timer TOU0_0 counter underflow interrupt	

(5) INT2B4 register: ATU-IIIS module detailed sources

Per-module interrupt source register 4 (INT2B4) <P4 address: location H'FFFF F450>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TOU47 UDF	TOU46 UDF	TOU45 UDF	TOU44 UDF	TOU43 UDF	TOU42 UDF	TOU41 UDF	TOU40 UDF	TOU37 UDF	TOU36 UDF	TOU35 UDF	TOU34 UDF	TOU33 UDF	TOU32 UDF	TOU31 UDF	TOU30 UDF
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 16	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding ATU-IIIS interrupt source. The indications provided by this register are not cleared even if the ATU-IIIS module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
15	TOU47UDF	0	R	—	Timer TOU4_7 counter underflow interrupt	
14	TOU46UDF	0	R	—	Timer TOU4_6 counter underflow interrupt	
13	TOU45UDF	0	R	—	Timer TOU4_5 counter underflow interrupt	
12	TOU44UDF	0	R	—	Timer TOU4_4 counter underflow interrupt	
11	TOU43UDF	0	R	—	Timer TOU4_3 counter underflow interrupt	
10	TOU42UDF	0	R	—	Timer TOU4_2 counter underflow interrupt	
9	TOU41UDF	0	R	—	Timer TOU4_1 counter underflow interrupt	
8	TOU40UDF	0	R	—	Timer TOU4_0 counter underflow interrupt	
7	TOU37UDF	0	R	—	Timer TOU3_7 counter underflow interrupt	
6	TOU36UDF	0	R	—	Timer TOU3_6 counter underflow interrupt	
5	TOU35UDF	0	R	—	Timer TOU3_5 counter underflow interrupt	
4	TOU34UDF	0	R	—	Timer TOU3_4 counter underflow interrupt	
3	TOU33UDF	0	R	—	Timer TOU3_3 counter underflow interrupt	
2	TOU32UDF	0	R	—	Timer TOU3_2 counter underflow interrupt	
1	TOU31UDF	0	R	—	Timer TOU3_1 counter underflow interrupt	
0	TOU30UDF	0	R	—	Timer TOU3_0 counter underflow interrupt	

(6) INT2B5 register: SCIF module detailed sources

Per-module interrupt source register 5 (INT2B5) <P4 address: location H'FFFF F454>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXI3	BRI3	RXI3	ERI3	TXI2	BRI2	RXI2	ERI2	TXI1	BRI1	RXI1	ERI1	TXI0	BRI0	RXI0	ERI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 16	—	All 0	0	—	Reserved Bits These bits are always read as "0".	
15	TXI3	0	R	—	SCIF3 transmit FIFO data empty interrupt	These bits indicate the state of the corresponding SCIF interrupt source. The indications provided by this register are not cleared even if the SCIF module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
14	BRI3	0	R	—	SCIF3 break or overrun error interrupt	
13	RXI3	0	R	—	SCIF3 receive FIFO data full or receive data ready interrupt	
12	ERI3	0	R	—	SCIF3 receive error interrupt	
11	TXI2	0	R	—	SCIF2 transmit FIFO data empty interrupt	
10	BRI2	0	R	—	SCIF2 break or overrun error interrupt	
9	RXI2	0	R	—	SCIF2 receive FIFO data full or receive data ready interrupt	
8	ERI2	0	R	—	SCIF2 receive error interrupt	
7	TXI1	0	R	—	SCIF1 transmit FIFO data empty interrupt	
6	BRI1	0	R	—	SCIF1 break or overrun error interrupt	
5	RXI1	0	R	—	SCIF1 receive FIFO data full or receive data ready interrupt	
4	ERI1	0	R	—	SCIF1 receive error interrupt	
3	TXI0	0	R	—	SCIF0 transmit FIFO data empty interrupt	
2	BRI0	0	R	—	SCIF0 break or overrun error interrupt	
1	RXI0	0	R	—	SCIF0 receive FIFO data full or receive data ready interrupt	
0	ERI0	0	R	—	SCIF0 receive error interrupt	

(7) INT2B6 register: RSPI module detailed sources

Per-module interrupt source register 6 (INT2B6) <P4 address: location H'FFFF F458>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SPTI2	SPRI2	SPEI2	SPTI1	SPRI1	SPEI1	SPTI0	SPRI0	SPEI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 9	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding RSPI interrupt source. The indications provided by this register are not cleared even if the RSPI module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
8	SPTI2	0	R	—	RSPI2 transmit interrupt	
7	SPRI2	0	R	—	RSPI2 receive interrupt	
6	SPEI2	0	R	—	RSPI2 error interrupt	
5	SPTI1	0	R	—	RSPI1 transmit interrupt	
4	SPRI1	0	R	—	RSPI1 receive interrupt	
3	SPEI1	0	R	—	RSPI1 error interrupt	
2	SPTI0	0	R	—	RSPI0 transmit interrupt	
1	SPRI0	0	R	—	RSPI0 receive interrupt	
0	SPEI0	0	R	—	RSPI0 error interrupt	

(8) INT2B7 register: DRI module detailed sources

Per-module interrupt source register 7 (INT2B7) <P4 address: location H'FFFF F45C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DRI2 TRM	DRI2 DEC	DRI2 EVENT	DRI1 TRM	DRI1 DEC	DRI1 EVENT	DRI0 TRM	DRI0 DEC	DRI0 EVENT
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 9	—	All 0	0	—	Reserved Bits These bits are always read as "0".	
8	DRI2TRM	0	R	—	DRI2 transfer interrupt	These bits indicate the state of the corresponding DRI interrupt source. The indications provided by this register are not cleared even if the DRI module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
7	DRI2DEC	0	R	—	DRI2 counter interrupt	
6	DRI2EVENT	0	R	—	DRI2 event detection interrupt	
5	DRI1TRM	0	R	—	DRI1 transfer interrupt	
4	DRI1DEC	0	R	—	DRI1 counter interrupt	
3	DRI1EVENT	0	R	—	DRI1 event detection interrupt	
2	DRI0TRM	0	R	—	DRI0 transfer interrupt	
1	DRI0DEC	0	R	—	DRI0 counter interrupt	
0	DRI0EVENT	0	R	—	DRI0 event detection interrupt	

(9) INT2B8 register: CAN module detailed sources

Per-module interrupt source register 8 (INT2B8) <P4 address: location H'FFFF F460>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	TXM3	RXM 13	RXM 03	—	TXF3	RXF3	ERS3	—	TXM2	RXM 12	RXM 02	—	TXF2	RXF2	ERS2
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TXM1	RXM 11	RXM 01	—	TXF1	RXF1	ERS1	—	TXM0	RXM 10	RXM 00	—	TXF0	RXF0	ERS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31	—	0	0	—	Reserved Bit This bit is always read as "0".	These bits indicate the state of the corresponding CAN interrupt source. The indications provided by this register are not cleared even if the CAN module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
30	TXM3	0	R	—	CAN3 mailbox 0 to 63 transmit interrupt	
29	RXM13	0	R	—	CAN3 mailbox 1 to 63 receive interrupt	
28	RXM03	0	R	—	CAN3 mailbox 0 receive interrupt	
27	—	0	0	—	Reserved Bit This bit is always read as "0".	
26	TXF3	0	R	—	CAN3 transmit FIFO interrupt	
25	RXF3	0	R	—	CAN3 receive FIFO interrupt	
24	ERS3	0	R	—	CAN3 error interrupt	
23	—	0	0	—	Reserved Bit This bit is always read as "0".	
22	TXM2	0	R	—	CAN2 mailbox 0 to 63 transmit interrupt	
21	RXM12	0	R	—	CAN2 mailbox 1 to 63 receive interrupt	
20	RXM02	0	R	—	CAN2 mailbox 0 receive interrupt	
19	—	0	0	—	Reserved Bit This bit is always read as "0".	
18	TXF2	0	R	—	CAN2 transmit FIFO interrupt	
17	RXF2	0	R	—	CAN2 receive FIFO interrupt	
16	ERS2	0	R	—	CAN2 error interrupt	
15	—	0	0	—	Reserved Bit This bit is always read as "0".	
14	TXM1	0	R	—	CAN1 mailbox 0 to 63 transmit interrupt	
13	RXM11	0	R	—	CAN1 mailbox 1 to 63 receive interrupt	
12	RXM01	0	R	—	CAN1 mailbox 0 receive interrupt	
11	—	0	0	—	Reserved Bit This bit is always read as "0".	

Bit	Abbreviation	After Reset	R	W	Description	
10	TXF1	0	R	—	CAN1 transmit FIFO interrupt	These bits indicate the state of the corresponding CAN interrupt source. The indications provided by this register are not cleared even if the CAN module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
9	RXF1	0	R	—	CAN1 receive FIFO interrupt	
8	ERS1	0	R	—	CAN1 error interrupt	
7	—	0	0	—	Reserved Bit This bit is always read as "0".	
6	TXM0	0	R	—	CAN0 mailbox 0 to 63 transmit interrupt	
5	RXM10	0	R	—	CAN0 mailbox 1 to 63 receive interrupt	
4	RXM00	0	R	—	CAN0 mailbox 0 receive interrupt	
3	—	0	0	—	Reserved Bit This bit is always read as "0".	
2	TXF0	0	R	—	CAN0 transmit FIFO interrupt	
1	RXF0	0	R	—	CAN0 receive FIFO interrupt	
0	ERS0	0	R	—	CAN0 error interrupt	

(10) INT2B10 register: ADC module detailed sources

Per-module interrupt source register 10 (INT2B10) <P4 address: location H'FFFF F468>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	AD1 ID5	AD1 ID4	—	—	AD1 ID1	AD1 ID0	AD0 ID15	AD0 ID14
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD0 ID13	AD0 ID12	AD0 ID11	AD0 ID10	AD0 ID9	AD0 ID8	—	AD0 ID6	—	AD0 ID4	—	AD0 ID2	—	AD0 ID0	AD1I	AD0I
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 24	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding ADC interrupt source. The indications provided by this register are not cleared even if the ADC module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
23	AD1ID5	0	R	—	AD1ID5 interrupt conversion complete interrupt	
22	AD1ID4	0	R	—	AD1ID4 interrupt conversion complete interrupt	
21, 20	—	All 0	0	—	Reserved Bits These bits are always read as "0".	
19	AD1ID1	0	R	—	AD1ID1 interrupt conversion complete interrupt	
18	AD1ID0	0	R	—	AD1ID0 interrupt conversion complete interrupt	
17	AD0ID15	0	R	—	AD0ID15 interrupt conversion complete interrupt	
16	AD0ID14	0	R	—	AD0ID14 interrupt conversion complete interrupt	
15	AD0ID13	0	R	—	AD0ID13 interrupt conversion complete interrupt	
14	AD0ID12	0	R	—	AD0ID12 interrupt conversion complete interrupt	
13	AD0ID11	0	R	—	AD0ID11 interrupt conversion complete interrupt	
12	AD0ID10	0	R	—	AD0ID10 interrupt conversion complete interrupt	
11	AD0ID9	0	R	—	AD0ID9 interrupt conversion complete interrupt	
10	AD0ID8	0	R	—	AD0ID8 interrupt conversion complete interrupt	
9	—	0	0	—	Reserved Bit This bit is always read as "0".	
8	AD0ID6	0	R	—	AD0ID6 interrupt conversion complete interrupt	

Bit	Abbreviation	After Reset	R	W	Description	
7	—	0	0	—	Reserved Bit This bit is always read as "0".	These bits indicate the state of the corresponding ADC interrupt source. The indications provided by this register are not cleared even if the ADC module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
6	AD0ID4	0	R	—	AD0ID4 interrupt conversion complete interrupt	
5	—	0	0	—	Reserved Bit This bit is always read as "0".	
4	AD0ID2	0	R	—	AD0ID2 interrupt conversion complete interrupt	
3	—	0	0	—	Reserved Bit This bit is always read as "0".	
2	AD0ID0	0	R	—	AD0ID0 interrupt conversion complete interrupt	
1	AD1I	0	R	—	AD1 scan conversion complete interrupt	
0	AD0I	0	R	—	AD0 scan conversion complete interrupt	

(11) INT2B11 register: FlexRay module detailed sources

The SH7456 Group has no FlexRay module. In the SH7456 Group, the bits in the INT2B11 register are reserved. These bits are always read as "0". Writing to these bits is invalid.

Per-module interrupt source register 11 (INT2B11) <P4 address: location H'FFFF F46C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	FRTINT1	FRTINT0	FRINT1	FRINT0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 4	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding FlexRay interrupt source. The indications provided by this register are not cleared even if the FlexRay module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
3	FRTINT1	0	R	—	FlexRay timer 1 interrupt	
2	FRTINT0	0	R	—	FlexRay timer 0 interrupt	
1	FRINT1	0	R	—	FlexRay interrupt 1	
0	FRINT0	0	R	—	FlexRay interrupt 0	

(12) INT2B12 register: PDAC module detailed sources

Per-module interrupt source register 12 (INT2B12) <P4 address: location H'FFFF F494>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PDINT
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description	
31 to 1	—	All 0	0	—	Reserved Bits These bits are always read as "0".	These bits indicate the state of the corresponding PDAC interrupt source. The indications provided by this register are not cleared even if the PDAC module is masked with the interrupt mask register. 0: No interrupts 1: Interrupts are generated
0	PDINT	0	R	—	Final modulation complete interrupt	

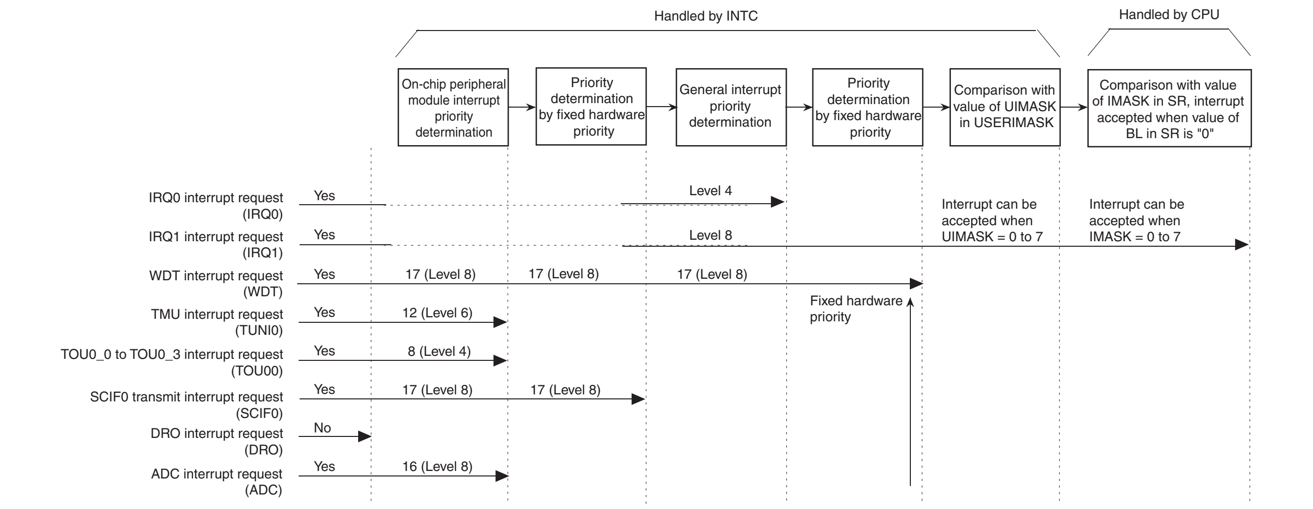
15.4 Operation

15.4.1 Interrupt Sources and Priorities

Interrupt sources fall into three categories: NMI, IRQ, and on-chip peripheral module interrupts. The priority of each interrupt source is indicated by its interrupt priority level (level 15 to level 0), with level 15 being the highest priority and level 1 the lowest. A priority level setting of level 0 causes the interrupt to be masked and associated interrupt requests to be ignored.

The vector address of the interrupt source (NMI, IRQ, and on-chip peripheral module interrupts) is fixed at the VBR register value + H'600. Individual sources are reflected in the interrupt event register (INTEVT), so the source can be identified by using the INTEVT register value as an offset when branching in the exception processing routine. When multiple interrupt sources are set to the same priority level, and more than one of these sources occurs simultaneously, processing is based on the hardware priority as shown in tables 15.2 and 15.3.

Figure 15.2 shows examples of priority determination at interrupt request acceptance, and table 15.8 lists priorities and the IMASK values at which they are accepted.


Handled by INTC
Handled by CPU
On-chip peripheral module interrupt priority determination
Priority determination by fixed hardware priority
General interrupt priority determination
Priority determination by fixed hardware priority
Comparison with value of UIMASK in USERIMASK
Comparison with value of IMASK in SR, interrupt accepted when value of BL in SR is "0"
IRQ0 interrupt request (IRQ0)
IRQ1 interrupt request (IRQ1)
WDT interrupt request (WDT)
TMU interrupt request (TUNI0)
TOU0_0 to TOU0_3 interrupt request (TOU00)
SCIF0 transmit interrupt request (SCIF0)
DRO interrupt request (DRO)
ADC interrupt request (ADC)
Yes
Level 4
Yes
Level 8
Yes
17 (Level 8)
17 (Level 8)
17 (Level 8)
Yes
12 (Level 6)
Yes
8 (Level 4)
Yes
17 (Level 8)
17 (Level 8)
No
Yes
16 (Level 8)
Interrupt can be accepted when UIMASK = 0 to 7
Interrupt can be accepted when IMASK = 0 to 7
Fixed hardware priority

Note: Interrupt priority levels shown in parentheses indicate priority levels that have been converted into four-bit values by discarding the lowest bit.

Figure 15.2 Priority Determination at Interrupt Request Acceptance

Table 15.8 Interrupt Priorities and IMASK Values at Which They Are Accepted

On-Chip Peripheral Module Interrupt Priority	Priority of General Interrupts	IMASK (UIMASK) Value at Which Interrupt Is Accepted
0, 1	Level 0	Not accepted (interrupt disabled state)
2, 3	Level 1	Accepted when IMASK (UIMASK) value is 0
4, 5	Level 2	Accepted when IMASK (UIMASK) value is 0 or 1
6, 7	Level 3	Accepted when IMASK (UIMASK) value is 0 to 2
8, 9	Level 4	Accepted when IMASK (UIMASK) value is 0 to 3
10, 11	Level 5	Accepted when IMASK (UIMASK) value is 0 to 4
12, 13	Level 6	Accepted when IMASK (UIMASK) value is 0 to 5
14, 15	Level 7	Accepted when IMASK (UIMASK) value is 0 to 6
16, 17	Level 8	Accepted when IMASK (UIMASK) value is 0 to 7
18, 19	Level 9	Accepted when IMASK (UIMASK) value is 0 to 8
20, 21	Level 10	Accepted when IMASK (UIMASK) value is 0 to 9
22, 23	Level 11	Accepted when IMASK (UIMASK) value is 0 to 10
24, 25	Level 12	Accepted when IMASK (UIMASK) value is 0 to 11
26, 27	Level 13	Accepted when IMASK (UIMASK) value is 0 to 12
28, 29	Level 14	Accepted when IMASK (UIMASK) value is 0 to 13
30, 31	Level 15	Accepted when IMASK (UIMASK) value is 0 to 14

(1) NMI Interrupt

The NMI interrupt has a higher priority than IRQ interrupts and on-chip peripheral module interrupts. When the NMIB bit in ICR0 is set to "1", an NMI interrupt is always accepted immediately by the CPU, regardless of the value of the BL bit in SR. When the value of the NMIB bit in ICR0 is "0", acceptance by the CPU of an NMI interrupt must wait until the BL bit in SR is cleared to "0".

Input from the NMI pin is edge-detected. The NMIE bit in the ICR0 register is used to select either the rising or the falling edge as the detection edge. When the NMIE bit in the ICR0 is modified, the NMI interrupts is not detected for a period of up to 6 Pck cycles after the value of the bit is changed.

By setting the MAI bit in ICR0 to "1", it is possible to mask all interrupts (NMI, IRQ, and on-chip peripheral module interrupts) for as long as input to the NMI pin is "L" level, regardless of the values of the BL bit and IMASK bits in SR. In this case only an NMI interrupt triggered by a change in the NMI pin state can be generated.

(2) IRQ Interrupts

The IRQnS bits (n = 0 to 2, 5 to 7) in ICR1 are use to set the detection mode to falling edge, rising edge, "L" level, or "H" level. Also, the interrupt priority is set in the INTPRI register.

When the IRQ interrupt request detection mode is "L" level or "H" level, the IRQ interrupt pin state should be maintained for the period from when the interrupt is accepted until the start of interrupt handling.

Note that after an IRQ interrupt request is detected, the interrupt source is maintained in the INTREQ register even if the IRQ interrupt pin state changes and the request is canceled before acceptance by the CPU. For details on the method of clearing IRQ interrupt requests when level detection has been selected, see section 15.7.2, To Clear IRQ Interrupt Requests When Level Detection is Selected.

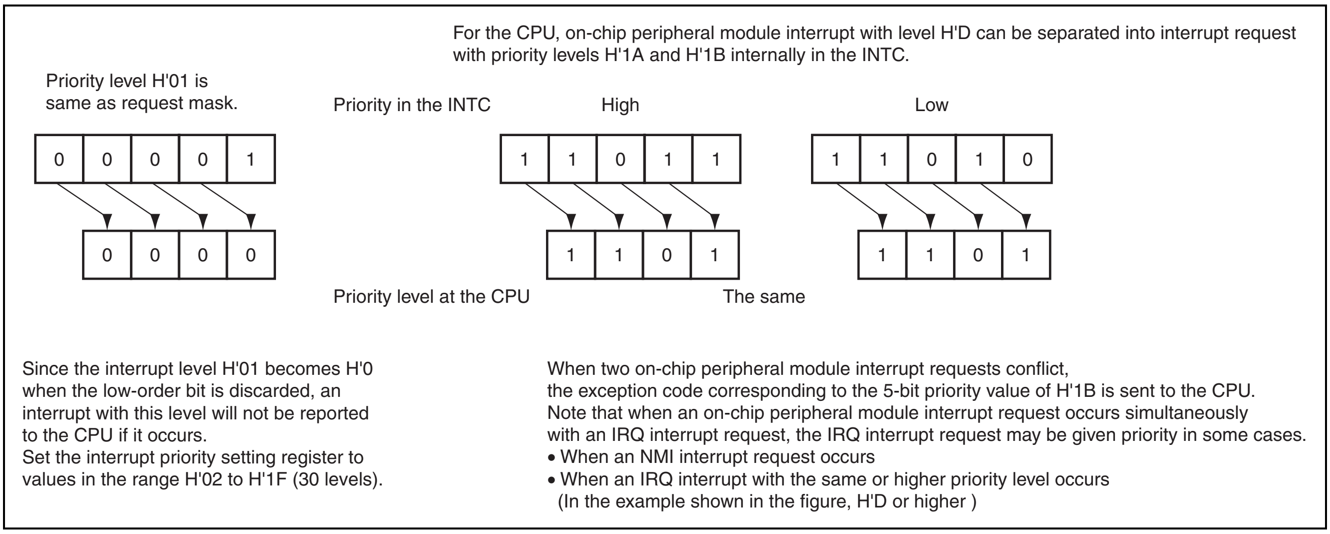
For details on the method of clearing IRQ interrupt requests when edge detection has been selected, see section 15.7.3, To Clear IRQ Interrupt Requests When Edge Detection is Selected.

When the INTMU bit in the CPU operation mode (CPUOPM) register is set to "1", the SR.IMASK bits are automatically modified to the level of the accepted interrupt. When the CPUOPM.INTMU bit is cleared to "0", the SR.IMASK bits are not affected by the accepted interrupt. For details on the CPUOPM register, see appendix A, CPU Operation Mode Register (CPUOPM).

(3) On-Chip Peripheral Module Interrupts

On-chip peripheral module interrupts are generated by the on-chip peripheral modules. When an on-chip peripheral module interrupt occurs, an exception code is output to the CPU by the interrupt request as a unique identifier of the interrupt source. When the CPU accepts the interrupt, the corresponding exception code is indicated in the INTEVT register of the CPU. This enables the interrupt handler to determine the source by reading the INTEVT register, and there is no need to read the source indication registers of the INTC. For the correspondences between the on-chip peripheral module interrupt sources and the exception codes, see table 15.3.

As shown in figure 15.1, each on-chip peripheral module interrupt source can be given one of 30 priority levels by assigning one of 32 5-bit values. (The higher the setting value, the higher the priority. Specifying a value of "H'00" or "H'01" has the same effect as masking interrupt requests for the corresponding source.) The CPU determines the interrupt priority on the basis of 15 levels expressed as 4-bit values (interrupt requests with a value of "H'0" are effectively masked), so the lowest bit of the on-chip peripheral module interrupt priority setting is discarded, converting it into a 4-bit value that is sent to the CPU. For example, interrupts for two sources with priority values of "H'1A" and "H'1B", respectively, will both have the same 4-bit output value of "H'1A". The two interrupts both have the same value, but when they are in contention, the exception code assigned to the one with the higher 5-bit priority setting of "H'1B" is sent to the CPU. When a contention occurs between two interrupts with the same priority setting, the exception code is sent to the CPU according to the listing of exception codes in table 15.3. Figure 15.3 shows the handling of priority for on-chip peripheral module interrupts.



For the CPU, on-chip peripheral module interrupt with level H'D can be separated into interrupt request with priority levels H'1A and H'1B internally in the INTC.

Since the interrupt level H'01 becomes H'0 when the low-order bit is discarded, an interrupt with this level will not be reported to the CPU if it occurs.
Set the interrupt priority setting register to values in the range H'02 to H'1F (30 levels).

When two on-chip peripheral module interrupt requests conflict, the exception code corresponding to the 5-bit priority value of H'1B is sent to the CPU.
Note that when an on-chip peripheral module interrupt request occurs simultaneously with an IRQ interrupt request, the IRQ interrupt request may be given priority in some cases.

- When an NMI interrupt request occurs
- When an IRQ interrupt with the same or higher priority level occurs
  (In the example shown in the figure, H'D or higher)

Figure 15.3 On-Chip Peripheral Module Interrupt Priority

15.4.2 General Interrupt Sequence

(1) General Interrupt Operation Sequence 1 (Clearing the Interrupt Request When the BL Bit in SR Has Been Set to "1")

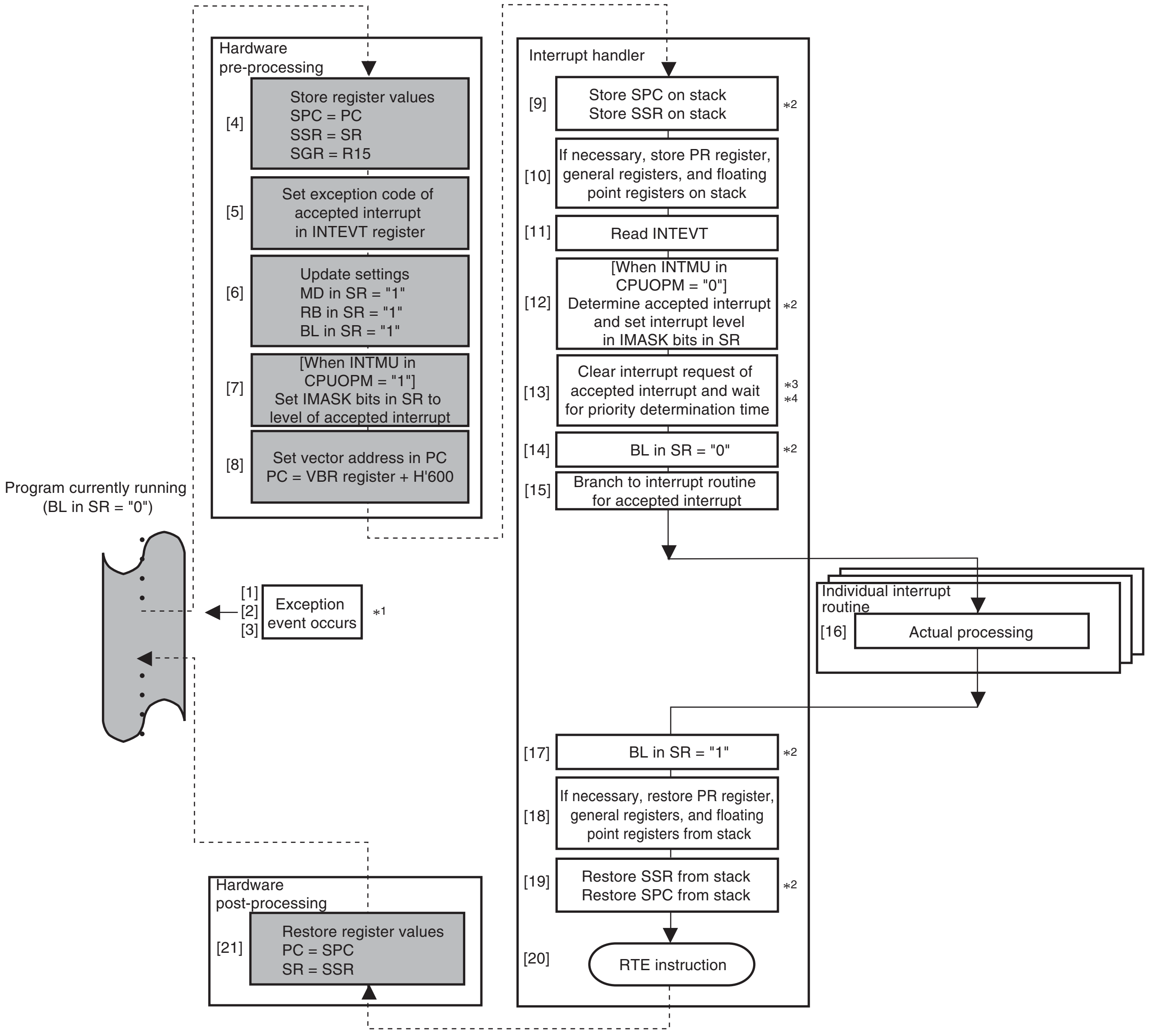
The sequence of operations when a general interrupt occurs for clearing the interrupt request when the value of the BL bit in SR is "1" is described below, considering the possibility of multiple interrupts occurring. When the interrupt handler is configured as described below and multiple interrupts occur, an interrupt with a higher priority level can be accepted immediately after step 14. This ensures a short interrupt response time for very urgent processing tasks. Steps 1 to 8 cover hardware pre-processing and step 21 covers hardware post-processing.

Note that steps 9, 12, 14, 17, and 19 may be omitted if multiple simultaneous interrupts will not be used.

1. Interrupt request sources send interrupt request signals to the INTC.
2. The INTC selects the highest-priority interrupt from the interrupt requests sent, according to the priority level settings in the INTPRI register and registers INT2PRI0 to INT2PRI12, and the lower-priority interrupts are held pending. If two interrupts have the same priority setting or if multiple interrupts occur within a single module, the interrupt with the highest priority according to tables 15.2 and 15.3 is selected.
3. The CPU compares the priority level of the interrupt selected by the INTC with the IMASK bits in SR. When the value of the BL bit in SR is "0", the CPU accepts the interrupt at the next gap between instructions only if its priority is higher than the level indicated by the IMASK bits in SR.
4. The contents of the status register (SR) and program counter (PC) are stored in the SSR register and SPC counter, respectively. The contents of R15 is saved to the SGR register at this time.
5. The exception code (interrupt source code) is set in the interrupt event register (INTEVT).
6. The BL, MD, and RB bits in SR are set to "1".
7. If the INTMU bit in CPUOPM has been set to "1", the IMASK bits in SR are automatically set to the level of the interrupt that was accepted. This ensures that interrupts with a lower priority level than the accepted interrupt will not be accepted. However, if the INTMU bit in CPUOPM has been cleared to "0", the IMASK bits in SR are not affected when an interrupt is accepted.
8. Execution jumps to the start address of the interrupt handler (the sum of the value set in the vector base register (VBR) and H'0000 0600).
9. The values of the SPC counter and SSR register are saved on the stack.
10. If necessary, the contents of the PR register, general registers, and floating point registers are saved on the stack.
11. The value of the INTEVT register is read to determine the source of the interrupt accepted by the CPU.
12. If the INTMU bit in CPUOPM has been cleared to "0", the IMASK bits in SR must be set to the level of the interrupt that was accepted by software. This ensures that interrupts with a lower priority level than the accepted interrupt will not be accepted. However, if the INTMU bit in CPUOPM has been set to "1", the IMASK bits in SR will have automatically been set to the level of the accepted interrupt in step 7, above.
13. The corresponding interrupt request is cleared within the interrupt routine for the accepted interrupt. After this, it is necessary to ensure that the priority determination time shown in table 15.9 elapses before the BL bit in SR is cleared (step 14, below). This will prevent the interrupt request that was supposed to be cleared from being reaccepted erroneously.
14. The BL bit in the SR register is cleared to "0".
15. The value of the INTEVT register read in step 11 is used as an offset to branch to the interrupt routine for the interrupt source.
16. The next lines of code specify the actual processing to be performed.
17. The BL bit in the SR register is set to "1".
18. The values of the PR register, general registers, and floating point registers saved in step 10 are restored from the stack.

19. The values of the SSR register and SPC counter are restored from the stack.
20. The RTE instruction is executed.
21. Execution of the RTE instruction causes the contents of the status register (SR) and program counter (PC) stored, respectively, in the SSR register and SPC counter in step 4 to be restored automatically.

Note: • The IRQ and on-chip peripheral module interrupts are initialized to the interrupt masked state by a hardware reset. To disable masking of an interrupt, write "0" to the corresponding bit in the INTMSKCLR, INT2MSKCR, or INT2MSKCR1 register.


Hardware pre-processing
[4] Store register values SPC = PC SSR = SR SGR = R15
[5] Set exception code of accepted interrupt in INTEVT register
[6] Update settings MD in SR = "1" RB in SR = "1" BL in SR = "1"
[7] [When INTMU in CPUOPM = "1"] Set IMASK bits in SR to level of accepted interrupt
[8] Set vector address in PC PC = VBR register + H'600
Program currently running (BL in SR = "0")
[1] [2] [3] Exception event occurs *1
Interrupt handler
[9] Store SPC on stack Store SSR on stack *2
[10] If necessary, store PR register, general registers, and floating point registers on stack
[11] Read INTEVT
[12] [When INTMU in CPUOPM = "0"] Determine accepted interrupt and set interrupt level in IMASK bits in SR *2
[13] Clear interrupt request of accepted interrupt and wait for priority determination time *3 *4
[14] BL in SR = "0" *2
[15] Branch to interrupt routine for accepted interrupt
Individual interrupt routine
[16] Actual processing
[17] BL in SR = "1" *2
[18] If necessary, restore PR register, general registers, and floating point registers from stack
[19] Restore SSR from stack Restore SPC from stack *2
[20] RTE instruction
Hardware post-processing
[21] Restore register values PC = SPC SR = SSR

Notes *1 The interrupt is accepted at the next gap between instructions when its priority as reported to the CPU by the INTC is higher than the level indicated by the IMASK bits in SR and the value of the BL bit in SR is "0".
*2 Necessary only if multiple interrupts may occur simultaneously.
*3 When level detection is selected for IRQ interrupts, the following steps are required before clearing the request:
a. Clear the request source of the IRQ level interrupt.
b. Wait (8 Pck) for the request of the IRQ level interrupt to clear.
*4 After the corresponding interrupt request is cleared, allow the priority determination time shown in table 15.9 to elapse before clearing the BL bit in SR. This will prevent the interrupt request that was supposed to be cleared from being reaccepted erroneously.

Figure 15.4 General Interrupt Operation Example 1

(2) General Interrupt Operation Sequence 2 (Clearing the Interrupt Request When the BL Bit in SR Has Been Cleared to "0")

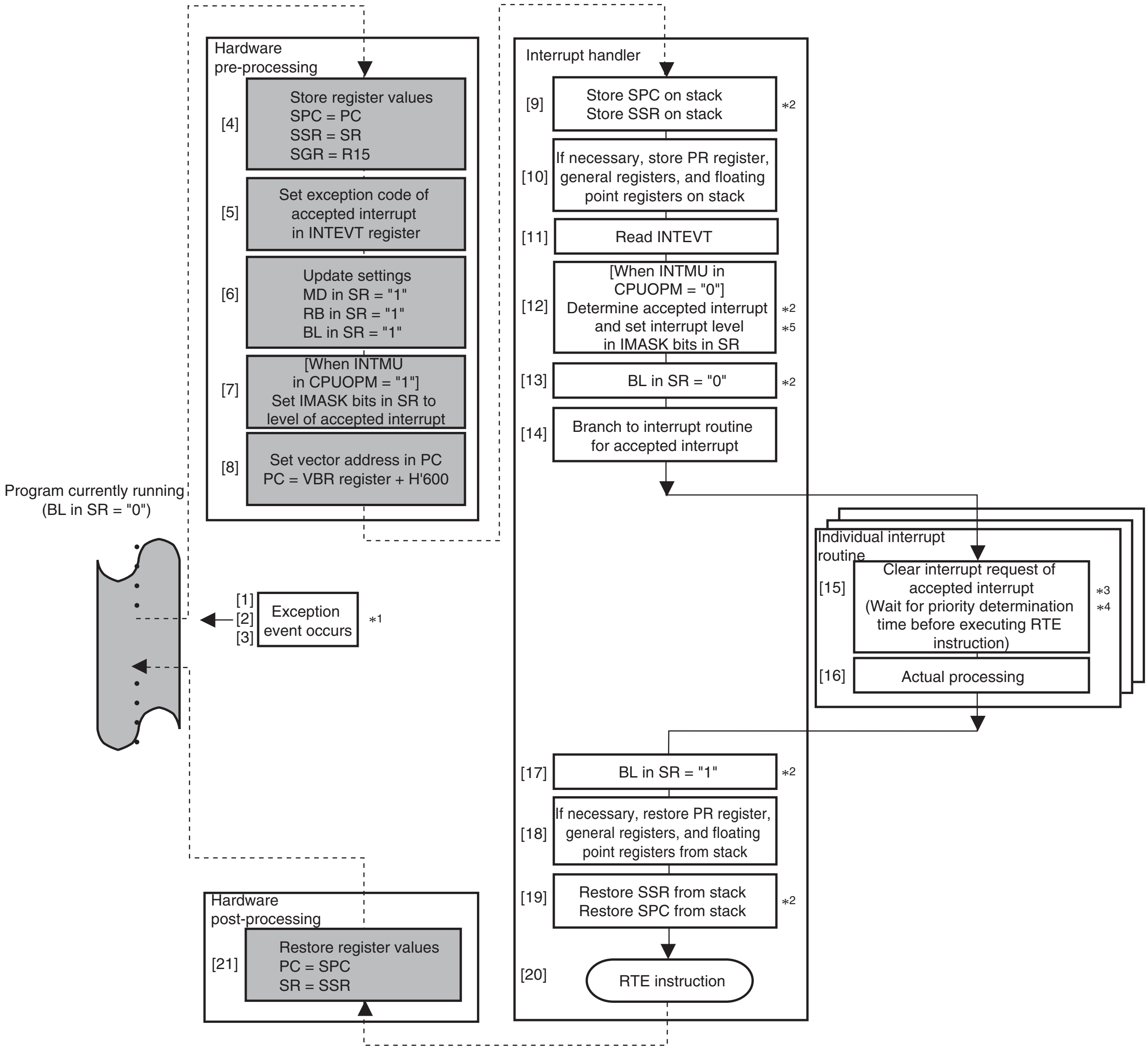
The sequence of operations when a general interrupt occurs for clearing the interrupt request when the value of the BL bit in SR is "0" is described below, considering the possibility of multiple interrupts occurring. When the interrupt handler is configured as described below and multiple interrupts occur, an interrupt with a higher priority level can be accepted immediately after step 13. This ensures a short interrupt response time for very urgent processing tasks. Steps 1 to 8 cover hardware pre-processing and step 21 covers hardware post-processing.

Note that steps 9, 12, 13, 17, and 19 may be omitted if multiple simultaneous interrupts will not be used.

1. Interrupt request sources send interrupt request signals to the INTC.
2. The INTC selects the highest-priority interrupt from the interrupt requests sent, according to the priority level settings in the INTPRI register and registers INT2PRI0 to INT2PRI12, and the lower-priority interrupts are held pending. If two interrupts have the same priority setting or if multiple interrupts occur within a single module, the interrupt with the highest priority according to tables 15.2 and 15.3 is selected.
3. The CPU compares the priority level of the interrupt selected by the INTC with the IMASK bits in SR. When the value of the BL bit in SR is "0", the CPU accepts the interrupt at the next gap between instructions only if its priority is higher than the level indicated by the IMASK bits in SR.
4. The contents of the status register (SR) and program counter (PC) are stored in the SSR register and SPC counter, respectively. The contents of R15 is saved to the SGR register at this time.
5. The exception code (interrupt source code) is set in the interrupt event register (INTEVT).
6. The block (BL), mode (MD), and register bank (RB) bits in the SR register are set to "1".
7. If the INTMU bit in CPUOPM has been set to "1", the IMASK bits in SR are automatically set to the level of the interrupt that was accepted. This ensures that interrupts with a lower priority level than the accepted interrupt will not be accepted. However, if the INTMU bit in CPUOPM has been cleared to "0", the IMASK bits in SR are not affected when an interrupt is accepted.
8. Execution jumps to the start address of the interrupt handler (the sum of the value set in the vector base register (VBR) and H'0000 0600).
9. The values of the SPC counter and SSR register are saved on the stack.
10. If necessary, the contents of the PR register, general registers, and floating point registers are saved on the stack.
11. The value of the INTEVT register is read to determine the source of the interrupt accepted by the CPU.
12. If the INTMU bit in CPUOPM has been cleared to "0", the IMASK bits in SR must be set equal to or the value higher than the level of the interrupt that was accepted by software. This ensures that interrupts with a lower priority level than the accepted interrupt will not be accepted. However, if the INTMU bit in CPUOPM has been set to "1", the IMASK bits in SR will have automatically been set to the level of the accepted interrupt in step 7, above.
13. The BL bit in the SR register is cleared to "0".
14. The value of the INTEVT register read in step 11 is used as an offset to branch to the interrupt routine for the interrupt source.
15. The corresponding interrupt request is cleared within the interrupt routine for the accepted interrupt. After this, it is necessary to ensure that the priority determination time shown in table 15.9 elapses before the RTE instruction is executed (step 20, below). This will prevent the interrupt request that was supposed to be cleared from being reaccepted erroneously.
16. The next lines of code specify the actual processing to be performed.
17. The BL bit in the SR register is set to "1".
18. The values of the PR register, general registers, and floating point registers saved in step 10 are restored from the stack.
19. The values of the SSR register and SPC counter are restored from the stack.
20. The RTE instruction is executed.

21. Execution of the RTE instruction causes the contents of the status register (SR) and program counter (PC) stored, respectively, in the SSR register and SPC counter in step 4 to be restored automatically.

Note: • The IRQ and on-chip peripheral module interrupts are initialized to the interrupt masked state by a hardware reset. To disable masking of an interrupt, write "0" to the corresponding bit in the INTMSKCLR, INT2MSKCR, or INT2MSKCR1 register.


Hardware pre-processing
[4] Store register values SPC = PC SSR = SR SGR = R15
[5] Set exception code of accepted interrupt in INTEVT register
[6] Update settings MD in SR = "1" RB in SR = "1" BL in SR = "1"
[7] [When INTMU in CPUOPM = "1"] Set IMASK bits in SR to level of accepted interrupt
[8] Set vector address in PC PC = VBR register + H'600
Program currently running (BL in SR = "0")
[1] [2] [3] Exception event occurs *1
Interrupt handler
[9] Store SPC on stack Store SSR on stack *2
[10] If necessary, store PR register, general registers, and floating point registers on stack
[11] Read INTEVT
[12] [When INTMU in CPUOPM = "0"] Determine accepted interrupt and set interrupt level in IMASK bits in SR *2 *5
[13] BL in SR = "0" *2
[14] Branch to interrupt routine for accepted interrupt
Individual interrupt routine
[15] Clear interrupt request of accepted interrupt (Wait for priority determination time before executing RTE instruction) *3 *4
[16] Actual processing
[17] BL in SR = "1" *2
[18] If necessary, restore PR register, general registers, and floating point registers from stack
[19] Restore SSR from stack Restore SPC from stack *2
[20] RTE instruction
Hardware post-processing
[21] Restore register values PC = SPC SR = SSR

Notes *1 The interrupt is accepted at the next gap between instructions when its priority as reported to the CPU by the INTC is higher than the level indicated by the IMASK bits in SR and the value of the BL bit in SR is "0".

*2 Necessary only if multiple interrupts may occur simultaneously.

*3 When level detection is selected for IRQ interrupts, the following steps are required before clearing the request:
a. Clear the request source of the IRQ level interrupt.
b. Wait (8 Pck) for the request of the IRQ level interrupt to clear.

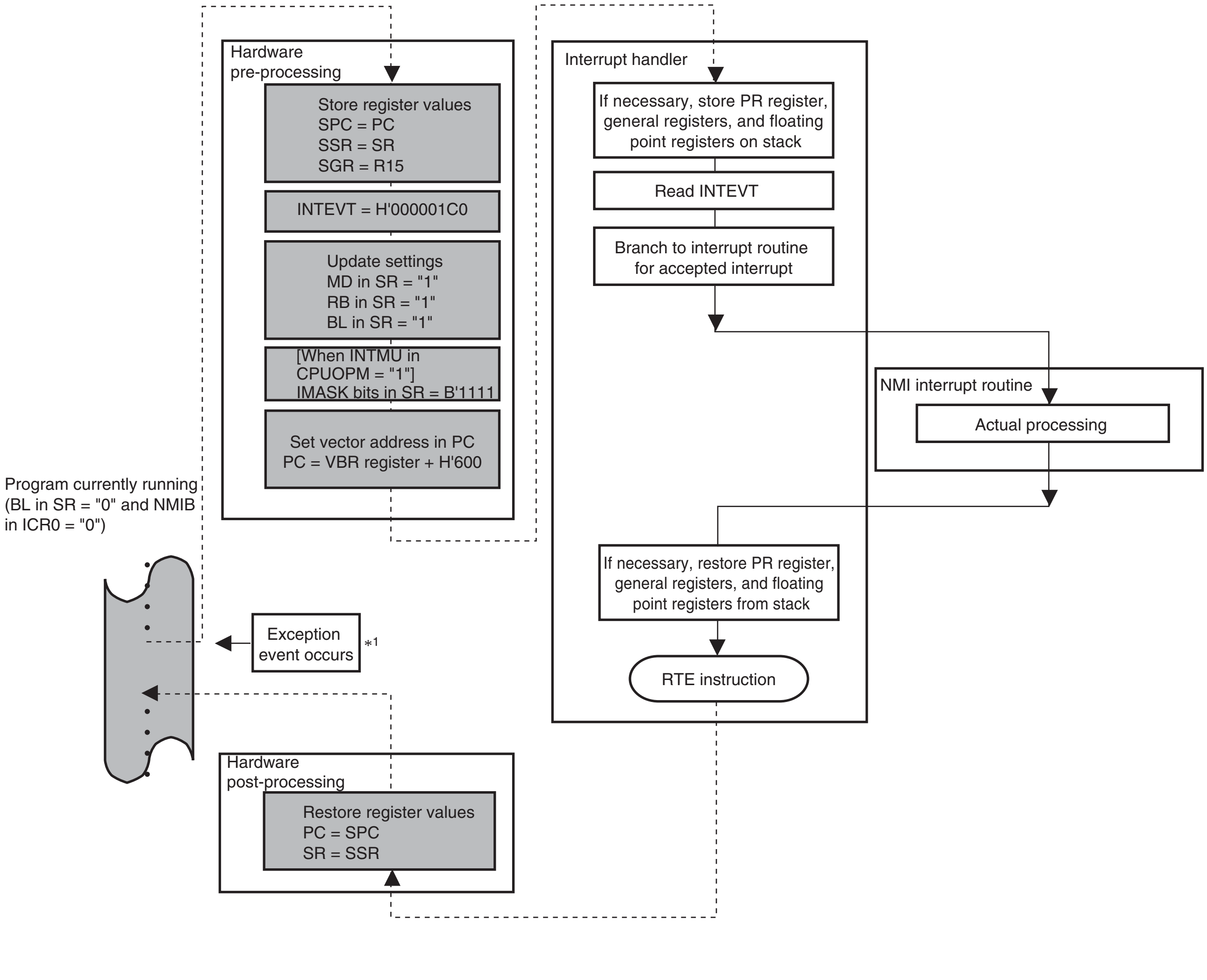
*4 After the corresponding interrupt request is cleared, allow the priority determination time shown in table 15.9 to elapse before executing the RTE instruction. This will prevent the interrupt request that was supposed to be cleared from being reaccepted erroneously.

*5 Set the value equal to or more than the accepted interrupt level to the IMASK bits in SR by software.

Figure 15.5 General Interrupt Operation Example 2

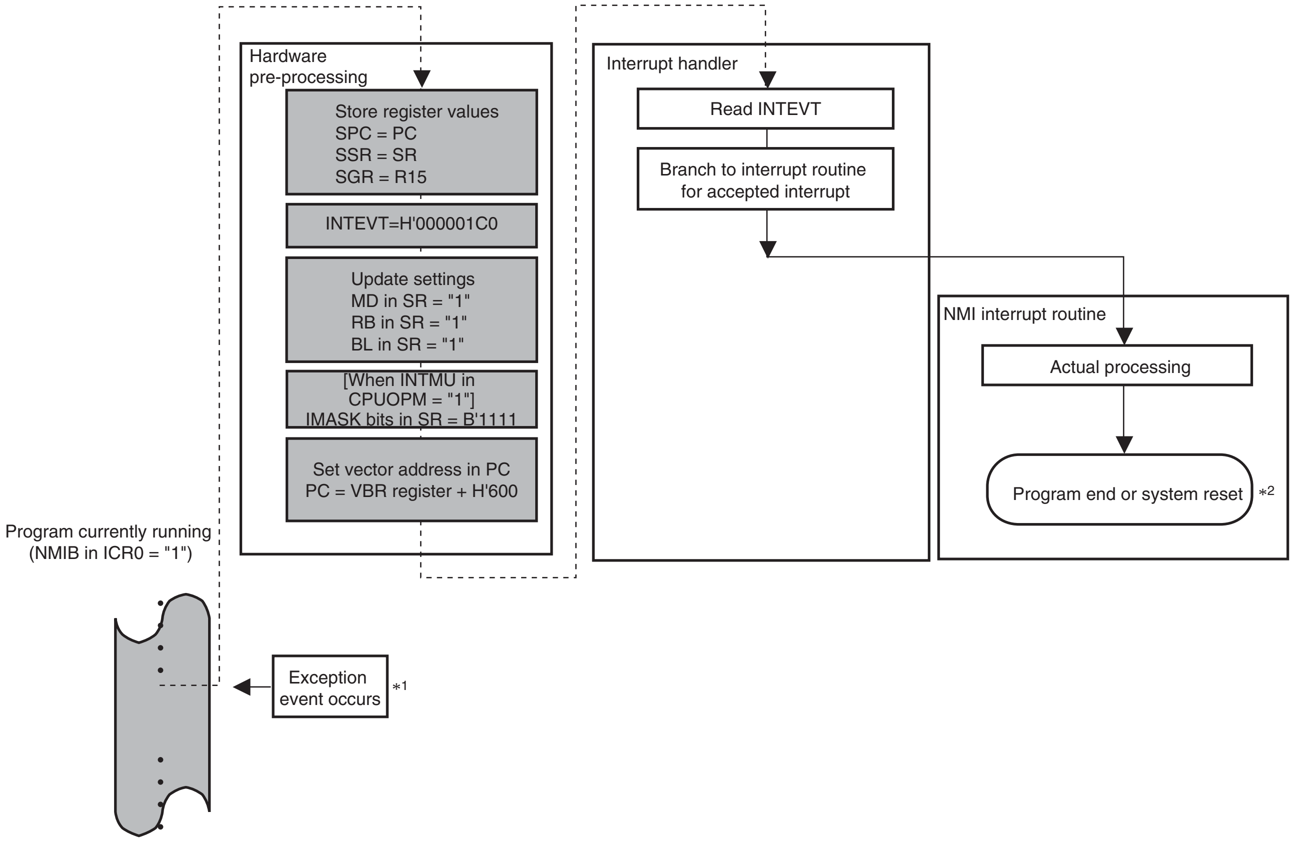
15.4.3 NMI Interrupt Operation Sequence

NMI is an emergency interrupt request used in cases such as when a malfunction is detected in the power supply or an external watchdog timer. Figures 15.6 and 15.7 show operation examples for the NMI interrupt.


Hardware pre-processing
Store register values
SPC = PC
SSR = SR
SGR = R15
INTEVT = H'000001C0
Update settings
MD in SR = "1"
RB in SR = "1"
BL in SR = "1"
[When INTMU in CPUOPM = "1"]
IMASK bits in SR = B'1111
Set vector address in PC
PC = VBR register + H'600
Interrupt handler
If necessary, store PR register, general registers, and floating point registers on stack
Read INTEVT
Branch to interrupt routine for accepted interrupt
NMI interrupt routine
Actual processing
If necessary, restore PR register, general registers, and floating point registers from stack
RTE instruction
Program currently running (BL in SR = "0" and NMIB in ICR0 = "0")
Exception event occurs
*1
Hardware post-processing
Restore register values
PC = SPC
SR = SSR

Note *1 The interrupt is accepted at the next gap between instructions when the value of the BL bit in SR is "0" and the active edge has been detected on the NMI pin.

Figure 15.6 NMI Interrupt Operation Example (When Value of NMIB Bit in ICR0 is "0")



Note *1 The interrupt is accepted at the next gap between instructions when the active edge has been detected on the NMI pin.
*2 When an NMI interrupt is accepted while the value of BL in SR is "1", earlier exception data (SSR, SPC, SGR, and INTEVT) may be lost. Control must therefore not be returned to the program that was running when the NMI interrupt occurred.

Figure 15.7 NMI Interrupt Operation Example (When Value of NMIB Bit in ICR0 is "1")

15.5 Interrupt Response Time

Table 15.9 shows the interrupt response time, which is the interval from when an interrupt request occurs until the interrupt exception handling is started and the start instruction of the interrupt handling is fetched.

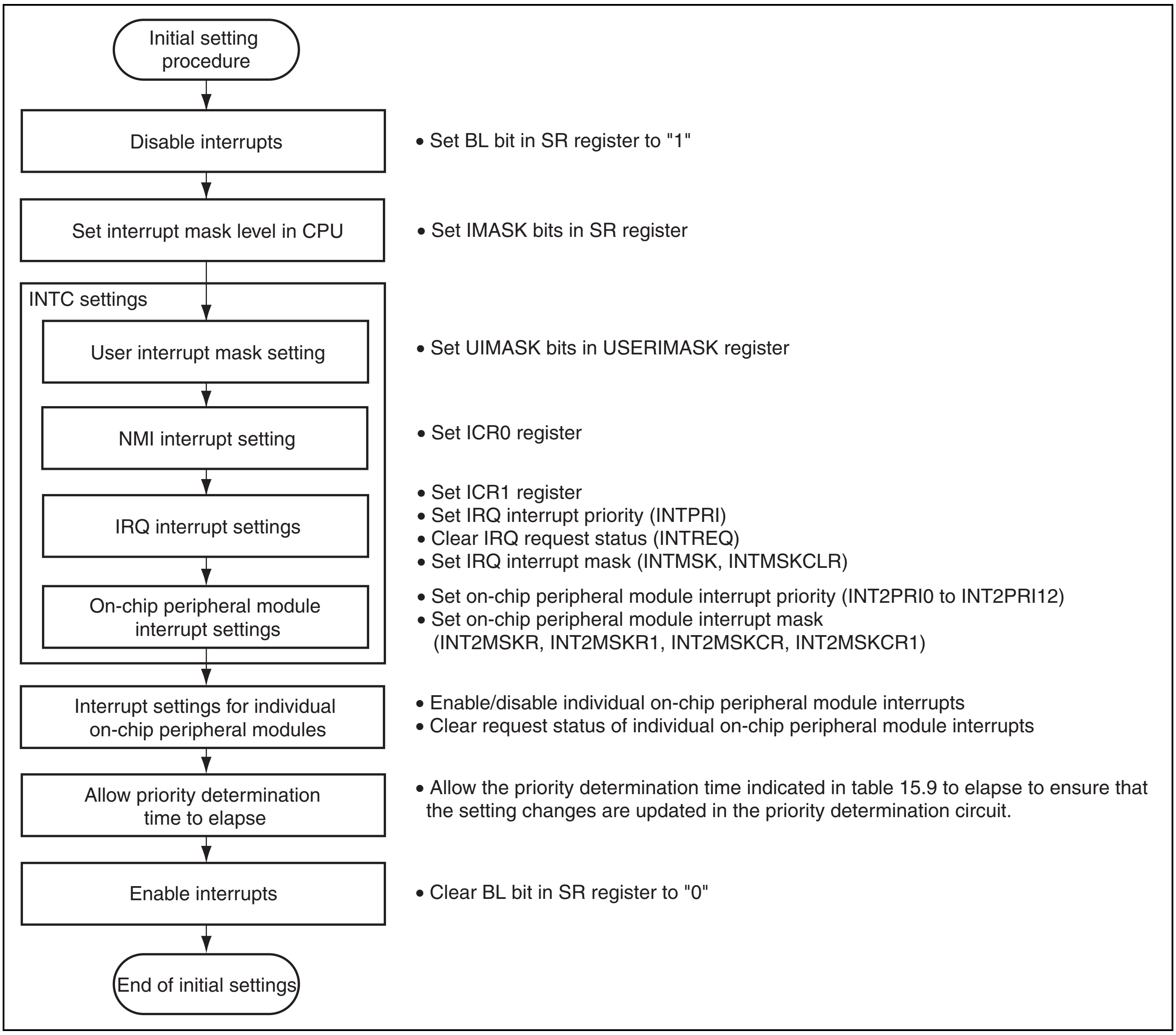
Table 15.9 Interrupt Response Time

Item		Number of State			Remarks
		NMI	IRQ	Peripheral Module	
Priority determination time		7 Pcyc	6 Pcyc	5Pcyc	
Wait time until the CPU finishes the current sequence			S-1 (≥ 0) × Icyc		
Interval from when interrupt exception handling begins (saving SR and PC) until a SHwy bus request is issued to fetch the start instruction of the interrupt handling			11Icyc + 1Scyc		
Response time	Total	(S + 10) Icyc + 1Scyc + 7 Pcyc	(S + 10) Icyc + 1Scyc + 6 Pcyc	(S + 10) Icyc + 1Scyc + 5Pcyc	
	Minimum	40Icyc + S × Icyc	36Icyc + S × Icyc	32Icyc + S × Icyc	When Icyc:Scyc: Pcyc = 4:2:1

Legend:
Icyc: Period for one CPU clock cycle
Scyc: Period for one SHwy clock cycle
Pcyc: Period for one peripheral clock cycle
S: Number of instruction execution states

15.6 Initial Setting Procedure Example

Figure 15.8 shows an example of the initial setting procedure of the INTC. Before enabling interrupts, it is necessary to allow the priority determination time indicated in table 15.9 to elapse to ensure that the setting changes are updated in the priority determination circuit.


Initial setting procedure
Disable interrupts
Set interrupt mask level in CPU
INTC settings
User interrupt mask setting
NMI interrupt setting
IRQ interrupt settings
On-chip peripheral module interrupt settings
Interrupt settings for individual on-chip peripheral modules
Allow priority determination time to elapse
Enable interrupts
End of initial settings

- Set BL bit in SR register to "1"
- Set IMASK bits in SR register
- Set UIMASK bits in USERIMASK register
- Set ICR0 register
- Set ICR1 register
- Set IRQ interrupt priority (INTPRI)
- Clear IRQ request status (INTREQ)
- Set IRQ interrupt mask (INTMSK, INTMSKCLR)
- Set on-chip peripheral module interrupt priority (INT2PRI0 to INT2PRI12)
- Set on-chip peripheral module interrupt mask (INT2MSKR, INT2MSKR1, INT2MSKCR, INT2MSKCR1)
- Enable/disable individual on-chip peripheral module interrupts
- Clear request status of individual on-chip peripheral module interrupts
- Allow the priority determination time indicated in table 15.9 to elapse to ensure that the setting changes are updated in the priority determination circuit.
- Clear BL bit in SR register to "0"

Figure 15.8 INTC Initial Setting Procedure Example

15.7 Usage Notes

15.7.1 Notes on Setting IRQ Pin Function

Pins IRQ0 to IRQ2 and IRQ5 to IRQ7 are multiplexed, meaning that they can be set to other functions. As a result, when these pins are switched to function as IRQ0 to IRQ2 and IRQ5 to IRQ7, interrupt requests may be detected erroneously and maintained internally by the INTC. It is therefore necessary to mask IRQ interrupt requests before switching the functions of the pins to IRQ0 to IRQ2 and IRQ5 to IRQ7 in the pin function unit.

Table 15.10 Switching Sequence of IRQ Pin Function

Sequence	Item	Procedure
1	IRQ interrupt request masking	Write "1" to all bits in the INTMSK register.
2	Setting the IRQ0 to IRQ2, and IRQ5 to IRQ7 functions by the pin function unit	Select the pins to be used with the corresponding port control register.
3	Starting IRQ interrupt request detection	Write "1" to the corresponding bit or bits in the INTMSKCLR register.

15.7.2 To Clear IRQ Interrupt Requests When Level Detection is Selected

To clear IRQ interrupt requests that are set up for level detection, write "1" to the INTMSK.IRQn (n = 0 to 2, 5 to 7) bit that correspond to the requests to be cleared. It is not possible to clear detected IRQ interrupt requests by writing "0" to the corresponding bits in the INTPRI register. Which IRQ interrupt requests have been detected can be checked by reading the INTREQ register. The clearing procedure of the IRQ interrupt request when level detection is as follows.

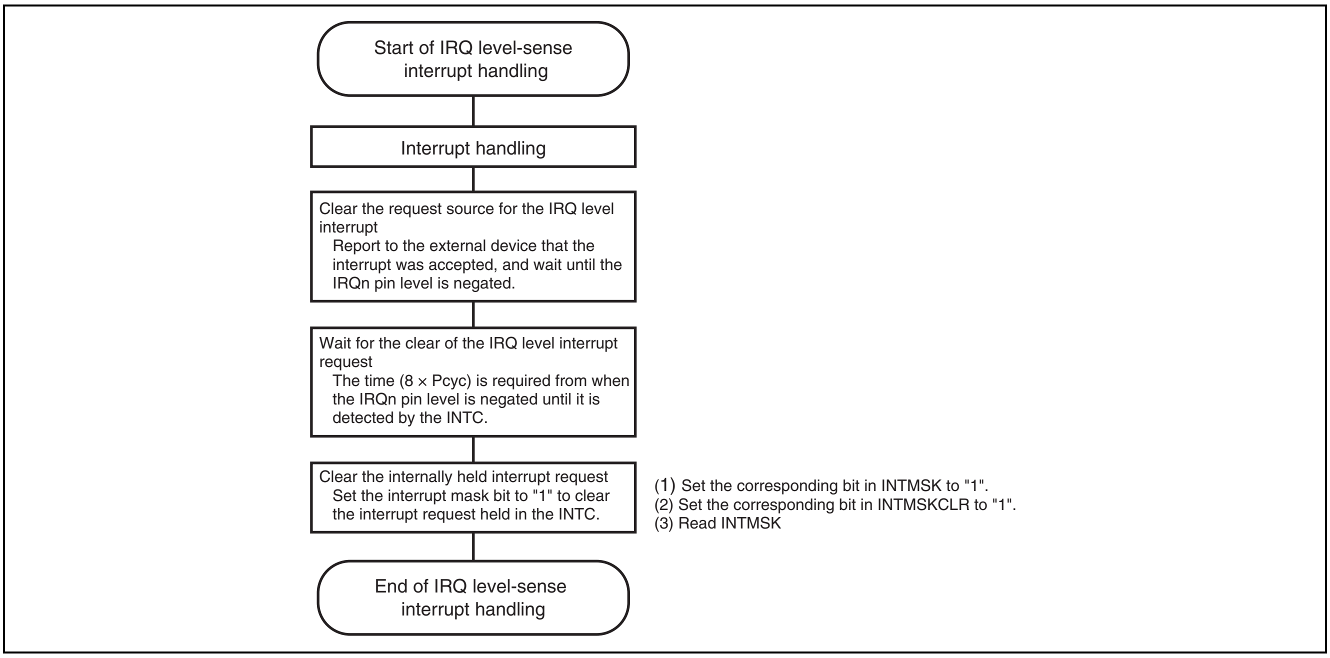
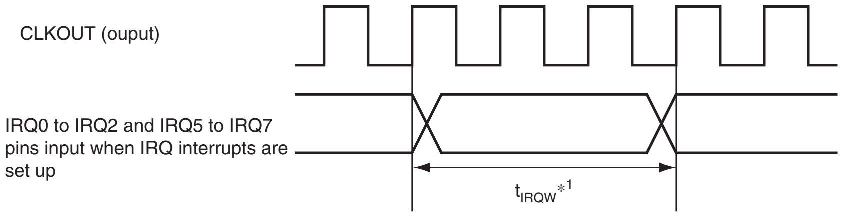

Start of IRQ level-sense interrupt handling
Interrupt handling
Clear the request source for the IRQ level interrupt
Report to the external device that the interrupt was accepted, and wait until the IRQn pin level is negated.
Wait for the clear of the IRQ level interrupt request
The time (8 × Pcyc) is required from when the IRQn pin level is negated until it is detected by the INTC.
Clear the internally held interrupt request
Set the interrupt mask bit to "1" to clear the interrupt request held in the INTC.
(1) Set the corresponding bit in INTMSK to "1".
(2) Set the corresponding bit in INTMSKCLR to "1".
(3) Read INTMSK
End of IRQ level-sense interrupt handling

Figure 15.9 Example of Interrupt Handling Routine


CLKOUT (ouput)
IRQ0 to IRQ2 and IRQ5 to IRQ7 pins input when IRQ interrupts are set up
tIRQW*1

Note: *1 Refer to section 38, Electrical Characteristics.

Figure 15.10 Time Requested to Detect Interrupts from IRQ

15.7.3 To Clear IRQ Interrupt Requests When Edge Detection is Selected

To clear IRQ interrupt requests that are set up for edge detection, write "0" to the INTREQ.IRQn (n = 0 to 2, 5 to 7) bit that correspond to the requests to be cleared after reading "1". Write "1" to the bits whose value was read as "0". It is not possible to clear detected IRQ interrupt requests by writing "1" to the corresponding bits in the INTMSK register.

This page is blank for reasons of layout.

Section 16 Reset

16.1 Reset Operation

When a "L" level pulse with a width greater than the noise cancel width (t_{RESNCW}) is applied to the RESET# pin, this MCU will accept the reset.

When a reset is accepted, each pin is switched to its reset state. After the reset is accepted and after a delay of 3 or 4 Pck cycles, internal circuits including the CPU are reset. The width of the "L" level pulse input to the RESET# pin must be longer than the RESET# pulse width (t_{RESW}). After that, the internal circuit reset will be cleared 2150 Pck cycles after a "H" level signal is input to the RESET# pin. CPU reset exception handling starts at that point.

The RESET# noise cancellation width is stipulated by t_{RESNCW}, and the RESET# pulse width is stipulated by t_{RESW}. See section 38, Electrical Characteristics, for details.

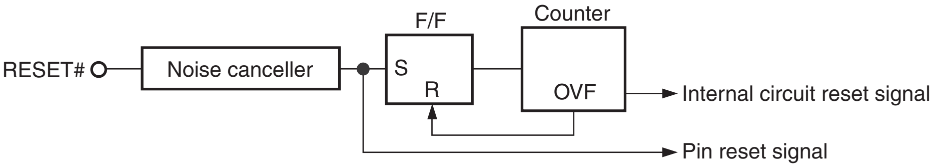


Figure 16.1 Reset Circuit

16.2 Input/Output Pins

Table 16.1 lists the configuration and functions of the RESET# pin.

This pin is not multiplexed with other functions.

Table 16.1 Pin Configuration

Pin Name	I/O	Function
RESET#	Input	Reset input pin This MCU switches to the reset state when a "L" level is input.

16.3 Operation

16.3.1 Reset Request

This section describes the reset generation request and transition operation.

Unless specifically stated otherwise, the term "reset" is used to refer generally to any one of the reset factors occurring.

- Sources

1. When a "L" level signal is input to the RESET# pin
2. When the WDTCNT register overflows in the state when the WTIT bit in the WDTCSR register is set to "1"
3. When a reset occurs due to an H-UDI command being issued
   See section 37, User Debugging Interface (H-UDI), for details.
4. When an instruction TLB multiple hit exception occurs
5. When a data TLB multiple hit exception occurs
   For details, see section 7, Memory Management Unit (MMU).
6. A general exception other than a user break is generated while SR.BL = "1".
   A reset triggered by source 1. or 2. of the above is referred to as a hardware reset.

- Transition target address: H'A000 0000
- Transition operation
  When a reset source occurs, the CPU starts reset exception handling.
  For details, see section 5, Exception Handling.

16.3.2 Reset with the RESET# Pin

Always input a "L" level signal to the RESET# pin when the system is powered on. Also, it is necessary to input a "L" level signal to the TRST# pin to initialize the H-UDI.

After the RESET# pin is changed from "L" to "H", the reset state continues internally to the IC until the reset hold time has passed. The reset hold time is a period greater than or equal to 1075 cycles of the EXTAL pin input signal period.

(1) Power On Reset

When power is first applied, a "L" level signal must be input to the RESET# pin during the total period of the oscillation circuit and PLL oscillation stabilization time after the supply voltage rises within the limits. See section 34, Power Supply Circuit, for details on the power on sequence. The oscillation stabilization time is specified as t_{OSC1}. For details, see section 38, Electrical Characteristics. Also, a "L" level signal must be input to the TRST# pin to initialize the H-UDI module.

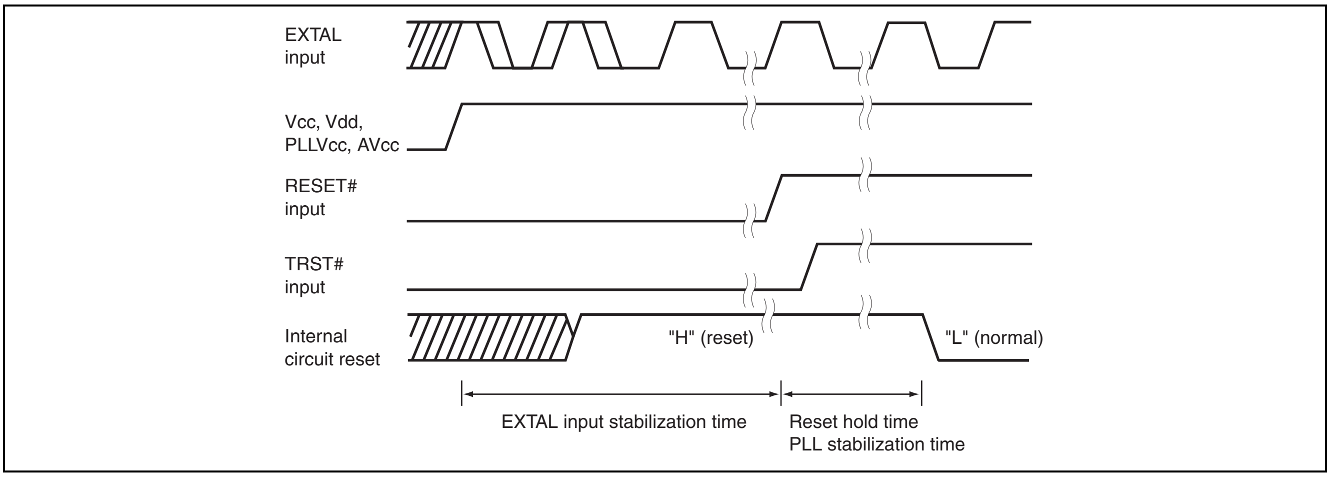


Figure 16.2 Power On Reset

(2) Reset During MCU Operation

When generating a reset during normal operation, a "L" level with a width greater than the reset pulse width must be applied to the RESET# pin. The reset pulse width is specified as t_{RESW}. For details, see section 38, Electrical Characteristics.

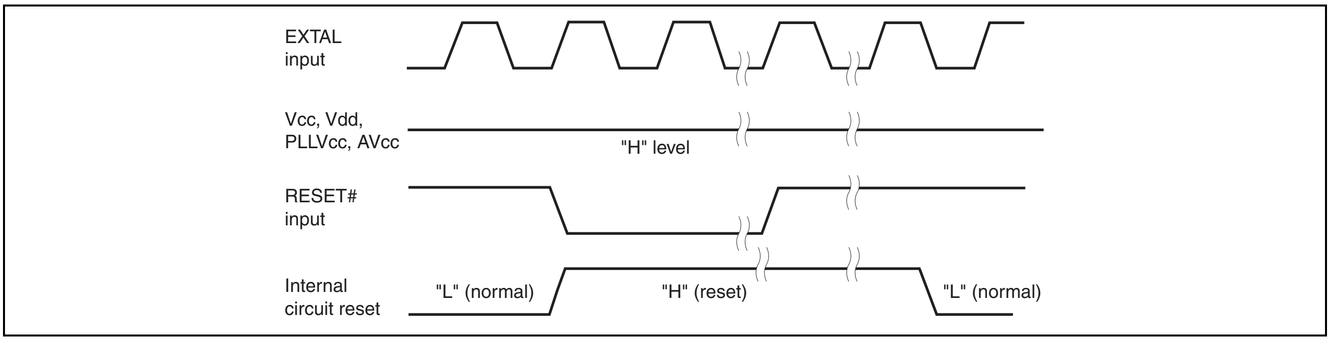


Figure 16.3 Reset During MCU Operation

16.4 Usage Notes

16.4.1 Usage Notes Regarding Internal Reset

When the microcontroller is changing to reset status caused by an internal reset, there will be a period of one peripheral clock (Pck), where I/O ports will be undefined. Undefined state could be either high level output or low level output or high impedance states. The microcontroller, including I/O ports, will correctly transfer to reset status after this period.

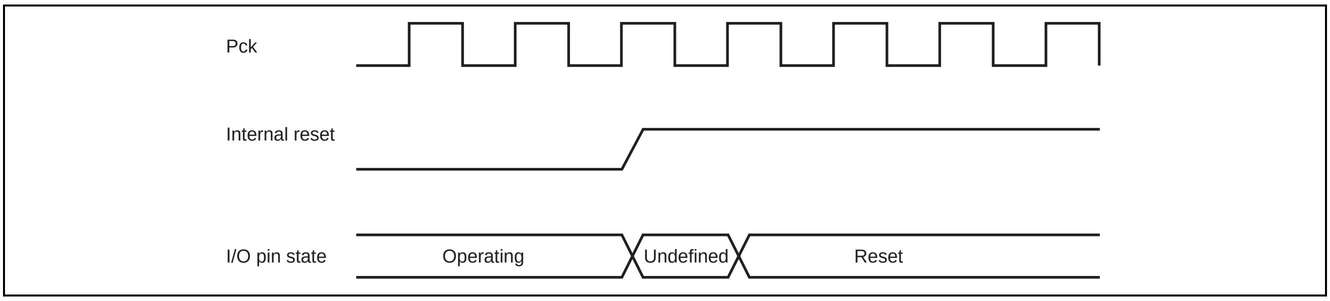


Figure 16.4 I/O Pin Operations Caused by an Internal Reset

Internal resets can be caused by the following.

- Reset by watchdog timer (WDT) overflow
- H-UDI reset during debug
- Reset by exception

Reset by inputting a low signal to RESET# pin is not an internal reset.

Table 16.2 lists the affected input/output pins. For the other pins, when receiving internal reset, the pin output becomes high-impedance simultaneously.

Table 16.2 Corresponding Input/Output Pins

Corresponding Input/Output Pins		I/O Pin Operations During a Reset by an Internal Source	
Functions	**Pin functions** **The names of pins are given within the brackets.**	**Pin state during operations → pin state while the chip's state is undefined → pin state in the reset state**	**Descriptions**
ATU-III	TOnm [PA0 to PA13, PC0 to PC3, PC5, PC6]	"H" or "L" output → "L" output → Hi-Z	When receiving internal reset, "L" level is output among Max. of 1 Pck cycle. The "L" level is initial value of port data register.
PSEL	PSLCLKB [PA8], PSLCLKA [PA9], PSLDATA0 [PA10], PSLDATA1 [PA11], PSLDATA2 [PA12], PSLDATA3 [PA13]		
ADC	AD0END [PJ11]		
RCAN	CTXi [PF1, PJ1, PJ3, PJ5, PJ7]	"H" or "L" output → "H" output → Hi-Z	When receiving internal reset, "H" level is output among Max. of 1 Pck cycle. The "H" level is initial value of port data register.
AUDR	AUDREVT# [PL9]		
ATU-III	TOnm (m=0 to 5) (When PWM output-prohibit function is enable) [PA0 to PA5, PA8 to PA13, PC0 to PC3, PC5]	Hi-Z → "L" output → Hi-Z	When receiving internal reset, "L" level is output among Max. of 1 Pck cycle. The "L" level is initial value of port data register.

Notes: 1. If the corresponding function is not selected, the pin output becomes high-impedance simultaneously, when receiving internal reset.

2. The symbols "H", "L", and "Hi-Z" in the table indicate the "H" level, the "L" level, and the high-impedance state, respectively.

This page is blank for reasons of layout.

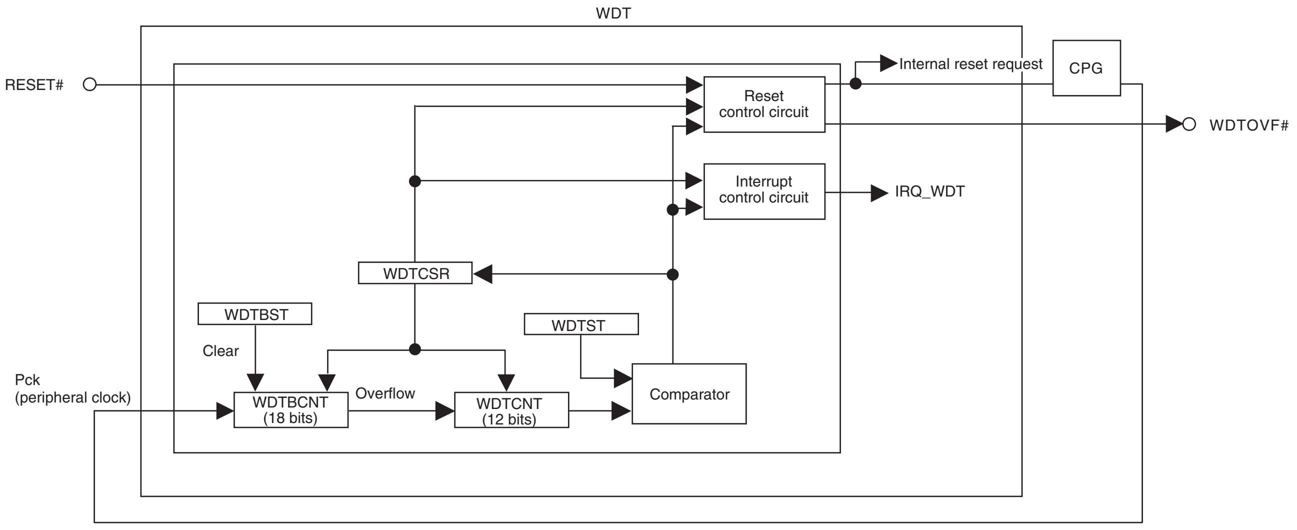
Section 17 Watchdog Timer (WDT)

The watchdog timer module consists of a single timer channel that can be used as either a watchdog timer or an interval timer.

17.1 Overview

- Implements a system runaway monitoring function using a timer that is incremented with a fixed period.
- Provides both a watchdog timer mode, in which a hardware reset is issued when a counter overflows, and a interval timer mode that generates periodic interrupts.
- In watchdog timer mode, a hardware reset is issued when a counter overflows and the WDTOVF# signal is output.
- In interval timer mode, the interval timer interrupt is generated by counter overflow.
- A code value is stored in the upper 8 bits of the watchdog timer related registers so that they cannot be rewritten easily.
- The maximum time until the counter overflows is about 27 seconds (when the peripheral clock Pck is 40 MHz).

Figure 17.1 shows the block diagram of the WDT module.



Note: • IRQ_n indicates the interrupt number and the same number indicates the same group interrupt.

Legend:

WDTBCNT:	Watchdog timer base counter
WDTBST:	Watchdog timer base stop time register
WDTCNT:	Watchdog timer counter
WDTCSR:	Watchdog timer control/status register
WDTST:	Watchdog timer stop time register

Figure 17.1 Block Diagram of WDT

17.2 Input/Output Pins

Table 17.1 lists the WDT module pins.

This pin is not multiplexed with other functions.

Table 17.1 Pin Configuration

Pin Name	I/O	Function
WDTOVF#	Output	Counter overflow output pin Outputs the counter overflow signal in watchdog timer mode.

17.3 Register Descriptions

Table 17.2 lists the WDT module registers.

Table 17.2 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Watchdog timer stop time register	WDTST	H'0000 0000	H'FFFF 1000	32	17-3
Watchdog timer control/status register	WDTCSR	H'0000 0000	H'FFFF 1004	32	17-4
Watchdog timer base stop time register	WDTBST	H'0000 0000	H'FFFF 1008	32	17-5
Watchdog timer counter	WDTCNT	H'0000 0000	H'FFFF 1010	32	17-5
Watchdog timer base counter	WDTBCNT	H'0000 0000	H'FFFF 1018	32	17-6

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

17.3.1 Watchdog Timer Stop Time Register (WDTST)

The WDTST register specifies the WDTCNT counter overflow value. Setting this register to H'0000 0001 sets the shortest time until overflow, and setting it to H'0000 0000 sets the longest time until overflow.

When writing to the WDTST register, the WDTSTKEY field must be set to H'5A and the long word access size must be used. The WDTSTKEY field always returns "0" when read.

The WDTST register is reset by the RESET# pin and initialized to H'0000 0000. Note, however, that the value set prior to the reset will be retained for resets due to counter overflows in watchdog timer mode and resets due to H-UDI.

Watchdog timer stop time register (WDTST) <P4 address: location H'FFFF 1000>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDTSTKEY								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	WDTST											
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	WDTSTKEY	All 0	0	W	WDTST Register Write Key Code Bits These bits enable or disable WDTST bit modification. The data written to these bits are not retained. These bits are always read as "0". H'5A: Enable WDTST bit modification. Other than H'5A: Disable WDTST bit modification.
23 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11 to 0	WDTST	H'000	R	W	WDTCNT Counter Overflow Value

17.3.2 Watchdog Timer Control/Status Register (WDTCSR)

When writing to the WDTCSR register, the WDTCSRKEY field must be set to H'A5 and the long word access size must be used. The WDTSTKEY field always returns "0" when read. The WDTCSR register is reset by the RESET# pin and initialized to "H'0000 0000". Although the value set prior to the reset will be retained for resets due to counter overflows in watchdog timer mode, the WOVF bit is set to "1". The value set prior to the reset will be retained for resets due to the H-UDI.

Watchdog timer control/status register (WDTCSR) <P4 address: location H'FFFF 1004>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDTCSRKEY								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TME	WTIT	—	WOVF	IOVF	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	WDTCSRKEY	All 0	0	W	WDTCSR Register Write Key Code Bits These bits enable or disable lower bit modification. The data written to these bits are not retained. These bits are always read as "0". H'A5: Enable lower bit modification. Other than H'A5: Disable lower bit modification.
23 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7	TME	0	R	W	Timer Enable Bit Starts or stops timer operation. 0: Stops the counting operation 1: Starts the counting operation
6	WTIT	0	R	W	Timer Mode Select Bit Specifies whether this module is used as a watchdog timer or as an interval timer. Note that writing the WTIT bit is not allowed during watchdog timer operation. 0: Interval timer mode 1: Watchdog timer mode
5	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
4	WOVF	0	R	*[1]	Watchdog Timer Overflow Flag Indicates whether or not the WDTCNT counter has overflowed in watchdog timer mode. This bit is not set in interval timer mode. 0: No overflow has occurred 1: The WDTCNT counter overflowed in watchdog timer mode
3	IOVF	0	R	*[1]	Interval Timer Overflow Flag Indicates whether or not the WDTCNT counter has overflowed in interval timer mode. This bit is not set in watchdog timer mode. 0: No overflow has occurred 1: The WDTCNT counter overflowed in interval timer mode

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Note: *1 Only "0" is valid on write. The previous value is retained if a "1" is written.

17.3.3 Watchdog Timer Base Stop Time Register (WDTBST)

When clearing the WDTBST register, the WDTCSRKEY field must be set to H'55 and the long word access size must be used. When read, the WDTBST register always returns H'0000 0000.

Watchdog Timer base stop time register (WDTBST) <P4 address: location H'FFFF 1008>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDTBSTKEY								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	WDTBSTKEY	All 0	0	W	WDTBST Register Write Key Code Bits These bits clear the WDTBCNT counter. The data written to these bits are not retained. These bits are always read as "0". H'55: Clear the WDTBCNT counter. Other than H'55: Data write is invalid.
23 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

17.3.4 Watchdog Timer Counter (WDTCNT)

The WDTCNT counter is incremented by the overflow of the WDTBCNT counter. When the WDTCNT counter itself overflows, in watchdog timer mode the reset will be issued or in interval timer mode an interrupt will be generated. Note that the WDTCNT counter is a read-only register and that writing this register is invalid.

The WDTCNT counter is reset by the RESET# pin and initialized to H'0000 0000. Note, however, that although the counter is reset by resets due to counter overflows in watchdog timer mode and resets due to H-UDI, after these resets are cleared, the increment operation will restart.

Watchdog Timer Counter (WDTCNT) <P4 address: location H'FFFF 1010>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	WDTCNT											
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 12	—	All 0	0	—	Reserved Bits These bits are always read as "0".
11 to 0	WDTCNT	All 0	R	—	WDTCNT Count Value

17.3.5 Watchdog Timer Base Counter (WDTBCNT)

The WDTBCNT counter is incremented by the peripheral clock (Pck). When the WDTBCNT counter overflows, the WDTCNT counter is incremented and the WDTBCNT counter is cleared to H'0000 0000. Note that the WDTBCNT counter is a read-only register and that writing this register is invalid.

The WDTBCNT counter is reset by the RESET# pin and initialized to H'0000 0000. Note, however, that although the counter is reset by resets due to counter overflows in watchdog timer mode and resets due to H-UDI, after these resets are cleared, the increment operation will restart.

Watchdog timer base counter (WDTBCNT) <P4 address: location H'FFFF 1018>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTBCNT	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDTBCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 18	—	All 0	0	—	Reserved Bits These bits are always read as "0".
17 to 0	WDTBCNT	All 0	R	—	WDTBCNT Count Value

17.4 Operation

17.4.1 Using Watchdog Timer Mode

1. Set the WDTCNT counter overflow time in the WDTST register.
2. Set the WDTCSR register WTIT bit to "1".
3. Set the WDTCSR register TME bit to "1": this starts the WDT counter count operation.
4. In watchdog timer mode, the application must periodically clear the WDTCNT or the WDTBCNT counter so that the WDTCNT counter does not overflow. See section 17.4.4, Clearing the WDT Counter, for the procedure for clearing this counter.
5. If the WDTCNT counter overflows, the WDTCSR register WOVF flag will be set to "1" and a hardware reset will be issued.

17.4.2 Using Interval Timer Mode

In interval timer mode, the interval timer interrupt is generated each time the counter overflows. Thus an interrupt is generated once every fixed period.

1. Set the WDTCNT counter overflow time in the WDTST register.
2. Set the WDTCSR register WTIT bit to "0".
3. Set the WDTCSR register TME bit to "1": this starts the WDT counter count operation.
4. When the WDTCNT counter overflows, the WDTCSR register IOVF flag will be set to "1" and an interval timer interrupt request will be generated. The WDTCNT counter and the WDTBCNT counter count operation will continue at this time.

17.4.3 Time Until a WDT Overflow Occurs

Figure 17.2 shows the relationship between WDTCNT counter and the WDTBCNT counter.

The figure presents an example of operation in interval timer mode, and counting operation continues after the WDTCNT counter overflows. In watchdog timer mode, after the reset due to the counter overflow is cleared, the WDTCNT counter and the WDTBCNT counter are cleared to "0" and the counting operation is restarted.

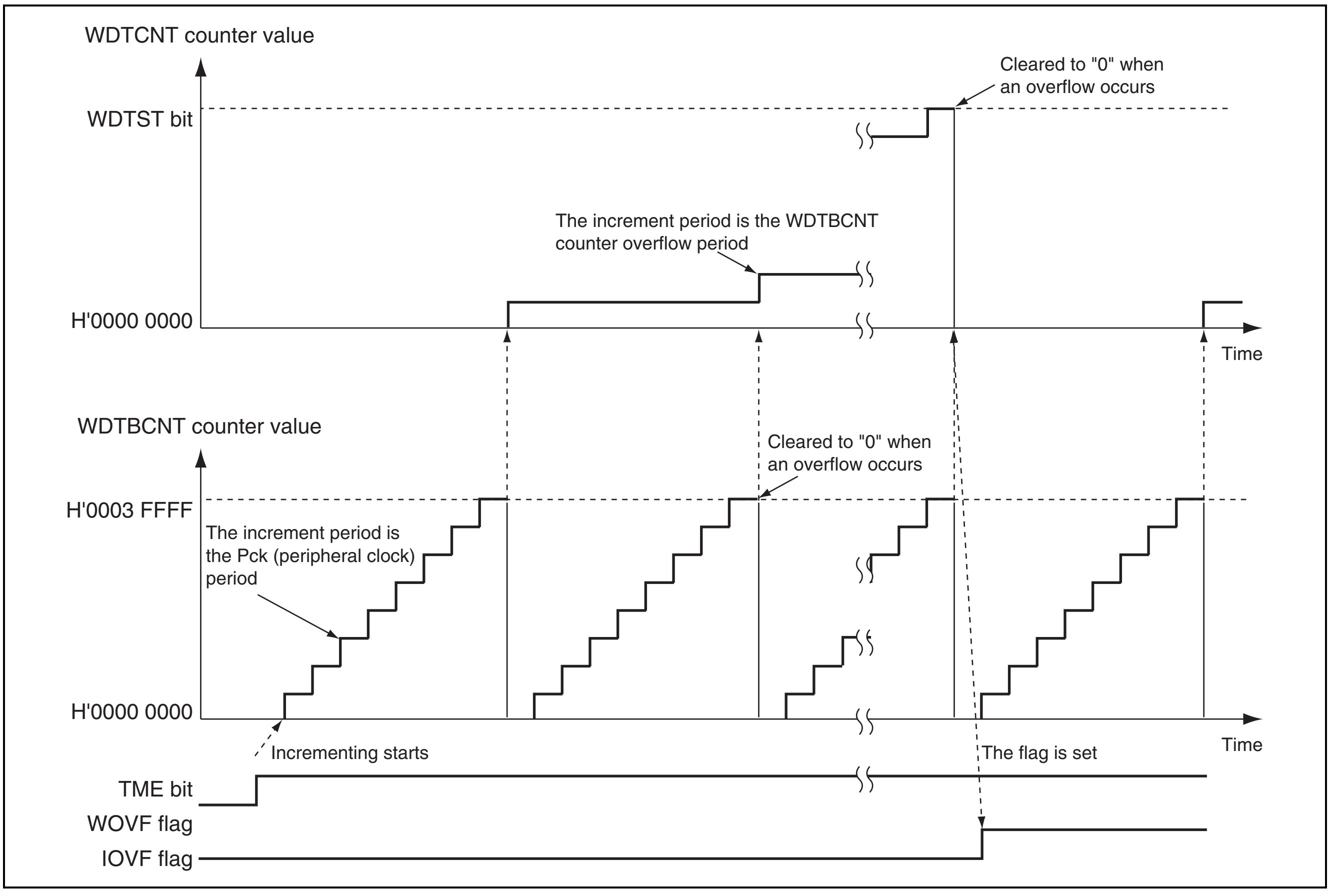

WDTCNT counter value
WDTST bit
Cleared to "0" when an overflow occurs
The increment period is the WDTBCNT counter overflow period
H'0000 0000
Time
WDTBCNT counter value
Cleared to "0" when an overflow occurs
H'0003 FFFF
The increment period is the Pck (peripheral clock) period
H'0000 0000
Incrementing starts
The flag is set
Time
TME bit
WOVF flag
IOVF flag

Figure 17.2 WDT Increment Operations (in interval timer mode)

The WDTBCNT counter is an 18-bit counter that counts (is incremented) every cycle of the peripheral clock (Pck). Writing H'5500 0000 to the WDTBST register clears the WDTBCNT counter. When the peripheral clock frequency is 40 MHz, the duration from when the WDTBCNT counter value is "0" until overflow is as follows:

2^18 [bits] × 1 / 40 [MHz] = approx. 6.554 [ms]

The WDTCNT counter is a 12-bit counter that counts (is incremented by) WDTBCNT counter overflow events. The WDTCNT counter is cleared by writing an overflow value to the WDTST register.

Overflow occurs when the WDTCNT counter value matches the overflow value set in the WDTST bits. The duration until WDTCNT counter overflow is calculated as follows:

Setting value of WDTST bits × approx. 6.554 [ms]

Note that setting the WDTST bits to "H'000" sets the counter value to "H'1000".

Therefore, the maximum duration until overflow is when the WDTST bits in the WDTST register are set to "H'000":

2^12 [bits] × 6.554 [ms] = approx. 26.84 [s]

The minimum duration until overflow is when the WDTST bits in the WDTST register are set to "H'001":

2^0 [bits] × 6.554 [ms] = approx. 6.554 [ms]

17.4.4 Clearing the WDT Counter

The WDTBCNT and WDTCNT counters can be cleared in the following ways:

(1) WDTBCNT counter clearing conditions

- Write H'5500 0000 to the WDTBST register in a longword unit
- Reset by the RESET# pin
- Reset by a counter overflow in watchdog timer mode
- Reset by using the H-UDI

(2) WDTCNT counter clearing conditions

- Write the WDTST bits in the WDTST register (The counter is not cleared if the wrong key code is written.)
- Reset by the RESET# pin
- Reset by a counter overflow in watchdog timer mode
- Reset by using the H-UDI

17.4.5 Hardware Reset due to WDT Overflow

The hardware reset period (WDT reset period) due to a WDT overflow is a range from 9 to 18 of EXTAL pin input cycles. Also, the time from a WDT overflow to the point this MCU has transitioned to the hardware reset state (the WDT reset request period) is at least 1 EXTAL pin input cycle.

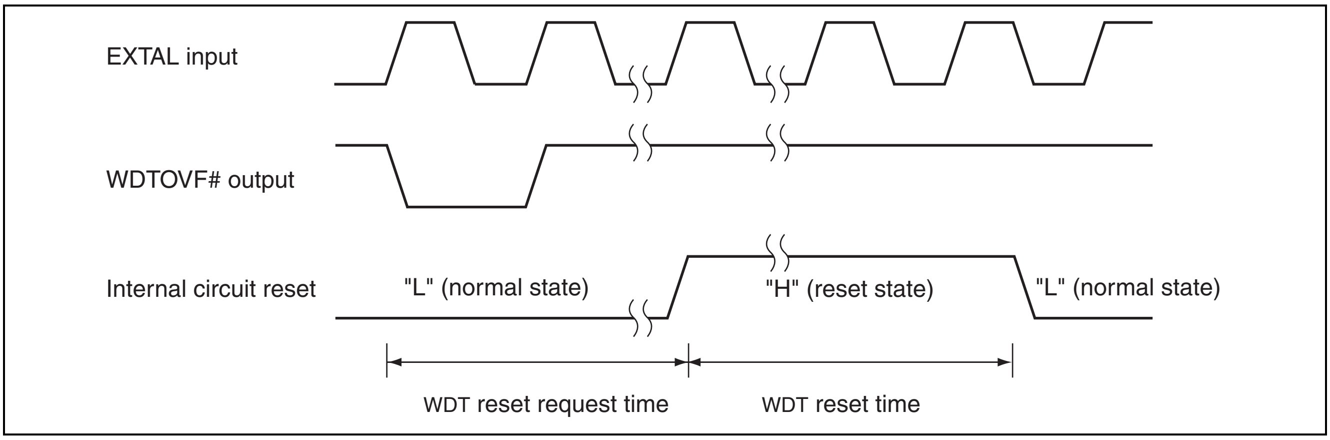


Figure 17.3 WDT Overflow Hardware Reset During Normal Operation

This page is blank for reasons of layout.

Section 18 I/O Ports and Pin Function Unit

18.1 Overview

This MCU provides 108 I/O port pins, designated as A, B, C, D, E, F, G, H, J, K, L, M, and N.

Each port can be used as input or output by setting the respective port IO registers. Each port also functions as a multiplexed pin shared with on-chip peripheral module signals. The function of each multiplexed pin function is selected by using the corresponding control register.

Other port functions, including an output driving ability setting function, input threshold value switching function are also incorporated.

Table 18.1 lists the overview of I/O ports and the pin function unit.

Table 18.1 Overview of I/O Ports and Pin Function Unit

Item	Overview
Number of ports	Total 108 pins Port A: PA0 to PA13 (14 pins) Port B: PB0, PB1, and PB3 (3 pins) Port C: PC0 to PC3, PC5, PC6, and PC14 (7 pins) Port D: PD0 to PD10 (11 pins) Port E: PE15 (1 pin) Port F: PF0, PF1, PF4, and PF5 (4 pins) Port G: PG0 to PG4 (5 pins) Port H: PH0 to PH15 (16 pins) Port J: PJ0 to PJ7 and PJ10 to PJ15 (14 pins) Port K: PK0, PK5, PK6, and PK8 to PK14 (10 pins) Port L: PL2 to PL6, PL8, and PL9 (7 pins) Port M: PM0, PM2, PM4, PM6, and PM8 to PM15 (12 pins) Port N: PN0, PN1, PN4, and PN5 (4 pins)
Port function	Input or output can be set in port units by using the port IO register (port M and port N are input-only).
Driving ability setting function	The output driving ability of the corresponding port can be increased by using the driving ability setting register (except for port M and port N).
Input threshold value switching function	The input threshold value can be specified to three voltage levels (0.35 Vcc, 0.5 Vcc, or 0.7 Vcc) in port group units. Note that the ports that the input threshold value can be specified are ports except for PG0 to PG3, PJ1, PJ3 to PJ5, PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, and PN5.

18.2 Multiplex Pin Functions

Each port pin functions as a multiplexed pin shared with general input/output and another function. The function of each multiplexed pin functions are is selected by the corresponding control register.

Tables 18.2 to 18.14 list the multiplexed pin functions for each port. Table 18.15 lists the pin functions allowed for input/output at two pins and pin assignments.

Table 18.2 Multiplexed Pin Functions for Port A

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PA0 (port)	—	TO00 (ATU-IIIS)	DDB00 (DRI)	—	—
PA1 (port)	—	TO01 (ATU-IIIS)	DDB01 (DRI)	—	—
PA2 (port)	—	TO02 (ATU-IIIS)	DDB02 (DRI)	—	—
PA3 (port)	—	TO03 (ATU-IIIS)	DDB03 (DRI)	—	—
PA4 (port)	—	TO04 (ATU-IIIS)	DDB04 (DRI)	—	—
PA5 (port)	—	TO05 (ATU-IIIS)	DDB05 (DRI)	—	—
PA6 (port)	—	TO06 (ATU-IIIS)	DDB06 (DRI)	—	—
PA7 (port)	—	TO07 (ATU-IIIS)	DDB07 (DRI)	—	—
PA8 (port)	—	TO10 (ATU-IIIS)	DDB08 (DRI)	PSLCLKB (PSEL)	—
PA9 (port)	—	TO11 (ATU-IIIS)	DDB09 (DRI)	PSLCLKA (PSEL)	—
PA10 (port)	—	TO12 (ATU-IIIS)	DDB10 (DRI)	PSLDATA0 (PSEL)	—
PA11 (port)	—	TO13 (ATU-IIIS)	DDB11 (DRI)	PSLDATA1 (PSEL)	—
PA12 (port)	—	TO14 (ATU-IIIS)	DDB12 (DRI)	PSLDATA2 (PSEL)	—
PA13 (port)	—	TO15 (ATU-IIIS)	DDB13 (DRI)	PSLDATA3 (PSEL)	—

Table 18.3 Multiplexed Pin Functions for Port B

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PB0 (port)	—	PWMOFF0 (ATU-IIIS)	DINB0 (DRI)	—	—
PB1 (port)	—	PWMOFF1 (ATU-IIIS)	DINB1 (DRI)	—	—
PB3 (port)	—	PWMOFF3 (ATU-IIIS)	DINB3 (DRI)	—	—

Table 18.4 Multiplexed Pin Functions for Port C

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PC0 (port)	—	TO30 (ATU-IIIS)	—	MOSI2 (RSPI)	(IRQ6) (INTC)
PC1 (port)	—	TO31 (ATU-IIIS)	—	MISO2 (RSPI)	—
PC2 (port)	—	TO32 (ATU-IIIS)	—	RSPCK2 (RSPI)	DREQ0 (DMAC)
PC3 (port)	—	TO33 (ATU-IIIS)	—	SSL20 (RSPI)	IRQ0 (INTC)
PC5 (port)	—	TO35 (ATU-IIIS)	—	—	—
PC6 (port)	CLKOUT (CPG)	TO36 (ATU-IIIS)	—	—	—
PC14 (port)	—	—	—	—	—

Table 18.5 Multiplexed Pin Functions for Port D

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PD0 (port)	—	PDIDATA0 (PDAC)	—	—	—
PD1 (port)	—	PDIDATA1 (PDAC)	—	—	—
PD2 (port)	—	PDIDATA2 (PDAC)	—	—	—
PD3 (port)	—	PDIDATA3 (PDAC)	—	—	—
PD4 (port)	—	PDIDATA4 (PDAC)	—	—	—
PD5 (port)	—	PDIDATA5 (PDAC)	—	—	—
PD6 (port)	—	PDIDATA6 (PDAC)	—	—	—
PD7 (port)	—	PDIDATA7 (PDAC)	—	—	—
PD8 (port)	—	PDIDATA8 (PDAC)	—	—	—
PD9 (port)	—	PDIDATA9 (PDAC)	—	—	—
PD10 (port)	—	PDIWR (PDAC)	—	—	—

Table 18.6 Multiplexed Pin Functions for Port E

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PE15 (port)	—	TO27 (ATU-IIIS)	—	PSLCLR (PSEL)	—

Table 18.7 Multiplexed Pin Functions for Port F

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PF0 (port)	CRX0 (CAN)	—	—	—	—
PF1 (port)	CTX0 (CAN)	—	—	—	—
PF4 (port)	SDA (IIC3)	—	—	—	(CRX3) (CAN)
PF5 (port)	SCL (IIC3)	—	—	—	(CTX3) (CAN)

Table 18.8 Multiplexed Pin Functions for Port G

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PG0 (port)	MOSI0 (RSPI)	TO40 (ATU-IIIS)	—	—	—
PG1 (port)	MISO0 (RSPI)	TO41 (ATU-IIIS)	—	—	—
PG2 (port)	RSPCK0 (RSPI)	TO42 (ATU-IIIS)	—	—	—
PG3 (port)	—	TO43 (ATU-IIIS)	SSL00 (RSPI)	—	(IRQ7) (INTC)
PG4 (port)	IRQ2 (INTC)	TO44 (ATU-IIIS)	SSL01 (RSPI)	—	—

Table 18.9 Multiplexed Pin Functions for Port H

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PH0 (port)	DROD8 (DRO)	TO20 (ATU-IIIS)	DDC00 (DRI)	TIF0A (ATU-IIIS)	—
PH1 (port)	DROD9 (DRO)	TO21 (ATU-IIIS)	DDC01 (DRI)	TIF0B (ATU-IIIS)	—
PH2 (port)	DROD10 (DRO)	TO22 (ATU-IIIS)	DDC02 (DRI)	TIF1A (ATU-IIIS)	—
PH3 (port)	DROD11 (DRO)	TO23 (ATU-IIIS)	DDC03 (DRI)	TIF1B (ATU-IIIS)	—
PH4 (port)	DROD12 (DRO)	TO24 (ATU-IIIS)	DDC04 (DRI)	TIA00 (ATU-IIIS)	—
PH5 (port)	DROD13 (DRO)	TO25 (ATU-IIIS)	DDC05 (DRI)	TIA01 (ATU-IIIS)	—
PH6 (port)	DROD14 (DRO)	TO26 (ATU-IIIS)	DDC06 (DRI)	TIA02 (ATU-IIIS)	—
PH7 (port)	DROD15 (DRO)	(TO27) (ATU-IIIS)	DDC07 (DRI)	TIA03 (ATU-IIIS)	—
PH8 (port)	DROD0 (DRO)	(TO30) (ATU-IIIS)	DDC08 (DRI)	RTS2# (SCIF)	—
PH9 (port)	DROD1 (DRO)	(TO31) (ATU-IIIS)	DDC09 (DRI)	CTS2# (SCIF)	—
PH10 (port)	DROD2 (DRO)	(TO32) (ATU-IIIS)	DDC10 (DRI)	—	—
PH11 (port)	DROD3 (DRO)	(TO33) (ATU-IIIS)	DDC11 (DRI)	—	—
PH12 (port)	DROD4 (DRO)	TO34 (ATU-IIIS)	DDC12 (DRI)	—	—
PH13 (port)	DROD5 (DRO)	(TO35) (ATU-IIIS)	DDC13 (DRI)	—	—
PH14 (port)	DROD6 (DRO)	(TO36) (ATU-IIIS)	DDC14 (DRI)	—	IRQ1 (INTC)
PH15 (port)	DROD7 (DRO)	TO37 (ATU-IIIS)	DDC15 (DRI)	—	—

Table 18.10 Multiplexed Pin Functions for Port J

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PJ0 (port)	(CRX0) (CAN)	FRXA (FlexRay)	—	—	—
PJ1 (port)	(CTX0) (CAN)	FTXA (FlexRay)	—	—	—
PJ2 (port)	CRX1 (CAN)	FRXB (FlexRay)	—	—	—
PJ3 (port)	CTX1 (CAN)	FTXB (FlexRay)	—	RTS0# (SCIF)	—
PJ4 (port)	CRX2 (CAN)	FTXENA (FlexRay)	—	CTS0# (SCIF)	—
PJ5 (port)	CTX2 (CAN)	FTXENB (FlexRay)	—	SCK2 (SCIF)	—
PJ6 (port)	CRX3 (CAN)	TIF2A (ATU-IIIS)	—	RXD2 (SCIF)	TIA04 (ATU-IIIS)
PJ7 (port)	CTX3 (CAN)	TIF2B (ATU-IIIS)	—	TXD2 (SCIF)	—
PJ10 (port)	RXD0 (SCIF)	PWMOFF4 (ATU-IIIS)	—	AD0TRG# (ADC)	—
PJ11 (port)	TXD0 (SCIF)	—	—	AD0END (ADC)	—
PJ12 (port)	SCK0 (SCIF)	TCLKB (ATU-IIIS)	—	—	(IRQ0) (INTC)
PJ13 (port)	RXD1 (SCIF)	MISO1 (RSPI)	—	—	—
PJ14 (port)	TXD1 (SCIF)	MOSI1 (RSPI)	—	—	—
PJ15 (port)	SCK1 (SCIF)	PSPCK1 (RSPI)	—	—	—

Table 18.11 Multiplexed Pin Functions for Port K

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PK0 (port)	IRQ5 (INTC)	SSL10 (RSPI)	—	—	—
PK5 (port)	—	—	DINC4 (DRI)	RXD3 (SCIF)	—
PK6 (port)	—	—	—	TXD3 (SCIF)	—
PK8 (port)	DREQ2 (DMAC)	—	—	—	—
PK9 (port)	AUDRD0 (AUDR)	—	—	RTS3# (SCIF)	—
PK10 (port)	AUDRD1 (AUDR)	—	—	CTS3# (SCIF)	—
PK11 (port)	AUDRD2 (AUDR)	—	—	—	—
PK12 (port)	AUDRD3 (AUDR)	—	—	—	—
PK13 (port)	AUDRCLK (AUDR)	—	—	—	—
PK14 (port)	AUDRSYN# (AUDR)	—	—	—	—

Table 18.12 Multiplexed Pin Functions for Port L

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PL2 (port)	DROWR (DRO)	—	—	—	—
PL3 (port)	—	IRQ6 (INTC)	—	—	—
PL4 (port)	TIA10 (ATU-IIIS)	(TIF0A) (ATU-IIIS)	—	—	—
PL5 (port)	TIA11 (ATU-IIIS)	(TIF0B) (ATU-IIIS)	—	—	—
PL6 (port)	TIA12 (ATU-IIIS)	(TIF1A) (ATU-IIIS)	—	—	—
PL8 (port)	TIA14 (ATU-IIIS)	IRQ7 (INTC)	—	DREQ3 (DMAC)	—
PL9 (port)	TIA15 (ATU-IIIS)	—	—	—	AUDREVT# (AUDR)

Table 18.13 Multiplexed Pin Functions for Port M

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PM0 (port)	AD0IN0 (ADC)	—	—	—	—
PM2 (port)	AD0IN2 (ADC)	—	—	—	—
PM4 (port)	AD0IN4 (ADC)	—	—	—	—
PM6 (port)	AD0IN6 (ADC)	—	—	—	—
PM8 (port)	AD0IN8 (ADC)	—	—	—	—
PM9 (port)	AD0IN9 (ADC)	—	—	—	—
PM10 (port)	AD0IN10 (ADC)	—	—	—	—
PM11 (port)	AD0IN11 (ADC)	—	—	—	—
PM12 (port)	AD0IN12 (ADC)	—	—	—	—
PM13 (port)	AD0IN13 (ADC)	—	—	—	—
PM14 (port)	AD0IN14 (ADC)	—	—	—	—
PM15 (port)	AD0IN15 (ADC)	—	—	—	—

Table 18.14 Multiplexed Pin Functions for Port N

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)
PN0 (port)	AD1IN0 (ADC)	—	—	—	—
PN1 (port)	AD1IN1 (ADC)	—	—	—	—
PN4 (port)	AD1IN4 (ADC)	—	—	—	—
PN5 (port)	AD1IN5 (ADC)	—	—	—	—

Note: • The same function can be assigned to two separate pins. See table 18.15 for details.

Table 18.15 Pin Functions Allowed for Input/Output at Two Pins and Pin Assignments

Module	Signal Name	Pin Group A	Pin Group B	Notes
INTC	IRQ0	PC3/TO33/SSL20/IRQ0	PJ12/SCK0/TCLKB/IRQ0	*[1]
	IRQ1	—	PH14/DROD6/TO36/DDC14/IRQ1	
	IRQ2	PG4/IRQ2/TO44/SSL01	—	
	IRQ5	PK0/IRQ5/SSL10	—	
	IRQ6	PL3/IRQ6	PC0/TO30/MOSI2/IRQ6	
	IRQ7	PL8/TIA14/IRQ7/DREQ3	PG3/TO43/SSL00/IRQ7	
PSEL	PSLCLKB	PA8/TO10/DDB08/PSLCLKB	—	*[2]
	PSLCLKA	PA9/TO11/DDB09/PSLCLKA	—	
	PSLDATA0	PA10/TO12/DDB10/PSLDATA0	—	
	PSLDATA1	PA11/TO13/DDB11/PSLDATA1	—	
	PSLDATA2	PA12/TO14/DDB12/PSLDATA2	—	
	PSLDATA3	PA13/TO15/DDB13/PSLDATA3	—	
	PSLCLR	—	PE15/TO27/PSLCLR	
CAN	CRX0	PF0/CRX0	PJ0/CRX0/FRXA	*[1]
	CTX0	PF1/CTX0	PJ1/CTX0/FTXA	*[2]
	CRX1	—	PJ2/CRX1/FRXB	*[1]
	CTX1	—	PJ3/CTX1/FTXB/RTS0#	*[2]
	CRX2	—	PJ4/CRX2/FTXENA/CTS0#	*[1]
	CTX2	—	PJ5/CTX2/FTXENB/SCK2	*[2]
	CRX3	PJ6/CRX3/TIF2A/RXD2/TIA04	PF4/SDA/CRX3	*[1]
	CTX3	PJ7/CTX3/TIF2B/TXD2	PF5/SCL/CTX3	*[2]

Module	Signal Name	Pin Group A	Pin Group B	Notes
ATU-IIIS	TO20	—	PH0/DROD8/TO20/DDC00/TIF0A	*2
	TO21	—	PH1/DROD9/TO21/DDC01/TIF0B	
	TO22	—	PH2/DROD10/TO22/DDC02/TIF1A	
	TO23	—	PH3/DROD11/TO23/DDC03/TIF1B	
	TO24	—	PH4/DROD12/TO24/DDC04/TIA00	
	TO25	—	PH5/DROD13/TO25/DDC05/TIA01	
	TO26	—	PH6/DROD14/TO26/DDC06/TIA02	
	TO27	PE15/TO27/PSLCLR	PH7/DROD15/TO27/DDC07/TIA03	
	TO30	PC0/TO30/MOSI2/IRQ6	PH8/DROD0/TO30/DDC08/RTS2#	
	TO31	PC1/TO31/MISO2	PH9/DROD1/TO31/DDC09/CTS2#	
	TO32	PC2/TO32/RSPCK2/DREQ0	PH10/DROD2/TO32/DDC10	
	TO33	PC3/TO33/SSL20/IRQ0	PH11/DROD3/TO33/DDC11	
	TO34	—	PH12/DROD4/TO34/DDC12	
	TO35	PC5/TO35	PH13/DROD5/TO35/DDC13	
	TO36	PC6/CLKOUT/TO36	PH14/DROD6/TO36/DDC14/IRQ1	
	TO37	—	PH15/DROD7/TO37/DDC15	
	TIF0A	PH0/DRO08/TO20/DDC00/TIF0A	PL4/TIA10/TIF0A	*1
	TIF0B	PH1/DRO09/TO21/DDC01/TIF0B	PL5/TIA11/TIF0B	
	TIF1A	PH2/DRO10/TO22/DDC02/TIF1A	PL6/TIA12/TIF1A	
	TIF1B	PH3/DRO11/TO23/DDC03/TIF1B	—	

Notes: *1 If pin group A and pin group B have the same input function set, the setting for pin group A comes into effect so that input from Pin group A is accepted as input for the relevant input function.

*2 If pin group A and pin group B have the same output function set, the signal is output from both pins.

18.3 Register Descriptions

The I/O port registers are listed in table 18.16.

Table 18.16 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Port A data register	PADR	H'0000	H'FFFF 5002	8, 16	18-12
Port A I/O register	PAIOR	H'0000	H'FFFF 5006	8, 16	18-27
Port A driving ability setting register	PADSR	H'0000	H'FFFF 509A	8, 16	18-18
Port A control register 4	PACR4	H'0000	H'FFFF 5010	8, 16, 32	18-30
Port A control register 3	PACR3	H'0000	H'FFFF 5012	8, 16, 32	18-31
Port A control register 2	PACR2	H'0000	H'FFFF 5014	8, 16, 32	18-32
Port A control register 1	PACR1	H'0000	H'FFFF 5016	8, 16, 32	18-33
Port A port register	PAPR	Pin state	H'FFFF 501E	8, 16	18-15
Port B data register	PBDR	H'0000	H'FFFF 5102	8, 16	18-12
Port B I/O register	PBIOR	H'0000	H'FFFF 5106	8, 16	18-27
Port B control register 2	PBCR2	H'0000	H'FFFF 5114	8, 16, 32	18-34
Port B control register 1	PBCR1	H'0000	H'FFFF 5116	8, 16, 32	18-35
Port B port register	PBPR	Pin state	H'FFFF 511E	8, 16	18-15
Port B driving ability setting register	PBDSR	H'0000	H'FFFF 519A	8, 16	18-18
Port C data register	PCDR	H'0000	H'FFFF 5202	8, 16	18-12
Port C I/O register	PCIOR	H'0000	H'FFFF 5206	8, 16	18-28
Port C control register 4	PCCR4	H'0000	H'FFFF 5210	8, 16, 32	18-36
Port C control register 3	PCCR3	H'0000	H'FFFF 5212	8, 16, 32	18-36
Port C control register 2	PCCR2	H'0000	H'FFFF 5214	8, 16, 32	18-37
Port C control register 1	PCCR1	H'0000	H'FFFF 5216	8, 16, 32	18-38
Port C port register	PCPR	Pin state	H'FFFF 521E	8, 16	18-15
Port C driving ability setting register	PCDSR	H'0000	H'FFFF 529A	8, 16	18-18
Port ABC input threshold value switching register	PALVR	H'0000	H'FFFF 5300	8, 16	18-21
Port DRI input channel switching register	PDRIR	H'0000	H'FFFF 5340	8	18-70
Port D data register	PDDR	H'0000	H'FFFF 5402	8, 16	18-12
Port D I/O register	PDIOR	H'0000	H'FFFF 5406	8, 16	18-28
Port D control register 4	PDCR4	H'0000	H'FFFF 5410	8, 16, 32	18-39
Port D control register 3	PDCR3	H'0000	H'FFFF 5412	8, 16, 32	18-40
Port D control register 2	PDCR2	H'0000	H'FFFF 5414	8, 16, 32	18-41
Port D control register 1	PDCR1	H'0000	H'FFFF 5416	8, 16, 32	18-42
Port D port register	PDPR	Pin state	H'FFFF 541E	8, 16	18-15
Port D driving ability setting register	PDDSR	H'0000	H'FFFF 549A	8, 16	18-18
Port E data register	PEDR	H'0000	H'FFFF 5502	8, 16	18-12
Port E I/O register	PEIOR	H'0000	H'FFFF 5506	8, 16	18-28
Port E control register 4	PECR4	H'0000	H'FFFF 5510	8, 16, 32	18-43
Port E control register 3	PECR3	H'0000	H'FFFF 5512	8, 16, 32	18-43

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Port E port register	PEPR	Pin state	H'FFFF 551E	8, 16	18-15
Port E driving ability setting register	PEDSR	H'0000	H'FFFF 559A	8, 16	18-18
Port F data register	PFDR	H'0000	H'FFFF 5602	8, 16	18-12
Port F I/O register	PFIOR	H'0000	H'FFFF 5606	8, 16	18-28
Port F control register 2	PFCR2	H'0000	H'FFFF 5614	8, 16, 32	18-44
Port F control register 1	PFCR1	H'0000	H'FFFF 5616	8, 16, 32	18-45
Port F port register	PFPR	Pin state	H'FFFF 561E	8, 16	18-15
Port F driving ability setting register	PFDSR	H'0000	H'FFFF 569A	8, 16	18-18
Port DEF input threshold value switching register	PDLVR	H'0000	H'FFFF 5700	8, 16	18-23
Port G data register	PGDR	H'0000	H'FFFF 5802	8, 16	18-13
Port G I/O register	PGIOR	H'0000	H'FFFF 5806	8, 16	18-28
Port G control register 2	PGCR2	H'0000	H'FFFF 5814	8, 16, 32	18-46
Port G control register 1	PGCR1	H'0000	H'FFFF 5816	8, 16, 32	18-47
Port G port register	PGPR	Pin state	H'FFFF 581E	8, 16	18-15
Port G driving ability setting register	PGDSR	H'0000	H'FFFF 589A	8, 16	18-18
Port H data register	PHDR	H'0000	H'FFFF 5902	8, 16	18-13
Port H I/O register	PHIOR	H'0000	H'FFFF 5906	8, 16	18-28
Port H control register 4	PHCR4	H'0000	H'FFFF 5910	8, 16, 32	18-48
Port H control register 3	PHCR3	H'0000	H'FFFF 5912	8, 16, 32	18-49
Port H control register 2	PHCR2	H'0000	H'FFFF 5914	8, 16, 32	18-50
Port H control register 1	PHCR1	H'0000	H'FFFF 5916	8, 16, 32	18-51
Port H port register	PHPR	Pin state	H'FFFF 591E	8, 16	18-16
Port H driving ability setting register	PHDSR	H'0000	H'FFFF 599A	8, 16	18-19
Port J data register	PJDR	H'0000	H'FFFF 5A02	8, 16	18-13
Port J I/O register	PJIOR	H'0000	H'FFFF 5A06	8, 16	18-28
Port J control register 4	PJCR4	H'0000	H'FFFF 5A10	8, 16, 32	18-53
Port J control register 3	PJCR3	H'0000	H'FFFF 5A12	8, 16, 32	18-54
Port J control register 2	PJCR2	H'0000	H'FFFF 5A14	8, 16, 32	18-55
Port J control register 1	PJCR1	H'0000	H'FFFF 5A16	8, 16, 32	18-56
Port J port register	PJPR	Pin state	H'FFFF 5A1E	8, 16	18-16
Port J driving ability setting register	PJDSR	H'0000	H'FFFF 5A9A	8, 16	18-19
Port GHJ input threshold value switching register	PGLVR	H'0000	H'FFFF 5B00	8, 16	18-25
Port K data register	PKDR	H'0000	H'FFFF 5C02	8, 16	18-13
Port K I/O register	PKIOR	H'0000	H'FFFF 5C06	8, 16	18-28
Port K control register 4	PKCR4	H'0000	H'FFFF 5C10	8, 16, 32	18-57
Port K control register 3	PKCR3	H'0000	H'FFFF 5C12	8, 16, 32	18-58
Port K control register 2	PKCR2	H'0000	H'FFFF 5C14	8, 16, 32	18-59
Port K control register 1	PKCR1	H'0000	H'FFFF 5C16	8, 16, 32	18-60
Port K port register	PKPR	Pin state	H'FFFF 5C1E	8, 16	18-16
Port K driving ability setting register	PKDSR	H'0000	H'FFFF 5C9A	8, 16	18-19

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Port L data register	PLDR	H'0000	H'FFFF 5D02	8, 16	18-13
Port L I/O register	PLIOR	H'0000	H'FFFF 5D06	8, 16	18-29
Port L control register 3	PLCR3	H'0000	H'FFFF 5D12	8, 16	18-61
Port L control register 2	PLCR2	H'0000	H'FFFF 5D14	8, 16, 32	18-62
Port L control register 1	PLCR1	H'0000	H'FFFF 5D16	8, 16, 32	18-63
Port L port register	PLPR	Pin state	H'FFFF 5D1E	8, 16	18-16
Port L driving ability setting register	PLDSR	H'0000	H'FFFF 5D9A	8, 16	18-19
Port KL input threshold value switching register	PKLVR	H'0009	H'FFFF 5E00	8, 16	18-26
Port M control register 4	PMCR4	H'1111	H'FFFF 5E10	8, 16, 32	18-64
Port M control register 3	PMCR3	H'1111	H'FFFF 5E12	8, 16, 32	18-65
Port M control register 2	PMCR2	H'1111	H'FFFF 5E14	8, 16, 32	18-66
Port M control register 1	PMCR1	H'1111	H'FFFF 5E16	8, 16, 32	18-67
Port M port register	PMPR	H'0000	H'FFFF 5E1E	8, 16	18-16
Port N control register 2	PNCR2	H'1111	H'FFFF 5F14	8, 16, 32	18-68
Port N control register 1	PNCR1	H'1111	H'FFFF 5F16	8, 16, 32	18-69
Port N port register	PNPR	H'0000	H'FFFF 5F1E	8, 16	18-16

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

18.3.1 Port A to H and J to L Data Registers (PADR to PHDR and PJDR to PLDR)

The PADR to PHDR and PJDR to PLDR register store input and output data corresponding to each port.

When a pin functions as a general output, the value written to the corresponding register is output on the pin. The register value can be read directly regardless of the pin state by reading the register.

When a pin functions as a general input, reading the corresponding register returns the pin state directly rather than the register value. Also, a value can be written to the corresponding register but does not affect the pin state.

Table 18.17 lists the read and write operations.

Port A Data Regoster (PADR) <P4 address: location H'FFFF 5002>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PA13 DR	PA12 DR	PA11 DR	PA10 DR	PA9 DR	PA8 DR	PA7 DR	PA6 DR	PA5 DR	PA4 DR	PA3 DR	PA2 DR	PA1 DR	PA0 DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port B Data Register (PBDR) <P4 address: location H'FFFF 5102>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PB3 DR	—	PB1 DR	PB0 DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port C Data Register (PCDR) <P4 address: location H'FFFF 5202>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PC14 DR	—	—	—	—	—	—	—	PC6 DR	PC5 DR	—	PC3 DR	PC2 DR	PC1 DR	PC0 DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port D Data Register (PDDR) <P4 address: location H'FFFF 5402>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PD10 DR	PD9 DR	PD8 DR	PD7 DR	PD6 DR	PD5 DR	PD4 DR	PD3 DR	PD2 DR	PD1 DR	PD0 DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port E Data Register (PEDR) <P4 address: location H'FFFF 5502>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15 DR	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port F Data Register (PFDR) <P4 address: location H'FFFF 5602>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PF5 DR	PF4 DR	—	—	PF1 DR	PF0 DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port G Data Register (PGDR) <P4 address: location H'FFFF 5802>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PG4DR	PG3DR	PG2DR	PG1DR	PG0DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port H Data Register (PHDR) <P4 address: location H'FFFF 5902>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH15DR	PH14DR	PH13DR	PH12DR	PH11DR	PH10DR	PH9DR	PH8DR	PH7DR	PH6DR	PH5DR	PH4DR	PH3DR	PH2DR	PH1DR	PH0DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port J Data Register (PJDR) <P4 address: location H'FFFF 5A02>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ15DR	PJ14DR	PJ13DR	PJ12DR	PJ11DR	PJ10DR	—	—	PJ7DR	PJ6DR	PJ5DR	PJ4DR	PJ3DR	PJ2DR	PJ1DR	PJ0DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port K Data Register (PKDR) <P4 address: location H'FFFF 5C02>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PK14DR	PK13DR	PK12DR	PK11DR	PK10DR	PK9DR	PK8DR	—	PK6DR	PK5DR	—	—	—	—	PK0DR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port L Data Register (PLDR) <P4 address: location H'FFFF 5D02>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PL9DR	PL8DR	—	PL6DR	PL5DR	PL4DR	PL3DR	PL2DR	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	Pm15DR	0	R	W	Refer to table 18.17.
14	Pm14DR	0	R	W	
13	Pm13DR	0	R	W	
12	Pm12DR	0	R	W	
11	Pm11DR	0	R	W	
10	Pm10DR	0	R	W	
9	Pm9DR	0	R	W	
8	Pm8DR	0	R	W	
7	Pm7DR	0	R	W	
6	Pm6DR	0	R	W	
5	Pm5DR	0	R	W	
4	Pm4DR	0	R	W	
3	Pm3DR	0	R	W	
2	Pm2DR	0	R	W	
1	Pm1DR	0	R	W	
0	Pm0DR	0	R	W	

Note: • Following port bits are reserved and nothing is assigned. These bits are always read as "0". The write value should always be "0".

Port A: Bits 15, 14

Port B: Bits 15 to 4, and 2

Port C: Bits 15, 13 to 7, and 4

Port D: Bits 15 to 11

Port E: Bits 14 to 0

Port F: Bits 15 to 6, 3, and 2

Port G: Bits 15 to 5

Port J: Bits 9, 8

Port K: Bits 15, 7, and 4 to 1

Port L: Bits 15 to 10, 7, 1, and 0

Legend: m = A to H and J to L

Table 18.17 Read and Write Operations of Port A to H and J to L Data Registers (PADR to PHDR and PJDR to PLDR)

PmIOR Register Bit Value	Pin Function	Read	Write
0	General input	Pin state	Data can be written to the PmDR register but data does not affect the pin state.
	Other than general input	Pin state	Data can be written to the PmDR register but data does not affect the pin state.
1	General output	PmDR register value	Data written to the PmDR register is output on the pin.
	Other than general output	PmDR register value	Data can be written to the PmDR register but data does not affect the pin state.

Legend: m = A to H and J to L

18.3.2 Port A to H and J to N Port Registers (PAPR to PHPR and PJPR to PNPR)

The PAPR to PHPR and PJPR to PNPR register always store the port pin states, so it cannot be written to directly by the CPU.

Ports M and N are input-only. When the analog input pin for the A/D converter is selected, these functions cannot be used.

Port A Port Register (PAPR) <P4 address: location H'FFFF 501E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PA13 PR	PA12 PR	PA11 PR	PA10 PR	PA9 PR	PA8 PR	PA7 PR	PA6 PR	PA5 PR	PA4 PR	PA3 PR	PA2 PR	PA1 PR	PA0 PR
After Reset:	?	?	PA13	PA12	PA11	PA10	PA9	PA8	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0

Port B Port Register (PBPR) <P4 address: location H'FFFF 511E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PB3 PR	—	PB1 PR	PB0 PR
After Reset:	0	0	0	0	0	0	0	0	0	?	?	?	PB3	?	PB1	PB0

Port C Port Register (PCPR) <P4 address: location H'FFFF 521E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PC14 PR	—	—	—	—	—	—	—	PC6 PR	PC5 PR	—	PC3 PR	PC2 PR	PC1 PR	PC0 PR
After Reset:	?	PC14	?	?	?	?	?	?	?	PC6	PC5	?	PC3	PC2	PC1	PC0

Port D Port Register (PDPR) <P4 address: location H'FFFF 541E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PD10 PR	PD9 PR	PD8 PR	PD7 PR	PD6 PR	PD5 PR	PD4 PR	PD3 PR	PD2 PR	PD1 PR	PD0 PR
After Reset:	?	?	?	?	?	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0

Port E Port Register (PEPR) <P4 address: location H'FFFF 551E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15 PR	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	PE15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Port F Port Register (PFPR) <P4 address: location H'FFFF 561E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PF5 PR	PF4 PR	—	—	PF1 PR	PF0 PR
After Reset:	0	0	0	0	0	0	0	0	0	0	PF5	PF4	?	?	PF1	PF0

Port G Port Register (PGPR) <P4 address: location H'FFFF 581E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PG4 PR	PG3 PR	PG2 PR	PG1 PR	PG0 PR
After Reset:	0	0	0	0	0	0	0	0	?	?	?	PG4	PG3	PG2	PG1	PG0

Port H Port Register (PHPR) <P4 address: location H'FFFF 591E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH15 PR	PH14 PR	PH13 PR	PH12 PR	PH11 PR	PH10 PR	PH9 PR	PH8 PR	PH7 PR	PH6 PR	PH5 PR	PH4 PR	PH3 PR	PH2 PR	PH1 PR	PH0 PR
After Reset:	PH15	PH14	PH13	PH12	PH11	PH10	PH9	PH8	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0

Port J Port Register (PJPR) <P4 address: location H'FFFF 5A1E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ15 PR	PJ14 PR	PJ13 PR	PJ12 PR	PJ11 PR	PJ10 PR	—	—	PJ7 PR	PJ6 PR	PJ5 PR	PJ4 PR	PJ3 PR	PJ2 PR	PJ1 PR	PJ0 PR
After Reset:	PJ15	PJ14	PJ13	PJ12	PJ11	PJ10	?	?	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0

Port K Port Register (PKPR) <P4 address: location H'FFFF 5C1E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PK14 PR	PK13 PR	PK12 PR	PK11 PR	PK10 PR	PK9 PR	PK8 PR	—	PK6 PR	PK5 PR	—	—	—	—	PK0 PR
After Reset:	0	PK14	PK13	PK12	PK11	PK10	PK9	PK8	?	PK6	PK5	?	?	?	?	PK0

Port L Port Register (PLPR) <P4 address: location H'FFFF 5D1E>

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PL9 PR	PL8 PR	—	PL6 PR	PL5 PR	PL4 PR	PL3 PR	PL2 PR	—	—
After Reset:	0	0	0	0	0	0	PL9	PL8	?	PL6	PL5	PL4	PL3	PL2	?	?

Port M Port Register (PMPR) <P4 address: location H'FFFF 5E1E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PM15 PR	PM14 PR	PM13 PR	PM12 PR	PM11 PR	PM10 PR	PM9 PR	PM8 PR	—	PM6 PR	—	PM4 PR	—	PM2 PR	—	PM0 PR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port N Port Register (PNPR) <P4 address: location H'FFFF 5F1E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PN5 PR	PN4 PR	—	—	PN1 PR	PN0 PR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: Pin state>

Bit	Abbreviation	After Reset	R	W	Description
15	Pm15PR	Pin state	R	—	When these bits are read, the pin states can be read.
14	Pm14PR	Pin state	R	—	
13	Pm13PR	Pin state	R	—	
12	Pm12PR	Pin state	R	—	
11	Pm11PR	Pin state	R	—	
10	Pm10PR	Pin state	R	—	
9	Pm9PR	Pin state	R	—	
8	Pm8PR	Pin state	R	—	
7	Pm7PR	Pin state	R	—	
6	Pm6PR	Pin state	R	—	
5	Pm5PR	Pin state	R	—	
4	Pm4PR	Pin state	R	—	
3	Pm3PR	Pin state	R	—	
2	Pm2PR	Pin state	R	—	
1	Pm1PR	Pin state	R	—	
0	Pm0PR	Pin state	R	—	

Notes:
- Following port bits are reserved and nothing is assigned.
  Port A: Bits 15, 14
  Port B: Bits 15 to 4, and 2
  Port C: Bits 15, 13 to 7, and 4
  Port D: Bits 15 to 11
  Port E: Bits 14 to 0
  Port F: Bits 15 to 6, 3, and 2
  Port G: Bits 15 to 5
  Port J: Bits 9, 8
  Port K: Bits 15, 7, and 4 to 1
  Port L: Bits 15 to 10, 7, 1, and 0
- The value of ports M and N is set to "0" as the analog input pin for the A/D converter is selected after a reset.

Legend: m = A to H and J to N

18.3.3 Port A to H and J to L Driving Ability Setting Registers (PADSR to PHDSR and PJDSR to PLDSR)

The PADSR to PHDSR and PJDSR to PLDSR register are used to specify the drive capacity of the port pins. The settings of these registers are valid regardless of the selected pin functions.

However, the settings of the PFDSR register have no effect when the PF4 pin is set to SDA output (IIC3) or the PF5 pin is set to SCL output (IIC3).

Port A Driving Ability Setting Register (PADSR) <P4 address: location H'FFFF 509A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PA13 DSR	PA12 DSR	PA11 DSR	PA10 DSR	PA9 DSR	PA8 DSR	PA7 DSR	PA6 DSR	PA5 DSR	PA4 DSR	PA3 DSR	PA2 DSR	PA1 DSR	PA0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port B Driving Ability Setting Register (PBDSR) <P4 address: location H'FFFF 519A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PB3 DSR	—	PB1 DSR	PB0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port C Driving Ability Setting Register (PCDSR) <P4 address: location H'FFFF 529A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PC14 DSR	—	—	—	—	—	—	—	PC6 DSR	PC5 DSR	—	PC3 DSR	PC2 DSR	PC1 DSR	PC0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port D Driving Ability Setting Register (PDDSR) <P4 address: location H'FFFF 549A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PD10 DSR	PD9 DSR	PD8 DSR	PD7 DSR	PD6 DSR	PD5 DSR	PD4 DSR	PD3 DSR	PD2 DSR	PD1 DSR	PD0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port E Driving Ability Setting Register (PEDSR) <P4 address: location H'FFFF 559A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15 DSR	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port F Driving Ability Setting Register (PFDSR) <P4 address: location H'FFFF 569A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PF5 DSR	PF4 DSR	—	—	PF1 DSR	PF0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port G Driving Ability Setting Register (PGDSR) <P4 address: location H'FFFF 589A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PG4 DSR	PG3 DSR	PG2 DSR	PG1 DSR	PG0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port H Driving Ability Setting Register (PHDSR) <P4 address: location H'FFFF 599A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH15 DSR	PH14 DSR	PH13 DSR	PH12 DSR	PH11 DSR	PH10 DSR	PH9 DSR	PH8 DSR	PH7 DSR	PH6 DSR	PH5 DSR	PH4 DSR	PH3 DSR	PH2 DSR	PH1 DSR	PH0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port J Driving Ability Setting Register (PJDSR) <P4 address: location H'FFFF 5A9A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ15 DSR	PJ14 DSR	PJ13 DSR	PJ12 DSR	PJ11 DSR	PJ10 DSR	—	—	PJ7 DSR	PJ6 DSR	PJ5 DSR	PJ4 DSR	PJ3 DSR	PJ2 DSR	PJ1 DSR	PJ0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port K Driving Ability Setting Register (PKDSR) <P4 address: location H'FFFF 5C9A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PK14 DSR	PK13 DSR	PK12 DSR	PK11 DSR	PK10 DSR	PK9 DSR	PK8 DSR	—	PK6 DSR	PK5 DSR	—	—	—	—	PK0 DSR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port L Driving Ability Setting Register (PLDSR) <P4 address: location H'FFFF 5D9A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PL9 DSR	PL8 DSR	—	PL6 DSR	PL5 DSR	PL4 DSR	PL3 DSR	PL2 DSR	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	Pm15DSR	0	R	W	When these bits are set to "1", the driving ability of the corresponding pin is higher than normal. 0: Normal output 1: Drive capacity of output pin is increased
14	Pm14DSR	0	R	W	
13	Pm13DSR	0	R	W	
12	Pm12DSR	0	R	W	
11	Pm11DSR	0	R	W	
10	Pm10DSR	0	R	W	
9	Pm9DSR	0	R	W	
8	Pm8DSR	0	R	W	
7	Pm7DSR	0	R	W	
6	Pm6DSR	0	R	W	
5	Pm5DSR	0	R	W	
4	Pm4DSR	0	R	W	
3	Pm3DSR	0	R	W	
2	Pm2DSR	0	R	W	
1	Pm1DSR	0	R	W	
0	Pm0DSR	0	R	W	

Note: • Following port bits are reserved and nothing is assigned. These bits are always read as "0". The write value should always be "0".

Port A: Bits 15, 14

Port B: Bits 15 to 4, and 2

Port C: Bits 15, 13 to 7, and 4

Port D: Bits 15 to 11

Port E: Bits 14 to 0

Port F: Bits 15 to 6, 3, and 2

Port G: Bits 15 to 5

Port J: Bits 9, 8

Port K: Bits 15, 7, and 4 to 1

Port L: Bits 15 to 10, 7, 1, and 0

Legend: m = A to H and J to L

18.3.4 Port ABC Input Threshold Value Switching Register (PALVR)

The PALVR register is used to specify the input threshold values for ports A, B, and C, in port group units, from among three voltage levels (and to enable or disable Schmitt input). Bits 11 to 8 in the PALVR register correspond to port C pins; bits 7 to 4 correspond to port B pins; and bits 3 to 0 correspond to port A pins, respectively. The setting of the PALVR register is valid regardless of the selected pin functions.

After a reset the pins are in the input prohibited state, so it is necessary to set the PmPIEN (port m input level setting) bit to "1" to perform input processing. The PCPIEN bit corresponds to port C, the PBPIEN bit to port B, and the PAPIEN bit to port A.

When the port input is disabled, the pins are in a state equivalent to when a "H" level signal is being input. If the peripheral input function is selected by the port control register in the input prohibited state, the input of a "H" level signal input may cause unintended operation. The setting sequence for selecting the peripheral input function is shown below.

1. Enable the port input after the pin level has been set.

2. Select the pin function with the setting of the port control register.

Port ABC Input Threshold Value Switching Register (PALVR) <P4 address: location H'FFFF 5300>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PC PIEN	PC SCSEL	PC SEL0	PC SEL1	PB PIEN	PB SCSEL	PB SEL0	PB SEL1	PA PIEN	PA SCSEL	PA SEL0	PA SEL1
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	PCPIEN	0	R	W	Port C Input Level Setting Bits 0xxx: Input prohibited state 1000: CMOS input, 0.35 Vcc 1001: CMOS input, 0.50 Vcc 1010: Setting prohibited 1011: CMOS input, 0.70 Vcc 1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc 1101: Setting prohibited 1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc 1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc
10	PCSCSEL	0	R	W	
9	PCSEL0	0	R	W	
8	PCSEL1	0	R	W	

Bit	Abbreviation	After Reset	R	W	Description
7	PBPIEN	0	R	W	Port B Input Level Setting Bits
6	PBSCSEL	0	R	W	0xxx: Input prohibited state
5	PBSEL0	0	R	W	1000: CMOS input, 0.35 Vcc
4	PBSEL1	0	R	W	1001: CMOS input, 0.50 Vcc 1010: Setting prohibited 1011: CMOS input, 0.70 Vcc 1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc 1101: Setting prohibited 1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc 1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc
3	PAPIEN	0	R	W	Port A Input Level Setting Bits
2	PASCSEL	0	R	W	0xxx: Input prohibited state
1	PASEL0	0	R	W	1000: CMOS input, 0.35 Vcc
0	PASEL1	0	R	W	1001: CMOS input, 0.50 Vcc 1010: Setting prohibited 1011: CMOS input, 0.70 Vcc 1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc 1101: Setting prohibited 1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc 1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc

18.3.5 Port DEF Input Threshold Value Switching Register (PDLVR)

The PDLVR register is used to specify the input threshold values for ports D, E, and F, in port group units, from among three voltage levels (and to enable or disable Schmitt input). Bits 11 to 8 in the PDLVR register correspond to port F pins; bits 7 to 4 correspond to port E pins; and bits 3 to 0 correspond to port D pins, respectively. The setting of the PDLVR register is valid regardless of the selected pin functions.

After a reset the pins are in the input prohibited state, so it is necessary to set the PmPIEN (port m input level setting) bit to "1" to perform input processing.

When the port input is disabled, the pins are in a state equivalent to when a "H" level signal is being input. If the peripheral input function is selected by the port control register in the input prohibited state, the input of a "H" level signal input may cause unintended operation. The setting sequence for selecting the peripheral input function is shown below.

1. Enable the port input after the pin level has been set.

2. Select the pin function with the setting of the port control register.

Port DEF Input Threshold Value Switching Register (PDLVR) <P4 address: location H'FFFF 5700>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PF PIEN	PF SCSEL	PF SEL0	PF SEL1	PE PIEN	PE SCSEL	PE SEL0	PE SEL1	PD PIEN	PD SCSEL	PD SEL0	PD SEL1
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	PFPIEN	0	R	W	Port F Input Level Setting Bits*[1] 0xxx: Input prohibited state 1000: CMOS input, 0.35 Vcc 1001: CMOS input, 0.50 Vcc 1010: Setting prohibited 1011: CMOS input, 0.70 Vcc 1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc 1101: Setting prohibited 1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc 1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc Note: *1 When SDA and SCL are selected as the pin functions of PF4 and PF5 in the PFCR2 register, the SDA and SCL input threshold values are fixed at 0.7 Vcc and 0.3 Vcc, respectively. When PF4 and PF5 are used as SDA and SCL, set these bits to a value of "1xxx" which is to say any value other than an input prohibited setting.
10	PFSCSEL	0	R	W	
9	PFSEL0	0	R	W	
8	PFSEL1	0	R	W	

Bit	Abbreviation	After Reset	R	W	Description
7	PEPIEN	0	R	W	Port E Input Level Setting Bits
6	PESCSEL	0	R	W	0xxx: Input prohibited state
5	PESEL0	0	R	W	1000: CMOS input, 0.35 Vcc
4	PESEL1	0	R	W	1001: CMOS input, 0.50 Vcc
					1010: Setting prohibited
					1011: CMOS input, 0.70 Vcc
					1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc
					1101: Setting prohibited
					1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc
					1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc
3	PDPIEN	0	R	W	Port D Input Level Setting Bits
2	PDSCSEL	0	R	W	0xxx: Input prohibited state
1	PDSEL0	0	R	W	1000: CMOS input, 0.35 Vcc
0	PDSEL1	0	R	W	1001: CMOS input, 0.50 Vcc
					1010: Setting prohibited
					1011: CMOS input, 0.70 Vcc
					1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc
					1101: Setting prohibited
					1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc
					1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc

18.3.6 Port GHJ Input Threshold Value Switching Register (PGLVR)

The PGLVR register is used to specify the input threshold values for ports G, H, and J, in port group units, from among three voltage levels (and to enable or disable Schmitt input). Bits 11 to 8 in the PGLVR register correspond to port J pins; bits 7 to 4 correspond to port H pins; and bits 3 to 0 correspond to port G pins, respectively. No bits in this register correspond to pins PG0, PG1, PG2, PG3, PJ1, PJ3, PJ4, and PJ5. The setting of the PGLVR register is valid regardless of the selected pin functions.

After a reset the pins are in the input prohibited state, so it is necessary to set the PmPIEN (port m input level setting) bit to "1" to perform input processing. Pins PG0, PG1, PG2, PG3, PJ1, PJ3, PJ4, and PJ5 can be used as inputs without the need to make settings to the PGLVR register.

When the port input is disabled, the pins are in a state equivalent to when a "H" level signal is being input. If the peripheral input function is selected by the port control register in the input prohibited state, the input of a "H" level signal input may cause unintended operation. The setting sequence for selecting the peripheral input function is shown below.

1. Enable the port input after the pin level has been set.

2. Select the pin function with the setting of the port control register.

Port GHJ Input Threshold Value Switching Register (PGLVR) <P4 address: location H'FFFF 5B00>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PJ PIEN	PJ SCSEL	PJ SEL0	PJ SEL1	PH PIEN	PH SCSEL	PH SEL0	PH SEL1	PG PIEN	PG SCSEL	PG SEL0	PG SEL1
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	PJPIEN	0	R	W	Port J Input Level Setting Bits
10	PJSCSEL	0	R	W	0xxx: Input prohibited state
9	PJSEL0	0	R	W	1000: CMOS input, 0.35 Vcc
8	PJSEL1	0	R	W	1001: CMOS input, 0.50 Vcc 1010: Setting prohibited 1011: CMOS input, 0.70 Vcc 1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc 1101: Setting prohibited 1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc 1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc

Bit	Abbreviation	After Reset	R	W	Description
7	PHPIEN	0	R	W	Port H Input Level Setting Bits
6	PHSCSEL	0	R	W	0xxx: Input prohibited state
5	PHSEL0	0	R	W	1000: CMOS input, 0.35 Vcc
4	PHSEL1	0	R	W	1001: CMOS input, 0.50 Vcc
					1010: Setting prohibited
					1011: CMOS input, 0.70 Vcc
					1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc
					1101: Setting prohibited
					1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc
					1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc
3	PGPIEN	0	R	W	Port G Input Level Setting Bits
2	PGSCSEL	0	R	W	0xxx: Input prohibited state
1	PGSEL0	0	R	W	1000: CMOS input, 0.35 Vcc
0	PGSEL1	0	R	W	1001: CMOS input, 0.50 Vcc
					1010: Setting prohibited
					1011: CMOS input, 0.70 Vcc
					1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc
					1101: Setting prohibited
					1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc
					1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc

18.3.7 Port KL Input Threshold Value Switching Register (PKLVR)

The PKLVR register is used to specify the input threshold values for ports K and L, in port group units, from among three voltage levels (and to enable or disable Schmitt input). Bits 7 to 4 in the PKLVR register correspond to port L pins and bits 3 to 0 correspond to port K pins. The setting of the PKLVR register is valid regardless of the selected pin functions.

After a reset the port L pins are in the input prohibited state, so it is necessary to set the PmPIEN (port m input level setting) bit to "1" to perform input processing. The port K pins, in contrast, are in the input enabled state (CMOS input, 0.50 Vcc) after a reset.

When the port input is disabled, the pins are in a state equivalent to when a "H" level signal is being input. If the peripheral input function is selected by the port control register in the input prohibited state, the input of a "H" level signal input may cause unintended operation. The setting sequence for selecting the peripheral input function is shown below.

1. Enable the port input after the pin level has been set.

2. Select the pin function with the setting of the port control register.

Port KL Input Threshold Value Switching Register (PKLVR) <P4 address: location H'FFFF 5E00>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PL PIEN	PL SCSEL	PL SEL0	PL SEL1	PK PIEN	PK SCSEL	PK SEL0	PK SEL1
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

<After Reset: H'0009>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7	PLPIEN	0	R	W	Port L Input Level Setting Bits
6	PLSCSEL	0	R	W	0xxx: Input prohibited state
5	PLSEL0	0	R	W	1000: CMOS input, 0.35 Vcc
4	PLSEL1	0	R	W	1001: CMOS input, 0.50 Vcc 1010: Setting prohibited 1011: CMOS input, 0.70 Vcc 1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc 1101: Setting prohibited 1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc 1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc
3	PKPIEN	1	R	W	Port K Input Level Setting Bits
2	PKSCSEL	0	R	W	0xxx: Input prohibited state
1	PKSEL0	0	R	W	1000: CMOS input, 0.35 Vcc
0	PKSEL1	1	R	W	1001: CMOS input, 0.50 Vcc 1010: Setting prohibited 1011: CMOS input, 0.70 Vcc 1100: Schmitt input, VT+ = 0.50 Vcc, VT– = 0.35 Vcc 1101: Setting prohibited 1110: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.35 Vcc 1111: Schmitt input, VT+ = 0.70 Vcc, VT– = 0.50 Vcc

18.3.8 Port A to H and J to L I/O Registers (PAIOR to PHIOR and PJIOR to PLIOR)

The PAIOR to PHIOR and PJIOR to PLIOR registers are used to set the I/O direction of the port pins.

These registers are enabled only when the port pins function as general I/O pins. Otherwise, the set values of these registers have no effect on the pin status.

Port A I/O Register (PAIOR) <P4 address: location H'FFFF 5006>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PA15 IOR	PA14 IOR	PA13 IOR	PA12 IOR	PA11 IOR	PA10 IOR	PA9 IOR	PA8 IOR	PA7 IOR	PA6 IOR	PA5 IOR	PA4 IOR	PA3 IOR	PA2 IOR	PA1 IOR	PA0 IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port B I/O Register (PBIOR) <P4 address: location H'FFFF 5106>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PB6 IOR	PB5 IOR	PB4 IOR	PB3 IOR	PB2 IOR	PB1 IOR	PB0 IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port C I/O Register (PCIOR) <P4 address: location H'FFFF 5206>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PC15IOR	PC14IOR	PC13IOR	PC12IOR	PC11IOR	PC10IOR	PC9IOR	PC8IOR	PC7IOR	PC6IOR	PC5IOR	PC4IOR	PC3IOR	PC2IOR	PC1IOR	PC0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port D I/O Register (PDIOR) <P4 address: location H'FFFF 5406>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15IOR	PD14IOR	PD13IOR	PD12IOR	PD11IOR	PD10IOR	PD9IOR	PD8IOR	PD7IOR	PD6IOR	PD5IOR	PD4IOR	PD3IOR	PD2IOR	PD1IOR	PD0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port E I/O Register (PEIOR) <P4 address: location H'FFFF 5506>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15IOR	PE14IOR	PE13IOR	PE12IOR	PE11IOR	PE10IOR	PE9IOR	PE8IOR	PE7IOR	PE6IOR	PE5IOR	PE4IOR	PE3IOR	PE2IOR	PE1IOR	PE0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port F I/O Register (PFIOR) <P4 address: location H'FFFF 5606>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PF5IOR	PF4IOR	PF3IOR	PF2IOR	PF1IOR	PF0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port G I/O Register (PGIOR) <P4 address: location H'FFFF 5806>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PG7IOR	PG6IOR	PG5IOR	PG4IOR	PG3IOR	PG2IOR	PG1IOR	PG0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port H I/O Register (PHIOR) <P4 address: location H'FFFF 5906>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH15IOR	PH14IOR	PH13IOR	PH12IOR	PH11IOR	PH10IOR	PH9IOR	PH8IOR	PH7IOR	PH6IOR	PH5IOR	PH4IOR	PH3IOR	PH2IOR	PH1IOR	PH0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port J I/O Register (PJIOR) <P4 address: location H'FFFF 5A06>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ15IOR	PJ14IOR	PJ13IOR	PJ12IOR	PJ11IOR	PJ10IOR	PJ9IOR	PJ8IOR	PJ7IOR	PJ6IOR	PJ5IOR	PJ4IOR	PJ3IOR	PJ2IOR	PJ1IOR	PJ0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port K I/O Register (PKIOR) <P4 address: location H'FFFF 5C06>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PK14IOR	PK13IOR	PK12IOR	PK11IOR	PK10IOR	PK9IOR	PK8IOR	PK7IOR	PK6IOR	PK5IOR	PK4IOR	PK3IOR	PK2IOR	PK1IOR	PK0IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Port L I/O Register (PLIOR) <P4 address: location H'FFFF 5D06>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PL9 IOR	PL8 IOR	PL7 IOR	PL6 IOR	PL5 IOR	PL4 IOR	PL3 IOR	PL2 IOR	PL1 IOR	PL0 IOR
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	Pm15IOR to Pm0IOR	All 0	R	W	The Pm15IOR to Pm0IOR bits correspond to the Pm15 to Pm0 pins respectively (the multiplex pin names other than the port names are omitted from pin names). Setting these bits specifies the I/O direction of the corresponding pins. 0: The corresponding pin is set to input. 1: The corresponding pin is set to output.

Note:
- After a reset is canceled, set the following reserved bits to "1". For details, see section 18.4, I/O Port Initial Setting Procedure Examples.

  Port A: Bits PA15IOR and PA14IOR

  Port B: Bits PB6IOR to PB4IOR and PB2IOR

  Port C: Bits PC15IOR, PC13IOR to PC7IOR, and PC4IOR

  Port D: Bits PD15IOR to PD11IOR

  Port E: Bits PE14IOR to PE0IOR

  Port F: Bits PF3IOR and PF2IOR

  Port G: Bits PG7IOR to PG5IOR

  Port J: Bits PJ9IOR and PJ8IOR

  Port K: Bits PK7IOR and PK4IOR to PK1IOR

  Port L: Bits PL7IOR, PL1IOR, and PL0IOR
- Following port bits are reserved and nothing is assigned. These bits are always read as "0". The write value should always be "0".

  Port B: Bits 15 to 7

  Port F: Bits 15 to 6

  Port G: Bits 15 to 8

  Port K: Bit 15

  Port L: Bits 15 to 10

Legend: m = A to H and J to L

18.3.9 Port A Control Registers 1 to 4 (PACR1 to PACR4)

The PACR1 to PACR4 registers are used to select the functions of the multiplexed pins of port A.

(1) Port A Control Register 4 (PACR4)

Port A Control Register 4 (PACR4) <P4 address: location H'FFFF 5010>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PA13MD			—	PA12MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6 to 4	PA13MD	000	R	W	PA13 Mode Bits 000: PA13 input/output (port) 001: Setting prohibited 010: TO15 output (ATU-IIIS) 011: DDB13 input (DRI) 100: PSLDATA3 output (PSEL) 101: Setting prohibited 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PA12MD	000	R	W	PA12 Mode Bits 000: PA12 input/output (port) 001: Setting prohibited 010: TO14 output (ATU-IIIS) 011: DDB12 input (DRI) 100: PSLDATA2 output (PSEL) 101: Setting prohibited 11x: Setting prohibited

(2) Port A Control Register 3 (PACR3)

Port A Control Register 3 (PACR3) <P4 address: location H'FFFF 5012>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PA11MD			—	PA10MD			—	PA9MD			—	PA8MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PA11MD	000	R	W	PA11 Mode Bits 000: PA11 input/output (port) 001: Setting prohibited 010: TO13 output (ATU-IIIS) 011: DDB11 input (DRI) 100: PSLDATA1 output (PSEL) 101: Setting prohibited 11x: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PA10MD	000	R	W	PA10 Mode Bits 000: PA10 input/output (port) 001: Setting prohibited 010: TO12 output (ATU-IIIS) 011: DDB10 input (DRI) 100: PSLDATA0 output (PSEL) 101: Setting prohibited 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PA9MD	000	R	W	PA9 Mode Bits 000: PA9 input/output (port) 001: Setting prohibited 010: TO11 output (ATU-IIIS) 011: DDB09 input (DRI) 100: PSLCLKA output (PSEL) 101: Setting prohibited 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	PA8MD	000	R	W	PA8 Mode Bits 000: PA8 input/output (port) 001: Setting prohibited 010: TO10 output (ATU-IIIS) 011: DDB08 input (DRI) 100: PSLCLKB output (PSEL) 101: Setting prohibited 11x: Setting prohibited

(3) Port A Control Register 2 (PACR2)

Port A Control Register 2 (PACR2) <P4 address: location H'FFFF 5014>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PA7MD			—	PA6MD			—	PA5MD			—	PA4MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PA7MD	000	R	W	PA7 Mode Bits 000: PA7 input/output (port) 001: Setting prohibited 010: TO07 output (ATU-IIIS) 011: DDB07 input (DRI) 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PA6MD	000	R	W	PA6 Mode Bits 000: PA6 input/output (port) 001: Setting prohibited 010: TO06 output (ATU-IIIS) 011: DDB06 input (DRI) 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PA5MD	000	R	W	PA5 Mode Bits 000: PA5 input/output (port) 001: Setting prohibited 010: TO05 output (ATU-IIIS) 011: DDB05 input (DRI) 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	PA4MD	000	R	W	PA4 Mode Bits 000: PA4 input/output (port) 001: Setting prohibited 010: TO04 output (ATU-IIIS) 011: DDB04 input (DRI) 1xx: Setting prohibited

(4) Port A Control Register 1 (PACR1)

Port A Control Register 1 (PACR1) <P4 address: location H'FFFF 5016>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PA3MD			—	PA2MD			—	PA1MD			—	PA0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PA3MD	000	R	W	PA3 Mode Bits 000: PA3 input/output (port) 001: Setting prohibited 010: TO03 output (ATU-IIIS) 011: DDB03 input (DRI) 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PA2MD	000	R	W	PA2 Mode Bits 000: PA2 input/output (port) 001: Setting prohibited 010: TO02 output (ATU-IIIS) 011: DDB02 input (DRI) 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PA1MD	000	R	W	PA1 Mode Bits 000: PA1 input/output (port) 001: Setting prohibited 010: TO01 output (ATU-IIIS) 011: DDB01 input (DRI) 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	PA0MD	000	R	W	PA0 Mode Bits 000: PA0 input/output (port) 001: Setting prohibited 010: TO00 output (ATU-IIIS) 011: DDB00 input (DRI) 1xx: Setting prohibited

18.3.10 Port B Control Registers 1 and 2 (PBCR1 and PBCR2)

The PBCR1 and PBCR2 registers are used to select the functions of the multiplexed pins of port B.

(1) Port B Control Register 2 (PBCR2)

Port B Control Register 2 (PBCR2) <P4 address: location H'FFFF 5114>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(2) Port B Control Register 1 (PBCR1)

Port B Control Register 1 (PBCR1) <P4 address: location H'FFFF 5116>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PB3MD			—	—	—	—	—	PB1MD			—	PB0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PB3MD	000	R	W	PB3 Mode Bits 000: PB3 input/output (port) 001: Setting prohibited 010: PWMOFF3 input (ATU-IIIS) 011: DINB3 input (DRI) 1xx: Setting prohibited
11 to 7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6 to 4	PB1MD	000	R	W	PB1 Mode Bits 000: PB1 input/output (port) 001: Setting prohibited 010: PWMOFF1 input (ATU-IIIS) 011: DINB1 input (DRI) 1xx: Setting prohibited
3	—	0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	PB0MD	000	R	W	PB0 Mode Bits 000: PB0 input/output (port) 001: Setting prohibited 010: PWMOFF0 input (ATU-IIIS) 011: DINB0 input (DRI) 1xx: Setting prohibited

18.3.11 Port C Control Registers 1 to 4 (PCCR1 to PCCR4)

The PCCR1 to PCCR4 registers select the functions of the multiplexed pins of port C.

(1) Port C Control Register 4 (PCCR4)

Port C Control Register 4 (PCCR4) <P4 address: location H'FFFF 5210>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PC14MD			—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 8	PC14MD	000	R	W	PC14 Mode Bits 000: PC14 input/output (port) 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 1xx: Setting prohibited
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(2) Port C Control Register 3 (PCCR3)

Port C Control Register 3 (PCCR3) <P4 address: location H'FFFF 5212>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(3) Port C Control Register 2 (PCCR2)

Port C Control Register 2 (PCCR2) <P4 address: location H'FFFF 5214>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PC6MD			—	PC5MD			—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 8	PC6MD	000	R	W	PC6 Mode Bits 000: PC6 input/output (port) 001: CLKOUT output (CPG) 010: TO36 output (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PC5MD	000	R	W	PC5 Mode Bits 000: PC5 input/output (port) 001: Setting prohibited 010: TO35 output (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited
3 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(4) Port C Control Register 1 (PCCR1)

Port C Control Register 1 (PCCR1) <P4 address: location H'FFFF 5216>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PC3MD			—	PC2MD			—	PC1MD			—	PC0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PC3MD	000	R	W	PC3 Mode Bits 000: PC3 input/output (port) 001: Setting prohibited 010: TO33 output (ATU-IIIS) 011: Setting prohibited 100: SSL20 input/output (RSPI) 101: IRQ0 input (INTC) 11x: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PC2MD	000	R	W	PC2 Mode Bits 000: PC2 input/output (port) 001: Setting prohibited 010: TO32 output (ATU-IIIS) 011: Setting prohibited 100: RSPCK2 input/output (RSPI) 101: DREQ0 input (DMAC) 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PC1MD	000	R	W	PC1 Mode Bits 000: PC1 input/output (port) 001: Setting prohibited 010: TO31 output (ATU-IIIS) 011: Setting prohibited 100: MISO2 input/output (RSPI) 101: Setting prohibited 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	PC0MD	000	R	W	PC0 Mode Bits 000: PC0 input/output (port) 001: Setting prohibited 010: TO30 output (ATU-IIIS) 011: Setting prohibited 100: MOSI2 input/output (RSPI) 101: IRQ6 input (INTC) 11x: Setting prohibited

18.3.12 Port D Control Registers 1 to 4 (PDCR1 to PDCR4)

The PDCR1 to PDCR4 registers are used to select the functions of the multiplexed pins of port D.

(1) Port D Control Register 4 (PDCR4)

Port D Control Register 4 (PDCR4) <P4 address: location H'FFFF 5410>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(2) Port D Control Register 3 (PDCR3)

Port D Control Register 3 (PDCR3) <P4 address: location H'FFFF 5412>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PD10MD			—	PD9MD			—	PD8MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 8	PD10MD	000	R	W	PD10 Mode Bits 000: PD10 input/output (port) 001: Setting prohibited 010: PDIWR output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PD9MD	000	R	W	PD9 Mode Bits 000: PD9 input/output (port) 001: Setting prohibited 010: PDIDATA9 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PD8MD	000	R	W	PD8 Mode Bits 000: PD8 input/output (port) 001: Setting prohibited 010: PDIDATA8 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited

(3) Port D Control Register 2 (PDCR2)

Port D Control Register 2 (PDCR2) <P4 address: location H'FFFF 5414>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PD7MD			—	PD6MD			—	PD5MD			—	PD4MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PD7MD	000	R	W	PD7 Mode Bits 000: PD7 input/output (port) 001: Setting prohibited 010: PDIDATA7 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PD6MD	000	R	W	PD6 Mode Bits 000: PD6 input/output (port) 001: Setting prohibited 010: PDIDATA6 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PD5MD	000	R	W	PD5 Mode Bits 000: PD5 input/output (port) 001: Setting prohibited 010: PDIDATA5 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PD4MD	000	R	W	PD4 Mode Bits 000: PD4 input/output (port) 001: Setting prohibited 010: PDIDATA4 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited

(4) Port D Control Register 1 (PDCR1)

Port D Control Register 1 (PDCR1) <P4 address: location H'FFFF 5416>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PD3MD			—	PD2MD			—	PD1MD			—	PD0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PD3MD	000	R	W	PD3 Mode Bits 000: PD3 input/output (port) 001: Setting prohibited 010: PDIDATA3 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PD2MD	000	R	W	PD2 Mode Bits 000: PD2 input/output (port) 001: Setting prohibited 010: PDIDATA2 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PD1MD	000	R	W	PD1 Mode Bits 000: PD1 input/output (port) 001: Setting prohibited 010: PDIDATA1 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PD0MD	000	R	W	PD0 Mode Bits 000: PD0 input/output (port) 001: Setting prohibited 010: PDIDATA0 output (PDAC) 011: Setting prohibited 1xx: Setting prohibited

18.3.13 Port E Control Registers 3 and 4 (PECR3 and PECR4)

The PECR3 and PECR4 registers are used to select the functions of the multiplexed pins of port E.

(1) Port E Control Register 4 (PECR4)

Port E Control Register 4 (PECR4) <P4 address: location H'FFFF 5510>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PE15MD			—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PE15MD	000	R	W	PE15 Mode Bits 000: PE15 input/output (port) 001: Setting prohibited 010: TO27 output (ATU-IIIS) 011: Setting prohibited 100: PSLCLR output (PSEL) 101: Setting prohibited 1xx: Setting prohibited
11 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(2) Port E Control Register 3 (PECR3)

Port E Control Register 3 (PECR3) <P4 address: location H'FFFF 5512>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

18.3.14 Port F Control Registers 1 and 2 (PFCR1 and PFCR2)

The PFCR1 and PFCR2 registers are used to select the functions of the multiplexed pins of port F.

(1) Port F Control Register 2 (PFCR2)

Port F Control Register 2 (PFCR2) <P4 address: location H'FFFF 5614>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PF5MD			—	PF4MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6 to 4	PF5MD	000	R	W	PF5 Mode Bits 000: PF5 input/output (port) 001: SCL input/output (IIC3) 01x: Setting prohibited 100: Setting prohibited 101: CTX3 output (CAN) 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PF4MD	000	R	W	PF4 Mode Bits 000: PF4 input/output (port) 001: SDA input/output (IIC3) 01x: Setting prohibited 100: Setting prohibited 101: CRX3 input (CAN) 11x: Setting prohibited

(2) Port F Control Register 1 (PFCR1)

Port F Control Register 1 (PFCR1) <P4 address: location H'FFFF 5616>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PF1MD			—	PF0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6 to 4	PF1MD	000	R	W	PF1 Mode Bits 000: PF1 input/output (port) 001: CTX0 output (CAN) 01x: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PF0MD	000	R	W	PF0 Mode Bits 000: PF0 input/output (port) 001: CRX0 input (CAN) 01x: Setting prohibited 1xx: Setting prohibited

18.3.15 Port G Control Registers 1 and 2 (PGCR1 and PGCR2)

The PGCR1 and PGCR2 registers are used to select the functions of the multiplexed pins of port G.

(1) Port G Control Register 2 (PGCR2)

Port G Control Register 2 (PGCR2) <P4 address: location H'FFFF 5814>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PG4MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	PG4MD	000	R	W	PG4 Mode Bits 000: PG4 input/output (port) 001: IRQ2 input (INTC) 010: TO44 output (ATU-IIIS) 011: SSL01 output (RSPI) 1xx: Setting prohibited

(2) Port G Control Register 1 (PGCR1)

Port G Control Register 1 (PGCR1) <P4 address: location H'FFFF 5816>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PG3MD			—	PG2MD			—	PG1MD			—	PG0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PG3MD	000	R	W	PG3 Mode Bits 000: PG3 input/output (port) 001: Setting prohibited 010: TO43 output (ATU-IIIS) 011: SSL00 input/output (RSPI) 100: Setting prohibited 101: IRQ7 input (INTC) 11x: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PG2MD	000	R	W	PG2 Mode Bits 000: PG2 input/output (port) 001: RSPCK0 input/output (RSPI) 010: TO42 output (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PG1MD	000	R	W	PG1 Mode Bits 000: PG1 input/output (port) 001: MISO0 input/output (RSPI) 010: TO41 output (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PG0MD	000	R	W	PG0 Mode Bits 000: PG0 input/output (port) 001: MOSI0 input/output (RSPI) 010: TO40 output (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited

18.3.16 Port H Control Registers 1 to 4 (PHCR1 to PHCR4)

The PHCR1 to PHCR4 registers are used to select the functions of the multiplexed pins of port H.

(1) Port H Control Register 4 (PHCR4)

Port H Control Register 4 (PHCR4) <P4 address: location H'FFFF 5910>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PH15MD			—	PH14MD			—	PH13MD			—	PH12MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PH15MD	000	R	W	PH15 Mode Bits 000: PH15 input/output (port) 001: DROD7 output (DRO) 010: TO37 output (ATU-IIIS) 011: DDC15 input (DRI) 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PH14MD	000	R	W	PH14 Mode Bits 000: PH14 input/output (port) 001: DROD6 output (DRO) 010: TO36 output (ATU-IIIS) 011: DDC14 input (DRI) 100: Setting prohibited 101: IRQ1 input (INTC) 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PH13MD	000	R	W	PH13 Mode Bits 000: PH13 input/output (port) 001: DROD5 output (DRO) 010: TO35 output (ATU-IIIS) 011: DDC13 input (DRI) 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PH12MD	000	R	W	PH12 Mode Bits 000: PH12 input/output (port) 001: DROD4 output (DRO) 010: TO34 output (ATU-IIIS) 011: DDC12 input (DRI) 1xx: Setting prohibited

(2) Port H Control Register 3 (PHCR3)

Port H Control Register 3 (PHCR3) <P4 address: location H'FFFF 5912>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PH11MD			—	PH10MD			—	PH9MD			—	PH8MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PH11MD	000	R	W	PH11 Mode Bits 000: PH11 input/output (port) 001: DROD3 output (DRO) 010: TO33 output (ATU-IIIS) 011: DDC11 input (DRI) 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PH10MD	000	R	W	PH10 Mode Bits 000: PH10 input/output (port) 001: DROD2 output (DRO) 010: TO32 output (ATU-IIIS) 011: DDC10 input (DRI) 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PH9MD	000	R	W	PH9 Mode Bits 000: PH9 input/output (port) 001: DROD1 output (DRO) 010: TO31 output (ATU-IIIS) 011: DDC09 input (DRI) 100: CTS2# input/output (SCIF) 101: Setting prohibited 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PH8MD	000	R	W	PH8 Mode Bits 000: PH8 input/output (port) 001: DROD0 output (DRO) 010: TO30 output (ATU-IIIS) 011: DDC08 input (DRI) 100: RTS2# input/output (SCIF) 101: Setting prohibited 11x: Setting prohibited

(3) Port H Control Register 2 (PHCR2)

Port H Control Register 2 (PHCR2) <P4 address: location H'FFFF 5914>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PH7MD			—	PH6MD			—	PH5MD			—	PH4MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PH7MD	000	R	W	PH7 Mode Bits 000: PH7 input/output (port) 001: DROD15 output (DRO) 010: TO27 output (ATU-IIIS) 011: DDC07 input (DRI) 100: TIA03 input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PH6MD	000	R	W	PH6 Mode Bits 000: PH6 input/output (port) 001: DROD14 output (DRO) 010: TO26 output (ATU-IIIS) 011: DDC06 input (DRI) 100: TIA02 input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PH5MD	000	R	W	PH5 Mode Bits 000: PH5 input/output (port) 001: DROD13 output (DRO) 010: TO25 output (ATU-IIIS) 011: DDC05 input (DRI) 100: TIA01 input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PH4MD	000	R	W	PH4 Mode Bits 000: PH4 input/output (port) 001: DROD12 output (DRO) 010: TO24 output (ATU-IIIS) 011: DDC04 input (DRI) 100: TIA00 input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited

(4) Port H Control Register 1 (PHCR1)

Port H Control Register 1 (PHCR1) <P4 address: location H'FFFF 5916>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PH3MD			—	PH2MD			—	PH1MD			—	PH0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PH3MD	000	R	W	PH3 Mode Bits 000: PH3 input/output (port) 001: DROD11 output (DRO) 010: TO23 output (ATU-IIIS) 011: DDC03 input (DRI) 100: TIF1B input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PH2MD	000	R	W	PH2 Mode Bits 000: PH2 input/output (port) 001: DROD10 output (DRO) 010: TO22 output (ATU-IIIS) 011: DDC02 input (DRI) 100: TIF1A input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
6 to 4	PH1MD	000	R	W	PH1 Mode Bits 000: PH1 input/output (port) 001: DROD9 output (DRO) 010: TO21 output (ATU-IIIS) 011: DDC01 input (DRI) 100: TIF0B input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PH0MD	000	R	W	PH0 Mode Bits 000: PH0 input/output (port) 001: DROD8 output (DRO) 010: TO20 output (ATU-IIIS) 011: DDC00 input (DRI) 100: TIF0A input (ATU-IIIS) 101: Setting prohibited 11x: Setting prohibited

18.3.17 Port J Control Registers 1 to 4 (PJCR1 to PJCR4)

The PJCR1 to PJCR4 registers are used to select the functions of the multiplexed pins of port J.

(1) Port J Control Register 4 (PJCR4)

Port J Control Register 4 (PJCR4) <P4 address: location H'FFFF 5A10>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ15MD			—	PJ14MD			—	PJ13MD			—	PJ12MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PJ15MD	000	R	W	PJ15 Mode Bits 000: PJ15 input/output (port) 001: SCK1 input/output (SCIF) 010: PSPCK1 output (RSPI) 011: Setting prohibited 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PJ14MD	000	R	W	PJ14 Mode Bits 000: PJ14 input/output (port) 001: TXD1 output (SCIF) 010: MOSI1 input/output (RSPI) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PJ13MD	000	R	W	PJ13 Mode Bits 000: PJ13 input/output (port) 001: RXD1 input (SCIF) 010: MISO1 input/output (RSPI) 011: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PJ12MD	000	R	W	PJ12 Mode Bits 000: PJ12 input/output (port) 001: SCK0 input/output (SCIF) 010: TCLKB input (ATU-IIIS) 011: Setting prohibited 100: Setting prohibited 101: IRQ0 input (INTC) 11x: Setting prohibited

(2) Port J Control Register 3 (PJCR3)

Port J Control Register 3 (PJCR3) <P4 address: location H'FFFF 5A12>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ11MD			—	PJ10MD			—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PJ11MD	000	R	W	PJ11 Mode Bits 000: PJ11 input/output (port) 001: TXD0 output (SCIF) 010: Setting prohibited 011: Setting prohibited 100: AD0END output (ADC) 101: Setting prohibited 11x: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PJ10MD	000	R	W	PJ10 Mode Bits 000: PJ10 input/output (port) 001: RXD0 input (SCIF) 010: PWMOFF4 input (ATU-IIIS) 011: Setting prohibited 100: AD0TRG# input (ADC) 101: Setting prohibited 11x: Setting prohibited
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(3) Port J Control Register 2 (PJCR2)

Port J Control Register 2 (PJCR2) <P4 address: location H'FFFF 5A14>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ7MD			—	PJ6MD			—	PJ5MD			—	PJ4MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PJ7MD	000	R	W	PJ7 Mode Bits 000: PJ7 input/output (port) 001: CTX3 output (CAN) 010: TIF2B input (ATU-IIIS) 011: Setting prohibited 100: TXD2 output (SCIF) 101: Setting prohibited 11x: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PJ6MD	000	R	W	PJ6 Mode Bits 000: PJ6 input/output (port) 001: CRX3 input (CAN) 010: TIF2A input (ATU-IIIS) 011: Setting prohibited 100: RXD2 input (SCIF) 101: TIA04 input (ATU-IIIS) 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PJ5MD	000	R	W	PJ5 Mode Bits 000: PJ5 input/output (port) 001: CTX2 output (CAN) 010: FTXENB output (FlexRay) 011: Setting prohibited 100: SCK2 input/output (SCIF) 101: Setting prohibited 111: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	PJ4MD	000	R	W	PJ4 Mode Bits 000: PJ4 input/output (port) 001: CRX2 input (CAN) 010: FTXENA output (FlexRay) 011: Setting prohibited 100: CTS0# input/output (SCIF) 101: Setting prohibited 111: Setting prohibited

Note: • On the SH7456 Group, do not select the FlexRay-related pins (FRXA, FTXA, FRXB, FTXB, FTXENA, and FTXENB).

(4) Port J Control Register 1 (PJCR1)

Port J Control Register 1 (PJCR1) <P4 address: location H'FFFF 5A16>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ3MD			—	PJ2MD			—	PJ1MD			—	PJ0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PJ3MD	000	R	W	PJ3 Mode Bits 000: PJ3 input/output (port) 001: CTX1 output (CAN) 010: FTXB output (FlexRay) 011: Setting prohibited 100: RTS0# input/output (SCIF) 101: Setting prohibited 111: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PJ2MD	000	R	W	PJ2 Mode Bits 000: PJ2 input/output (port) 001: CRX1 input (CAN) 010: FRXB input (FlexRay) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PJ1MD	000	R	W	PJ1 Mode Bits 000: PJ1 input/output (port) 001: CTX0 output (CAN) 010: FTXA output (FlexRay) 011: Setting prohibited 1xx: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PJ0MD	000	R	W	PJ0 Mode Bits 000: PJ0 input/output (port) 001: CRX0 input (CAN) 010: FRXA input (FlexRay) 011: Setting prohibited 1xx: Setting prohibited

Note: • On the SH7456 Group, do not select the FlexRay-related pins (FRXA, FTXA, FRXB, FTXB, FTXENA, and FTXENB).

18.3.18 Port K Control Registers 1 to 4 (PKCR1 to PKCR4)

The PKCR1 to PKCR4 registers are used to select the functions of the multiplexed pins of port K.

(1) Port K Control Register 4 (PKCR4)

Port K Control Register 4 (PKCR4) <P4 address: location H'FFFF 5C10>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PK14MD			—	PK13MD			—	PK12MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 8	PK14MD	000	R	W	PK14 Mode Bits 000: PK14 input/output (port) 001: AUDRSYN# input (AUDR) 01x: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PK13MD	000	R	W	PK13 Mode Bits 000: PK13 input/output (port) 001: AUDRCLK input (AUDR) 01x: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PK12MD	000	R	W	PK12 Mode Bits 000: PK12 input/output (port) 001: AUDRD3 input/output (AUDR) 01x: Setting prohibited 1xx: Setting prohibited

(2) Port K Control Register 3 (PKCR3)

Port K Control Register 3 (PKCR3) <P4 address: location H'FFFF 5C12>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PK11MD			—	PK10MD			—	PK9MD			—	PK8MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PK11MD	000	R	W	PK11 Mode Bits 000: PK11 input/output (port) 001: AUDRD2 input/output (AUDR) 01x: Setting prohibited 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PK10MD	000	R	W	PK10 Mode Bits 000: PK10 input/output (port) 001: AUDRD1 input/output (AUDR) 01x: Setting prohibited 100: CTS3# input/output (SCIF) 101: Setting prohibited 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PK9MD	000	R	W	PK9 Mode Bits 000: PK9 input/output (port) 001: AUDRD0 input/output (AUDR) 01x: Setting prohibited 100: RTS3# input/output (SCIF) 101: Setting prohibited 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PK8MD	000	R	W	PK8 Mode Bits 000: PK8 input/output (port) 001: DREQ2 input (DMAC) 01x: Setting prohibited 1xx: Setting prohibited

(3) Port K Control Register 2 (PKCR2)

Port K Control Register 2 (PKCR2) <P4 address: location H'FFFF 5C14>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PK6MD			—	PK5MD			—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 8	PK6MD	000	R	W	PK6 Mode Bits 000: PK6 input/output (port) 001: Setting prohibited 01x: Setting prohibited 100: TXD3 output (SCIF) 101: Setting prohibited 11x: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PK5MD	000	R	W	PK5 Mode Bits 000: PK5 input/output (port) 001: Setting prohibited 010: Setting prohibited 011: DINC4 input (DRI) 100: RXD3 input (SCIF) 101: Setting prohibited 11x: Setting prohibited
3 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

(4) Port K Control Register 1 (PKCR1)

Port K Control Register 1 (PKCR1) <P4 address: location H'FFFF 5C16>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PK0MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	PK0MD	000	R	W	PK0 Mode Bits 000: PK0 input/output (port) 001: IRQ5 input (INTC) 010: SSL10 input/output (RSPI) 011: Setting prohibited 1xx: Setting prohibited

18.3.19 Port L Control Registers 1 to 3 (PLCR1 to PLCR3)

The PLCR1 to PLCR3 registers are used to select the functions of the multiplexed pins of port L.

(1) Port L Control Register 3 (PLCR3)

Port L Control Register 3 (PLCR3) <P4 address: location H'FFFF 5D12>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PL9MD			—	PL8MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6 to 4	PL9MD	000	R	W	PL9 Mode Bits 000: PL9 input/output (port) 001: TIA15 input (ATU-IIIS) 01x: Setting prohibited 100: Setting prohibited 101: AUDREVT# output (AUDR) 11x: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PL8MD	000	R	W	PL8 Mode Bits 000: PL8 input/output (port) 001: TIA14 input (ATU-IIIS) 010: IRQ7 input (INTC) 011: Setting prohibited 100: DREQ3 input (DMAC) 101: Setting prohibited 11x: Setting prohibited

(2) Port L Control Register 2 (PLCR2)

Port L Control Register 2 (PLCR2) <P4 address: location H'FFFF 5D14>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PL6MD			—	PL5MD			—	PL4MD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 8	PL6MD	000	R	W	PL6 Mode Bits 000: PL6 input/output (port) 001: TIA12 input (ATU-IIIS) 010: TIF1A input (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PL5MD	000	R	W	PL5 Mode Bits 000: PL5 input/output (port) 001: TIA11 input (ATU-IIIS) 010: TIF0B input (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PL4MD	000	R	W	PL4 Mode Bits 000: PL4 input/output (port) 001: TIA10 input (ATU-IIIS) 010: TIF0A input (ATU-IIIS) 011: Setting prohibited 1xx: Setting prohibited

(3) Port L Control Register 1 (PLCR1)

Port L Control Register 1 (PLCR1) <P4 address: location H'FFFF 5D16>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PL3MD			—	PL2MD			—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PL3MD	000	R	W	PL3 Mode Bits 000: PL3 input/output (port) 001: Setting prohibited 010: IRQ6 input (INTC) 011: Setting prohibited 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PL2MD	000	R	W	PL2 Mode Bits 000: PL2 input/output (port) 001: DROWR output (DRO) 01x: Setting prohibited 1xx: Setting prohibited
7 to 0	—	All 0	0	0	Reserved Bits These bits are read as "0". The write value should always be "0".

18.3.20 Port M Control Registers 1 to 4 (PMCR1 to PMCR4)

The PMCR1 to PMCR4 registers are used to select the functions of the multiplexed pins of port M.

(1) Port M Control Register 4 (PMCR4)

Port M Control Register 4 (PMCR4) <P4 address: location H'FFFF 5E10>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PM15MD			—	PM14MD			—	PM13MD			—	PM12MD		
After Reset:	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1

<After Reset: H'1111>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PM15MD	001	R	W	PM15 Mode Bits 000: PM15 input (port) 001: AD0IN15 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PM14MD	001	R	W	PM14 Mode Bits 000: PM14 input (port) 001: AD0IN14 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PM13MD	001	R	W	PM13 Mode Bits 000: PM13 input (port) 001: AD0IN13 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PM12MD	001	R	W	PM12 Mode Bits 000: PM12 input (port) 001: AD0IN12 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited

Note: • Set PM0 to PM15 as all analog input pins or all general ports. Setting and using them as a mixture of analog input pins and general ports is not supported.

(2) Port M Control Register 3 (PMCR3)

Port M Control Register 3 (PMCR3) <P4 address: location H'FFFF 5E12>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PM11MD			—	PM10MD			—	PM9MD			—	PM8MD		
After Reset:	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1

<After Reset: H'1111>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 12	PM11MD	001	R	W	PM11 Mode Bits 000: PM11 input (port) 001: AD0IN11 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PM10MD	001	R	W	PM10 Mode Bits 000: PM10 input (port) 001: AD0IN10 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PM9MD	001	R	W	PM9 Mode Bits 000: PM9 input (port) 001: AD0IN9 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PM8MD	001	R	W	PM8 Mode Bits 000: PM8 input (port) 001: AD0IN8 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited

Note: • Set PM0 to PM15 as all analog input pins or all general ports. Setting and using them as a mixture of analog input pins and general ports is not supported.

(3) Port M Control Register 2 (PMCR2)

Port M Control Register 2 (PMCR2) <P4 address: location H'FFFF 5E14>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PM6MD			—	—	—	—	—	PM4MD		
After Reset:	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1

<After Reset: H'1111>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PM6MD	001	R	W	PM6 Mode Bits 000: PM6 input (port) 001: AD0IN6 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PM4MD	001	R	W	PM4 Mode Bits 000: PM4 input (port) 001: AD0IN4 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited

Note: • Set PM0 to PM15 as all analog input pins or all general ports. Setting and using them as a mixture of analog input pins and general ports is not supported.

(4) Port M Control Register 1 (PMCR1)

Port M Control Register 1 (PMCR1) <P4 address: location H'FFFF 5E16>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PM2MD			—	—	—	—	—	PM0MD		
After Reset:	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1

<After Reset: H'1111>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10 to 8	PM2MD	001	R	W	PM2 Mode Bits 000: PM2 input (port) 001: AD0IN2 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PM0MD	001	R	W	PM0 Mode Bits 000: PM0 input (port) 001: AD0IN0 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited

Note: • Set PM0 to PM15 as all analog input pins or all general ports. Setting and using them as a mixture of analog input pins and general ports is not supported.

18.3.21 Port N Control Registers 1 and 2 (PNCR1 and PNCR2)

The PNCR1 and PNCR2 registers are used to select the functions of the multiplexed pins of port N.

(1) Port N Control Register 2 (PNCR2)

<P4 address: location H'FFFF 5F14>

Port N Control Register 2 (PNCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PN5MD			—	PN4MD		
After Reset:	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1

<After Reset: H'1111>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
11 to 9	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
8	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PN5MD	001	R	W	PN5 Mode Bits 000: PN5 input (port) 001: AD1IN5 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PN4MD	001	R	W	PN4 Mode Bits 000: PN4 input (port) 001: AD1IN4 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited

Note: • Set PN0 to PN7 as all analog input pins or all general ports. Setting and using them as a mixture of analog input pins and general ports is not supported.

(2) Port N Control Register 1 (PNCR1)

Port N Control Register 1 (PNCR1) <P4 address: location H'FFFF 5F16>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PN1MD			—	PN0MD		
After Reset:	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1

<After Reset: H'1111>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
11 to 9	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
8	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	PN1MD	001	R	W	PN1 Mode Bits 000: PN1 input (port) 001: AD1IN1 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2 to 0	PN0MD	001	R	W	PN0 Mode Bits 000: PN0 input (port) 001: AD1IN0 input (ADC) 01x: Setting prohibited 1xx: Setting prohibited

Note: • Set PN0 to PN7 as all analog input pins or all general ports. Setting and using them as a mixture of analog input pins and general ports is not supported.

18.3.22 Port DRI Input Channel Switching Register (PDRIR)

The PDRIR register is used to select the DRI channels (DRI0 to DRI2) used for data input to the DRI pins.

Port DRI input channel switching register (PDRIR) <P4 address: location H'FFFF 5340>

Bit:	7	6	5	4	3	2	1	0
	—	—	PDRI2MD		PDRI1MD		PDRI0MD	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5, 4	PDRI2MD	00	R	W	PDRI2 Mode Bits Select the pins on which input to DRI2 is applied. 00: Input has no effect 01: Input on pins DINB0, DINB1, DINB3, and DDB00 to DDB13 10: Input on pins DINC4 and DDC00 to DDC15 11: Setting prohibited
3, 2	PDRI1MD	00	R	W	PDRI1 Mode Bits Select the pins on which input to DRI1 is applied. 00: Input has no effect 01: Input on pins DINB0, DINB1, DINB3, and DDB00 to DDB13 10: Input on pins DINC4 and DDC00 to DDC15 11: Setting prohibited
1, 0	PDRI0MD	00	R	W	PDRI0 Mode Bits Select the pins on which input to DRI0 is applied. 00: Input has no effect 01: Input on pins DINB0, DINB1, DINB3, and DDB00 to DDB13 10: Input on pins DINC4 and DDC00 to DDC15 11: Setting prohibited

18.4 I/O Port Initial Setting Procedure Examples

Examples of initial setting procedures for I/O port related registers are shown below.

(1) Setting procedure when using pins as input ports

1. Setting of reserved bits in the port m I/O register (PmIOR)
   Set the bits listed in note 1 below to "1".
2. Setting of port m control register n (PmCRn)
   Set the pins to port operation (after reset cancellation, port operation).
3. Setting of port m I/O register (PmIOR)
   Set the port pins to input operation (after reset cancellation, input operation).
4. Setting of port m input threshold value switching register (PmLVR)
   Set the input threshold value (port input enabled).

(2) Setting procedure when using pins as output ports

1. Setting of reserved bits in the port m I/O register (PmIOR)
   Set the bits listed in note 1 below to "1".
2. Setting of port m control register n (PmCRn)
   Set the pins to port operation (after reset cancellation, port operation).
3. Setting of port m data register (PmDR)
   Set the output data.
4. Setting of port m driving ability setting register (PmDSR)
   Set the drive capacity of the output pins.
5. Setting of port m I/O register (PmIOR)
   Set the port pins to output operation.

(3) Setting procedure when using pins as peripheral function inputs

1. Setting of reserved bits in the port m I/O register (PmIOR)
   Set the bits listed in note 1 below to "1".
2. Setting of port m input threshold value switching register (PmLVR)
   Set the input threshold value (port input enabled).
3. Setting of port m control register n (PmCRn)
   Set the pins to peripheral function operation.

(4) Setting procedure when using pins as peripheral function outputs

1. Setting of reserved bits in the port m I/O register (PmIOR)
   Set the bits listed in note 1 below to "1".
2. Setting of port m driving ability setting register (PmDSR)
   Set the drive capacity of the output pins.
3. Setting of port m control register n (PmCRn)
   Set the pins to peripheral function operation.

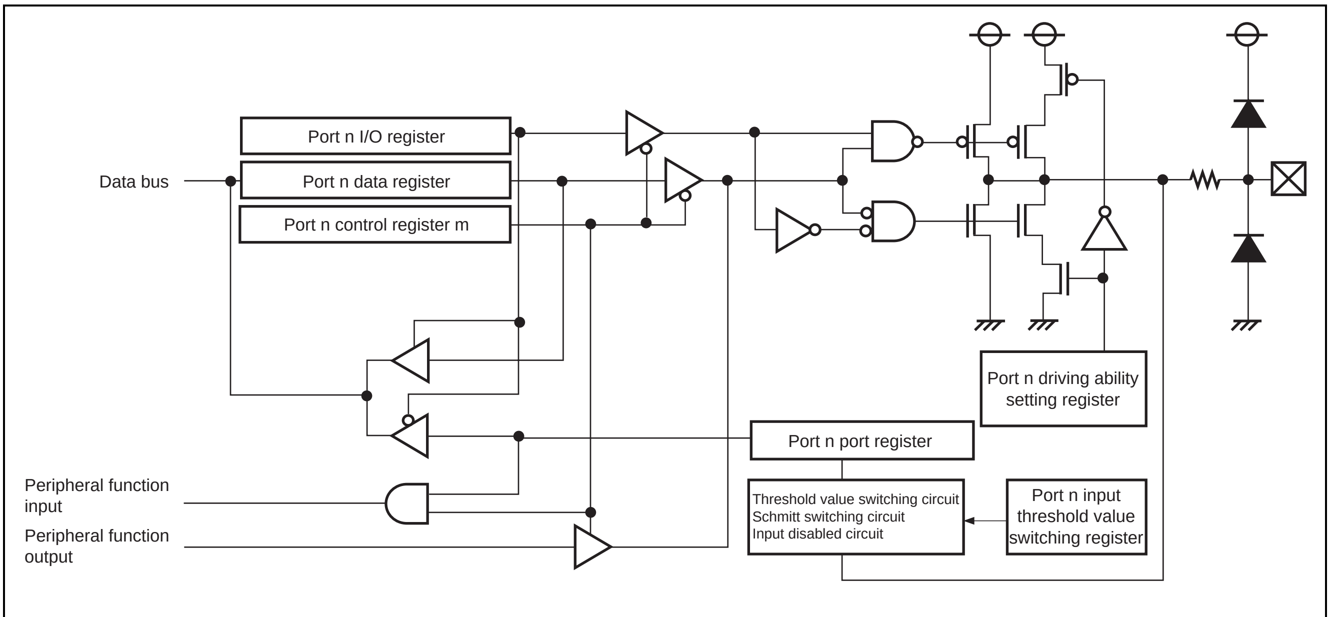
Note: *1 After a reset is canceled, set the following reserved bits to "1".

Port A: Bits PA15IOR and PA14IOR
Port B: Bits PB6IOR to PB4IOR and PB2IOR
Port C: Bits PC15IOR, PC13IOR to PC7IOR, and PC4IOR
Port D: Bits PD15IOR to PD11IOR
Port E: Bits PE14IOR to PE0IOR
Port F: Bits PF3IOR and PF2IOR
Port G: Bits PG7IOR to PG5IOR
Port J: Bits PJ9IOR and PJ8IOR
Port K: Bits PK7IOR and PK4IOR to PK1IOR
Port L: Bits PL7IOR, PL1IOR, and PL0IOR

18.5 Port Peripheral Circuits

Figures 18.1 to 18.12 show port peripheral circuit diagrams.

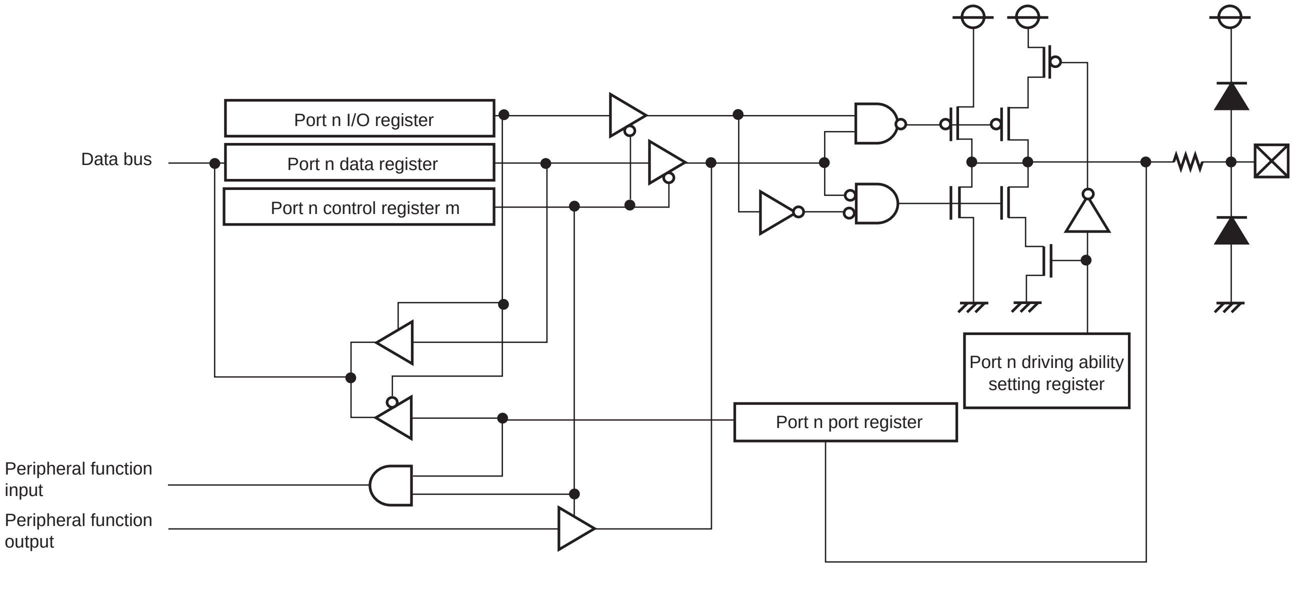
- Applicable ports
  PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PG4, PH0 to PH15, PJ0, PJ2, PJ6, PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9


Port n I/O register
Data bus
Port n data register
Port n control register m
Port n driving ability setting register
Port n port register
Peripheral function input
Peripheral function output
Threshold value switching circuit
Schmitt switching circuit
Input disabled circuit
Port n input threshold value switching register

Note: • Route that programming a value to register (connects to a data bus) is omitted.

Figure 18.1 Port Peripheral Circuit Diagram (Input/Output Ports 1)

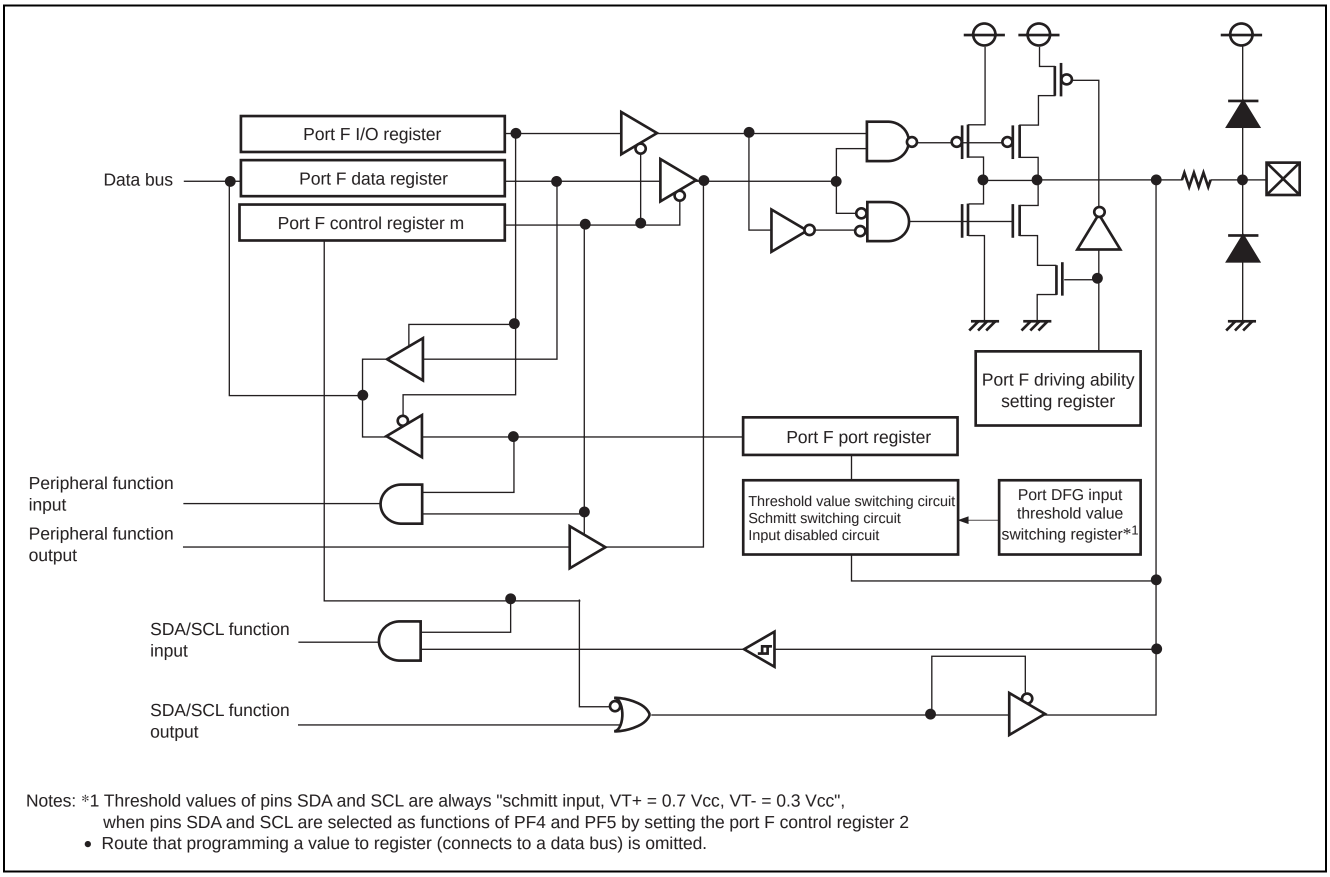
- Applicable ports
  PG0 to PG3, PJ1, PJ3, PJ4, PJ5


Port n I/O register
Data bus
Port n data register
Port n control register m
Port n driving ability setting register
Port n port register
Peripheral function input
Peripheral function output

Notes: • Threshold value is always "CMOS input, 0.50 Vcc".
• Route that programming a value to register (connects to a data bus) is omitted.

Figure 18.2 Port Peripheral Circuit Diagram (Input/Output Ports 2)

- Applicable ports
  PF4, PF5


Port F I/O register
Data bus
Port F data register
Port F control register m
Port F driving ability setting register
Port F port register
Peripheral function input
Peripheral function output
Threshold value switching circuit
Schmitt switching circuit
Input disabled circuit
Port DFG input threshold value switching register*1
SDA/SCL function input
SDA/SCL function output

Notes: *1 Threshold values of pins SDA and SCL are always "schmitt input, VT+ = 0.7 Vcc, VT- = 0.3 Vcc", when pins SDA and SCL are selected as functions of PF4 and PF5 by setting the port F control register 2

- Route that programming a value to register (connects to a data bus) is omitted.

Figure 18.3 Port Peripheral Circuit Diagram (Input/Output Ports 3)

- Applicable ports
  ASEBRK#/BRKACK

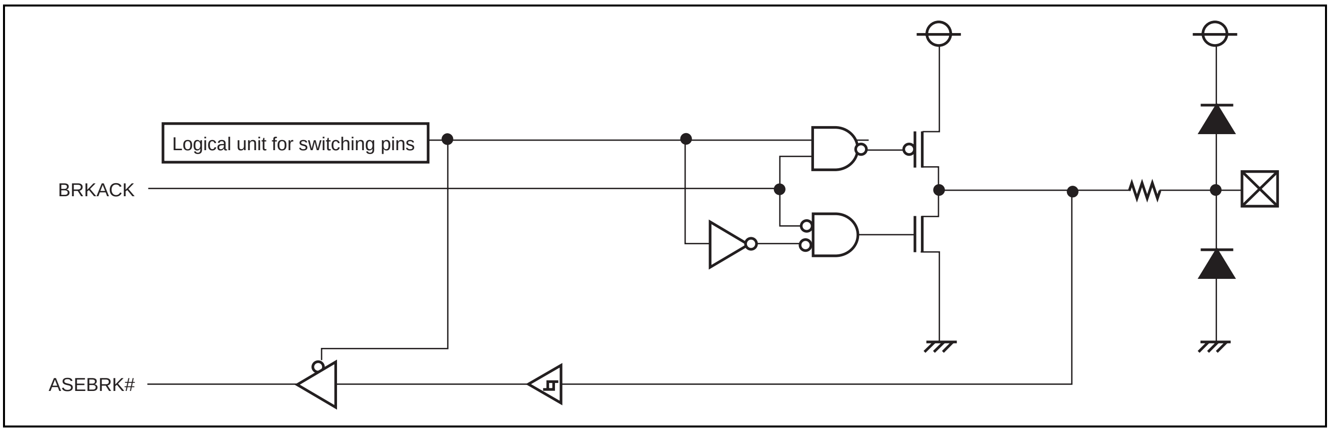


Figure 18.4 Port Peripheral Circuit Diagram (Input/Output Ports 4)

- Applicable ports (port input and analog input)
  PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5

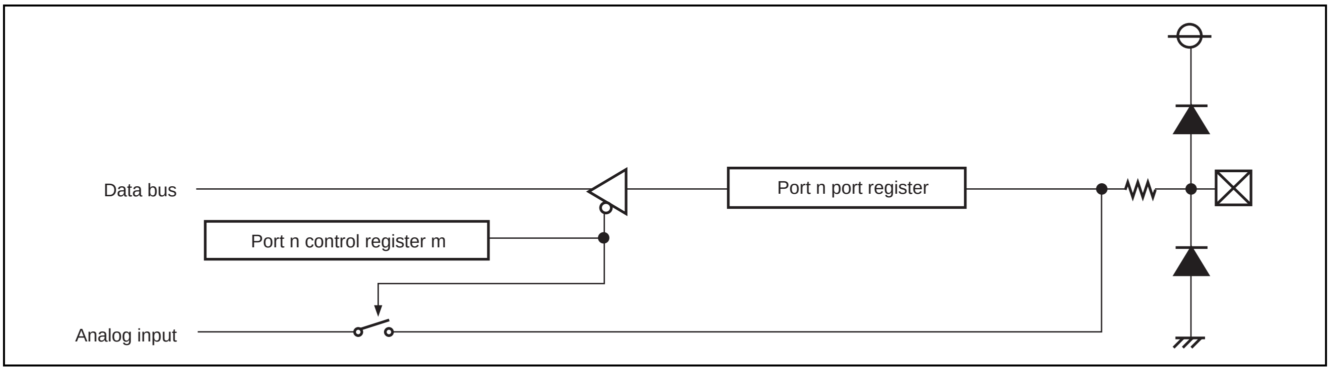


Figure 18.5 Port Peripheral Circuit Diagram (Input Port 1)

- Applicable ports (with noise canceller function)
  NMI, RESET#

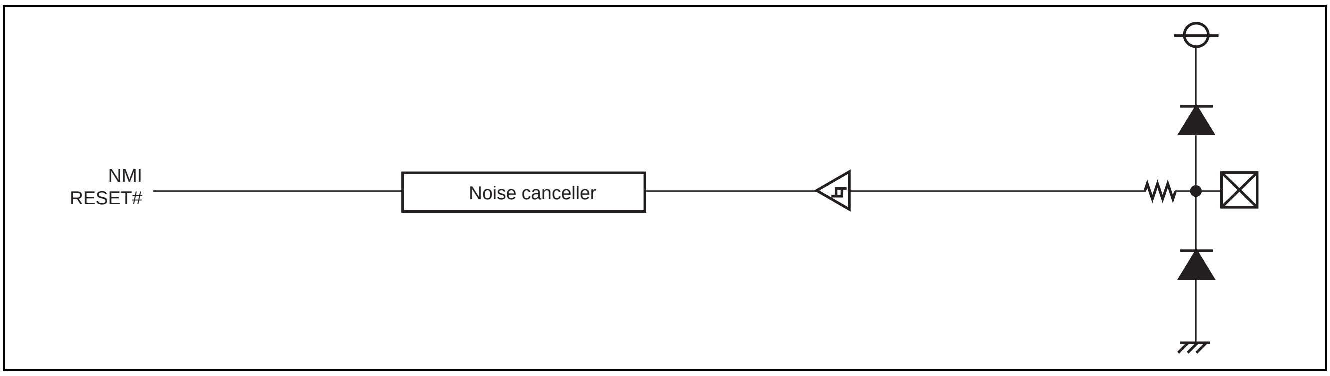
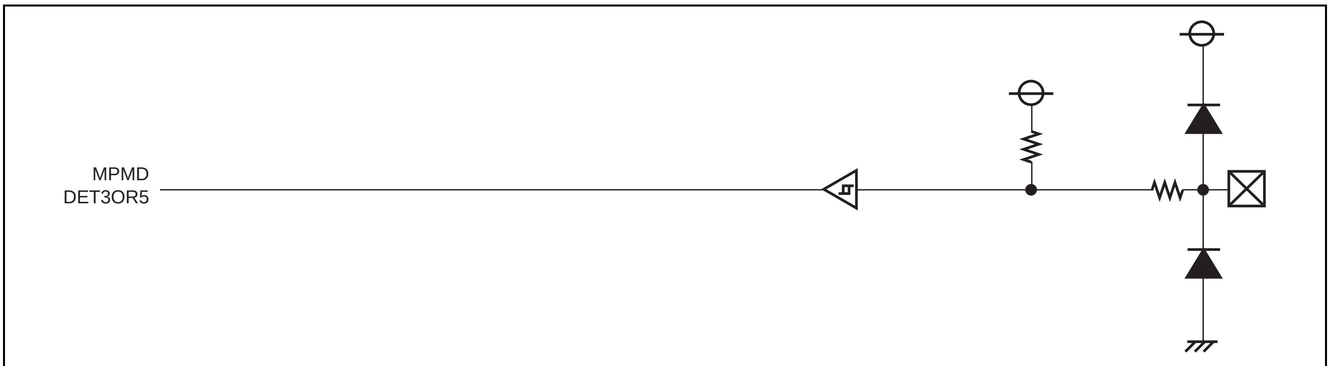


Figure 18.6 Port Peripheral Circuit Diagram (Input Port 2)

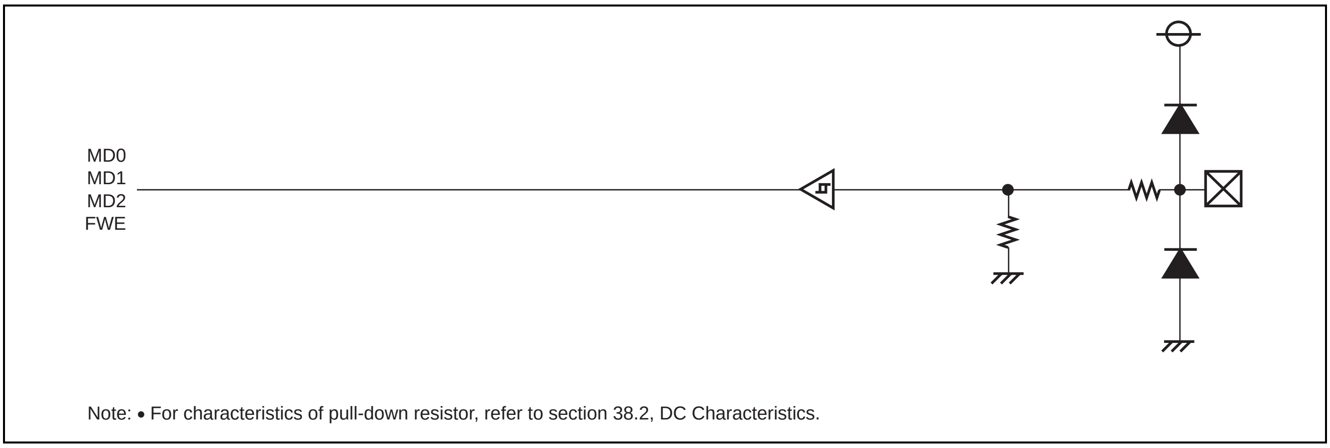
- Applicable ports (with pull-up function)
  MPMD, DET3OR5



Note: • For characteristics of pull-up resistor, refer to section 38.2, DC Characteristics.

Figure 18.7 Port Peripheral Circuit Diagram (Input Port 3)

- Applicable ports (with pull-down function)
  MD0 to MD2, FWE



Note: • For characteristics of pull-down resistor, refer to section 38.2, DC Characteristics.

Figure 18.8 Port Peripheral Circuit Diagram (Input Port 4)

- Applicable ports
  TMS, TDI, TRST#

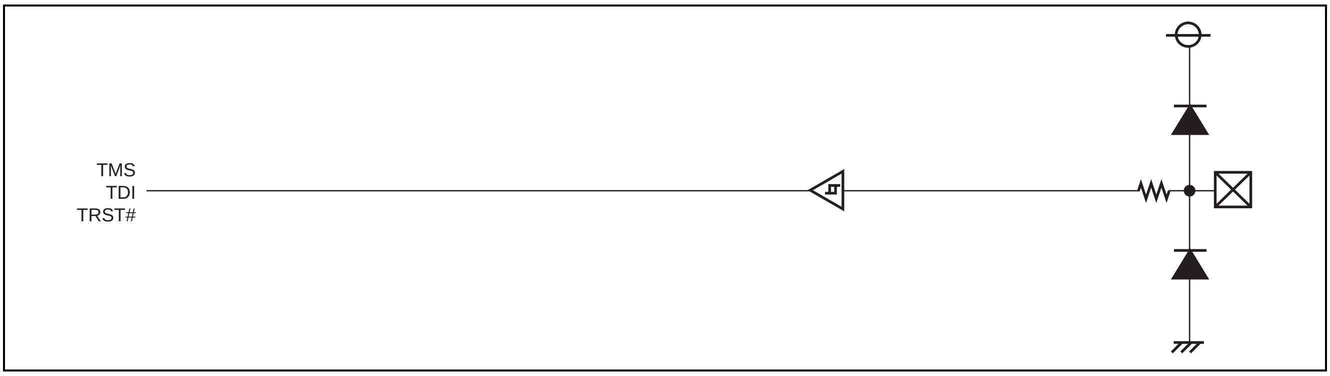


Figure 18.9 Port Peripheral Circuit Diagram (Input Port 5)

- Applicable port (clock input)
  TCK

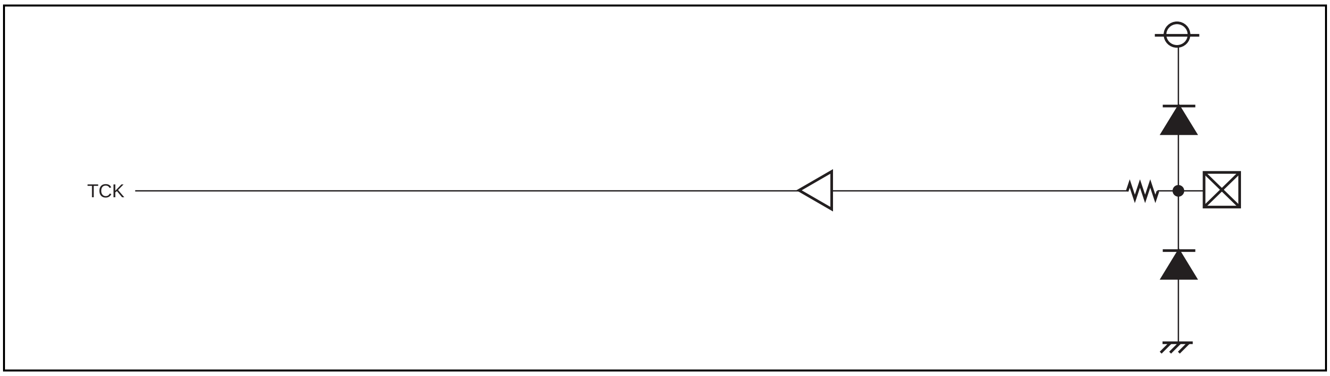


Figure 18.10 Port Peripheral Circuit Diagram (Input Port 6)

- Applicable ports (clock input and clock output)
  EXTAL, XTAL

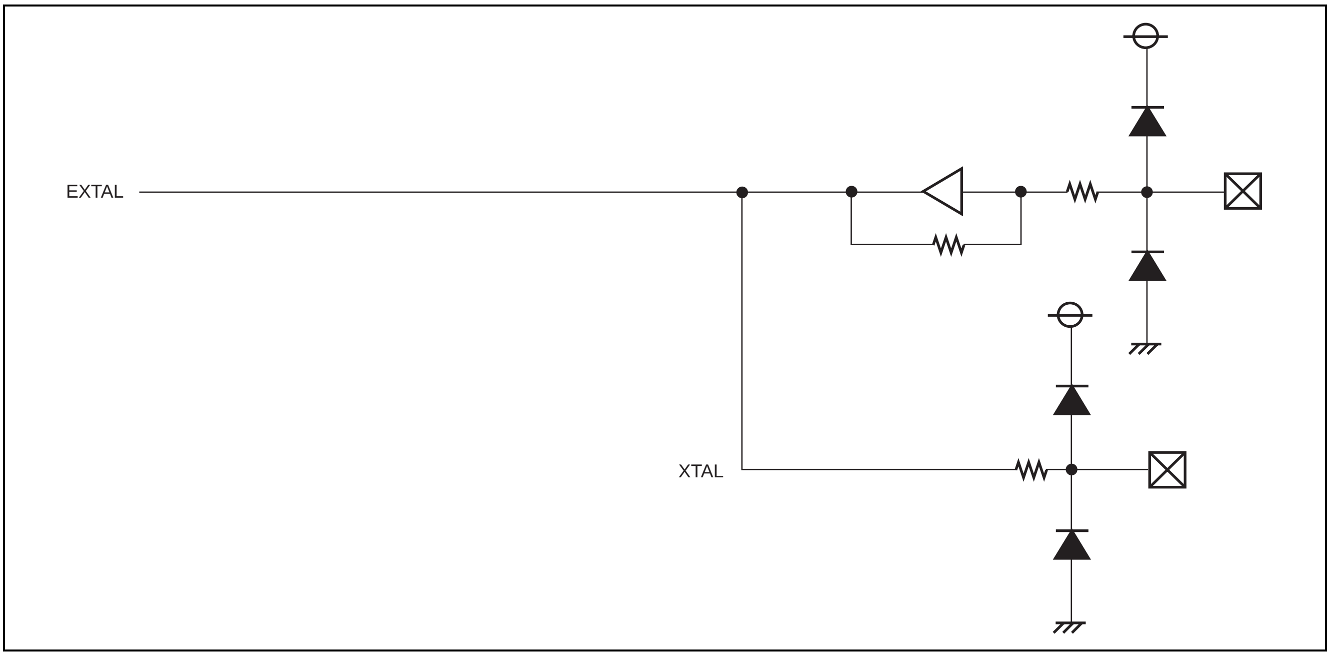


Figure 18.11 Port Peripheral Circuit Diagram (Input Port and Output Port)

- Applicable Ports
  TDO, WDTOVF#

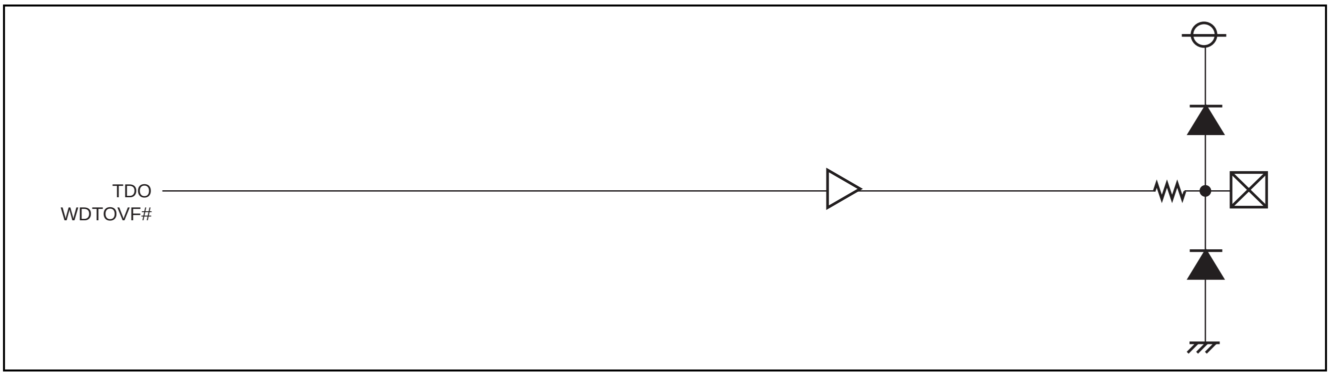


Figure 18.12 Port Peripheral Circuit Diagram (Output Port 1)

18.6 Usage Notes

18.6.1 Port Input Disable Function

To use the input function of a port on which input is disabled after the reset state is released, it is necessary to enable input by setting the port input level settings bits in the port input threshold value switching register to a value other than "B'0xxx". When port input is disabled, the state is equivalent to when a "H" level input is being received on the pin. Therefore, selecting the peripheral input function in the port operation mode register while input is disabled can result in unintended operation caused by "H" level input.

18.6.2 Peripheral Function Input When Pins are Set as General Port Pins

When the peripheral function input/general port pins listed below are set as general port pins in the operation mode register, the peripheral function input is "H" level. Therefore, when an "L" level signal is being input to a peripheral function input pin, an edge signal is input to the peripheral function input when the operation mode register is manipulated.

CRX0 to CRX3, RXD0 to RXD3, RTS0#, RTS2#, RTS3#, CTS0#, CTS2#, CTS3#, SDA, SCL, AD0TRG#, IRQ0 to IRQ2, IRQ5 to IRQ7, AUDRSYN#

When peripheral function input/general port pins other than the above are set as general port pins in the operation mode register, the peripheral function input is "L" level.

18.6.3 I/O Port Initial Setting

After a reset is canceled, set the reserved bits in the port m I/O register (PmIOR) listed as note 1 below to "1". For details of the I/O port initial setting procedure, see section 18.4, I/O Port Initial Setting Procedure Examples.

Note: *1 Port A: Bits PA15IOR and PA14IOR
Port B: Bits PB6IOR to PB4IOR and PB2IOR
Port C: Bits PC15IOR, PC13IOR to PC7IOR, and PC4IOR
Port D: Bits PD15IOR to PD11IOR
Port E: Bits PE14IOR to PE0IOR
Port F: Bits PF3IOR and PF2IOR
Port G: Bits PG7IOR to PG5IOR
Port J: Bits PJ9IOR and PJ8IOR
Port K: Bits PK7IOR and PK4IOR to PK1IOR
Port L: Bits PL7IOR, PL1IOR, and PL0IOR

This page is blank for reasons of layout.

Section 19 Bus State Controller (BSC)

The external bus state controller (BSC) function is not implemented on the SH7455 Group and SH7456 Group.

This page is blank for reasons of layout.

Section 20 Direct Memory Access Controller (DMAC)

20.1 Overview

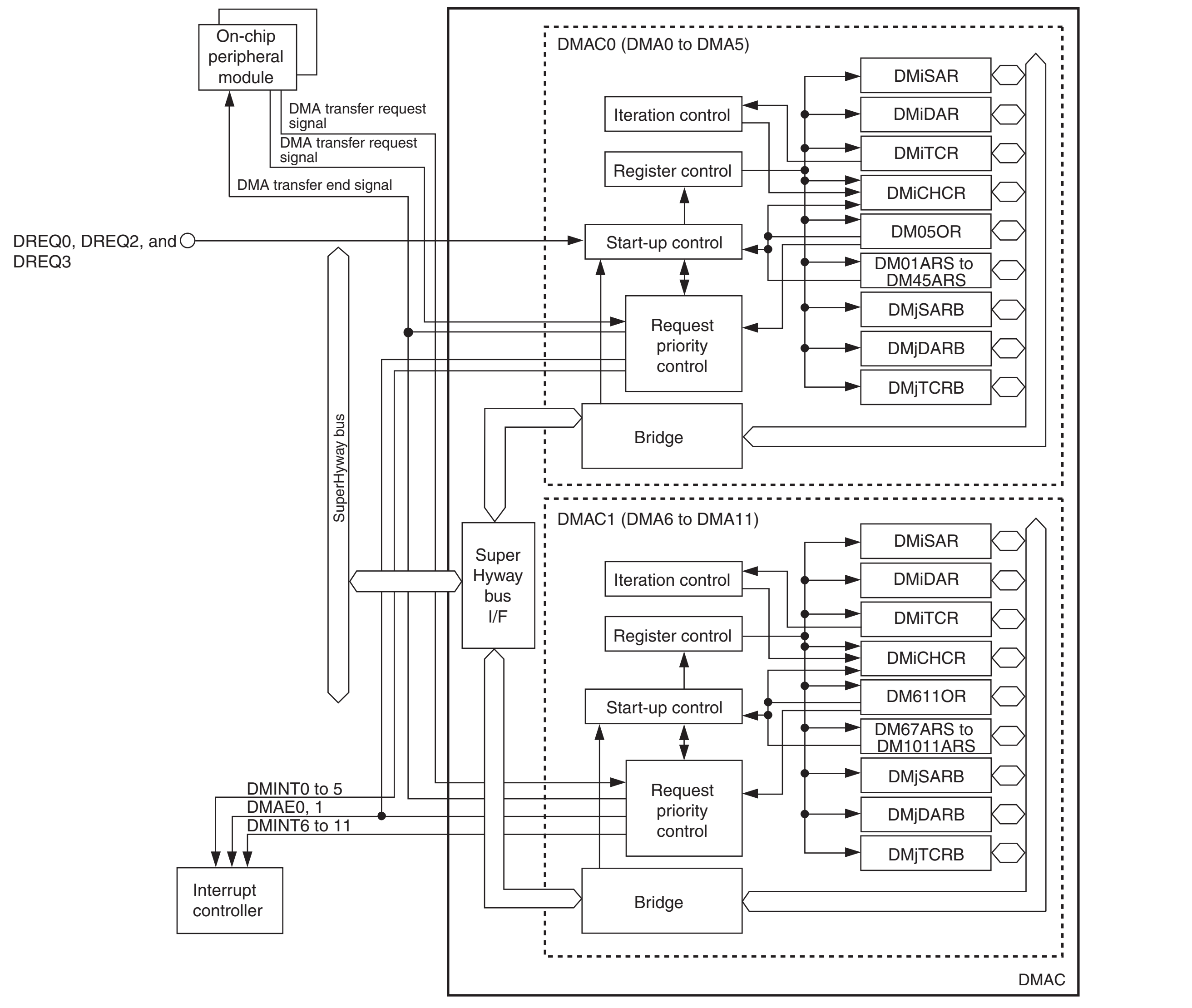
This MCU includes two modules of the direct memory access controller (DMAC). The DMAC0 module consists of six channels DMA0 to DMA5 and the DMAC1 module consists of six channels DMA6 to DMA11.

The DMAC can be used in place of the CPU to perform high-speed transfers between on-chip peripheral modules, IL memory/OL memory, SHwyRAM, and ROM. Note that in register and pin names appearing in this section, the i in the notation DMAi represents values from 0 to 11, and the j in the notation DMAj represents values from 0 to 3 and 6 to 9. Table 20.1 lists the overview of the DMAC.

Table 20.1 Overview of DMAC

Item	Description
Number of channels	6 channels (DMA0 to DMA5) + 6 channels (DMA6 to DMA11)
Transfer request sources	• Auto request (software request) • On-chip peripheral module request (SCIF, RSPI, IIC3, ATU-IIIS, ADC, DRI) • External request (DMA0 , DMA2, and DMA3 only)
Maximum number of transfers	• 16,777,216
Transferable address space	• 4-Gbyte space
Transfer areas	• Supports transfers between on-chip peripheral modules, IL memory/OL memory, SHwyRAM, and ROM.
Transfer data size	• 1 byte, 2 bytes (word), 4 bytes (long word), 16 bytes, 32 bytes
Transfer address method	• Dual address
Transfer modes	• Cycle steal mode 1, cycle steal mode 2, or burst mode
Transfer direction	• Three types can be selected for the source and destination. Address fixed Address incremented Address decremented
Channel priority	• Priority between modules The priority between the DMAC0 module (DMA0 to DMA5) and the DMAC1 module (DMA6 to DMA11) is round robin. • Channel priority within a module Either a fixed priority (DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5) or round robin can be selected for DMA0 to DMA5. Either a fixed priority (DMA6 > DMA7 > DMA8 > DMA9 > DMA10 > DMA11) or round robin can be selected for DMA6 to DMA11.
Interrupt request	• After half of the transfers ended • After all transfers ended • After an error occurred
External request detection	• DREQ0, DREQ2, and DREQ3 pin input rising edge or falling edge detection can be selected.
Others	• Repeat function and reload function

Figure 20.1 shows the block diagram of the DMAC.



Note: *1 The half-end interrupt request is available in DMA0 to DMA3 and DMA6 to DMA9 only.

Legend:

DMiCHCR	: DMAi channel control register
DMiDAR	: DMAi destination address register
DMjDARB	: DMAj destination address register B
DM05OR	: DMA05 operation register
DM611OR	: DMA611 operation register
DM01ARS to DM1011ARS	: DMA01 to DMA1011 extended resource registers
DMAINTi	: DMAi transfer end/half-end interrupt request*1
DMAE0, 1	: Channels 0 to 5 Address error interrupt request, Channels 6 to 11 Address error interrupt request
DMiSAR	: DMAi source address register
DMjSARB	: DMAj source address register B
SuperHyway bus	: SuperHyway bus (Internal bus)
DMiTCR	: DMAi transfer count register
DMjTCRB	: DMAj transfer count register B

i = 0 to 11, j = 0 to 3, and 6 to 9

Figure 20.1 Block Diagram of DMAC

20.2 Input/Output Pins

Table 20.2 lists the configuration of the pins that are connected to external devices. The DMAC has pins for three channels for external devices use.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 20.2 Pin Configuration

Channel	Pin Name	I/O	Description
DMA0	DREQ0	Input	DMA transfer request input from external device to DMA0
DMA2	DREQ2	Input	DMA transfer request input from external device to DMA2
DMA3	DREQ3	Input	DMA transfer request input from external device to DMA3

Note: • When a channel from DMA0, DMA2, and DMA3 is used for other than external request, input to the pin (DREQ0, DREQ2, and DREQ3) is ignored.

20.3 Register Descriptions

Table 20.3 shows the configuration of registers of the DMAC0 Module.

Table 20.4 shows the configuration of registers of the DMAC1 Module.

Table 20.3 Register Configuration of DMAC0 Module

Channel	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DMA0	DMA0 source address register	DM0SAR	Undefined	H'FF60 8020	32	20-8
	DMA0 destination address register	DM0DAR	Undefined	H'FF60 8024	32	20-10
	DMA0 transfer count register	DM0TCR	Undefined	H'FF60 8028	32	20-12
	DMA0 channel control register	DM0CHCR	H'4000 0040	H'FF60 802C	32	20-14
	DMA0 source address register B	DM0SARB	Undefined	H'FF60 8120	32	20-9
	DMA0 destination address register B	DM0DARB	Undefined	H'FF60 8124	32	20-11
	DMA0 transfer count register B	DM0TCRB	Undefined	H'FF60 8128	32	20-13
DMA1	DMA1 source address register	DM1SAR	Undefined	H'FF60 8030	32	20-8
	DMA1 destination address register	DM1DAR	Undefined	H'FF60 8034	32	20-10
	DMA1 transfer count register	DM1TCR	Undefined	H'FF60 8038	32	20-12
	DMA1 channel control register	DM1CHCR	H'4000 0040	H'FF60 803C	32	20-14
	DMA1 source address register B	DM1SARB	Undefined	H'FF60 8130	32	20-9
	DMA1 destination address register B	DM1DARB	Undefined	H'FF60 8134	32	20-11
	DMA1 transfer count register B	DM1TCRB	Undefined	H'FF60 8138	32	20-13
DMA2	DMA2 source address register	DM2SAR	Undefined	H'FF60 8040	32	20-8
	DMA2 destination address register	DM2DAR	Undefined	H'FF60 8044	32	20-10
	DMA2 transfer count register	DM2TCR	Undefined	H'FF60 8048	32	20-12
	DMA2 channel control register	DM2CHCR	H'4000 0040	H'FF60 804C	32	20-14
	DMA2 source address register B	DM2SARB	Undefined	H'FF60 8140	32	20-9
	DMA2 destination address register B	DM2DARB	Undefined	H'FF60 8144	32	20-11
	DMA2 transfer count register B	DM2TCRB	Undefined	H'FF60 8148	32	20-13
DMA3	DMA3 source address register	DM3SAR	Undefined	H'FF60 8050	32	20-8
	DMA3 destination address register	DM3DAR	Undefined	H'FF60 8054	32	20-10
	DMA3 transfer count register	DM3TCR	Undefined	H'FF60 8058	32	20-12
	DMA3 channel control register	DM3CHCR	H'4000 0040	H'FF60 805C	32	20-14
	DMA3 source address register B	DM3SARB	Undefined	H'FF60 8150	32	20-9
	DMA3 destination address register B	DM3DARB	Undefined	H'FF60 8154	32	20-11
	DMA3 transfer count register B	DM3TCRB	Undefined	H'FF60 8158	32	20-13
DMA4	DMA4 source address register	DM4SAR	Undefined	H'FF60 8070	32	20-8
	DMA4 destination address register	DM4DAR	Undefined	H'FF60 8074	32	20-10
	DMA4 transfer count register	DM4TCR	Undefined	H'FF60 8078	32	20-12
	DMA4 channel control register	DM4CHCR	H'4000 0040	H'FF60 807C	32	20-14

Channel	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DMA5	DMA5 source address register	DM5SAR	Undefined	H'FF60 8080	32	20-8
	DMA5 destination address register	DM5DAR	Undefined	H'FF60 8084	32	20-10
	DMA5 transfer count register	DM5TCR	Undefined	H'FF60 8088	32	20-12
	DMA5 channel control register	DM5CHCR	H'4000 0040	H'FF60 808C	32	20-14
DMA0, DMA1	DMA01 extended resource select register	DM01ARS	H'0000	H'FF60 9000	16	20-22
DMA2, DMA3	DMA23 extended resource select register	DM23ARS	H'0000	H'FF60 9004	16	20-22
DMA4, DMA5	DMA45 extended resource select register	DM45ARS	H'0000	H'FF60 9008	16	20-23
DMA0 to DMA5 common	DMA05 operation register	DM05OR	H'0000	H'FF60 8060	16	20-19

Notes:
- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.
- Do to perform accesses using other than the specified access size.

Table 20.4 Register Configuration of DMAC1 Module

Channel	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DMA6	DMA6 source address register	DM6SAR	Undefined	H'FF61 8020	32	20-8
	DMA6 destination address register	DM6DAR	Undefined	H'FF61 8024	32	20-10
	DMA6 transfer count register	DM6TCR	Undefined	H'FF61 8028	32	20-12
	DMA6 channel control register	DM6CHCR	H'4000 0040	H'FF61 802C	32	20-14
	DMA6 source address register B	DM6SARB	Undefined	H'FF61 8120	32	20-9
	DMA6 destination address register B	DM6DARB	Undefined	H'FF61 8124	32	20-11
	DMA6 transfer count register B	DM6TCRB	Undefined	H'FF61 8128	32	20-13
DMA7	DMA7 source address register	DM7SAR	Undefined	H'FF61 8030	32	20-8
	DMA7 destination address register	DM7DAR	Undefined	H'FF61 8034	32	20-10
	DMA7 transfer count register	DM7TCR	Undefined	H'FF61 8038	32	20-12
	DMA7 channel control register	DM7CHCR	H'4000 0040	H'FF61 803C	32	20-14
	DMA7 source address register B	DM7SARB	Undefined	H'FF61 8130	32	20-9
	DMA7 destination address register B	DM7DARB	Undefined	H'FF61 8134	32	20-11
	DMA7 transfer count register B	DM7TCRB	Undefined	H'FF61 8138	32	20-13
DMA8	DMA8 source address register	DM8SAR	Undefined	H'FF61 8040	32	20-8
	DMA8 destination address register	DM8DAR	Undefined	H'FF61 8044	32	20-10
	DMA8 transfer count register	DM8TCR	Undefined	H'FF61 8048	32	20-12
	DMA8 channel control register	DM8CHCR	H'4000 0040	H'FF61 804C	32	20-14
	DMA8 source address register B	DM8SARB	Undefined	H'FF61 8140	32	20-9

Channel	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DMA8	DMA8 destination address register B	DM8DARB	Undefined	H'FF61 8144	32	20-11
	DMA8 transfer count register B	DM8TCRB	Undefined	H'FF61 8148	32	20-13
DMA9	DMA9 source address register	DM9SAR	Undefined	H'FF61 8050	32	20-8
	DMA9 destination address register	DM9DAR	Undefined	H'FF61 8054	32	20-10
	DMA9 transfer count register	DM9TCR	Undefined	H'FF61 8058	32	20-12
	DMA9 channel control register	DM9CHCR	H'4000 0040	H'FF61 805C	32	20-14
	DMA9 source address register B	DM9SARB	Undefined	H'FF61 8150	32	20-9
	DMA9 destination address register B	DM9DARB	Undefined	H'FF61 8154	32	20-11
	DMA9 transfer count register B	DM9TCRB	Undefined	H'FF61 8158	32	20-13
DMA10	DMA10 source address register	DM10SAR	Undefined	H'FF61 8070	32	20-8
	DMA10 destination address register	DM10DAR	Undefined	H'FF61 8074	32	20-10
	DMA10 transfer count register	DM10TCR	Undefined	H'FF61 8078	32	20-12
	DMA10 channel control register	DM10CHCR	H'4000 0040	H'FF61 807C	32	20-14
DMA11	DMA11 source address register	DM11SAR	Undefined	H'FF61 8080	32	20-8
	DMA11 destination address register	DM11DAR	Undefined	H'FF61 8084	32	20-10
	DMA11 transfer count register	DM11TCR	Undefined	H'FF61 8088	32	20-12
	DMA11 channel control register	DM11CHCR	H'4000 0040	H'FF61 808C	32	20-14
DMA6, DMA7	DMA67 extended resource select register	DM67ARS	H'0000	H'FF61 9000	16	20-23
DMA8, DMA9	DMA89 extended resource select register	DM89ARS	H'0000	H'FF61 9004	16	20-23
DMA10, DMA11	DMA1011 extended resource select register	DM1011ARS	H'0000	H'FF61 9008	16	20-24
DMA6 to DMA11 common	DMA611 operation register	DM611OR	H'0000	H'FF61 8060	16	20-19

Notes:
- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.
- Do to perform accesses using other than the specified access size.

20.3.1 DMAi Source Address Register (DMiSAR)

The DMiSAR register specifies the source address of a DMA transfer.

For the address to be specified as the transfer source, set it according to the address boundary for the transfer data size. In addition, the DMAC is constrained by the 32-bit physical address space, so addresses in the P0 or P4 area should be set as the transfer source.

DMA0 Source Address Register (DM0SAR)	<P4 address: location H'FF60 8020>
DMA1 Source Address Register (DM1SAR)	<P4 address: location H'FF60 8030>
DMA2 Source Address Register (DM2SAR)	<P4 address: location H'FF60 8040>
DMA3 Source Address Register (DM3SAR)	<P4 address: location H'FF60 8050>
DMA4 Source Address Register (DM4SAR)	<P4 address: location H'FF60 8070>
DMA5 Source Address Register (DM5SAR)	<P4 address: location H'FF60 8080>
DMA6 Source Address Register (DM6SAR)	<P4 address: location H'FF61 8020>
DMA7 Source Address Register (DM7SAR)	<P4 address: location H'FF61 8030>
DMA8 Source Address Register (DM8SAR)	<P4 address: location H'FF61 8040>
DMA9 Source Address Register (DM9SAR)	<P4 address: location H'FF61 8050>
DMA10 Source Address Register (DM10SAR)	<P4 address: location H'FF61 8070>
DMA11 Source Address Register (DM11SAR)	<P4 address: location H'FF61 8080>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMiSAR															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMiSAR															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	DMiSAR	Undefined	R	W	Specifies the DMA transfer source address. When read, these bits return the next DMA transfer source address.

20.3.2 DMAj Source Address Register B (DMjSARB)

The DMjSARB register specifies the source address of a DMA transfer that is set in the DMjSAR register again with the repeat/reload function. For the repeat function, see section 20.4.7, Repeat Function. For the reload function, see section 20.4.8, Reload Function.

Data to be written to the DMjSAR register is also automatically written to the DMjSARB register. To set the DMjSARB register address that differs from the DMjSAR address, write data to the DMjSARB register after the DMjSAR register.

For the address to be specified as the transfer source, set it according to the address boundary for the transfer data size. In addition, the DMAC is constrained by the 32-bit physical address space, so addresses in the P0 or P4 area should be set as the transfer source.

DMA0 Source Address Register B (DM0SARB)	<P4 address: location H'FF60 8120>
DMA1 Source Address Register B (DM1SARB)	<P4 address: location H'FF60 8130>
DMA2 Source Address Register B (DM2SARB)	<P4 address: location H'FF60 8140>
DMA3 Source Address Register B (DM3SARB)	<P4 address: location H'FF60 8150>
DMA6 Source Address Register B (DM6SARB)	<P4 address: location H'FF61 8120>
DMA7 Source Address Register B (DM7SARB)	<P4 address: location H'FF61 8130>
DMA8 Source Address Register B (DM8SARB)	<P4 address: location H'FF61 8140>
DMA9 Source Address Register B (DM9SARB)	<P4 address: location H'FF61 8150>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMjSARB															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMjSARB															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	DMjSARB	Undefined	R	W	Specifies the DMA transfer source address reset in the DMjSAR register with the repeat/reload function.

20.3.3 DMAi Destination Address Register (DMiDAR)

The DMiDAR register specifies the destination address of a DMA transfer.

For the address to be specified as the transfer destination, set it according to the address boundary for the transfer data size. In addition, the DMAC is constrained by the 32-bit physical address space, so addresses in the P0 or P4 area should be set as the transfer destination.

DMA0 Destination Address Register (DM0DAR)	<P4 address: location H'FF60 8024>
DMA1 Destination Address Register (DM1DAR)	<P4 address: location H'FF60 8034>
DMA2 Destination Address Register (DM2DAR)	<P4 address: location H'FF60 8044>
DMA3 Destination Address Register (DM3DAR)	<P4 address: location H'FF60 8054>
DMA4 Destination Address Register (DM4DAR)	<P4 address: location H'FF60 8074>
DMA5 Destination Address Register (DM5DAR)	<P4 address: location H'FF60 8084>
DMA6 Destination Address Register (DM6DAR)	<P4 address: location H'FF61 8024>
DMA7 Destination Address Register (DM7DAR)	<P4 address: location H'FF61 8034>
DMA8 Destination Address Register (DM8DAR)	<P4 address: location H'FF61 8044>
DMA9 Destination Address Register (DM9DAR)	<P4 address: location H'FF61 8054>
DMA10 Destination Address Register (DM10DAR)	<P4 address: location H'FF61 8074>
DMA11 Destination Address Register (DM11DAR)	<P4 address: location H'FF61 8084>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMiDAR															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMiDAR															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	DMiDAR	Undefined	R	W	Specifies the DMA transfer source address. When read, these bits return the next DMA transfer destination address.

20.3.4 DMAj Destination Address Register B (DMjDARB)

The DMjDARB register specifies the destination address of a DMA transfer that is set in the DMjDAR register again with the repeat/reload function. For the repeat function, see section 20.4.7, Repeat Function. For the reload function, see section 20.4.8, Reload Function.

Data to be written to the DMjDAR register is also automatically written to the DMjDARB register. To set DMjDARB address that differs from the DMjDAR address, write data to the DMjDARB register after the DMjDAR register.

For the address to be specified as the transfer destination, set it according to the address boundary for the transfer data size. In addition, the DMAC is constrained by the 32-bit physical address space, so addresses in the P0 or P4 area should be set as the transfer destination.

DMA0 Destination Address Register B (DM0DARB)	<P4 address: location H'FF60 8124>
DMA1 Destination Address Register B (DM1DARB)	<P4 address: location H'FF60 8134>
DMA2 Destination Address Register B (DM2DARB)	<P4 address: location H'FF60 8144>
DMA3 Destination Address Register B (DM3DARB)	<P4 address: location H'FF60 8154>
DMA6 Destination Address Register B (DM6DARB)	<P4 address: location H'FF61 8124>
DMA7 Destination Address Register B (DM7DARB)	<P4 address: location H'FF61 8134>
DMA8 Destination Address Register B (DM8DARB)	<P4 address: location H'FF61 8144>
DMA9 Destination Address Register B (DM9DARB)	<P4 address: location H'FF61 8154>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMjDARB															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMjDARB															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	DMjDARB	Undefined	R	W	Specifies the DMA transfer destination address reset in the DMjDAR register with the repeat/reload function.

20.3.5 DMAi Transfer Count Register (DMiTCR)

The DMiTCR register specifies the DMA transfer count. The number of transfers is 1 when the setting is H'0000 0001, 16,777,215 when H'00FF FFFF is set, and 16,777,216 (the maximum) when H'0000 0000 is set. During a DMA transfer, these registers indicate the remaining transfer count.

DMA0 Transfer Count Register (DM0TCR)	<P4 address: location H'FF60 8028>
DMA1 Transfer Count Register (DM1TCR)	<P4 address: location H'FF60 8038>
DMA2 Transfer Count Register (DM2TCR)	<P4 address: location H'FF60 8048>
DMA3 Transfer Count Register (DM3TCR)	<P4 address: location H'FF60 8058>
DMA4 Transfer Count Register (DM4TCR)	<P4 address: location H'FF60 8078>
DMA5 Transfer Count Register (DM5TCR)	<P4 address: location H'FF60 8088>
DMA6 Transfer Count Register (DM6TCR)	<P4 address: location H'FF61 8028>
DMA7 Transfer Count Register (DM7TCR)	<P4 address: location H'FF61 8038>
DMA8 Transfer Count Register (DM8TCR)	<P4 address: location H'FF61 8048>
DMA9 Transfer Count Register (DM9TCR)	<P4 address: location H'FF61 8058>
DMA10 Transfer Count Register (DM10TCR)	<P4 address: location H'FF61 8078>
DMA11 Transfer Count Register (DM11TCR)	<P4 address: location H'FF61 8088>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DMiTCR							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMiTCR															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	—	Undefined	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
23 to 0	DMiTCR	Undefined	R	W	Specifies the number of DMA transfers. When read, these bits return the number of DMA transfers remaining. After transfers are completed, these bits are read as "0".

20.3.6 DMAj Transfer Count Register B (DMjTCRB)

Data to be written to the DMjTCR is also automatically written to the DMjTCRB register. To set the DMjTCRB register address that differs from the DMjTCR address, write data to the DMjTCRB register after the DMjTCR register.

When the repeat function is enabled, the DMjTCRB specifies the number of DMA transfers that are reset in the DMjTCR register. When the half-end function is enabled, the DMjTCRB register is used as the initial value hold register for half-end detection.

When the reload function is enabled, the DMjTCRB sets the number of DMA transfers and is used as a transfer counter. Bits 7 to 0 operate as a transfer counter. The contents of the DMjSAR and DMjDAR registers are updated when the value of these bits reaches "0", and then the value of bits 23 to 16 in the DMjTCRB register are loaded in bits 7 to 0. Bits 23 to 16 specify the number of transfers until reload. Set bits 23 to 16 and 7 to 0 to the same value, and clear bits 15 to 8 to "0". Also, clear the HIE bit in the DMjCHCR register to "0" and do not use the half-end function when the reload function is used. For the repeat function, see section 20.4.7, Repeat Function. For the reload function, see section 20.4.8, Reload Function.

DMA0 Transfer Count Register B (DM0TCRB)	<P4 address: location H'FF60 8128>
DMA1 Transfer Count Register B (DM1TCRB)	<P4 address: location H'FF60 8138>
DMA2 Transfer Count Register B (DM2TCRB)	<P4 address: location H'FF60 8148>
DMA3 Transfer Count Register B (DM3TCRB)	<P4 address: location H'FF60 8158>
DMA6 Transfer Count Register B (DM6TCRB)	<P4 address: location H'FF61 8128>
DMA7 Transfer Count Register B (DM7TCRB)	<P4 address: location H'FF61 8138>
DMA8 Transfer Count Register B (DM8TCRB)	<P4 address: location H'FF61 8148>
DMA9 Transfer Count Register B (DM9TCRB)	<P4 address: location H'FF61 8158>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DMjTCRB							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMjTCRB															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	—	Undefined	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
23 to 0	DMjTCRB	Undefined	R	W	• When the repeat function is enabled, the DMjTCRB specifies the number of DMA transfers that are reset in the DMjTCR register. When the half-end function is enabled, the DMjTCRB register is used as the initial value hold register for half-end detection. • When the reload function is enabled: Bits 23 to 16: Specify the number of transfers until reload. Bits 15 to 8: Cleared to "0". Bits 7 to 0: Operate as a DMA transfer counter. Set bits 23 to 16 and 7 to 0 to the same value. Clear the HIE bit in the DMjCHCR register to "0" and do not use the half-end function when the reload function is enabled.

20.3.7 DMAi Channel Control Register (DMiCHCR)

The DMiCHCR register controls the DMA transfer mode.

DMA0 Channel Control Register (DM0CHCR)	<P4 address: location H'FF60 802C>
DMA1 Channel Control Register (DM1CHCR)	<P4 address: location H'FF60 803C>
DMA2 Channel Control Register (DM2CHCR)	<P4 address: location H'FF60 804C>
DMA3 Channel Control Register (DM3CHCR)	<P4 address: location H'FF60 805C>
DMA4 Channel Control Register (DM4CHCR)	<P4 address: location H'FF60 807C>
DMA5 Channel Control Register (DM5CHCR)	<P4 address: location H'FF60 808C>
DMA6 Channel Control Register (DM6CHCR)	<P4 address: location H'FF61 802C>
DMA7 Channel Control Register (DM7CHCR)	<P4 address: location H'FF61 803C>
DMA8 Channel Control Register (DM8CHCR)	<P4 address: location H'FF61 804C>
DMA9 Channel Control Register (DM9CHCR)	<P4 address: location H'FF61 805C>
DMA10 Channel Control Register (DM10CHCR)	<P4 address: location H'FF61 807C>
DMA11 Channel Control Register (DM11CHCR)	<P4 address: location H'FF61 808C>

<table>
<tr><td>Bit:</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td></td><td>—</td><td>LCKN</td><td>—</td><td>—</td><td colspan="3">RPT</td><td>—</td><td>—</td><td>—</td><td>—</td><td>TS2</td><td>HE</td><td>HIE</td><td>—</td><td>—</td></tr>
<tr><td>After Reset:</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

<table>
<tr><td>Bit:</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td></td><td colspan="2">DM</td><td colspan="2">SM</td><td colspan="4">RS</td><td>DL</td><td>—</td><td>TB</td><td>TS1</td><td>TS0</td><td>IE</td><td>TE</td><td>DE</td></tr>
<tr><td>After Reset:</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

<After Reset: H'4000 0040>

Bit	Abbreviation	After Reset	R	W	Description
31	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
30	LCKN	1	R	W	Bus Mastership Release Bit Specifies whether to keep or release the bus mastership for read/write cycles in one transfer. Releasing the bus mastership allows the bus mastership request for this bus master to be accepted. For details, see section 20.4.2, DMA Transfer Modes. Set this bit to "0" in burst mode. 0: Bus mastership is kept 1: Bus mastership is released
29, 28	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
27 to 25	RPT	All 0	R	W	DMA Setting Renewal Specify Bits These bits are enabled in the DMjCHCR register. In the DMA4CHCR, DMA5CHCR, DMA10CHCR, and DMA11CHCR registers, the write value should be always be "000". These bits are always read as "000". 000: Repeat function, reload function disabled 001: Repeat function enabled (SAR, DAR, TCR) Reload DMjSARB to DMjSAR, DMjDARB to DMjDAR, and DMjTCRB to DMjTCR 010: Repeat function enabled (DAR, TCR) Reload DMjDARB to DMjDAR, and DMjTCRB to DMjTCR 011: Repeat function enabled (SAR, TCR) Reload DMjSARB to DMjSAR, and DMjTCRB to DMjTCR 100: Reserved (setting prohibited) 101: Reload function enabled (SAR, DAR, TCRB) Reload DMjSARB to DMjSAR, DMjDARB to DMjDAR, and DMjTCRB [23:16] to DMjTCRB [7:0] 110: Reload function enabled (DAR, TCRB) Reload DMjDARB to DMjDAR, and DMjTCRB [23:16] to DMjTCRB [7:0] 111: Reload function enabled (SAR, TCRB) Reload DMjSARB to DMjSAR, and DMjTCRB [23:16] to DMjTCRB [7:0]
24 to 21	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
20	TS2	0	R	W	Transfer Size Specify Bit With TS1 and TS0, this bit specifies the DMA data transfer size. When the transfer source or transfer destination is a register of an on-chip peripheral module with a transfer size set, set a transfer size according to the specified access size . For the address to be specified as the transfer source or destination, set it according to the address boundary for the transfer data size. b20 b4 b3 0 0 0 : 1-byte units transfer 0 0 1 : Word (2-byte) units transfer 0 1 0 : Longword (4-byte) units transfer 0 1 1 : 16-byte units transfer (8 bytes × 2) 1 0 0 : 32-byte units transfer (8 bytes × 4) Other than above: Setting prohibited Note: 16-byte and 32-byte accesses are possible between IL memory/OL memory, SHwyRAM, and ROM only.

Bit	Abbreviation	After Reset	R	W	Description
19	HE	0	R	*[1]	Half End Interrupt Request Status Flag After the HIE bit is set to "1" and the DMjTCR register value becomes half of the DMjTCRB register value (1 bit shift to right), the HE bit becomes "1". This bit is valid only in the DMjCHCR register. In the DMA4CHCR, DMA5CHCR, DMA10CHCR, and DMA11CHCR registers, the write value should be always "0". This bit is always read as "0". 0: Half end interrupt is not requested 1: Half end interrupt is requested (DMjTCR register = DMjTCRB register ÷ 2) [Condition for clearing to "0"] Writing "0" to the HE bit after reading it as "1"*[1]
18	HIE	0	R	W	Half End Interrupt Enable Bit When the HIE bit is set to "1" and the HE bit is set, an interrupt request is generated to the INTC. Clear this bit to "0" while the reload function is enabled. This bit is valid in the DMjCHCR register. In the DMA4CHCR, DMA5CHCR, DMA10CHCR, and DMA11CHCR registers, the write value should be always "0". This bit is always read as "0". 0: Half end interrupt request is disabled 1: Half end interrupt request is enabled
17, 16	—	All 0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
15, 14	DM	00	R	W	Destination Address Direction Select Bits Specify whether the DMA destination address is incremented, decremented, or fixed. 00: Address fixed 01: Address incremented +1 in byte units transfer +2 in 2-byte (word) units transfer +4 in 4-byte (longword) units transfer +16 in 16-byte units transfer +32 in 32-byte units transfer 10: Address decremented –1 in byte units transfer –2 in 2-byte (word) units transfer –4 in 4-byte (longword) units transfer Setting prohibited in 16/32-byte units transfer 11: Setting prohibited Note: When the transfer data size is set as 16-byte or 32-byte units with the TS0 to TS2 bits, do not set the DM bit to address decremented.

Bit	Abbreviation	After Reset	R	W	Description
13, 12	SM	00	R	W	Source Address Direction Select Bits Specify whether the DMA destination address is incremented, decremented, or fixed. 00: Address fixed 01: Address incremented +1 in byte units transfer +2 in 2-byte (word) units transfer +4 in 4-byte (longword) units transfer +16 in 16-byte units transfer +32 in 32-byte units transfer 10: Address decremented –1 in byte units transfer –2 in 2-byte (word) units transfer –4 in 4-byte (longword) units transfer Setting prohibited in 16/32-byte units transfer 11: Setting prohibited Note: When the transfer data size is set as 16-byte or 32-byte units with the TS0 to TS2 bits, do not set the DM bit to address decremented.
11 to 8	RS	All 0	R	W	Resource Select Bits These bits are used to set the transfer request source. The changing of transfer request source should be done in the state that the DMA enable DE bit is cleared to "0". 0000: External request 0100: Auto request (software) 1000: On-chip peripheral module request selected by DMA extended resource selector (DM01ARS to DM1011ARS) Other than above: Setting prohibited Note: No DMA transfer for external request can be selected in the DM1CHCR and DM4CHCR to DM11CHCR registers.
7	DL	0	R	W	DREQ Edge Select Bit Specifies the detecting method of the DREQ pin input. This bit is valid only in the DM0CHCR, DM2CHCR, and DM3CHCR registers. When setting the DM1CHCR and DM4CHCR to DM11CHCR registers, the write value should be always "0". The read value is always "0". Also, if the transfer request source is specified as an on-chip peripheral module or if an auto-request is specified, these bits are invalid. 0: DREQ detected at falling edge 1: DREQ detected at rising edge
6	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".

Bit	Abbreviation	After Reset	R	W	Description
5	TB	0	R	W	Transfer Bus Mode Bit Specifies the DMA transfer mode. When on-chip module request is set, select cycle steal mode 1 or cycle steal mode 2. Use the LCKN bit in the DMiCHCR register to select cycle steal mode 1 or cycle steal mode 2. 0: Cycle steal mode 1 or cycle steal mode 2 1: Burst mode
4	TS1	0	R	W	Transfer Size Specify Bit See the description of TS2 bit.
3	TS0	0	R	W	
2	IE	0	R	W	DMA Transfer Complete Interrupt Enable Bit Specifies whether or not to notify an interrupt request specified by the TE bit to the INTC at the final DMA transfer. 0: DMA transfer complete interrupt is disabled 1: DMA transfer complete interrupt is enabled
1	TE	0	R	*1	DMA Transfer Complete Interrupt Request Status Flag After the bus mastership is obtained at the the final transfer, the DMiTCR register value reaches "0", a read cycle is performed, and the TE bit is set to "1". If the IE bit is "1" at this timing, an interrupt request is notified to the INTC. Then a write cycle at the final transfer is performed. If the LCKN bit is "1", however, the bus mastership is released once, and a write cycle is performed after obtaining the mastership again. During DMA operation, if the DMA transfer complete conditions are satisfied before the DMiTCR register value reaches "0" and the DMA transfer is forcibly terminated, the TE bit is not set. 0: No DMA transfer complete interrupt request 1: DMA transfer complete interrupt requested [Condition for clearing to "0"] Writing "0" to the TE bit after reading it as "1"*1
0	DE	0	R	W	DMA Enable Bit Enables or disables the DMA transfer. If the DMA transfer end conditions (including clearing the DE bit to "0") are satisfied during DMA transfer, the transfer is forcibly terminated. When terminating the DMA transfer, the transfer request from the corresponding on-chip peripheral module must be cleared. If the CPU attempts to write "1" to the DE bit during burst transfer, this bit is set to "1" after the bust transfer is completed because the DMAC keeps the bus mastership. For auto request, if the DMA transfer start conditions (including the DE bit) are satisfied, the transfer starts. For external request or on-chip peripheral module request, if DMA transfer request is generated from the corresponding device or on-chip peripheral module after the DMA transfer start conditions (including the DE bit) are satisfied, the transfer starts. 0: DMA transfer disabled 1: DMA transfer enabled For details, refer to section 20.4.3, DMA Transfer Start Conditions and DMA Transfer End Conditions.

Note: *1 To clear the TE and HE bits, read the register and then write "0" to the flag bit only if its value was read as "1". Write "1" to the bits whose value was read as "0". Always write "1" to flag bits you do not wish to clear. Write the previous values to the bits other than TE and HE.

20.3.8 DMA05 and DMA611 Operation Registers (DM05OR and DM611OR)

The DM05OR and DM611OR registers specify the priority level of channels at the DMA transfer. This register shows the DMA transfer status.

DM05OR is a common register for DMA0 to DMA5, and DM611OR is a common register for DMA6 to DMA11.

DMA05 Operation Register (DM05OR) <P4 address: location H'FF60 8060>
DMA611 Operation Register (DM611OR) <P4 address: location H'FF61 8060>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PR		—	—	—	—	—	AE	NMIF	DME
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9, 8	PR	00	R	W	Priority Select Bits Select the priority level between channels when there are transfer requests for multiple channels simultaneously. When the PR bits in the DM05OR register are set to "11", do not the mix cycle steal mode 1 (cycle steal mode 2) and burst mode among DMA6 to DMA11. 00: DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5 (DM05OR register) DMA6 > DMA7 > DMA8 > DMA9 > DMA10 > DMA11 (DM611OR register) 01: DMA0 > DMA2 > DMA3 > DMA1 > DMA4 > DMA5 (DM05OR register) DMA6 > DMA8 > DMA9 > DMA7 > DMA10 > DMA11 (DM611OR register) 10: Setting prohibited 11: DMA0 to DMA5 in round robin (DM05OR register) DMA6 to DMA11 in round robin (DM611OR register)
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2	AE	0	R	*1	Address Error Flag Indicates that an address error encountered by the DMAC during DMA transfer. This bit is set to "1" under the following conditions: • The value set in the DMiSAR or DMiDAR register does not match to the transfer size boundary. • The transfer source or transfer destination is invalid space. 0: No address error 1: Address error occurred [Condition for clearing to "0"] Writing "0" to the AE bit after reading it as "1"*1
1	NMIF	0	R	*1	NMI Flag This flag indicates that NMI is input. When the NMI is input, the DMA transfer in progress can be done in at least one transfer unit, and the DMA transfers on all channels are forcibly terminated. To start a new transfer, set all channels again after returning from the NMI interrupt routine. When the DMAC is not in operational, the NMIF bit is set to "1" even if the NMI is input. The NMIF bit is also set to "1" even if the NMI is input while the NMI block bit (NMIB) in the ICR0 register is "0" and the BL bit in the SR register is "1". 0: No NMI interrupt 1: NMI interrupt occurred [Condition for clearing to "0"] Writing "0" to the NMIF bit after reading it as "1"*1

Bit	Abbreviation	After Reset	R	W	Description
0	DME	0	R	W	DMA Master Enable Bit This bit enables or disables DMA transfers on all channels corresponding to the DM05OR register (DMA0 to DMA5) and all channels corresponding to the DM611OR register (DMA6 to DMA11). If the DMA transfer end conditions (including clearing the DME bit to "0") are satisfied during DMA transfers, the transfers are forcibly terminated. To terminate DMA transfers, the DMA transfer request from the corresponding on-chip peripheral module must be cleared. If the CPU attempts to write "1" to the DME bit during burst transfers, this bit is set to "1" after the burst transfers are completed because the DMAC keeps the bus mastership. For auto request, if the DMA transfer start conditions (including the DME bit) are satisfied, the transfers start. For external request or on-chip peripheral module request, if the DMA transfer request is generated from the corresponding device or on-chip peripheral module after the DMA transfer start conditions (including the DME bit) are satisfied, the transfers start. 0: Disables DMA transfers on DMA0 to DMA5 (DM05OR register) Disables DMA transfers on DMA6 to DMA11 (DM611OR register) 1: Enables DMA transfers on channels DMA0 to DMA5 (DM05OR register) Enables DMA transfers on DMA6 to DMA11 (DM611OR register) For details, refer to section 20.4.3, DMA Transfer Start Conditions and DMA Transfer End Conditions.

Note: *1 To clear the AE and NMIF bits, read the register and then write "0" to the flag bit only if its value was read as "1". Write "1" to the bits whose value was read as "0". Always write "1" to flag bits you do not wish to clear. Write the previous values to the bits other than AE and NMIF.

20.3.9 DMA01 to DMA1011 Extended Resource Select Registers (DM01ARS to DM1011ARS)

The DM01ARS to DM1011ARS registers transfer requests from on-chip peripheral modules (the SCIF, RSPI, IIC, ATU-IIIS, ADC, and DRI).

When a value other than the values listed in table 20.5 is set in the CnMRID bit, DMA operation cannot be guaranteed. In addition, a transfer request from an on-chip peripheral module should not be assigned to multiple DMAC channels as a resource. Otherwise, correct operation cannot be guaranteed.

(1) DMA01 Extended Resource Select Register (DM01ARS)

DMA01 Extended Resource Select Register (DM01ARS) <P4 address: location H'FF60 9000>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C1MRID								C0MRID							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	C1MRID	All 0	R	W	Transfer request module ID for DMA1 See table 20.5.
7 to 0	C0MRID	All 0	R	W	Transfer request module ID for DMA0 See table 20.5.

(2) DMA23 Extended Resource Select Register (DM23ARS)

DMA23 Extended Resource Select Register (DM23ARS) <P4 address: location H'FF60 9004>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C3MRID								C2MRID							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	C3MRID	All 0	R	W	Transfer request module ID for DMA3 See table 20.5.
7 to 0	C2MRID	All 0	R	W	Transfer request module ID for DMA2 See table 20.5.

(3) DMA45 Extended Resource Select Register (DM45ARS)

DMA45 Extended Resource Select Register (DM45ARS) <P4 address: location H'FF60 9008>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C5MRID								C4MRID							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	C5MRID	All 0	R	W	Transfer request module ID for DMA5 See table 20.5.
7 to 0	C4MRID	All 0	R	W	Transfer request module ID for DMA4 See table 20.5.

(4) DMA67 Extended Resource Select Register (DM67ARS)

DMA67 Extended Resource Select Register (DM67ARS) <P4 address: location H'FF61 9000>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C7MRID								C6MRID							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	C7MRID	All 0	R	W	Transfer request module ID for DMA7 See table 20.5.
7 to 0	C6MRID	All 0	R	W	Transfer request module ID for DMA6 See table 20.5.

(5) DMA89 Extended Resource Select Register (DM89ARS)

DMA89 Extended Resource Select Register (DM89ARS) <P4 address: location H'FF61 9004>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C9MRID								C8MRID							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	C9MRID	All 0	R	W	Transfer request module ID for DMA9 See table 20.5.
7 to 0	C8MRID	All 0	R	W	Transfer request module ID for DMA8 See table 20.5.

(6) DMA1011 Extended Resource Select Register (DM1011ARS)

DMA1011 Extended Resource Select Register (DM1011ARS) <P4 address: location H'FF61 9008>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C11MRID								C10MRID							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	C11MRID	All 0	R	W	Transfer request module ID for DMA11 See table 20.5.
7 to 0	C10MRID	All 0	R	W	Transfer request module ID for DMA10 See table 20.5.

Table 20.5 Transfer Request Sources for On-Chip Peripheral Module Request

On-Chip Peripheral Module	DMA Transfer Request Sources	Setting Value for CnMRID Bit	Transfer Source	Transfer Destination
SCIF	SCIF0 transmit FIFO data empty	H'01	Any	SC0FTDR
	SCIF0 receive FIFO data full	H'02	SC0FRDR	Any
	SCIF1 transmit FIFO data empty	H'05	Any	SC1FTDR
	SCIF1 receive FIFO data full	H'06	SC1FRDR	Any
	SCIF2 transmit FIFO data empty	H'09	Any	SC2FTDR
	SCIF2 receive FIFO data full	H'0A	SC2FRDR	Any
	SCIF3 transmit FIFO data empty	H'0D	Any	SC3FTDR
	SCIF3 receive FIFO data full	H'0E	SC3FRDR	Any
RSPI	RSPI0 transmit buffer empty	H'11	Any	SP0DR
	RSPI0 receive buffer full	H'12	SP0DR	Any
	RSPI1 transmit buffer empty	H'15	Any	SP1DR
	RSPI1 receive buffer full	H'16	SP1DR	Any
	RSPI2 transmit buffer empty	H'19	Any	SP2DR
	RSPI2 receive buffer full	H'1A	SP2DR	Any
IIC3	IIC3 transmit data empty	H'1D	Any	ICDRT
	IIC3 receive data full	H'1E	ICDRR	Any
ATU-IIIS	Timer A0 channel0 input capture	H'27	Any	Any
	Timer A0 channel1 input capture	H'2B	Any	Any
	Timer A0 channel2 input capture	H'2F	Any	Any
	Timer A0 channel3 input capture	H'33	Any	Any
	Timer A0 channel4 input capture	H'37	Any	Any
	(Reserved)	H'3B	—	—
	Timer F0 input capture	H'3F	TF0CDR	Any
	Timer F1 input capture	H'43	TF1CDR	Any
	Timer G3 compare-match	H'47	Any	Any
	Timer G4 compare-match	H'4B	Any	Any
	Timer G5 compare-match	H'4F	Any	Any
	Timer TOU0_0 counter underflow	H'53	Any	Any
	Timer TOU0_7 counter underflow	H'57	Any	Any
	Timer TOU1_0 counter underflow	H'5B	Any	Any
	Timer TOU1_7 counter underflow	H'5F	Any	Any
	Timer TOU2_0 counter underflow	H'63	Any	Any
	Timer TOU2_7 counter underflow	H'67	Any	Any
	Timer TOU3_0 counter underflow	H'6B	Any	Any
	Timer TOU3_7 counter underflow	H'6F	Any	Any
	Timer TOU4_0 counter underflow	H'73	Any	Any
	Timer TOU4_7 counter underflow	H'77	Any	Any

On-Chip Peripheral Module	DMA Transfer Request Sources	Setting Value for CnMRID Bit	Transfer Source	Transfer Destination
ADC	AD0 scan transfer end	H'7B	Any	Any
	AD1 scan transfer end	H'7F	Any	Any
	AD0IN0 interrupt transfer end	H'83	Any	Any
	(Reserved)	H'87	—	—
	AD0IN2 interrupt transfer end	H'8B	Any	Any
	(Reserved)	H'8F	—	—
	AD0IN15 interrupt transfer end	H'93	Any	Any
DRI	DRI0 DIN0 event detection	H'97	Any	Any
	DRI0 DIN1 event detection	H'9B	Any	Any
	(Reserved)	H'9F	—	—
	DRI0 DIN3 event detection	H'A3	Any	Any
	DRI0 DIN4 event detection	H'A7	Any	Any
	DRI0 DIN5 event detection	H'AB	Any	Any
	DRI0 DEC0 underflow	H'AF	Any	Any
	DRI0 DEC1 underflow	H'B3	Any	Any
	DRI0 DEC2 underflow	H'B7	Any	Any
	DRI0 DEC3 underflow	H'BB	Any	Any
	DRI0 DEC4 underflow	H'BF	Any	Any
	DRI0 DEC5 underflow	H'C3	Any	Any
	DRI0 DRI address counter 0 transfer end	H'C7	Any	Any
	DRI0 DRI address counter 1 transfer end	H'CB	Any	Any
	DRI0 DRI acquisition event counter underflow	H'CF	Any	Any
	DRI0 DRI transfer counter underflow	H'D3	Any	Any

Note: • When on-chip peripheral module request is selected, select cycle steal mode 1 (cycle steal mode 2).

20.4 Operation

When the DMA start conditions are satisfied during DMA operation, the DMA starts the transfer according to the predetermined channel priority; when the transfer end conditions are satisfied, it ends the transfer.

Completion of DMA transfer refers to when the DMiTCR register value reaches "0" and the final DMA transfer is completed. Completion of DMA forcible termination refers to when the DMA transfer end conditions are satisfied before the DMiTCR register value reaches "0".

20.4.1 DMA Transfer Request Sources

There are three DMA transfer request sources: auto request (software request), external request, and on-chip peripheral module request. The transfer request source is selected by the setting of the RS bit in the DMiCHCR register corresponding to the channel and the settings of registers DM01ARS to DM1011ARS.

(1) Auto-Request (Software Request)

When auto-request is selected as the DMA transfer request source, the DMAC automatically generates transfer request signals internally if the DMA transfer start conditions are satisfied. Transfer is performed continuously until the DMiTCR register becomes "0" regardless of the transfer mode (cycle steal mode 1, cycle steal mode 2, or burst mode).

Table 20.6 lists the DMA transfer matrix in auto request mode.

Table 20.6 DMA Transfer Matrix in Auto-Request Mode

	Transfer Destination				
Transfer Source	External Address Space	On-Chip Peripheral Module*1	IL Memory/ OL Memory	SHwyRAM	ROM
External address space	×	×	×	×	×
On-chip peripheral module*1	×	○	○	○	×
IL memory/OL memory	×	○	○	○	×
SHwyRAM	×	○	○	○	×
ROM	×	○	○	○	×

Note: *1 When the transfer source or destination is on-chip peripheral module register, the transfer size should be the same value of its access size.

Legend:

○: Transfer is available.

×: Transfer is not available.

(2) External Request

When external request is selected as the DMA transfer request source, DMA transfer is performed upon the request signal (DREQ0, DREQ2, and DREQ3) from an external device in this MCU while the DMA transfer start conditions are satisfied. External request is selected as the DMA transfer request source for DMA0, DMA2, and DMA3 only.

Choose to detect the DREQ signal by either the edge of the signal input with the DL bit in the DM0CHCR, DM2CHCR and DM3CHCR registers.

Table 20.7 lists the DMA transfer matrix in external request mode.

Table 20.7 DMA Transfer Matrix in External Request Mode*2

	Transfer Destination				
Transfer Source	**External Address Space**	**On-Chip Peripheral Module*1**	**IL Memory/ OL Memory**	**SHwyRAM**	**ROM**
External address space	×	×	×	×	×
On-chip peripheral module*1	×	○	○	○	×
IL memory/OL memory	×	○	○	○	×
SHwyRAM	×	○	○	○	×
ROM	×	○	○	○	×

Notes: *1 When the transfer source or destination is an on-chip peripheral module, a transfer size permitted by the source or destination register should be used.

*2 External requests are valid for channels 0, 2, and 3 only.

Legend:

○: Transfer is available.

×: Transfer is not available.

(3) On-Chip Peripheral Module Request

When on-chip peripheral request is selected as the DMA transfer request source, DMA transfer is performed upon the request signal from an on-chip peripheral module while the DMA transfer start conditions are satisfied. DMA transfer request signals can be issued by the SCIF, RSPI, IIC, ATU-IIIS, ADC, and DRI, based on the settings in registers DM01ARS to DM1011ARS.

Table 20.8 lists the DMA transfer matrix in on-chip peripheral module request mode.

Table 20.8 DMA Transfer Matrix in On-Chip Peripheral Module Request Mode*2

	Transfer Destination				
Transfer Source	**External Address Space**	**On-Chip Peripheral Module*1**	**IL Memory/ OL memory**	**SHwyRAM**	**ROM**
External address space	×	×	×	×	×
On-chip peripheral module*1	×	○	○	○	×
IL memory/OL memory	×	○	○	○	×
SHwyRAM	×	○	○	○	×
ROM	×	○	○	○	×

Notes: *1 When the transfer source or destination is on-chip peripheral module register, set the transfer size to the same value of its access size.

*2 Settings are available in cycle steal mode 1 and cycle steal mode 2.

Legend:

○: Transfer is available.

×: Transfer is not available.

20.4.2 DMA Transfer Modes

Cycle steal mode 1, cycle steal mode 2, and burst mode are available as DMA transfer modes. Select the mode in the TB and LCKN bits in DMiCHCR register.

One transfer unit described in this section refers to the total of 1 to 32-byte read and write cycles set by the TS0 to TS2 bits in the DMiCHCR register. Also, other bus masters refers to other DMAC modules, CPU, H-UDI, and AUDR.

(1) Cycle Steal Mode 1 (TB = "0", LCKN = "0")

In cycle steal mode 1, the SuperHyway bus mastership is given to another bus master after a DMA transfer of one transfer unit. When the next transfer request occurs, the DMAC issues the next transfer request, the bus mastership is obtained from the other bus master, and a transfer is performed for one transfer unit. When that transfer ends, the bus mastership is passed to the other bus master. This is repeated until the transfer end conditions are satisfied.

Figure 20.2 shows an example of DMA transfer timing in cycle-steal mode 1.

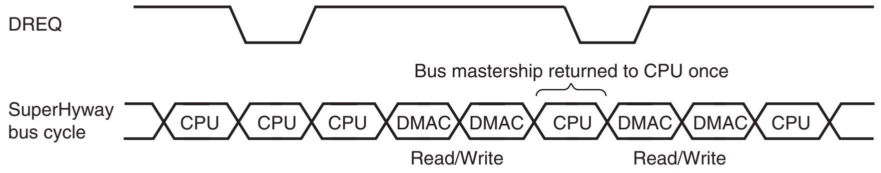


Figure 20.2 DMA Transfer Timing Example in Cycle Steal Mode 1 (DREQ Signal Falling Edge Detection Enabled)

(2) Cycle Steal Mode 2 (TB = "0", LCKN = "1")

In cycle steal mode 2, once the SuperHyway bus mastership is given to another bus master after a read cycle of one transfer unit, the DMAC issues a transfer request for a write cycle. When the next DMA transfer request occurs, the DMAC issues the next transfer request for a read cycle and the cycle is performed. This is repeated until the transfer end conditions are satisfied.

Figure 20.3 shows an example of DMA transfer timing in cycle steal mode 2.

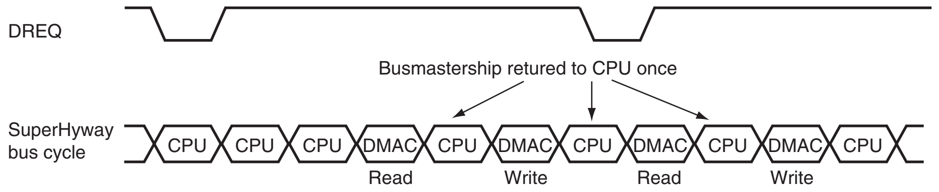


Figure 20.3 DMA Transfer Timing Example in Cycle-Steal Mode 2 (DREQ Signal Falling Edge Detection Enabled)

(3) Burst Mode (TB = "1", LCKN = "0")

In burst mode, once the DMAC obtains the SuperHyway bus mastership, the transfer is performed continuously without releasing the bus mastership until the transfer end conditions are satisfied.

Do not use burst mode when the on-chip peripheral module is selected as the transfer request source.

Figure 20.4 shows DMA transfer timing in burst mode.

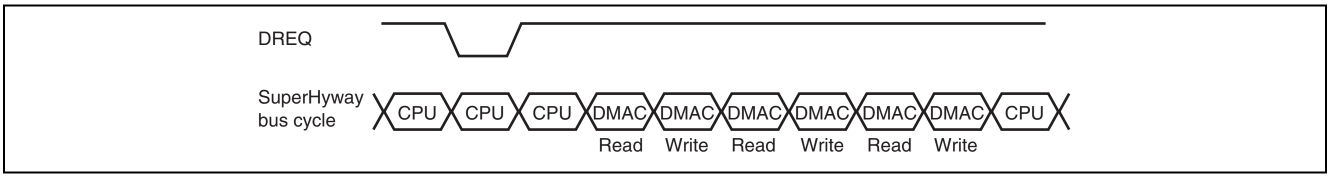


Figure 20.4 DMA Transfer Timing Example in Burst Mode (DREQ Signal Falling Edge Detection Enabled)

20.4.3 DMA Transfer Start Conditions and DMA Transfer End Conditions

The following shows the DMA transfer start and end conditions:

(1) DMA Transfer Start Conditions

(a) When the repeat function is disabled, or when the repeat function is enabled and the DMiCHCR.HIE bit = "0"

- The DM05OR (DM611OR).NMIF bit ="0", AE bit ="0", DME bit ="1",
  DMiCHCR.TE bit ="0", and DE bit ="1"

(b) When the repeat function is enabled and the DMiCHCR.HIE bit = "1"

- The DM05OR (DM611OR).NMIF bit = "0", AE bit = "0", DME bit ="1",
  DMiCHCR.TE bit = "0", HE bit = "0", and DE bit ="1"
- The DM05OR (DM611OR).NMIF bit = "0", AE bit = "0", DME bit ="1",
  DMiCHCR.TE bit = "0", HE bit ="1", and DE bit ="1"
- The DM05OR (DM611OR).NMIF bit = "0", AE bit = "0", DME bit ="1",
  DMiCHCR.TE bit = "1", HE bit = "0", and DE bit ="1"

(2) DMA Transfer End Conditions

(a) When the repeat function is disabled, or when the repeat function is enabled and the DMiCHCR.HIE bit = "0"

- NMI is input
- An address error occurred
- The DMiCHCR.DE bit is set to "0"
- The DM05OR (DM611OR).DME bit is set to "0"
- The DMiCHCR.TE bit = "1"

(b) When the repeat function is enabled and the DMiCHCR.HIE bit = "1"

- NMI is input
- An address error occurred
- The DMiCHCR.DE bit is set to "0"
- The DM05OR (DM611OR).DME bit is set to "0"
- The DMiCHCR.HE bit = "1" and TE bit = "1"

20.4.4 Channel Priority in Modules

When the DMAC receives simultaneous transfer requests on two or more channels, it transfers data according to a predetermined priority. Two channel priority modes are available: fixed and round-robin. The priority is selected by setting the PR bit in the DM05OR register (DMA0 to 5) or DM611OR register (DMA6 to 11).

Round robin is used as the priority relationship for DMAC0 module (DMA0 to 5) and DMAC1 module (DMA6 to 11).

For details, see section 20.4.6, Priority between DMAC Modules.

(1) Fixed Priority

When the priority levels among the channels are selected as fixed, the priority does not change. There are two kinds of fixed priority as follows:

(a) DMA0 to 5

- DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5
- DMA0 > DMA2 > DMA3 > DMA1 > DMA4 >DMA5

(b) DMA6 to 11

- DMA6 > DMA7 > DMA8 > DMA9 > DMA10 > DMA11
- DMA6 > DMA8 > DMA9 > DMA7 > DMA10 > DMA11

(2) Round Robin

When the priority levels among the channels are selected as round robin, each time data of one transfer unit is transferred on one channel, the priority is rotated. The channel on which the transfer just finished rotates to the bottom of the priority. An example of changes in channel priority is shown in figure 20.5.

The priority of round robin immediately after reset is shown as follows:

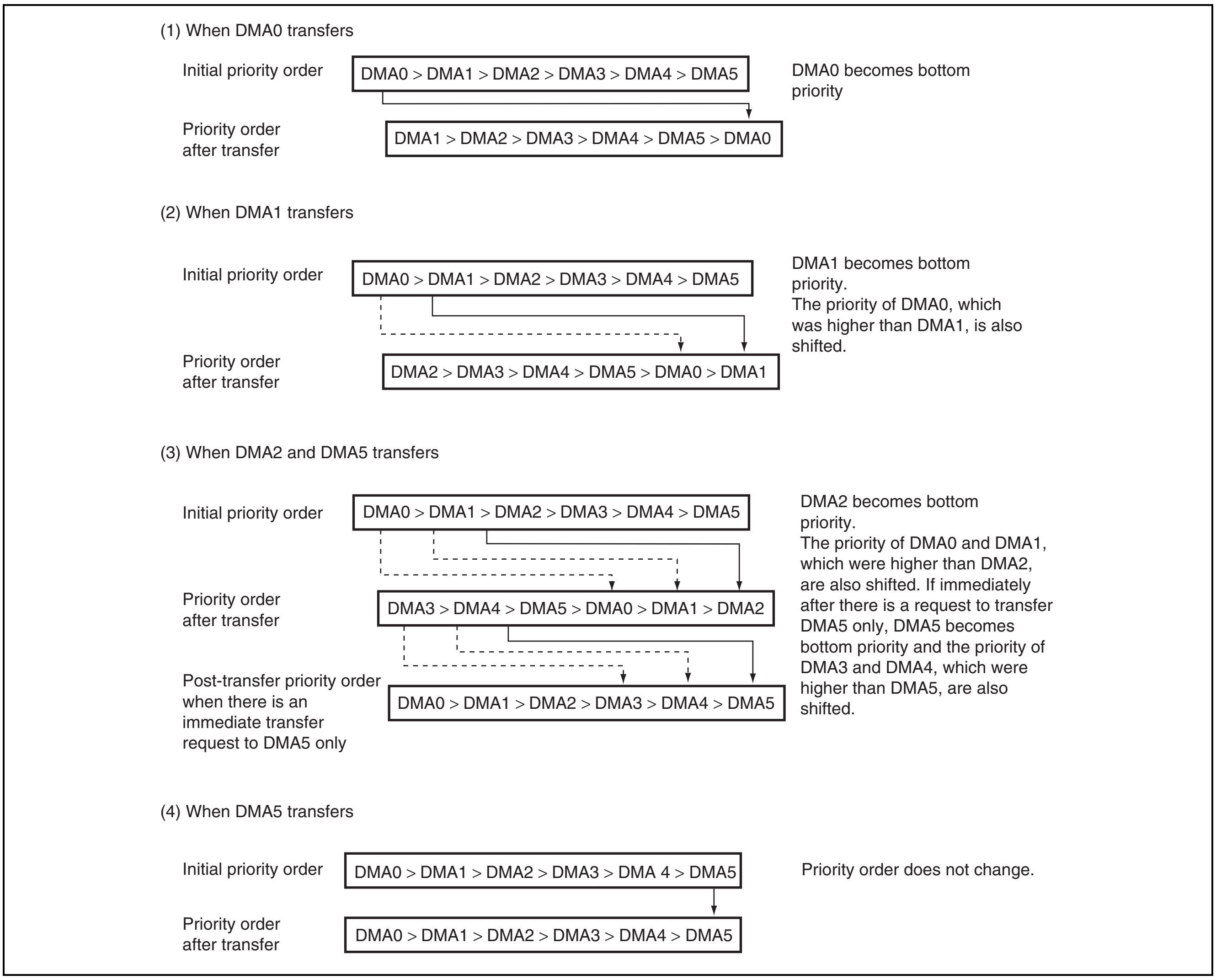
(a) DMA0 to 5

- DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5

(b) DMA6 to 11

- DMA6 > DMA7 > DMA8 > DMA9 > DMA10 > DMA11

When the priority levels among the channels are selected as round robin, do not mix cycle steal mode 1 (cycle steal mode 2) and burst mode among DMA0 to DMA5. Also, do not mix cycle steal mode 1 (cycle steal mode 2) and burst mode among DMA6 to DMA11.


(1) When DMA0 transfers
Initial priority order
DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5
Priority order after transfer
DMA1 > DMA2 > DMA3 > DMA4 > DMA5 > DMA0
(2) When DMA1 transfers
Initial priority order
DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5
Priority order after transfer
DMA2 > DMA3 > DMA4 > DMA5 > DMA0 > DMA1
(3) When DMA2 and DMA5 transfers
Initial priority order
DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5
Priority order after transfer
DMA3 > DMA4 > DMA5 > DMA0 > DMA1 > DMA2
Post-transfer priority order when there is an immediate transfer request to DMA5 only
DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5
(4) When DMA5 transfers
Initial priority order
DMA0 > DMA1 > DMA2 > DMA3 > DMA 4 > DMA5
Priority order after transfer
DMA0 > DMA1 > DMA2 > DMA3 > DMA4 > DMA5

DMA0 becomes bottom priority

DMA1 becomes bottom priority. The priority of DMA0, which was higher than DMA1, is also shifted.

DMA2 becomes bottom priority. The priority of DMA0 and DMA1, which were higher than DMA2, are also shifted. If immediately after there is a request to transfer DMA5 only, DMA5 becomes bottom priority and the priority of DMA3 and DMA4, which were higher than DMA5, are also shifted.

Priority order does not change.

Figure 20.5 Example 1 of Changes in Channel Priority (Round Robin)

Figure 20.6 shows how the priority changes when DMA0 and DMA3 transfers are requested simultaneously and a DMA1 transfer is requested during the DMA0 transfer. The DMAC operates as follows:

1. Transfer requests are generated simultaneously to DMA0 and DMA3.
2. DMA0 has a higher priority, so the DMA0 transfer begins first (DMA3 waits for transfer).
3. A DMA1 transfer request occurs during the DMA0 transfer (DMA1 and DMA3 are both waiting).
4. When the DMA0 transfer ends, DMA0 becomes lowest priority.
5. At this point, DMA1 has a higher priority than DMA3, so the DMA1 transfer begins (DMA3 waits for transfer).
6. When the DMA1 transfer ends, DMA1 becomes lowest priority.
7. The DMA3 transfer begins.
8. When the DMA3 transfer ends, DMA3 and DMA2 shift downward in priority so that DMA3 becomes the lowest priority.

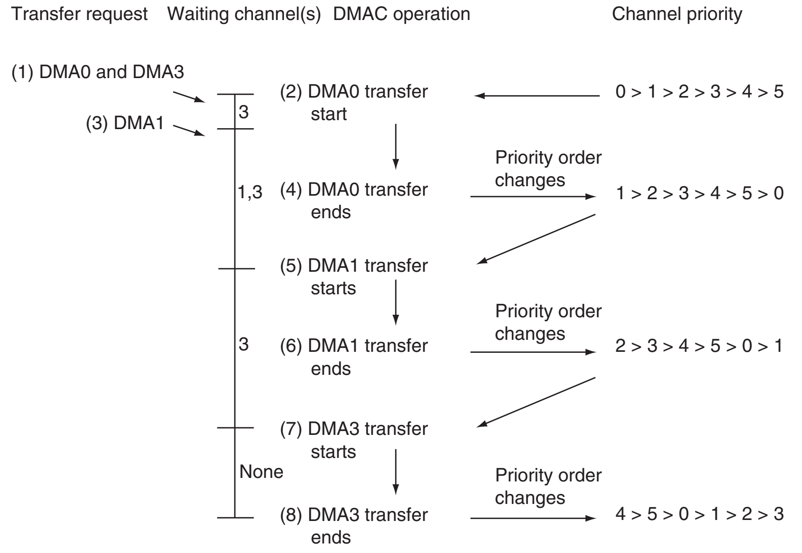


Figure 20.6 Example 2 of Changes in Channel Priority (Round Robin)

20.4.5 Operation Example of Multiple Channels in the Same DMAC Module

The channel priority in the same DMAC module is determined each one transfer unit.

If there is a channel which is transferring in burst mode in the same DMAC module, the DMAC module does not give the bus mastership to another bus master.

With the DMAC0 module as an example, the operation of multiple channels in the same DMAC module is shown as follows.

(1) Example of Mixing Cycle Steal Mode 1 and Burst Mode

The following shows an operation example when DMA0 is set to cycle steal mode 1 and DMA1 is set to burst mode.

When the priority is selected as fixed priority (DMA0 > DMA1....> DMA5), and DMA1 is transferring in burst mode, if there is a transfer request to DMA0 with a higher priority in cycle steal mode 1, DMA0 transfer will begin after the one transfer unit of DMA1.

In cycle steal mode 1, the bus mastership is usually given to another bus master each one transfer unit. However, if there is a channel which is transferring in burst mode in the DMAC0 module, first DMA0 with a higher priority in cycle steal mode 1 performs the transfer of one transfer unit, and DMA1 transfer is continuously performed without releasing the bus mastership.

The bus mastership will then switch between the two in the order DMA0, DMA1, DMA0, and DMA1. This bus status is referred to as burst mode prioritized execution.

This example is shown in figure 20.7. When multiple channels are operating in burst modes, the channel with the highest priority is executed first. When DMA transfer is executed in the multiple channels, the bus mastership will not be released to another bus master until all transfers in burst mode are complete.

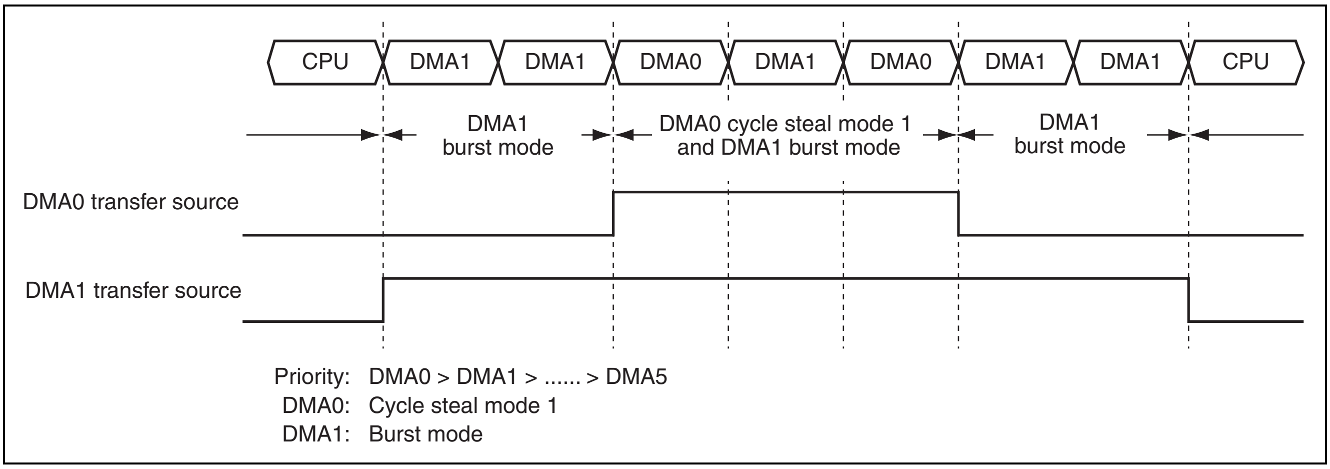


Figure 20.7 Example of Operations in Priority Order in Burst Mode

(2) Example of Mixing Burst Mode and Burst Mode

The following shows an operation example when DMA0 is set to burst mode and DMA1 is set to burst mode.

When the priority is selected as fixed priority (DMA0 > DMA1....> DMA5), and DMA1 is transferring in burst mode, if there is a transfer request to DMA0 with a higher priority, DMA0 transfer will begin after the one transfer unit of DMA1. After all DMA0 transfers are complete, DMA1 transfer in burst mode is executed without releasing the bus mastership.

When the priority is selected as round robin and DMA1 is transferring in burst mode, if there is a transfer request to DMA0 with a higher priority, DMA0 transfer will begin after the one transfer unit of DMA1. As the channel priority is rotated or determined each transfer unit, after the one transfer unit of DMA1 transfer, the one transfer unit of DMA0 transfer in burst mode is executed without releasing the bus mastership. The bus mastership will then switch between the two in the order DMA1, DMA0, DMA1, and DMA0.

20.4.6 Priority between DMAC Modules

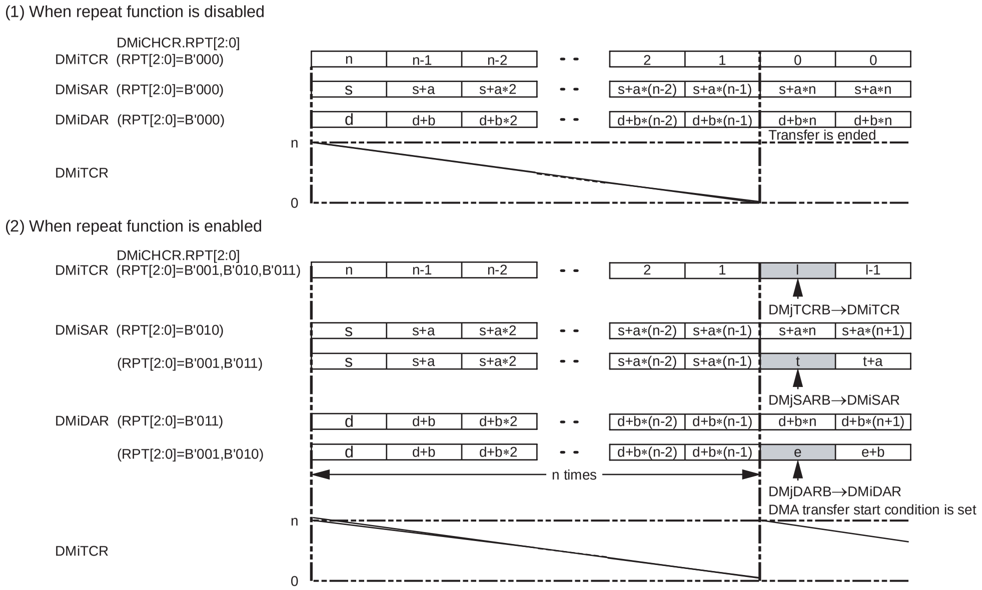
Round robin is used as the priority relationship for the DMAC0 module (DMA0 to 5) and DMAC1 module (DMA6 to 11) in the DMAC. The priority is DMAC0 > DMAC1 immediately after reset. Each time the DMAC0 (DAMC1) module releases the bus mastership to another bus master, the priority is rotated so that the DMAC0 (DAMC1) module which performed the DMA transfer rotates to the bottom of the priority.

20.4.7 Repeat Function

When the repeat function is enabled, the setting value is loaded from the DMjTCRB register to the DMjTCR register each time the DMjTCR register value reaches "0".

According to setting the RPT bits in the DMjCHCR register, the setting value can be loaded from the DMjSARB register to the DMjSAR register and the DMjDARB register to DMjDAR register at this time. To start DMA transfer after the setting value is loaded, the conditions described in section 20.4.3, DMA Transfer Start Conditions and DMA Transfer End Conditions should be satisfied by software.

Figure 20.8 shows an operation of data transfer processing when the repeat function is disabled and enabled.



[Legend]
n: Transfer count set in DMiTCR; any integral number (< 2 ^ 24)
l: Transfer count set in DMjTCRB; any integral number (< 2 ^ 24)
s: Address set in DMiSAR
t: Address set in DMjSARB
d: Address set in DMiDAR
e: Address set in DMjDARB
a: Difference when DMiSAR is updated (±1 for byte; ±2 for word; ±4 for longword, 8-byte × 2: ±16, 8-byte × 4: ±32)*1
b: Difference when DMiDAR is updated (±1 for byte; ±2 for word; ±4 for longword, 8-byte × 2: ±16, 8-byte × 4: ±32)*2

[Notes]
*1: When DMjCHCR.SM = "0" (the source address is fixed), a = 0.
*2: When DMjCHCR.DM = "0" (the destination address is fixed), b = 0.

Figure 20.8 Transfer when Repeat Function is Enabled and Data Transfer when Disabled

Using the repeat function with the half end function allows a double buffer transfer executed virtually. The setting example of the half end function is described below.

In the following example, handling 40-byte data every data reception is explained.

1. DMAC settings
   - Set address of the SCi receive FIFO data register (SCiFRDR) in the SCIF to the DMjSAR register
   - Set address of an internal memory data store area to the DMjDAR register
   - Set the DMjTCR register to 80 (H'50)
   - Satisfy the following settings of the DMjCHCR register

     RPT bits = B'010: Repeat function enabled (DMjDAR, DMjTCR)

     HIE bit = B'1: DMjTCR register/2 interrupt generated

     DM bits = B'01: DMjDAR register incremented

     SM bits = B'00: DMjSAR register fixed

     IE bit = B'1: Interrupt enabled

     DE bit = B'1: DMA transfer enabled
   - Set such as TB and TS bits according to use conditions
   - Set the PR bit in the DM05OR or DM611OR register according to the usage conditions, and set the DME bit to "1".
2. DMA transfer is executed.
3. The DMjTCR register is decreased to half of its initial value and an interrupt is generated

   In the interrupt handler, after reading out the DMjCHCR register and verifying that HE (bit 19) is set to "1", clear that bit to "0" and process 40 bytes of data starting at the address set in the DMjDAR register.
4. The DMjTCR register is cleared to "0" and an interrupt is generated

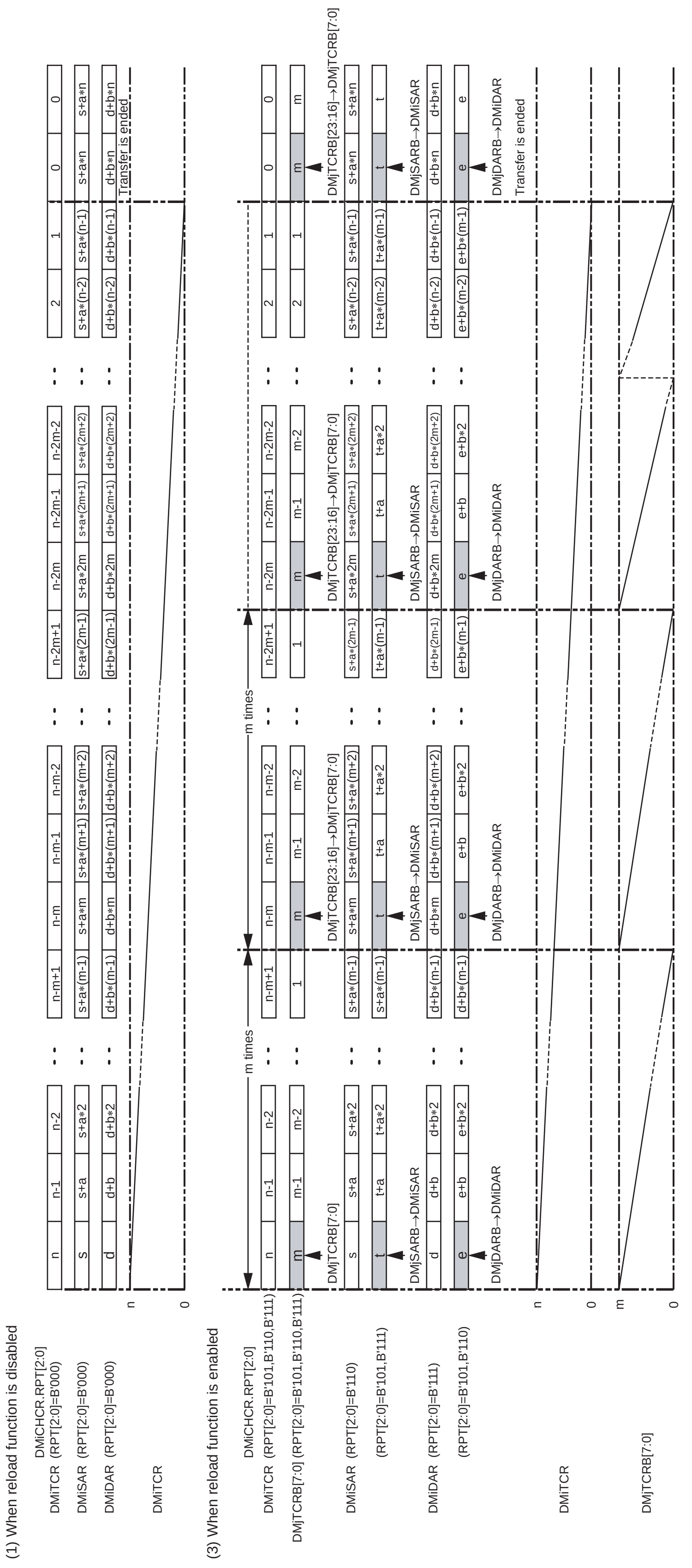
   In the interrupt handler, after reading out the DMjCHCR register and verifying that TE (bit 1) is set to "1", clear that bit to "0" and process 40 bytes of data starting at the address which is 40 added to the address set in the DMjDAR register. At this time, the values of the DMjDARB and DMjTCRB registers are copied to the DMjDAR and DMjTCR registers, respectively, in the DMAC.
5. After that, the operations of items 3 and 4 are repeated until either the condition of both HE and TE being "1" and either DME or DE being "0" is established or an NMI interrupt occurs.

That is, using this function makes it possible to perform sequential processing while interchanging the data reception storage buffer and the data processing buffer.

20.4.8 Reload Function

When the reload function is executed, the DMjTCRB register is used as a reload counter. When the DMAC reload function is enabled, according to the setting of the RPT bits in the DMjCHCR register, the setting value is loaded from the DMjSARB register to the DMjSAR register, the DMjDARB register to the DMjDAR register, and bits 23 to 16 in the DMjTCRB register to bits 7 to 0 in the DMjTCRB register when transfer starts and each the number of transfers set in bits 7 to 0 in the DMjTCRB register. Transfer is repeated without having to re-specify the transfer settings until the DMjTCR register value reaches "0". This function is effective when repeating data transfer with a specific area.

Figure 20.9 shows an operation of data transfer processing when the reload function is disabled and enabled.



[Legend]
n: Transfer count set in DMiTCR; any integral number (<2^24)
m: Transfer count for reload function set in DMjTCRB[23:16]; any integral number (<2^8)
s: Address set in DMiSAR
t: Address set in DMjSARB
d: Address set in DMiDAR
e: Address set in DMjDARB
a: Difference when DMiSAR is updated (±1 for byte; ±2 for word; ±4 for longword, 8-byte × 2: ±16, 8-byte × 4: ±32)*1
b: Difference when DMiDAR is updated (±1 for byte; ±2 for word; ±4 for longword, 8-byte × 2: ±16, 8-byte × 4: ±32)*2

[Notes]
*1: When DMjCHCR.SM = "0" (the source address is fixed), a = 0.
*2: When DMjCHCR.DM = "0" (the destination address is fixed), b = 0.

Figure 20.9 Data Transfer when Reload Function is Enabled and Data Transfer when Disabled

20.5 Usage Notes

20.5.1 Address Error

When a DMA address error occurs, perform the following procedure. Then reset all channels corresponding to registers DM05OR and DM611OR and restart the transfer.

1. Dummy read for the below listed registers. However, dummy read of the space not specified as the access destination by the DMA can be omitted.
   - On-chip peripheral module (Pck): Dummy read of the dummy access area (addresses H'FFFF 5020 to H'FFFF 5023)
   - On-chip peripheral module (PAck): Dummy read of the dummy access area (addresses H'FFA0 0000 to H'FFA0 0003)
   - IL memory/OL memory: Dummy read of any area
   - SHwyRAM: Dummy read of any area
   - ROM: Dummy read of any area
2. Issue the SYNCO instruction.
3. Reset all channels corresponding to the register, DM05OR or DM611OR, where the DMA address error occurred.
   - Reset DMA0 to 5 if the DM05OR.AE bit was set to "1".
   - Reset DMA6 to 11 if the DM611OR.AE bit was set to "1".

20.5.2 DMA Transfer to DMAC Prohibited

Do not perform DMA transfer with the DMAC register specified as the transfer source or transfer destination.

20.5.3 NMI Interrupt

When an NMI interrupt occurs, the DMA transfer is stopped. After returning from the NMI interrupt routine, set all channels again, and then restart the DMA transfer.

20.5.4 Accessing Registers during DMA Operation

During DMA operation (while the DMA start conditions are satisfied), the following constraints apply when accessing registers. Before forcibly termination, make sure the DMA transfer request from the corresponding on-chip module is cleared.

- DMAC0 module

  No write access is allowed for registers other than the DM0CHCR to DM5CHCR.DE bit, HE bit, TE bit, and DM05OR.DME bit.

  Write the same value to the other bits in the DM0CHCR to DM5CHCR registers and the DM05OR register.

- DMAC1 module

  No write access is allowed for registers other than the DM6CHCR to DM11CHCR.DE bit, HE bit, TE bit, and DM611OR.DME bit.

  Write the same value to the other bits in the DM6CHCR to DM11CHCR registers and the DM611OR register.

This page is blank for reasons of layout.

Section 21 Advanced Timer Unit IIIS (ATU-IIIS)

21.1 Overview

The ATU-IIIS comprises the timer blocks timer A (two subblocks, each with five channels), timer F (three channels), timer G (six channels), and timer TOU (five subblocks, each with eight channels), prescalers, and a common controller. The timer blocks have different functions and each can operate independently; timer blocks can also be linked via the clock bus. Each timer block consists of one or more timer subblocks and each subblock has one or more channels.

In the descriptions in this section, the following notation is used to refer to the names of the registers associated with each of the timers:

- Timer A Registers: i = 0 or 1, k = 0 to 5
- Timer F Registers: j = 0 to 2
- Timer G Registers: k = 0 to 5
- Timer TOU Registers: n = 0 to 4, m = 0 to 7

Table 21.1 lists the overview of the ATU-IIIS module.

Table 21.1 ATU-IIIS Overview

Item	Description
Function	• 64 interrupt sources can be generated. This enables to activate the DMAC, and interrupt processing by the CPU. • 6 pulse output dedicated for A/D (timer G) • On-chip 4-channel prescaler provided, which generates four types of clocks by dividing peripheral clock (Pck) by 1/1 to 1/1024 • Each channel for a timer can select a count source from among four divided clocks generated by prescaler and two external clocks.
Timer A	Timer A has a 32-bit free-run counter and 10 (for two subblocks, each with five channels) 32-bit input capture registers. • Detection by rising edges, falling edges, or both edges • DMAC activation at capture timing • Noise canceling function for each external pin with maximum length of 0.82 ms • Capture interrupt and counter overflow interrupt are available
Timer F	Timer F consists of three subblocks. Each subblock is provided with two 24-bit counters, a 16-bit counter, three 24-bit general registers, and a 16-bit general register. This provides the following operations: • Noise canceling function for each external pin with maximum length of 0.82 ms • Seven operation modes: edge counting in a specified period, valid edge interval counting, measurement of time during "H"/"L" input levels, measurement of PWM input waveform timing, rotation speed/pulse measurement, up/down event count, and four-time multiplication event count. • Activates DMAC by input capture interrupt • Overflow interrupt generation is available

Item	Description
Timer G	Timer G consists of six subblocks that have the same function. Each channel is provided with a 16-bit free-run counter and output compare register. This provides the following operations: • Outputs event that is triggered by compare match. This output can be used as a trigger for AD activation/interrupt. • Activates DMAC by compare match interrupt
Timer TOU	Timer TOU (Timer Output Unification) comprises a 24-bit output timer with a total of 40 channels (five subblocks, each with eight channels). The operation mode of each timer TOU channel can be selected by software from among the following: • PWM output mode • One-shot PWM output mode • One-shot output mode • Continuous output mode

Table 21.2 ATU-IIIS Interrupt Generation Functions

Signal Name	Interrupt Source	Source Input Count
IRQ_TA_OVF	Timer A overflow	2
IRQ_TA_IC	Timer A input capture	10
IRQ_TF_OVF	Timer F overflow	3
IRQ_TF_IC	Timer F input capture	3
IRQ_TG0	Timer G0 compare-match	1
IRQ_TG1	Timer G1 compare-match	1
IRQ_TG2	Timer G2 compare-match	1
IRQ_TG3	Timer G3 compare-match	1
IRQ_TG4	Timer G4 compare-match	1
IRQ_TG5	Timer G5 compare-match	1
IRQ_TOU00	TOU0_0 to TOU0_3 underflow	4
IRQ_TOU04	TOU0_4 to TOU0_7 underflow	4
IRQ_TOU10	TOU1_0 to TOU1_3 underflow	4
IRQ_TOU14	TOU1_4 to TOU1_7 underflow	4
IRQ_TOU20	TOU2_0 to TOU2_3 underflow	4
IRQ_TOU24	TOU2_4 to TOU2_7 underflow	4
IRQ_TOU30	TOU3_0 to TOU3_3 underflow	4
IRQ_TOU34	TOU3_4 to TOU3_7 underflow	4
IRQ_TOU40	TOU4_0 to TOU4_3 underflow	4
IRQ_TOU44	TOU4_4 to TOU4_7 underflow	4

21.2 Block Diagram

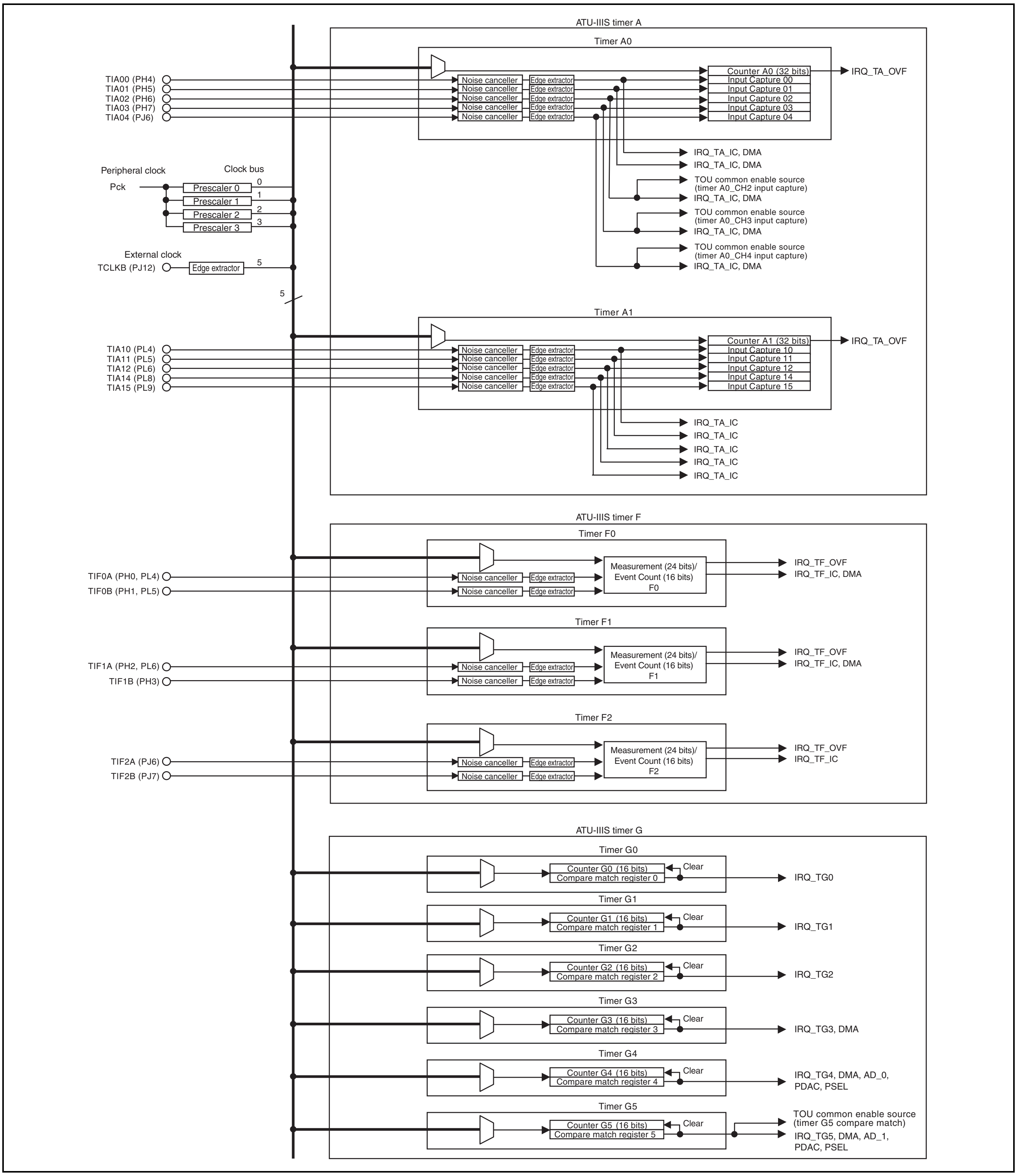


Figure 21.1 Block Diagram of ATU-IIIS (1)

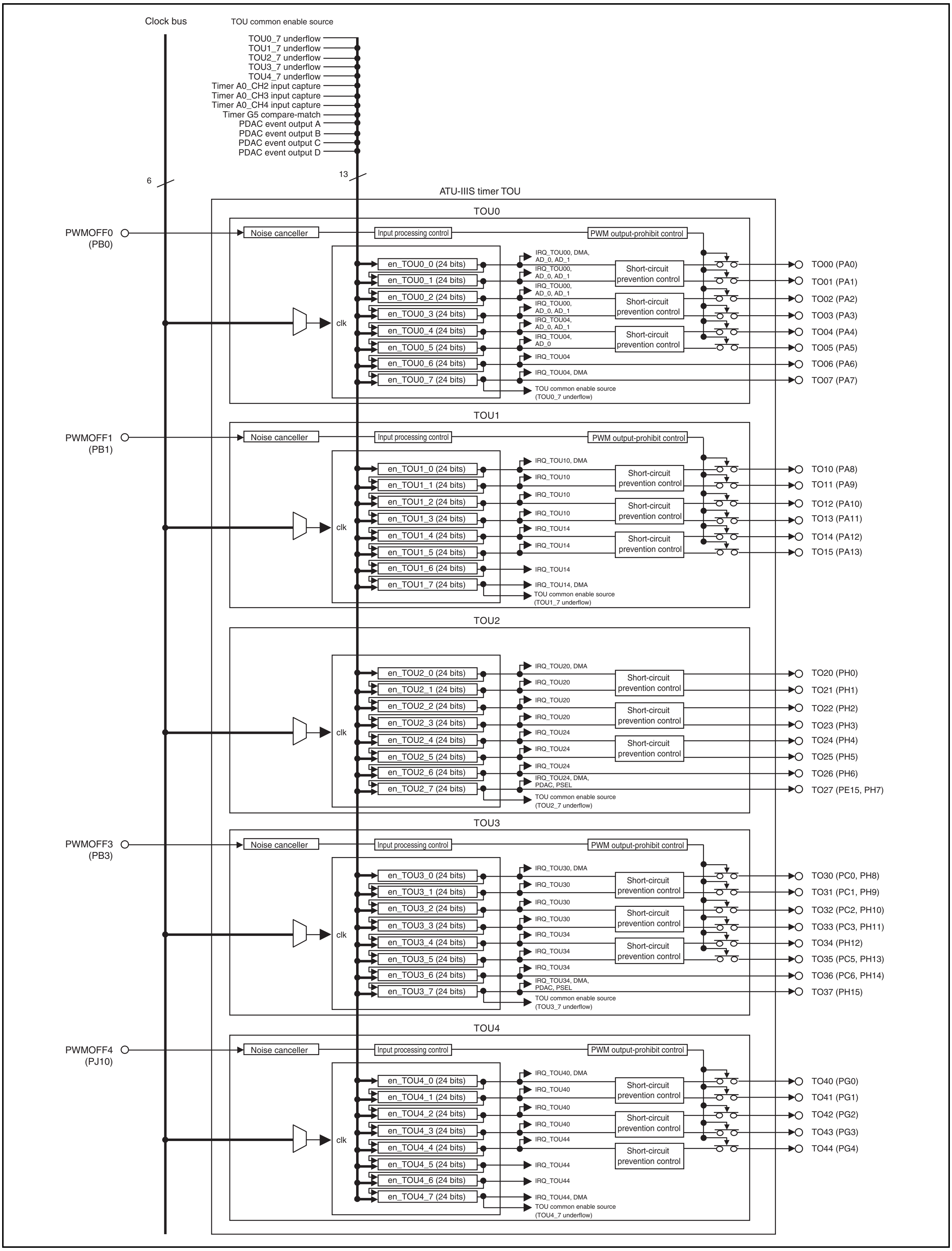
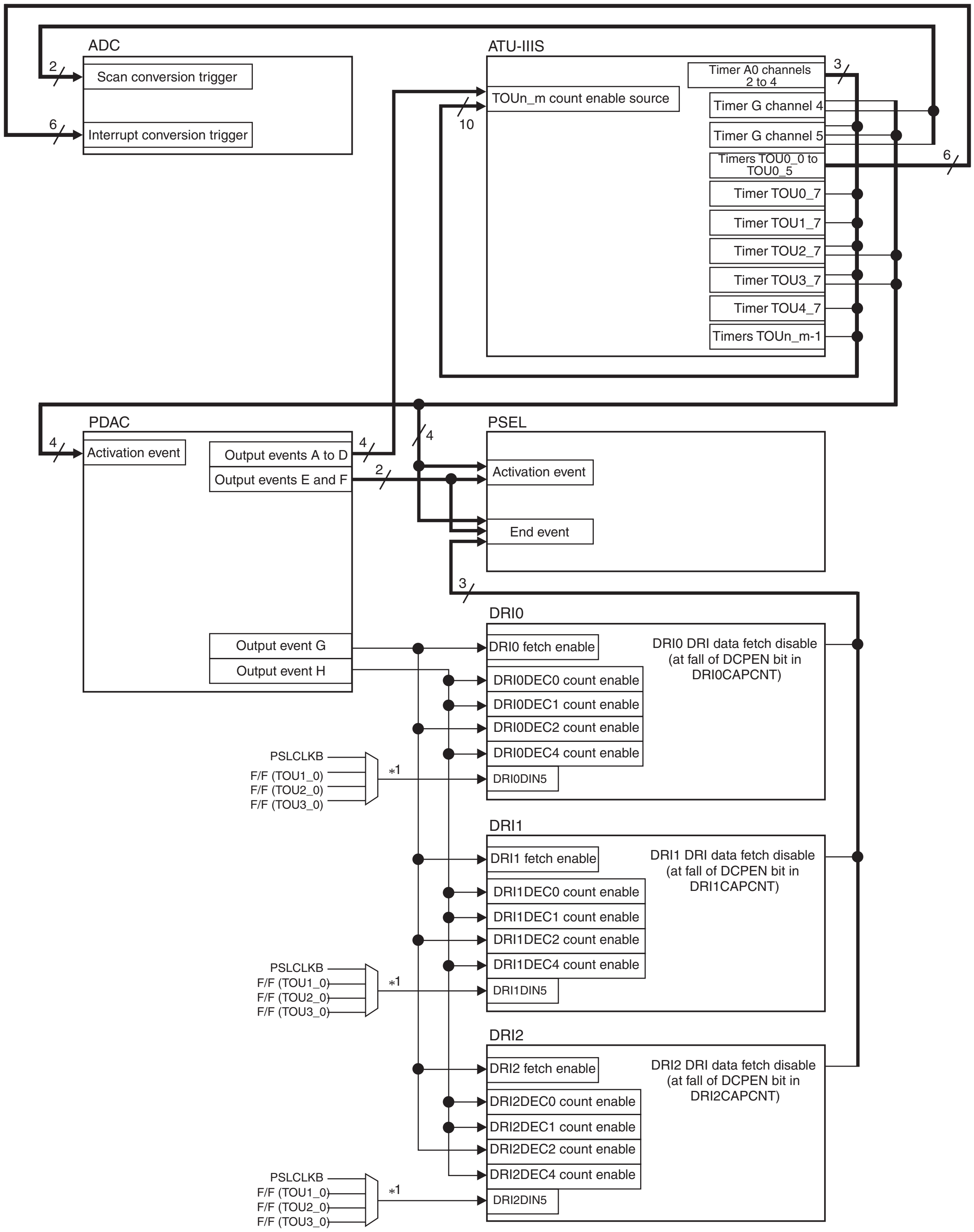


Figure 21.1 Block Diagram of ATU-IIIS (2)

Figure 21.2 is a wiring diagram of the event signals.



Note *1 Clock output from PSEL or F/F output from timer TOU of ATU-IIIS can be input as an internal signal to DIN5. For example, when F/F (TOU1_0) is selected in the DRIiDINSEL register, the value at output from timer TOU1_0 (the value of the FFDT10 bit in TO1FFDR) is input to DIN5 as an internal signal.

Legend: m = 0 to 7, n = 0 to 4

Figure 21.2 Event Signal Wiring Diagram

21.3 Input/Output Pins

Table 21.3 lists the ATU-IIIS pins.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 21.3 Pin Configuration

Item	Pin Name	I/O	Function
Common controller	TCLKB	Input	External clock input to clock bus 5
Timer A	TIA00 to TIA04 TIA10 to TIA12 TIA14, TIA15	Input	Input-capture triggers for timer A channels
Timer F	TIF0A, TIF0B TIF1A, TIF1B TIF2A, TIF2B	Input	Event inputs for timer F channels
Timer TOU	PWMOFF0, PWMOFF1 PWMOFF3, PWMOFF4	Input	Timer TOU PWM output-prohibit control signal inputs
	TO00 to TO07 TO10 to TO15 TO20 to TO27 TO30 to TO37 TO40 to TO44	Output	Pulse/PWM outputs for timer TOU channels

21.4 Register Descriptions

Addresses of the ATU-IIIS registers are shown below. To access the registers, the following procedure should be followed.

- When writing to reserved bits, the value written must be "0".
- Registers which have more than 16 bits must be read from and written to in 32 bit units. These registers cannot be accessed in 16- or 8-bit units.

Table 21.4 Register Configuration

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Common controller	ATU-IIIS Master Enable Register	ATUENR	H'0000	H'FFFF E000	16	21-18
	ATU-IIIS Clock Bus Control Register	ATCBCNT	H'00	H'FFFF E002	8	21-20
	ATU-IIIS Noise Cancellation Mode Register	ATNCMR	H'00	H'FFFF E003	8	21-21
	ATU-IIIS Interrupt Select Register A0	ATISRA0	H'00	H'FFFF E010	8	21-24
	ATU-IIIS Interrupt Select Register A1	ATISRA1	H'00	H'FFFF E011	8	21-24
	ATU-IIIS Interrupt Select Register F	ATISRF	H'00	H'FFFF E014	8	21-25
	ATU-IIIS Interrupt Select Register G	ATISRG	H'00	H'FFFF E018	8	21-26
	ATU-IIIS Interrupt Select Register TOU0	ATISRT0	H'00	H'FFFF E020	8	21-27
	ATU-IIIS Interrupt Select Register TOU1	ATISRT1	H'00	H'FFFF E021	8	21-27
	ATU-IIIS Interrupt Select Register TOU2	ATISRT2	H'00	H'FFFF E022	8	21-27
	ATU-IIIS Interrupt Select Register TOU3	ATISRT3	H'00	H'FFFF E023	8	21-27
	ATU-IIIS Interrupt Select Register TOU4	ATISRT4	H'00	H'FFFF E024	8	21-27
	ATU-IIIS Prescaler Register 0	ATPSCR0	H'0000	H'FFFF E100	16	21-29
	ATU-IIIS Prescaler Register 1	ATPSCR1	H'0000	H'FFFF E102	16	21-29
	ATU-IIIS Prescaler Register 2	ATPSCR2	H'0000	H'FFFF E104	16	21-29
	ATU-IIIS Prescaler Register 3	ATPSCR3	H'0000	H'FFFF E106	16	21-29
Timer A0	TA0 Control Register	TA0CR	H'00	H'FFFF E202	8	21-33
	TA0I/O Control Register 1	TA0IO1	H'0000	H'FFFF E204	16	21-34
	TA0I/O Control Register 2	TA0IO2	H'0000	H'FFFF E206	16	21-35
	TA0 Status Register	TA0SR	H'00	H'FFFF E208	8	21-37
	TA0 Interrupt Enable Register	TA0IER	H'00	H'FFFF E209	8	21-39
	TA00 Noise Canceler Counter	TA00NCNT	H'00	H'FFFF E210	8	21-43
	TA00 Noise Canceler Register	TA00NCR	H'00	H'FFFF E211	8	21-44
	TA01 Noise Canceler Counter	TA01NCNT	H'00	H'FFFF E212	8	21-43
	TA01 Noise Canceler Register	TA01NCR	H'00	H'FFFF E213	8	21-44
	TA02 Noise Canceler Counter	TA02NCNT	H'00	H'FFFF E214	8	21-43
	TA02 Noise Canceler Register	TA02NCR	H'00	H'FFFF E215	8	21-44
	TA03 Noise Canceler Counter	TA03NCNT	H'00	H'FFFF E216	8	21-43
	TA03 Noise Canceler Register	TA03NCR	H'00	H'FFFF E217	8	21-44
	TA04 Noise Canceler Counter	TA04NCNT	H'00	H'FFFF E218	8	21-43
	TA04 Noise Canceler Register	TA04NCR	H'00	H'FFFF E219	8	21-44
	TA0 Free-Running Counter	TA0TCNT	H'0000 0000	H'FFFF E220	32	21-41
	TA00 Input Capture Register	TA00ICR	H'0000 0000	H'FFFF E228	32	21-40

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer A0	TA01 Input Capture Register	TA01ICR	H'0000 0000	H'FFFF E22C	32	21-40
	TA02 Input Capture Register	TA02ICR	H'0000 0000	H'FFFF E230	32	21-40
	TA03 Input Capture Register	TA03ICR	H'0000 0000	H'FFFF E234	32	21-40
	TA04 Input Capture Register	TA04ICR	H'0000 0000	H'FFFF E238	32	21-40
Timer A1	TA1 Control Register	TA1CR	H'00	H'FFFF E302	8	21-33
	TA1I/O Control Register 1	TA1IO1	H'0000	H'FFFF E304	16	21-34
	TA1I/O Control Register 2	TA1IO2	H'0000	H'FFFF E306	16	21-35
	TA1 Status Register	TA1SR	H'00	H'FFFF E308	8	21-37
	TA1 Interrupt Enable Register	TA1IER	H'00	H'FFFF E309	8	21-39
	TA10 Noise Canceler Counter	TA10NCNT	H'00	H'FFFF E310	8	21-43
	TA10 Noise Canceler Register	TA10NCR	H'00	H'FFFF E311	8	21-44
	TA11 Noise Canceler Counter	TA11NCNT	H'00	H'FFFF E312	8	21-43
	TA11 Noise Canceler Register	TA11NCR	H'00	H'FFFF E313	8	21-44
	TA12 Noise Canceler Counter	TA12NCNT	H'00	H'FFFF E314	8	21-43
	TA12 Noise Canceler Register	TA12NCR	H'00	H'FFFF E315	8	21-44
	TA14 Noise Canceler Counter	TA14NCNT	H'00	H'FFFF E318	8	21-43
	TA14 Noise Canceler Register	TA14NCR	H'00	H'FFFF E319	8	21-44
	TA15 Noise Canceler Counter	TA15NCNT	H'00	H'FFFF E31A	8	21-43
	TA15 Noise Canceler Register	TA15NCR	H'00	H'FFFF E31B	8	21-44
	TA1 Free-Running Counter	TA1TCNT	H'0000 0000	H'FFFF E320	32	21-41
	TA10 Input Capture Register	TA10ICR	H'0000 0000	H'FFFF E328	32	21-40
	TA11 Input Capture Register	TA11ICR	H'0000 0000	H'FFFF E32C	32	21-40
	TA12 Input Capture Register	TA12ICR	H'0000 0000	H'FFFF E330	32	21-40
	TA14 Input Capture Register	TA14ICR	H'0000 0000	H'FFFF E338	32	21-40
	TA15 Input Capture Register	TA15ICR	H'0000 0000	H'FFFF E33C	32	21-40
Timer F common controller	TF Start Register	TFSTR	H'0000 0000	H'FFFF E400	32	21-52
	TF Noise Canceller Control Register	TFNCCR	H'0000 0000	H'FFFF E404	32	21-53
Timer F0	TF0 Noise Canceler Counter A	TF0NCNTA	H'00	H'FFFF E410	8	21-68
	TF0 Noise Cancel Register A	TF0NCRA	H'00	H'FFFF E411	8	21-71
	TF0 Noise Canceler Counter B	TF0NCNTB	H'00	H'FFFF E450	8	21-70
	TF0 Noise Cancel Register B	TF0NCRB	H'00	H'FFFF E451	8	21-72
	TF0 Control Register	TF0CR	H'00	H'FFFF E480	8	21-55
	TF0 Interrupt Enable Register	TF0IER	H'00	H'FFFF E481	8	21-57
	TF0 Status Register	TF0SR	H'00	H'FFFF E483	8	21-58
	TF0 Timer Counter A	TF0ECNTA	H'0000 0000	H'FFFF E484	32	21-60
	TF0 Event Counter	TF0ECNTB	H'0000	H'FFFF E488	16	21-61
	TF0 General Register B	TF0GRB	H'FFFF	H'FFFF E48A	16	21-64
	TF0 Time Counter C	TF0ECNTC	H'0000 0000	H'FFFF E48C	32	21-62
	TF0 General Register A	TF0GRA	H'FFFF FF00	H'FFFF E490	32	21-63
	TF0 Capture Output Register	TF0CDR	H'FFFF FF00	H'FFFF E494	32	21-67
	TF0 General Register C	TF0GRC	H'FFFF FF00	H'FFFF E498	32	21-65

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer F0	TF0 General Register D	TF0GRD	H'FFFF FF00	H'FFFF E49C	32	21-66
Timer F1	TF1 Noise Canceler Counter A	TF1NCNTA	H'00	H'FFFF E412	8	21-68
	TF1 Noise Cancel Register A	TF1NCRA	H'00	H'FFFF E413	8	21-71
	TF1 Noise Canceler Counter B	TF1NCNTB	H'00	H'FFFF E452	8	21-70
	TF1 Noise Cancel Register B	TF1NCRB	H'00	H'FFFF E453	8	21-72
	TF1 Control Register	TF1CR	H'00	H'FFFF E4A0	8	21-55
	TF1 Interrupt Enable Register	TF1IER	H'00	H'FFFF E4A1	8	21-57
	TF1 Status Register	TF1SR	H'00	H'FFFF E4A3	8	21-58
	TF1 Timer Counter A	TF1ECNTA	H'0000 0000	H'FFFF E4A4	32	21-60
	TF1 Event Counter	TF1ECNTB	H'0000	H'FFFF E4A8	16	21-61
	TF1 General Register B	TF1GRB	H'FFFF	H'FFFF E4AA	16	21-64
	TF1 Time Counter C	TF1ECNTC	H'0000 0000	H'FFFF E4AC	32	21-62
	TF1 General Register A	TF1GRA	H'FFFF FF00	H'FFFF E4B0	32	21-63
	TF1 Capture Output Register	TF1CDR	H'FFFF FF00	H'FFFF E4B4	32	21-67
	TF1 General Register C	TF1GRC	H'FFFF FF00	H'FFFF E4B8	32	21-65
	TF1 General Register D	TF1GRD	H'FFFF FF00	H'FFFF E4BC	32	21-66
Timer F2	TF2 Noise Canceler Counter A	TF2NCNTA	H'00	H'FFFF E414	8	21-68
	TF2 Noise Cancel Register A	TF2NCRA	H'00	H'FFFF E415	8	21-71
	TF2 Noise Canceler Counter B	TF2NCNTB	H'00	H'FFFF E454	8	21-70
	TF2 Noise Cancel Register B	TF2NCRB	H'00	H'FFFF E455	8	21-72
	TF2 Control Register	TF2CR	H'00	H'FFFF E4C0	8	21-55
	TF2 Interrupt Enable Register	TF2IER	H'00	H'FFFF E4C1	8	21-57
	TF2 Status Register	TF2SR	H'00	H'FFFF E4C3	8	21-58
	TF2 Timer Counter A	TF2ECNTA	H'0000 0000	H'FFFF E4C4	32	21-60
	TF2 Event Counter	TF2ECNTB	H'0000	H'FFFF E4C8	16	21-61
	TF2 General Register B	TF2GRB	H'FFFF	H'FFFF E4CA	16	21-64
	TF2 Time Counter C	TF2ECNTC	H'0000 0000	H'FFFF E4CC	32	21-62
	TF2 General Register A	TF2GRA	H'FFFF FF00	H'FFFF E4D0	32	21-63
	TF2 Capture Output Register	TF2CDR	H'FFFF FF00	H'FFFF E4D4	32	21-67
	TF2 General Register C	TF2GRC	H'FFFF FF00	H'FFFF E4D8	32	21-65
	TF2 General Register D	TF2GRD	H'FFFF FF00	H'FFFF E4DC	32	21-66
Timer G common controller	TG Start Register	TGSTR	H'00	H'FFFF E501	8	21-84
Timer G0	TG0 Control Register	TG0CR	H'00	H'FFFF E580	8	21-85
	TG0 Status Register	TG0SR	H'00	H'FFFF E581	8	21-86
	TG0 Counter	TG0CNT	H'0000	H'FFFF E584	16	21-88
	TG0 Compare Match Register	TG0OCR	H'FFFF	H'FFFF E586	16	21-89
Timer G1	TG1 Control Register	TG1CR	H'00	H'FFFF E590	8	21-85
	TG1 Status Register	TG1SR	H'00	H'FFFF E591	8	21-86
	TG1 Counter	TG1CNT	H'0000	H'FFFF E594	16	21-88
	TG1 Compare Match Register	TG1OCR	H'FFFF	H'FFFF E596	16	21-89

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer G2	TG2 Control Register	TG2CR	H'00	H'FFFF E5A0	8	21-85
	TG2 Status Register	TG2SR	H'00	H'FFFF E5A1	8	21-86
	TG2 Counter	TG2CNT	H'0000	H'FFFF E5A4	16	21-88
	TG2 Compare Match Register	TG2OCR	H'FFFF	H'FFFF E5A6	16	21-89
Timer G3	TG3 Control Register	TG3CR	H'00	H'FFFF E5B0	8	21-85
	TG3 Status Register	TG3SR	H'00	H'FFFF E5B1	8	21-86
	TG3 Counter	TG3CNT	H'0000	H'FFFF E5B4	16	21-88
	TG3 Compare Match Register	TG3OCR	H'FFFF	H'FFFF E5B6	16	21-89
Timer G4	TG4 Control Register	TG4CR	H'00	H'FFFF E5C0	8	21-85
	TG4 Status Register	TG4SR	H'00	H'FFFF E5C1	8	21-86
	TG4 Counter	TG4CNT	H'0000	H'FFFF E5C4	16	21-88
	TG4 Compare Match Register	TG4OCR	H'FFFF	H'FFFF E5C6	16	21-89
Timer G5	TG5 Control Register	TG5CR	H'00	H'FFFF E5D0	8	21-85
	TG5 Status Register	TG5SR	H'00	H'FFFF E5D1	8	21-86
	TG5 Counter	TG5CNT	H'0000	H'FFFF E5D4	16	21-88
	TG5 Compare Match Register	TG5OCR	H'FFFF	H'FFFF E5D6	16	21-89
Timer TOU0	TOU0 Control Register	TO0CR	H'00	H'FFFF E600	8	21-95
	TOU0 Timer Interrupt Enable Register	TO0IER	H'00	H'FFFF E601	8	21-96
	TOU0 Output Control Register	TO0OUCR	H'00	H'FFFF E602	8	21-97
	TOU0 Status Register	TO0SR	H'00	H'FFFF E603	8	21-98
	TOU0 Counter Enable Protect Register	TO0CEPR	H'00	H'FFFF E604	8	21-99
	Flip-Flop Output Protect Register for TOU0 Short-Circuit Prevention Function	TO0SHFFPR	H'00	H'FFFF E605	8	21-102
	TOU0 Flip-Flop Output Protect Register	TO0FFPR	H'00	H'FFFF E606	8	21-104
	TOU0 Counter Enable Register	TO0CENR	H'00	H'FFFF E608	8	21-100
	Flip-Flop Output Data Register for TOU0 Short-Circuit Prevention Function	TO0SHFFDR	H'00	H'FFFF E609	8	21-103
	TOU0 Flip-Flop Output Data Register	TO0FFDR	H'00	H'FFFF E60A	8	21-105
	TOU0 Noise Canceler Control Register	TO0NCCR	H'00	H'FFFF E60C	8	21-106
	TOU0 Noise Canceler Counter	TO0NCNT	H'00	H'FFFF E60E	8	21-108
	TOU0 Noise Canceler Register	TO0NCR	H'00	H'FFFF E60F	8	21-109
	TOU0PWMOFF Input Processing Register	TO0POCR	H'0000	H'FFFF E610	16	21-110
	TOU0PWMOFF Function Enable Register	TO0POER	H'00	H'FFFF E613	8	21-111
	TOU0PWM Output-Prohibit Control Register	TO0PODISCR	H'0000	H'FFFF E614	16	21-112
	TOU0PWM Output-Prohibit Level Control Register	TO0POLVCR	H'00	H'FFFF E617	8	21-113
	TOU00 Mode Control Register	TO00MCR	H'00	H'FFFF E620	8	21-115
	TOU00 Counter	TO00CNT	H'0000 0000	H'FFFF E624	32	21-118, 21-120
	TOU00 Reload Register	TO00RLD	H'0000 0000	H'FFFF E628	32	21-122
	TOU01 Mode Control Register	TO01MCR	H'00	H'FFFF E630	8	21-115
	TOU01 Counter	TO01CNT	H'0000 0000	H'FFFF E634	32	21-118, 21-120

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer TOU0	TOU01 Reload Register	TO01RLD	H'0000 0000	H'FFFF E638	32	21-122
	TOU02 Mode Control Register	TO02MCR	H'00	H'FFFF E640	8	21-115
	TOU02 Counter	TO02CNT	H'0000 0000	H'FFFF E644	32	21-118, 21-120
	TOU02 Reload Register	TO02RLD	H'0000 0000	H'FFFF E648	32	21-122
	TOU03 Mode Control Register	TO03MCR	H'00	H'FFFF E650	8	21-115
	TOU03 Counter	TO03CNT	H'0000 0000	H'FFFF E654	32	21-118, 21-120
	TOU03 Reload Register	TO03RLD	H'0000 0000	H'FFFF E658	32	21-122
	TOU04 Mode Control Register	TO04MCR	H'00	H'FFFF E660	8	21-115
	TOU04 Counter	TO04CNT	H'0000 0000	H'FFFF E664	32	21-118, 21-120
	TOU04 Reload Register	TO04RLD	H'0000 0000	H'FFFF E668	32	21-122
	TOU05 Mode Control Register	TO05MCR	H'00	H'FFFF E670	8	21-115
	TOU05 Counter	TO05CNT	H'0000 0000	H'FFFF E674	32	21-118, 21-120
	TOU05 Reload Register	TO05RLD	H'0000 0000	H'FFFF E678	32	21-122
	TOU06 Mode Control Register	TO06MCR	H'00	H'FFFF E680	8	21-115
	TOU06 Counter	TO06CNT	H'0000 0000	H'FFFF E684	32	21-118, 21-120
	TOU06 Reload Register	TO06RLD	H'0000 0000	H'FFFF E688	32	21-122
	TOU07 Mode Control Register	TO07MCR	H'00	H'FFFF E690	8	21-115
	TOU07 Counter	TO07CNT	H'0000 0000	H'FFFF E694	32	21-118, 21-120
	TOU07 Reload Register	TO07RLD	H'0000 0000	H'FFFF E698	32	21-122
Timer TOU1	TOU1 Control Register	TO1CR	H'00	H'FFFF E700	8	21-95
	TOU1 Timer Interrupt Enable Register	TO1IER	H'00	H'FFFF E701	8	21-96
	TOU1 Output Control Register	TO1OUCR	H'00	H'FFFF E702	8	21-97
	TOU1 Status Register	TO1SR	H'00	H'FFFF E703	8	21-98
	TOU1 Counter Enable Protect Register	TO1CEPR	H'00	H'FFFF E704	8	21-99
	Flip-Flop Output Protect Register for TOU1 Short-Circuit Prevention Function	TO1SHFFPR	H'00	H'FFFF E705	8	21-102
	TOU1 Flip-Flop Output Protect Register	TO1FFPR	H'00	H'FFFF E706	8	21-104
	TOU1 Counter Enable Register	TO1CENR	H'00	H'FFFF E708	8	21-100
	Flip-Flop Output Data Register for TOU1 Short-Circuit Prevention Function	TO1SHFFDR	H'00	H'FFFF E709	8	21-103
	TOU1 Flip-Flop Output Data Register	TO1FFDR	H'00	H'FFFF E70A	8	21-105
	TOU1 Noise Canceler Control Register	TO1NCCR	H'00	H'FFFF E70C	8	21-106
	TOU1 Noise Canceler Counter	TO1NCNT	H'00	H'FFFF E70E	8	21-108
	TOU1 Noise Canceler Register	TO1NCR	H'00	H'FFFF E70F	8	21-109
	TOU1PWMOFF Input Processing Register	TO1POCR	H'0000	H'FFFF E710	16	21-110
	TOU1PWMOFF Function Enable Register	TO1POER	H'00	H'FFFF E713	8	21-111
	TOU1PWM Output-Prohibit Control Register	TO1PODISCR	H'0000	H'FFFF E714	16	21-112

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer TOU1	TOU1PWM Output-Prohibit Level Control Register	TO1POLVCR	H'00	H'FFFF E717	8	21-113
	TOU10 Mode Control Register	TO10MCR	H'00	H'FFFF E720	8	21-115
	TOU10 Counter	TO10CNT	H'0000 0000	H'FFFF E724	32	21-118, 21-120
	TOU10 Reload Register	TO10RLD	H'0000 0000	H'FFFF E728	32	21-122
	TOU11 Mode Control Register	TO11MCR	H'00	H'FFFF E730	8	21-115
	TOU11 Counter	TO11CNT	H'0000 0000	H'FFFF E734	32	21-118, 21-120
	TOU11 Reload Register	TO11RLD	H'0000 0000	H'FFFF E738	32	21-122
	TOU12 Mode Control Register	TO12MCR	H'00	H'FFFF E740	8	21-115
	TOU12 Counter	TO12CNT	H'0000 0000	H'FFFF E744	32	21-118, 21-120
	TOU12 Reload Register	TO12RLD	H'0000 0000	H'FFFF E748	32	21-122
	TOU13 Mode Control Register	TO13MCR	H'00	H'FFFF E750	8	21-115
	TOU13 Counter	TO13CNT	H'0000 0000	H'FFFF E754	32	21-118, 21-120
	TOU13 Reload Register	TO13RLD	H'0000 0000	H'FFFF E758	32	21-122
	TOU14 Mode Control Register	TO14MCR	H'00	H'FFFF E760	8	21-115
	TOU14 Counter	TO14CNT	H'0000 0000	H'FFFF E764	32	21-118, 21-120
	TOU14 Reload Register	TO14RLD	H'0000 0000	H'FFFF E768	32	21-122
	TOU15 Mode Control Register	TO15MCR	H'00	H'FFFF E770	8	21-115
	TOU15 Counter	TO15CNT	H'0000 0000	H'FFFF E774	32	21-118, 21-120
	TOU15 Reload Register	TO15RLD	H'0000 0000	H'FFFF E778	32	21-122
	TOU16 Mode Control Register	TO16MCR	H'00	H'FFFF E780	8	21-115
	TOU16 Counter	TO16CNT	H'0000 0000	H'FFFF E784	32	21-118, 21-120
	TOU16 Reload Register	TO16RLD	H'0000 0000	H'FFFF E788	32	21-122
	TOU17 Mode Control Register	TO17MCR	H'00	H'FFFF E790	8	21-115
	TOU17 Counter	TO17CNT	H'0000 0000	H'FFFF E794	32	21-118, 21-120
	TOU17 Reload Register	TO17RLD	H'0000 0000	H'FFFF E798	32	21-122
Timer TOU2	TOU2 Control Register	TO2CR	H'00	H'FFFF E800	8	21-95
	TOU2 Timer Interrupt Enable Register	TO2IER	H'00	H'FFFF E801	8	21-96
	TOU2 Output Control Register	TO2OUCR	H'00	H'FFFF E802	8	21-97
	TOU2 Status Register	TO2SR	H'00	H'FFFF E803	8	21-98
	TOU2 Counter Enable Protect Register	TO2CEPR	H'00	H'FFFF E804	8	21-99
	Flip-Flop Output Protect Register for TOU2 Short-Circuit Prevention Function	TO2SHFFPR	H'00	H'FFFF E805	8	21-102
	TOU2 Flip-Flop Output Protect Register	TO2FFPR	H'00	H'FFFF E806	8	21-104
	TOU2 Counter Enable Register	TO2CENR	H'00	H'FFFF E808	8	21-100
	Flip-Flop Output Data Register for TOU2 Short-Circuit Prevention Function	TO2SHFFDR	H'00	H'FFFF E809	8	21-103

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer TOU2	TOU2 Flip-Flop Output Data Register	TO2FFDR	H'00	H'FFFF E80A	8	21-105
	TOU20 Mode Control Register	TO20MCR	H'00	H'FFFF E820	8	21-115
	TOU20 Counter	TO20CNT	H'0000 0000	H'FFFF E824	32	21-118, 21-120
	TOU20 Reload Register	TO20RLD	H'0000 0000	H'FFFF E828	32	21-123
	TOU21 Mode Control Register	TO21MCR	H'00	H'FFFF E830	8	21-115
	TOU21 Counter	TO21CNT	H'0000 0000	H'FFFF E834	32	21-118, 21-120
	TOU21 Reload Register	TO21RLD	H'0000 0000	H'FFFF E838	32	21-123
	TOU22 Mode Control Register	TO22MCR	H'00	H'FFFF E840	8	21-115
	TOU22 Counter	TO22CNT	H'0000 0000	H'FFFF E844	32	21-118, 21-120
	TOU22 Reload Register	TO22RLD	H'0000 0000	H'FFFF E848	32	21-123
	TOU23 Mode Control Register	TO23MCR	H'00	H'FFFF E850	8	21-115
	TOU23 Counter	TO23CNT	H'0000 0000	H'FFFF E854	32	21-118, 21-120
	TOU23 Reload Register	TO23RLD	H'0000 0000	H'FFFF E858	32	21-123
	TOU24 Mode Control Register	TO24MCR	H'00	H'FFFF E860	8	21-115
	TOU24 Counter	TO24CNT	H'0000 0000	H'FFFF E864	32	21-118, 21-120
	TOU24 Reload Register	TO24RLD	H'0000 0000	H'FFFF E868	32	21-123
	TOU25 Mode Control Register	TO25MCR	H'00	H'FFFF E870	8	21-115
	TOU25 Counter	TO25CNT	H'0000 0000	H'FFFF E874	32	21-118, 21-120
	TOU25 Reload Register	TO25RLD	H'0000 0000	H'FFFF E878	32	21-123
	TOU26 Mode Control Register	TO26MCR	H'00	H'FFFF E880	8	21-115
	TOU26 Counter	TO26CNT	H'0000 0000	H'FFFF E884	32	21-118, 21-120
	TOU26 Reload Register	TO26RLD	H'0000 0000	H'FFFF E888	32	21-123
	TOU27 Mode Control Register	TO27MCR	H'00	H'FFFF E890	8	21-115
	TOU27 Counter	TO27CNT	H'0000 0000	H'FFFF E894	32	21-118, 21-120
	TOU27 Reload Register	TO27RLD	H'0000 0000	H'FFFF E898	32	21-123
Timer TOU3	TOU3 Control Register	TO3CR	H'00	H'FFFF E900	8	21-95
	TOU3 Timer Interrupt Enable Register	TO3IER	H'00	H'FFFF E901	8	21-96
	TOU3 Output Control Register	TO3OUCR	H'00	H'FFFF E902	8	21-97
	TOU3 Status Register	TO3SR	H'00	H'FFFF E903	8	21-98
	TOU3 Counter Enable Protect Register	TO3CEPR	H'00	H'FFFF E904	8	21-99
	Flip-Flop Output Protect Register for TOU3 Short-Circuit Prevention Function	TO3SHFFPR	H'00	H'FFFF E905	8	21-102
	TOU3 Flip-Flop Output Protect Register	TO3FFPR	H'00	H'FFFF E906	8	21-104
	TOU3 Counter Enable Register	TO3CENR	H'00	H'FFFF E908	8	21-100
	Flip-Flop Output Data Register for TOU3 Short-Circuit Prevention Function	TO3SHFFDR	H'00	H'FFFF E909	8	21-103
	TOU3 Flip-Flop Output Data Register	TO3FFDR	H'00	H'FFFF E90A	8	21-105

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer TOU3	TOU3 Noise Canceler Control Register	TO3NCCR	H'00	H'FFFF E90C	8	21-106
	TOU3 Noise Canceler Counter	TO3NCNT	H'00	H'FFFF E90E	8	21-108
	TOU3 Noise Canceler Register	TO3NCR	H'00	H'FFFF E90F	8	21-109
	TOU3PWMOFF Input Processing Register	TO3POCR	H'0000	H'FFFF E910	16	21-110
	TOU3PWMOFF Function Enable Register	TO3POER	H'00	H'FFFF E913	8	21-111
	TOU3PWM Output-Prohibit Control Register	TO3PODISCR	H'0000	H'FFFF E914	16	21-112
	TOU3PWM Output-Prohibit Level Control Register	TO3POLVCR	H'00	H'FFFF E917	8	21-113
	TOU30 Mode Control Register	TO30MCR	H'00	H'FFFF E920	8	21-116
	TOU30 Counter	TO30CNT	H'0000 0000	H'FFFF E924	32	21-119, 21-121
	TOU30 Reload Register	TO30RLD	H'0000 0000	H'FFFF E928	32	21-123
	TOU31 Mode Control Register	TO31MCR	H'00	H'FFFF E930	8	21-116
	TOU31 Counter	TO31CNT	H'0000 0000	H'FFFF E934	32	21-119, 21-121
	TOU31 Reload Register	TO31RLD	H'0000 0000	H'FFFF E938	32	21-123
	TOU32 Mode Control Register	TO32MCR	H'00	H'FFFF E940	8	21-116
	TOU32 Counter	TO32CNT	H'0000 0000	H'FFFF E944	32	21-119, 21-121
	TOU32 Reload Register	TO32RLD	H'0000 0000	H'FFFF E948	32	21-123
	TOU33 Mode Control Register	TO33MCR	H'00	H'FFFF E950	8	21-116
	TOU33 Counter	TO33CNT	H'0000 0000	H'FFFF E954	32	21-119, 21-121
	TOU33 Reload Register	TO33RLD	H'0000 0000	H'FFFF E958	32	21-123
	TOU34 Mode Control Register	TO34MCR	H'00	H'FFFF E960	8	21-116
	TOU34 Counter	TO34CNT	H'0000 0000	H'FFFF E964	32	21-119, 21-121
	TOU34 Reload Register	TO34RLD	H'0000 0000	H'FFFF E968	32	21-123
	TOU35 Mode Control Register	TO35MCR	H'00	H'FFFF E970	8	21-116
	TOU35 Counter	TO35CNT	H'0000 0000	H'FFFF E974	32	21-119, 21-121
	TOU35 Reload Register	TO35RLD	H'0000 0000	H'FFFF E978	32	21-123
	TOU36 Mode Control Register	TO36MCR	H'00	H'FFFF E980	8	21-116
	TOU36 Counter	TO36CNT	H'0000 0000	H'FFFF E984	32	21-119, 21-121
	TOU36 Reload Register	TO36RLD	H'0000 0000	H'FFFF E988	32	21-123
	TOU37 Mode Control Register	TO37MCR	H'00	H'FFFF E990	8	21-116
	TOU37 Counter	TO37CNT	H'0000 0000	H'FFFF E994	32	21-119, 21-121
	TOU37 Reload Register	TO37RLD	H'0000 0000	H'FFFF E998	32	21-123
Timer TOU4	TOU4 Control Register	TO4CR	H'00	H'FFFF EA00	8	21-95
	TOU4 Timer Interrupt Enable Register	TO4IER	H'00	H'FFFF EA01	8	21-96
	TOU4 Output Control Register	TO4OUCR	H'00	H'FFFF EA02	8	21-97
	TOU4 Status Register	TO4SR	H'00	H'FFFF EA03	8	21-98

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer TOU4	TOU4 Counter Enable Protect Register	TO4CEPR	H'00	H'FFFF EA04	8	21-99
	Flip-Flop Output Protect Register for TOU4 Short-Circuit Prevention Function	TO4SHFFPR	H'00	H'FFFF EA05	8	21-102
	TOU4 Flip-Flop Output Protect Register	TO4FFPR	H'00	H'FFFF EA06	8	21-104
	TOU4 Counter Enable Register	TO4CENR	H'00	H'FFFF EA08	8	21-100
	Flip-Flop Output Data Register for TOU4 Short-Circuit Prevention Function	TO4SHFFDR	H'00	H'FFFF EA09	8	21-103
	TOU4 Flip-Flop Output Data Register	TO4FFDR	H'00	H'FFFF EA0A	8	21-105
	TOU4 Noise Canceler Control Register	TO4NCCR	H'00	H'FFFF EA0C	8	21-106
	TOU4 Noise Canceler Counter	TO4NCNT	H'00	H'FFFF EA0E	8	21-108
	TOU4 Noise Canceler Register	TO4NCR	H'00	H'FFFF EA0F	8	21-109
	TOU4PWMOFF Input Processing Register	TO4POCR	H'0000	H'FFFF EA10	16	21-110
	TOU4PWMOFF Function Enable Register	TO4POER	H'00	H'FFFF EA13	8	21-111
	TOU4PWM Output-Prohibit Control Register	TO4PODISCR	H'0000	H'FFFF EA14	16	21-112
	TOU4PWM Output-Prohibit Level Control Register	TO4POLVCR	H'00	H'FFFF EA17	8	21-113
	TOU40 Mode Control Register	TO40MCR	H'00	H'FFFF EA20	8	21-116
	TOU40 Counter	TO40CNT	H'0000 0000	H'FFFF EA24	32	21-119, 21-121
	TOU40 Reload Register	TO40RLD	H'0000 0000	H'FFFF EA28	32	21-123
	TOU41 Mode Control Register	TO41MCR	H'00	H'FFFF EA30	8	21-116
	TOU41 Counter	TO41CNT	H'0000 0000	H'FFFF EA34	32	21-119, 21-121
	TOU41 Reload Register	TO41RLD	H'0000 0000	H'FFFF EA38	32	21-123
	TOU42 Mode Control Register	TO42MCR	H'00	H'FFFF EA40	8	21-116
	TOU42 Counter	TO42CNT	H'0000 0000	H'FFFF EA44	32	21-119, 21-121
	TOU42 Reload Register	TO42RLD	H'0000 0000	H'FFFF EA48	32	21-123
	TOU43 Mode Control Register	TO43MCR	H'00	H'FFFF EA50	8	21-116
	TOU43 Counter	TO43CNT	H'0000 0000	H'FFFF EA54	32	21-119, 21-121
	TOU43 Reload Register	TO43RLD	H'0000 0000	H'FFFF EA58	32	21-123
	TOU44 Mode Control Register	TO44MCR	H'00	H'FFFF EA60	8	21-116
	TOU44 Counter	TO44CNT	H'0000 0000	H'FFFF EA64	32	21-119, 21-121
	TOU44 Reload Register	TO44RLD	H'0000 0000	H'FFFF EA68	32	21-123
	TOU45 Mode Control Register	TO45MCR	H'00	H'FFFF EA70	8	21-116
	TOU45 Counter	TO45CNT	H'0000 0000	H'FFFF EA74	32	21-119, 21-121
	TOU45 Reload Register	TO45RLD	H'0000 0000	H'FFFF EA78	32	21-123
	TOU46 Mode Control Register	TO46MCR	H'00	H'FFFF EA80	8	21-116
	TOU46 Counter	TO46CNT	H'0000 0000	H'FFFF EA84	32	21-119, 21-121
	TOU46 Reload Register	TO46RLD	H'0000 0000	H'FFFF EA88	32	21-123
	TOU47 Mode Control Register	TO47MCR	H'00	H'FFFF EA90	8	21-116

Item	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Timer TOU4	TOU47 Counter	TO47CNT	H'0000 0000	H'FFFF EA94	32	21-119, 21-121
	TOU47 Reload Register	TO47RLD	H'0000 0000	H'FFFF EA98	32	21-123

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

21.5 Overview of Common Controller

The common controller controls the ATU-IIIS module as a whole. For example, it enables and disables the prescalers and timer counters for timers A, F, G, TOU, and controls the clock bus.

21.5.1 Clock Bus

The clock bus consists of five signal lines used to distribute the source signals for counting (count enabling signals) to the timer channels. The timer counters on each of the channels run in synchronization with the peripheral clock (Pck). The clock bus signals function as count enabling signals for the counters.

Table 21.5 shows the signals which are available for input on each clock bus.

Table 21.5 Signals to be Input on Each Clock Bus

Bit Number of Clock Bus	Input Signals
5	External input clock B (TCLKB)
4	The MCU does not support this setting. Do not set.
3	Output signal from prescaler 3
2	Output signal from prescaler 2
1	Output signal from prescaler 1
0	Output signal from prescaler 0

21.6 Register Description of Common Controller

21.6.1 ATU-IIIS Master Enable Register (ATUENR)

The ATUENR register is used to enable and disable the prescalers and the individual timers in ATU-IIIS. Setting an enable bit to "1" enables the corresponding timer. Clearing the bit to "0" disables the corresponding timer. Even when the enable bit is cleared to "0", the registers of the corresponding timer remain accessible.

Timers can be synchronized by simultaneously setting multiple bits to "1". Note that a particular subblock cannot be synchronized with other subblocks while they are counting.

ATU-IIIS Master Enable Register (ATUENR) <P4 address: location H'FFFF E000>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTE	—	—	—	—	—	—	—	TGE	TFE	—	—	—	—	TAE	PSCE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	TTE	0	R	W	Timer TOU Enable Bit Enables and disables counter operation of timers TOU0 to TOU4. When the counters are disabled, their values are retained. When the TTE bit is again set to "1", the counters resume counting from the retained values. However, do not change the operating mode, clock source, or count enable source. 0: Operation of timers TOU0 to TOU4 disabled 1: Operation of timers TOU0 to TOU4 enabled
14 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7	TGE	0	R	W	Timer G Enable Bit Enables and disables counter operation of timer G. When the counter is disabled, its value is retained. When this bit is again set to "1", the counter resumes counting from the retained value. However, the corresponding bit in the timer G start register must also be set to 1 to enable the operation of either of the subblock counters. 0: Timer G counter operation disabled 1: Timer G counter operation enabled
6	TFE	0	R	W	Timer F Enable Bit Enables and disables counter operation of timer F. When the counter is disabled, its value is retained. When this bit is again set to "1", the counter resumes counting from the retained value. However, the corresponding bit in the timer F start register must also be set to 1 to enable the operation of either of the subblock counters. 0: Timer F counter operation disabled 1: Timer F counter operation enabled
5 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
1	TAE	0	R	W	Timer A Enable Bit Enables and disables counter operation of timer Ai. When the counter is disabled, its value is retained. When this bit is again set to "1", the counter resumes counting from the retained value. 0: Timer Ai counter operation disabled 1: Timer Ai counter operation enabled
0	PSCE	0	R	W	Prescaler Enable Bit Enables and disables the prescaler counters. When the prescaler counters are disabled, the counter values are retained. Once the bit is set to "1" again, the counter resumes counting from the retained value. 0: Prescaler counter operation disabled 1: Prescaler counter operation enabled

21.6.2 ATU-IIIS Clock Bus Control Register (ATCBCNT)

The ATCBCNT register selects the source of the clock signal to be supplied on the clock bus and the valid edge of external clock signals.

ATU-IIIS Clock Bus Control Register (ATCBCNT) <P4 address: location H'FFFF E002>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CB5EG	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1, 0	CB5EG	00	R	W	Clock Bus 5 Edge Select Bits These bits select the edge sense for external input clock B (TCLKB). The clock signal is output on signal line 5 of the clock bus. Counters for which signal line 5 of the clock bus has been selected as the source for counting count on the edge selected by these bits. The setting of these bits is only valid when the TCLKB signal is selected as the source for line 5 of the clock bus. 00: Neither edge of the external clock is sensed 01: Rising edges of the external clock are sensed 10: Falling edges of the external clock are sensed 11: Both rising and falling edges of the external clock are sensed

21.6.3 ATU-IIIS Noise Cancellation Mode Register (ATNCMR)

The ATNCMR register selects the mode and clock to drive the counter for of the noise canceler in each of timers A, F, and TOU.

In premature-transition cancellation mode, subsequent changes to the input signal level are ignored if they come within a given period of a detected change. That is, level changes within a certain period of an initial one are treated as noise.

In minimum time-at-level cancellation mode, the first and subsequent level changes are ignored unless the input signal level remains the same over a given period. Level changes occurring within a shorter period are considered to indicate an unstable signal, and such signals are treated as noise.

The period is set by noise canceler registers in each of the applicable blocks (i.e. in timers A, F, and TOU) and is counted by a noise canceler counter.

Figures 21.3 and 21.4 show outlines of noise cancellation operation (using the TIA00 input signal of timer A as an example) in premature-transition cancellation mode and minimum time-at-level cancellation mode, respectively.

The edge for counting is detected from signals after noise removal in timers A, F, and TOU. Rising edges are being detected in figures 21.3 and 21.4.

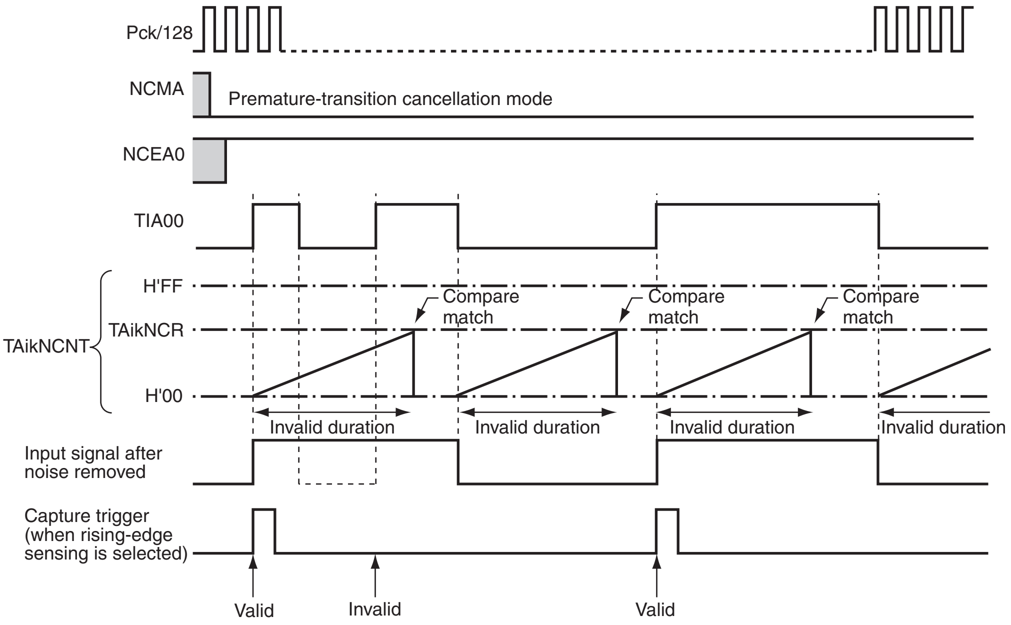
ATU-IIIS Noise Cancellation Mode Register (ATNCMR) <P4 address: location H'FFFF E003>

Bit:	7	6	5	4	3	2	1	0
	NCC SEL	NCMT	—	—	—	NCMF	—	NCMA
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

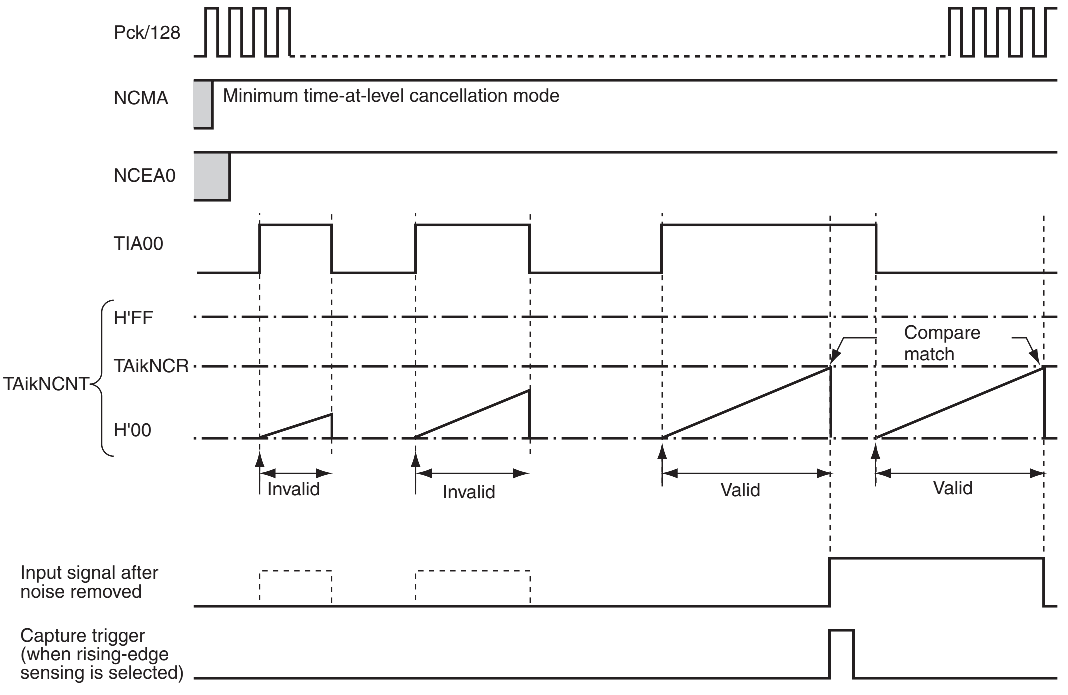
Bit	Abbreviation	After Reset	R	W	Description
7	NCCSEL	0	R	W	Noise Canceler Counter Clock Select Bit Selects the clock for counting by the noise cancelers. The peripheral clock (Pck) or Pck divided by 128 can be selected. The default setting is the clock divided by 128. The same counter clock must be used for all timers other than timer A. It is also possible to select the clock signal on clock bus 5 as the count source. For details, see section 21.11.3, TAiI/O Control Register 2 (TAiIO2). 0: Peripheral clock (Pck) divided by 128 is used as the counter clock 1: Peripheral clock (Pck) is used as the counter clock
6	NCMT	0	R	W	Timer TOU Noise Cancellation Mode Bit Selects the noise cancellation mode for timer TOU. The same mode is used on all channels of timer TOU. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode
5 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2	NCMF	0	R	W	Timer F Noise Cancellation Mode Bit Selects the noise cancellation mode for timer F. The same mode is used on both channels of timer F. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode
1	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
0	NCMA	0	R	W	Timer A Noise Cancellation Mode Bit Selects the noise cancellation mode for timer A. The same mode is used on both channels of timer A. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode



Note: • This figure is purely schematic and does not show accurate timing.

Figure 21.3 Operation of Noise Canceler in Premature-Transition Cancellation Mode


Pck/128
NCMA
Minimum time-at-level cancellation mode
NCEA0
TIA00
H'FF
TAikNCR
TAikNCNT
H'00
Compare match
Invalid
Invalid
Valid
Valid
Input signal after noise removed
Capture trigger (when rising-edge sensing is selected)

Note: • This figure is purely schematic and does not show accurate timing.

Figure 21.4 Operation of Noise Canceler in Minimum-Time-at Level Cancellation Mode

21.6.4 ATU-IIIS Interrupt Select Register Ai (ATISRAi)

The ATISRAi register selects whether an interrupt request triggered according to the setting of the TAik input capture register (TAikICR) of the TAi free-running counter (TAiTCNT) is output to the INTC or to the DMAC. By selecting output to the DMAC, it is possible to activate the DMAC when an interrupt request is triggered according to the above input capture condition.

ATU-IIIS Interrupt Select Register A0 (ATISRA0) <P4 address: location H'FFFF E010>
ATU-IIIS Interrupt Select Register A1 (ATISRA1) <P4 address: location H'FFFF E011>

Bit:	7	6	5	4	3	2	1	0
	—	—	IRSEL Ai5	IRSEL Ai4	IRSEL Ai3	IRSEL Ai2	IRSEL Ai1	IRSEL Ai0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	IRSELAi5	0	R	W	Interrupt Request Select Bits (Aik) These bits select whether an interrupt request triggered by the ICFAk bit, while the ICFAk bit in the TAiSR register is set to "1", is output to the INTC or to the DMAC. The DMA transfer request sources are not included in timer A1 channels 0 to 5. Always write "0" to bits IRSELA10 to IRSELA15 in the ATISRA1 register. 0: Input capture interrupt request output to INTC 1: Input capture interrupt request output to DMAC
4	IRSELAi4	0	R	W	
3	IRSELAi3	0	R	W	
2	IRSELAi2	0	R	W	
1	IRSELAi1	0	R	W	
0	IRSELAi0	0	R	W	

Legend: i = 0 or 1, k = 0 to 5

21.6.5 ATU-IIIS Interrupt Select Register F (ATISRF)

The ATISRF register selects whether an interrupt request triggered by the ICFFj bit in the TFjSR register is output to the INTC or to the DMAC. By selecting output to the DMAC, it is possible to activate the DMAC when an interrupt request is triggered according to the above input capture condition.

ATU-IIIS Interrupt Select Register F (ATISRF) <P4 address: location H'FFFF E014>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	IRSEL F2	IRSEL F1	IRSEL F0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	IRSELF2	0	R	0	Interrupt Request Select Bits (Fj)
1	IRSELF1	0	R	W	These bits select whether an interrupt request triggered by the ICFFj bit is output to the INTC or to the DMAC. The DMA transfer request source is not included in timer F2. Always write "0" to bit IRSELF2. 0: Input capture interrupt request output to INTC 1: Input capture interrupt request output to DMAC
0	IRSELF0	0	R	W	

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.6.6 ATU-IIIS Interrupt Select Register G (ATISRG)

The ATISRG register selects whether an interrupt request triggered according to the setting of the TGk compare match register (TGkOCR) is output to the INTC or to the DMAC. By selecting output to the DMAC, it is possible to activate the DMAC when an interrupt request is triggered according to the above compare match condition.

ATU-IIIS Interrupt Select Register G (ATISRG) <P4 address: location H'FFFF E018>

Bit:	7	6	5	4	3	2	1	0
	—	—	IRSEL G5	IRSEL G4	IRSEL G3	IRSEL G2	IRSEL G1	IRSEL G0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	IRSELG5	0	R	W	Interrupt Request Select Bits (Gk) These bits select whether an interrupt request triggered by the CMFGk bit, while the CMFGk bit in the TGkSR register is set to "1", is output to the INTC or to the DMAC. By selecting output to the DMAC, it is possible to activate the DMAC when an interrupt request is triggered. The DMA transfer request sources are not included in timers G0 and G2. Always write "0" to bits IRSELG0 to IRSELG2. 0: Compare match interrupt request output to INTC 1: Compare match interrupt request output to DMAC
4	IRSELG4	0	R	W	
3	IRSELG3	0	R	W	
2	IRSELG2	0	R	0	
1	IRSELG1	0	R	0	
0	IRSELG0	0	R	0	

Legend: k = 0 to 5, corresponding to subclocks G0 to G5, respectively

21.6.7 ATU-IIIS Interrupt Select Registers TOU0 to TOU4 (ATISRT0 to ATISRT4)

The ATISRT0 to ATISRT4 registers select whether an interrupt request triggered by an underflow of one of timer counters TOU00 to TOU47 is output to the INTC or to the DMAC. By selecting output to the DMAC, it is possible to activate the DMAC when an interrupt request is triggered

ATU-IIIS Interrupt Select Register TOU0 (ATISRT0)	<P4 address: location H'FFFF E020>
ATU-IIIS Interrupt Select Register TOU1 (ATISRT1)	<P4 address: location H'FFFF E021>
ATU-IIIS Interrupt Select Register TOU2 (ATISRT2)	<P4 address: location H'FFFF E022>
ATU-IIIS Interrupt Select Register TOU3 (ATISRT3)	<P4 address: location H'FFFF E023>
ATU-IIIS Interrupt Select Register TOU4 (ATISRT4)	<P4 address: location H'FFFF E024>

Bit:	7	6	5	4	3	2	1	0
	IRSEL Tn7	IRSEL Tn6	IRSEL Tn5	IRSEL Tn4	IRSEL Tn3	IRSEL Tn2	IRSEL Tn1	IRSEL Tn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

		After			
Bit	**Abbreviation**	**Reset**	**R**	**W**	**Description**
7	IRSELTn7	0	R	W	Interrupt Request Select Bits (Tnk) These bits select whether an interrupt request triggered by an underflow of one of timer counters TOU00 to TOU47 is output to the INTC or to the DMAC. The DMA transfer request sources are not included in timers TOUn_1 to TOUn_6. Always write "0" to bits IRSELTn1 to IRSELTn6. 0: Underflow interrupt request output to INTC 1: Underflow interrupt request output to DMAC
6	IRSELTn6	0	R	0	
5	IRSELTn5	0	R	0	
4	IRSELTn4	0	R	0	
3	IRSELTn3	0	R	0	
2	IRSELTn2	0	R	0	
1	IRSELTn1	0	R	0	
0	IRSELTn0	0	R	W	

Legend: n = 0 to 4, k = 0 to 7

21.7 Overview of Prescalers

ATU-IIIS includes four general prescalers and a prescaler for the noise-canceler clock.

The general prescalers are implemented as 10-bit down-counters, in which the prescaled clock signals are derived by frequency-dividing the peripheral clock (Pck) by N ($1 \leq N \leq 1024$).

The division ratio is obtained from the following expression.

$$\text{Division ratio of prescaler} = \frac{1}{\text{PSCn[9:0]} + 1} \quad \text{(Settable value: 1/1 to 1/1024)}$$

A duty cycle of 50% is not guaranteed for the clock-signal outputs of the prescalers. Instead, the high level is output in one cycle of the Pck clock and the low level is output in the remaining cycles within the prescaled period. When 1/1 is selected as the division ratio, the high level is always output on the clock bus.

The generated clock signals are supplied to the individual timers via the clock bus. The prescalers for each of the channels operate independently. The prescalers can, however, be started in synchronization with each other after a reset by setting the PSCE bit in the ATUENR register to "1" after the appropriate values have been set. Synchronization of the prescalers is not possible when its division ratio has been changed after the prescalers has started.

The prescaler for the noise-canceler clock is implemented as a 7-bit down-counter. This generates a clock signal by frequency-dividing the peripheral clock (Pck) by 128. The clock signal thus generated is supplied to the noise cancelers of timers A, F, and TOU.

A division ratio of 1/1 or 1/128 can be selected for the noise-canceler clock by means of the NCCSEL bit in the ATNCMR register of the common controller. Further division ratios are not available.

The down-counters of the prescalers are initialized to H'000 by a hardware reset.

21.8 Register Description of Prescalers

21.8.1 ATU-IIIS Prescaler Registers 0 to 3 (ATPSCR0 to ATPSCR3)

The ATPSCR0 to ATPSCR3 registers, each of which holds a division ratio for one of the four prescalers.

After a prescaler counter underflows, counting restarts from the setting in this register. Settable values range from H'000 to H'3FF.

ATU-IIIS Prescaler Register 0 (ATPSCR0)	<P4 address: location H'FFFF E100>
ATU-IIIS Prescaler Register 1 (ATPSCR1)	<P4 address: location H'FFFF E102>
ATU-IIIS Prescaler Register 2 (ATPSCR2)	<P4 address: location H'FFFF E104>
ATU-IIIS Prescaler Register 3 (ATPSCR3)	<P4 address: location H'FFFF E106>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PSC									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	PSC	All 0	R	W	Division Ratio These bits specify the division ratio for the corresponding prescaler.

21.9 Operation of Prescalers

21.9.1 Starting Prescalers

The prescalers start operating when the PSCE bit in the ATU-IIIS master enable register (ATUENR) is set to "1" and generates a clock with a frequency given by the division ratio in the PSC bit. While a prescaler is operating, the "H" level is output for one cycle of the Pck clock each time the corresponding prescaler counter underflows.

When the setting in the PSC bit is changed during operation, the division ratio of the output clock is updated on the first subsequent counter underflow.

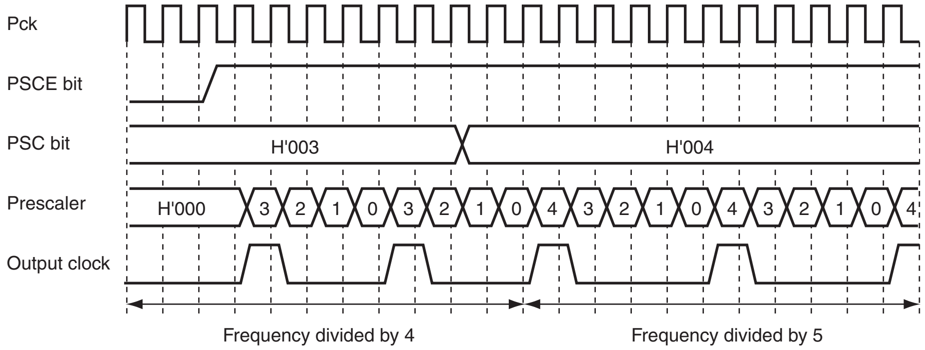


Figure 21.5 Starting Prescaler

21.9.2 Stopping and Restarting Operation

The prescaler stops operating when the PSCE bit in the ATU-IIIS master enable register (ATUENR) is cleared to "0". The clock signal stays at the "L" level and the value in the prescaler counter is retained while the prescaler is stopped. Setting the PSCE bit to "1" makes counting restart from the retained value.

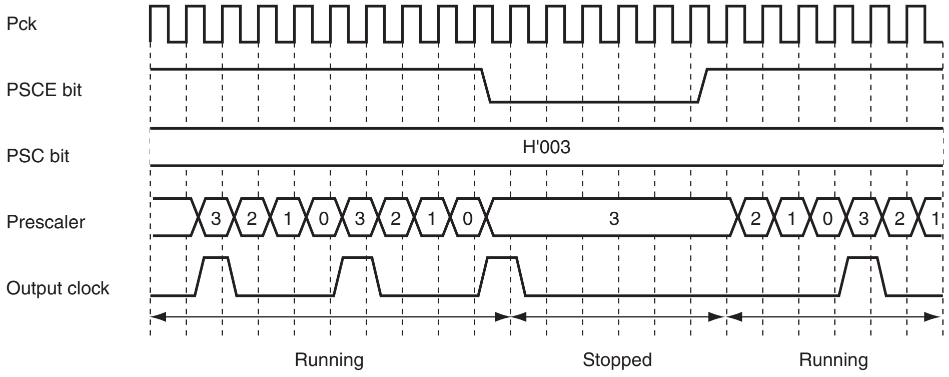


Figure 21.6 Stopping Prescaler

21.10 Overview of Timer A

On this MCU, the ATU-IIIS includes two timer A subblocks, each with five channels.

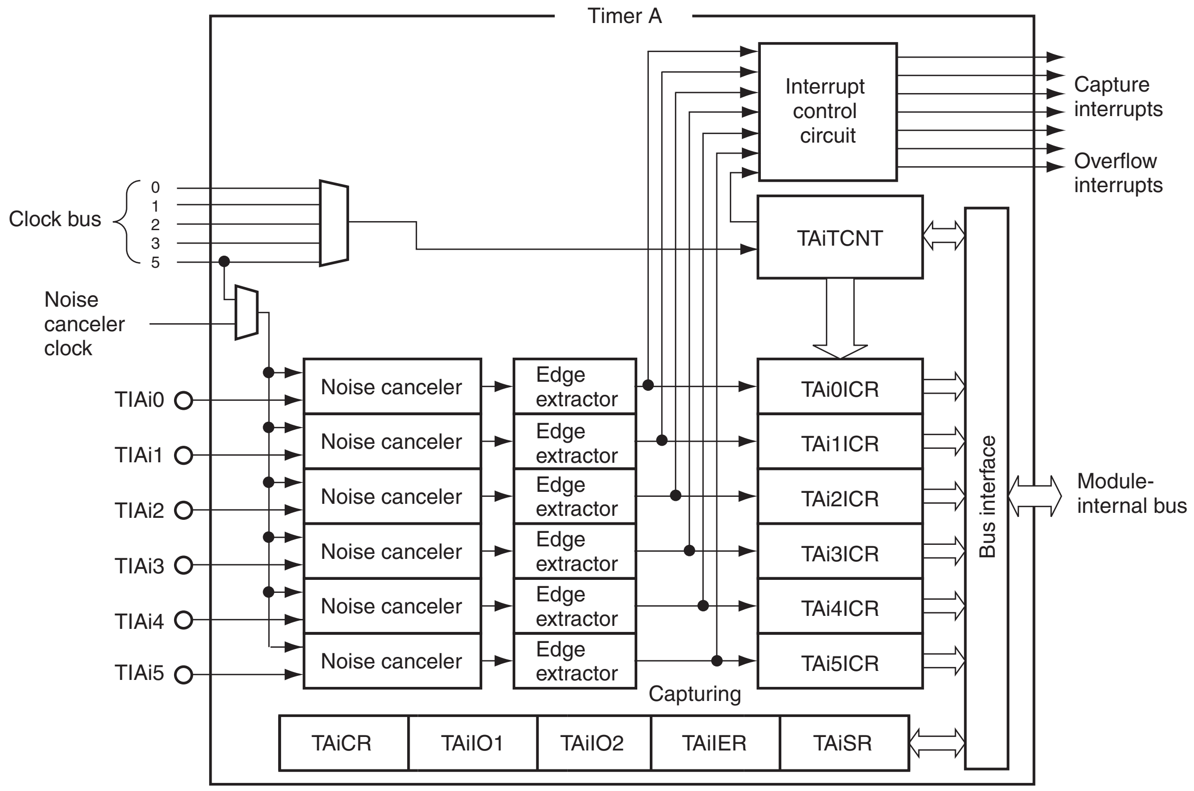
Timer A incorporates a TAi free-running counter (TAiTCNT) and six TAik input capture registers (TAikICR). TAiTCNT is a free-running up counter. An interrupt request can be output when the counter overflows.

The six TAik input capture registers (TAikICR) capture the value of the TAi free-running counter (TAiTCNT) at input on the corresponding external signal input pin (TIAi0 to TIAi5). The rising or falling edge, or both edges, of the input on the external signal input pin can be selected as the trigger for capture by setting TAiI/O control register 1 (TAiIO1). In each case, the DMAC can be activated or an interrupt request generated according to the capture timing.

Noise in the input on the external signal input pin can be removed by using the noise canceling function.

21.10.1 Block Diagram

Figure 21.7 is a block diagram of timer Ai.



Legend:

TAiCR	: TAi control register
TAiIO1	: TAi I/O control register 1
TAiIO2	: TAi I/O control register 2
TAiIER	: TAi interrupt enable register
TAiSR	: TAi status register
TAikICR	: TAi input capture register
TAiTCNT	: TAi free-running counter

i = 0, 1

Figure 21.7 Block Diagram of Timer A

21.11 Description of Timer A Registers

21.11.1 TAi Control Register (TAiCR)

The TAiCR register selsects the counter clock.

TA0 Control Register (TA0CR) <P4 address: location H'FFFF E202>
TA1 Control Register (TA1CR) <P4 address: location H'FFFF E302>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	CKSELA		
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	CKSELA	000	R	W	Clock Select Bit A These bits select the signal on one of clock-bus lines 0 to 3 and 5 as the clock signal for counting. The signal on lines 0 to 3 have been frequency-divided by prescalers 0 to 3, respectively. Clock-bus line 5 supplies externally input clock B (TCLKB). Stop timer A before making or changing the counter-clock selection. 000: Clock-bus line 0 (prescaler 0) 001: Clock-bus line 1 (prescaler 1) 010: Clock-bus line 2 (prescaler 2) 011: Clock-bus line 3 (prescaler 3) 100: Setting prohibited 101: Clock-bus line 5 (TCLKB) 110: Setting prohibited 111: Setting prohibited

Note: • The edge of an external input clock is extracted before it is output on a clock bus. When using external input clock B, select the edge to be extracted by setting the CB5EG bit in the ATU-IIIS clock bus control register (ATCBCNT).

21.11.2 TAiI/O Control Register 1 (TAiIO1)

The TAiIO1 register sets the edge of external inputs (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15) to be extracted.

TA0I/O Control Register 1 (TA0IO1) <P4 address: location H'FFFF E204>
TA1I/O Control Register 1 (TA1IO1) <P4 address: location H'FFFF E304>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	IOA5		IOA4		IOA3		IOA2		IOA1		IOA0	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11, 10	IOA5	00	R	W	I/O Control Bits (Ak) These bits select the edge of external inputs (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15) that is to be extracted for use in input-capture triggering. When these bits are set to "B'00", input capturing is not performed. When a value other than "B'00" is set, the TAiTCNT counter value is transferred to the corresponding TAikICR register at extraction of the selected edge. Edge extraction is synchronized with the Pck clock. Make sure that the frequency of the Pck clock is at least twice the frequency of the external input signal. Otherwise, edge extraction will not performed correctly. Edge extraction is executed to a signal after noise removed. When the noise canceler is disabled, the selected edge is simply extracted from the external inputs (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15). When the noise canceler is enabled, the selected edge is extracted from the signals after noise removal. 00: Input capturing is not performed 01: The TAiTCNT counter value is captured in the TAikICR register at the rising edge of TIA 10: The TAiTCNT counter value is captured in the TAikICR register at the falling edge of TIA 11: The TAiTCNT counter value is captured in the TAikICR register at both the rising and falling edges of TIA
9, 8	IOA4	00	R	W	
7, 6	IOA3	00	R	W	
5, 4	IOA2	00	R	W	
3, 2	IOA1	00	R	W	
1, 0	IOA0	00	R	W	

Legend: i = 0, 1, k = 0 to 5

21.11.3 TAiI/O Control Register 2 (TAiIO2)

The TAiIO2 register selects the noise canceler clock and enables and disables the noise cancelers for externally input signals (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15).

TA0I/O Control Register 2 (TA0IO2) <P4 address: location H'FFFF E206>
TA1I/O Control Register 2 (TA1IO2) <P4 address: location H'FFFF E306>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	NCKA5	NCKA4	NCKA3	NCKA2	NCKA1	NCKA0	—	—	NCEA5	NCEA4	NCEA3	NCEA2	NCEA1	NCEA0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
15, 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13	NCKA5	0	R	W	Noise Canceler Clock Select Bits (Ak)
12	NCKA4	0	R	W	These bits select the count source clock of the TAik noise canceler counter (TAikNCNT). The noise canceler count clock or the signal on clock-bus line 5 can be selected as the signal for counting. Either the Pck clock frequency divided by 128 or the Pck clock can be selected as the noise canceler count clock by setting the NCCSEL bit of the common controller. 0: Noise canceler count clock is selected as the signal for counting by the TAikNCNT counter 1: Clock-bus line 5 is selected as the signal for counting by TAikNCNT counter
11	NCKA3	0	R	W	
10	NCKA2	0	R	W	
9	NCKA1	0	R	W	
8	NCKA0	0	R	W	
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
5	NCEA5	0	R	W	Noise Canceler Enable Bits (Ak) These bits enable and disable the noise cancelers for externally input signals (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15) When a level change on externally input signals TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15 is detected while this bit is set to "1", it is processed in premature-transition cancellation or minimum time-at-level cancellation mode depending on the setting in the noise cancellation mode register (ATNCMR) of the common controller. In premature-transition cancellation mode When a level change of the externally input signal is detected, the change is output as the signal whose noise is removed and the corresponding TAik noise canceler counter (TAikNCNT) is started for counting up. Subsequent level changes are masked until the value in the TAikNCNT counter reaches the value in the TAik noise canceler register (TAikNCNT). The level of the externally input signal is output on this compare match. When these bits are cleared to "0" while the TAikNCNT counter is being incremented, counting continues after the clearing of the bits until a compare match occurs, and level changes in TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15 are masked over this period. In minimum time-at-level cancellation mode When a level change of the externally input signal is detected, the corresponding TAik noise canceler counter (TAikNCNT) is started for counting up. If subsequent level changes are not detected until the value in the counter reaches the value in the TAik noise canceler register (TAikNCR), the previously accepted level change is output as the signal whose noise is removed on compare match of the counter and noise canceler register. When the subsequent level change is detected before the values in the counter and noise canceler register match, all the changes are treated as noise. Therefore the signal whose noise is removed is not changed. When these bits are cleared to "0" while the TAikNCNTcounter is being incremented, counting continues until the values in the counter and the noise canceler register match or a level change on the externally input signal is detected. For details on operations in each mode, see figures 21.3 and 21.4. 0: Noise cancelers for TIA are disabled 1: Noise cancelers for TIA are enabled
4	NCEA4	0	R	W	
3	NCEA3	0	R	W	
2	NCEA2	0	R	W	
1	NCEA1	0	R	W	
0	NCEA0	0	R	W	

Legend: i = 0, 1, k = 0 to 5

21.11.4 TAi Status Register (TAiSR)

The TAiSR register indicates overflow on TAi free-running counter (TAiTCNT) and input capture on TAik input capture register (TAikICR).

The flags in this register are interrupt sources. When an interrupt is enabled by the setting of the corresponding bit in the TAi interrupt enable register (TAiIER), a DMA transfer request can be sent to the DMAC or an interrupt request can be sent to the CPU by setting a flag.

TA0 Status Register (TA0SR) <P4 address: location H'FFFF E208>
TA1 Status Register (TA1SR) <P4 address: location H'FFFF E308>

Bit:	7	6	5	4	3	2	1	0
	OVFA	—	ICFA5	ICFA4	ICFA3	ICFA2	ICFA1	ICFA0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	OVFA	0	R	*[1]	Overflow Flag A This status flag indicates that the TAi free-running counter (TAiTCNT) has overflowed. When this flag returns a value of "1" when read, the counter TAiTCNT has overflowed. This flag cannot be set to "1" by software. To clear the OVFA bit, write "0" to it after reading it as "1". Writing "0" before reading it as "1" has no effect. 0: TAiTCNT has not overflowed 1: TAiTCNT has overflowed [Condition for clearing to "0"] • Writing "0" to OVFA after reading it as "1" [Condition for setting to "1"] • Overflow of TAiTCNT counter value (transition from H'FFFF FFFF to H'0000 0000)
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
5	ICFA5	0	R	*1	Input Capture Flag (Ak) These status flags indicate that the an input capture to the TAik input capture register (TAikICR) has occurred. When a flag returns a value of "1" when read, an input capture to the corresponding input capture register has occurred. These flags cannot be set to "1" by software. A flag is automatically cleared to "0" when the DMAC accepts a DMA transfer request triggered by the corresponding input capture interrupt. An input capture flag can be cleared to "0" by writing "0" to it after reading it as "1". Writing "0" before reading it as "1" has no effect. 0: No input capture has occurred 1: Input capture has occurred [Conditions for clearing to "0"] • Writing "0" to ICFAk after reading it as "1" • When the DMAC accepts a DMA transfer request triggered by the input capture interrupt corresponding to the flag [Condition for setting to "1"] • When the TAiTCNT counter value is transferred to the TAikICR register at assertion of the input capture signal (TIA)
4	ICFA4	0	R	*1	
3	ICFA3	0	R	*1	
2	ICFA2	0	R	*1	
1	ICFA1	0	R	*1	
0	ICFA0	0	R	*1	

Note: *1 Only writing "0" to this bit after reading it as "1" to clear the flag. Writing "1" to this bit has no effect.

Legend: k = 0 to 5

21.11.5 TAi Interrupt Enable Register (TAiIER)

The TAiIER register enables and disables an interrupt request of overflow on TAi free-running counter (TAiTCNT) or input capture on TAik input capture register (TAikICR).

TA0 Interrupt Enable Register (TA0IER) <P4 address: location H'FFFF E209>
TA1 Interrupt Enable Register (TA1IER) <P4 address: location H'FFFF E309>

Bit:	7	6	5	4	3	2	1	0
	OVEA	—	ICEA5	ICEA4	ICEA3	ICEA2	ICEA1	ICEA0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	OVEA	0	R	W	Overflow Interrupt A Enable Bit Enables and disables an OVFA interrupt request when overflow flag A (OVFA) in TAi status register (TAiSR) is set to "1". 0: Disables an overflow interrupt A request 1: Enables an overflow interrupt A request
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5	ICEA5	0	R	W	Input Capture Interrupt Ak Enable Bits Enables and disables an ICFAk bit interrupt request when bit ICFAk in the TAiSR register is set to "1". 0: Disables a request of input capture interrupt Ak 1: Enables a request of input capture interrupt Ak
4	ICEA4	0	R	W	
3	ICEA3	0	R	W	
2	ICEA2	0	R	W	
1	ICEA1	0	R	W	
0	ICEA0	0	R	W	

Legend: i = 0, 1, k = 0 to 5

21.11.6 TAik Input Capture Register (TAikICR)

The TAikICR register is used for input capturing. Writing to these registers is prohibited.

This register holds the contents of TAi free-running counter A (TAiTCNT) when an assertion of input capture signals (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15) is detected. At this time, the ICFAk bit in TAi status register (TAiSR) is set to "1". In addition, when the DMAC accepts a DMA transfer request triggered by an input capture interrupt, the corresponding input capture flag in the TAiSR register is cleared to "0".

The edge to be detected is selected by I/O control bit (IOA) in TAiI/O control register 1 (TAiIO1).

TA00 Input Capture Register (TA00ICR)	<P4 address: location H'FFFF E228>
TA01 Input Capture Register (TA01ICR)	<P4 address: location H'FFFF E22C>
TA02 Input Capture Register (TA02ICR)	<P4 address: location H'FFFF E230>
TA03 Input Capture Register (TA03ICR)	<P4 address: location H'FFFF E234>
TA04 Input Capture Register (TA04ICR)	<P4 address: location H'FFFF E238>
TA10 Input Capture Register (TA10ICR)	<P4 address: location H'FFFF E328>
TA11 Input Capture Register (TA11ICR)	<P4 address: location H'FFFF E32C>
TA12 Input Capture Register (TA12ICR)	<P4 address: location H'FFFF E330>
TA14 Input Capture Register (TA14ICR)	<P4 address: location H'FFFF E338>
TA15 Input Capture Register (TA15ICR)	<P4 address: location H'FFFF E33C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICAik															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICAik															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	ICAik	All 0	R	N	Input Capture Aik These bits hold the 32-bit input capture value.

Legend: i = 0, k = 0 to 5

21.11.7 TAi Free-Running Counter (TAiTCNT)

The TAiTCNT counter counts on the signal output by the prescaler via the clock bus, externally input clock signal.

Timer A is started for counting up by setting the TAE bit in the ATU-IIIS master enable register (ATUENR) to "1". The clock input to the counter is selected by setting the clock select bit (CKSELA) in TAi control register A (TAiCR).

When the TAiTCNT counter overflows (from H'FFFF FFFF to H'0000 0000), the overflow flag A (OVFA) in TAi status register A (TAiSR) is set to "1".

TA0 Free-Running Counter (TA0TCNT) <P4 address: location H'FFFF E220>
TA1 Free-Running Counter (TA1TCNT) <P4 address: location H'FFFF E320>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTA															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTA															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	CNTA	All 0	R	W	Timer Count A These bits hold the counter value in 32-bit units.

21.11.8 TAik Noise Canceler Counter (TAikNCNT)

When the noise canceling function is enabled by setting the noise canceler enable bits (NCEA5 to NCEA0) in TAiI/O control register 2 (TAiIO2), incrementing of the counter begins with signals from the external input pins (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15) as the trigger. Either the noise canceler count clock or clock bus 5 may be selected as the count source by means of the noise canceler clock select bits (NCKA5 to NCKA0) in the TAiIO2 register.

Operation proceeds in premature-transition cancellation mode or minimum time-at-level cancellation mode according to the setting of the NCMA bit in the ATNCMR register of the common controller.

- Premature-Transition Cancellation Mode

  When a level change of the externally input signal (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15) is detected while bits NCEA5 to NCEA0 are set to 1 and NCNTA0 to NCNTA5 are stopped, the TAikNCNT counter is started for counting up. These counters are cleared to "H'00" and stopped on the first edge of the Pck clock after the value in NCNTA0 to NCNTA5 matches the value in TAi noise canceler register (TAikNCR).

  The TAikNCNT counter is incremented regardless of the TAE bits in the ATUENR register.

  The first level change after the start of count operation is output as the signal with the noise canceled and is subject to edge extraction. All subsequent level changes are masked until the counter value matches that of the TAikNCR register, so the noise-cancelled signal does not change. When the counter value matches that of the TAikNCR register, the input signal level at that point is output as the noise-cancelled signal.

  Even if the NCEA bits are cleared to "0" while the counter is operating, the counter continues to operate until its value matches that of the TAikNCR register. The input signal is masked over this period.

- Minimum Time-at-Level Cancellation Mode

  When a level change of the externally input signal (TIA00 to TIA04, TIA10 to TIA12, TIA14, and TIA15) is detected while bits NCEA5 to NCEA0 are set to "1" and the TAikNCNT counter is stopped, the TAikNCNT counter is started for counting up. These counters are cleared to "H'00" and stopped on the first edge of the Pck clock after the value in the TAikNCNT counter matches the value in TAi noise canceler register (TAikNCR) or after the level of the externally input signal is changed.

  The TAikNCNT counter is incremented regardless of the TAE bits in the ATUENR register.

  The noise-cancelled signal changes along with a level change at the start of count operation only when the counter value matches that of the TAikNCR register. If count operation stops before the value of the TAikNCR register is matched, level changes at the start and end of count operation are masked, so the noise-cancelled signal does not change.

  Even if the NCEA bits are cleared to "0" while the counter is operating, the counter does not stop and noise cancelation continues until a compare match occurs or the input signal level changes.

TA00 Noise Canceler Counter (TA00NCNT)	<P4 address: location H'FFFF E210>
TA01 Noise Canceler Counter (TA01NCNT)	<P4 address: location H'FFFF E212>
TA02 Noise Canceler Counter (TA02NCNT)	<P4 address: location H'FFFF E214>
TA03 Noise Canceler Counter (TA03NCNT)	<P4 address: location H'FFFF E216>
TA04 Noise Canceler Counter (TA04NCNT)	<P4 address: location H'FFFF E218>
TA10 Noise Canceler Counter (TA10NCNT)	<P4 address: location H'FFFF E310>
TA11 Noise Canceler Counter (TA11NCNT)	<P4 address: location H'FFFF E312>
TA12 Noise Canceler Counter (TA12NCNT)	<P4 address: location H'FFFF E314>
TA14 Noise Canceler Counter (TA14NCNT)	<P4 address: location H'FFFF E318>
TA15 Noise Canceler Counter (TA15NCNT)	<P4 address: location H'FFFF E31A>

Bit:	7	6	5	4	3	2	1	0
	TAikNCNT							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TAikNCNT	All 0	R	W	TAik Noise Canceler Count These bits hold the counter value in 8-bit units.

Legend: i = 0 or 1, k = 0 to 5

21.11.9 TAik Noise Canceler Register (TAikNCR)

The TAikNCR register sets the upper limitations of the TAik noise canceler counter (TAikNCNT). For example, when the noise canceler is driven by the Pck clock divided by 128 and this register is set to H'FF, a pulse whose width is 0.82 ms (when Pck = 40 MHz) can be treated as noise.
Operation proceeds in premature-transition cancellation mode or minimum time-at-level cancellation mode according to the setting of the NCMA bit in the ATNCMR register of the common controller.

- Premature-transition cancellation mode

  When the TAikNCNT counter is operating, further changes in the level of the input signal are masked. The values of the TAikNCNT counter and the TAikNCR register are compared constantly. When a compare match occurs, the count value of the TAikNCNT counter is cleared in synchronization with the next Pck clock cycle, counter operation stops, and masking of the input signal is canceled.
- Minimum time-at-level cancellation mode

  The ATU-IIIS is in noise cancelation standby status when the TAikNCNT counter is operating. The values of the TAikNCNT counter and the TAikNCR register are compared constantly. When a compare match occurs, the count value of the TAikNCNT counter is cleared in synchronization with the next Pck clock cycle, counter operation stops, and simultaneously the noise canceler outputs the input signal with the noise removed.

TA00 Noise Canceler Register (TA00NCR)	<P4 address: location H'FFFF E211>
TA01 Noise Canceler Register (TA01NCR)	<P4 address: location H'FFFF E213>
TA02 Noise Canceler Register (TA02NCR)	<P4 address: location H'FFFF E215>
TA03 Noise Canceler Register (TA03NCR)	<P4 address: location H'FFFF E217>
TA04 Noise Canceler Register (TA04NCR)	<P4 address: location H'FFFF E219>
TA10 Noise Canceler Register (TA10NCR)	<P4 address: location H'FFFF E311>
TA11 Noise Canceler Register (TA11NCR)	<P4 address: location H'FFFF E313>
TA12 Noise Canceler Register (TA12NCR)	<P4 address: location H'FFFF E315>
TA14 Noise Canceler Register (TA14NCR)	<P4 address: location H'FFFF E319>
TA15 Noise Canceler Register (TA15NCR)	<P4 address: location H'FFFF E31B>

Bit:	7	6	5	4	3	2	1	0
	TAikNCT							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TAikNCT	All 0	R	W	TAik Noise Cancellation Time These bits set a period for TIA noise cancellation (8-bit compare value).

Legend: i = 0 or 1, k = 0 to 5

21.12 Operations of Timer A

21.12.1 Operation of Noise Canceler

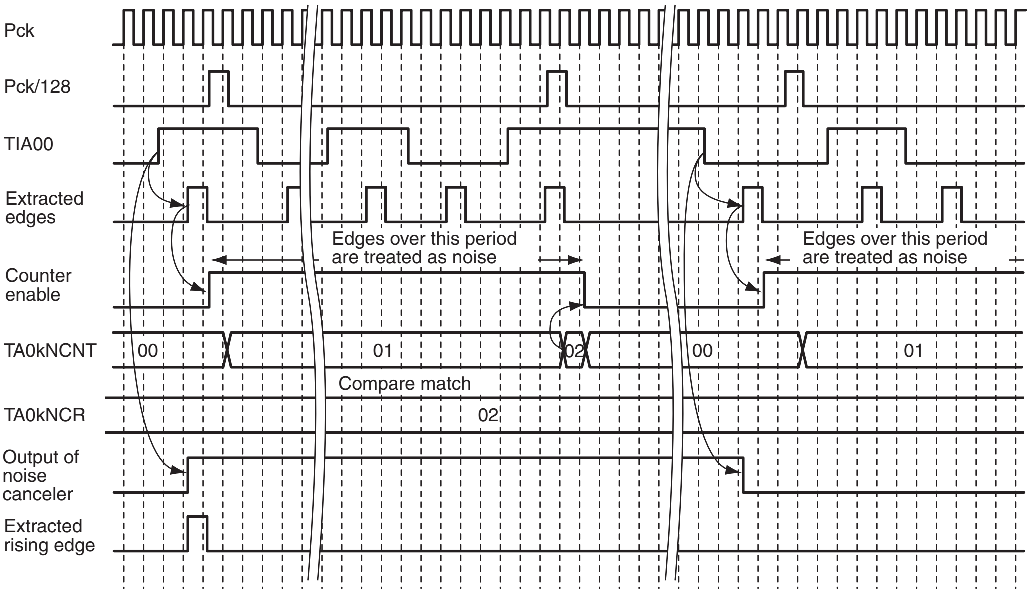
Input edges are processed in premature-transition cancellation or minimum time-at-level cancellation mode depending on the setting in the NCMA bit in the ATU-IIIS noise cancellation mode register (ATNCMR) of the common controller.

Figures 21.8 and 21.10 show examples of noise cancellation in premature-transition cancellation and minimum time-at-level cancellation modes, respectively. In these examples, edges are input on pin TIA00 and the rising edge sensing is selected.

In premature-transition cancellation mode, TAik noise canceler counter (TAikNCNT) is started by the level change of the externally input signal as a trigger. At the same time, the level change is output as the signal after noise removal.

Counting continues until the counter value match the value in TAik noise canceler register (TAikNCR). Level changes within the period are ignored and are not output. When the counter value matches that of the TAikNCR register, the input signal level at that point is output as the noise-cancelled signal. Note that the levels are changed on the compare match when the input levels at the start of counting (after the first level change) and on the compare match differ.

Figure 21.9 shows an example of noise cancellation for two types of waveforms.



Note: • When rising-edge sensing and premature-transition cancellation mode are selected

Figure 21.8 Example of Noise Cancellation in Premature-Transition Cancellation

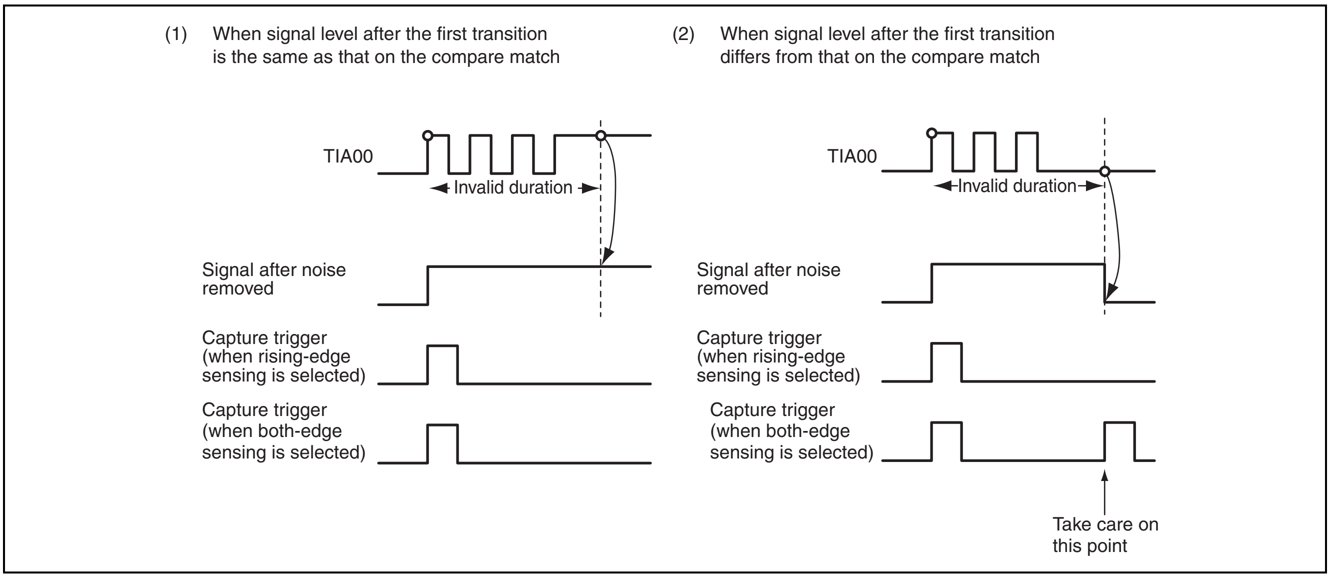
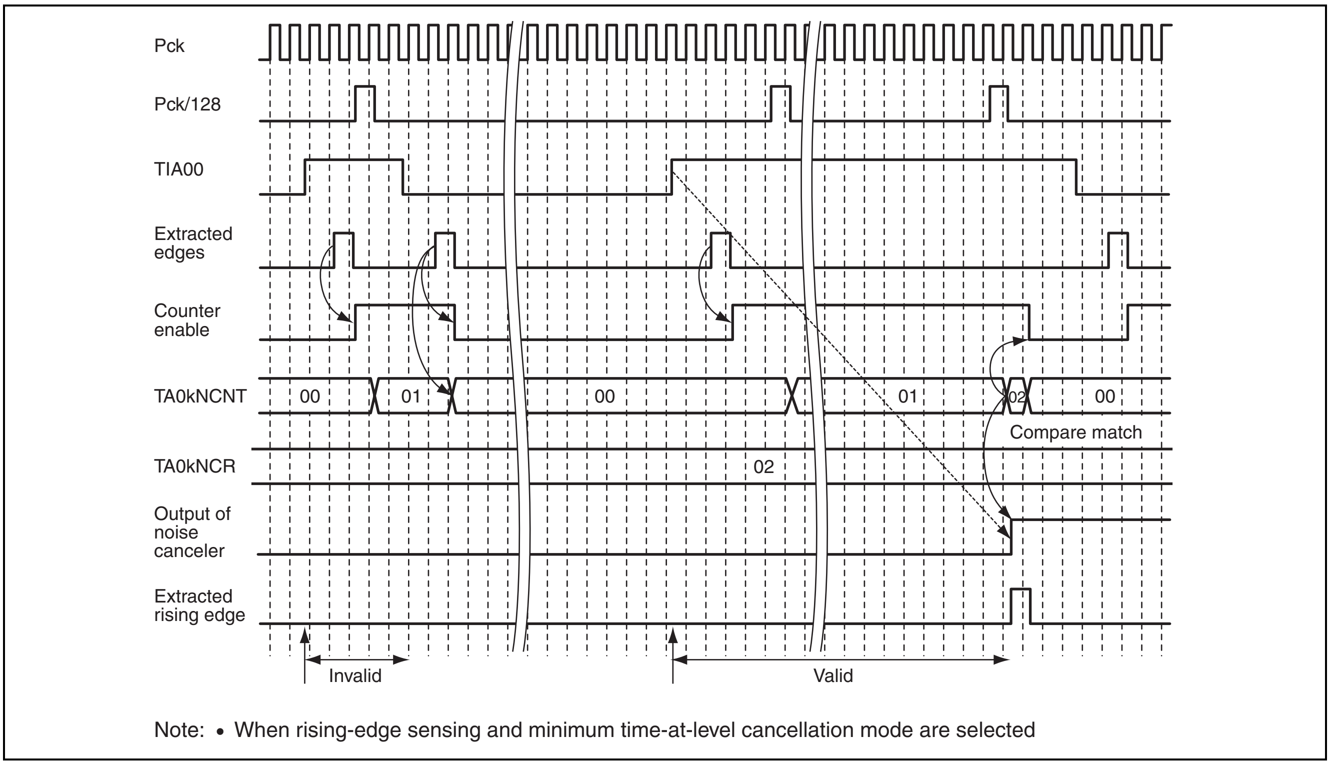

(1) When signal level after the first transition is the same as that on the compare match
(2) When signal level after the first transition differs from that on the compare match
TIA00
Invalid duration
Signal after noise removed
Capture trigger (when rising-edge sensing is selected)
Capture trigger (when both-edge sensing is selected)
Take care on this point

Figure 21.9 Example of Noise Cancellation in Premature-Transition Cancellation for Two Types of Waveforms

In minimum time-at-level cancellation, TAik noise canceler counter (TAikNCNT) is started by the level change of the externally input signal as a trigger. Counting continues until the counter value match the value in TAi noise canceler register (TAikNCR) or the level change of the input signal is detected.

When the values in the counter and TAi inoise canceler register match, the level change at the start of counting is output as the signal after noise removal. If the level of the input signal changes before the count value matches the value set in the TAi noise cancel register (TAikNCR), these level changes as well as any that occur during timer operation are treated as noise and not output as the noise-cancelled signal.



Note: • When rising-edge sensing and minimum time-at-level cancellation mode are selected

Figure 21.10 Example of Noise Cancellation in Minimum Time-at-Level Cancellation

21.12.2 Operation of Free-Running Counter

The TAi Free-running counter (TAiTCNT) is started for counting up by setting the TAE bit in ATU-IIIS master enable register (ATUENR) to "1". When the TAiTCNT counter overflows (from H'FFFF FFFF to H'0000 0000), the OVFA bit in TAi status register (TAiSR) is set to "1". An interrupt request is issued for the CPU when the OVEA bit in TAi interrupt enable register (TAiIER) is set to "1". After overflow, the TAiTCNT counter continues counting up from H'0000 0000.

When the TAE bit in the AUTENR register is cleared to "0", the TAiTCNT counter is stopped but is not cleared. By setting the TAE bit to "1" again, the TAiTCNT counter is resumed from the value when stopped.

The TAiTCNT counter can be written during operation and writing takes priority over counting. After that, the TAiTCNT counter is started from the written value. Regardless of the clock driving the counter, the write access is completed in two cycles of the Pck clock.

The prescalers run regardless of the TAE bit and are not synchronized with the timing at which the TAE bit is set. Therefore, the time from when the TAE bit is set to when the TAiTCNT counter is incremented for the first time is less than the cycle of the clock of the TAiTCNT counter.

Figure 21.11 shows an example of TAi free-running counter (TAiTCNT) operation.

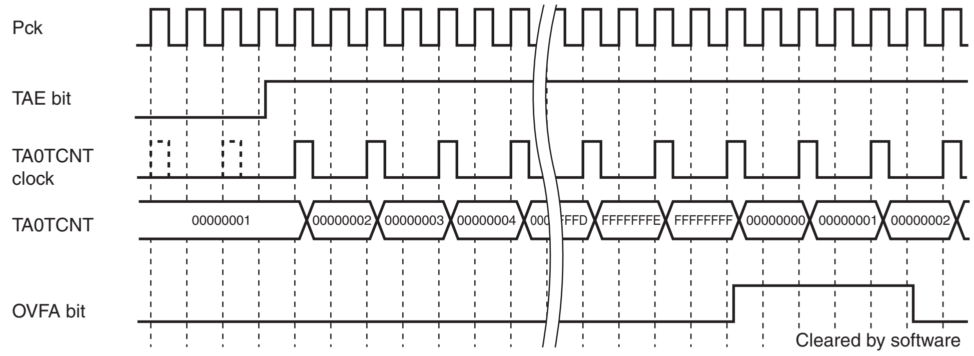


Figure 21.11 Timer A Free-Running Operation: Overflow Timing

21.12.3 Input Capture

The TAik input capture register (TAikICR) performs input capture of edge input from the corresponding external input pins (TIAik) when input capture operation is specified by bits IOA5 to IOA0 in TAiI/O control register 1 (TAiIO1).

Noise on the signals can be removed by the noise cancelers.

The TAi free-running counter (TAiTCNT) begins counting up when the TAE bit in the ATUENR register is set. When an edge is input on the external signal input pin corresponding to the TAikICR register, the matching bit in the TAi status register (TAiSR) is set to "1" and the TAiTCNT counter value is transferred to the TAikICR register. The rising or falling edge, or both edges, of the input can be selected. Interrupt requests can be sent to the CPU by making the appropriate setting in the TAi interrupt enable register (TAiIER). It is also possible to start a DMA transfer by an interrupt request by specifying the DMAC.

When the TAik input capture register (TAikICR) and the TAi free-running counter (TAiTCNT) are written simultaneously, ICRA0 to ICRA5 capture the previous value stored in TCNTA.

Figure 21.12 shows an example of input capture when the edges to be sensed are rising edges for TIA00, falling edges for TIA01, and both edges for TIA02.

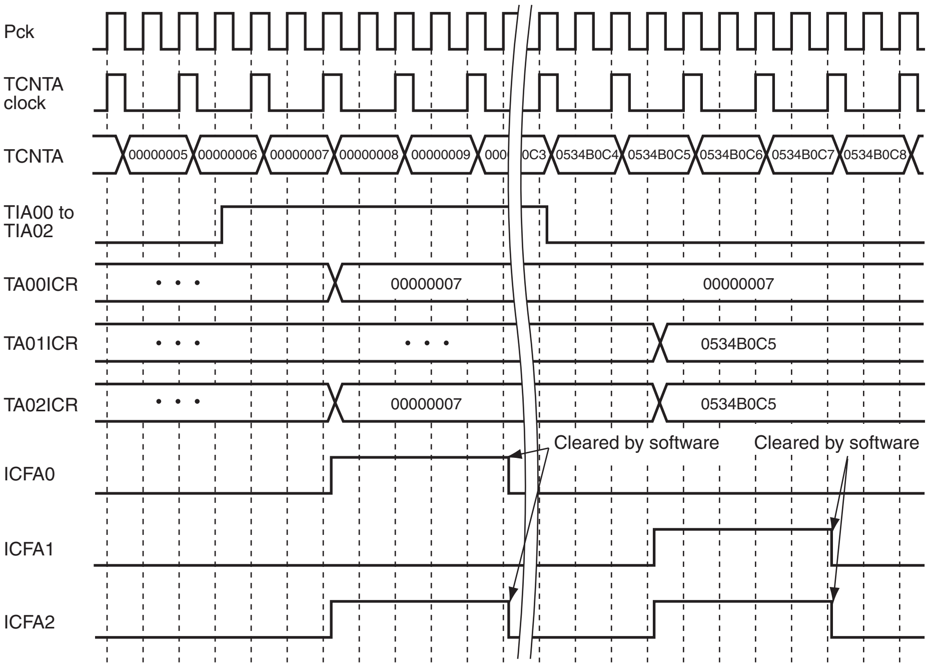


Figure 21.12 Example of Input Capture of Timer A

21.12.4 DMA Transfer

The DMAC can be activated by an input capture interrupt request. The corresponding bit in the TAi status register is cleared to "0" when a DMA transfer request triggered by an input capture interrupt is accepted by the DMAC.

21.13 Overview of Timer F

The timer F consists of three subblocks, featuring functions shown below.

- Edge counting in a specified period

  Counts the number of edges input to the external input pin (TIFjA) in a specified period.
- Valid edge interval counting

  Measures time until a specified number of edges is input to the external pin (TIFjA).
- Measurement of time during "H"/"L" input levels

  Measures a toal amount of time when a "H"or "L" level is input to the external input pin (TIFjA). The duration of measurement is designated as the number of pulses input to the external pin.
- Measurement of PWM input waveform timing

  Measures the off-duty period and cycle time of the PWM waveform input to the external pin (TIFjA). The duration of measurement is designated as the number of PWM cycles input to the external pin.
- Rotation speed/pulse measurement

  Every time an edge is input to the external pin (TIFjA), the following values are retained — edge count, time stamp at edge input, edge input interval (cycle), and "H"/"L" input level immediately before input.
- Up/down event count

  TIFjA of the two external pins (TIFjA, TIFjB) is used to count as the count source. TIFjB switches between upcounting and downcounting.
- Four-time multiplication event count

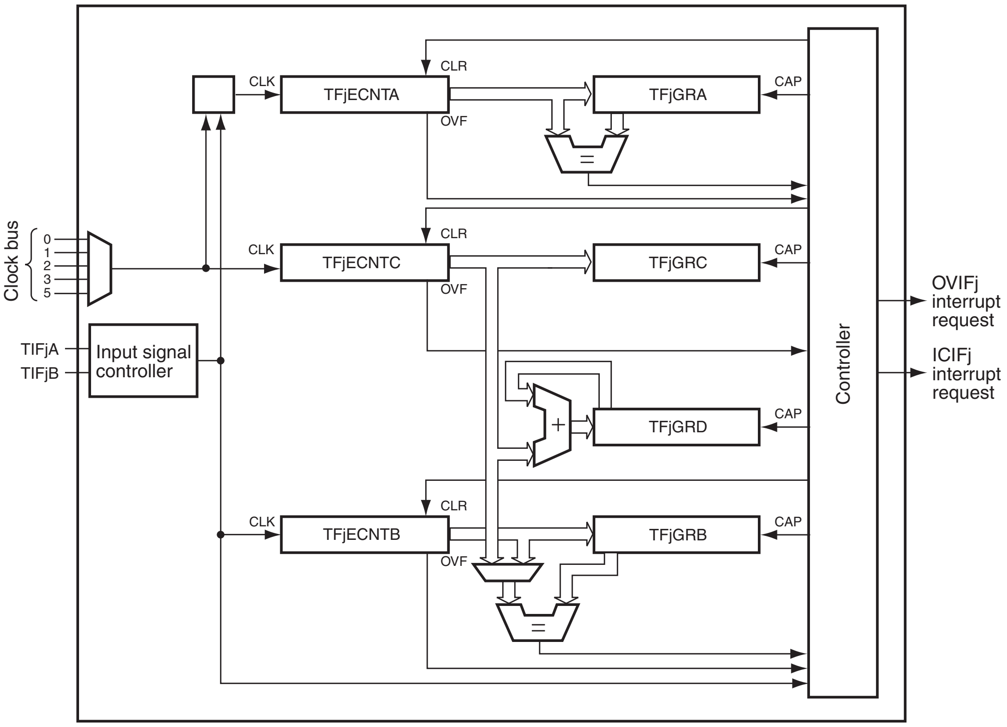
  Counting operation is executed using two external input pins (TIFjA, TIFjB) as the count sources. Signals in the pins switch between upcounting and downcounting.

Input signals from the external input pins TIFjA and TIFjB can be subject to the noise cancellation function using the noise cancellation function.

21.13.1 Block Diagram

The timer F consists of three subblocks. Each subblock consists of such units as two 24-bit time counters (TFjECNTA, TFjECNTC), three 24-bit general registers (TFjGRA, TFjGRC, TFjGRD), 16-bit event counter (TFjENCTB), 16-bit general register (TFjGRB), input processing unit (edge detection, noise canceller), controllers etc.

Figure 21.13 is a block diagram of timer F.



Notes:
- TIFjB is used only in the up or down counting and 4-time multiplication event counting modes. In other modes the TIFjB input does not affect the operation of timer Fj.
- TFjGBD is used only in the rotation speed/pulse measurement mode.
- j = 0 to 2

Legend:

TFjECNTA: Timer counter A	TFjGRB: General register B
TFjECNTB: Event counter	TFjGRC: General register C
TFjECNTC: Timer counter C	TFjGRD: General register D
TFjGRA: General register A	

Figure 21.13 Block Diagram of Subblocks of Timer F

21.13.2 Interrupts

The timer F can output two types of interrupts totaling six interrupts.

- OVIF0 to OVIF2 interrupts

  An interrupt is output when one of the three counters (TFjECNTA, TFjECNTB, TFjECNTC) in the subblock Fj has overflown or underflown (only in TFjECNTB). To which counter the interrupt belongs can be known by referring to the TFj status register F (TFjSR). This request is received by the INTC module and the designated processing is performed.
- ICIF0 to ICIF2 interrupts

  The interrupt is output when a count value capturing in the subblock Fj occurs. This request is received by the DMAC or INTC module. DMA transfer by DMAC enables to transfer captured data obtained by using compare match as a trigger to the on-chip SRAM (IL memory, OL memory, and SHwyRAM) or perform designated processing by interrupts. For details on DMA transfer by DMAC, see section 20, Direct Memory Access Controller (DMAC).

21.14 Description of Timer F Registers

21.14.1 TF Start Register (TFSTR)

The TFSTR register specifies whether to operate or stop each subblock (timer F0 to F2) in the timer F. Count operation is not executed unless TFE bit in ATU-IIIS master enable register (ATUENR) is enabled even if the start bit in timer F is set to enable the count operation.

TF Start Register (TFSTR) <P4 address: location H'FFFF E400>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	STRF2	STRF1	STRF0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	STRF2	0	R	W	Counter Fj Start Bits These bits specify whether to operate or stop two timer counters in subblocks (TFjECNTA, TFjECNTC) and event counter (TFjECNTB). Counter value is retained at stop state. When this bit is set to "1" once again, the operation starts at the retained value. Note that count operation does not start when this bit is set to "1" unless the TFE bit in the ATUENR register is also set to "1". 0: Stop the counting operation of TFjECNTA, TFjECNTB, and TFjECNTC. 1: Enable the counting operation of TFjECNTA, TFjECNTB, and TFjECNTC. Note: • The prescaler is operating regardless of the setting of the counter F start bit, and not initialized at the start of counter. Therefore, during the time between the activation and the start of actual count operation by the above counter, hardware-related uncertainty shorter than the period of selected count source (resolution) accompanies
1	STRF1	0	R	W	
0	STRF0	0	R	W	

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.2 TF Noise Canceller Control Register (TFNCCR)

The TFNCCR register specifies to enable/disable the noise canceller in the subblocks F0 to F2.

TF Noise Canceller Control Register (TFNCCR) <P4 address: location H'FFFF E404>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	NCEF2	NCEF1	NCEF0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2	NCEF2	0	R	W	Noise Canceller Enable Bits (Fj) Specify to enable/disable the noise canceller in each subblock. Regarding the subblocks F0 to F2, each subblock has noise cancellers TIFjA and TIFjB but enabling/disabling these cancellers cannot be specified independently. Setting the NCEFj bit to "1" enables each noise canceller in TIFjA and TIFjB. When the noise canceller function is enabled and the level change in the external input signals (TIFjA, TIFjB) is detected, either subsequent edge cancel mode or preceding edge cancel mode starts according to the setting of ATU-IIIS noise cancellation mode register (ATNCMR) in common controller. In subsequent edge cancel mode, when the input signal level change is detected, the change is output as the signal that has passed through TFj noise canceling. Simultaneously, corresponding noise canceler counters A and B (TFjNCNTA, TFjNCNTB) start upcounting. The input signal level change is masked until a compare match occurs between the value in the noise canceler counter and the values in the TFj noise cancel registers A and B (TFjNCRA, TFjNCRB). When a compare match occurs, the input signal level at this moment is output as the signal after noise canceling. When these bits are cleared to "0" while the TFjNCNTA and TFjNCNTB counters are in count operation, the count operation continues until a compare match occurs and the level change of the corresponding external input (TIFAj, TIFBj) is kept being masked. In preceding edge cancel mode, when the level change of the input signal is detected, the TFjNCNTA and TFjNCNTB counters starts upcounting. If a level change of input signal is not detected during the period until a compare match occurs between the value in the noise canceler counter and the values in the TFjNCRA and TFjNCRB registers, the level change at the compare match is output as the signal after a noise cancellation. If a noise change is detected until a compare match occurs, this change is regarded as noise and the noise canceller does not change the signal after a noise cancel regarding that no level change of input signal occurred. When these bits are cleared to "0" while the TFjNCNTA and TFjNCNTB counters are in count operation, the count operation continues to keep noise canceling processing until a compare match or input signal change occurs. For a operating example in cancel mode, see figures 21.3 and 21.4. 0: Noise cancel function of TIFjA and TIFjB is disabled 1: Noise cancel function of TIFjA and TIFjB is enabled
1	NCEF1	0	R	W	
0	NCEF0	0	R	W	

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.3 TFj Control Register (TFjCR)

The TFjCR register specifies the operation mode of the subblocks F0 to F2.

TF0 Control Register (TF0CR)	<P4 address: location H'FFFF E480>
TF1 Control Register (TF1CR)	<P4 address: location H'FFFF E4A0>
TF2 Control Register (TF2CR)	<P4 address: location H'FFFF E4C0>

Bit:	7	6	5	4	3	2	1	0
	CKSELFj			MDFj			EGSELFj	
After Reset:	0	0	0	0	0	0	0	0

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	CKSELFj	000	R	W	Clock Select Bits (Fj) Specify the signal on one of clock-bus lines 0 to 3 and 5 as the clock sources for the two timer counters A and C (TFjECNTA, TFjECNTC) in the subblocks F0 to F2. The signal on lines 0 to 3 have been frequency-divided by prescalers 0 to 3, respectively. Clock bus 5 supplies externally input clock B (TCLKB). Stop timer F before making or changing the counter-clock selection. 000: Clock-bus line 0 (prescaler 0) 001: Clock-bus line 1 (prescaler 1) 010: Clock-bus line 2 (prescaler 2) 011: Clock-bus line 3 (prescaler 3) 100: Setting prohibited 101: Clock-bus line 5 (TCLKB) 110: Setting prohibited 111: Setting prohibited Note: • The edge of an external input clock is extracted before it is output on a clock bus. When using external input clock B, select the edge to be extracted by setting the CB5EG bit in the ATU-IIIS clock bus control register (ATCBCNT).
4 to 2	MDFj	000	R	W	Timer Operation Mode Bits (Fj) Specify the operation mode for the corresponding subblocks F0 to F2. There are seven modes: up/down event count, four-time multiplication event count, edge counting in a specified period, valid edge interval counting, measurement of time during "H"/"L" input levels, measurement of PWM input waveform timing, and rotation speed/pulse measurement. 000: Edge counting in a specified period 001: Valid edge interval counting 010: Measurement of time during "H"/"L" input levels 011: Setting prohibited 100: Measurement of PWM input waveform timing 101: Rotation speed/pulse measurement 110: Up/down event count 111: Four-time multiplication event count

Bit	Abbreviation	After Reset	R	W	Description
1, 0	EGSELFj	00	R	W	Edge Select Bits (Fj) Specify the edge sense modes for event input (TIFjA) in the subblocks F0 to F2. Edge detection is done for signals that have passed through the noise canceller. Therefore, edge detection is done to the external input (TIFjA, TIFjB) if the noise cancel function is disabled, and to signals after noise cancel if the noise cancel function is enabled. While 'measurement of time during "H"/"L" input levels' is specified, when this bit selects the falling edge, measurement of time during "H" level is specified. When this bit selects the rising edge, measurement of time during "L" level is specified. Do not select both edges. While 'measurement of PWM input waveform timing' and 'rotation speed/pulse measurement' are specified, when this bit selects the rising edge, the period between the two rising edges is regarded as the PWM cycle and the low-level period is regarded as the off-duty period. If the falling edge is selected, the period between the two falling edges is regarded as the PWM cycle and the high-level period is regarded as the off-duty period. Do not select both edges. When 'up/down event count' mode and 'four-time multiplication event count' mode are specified, be sure to designate both the rising and falling edges. If otherwise selected, the operation is not guaranteed (see table 21.16). 00: Edge detection is invalid 01: Rising edge 10: Falling edge 11: Both edges Note: • TIFjB pin is available only when 'up/down event count' and 'four-time multiplication event count' are specified. TIFjB operates always detecting both the rising and falling edges. In other modes, TIFjB does not detect edges.

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.4 TFj Interrupt Enable Register (TFjIER)

The TFjIER register specifies whether to enable or disable the interrupt corresponding to the TFj status register (TFjSR).

TF0 Interrupt Enable Register (TF0IER)	<P4 address: location H'FFFF E481>
TF1 Interrupt Enable Register (TF1IER)	<P4 address: location H'FFFF E4A1>
TF2 Interrupt Enable Register (TF2IER)	<P4 address: location H'FFFF E4C1>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	OVE CFj	OVE BFj	OVE AFj	ICEFj
After Reset:	0	0	0	0	0	0	0	0

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3	OVECFj	0	R	W	Overflow Interrupt Enable Bit CFj*1 Specifies whether to enable or disable the interrupt by OVFCFj to the status corresponding to the overflow of the TFj timer counter C (TFjECNTC) (in ‘measurement of PWM input waveform timing’ mode) or a compare match between the TFjECNTC counter and TFjGRB register (in ‘rotation speed/pulse measurement’ mode) 0: Interrupt by OVFCFj disabled 1: Interrupt by OVFCFj enabled
2	OVEBFj	0	R	W	Overflow Interrupt Enable Bit BFj*1 Specifies whether to enable or disable the interrupt by OVFBFj to the status corresponding to the overflow/underflow of the TFj event counter (TFjECNTB). 0: Interrupt by OVFBFj disabled 1: Interrupt by OVFBFj enabled
1	OVEAFj	0	R	W	Overflow Interrupt Enable Bit AFj*1 Specifies whether to enable or disable the interrupt by OVFAFj to the status corresponding to the overflow of the TFj timer counter A (TFjECNTA). 0: Interrupt by OVFAFj disabled 1: Interrupt by OVFAFj enabled
0	ICEFj	0	R	W	Input Capture Interrupt Enable Bit Fj Specifies whether to enable or disable the interrupt by ICFFj to the status corresponding to the input capture detection in the subblock Fj. 0: Interrupt by ICFFj disabled 1: Interrupt by ICFFj enabled

Note: *1 The overflow of interrupt of the subblock Fj is requested as the logical sum of the interrupts OVFAFj, OVFBFj, and OVFCFj. By referring to TFjSR, which counter generated the interrupt by overflow or underflow can be known.

21.14.5 TFj Status Register (TFjSR)

The TFjSR register indicates overflows in the timer counters A and C, overflow or underflow in the event counter, and input capture occurrence.

These flags are interrupt sources and requests the CPU interrupts if the corresponding bits to the TFj interrupt enable register (TFjIER) enable interrupts.

TF0 Status Register (TF0SR)	<P4 address: location H'FFFF E483>
TF1 Status Register (TF1SR)	<P4 address: location H'FFFF E4A3>
TF2 Status Register (TF2SR)	<P4 address: location H'FFFF E4C3>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	OVFCFj	OVFBFj	OVFAFj	ICFFj
After Reset:	0	0	0	0	0	0	0	0

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3	OVFCFj	0	R	*[1]	Overflow/Compare Match Flag CFj The values in this flag show different states depending on the operation mode. In 'measurement of PWM input waveform timing', the flag shows the overflow of the TFj timer counter C (TFjECNTC). In 'rotation speed/pulse measurement' mode, the flag shows a compare match between the TFjECNTC counter and TFjGRB register. This flag cannot be set to "1" by software. 0: No overflow in the TFjECNTC counter 1: The TFjECNTC counter has overflowed [Condition for clearing to "0"] • After reading OVFCFn = "1", "0" is written to OVFCFj. [Conditions for setting to "1"] • Measurement of PWM input waveform timing mode When the TFjECNTC counter overflowed (H'FFFF FF → H'0000 00) • Rotation speed/pulse measurement mode When values in the TFjECNTC counter and TFjGRB register (with the value zero extended to lower 8 bits) match

Bit	Abbreviation	After Reset	R	W	Description
2	OVFBFj	0	R	*[1]	Overflow Flag BFj By this bit, overflow or underflow of the TFj event counter (TFjECNTB) can be monitored. This flag cannot be set to "1" by software. 0: No overflow or underflow in the TFjECNTB counter 1: The TFjECNTB counter has overflowed or underflowed [Condition for clearing to "0"] • After reading OVFBFj = "1", "0" is written to OVFBFj. [Condition for setting to "1"] • When the TFjECNTB counter overflowed (H'FFFF → H'0000) or underflowed (H'0000 → H'FFFF)
1	OVFAFj	0	R	*[1]	Overflow Flag AFj By this bit, overflow of the TFj timer counter A (TFjCNTA) can be monitored. This flag cannot be set to "1" by software. 0: No overflow in the TFjECNTA counter 1: The TFjECNTA counter has overflowed [Condition for clearing to "0"] • After reading OVFAFj = "1", "0" is written to OVFAFj. [Condition for setting to "1"] • When the TFjECNTA counter overflowed (H'FFFF FF → H'0000 00)
0	ICFFj	0	R	*[1]	Input capture Flag Fj By this bit, the detection state of input capture in the subblock Fj can be monitored. This flag cannot be set to "1" by software. 0: Input capture is not detected in the subblock Fj. 1: Input capture is detected in the subblock Fj. [Conditions for clearing to "0"] • After reading ICFFj = "1", "0" is written to ICFFj. • When the TFj capture output register (TFjCDR) are read by DMAC access. [Condition for setting to "1"] • When input capture is detected in the subblock Fj.

Note: *1 To clear the flag, only writing "0" after reading "1" is possible. Writing "1" is invalid.

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.6 TFj Timer Counter A (TFjECNTA)

The TFjECNTA counter, with one provided to each subblock, executes upcount operation using the input clock. One clock bus from clock buses 0 to 5 can be selected as the input clock according to the setting of the corresponding control register. The input clocks for the TFjECNTA and TFjECNTC couneters are the same. Clock source cannot be set independently.

When clearing the counter is done at the countup timing, the TFjECNTA counter is cleared to H'0000 0100, and to H'0000 0000 in other cases.

TF0 Timer Counter A (TF0ECNTA)	<P4 address: location H'FFFF E484>
TF1 Timer Counter A (TF1ECNTA)	<P4 address: location H'FFFF E4A4>
TF2 Timer Counter A (TF2ECNTA)	<P4 address: location H'FFFF E4C4>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TFjECNTA															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjECNTA								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 8	TFjECNTA	All 0	R	W	TFj Time Count Upcounter A
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively.

21.14.7 TFj Event Counter (TFjECNTB)

The TFjECNTB counter, with one provided to each subblock, executes upcount/downcount operation using the input clock. The input clock is given two external input pins (TIFjA, TIFjB). The external pin and edge used to count differs according to the setting of the corresponding control register (operation mode and edge select). The input clock in each mode is listed in table 21.6.

When clearing the counter is done at the count-up timing, the TFjECNTB counter is cleared to "H'0001" and to "H'0000" in other cases.

TF0 Event Counter (TF0ECNTB)	<P4 address: location H'FFFF E488>
TF1 Event Counter (TF1ECNTB)	<P4 address: location H'FFFF E4A8>
TF2 Event Counter (TF2ECNTB)	<P4 address: location H'FFFF E4C8>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjECNTB															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	TFjECNTB	All 0	R	W	TFj Event Count Up/down event counter

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

Table 21.6 Event Counter Input Clock and Count Edge for Each Timer F Operation Mode

Operation Mode	Input Clock	Count Edge
Edge counting in a specified period	TIFjA	Selectable by EGSELFj bit in the TFjCR register
Valid edge interval counting	TIFjA	Selectable by EGSELFj bit in the TFjCR register
Measurement of time during "H"/"L" input levels	TIFjA	Selectable by EGSELFj bit in the TFjCR register (other than both edges)
Measurement of PWM input waveform timing	TIFjA	Selectable by EGSELFj bit in the TFjCR register (other than both edges)
Rotation speed/pulse measurement	TIFjA	Selectable by EGSELFj bit in the TFjCR register (other than both edges)
Up/down event count	TIFjA (Count direction is specified by TIFjB level)	Both rising/falling edges
Four-time multiplication event count	TIFjA, TIFjB	Both rising/falling edges

21.14.8 TFj Timer Counter C (TFjECNTC)

The TFjECNTC counter, with one provided to each subblock, is enabled only in ‘measurement of PWM input waveform timing’ and ‘rotation speed/pulse measurement’ modes. This counter does not execute count operation in other modes. This counter executes upcount operation using the input clock. One clock bus from clock buses 0 to 5 can be selected as the input clock according to the setting of the corresponding control register. The input clocks for the TFjECNTA and TFjECNTC counters are the same. Clock source cannot be set independently.

When clearing the counter is done at the external input timing or triggered by the TFjECNTB counter, the TFjECNTC counter is cleared synchronized with TFjECNTC count clock. At this moment, the TFjECNTC counter is cleared to H'0000 0100.

TF0 Time Counter C (TF0ECNTC) <P4 address: location H'FFFF E48C>
TF1 Time Counter C (TF1ECNTC) <P4 address: location H'FFFF E4AC>
TF2 Time Counter C (TF2ECNTC) <P4 address: location H'FFFF E4CC>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TFjECNTC															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjECNTC								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 8	TFjECNTC	All 0	R	W	TFj Time Count Upcounter C
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.9 TFj General Register A (TFjGRA)

The TFjGRA register, with one provided to each subblock, has two functions such as input capture register and output compare register for the TFj timer counter A (TFjECNTA).

Do not set TFjGRA register to H'0000 0000 to function this register as the output compare register. Note that if H'0000 0000 is set, incorrect measurement may occur.

TF0 General Register A (TF0GRA)	<P4 address: location H'FFFF E490>
TF1 General Register A (TF1GRA)	<P4 address: location H'FFFF E4B0>
TF2 General Register A (TF2GRA)	<P4 address: location H'FFFF E4D0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TFjGRA															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjGRA								—	—	—	—	—	—	—	—
After Reset:	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

<After Reset: H'FFFF FF00>

Bit	Abbreviation	After Reset	R	W	Description
31 to 8	TFjGRA	All 1	R	W	TFj General Registers A Input capture value or output compare match value for the timer counter A.
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.10 TFj General Register B (TFjGRB)

The TFjGRB register, with one provided to each subblock, has two functions such as input capture register and output compare register for the TFj event counter (TFjECNTB).

Do not set the TFjGRB register to H'0000 to function this register as the output compare register. Note that if H'0000 is set, incorrect measurement may occur.

TF0 General Register B (TF0GRB)	<P4 address: location H'FFFF E48A>
TF1 General Register B (TF1GRB)	<P4 address: location H'FFFF E4AA>
TF2 General Register B (TF2GRB)	<P4 address: location H'FFFF E4CA>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjGRB															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

<After Reset: H'FFFF>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	TFjGRB	All 1	R	W	TFj General Registers B Input capture value or output compare match value for event counter.

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.11 TFj General Register C (TFjGRC)

The TFjGRC register, with one provided to each subblock, has a function as the input capture register for the TFj timer counter C (TFjECNTC). Triggered by a compare match between the TFjECNTB counter and the TFjGRB register (in ‘measurement of PWM input waveform timing’ mode) or edge input of the TIFjA pin (in ‘rotation speed/pulse measurement’ mode), TFjECNTC count number is taken in at the next TFjECNTC upcount timing. These registers are valid only in ‘measurement of PWM input waveform timing’ or ‘rotation speed/pulse measurement’ mode. Capture operation is not executed in other modes.

TF0 General Register C (TF0GRC)	<P4 address: location H'FFFF E498>
TF1 General Register C (TF1GRC)	<P4 address: location H'FFFF E4B8>
TF2 General Register C (TF2GRC)	<P4 address: location H'FFFF E4D8>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TFjGRC															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjGRC								—	—	—	—	—	—	—	—
After Reset:	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

<After Reset: H'FFFF FF00>

Bit	Abbreviation	After Reset	R	W	Description
31 to 8	TFjGRC	All 1	R	W	TFj General Registers C Input capture value for the timer counter C
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.12 TFj General Register D (TFjGRD)

The TFjGRD register is provided to subblocks F0 to F2. Triggered by edge input of the TIFj pin, accumulated number in the TFj timer counter A (TFjECNTA) is taken in at the next TFjECNTC upcount timing. This register is valid only in ‘rotation speed/pulse measurement’ mode. Capture operation is not executed in other modes.

TF0 General Register D (TF0GRD)	<P4 address: location H'FFFF E49C>
TF1 General Register D (TF1GRD)	<P4 address: location H'FFFF E4BC>
TF2 General Register D (TF2GRD)	<P4 address: location H'FFFF E4DC>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TFjGRD															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjGRD								—	—	—	—	—	—	—	—
After Reset:	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

<After Reset: H'FFFF FF00>

Bit	Abbreviation	After Reset	R	W	Description
31 to 8	TFjGRD	All 1	R	W	TFj General Registers D Input capture value for the timer counter A
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.13 TFj Capture Output Register (TFjCDR)

The TFjCDR register is provided to each subblock. When this register is read, values in the TFjGRA, TFjGRB registers, or the TFjECNTB counter is read according to the operation mode. A 16-bit value in the TFjGRB register is read from the upper 16 bits in the TFjCDR bits. In this case, the lower eight bits in the TFjCDR bits are read as 0.

TF0 Capture Output Register (TF0CDR)	<P4 address: location H'FFFF E494>
TF1 Capture Output Register (TF1CDR)	<P4 address: location H'FFFF E4B4>
TF2 Capture Output Register (TF2CDR)	<P4 address: location H'FFFF E4D4>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TFjCDR															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFjCDR								—	—	—	—	—	—	—	—
After Reset:	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

<After Reset: H'FFFF FF00>

Bit	Abbreviation	After Reset	R	W	Description
31 to 8	TFjCDR	H'FFFFFF	R	—	TFj Capture Output Register Data stored in the TFjGRA register or the TFjGRB register is read according to the operation mode. Registers corresponding to various modes are listed below. Writing to these registers are ignored. Edge counting in a specified period mode: TFjGRB register Valid edge interval counting mode: TFjGRA register Measurement of time during high input levels mode: TFjGRA register Measurement of time during low input levels mode: TFjGRA register Measurement of PWM input waveform timing mode: TFjGRA register Rotation speed/pulse measurement mode: ECNTB counter Up/down event count mode mode: TFjGRB register Four-time multiplication event count mode: TFjGRB register
7 to 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.14 TFj Noise Canceler Counter A (TFjNCNTA)

When the TF noise canceler control register (TFNCCR) enables the noise canceler function, these registers start upcount operation using the count clock for noise canceler supplied by the pre-scaler, triggered by the level change in the external input pin (TIFjA).

Operation proceeds in premature-transition cancellation mode or minimum time-at-level cancellation mode according to the setting of the timer F noise cancel mode bit (NCMF) in the ATU-IIIS noise cancellation mode register (ATNCMR) of the common controller.

- Premature-transition cancellation mode

  The TFjNCNTA counter starts upcount operation triggered by the level change of input signal in TIFjA under the condition that the NCEFj bit is set to "1" and the TFjNCNTA counter is not in count operation. When the count number matches the value in the TFj noise cancel register A (TFjNCRA), this register stops the count operation, clearing the count value to "H'00" synchronizing with the next Pck clock. the TFjNCNTA counter executes the count operation regardless of the setting of the TFE bit in the ATU-IIIS master enable register (ATUENR).

  A level change at the start of count operation is output as it is as the signal that passed through the noise canceling operation. Although this signal is the subject of edge detection, the signal does not change because any input level change is masked until the count number matches the value of the TFjNCRA register.

  Even if the NCEFj bit is cleared during the count operation, the count operation continues until the count number matches the value of the TFjNCRA register. The input signal is masked during all that time.

- Minimum time-at-level cancellation mode

  The TFjNCNTA counter starts upcount operation triggered by the level change of input signal in TIFjA under the condition that the NCEFj bit is set to "1" and the TFjNCNTA counter is not in count operation. When the level change of the input signal occurs during the count operation or the count number matches the value in the TFj noise cancel register A (TFjNCRA), this register stops the count operation, clearing the count value to "H'00" synchronizing with the next Pck clock. The TFjNCNTA counter executes the count operation regardless of the setting of the TFE bit in the ATU-III master enable register (ATUENR).

  Signals that passed through the noise canceling operation can change only when the count number matches the value of the TFjNCRA register according to the level change at the count strart. If the count operation stops before the count number matches the value of the TFjNCRA register, signals that passed through the noise canceling operation do not change because the level changes at the start or stop of counting are masked.

  Even if the NCEFj bit is cleared during the count operation, the count operation and noise canceling processing continue until a compare match occurs or the level of the input signal changes.

TF0 Noise Canceler Counter A (TF0NCNTA)	<P4 address: location H'FFFF E410>
TF1 Noise Canceler Counter A (TF1NCNTA)	<P4 address: location H'FFFF E412>
TF2 Noise Canceler Counter A (TF2NCNTA)	<P4 address: location H'FFFF E414>

Bit:	7	6	5	4	3	2	1	0
	TFjNCNTA							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TFjNCNTA	All 0	R	W	TFj Noise Cancel Count A 8-bit count value

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.15 TFj Noise Canceler Counter B (TFjNCNTB)

The TFjNCNTB counter is available only in ‘up/down event count’ mode and ‘4-time multiplication event count’ mode.

When the TF noise canceler control register (TFNCCR) enables the noise canceler function, these registers start upcount operation using the count clock for noise canceler supplied by the pre-scaler, triggered by the level change in the external input pin (TIFjB).

Two types of operation modes — subsequent edge cancel mode and preceding rdge cancel mode — can be set according to the setting of the timer F niose cancel mode bit (NCMF) in the ATU-IIIS noise cancelltion mode register (ATNCMR) in the common controller.

- Premature-transition cancellation mode

  The TFjNCNTB counter starts upcount operation triggered by the level change of input signal in TIFjB under the condition that the NCEFj bit is set to "1" and the TFjNCNTB counter is not in count operation. When the count number matches the value in the TFj noise cancel register B (TFjNCRB), this register stops the count operation, clearing the count value to "H'00" synchronizing with the next Pck clock.

  The TFjNCNTB counter executes the count operation regardless of the setting of the TFE bit in the ATU-IIIS master enable register (ATUENR).

  A level change at the start of count operation is output as it is as the signal that passed through the noise canceling operation. Although this signal is the subject of edge detection, the signal does not change because any input level change is masked until the count number matches the value of the TFjNCRFB register.

  Even if the NCEFj bit is cleared during the count operation, the count operation continues until the count number matches the value of the TFjNCRFB register. The input signal is masked during all that time.
- Minimum time-at-level cancellation mode

  The TFjNCNTB counter starts upcount operation triggered by the level change of input signal in TIFjB under the condition that the NCEFj bit is set to "1" and the TFjNCNTB counter is not in count operation. When the level change of the input signal occurs during the count operation or the count number matches the value in the TFj noise cancel register B (TFjNCRB), this register stops the count operation, clearing the count value to "H'00" synchronizing with the next Pck clock.

  The TFjNCNTB counter executes the count operation regardless of the setting of the TFE bit in the ATU-III master enable register (ATUENR).

  Signals that passed through the noise canceling operation can change only when the count number matches the value of the TFjNCRB register, according to the level change at the count strart. If the count operation stops before the count number matches the value of the TFjNCRB register, signals that passed through the noise canceling operation do not change because the level changes at the start or stop of counting are masked.

  Even if the NCEFj bit is cleared during the count operation, the count operation and noise canceling processing continue until a compare match occurs or the level of the input signal changes.

TF0 Noise Canceler Counter B (TF0NCNTB) <P4 address: location H'FFFF E450>
TF1 Noise Canceler Counter B (TF1NCNTB) <P4 address: location H'FFFF E452>
TF2 Noise Canceler Counter B (TF2NCNTB) <P4 address: location H'FFFF E454>

Bit:	7	6	5	4	3	2	1	0
	TFjNCNTB							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TFjNCNTB	All 0	R	W	TFj Noise Cancel Count B 8-bit count value

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.16 TFj Noise Cancel Register A (TFjNCRA)

The TFjNCRA register sets the upper limit of the TFj noise canceler counter A (TFjNCNTA). Noise in the period up to 0.82 ms (when Pck = 40MHz) can be cancelled by setting the register to H'FF.

Two types of operation modes — subsequent edge cancel mode and preceding rdge cancel mode — can be set according to the setting of the timer F niose cancel mode bit (NCMF) in the ATU-IIIS noise cancelltion mode register (ATNCMR) in the common controller.

- Premature-transition cancellation mode

  While the TFjNCNTA counter is in count operation, the level change of the subsequent input signal is masked. Values in the TFjNCNTA counter and the TFjNCRA register are always compared. If a compare match occurs, these registers clear the value in the TFjNCNTA counter synchronizing with the next Pck clock, stop the count operation, and cancel the masking of the input signal.
- Minimum time-at-level cancellation mode

  While the TFjNCNTA counter is in count operation, noise canceler processing waiting state is entered. Values in the TFjNCNTA counter and the TFjNCRA register are always compared. If a compare match occurs, these registers clear the value in the TFjNCNTA counter synchronizing with the next Pck clock, stop the count operation, and then cancel the masking of the input signal and the noise canceler outputs the input signal that has passed through the noise canceling processing.

TF0 Noise Cancel Register A (TF0NCRA)	<P4 address: location H'FFFF E411>
TF1 Noise Cancel Register A (TF1NCRA)	<P4 address: location H'FFFF E413>
TF2 Noise Cancel Register A (TF2NCRA)	<P4 address: location H'FFFF E415>

Bit:	7	6	5	4	3	2	1	0
	TFjNCRA							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TFjNCRA	All 0	R	W	TFj Noise Cancel Time A TIFjA noise cancel period (8-bit compare value)

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.14.17 TFj Noise Cancel Register B (TFjNCRB)

The TFjNCRB register sets the upper limit of the TFj noise canceler counter B (TFjNCNTB). Noise in the period up to 0.82 ms (when Pck = 40MHz) can be cancelled by setting the register to H'FF.

Operation proceeds in premature-transition cancellation mode or minimum time-at-level cancellation mode according to the setting of the timer F noise cancel mode bit (NCMF) in the ATU-IIIS noise cancellation mode register (ATNCMR) of the common controller.

- Premature-transition cancellation mode

  While the TFjNCNTB counter is in count operation, the level change of the subsequent input signal is masked. Values in the TFjNCNTB counter and the TFjNCRB register are always compared. If a compare match occurs, these registers clear the value in the TFjNCNTB counter synchronizing with the next Pck clock, stop the count operation, and cancel the masking of the input signal.
- Minimum time-at-level cancellation mode

  While the TFjNCNTB counter is in count operation, noise canceler processing waiting state is entered. Values in the TFjNCNTB counter and the TFjNCRB register are always compared. If a compare match occurs, these registers clear the value in the TFjNCNTB counter synchronizing with the next Pck clock, stop the count operation. Simultaneously the noise canceler outputs the input signal that has passed through noise canceling processing.

TF0 Noise Cancel Register B (TF0NCRB) <P4 address: location H'FFFF E451>
TF1 Noise Cancel Register B (TF1NCRB) <P4 address: location H'FFFF E453>
TF2 Noise Cancel Register B (TF2NCRB) <P4 address: location H'FFFF E455>

Bit:	7	6	5	4	3	2	1	0
	TFjNCRB							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TFjNCRB	All 0	R	W	TFj Noise Cancel Time B TIFjB noise cancel period (8-bit compare value)

Legend: j = 0 to 2, corresponding to subclocks F0 to F2, respectively

21.15 Operations of Timer F

21.15.1 Fixed-Period Edge Counting

When a period over which edges are counted is set in the TFjGRA register, the number of edges within the period is obtained in the TFjGRB regsiter. When no edge is detected within the period, 0 is set to the TFjGRB register. The period set to count is eqivqlent to the cycle of the TFjECNTA counter clock (the TFjGRA register value). Operation of the timer Fj is described below. Figure 21.14 shows an operation example. In this example, eight edges are input to 12 cycles of the count source clock. Here the TFjECNTA and TFjECNTB clocks indicate the timing of count and clear operation by the TFjECNTA and TFjECNTB counters, respectively.

The operation of registers in fixed-period edge counting mode is described below.

- TFjECNTA: Measures time using one of the clock buses 0 to 5. When a compare match is detected, the count value is cleared synchronized with the next Pck clock.
- TFjECNTB: Counts edges of the signals provided from TIFjA input. Edge types subject to count can be selected from among rising, falling, or both edges. The example given here counts the falling edges. A delay of two cycles in TIFjA occurs because of synchronization processing. When a compare match in the TFjECNTA counter is detected, the count number is cleared synchronized with the next Pck clock. In a case where edges subject to count are given simultaneously at a count clearing by a compare match, both operations are regarded to be done in one cycle, setting the count value to H'0001. Figure 21.15 shows an example of this.
- TFjGRA: Functions as the compare match register for the TFjECNTA counter. A compare match is detected when the count values in the TFjECNTA counter and TFjGRA agree.
- TFjGRB: Functions as the capture register for the TFjECNTB counter. When a compare match in the TFjECNTA counter is detected, this register captures the TFjECNTB counter count number synchronizing with the next Pck clock.
- ICFFj flag: After detecting a compare match in the TFjECNTA counter, sets the ICFFj flag synchronized with the next Pck clock.
- TFjECNTC, TFjGRC, TFjGRD: Do not function.

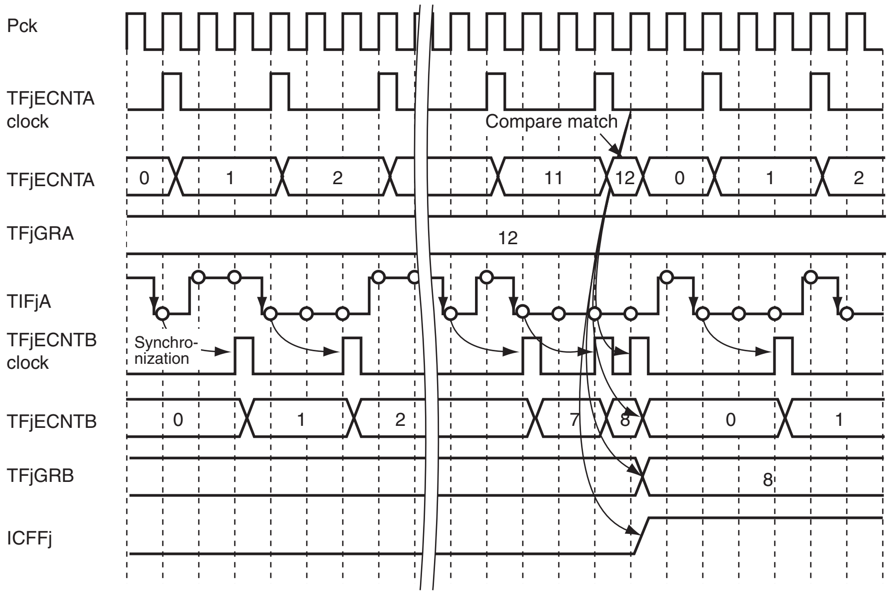


Figure 21.14 Operation Example of Edge Count in Given Time

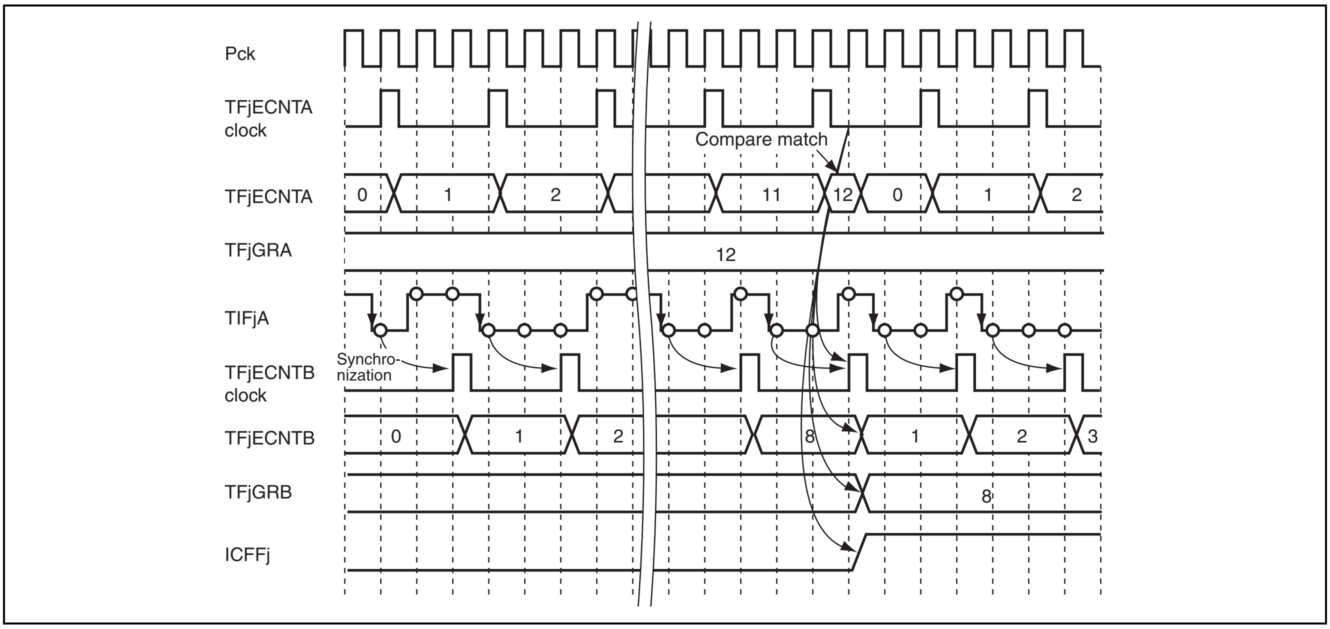


Figure 21.15 Operation Example of Edge Count in Given Time (Compare Match and Event Occur Simultaneously)

21.15.2 Valid Edge Interval Counting

When a number of edges are set in the TFjGRB register, the time necessary to count these edges is notified to the TFjGRA register. The average of input edge intervals is obtained by dividing the time by the number of edges. The outcome is given as the period of the TFjECNTA counter count source clock (TFjGRA register value). Operation of the timer Fj is described below. Figure 21.16 shows an operation example. In this example, 13 cycles of the counter clock are needed to detect 12 input edges. Here the TFjECNTA and TFjECNTB clocks indicate the timing of count and clear operation by the TFjECNTA and TFjECNTB counters, respectively.

Operation in valid edge interval counting mode is described below.

- TFjECNTA: Measures time using one of the clock buses 0 to 5. When a compare match between the TFjECNTB counter and the TFjGRB register is detected, the count value is cleared synchronized with the next TFjECNTA clock. Since TFjECNTA count clear occurs at the same time with countup, the cleared value becomes H'0000 0100.
- TFjECNTB: Counts edges provided from TIFjA. Edge types subject to count can be selected from among rising, falling, or both edges. The example given here counts the falling edges. A delay of two cycles in TIFjA occurs because of synchronization processing. When a compare match is detected, the count number is cleared synchronized with the next Pck clock.
- TFjGRA: Functions as the capture register for the TFjECNTA counter. When a compare match in the TFjECNTB counter is detected, this register captures the TFjECNTA counter count number synchronizing with the next TFjECNTA clock.
- TFjGRB: Functions as the compare match register for the TFjECNTB counter. A compare match is detected when the count values in the TFjECNTB counter and the TFjGRB register match.
- ICFFj flag: After detecting a compare match in the TFjECNTB counter, sets the ICFFj flag synchronized with the next TFjECNTA clock.
- TFjECNTC, TFjGRC, TFjGRD: Do not function.

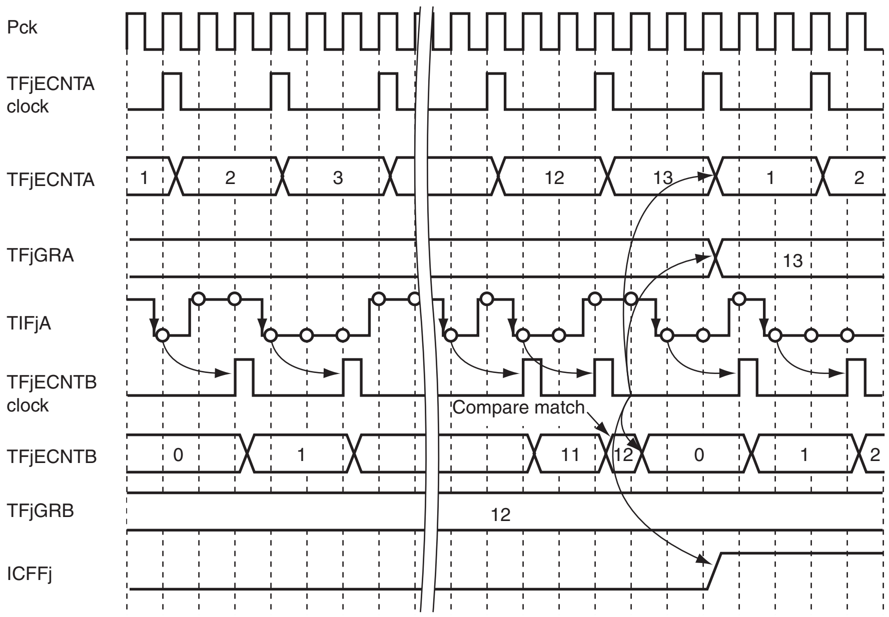


Figure 21.16 Operation Example of Valid Edge Interval Counting

21.15.3 Measurement of Time during "H"/"L" Input Levels

Measures the time while TIFjA is driven "H" or "L". The time obtained is indicated using the TFjECNTA counter clock source as the standard. The width of the measument time is specified to the TFjGRB register in the form of the pulse number provided for TIFjA (TFjGRB register value). Operation of the timer F is described below. Figure 21.17 shows an operation example. This is the example in which the "H" level periods of the three pulses are measured as nine count source cycles. Here the TFjECNTA and TFjECNTB clocks indicate the timing of count and clear operation by the TFjECNTA and TFjECNTB counters, respectively.

Operation of registers in ‘counting "H" or "L" level of input’ mode is shown below.

- TFjECNTA: Executes upcount using one of the clock buses 0 to 5 as a count source and TIFjA level as enable. Therefore, the time period in which TIFjA is in "H" level is measured. After detecting a compare match in the TFjECNTB counter, this register clears the count number synchronizng with the next count source clock. If TIFjA is driven "H" level at clearing count by the compare match, the count value becomes H'0000 0100. Figure 21.18 is an example of this.
- TFjECNTB: Counts the falling edge of TIFjA. A delay of two cycles occurs because of synchronization processing. When a compare match is detected, the count number is cleared synchronized with the next Pck clock.
- TFjGRA: Functions as the capture register for the TFjECNTA counter. When a compare match in the TFjECNTB counter is detected, this register captures the TFjECNTA count number synchronizing with the next TFjECNTA clock.
- TFjGRB: Functions as the compare match register for the TFjECNTB counter. A compare match is detected when the count number in the TFjECNTB counter and TFjGRB match.
- ICFFj flag: After detecting a compare match in the TFjECNTB coutner, sets the ICFFj flag synchronized with the next TFjECNTA clock.
- TFjECNTC, TFjGRC, TFjGRD: Do not function.

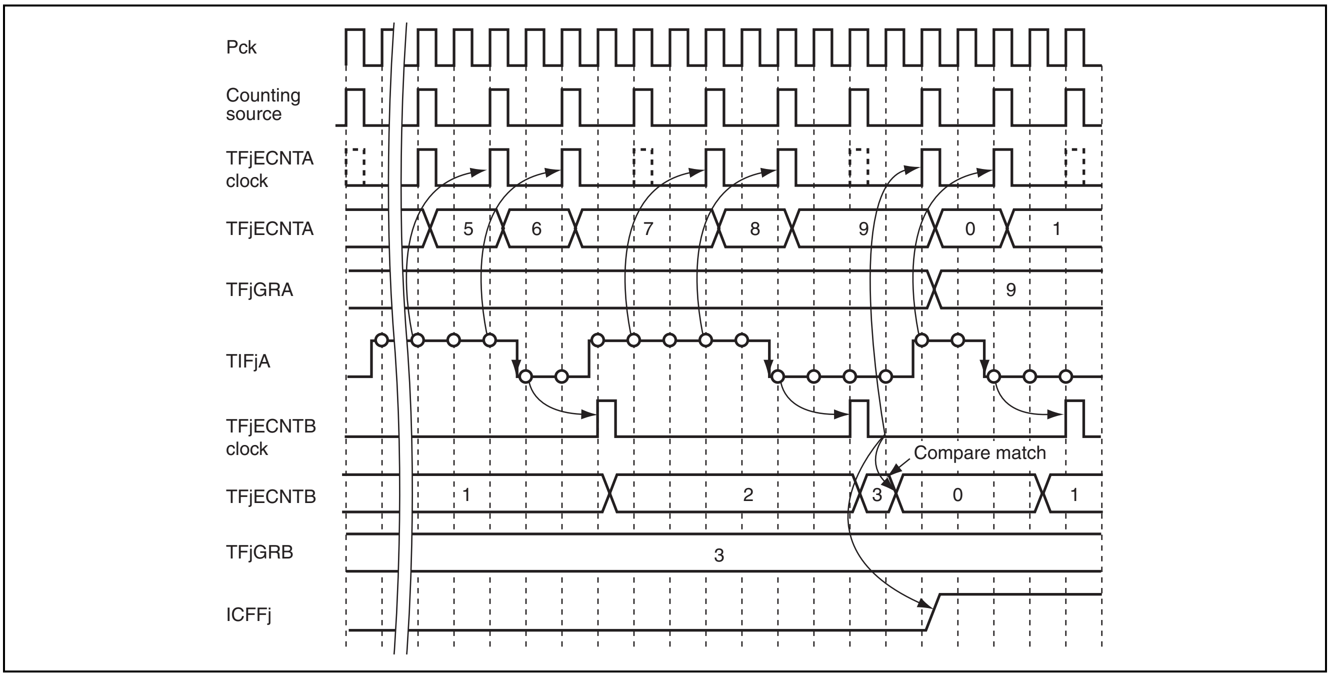


Figure 21.17 Operation Example of Measurement of Time during "H" Input Levels

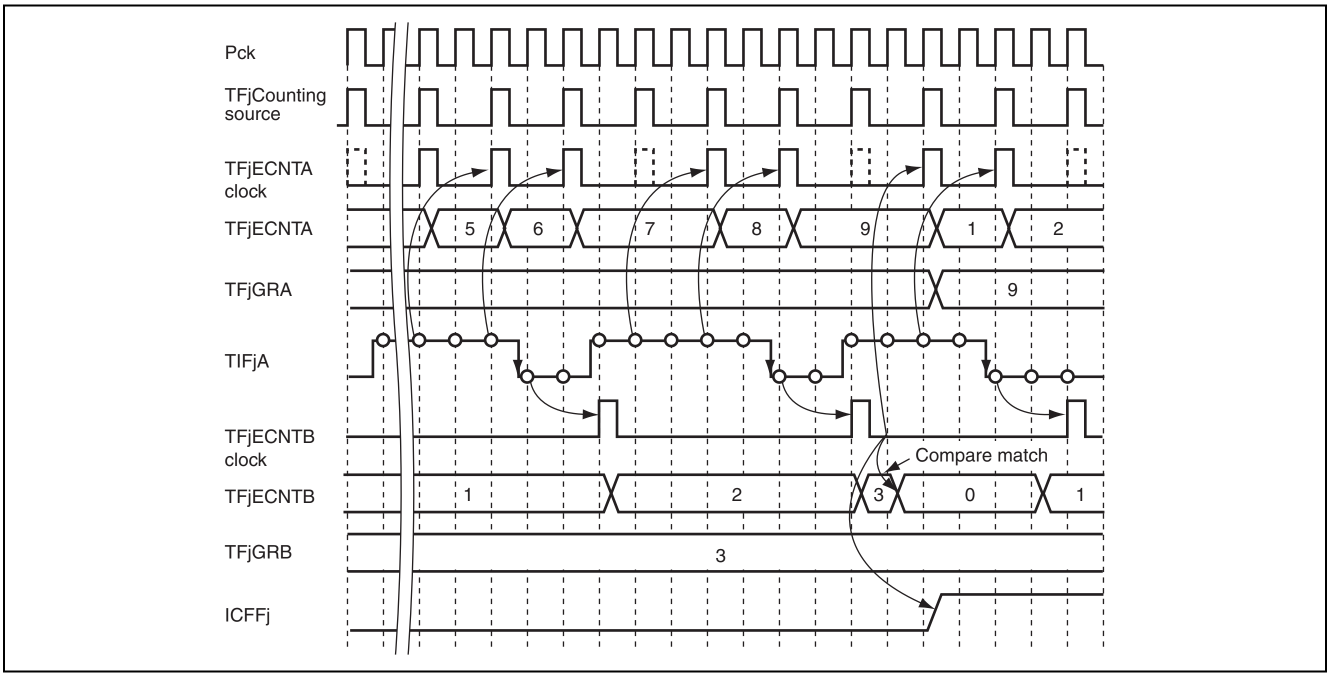


Figure 21.18 Operation Example of Measurement of Time during "H" Input Levels (TIFjA is in "H" Level When Capture is in Operation)

21.15.4 Measurement of PWM Input Waveform Timing

Measures the off-duty (non-active) period and cycle time of the PWM waveform input to TIFjA. The off-duty period is measured as the period of either the "H" or "L" level input on TIFjA, and the PWN cycle is measured as the interval between two rising or falling edges. Both are measured concurrently. The time obtained is indicated using the TFjECNTA counter clock source as the standard. The duration of the measument is set in the TFjGRB register, which is specified as the number of PWM pulses (TFjGRB register value) input to TIFjA.

Operation of timer F is described below. Figure 21.19 shows an operation example. This is the example in which two PWM cycles in PWM waveform are measured as six counter clock cycles and the off-duty period (low-level period) is measured as four counter clock cycles.

Here the TFjECNTA, TFjECNTB, and TFjECNTC clocks indicate the timing of count and clear operation by the TFjECNTA, TFjECNTB, and TFjECNTC counters, respectively.

The operation of each register in ‘measurement of PWM input waveform timing’ mode are as follows:

- TFjECNTA: Executes upcount using one of the clock buses 0 to 5 as a count source and TIFjA level as enable. Therefore, the time period in which TIFjA is in "L" level is measured. After detecting a compare match in the TFjECNTB coutner, this register clears the count number synchronizng with the next count source clock. If TIFjA is driven "L" level at clearing count by the compare match, the count value becomes H'0000 0100.
- TFjECNTB: Counts the rising edge of TIFjA. A delay of two cycles occurs because of sybchronization processing. When a compare match is detected, the count number is cleared synchronized with the next Pck clock.
- TFjGRA: Functions as the capture register for the TFjECNTA counter. When a compare match in the TFjECNTB counter is detected, this register captures the TFjECNTA counter count number synchronizing with the next TFjECNTA clock.
- TFjGRB: Functions as the compare match register for the TFjECNTB counter. A compare match is detected when the count number in the TFjECNTB counter and TFjGRB match.
- TFjECNTC: Measures time using the same count source as the TFjECNTA counter. This register clears the count number synchronizing with the next TFjECNTA clock after detecting a compare match in the TFjECNTB counter. Since the TFjECNTC counter count clear occurs at the same time with countup, the cleared value is H'0000 0100.
- TFjGRC: Functions as the capture register for the TFjECNTC counter. This register captures the TFjECNTC counter count number synchronizing with the next TFjECNTA clock after detecting a compare match in the TFjECNTB counter.
- ICFFj flag: After detecting a compare match in the TFjECNTB counter, sets the ICFFj flag synchronized with the next TFjECNTA clock.
- TFjGRD: Does not function.

Therefore, TFjECNTB (TFjGRB) and TFjECNTA (TFjGRA) are operating in measurement of time during "L" input levels mode and TFjECNTB (TFjGRB), and TFjECNTC (TFjGRC) are operating in valid edge interval counting mode.

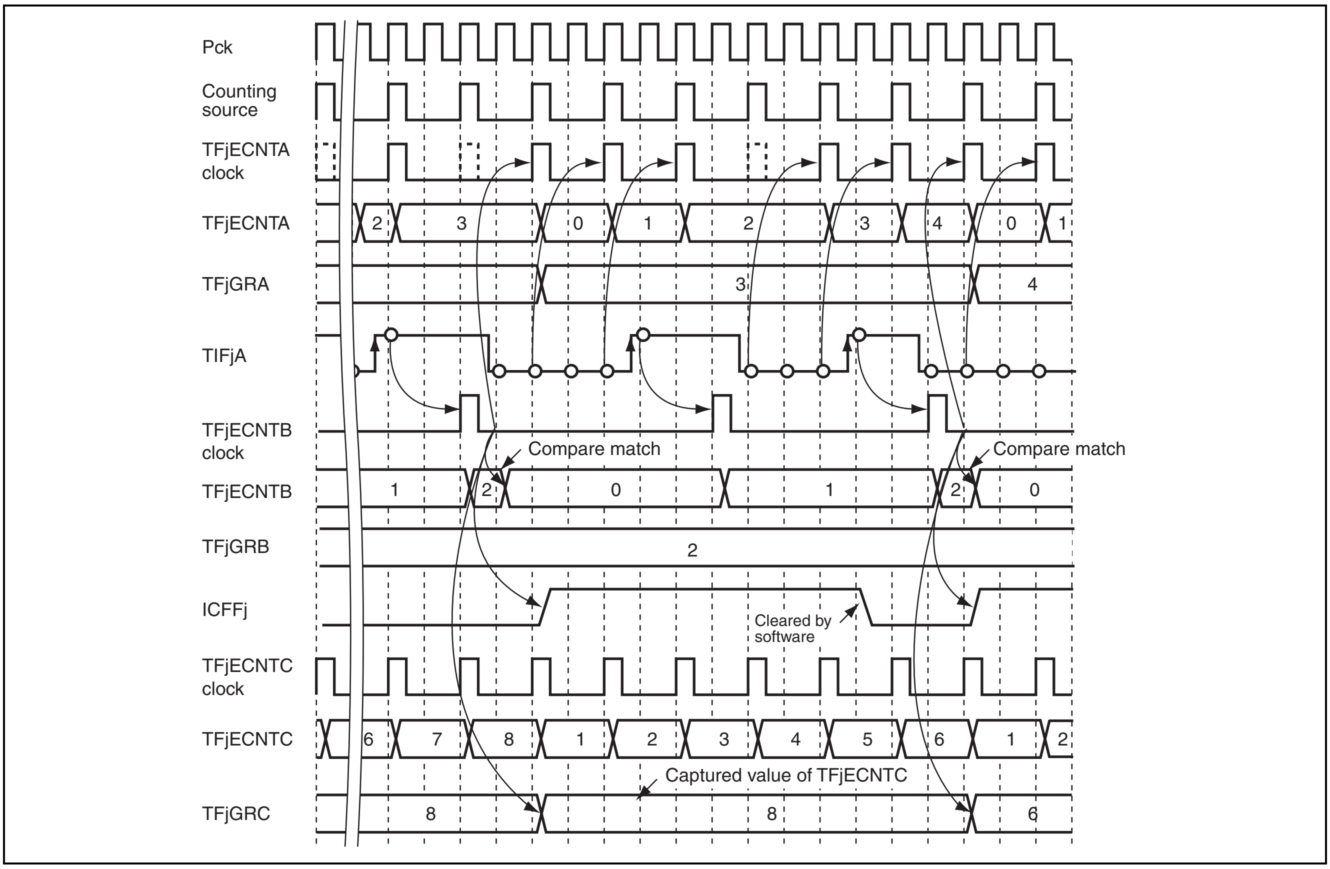


Figure 21.19 Operation Example of Measurement of PWM Input Waveform Timing

21.15.5 Rotation Speed/Pulse Measurement

Measures the number of edges input to TIFjA, the edge input time (time stamp), the off-duty period and PWM cycle time in the PWM waveform that emerges between the last input edge and the edge this time.

The time obtained is indicated using the TFjECNTA counter clock source as the standard. The maximum interval of edge input can be set to the TFjGRB register, which enables to output an interrupt request if the edge input interval exceeds the maximum value.

At this moment, the timer F operates as shown below. Figure 21.20 shows an example of operation. Here the TFjECNTA, TFjECNTB, and TFjECNTC clocks indicate the timing of count and clear operation by the TFjECNTA, TFjECNTB, and TFjECNTC counters, respectively.

The operations of each register in ‘rotation speed/pulse measurement’ mode are as follows:

- TFjECNTA: Executes upcount using one of the clock buses 0 to 5 as a count source and TIFjA input level as enable. Therefore, the time period in which TIFjA is in "L" level is measured. After inputting the edge to TIFjA, this register clears the count number synchronizng with the next count source clock. If TIFjA is driven low at clearing count, the count value becomes H'0000 0100.
- TFjECNTB: Counts the rising edge of TIFjA. A delay of two cycles occurs because of sybchronization processing.
- TFjGRA: Functions as the capture register for the TFjECNTA counter. This register captures the TFjECNTA counter count number synchronizing with the next TFjECNTA clock after inputting the edge to TIFjA.
- TFjGRB: Functions as the capture register for the TFjECNTC counter. When the TFjECNTC counter count and the value in lower eight bits in the TFjGRB register extended with 0 match, this register detects a compare match and set the OVFCFj flag to "1".

- TFjECNTC: Measures time using the same count source as the TFjECNTA counter. This register clears the count number synchronizing with the next TFjECNTA clock after inputting the edge to TIFjA. Since the TFjECNTC counter count clear occurs in the same timing, the cleared value is H'0000 0100.
- TFjGRC: Functions as the capture register for the TFjECNTC counter. This register captures the TFjECNTC counter count number synchronizing with the next TFjECNTA clock after inputting the edge to TIFjA.
- TFjGRD: Functions as the capture register for the TFjECNTC counter. This register captures the TFjECNTA counter, whose number being accumulated to the TFjGRD register, synchronizing with the next TFjECNTA clock after inputting the edge to TIFjA. The value to be added is the TFjECNTC value before clearing.
- ICFFj flag: Sets the ICFFj flag synchronizing with the next TFjECNTA clock after inputting the edge to TIFjA.
- OVFCFj flag: Sets the OVFCFj synchronizing with the next Pck clock after the values in the TFjECNTC counter and TFjGRB register (in lower eight bits extended with 0) match.

While the ICFFj flag is set to "1", information on edge number, off duty cycle, PWM cycle, and edge input time can be obtained by reading the TFjECNTB counter, TFjGRA, TFjGRC, and TFjGRD registers, respectively. The capture timing of TFjGRA, TFjGRC, and TFjGRD registers synchronizes with the count clock of the TFjECNTA counter. Note that if the edge input cycle is shorter than the TFjECNTA counter count clock cycle, incorrect measurement may occur.

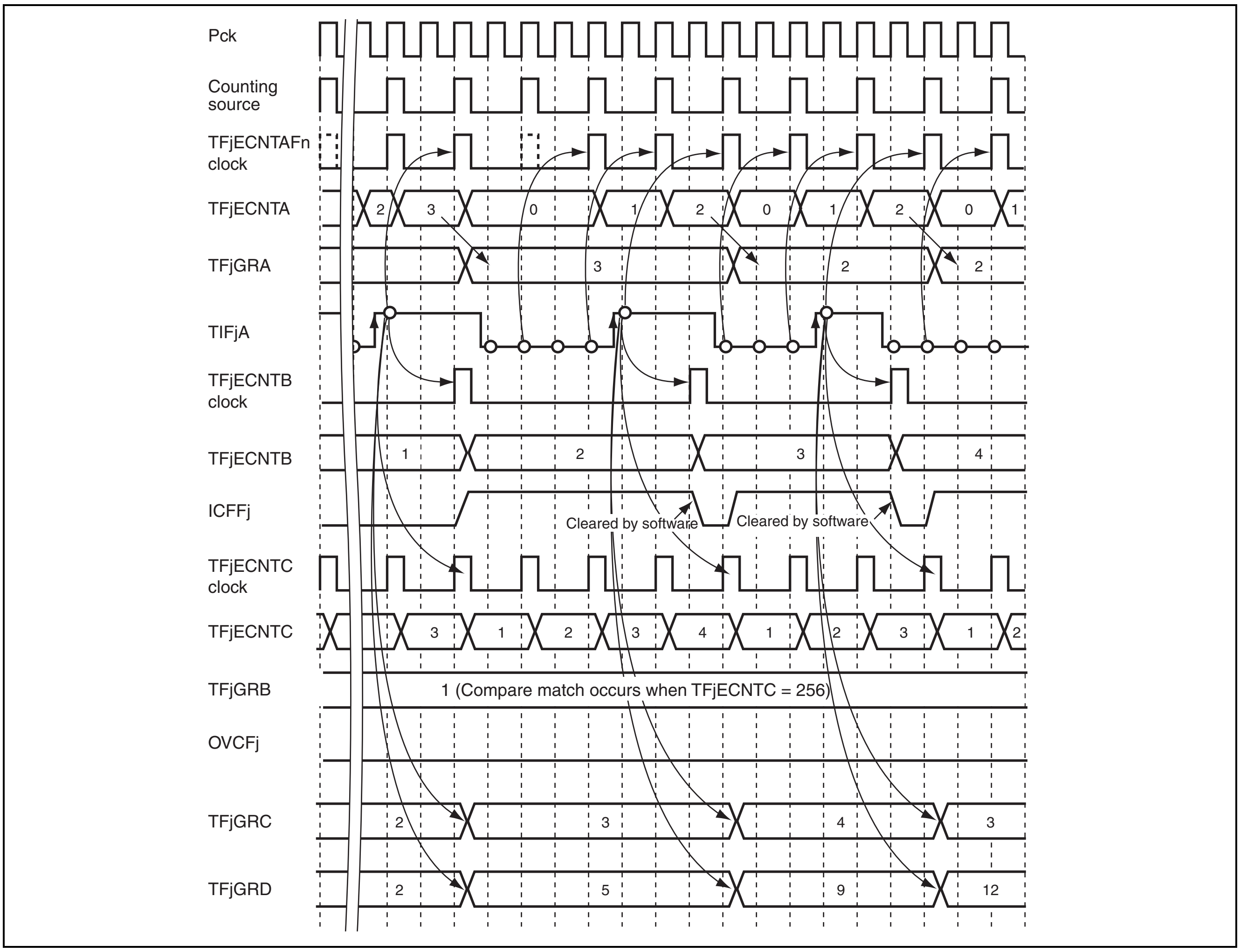


Figure 21.20 Operation Example of Rotation Speed/Pulse Measurement

21.15.6 Up/Down Event Count

This register uses the TIFjA pin, one of the two external input pins (TIFjA, TIFjB), as the count source, and TIFjB switches upcount to and from downcount. If a count period is designated to the TFjGRA regsiter, the count number after designation can be obtained in the TFjGRB register. The counting period is the period of the TFjCNTA counter count source clock (TFjGRA register value). At this moment, the timer F operates as shown below. Figure 21.21 shows an example of operation. Here the TFjECNTA and TFjECNTB clocks indicate the timing of count and clear operation by the TFjECNTA and TFjECNTB counters, respectively.

The operations of each register in up/down count operation mode are as follows:

- TFjECNTA: Measures time using one of the clock buses 0 to 5. When a compare match is detected, the count value is cleared synchronized with the next Pck clock.
- TFjECNTB: Upcount/downcount operation is performed at both rising and falling edges of TIFjA. Count direction is determined by the TIFjB input level. (See table 21.7.) Because of synchronization processing, a delay of two cycles occurs in TIFjA and TIFjB.
- TFjGRA: Functions as the compare match register for the TFjECNTA counter. A compare match is detected when the count number in the TFjECNTA counter and GRA register match.
- TFjGRB: Functions as the capture register for the TFjECNTB counter. When a compare match in the TFjECNTA counter is detected, this register captures the TFjECNTB counter count number synchronizing with the next Pck clock.
- ICFFj flag: After detecting a compare match in the TFjECNTA counter, sets the ICFFj flag synchronized with the next Pck clock.
- TFjECNTC, TFjGRC, TFjGRD: Do not function.

Table 21.7 Count Direction in Up/Down Event Count Mode

	Count Direction			
Input	**Upcount**		**Downcount**	
TIFjA	↑	↓	↑	↓
TIFjB	"L" level		"H" level	

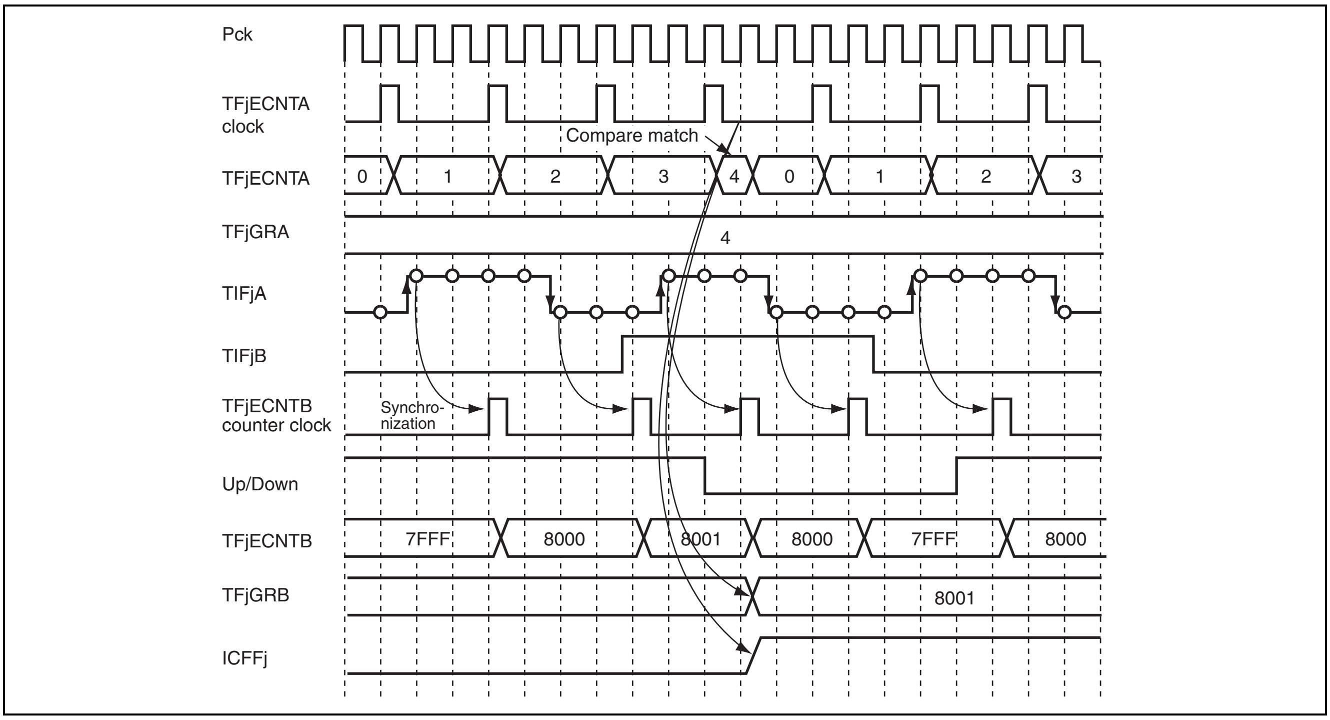


Figure 21.21 Operation Example of Up/Down Event Count

21.15.7 Four-time Multiplication Event Count

The count operation is executed using the external two input pins (TIFjA, TIFjB) as the count sources. Upcount or downcount is switched according to their input states. If a count period is designated to the TFjGRA register, the count number after designation can be obtained in the TFjGRB register. The counting period is the period of the TFjECNTA counter cont source clock (TFjGRA register value).

At this moment, the timer F operates as shown below. Figure 21.22 shows an example of operation. Here the TFjECNTA and TFjECNTB clocks indicate the timing of count and clear operation by the TFjECNTA and TFjECNTB counters, respectively.

The operations of each register in ‘four-time event count operation’ mode are as follows:

- TFjECNTA: Measures time using one of the clock buses 0 to 5. When a compare match is detected, the count value is cleared synchronized with the next Pck clock.
- TFjECNTB: Upcount/downcount operation is performed at both rising and falling edges of TIFjA and TIFjB respectively. Count direction is determined by the other signal input level. (See table 21.8.) Because of synchronization processing, a delay of two cycles occurs in TIFjA and TIFjB.
- TFjGRA: Functions as the compare match register for the TFjECNTA coutner. A compare match is detected when the count number in the TFjECNTA counter and TFjGRA register agree.
- TFjGRB: Functions as the capture register for the TFjECNTB coutner. When a compare match in the TFjECNTA counter is detected, this register captures the TFjECNTB counter count number synchronizing with the next Pck clock.
- ICFFj flag: After detecting a compare match in the TFjECNTA counter, sets the ICFFj flag synchronized with the next Pck clock.
- TFjECNTC, TFjGRC, TFjGRD: Do not function.

Table 21.8 Count Direction in Four-time Multiplication Event Count Mode

	Count Direction							
Input	**Upcount**				**Downcount**			
TIFnA	"H" level	↓	"L" level	↑	"H" level	↓	"L" level	↑
TIFnB	↑	"H" level	↓	"L" level	↓	"L" level	↑	"H" level

Note: • Operation when edge inputs in TIFjA and TIFjB are detected simultaneously is not guaranteed. The interval between edge inputs in TIFjA and TIFjB must be at least 1.5 cycles (Pck clock).

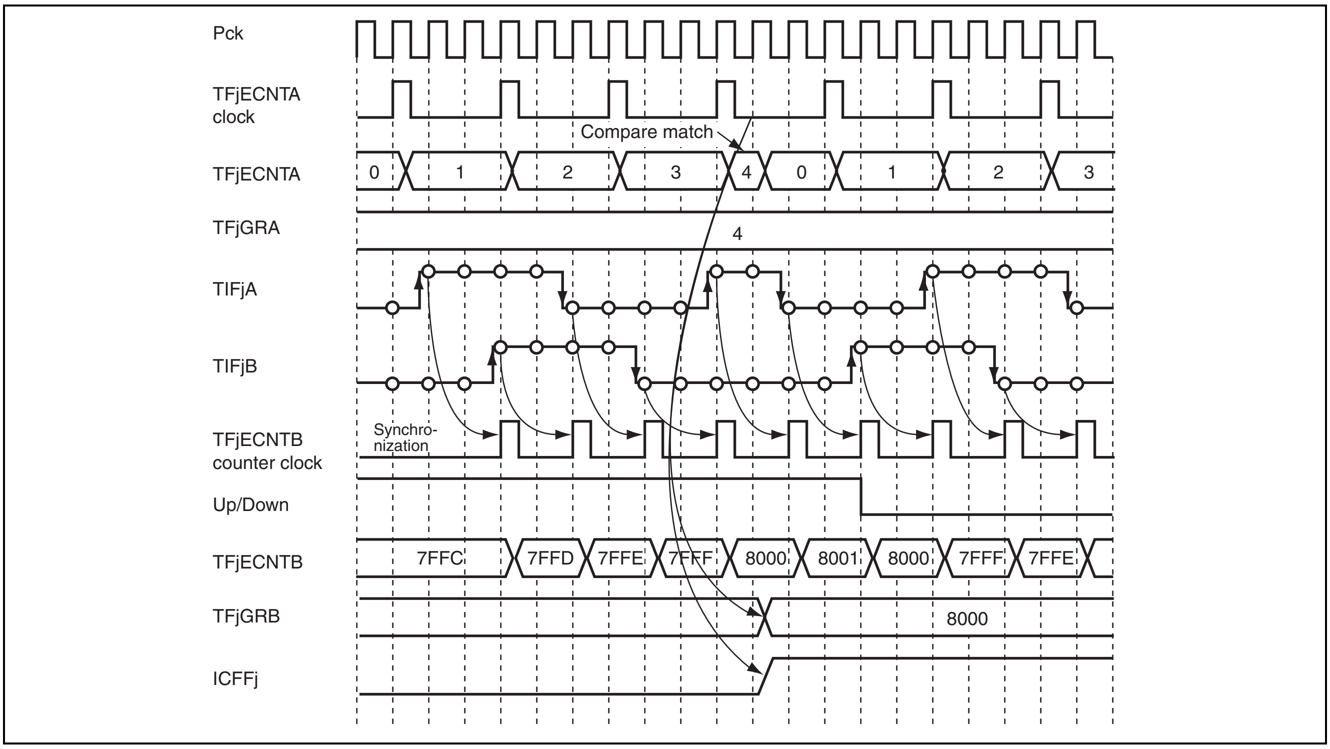


Figure 21.22 Operation Example of Four-time Multiplication Event Count

21.15.8 Overflow and Underflow

When a counter value changes H'FFFF FF00 (TFjECNTA, TFjECNTC) to H'0000 0000 (TFjECNTA, TFjECNTC) and H'FFFF (TFjECNTB) to H'0000 (TFjECNTB) except counter clear operation, overflow is detected. Overflow flags are set at the same time when the counter value changes H'0000 0000 (or H'0000). OVFAFj, OVFBFj, and OVFCFj are set when overflow is detected in the TFjECNTA, TFjECNTB, or TFjECNTC counters, respectively.

When a counter value changes H'0000 (TFjECNTB) to H'FFFF (TFjECNTB), underflow is detected. Underflow flag is set at the same time when the counter value changes from H'0000 to H'FFFF. Underflow occurs only in the TFjECNTB counter, and OVFBFj is set upon its detection.

21.16 Overview of Timer G

Timer G consists of six subblocks that are identical to each other.

Each subblock counts the input clock signal and generates a negative logic pulse signal when the designated time elapses at each cycle of the Pck clock. The generated pulse is used to activate an interrupt trigger of the A/D converter. Interrupt requests can be issued other than the pulse signal and can request DMA transfer to the DMAC. The counter clock is selected from five lines of the clock bus.

21.16.1 Block Diagram of Robots

Timer G subblocks consist of one 16-bit timer TGk counter (TGkCNT), one TGk compare match register (TGkOCR), and controller.

Figure 21.23 is a block diagram of timer G.

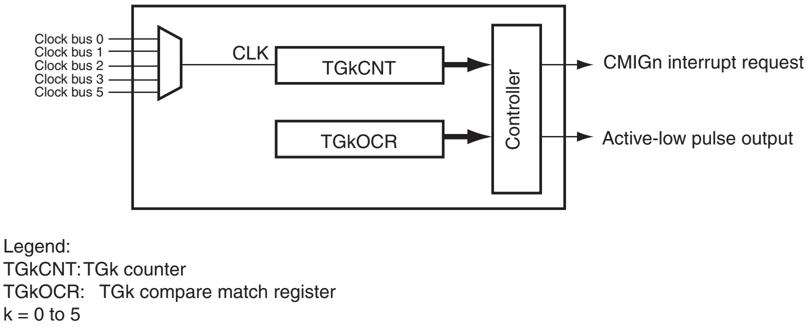


Figure 21.23 Block Diagram of Timer G

21.16.2 Interrupt Requests

Six timer G interrupts, CMIG0 to CMIF5, are available for timer G. When a compare match is detected, an interrupt request is output. The interrupt request is received by the DMAC and the INTC. The designated processing takes place according to the settings of each.

21.17 Description of Timer G Registers

21.17.1 TG Start Register (TGSTR)

The TGSTR register enables and disables the subblocks of timer G. Timer G counters run when the STRGk bit and the TGE bit in the ATU-III master enable register (ATUENR) are both set to "1".

TG Start Register (TGSTR) <P4 address: location H'FFFF E501>

Bit:	7	6	5	4	3	2	1	0
	—	—	STRG 5	STRG 4	STRG 3	STRG 2	STRG 1	STRG 0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	STRG5	0	R	W	Counter G Start Bits These bits enable and disable TGk counter (TGkCNT) in the subblock. When these bits are cleared to "0", the TGkCNT counter is stopped. While TCNTGk is stopped, it retains the previous value. When these bits are set to "1", TCNTGn is resumed from the previous value. The TGkCNT counter runs when these bits and the TGE bit in the ATUENR register are both set to "1". 0: TGkCNT counter is disabled 1: TGkCNT counter is enabled Note: • The prescalers run regardless of the setting of the counter G start bit and are not synchronized with the timing at which the TGkCNT counter is started. Therefore, the period from startup until when the TGkCNT counter is incremented for the first time can vary due to hardware conditions within the selected count source cycle (resolution).
4	STRG4	0	R	W	
3	STRG3	0	R	W	
2	STRG2	0	R	W	
1	STRG1	0	R	W	
0	STRG0	0	R	W	

Legend: k = 0 to 5, corresponding to subclocks G0 to G5, respectively

21.17.2 TGk Control Register (TGkCR)

The TGkCR register sets the operating mode of each subblock of timer G.

TG0 Control Register (TG0CR)	<P4 address: location H'FFFF E580>
TG1 Control Register (TG1CR)	<P4 address: location H'FFFF E590>
TG2 Control Register (TG2CR)	<P4 address: location H'FFFF E5A0>
TG3 Control Register (TG3CR)	<P4 address: location H'FFFF E5B0>
TG4 Control Register (TG4CR)	<P4 address: location H'FFFF E5C0>
TG5 Control Register (TG5CR)	<P4 address: location H'FFFF E5D0>

Bit:	7	6	5	4	3	2	1	0
	—	CKSELGk			—	—	CMPO EGk	CM EGk
After Reset:	0	0	0	0	0	0	0	0

Legend: k = 0 to 5, corresponding to subclocks G0 to G5, respectively

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	CKSELGk	000	R	W	Clock Select Bits (Gk) Specify the signal on one of clock-bus lines 0 to 3 and 5 as the clock source of TGk counter (TGkCNT) of the subblock. The signal on lines 0 to 3 have been frequency-divided by prescalers 0 to 3, respectively. Clock bus 5 supplies externally input clock B (TCLKB). Stop timer G before making or changing the counter-clock selection. 000: Clock-bus line 0 (prescaler 0) 001: Clock-bus line 1 (prescaler 1) 010: Clock-bus line 2 (prescaler 2) 011: Clock-bus line 3 (prescaler 3) 100: Setting prohibited 101: Clock-bus line 5 (TCLKB) 110: Setting prohibited 111: Setting prohibited Note: • The edge of an external input clock is extracted before it is output on a clock bus. When using external input clock B, select the edge to be extracted by setting the CB5EG bit in the ATU-IIIS clock bus control register (ATCBCNT).
3, 2	—	All 0	0	0	Reserved Bits This bit is always read as "0". The write value should always be "0".
1	CMPOEGk	0	R	W	Pulse Output Enable Bit (Gk) Selects whether or not a compare match pulse is externally output on compare match between TGk counter (TGkCNT) and TGk compare match register (TGkOCR). 0: Pulse is not output on compare match between the TGkCNT counter and TGkOCR register 1: Pulse is output on compare match between the TGkCNT counter and TGkOCR register

Bit	Abbreviation	After Reset	R	W	Description
0	CMEGk	0	R	W	Compare Match Interrupt Enable Bit (Gk) Enables and disables output of interrupt requests on compare match between TGk status register (TGkSR) and compare match flag Gk (CMFGk). 0: Interrupt request is not issued on compare match of CMFGk 1: Interrupt request is issued on compare match of CMFGk

Legend: k = 0 to 5, corresponding to subclocks G0 to G5, respectively

21.17.3 TGk Status Register (TGkSR)

The TGkSR register indicates occurrence of TGk counter overflow and compare match.

These flags are interrupt sources. When an interrupt is enabled by the setting of the corresponding bit in the TGk control register (TGkCR), they can be used to send an interrupt request to the CPU or a DMA transfer request to the DMAC.

TG0 Status Register (TG0SR)	<P4 address: location H'FFFF E581>
TG1 Status Register (TG1SR)	<P4 address: location H'FFFF E591>
TG2 Status Register (TG2SR)	<P4 address: location H'FFFF E5A1>
TG3 Status Register (TG3SR)	<P4 address: location H'FFFF E5B1>
TG4 Status Register (TG4SR)	<P4 address: location H'FFFF E5C1>
TG5 Status Register (TG5SR)	<P4 address: location H'FFFF E5D1>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OVFGk	CMFGk
After Reset:	0	0	0	0	0	0	0	0

Legend: k = 0 to 5, corresponding to subclocks G0 to G5, respectively

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
1	OVFGk	0	R	*[1]	Overflow Flag Gk Indicates whether or not TGk counter Gn (TGkCNT) has overflowed. This bit cannot be set to "1" by software. No interrupt corresponds to this bit. 0: TGkCNT counter has not overflowed 1: TGkCNT counter has overflowed [Condition for clearing to "0"] • Writing "0" to OVFGk after reading it as "1" [Condition for setting to "1"] • When TGkCNT counter overflowed (from H'FFFF to H'0000)
0	CMFGk	0	R	*[1]	Compare Match Flag Gk Indicates whether or not a compare match has occurred in subblocks. This flag cannot be set to "1" by software. When the CMEGk bit in the timer control register is set to "1", setting this flag to "1" causes a compare match interrupt to be issued. 0: Compare match has not occurred in subblock Gk 1: Compare match has occurred in subblock Gk [Conditions for clearing to "0"] • Writing "0" to CMFGk after reading it as "1" • When a DMA transfer request triggered by an compare match interrupt is accepted by the DMAC [Condition for setting to "1"] • When compare match occurred in subblocks Gk

Note: *1 Only writing "0" to this bit after reading it as "1" to clear the flag. Writing "1" to this bit has no effect.

Legend: k = 0 to 5, corresponding to subclocks G0 to G5, respectively

21.17.4 TGk Counter (TGkCNT)

The TGkCNT counter is provided one for each subblock and are incremented by the clock selected in the corresponding control register. Lines 0 to 5 of the clock bus can be selected as the input clock.

These counter values are constantly compared with the value in TGk compare match register (TGkOCR). When they match, Compare match flag G (CMFG) is set and the counter value is cleared to "H'0000" in the next Pck clock cycle. If counter clearing by compare match and incrementation occur simultaneously, the TGkCNT counter is initialized to "H'0001". This occurs when the TGkCNT counter is driven by the clock whose frequency is equal to the Pck clock.

TG0 Counter (TG0CNT)	<P4 address: location H'FFFF E584>
TG1 Counter (TG1CNT)	<P4 address: location H'FFFF E594>
TG2 Counter (TG2CNT)	<P4 address: location H'FFFF E5A4>
TG3 Counter (TG3CNT)	<P4 address: location H'FFFF E5B4>
TG4 Counter (TG4CNT)	<P4 address: location H'FFFF E5C4>
TG5 Counter (TG5CNT)	<P4 address: location H'FFFF E5D4>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TGkCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	TGkCNT	All 0	R	W	TGk Timer Counter These bits store the up-counter value.

Legend: k = 0 to 5, corresponding to subblocks G0 to G5, respectively

21.17.5 TGk Compare Match Register (TGkOCR)

The TGkOCR register is provided one for each subblock and function as the output compare register for timer counter G (TGkCNT).

Do not set the TGkOCR register to H'0000. If H'0000 is set, compare matches occur at incorrected cycles.

TG0 Compare Match Register (TG0OCR)	<P4 address: location H'FFFF E586>
TG1 Compare Match Register (TG1OCR)	<P4 address: location H'FFFF E596>
TG2 Compare Match Register (TG2OCR)	<P4 address: location H'FFFF E5A6>
TG3 Compare Match Register (TG3OCR)	<P4 address: location H'FFFF E5B6>
TG4 Compare Match Register (TG4OCR)	<P4 address: location H'FFFF E5C6>
TG5 Compare Match Register (TG5OCR)	<P4 address: location H'FFFF E5D6>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TGkOCR															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

<After Reset: H'FFFF>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	TGkOCR	All 1	R	W	TGk Compare Match These bits set the compare match value.

Legend: k = 0 to 5, corresponding to subclocks G0 to G5, respectively

21.18 Operations of Timer G

An active-low pulse is output for one cycle of the Pck clock after a time set in the TGkOCR register has elapsed. The initial level on the output pin is a level of 1. Set the number of cycles of the TGkCNT counter clock.

The generated pulse can be used to activate the A/D converter by setting the compare match pulse output enable bit (CMPOEG) in TGk control register G (TGkCR).

When compare match occurs, the compare match flag (CMFG) in TGk status register (TGkSR) is set. DMA transfer or interrupt requests can be issued for the DMAC or CPU by setting the compare match interrupt enable bit (CMEG) in the TGkCR register.

Figure 21.24 shows operation example for counters and compare match. Here the TGkCNT clock indicates the timing of count and clear operation by the TGkCNT counter.

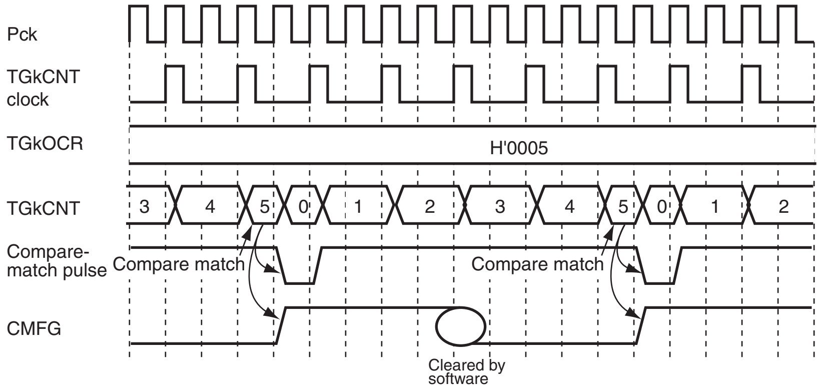


Figure 21.24 Operation Example of Counter and Compare Match

21.19 Overview of Timer TOU

Timer TOU (Timer Output Unification) comprises a 24-bit output timer with a total of 40 channels (five subblocks, each with eight channels). The operation mode of each timer TOU channel can be selected by software from among the following.

Output modes with no correction

- PWM output mode
- One-shot PWM output mode
- One-shot output mode
- Continuous output mode

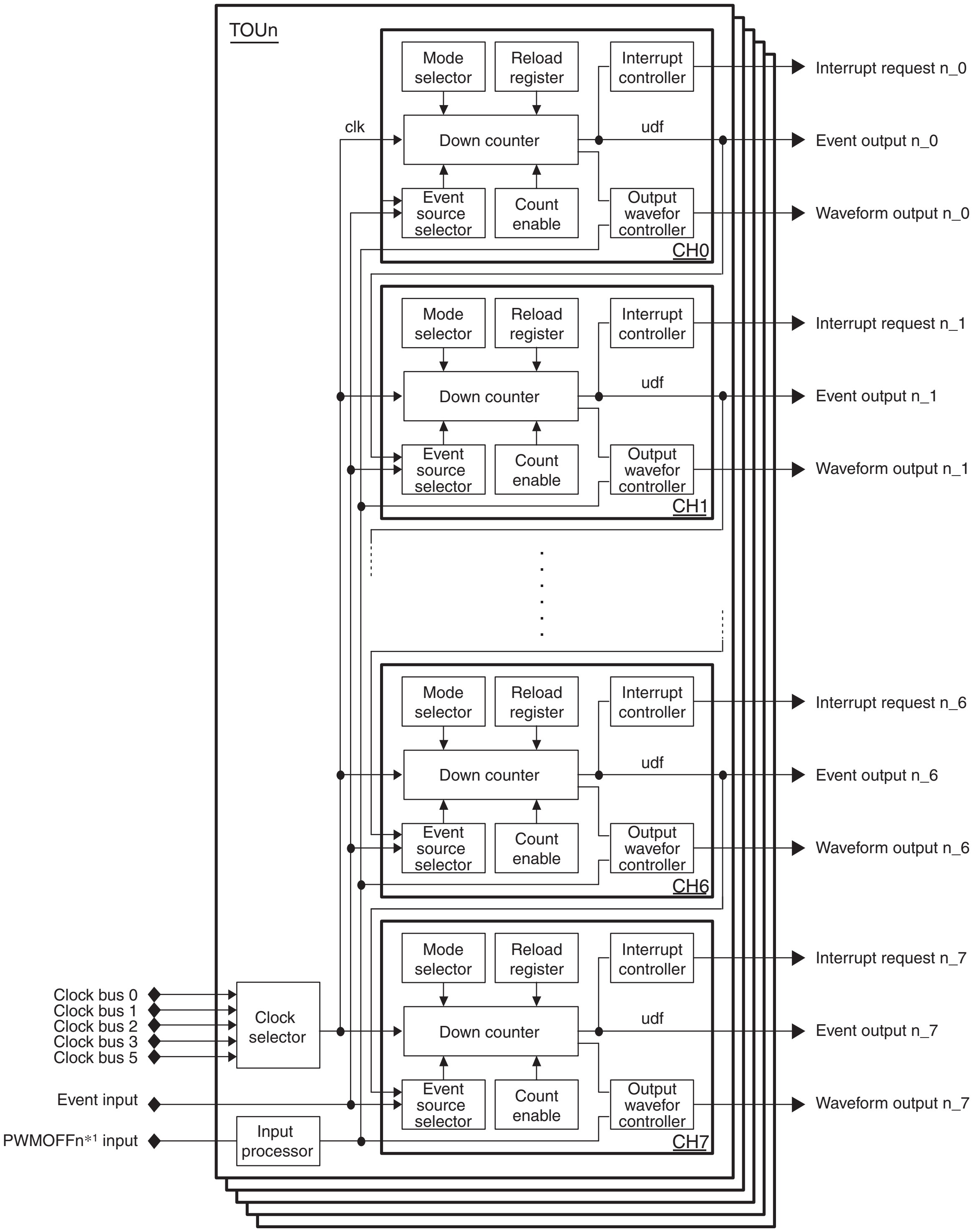
Table 21.9 lists the specifications of timer TOU.

Table 21.9 Specifications of Timer TOU (24-Bit Output Timer)

Item	Specification
Channels	8 channels × 5 subblocks
Counters	24-bit down counters (16-bit down counters in PWM output or one-shot PWM output mode)
Reload registers	24-bit reload registers (16-bit reload registers in PWM output or one-shot PWM output mode)
Timer activation	• TOU0_7 underflow • TOU1_7 underflow • TOU2_7 underflow • TOU3_7 underflow • TOU4_7 underflow • Timer A channel 2 input capture • Timer A channel 3 input capture • Timer A channel 4 input capture • Timer G channel 5 compare match • TOUn_m – 1 (except TOUn_0) • PDAC event output signals A to D
Operating modes	[Output modes with no correction] • PWM output mode • One-shot PWM output mode • One-shot output mode • Continuous output mode
Interrupt request generation	Generation at counter underflow supported
DMA transfer request generation	Generation at counter underflow supported

21.19.1 Block Diagram

Figure 21.25 is a block diagram of timer TOU.



Legend: n = 0 to 4

Note: *1 The PWMOFF inputs are PWMOFF1, PWMOFF1, PWMOFF3, and PWMOFF4. PWMOFF2 is not implemented.

Figure 21.25 Block Diagram of Timer TOU (24-Bit Output Timer)

21.19.2 Overview of Timer TOU Operating Modes

An overview of the operating modes supported by timer TOU is provided below. Note that for each timer TOU channel only one of the following modes may be selected.

(1) PWM Output Mode (No Correction Function)

In PWM output mode, two reload registers are used to generate a waveform with a user-defined duty cycle. In PWM output mode, timer TOU operates as a 16-bit timer.

When the timer counter is enabled after setting the reload 0 register and reload 1 register to their initial values, the value of the reload 0 register is loaded in the counter in synchronization with the count clock and down counting starts. At the first counter underflow cycle, the contents of the reload 1 register are loaded in the counter, and at each subsequent underflow the counter is reloaded with the value of the reload 0 register or reload 1 register, alternately. The setting of the reload 0 register plus 1 or the setting of the reload 1 register plus 1 are each valid as counter values.

The timer stops operating when the value specifying counter disabled is written to the counter enable bit. (The halt of timer operation is not synchronized with the PWM output period.)

In PWM output mode, the F/F output waveform consists of the inverted value of the TOUn output control register, output at counter start and at each even-numbered underflow, and the value of the TOUn output control register, output at each odd-numbered underflow.

It is also possible to generate an interrupt request or DMA transfer request at each even-numbered underflow after the counter starts.

(2) One-Shot PWM Output Mode (No Correction Function)

In one-shot PWM output mode, two reload registers are used to generate one instance only of a waveform with a user-defined duty cycle. In one-shot PWM output mode, timer TOU operates as a 16-bit timer.

When the timer counter is enabled after setting the reload 0 register and reload 1 register to their initial values, the value of the reload 0 register is loaded in the counter in synchronization with the count clock and down counting starts. At the first counter underflow cycle, the contents of the reload 1 register are loaded in the counter, and at the second underflow the counter stops. The setting of the reload 0 register plus 1 or the setting of the reload 1 register plus 1 are each valid as counter values.

Stopping the timer by software is accomplished by writing the value specifying counter disabled to the counter enable bit. (The halt of timer operation is not synchronized with the PWM output period.)

In one-shot PWM output mode, the F/F output waveform consists of the value of the TOUn output control register, output at counter start and at the second underflow, and the inverted value of the TOUn output control register, output at the first underflow. (In contrast to PWM output mode, the F/F output is not inverted at counter start.)

It is also possible to generate an interrupt request or DMA transfer request at the first and second counter underflows. Generation of the first interrupt request or DMA transfer request can be enabled or disabled by software.

(3) One-Shot Output Mode (No Correction Function)

In one-shot output mode, a single pulse with a width equal to the reload register setting value plus 1 is generated, then counter operation stops.

When the timer counter is enabled after setting the reload register, the contents of the reload register are loaded in the counter in synchronization with the count clock and counter operation starts. The counter counts down until an underflow occurs, then stops.

In one-shot output mode, the F/F output waveform consists of the inverted value of the TOUn output control register, output at counter start, and the value of the TOUn output control register, output when an underflow occurs. The result is a single one-shot pulse waveform equivalent to the setting of the reload register plus 1.

It is also possible to generate an interrupt request or DMA transfer request at the counter underflow.

(4) Continuous Output Mode (No Correction Function)

In continuous output mode, the counter counts down from its setting value, and the value of the reload register is loaded when an underflow occurs. This operation is repeated at each subsequent counter underflow, generating successive pulses equivalent to the inverted value of the reload register plus 1.

When the timer counter is enabled after setting the timer and reload register, down counting starts from the setting value of the timer and continues until an underflow occurs.

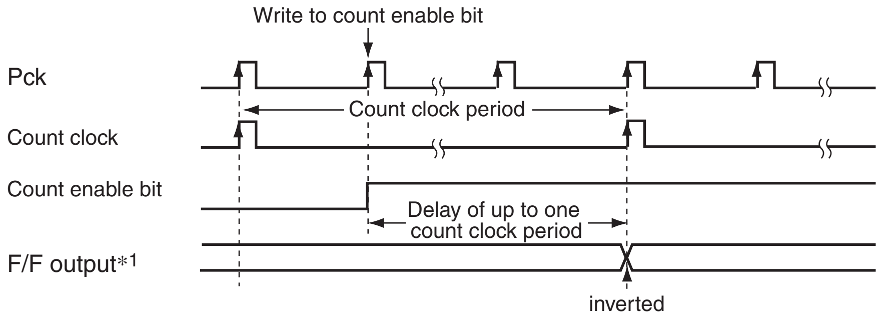
At the underflow cycle, the contents of the reload register are loaded in the counter and down counting begins again. This operation is repeated at each subsequent underflow. The timer stops operating when the value specifying counter disabled is written to the counter enable bit.

In continuous output mode, the F/F output waveform consists of the inverted value of the TOUn output control register, output at counter start and at each even-numbered underflow, and the value of the TOUn output control register, output at each odd-numbered underflow. This pulse waveform is output continuously until the counter stops.

It is also possible to generate an interrupt request or DMA transfer request at each counter underflow.

Count clock delay

- Timer operation is synchronized with the count clock, so there is a delay of up to one count clock period between the point at which the counter enable bit is written to and the start of timer operation. In operating modes in which F/F output is inverted at counter start, the F/F output is inverted in synchronization with the count clock.



Note: *1 Example of operating mode with F/F output inverted at counter start

Figure 21.26 Count Clock Delay

21.20 Descriptions of Timer TOU Registers

21.20.1 TOUn Control Register (TOnCR)

The TOnCR register is used to select the count clock and to enable or disable the short-circuit prevention function.

Note: • Only set or update the TOnCR register while the TTE bit in the ATUENR register and the TCENTnm bit in the TOnCENR register are both cleared to "0".

TOU0 Control Register (TO0CR)	<P4 address: location H'FFFF E600>
TOU1 Control Register (TO1CR)	<P4 address: location H'FFFF E700>
TOU2 Control Register (TO2CR)	<P4 address: location H'FFFF E800>
TOU3 Control Register (TO3CR)	<P4 address: location H'FFFF E900>
TOU4 Control Register (TO4CR)	<P4 address: location H'FFFF EA00>

Bit:	7	6	5	4	3	2	1	0
	—	CKSELTn			—	—	—	SHENTn
After Reset:	0	0	0	0	0	0	0	0

Legend: n = 0 to 4

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 4	CKSELTn	000	R	W	TOUn Clock Select Bits These bits select one among clock buses 0 to 3 and 5 as the count clock. Clock buses 0 to 3 correspond to the divided clocks output by prescalers 0 to 3 and clock bus 5 to external clock input B (TCLKB). Select the count clock when TOUn operation is halted. 000: Clock-bus line 0 (prescaler 0) 001: Clock-bus line 1 (prescaler 1) 010: Clock-bus line 2 (prescaler 2) 011: Clock-bus line 3 (prescaler 3) 100: Setting prohibited 101: Clock-bus line 5 (TCLKB) 110: Setting prohibited 111: Setting prohibited Note: • The edge of an external input clock is extracted before it is output on a clock bus. When using external input clock B, select the edge to be extracted by setting the CB5EG bit in the ATU-IIIS clock bus control register (ATCBCNT).
3 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	SHENTn	0	R	W	TOUn Short-Circuit Prevention Function Enable Bit 0: Short-circuit prevention function disabled 1: Short-circuit prevention function enabled

21.20.2 TOUn Timer Interrupt Enable Register (TOnIER)

The TOnIER register is used to enable or disable underflow interrupt requests. Setting a bit to "1" enables, clearing it to "0" disables, the corresponding interrupt.

TOU0 Timer Interrupt Enable Register (TO0IER)	<P4 address: location H'FFFF E601>
TOU1 Timer Interrupt Enable Register (TO1IER)	<P4 address: location H'FFFF E701>
TOU2 Timer Interrupt Enable Register (TO2IER)	<P4 address: location H'FFFF E801>
TOU3 Timer Interrupt Enable Register (TO3IER)	<P4 address: location H'FFFF E901>
TOU4 Timer Interrupt Enable Register (TO4IER)	<P4 address: location H'FFFF EA01>

Bit:	7	6	5	4	3	2	1	0
	UDE Tn7	UDE Tn6	UDE Tn5	UDE Tn4	UDE Tn3	UDE Tn2	UDE Tn1	UDE Tn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	UDETn7	0	R	W	TOUn_0 to TOUn_7 Underflow Interrupt Enable Bits These bits enable or disable output of interrupt requests triggered by UDFTn0 to UDFTn7 when one of underflow flags TOUn_0 to TOUn_7 (UDFTn0 to UDFTn7) in the TOUn status register (TOnSR) is set to "1". 0: Output of underflow interrupt requests triggered by TOUn_0 to TOUn_7 disabled 1: Output of underflow interrupt requests triggered by TOUn_0 to TOUn_7 enabled
6	UDETn6	0	R	W	
5	UDETn5	0	R	W	
4	UDETn4	0	R	W	
3	UDETn3	0	R	W	
2	UDETn2	0	R	W	
1	UDETn1	0	R	W	
0	UDETn0	0	R	W	

Legend: n = 0 to 4

21.20.3 TOUn Output Control Register (TOnOUCR)

The TOnOUCR register is used to specify the initial value of the TOUn F/F (flip flop) output signals (TOn0 to TOn7). Writing to this register causes the same value to be written simultaneously to the TOUn flip-flop output data register (TOnFFDR). The contents of the TOnOUCR register can be manipulated only when writing is enabled by the setting of the TOnFFPR register.

TOU0 Output Control Register (TO0OUCR)	<P4 address: location H'FFFF E602>
TOU1 Output Control Register (TO1OUCR)	<P4 address: location H'FFFF E702>
TOU2 Output Control Register (TO2OUCR)	<P4 address: location H'FFFF E802>
TOU3 Output Control Register (TO3OUCR)	<P4 address: location H'FFFF E902>
TOU4 Output Control Register (TO4OUCR)	<P4 address: location H'FFFF EA02>

Bit:	7	6	5	4	3	2	1	0
	IOC Tn7	IOC Tn6	IOC Tn5	IOC Tn4	IOC Tn3	IOC Tn2	IOC Tn1	IOC Tn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	IOCTn7	0	R	W	TOUn_0 to TOUn_7 Output Control Bits These bits specify the initial value of the TOUn F/F output signals. 0: Initial value of TOUn F/F output signal (TOn0 to TOn7) is "L" level 1: Initial value of TOUn F/F output signal (TOn0 to TOn7) is "H" level
6	IOCTn6	0	R	W	
5	IOCTn5	0	R	W	
4	IOCTn4	0	R	W	
3	IOCTn3	0	R	W	
2	IOCTn2	0	R	W	
1	IOCTn1	0	R	W	
0	IOCTn0	0	R	W	

Legend: n = 0 to 4

21.20.4 TOUn Status Register (TOnSR)

Each bit in the TOnSR register is a status bit for distinguishing whether or not an interrupt request has occurred. These bits are set by hardware when an interrupt request is generated, and they cannot be set by software. Each status bit (flag) can be cleared by writing "0" to it after reading it as "1". The operation of the TOnSR register is not affected by the values of the bits in the TOnIER register. The TOnSR register can therefore be used to confirm the operation of peripheral functions. During interrupt handling, only clear the status bit relevant to the interrupt handling routine. Clearing status bits related to interrupts for which interrupt handling has not taken place, has the effect of clearing unexecuted interrupt requests.

TOU0 Status Register (TO0SR)	<P4 address: location H'FFFF E603>
TOU1 Status Register (TO1SR)	<P4 address: location H'FFFF E703>
TOU2 Status Register (TO2SR)	<P4 address: location H'FFFF E803>
TOU3 Status Register (TO3SR)	<P4 address: location H'FFFF E903>
TOU4 Status Register (TO4SR)	<P4 address: location H'FFFF EA03>

Bit:	7	6	5	4	3	2	1	0
	UDF Tn7	UDF Tn6	UDF Tn5	UDF Tn4	UDF Tn3	UDF Tn2	UDF Tn1	UDF Tn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	UDFTn7	0	R	*1	TOUn_0 to TOUn_7 Underflow Flags Each of these status flags indicates the occurrence of an underflow by the corresponding TOUnm counter (TOnmCNT). When the flag returns a value of "1" when read, the counter TOnmCNT has overflowed. These flags cannot be set to "1" by software. To one of flags UDFTn0 to UDFTn7, write "0" to it after reading it as "1". Writing "0" before reading it as "1" has no effect. 0: No underflow interrupt request generated 1: Underflow interrupt request generated [Conditions for clearing to "0"] • When "0" is written one of bits UDFTn0 to UDFTn7 after reading it as "1" • When a DMA transfer request triggered by the underflow interrupt corresponding to the flag is accepted by the DMAC [Condition for setting to "1"] • When the TOnmCNT counter value underflows (transitions from H'0000 0000 to H'FFFF FFFF)
6	UDFTn6	0	R	*1	
5	UDFTn5	0	R	*1	
4	UDFTn4	0	R	*1	
3	UDFTn3	0	R	*1	
2	UDFTn2	0	R	*1	
1	UDFTn1	0	R	*1	
0	UDFTn0	0	R	*1	

Note: *1 "0" may be written to this bit only after reading it as "1" to clear the flag. Writing "1" to this bit has no effect.

Legend: n = 0 to 4

21.20.5 TOUn Counter Enable Protect Register (TOnCEPR)

The TOnCEPR register is used to enable or disable overwriting of the counter enable bits described in section 21.20.6, TOUn Counter Enable Register (TOnCENR).

TOU0 Counter Enable Protect Register (TO0CEPR)	<P4 address: location H'FFFF E604>
TOU1 Counter Enable Protect Register (TO1CEPR)	<P4 address: location H'FFFF E704>
TOU2 Counter Enable Protect Register (TO2CEPR)	<P4 address: location H'FFFF E804>
TOU3 Counter Enable Protect Register (TO3CEPR)	<P4 address: location H'FFFF E904>
TOU4 Counter Enable Protect Register (TO4CEPR)	<P4 address: location H'FFFF EA04>

Bit:	7	6	5	4	3	2	1	0
	TCEPTn7	TCEPTn6	TCEPTn5	TCEPTn4	TCEPTn3	TCEPTn2	TCEPTn1	TCEPTn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	TCEPTn7	0	R	W	TOUn_0 to TOUn_7 Timer Counter Enable Protect Bits 0: Overwriting of counter enable bit enabled 1: Overwriting of counter enable bit disabled
6	TCEPTn6	0	R	W	
5	TCEPTn5	0	R	W	
4	TCEPTn4	0	R	W	
3	TCEPTn3	0	R	W	
2	TCEPTn2	0	R	W	
1	TCEPTn1	0	R	W	
0	TCEPTn0	0	R	W	

Legend: n = 0 to 4

21.20.6 TOUn Counter Enable Register (TOnCENR)

The TOnCENR register controls the operation of the TOnmCNT counters. To enable a counter by software, set the corresponding timer counter enable protect bit to enable overwriting and then write "1" to the timer counter enable bit. Note that counter operation will not take place even when the timer TOU timer counter enable bit is set to counter enabled unless the TTE bit in the ATU-IIIS master enable register (ATUENR) is set to enabled. To stop counter operation, set the corresponding timer counter enable protect bit to enable overwriting and then write "0" to the timer counter enable bit.

In one-shot output or one-shot PWM output mode, the timer counter enable bit is cleared to "0" automatically when an underflow occurs and the timer stops. Consequently, when the TOnCENR register is read it functions as a status register showing the counter operating status (running or stopped).

TOU0 Counter Enable Register (TO0CENR)	<P4 address: location H'FFFF E608>
TOU1 Counter Enable Register (TO1CENR)	<P4 address: location H'FFFF E708>
TOU2 Counter Enable Register (TO2CENR)	<P4 address: location H'FFFF E808>
TOU3 Counter Enable Register (TO3CENR)	<P4 address: location H'FFFF E908>
TOU4 Counter Enable Register (TO4CENR)	<P4 address: location H'FFFF EA08>

Bit:	7	6	5	4	3	2	1	0
	TCEN Tn7	TCEN Tn6	TCEN Tn5	TCEN Tn4	TCEN Tn3	TCEN Tn2	TCEN Tn1	TCEN Tn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	TCENTn7	0	R	W	TOUn_0 to TOUn_7 Timer Counter Enable Bits 0: Counter disabled 1: Counter enabled
6	TCENTn6	0	R	W	
5	TCENTn5	0	R	W	
4	TCENTn4	0	R	W	
3	TCENTn3	0	R	W	
2	TCENTn2	0	R	W	
1	TCENTn1	0	R	W	
0	TCENTn0	0	R	W	

Legend: n = 0 to 4

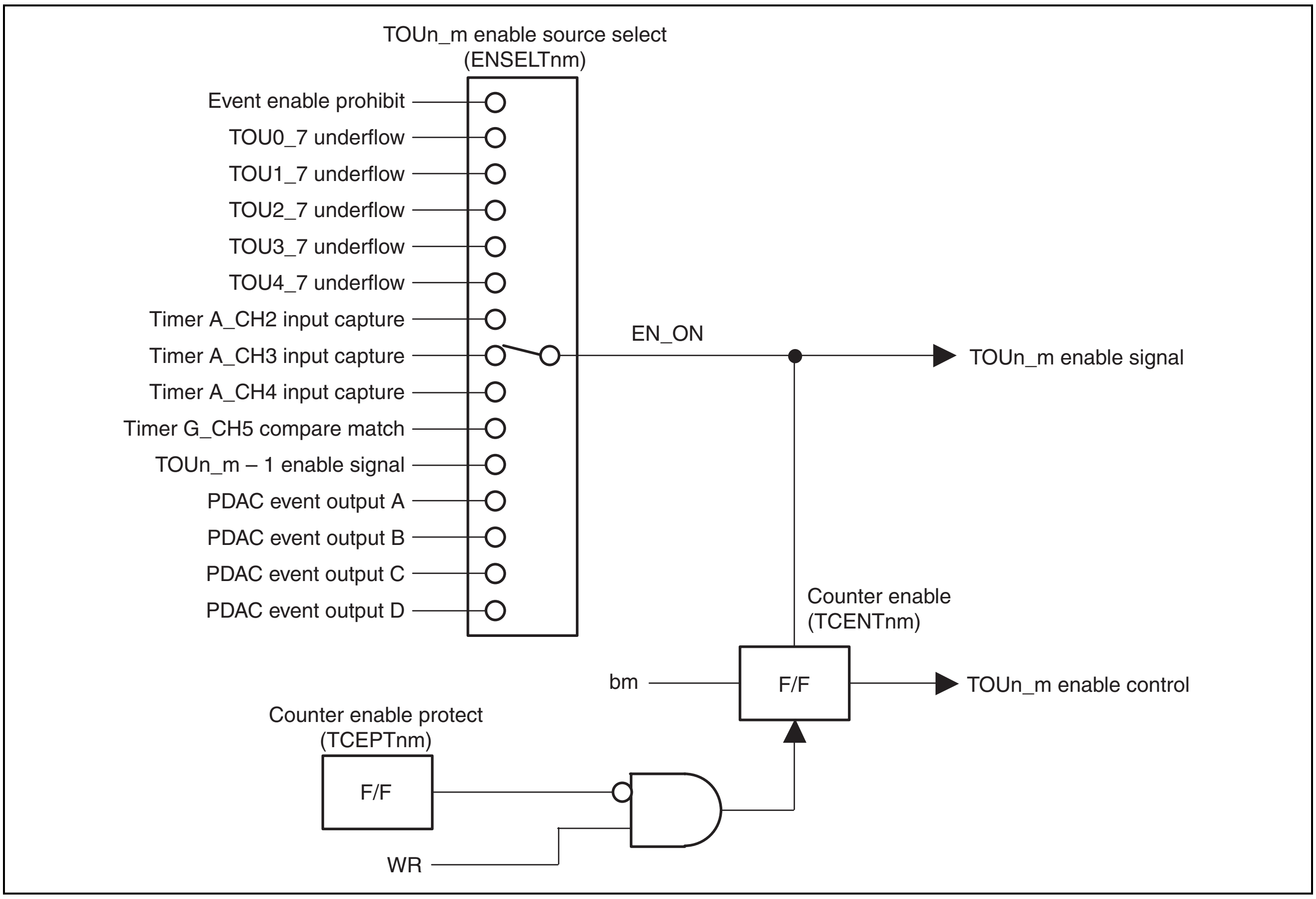


Figure 21.27 TOU Enable Circuit Configuration Diagram

21.20.7 Flip-Flop Output Protect Register for TOUn Short-Circuit Prevention Function (TOnSHFFPR)

The TOnSHFFPR register is used to enable or disable writing to the F/F used by the short-circuit prevention function. Writing to the TOnSHFFDR register has no effect when writes are disabled.

Flip-Flop Output Protect Register for TOU0 Short-Circuit Prevention Function (TO0SHFFPR)	<P4 address: location H'FFFF E605>
Flip-Flop Output Protect Register for TOU1 Short-Circuit Prevention Function (TO1SHFFPR)	<P4 address: location H'FFFF E705>
Flip-Flop Output Protect Register for TOU2 Short-Circuit Prevention Function (TO2SHFFPR)	<P4 address: location H'FFFF E805>
Flip-Flop Output Protect Register for TOU3 Short-Circuit Prevention Function (TO3SHFFPR)	<P4 address: location H'FFFF E905>
Flip-Flop Output Protect Register for TOU4 Short-Circuit Prevention Function (TO4SHFFPR)	<P4 address: location H'FFFF EA05>

Bit:	7	6	5	4	3	2	1	0
	—	—	SHFPTn5	SHFPTn4	SHFPTn3	SHFPTn2	SHFPTn1	SHFPTn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	SHFPTn5	0	R	W	TOUn_0 to TOUn_5 F/F Output Protect Bits for Short-Circuit Prevention Function 0: Writing to short-circuit prevention function F/F output data bits enabled 1: Writing to short-circuit prevention function F/F output data bits disabled
4	SHFPTn4	0	R	W	
3	SHFPTn3	0	R	W	
2	SHFPTn2	0	R	W	
1	SHFPTn1	0	R	W	
0	SHFPTn0	0	R	W	

Legend: n = 0 to 4

21.20.8 Flip-Flop Output Data Register for TOUn Short-Circuit Prevention Function (TOnSHFFDR)

The TOnSHFFDR register is used to specify the output of the F/F used by the short-circuit prevention function.

The contents of the TOnSHFFDR register can be manipulated only when writing is enabled by the setting of the TOnSHFFPR register.

Flip-Flop Output Data Register for TOU0 Short-Circuit Prevention Function (TO0SHFFDR)	<P4 address: location H'FFFF E609>
Flip-Flop Output Data Register for TOU1 Short-Circuit Prevention Function (TO1SHFFDR)	<P4 address: location H'FFFF E709>
Flip-Flop Output Data Register for TOU2 Short-Circuit Prevention Function (TO2SHFFDR)	<P4 address: location H'FFFF E809>
Flip-Flop Output Data Register for TOU3 Short-Circuit Prevention Function (TO3SHFFDR)	<P4 address: location H'FFFF E909>
Flip-Flop Output Data Register for TOU4 Short-Circuit Prevention Function (TO4SHFFDR)	<P4 address: location H'FFFF EA09>

Bit:	7	6	5	4	3	2	1	0
	—	—	SHFD Tn5	SHFD Tn4	SHFD Tn3	SHFD Tn2	SHFD Tn1	SHFD Tn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	SHFDTn5	0	R	W	TOUn_0 to TOUn_5 F/F Output Protect Bits for Short-Circuit Prevention Function 0: Short-circuit prevention function F/F output data = 0 1: short-circuit prevention function F/F output data = 1
4	SHFDTn4	0	R	W	
3	SHFDTn3	0	R	W	
2	SHFDTn2	0	R	W	
1	SHFDTn1	0	R	W	
0	SHFDTn0	0	R	W	

Legend: n = 0 to 4

21.20.9 TOUn Flip-Flop Output Protect Register (TOnFFPR)

The TOnFFPR register is used to enable or disable writing to the output bits of the timer TOU flip-flops (F/F) and output control bits in the TOnOUCR register. Writing to the TOnFFDR or TOnOUCR register has no effect when writes are disabled.

TOU0 Flip-Flop Output Protect Register (TO0FFPR)	<P4 address: location H'FFFF E606>
TOU1 Flip-Flop Output Protect Register (TO1FFPR)	<P4 address: location H'FFFF E706>
TOU2 Flip-Flop Output Protect Register (TO2FFPR)	<P4 address: location H'FFFF E806>
TOU3 Flip-Flop Output ProtectRegister (TO3FFPR)	<P4 address: location H'FFFF E906>
TOU4 Flip-Flop Output ProtectRegister (TO4FFPR)	<P4 address: location H'FFFF EA06>

Bit:	7	6	5	4	3	2	1	0
	FFPTn7	FFPTn6	FFPTn5	FFPTn4	FFPTn3	FFPTn2	FFPTn1	FFPTn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	FFPTn7	0	R	W	TOUn_0 to TOUn_7 F/F Output Protect Bits 0: Writing to F/F output bits and output control bits enabled 1: Writing to F/F output bits and output control bits disabled
6	FFPTn6	0	R	W	
5	FFPTn5	0	R	W	
4	FFPTn4	0	R	W	
3	FFPTn3	0	R	W	
2	FFPTn2	0	R	W	
1	FFPTn1	0	R	W	
0	FFPTn0	0	R	W	

Legend: n = 0 to 4

21.20.10 TOUn Flip-Flop Output Data Register (TOnFFDR)

The TOnFFDR register is used to set the output of the timer TOU flip-flops (F/F). Writing to the TOnOUCR register causes the same value to be written simultaneously to the TOnFFDR register. The TOnFFDR register is writable only when writing is enabled by the setting of the TOnFFPR register described above. Normally the F/F output changes at counter start and underflow, based on the value of the TOnOUCR register, but it is possible to change the F/F output as desired by manipulating the TOnFFDR register (see figure 21.43).

TOU0 Flip-Flop Output Data Register (TO0FFDR)	<P4 address: location H'FFFF E60A>
TOU1 Flip-Flop Output Data Register (TO1FFDR)	<P4 address: location H'FFFF E70A>
TOU2 Flip-Flop Output Data Register (TO2FFDR)	<P4 address: location H'FFFF E80A>
TOU3 Flip-Flop Output Data Register (TO3FFDR)	<P4 address: location H'FFFF E90A>
TOU4 Flip-Flop Output Data Register (TO4FFDR)	<P4 address: location H'FFFF EA0A>

Bit:	7	6	5	4	3	2	1	0
	FFD Tn7	FFD Tn6	FFD Tn5	FFD Tn4	FFD Tn3	FFD Tn2	FFD Tn1	FFD Tn0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	FFDTn7	0	R	W	TOUn_0 to TOUn_7 F/F Output Data Bits 0: F/F output bit = 0 1: F/F output bit = 1
6	FFDTn6	0	R	W	
5	FFDTn5	0	R	W	
4	FFDTn4	0	R	W	
3	FFDTn3	0	R	W	
2	FFDTn2	0	R	W	
1	FFDTn1	0	R	W	
0	FFDTn0	0	R	W	

Legend: n = 0 to 4

21.20.11 TOUn Noise Canceler Control Register (TOnNCCR)

The TOnNCCR register is used to enable or disable the noise canceler function for external input (PWMOFFn) and to select the noise canceler clock.

TOU0 Noise Canceler Control Register (TO0NCCR)	<P4 address: location H'FFFF E60C>
TOU1 Noise Canceler Control Register (TO1NCCR)	<P4 address: location H'FFFF E70C>
TOU3 Noise Canceler Control Register (TO3NCCR)	<P4 address: location H'FFFF E90C>
TOU4 Noise Canceler Control Register (TO4NCCR)	<P4 address: location H'FFFF EA0C>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	NCK Tn	—	—	—	NCE Tn
After Reset:	0	0	0	0	0	0	0	0

Legend: n = 0, 1, 3, 4

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	NOKTn	0	R	W	TOUn Noise Canceler Clock Select Bit This bit selects the count source clock of the TOUn noise canceler counter (TOnNCNT). Either the noise canceler count clock or clock bus 5 may be selected as the count source clock. As the noise canceler count clock, either the Pck clock divided by 128 or the Pck clock can be selected by setting the NCCSEL bit in the ATNCMR register of the common controller. 0: Noise canceler count clock selected as count source clock of TOnNCNT counter 1: Clock bus 5 selected as count source clock of TOnNCNT counter
3 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
0	NCETn	0	R	W	TOUn Noise Canceler Enable Bit This bit enables or disables the noise canceler function for the corresponding external input (PWMOFFn). When this bit is set to "1", at edge detection on the corresponding external input PWMOFFn, processing starts in either premature-transition cancellation mode or minimum time-at-level cancellation mode, according to the setting of the noise cancellation mode register (NCMR) of the common controller. In premature-transition cancellation mode, when a level change of the external input signal is detected, the change is output as the noise-canceled signal. At the same time, the corresponding TOUn noise canceler counter (TOnNCNT) begins counting up. Level changes in the external input signal are masked from this point until a compare match occurs between the noise canceler counter and TOUn noise canceler register (TOnNCR) values. When a compare match occurs, the external input signal level at that point is output as the noise-canceled signal. If an NCETn bit is cleared to "0" while the corresponding TOnNCNT counter is operating, counter operation does not stop until a compare match occurs and level changes in the corresponding external input PWMOFFn continue to be masked. In minimum time-at-level cancellation mode, the corresponding TOUn noise canceler counter (TOnNCNT) begins counting up when a level change of the external input signal is detected. If the level of the external input signal does not change from this point until a compare match occurs with the TOUn noise canceler register (TOnNCR) value, a level change is first output as the noise-canceled signal at the timing of the compare match. If the level of the external input signal changes before a compare match occurs, it is treated as noise, the noise canceler considers no change to have occurred in the external input signal, and the level of the noise-canceled signal does not change. If an NCETn bit is cleared to "0" while the corresponding TOnNCNT counter is operating, counter operation does not stop and noise canceler processing continues until a compare match occurs or the level of the input signal changes. For examples of operation in the two cancellation modes, see figures 21.3 and 21.4. 0: Noise canceler function for PWMOFFn input disabled 1: Noise canceler function for PWMOFFn input enabled

Legend: n = 0, 1, 3, 4

21.20.12 TOUn Noise Canceler Counter (TOnNCNT)

When the noise canceler function is enabled by a noise canceler enable bit (among NCET4 to NCET0) in the TOUn noise canceler control register (TOnNCCR), the counter is incremented when triggered by a signal on the corresponding external input pin (PWMOFFn). Either the noise canceler count clock or clock bus 5 may be selected as the count source clock by means of the corresponding noise canceler clock select bit (among NCKT4 to NCKT0).

Operation proceeds in premature-transition cancellation mode or minimum time-at-level cancellation mode, according to the setting of the NCMT bit in the NCMR register of the common controller.

- Premature-Transition Cancellation Mode

  When the input level of a signal PWMOFFn changes while the corresponding bit among NCET4 to NCET0 is set to "1" and the TOnNCNT counter is stopped, the TOnNCNT counter starts to count up. When the counter value matches that of the TOUn noise canceler register (TOnNCR), the counter value is cleared to "H'00" in synchronization with the next Pck clock cycle.

  The TOnNCNT counter operates regardless of the setting of the TTE bit in the ATUENR register. If a level change occurs at counter start, it is output unchanged as the noise-canceled signal and is subject to edge detection. From that point until the counter value matches that of the TOnNCR register the noise-canceled signal does not change, because all input level changes are masked. When the counter value matches that of the TOnNCR register, the input signal level at that point is output as the noise-canceled signal.

  Even if the NCET bit is cleared to "0" while the counter is operating, counter operation continues until the counter value matches that of the TOnNCR register. The input signal is masked during this period.
- Minimum Time-at-Level Cancellation Mode

  When the input level of a signal PWMOFFn changes while the corresponding bit among NCET4 to NCET0 is set to "1" and the TOnNCNT counter is stopped, the TOnNCNT counter starts to count up. If the input signal level changes during counter operation or the counter value matches that of the TOUn noise canceler register (TOnNCR), the counter value is cleared to "H'00" in synchronization with the next Pck clock cycle.

  The TOnNCNT counter operates regardless of the setting of the TTE bit in the ATUENR register. The noise-canceled signal changes only when the counter value matches that of the TOnNCR register and the level changes at counter start. The noise-canceled signal does not change if counter operation stops before a match occurs with the value of the TOnNCR register, because level changes at the start and end of counter operation are masked.

  Even if the NCET bit is cleared to "0" while the counter is operating, counter operation does not stop and noise canceler processing continues until a compare match occurs or the level of the input signal changes.

TOU0 Noise Canceler Counter (TO0NCNT)	<P4 address: location H'FFFF E60E>
TOU1 Noise Canceler Counter (TO1NCNT)	<P4 address: location H'FFFF E70E>
TOU3 Noise Canceler Counter (TO3NCNT)	<P4 address: location H'FFFF E90E>
TOU4 Noise Canceler Counter (TO4NCNT)	<P4 address: location H'FFFF EA0E>

Bit:	7	6	5	4	3	2	1	0
	TOnNCNT							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TOnNCNT	All 0	R	W	TOUn Noise Canceler Counter Bits These bits store the 8-bit counter value.

Legend: n = 0, 1, 3, 4

21.20.13 TOUn Noise Canceler Register (TOnNCR)

The TOnNCR register is used to set the upper limit value of the TOUn noise canceler counter (TOnNCNT). Noise with a duration of up to 0.82 ms (when Pck = 40 MHz) can be canceled by setting this register to H'FF when Pck clock divided by 128 is selected as the noise canceler clock.

Operation takes place in either premature-transition cancellation mode or minimum time-at-level cancellation mode, according to the setting of the NCMT bit in the NCMR register of the common controller.

- Premature-Transition Cancellation Mode

  In this mode, further changes in the input signal level are masked while the TOnNCNT counter is operating. The values of the TOnNCNT counter and the TOnNCR register are compared constantly. When a compare match occurs, the count value of the TOnNCNT counter is cleared in synchronization with the next Pck clock cycle, counter operation stops, and masking of the input signal cancels.
- Minimum Time-at-Level Cancellation Mode

  The noise canceler is in standby status when the TOnNCNT counter is operating. The values of the TOnNCNT counter and the TOnNCR register are compared constantly. When a compare match occurs, the count value of the TOnNCNT counter is cleared in synchronization with the next Pck clock cycle, counter operation stops, and simultaneously the noise canceler outputs the input signal with the noise removed.

TOU0 Noise Canceler Register (TO0NCR)	<P4 address: location H'FFFF E60F>
TOU1 Noise Canceler Register (TO1NCR)	<P4 address: location H'FFFF E70F>
TOU3 Noise Canceler Register (TO3NCR)	<P4 address: location H'FFFF E90F>
TOU4 Noise Canceler Register (TO4NCR)	<P4 address: location H'FFFF EA0F>

Bit:	7	6	5	4	3	2	1	0
	TOnNCR							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	TOnNCR	All 0	R	W	TOUn Noise Cancellation Time PWMOFFn noise cancellation duration (8-bit compare value)

Legend: n = 0, 1, 3, 4

21.20.14 TOUnPWMOFF Input Processing Register (TOnPOCR)

The TOnPOCR register is used to make settings for PWM output-prohibit control using external pins. For details on the PWM output-prohibit function, see section 21.21.7, PWM Output-Prohibit Function.

TOU0PWMOFF Input Processing Register (TO0POCR)	<P4 address: location H'FFFF E610>
TOU1PWMOFF Input Processing Register (TO1POCR)	<P4 address: location H'FFFF E710>
TOU3PWMOFF Input Processing Register (TO3POCR)	<P4 address: location H'FFFF E910>
TOU4PWMOFF Input Processing Register (TO4POCR)	<P4 address: location H'FFFF EA10>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ATUKEY								—	—	—	—	—	POSTn		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	ATUKEY	All 0	0	W	TOnPOCR Register Write Key Code Bits These bits enable or disable POSTn bit modification. The data written to these bits are not retained. These bits are always read as "0". H'C9: Enable POSTn bit modification. Other than H'C9: Disable POSTn bit modification.
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	POSTn	000	R	W	TOUn PWMOFF Input Processing Control Bits 000: Input disabled 001: Rising edge 010: Falling edge 011: Both edges 10x: "L" level 11x: "H" level

Legend: n = 0, 1, 3, 4

21.20.15 TOUnPWMOFF Function Enable Register (TOnPOER)

The TOnPOER register is used to enable or disable the PWM output-prohibit function using the PWMOFF input pins. The PWM output-prohibit function can be used in all the output modes of timer TOU. It cannot be used when an I/O port is used in a mode other than timer output. For details, see section 21.21.7, PWM Output-Prohibit Function.

TOU0PWMOFF Function Enable Register (TO0POER) <P4 address: location H'FFFF E613>
TOU1PWMOFF Function Enable Register (TO1POER) <P4 address: location H'FFFF E713>
TOU4PWMOFF Function Enable Register (TO4POER) <P4 address: location H'FFFF EA13>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	POEN Tn
After Reset:	0	0	0	0	0	0	0	0

TOU3PWMOFF Function Enable Register (TO3POER) <P4 address: location H'FFFF E913>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	POEN 2Tp	—	—	—	POEN Tn
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	POEN2Tp	0	R	W	TOUp PWMOFF Function Select Bit for Pin Group B Controls whether to enable or disable the PWMOFF function of the pins assigned to pin group B (PH0 to PH15). For details, see table 18.5. Since TOU0, TOU1, and TOU4 are not assigned to pin group B, this bit is reserved in registers TO0POER, TO1POER, and TO4TOER. This bit is always read as "0". The write value should always be "0". 0: PWMOFF function disabled 1: PWMOFF function enabled
3 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	POENTn	0	R	W	TOUn PWMOFF Function Select Bit Controls whether to enable or disable the PWMOFF function of the pins other than pins assigned to pin group B (PH0 to PH15). 0: PWMOFF function disabled 1: PWMOFF function enabled

Legend: n = 0, 1, 3, 4, p = 3

21.20.16 TOUnPWM Output-Prohibit Control Register (TOnPODISCR)

The TOnPODISCR register is used to enable or disable PWM output from pins TOn0 to TOn5. These pins are used to control three-phase PWM output by timer TOU. Three-phase PWM output can be forcibly disabled (high-impedance state) by controlling the TOnPODISCR register. This functionality is available in all the output modes of timer TOU. It cannot be used when an I/O port is used in a mode other than timer output. For details, see section 21.21.7, PWM Output-Prohibit Function. The PODISTn bit can also be read as a status bit indicating the output-prohibit state.

TOU0PWM Output-Prohibit Control Register (TO0PODISCR) <P4 address: location H'FFFF E614>
TOU1PWM Output-Prohibit Control Register (TO1PODISCR) <P4 address: location H'FFFF E714>
TOU4PWM Output-Prohibit Control Register (TO4PODISCR) <P4 address: location H'FFFF EA14>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ATUKEY								—	—	—	—	—	—	—	PODISTn
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TOU3PWM Output-Prohibit Control Register (TO3PODISCR) <P4 address: location H'FFFF E914>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ATUKEY								—	—	—	PODIS2Tp	—	—	—	PODISTn
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	ATUKEY	All 0	0	W	TOnPODISCR Register Write Key Code Bits These bits enable or disable lower bit modification. The data written to these bits are not retained. These bits are always read as "0". H'C9: Enable lower bit modification. Other than H'C9: Disable lower bit modification.
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	PODIS2Tp	0	R	W	TOp0 to TOp5 Output-Prohibit Select Bit for Pin Group B Controls whether to enable or disable the output of the pins assigned to pin group B (PH0 to PH15). For details, see table 18.5. Since TOU0, TOU1, and TOU4 are not assigned to pin group B, this bit is reserved in registers TO0PODISCR, TO1PODISCR, and TO4PODISCR. This bit is always read as "0". The write value should always be "0". 0: Output enabled 1: Output disabled
3 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	PODISTn	0	R	W	TOn0 to TOn5 Output-Prohibit Select Bit Controls whether to enable or disable the output of the pins other than pins assigned to pin group B (PH0 to PH15). 0: Output enabled 1: Output disabled

Legend: n = 0, 1, 3, 4, p = 3

21.20.17 TOUnPWM Output-Prohibit Level Control Register (TOnPOLVCR)

The output-prohibit level select function is used to forcibly disable (high-impedance state) timer output based on the output state of the timer.

The output-prohibit level select function can be used, for example, to determine when the three-phase PWM signals are simultaneously in the ON state. This functionality is available in all the output modes of timer TOU. It cannot be used when an I/O port is used in a mode other than timer output. For details, see section 21.21.7, PWM Output-Prohibit Function.

TOU0PWM Output-Prohibit Level Control Register (TO0POLVCR) <P4 address: location H'FFFF E617>
TOU1PWM Output-Prohibit Level Control Register (TO1POLVCR) <P4 address: location H'FFFF E717>
TOU4PWM Output-Prohibit Level Control Register (TO4POLVCR) <P4 address: location H'FFFF EA17>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	POLV SELTn	POLV ENTn
After Reset:	0	0	0	0	0	0	0	0

TOU3PWM Output-Prohibit Level Control Register (TO3POLVCR) <P4 address: location H'FFFF E917>

Bit:	7	6	5	4	3	2	1	0
	—	—	POLV SEL2Tp	POLV EN2Tp	—	—	POLV SELTn	POLV ENTn
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	POLVSEL2Tp	0	R	W	TOp0 to TOp5 Output-Prohibit Level Select Bit for Pin Group B Selects the output-prohibit level of the pins assigned to pin group B (PH0 to PH15). For details, see table 18.5. Since TOU0, TOU1, and TOU4 are not assigned to pin group B, this bit is reserved in registers TO0POLVCR, TO1POLVCR, and TO4POLVCR. This bit is always read as "0". The write value should always be "0". 0: Output-prohibit level "L" selected 1: Output-prohibit level "H" selected
4	POLVEN2Tp	0	R	W	TOUp Output-Prohibit Level Enable/Disable Select Bit for Pin Group B Controls whether to disable or enable the output-prohibit level selection of the pins assigned to pin group B (PH0 to PH15). For details, see table 18.5. Since TOU0, TOU1, and TOU4 are not assigned to pin group B, this bit is reserved in registers TO0POLVCR, TO1POLVCR, and TO4POLVCR. This bit is always read as "0". The write value should always be "0". 0: Output-prohibit level select disabled 1: Output-prohibit level select enabled
3, 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
1	POLVSELTn	0	R	W	TOn0 to TOn5 Output-Prohibit Level Select Bit Selects the output-prohibit level of the pins other than pins assigned to pin group B (PH0 to PH15). 0: Output-prohibit level "L" selected 1: Output-prohibit level "H" selected
0	POLVENTn	0	R	W	TOUn Output-Prohibit Level Enable/Disable Select Bit Controls whether to enable or disable the output-prohibit level selection of the pins other than pins assigned to pin group B (PH0 to PH15). 0: Output-prohibit level select disabled 1: Output-prohibit level select enabled

Legend: n = 0, 1, 3, 4, p = 3

(1) POLVSELTn (Output-Prohibit Level Select) Bit (Bit 1) and POLVSEL2Tp (Output-Prohibit Level Select for Pin Group B) Bit (Bit 5)

This bit specifies at what output level ("L" or "H" level) output is disabled.

"L" level output is disabled when this bit is cleared to "0", and "H" level output is disabled when it is set to "1".

The conditions under which timer output is turned off according to the timer output state are described below.

- POLVSELTn = "0"

  Output from TOn0 to TOn5 (output pins TOUn_0 to TOUn_5) is disabled when any one of the following conditions is satisfied:

  Output from TOn0 (output pin TOUn_0) and TOn1 (output pin TOUn_1) are both "L" level.

  Output from TOn2 (output pin TOUn_2) and TOn3 (output pin TOUn_3) are both "L" level.

  Output from TOn4 (output pin TOUn_4) and TOn5 (output pin TOUn_5) are both "L" level.
- POLVSELTn = "1"

  Output from TOn0 to TOn5 (output pins TOUn_0 to TOUn_5) is disabled when any one of the following conditions is satisfied:

  Output from TOn0 (output pin TOUn_0) and TOn1 (output pin TOUn_1) are both "H" level.

  Output from TOn2 (output pin TOUn_2) and TOn3 (output pin TOUn_3) are both "H" level.

  Output from TOn4 (output pin TOUn_4) and TOn5 (output pin TOUn_5) are both "H" level.

(2) POLVENTn (Output-Prohibit Level Enable/Disable Select) Bit (Bit 0) and POLVEN2Tp (Output-Prohibit Level Enable/Disable Select for Pin Group B) Bit (Bit 4)

This bit enables or disables the output-prohibit function using the level selected by the POLVSELTn bit. The output-prohibit level selected by the POLVSELTn bit is enabled when this bit is set to "1", the output-prohibit level selected by the POLVSELTn bit is disabled when it is cleared to "0".

Legend: n = 0, 1, 3, 4

21.20.18 TOUnm Mode Control Register (TOnmMCR)

The TOnmMCR register is used to select the operation mode and enable source for each TOU channel, and to enable or disable generation of an interrupt or DMA transfer request at the first underflow in one-shot PWM mode.

Note: • Only set or update the TOnmMCR register while the TTE bit in the ATUENR register and the TCENTnm bit in the TOnCENR register are both cleared to "0".

- Timer TOU0

TOU00 Mode Control Register (TO00MCR)	<P4 address: location H'FFFF E620>
TOU01 Mode Control Register (TO01MCR)	<P4 address: location H'FFFF E630>
TOU02 Mode Control Register (TO02MCR)	<P4 address: location H'FFFF E640>
TOU03 Mode Control Register (TO03MCR)	<P4 address: location H'FFFF E650>
TOU04 Mode Control Register (TO04MCR)	<P4 address: location H'FFFF E660>
TOU05 Mode Control Register (TO05MCR)	<P4 address: location H'FFFF E670>
TOU06 Mode Control Register (TO06MCR)	<P4 address: location H'FFFF E680>
TOU07 Mode Control Register (TO07MCR)	<P4 address: location H'FFFF E690>

Bit:	7	6	5	4	3	2	1	0
	—	UD1ET0m	MDSELT0m		ENSELT0m			
After Reset:	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU1

TOU10 Mode Control Register (TO10MCR)	<P4 address: location H'FFFF E720>
TOU11 Mode Control Register (TO11MCR)	<P4 address: location H'FFFF E730>
TOU12 Mode Control Register (TO12MCR)	<P4 address: location H'FFFF E740>
TOU13 Mode Control Register (TO13MCR)	<P4 address: location H'FFFF E750>
TOU14 Mode Control Register (TO14MCR)	<P4 address: location H'FFFF E760>
TOU15 Mode Control Register (TO15MCR)	<P4 address: location H'FFFF E770>
TOU16 Mode Control Register (TO16MCR)	<P4 address: location H'FFFF E780>
TOU17 Mode Control Register (TO17MCR)	<P4 address: location H'FFFF E790>

Bit:	7	6	5	4	3	2	1	0
	—	UD1ET1m	MDSELT1m		ENSELT1m			
After Reset:	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU2

TOU20 Mode Control Register (TO20MCR)	<P4 address: location H'FFFF E820>
TOU21 Mode Control Register (TO21MCR)	<P4 address: location H'FFFF E830>
TOU22 Mode Control Register (TO22MCR)	<P4 address: location H'FFFF E840>
TOU23 Mode Control Register (TO23MCR)	<P4 address: location H'FFFF E850>
TOU24 Mode Control Register (TO24MCR)	<P4 address: location H'FFFF E860>
TOU25 Mode Control Register (TO25MCR)	<P4 address: location H'FFFF E870>
TOU26 Mode Control Register (TO26MCR)	<P4 address: location H'FFFF E880>
TOU27 Mode Control Register (TO27MCR)	<P4 address: location H'FFFF E890>

Bit:	7	6	5	4	3	2	1	0
	—	UD1ET2m	MDSELT2m		ENSELT2m			
After Reset:	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU3

TOU30 Mode Control Register (TO30MCR)	<P4 address: location H'FFFF E920>
TOU31 Mode Control Register (TO31MCR)	<P4 address: location H'FFFF E930>
TOU32 Mode Control Register (TO32MCR)	<P4 address: location H'FFFF E940>
TOU33 Mode Control Register (TO33MCR)	<P4 address: location H'FFFF E950>
TOU34 Mode Control Register (TO34MCR)	<P4 address: location H'FFFF E960>
TOU35 Mode Control Register (TO35MCR)	<P4 address: location H'FFFF E970>
TOU36 Mode Control Register (TO36MCR)	<P4 address: location H'FFFF E980>
TOU37 Mode Control Register (TO37MCR)	<P4 address: location H'FFFF E990>

Bit:	7	6	5	4	3	2	1	0
	—	UD1ET3m	MDSELT3m		ENSELT3m			
After Reset:	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU4

TOU40 Mode Control Register (TO40MCR)	<P4 address: location H'FFFF EA20>
TOU41 Mode Control Register (TO41MCR)	<P4 address: location H'FFFF EA30>
TOU42 Mode Control Register (TO42MCR)	<P4 address: location H'FFFF EA40>
TOU43 Mode Control Register (TO43MCR)	<P4 address: location H'FFFF EA50>
TOU44 Mode Control Register (TO44MCR)	<P4 address: location H'FFFF EA60>
TOU45 Mode Control Register (TO45MCR)	<P4 address: location H'FFFF EA70>
TOU46 Mode Control Register (TO46MCR)	<P4 address: location H'FFFF EA80>
TOU47 Mode Control Register (TO47MCR)	<P4 address: location H'FFFF EA90>

Bit:	7	6	5	4	3	2	1	0
	—	UD1ET4m	MDSELT4m		ENSELT4m			
After Reset:	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6	UD1ETnm	0	R	W	TOUn_m One-Shot PWM Mode Interrupt/DMA Transfer Request Generation Enable/Disable Bit This bit enables or disables generation of an interrupt or DMA transfer request at the first underflow when one-shot PWM mode is selected by the setting of the MDSELTnm register. 0: Generation of interrupt/DMA transfer request at first underflow disabled 1: Generation of interrupt/DMA transfer request at first underflow enabled
5, 4	MDSELTnm	00	R	W	TOUn_m Operating Mode Select Bits 00: One-shot output mode 01: One-shot PWM output mode 10: Continuous output mode 11: PWM output mode

Bit	Abbreviation	After Reset	R	W	Description
3 to 0	ENSELTnm	0000	R	W	TOUn_m Enable Source Select Bits These bits are used to select the enable source for each timer TOU channel. Note that counter operation will not occur, even if an enable source setting other than event enable disabled is selected for a timer TOU channel, unless the TTE bit in the ATU-IIIS master enable register (ATUENR) is set to enable. 0000: Event enable disabled 0001: TOU0_7 underflow 0010: TOU1_7 underflow 0011: TOU2_7 underflow 0100: TOU3_7 underflow 0101: TOU4_7 underflow 0110: Timer A channel 2 input capture 0111: Timer A channel 3 input capture 1000: Timer A channel 4 input capture 1001: Setting prohibited 1010: Timer G channel 5 compare match 1011: Previous TOU channel (TOUn_0: setting prohibited, other than TOUn_0 = TOUn_m-1) 1100: PDAC event output signal A 1101: PDAC event output signal B 1110: PDAC event output signal C 1111: PDAC event output signal D

Note: • Only set or update the TOnmMCR register while the TTE bit in the ATUENR register and the TCENTnm bit in the TOnCENR register are both cleared to "0".

21.20.19 TOUnm Counter (TOnmCNT)

Each TOnmCNT counter implements one of timers TOU00 to TOU47. The function of this register differs depending on the timer's operation mode.

(1) TOU00 to TOU47 Timer Counter in One-Shot Output/Continuous Output Mode

In one-shot output or continuous output mode, TOnmCNT operates as a 24-bit down counter. After the timer is enabled (by writing to the counter enable bit by software or by the occurrence of the event matching the setting of the enable source select bits), the counter starts operating in synchronization with the count clock. Bits 31 to 24 are ignored.

In PWM output or one-shot PWM output mode, TOnmCNT operates as a 16-bit down counter; only 16 of its bits (bits 15 to 0) are valid. For details, see section 21.20.19 (2), TOU00 to TOU47 Timer Counter in PWM Output/One-Shot PWM Output Mode.

- Timer TOU0

TOU00 Counter (TO00CNT)	<P4 address: location H'FFFF E624>
TOU01 Counter (TO01CNT)	<P4 address: location H'FFFF E634>
TOU02 Counter (TO02CNT)	<P4 address: location H'FFFF E644>
TOU03 Counter (TO03CNT)	<P4 address: location H'FFFF E654>
TOU04 Counter (TO04CNT)	<P4 address: location H'FFFF E664>
TOU05 Counter (TO05CNT)	<P4 address: location H'FFFF E674>
TOU06 Counter (TO06CNT)	<P4 address: location H'FFFF E684>
TOU07 Counter (TO07CNT)	<P4 address: location H'FFFF E694>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO0mCNT							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO0mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU1

TOU10 Counter (TO10CNT)	<P4 address: location H'FFFF E724>
TOU11 Counter (TO11CNT)	<P4 address: location H'FFFF E734>
TOU12 Counter (TO12CNT)	<P4 address: location H'FFFF E744>
TOU13 Counter (TO13CNT)	<P4 address: location H'FFFF E754>
TOU14 Counter (TO14CNT)	<P4 address: location H'FFFF E764>
TOU15 Counter (TO15CNT)	<P4 address: location H'FFFF E774>
TOU16 Counter (TO16CNT)	<P4 address: location H'FFFF E784>
TOU17 Counter (TO17CNT)	<P4 address: location H'FFFF E794>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO1mCNT							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO1mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU2

TOU20 Counter (TO20CNT)	<P4 address: location H'FFFF E824>
TOU21 Counter (TO21CNT)	<P4 address: location H'FFFF E834>
TOU22 Counter (TO22CNT)	<P4 address: location H'FFFF E844>
TOU23 Counter (TO23CNT)	<P4 address: location H'FFFF E854>
TOU24 Counter (TO24CNT)	<P4 address: location H'FFFF E864>
TOU25 Counter (TO25CNT)	<P4 address: location H'FFFF E874>
TOU26 Counter (TO26CNT)	<P4 address: location H'FFFF E884>
TOU27 Counter (TO27CNT)	<P4 address: location H'FFFF E894>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO2mCNT							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO2mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU3

TOU30 Counter (TO30CNT)	<P4 address: location H'FFFF E924>
TOU31 Counter (TO31CNT)	<P4 address: location H'FFFF E934>
TOU32 Counter (TO32CNT)	<P4 address: location H'FFFF E944>
TOU33 Counter (TO33CNT)	<P4 address: location H'FFFF E954>
TOU34 Counter (TO34CNT)	<P4 address: location H'FFFF E964>
TOU35 Counter (TO35CNT)	<P4 address: location H'FFFF E974>
TOU36 Counter (TO36CNT)	<P4 address: location H'FFFF E984>
TOU37 Counter (TO37CNT)	<P4 address: location H'FFFF E994>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO3mCNT							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO3mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU4

TOU40 Counter (TO40CNT)	<P4 address: location H'FFFF EA24>
TOU41 Counter (TO41CNT)	<P4 address: location H'FFFF EA34>
TOU42 Counter (TO42CNT)	<P4 address: location H'FFFF EA44>
TOU43 Counter (TO43CNT)	<P4 address: location H'FFFF EA54>
TOU44 Counter (TO44CNT)	<P4 address: location H'FFFF EA64>
TOU45 Counter (TO45CNT)	<P4 address: location H'FFFF EA74>
TOU46 Counter (TO46CNT)	<P4 address: location H'FFFF EA84>
TOU47 Counter (TO47CNT)	<P4 address: location H'FFFF EA94>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO4mCNT							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO4mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
23 to 0	TOnmCNT	All 0	R	W	24-bit counter value

Note: • Always access this register in longword (32-bit units).

Legend: n = 0 to 4, m = 0 to 7

(2) TOU00 to TOU47 Timer Counter in PWM Output/One-Shot PWM Output Mode

In PWM output or one-shot PWM output mode, TOnmCNT operates as a 16-bit down counter. After the timer is enabled (by writing to the counter enable bit by software or by the occurrence of the event matching the setting of the enable source select bits), the counter starts operating in synchronization with the count clock. Bits 31 to 16 are ignored.

In one-shot output or continuous output mode, TOnmCNT operates as a 24-bit down counter by using eight additional bits (bits 23 to 0). For details, see section 21.20.19 (1), TOU00 to TOU47 Timer Counter in One-Shot Output/Continuous Output Mode.

- Timer TOU0

TOU00 Counter (TO00CNT)	<P4 address: location H'FFFF E624>
TOU01 Counter (TO01CNT)	<P4 address: location H'FFFF E634>
TOU02 Counter (TO02CNT)	<P4 address: location H'FFFF E644>
TOU03 Counter (TO03CNT)	<P4 address: location H'FFFF E654>
TOU04 Counter (TO04CNT)	<P4 address: location H'FFFF E664>
TOU05 Counter (TO05CNT)	<P4 address: location H'FFFF E674>
TOU06 Counter (TO06CNT)	<P4 address: location H'FFFF E684>
TOU07 Counter (TO07CNT)	<P4 address: location H'FFFF E694>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO0mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU1

TOU10 Counter (TO10CNT)	<P4 address: location H'FFFF E724>
TOU11 Counter (TO11CNT)	<P4 address: location H'FFFF E734>
TOU12 Counter (TO12CNT)	<P4 address: location H'FFFF E744>
TOU13 Counter (TO13CNT)	<P4 address: location H'FFFF E754>
TOU14 Counter (TO14CNT)	<P4 address: location H'FFFF E764>
TOU15 Counter (TO15CNT)	<P4 address: location H'FFFF E774>
TOU16 Counter (TO16CNT)	<P4 address: location H'FFFF E784>
TOU17 Counter (TO17CNT)	<P4 address: location H'FFFF E794>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO1mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU2

TOU20 Counter (TO20CNT)	<P4 address: location H'FFFF E824>
TOU21 Counter (TO21CNT)	<P4 address: location H'FFFF E834>
TOU22 Counter (TO22CNT)	<P4 address: location H'FFFF E844>
TOU23 Counter (TO23CNT)	<P4 address: location H'FFFF E854>
TOU24 Counter (TO24CNT)	<P4 address: location H'FFFF E864>
TOU25 Counter (TO25CNT)	<P4 address: location H'FFFF E874>
TOU26 Counter (TO26CNT)	<P4 address: location H'FFFF E884>
TOU27 Counter (TO27CNT)	<P4 address: location H'FFFF E894>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO2mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU3

TOU30 Counter (TO30CNT)	<P4 address: location H'FFFF E924>
TOU31 Counter (TO31CNT)	<P4 address: location H'FFFF E934>
TOU32 Counter (TO32CNT)	<P4 address: location H'FFFF E944>
TOU33 Counter (TO33CNT)	<P4 address: location H'FFFF E954>
TOU34 Counter (TO34CNT)	<P4 address: location H'FFFF E964>
TOU35 Counter (TO35CNT)	<P4 address: location H'FFFF E974>
TOU36 Counter (TO36CNT)	<P4 address: location H'FFFF E984>
TOU37 Counter (TO37CNT)	<P4 address: location H'FFFF E994>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO3mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU4

TOU40 Counter (TO40CNT)	<P4 address: location H'FFFF EA24>
TOU41 Counter (TO41CNT)	<P4 address: location H'FFFF EA34>
TOU42 Counter (TO42CNT)	<P4 address: location H'FFFF EA44>
TOU43 Counter (TO43CNT)	<P4 address: location H'FFFF EA54>
TOU44 Counter (TO44CNT)	<P4 address: location H'FFFF EA64>
TOU45 Counter (TO45CNT)	<P4 address: location H'FFFF EA74>
TOU46 Counter (TO46CNT)	<P4 address: location H'FFFF EA84>
TOU47 Counter (TO47CNT)	<P4 address: location H'FFFF EA94>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO4mCNT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 16	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
15 to 0	TOnmCNT	All 0	R	W	16-bit counter value

Note: • Always access this register in longword (32-bit units).

Legend: n = 0 to 4, m = 0 to 7

21.20.20 TOUnm Reload Register (TOnmRLD)

The TOnmRLD register is used to reload data in the TOnmCNT register. The function of this register differs depending on the timer's operation mode.

(1) TOU00 to TOU47 Reload Register in One-Shot Output/Continuous Output Mode

In one-shot output or continuous output mode, TOnmRLD operates as a 24-bit reload register. The value set in the 24 bits (bits 23 to 0) of this register is reloaded in the counter. Bits 31 to 24 are ignored.

The contents of the reload register are loaded in the counter at the following times, in synchronization with the count clock:

- In one-shot output mode, at the next cycle after the counter is enabled
- In continuous output mode, at the cycle when the counter underflows

Data is not loaded in the counter at the point in time when data is written to the reload register.

In PWM output or one-shot PWM output mode, TOnmRLD operates as a 16-bit reload 0 register and 16-bit reload 1 register. For details, see section 21.20.20 (2), TOU00 to TOU47 Reload Register in PWM Output/One-Shot PWM Output Mode.

- Timer TOU0

TOU00 Reload Register (TO00RLD)	<P4 address: location H'FFFF E628>
TOU01 Reload Register (TO01RLD)	<P4 address: location H'FFFF E638>
TOU02 Reload Register (TO02RLD)	<P4 address: location H'FFFF E648>
TOU03 Reload Register (TO03RLD)	<P4 address: location H'FFFF E658>
TOU04 Reload Register (TO04RLD)	<P4 address: location H'FFFF E668>
TOU05 Reload Register (TO05RLD)	<P4 address: location H'FFFF E678>
TOU06 Reload Register (TO06RLD)	<P4 address: location H'FFFF E688>
TOU07 Reload Register (TO07RLD)	<P4 address: location H'FFFF E698>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO0mRLD							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO0mRLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU1

TOU10 Reload Register (TO10RLD)	<P4 address: location H'FFFF E728>
TOU11 Reload Register (TO11RLD)	<P4 address: location H'FFFF E738>
TOU12 Reload Register (TO12RLD)	<P4 address: location H'FFFF E748>
TOU13 Reload Register (TO13RLD)	<P4 address: location H'FFFF E758>
TOU14 Reload Register (TO14RLD)	<P4 address: location H'FFFF E768>
TOU15 Reload Register (TO15RLD)	<P4 address: location H'FFFF E778>
TOU16 Reload Register (TO16RLD)	<P4 address: location H'FFFF E788>
TOU17 Reload Register (TO17RLD)	<P4 address: location H'FFFF E798>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO1mRLD							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO1mRLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU2

TOU20 Reload Register (TO20RLD)	<P4 address: location H'FFFF E828>
TOU21 Reload Register (TO21RLD)	<P4 address: location H'FFFF E838>
TOU22 Reload Register (TO22RLD)	<P4 address: location H'FFFF E848>
TOU23 Reload Register (TO23RLD)	<P4 address: location H'FFFF E858>
TOU24 Reload Register (TO24RLD)	<P4 address: location H'FFFF E868>
TOU25 Reload Register (TO25RLD)	<P4 address: location H'FFFF E878>
TOU26 Reload Register (TO26RLD)	<P4 address: location H'FFFF E888>
TOU27 Reload Register (TO27RLD)	<P4 address: location H'FFFF E898>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO2mRLD							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO2mRLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU3

TOU30 Reload Register (TO30RLD)	<P4 address: location H'FFFF E928>
TOU31 Reload Register (TO31RLD)	<P4 address: location H'FFFF E938>
TOU32 Reload Register (TO32RLD)	<P4 address: location H'FFFF E948>
TOU33 Reload Register (TO33RLD)	<P4 address: location H'FFFF E958>
TOU34 Reload Register (TO34RLD)	<P4 address: location H'FFFF E968>
TOU35 Reload Register (TO35RLD)	<P4 address: location H'FFFF E978>
TOU36 Reload Register (TO36RLD)	<P4 address: location H'FFFF E988>
TOU37 Reload Register (TO37RLD)	<P4 address: location H'FFFF E998>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO3mRLD							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO3mRLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU4

TOU40 Reload Register (TO40RLD)	<P4 address: location H'FFFF EA28>
TOU41 Reload Register (TO41RLD)	<P4 address: location H'FFFF EA38>
TOU42 Reload Register (TO42RLD)	<P4 address: location H'FFFF EA48>
TOU43 Reload Register (TO43RLD)	<P4 address: location H'FFFF EA58>
TOU44 Reload Register (TO44RLD)	<P4 address: location H'FFFF EA68>
TOU45 Reload Register (TO45RLD)	<P4 address: location H'FFFF EA78>
TOU46 Reload Register (TO46RLD)	<P4 address: location H'FFFF EA88>
TOU47 Reload Register (TO47RLD)	<P4 address: location H'FFFF EA98>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TO4mRLD							
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO4mRLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
23 to 0	TOnmRLD	All 0	R	W	24-bit reload register value

Note: • Always access this register in longword (32-bit units).

Legend: n = 0 to 4, m = 0 to 7

(2) TOU00 to TOU47 Reload Register in PWM Output/One-Shot PWM Output Mode

In PWM output or one-shot PWM output mode, TOnmRLD operates as a 16-bit reload 0 register and 16-bit reload 1 register. The values set in the 16 bits of the reload 0 register and reload 1 register are loaded when the counter is enabled.

The contents of the reload registers are loaded in the counter at the following times, in synchronization with the count clock:

- At the next cycle after the counter is enabled
- In PWM output mode, at the cycle when the counter underflows from the value set in the reload register

Data is not loaded in the counter at the point in time when data is written to the reload registers.

Counting of non-inverted PWM output (0% or 100% duty cycle) can be performed by setting a reload register to H'FFFF. For details, see section 21.21.5, 0% or 100% Duty Cycle in PWM Output/One-Shot PWM Output Mode.

In one-shot output or continuous output mode, the reload 0 register and reload 1 register together operate as a 24-bit (bits 23 to 0) reload register. For details, see section 21.20.20 (1), TOU00 to TOU47 Reload Register in One-Shot Output/Continuous Output Mode.

- Timer TOU0

TOU00 Reload Register (TO00RLD)	<P4 address: location H'FFFF E628>
TOU01 Reload Register (TO01RLD)	<P4 address: location H'FFFF E638>
TOU02 Reload Register (TO02RLD)	<P4 address: location H'FFFF E648>
TOU03 Reload Register (TO03RLD)	<P4 address: location H'FFFF E658>
TOU04 Reload Register (TO04RLD)	<P4 address: location H'FFFF E668>
TOU05 Reload Register (TO05RLD)	<P4 address: location H'FFFF E678>
TOU06 Reload Register (TO06RLD)	<P4 address: location H'FFFF E688>
TOU07 Reload Register (TO07RLD)	<P4 address: location H'FFFF E698>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TO0mRLD1															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO0mRLD0															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU1

TOU10 Reload Register (TO10RLD)	<P4 address: location H'FFFF E728>
TOU11 Reload Register (TO11RLD)	<P4 address: location H'FFFF E738>
TOU12 Reload Register (TO12RLD)	<P4 address: location H'FFFF E748>
TOU13 Reload Register (TO13RLD)	<P4 address: location H'FFFF E758>
TOU14 Reload Register (TO14RLD)	<P4 address: location H'FFFF E768>
TOU15 Reload Register (TO15RLD)	<P4 address: location H'FFFF E778>
TOU16 Reload Register (TO16RLD)	<P4 address: location H'FFFF E788>
TOU17 Reload Register (TO17RLD)	<P4 address: location H'FFFF E798>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TO1mRLD1															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO1mRLD0															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU2

TOU20 Reload Register (TO20RLD)	<P4 address: location H'FFFF E828>
TOU21 Reload Register (TO21RLD)	<P4 address: location H'FFFF E838>
TOU22 Reload Register (TO22RLD)	<P4 address: location H'FFFF E848>
TOU23 Reload Register (TO23RLD)	<P4 address: location H'FFFF E858>
TOU24 Reload Register (TO24RLD)	<P4 address: location H'FFFF E868>
TOU25 Reload Register (TO25RLD)	<P4 address: location H'FFFF E878>
TOU26 Reload Register (TO26RLD)	<P4 address: location H'FFFF E888>
TOU27 Reload Register (TO27RLD)	<P4 address: location H'FFFF E898>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TO2mRLD1															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO2mRLD0															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU3

TOU30 Reload Register (TO30RLD)	<P4 address: location H'FFFF E928>
TOU31 Reload Register (TO31RLD)	<P4 address: location H'FFFF E938>
TOU32 Reload Register (TO32RLD)	<P4 address: location H'FFFF E948>
TOU33 Reload Register (TO33RLD)	<P4 address: location H'FFFF E958>
TOU34 Reload Register (TO34RLD)	<P4 address: location H'FFFF E968>
TOU35 Reload Register (TO35RLD)	<P4 address: location H'FFFF E978>
TOU36 Reload Register (TO36RLD)	<P4 address: location H'FFFF E988>
TOU37 Reload Register (TO37RLD)	<P4 address: location H'FFFF E998>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TO3mRLD1															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO3mRLD0															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

- Timer TOU4

TOU40 Reload Register (TO40RLD)	<P4 address: location H'FFFF EA28>
TOU41 Reload Register (TO41RLD)	<P4 address: location H'FFFF EA38>
TOU42 Reload Register (TO42RLD)	<P4 address: location H'FFFF EA48>
TOU43 Reload Register (TO43RLD)	<P4 address: location H'FFFF EA58>
TOU44 Reload Register (TO44RLD)	<P4 address: location H'FFFF EA68>
TOU45 Reload Register (TO45RLD)	<P4 address: location H'FFFF EA78>
TOU46 Reload Register (TO46RLD)	<P4 address: location H'FFFF EA88>
TOU47 Reload Register (TO47RLD)	<P4 address: location H'FFFF EA98>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TO4mRLD1															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TO4mRLD0															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Legend: m = 0 to 7

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 16	TOnmRLD1	All 0	R	W	16-bit reload 1 register value
15 to 0	TOnmRLD0	All 0	R	W	16-bit reload 0 register value

Note: • Always access this register in longword (32-bit units).

Legend: n = 0 to 4, m = 0 to 7

21.21 Operation of Timer TOU

21.21.1 Operation in PWM Output Mode

(1) Overview of PWM Output Mode

In PWM output mode, two reload registers are used to generate a waveform with a user-defined duty cycle. Timer TOU functions as a 16-bit timer in PWM output mode.

When the timer counter is enabled after setting the reload 0 register and reload 1 register to their initial values, the value of the reload 0 register is loaded in the counter in synchronization with the count clock and down counting starts. At the first counter underflow cycle, the contents of the reload 1 register are loaded in the counter, and at each subsequent underflow the counter is reloaded with the value of the reload 0 register or reload 1 register, alternately. The setting of the reload 0 register plus 1 or the setting of the reload 1 register plus 1 are each valid as count values.

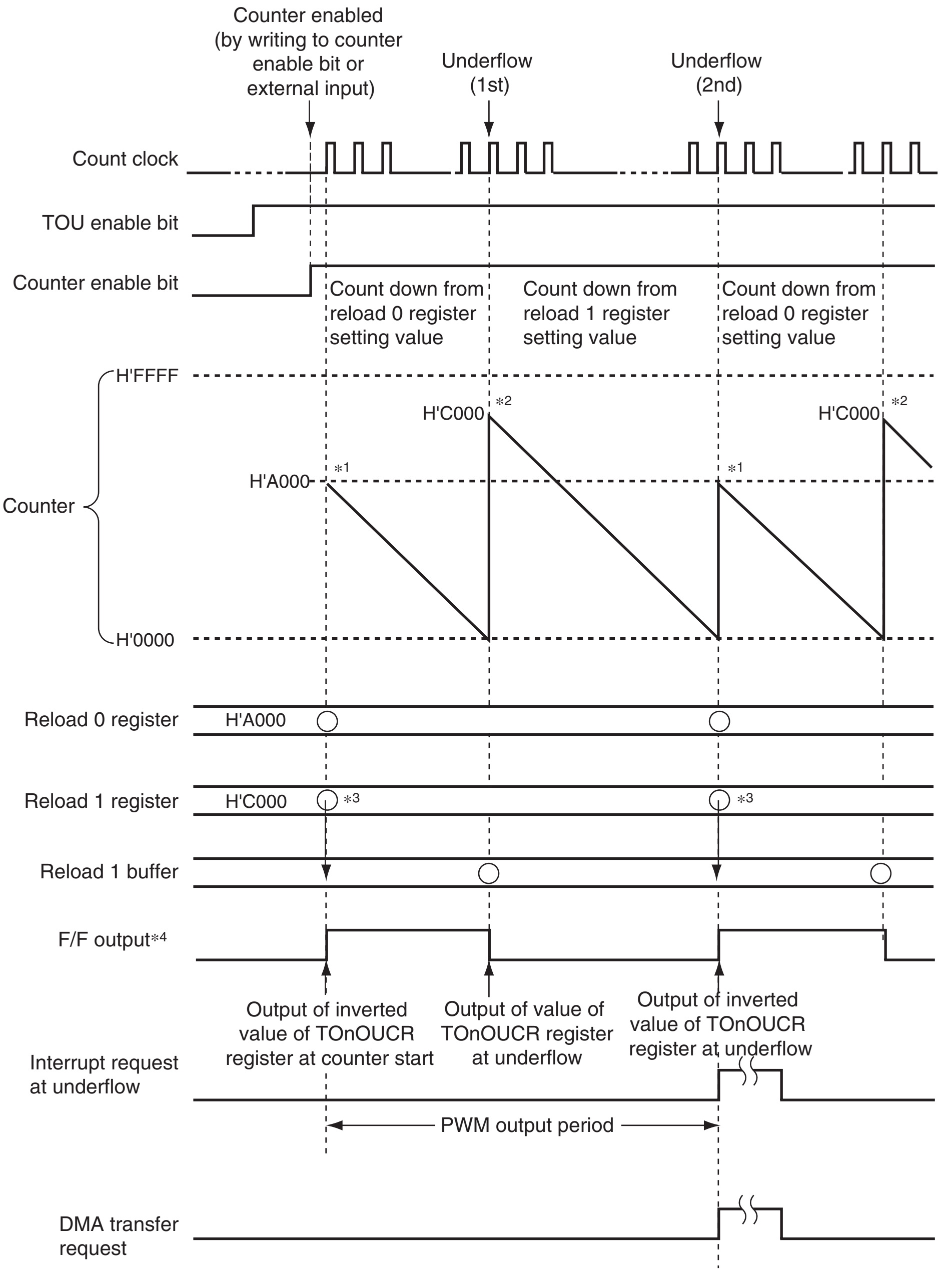
The timer stops operating when the value specifying counter disabled is written to the counter enable bit. (The halt of timer operation is not synchronized with the PWM output period.)

In PWM output mode, the F/F output waveform consists of the inverted value of the TOUn output control register, output at counter start and at each even-numbered underflow, and the value of the TOUn output control register, output at each odd-numbered underflow.

It is also possible to generate an interrupt request or DMA transfer request at each even-numbered underflow after the counter starts.

It is possible to output a PWM waveform consisting of the non-inverted F/F output with a 0% or 100% duty cycle by setting the reload 0 register or reload 1 register to H'FFFF, though interrupt requests are still generated. For details, see section 21.21.5, 0% or 100% Duty Cycle in PWM Output/One-Shot PWM Output Mode.

Note that timer TOU does not support a correction function in PWM output mode.


Counter enabled
(by writing to counter enable bit or external input)
Underflow (1st)
Underflow (2nd)
Count clock
TOU enable bit
Counter enable bit
Count down from reload 0 register setting value
Count down from reload 1 register setting value
Count down from reload 0 register setting value
H'FFFF
H'C000 *2
H'C000 *2
H'A000 *1
*1
Counter
H'0000
Reload 0 register
H'A000
Reload 1 register
H'C000 *3
*3
Reload 1 buffer
F/F output*4
Output of inverted value of TOnOUCR register at counter start
Output of value of TOnOUCR register at underflow
Output of inverted value of TOnOUCR register at underflow
Interrupt request at underflow
PWM output period
DMA transfer request

Notes: This figure does not include detailed timing information.

*1 The value of the reload 0 register is reloaded in the counter.

*2 The value of the reload 1 buffer is reloaded in the counter.

*3 Transfer to the reload 1 buffer occurs when the value of the reload 0 register is reloaded in the counter after updating of the reload 0 register and reload 1 register.

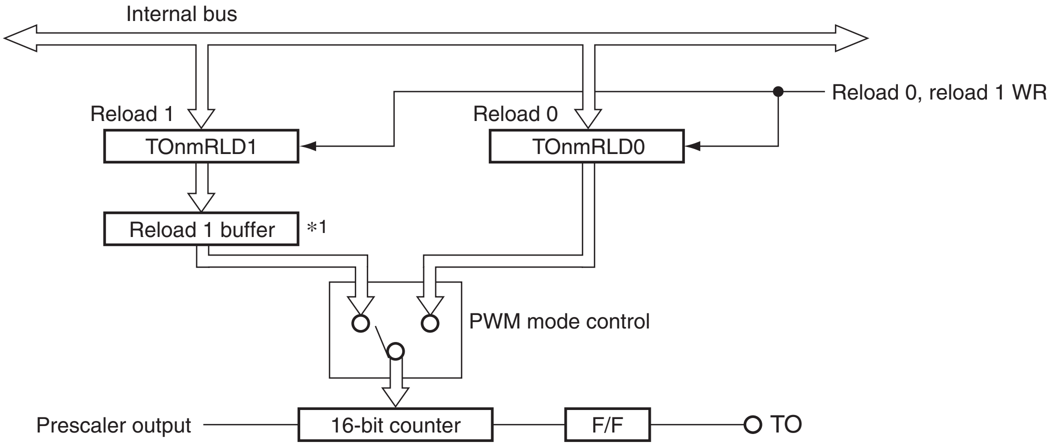
*4 Updating of the reload 0 register and reload 1 register while the timer is operating has no effect on the PWM waveform being output at the time. The updated values take effect from the next PWM period after the reload 0 register update.

Figure 21.28 Operation Example of PWM Output Mode

(2) Reload Register Updating in PWM Output Mode

In PWM output mode, updating of the reload 0 and reload 1 registers takes place simultaneously with data writes to the registers. Also, reading the reload 0 and reload 1 registers always returns the data that was written to them.

During counter operation, when the reload 0 register is reloaded in the counter following updating of the reload 0 and reload 1 registers, data is transferred from the reload 1 register to the reload 1 buffer.



Note: *1 During counter operation, when the reload 0 register is reloaded in the counter following updating of the reload 0 and reload 1 registers, data is transferred from the reload 1 register to the reload 1 buffer.

Figure 21.29 PWM Circuit Diagram

Normally this operation is performed all at once, with a 32-bit access starting at the reload 1 register address. Note that when the PWM period ends by the time of the write to the reload 0 and reload 1 registers during a PWM period rewrite, updating of the PWM period does not occur immediately and takes effect in the next period.

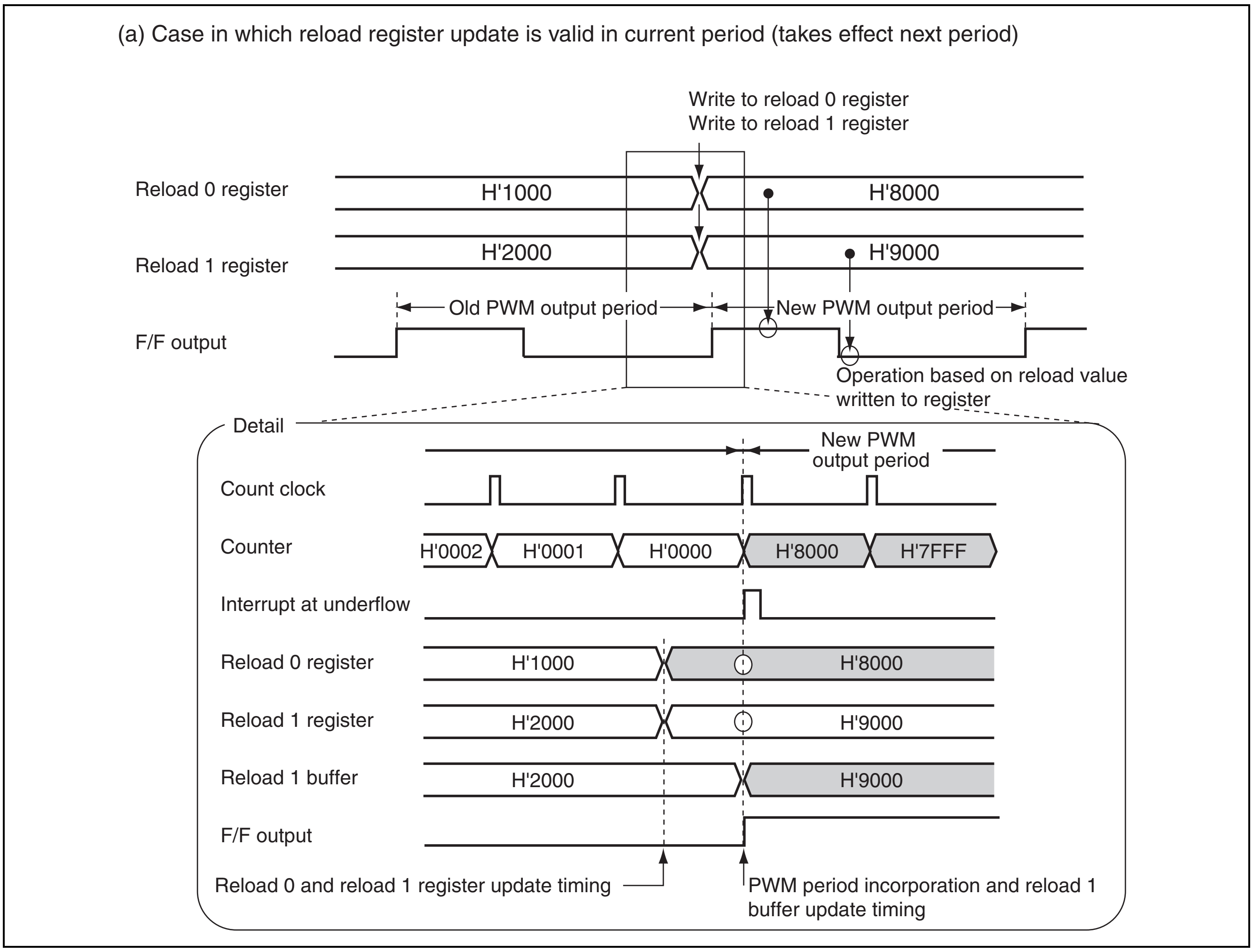


Figure 21.30 Reload 0 and Reload 1 Register Updating in PWM Output Mode

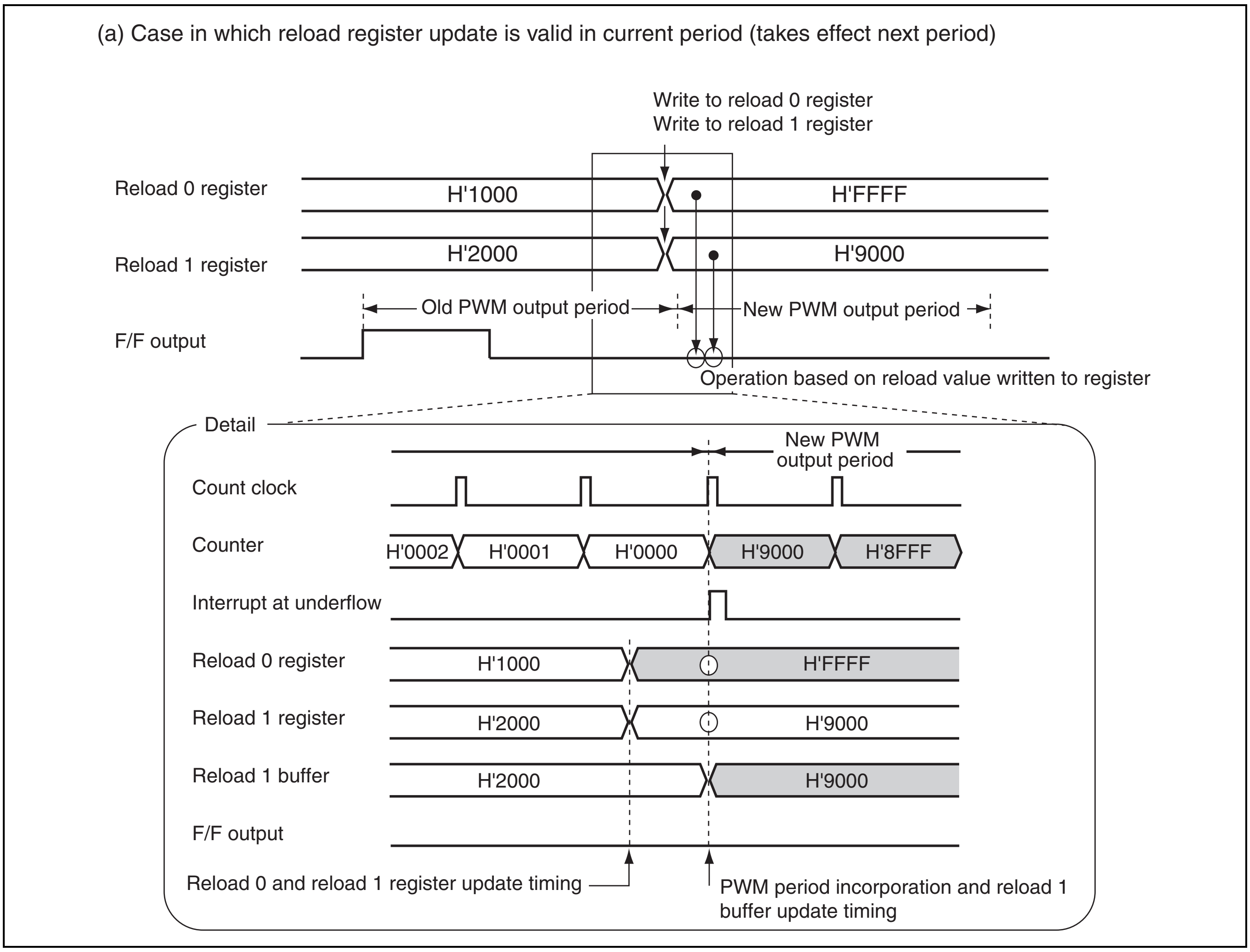


Figure 21.31 Reload 0 and Reload 1 Register Updating in PWM Output Mode (0% Duty Cycle)

(3) Notes on PWM Output Mode

Keep the following points in mind when using PWM output mode:

- If counter enable by external input and a write of the value for counter disabled to the counter enable bit occur in the same clock cycle, the write of the value for counter disabled to the counter enable bit takes precedence.
- When the counter is read in a cycle where an underflow occurs, the value of the reload register is returned. Reading the counter in the clock cycle immediately following returns a value equivalent to the reload register value minus 1.
- Timer operation is synchronized with the count clock, so there is a delay of up to one count clock period between counter enable and F/F output inversion.

21.21.2 Operation in One-Shot PWM Output Mode

(1) Overview of One-Shot PWM Output Mode

In one-shot PWM output mode, two reload registers are used to generate one instance only of a waveform with a user-defined duty cycle. In one-shot PWM output mode, timer TOU operates as a 16-bit timer.

When the timer counter is enabled after setting the reload 0 register and reload 1 register to their initial values, the value of the reload 0 register is loaded in the counter in synchronization with the count clock and down counting starts. At the first counter underflow cycle, the contents of the reload 1 register are loaded in the counter, and at the second underflow the counter stops. The setting of the reload 0 register plus 1 or the setting of the reload 1 register plus 1 are each valid as count values.

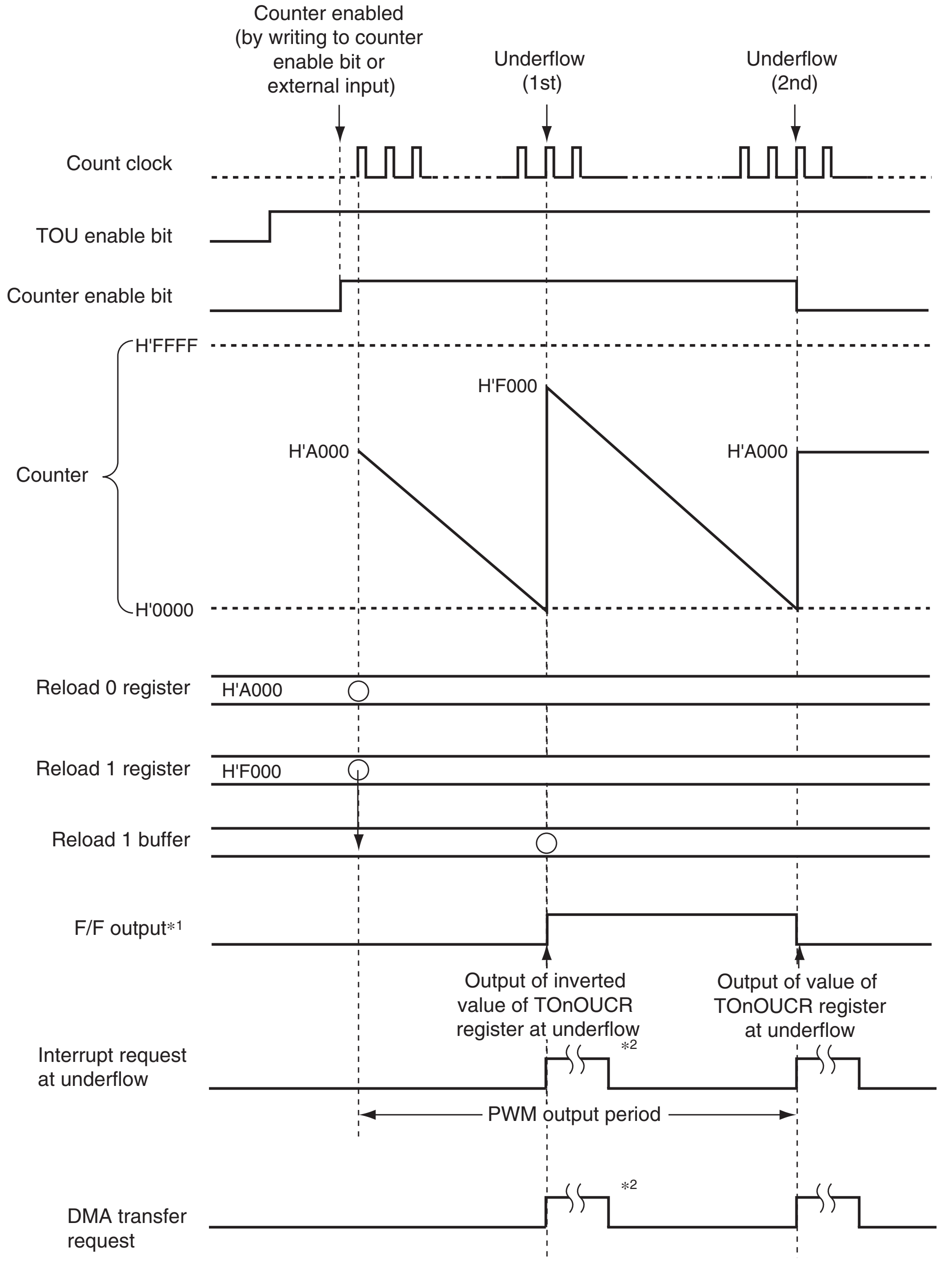
Stopping the timer by software is accomplished by writing the value specifying counter disabled to the counter enable bit. (The halt of timer operation is not synchronized with the PWM output period.)

In one-shot PWM output mode, the F/F output waveform consists of the value of the TOUn output control register, output at counter start and at the second underflow, and the inverted value of the TOUn output control register, output at the first underflow. (In contrast to PWM output mode, the F/F output is not inverted at counter start.)

It is also possible to generate an interrupt request or DMA transfer request at the first and second counter underflows. Generation of the first interrupt request or DMA transfer request can be enabled or disabled by software.

It is possible to output a PWM waveform consisting of the non-inverted F/F output with a 0% or 100% duty cycle by setting the reload 0 register or reload 1 register to H'FFFF, though interrupt requests are still generated. For details, see section 21.21.5, 0% or 100% Duty Cycle in PWM Output/One-Shot PWM Output Mode.

Note that timer TOU does not support a correction function in one-shot PWM output mode.



Notes: This figure does not include detailed timing information.

*1 Updating of the reload 0 register and reload 1 register while the timer is operating has no effect on the PWM waveform being output at the time. The updated values take effect from the next PWM period after the update of the reload 0 register and reload 1 register.

*2 This can be enabled or disabled by software.

Figure 21.32 Operation Example of One-Shot PWM Output Mode

(2) Notes on One-Shot PWM Output Mode

Keep the following points in mind when using one-shot PWM output mode:

- If counter enable by external input and a write of the value for counter disabled to the counter enable bit occur in the same clock cycle, the write of the value for counter disabled to the counter enable bit takes precedence.
- When the counter is read in a cycle where an underflow occurs, the value of the reload register is returned. Reading the counter in the clock cycle immediately following returns a value equivalent to the reload register value minus 1.
- An update of the reload 0 and reload 1 registers during counter operation does not affect the PWM waveform that is currently being output. The update takes effect from the PWM period at the next counter start.

21.21.3 Operation in One-Shot Output Mode

(1) Overview of One-Shot Output Mode

In one-shot output mode, a single pulse with a width equal to the reload register setting value plus 1 is generated, then counter operation stops.

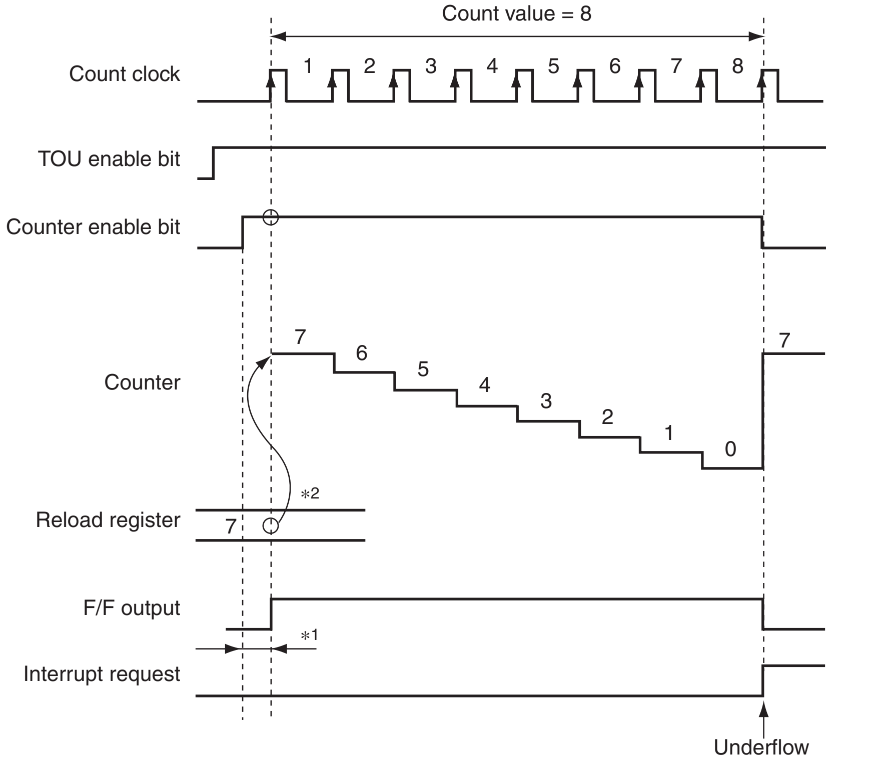
When the timer counter is enabled after setting the reload register, the contents of the reload register are loaded in the counter in synchronization with the count clock and counter operation starts. The counter counts down until an underflow occurs, then stops.

In one-shot output mode, the F/F output waveform consists of the inverted value of the TOUn output control register, output at counter start, and the value of the TOUn output control register, output when an underflow occurs. The result is a single one-shot pulse waveform equivalent to the setting of the reload register plus 1.

It is also possible to generate an interrupt request or DMA transfer request at the counter underflow.

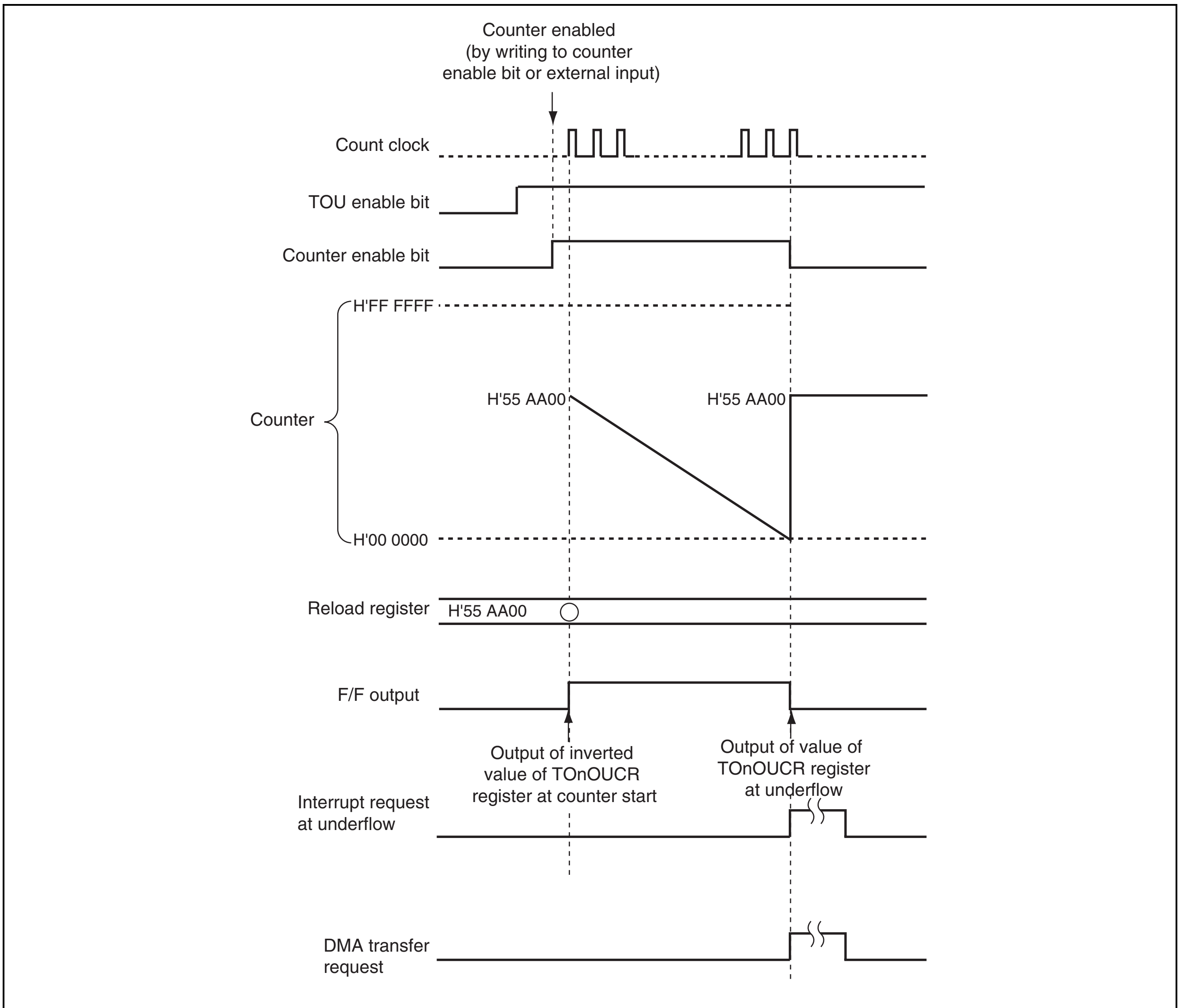
The count value is the setting of the reload register plus 1.

For example, if the initial value of the reload register is 7, the count value is 8.



Notes: This figure does not include detailed timing information.
*1 A count clock delay is included between counter enable and the change in F/F output.
*2 The value of the reload register is reloaded in the counter.

Figure 21.33 Example of Counter Operation in One-Shot Output Mode



Note: • This figure does not include detailed timing information.

Figure 21.34 Operation Example of One-Shot Output Mode

(2) Notes on One-Shot Output Mode

Keep the following points in mind when using one-shot output mode:

- If counter stop at underflow and counter enable by external input occur in the same clock cycle, counter stop at underflow takes precedence.
- If counter stop at underflow and a write of the value for counter enabled to the counter enable bit occur in the same clock cycle, the write of the value for counter enabled to the counter enable bit takes precedence.
- If counter enable by external input and a write of the value for counter disabled to the counter enable bit occur in the same clock cycle, the write of the value for counter disabled to the counter enable bit takes precedence.
- Timer operation is synchronized with the count clock, so there is a delay of up to one count clock period between counter enable and F/F output inversion.

21.21.4 Operation in Continuous Output Mode

(1) Overview of Continuous Output Mode

In continuous output mode, the counter counts down from its setting value, and the value of the reload register is loaded when an underflow occurs. This operation is repeated at each subsequent counter underflow, generating successive pulses equivalent to the inverted value of the reload register plus 1.

When the timer counter is enabled after setting the timer and reload register, down counting starts, in synchronization with the count clock, from the setting value of the timer and continues until an underflow occurs.

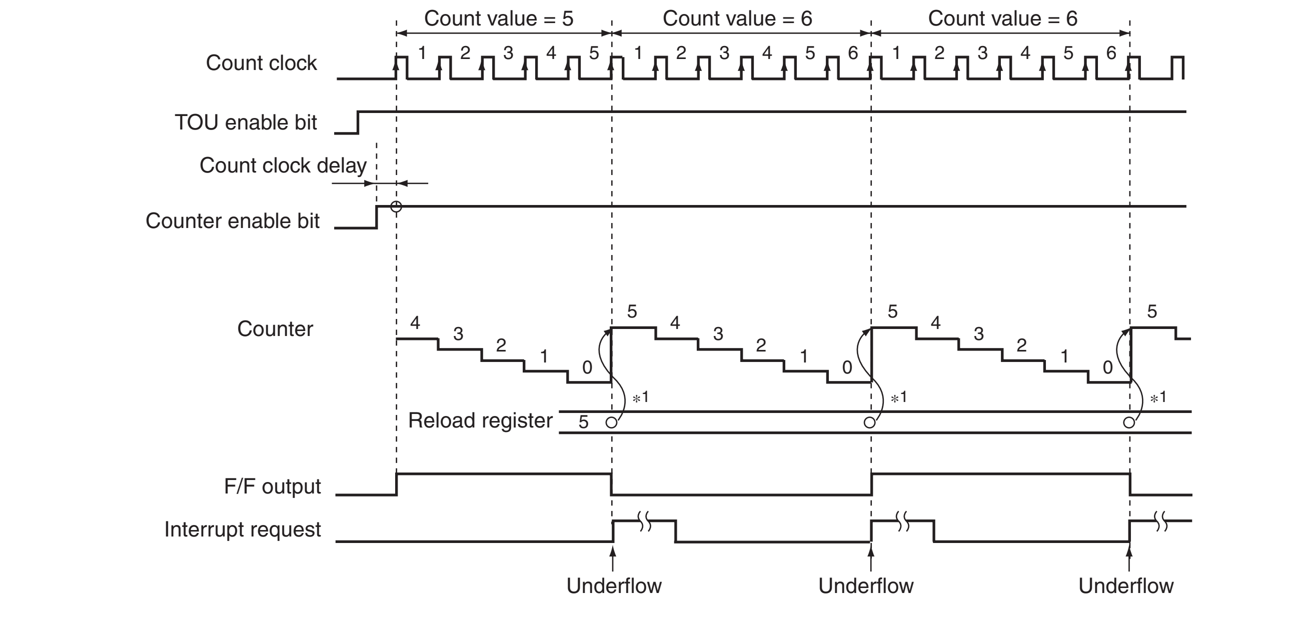
At the underflow cycle, the contents of the reload register are loaded in the counter and down counting begins again. This operation is repeated at each subsequent underflow. The timer stops operating when the value specifying counter disabled is written to the counter enable bit.

In continuous output mode, the F/F output waveform consists of the inverted value of the TOUn output control register, output at counter start and at each even-numbered underflow, and the value of the TOUn output control register, output at each odd-numbered underflow. This pulse waveform is output continuously until the counter stops.

It is also possible to generate an interrupt request or DMA transfer request at each counter underflow.

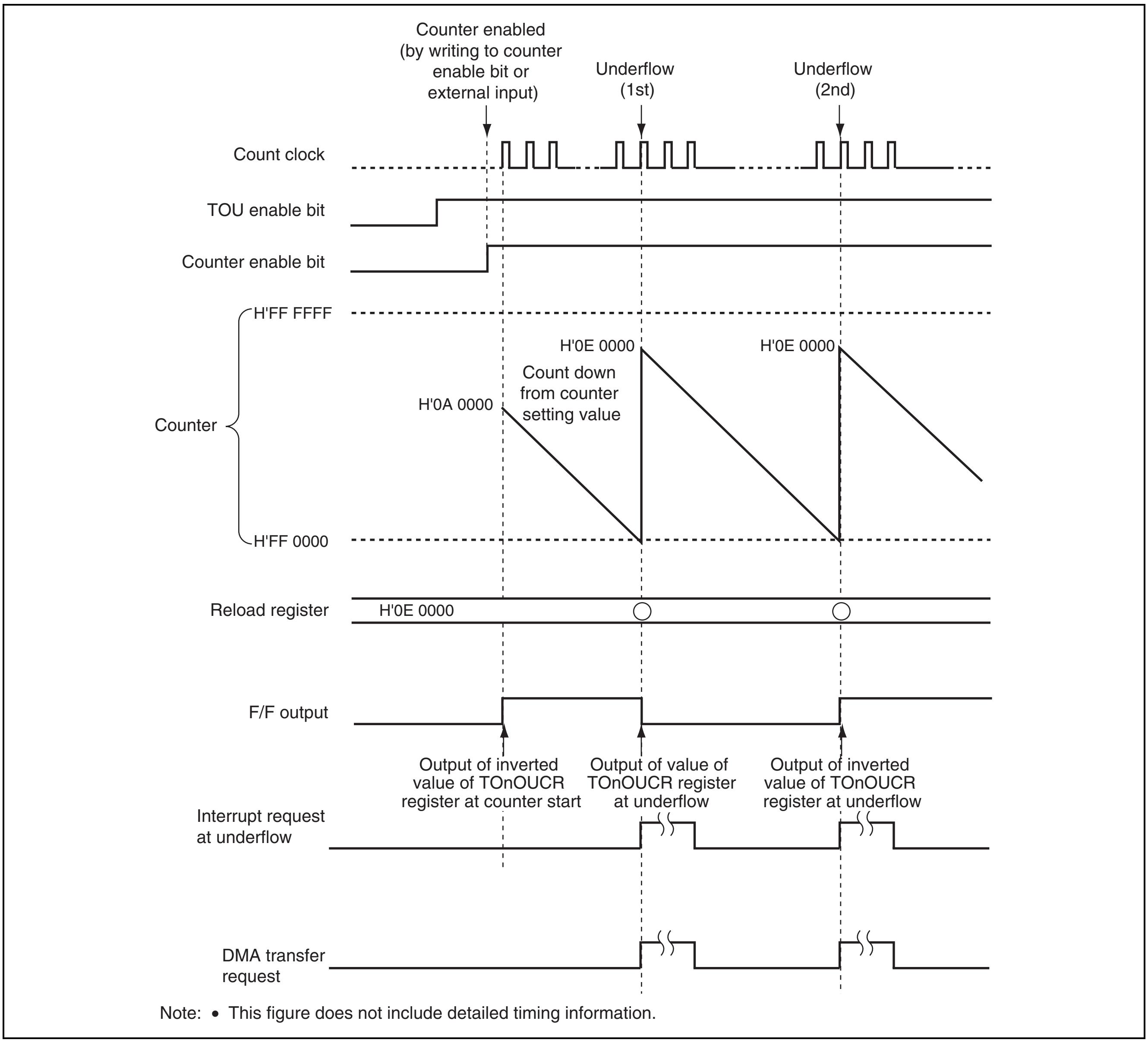
The setting of the counter plus 1 and the setting of the of the reload register plus 1 are valid as count values.

For example, operation is as shown below when the initial value of the counter is 4 and the initial value of the reload register is 5.



Notes: This figure does not include detailed timing information.
*1 The value of the reload register is reloaded in the counter.

Figure 21.35 Example of Counter Operation in Continuous Output Mode


Counter enabled (by writing to counter enable bit or external input)
Underflow (1st)
Underflow (2nd)
Count clock
TOU enable bit
Counter enable bit
H'FF FFFF
H'0E 0000
H'0E 0000
Count down from counter setting value
H'0A 0000
Counter
H'FF 0000
Reload register
H'0E 0000
F/F output
Output of inverted value of TOnOUCR register at counter start
Output of value of TOnOUCR register at underflow
Output of inverted value of TOnOUCR register at underflow
Interrupt request at underflow
DMA transfer request

Note: • This figure does not include detailed timing information.

Figure 21.36 Operation Example of Continuous Output Mode

(2) Notes on Continuous Output Mode

Keep the following points in mind when using continuous output mode:

- If counter enable by external input and a write of the value for counter disabled to the counter enable bit occur in the same clock cycle, the write of the value for counter disabled to the counter enable bit takes precedence.
- When the counter is read in a cycle where an underflow occurs, the value of the reload register is returned. Reading the counter in the clock cycle immediately following returns a value equivalent to the reload register value minus 1.
- Timer operation is synchronized with the count clock, so there is a delay of up to one count clock period between counter enable and F/F output inversion.

21.21.5 0% or 100% Duty Cycle in PWM Output/One-Shot PWM Output Mode

In PWM output or one-shot PWM output mode, it is possible to output a PWM waveform consisting of the non-inverted F/F output with a 0% or 100% duty cycle by setting the reload 0 register or reload 1 register to H'FFFF.

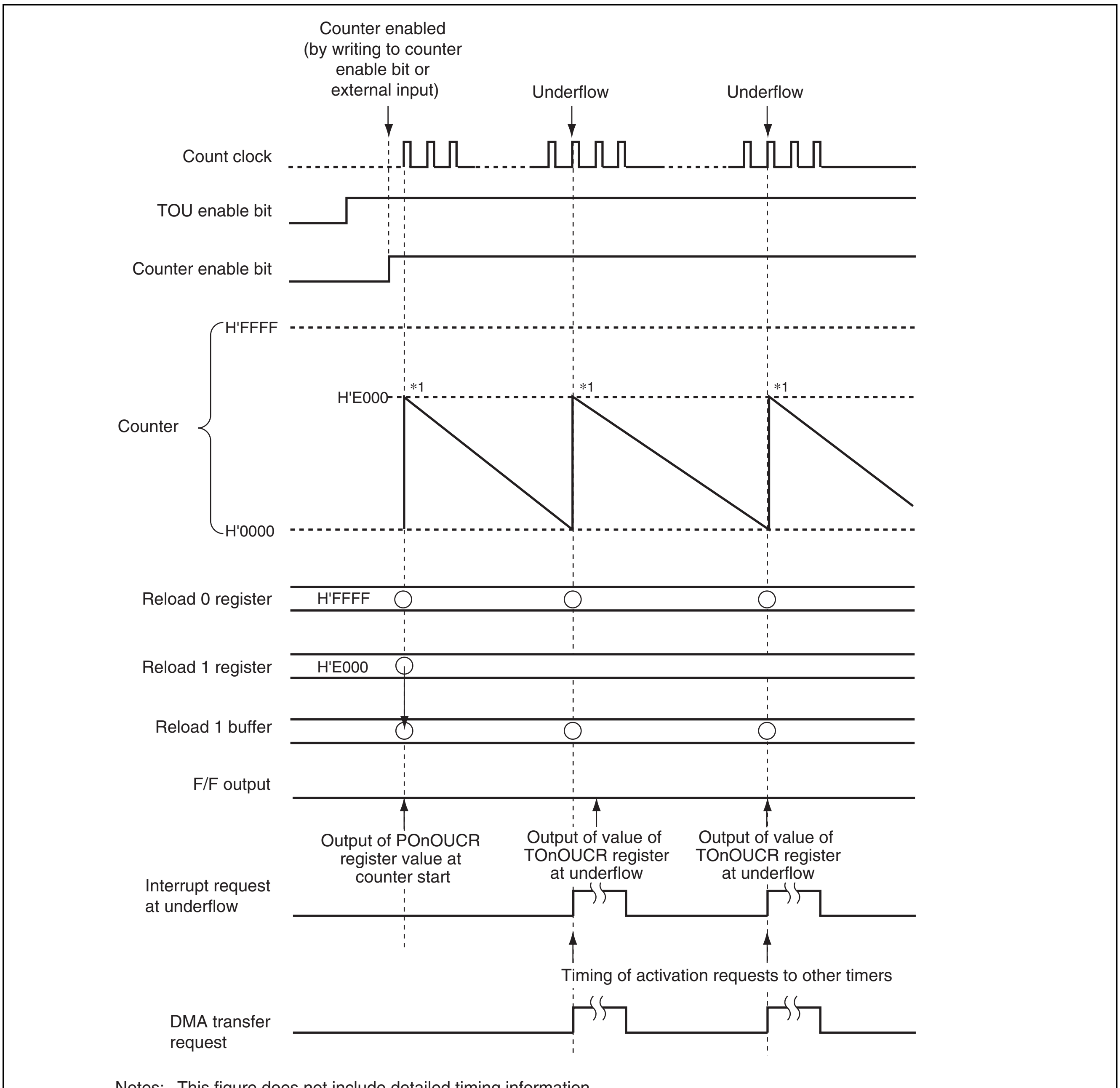
Keep the following in mind to determine whether or not the reload value is H'FFFF in PWM output or one-shot PWM output mode.

(Example) Desired output period is 10 count cycles

Period ratio	50% : 50%	80% : 20%	90% : 10%	100% : 0%
Count ratio	5 : 5	8 : 2	9 : 1	10 : 0
Register settings	0004 : 0004	0007 : 0001	0008 : 0000	0009 : FFFF

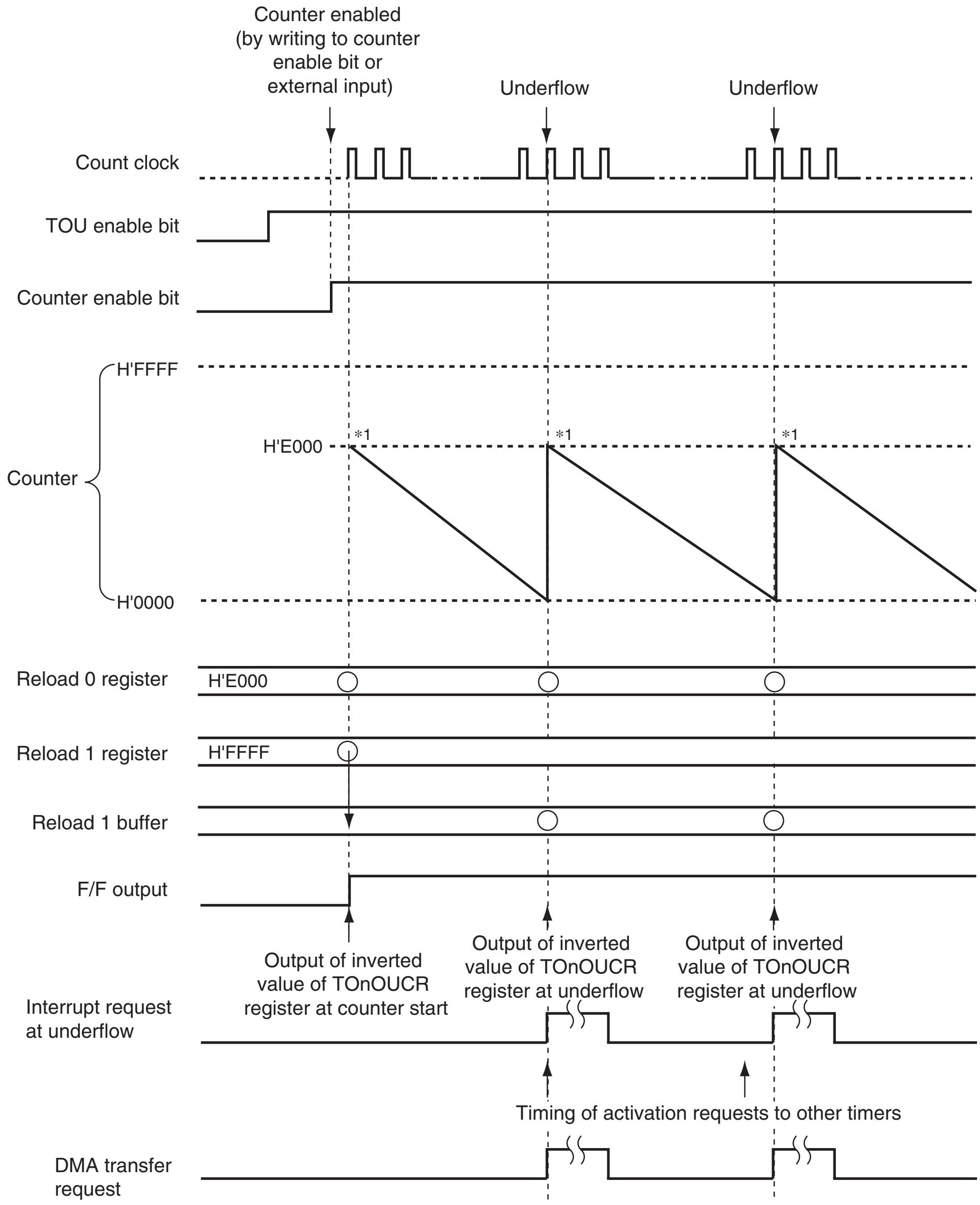
Since the number of cycles counted is n + 1, the actual setting value must be the desired value minus 1.

- Setting a reload register to H'FFFF results in 0% or 100% duty, so it is not possible simply to count a value of H'FFFF.
- Setting both the reload 0 register and reload 1 register to H'FFFF is prohibited.
- Writing H'FFFF when the counter is operating is prohibited.
- Interrupt requests and activation requests to other timers continue to be generated when the duty cycle is 0% or 100%.



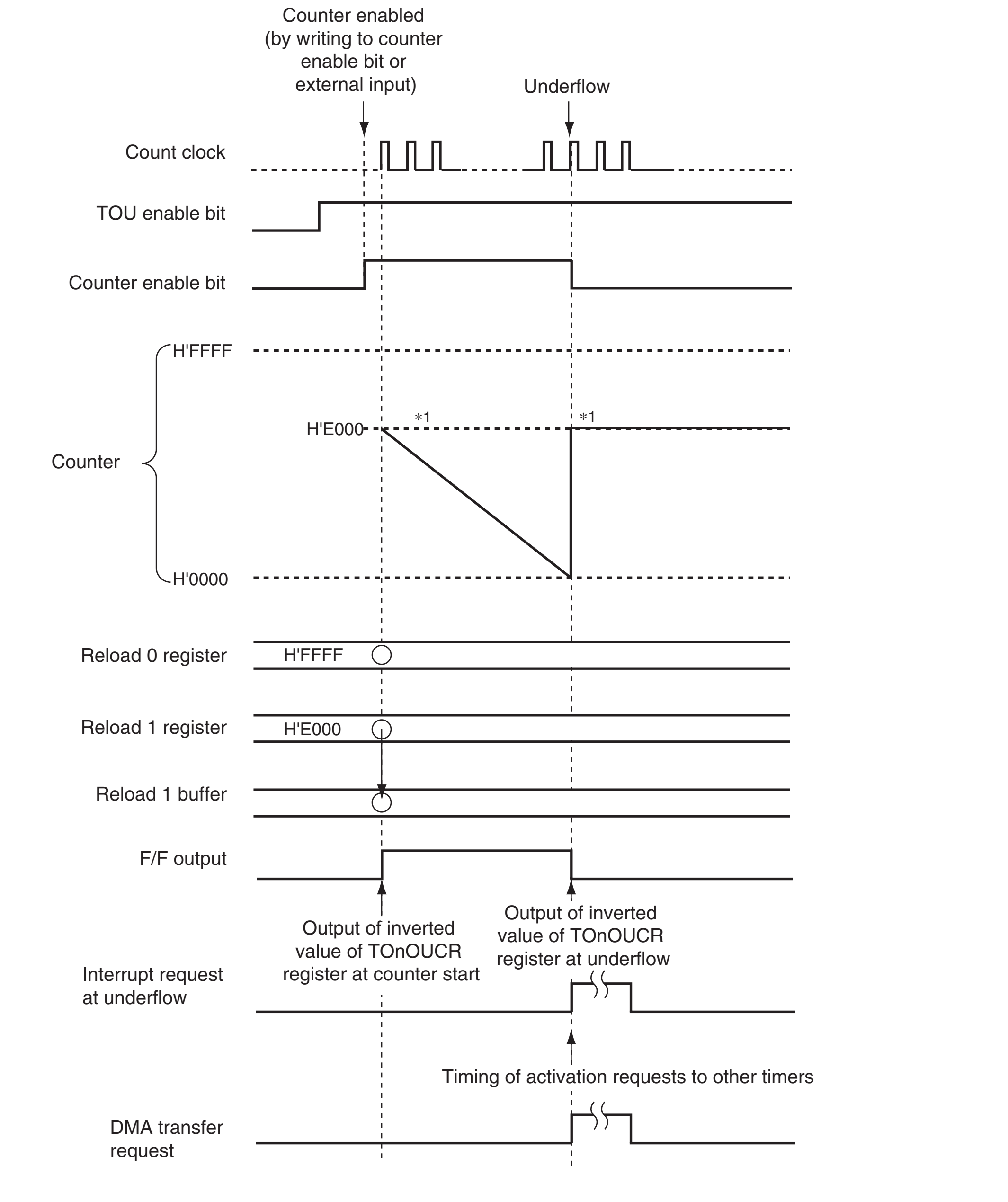
Notes: This figure does not include detailed timing information.
*1 The value of the reload 1 register is reloaded in the counter because the reload 0 register is set to H'FFFF.

Figure 21.37 Operation Example of PWM Output Mode (Reload 0 Register: H'FFFF)


Counter enabled
(by writing to counter
enable bit or
external input)
Underflow
Underflow
Count clock
TOU enable bit
Counter enable bit
H'FFFF
Counter
H'E000
*1
*1
*1
H'0000
Reload 0 register
H'E000
Reload 1 register
H'FFFF
Reload 1 buffer
F/F output
Output of inverted value of TOnOUCR register at counter start
Output of inverted value of TOnOUCR register at underflow
Output of inverted value of TOnOUCR register at underflow
Interrupt request at underflow
Timing of activation requests to other timers
DMA transfer request

Notes: This figure does not include detailed timing information.
*1 The value of the reload 0 register is reloaded in the counter because the reload 1 register is set to H'FFFF.

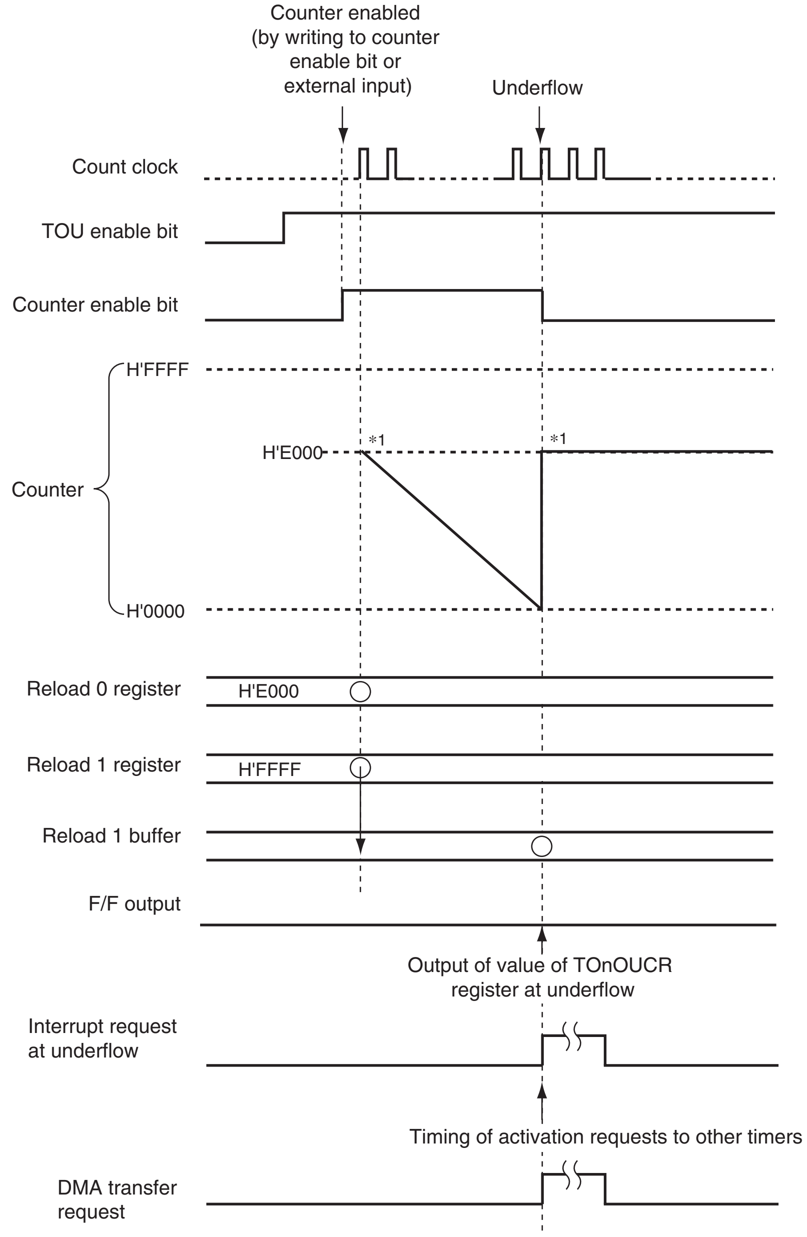
Figure 21.38 Operation Example of PWM Output Mode (Reload 1 Register: H'FFFF)



Notes: This figure does not include detailed timing information.

*1 The value of the reload 0 register is reloaded in the counter because the reload 1 register is set to H'FFFF.

Figure 21.39 Operation Example of One-Shot PWM Output Mode (Reload 0 Register: H'FFFF)



Notes: This figure does not include detailed timing information.
*1 The value of the reload 0 register is reloaded in the counter.

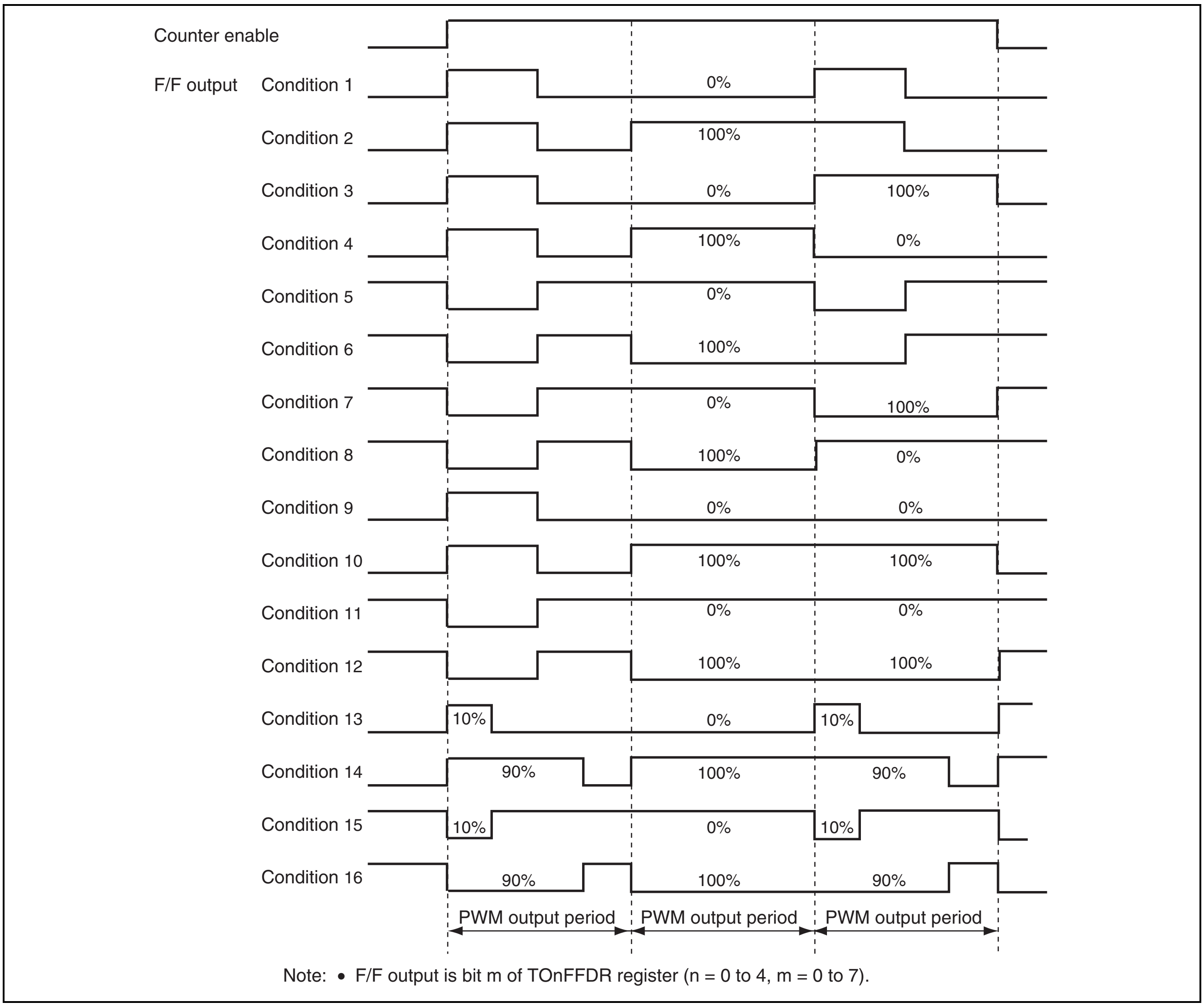
Figure 21.40 Operation Example of One-Shot PWM Output Mode (Reload 1 Register: H'FFFF)

In PWM output mode and one-shot PWM output mode, the F/F output is as shown below when the duty cycle is 0% or 100%.

(1) PWM Output Mode

0%: Value of TOnOUCR register is output.

100%: Inverted value of TOnOUCR register is output.



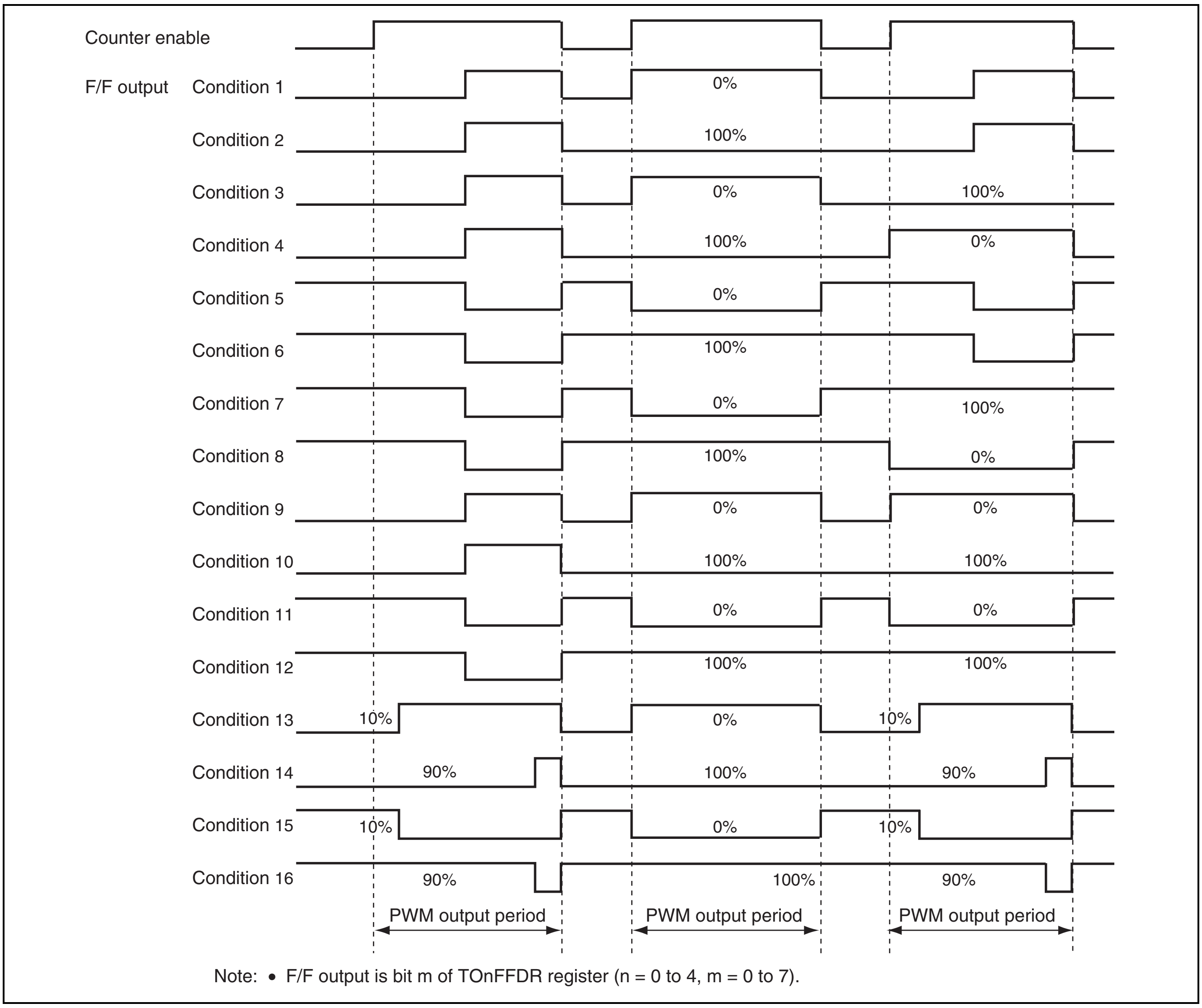
Note: • F/F output is bit m of TOnFFDR register (n = 0 to 4, m = 0 to 7).

Figure 21.41 Example of F/F Output with 0% or 100% Duty Cycle in PWM Output Mode

(2) One-Shot PWM Output Mode

0%: Inverted value of TOnOUCR register is output.

100%: Value of TOnOUCR register is output.



Note: • F/F output is bit m of TOnFFDR register (n = 0 to 4, m = 0 to 7).

Figure 21.42 Example of F/F Output with 0% or 100% Duty Cycle in One-Shot PWM Output Mode

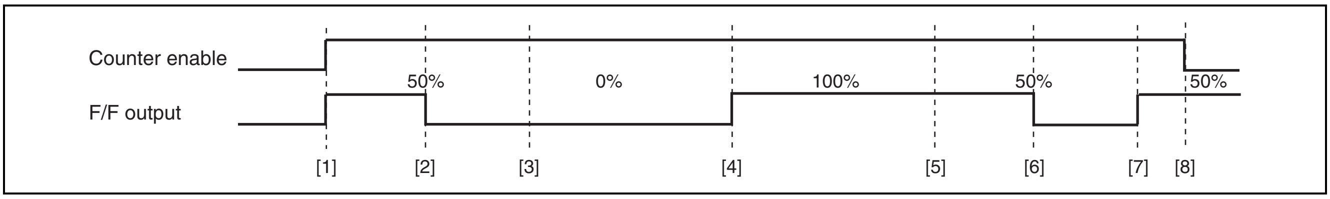
21.21.6 F/F Output in Each Operating Mode

F/F output in each operating mode is controlled by the settings of the TOUn flip-flop output data register (TOnFFDR) and TOUn output control register (TOnOUCR). The correspondence between the settings of the TOnFFDR and TOnOUCR registers, and the F/F output, is described below.

(1) Common to All Operating Modes

- When data is written to the TOnOUCR register, the TOnOUCR and TOnFFDR registers are both updated with the new TOnOUCR register contents.
- When data is written to the TOnFFDR register, only the TOnFFDR register is updated with the new TOnFFDR register contents. (The updated TOnFFDR register contents are reflected immediately in the F/F output.)
- The TOnOUCR and TOnFFDR registers can be rewritten regardless of the counter enable status.

(2) F/F Output in PWM Output Mode



[1] At counter start, the F/F output is the inverted value of the TOnOUCR register.

[2] At underflow, the F/F output is the value of the TOnOUCR register.

[3] At underflow, the F/F output is the value of the TOnOUCR register if the reload 0 register value is H'FFFF and the reload 1 register value is H'xxxx (any value other than H'FFFF). The output period is determined by the value of the reload 1 register.

[4] At underflow, the F/F output is the inverted value of the TOnOUCR register if the reload 0 register value is H'xxxx (any value other than H'FFFF) and the reload 1 register value is H'FFFF. The output period is determined by the value of the reload 0 register.

[5] At underflow, the F/F output is the inverted value of the TOnOUCR register.

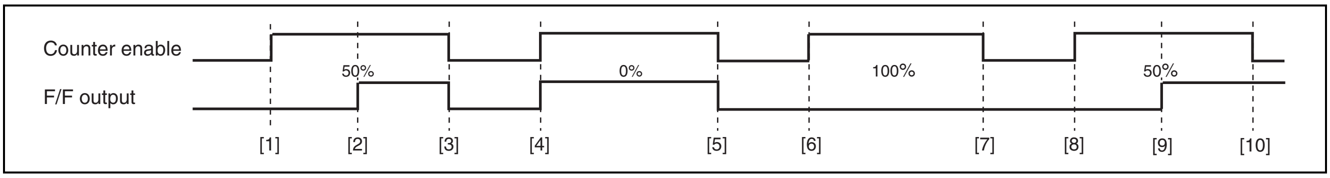
[6] At underflow, the F/F output is the value of the TOnOUCR register.

[7] At underflow, the F/F output is the inverted value of the TOnOUCR register.

[8] When the counter enable bit is set to disabled (operation stop), F/F output maintains the value of the TOnFFDR register.

Note: • Setting both the reload 0 register and reload 1 register to H'FFFF is prohibited.

(3) F/F Output in One-Shot PWM Output Mode



[1] At counter start, the F/F output is the value of the TOnOUCR register.

[2] At underflow, the F/F output is the inverted value of the TOnOUCR register.

[3] At counter stop at underflow, the F/F output is the value of the TOnOUCR register.

[4] At counter start, the F/F output is the inverted value of the TOnOUCR register if the reload 0 register value is H'FFFF and the reload 1 register value is H'xxxx (any value other than H'FFFF). The output period is determined by the value of the reload 1 register.

[5] At counter stop at underflow, the F/F output is the value of the TOnOUCR register.

[6] At counter start, the F/F output is the inverted value of the TOnOUCR register if the reload 0 register value is H'xxxx (any value other than H'FFFF) and the reload 1 register value is H'FFFF. The output period is determined by the value of the reload 0 register.

[7] At counter stop at underflow, the F/F output is the value of the TOnOUCR register.

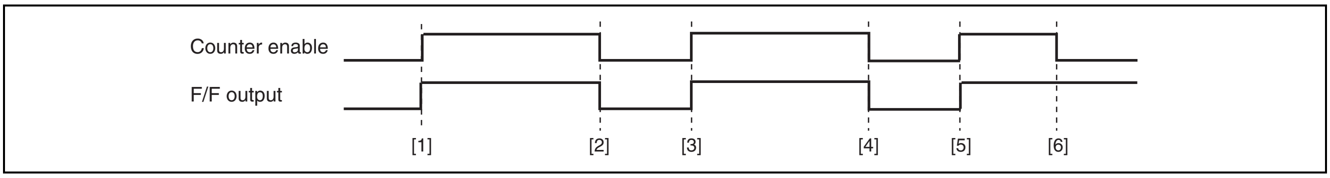
[8] At counter start, the F/F output is the value of the TOnOUCR register.

[9] At underflow, the F/F output is the inverted value of the TOnOUCR register.

[10] When the counter enable bit is set to disabled (operation stop), F/F output maintains the value of the TOnFFDR register.

Note: • Setting both the reload 0 register and reload 1 register to H'FFFF is prohibited.

(4) F/F Output in One-Shot Output Mode



[1] At counter start, the F/F output is the inverted value of the TOnOUCR register.

[2] At counter stop at underflow, the F/F output is the value of the TOnOUCR register.

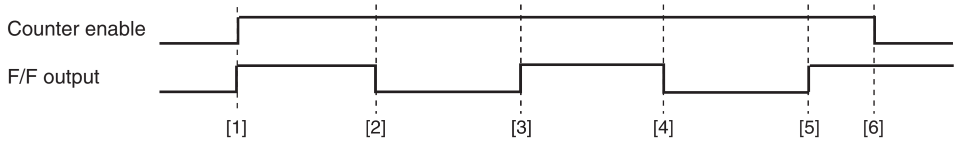
[3] At counter start, the F/F output is the inverted value of the TOnOUCR register.

[4] At counter stop at underflow, the F/F output is the value of the TOnOUCR register.

[5] At counter start, the F/F output is the inverted value of the TOnOUCR register.

[6] When the counter enable bit is set to disabled (operation stop), F/F output maintains the value of the TOnFFDR register.

(5) F/F Output in Continuous Output Mode



[1] At counter start, the F/F output is the inverted value of the TOnOUCR register.

[2] At underflow, the F/F output is the value of the TOnOUCR register.

[3] At underflow, the F/F output is the inverted value of the TOnOUCR register.

[4] At underflow, the F/F output is the value of the TOnOUCR register.

[5] At underflow, the F/F output is the inverted value of the TOnOUCR register.

[6] When the counter enable bit is set to disabled (operation stop), F/F output maintains the value of the TOnFFDR register.

21.21.7 PWM Output-Prohibit Function

The ATU-IIIS provides a function for forcibly disabling output from TOn0 to TOn5, the output pins of timers TOUn_0 to TOUn_5. This function can be used for protection when an abnormal state is detected, such as a short circuit during three-phase PWM control, but it is available for use in all timer TOU output modes. The PWM output-prohibit function cannot be used for I/O ports set to modes other than timer output. Figure 21.43 shows the circuit configuration of the PWM output-prohibit function.

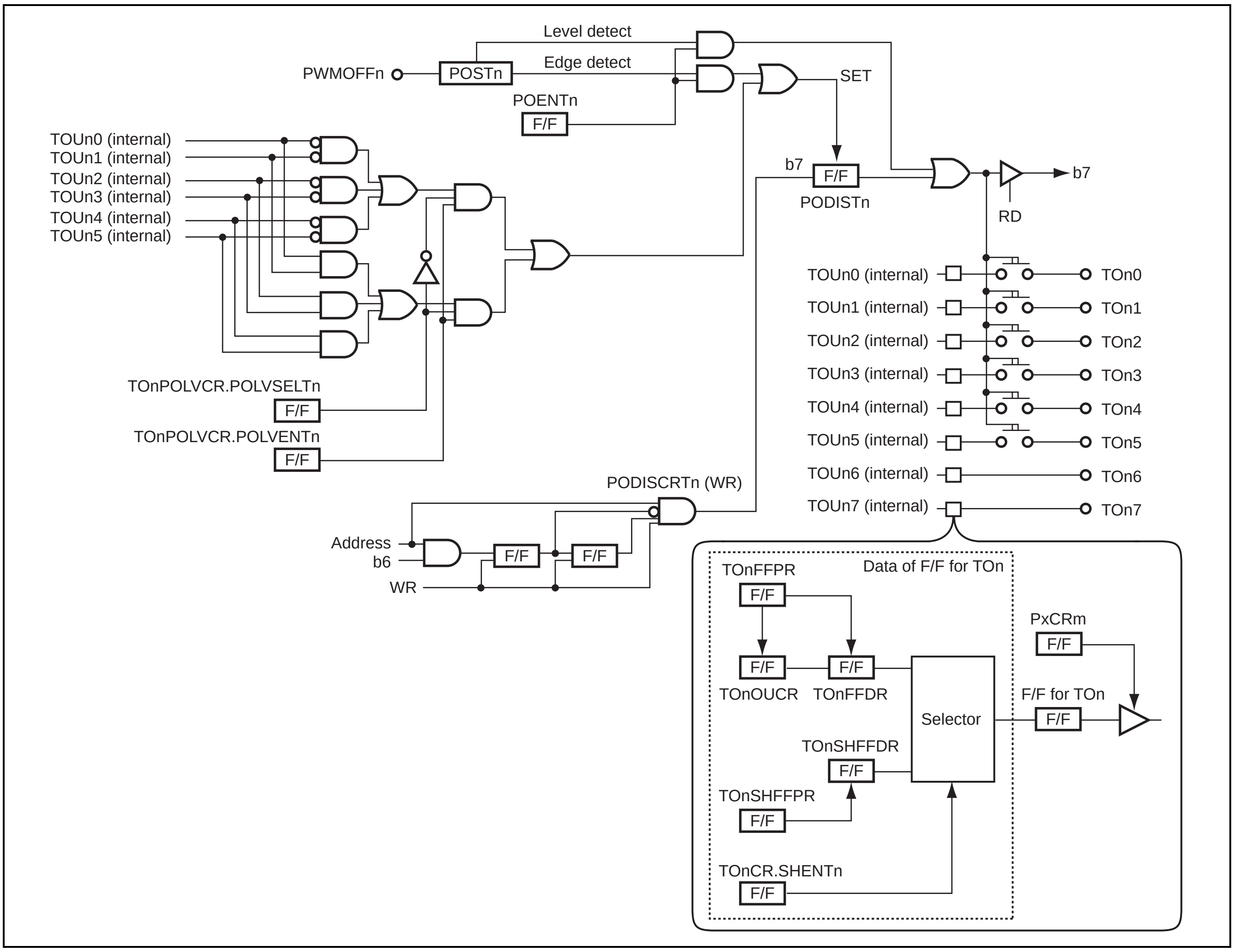


Figure 21.43 Circuit Configuration of PWM Output-Prohibit Function

PWM output can be disabled using any of the following three methods:

(1) Disabling PWM Output by Using a Signal on an External Pin (PWMOFFn)

Output on ports TOn0 to TOn5, the PWM outputs for timers TOUn_0 to TOUn_5, respectively, can be disabled by inputting a signal on external pin PWMOFFn.

- When the rising edge, falling edge, or both edges setting is selected by the POSTn bits in the TOUnPWMOFF input processing register (TOnPOCR)

  PWM output is disabled when edge detection occurs on the signal input an external pin (PWMOFFn). When edge detection occurs the PODISTn bit in the TOUnPWM output-prohibit control register (TOnPODISCR) is set to "1".

  Returning to the PWM output-enabled state is accomplished by clearing the PODISTn bit in the TOUnPWM output-prohibit control register (TOnPODISCR) to "0".
- When the "L" level or "H" level setting is selected by the POSTn bits in the TOUnPWMOFF input processing register (TOnPOCR)

  PWM output is disabled during input on an external pin (PWMOFFn) of the PWM output-prohibit level. When this occurs, the PODISTn bit in the TOUnPWM output-prohibit control register is set to "1".

  Returning to the PWM output-enabled state is accomplished by halting input of the PWM output-prohibit level on the external pin (PWMOFFn). At this point, reading the PODISTn bit in the TOUnPWM output-prohibit control register returns the setting value last written to it.

Note: • When the PODISTn bit in the TOUnPWM output-prohibit control register is written to while the PWM output-prohibit level is being input on the external pin (PWMOFFn), the value written in stored in the register. However, reading the PODISTn bit returns a value of "1". The setting value of the PODISTn bit can be read once input of the PWM output-prohibit level on the external pin stops, and PWM output is then controlled according to that setting value.

To disable PWM output when a signal is input on the external pin (PWMOFFn), make the following settings in the TOUnPWMOFF input processing control register (TOnPOCR) and TOUnPWMOFF function enable register (TOnPOER):

- Disabling PWM output when a signal is input on PWMOFFn

  A. Write an appropriate setting value ("001", "010", "011", "10X" or "11X") to the POSTn bits in the TOnPOCR register.

  B. Write "1" to the POENTn bit in the TOnPOER register to enable the PWMOFF function.

(2) Disabling PWM Output by Using the PWM Output-Prohibit Control Register

Output on ports TOn0 to TOn5, the PWM outputs for timers TOUn_0 to TOUn_5, respectively, can be disabled by setting the TOUnPWM output-prohibit control register (TOnPODISCR).

To disable PWM output by using the TOUnPWM output-prohibit control register (TOnPODISCR), make the following settings:

- Disabling PWM output by using the TOUnPWM output-prohibit control register (TOnPODISCR)

  A. Set the PODISTn bit in the TOnPODISCR register to "1" (output disabled).

(3) Disabling PWM Output by Using the Pin Level of Ports TOn0 to TOn5

Output on ports TOn0 to TOn5, the PWM outputs for timers TOUn_0 to TOUn_5, respectively, can be disabled based on the pin level ("L" or "H" level) of ports TOn0 to TOn5.

PWM output is disabled after the PWM output-prohibit level is detected on a port among TOUn0 to TOUn5. The PODISTn bit in the TOUnPWM output-prohibit control register (TOnPODISCR) is set to "1" when PWM output is disabled. Returning to the PWM output-enabled state is accomplished by first halting input of the PWM output-prohibit level on the port among TOUn0 to TOUn5 and then clearing the PODISTn bit in the TOUnPWM output-prohibit control register to "0".

Note: • When the PODISTn bit in the TOUnPWM output-prohibit control register is written to while the PWM output-prohibit level is being output on a port among TOUn0 to TOUn5, the value written is ignored.

To disable PWM output by using the port pin level, make the following settings in the TOUnPWM output-prohibit control register (TOnPODISCR) and TOUnPWMOFF function enable register (TOnPOER):

- Disabling PWM output by using the level of ports TOn0 to TOn5
  A. Set the level ("L" or "H" level) at which PWM output is to be disabled by setting the POLVSELTn bit in the TOnPOLVCR register.
  B. Set the POLVENTn bit in the TOnPOLVCR register to "1" (output-prohibit enabled).
  C. Write "1" to the POENTn bit in the TOnPOER register to enable the PWMOFF function.

21.21.8 Short-Circuit Prevention Function

(1) Overview of Short-Circuit Prevention Function

Set the short-circuit prevention function enable/disable bit when counters TOUn_0 to TOUn_5 are stopped. (The setting of this bit in counter enabled status is prohibited.)

When the short-circuit prevention function is enabled, the operating modes of the timers should be as follows (the function cannot be used with the timers set to other modes):

TOUn_0 (TOUn_2, TOUn_4): One-shot PWM output mode

TOUn_1 (TOUn_3, TOUn_5): One-shot output mode

Note:
- TOUn_0 (2, 4) includes TOUn_0, TOUn_2, and TOUn_4.
- TOUn_1 (3, 5) includes TOUn_1, TOUn_3, and TOUn_5.

When the short-circuit prevention function is enabled, the TOUn enable source select bits for TOUn_1 (TOUn_3, TOUn_5) have no effect, and underflow of TOUn_0 (TOUn_2, TOUn_4) is the activation source for TOUn_1 (TOUn_3, TOUn_5).

The short-circuit prevention duration is specified by the setting of the TOUn_1 (TOUn_3, TOUn_5) reload register. The short-circuit prevention duration is the setting value of the reload register plus 2. Note that the setting value of the reload register must meet the following conditions:

TOUn_1 (TOUn_3, TOUn_5) reload register setting value ≤ TOUn_0 (TOUn_2, TOUn_4) reload register setting value – 2

When enabling the short-circuit prevention function, values must be set in the TOUn output control register and flip-flop output data register for the TOUn short-circuit prevention function.

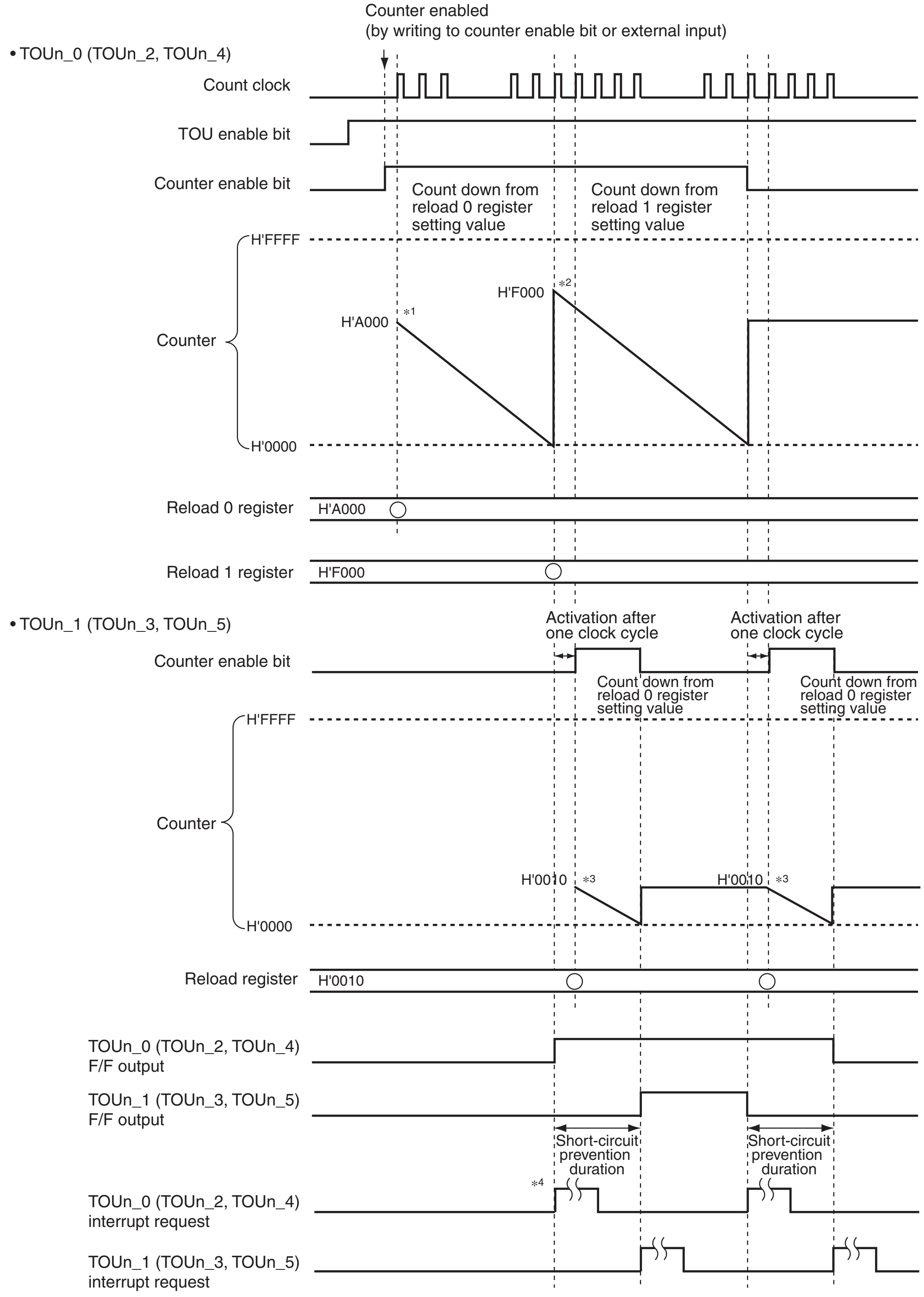
- To output "H" level initially

  Set the TOUn output control register to "1" and the flip-flop output data register for the TOUn short-circuit prevention function to "0".
- To output "L" level initially

  Set the TOUn output control register to "0" and the flip-flop output data register for the TOUn short-circuit prevention function to "1".

Writing the same value to the TOUn output control register and to the flip-flop output data register for the TOUn short-circuit prevention function results in fixed output.

When the short-circuit prevention function is enabled, writing "H'FFFF" to the TOUn_0 (TOUn_2, TOUn_4) reload 0 register and TOUn_0 (TOUn_2, TOUn_4) reload 1 register is prohibited.


Counter enabled
(by writing to counter enable bit or external input)
• TOUn_0 (TOUn_2, TOUn_4)
Count clock
TOU enable bit
Counter enable bit
Count down from reload 0 register setting value
Count down from reload 1 register setting value
H'FFFF
H'F000
*2
H'A000
*1
Counter
H'0000
Reload 0 register
H'A000
Reload 1 register
H'F000
• TOUn_1 (TOUn_3, TOUn_5)
Activation after one clock cycle
Activation after one clock cycle
Counter enable bit
Count down from reload 0 register setting value
Count down from reload 0 register setting value
H'FFFF
Counter
H'0010
*3
H'0010
*3
H'0000
Reload register
H'0010
TOUn_0 (TOUn_2, TOUn_4) F/F output
TOUn_1 (TOUn_3, TOUn_5) F/F output
Short-circuit prevention duration
Short-circuit prevention duration
*4
TOUn_0 (TOUn_2, TOUn_4) interrupt request
TOUn_1 (TOUn_3, TOUn_5) interrupt request

Notes: This figure does not include detailed timing information.
*1 The value of the reload 0 register is reloaded in the counter.
*2 The value of the reload 1 register is reloaded in the counter.
*3 The value of a reload register is reloaded in the counter.
*4 An interrupt request can be output by making a register setting.

Figure 21.44 Overview of Short-Circuit Prevention Function Operation

To disable the short-circuit prevention function when fixing output forcibly by software, perform the following procedure:

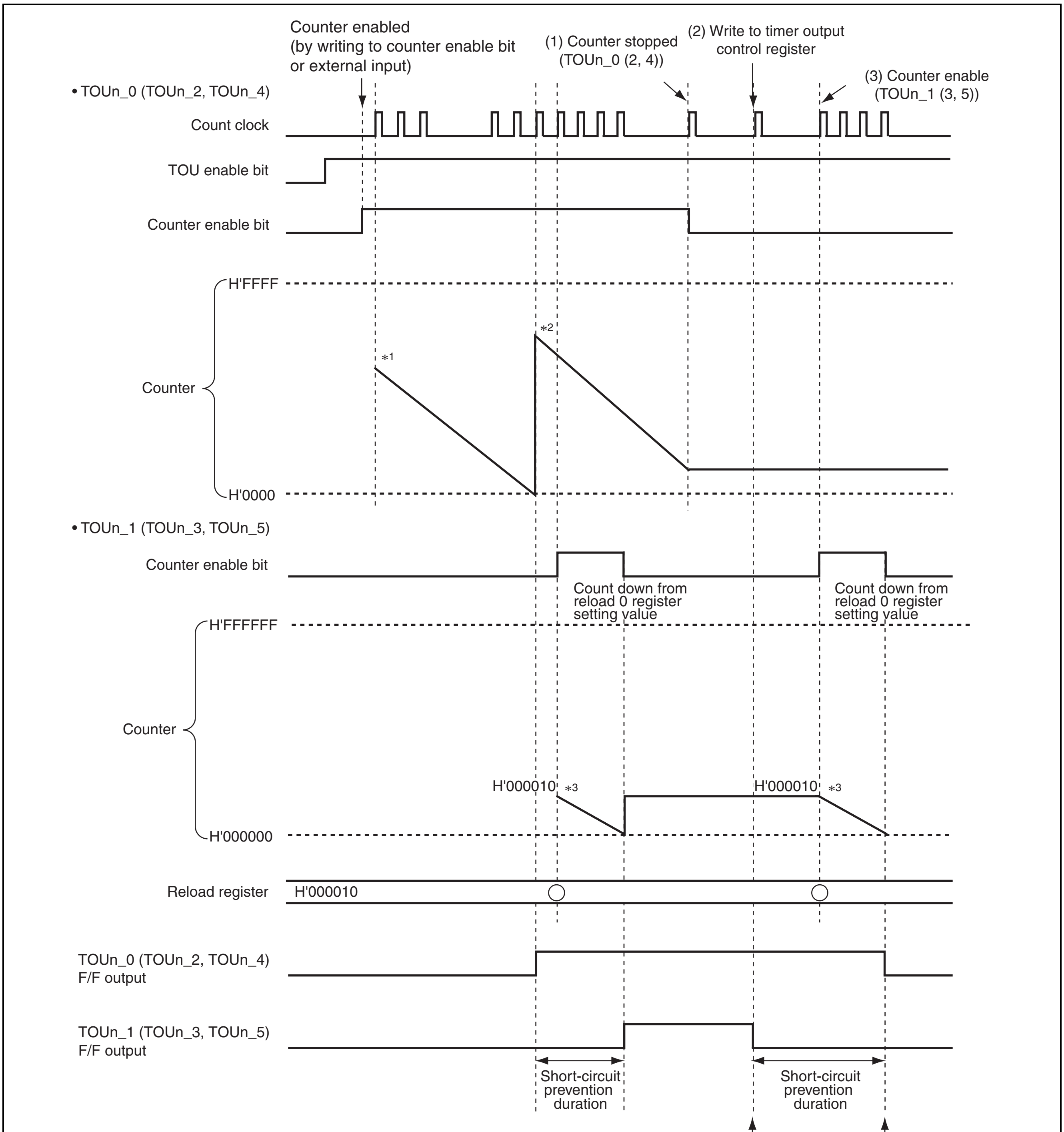
- Write "0" to the count enable bits for TOUn_0 (TOUn_2, TOUn_4).
- Write to the TOUn output control register for TOUn_1 (TOUn3, TOUn5) with a value to enable short-circuit prevention.
- Write "1" to the count enable bits for TOUn_1 (TOUn_3, TOUn_5).

At this time, short-circuit prevention duration is as follows:

Duration between writing to the TOUn output control register for TOUn_1 (TOUn_3, TOUn_5) and TOUn_1 (TOUn_3, TOUn_5) count enable + TOUn_1 (TOUn_3, TOUn_5) reload register setting value + 1

When stopping the counters by software, make the settings so that both TOUn_0 (TOUn_2, TOUn_4) and TOUn_1 (TOUn_3, TOUn_5) are stopped.

When writing to "1" to the count enable bits for TOUn_1 (TOUn_3, TOUn_5), both TOUn_0 (TOUn_2, TOUn_4) and TOUn_1 (TOUn_3, TOUn_5) must be stopped.



Notes: This figure does not include detailed timing information.

*1 The value of the reload 0 register is reloaded in the counter.

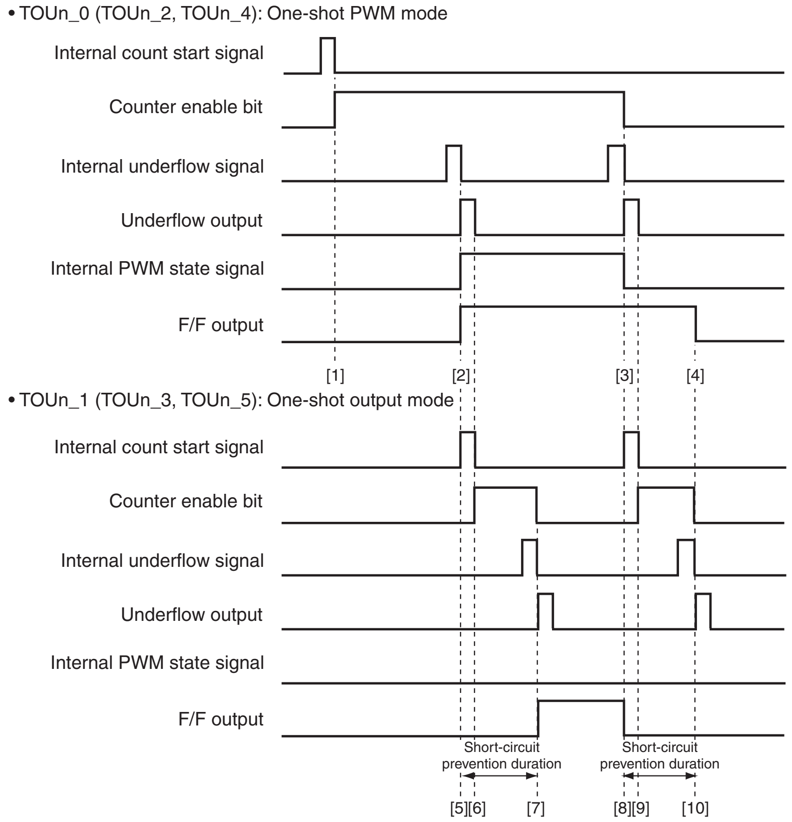
*2 The value of the reload 1 register is reloaded in the counter.

*3 The value of a reload register is reloaded in the counter.

Figure 21.45 Overview of Operation when Forcibly Fixing Output by Software

(2) F/F Output and Register Settings when Using the Short-Circuit Prevention Function

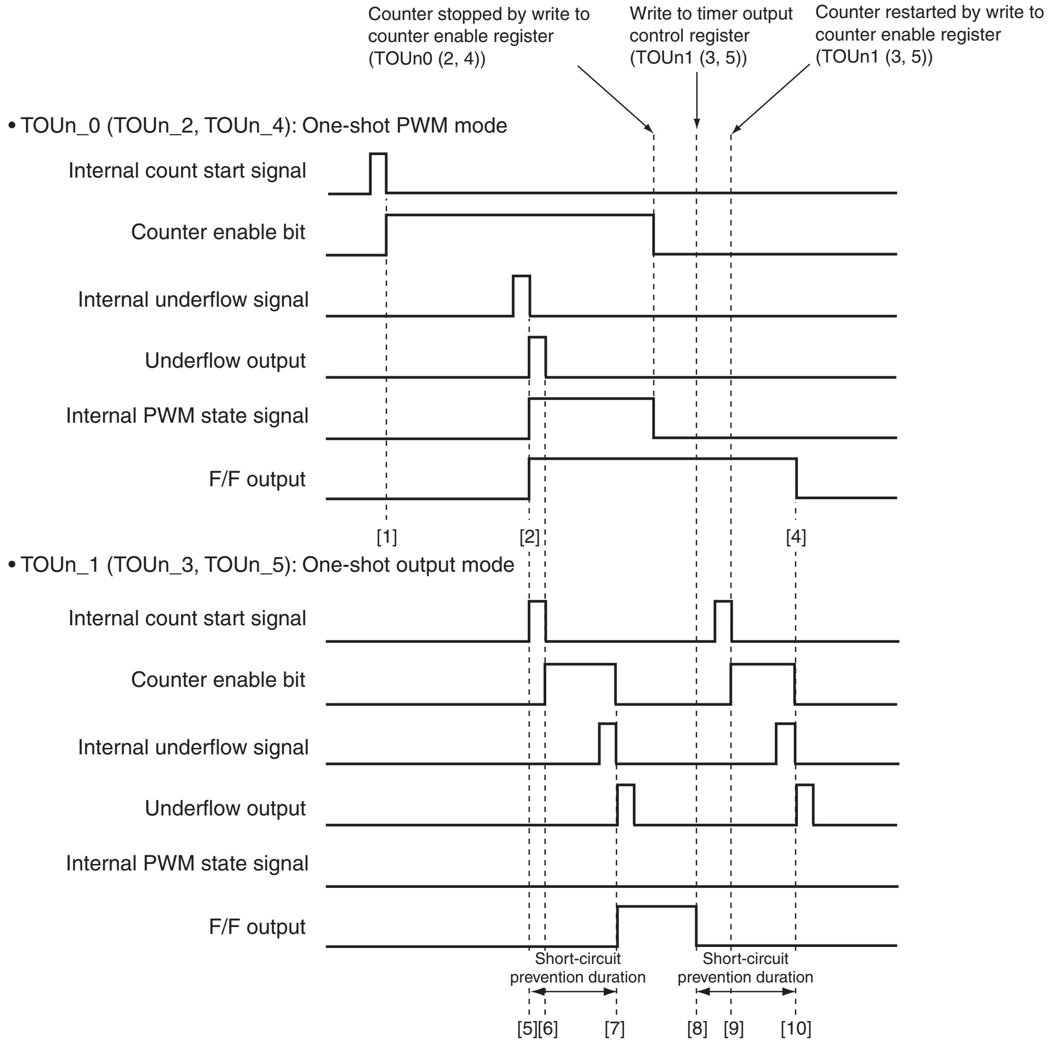
Figure 21.46 shows the correspondence between F/F output and register setting values when the short-circuit prevention function is enabled.


• TOUn_0 (TOUn_2, TOUn_4): One-shot PWM mode
Internal count start signal
Counter enable bit
Internal underflow signal
Underflow output
Internal PWM state signal
F/F output
[1]
[2]
[3]
[4]
• TOUn_1 (TOUn_3, TOUn_5): One-shot output mode
Internal count start signal
Counter enable bit
Internal underflow signal
Underflow output
Internal PWM state signal
F/F output
Short-circuit prevention duration
Short-circuit prevention duration
[5][6]
[7]
[8][9]
[10]

Notes:
- Writing the same value to the TOnFFDR and TOnSHFFDR registers results in fixed output.
- When the short-circuit prevention function is enabled for TOUn_1 (TOUn_3, TOUn_5) by the SHENTn bit in the TOnCR register, activation occurs at underflow of the previous TOU channel regardless of the setting of the ENSELTn1 (ENSELTn3, ENSELTn5) bits in the TOn1MCR (TOn1MCR3, TOn1MCR5) register.

Legend:
[1] The TOUn_0 (TOUn_2, TOUn_4) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal count start signal)
[2] The TOUn_0 (TOUn_2, TOUn_4) F/F output is the inverted value of the TOnSHFFDR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal underflow signal)
[3] The TOUn_0 (TOUn_2, TOUn_4) F/F output is the inverted value of the TOnSHFFDR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal underflow signal)
[4] The TOUn_0 (TOUn_2, TOUn_4) F/F output is the inverted value of the TOnOUCR register.
(At the "L" level of the TOUn_0 (TOUn_2, TOUn_4) internal PWM state signal,
at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal underflow signal)
[5] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal underflow signal)
[6] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) underflow output,
= at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal count start signal)
[7] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnSHFFDR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal PWM state signal,
at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal underflow signal)
[8] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal underflow signal)
[9] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) underflow output,
= at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal count start signal)
[10] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "L" level of the TOUn_0 (TOUn_2, TOUn_4) internal PWM state signal,
at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal underflow signal)

Figure 21.46 F/F Output and Register Setting Values with Short-Circuit Prevention Function Enabled


Counter stopped by write to counter enable register (TOUn0 (2, 4))
Write to timer output control register (TOUn1 (3, 5))
Counter restarted by write to counter enable register (TOUn1 (3, 5))
• TOUn_0 (TOUn_2, TOUn_4): One-shot PWM mode
Internal count start signal
Counter enable bit
Internal underflow signal
Underflow output
Internal PWM state signal
F/F output
[1]
[2]
[4]
• TOUn_1 (TOUn_3, TOUn_5): One-shot output mode
Internal count start signal
Counter enable bit
Internal underflow signal
Underflow output
Internal PWM state signal
F/F output
Short-circuit prevention duration
Short-circuit prevention duration
[5][6]
[7]
[8]
[9]
[10]

Legend:

[1] The TOUn_0 (TOUn_2, TOUn_4) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal count start signal)

[2] The TOUn_0 (TOUn_2, TOUn_4) F/F output is the inverted value of the TOnSHFFDR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal underflow signal)

[4] The TOUn_0 (TOUn_2, TOUn_4) F/F output is the inverted value of the TOnOUCR register.
(At the "L" level of the TOUn_0 (TOUn_2, TOUn_4) internal PWM state signal,
at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal underflow signal)

[5] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal underflow signal)

[6] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) underflow output,
= at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal count start signal)

[7] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnSHFFDR register.
(At the "H" level of the TOUn_0 (TOUn_2, TOUn_4) internal PWM state signal,
at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal underflow signal)

[8] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnFFDR register.
(By writing "1" to the TOUn_1 (TOUn_3, TOUn_5) F/F data register)

[9] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(By writing "1" to the TOUn_1 (TOUn_3, TOUn_5) count enable register)

[10] The TOUn_1 (TOUn_3, TOUn_5) F/F output is the inverted value of the TOnOUCR register.
(At the "L" level of the TOUn_0 (TOUn_2, TOUn_4) internal PWM state signal,
at the "H" level of the TOUn_1 (TOUn_3, TOUn_5) internal underflow signal)

Figure 21.47 F/F Output and Register Settings with Output Forcibly Fixed by Software

(3) Timer TOU Application (Three-Phase Motor Control)

To maximize the PWM resolution, Pck (40 MHz) with a prescaler division ratio of 1 is used as the count source.

TOU0_7 is set to continuous output mode to generate a 20 kHz carrier period. TOU0_0, TOU0_2, and TOU0_4 are set to one-shot PWM output mode, and TOU0_6 to one-shot output mode. TOU0_0, TOU0_2, TOU0_4, and TOU0_6 are activated at TOU0_7 underflow.

TOU0_1, TOU0_3, and TOU0_5 are set to one-shot output mode for use as a short-circuit prevention function.

The PWM output waveforms generated by TOU0_0 to TOU0_5 are output on external pins TO00 to TO05. The initial values of TO00 to TO05 are set in the TOUn output control register.

The value set in the reload register causes TOU0_6 to underflow three-quarters of the way into the carrier period.

In addition, the PWM output-prohibit function is used with external pin input and a port level setting.

Figure 21.48 shows a timing chart of timer operation. Figure 21.49 shows a diagram of timer configuration.

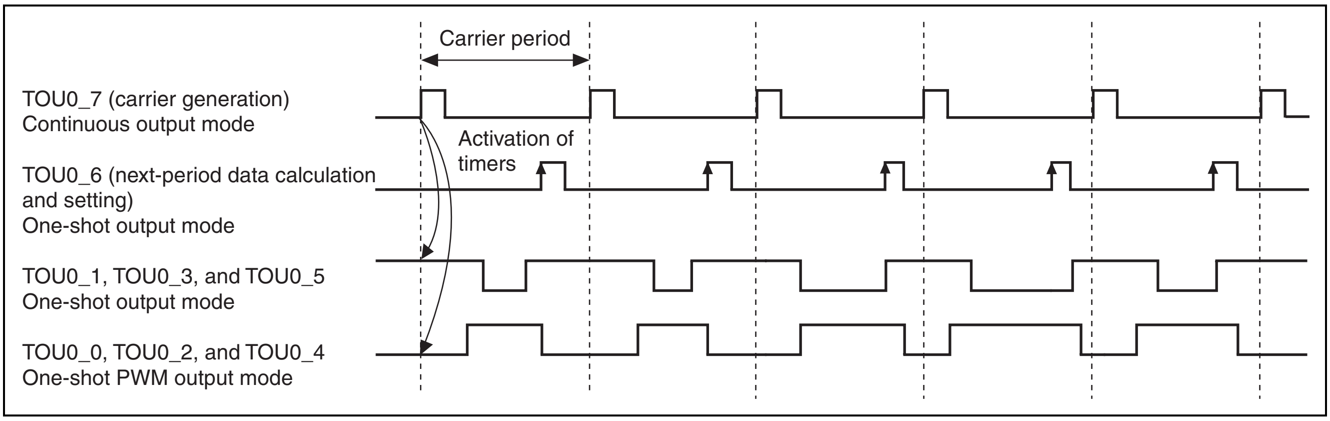

Carrier period
TOU0_7 (carrier generation)
Continuous output mode
Activation of timers
TOU0_6 (next-period data calculation and setting)
One-shot output mode
TOU0_1, TOU0_3, and TOU0_5
One-shot output mode
TOU0_0, TOU0_2, and TOU0_4
One-shot PWM output mode

Figure 21.48 Timer Operation Timing Chart

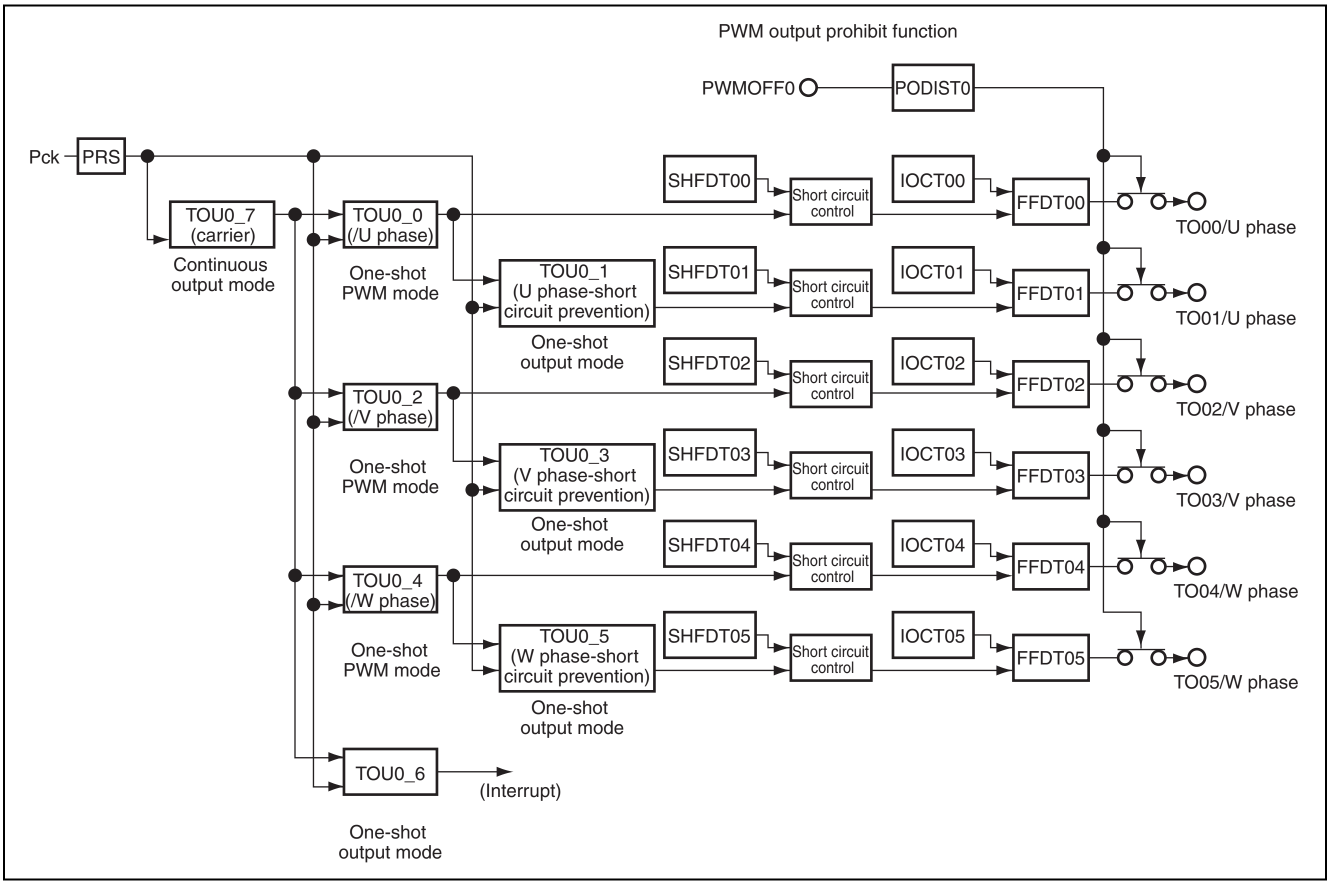


Figure 21.49 Timer Configuration Diagram

This page is blank for reasons of layout.

Section 22 Timer Unit (TMU)

This MCU includes a timer unit (TMU) that consists of three 32-bit timers, TM0 to TM2. In this section, the n in the "TMn" notations specifies the values 0 to 2.

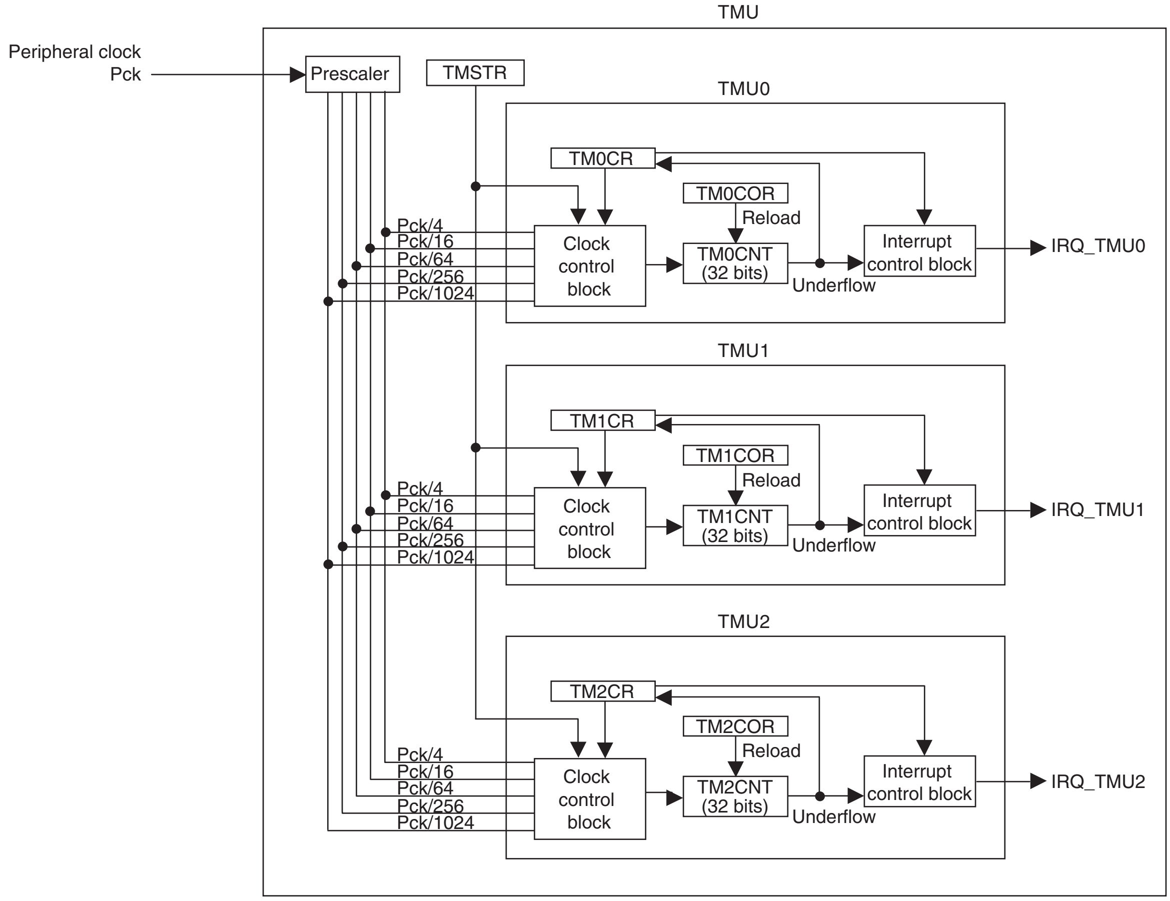
22.1 Overview

- Auto-reload type 32-bit down-counter provided for each channel
- 32-bit timer constant register for auto-reload use, readable/writable at any time, and 32-bit down-counter provided for each channel
- Selection of five counter input clocks: Channel TM0 to TM2

  Five peripheral clocks (Pck/4, Pck/16, Pck/64, Pck/256, and Pck/1024) (Pck is the peripheral clock).
- One interrupt source

  One underflow source (each channel)

Figure 22.1 shows a block diagram of the TMU.



Notes: • IRQ_n indicates an interrupt number, and the same number indicates interrupts in the same group.

Legend:
TMnCNT: TMn counter
TMnCOR: TMn constant register
TMnCR: TMn control register
TMSTR: TM status register
n = 0 to 2

Figure 22.1 Block Diagram of TMU

22.2 Register Descriptions

Table 22.1 shows the TMU register configuration.

Table 22.1 Register Configuration

Channel	Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Common	TM start register	TMSTR	H'00	H'FFFF D004	8	22-2
0	TM0 constant register	TM0COR	H'FFFF FFFF	H'FFFF D008	32	22-3
	TM0 counter	TM0CNT	H'FFFF FFFF	H'FFFF D00C	32	22-3
	TM0 control register	TM0CR	H'0000	H'FFFF D010	16	22-4
1	TM1 constant register	TM1COR	H'FFFF FFFF	H'FFFF D014	32	22-3
	TM1 counter	TM1CNT	H'FFFF FFFF	H'FFFF D018	32	22-3
	TM1 control register	TM1CR	H'0000	H'FFFF D01C	16	22-4
2	TM2 constant register	TM2COR	H'FFFF FFFF	H'FFFF D020	32	22-3
	TM2 counter	TM2CNT	H'FFFF FFFF	H'FFFF D024	32	22-3
	TM2 control register	TM2CR	H'0000	H'FFFF D028	16	22-4

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

22.2.1 TM Start Register (TMSTR)

The TMSTR register selects whether the TMnCNT counters operate or are stopped.

TM Start Register (TMSTR) <P4 address: location H'FFFF D004>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	STR2	STR1	STR0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	STR2	0	R	W	TM2 Counter Start 2 Bit Specifies whether the TM2CNT counter is operated. 0: The TM2CNT counter is stopped 1: The TM2CNT counter is operated
1	STR1	0	R	W	TM1 Counter Start 1 Bit Specifies whether the TM1CNT counter is operated. 0: The TM1CNT counter is stopped 1: The TM1CNT counter is operated
0	STR0	0	R	W	TM0 Counter Start 0 Bit Specifies whether the TM0CNT counter is operated. 0: The TM0CNT counter is stopped 1: The TM0CNT counter is operated

22.2.2 TMn Constant Register (TMnCOR)

The value of the TMnCOR register is loaded into the TMnCNT counter when an underflow occurs as the result of decrementing the TMnCNT counter. The decrementing of the TMnCNT counter then continues starting from the loaded value.

TM0 Constant Register (TM0COR)	<P4 address: location H'FFFF D008>
TM1 Constant Register (TM1COR)	<P4 address: location H'FFFF D014>
TM2 Constant Register (TM2COR)	<P4 address: location H'FFFF D020>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMnCOR															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMnCOR															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

<After Reset: H'FFFF FFFF>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TMnCOR	All 1	R	W	Holds a 32-bit value that will be loaded into the TMnCNT counter when the TMnCNT counter underflows.

22.2.3 TMn Counter (TMnCNT)

The TMnCNT counter is a down counter that is decremented by the input clock signal selected by the TMnCR register TPSC bits.

When an underflow occurs as the result of decrementing the TMnCNT counter, the corresponding channel TMnCR register UNF bit is set to "1". Also, at the same time, the value of the TMnCOR register is loaded into the TMnCNT counter and the decrementing operation continues starting from the loaded value.

TM0 Counter (TM0CNT)	<P4 address: location H'FFFF D00C>
TM1 Counter (TM1CNT)	<P4 address: location H'FFFF D018>
TM2 Counter (TM2CNT)	<P4 address: location H'FFFF D024>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMnCNT															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMnCNT															
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

<After Reset: H'FFFF FFFF>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TMnCNT	All 1	R	W	The value of the 32-bit counter

22.2.4 TMn Control Register (TMnCR)

The TMnCR register selects the counter clock and control the generations of interrupts when the TMnCNT counter underflows.

TM0 Control Register (TM0CR) <P4 address: location H'FFFF D010>
TM1 Control Register (TM1CR) <P4 address: location H'FFFF D01C>
TM2 Control Register (TM2CR) <P4 address: location H'FFFF D028>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UNF	—	—	UNIE	—	—	TPSC		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 9	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
8	UNF	0	R	*[1]	Underflow Flag Status flag that indicates the occurrence of a TMnCNT counter underflow. 0: Indicates that a TMnCNT counter underflow has not occurred. 1: Indicates that a TMnCNT counter underflow has occurred. [Condition for clearing to "0"] • When "0" is written to the UNF bit [Condition for setting to "1"] • When the TMnCNT counter underflows
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	UNIE	0	R	W	Underflow Interrupt Control Bit Controls whether or not the underflow interrupt is enabled when the status flag UNF bit that indicates a TMnCNT counter underflow is set to "1". 0: The underflow interrupt (TUNI) is disabled. 1: The underflow interrupt (TUNI) is enabled.
4, 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	TPSC	000	R	W	Timer Prescaler These bits select the count clock for the TMnCNT counter. 000: Counted by Pck/4 001: Counted by Pck/16 010: Counted by Pck/64 011: Counted by Pck/256 100: Counted by Pck/1024 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited

Note: *1 The original value is retained when "1" is written to this bit.

22.3 Operation

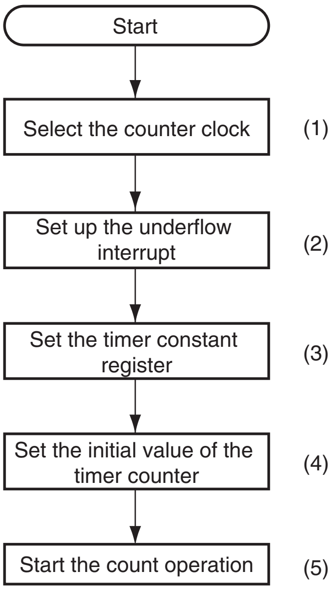
Each channel has a 32-bit timer/counter (TMnCNT) and a 32-bit TMn constant register (TMnCOR). The TMnCNT is a down counter which is decremented. This module supports a periodic counting operation based on an auto-reload function.

22.3.1 Counter Operation

When one of the TMSTR register STR2 to STR0 bits is set to "1", the TMnCNT counter for the corresponding channel starts to decrement. When the TMnCNT counter underflows, the corresponding TMnCR register UNF flag is set to "1". At this time, if the TMnCR register UNIE bit is "1", an interrupt request will be issued to the CPU. Also at this time, the value of the TMnCOR register will be loaded to the TMnCNT counter and the decrementing operation will continue (auto-reload function).

(1) Example of Count Operation Setting Procedure

Figure 22.2 shows an example of the count operation setting procedure.


Start
Select the counter clock
(1)
Set up the underflow interrupt
(2)
Set the timer constant register
(3)
Set the initial value of the timer counter
(4)
Start the count operation
(5)

(1) Select the counter clock with the TMnCR register TPSC[2:0] bits.

(2) Set whether or not an interrupt is generated by the TMnCR register UNIE bit when the TMnCNT counter underflows.

(3) Set the value of the TMnCOR register.

(4) Set the initial value of the TMnCNT counter.

(5) Start the count operation by setting the TMnSTR register STR bit to "1".

Note: When the interrupt occurs, the interrupt handler must clear the flag that corresponds to the interrupt source. If interrupts are enabled without clearing the flag, the interrupt will occur again.

Figure 22.2 Example of Count Operation Setting Procedure

(2) Auto-Reload Count Operation

Figure 22.3 shows the TMnCNT counter auto-reload operation.

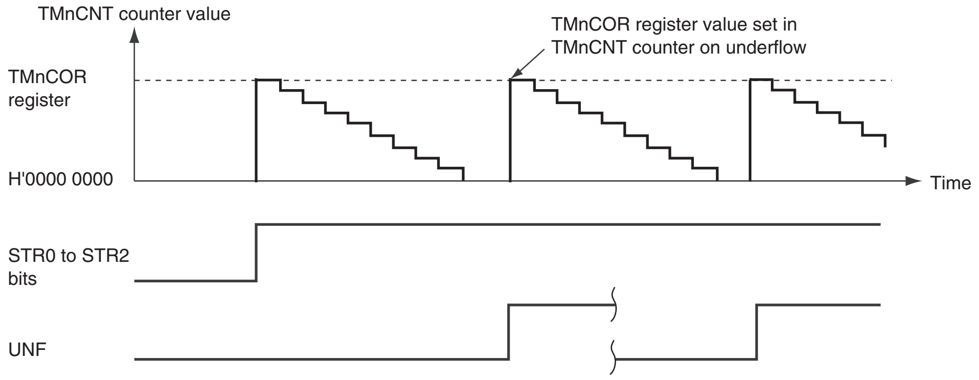


Figure 22.3 TMnCNT Counter Auto-Reload Operation

(3) TMnCNT Counter Count Timing

Any of five count clocks (Pck/4, Pck/16, Pck/64, Pck/256, or Pck/1024) scaled from the peripheral clock can be selected as the count clock by the TMnCR register TPSC bits.

Figure 22.4 shows the timing in this case.

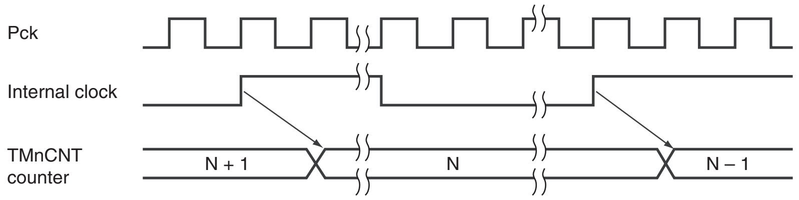


Figure 22.4 Count Timing when Operating on Internal Clock

22.4 Interrupts

The TMU interrupt is an underflow interrupt. Each channel can generate this underflow interrupt.

An underflow interrupt request is generated (for each channel) when both the UNF bit and the interrupt enable bit (UNIE) for that channel are set to "1".

The TMU interrupt sources are summarized in table 22.2.

Table 22.2 TMU Interrupt Sources

Channel	Interrupt Source	Description
0	TUNI0	Underflow interrupt 0
1	TUNI1	Underflow interrupt 1
2	TUNI2	Underflow interrupt 2

22.5 Usage Notes

22.5.1 Register Writes

When writing to a TMU register, timer count operation must be stopped by clearing the TMSTR register start bit (STR2 to STR0) for the corresponding channel.

Note, however, that the TMSTR register may be written and the TMnCR register UNF bit may be cleared while a count operation is in progress. When clearing a flag (UNF) during a count operation, be sure not to change the values of any other bits.

22.5.2 Reading from TMnCNT Counter

Reading from the TMnCNT counter is performed synchronously with the timer count operation. Note that when the timer count operation is performed simultaneously with reading from a register, the synchronous processing causes the TMnCNT counter value before the count-down operation to be read as the TMnCNT counter value.

Section 23 Serial Communication Interface with FIFO (SCIF)

This MCU has a four-channel (SCIF0 to SCIF3) serial communication interface with FIFO (SCIF) that supports both asynchronous and clock synchronous serial communication. It also has 16-stage FIFO registers for both transmission and reception independently for each channel that enable this MCU to perform efficient high-speed continuous communication. Note that the "i" used in the "SCi" register, pin, and signal names in this chapter indicates the numbers 0 to 3, corresponding to the four SCI channels. (For pin specifications, see table 23.1.)

23.1 Overview

- Asynchronous serial communication:
  - Serial data communication is performed by start-stop in character units. The SCIF can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other communications chip that employs a standard asynchronous serial system. There are eight selectable serial data communication formats.
  - Data length: 7 or 8 bits
  - Stop bit length: 1 or 2 bits
  - Parity: Even, odd, or none
  - Receive error detection: Parity, framing, and overrun errors
  - Break detection: Break is detected when a framing error is followed by at least one frame at the space 0 level ("L" level). It is also detected by reading the RxDi level directly from the SCi serial port register when a framing error occurs.
- Clock synchronous serial communication:
  - Serial data communication is synchronized with a clock signal. The SCIF can communicate with other chips having a clock synchronous communication function. There is one serial data communication format.
  - Data length: 8 bits
  - Receive error detection: Overrun errors
- Full duplex communication: The transmitting and receiving sections are independent, so the SCIF can transmit and receive simultaneously. Both sections use 16-stage FIFO buffering, so high-speed continuous data transfer is possible in both the transmit and receive directions.
- On-chip baud rate generator with selectable bit rates
- Internal or external transmit/receive clock source: From either baud rate generator (internal) or SCKi pin (external)
- Four types of interrupts: Transmit-FIFO-data-empty interrupt, break interrupt, receive-FIFO-data-full interrupt, and receive-error interrupts are requested independently.
- In asynchronous mode, on-chip modem control functions (using the RTSi# and CTSi# pins).
- The quantity of data in the transmit and SCi receive FIFO data registers and the number of receive errors of the SCi receive data in the receive FIFO data register can be ascertained.
- A time-out error (DR) can be detected when receiving in asynchronous mode.
- In asynchronous mode, the base clock frequency can be either 16 or 8 times the bit rate.
- When an internal clock is selected as a clock source and the SCKi pin is used as an input pin in asynchronous mode, either normal mode or double-speed mode can be selected for the baud rate generator.
- Maximum transfer rate in clock synchronous mode : 3.3 Mbps.
- Maximum transfer rate in asynchronous mode : 5 Mbps.

Figure 23.1 shows a block diagram of the SCIFi.

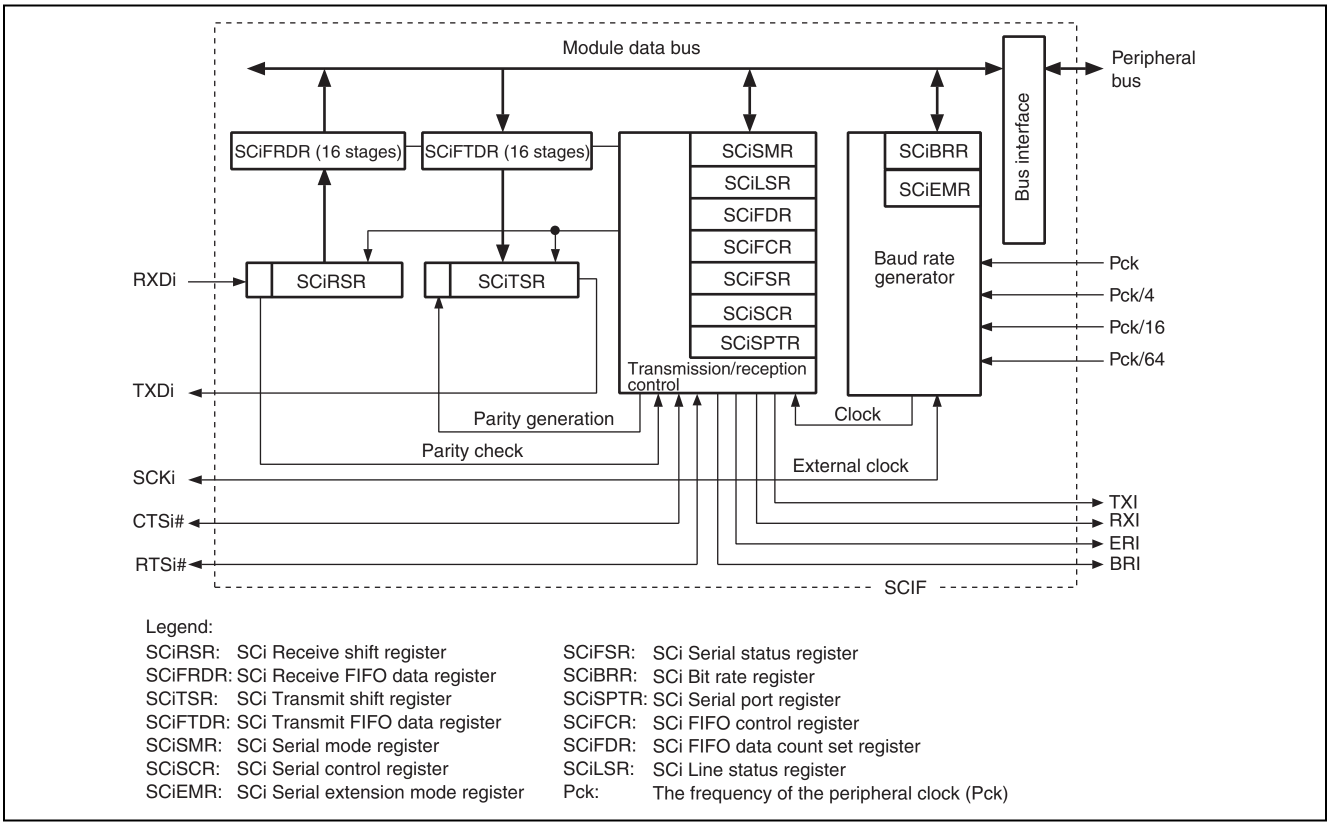

Module data bus
Peripheral bus
Bus interface
SCiFRDR (16 stages)
SCiFTDR (16 stages)
SCiSMR
SCiLSR
SCiFDR
SCiFCR
SCiFSR
SCiSCR
SCiSPTR
Transmission/reception control
SCiBRR
SCiEMR
Baud rate generator
Pck
Pck/4
Pck/16
Pck/64
RXDi
SCiRSR
SCiTSR
TXDi
Parity generation
Parity check
Clock
External clock
SCKi
CTSi#
RTSi#
TXI
RXI
ERI
BRI
SCIF
Legend:
SCiRSR: SCi Receive shift register
SCiFRDR: SCi Receive FIFO data register
SCiTSR: SCi Transmit shift register
SCiFTDR: SCi Transmit FIFO data register
SCiSMR: SCi Serial mode register
SCiSCR: SCi Serial control register
SCiEMR: SCi Serial extension mode register
SCiFSR: SCi Serial status register
SCiBRR: SCi Bit rate register
SCiSPTR: SCi Serial port register
SCiFCR: SCi FIFO control register
SCiFDR: SCi FIFO data count set register
SCiLSR: SCi Line status register
Pck: The frequency of the peripheral clock (Pck)

Figure 23.1 Block Diagram of SCIF

23.2 Input/Output Pins

Table 23.1 shows the pin configuration of the SCIFi.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 23.1 Pin Configuration

Channel	Symbol	I/O	Function
0	SCK0	I/O	Clock I/O
	RXD0	Input	Receive data input
	TXD0	Output	Transmit data output
	RTS0#	I/O	Request to send
	CTS0#	I/O	Clear to send
1*[1]	SCK1	I/O	Clock I/O
	RXD1	Input	Receive data input
	TXD1	Output	Transmit data output
2	SCK2	I/O	Clock I/O
	RXD2	Input	Receive data input
	TXD2	Output	Transmit data output
	RTS2#	I/O	Request to send
	CTS2#	I/O	Clear to send
3*[1]	RXD3	Input	Receive data input
	TXD3	Output	Transmit data output
	RTS3#	I/O	Request to send
	CTS3#	I/O	Clear to send

Note: *1 Pins RTS1#, CTS1#, and SCK3 are not implemented in this MCU.

23.3 Register Descriptions

Table 23.2 shows the SCIFi register configuration.

Table 23.2 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
SC0 Serial mode register	SC0SMR	H'0000	H'FFFF C000	16	23-8
SC0 Bit rate register	SC0BRR	H'FF	H'FFFF C004	8	23-19
SC0 Serial control register	SC0SCR	H'0000	H'FFFF C008	16	23-10
SC0 Transmit FIFO data register	SC0FTDR	Undefined	H'FFFF C00C	8	23-7
SC0 Serial status register	SC0FSR	H'0060	H'FFFF C010	16	23-13
SC0 Receive FIFO data register	SC0FRDR	Undefined	H'FFFF C014	8	23-6
SC0 FIFO control register	SC0FCR	H'0000	H'FFFF C018	16	23-23
SC0 FIFO data count set register	SC0FDR	H'0000	H'FFFF C01C	16	23-25
SC0 Serial port register	SC0SPTR	H'0050	H'FFFF C020	16	23-26
SC0 Line status register	SC0LSR	H'0000	H'FFFF C024	16	23-28
SC0 Serial extension mode register	SC0EMR	H'0000	H'FFFF C028	16	23-29
SC1 Serial mode register	SC1SMR	H'0000	H'FFFF C100	16	23-8
SC1 Bit rate register	SC1BRR	H'FF	H'FFFF C104	8	23-19
SC1 Serial control register	SC1SCR	H'0000	H'FFFF C108	16	23-10
SC1 Transmit FIFO data register	SC1FTDR	Undefined	H'FFFF C10C	8	23-7
SC1 Serial status register	SC1FSR	H'0060	H'FFFF C110	16	23-13
SC1 Receive FIFO data register	SC1FRDR	Undefined	H'FFFF C114	8	23-6
SC1 FIFO control register	SC1FCR	H'0000	H'FFFF C118	16	23-23
SC1 FIFO data count set register	SC1FDR	H'0000	H'FFFF C11C	16	23-25
SC1 Serial port register	SC1SPTR	H'0050	H'FFFF C120	16	23-26
SC1 Line status register	SC1LSR	H'0000	H'FFFF C124	16	23-28
SC1 Serial extension mode register	SC1EMR	H'0000	H'FFFF C128	16	23-29
SC2 Serial mode register	SC2SMR	H'0000	H'FFFF C200	16	23-8
SC2 Bit rate register	SC2BRR	H'FF	H'FFFF C204	8	23-19
SC2 Serial control register	SC2SCR	H'0000	H'FFFF C208	16	23-10
SC2 Transmit FIFO data register	SC2FTDR	Undefined	H'FFFF C20C	8	23-7
SC2 Serial status register	SC2FSR	H'0060	H'FFFF C210	16	23-13
SC2 Receive FIFO data register	SC2FRDR	Undefined	H'FFFF C214	8	23-6
SC2 FIFO control register	SC2FCR	H'0000	H'FFFF C218	16	23-23
SC2 FIFO data count set register	SC2FDR	H'0000	H'FFFF C21C	16	23-25
SC2 Serial port register	SC2SPTR	H'0050	H'FFFF C220	16	23-26
SC2 Line status register	SC2LSR	H'0000	H'FFFF C224	16	23-28
SC2 Serial extension mode register	SC2EMR	H'0000	H'FFFF C228	16	23-29
SC3 Serial mode register	SC3SMR	H'0000	H'FFFF C300	16	23-8
SC3 Bit rate register	SC3BRR	H'FF	H'FFFF C304	8	23-19
SC3 Serial control register	SC3SCR	H'0000	H'FFFF C308	16	23-10
SC3 Transmit FIFO data register	SC3FTDR	Undefined	H'FFFF C30C	8	23-7

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
SC3 Serial status register	SC3FSR	H'0060	H'FFFF C310	16	23-13
SC3 Receive FIFO data register	SC3FRDR	Undefined	H'FFFF C314	8	23-6
SC3 FIFO control register	SC3FCR	H'0000	H'FFFF C318	16	23-23
SC3 FIFO data count set register	SC3FDR	H'0000	H'FFFF C31C	16	23-25
SC3 Serial port register	SC3SPTR	H'0050	H'FFFF C320	16	23-26
SC3 Line status register	SC3LSR	H'0000	H'FFFF C324	16	23-28
SC3 Serial extension mode register	SC3EMR	H'0000	H'FFFF C328	16	23-29

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

23.3.1 SCi Receive Shift Register (SCiRSR)

The SCiRSR register is an internal register used to receive serial data. Data input at the RXDi pin is loaded into the SCiRSR register in the order received, LSB (bit 0) first, converting the data to parallel form. When one byte has been received, it is automatically transferred to the SCi receive FIFO data register (SCiFRDR).

The CPU cannot read or write to the SCiRSR register directly.

SC0 Receive Shift Register (SC0RSR)
SC1 Receive Shift Register (SC1RSR)
SC2 Receive Shift Register (SC2RSR)
SC3 Receive Shift Register (SC3RSR)

Bit:	7	6	5	4	3	2	1	0
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

23.3.2 SCi Receive FIFO Data Register (SCiFRDR)

The SCiFRDR register is a 16-byte FIFO register that stores serial receive data. The SCIFi completes the reception of one byte of serial data by moving the received data from the SCi receive shift register (SCiRSR) into SCiFRDR for storage. Continuous reception is possible until 16 bytes are stored.
SCiFRDR is a read-only register; it cannot be written. If data is read when there is no receive data in the SCFRDR, the value is undefined.

When the SCiFRDR register has been filled with received data, subsequently received serial data will be lost.

SC0 Receive FIFO Data Register (SC0FRDR)	<P4 address: location H'FFFF C014>
SC1 Receive FIFO Data Register (SC1FRDR)	<P4 address: location H'FFFF C114>
SC2 Receive FIFO Data Register (SC2FRDR)	<P4 address: location H'FFFF C214>
SC3 Receive FIFO Data Register (SC3FRDR)	<P4 address: location H'FFFF C314>

Bit:	7	6	5	4	3	2	1	0
	SCiFRDR							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	SCiFRDR	Undefined	R	N	This is a 16-stage FIFO register that hold received serial data.

23.3.3 SCi Transmit Shift Register (SCiTSR)

The SCiTSR register is an internal register used to transmit serial data. The SCIFi loads transmit data from the SCi transmit FIFO data register (SCiFTDR) into the SCiTSR register, then transmits the data serially from the TXDi pin, LSB (bit 0) first. After transmitting one data byte, the SCIF automatically loads the next transmit data from the SCiFTDR register into SCiTSR and starts transmitting again.

The CPU cannot read from or write to the SCiTSR register directly.

SC0 Transmit Shift Register (SC0TSR)
SC1 Transmit Shift Register (SC1TSR)
SC2 Transmit Shift Register (SC2TSR)
SC3 Transmit Shift Register (SC3TSR)

Bit:	7	6	5	4	3	2	1	0
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

23.3.4 SCi Transmit FIFO Data Register (SCiFTDR)

The SCiFTDR register is a 16-byte FIFO register that stores data for serial transmission. When the SCIF detects that the SCi transmit shift register (SCiTSR) is empty, it moves transmit data written in the SCiFTDR register into the SCiTSR register and starts serial transmission. Continuous serial transmission is performed until there is no transmit data left in the SCiFTDR register. The SCiFTDR register is write-only register. Do not write to the SCiFTDR register when the value of the TE bit in SCiSCR is "0" (transmit operation disabled).

When the SCiFTDR register is full of transmit data (16 bytes), no more data can be written. If writing of new data is attempted, the data is ignored.

SC0 Transmit FIFO Data Register (SC0FTDR)	<P4 address: location H'FFFF C00C>
SC1 Transmit FIFO Data Register (SC1FTDR)	<P4 address: location H'FFFF C10C>
SC2 Transmit FIFO Data Register (SC2FTDR)	<P4 address: location H'FFFF C20C>
SC3 Transmit FIFO Data Register (SC3FTDR)	<P4 address: location H'FFFF C30C>

Bit:	7	6	5	4	3	2	1	0
	SCiFTDR							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	SCiFTDR	Undefined	R	W	This is a 16-stage FIFO register that hold serial transmit data.

23.3.5 SCi Serial Mode Register (SCiSMR)

The SCiSMR register specifies the SCIFi serial communication format and selects the clock source for the baud rate generator.

SC0 Serial Mode Register (SC0SMR)	<P4 address: location H'FFFF C000>
SC1 Serial Mode Register (SC1SMR)	<P4 address: location H'FFFF C100>
SC2 Serial Mode Register (SC2SMR)	<P4 address: location H'FFFF C200>
SC3 Serial Mode Register (SC3SMR)	<P4 address: location H'FFFF C300>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CA	CHR	PE	OE	STOP	—	CKS	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7	CA	0	R	W	Communication Mode Bit Selects whether the SCIFi operates in asynchronous or clock synchronous mode. 0: Asynchronous mode 1: Clock synchronous mode*[1] Note: *1 Not setting in the unit SC3.
6	CHR	0	R	W	Character Length Bit Selects 7-bit or 8-bit data length in asynchronous mode. In the clock synchronous mode, the data length is always 8 bits, regardless of the CHR bit setting. 0: 8-bit data 1: 7-bit data*[1] Note: *1 When 7-bit data is selected, the MSB (bit 7) of the SCi transmit FIFO data register is not transmitted.
5	PE	0	R	W	Parity Enable Bit Selects whether to add a parity bit to transmit data and to check the parity of receive data, in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the PE setting. 0: Parity bit is not added or checked 1: Parity bit is added or checked*[1] Note: *1 When PE is set to "1", an even or odd parity bit is added to transmit data, depending on the parity mode (OE) setting. Receive data parity is checked according to the even/odd (OE) mode setting.

Bit	Abbreviation	After Reset	R	W	Description
4	OE	0	R	W	Parity Mode Bit Selects even or odd parity when parity bits are added and checked. The OE setting is used only in asynchronous mode and only when the parity enable bit (PE) is set to "1" to enable parity addition and checking. The OE setting is ignored in clock synchronous mode, or in asynchronous mode when parity addition and checking is disabled. 0: Even parity*[1] 1: Odd parity*[2] Notes: *1 If even parity is selected, the parity bit is added to transmit data to make an even number of "1s" in the transmitted character and parity bit combined. Receive data is checked to see if it has an even number of "1s" in the received character and parity bit combined. *2 If odd parity is selected, the parity bit is added to transmit data to make an odd number of "1s" in the transmitted character and parity bit combined. Receive data is checked to see if it has an odd number of "1s" in the received character and parity bit combined.
3	STOP	0	R	W	Stop Bit Length Bit Selects one 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. 0: One stop bit When transmitting, a single "1" bit is added at the end of each transmitted character. 1: Two stop bits When transmitting, two "1" bits are added at the end of each transmitted character.
2	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
1, 0	CKS	00	R	W	Clock Select Bits Select the internal clock source of the on-chip baud rate generator. For further information on the clock source, SCi bit rate register settings, and baud rate, see section 23.3.8, SCi Bit Rate Register (SCiBRR). 00: Pck clock 01: Pck/4 clock 10: Pck/16 clock 11: Pck/64 clock

23.3.6 SCi Serial Control Register (SCiSCR)

The SCiSCR register operates the SCIF transmitter/receiver, enables/disables interrupt requests, and selects the transmit/receive clock source.

SC0 Serial Control Register (SC0SCR)	<P4 address: location H'FFFF C008>
SC1 Serial Control Register (SC1SCR)	<P4 address: location H'FFFF C108>
SC2 Serial Control Register (SC2SCR)	<P4 address: location H'FFFF C208>
SC3 Serial Control Register (SC3SCR)	<P4 address: location H'FFFF C308>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	—	CKE	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7	TIE	0	R	W	Transmit Interrupt Enable Bit Enables or disables the transmit-FIFO-data-empty interrupt (TXI) requested when the serial transmit data is transferred from the SCi transmit FIFO data register (SCiFTDR) to the SCi transmit shift register (SCiTSR), when the quantity of data in the SCi transmit FIFO register becomes less than the specified number of transmission triggers, and when the TDFE flag in the serial status register (SCiFSR) is set to "1". The TIE bit only enables or disables the TXI request. As long as TXI is not cancelled, the TXI interrupt is generated whenever the request is enabled. 0: Transmit-FIFO-data-empty interrupt request (TXI) is disabled 1: Transmit-FIFO-data-empty interrupt request (TXI) is enabled*1 Note: *1 The TXI interrupt request can be cleared by writing a greater quantity of transmit data than the specified transmission trigger number to the SCiFTDR register and by clearing TDFE to "0" after reading "1" from TDFE, or can be cleared by clearing TIE to "0". For the relationship with the DMAC, see section 23.5, SCIFi Interrupt Sources and DMAC.

Bit	Abbreviation	After Reset	R	W	Description
6	RIE	0	R	W	Receive Interrupt Enable Bit Enables or disables the receive FIFO data full (RXI) interrupts requested when the RDF flag or DR flag in SCi serial status register (SCiFSR) is set to "1", receive-error (ERI) interrupts requested when the ER flag in the SCiFSR register is set to "1", and break (BRI) interrupts requested when the BRK flag in the SCiFSR register or the ORER flag in SCi line status register (SCiLSR) is set to "1". The RIE bit only enables or disables RXI, ERI, and BRI requests. As long as RXI, ERI, and BRI are not cleared, RXI, ERI, and BRI interrupts are generated when the requests are enabled. 0: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled 1: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled*1 Note: *1 RXI interrupt requests can be cleared by reading the DR or RDF flag after it has been set to "1", then clearing the flag to "0", or by clearing RIE to "0". ERI or BRI interrupt requests can be cleared by reading the ER, BR or ORER flag after it has been set to "1", then clearing the flag to "0", or by clearing RIE and REIE to "0". For the relationship with the DMAC, see section 23.5, SCIFi Interrupt Sources and DMAC.
5	TE	0	R	W	Transmit Enable Bit Enables or disables the serial transmitter. Do not clear the TE bit during transmission. 0: Transmitter disabled 1: Transmitter enabled*1 Note: *1 Serial transmission starts after writing of transmit data into the SCiFTDR register. Select the transmit format in the SCiSMR and SCiFCR registers and reset the transmit FIFO before setting TE to "1".
4	RE	0	R	W	Receive Enable Bit Enables or disables the serial receiver. Do not clear the RE bit during reception. 0: Receiver disabled*1 1: Receiver enabled*2 Notes: *1 Clearing RE to "0" does not affect the receive flags (DR, ER, BRK, RDF, FER, PER, and ORER). These flags retain their previous values. *2 Serial reception starts when a start bit is detected in asynchronous mode, or synchronous clock is detected in clock synchronous mode. Note that the following must be performed before setting the RE bit to "1": set the SCi serial mode register (SCiSMR) and the SCiFIFO control register (SCiFCR) as required, determine the reception format, and reset the receive FIFO.

Bit	Abbreviation	After Reset	R	W	Description
3	REIE	0	R	W	Receive Error Interrupt Enable Bit Enables or disables the receive-error (ERI) interrupts and break (BRI) interrupts. The setting of REIE bit is valid only when RIE bit is set to "0". 0: Receive-error interrupt (ERI) and break interrupt (BRI) requests are disabled 1: Receive-error interrupt (ERI) and break interrupt (BRI) requests are enabled*1 Note: *1 ERI or BRI interrupt requests can be cleared by reading the ER, BR or ORER flag after it has been set to "1", then clearing the flag to "0", or by clearing RIE and REIE bits to "0". Even if RIE bit is set to "0", when REIE bit is set to "1", ERI or BRI interrupt requests are enabled. Set this bit to report ERI and BRI interrupt requests to the interrupt controller during DMA transfers. For the relationship with the DMAC, see section 23.5, SCIFi Interrupt Sources and DMAC.
2	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
1, 0	CKE	00	R	W	Clock Enable Bits Select the SCIFi clock source and enable or disable clock output from the SCKi pin. Depending on CKE bit, the SCKi pin can be used for serial clock output or serial clock input. If serial clock output is set in clock synchronous mode, set the CA bit in the SCiSMR register to "1", and then set CKE bit. • Asynchronous mode 00: Internal clock, SCKi pin used for input pin (input signal is ignored) 01: Internal clock, SCKi pin used for clock output (The output clock frequency is either 16 or 8 times the bit rate.) 10: External clock, SCKi pin used for clock input (The input clock frequency is either 16 or 8 times the bit rate.) 11: Setting prohibited • Clock synchronous mode 0x: Internal clock, SCKi pin used for serial clock output 10: External clock, SCKi pin used for serial clock input 11: Setting prohibited

Legend:

x: Don't care

23.3.7 SCi Serial Status Register (SCiFSR)

The upper 8 bits indicate the number of receive errors in the SCi receive FIFO data register, and the lower 8 bits indicate the status flag indicating SCIF operating state.

The value "1" cannot be written to the ER, TEND, TDFE, BRK, RDF, and DR status flags. Also note that these flags cannot be cleared unless the value "1" has been read out first. The PER bits, the PERNUM bits, the FER bit, and the FERNUM bits are read only and cannot be written.

SC0 Serial Status Register (SC0FSR)	<P4 address: location H'FFFF C010>
SC1 Serial Status Register (SC1FSR)	<P4 address: location H'FFFF C110>
SC2 Serial Status Register (SC2FSR)	<P4 address: location H'FFFF C210>
SC3 Serial Status Register (SC3FSR)	<P4 address: location H'FFFF C310>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PERNUM				FERNUM				ER	TEND	TDFE	BRK	FER	PER	RDF	DR
After Reset:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0

<After Reset: H'0060>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	PERNUM	0000	R	0	Number of Parity Errors Bits Indicate the quantity of data including a parity error in the receive data stored in the SCi receive FIFO data register (SCiFRDR). The value indicated by bits 15 to 12 after the ER bit in the SCiFSR register is set, represents the number of parity errors in the SCiFRDR register. When parity errors have occurred in all 16-byte receive data in SCFRDR, the PERNUM bit shows "0000".
11 to 8	FERNUM	0000	R	0	Number of Framing Errors Bits Indicate the quantity of data including a framing error in the receive data stored in the SCi receive FIFO data register (SCiFRDR). The value indicated by bits 11 to 8 after the ER bit in the SCiFSR register is set, represents the number of framing errors in the SCiFRDR register. When framing errors have occurred in all 16-byte receive data in the SCFRDR register, the FERNUM bit shows "0000".

Bit	Abbreviation	After Reset	R	W	Description
7	ER	0	R	*1	Receive Error Flag Indicates the occurrence of a framing error, or of a parity error when receiving data that includes parity.*2 0: Receiving is in progress or has ended normally 1: Indicates that either a framing error or a parity error occurred during reception [Conditions for clearing to "0"] • ER is cleared to "0" when "0" is written after "1" is read from ER [Conditions for setting to "1"] • ER is set to "1" when the stop bit is "0" after checking whether or not the last stop bit of the received data is "1" at the end of one data receive operation*3 • ER is set to "1" when the total number of "1s" in the receive data plus parity bit does not match the even/odd parity specified by the OE bit in the SCiSMR register Notes: *2 Clearing the RE bit to "0" in the SCiSCR register does not affect the ER bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to the SCiFRDR register and the receive operation is continued. Whether or not the data read from the SCiFRDR register includes a receive error can be detected by the FER and PER bits in the SCiFSR register. *3 In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.
6	TEND	1	R	*1	Transmit End Flag Indicates that when the last bit of a serial character was transmitted, the SCiFTDR register did not contain valid data, so transmission has ended. 0: Transmission is in progress 1: Transmission has completed [Condition for clearing to "0"] • TEND is cleared to "0" when "0" is written after "1" is read from TEND after transmit data is written in the SCiFTDR register 1: End of transmission [Conditions for setting to "1"] • When the SCiSCR register TE bit is "0" • When there is no transmit data in the SCiFTDR register when the last bit in a one-byte serially transmitted character is transmitted

Bit	Abbreviation	After Reset	R	W	Description
5	TDFE	1	R	*1	Transmit FIFO Data Empty Flag Indicates that data has been transferred from the transmit FIFO data register (SCiFTDR) to the transmit shift register (SCiTSR), the quantity of data in the SCiFTDR register has become less than the transmission trigger number specified by the TTRG bit in the FIFO control register (SCiFCR), and writing of transmit data to the SCiFTDR register is enabled. 0: The quantity of transmit data written to the SCiFTDR register is greater than the specified transmission trigger number 1: Indicates that the number of transmit data items written to the SCiFTDR register is less than or equal to the specified transmit trigger count*2. [Conditions for clearing to "0"] • TDFE is cleared to "0" when data exceeding the specified transmission trigger number is written to the SCiFTDR register after "1" is read from the TDFE flag and then "0" is written • TDFE is cleared to "0" when DMAC is activated by transmit FIFO data empty interrupt (TXI) and write data exceeding the specified transmission trigger number to the SCiFTDR register [Conditions for setting to "1"] • TDFE is set to "1" when the quantity of transmit data in the SCiFTDR register becomes less than or equal to the specified transmission trigger number as a result of transmission Note: *2 Since the SCiFTDR register is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is "1" is "16 minus the specified transmission trigger number". If an attempt is made to write additional data, the data is ignored. The quantity of data in the SCiFTDR register is indicated by the upper 8 bits of the SCiFDR register.
4	BRK	0	R	*1	Break Detection Bit Indicates that a break signal has been detected in receive data. 0: No break signal received 1: A break signal was received*2 [Conditions for clearing to "0"] • BRK is cleared to "0" when "0" is written after "1" is read from BRK 1: Break signal received [Condition for setting to "1"] • BRK is set to "1" when data including a framing error is received, and then a framing error is followed by at least one frame at the space "0" level ("L" level). Note: *2 When a break is detected, transfer of the receive data (H'00) to the SCiFRDR register stops after detection. When the break ends and the receive signal becomes mark "1", the transfer of receive data resumes.

Bit	Abbreviation	After Reset	R	W	Description
3	FER	0	R	0	Framing Error Indication Flag Indicates a framing error in the data read from the next SCi receive FIFO data register (SCiFRDR) in asynchronous mode. 0: No receive framing error occurred in the next data read from the SCiFRDR register 1: Indicates that a framing error occurred for the next receive data that will be read from the SCiFRDR register [Conditions for clearing to "0"] • FER is cleared to "0" when no framing error is present in the next data read from the SCiFRDR register [Conditions for setting to "1"] • FER is set to "1" when a framing error is present in the next data read from the SCFRDR register
2	PER	0	R	0	Parity Error Indication Flag Indicates a parity error in the data read from the next SCi receive FIFO data register (SCiFRDR) in asynchronous mode. 0: No receive parity error occurred in the next data read from the SCiFRDR register 1: Indicates that a parity error occurred for the next receive data that will be read from the SCiFRDR register [Conditions for clearing to "0"] • PER is cleared to "0" when no parity error is present in the next data read from the SCiFRDR register [Condition for setting to "1"] • PER is set to "1" when a parity error is present in the next data read from the SCFRDR register

Bit	Abbreviation	After Reset	R	W	Description
1	RDF	0	R	*1	Receive FIFO Data Full Flag Indicates that receive data has been transferred to the SCi receive FIFO data register (SCiFRDR), and the quantity of data in the SCiFRDR register has become more than the receive trigger number specified by the RTRG bit in the SCi FIFO control register (SCiFCR). 0: The quantity of transmit data written to the SCiFRDR register is less than the specified receive trigger number 1: Indicates that the number of SCiFRDR register receive data items is greater than or equal to the specified receive trigger count [Conditions for clearing to "0"] • RDF is cleared to "0" when the SCiFRDR register is read until the quantity of receive data in the SCiFRDR register becomes less than the specified receive trigger number after "1" is read from RDF, and then "0" is written • RDF is cleared to "0" when DMAC is activated by receive FIFO data full interrupt (RXI) and read the SCiFRDR register until the quantity of receive data in the SCiFRDR register becomes less than the specified receive trigger number [Condition for setting to "1"] • RDF is set to "1" when a quantity of receive data more than or equal to the specified receive trigger number is stored in the SCiFRDR register*2 Note: *2 As the SCiFRDR register is a 16-byte FIFO register, the maximum quantity of data that can be read when RDF is "1" becomes the specified receive trigger number. If an attempt is made to read after all the data in the SCiFRDR register has been read, the data is undefined. The quantity of receive data in the SCiFRDR register is indicated by the lower 8 bits of the SCiFDR register.

Bit	Abbreviation	After Reset	R	W	Description
0	DR	0	R	*1	Receive Data Ready Flag Indicates that the quantity of data in the SCi receive FIFO data register (SCiFRDR) is less than the specified receive trigger number, and that the next data has not yet been received after the elapse of 15 ETU from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1. 0: Receiving is in progress, or no receive data remains in the SCiFRDR register after receiving ended normally 1: Indicates that the next receive data has not been received [Conditions for clearing to "0"] • After DR has been read out in the state where it is "1", read out all the receive data in the SCiFRDR register and write "0" to DR. • When the DMAC is has been started by the receive FIFO data full interrupt (RXI) and all the receive data has been read out from the SCiFRDR register. [Condition for setting to "1"] • DR is set to "1" when the SCiFRDR register contains less data than the specified receive trigger number, and the next data has not yet been received after the elapse of 15 ETU from the last stop bit.*2 Note: *2 This is equivalent to 1.5 frames with the 8-bit, 1-stop-bit format. (ETU: elementary time unit = 1-bit transfer period)

Note: *1 Only "0" can be written to clear the flag after "1" is read.

23.3.8 SCi Bit Rate Register (SCiBRR)

The SCiBRR register is an 8-bit register that sets the serial transmission and reception bit rate in conjunction with the SCi serial mode register (SCiSMR) CKS bit and the SCi serial extended mode register (SCiEMR) BGDM and ABCS bits.

Each channel has independent baud rate generator control, so different values can be set in four channels.

SC0 Bit Rate Register (SC0BRR)	<P4 address: location H'FFFF C004>
SC1 Bit Rate Register (SC1BRR)	<P4 address: location H'FFFF C104>
SC2 Bit Rate Register (SC2BRR)	<P4 address: location H'FFFF C204>
SC3 Bit Rate Register (SC3BRR)	<P4 address: location H'FFFF C304>

Bit:	7	6	5	4	3	2	1	0
	SCiBRR							
After Reset:	1	1	1	1	1	1	1	1

<After Reset: H'FF>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	SCiBRR	All 1	R	W	Baud rate generator setting (0 ≤ N ≤ 255)

The SCiBRR register setting is calculated as follows:

- Asynchronous mode:

When baud rate generator operates in normal mode (when the BGDM bit of the SCiEMR regiater is "0"):

$$N = \frac{Pck}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$ (Operation on a base clock with a frequency of 16 times the bit rate)

$$N = \frac{Pck}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$ (Operation on a base clock with a frequency of 8 times the bit rate)

When baud rate generator operates in double speed mode (when the BGDM bit of the SCiEMR regiater is "1"):

$$N = \frac{Pck}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$ (Operation on a base clock with a frequency of 16 times the bit rate)

$$N = \frac{Pck}{16 \times 2^{2n-1} \times B} \times 10^6 - 1$$ (Operation on a base clock with a frequency of 8 times the bit rate)

- Clock synchronous mode:

$$N = \frac{Pck}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bits/s)
N: SCiBRR register setting for baud rate generator (0 ≤ N ≤ 255)
(The setting must satisfy the electrical characteristics.)
Pck: Peripheral clock operating frequency (MHz)
n: Baud rate generator clock source (n = 0 to 3) (for the clock sources and values of n, see table 23.3.)

Table 23.3 SCiSMR Register Settings

n	Clock Source	SCiSMR register CKS field settings
0	Pck	00
1	Pck/4	01
2	Pck/16	10
3	Pck/64	11

The bit rate error in asynchronous mode is given by the following formula:

When baud rate generator operates in normal mode (the BGDM bit of the SCiEMR register is "0"):

$$\text{Error (\%)} = \left\{ \frac{Pck \times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$ (Operation on a base clock with a frequency of 16 times the bit rate)

$$\text{Error (\%)} = \left\{ \frac{Pck \times 10^6}{(N+1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$ (Operation on a base clock with a frequency of 8 times the bit rate)

When baud rate generator operates in double speed mode (the BGDM bit of the SCiEMR register is "1"):

$$\text{Error (\%)} = \left\{ \frac{Pck \times 10^6}{(N+1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$ (Operation on a base clock with a frequency of 16 times the bit rate)

$$\text{Error (\%)} = \left\{ \frac{Pck \times 10^6}{(N+1) \times B \times 16 \times 2^{2n-1}} - 1 \right\} \times 100$$ (Operation on a base clock with a frequency of 8 times the bit rate)

Table 23.4 shows the example of bit rate settings in asynchronous mode. Table 23.5 shows the example of bit rate settings in clock-synchronous mode.

Table 23.4 Example Bit Rate Settings in Asynchronous Mode

	BGDM and ABCS in SCiEMR = "0"				BGDM = "0" and ABCS = "1" in SCiEMR, or BGDM = "1" and ABCS = "0" in SCiEMR				BGDM and ABCS in SCiEMR = "1"			
Bit Rate (bps)	**n**	**N**	**Error (%)**	**Actual Bit Rate**	**n**	**N**	**Error (%)**	**Actual Bit Rate**	**n**	**N**	**Error (%)**	**Actual Bit Rate**
110	3	177	-0.25	109.73	—	—	—	—	—	—	—	—
150	3	129	0.16	150.24	3	255	1.73	152.59	—	—	—	—
300	3	64	0.16	300.48	3	129	0.16	300.48	3	255	1.73	305.18
600	2	129	0.16	600.96	3	64	0.16	600.96	3	129	0.16	600.96
1200	2	64	0.16	1201.92	2	129	0.16	1201.92	3	64	0.16	1201.92
2400	1	129	0.16	2403.85	2	64	0.16	2403.85	2	129	0.16	2403.85
4800	1	64	0.16	4807.69	1	129	0.16	4807.69	2	64	0.16	4807.69
9600	0	129	0.16	9615.38	1	64	0.16	9615.38	1	129	0.16	9615.38
19200	0	64	0.16	19230.77	0	129	0.16	19230.77	1	64	0.16	19230.77
31250	0	39	0.00	31250.00	0	79	0.00	31250.00	0	159	0.00	31250.00
38400	0	32	-1.36	37878.79	0	64	0.16	38461.54	0	129	0.16	38461.54
57600	0	21	-1.36	56818.18	0	42	0.94	58139.53	0	86	-0.22	57471.26
62500	0	19	0.00	62500.00	0	39	0.00	62500.00	0	79	0.00	62500.00
115200	0	10	-1.36	113636.36	0	21	-1.36	113636.36	0	42	0.94	116279.07
125000	0	9	0.00	125000.00	0	19	0.00	125000.00	0	39	0.00	125000.00
250000	0	4	0.00	250000.00	0	9	0.00	250000.00	0	19	0.00	250000.00
500000	—	—	—	—	0	4	0.00	500000.00	0	9	0.00	500000.00
625000	0	1	0.00	625000.00	0	3	0.00	625000.00	0	7	0.00	625000.00
1000000	—	—	—	—	—	—	—	—	0	4	0.00	1000000.00
1250000	0	0	0.00	1250000.00	0	1	0.00	1250000.00	0	3	0.00	1250000.00
2500000	—	—	—	—	0	0	0.00	2500000.00	0	1	0.00	2500000.00

Notes:
- Communication at the bit rates listed above is not guaranteed when the indicated settings are used. Make settings so as to satisfy the electrical characteristics of the MCU and the device being communicated with. In addition, make sure to evaluate and verify the environment of the system under development.
- n indicates the setting value of the clock select bits (CKS) in the SCiSMR register.
- N indicates the setting value of baud rate generator in the SCiBRR register.

Table 23.5 Example Bit Rate Settings in Clock-Synchronous Mode

Bit Rate (bps)	n	N	Actual Bit Rate
1000	3	155	1001.60
2500	3	62	2480.16
5000	2	124	5000.00
10000	2	62	9920.63
25000	1	99	25000.00
50000	1	49	50000.00
100000	0	99	100000.00
250000	0	39	250000.00
500000	0	19	500000.00
1000000	0	9	1000000.00
2000000	0	4	2000000.00
2500000	0	3	2500000.00
3333333	0	2	3333333.33

Notes
- Communication at the bit rates listed above is not guaranteed when the indicated settings are used. Make settings so as to satisfy the electrical characteristics of the MCU and the device being communicated with. In addition, make sure to evaluate and verify the environment of the system under development.
- n indicates the setting value of the clock select bits (CKS) in the SCiSMR register.
- N indicates the setting value of baud rate generator in the SCiBRR register.

Table 23.6 indicates the maximum bit rates in asynchronous mode when the baud rate generator is used. Table 23.7 lists the maximum bit rates in asynchronous mode when the external clock input is used. Table 23.8 lists the maximum bit rates in clock synchronous mode when the external clock input is used (when t_{Scyc} = 12tc (Pck)*1).

Note: *1 Make sure that the electrical characteristics of this MCU and that of a connected LSI are satisfied.

Table 23.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Asynchronous Mode)

	Settings				
Pck (MHz)	BGDM Bit	ABCS Bit	n	N	Maximum Bit Rate (bits/s)
40	0	0	0	0	1250000
		1	0	0	2500000
	1	0	0	0	2500000
		1	0	0	5000000

Table 23.7 Maximum Bit Rates with External Clock Input (Asynchronous Mode)

Pck (MHz)	External Input Clock (MHz)	ABCS Bit Settings	Maximum Bit Rate (bits/s)
40	10	0	625000
		1	1250000

Table 23.8 Maximum Bit Rates with External Clock Input (Clock Synchronous Mode, t_{Scyc} = 12tc (Pck))

Pck (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
40	3.3333333	3333333.3

23.3.9 SCi FIFO Control Register (SCiFCR)

The SCiFCR register resets the quantity of data in the SCi transmit and receive FIFO data registers, sets the trigger data quantity, and contains an enable bit for loop-back testing.

SC0 FIFO Control Register (SC0FCR)	<P4 address: location H'FFFF C018>
SC1 FIFO Control Register (SC1FCR)	<P4 address: location H'FFFF C118>
SC2 FIFO Control Register (SC2FCR)	<P4 address: location H'FFFF C218>
SC3 FIFO Control Register (SC3FCR)	<P4 address: location H'FFFF C318>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RSTRG			RTRG		TTRG		MCE	TFRST	RFRST	LOOP
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 8	RSTRG	000	R	W	RTS# Output Active Trigger Bits When the quantity of receive data in the SCi receive FIFO data register (SCiFRDR) becomes more than or equal to the number shown below, RTSi# signal is set to "H" level. 000: 15 001: 1 010: 4 011: 6 100: 8 101: 10 110: 12 111: 14
7, 6	RTRG	00	R	W	Receive FIFO Data Trigger Bits Set the quantity of receive data which sets the receive data full (RDF) flag in the SCi serial status register (SCiFSR). The RDF flag is set to "1" when the quantity of receive data stored in the SCi receive FIFO register (SCiFRDR) is increased more than or equal to the set trigger number shown below. • Asynchronous mode 00: 1 01: 4 10: 8 11: 14 • Clock synchronous mode 00: 1 01: 2 10: 8 11: 14 Note: • In clock synchronous mode, to transfer the receive data using DMAC, set the receive trigger number to "1". If set to other than "1", CPU must read the receive data left in the SCiFRDR register because the RXI interrupt is not generated when the quantity of received data is less than the received trigger number.

Bit	Abbreviation	After Reset	R	W	Description
5, 4	TTRG	00	R	W	Transmit FIFO Data Trigger Bits Set the quantity of remaining transmit data which sets the transmit FIFO data register empty (TDFE) flag in the SCi serial status register (SCiFSR). The TDFE flag is set to "1" when the quantity of transmit data in the SCi transmit FIFO data register (SCiFTDR) becomes less than or equal to the set trigger number shown below. 00: 8 (8) 01: 4 (12) 10: 2 (14) 11: 0 (16) Note: • Values in parentheses mean the number of empty bytes in the SCiFTDR register when the TDFE flag is set to "1".
3	MCE	0	R	W	Modem Control Enable Bit Enables modem control signals CTSi# and RTSi#. For channels 0 to 2 in clock synchronous mode, MCE bit should always be "0". 0: Modem signal disabled*1 1: Modem signal enabled Note: *1 CTSi# is fixed at active "L" regardless of the input value, and RTSi# is also fixed at "L" level.
2	TFRST	0	R	W	SCi Transmit FIFO Data Register Reset Bit Disables the transmit data in the SCi transmit FIFO data register and resets the data to the empty state. 0: Reset operation disabled*1 1: Reset operation enabled Note: *1 Reset operation is executed by a hardware reset.
1	RFRST	0	R	W	SCi Receive FIFO Data Register Reset Bit Disables the receive data in the SCi receive FIFO data register and resets the data to the empty state. 0: Reset operation disabled*1 1: Reset operation enabled Note: *1 Reset operation is executed by a hardware reset.
0	LOOP	0	R	W	Loop-Back Test Bit Internally connects the transmit output pin (TXDi) and receive input pin (RXDi) and internally connects the RTSi# pin and CTSi# pin and enables loop-back testing. 0: Loop back test disabled 1: Loop back test enabled

23.3.10 SCi FIFO Data Count Set Register (SCiFDR)

The SCiFDR register indicates the number of data items stored in the SCi transmit FIFO data register (SCiFTDR) and the SCi receive FIFO data register (SCiFRDR).

It indicates the quantity of transmit data in the SCiFTDR register with the upper 8 bits, and the quantity of receive data in the SCiFRDR register with the lower 8 bits. The SCiFDR register is a read-only register.

SC0 FIFO Data Count Set Register (SC0FDR)	<P4 address: location H'FFFF C01C>
SC1 FIFO Data Count Set Register (SC1FDR)	<P4 address: location H'FFFF C11C>
SC2 FIFO Data Count Set Register (SC2FDR)	<P4 address: location H'FFFF C21C>
SC3 FIFO Data Count Set Register (SC3FDR)	<P4 address: location H'FFFF C31C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—			T			—	—	—			R		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	N	Reserved Bits These bits are always read as "0".
12 to 8	T	00000	R	N	T4 to T0 bits indicate the quantity of non-transmitted data stored in the SCiFTDR register. "H'00" means no transmit data, and "H'10" means that the SCiFTDR register is full of transmit data.
7 to 5	—	All 0	0	N	Reserved Bits These bits are always read as "0".
4 to 0	R	00000	R	N	R4 to R0 bits indicate the quantity of receive data stored in the SCiFRDR register. "H'00" means no receive data, and "H'10" means that the SCiFRDR register full of receive data.

23.3.11 SCi Serial Port Register (SCiSPTR)

The SCiSPTR register controls input/output and data of pins multiplexed to SCIFi function. Bits 7 and 6 can control input/output data of RTSi# pin. Bits 5 and 4 can control input/output data of CTSi# pin. Bits 3 and 2 can control input/output data of SCKi pin. Bits 1 and 0 can input data from RXDi pin and output data to TXDi pin, so they control break of serial transmitting/receiving.

SC0 Serial Port Register (SC0SPTR)	<P4 address: location H'FFFF C020>
SC1 Serial Port Register (SC1SPTR)	<P4 address: location H'FFFF C120>
SC2 Serial Port Register (SC2SPTR)	<P4 address: location H'FFFF C220>
SC3 Serial Port Register (SC3SPTR)	<P4 address: location H'FFFF C320>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
After Reset:	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0

<After Reset: H'0050>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7	RTSIO	0	R	W	RTS# Port Input/Output Bit Indicates input or output of the serial port RTSi# pin. When the RTSi# pin is actually used as a port outputting the RTSDT bit value, the MCE bit in the SCiFCR register should be cleared to "0". 0: RTSDT bit value not output to RTSi# pin 1: RTSDT bit value output to RTSi# pin
6	RTSDT	1	R	W	RTS# Port Data Bit Indicates the input/output data of the serial port RTSi# pin. Input/output is specified by the RTSIO bit. For output, the RTSDT bit value is output to the RTSi# pin. The RTSi# pin status is read from the RTSDT bit regardless of the RTSIO bit setting. However, RTSi# input/output must be set in the pin function unit. 0: Input/output data is "L" level 1: Input/output data is "H" level
5	CTSIO	0	R	W	CTS# Port Input/Output Bit Indicates input or output of the serial port CTSi# pin. When the CTSi# pin is actually used as a port outputting the CTSDT bit value, the MCE bit in the SCiFCR register should be cleared to "0". 0: CTSDT bit value not output to CTSi# pin 1: CTSDT bit value output to CTSi# pin
4	CTSDT	1	R	W	CTS# Port Data Bit Indicates the input/output data of the serial port CTSi# pin. Input/output is specified by the CTSIO bit. For output, the CTSDT bit value is output to the CTSi# pin. The CTSi# pin status is read from the CTSDT bit regardless of the CTSIO bit setting. However, CTSi# input/output must be set in the pin function unit. 0: Input/output data is "L" level 1: Input/output data is "H" level

Bit	Abbreviation	After Reset	R	W	Description
3	SCKIO	0	R	W	SCK Port Input/Output Bit Indicates input or output of the serial port SCKi pin. When the SCKi pin is actually used as a port outputting the SCKDT bit value, the CKE bit in the SCiSCR register should be cleared to "00". 0: SCKDT bit value not output to SCKi pin 1: SCKDT bit value output to SCKi pin
2	SCKDT	0	R	W	SCK Port Data Bit Indicates the input/output data of the serial port SCKi pin. Input/output is specified by the SCKIO bit. For output, the SCKDT bit value is output to the SCK pin. The SCKi pin status is read from the SCKDT bit regardless of the SCKIO bit setting. However, SCKi input/output must be set in the pin function unit. 0: Input/output data is "L" level 1: Input/output data is "H" level
1	SPB2IO	0	R	W	Serial Port Break Input/Output Bit Indicates output condition of the serial port TXDi pin. When the TXDi pin is actually used as a port outputting the SPB2DT bit value, the TE bit in the SCiSCR register should be cleared to "0". 0: SPB2DT bit value not output to TXDi pin 1: SPB2DT bit value output to TXDi pin
0	SPB2DT	0	R	W	Serial Port Break Data Bit Indicates the input data of the RXDi pin and the output data of the TXDi pin used as serial ports. Input/output is specified by the SPB2IO bit. When the TXDi pin is set to output, the SPB2DT bit value is output to the TXDi pin. The RXDi pin status is read from the SPB2DT bit regardless of the SPB2IO bit setting. However, RXDi input and TXDi output must be set in the pin function unit. 0: Input/output data is "L" level 1: Input/output data is "H" level

23.3.12 SCi Line Status Register (SCiLSR)

The SCiLSR register indicates an abnormal termination due to the occurrence of an overrun error during reception. The value "1" cannot be written to the ORER bit status flag. To clear this bit to "0", a "1" must be read out in advance.

SC0 Line Status Register (SC0LSR)	<P4 address: location H'FFFF C024>
SC1 Line Status Register (SC1LSR)	<P4 address: location H'FFFF C124>
SC2 Line Status Register (SC2LSR)	<P4 address: location H'FFFF C224>
SC3 Line Status Register (SC3LSR)	<P4 address: location H'FFFF C324>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ORER
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	ORER	0	R	*[1]	Overrun Error Flag Indicates the occurrence of an overrun error. 0: Receiving is in progress or has ended normally*[2] 1: Indicates that an overrun error occurred during reception*[3] [Conditions for clearing to "0"] • ORER is cleared to "0" when "0" is written after "1" is read from ORER. [Condition for setting to "1"] • ORER is set to "1" when the next serial reception is completed while the receive FIFO contains 16-byte receive data. Notes: *[2] Clearing the RE bit to "0" in the SCiSCR register does not affect the ORER bit, which retains its previous value. *[3] The SCi receive FIFO data register (SCiFRDR) retains the data before an overrun error has occurred, and the next received data is discarded. When the ORER bit is set to "1", the SCIF cannot continue the next serial reception.

Note: *1 This flag can only be cleared by first reading the value "1" and only then writing a "0".

23.3.13 SCi Serial Extension Mode Register (SCiEMR)

Setting the BGDM bit in this register to "1" allows the baud rate generator in the SCIFi operates in double-speed mode when asynchronous mode is selected (by setting the CA bit in the SCiSMR register to "0") and an internal clock is selected as a clock source and the SCKi pin is set as an input pin (by setting the CKE bit in the SCiSCR register to "00"). Also, the basic clock for the 1-bit period in asynchronous mode by changing the ABCS bit setting.

SC0 Serial Extension Mode Register (SC0EMR)	<P4 address: location H'FFFF C028>
SC1 Serial Extension Mode Register (SC1EMR)	<P4 address: location H'FFFF C128>
SC2 Serial Extension Mode Register (SC2EMR)	<P4 address: location H'FFFF C228>
SC3 Serial Extension Mode Register (SC3EMR)	<P4 address: location H'FFFF C328>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BGDM	—	—	—	—	—	—	ABCS
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7	BGDM	0	R	W	Baud Rate Generator Double-Speed Mode Bit When the BGDM bit is set to "1", the baud rate generator in the SCIFi operates in double-speed mode. This bit is valid only when asynchronous mode is selected by setting the CA bit in the SCiSMR register to "0" and an internal clock is selected as a clock source and the SCKi pin is set as an input pin by setting the CKE bit in the SCiSCR register to "00". In other settings, use normal mode. 0: Normal mode 1: Double-speed mode
6 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	ABCS	0	R	W	Base Clock Select in Asynchronous Mode Bit This bit selects the base clock frequency within a bit period in asynchronous mode. This bit is valid only in asynchronous mode (when the CA bit in the SCiSMR register is "0"). 0: Base clock frequency is 16 times the bit rate 1: Base clock frequency is 8 times the bit rate

23.4 Operation

23.4.1 Overview

For serial communication, the SCIFi has an asynchronous mode in which characters are synchronized individually, and a clock synchronous mode in which communication is synchronized with clock pulses.

The SCIF has a 16-stage FIFO buffer for both transmission and reception respectively, reducing the overhead of the CPU, and enabling continuous high-speed communication. Furthermore, each channel has RTSi# and CTSi# signals to be used as modem control signals.

The transmission format is selected in the SCi serial mode register (SCiSMR), as shown in table 23.9. The SCIFi clock source is selected by the combination of the CKE bit in the SCi serial control register (SCiSCR), as shown in table 23.10.

(1) Asynchronous Mode

- Data length is selectable: 7 or 8 bits
- Parity bit and stop bit length (1 or 2 bits) are selectable respectively. The combination of the preceding selections constitutes the communication format and character length.
- In receiving, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The number of stored data bytes is indicated for both the transmit and receive FIFO registers.
- An internal or external clock can be selected as the SCIFi clock source.
  - When an internal clock is selected, the SCIFi operates using the clock of on-chip baud rate generator.
  - 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 has a fixed 8-bit data length.
- In receiving, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIFi clock source.
  - When an internal clock is selected, the SCIFi 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 SCIF operates on the input external synchronous clock not using the on-chip baud rate generator.

Table 23.9 SCiSMR Register Settings and SCIFi Communication Formats

SCiSMR Register Settings					SCIFi Communication Format		
Bit 7 CA	Bit 6 CHR	Bit 5 PE	Bit 3 STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous	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	x	x	x	Clock synchronous	8 bits	Not set	None

Legend:

x: Don't care

Table 23.10 SCiSMR and SCiSCR Register Settings and SCIFi Clock Source Selection

SCiSMR Register	SCiSCR Register			
Bit 7 CA	Bit 1, 0 CKE	Mode	Clock Source	SCKi Pin Function
0	00	Asynchronous	Internal	SCIFi does not use the SCKi pin
	01			Outputs a clock with a frequency 16 or 8 times the bit rate
	10		External	Inputs a clock with frequency 16 or 8 times the bit rate
	11		Setting prohibited	
1	0x	Clock synchronous	Internal	Outputs the serial clock
	10		External	Inputs the serial clock
	11		Setting prohibited	

Note: • When using the baud rate generator in double-speed mode (BGMD = "1"), select asynchronous mode by setting the CA bit to "0", and select an internal clock as a clock source and the SCKi pin is not used (CKE = "00").

Legend:

x: Don't care

23.4.2 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 SCIFi are independent, so full duplex communication is possible. The transmitter and receiver are 16-byte FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmitting and receiving.

Figure 23.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark ("H" level) state. The SCIF monitors the line and starts serial communication when the line goes to the space ("L" level) state, indicating a start bit. One serial character consists of a start bit ("L" level), data (LSB first), parity bit ("H" level or "L" level), and stop bit ("H" level), in that order.

When receiving in asynchronous mode, the SCIFi synchronizes at the falling edge of the start bit. The SCIFi samples each data bit on the eighth or fourth pulse of a clock with a frequency 16 or 8 times the bit rate. Receive data is latched at the center of each bit.

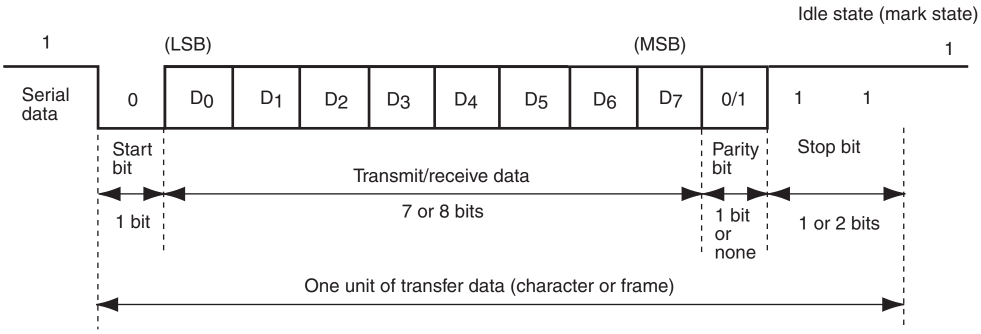


Figure 23.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits)

(1) Transmit/Receive Formats

Table 23.11 lists the eight communication formats that can be selected in asynchronous mode. The format is selected by settings in the serial mode register (SCiSMR).

Table 23.11 Serial Communication Formats (Asynchronous Mode)

SCiSMR Register Settings			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		
0	0	1	START	8-bit data								STOP	STOP	
0	1	0	START	8-bit data								P	STOP	
0	1	1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
1	0	1	START	7-bit data							STOP	STOP		
1	1	0	START	7-bit data							P	STOP		
1	1	1	START	7-bit data							P	STOP	STOP	

Legend:

START:	Start bit
STOP:	Stop bit
P:	Parity bit

(2) Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input to the SCKi pin can be selected as the SCIFi transmit/receive clock by the SCiSMR register CA bit and the SCi serial control register (SCiSCR) CKE bit. For clock source selection, refer to table 23.10.

When an external clock is input at the SCKi pin, it must have a frequency equal to 16 or 8 times the desired bit rate.

When the SCIFi operates on an internal clock, it can output a clock signal on the SCKi pin. The frequency of this output clock is 16 or 8 times the desired bit rate.

(3) Transmitting and Receiving Data

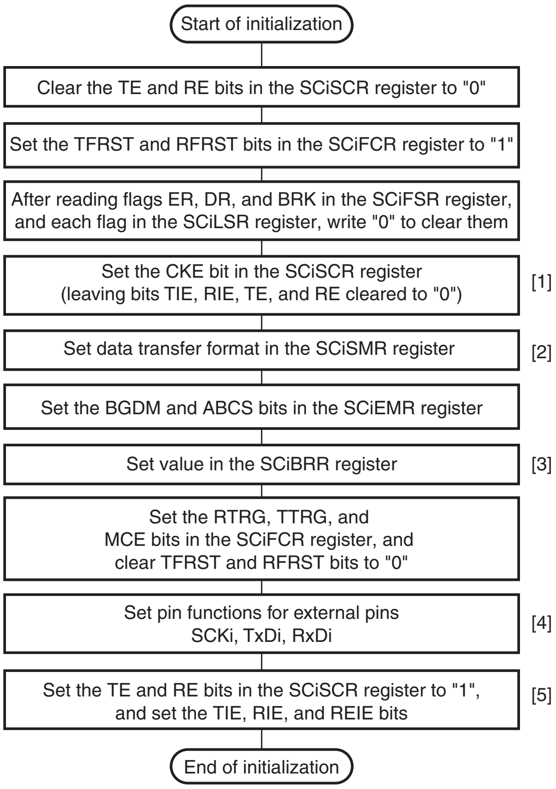
- SCIFi Initialization (Asynchronous Mode)

  Before transmitting or receiving, clear the TE and RE bits to "0" in the SCi serial control register (SCiSCR), then initialize the SCIFi as follows.

  When changing the operation mode or the communication format, always clear the TE and RE bits to "0" before following the procedure given below. Clearing TE bit to "0" initializes the SCi transmit shift register (SCiTSR). Clearing TE and RE bits to "0", however, does not initialize the SCi serial status register (SCiFSR), SCi transmit FIFO data register (SCiFTDR), or receive SCiFIFO data register (SCiFRDR), which retain their previous contents. Clear TE bit to "0" after all transmit data has been transmitted and the TEND flag in the SCiFSR register is set. The TE bit can be cleared to"0" during transmission, but the transmit data goes to the mark state after the bit is cleared to "0". Set the TFRST bit in the SCiFCR register to "1" and reset the SCiFTDR register before TE bit is set again to start transmission.

  When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIFi operation becomes unreliable if the clock is stopped.

  Figure 23.3 shows a sample flowchart for initializing the SCIFi.


Start of initialization
Clear the TE and RE bits in the SCiSCR register to "0"
Set the TFRST and RFRST bits in the SCiFCR register to "1"
After reading flags ER, DR, and BRK in the SCiFSR register, and each flag in the SCiLSR register, write "0" to clear them
Set the CKE bit in the SCiSCR register (leaving bits TIE, RIE, TE, and RE cleared to "0")
[1]
Set data transfer format in the SCiSMR register
[2]
Set the BGDM and ABCS bits in the SCiEMR register
Set value in the SCiBRR register
[3]
Set the RTRG, TTRG, and MCE bits in the SCiFCR register, and clear TFRST and RFRST bits to "0"
Set pin functions for external pins SCKi, TxDi, RxDi
[4]
Set the TE and RE bits in the SCiSCR register to "1", and set the TIE, RIE, and REIE bits
[5]
End of initialization

[1] Set the clock selection in the SCi serial controll register (SCiSCR). Be sure to clear bits TIE, RIE, TE, and RE to "0".

[2] Set the data transfer format in the SCi serial mode register (SCiSMR).

[3] Write a value that corresponds to the bit rate to the SCi bit rate register (SCiBRR). Note that this is not required if an external clock is used.

[4] Set the pin functions for the external pins. Set as RXDi input at receiving and TXDi at transmission.
Note, however, that no settings for the SCKi pin are required if the CKE field is "00" in asynchronous mode.

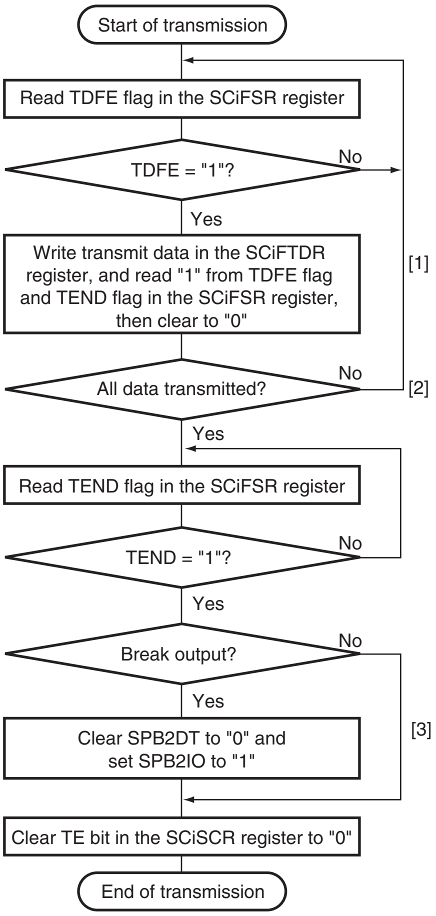
[5] Set the TE bit or RE bit in the SCi serial controll register (SCiSCR) to "1". Also set the RIE, REIE, and TIE bits. Setting the TE and RE bits enables the TXDi and RXDi pins to be used.
When transmitting, the SCIF will go to the mark state; when receiving, it will go to the idle state, waiting for a start bit.
Clock output from the SCKi pin will start at this point if the internal clock is set to be output.

Figure 23.3 Sample Flowchart for SCIF Initialization

- Transmitting Serial Data (Asynchronous Mode)

  Figure 23.4 shows a sample flowchart for serial transmission.

  Use the following procedure for serial data transmission after enabling the SCIFi for transmission.


Start of transmission
Read TDFE flag in the SCiFSR register
TDFE = "1"?
No
Yes
Write transmit data in the SCiFTDR register, and read "1" from TDFE flag and TEND flag in the SCiFSR register, then clear to "0"
[1]
All data transmitted?
No
[2]
Yes
Read TEND flag in the SCiFSR register
TEND = "1"?
No
Yes
Break output?
No
Yes
Clear SPB2DT to "0" and set SPB2IO to "1"
[3]
Clear TE bit in the SCiSCR register to "0"
End of transmission

[1] SCIF status check and transmit data write:
Read SCiFSR and check that the TDFE flag is set to "1", then write transmit data to SCFTDR, and read "1" from the TDFE and TEND flags, then clear to "0".
The quantity of transmit data that can be written is 16 - (transmit trigger set number).

[2] Serial transmission continuation procedure:
To continue serial transmission, read "1" from the TDFE flag to confirm that writing is possible, then write data to the SCiFTDR register, and then clear the TDFE flag to "0".

[3] Break output during serial transmission:
To output a break in serial transmission, clear the SPB2DT bit to "0" and set the SPB2IO bit to "1" in the SCi serial port register (SCiSPTR), then clear the TE bit in the SCi serial controll register (SCiSCR) to "0".

In [1] and [2], it is possible to ascertain the number of data bytes that can be written from the number of transmit data bytes in the SCiFTDR register indicated by the upper 8 bits of the SCiFIFO data register (SCiFDR).

Figure 23.4 Sample Flowchart for Transmitting Serial Data

In serial transmission, the SCIFi operates as described below.

1. When data is written into the SCi transmit FIFO data register (SCiFTDR), the SCIFi transfers the data from the SCiFTDR register to the SCi transmit shift register (SCiTSR). Confirm that the TDFE flag in the SCi serial status register (SCiFSR) is set to "1" before writing transmit data to the SCiFTDR register. The number of data bytes that can be written is (16 – transmit trigger setting).
2. When data is transferred from the SCiFTDR register to the SCiTSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the SCiFTDR register. When the number of transmit data bytes in the SCiFTDR register falls below the transmit trigger number set in the SCiFIFO control register (SCiFCR), the TDFE flag is set. If the TIE bit in the SCi serial control register (SCiSCR) is set to "1" at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.

   The serial transmit data is sent from the TXDi pin in the following order.

   A. Start bit: One-bit "0" is output.
   B. Transmit data: 8-bit or 7-bit data is output in LSB-first order.
   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 sent.
3. The SCIFi checks the SCiFTDR register transmit data at the timing for sending the stop bit. If data is present, the data is transferred from the SCiFTDR register to the SCiTSR register, the stop bit is sent, and then serial transmission of the next frame is started.

Figure 23.5 shows an example of the operation for transmission.

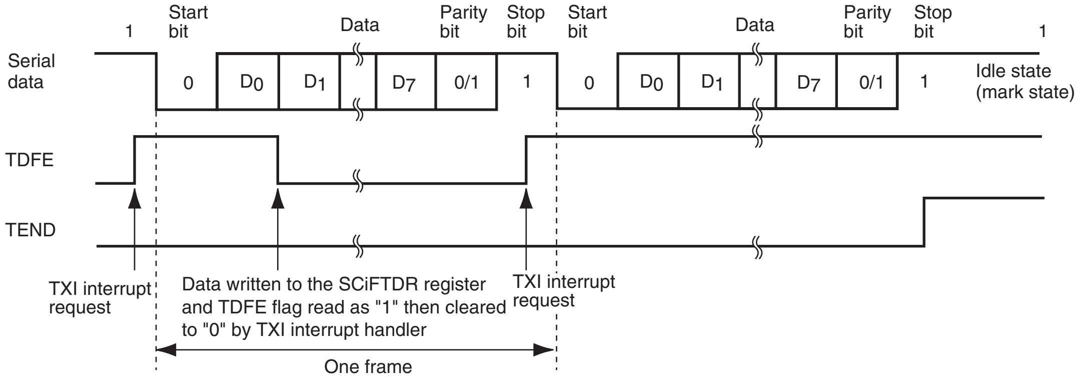


Figure 23.5 Example of Transmit Operation (8-Bit Data, Parity, 1 Stop Bit)

4. When modem control is enabled, transmission can be stopped and restarted in accordance with the CTSi# input value. When CTSi# is set to "H" level, if transmission is in progress, the line goes to the mark state after transmission of one frame. When CTSi# is set to "L" level, the next transmit data is output starting from the start bit.

Figure 23.6 shows an example of the operation when modem control is used.

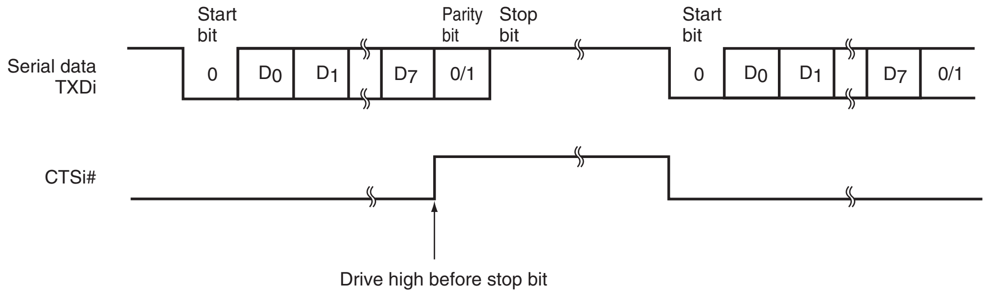
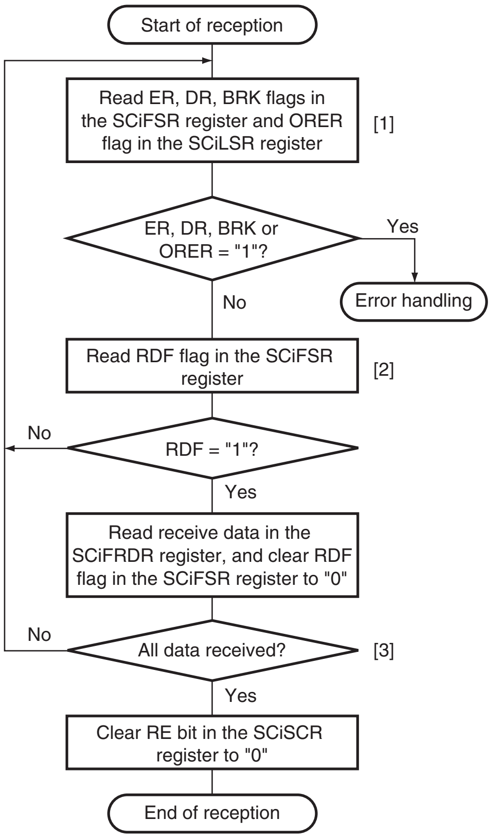


Figure 23.6 Example of Operation Using Modem Control (CTSi#)

- Receiving Serial Data (Asynchronous Mode)

  Figures 23.7 and 23.8 show sample flowcharts for serial reception.

  Use the following procedure for serial data reception after enabling the SCIFi for reception.


Start of reception
Read ER, DR, BRK flags in the SCiFSR register and ORER flag in the SCiLSR register
[1]
ER, DR, BRK or ORER = "1"?
Yes
Error handling
No
Read RDF flag in the SCiFSR register
[2]
No
RDF = "1"?
Yes
Read receive data in the SCiFRDR register, and clear RDF flag in the SCiFSR register to "0"
No
All data received?
[3]
Yes
Clear RE bit in the SCiSCR register to "0"
End of reception

[1] Receive error handling and break detection:

When a receive error occurs, read the DR, ER, and BRK flags in the SCi serial status register (SCiFSR), and the ORER flag in the SCi line status register (SCiLSR), to identify any error, perform the appropriate error handling, then clear the DR, ER, BRK, and ORER flags to "0". In the case of a framing error, a break can also be detected by reading the value of the RXDi pin.

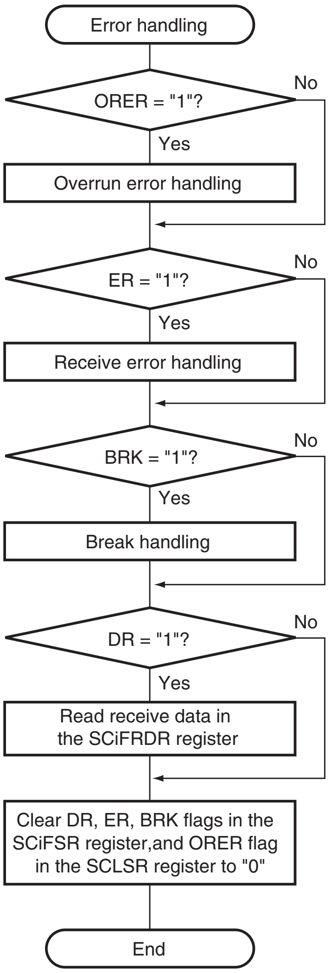
[2] SCIFi status check and receive data read:

Read the SCi serial status register (SCiFSR) and check that RDF flag = "1", then read the receive data in the SCi receive FIFO data register (SCiFRDR), read "1" from the RDF flag, and then clear the RDF flag to "0". The transition of the RDF flag from "0" to "1" can also be identified by a receive FIFO data full interrupt (RXI).

[3] Serial reception continuation procedure:

To continue serial reception, read at least the receive trigger set number of receive data bytes from the SCiFRDR register, read "1" from the RDF flag, then clear the RDF flag to "0". The number of receive data bytes in the SCiFRDR register can be ascertained by reading from the SCiFDR register.

Figure 23.7 Sample Flowchart for Receiving Serial Data (1)



- Whether a framing error or parity error has occurred in the receive data that is to be read from the SCi receive FIFO data register (SCiFRDR) can be ascertained from the FER and PER bits in the SCi serial status register (SCiFSR).
- When a break signal is received, receive data is not transferred to the SCiFRDR register while the BRK flag is set. However, note that the last data in the SCiFRDR register is "H'00", and the break data in which a framing error occurred is stored.

Figure 23.8 Sample Flowchart for Receiving Serial Data (2)

In serial reception, the SCIFi operates as described below.

1. The SCIFi monitors the transmission line, and if a "0" start bit is detected, performs internal synchronization and starts reception.
2. The received data is stored in the SCiRSR register in LSB-to-MSB order.
3. The parity bit and stop bit are received.

   After receiving these bits, the SCIFi carries out the following checks.

   A. Stop bit check: The SCIF checks whether the stop bit is "1". If there are two stop bits, only the first is checked.

   B. The SCIF checks whether receive data can be transferred from the SCi receive shift register (SCiRSR) to the SCiFRDR register.

   C. Overrun check: The SCIF checks that the ORER flag is "0", indicating that the overrun error has not occurred.

   D. Break check: The SCIF checks that 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 SCiFRDR register.

Note: • When a parity error or a framing error occurs, reception is not suspended.

4. If the RIE bit in the SCiSCR register is set to "1" when the RDF or DR flag changes to "1", a receive-FIFO-data-full interrupt (RXI) request is generated. If the RIE bit or the REIE bit in the SCiSCR register is set to "1" when the ER flag changes to "1", a receive-error interrupt (ERI) request is generated. If the RIE bit or the REIE bit in the SCiSCR register is set to "1" when the BRK or ORER flag changes to "1", a break reception interrupt (BRI) request is generated.

Figure 23.9 shows an example of the operation for reception.

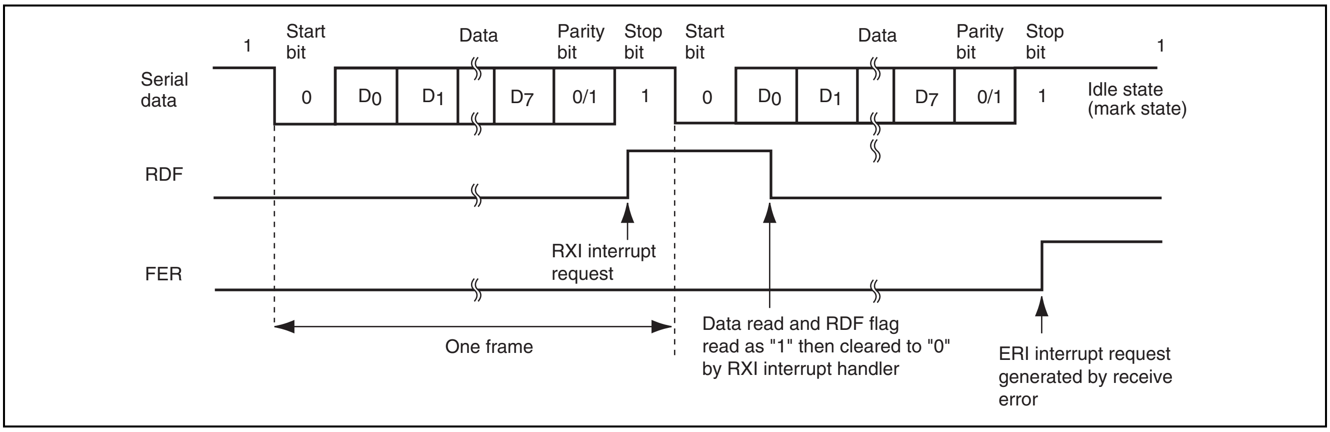


Figure 23.9 Example of SCIFi Receive Operation (8-Bit Data, Parity, 1 Stop Bit)

5. When modem control is enabled, the RTSi# signal is output when the SCiFRDR register is empty. When RTSi# is "L" level, reception is possible. When RTSi# is "H" level, this indicates that the SCiFRDR register exceeds the number set for the RTSi# output active trigger.

Figure 23.10 shows an example of the operation when modem control is used.

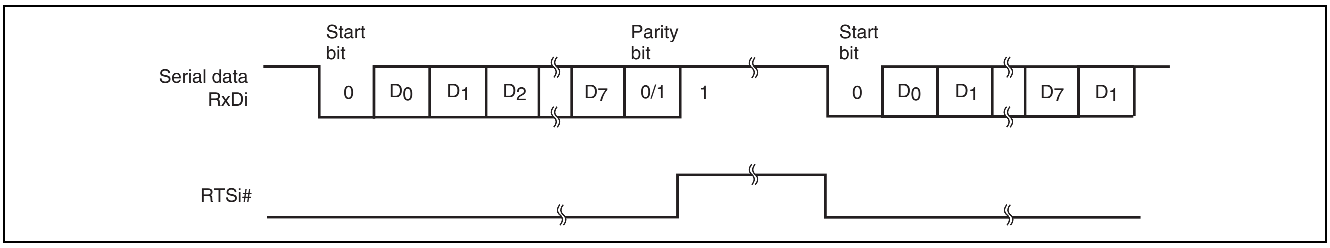


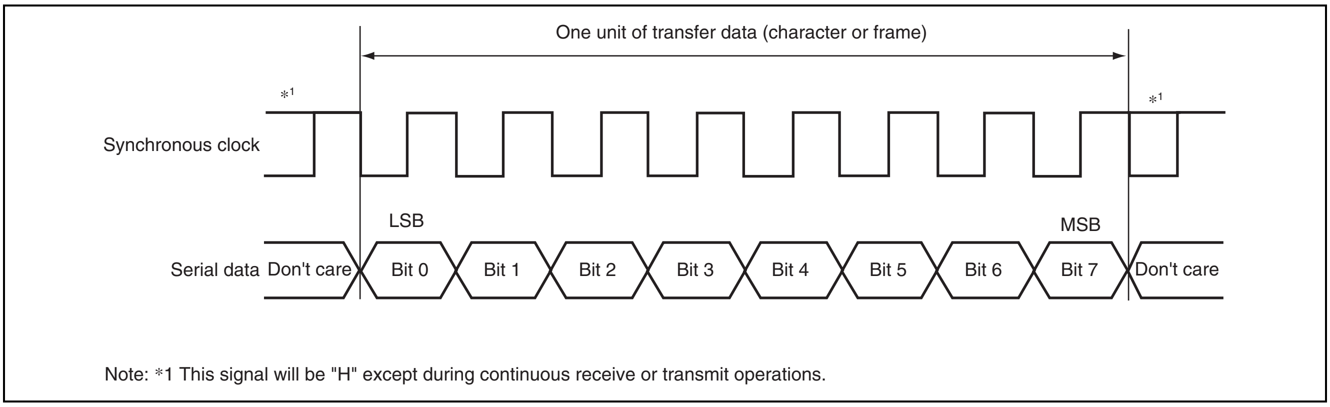
Figure 23.10 Example of Operation Using Modem Control (RTSi#)

23.4.3 Operation in Clock Synchronous Mode

In clock synchronous mode, the SCIFi transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

The SCIFi transmitter and receiver are independent, so full-duplex communication is possible while sharing the same clock. The transmitter and receiver are also 16-byte FIFO buffered, so continuous transmitting or receiving is possible by reading or writing data while transmitting or receiving is in progress.

Figure 23.11 shows the general format in clock synchronous serial communication.



Note: *1 This signal will be "H" except during continuous receive or transmit operations.

Figure 23.11 Data Format in Clock Synchronous Communication

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial 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.

In clock synchronous mode, the SCIFi receives data by synchronizing with the rising edge of the serial clock.

(1) Transmit/Receive Formats

The data length is fixed at eight bits. No parity bit can be added.

(2) Clock

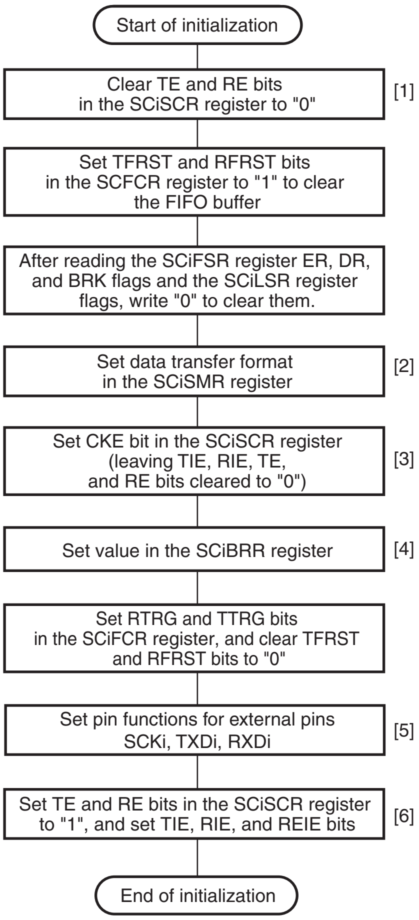
An internal clock generated by the on-chip baud rate generator by the setting of the CA bit in the SCiSMR register and CKE bit in the SCiSCR register, or an external clock input from the SCKi pin can be selected as the SCIF transmit/receive clock.

When the SCIF operates on an internal clock, it outputs the clock signal at the SCKi pin. Eight clock pulses are output per transmitted or received character. When the SCIF is not transmitting or receiving, the clock signal remains in the "H" state. When only receiving, the clock signal outputs while the RE bit in the SCiSCR register is "1" and the number of data in receive FIFO is more than the receive FIFO data trigger number.

(3) Transmitting and Receiving Data

- SCIFi Initialization (Clock Synchronous Mode)
  Before transmitting, receiving, or changing the mode or communication format, the software must clear the TE and RE bits to "0" in the SCi serial control register (SCiSCR), then initialize the SCIF. Clearing TE to "0" initializes the SCi transmit shift register (SCiTSR). Clearing RE to "0", however, does not initialize the RDF, PER, FER, and ORER flags and SCi receive data register (SCiRDR), which retain their previous contents.

Figure 23.12 shows a sample flowchart for initializing the SCIFi.


Start of initialization
Clear TE and RE bits in the SCiSCR register to "0"
[1]
Set TFRST and RFRST bits in the SCFCR register to "1" to clear the FIFO buffer
After reading the SCiFSR register ER, DR, and BRK flags and the SCiLSR register flags, write "0" to clear them.
Set data transfer format in the SCiSMR register
[2]
Set CKE bit in the SCiSCR register (leaving TIE, RIE, TE, and RE bits cleared to "0")
[3]
Set value in the SCiBRR register
[4]
Set RTRG and TTRG bits in the SCiFCR register, and clear TFRST and RFRST bits to "0"
Set pin functions for external pins SCKi, TXDi, RXDi
[5]
Set TE and RE bits in the SCiSCR register to "1", and set TIE, RIE, and REIE bits
[6]
End of initialization

[1] Leave the TE and RE bits cleared to 0 until the initialization almost ends.

[2] Set the data transfer format in the SCiSMR register.

[3] Set the CKE bit.

[4] Write a value corresponding to the bit rate into the SCi bit rate register (SCiBRR). This is not necessary if an external clock is used.

[5] Set the pin functions for the external pins. For reception, set the RXDi pin to input mode and for transmission set the TXDi pin to output mode.

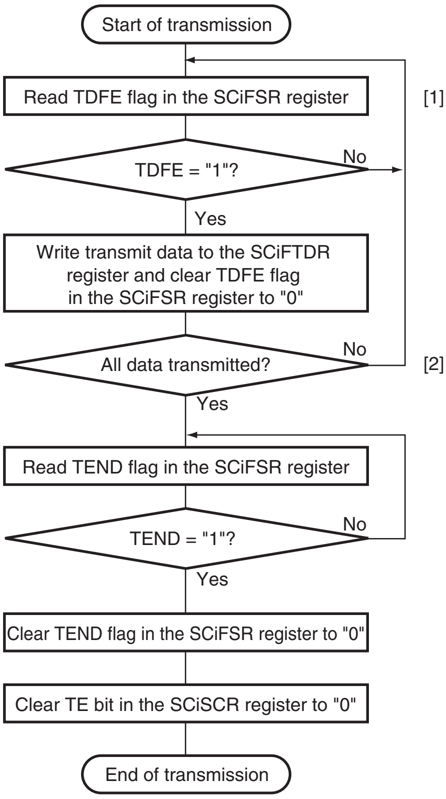
[6] Set the TE or RE bit in the SCiSCR register to "1". Also set the TIE, RIE, and REIE bits to enable the TXDi, RXDi, and SCKi pins to be used. When transmitting, the TXDi pin will go to the mark state. When receiving in clocked synchronous mode with the synchronization clock output (clock master) selected, a clock starts to be output from the SCKi pin at this point.

Figure 23.12 Sample Flowchart for SCIFi Initialization

- Transmitting Serial Data (Clock Synchronous Mode)

  Figure 23.13 shows a sample flowchart for transmitting serial data.

  Use the following procedure for serial data transmission after enabling the SCIFi for transmission.


Start of transmission
Read TDFE flag in the SCiFSR register
[1]
TDFE = "1"?
No
Yes
Write transmit data to the SCiFTDR register and clear TDFE flag in the SCiFSR register to "0"
All data transmitted?
No
[2]
Yes
Read TEND flag in the SCiFSR register
TEND = "1"?
No
Yes
Clear TEND flag in the SCiFSR register to "0"
Clear TE bit in the SCiSCR register to "0"
End of transmission

[1] SCIFi status check and transmit data write:

Read the SCiFSR register and check that the TDFE flag is set to "1", then write transmit data to the SCi transmit FIFO data register (SCiFTDR), and clear the TDFE flag to "0".

[2] Serial transmission continuation procedure:

To continue serial transmission, read "1" from the TDFE flag to confirm that writing is possible, and then write data to the SCiFTDR register, and then clear the TDFE flag to "0".

Figure 23.13 Sample Flowchart for Transmitting Serial Data

In serial transmission, the SCIFi operates as described below.

1. When data is written into the SCi transmit FIFO data register (SCiFTDR), the SCIFi transfers the data from the SCiFTDR register to the SCi transmit shift register (SCiTSR). Confirm that the TDFE flag in the SCi serial status register (SCiFSR) is set to "1" before writing transmit data to the SCiFTDR register. The number of data bytes that can be written is (16 – transmit trigger setting).
2. When data is transferred from the SCiFTDR register to the SCiTSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the SCiFTDR register. When the number of transmit data bytes in the SCiFTDR register falls below the transmit trigger number set in the SCiFIFO control register (SCiFCR), the TDFE flag is set. If the TIE bit in the SCi serial control register (SCiSCR) is set to "1" at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.

   If clock output mode is selected, the SCIFi outputs eight synchronous clock pulses. If an external clock source is selected, the SCIFi outputs data in synchronization with the input clock. Data is output from the TXDi pin in order from the LSB (bit 0) to the MSB (bit 7).
3. The SCIF checks the SCiFTDR register transmit data at the timing for sending the MSB (bit 7). If data is present, the data is transferred from the SCiFTDR register to the SCiTSR register, and then serial transmission of the next frame is started. If there is no data, the TXDi pin holds the state after the TEND flag in the SCiFSR register is set to "1" and the MSB (bit 7) is sent.
4. After the end of serial transmission, the SCKi pin is held in the "H" state.

Figure 23.14 shows an example of SCIFi transmit operation.

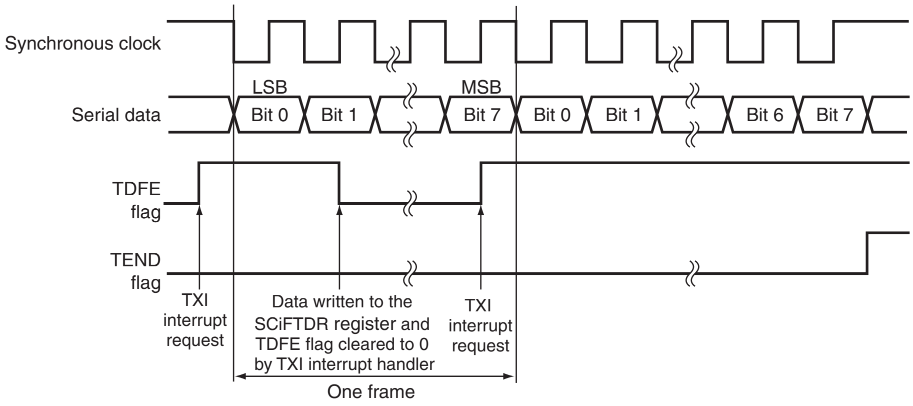
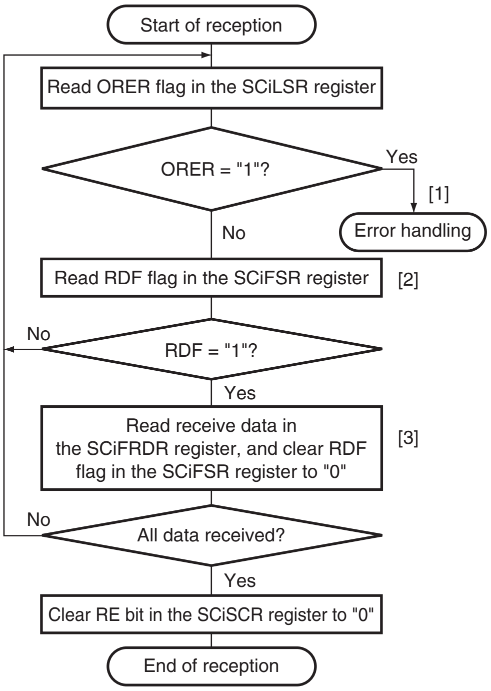


Figure 23.14 Example of SCIFi Transmit Operation

- Receiving Serial Data (Clock Synchronous Mode)

  Figures 23.15 and 23.16 show sample flowcharts for receiving serial data. Use the following procedure for serial data reception after enabling the SCIFi for reception.

  When switching from asynchronous mode to clock synchronous mode without SCIFi initialization, make sure that ORER, PER, and FER flags are cleared to "0".


Start of reception
Read ORER flag in the SCiLSR register
ORER = "1"?
Yes
[1]
Error handling
No
Read RDF flag in the SCiFSR register
[2]
No
RDF = "1"?
Yes
Read receive data in the SCiFRDR register, and clear RDF flag in the SCiFSR register to "0"
[3]
No
All data received?
Yes
Clear RE bit in the SCiSCR register to "0"
End of reception

[1] Receive error handling:
When a receive error occurs, read the ORER flag in the SCiLSR register to identify any error, perform the appropriate error handling, then clear the ORER flag to "0". Reception cannot be resumed while the ORER flag is set to "1".

[2] SCIFi status check and receive data read:
Read the SCiFSR register and check that RDF = "1", then read the receive data in the SCiFRDR register, and clear the RDF flag to "0". The transition of the RDF flag from "0" to "1" can also be identified by a receive FIFO data full interrupt (RXI).

[3] Serial reception continuation procedure:
To continue serial reception, read at least the receive trigger set number of receive data bytes from the SCiFRDR register, read "1" from the RDF flag, then clear the RDF flag to "0". The number of receive data bytes in the SCiFRDR register can be ascertained by reading the SCiFDR register. However, the RDF bit is cleared to "0" automatically when an RXI interrupt activates the DMAC to read the data in the SCiFRDR register.

Figure 23.15 Sample Flowchart for Receiving Serial Data (1)

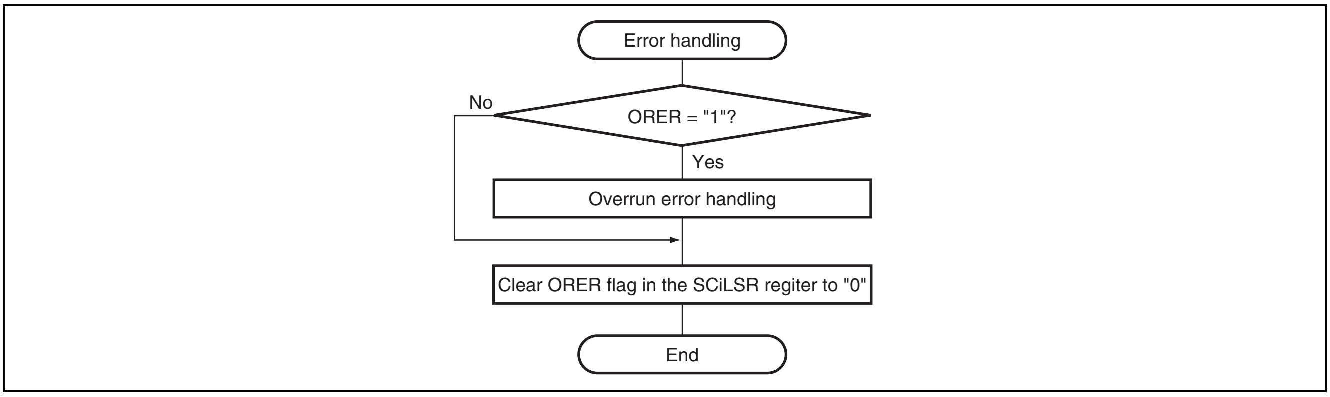


Figure 23.16 Sample Flowchart for Receiving Serial Data (2)

In serial reception, the SCIFi operates as described below.

1. The SCIFi synchronizes with serial clock input or output and starts the reception.
2. Receive data is shifted into the SCi receive shift register (SCiRSR) in order from the LSB to the MSB. After receiving the data, the SCIFi checks the receive data can be loaded from the SCiRSR register into the SCiFRDR register or not. If this check is passed, the RDF flag is set to "1" and the SCIF stores the received data in the SCiFRDR register. If the check is not passed (overrun error is detected), further reception is prevented.
3. After setting RDF to "1", if the receive FIFO data full interrupt enable bit (RIE) is set to "1" in the SCi serial control register (SCiSCR), the SCIF requests a receive-data-full interrupt (RXI). If the ORER bit is set to "1" and the receive-data-full interrupt enable bit (RIE) or the receive error interrupt enable bit (REIE) in the SCiSCR register is also set to "1", the SCIF requests a break interrupt (BRI).

Figure 23.17 shows an example of SCIFi receive operation.

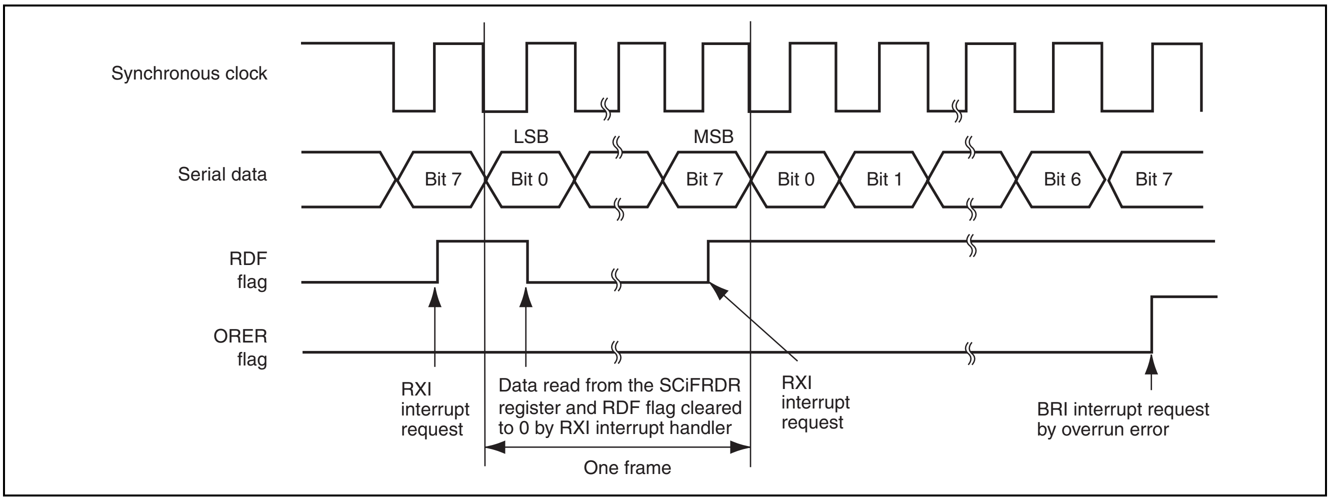
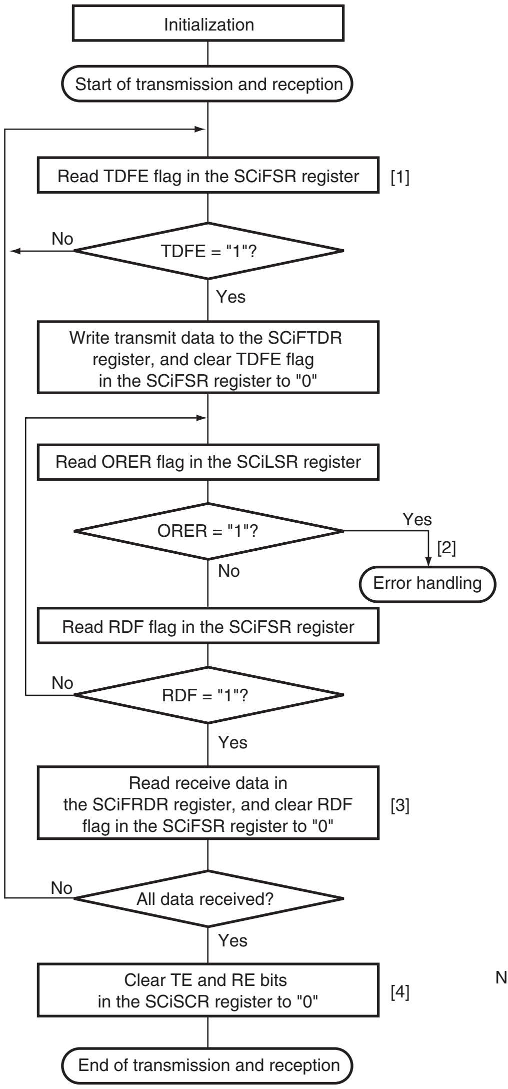


Figure 23.17 Example of SCIFi Receive Operation

- Transmitting and Receiving Serial Data Simultaneously (Clock Synchronous Mode)

  Figure 23.18 shows a sample flowchart for transmitting and receiving serial data simultaneously.

  Use the following procedure for the simultaneous transmission/reception of serial data, after enabling the SCIFi for transmission/reception.


Initialization
Start of transmission and reception
Read TDFE flag in the SCiFSR register
[1]
No
TDFE = "1"?
Yes
Write transmit data to the SCiFTDR register, and clear TDFE flag in the SCiFSR register to "0"
Read ORER flag in the SCiLSR register
ORER = "1"?
Yes
[2]
No
Error handling
Read RDF flag in the SCiFSR register
No
RDF = "1"?
Yes
Read receive data in the SCiFRDR register, and clear RDF flag in the SCiFSR register to "0"
[3]
No
All data received?
Yes
Clear TE and RE bits in the SCiSCR register to "0"
[4]
End of transmission and reception

[1] SCIFi status check and transmit data write:

Read the SCiFSR register and check that the TDFE flag is set to "1", then write transmit data to the SCiFTDR register, and clear the TDFE flag to "0". The transition of the TDFE flag from "0" to "1" can also be identified by a transmit FIFO data empty interrupt (TXI).

[2] Receive error handling:

When a receive error occurs, read the ORER flag in the SCiLSR register to identify any error, perform the appropriate error handling, then clear the ORER flag to "0". Reception cannot be resumed while the ORER flag is set to "1".

[3] SCIFi status check and receive data read:

Read the SCiFSR register and check that RDF flag = "1", then read the receive data in the SCiFRDR register, and clear the RDF flag to "0". The transition of the RDF flag from "0" to "1" can also be identified by a receive FIFO data full interrupt (RXI).

[4] Serial transmission and reception continuation procedure:

To continue serial transmission and reception, read "1" from the RDF flag and the receive data in the SCiFRDR register, and clear the RDF flag to "0" before receiving the MSB in the current frame. Similarly, read "1" from the TDFE flag to confirm that writing is possible before transmitting the MSB in the current frame. Then write data to the SCiFTDR register and clear the TDFE flag to "0".

Note: When switching from a transmit operation or receive operation to simultaneous transmission and reception operations, clear the TE and RE bits to "0", and then set them simultaneously to "1".

Figure 23.18 Sample Flowchart for Transmitting/Receiving Serial Data

23.5 SCIFi Interrupt Sources and DMAC

The SCIFi has four interrupt sources: transmit-FIFO-data-empty (TXI), receive-error (ERI), receive FIFO data full (RXI), and break (BRI).

Table 23.12 shows the interrupt sources and their order of priority. The interrupt sources are enabled or disabled by means of the TIE, RIE, and REIE bits in the SCiSCR register. A separate interrupt request is sent to the interrupt controller for each of these interrupt sources.

When TXI is enabled with the TIE bit, a TXI interrupt request and a transmit FIFO data empty DMA transfer request will be issued if the SCi serial status register (SCiFSR) TDFE flag is set to "1". When TXI is disabled with the TIE bit, only the transmit FIFO data empty DMA transfer request will be issued if the TDFE flag is set to "1". The DMAC can be activated and a data transfer performed by the transmit FIFO data empty DMA transfer request.

When RXI is enabled with the RIE bit, an RXI interrupt and a receive FIFO data full DMA transfer request will be issued if the SCiFSR register RDF flag or DR flag is set to "1". When RXI is disabled with the RIE bit, only the receive FIFO data full DMA transfer request will be issued if the SCiFSR register RDF flag or DR flag is set to "1". The DMAC can be activated and a data transfer performed by the receive FIFO data full DMA transfer request. Note that the RXI interrupt or receive FIFO data full DMA transfer request specified by setting the DR flag to "1" will only be issued in asynchronous mode.

A BRI interrupt request is issued if either the SCiFSR register BRK flag or the SCiLSR register ORER flag is set to "1".

If the DMAC is used for transmission or reception, first set up the DMAC, set it to the enabled state, and then set up SCIFi. Also, set up the module so that RXI and TXI interrupt requests are not sent to the interrupt controller. If settings that issue interrupt requests are used, interrupt requests to the interrupt controller will be cleared by the DMAC regardless of the interrupt handler.

Just an ERI interrupt request can be issued without issuing an RXI interrupt request by setting the SCiSCR register RIE bit to "0" and the REIE bit to "1".

Table 23.12 SCIF Interrupt Sources

Interrupt Source	Description	DMAC Activation	Priority on Reset Release
ERI	Interrupt due to a receive error (ER)	No	High
RXI	Interrupt due to receive FIFO data full (RDF) or data ready (DR)	Yes	↕
BRI	Interrupt due to break (BRK) or overrun (ORER)	No	
TXI	Interrupt due to transmit FIFO data empty (TDFE)	Yes	Low

23.6 Usage Notes

Note the following when using the SCIFi.

23.6.1 SCiFTDR Register Writing and TDFE Flag

The TDFE flag in the SCi serial status register (SCiFSR) is set when the number of transmit data bytes written in the SCi transmit FIFO data register (SCiFTDR) has fallen below the transmit trigger number set by bits TTRG bit in the SCiFIFO control register (SCiFCR). After the TDFE flag is set, transmit data up to the number of empty bytes in the SCiFTDR register can be written, allowing efficient continuous transmission.

However, if the number of data bytes written in the SCiFTDR register is equal to or less than the transmit trigger number, the TDFE flag will be set to "1" again after being read as "1" and cleared to "0". TDFE flag clearing should therefore be carried out when the SCiFTDR register contains more than the transmit trigger number of transmit data bytes after reading "1".

The number of transmit data bytes in the SCiFTDR register can be found from the upper 8 bits of the SCiFIFO data count register (SCiFDR).

23.6.2 SCiFRDR Reading and RDF Flag

The RDF flag in the SCi serial status register (SCiFSR) is set when the number of receive data bytes in the SCi receive FIFO data register (SCiFRDR) has become equal to or greater than the receive trigger number set by bits RTRG bit in the SCi FIFO control register (SCiFCR). After RDF flag is set, receive data equivalent to the trigger number can be read from the SCiFRDR register, allowing efficient continuous reception.

However, if the number of data bytes in the SCiFRDR register exceeds the trigger number, the RDF flag will be set to "1" again if it is cleared to "0". The RDF flag should therefore be cleared to "0" after being read as "1" after reading the number of the SCi received data in the receive FIFO data register (SCiFRDR) which is less than the trigger number.

The number of receive data bytes in the SCiFRDR register can be found from the lower 8 bits of the SCiFIFO data count register (SCiFDR).

23.6.3 Break Detection and Processing

Break signals can be detected by reading the RXDi pin directly when a framing error (FER) is detected. In the break state the input from the RXDi pin consists of all "0s", so the FER flag is set and the parity error flag (PER) may also be set.

Note that, although transfer of receive data to the SCiFRDR register is halted in the break state, the SCIFi receiver continues to operate.

23.6.4 Sending a Break Signal

The I/O condition and level of the TXDi pin are determined by the SPB2IO and SPB2DT bits in the SCi serial port register (SCiSPTR). This feature can be used to send a break signal.

Until TE bit is set to "1" (enabling transmission) after initializing, the TXDi pin does not work. During the period, mark status is performed by the SPB2DT bit. Therefore, the SPB2IO and SPB2DT bits should be set to "1" ("H" level output).

To send a break signal during serial transmission, clear the SPB2DT bit to "0" (designating "L" level), then clear the TE bit to "0" (halting transmission). When the TE bit is cleared to "0", the transmitter is initialized regardless of the current transmission state, and "0" is output from the TxDi pin.

23.6.5 Receive Data Sampling Timing in Asynchronous Mode

The SCIFi operates on a base clock with a frequency 16 or 8 times the bit rate. In reception, the SCIFi synchronizes internally with the fall of the start bit, which it samples on the base clock. Receive data is latched at the rising edge of the eighth or fourth base clock pulse. Figure 23.19 shows the receive data sampling timing in asynchronous mode.

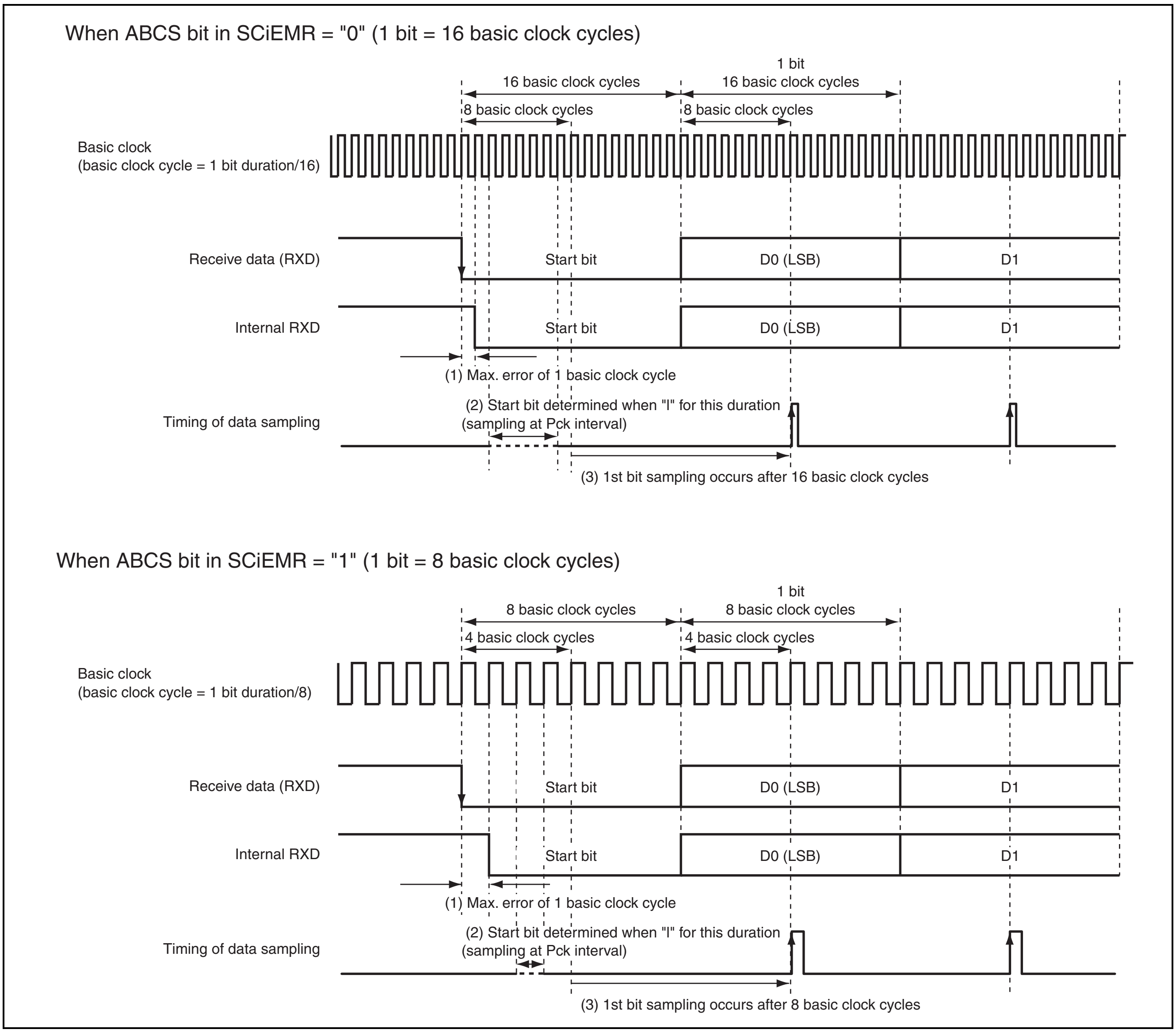

When ABCS bit in SCiEMR = "0" (1 bit = 16 basic clock cycles)
1 bit
16 basic clock cycles
16 basic clock cycles
8 basic clock cycles
8 basic clock cycles
Basic clock
(basic clock cycle = 1 bit duration/16)
Receive data (RXD)
Start bit
D0 (LSB)
D1
Internal RXD
Start bit
D0 (LSB)
D1
(1) Max. error of 1 basic clock cycle
(2) Start bit determined when "l" for this duration
(sampling at Pck interval)
Timing of data sampling
(3) 1st bit sampling occurs after 16 basic clock cycles
When ABCS bit in SCiEMR = "1" (1 bit = 8 basic clock cycles)
1 bit
8 basic clock cycles
8 basic clock cycles
4 basic clock cycles
4 basic clock cycles
Basic clock
(basic clock cycle = 1 bit duration/8)
Receive data (RXD)
Start bit
D0 (LSB)
D1
Internal RXD
Start bit
D0 (LSB)
D1
(1) Max. error of 1 basic clock cycle
(2) Start bit determined when "l" for this duration
(sampling at Pck interval)
Timing of data sampling
(3) 1st bit sampling occurs after 8 basic clock cycles

Figure 23.19 Receive Data Sampling Timing in Asynchronous Mode

Section 24 Renesas Serial Peripheral Interface (RSPI)

This MCU includes three-channel (RSPI0 to RSPI2) Renesas Serial Peripheral Interfaces (RSPI). The RSPI has three channels which are independent of each other and is capable of full-duplex high-speed serial communications with multiple processors and peripheral devices. Note that in pin and signal names appearing in this section, the i in the notation RSPIi represents values from 0 to 2. (For pin specifications, see table 24.1.)

24.1 Overview

- RSPI transfer function

  SPI (four-wire) and clock-synchronous (three-wire) serial communication are supported using the MOSI (master out slave in), MISO (master in slave out), SSL (slave select), and RSPCK (RSPI clock) signals.

  Serial communication in master or slave mode is supported.

  Mode fault error detection is supported.

  Overrun error detection is supported.

  The serial transfer clock polarity is selectable.

  The serial transfer clock phase is selectable.
- Data format

  Switchable between MSB first and LSB first.

  Transfer bit length selectable among 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 24, and 32 bits.

  128-bit transmit and receive buffers.

  Up to four frames (maximum frame size 32 bits) can be transferred in a single transmit or receive operation
- Buffer configuration

  Double-buffer transmit and receive buffer configuration.
- SSL control function

  Four SSL signals (SSL00, SSL01, SSL10, and SSL20).

  SSL00, SSL01, SSL10, and SSL20 are used for signal output in single-master mode.

  In multi-master mode, SSLi0 is used for signal input, and SSL01 is used for signal output or set as Hi-Z.

  In slave mode, SSLi0 is used for signal input, and SSL01 is set as Hi-Z.

  The delay time from SSL output assertion to RSPCK operation (RSPCK delay) is selectable.

  Setting range: 1 to 8 RSPCK cycles
  Setting unit: 1 RSPCK cycle

  The delay time from halt of RSPCK operation to SSL output negation (SSL negation delay) is selectable.

  Setting range: 1 to 8 RSPCK cycles
  Setting unit: 1 RSPCK cycle

  The wait until SSL output assertion for the next access (next-access delay) is selectable.

  Setting range: 1 to 8 RSPCK cycles
  Setting unit: 1 RSPCK cycle

  The SSL polarity is selectable.

- Control during master mode transfer

  Transfers composed of up to four commands can be executed sequentially as loops.

  For each command, the following items can be set:

  SSL signal value, bit rate, RSPCK polarity and phase, transfer data length, LSB or MSB first, burst, RSPCK, SSL negation delay, next-access delay

  Initiation of transfers when the CPU or the DMAC writes to the transmit buffer is supported.

  Initiation of transfers when the CPU clears the SPTEF bit is supported.

  The MOSI value at SSL negation is selectable.
- Interrupt sources

  The following maskable interrupt sources are supported:

  RSPI receive interrupt (receive buffer full)

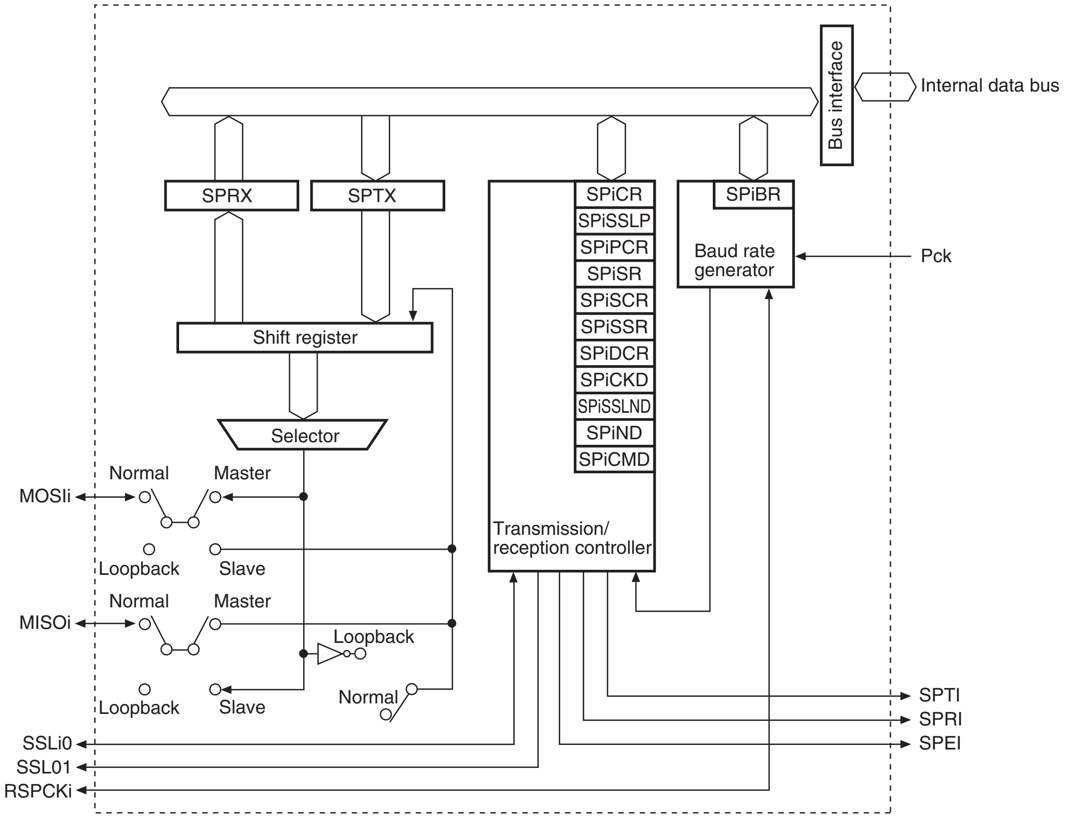
  RSPI transmit interrupt (transmit buffer empty)

  RSPI error interrupt (mode fault, overrun)
- Other

  Loopback mode.

  CMOS/open drain output switching is supported.

  RSPI disable (initialization) function.



Legend:

SPCRi: RSPIi control register
SPiSSLP: RSPIi slave select polarity register
SPiPCR: RSPIi pin control register
SPiSR: RSPIi status register
SPiSCR: RSPIi sequence control register
SPiSSR: RSPIi sequence status register
SPiDCR: RSPIi data controll register
SPiCKD: RSPIi clock delay register
SPiSSLND: RSPIi slave select negate delay register

SPiND: RSPIi next-access delay register
SPiCMD: RSPIi command registers (four, numbered 0 to 3)
SPiBR: RSPIi bit rate register
SPTX: RSPIi transmit buffer
SPRX: RSPIi receive buffer
SPTI: RSPIi transmit interrupt
SPRI: RSPIi receive interrupt
SPEI: RSPIi error interrupt

Figure 24.1 Block Diagram of RSPIi

24.2 Input/Output Pins

Table 24.1 shows the RSPI pin configuration. RSPIi automatically switches the input/output direction of the SSLi0 pin, which is set to output in single-master mode and to input in multi-master or slave mode. RSPIi automatically switches the input/output directions of the RSPCKi, MOSIi, and MISOi pins according to master/slave mode setting and the input level on the SSLi0 pin. (See section 24.4.2, Controlling RSPIi Pins.)

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 24.1 Pin Configuration

Channel	Abbreviation	I/O	Function
0	RSPCK0	I/O	RSPI0 clock input/output
	MOSI0	I/O	RSPI0 master transmit data
	MISO0	I/O	RSPI0 slave transmit data
	SSL00	I/O	RSPI0 slave select
	SSL01	Output	RSPI0 slave select
1	RSPCK1	I/O	RSPI1 clock input/output
	MOSI1	I/O	RSPI1 master transmit data
	MISO1	I/O	RSPI1 slave transmit data
	SSL10	I/O	RSPI1 slave select
2	RSPCK2	I/O	RSPI2 clock input/output
	MOSI2	I/O	RSPI2 master transmit data
	MISO2	I/O	RSPI2 slave transmit data
	SSL20	I/O	RSPI2 slave select

24.3 Register Descriptions

Table 24.2 shows the RSPI register configuration. These registers enable the RSPI to perform the following controls: specifying master/slave modes, specifying a transfer format, and controlling the transmitter and receiver.

Table 24.2 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
RSPI0 control register	SP0CR	H'00	H'FFFF B000	8, 16	24-7
RSPI0 slave select polarity register	SP0SSLP	H'00	H'FFFF B001	8, 16	24-9
RSPI0 pin control register	SP0PCR	H'00	H'FFFF B002	8, 16	24-10
RSPI0 status register	SP0SR	H'22	H'FFFF B003	8, 16	24-11
RSPI0 data register	SP0DR	H'0000 0000	H'FFFF B004	16, 32	24-14
RSPI0 sequence control register	SP0SCR	H'00	H'FFFF B008	8, 16	24-15
RSPI0 sequence status register	SP0SSR	H'00	H'FFFF B009	8, 16	24-16
RSPI0 bit rate register	SP0BR	H'FF	H'FFFF B00A	8, 16	24-17
RSPI0 data control register	SP0DCR	H'00	H'FFFF B00B	8, 16	24-18
RSPI0 clock delay register	SP0CKD	H'00	H'FFFF B00C	8, 16	24-21
RSPI0 slave select negation delay register	SP0SSLND	H'00	H'FFFF B00D	8, 16	24-22
RSPI0 next-access delay register	SP0ND	H'00	H'FFFF B00E	8	24-23
RSPI0 command register 0	SP0CMD0	H'070D	H'FFFF B010	16	24-24
RSPI0 command register 1	SP0CMD1	H'070D	H'FFFF B012	16	24-24
RSPI0 command register 2	SP0CMD2	H'070D	H'FFFF B014	16	24-24
RSPI0 command register 3	SP0CMD3	H'070D	H'FFFF B016	16	24-24
RSPI1 control register	SP1CR	H'00	H'FFFF B100	8, 16	24-7
RSPI1 slave select polarity register	SP1SSLP	H'00	H'FFFF B101	8, 16	24-9
RSPI1 pin control register	SP1PCR	H'00	H'FFFF B102	8, 16	24-10
RSPI1 status register	SP1SR	H'22	H'FFFF B103	8, 16	24-11
RSPI1 data register	SP1DR	H'0000 0000	H'FFFF B104	16, 32	24-14
RSPI1 sequence control register	SP1SCR	H'00	H'FFFF B108	8, 16	24-15
RSPI1 sequence status register	SP1SSR	H'00	H'FFFF B109	8, 16	24-16
RSPI1 bit rate register	SP1BR	H'FF	H'FFFF B10A	8, 16	24-17
RSPI1 data control register	SP1DCR	H'00	H'FFFF B10B	8, 16	24-18
RSPI1 clock delay register	SP1CKD	H'00	H'FFFF B10C	8, 16	24-21
RSPI1 slave select negation delay register	SP1SSLND	H'00	H'FFFF B10D	8, 16	24-22
RSPI1 next-access delay register	SP1ND	H'00	H'FFFF B10E	8	24-23
RSPI1 command register 0	SP1CMD0	H'070D	H'FFFF B110	16	24-24
RSPI1 command register 1	SP1CMD1	H'070D	H'FFFF B112	16	24-24
RSPI1 command register 2	SP1CMD2	H'070D	H'FFFF B114	16	24-24
RSPI1 command register 3	SP1CMD3	H'070D	H'FFFF B116	16	24-24

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
RSPI2 control register	SP2CR	H'00	H'FFFF B200	8, 16	24-7
RSPI2 slave select polarity register	SP2SSLP	H'00	H'FFFF B201	8, 16	24-9
RSPI2 pin control register	SP2PCR	H'00	H'FFFF B202	8, 16	24-10
RSPI2 status register	SP2SR	H'22	H'FFFF B203	8, 16	24-11
RSPI2 data register	SP2DR	H'0000 0000	H'FFFF B204	16, 32	24-14
RSPI2 sequence control register	SP2SCR	H'00	H'FFFF B208	8, 16	24-15
RSPI2 sequence status register	SP2SSR	H'00	H'FFFF B209	8, 16	24-16
RSPI2 bit rate register	SP2BR	H'FF	H'FFFF B20A	8, 16	24-17
RSPI2 data control register	SP2DCR	H'00	H'FFFF B20B	8, 16	24-18
RSPI2 clock delay register	SP2CKD	H'00	H'FFFF B20C	8, 16	24-21
RSPI2 slave select negation delay register	SP2SSLND	H'00	H'FFFF B20D	8, 16	24-22
RSPI2 next-access delay register	SP2ND	H'00	H'FFFF B20E	8	24-23
RSPI2 command register 0	SP2CMD0	H'070D	H'FFFF B210	16	24-24
RSPI2 command register 1	SP2CMD1	H'070D	H'FFFF B212	16	24-24
RSPI2 command register 2	SP2CMD2	H'070D	H'FFFF B214	16	24-24
RSPI2 command register 3	SP2CMD3	H'070D	H'FFFF B216	16	24-24

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

24.3.1 RSPIi Control Register (SPiCR)

The SPiCR rgister sets the operating mode of the RSPIi. If the MSTR and MODFEN bits are changed while the RSPI function is enabled by setting the SPE bit to "1", operation cannot be guaranteed.

RSPI0 Control Register (SP0CR)	<P4 address: location H'FFFF B000>
RSPI1 Control Register (SP1CR)	<P4 address: location H'FFFF B100>
RSPI2 Control Register (SP2CR)	<P4 address: location H'FFFF B200>

Bit:	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MOD FEN	—	SPMS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	SPRIE	0	R	W	RSPI Receive Interrupt Enable Bit If the RSPIi has detected a receive buffer write after completion of a serial transfer and the SPRF bit in the RSPIi status register (SPiSR) is set to "1", this bit enables or disables the generation of an RSPIi receive interrupt request. 0: Disables the generation of RSPIi receive interrupt requests. 1: Enables the generation of RSPIi receive interrupt requests.
6	SPE	0	R	W	RSPI Function Enable Bit Setting this bit to "1" enables the RSPI function. When the MODF bit in the RSPIi status register (SPiSR) is "1", the SPE bit cannot be set to "1" (see section 24.4.7, Error Detection). Setting the SPE bit to "0" disables the RSPI function, and initializes a part of the module function (see section 24.4.8, Initializing RSPI). 0: Disables the RSPIi function 1: Enables the RSPIi function
5	SPTIE	0	R	W	RSPI Transmit Interrupt Enable Bit Enables or disables the generation of RSPIi transmit interrupt requests when the RSPIi detects transmit buffer empty and sets the SPTEF bit in the RSPIi status register (SPiSR) to "1". In the RSPI disabled (with the SPE bit "0") status, the SPTEF bit is "1". Therefore, note that setting the SPTIE bit to "1" when the RSPI is in the disabled status generates an RSPIi transmit interrupt request. 0: Disables the generation of RSPIi transmit interrupt requests. 1: Enables the generation of RSPIi transmit interrupt requests.
4	SPEIE	0	R	W	RSPI Error Interrupt Enable Bit Enables or disables the generation of RSPIi error interrupt requests when the RSPIi detects a mode fault error and sets the MODF bit in the RSPIi status register (SPiSR) to "1", or when the RSPIi detects and sets the OVRF bit in the SPiSR register to "1" (see section 24.4.7, Error Detection). 0: Disables the generation of RSPIi error interrupt requests. 1: Enables the generation of RSPIi error interrupt requests.

Bit	Abbreviation	After Reset	R	W	Description
3	MSTR	0	R	W	RSPI Master/Slave Mode Select Bit Selects master/slave mode of RSPIi. According to the MSTR bit settings, the RSPIi determines the direction of pins RSPCKi, MOSIi, MISOi, SSLi0, and SSLi1. 0: Slave mode 1: Master mode
2	MODFEN	0	R	W	Mode Fault Error Detection Enable Bit Enables or disables the detection of mode fault error (see section 24.4.7, Error Detection). In addition, the RSPIi determines the input/output directions of the SSLi0 pin based on combinations of the MODFEN and MSTR bits (see section 24.4.2, Controlling RSPIi Pins). 0: Disables the detection of mode fault error 1: Enables the detection of mode fault error
1	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
0	SPMS	0	R	W	RSPI Mode Select Bit Selects SPI (four-wire) or clock-synchronous (three-wire) operation. In clock-synchronous mode, pins SSLi0 and SSLi1 are unused and three pins, RSPCKi, MOSIi, and MISOi, are used for communication. To use clock-synchronous mode, set the CPHA bit in RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3) to "1". Operation cannot be guaranteed if the CPHA bit is cleared to "0". 0: SPI (four-wire) operation 1: Clock-synchronous (three-wire) operation

Note: RSPIi operating mode is selected according to the combination of the MODFEN, MSTR, and SPMS bits. For details, see section 24.4.1, Overview of RSPIi Operations.

24.3.2 RSPIi Slave Select Polarity Register (SPiSSLP)

The SPiSSLP register sets the polarity of the SSLi0 and SSLi1 signals of the RSPIi. If the contents of the SPiSSLP register are changed while the RSPI function is enabled by setting the SPE bit in the RSPIi control register (SPiCR) to "1", operation cannot be guaranteed.

RSPI0 Slave Select Polarity Register (SP0SSLP) <P4 address: location H'FFFF B001>
RSPI1 Slave Select Polarity Register (SP1SSLP) <P4 address: location H'FFFF B101>
RSPI2 Slave Select Polarity Register (SP2SSLP) <P4 address: location H'FFFF B201>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SSL1P	SSL0P
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	SSL1P*1	0	R	W	SSLi1 Signal Polarity Setting Bit This bit sets the polarity of the SSLi1 signal. It also indicates the active polarity of the SSLi1 signal. 0: SSLi1 signal "L" active 1: SSLi1 signal "H" active
0	SSL0P	0	R	W	SSLi0 Signal Polarity Setting Bit This bit sets the polarity of the SSLi0 signal. It also indicates the active polarity of the SSLi0 signal. 0: SSLi0 signal "L" active 1: SSLi0 signal "H" active

Note: *1 This bit is a reserved bit when i = 1 or 2. And this bit is always read as "0". The write value should always be "0".

24.3.3 RSPIi Pin Control Register (SPiPCR)

The SPiPCR register sets the modes of the RSPIi pins. If the contents of this register are changed by the CPU while the RSPI function is enabled by setting the SPE bit in the SPiPCR register to "1", operation cannot be guaranteed.

RSPI0 Pin Control Register (SP0PCR)	<P4 address: location H'FFFF B002>
RSPI1 Pin Control Register (SP1PCR)	<P4 address: location H'FFFF B102>
RSPI2 Pin Control Register (SP2PCR)	<P4 address: location H'FFFF B202>

Bit:	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	SPOM	—	SPLP
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	MOIFE	0	R	W	MOSI Idle Value Fixing Enable Bit Fixes the MOSIi output value when RSPIi is in master mode and in an SSL negation period (including the SSL retention period during a burst transfer). When the MOIFE bit is cleared to "0", RSPIi outputs on the MOSIi pin the last data unit from the previous serial transfer during the SSL negation period. When the MOIFE bit is set to "1", RSPI outputs the fixed value specified by the MOIFV bit on the MOSIi pin. 0: MOSIi output value equals final data from previous transfer 1: MOSIi output value equals the value set in the MOIFV bit
4	MOIFV	0	R	W	MOSI Idle Fixed Value Bit If the MOIFE bit is "1" in master mode, the RSPIi, according to the MOIFV bit settings, determines the MOSIi signal value during the SSL negation period (including the SSL retention period during a burst transfer). 0: MOSIi Idle fixed value equals "L" level 1: MOSIi Idle fixed value equals "H" level
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2	SPOM	0	R	W	RSPI Output Pin Mode Bit Sets the RSPIi output pins to CMOS output/open drain output. 0: CMOS output 1: Open-drain output
1	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
0	SPLP	0	R	W	RSPI Loopback Bit When the SPLP bit is set to "1", the RSPIi shuts off the path between the MISOi pin and the shift register, and between the MOSIi pin and the shift register, and connects (reverses) the input path and the output path for the shift register. 0: Normal mode 1: Loopback mode

24.3.4 RSPIi Status Register (SPiSR)

The SPiSR register contains flag bits that indicate the operating status of the RSPI. Writing to the SPiSR register is valid only under specific conditions.

RSPI0 Status Register (SP0SR)	<P4 address: location H'FFFF B003>
RSPI1 Status Register (SP1SR)	<P4 address: location H'FFFF B103>
RSPI2 Status Register (SP2SR)	<P4 address: location H'FFFF B203>

Bit:	7	6	5	4	3	2	1	0
	SPRF	—	SPTEF	—	—	MODF	MIDLE	OVRF
After Reset:	0	0	1	0	0	0	1	0

<After Reset: H'22>

Bit	Abbreviation	After Reset	R	W	Description
7	SPRF	0	R	*[1]	RSPI Receive Buffer Full Flag Indicates the status of the receive buffer for the RSPIi data register (SPiDR). Upon completion of a serial transfer with the SPRF bit "0", the RSPIi transfers the receive data from the shift register to the SPiDR register, and sets this bit to "1". Since RSPIi performs full-duplex serial communication, this is also the time at which the last bit of the transmit data is sent. The SPRF bit is cleared to "0" under the following conditions: • The CPU reads the SPiSR register when the SPRF bit is "1", and then the CPU writes a "0" to the SPRF bit. • The CPU or DMAC reads received data from the SPiDR register. • Hardware reset If a serial transfer ends while the SPRF bit is "1", the RSPIi does not transfer the received data from the shift register to the SPiDR register. When the OVRF bit in the SPiSR register is "1", the SPRF bit cannot be changed from "0" to "1" (see section 24.4.7, Error Detection). 0: No valid data in the SPiDR register 1: Valid data found in the SPiDR register
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
5	SPTEF	1	R	*[1]	RSPI Transmit Buffer Empty Flag Indicates the status of the transmit buffer for the RSPIi data register (SPiDR). After the initialization of RSPIi or after transmit data is transferred from the transmit buffer to the shift register, the RSPIi sets the SPTEF bit to "1". The SPTEF bit is cleared to "0" under the following conditions. If the SPTEF bit is cleared and the shift register is empty, the data is copied from the transmit buffer to the shift register. • The CPU reads the SPiSR register when the SPTEF bit is "1", and then the CPU writes "0" to the SPTEF bit. • The CPU or DMAC writes the transmit data to the SPiDR register. The CPU or DMAC can write to the SPiDR register only when the SPTEF bit is set to "1". If the CPU or DMAC writes to the transmit buffer of the SPiDR register when the SPTEF bit is cleared to "0", the data in the transmit buffer is not updated. 0: Data found in the transmit buffer 1: No data in the transmit buffer
4, 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	MODF	0	R	*[1]	Mode Fault Error Flag Indicates the occurrence of a mode fault error. When the input level of the SSLi0 pin changes to the active level while the MSTR bit in the RSPIi control register (SPiCR) is "1" and the MODFEN bit is "1" with the RSPI being in multi-master mode, the RSPIi detects a mode fault error and sets the MODF bit to "1". Similarly, if the MODFEN bit is set to "1" when the MSTR bit is "0" and the RSPIi is in slave mode, and the SSLi0 pin is negated before the RSPCK cycle necessary for data transfer ends, the RSPI detects a mode fault error. The active level of the SSLi0 signal is determined by the SSL0P bit in the RSPIi slave select polarity register (SPiSSLP). The MODF bit is cleared to "0" under the following conditions. • The CPU reads the SPiSR register when the MODF bit is 1, and then writes "0" to the MODF bit. • Hardware reset 0: No mode fault error occurs 1: A mode fault error occurs

Bit	Abbreviation	After Reset	R	W	Description
1	MIDLE	1	R	0	RSPI Idle Flag Indicates the transfer status of RSPIi. The conditions under which the MIDLE bit is set to "1" in master mode (single or multi) are as follows: • The SPE bit in the SPiCR register is cleared to "0" (RSPIi initialization). or • The SPTEF bit in the SPiSR register is set to "1" (next unit of transmit data not set). • The SPCP bits in the SPiSSR register are set to "00" (sequence control command pointer positioned at start of loop). • The RSPIi internal sequence has transitioned to idle status (state in which operation has finished until access delay time elapses). The MIDLE bit is set to "1" when all three of the above conditions are satisfied. The MIDLE bit is cleared to "0" when the above conditions are not satisfied. The condition for setting the MIDLE bit is set to "1" in slave mode is as follows: • The SPE bit in the SPiCR register is cleared to "0" (RSPIi initialization). The MIDLE bit is cleared to "0" when SPE is set to "1". 0: RSPIi in transfer status 1: RSPIi in idle status
0	OVRF	0	R	W	Overrun Error Flag Indicates the occurrence of an overrun error. If a serial transfer ends while the SPRF bit is "1", the RSPIi detects an overrun error, and sets the OVRF bit to "1". The OVRF bit is cleared to "0" under the following conditions. • The CPU reads the SPiSR register when the OVRF bit is "1", and then writes "0" to the OVRF bit. • Hardware reset 0: No overrun error occurs 1: An overrun error occurs

Note: *1 Only "0" can be written to this bit after reading it as "1" to clear the flag.

24.3.5 RSPIi Data Register (SPiDR)

The SPiDR register is a buffer that stores RSPIi transmit and receive data. The transmit buffer (SPTX) and receive buffer (SPRX) are independent of each other and are mapped to the SPiDR register.

When the SPiDCR.SPLW bit is cleared to "0", bit 31 to bit 16 in the SPiDR register correspond to a buffer, and the range from bit 16 (LSB) to the assigned data length is treated as transfer data. Perform read and write accesses to the SPiDR register in word units.

When the SPiDCR.SPLW bit is set to "1", bit 31 to bit 0 in the SPiDR register correspond to a buffer, and the range from bit 0 (LSB) to the assigned data length is treated as transfer data. Perform read and write accesses to the SPiDR register in longword units.

When the SPLW bit is cleared to "0", the SPiDR register functions as a 64-bit buffer comprising four frames of up to 16 bits each. When the SPLW bit is set to "1", the SPiDR register functions as a 128-bit buffer comprising four frames of up to 32 bits each.

The frame length used by the SPiDR register is determined by the frame count setting bits (SPFC) in the RSPIi data control register (SPiDCR), and the bit length is determined by the RSPI data length setting bits (SPB) in RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3).

When the CPU or DMAC requests a write to the SPiDR register, and if the SPTEF bit in the RSPIi status register (SPiSR) is set to "1", RSPIi writes data to the transmit buffer of the SPiDR register. If the SPTEF bit is cleared to "0", RSPIi does not update the transmit buffer of the SPiDR register.

When the CPU or DMAC requests a read to the SPiDR register, the receive buffer is read when the RSPI receive/transmit data select bit (SPRDTD) in the RSPIi data control register (SPiDCR) is cleared to "0", and the transmit buffer is read when the SPRDTD bit is set to "1".

When the transmit buffer is read, the value written immediately previously is returned. The read value is "0" in all positions when the SPTEF bit in the RSPIi status register (SPiSR) is cleared to "0".

The normal operating method is to clear the SPRDTD bit to "0" and for the CPU or DMAC to read the receive buffer when the SPRF bit in the SPiSR register is set to "1" (unread data stored in receive buffer). When the SPRF or OVRF bit in the SPiSR register is set to "1", RSPIi does not update the receive buffer of SPiDR at the termination of a serial transfer.

RSPI0 Data Register (SP0DR) <P4 address: location H'FFFF B004>
RSPI1 Data Register (SP1DR) <P4 address: location H'FFFF B104>
RSPI2 Data Register (SP2DR) <P4 address: location H'FFFF B204>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	SPD31 to 0	All 0	R	W	Buffer for storing RSPIi transmit and receive data.

24.3.6 RSPIi Sequence Control Register (SPiSCR)

The SPiSCR register sets the sequence controlled method when the RSPI operates in master mode. If the contents of the SPiSCR register are changed by the CPU while the MSTR and SPE bits in the RSPIi control register (SPiCR) are "1" with the RSPI function enabled, the rewrite operation must be performed when the MIDLE bit in the RSPIi status register (SPiSR) is "1".

RSPI0 Sequence Control Register (SP0SCR)	<P4 address: location H'FFFF B008>
RSPI1 Sequence Control Register (SP1SCR)	<P4 address: location H'FFFF B108>
RSPI2 Sequence Control Register (SP2SCR)	<P4 address: location H'FFFF B208>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSLN	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1, 0	SPSLN	00	R	W	RSPI Sequence Length Setting Bits These bits set a sequence length when the RSPIi in master mode performs sequential operations. The RSPIi in master mode changes RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3) to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN bits. The relationship among the setting value of these bits sequence length, and referenced SPiCMD0 to SPiCMD3 register number is shown below. When the RSPIi is in slave mode, the SPiCMD0 register is always referenced. Sequence Length Referenced SPiCMD register (No) 00: 1 0 → 0 → … 01: 2 0 → 1 → 0 → … 10: 3 0 → 1 → 2 → 0 → … 11: 4 0 → 1 → 2 → 3 → 0 → …

24.3.7 RSPIi Sequence Status Register (SPiSSR)

The SPiSSR register indicates the sequence control status when the RSPI operates in master mode. SPiSSR is a read-only register.

RSPI0 Sequence Status Register (SP0SSR) <P4 address: location H'FFFF B009>
RSPI1 Sequence Status Register (SP1SSR) <P4 address: location H'FFFF B109>
RSPI2 Sequence Status Register (SP2SSR) <P4 address: location H'FFFF B209>

Bit:	7	6	5	4	3	2	1	0
	—	—	SPECM		—	—	SPCP	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	N	Reserved Bits These bits are always read as "0".
5, 4	SPECM	00	R	N	RSPI Error Command Bits These bits indicate RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3) that are pointed to by the command pointer (SPCP) when an error is detected during sequence control by the RSPIi. The RSPIi updates the SPECM bit only when an error is detected. If both the OVRF and MODF bits in the RSPIi status register (SPiSR) are "0" and there is no error, the values of the SPECM bit has no meaning. The correspondence between the value of the SPECM bits and registers SPiCMD0 to SPiCMD3 is shown below. For the RSPIi's error detection function, see section 24.4.7, Error Detection. For the RSPIi's sequence control, see section 24.4.9 (1), Master Mode Operation. 00: SPiCMD0 register 01: SPiCMD1 register 10: SPiCMD2 register 11: SPiCMD3 register
3, 2	—	All 0	0	N	Reserved Bits These bits are always read as "0".
1, 0	SPCP	00	R	N	RSPI Command Pointer Bits During RSPIi sequence control, these bits indicate RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3), which are currently pointed to by the pointers. For the RSPI's sequence control, see section 24.4.9 (1), Master Mode Operation. 00: SPiCMD0 register 01: SPiCMD1 register 10: SPiCMD2 register 11: SPiCMD3 register

24.3.8 RSPIi Bit Rate Register (SPiBR)

The SPiBR register is used to set the bit rate in master mode. If the contents of the SPiBR register are changed by the CPU while the MSTR and SPE bits in the RSPIi control register (SPiCR) are "1" with the RSPI function in master mode enabled, operation cannot be guaranteed.

RSPI0 Bit Rate Register (SP0BR)	<P4 address: location H'FFFF B00A>
RSPI1 Bit Rate Register (SP1BR)	<P4 address: location H'FFFF B10A>
RSPI2 Bit Rate Register (SP2BR)	<P4 address: location H'FFFF B20A>

Bit:	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
After Reset:	1	1	1	1	1	1	1	1

<After Reset: H'FF>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	SPR7 to 0	All 1	R	W	These bits set the bit rate in master mode.

When the RSPLi is used in slave mode, the RSPIi bit rate is dependent on the bit rate of the input clock, regardless of the settings of the SPiBR register and BRDV bits in SPiCMD0 to SPiCMD3 registers.

Otherwise, the bit rate is determined by the combination of the setting values of the SPiBR register and of the BRDV bits in RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3). The bit rate must be satisfy the specified electrical characteristics. The equation for calculating the bit rate is given below. In the equation, *n* denotes the setting value of the SPiBR register (0, 1, 2, …, 255), and *N* denotes the setting value of the BRDV bits (0, 1, 2, 3).

$$\text{Bit rate} = \frac{f\,(\text{Pck})}{2 \times (n + 1) \times 2^N}$$

Table 24.3 shows examples of the relationship between the SPiBR register and BRDV bit settings.

Table 24.3 Relationship between the SPiBR Register and BRDV Bit Settings

SPiBR Register Settings (n)	BRDV Bit Settings (N)	Division Ratio
1	0	4
2	0	6
3	0	8
4	0	10
5	0	12
5	1	24
5	2	48
5	3	96
255	3	4096

24.3.9 RSPIi Data Control Register (SPiDCR)

The SPiDCR register is used to specify the number of frames that can be stored in the SPiDR register, whether values read from the SPiDR register are treated as receive or transmit buffer values, and whether the SPiDR register is accessed in longword or word units. A maximum of four frames of data can be transferred by starting a single transmit or receive operation, according to the combination of the settings of the RSPI data length setting bits (SPB) in RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3), the sequence length setting bits (SPSLN) in the RSPIi sequence control register (SPiSCR), and the frame count setting bits (SPFC) in the RSPIi data control register (SPiDCR). If the CPU rewrites the SPFC bit in the SPiDCR register while the SPE bit in the RSPIi control register (SPiCR) is set to "1" with the RSPI function enabled, the rewrite operation must be performed when the MIDLE bit in the RSPIi register (SPiSR) is "1".

RSPI0 Data Control Register (SP0DCR) <P4 address: location H'FFFF B00B>
RSPI1 Data Control Register (SP1DCR) <P4 address: location H'FFFF B10B>
RSPI2 Data Control Register (SP2DCR) <P4 address: location H'FFFF B20B>

Bit:	7	6	5	4	3	2	1	0
	—	—	SPLW	SP RDTD	—	—	SPFC	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	SPLW	0	R	W	RSPI Longword Access/Word Access Setting Bit Specifies the access width for the RSPIi data register (SPiDR). Access bit 31 to bit 16 in word units when the SPLW bit is cleared to "0" and in longword units when the SPLW bit is set to "1". When the SPLW bit is cleared to "0", set the RSPI data length setting bits (SPB) in RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3) to specify between 8 and 16 bits. Operation cannot be guaranteed when 20, 24, or 32 bits is specified. 0: Word access to SPiDR register 1: Longword access to SPiDR register
4	SPRDTD	0	R	W	RSPI Receive/Transmit Data Select Bit Specifies whether values read from the RSPIi data register (SPiDR) are treated as receive or transmit buffer values. When the transmit buffer is read, the value written to the SPiDR register immediately previously is returned. Read the transmit buffer when the SPTEF bit in the RSPIi status register (SPiSR) is set to "1". 0: Value read from SPiDR register treated as receive buffer value 1: Value read from SPiDR register treated as transmit buffer value (provided SPTEF bit is set to "1")
3, 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
1, 0	SPFC	00	R	W	Frame Count Setting Bits These bits specify the number of frames that can be stored in the SPiDR register. A maximum of four frames of data can be transferred by starting a single transmit or receive operation, according to the combination of the settings of the RSPI data length setting bits (SPB) in RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3), the sequence length setting bits (SPSLN) in the RSPIi sequence control register (SPiSCR), and the frame count setting bits (SPFC) in the RSPIi data control register (SPiDCR). The SPFC bits also specify the number of receive data units at which the RSPI receive buffer full flag (SPRF) in the RSPIi status register (SPiSR) is set to "1". Table 24.4 and figure 24.2 show frame configurations that can be stored in the SPiDR register and example combinations of transmit and receive settings. Subsequent operation cannot be guaranteed when settings other than those shown in the example combinations are used. 00: 1 01: 2 10: 3 11: 4

Table 24.4 Frame Settings and Corresponding Bit Values

Settings	SPB Bits	SPSLN Bits	SPFC Bits	Number of Frames Transferred	Number of Frames when SPRF Bit = "1" and SPTEF Bit = "0"
1-1	N	00	00	1	1
1-2	N	00	01	2	2
1-3	N	00	10	3	3
1-4	N	00	11	4	4
2-1	N, M	01	01	2	2
2-2	N, M	01	11	4	4
3	N, M, O	10	10	3	3
4	N, M, O, P	11	11	4	4

Legend: N, M, O, P: Data lengths that can be specified by the SPB bits.

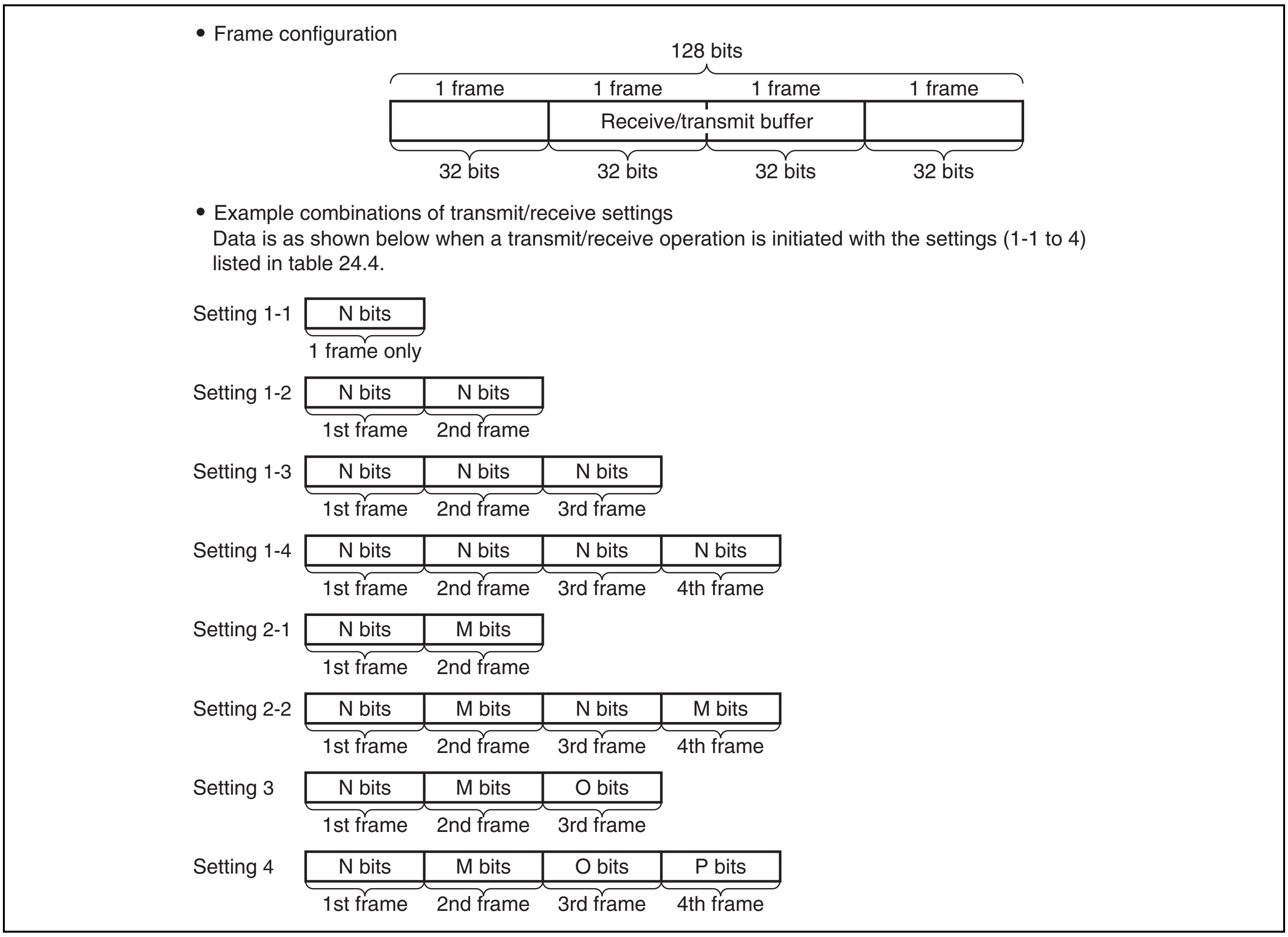


Figure 24.2 Frame Configurations and Example Combinations of Transmit/Receive Settings

24.3.10 RSPIi Clock Delay Register (SPiCKD)

The SPiCKD register sets a period from the beginning of SSL signal assertion to RSPCK oscillation (RSPCK delay) when the SCKDEN bit in RSPIi command registers 0 to 3 (SPiCMD to SPiCMD3) is "1". If the contents of the SPiCKD register are changed by the CPU while the MSTR and SPE bits in the RSPIi control register (SPiCR) are "1" with the RSPIi in master mode enabled, operation cannot be guaranteed.

Set the SCKDL bits to B'000 when the RSPI is used in slave mode.

RSPI0 Clock Delay Register (SP0CKD)	<P4 address: location H'FFFF B00C>
RSPI1 Clock Delay Register (SP1CKD)	<P4 address: location H'FFFF B10C>
RSPI2 Clock Delay Register (SP2CKD)	<P4 address: location H'FFFF B20C>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKDL		
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	SCKDL	000	R	W	RSPCK Delay Setting Bits These bits set an RSPCK delay value when the SCKDEN bit in the SPiCMD0 to SPiCMD3 registers is "1". The relationship between the setting value of the SCKDL bits and the delay value is as follows: 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK

24.3.11 RSPIi Slave Select Negation Delay Register (SPiSSLND)

The SPiSSLND register sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSL signal during a serial transfer by the RSPIi in master mode. If the contents of the SPiSSLND register are changed by the CPU while the MSTR and SPE bits in the RSPIi control register (SPiCR) are "1" with the RSPIi in master mode enabled, operation cannot be guaranteed.

To use RSPIi in slave mode, set the SLNDL bits to B'000.

RSPI0 Slave Select Negation Delay Register (SP0SSLND) <P4 address: location H'FFFF B00D>
RSPI1 Slave Select Negation Delay Register (SP1SSLND) <P4 address: location H'FFFF B10D>
RSPI2 Slave Select Negation Delay Register (SP2SSLND) <P4 address: location H'FFFF B20D>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL		
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	SLNDL	000	R	W	SSL Negation Delay Setting Bits These bits set an SSL negation delay value when the RSPIi is in master mode. The relationship between the setting value of the SLNDL bits and the SSL negation delay value is as follows: 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK

24.3.12 RSPIi Next-Access Delay Register (SPiND)

The SPiND register specifies the non-active period (next-access delay) of the SSL signal after termination of a serial transfer when the SPNDEN bit in RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3) is set to "1". Operation cannot be guaranteed if the SPiND register is rewritten when the MSTR and SPE bits in the RSPIi control register (SPiCR) are set to "1" and RSPIi is enabled in master mode.

To use RSPIi in slave mode, set the SPNDL bits to B'000.

RSPI0 Next-Access Delay Register (SP0ND)	<P4 address: location H'FFFF B00E>
RSPI1 Next-Access Delay Register (SP1ND)	<P4 address: location H'FFFF B10E>
RSPI2 Next-Access Delay Register (SP2ND)	<P4 address: location H'FFFF B20E>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPNDL		
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2 to 0	SPNDL	000	R	W	RSPI Next-Access Delay Setting Bits These bits set a next-access delay when the SPNDEN bit in the SPiCMD0 to SPiCMD3 registers is "1". The relationship between the setting value of the SPNDL bits and the next-access delay value is as follows: 000: 1 RSPCK + 2Pck 001: 2 RSPCK + 2Pck 010: 3 RSPCK + 2Pck 011: 4 RSPCK + 2Pck 100: 5 RSPCK + 2Pck 101: 6 RSPCK + 2Pck 110: 7 RSPCK + 2Pck 111: 8 RSPCK + 2Pck

24.3.13 RSPIi Command Registers 0 to 3 (SPiCMD0 to SPiCMD3)

Each RSPI channel has four RSPIi command registers (SPiCMD), numbered SPiCMD0 to SPiCMD3. Registers SPiCMD0 to SPiCMD3 are used to set the transfer format for RSPIi in master mode. Also, some of the bits in the SPiCMD0 register are used to set the transfer mode for RSPIi in slave mode. In master mode, RSPIi sequentially references registers SPiCMD0 to SPiCMD3, according to the setting of the SPSLN bits in the RSPIi sequence control register (SPiSCR), and executes the serial transfer specified in the referenced SPiCMD registers.

The SPiCMD register settings should be made by referencing the SPiCMD registers when the SPTEF bit in the RSPIi status register (SPiSR) is set to "1" and making the new settings before the data to be transmitted has been specified.

The SPiCMD registers referenced by the RSPIi in master mode can be confirmed by means of the SPCP bits in the RSPIi sequence status register (SPiSSR). When RSPIi enabled in slave mode, operation cannot be guaranteed if the SPiCMD0 register is rewritten.

RSPI0 Command Registers 0 to 3 (SP0CMD0 to SP0CMD3)	<P4 address: location H'FFFF B010 to H'FFFF B016>
RSPI1 Command Registers 0 to 3 (SP1CMD0 to SP1CMD3)	<P4 address: location H'FFFF B110 to H'FFFF B116>
RSPI2 Command Registers 0 to 3 (SP2CMD0 to SP2CMD3)	<P4 address: location H'FFFF B210 to H'FFFF B216>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB				SSLKP	SSLA			BRDV		CPOL	CPHA
After Reset:	0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	1

<After Reset: H'070D>

Bit	Abbreviation	After Reset	R	W	Description
15	SCKDEN	0	R	W	RSPCK Delay Setting Enable Bit Sets the period from the time the RSPIi in master mode sets the SSL signal active until the RSPI oscillates RSPCK (RSPCK delay). If the SCKDEN bit is "0", the RSPIi sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is "1", the RSPIi starts the oscillation of RSPCK at an RSPIi clock delay in compliance with the RSPIi clock delay register (SPiCKD) settings. To use the RSPIi in slave mode, the SCKDEN bit should be set to "0". 0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay equal to the RSPIi clock delay register (SPiCKD) settings
14	SLNDEN	0	R	W	SSL Negation Delay Setting Enable Bit Sets the period (SSL negation delay) from the time the RSPIi in master mode stops RSPCK oscillation until the RSPIi sets the SSL signal inactive. If the SLNDEN bit is "0", the RSPIi sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is "1", the RSPIi negates the SSL signal at an SSL negation delay in compliance with the RSPIi slave select negation delay register (SPiSSLND) settings. To use the RSPIi in slave mode, the SLNDEN bit should be set to "0". 0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay equal to the RSPIi slave select negation delay register (SPiSSLND) settings

Bit	Abbreviation	After Reset	R	W	Description
13	SPNDEN	0	R	W	RSPI Next-Access Delay Enable Bit Sets the period from the time the RSPIi in master mode terminates a serial transfer and sets the SSL signal inactive until the RSPIi enables the SSL signal assertion for the next access (next-access delay). If the SPNDEN bit is "0", the RSPIi sets the next-access delay to 1 RSPCK. If the SPNDEN bit is "1", the RSPIi inserts a next-access delay in compliance with the RSPIi next-access delay register (SPiND) settings. To use the RSPIi in slave mode, the SPNDEN bit should be set to "0". 0: A next-access delay of 1 RSPCK 1: A next-access delay equal to the RSPIi next-access delay register (SPiND) settings
12	LSBF	0	R	W	RSPI LSB First Sets the data format of the RSPIi in master mode or slave mode to MSB first or LSB first. 0: MSB first 1: LSB first
11 to 8	SPB	0111	R	W	RSPI Data Length Setting Bits These bits set a transfer data length for the RSPIi in master mode or slave mode. Set the SPB bits to a value corresponding to 8 to 16 bits when the value of the SPLW bit in SPiDCR is "0". Operation cannot be guaranteed when a setting of 20 to 32 bits is selected. 0100 to 0111: 8 bits 1000: 9 bits 1001: 10 bits 1010: 11 bits 1011: 12 bits 1100: 13 bits 1101: 14 bits 1110: 15 bits 1111: 16 bits 0000: 20 bits 0001: 24 bits 0010, 0011: 32 bits
7	SSLKP	0	R	W	SSL Signal Level Keeping Bit When the RSPIi in master mode performs a serial transfer, this bit specifies whether the SSL 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. To use the RSPIi in slave mode, the SSLKP bit should be set to "0". 0: Negates all SSL signals upon completion of transfer. 1: Keeps the SSL signal level from the end of the transfer until the beginning of the next access.

Bit	Abbreviation	After Reset	R	W	Description
6 to 4	SSLA	000	R	W	SSL Signal Assertion Setting Bits These bits control the SSLi signal assertion when the RSPIi performs serial transfers in master mode. Setting the SSLA bit controls the assertion for the signals SSLi0 and SSLi1. When an SSL signal is asserted, its polarity is determined by the set value in the corresponding RSPIi slave select polarity register (SPiSSLP). When the SSLA bit is set to "B'000" or "B'1xx" in multi-master mode, serial transfers are performed with all the SSLi signals in the neagted state (as SSLi0 acts as input). In single-master mode, serial transfer takes place with all SSL signals in the negated state when the SSLA bits are set to B'1xx. To use the RSPIi in slave mode, set B'000 to SSLA bit. 000: SSL0 001: SSL1*1 010: Setting prohibited 011: Setting prohibited 1xx: — Note: *1 This setting is only for the RSPI0 channel.
3, 2	BRDV	11	R	W	Bit Rate Division Setting Bits These bits are used to determine the bit rate. A bit rate is determined by combinations of the BRDV bits and the settings in the RSPIi bit rate register (SPiBR) (see section 24.3.8, RSPIi Bit Rate Register (SPiBR)). The settings in SPiBR determine the base bit rate. The settings in the BRDV bits are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In the SPiCMD0 to SPiCMD3 registers different BRDV bit settings can be specified. This permits the execution of serial transfers at a different bit rate for each command. 00: Select the base bit rate 01: Select the base bit rate divided by 2 10: Select the base bit rate divided by 4 11: Select the base bit rate divided by 8
1	CPOL	0	R	W	RSPCK Polarity Setting Bit Sets an RSPCK polarity of the RSPIi in master or slave mode. Data communications between RSPIi modules require the same RSPCK polarity setting between the modules. 0: RSPCK = "0" when idle 1: RSPCK = "1" when idle
0	CPHA	1	R	W	RSPCK Phase Setting Bit Sets an RSPCK phase of the RSPIi in master or slave mode. Data communications between RSPIi modules require the same RSPCK phase setting between the modules. 0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge

24.4 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.

24.4.1 Overview of RSPIi Operations

RSPIi can perform synchronous serial transfers in slave (SPI operation), single-master (SPI operation), multi-master (SPI operation), slave (clock-synchronous operation), and master (clock-synchronous operation) modes. A particular mode of the RSPI can be selected by using the MSTR, MODFEN, and SPMS bits in the RSPIi control register (SPiCR). Table 24.5 gives the relationship between RSPI modes and the SPiCR register settings, and a description of each mode.

Table 24.5 Relationship between RSPIi Modes and the SPiCR Register Settings and Description of Each Mode

Mode	Slave Mode (SPI Operation)	Single-Master Mode (SPI Operation)	Multi-Master Mode (SPI Operation)	Slave (Clock-Synchronous Operation)	Master (Clock-Synchronous Operation)
MSTR bit setting	0	1	1	0	1
MODFEN bit setting	0, 1	0	1	0	0
SPMS bit setting	0	0	0	1	1
RSPCKi signal	Input	Output	Output/Hi-Z	Input	Output
MOSIi signal	Input	Output	Output/Hi-Z	Input	Output
MISOi signal	Output/Hi-Z	Input	Input	Output	Input
SSLi0 signal	Input	Output	Input	Hi-Z	Hi-Z
SSL01 signal	Hi-Z	Output	Output/Hi-Z	Hi-Z	Hi-Z
Output pin mode	CMOS/open-drain	CMOS/open-drain	CMOS/open-drain	CMOS/open-drain	CMOS/open-drain
SSL polarity modification function	Supported	Supported	Supported	—	—
Clock source	RSPCK input	On-chip baud rate generator	On-chip baud rate generator	RSPCK input	On-chip baud rate generator
Clock polarity	Two	Two	Two	Two	Two
Clock phase	Two	Two	Two	One (CPHA = "1")	One (CPHA = "1")
First transfer bit	MSB/LSB	MSB/LSB	MSB/LSB	MSB/LSB	MSB/LSB
Transfer data length	8 to 32 bits	8 to 32 bits	8 to 32 bits	8 to 32 bits	8 to 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 starting method	SSL input active or RSPCK oscillation	Write to transmit buffer when SPTEF = "1"	Write to transmit buffer when SPTEF = "1"	RSPCK oscillation	Write to transmit buffer when SPTEF = "1"

Mode	Slave Mode (SPI Operation)	Single-Master Mode (SPI Operation)	Multi-Master Mode (SPI Operation)	Slave (Clock-Synchronous Operation)	Master (Clock-Synchronous Operation)
Sequence control	Not supported	Supported	Supported	Not supported	Supported
Transmit buffer empty detection	Supported	Supported	Supported	Supported	Supported
Receive buffer full detection	Supported	Supported	Supported	Supported	Supported
Overrun error detection	Supported	Supported	Supported	Supported	Supported
Mode fault error detection	Supported (MODFEN = "1")	Not supported	Supported	Not supported	Not supported

24.4.2 Controlling RSPIi Pins

According to the settings of the MSTR, MODFEN, and SPMS bits in the RSPIi control register (SPiCR) and the SPOM bit in the RSPIi pin control register (SPiPCR), the RSPIi can automatically switch pin directions and output modes. Table 24.6 shows the relationship between pin states and bit settings.

Table 24.6 Relationship between RSPIi Pin States and Control Bit Setting Values

		Pin State*[1]	
Mode	Pin	SPOM = "0"	SPOM = "1"
Single-master mode (SPI operation) (MSTR = "1", MODFEN = "0", SPMS = "0")	RSPCKi	CMOS output	Open-drain output
	SSLi0, SSL01	CMOS output	Open-drain output
	MOSIi	CMOS output	Open-drain output
	MISOi	Input	Input
Multi-master mode (SPI operation) (MSTR = "1", MODFEN = "1", SPMS = "0")	RSPCKi*[2]	CMOS output/Hi-Z	Open-drain output/Hi-Z
	SSLi0	Input	Input
	SSL01*[2]	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MOSIi*[2]	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MISOi	Input	Input
Slave mode (SPI operation) (MSTR = "0", SPMS = "0")	RSPCKi	Input	Input
	SSLi0	Input	Input
	SSL01	Hi-Z	Hi-Z
	MOSIi	Input	Input
	MISOi*[3]	CMOS output/Hi-Z	Open-drain output/Hi-Z
Master (Clock-Synchronous Operation) (MSTR = "1", MODFEN = "0", SPMS = "1")	RSPCKi	CMOS output	Open-drain output
	SSLi0, SSL01*[4]	Hi-Z	Hi-Z
	MOSIi	CMOS output	Open-drain output
	MISOi	Input	Input

		Pin State*1	
Mode	**Pin**	**SPOM = "0"**	**SPOM = "1"**
Slave (Clock-Synchronous Operation) (MSTR = "0", SPMS = "1")	RSPCKi	Input	Input
	SSLi0, SSL01*4	Hi-Z	Hi-Z
	MOSIi	Input	Input
	MISOi	CMOS output	Open-drain output

Notes: *1 The RSPIi settings are not indicated by the multifunction pins for which the RSPIi function is not selected.

*2 When the SSLi0 pin is at the active level, the pin state is Hi-Z.

*3 When the SSLi0 pin is at the inactive level or the SPE bit in the SPiCR register is cleared to "0", the pin state is Hi-Z.

*4 Pins SSLi0 and SSL01 can be used as I/O ports during clock-synchronous operation.

In single-master (SPI operation) or multi-master (SPI operation) mode, RSPIi determines the MOSIi signal values during the SSL negation period (including the SSL retention period during a burst transfer) according to the settings of the MOIFE and MOIFV bits in the SPiPCR register, as shown in table 24.7.

Table 24.7 MOSIi Signal Value Determination during SSL Negation Period

MOIFE	MOIFV	MOSIi Signal Value during SSL Negation Period
0	0, 1	Final data from previous transfer
1	0	Always "L"
1	1	Always "H"

24.4.3 RSPI System Configuration Example

(1) Single Master/Single Slave (with this MCU Acting as Master, i = 0)

Figure 24.3 shows a single-master/single-slave RSPI system configuration example when this MCU is used as a master. In the single-master/single-slave configuration, the SSLi0 and SSLi1 output of this MCU (master) is not used. The SSL input of the RSPI slave is fixed to the "L" level, and the RSPI slave is always maintained in a select state. In the transfer format corresponding to the case where the CPHA bit in the RSPIi command register (SPiCMD) 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 SSL output of this MCU should be connected to the SSL input of the slave device.

This MCU (master) always drives the RSPCKi and MOSIi. The RSPI slave always drives the MISOi.

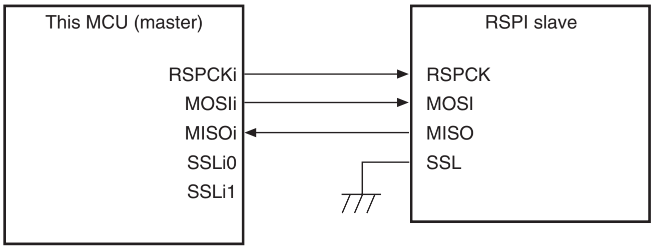


Figure 24.3 Single-Master/Single-Slave Configuration Example (This MCU = Master, i = 0)

(2) Single Master/Single Slave (with this MCU Acting as Slave, i = 0)

Figure 24.4 shows a single-master/single-slave RSPI system configuration example when this MCU is used as a slave. When this MCU is to operate as a slave, the SSLi0 pin is used as SSL input. The RSPI master always drives the RSPCK and MOSI. This MCU (slave) always drives the MISOi. When SSLi0 is at the inactive level, the pin state is Hi-Z.

In the single-slave configuration in which the CPHA bit in the RSPIi command register (SPiCMD) is set to "1", the SSLi0 input of this MCU (slave) is fixed to the "L" level, this MCU (slave) is always maintained in a selected state, and in this manner it is possible to execute serial transfer (figure 24.5).

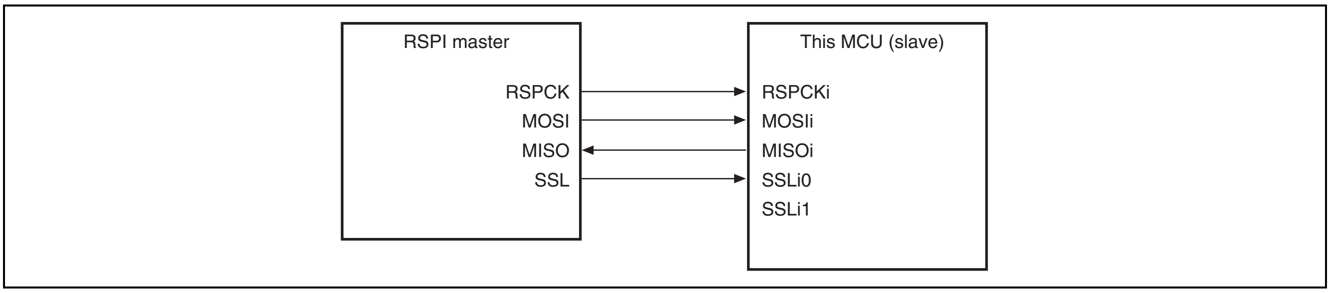


Figure 24.4 Single-Master/Single-Slave Configuration Example (This MCU = Slave, i = 0)

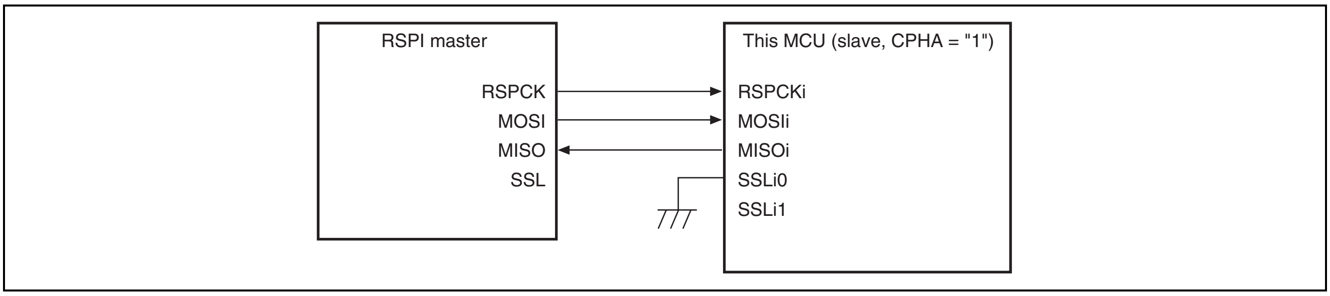


Figure 24.5 Single-Master/Single-Slave Configuration Example (This MCU = Slave, CPHA = "1", i = 0)

(3) Single Master/Multi-Slave (with this MCU Acting as Master, i = 0)

Figure 24.6 shows a single-master/multi-slave RSPI system configuration example when this MCU is used as a master. In the example of figure 24.6, the RSPI system is comprised of an this MCU (master) and two slaves (RSPI slave 0 and RSPI slave 1).

The RSPCKi and MOSIi outputs of this MCU (master) are connected to the RSPCK and MOSI inputs of RSPI slave 0 and RSPI slave 1. The MISO outputs of RSPI slave 0 and RSPI slave 1 are all connected to the MISOi input of this MCU (master). SSLi0 and SSLi1 outputs of this MCU (master) are connected to the SSL inputs of RSPI slave 0 and RSPI slave 1, respectively.

This MCU (master) always drives RSPCKi, MOSIi, SSLi0, and SSLi1. Of the RSPI slave 0 and RSPI slave 1, the slave that receives "L" level input into the SSL input drives MISO.

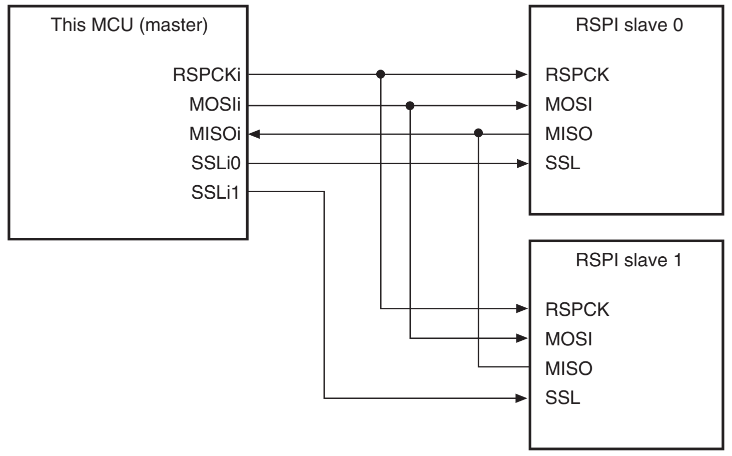


Figure 24.6 Single-Master/Multi-Slave Configuration Example (This MCU = Master, i = 0)

(4) Single Master/Multi-Slave (with this MCU Acting as Slave, i = 0)

Figure 24.7 shows a single-master/multi-slave RSPI system configuration example when this MCU is used as a slave. In the example of figure 24.7, the RSPI system is comprised of an RSPI master and two this MCUs (slave X and slave Y).

The RSPCK and MOSI outputs of the RSPI master are connected to the RSPCKi and MOSIi inputs of this MCUs (slave X and slave Y). The MISOi outputs of this MCUs (slave X and slave Y) are all connected to the MISO input of the RSPI master. The SSLX and SSLY outputs of the RSPI master are connected to the SSLi0 inputs of this MCUs (slave X and slave Y), respectively.

The RSPI master always drives RSPCK, MOSI, SSLX, and SSLY. Of this MCUs (slave X and slave Y), the slave that receives "L" level input into the SSLi0 input drives MISO.

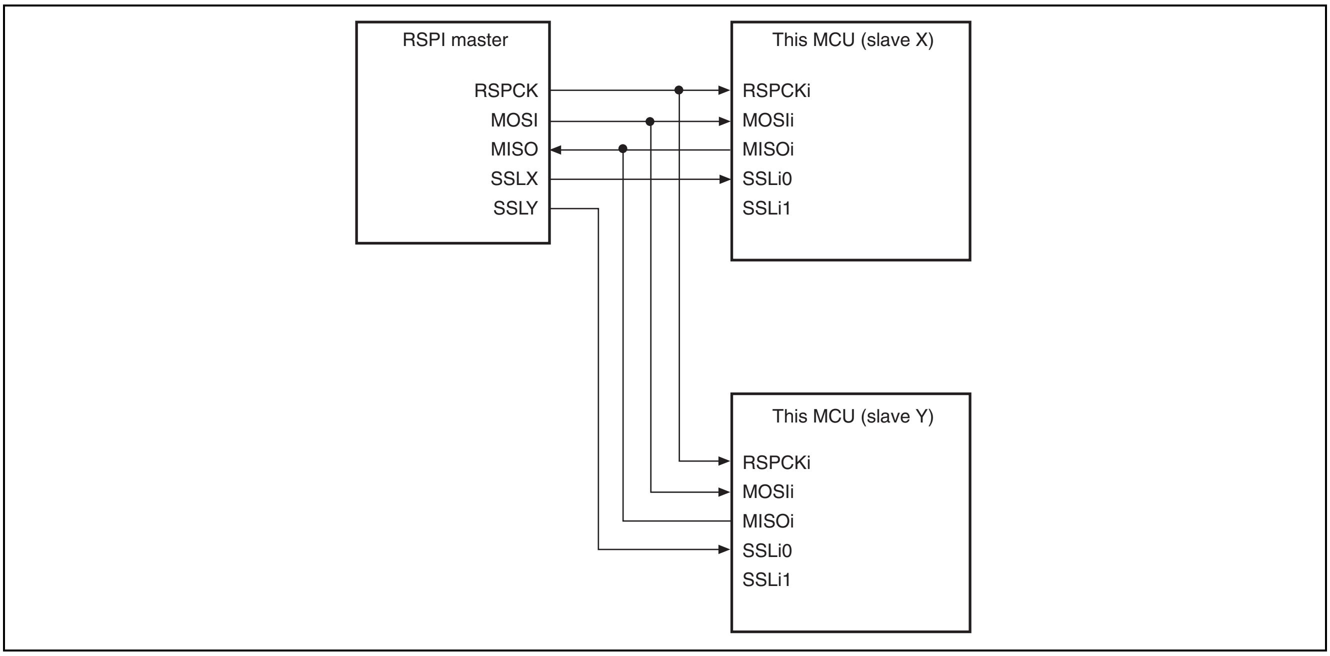


Figure 24.7 Single-Master/Multi-Slave Configuration Example (This MCU = Slave, i = 0)

(5) Multi-Master/Multi-Slave (with this MCU Acting as Master, i = 0)

Figure 24.8 shows a multi-master/multi-slave RSPI system configuration example when this MCU is used as a master. In the example of figure 24.8, the RSPI system is comprised of two this MCUs (master X, master Y) and RSPI slave 1.

The RSPCKi and MOSIi outputs of this MCU (master X, master Y) are connected to the RSPCK and MOSI inputs of RSPI slave 1. The MISO outputs of RSPI slave 1 are connected to the MISOi inputs of this MCU (master X, master Y). Any generic port Y output from this MCU (master X) is connected to the SSLi0 input of this MCU (master Y). Any generic port X output of this MCU (master Y) is connected to the SSLi0 input of this MCU (master X). The SSLi1 output of this MCU (master X, master Y) are connected to the SSL inputs of the RSPI slave 1.

This MCU drives RSPCKi, MOSIi, SSLi1, and SSLi2 when the SSLi0 input level is "H". When the SSLi0 input level is "L", this MCU detects a mode fault error, sets RSPCKi, MOSIi, SSLi1, and SSLi2 to Hi-Z, and releases the RSPI bus right to the other master.

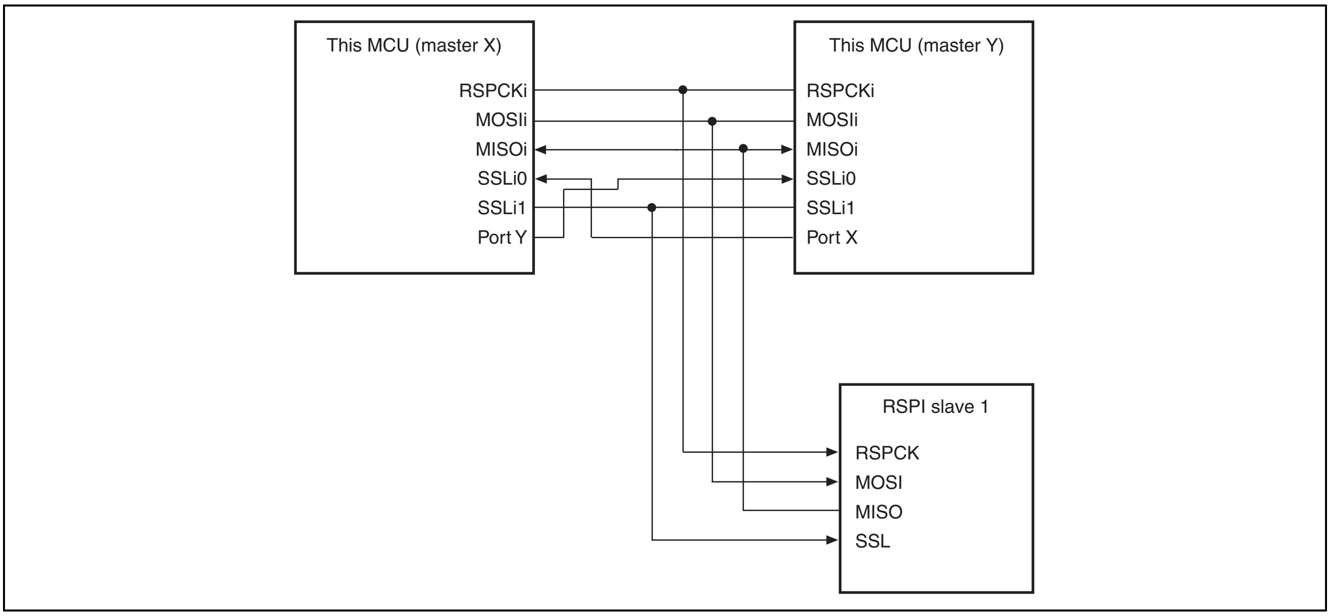


Figure 24.8 Multi-Master/Multi-Slave Configuration Example (This MCU = Master, i = 0)

(6) Master (Clock-Synchronous Operation)/Slave (Clock-Synchronous Operation) (with this MCU Acting as Master, i = 0)

Figure 24.9 shows a master (clock-synchronous operation)/slave (clock-synchronous operation) RSPI system configuration example when this MCU is used as a master. In this master (clock-synchronous operation)/slave (clock-synchronous operation) configuration, this MCU (master) does not use SSLi0 and SSLi1. This MCU (master) drives RSPCKi and MOSIi constantly. The RSPI slave drives MISO constantly.

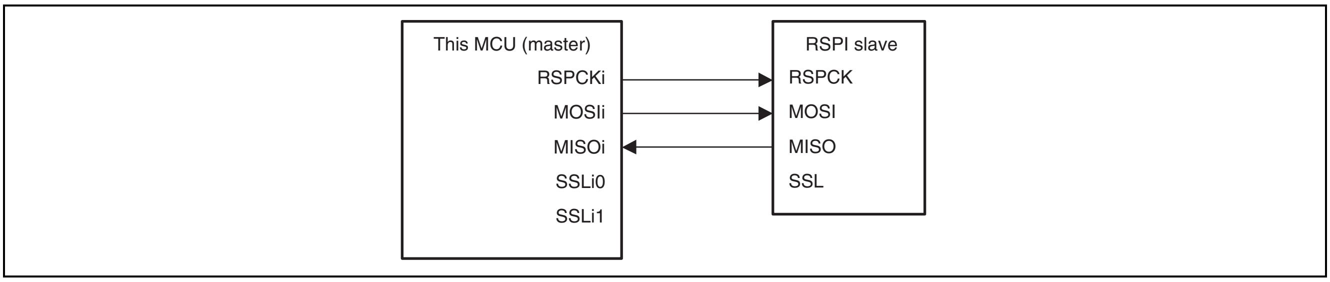


Figure 24.9 Master (Clock-Synchronous Operation)/Slave (Clock-Synchronous Operation) Configuration Example (This MCU = Master, i = 0)

(7) Master (Clock-Synchronous Operation)/Slave (Clock-Synchronous Operation) (with this MCU Acting as Slave, i = 0)

Figure 24.10 shows a master (clock-synchronous operation)/slave (clock-synchronous operation) RSPI system configuration example when this MCU is used as a slave. When this MCU is used as a slave (clock-synchronous operation), this MCU (slave) drives MISOi constantly, and the RSPI master drives RSPCK and MOSI constantly. This MCU (slave) can perform serial transfers only in a single-slave configuration in which the CPHA bit in the RSPIi command register (SPiCMD) is set to "1".

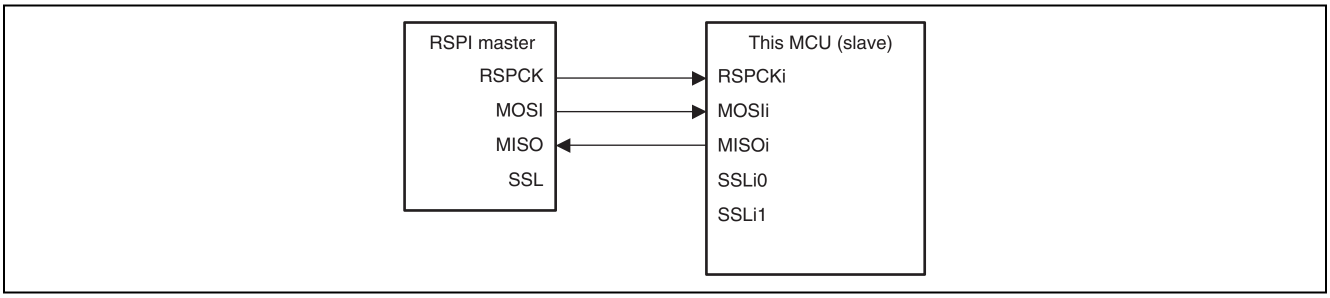


Figure 24.10 Master (Clock-Synchronous Operation)/Slave (Clock-Synchronous Operation) Configuration Example (This MCU = Slave, CPHA = "1", i = 0)

24.4.4 Transfer Format

(1) CPHA = "0"

Figure 24.11 shows an example transfer format for the serial transfer of 8-bit data when the CPHA bit in the RSPIi command register (SPiCMD) is "0". However, clock-synchronous operation (SPMS bit in RSPIi control register (SPiCR) set to "1") cannot be guaranteed with the CPHA bit is cleared to "0". In figure 24.11, RSPCK (CPOL = "0") indicates the RSPCK signal waveform when the CPOL bit in the SPiCMD register is "0"; RSPCK (CPOL = "1") indicates the RSPCK signal waveform when the CPOL bit is "1". The sampling timing represents the timing at which the RSPIi fetches serial transfer data into the shift register. The input/output directions of the signals depend on the RSPI settings. For details, see section 24.4.2, Controlling RSPIi Pins.

When the CPHA bit is cleared to "0", the output of valid data to the MOSIi signal and driving of valid data to the MISOi signal commences when the SSLi signal is asserted. The first RSPCKi signal change timing that occurs after the SSLi signal assertion becomes the first transfer data fetching timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for the MOSIi and MISOi signals is always 1/2 RSPCK cycle after the transfer data fetch timing. The settings in the CPOL bit do not affect the RSPCKi signal operation timing; they only affect the signal polarity.

t1 denotes a period from an SSLi signal assertion to RSPCKi oscillation (RSPCK delay). t2 denotes a period from the cessation of RSPCKi oscillation to an SSLi signal negation (SSL negation delay). t3 denotes a period in which SSLi 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 MCU is in master mode, see section 24.4.9 (1), Master Mode Operation.

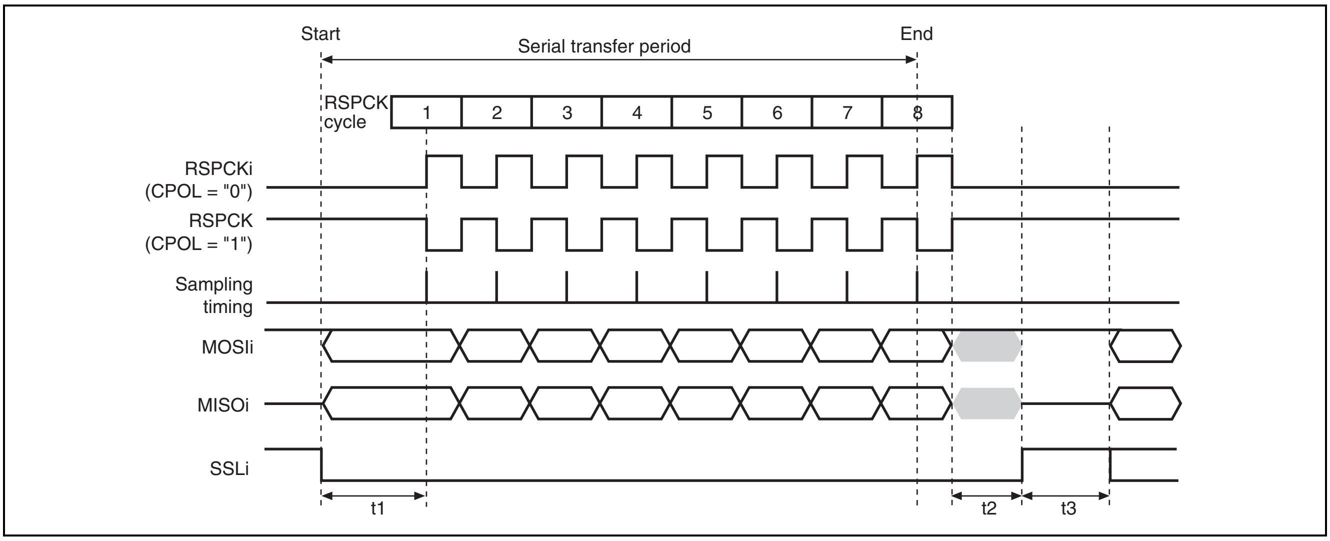


Figure 24.11 RSPI Transfer Format (CPHA = "0")

(2) CPHA = "1"

Figure 24.12 shows an example transfer format for the serial transfer of 8-bit data when the CPHA bit in the RSPIi command register (SPiCMD) is "1". However, the SSLi signal is not used when the SPMS bit in the RSPIi control register (SPiCR) is set to "1", and communication is performed using the RSPCKi, MOSIi, and MISOi signals only. In figure 24.12, RSPCK (CPOL = "0") indicates the RSPCK signal waveform when the CPOL bit in the SPiCMD register is "0"; RSPCK (CPOL = "1") indicates the RSPCK 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 input/output directions of the signals depend on the RSPI modes (master or slave). For details, see section 24.4.2, Controlling RSPIi Pins.

When the CPHA bit is "1", the driving of invalid data to the MISOi signal commences at an SSLi signal assertion timing. The driving of valid data to the MOSIi and MISOi signals commences at the first RSPCKi signal change timing that occurs after the SSLi signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycle after the data update timing. The settings in the CPOL bit do not affect the RSPCKi signal operation timing; they only affect 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 MCU is in master mode, see section 24.4.9 (1), Master Mode Operation.

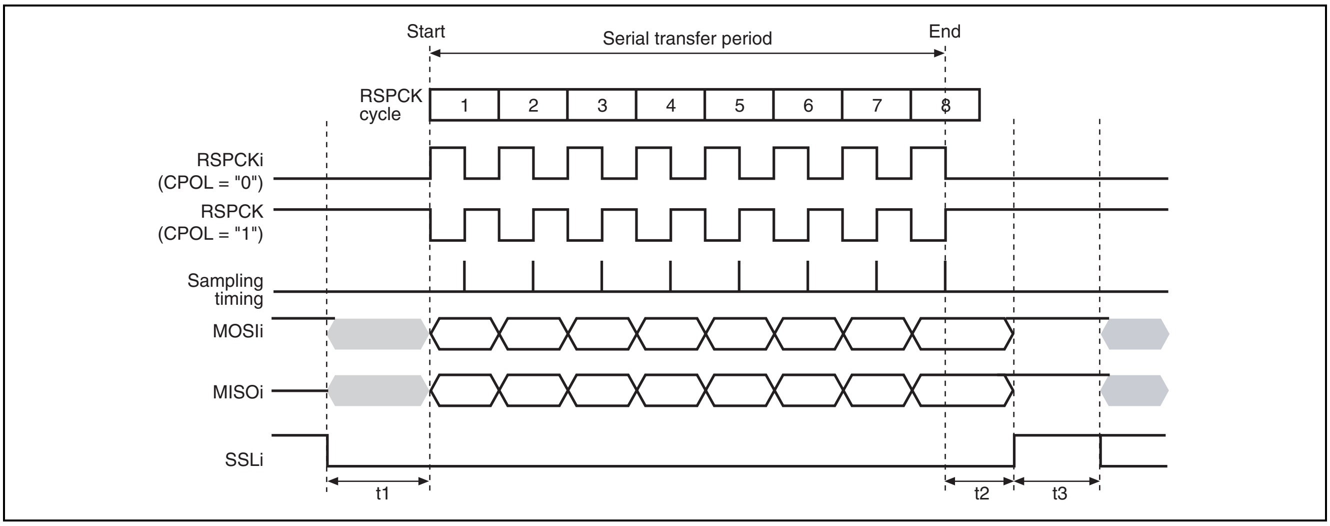


Figure 24.12 RSPI Transfer Format (CPHA = "1")

24.4.5 Data Format

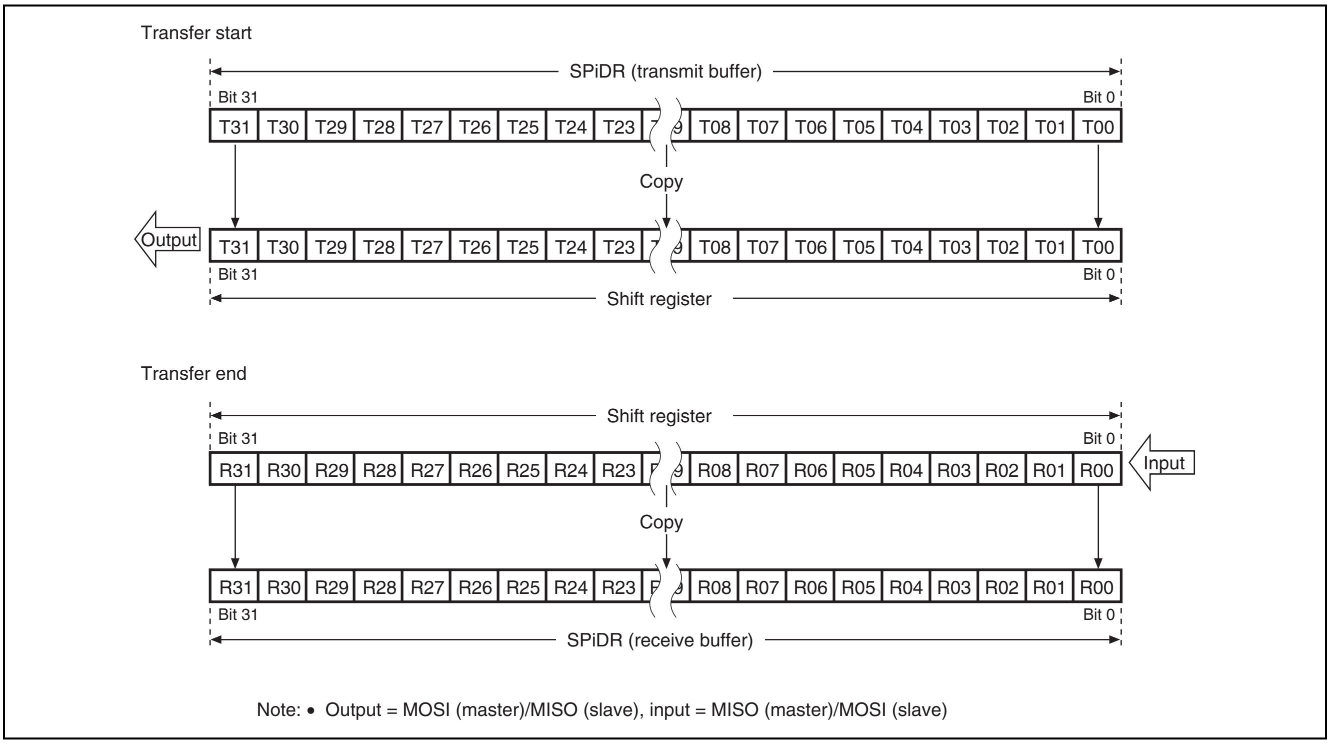
The RSPIi's data format depends on the settings in the RSPIi command register (SPiCMD). Irrespective of MSB/LSB first, the RSPIi treats the range from the LSB of the RSPIi data register (SPiDR) to the assigned data length as transfer data. Bit 16 of the SPiDR register is the LSB when the SPiDCR.SPLW bit is cleared to "0", and bit 0 of the SPiDR register is the LSB when the SPiDCR.SPLW bit is set to "1".

(1) MSB First Transfer (32-Bit Data)

Figure 24.13 shows the operation of the RSPIi data register (SPiDR) and the shift register when the RSPIi performs a 32-bit data length MSB-first data transfer.

The CPU or the DMAC writes T31 to T00 to the transmit buffer of the SPiDR register. If the SPTEF bit in the RSPIi status register (SPiSR) is "0" and the shift register is empty, the RSPIi copies the data in the transmit buffer of SPiDR to the shift register, and fully populates the shift register. When serial transfer starts, the RSPIi outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to R00 is stored in the shift register. In this state, the RSPIi copies the data from the shift register to the receive buffer of the SPiDR register, and empties the shift register.

If another serial transfer is started before the CPU or the DMAC writes to the transmit buffer of the SPiDR register, received data R31 to R00 is shifted out from the shift register.


Transfer start
SPiDR (transmit buffer)
Bit 31
Bit 0
T31 T30 T29 T28 T27 T26 T25 T24 T23 T08 T07 T06 T05 T04 T03 T02 T01 T00
Copy
Output
T31 T30 T29 T28 T27 T26 T25 T24 T23 T08 T07 T06 T05 T04 T03 T02 T01 T00
Bit 31
Bit 0
Shift register
Transfer end
Shift register
Bit 31
Bit 0
R31 R30 R29 R28 R27 R26 R25 R24 R23 R08 R07 R06 R05 R04 R03 R02 R01 R00
Input
Copy
R31 R30 R29 R28 R27 R26 R25 R24 R23 R08 R07 R06 R05 R04 R03 R02 R01 R00
Bit 31
Bit 0
SPiDR (receive buffer)

Note: • Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

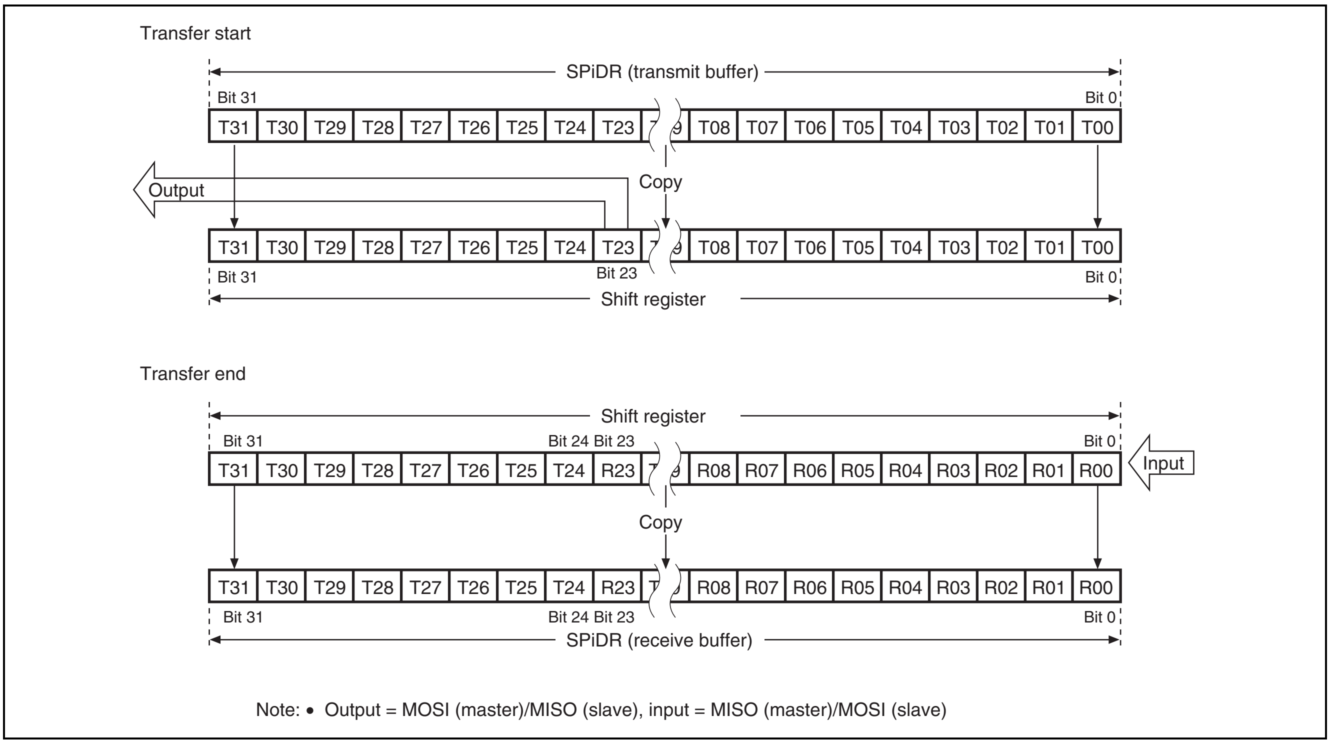
Figure 24.13 MSB First Transfer (32-Bit Data)

(2) MSB First Transfer (24-Bit Data)

As an example of MSB-first data transfer with a data length other than 32 bits, figure 24.14 shows the operation of the RSPIi data register (SPiDR) and the shift register when RSPIi performs a 24-bit data transfer.

The CPU or the DMAC writes T31 to T00 to the transmit buffer of the SPiDR register. If the SPTEF bit in the RSPIi status register (SPiSR) is "0" and the shift register is empty, the RSPIi copies the data in the transmit buffer of the SPiDR register to the shift register, and fully populates the shift register. When serial transfer starts, the RSPIi outputs data from bit 23 of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 24 bits has passed, received data R23 to R00 is stored in bits 23 to 0 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 24 in the shift register. In this state, the RSPIi copies the data from the shift register to the receive buffer of the SPiDR register, and empties the shift register.

If another serial transfer is started before the CPU or the DMAC writes to the transmit buffer of the SPiDR register, received data R23 to R00 is shifted out from the shift register.


Transfer start
SPiDR (transmit buffer)
Bit 31
Bit 0
T31 T30 T29 T28 T27 T26 T25 T24 T23 T08 T07 T06 T05 T04 T03 T02 T01 T00
Output
Copy
T31 T30 T29 T28 T27 T26 T25 T24 T23 T08 T07 T06 T05 T04 T03 T02 T01 T00
Bit 31
Bit 23
Bit 0
Shift register
Transfer end
Shift register
Bit 31
Bit 24 Bit 23
Bit 0
T31 T30 T29 T28 T27 T26 T25 T24 R23 R08 R07 R06 R05 R04 R03 R02 R01 R00
Input
Copy
T31 T30 T29 T28 T27 T26 T25 T24 R23 R08 R07 R06 R05 R04 R03 R02 R01 R00
Bit 31
Bit 24 Bit 23
Bit 0
SPiDR (receive buffer)

Note: • Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

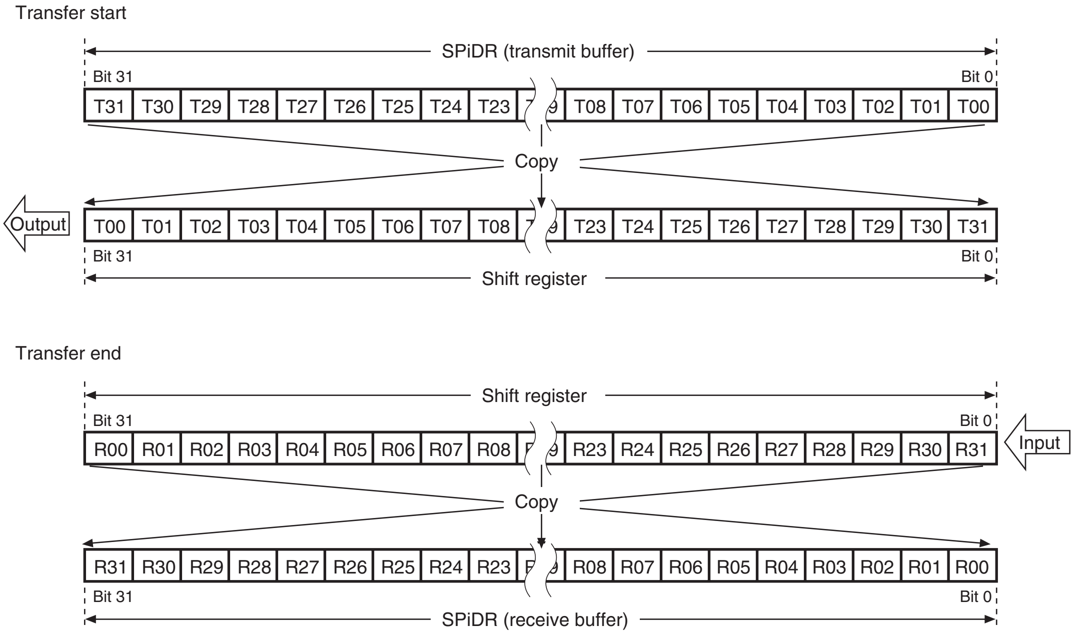
Figure 24.14 MSB First Transfer (24-Bit Data)

(3) LSB First Transfer (32-Bit Data)

Figure 24.15 shows the operation of the RSPIi data register (SPiDR) and the shift register when the RSPIi performs a 32-bit data length LSB-first data transfer.

The CPU or the DMAC writes T31 to T00 to the transmit buffer of the SPiDR register. If the SPTEF bit in the RSPIi status register (SPiSR) is 0 and the shift register is empty, the RSPIi reverses the order of the bits of the data in the transmit buffer of the SPiDR register, copies it to the shift register, and fully populates the shift register. When serial transfer starts, the RSPIi outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to R31 is stored in the shift register. In this state, the RSPIi copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of the SPiDR register, and empties the shift register.

If another serial transfer is started before the CPU or the DMAC writes to the transmit buffer of the SPiDR register, received data R00 to R31 is shifted out from the shift register.


Transfer start
SPiDR (transmit buffer)
Bit 31
Bit 0
T31 T30 T29 T28 T27 T26 T25 T24 T23 T08 T07 T06 T05 T04 T03 T02 T01 T00
Copy
Output
T00 T01 T02 T03 T04 T05 T06 T07 T08 T23 T24 T25 T26 T27 T28 T29 T30 T31
Bit 31
Bit 0
Shift register
Transfer end
Shift register
Bit 31
Bit 0
R00 R01 R02 R03 R04 R05 R06 R07 R08 R23 R24 R25 R26 R27 R28 R29 R30 R31
Input
Copy
R31 R30 R29 R28 R27 R26 R25 R24 R23 R08 R07 R06 R05 R04 R03 R02 R01 R00
Bit 31
Bit 0
SPiDR (receive buffer)

Note: • Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

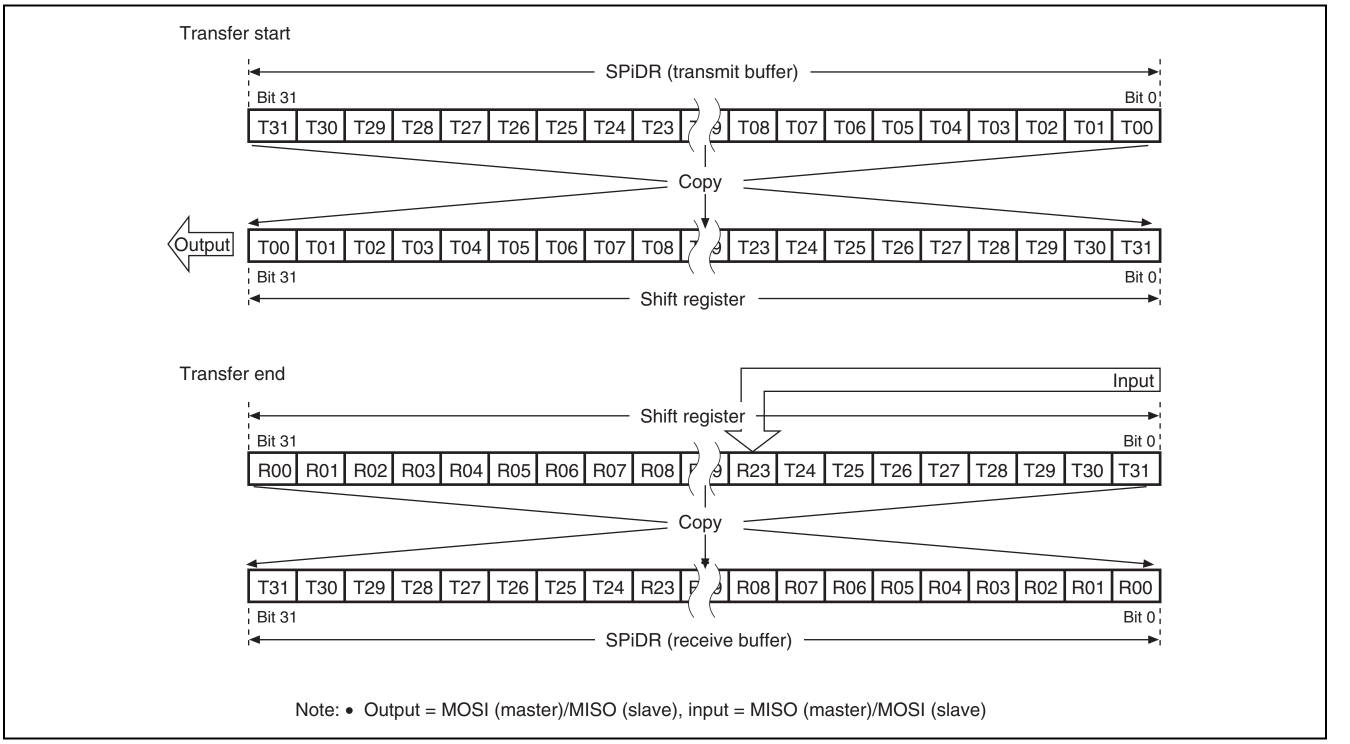
Figure 24.15 LSB First Transfer (32-Bit Data)

(4) LSB First Transfer (24-Bit Data)

As an example of LSB-first data transfer with a data length other than 32 bits, figure 24.16 shows the operation of the RSPIi data register (SPiDR) and the shift register when RSPIi performs a 24-bit data transfer.

The CPU or the DMAC writes T31 to T00 to the transmit buffer of the SPiDR register. If the SPTEF bit in the RSPIi status register (SPiSR) is "0" and the shift register is empty, the RSPI reverses the order of the bits of the data in the transmit buffer of the SPiDR register, copies it to the shift register, and fully populates the shift register. When serial transfer starts, the RSPIi outputs data from the MSB (bit 31) of the shift register, and shifts in the data from bit 8 of the shift register. When the RSPCK cycle required for the serial transfer of 24 bits has passed, received data R00 to R23 is stored in bits 31 to 8 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 7 to 0 of the shift register. In this state, the RSPIi copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of the SPiDR register, and empties the shift register.

If another serial transfer is started before the CPU or the DMAC writes to the transmit buffer of the SPiDR register, received data R00 to R23 is shifted out from the shift register.


Transfer start
SPiDR (transmit buffer)
Bit 31
Bit 0
T31 T30 T29 T28 T27 T26 T25 T24 T23 T08 T07 T06 T05 T04 T03 T02 T01 T00
Copy
Output
T00 T01 T02 T03 T04 T05 T06 T07 T08 T23 T24 T25 T26 T27 T28 T29 T30 T31
Bit 31
Bit 0
Shift register
Transfer end
Input
Shift register
Bit 31
Bit 0
R00 R01 R02 R03 R04 R05 R06 R07 R08 R23 T24 T25 T26 T27 T28 T29 T30 T31
Copy
T31 T30 T29 T28 T27 T26 T25 T24 R23 R08 R07 R06 R05 R04 R03 R02 R01 R00
Bit 31
Bit 0
SPiDR (receive buffer)

Note: • Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 24.16 LSB First Transfer (24-Bit Data)

24.4.6 Transmission Buffer Empty/Receive Buffer Full Flags

Figure 24.17 shows an example of operation of the RSPI transmit buffer empty flag (SPTEF) and the RSPI receive buffer full flag in the RSPIi status register (SPiSR). The SPiDR access depicted in figure 24.17 indicates the condition of access from the CPU or the DMAC to the RSPIi data register (SPiDR), where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example shown in figure 24.17, the RSPI performs an 8-bit serial transfer when the SPFC bits in the RSPIi data control register (SPiDCR) are set to "00" and the CPHA and CPOL bits in the RSPIi command register (SPiCMD) are set to "1" and "0", respectively. The numbers given under the RSPCKi waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

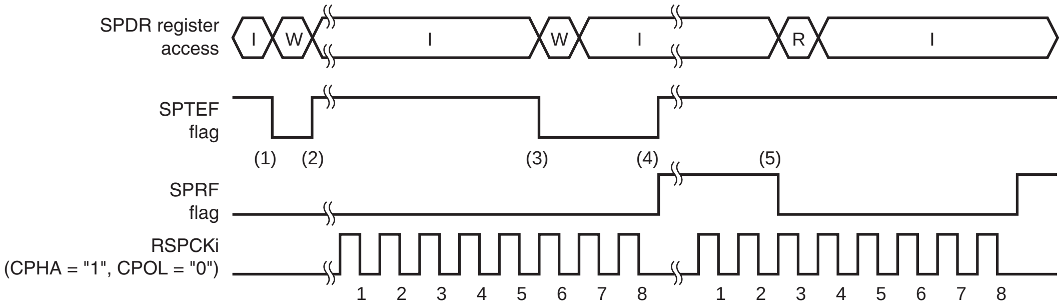


Figure 24.17 SPTEF and SPRF Bit Operation Example

The operation of the flags at timings shown in steps (1) to (5) in the figure is described below.

1. When the DMAC writes transmit data to the SPiDR register while the transmit buffer of the SPiDR register is empty, the RSPIi sets the SPTEF bit to "0", and writes data to the transmit buffer, with no change in the SPRF flag.
2. If the shift register is empty, the RSPIi sets the SPTEF bit to "1", and copies the data in the transmit buffer to the shift register, with no change in the SPRF flag. How a serial transfer is started depends on the mode of the RSPI. For details, see section 24.4.9, SPI Operation, and section 24.4.10, Clock Synchronous Operation.
3. When the CPU or the DMAC writes transmit data to SPiDR with the transmit buffer of the SPiDR register being empty, the RSPIi sets the SPTEF bit to 0, and writes data to the transmit buffer, while the SPRF flag remains unchanged. Because the data being transferred serially is stored in the shift register, the RSPIi does not copy the data in the transmit buffer to the shift register.
4. When the serial transfer ends with the receive buffer of the SPiDR register being empty, the RSPIi sets the SPRF bit to 1, and copies the receive data in the shift register to the receive buffer. Because the shift register becomes empty upon completion of serial transfer, if the transmit buffer was full before the serial transfer ended, the RSPIi sets the SPTEF bit to "1", and copies the data in the transmit buffer to the shift register. Even when received data is not copied from the shift register to the receive buffer in an overrun error status, upon completion of the serial transfer the RSPIi determines that the shift register is empty, and as a result data transfer from the transmit buffer to the shift register is enabled.
5. When the CPU or the DMAC reads SPiDR with the receive buffer being full, the RSPIi sets the SPRF bit to "0", and sends the data in the receive buffer to the internal bus.

If the CPU or the DMAC writes to the SPiDR register when the SPTEF bit is "0", the RSPIi does not update the data in the transmit buffer. When writing to the SPiDR register, make sure that the SPTEF bit is "1". That the SPTEF bit is "1" can be checked by reading the SPiSR register or by using an RSPI transmit interrupt. To use an RSPIi transmit interrupt, set the SPTIE bit in the SPiCR register to "1".

If the RSPI is disabled (the SPE bit in the SPiCR register being "0"), the SPTEF bit is initialized to "1". For this reason, setting the SPTIE bit to 1 when the RSPIi is disabled generates an RSPIi transmit interrupt.

When serial transfer ends with the SPRF bit being "1", the RSPIi does not copy data from the shift register to the receive buffer, and detects an overrun error (see section 24.4.7, Error Detection). To prevent a receive data overrun error, set the SPRF bit to "0" before the serial transfer ends. That the SPRF bit is "1" can be checked by either reading the SPiSR register or by using an RSPIi receive interrupt. To use an RSPI receive interrupt, set the SPRIE bit in the SPiCR register to "1".

24.4.7 Error Detection

In the normal RSPIi serial transfer, the data written from the RSPIi data register (SPiDR) to the transmit buffer by either the CPU or the DMAC is serially transmitted, and either the CPU or the DMAC can read the serially received data from the receive buffer of the SPiDR register. If access is made to the SPiDR register by either the CPU or the DMAC, depending on the status of the transmit buffer/receive buffer or the status of the RSPIi 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 RSPIi detects the event as an overrun error or a mode fault error. Table 24.8 shows the relationship between non-normal transfer operations and the RSPIi's error detection function.

Table 24.8 Relationship between Non-Normal Transfer Operations and RSPIi Error Detection Function

	Occurrence Condition	RSPI Operation	Error Detection
A	Either the CPU or the DMAC writes to the SPiDR register when the transmit buffer is full.	Retains the contents of the transmit buffer. Missing write data.	None
B	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 serially transmitted.	None
C	Either the CPU or the DMAC reads from the SPiDR register when the receive buffer is empty.	Previously received serial data is output to the CPU or the DMAC.	None
D	Serial transfer terminates when the receive buffer is full.	Retains the contents of the receive buffer. Missing serial receive data.	Overrun error
E	The SSLi0 input signal is asserted when the serial transfer is idle in multi-master mode.	Driving of the RSPCKi, MOSIi, and SSL01 output signals stopped. RSPI disabled.	Mode fault error
F	The SSLi0 input signal is asserted during serial transfer in multi-master mode.	Serial transfer suspended. Missing send/receive data. Driving of the RSPCKi, MOSIi, and SSL01 output signals stopped. RSPI disabled.	Mode fault error
G	The SSLi0 input signal is negated during serial transfer in slave mode.	Serial transfer suspended. Missing send/receive data. Driving of the MISOi output signal stopped. RSPI disabled.	Mode fault error

On operation A shown in table 24.8, the RSPIi does not detect an error. To prevent data omission during the writing to the SPiDR register by the CPU or the DMAC, write operations to the SPiDR register should be executed when the SPTEF bit in the RSPIi status register (SPiSR) is "1".

Likewise, the RSPIi does not detect an error on operation B. In a serial transfer that was started before the shift register was updated, the RSPIi sends the data that was received in the previous serial transfer, and does not treat the operation indicated in B as an error. Notice that the received data from the previous serial transfer is retained in the receive buffer of the SPiDR register, and thus it can be correctly read by the CPU or the DMAC (if the SPiDR register is not read before the end of the serial transfer, an overrun error may result).

Similarly, the RSPIi does not detect an error on operation C. To prevent the CPU or the DMAC from reading extraneous data, the SPiDR register read operation should be executed when the SPRF bit in the SPiSR register is "1".

An overrun error shown in D is described in section 24.4.7 (1), Overrun Error. A mode fault error shown in E to G is described in section 24.4.7 (2), Mode Fault Error. On operations of the SPTEF and SPRF bits in the SPiSR register, see section 24.4.6, Transmission Buffer Empty/Receive Buffer Full Flags.

(1) Overrun Error

If serial transfer ends when the receive buffer of the RSPIi data register (SPiDR) is full, the RSPIi detects an overrun error, and sets the OVRF bit in the SPiSR register to "1". When the OVRF bit is "1", the RSPIi does not copy data from the shift register to the receive buffer so that the data prior to the occurrence of the error is retained in the receive buffer. To reset the OVRF bit in the SPiSR register to "0", either execute hardware reset, or write a "0" to the OVRF bit after the CPU has read the SPiSR register with the OVRF bit set to "1".

Figure 24.18 shows an example of operation of the SPRF and OVRF bits in the SPiSR register. The SPiSR register access depicted in figure 24.18 indicates the condition of access from the CPU to the SPiSR register, and from the DMAC to SPiDR, respectively, where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of figure 24.18, the RSPI performs an 8-bit serial transfer in which the CPHA bit in the RSPIi command register (SPiCMD) is "1", and CPOL is "0". The numbers given under the RSPCKi waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

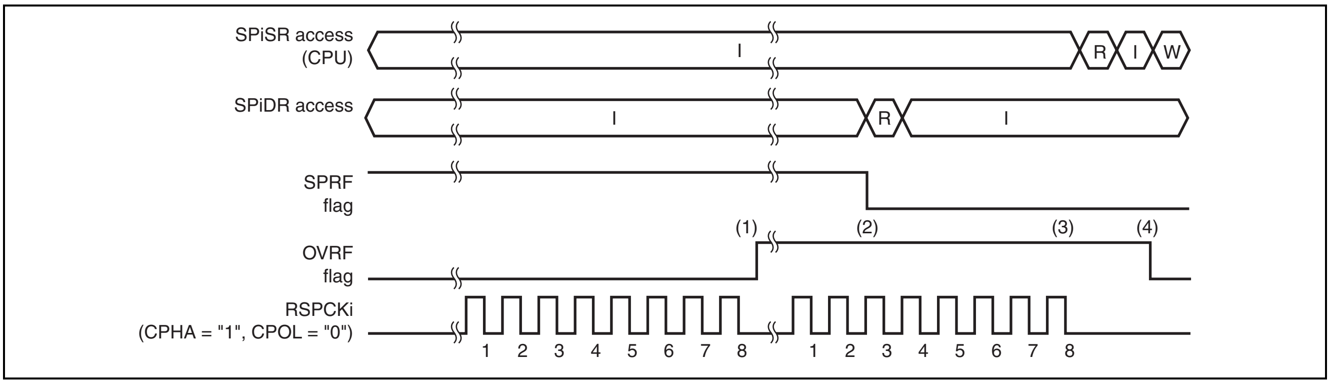


Figure 24.18 SPRF and OVRF Bit Operation Example

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 SPRF bit being "1" (receive buffer full), the RSPIi detects an overrun error, and sets the OVRF bit to "1". The RSPIi does not copy the data in the shift register to the receive buffer. In master mode, the RSPIi copies the value of the pointer to the RSPIi command register (SPiCMD) to the SPECM bit in the RSPIi sequence status register (SPiSSR).
2. When the CPU or the DMAC reads the SPiDR register, the RSPIi sets the SPRF bit to "0", and outputs the data in the receive buffer to an internal bus. The receive buffer becoming empty does not clear the OVRF bit.
3. If the serial transfer terminates with the OVRF bit being "1" (an overrun error), the RSPIi keeps the SPRF bit at "0" and does not update it. Likewise, the RSPIi does not copy the data in the shift register to the receive buffer. When in master mode, the RSPIi does not update SPECM bit of the SPiSSR register. If, in an overrun error state, the RSPIi does not copy the received data from the shift register to the receive buffer, upon termination of the serial transfer, the RSPIi determines that the shift register is empty; in this manner, data transfer is enabled from the transmit buffer to the shift register.

4. If the CPU writes a "0" to the OVRF bit after reading the SPiSR register when the OVRF bit is "1", the RSPIi clears the OVRF bit.

The occurrence of an overrun can be checked either by reading the SPiSR register or by using an RSPIi error interrupt and reading the SPiSR register. To use an RSPIi error interrupt, set the SPEIE bit in the RSPIi control register (SPiCR) to "1". When executing a serial transfer without using an RSPIi error interrupt, measures should be taken to ensure the early detection of overrun errors, such as reading the SPiSR register immediately after the SPiDR register is read. When the RSPIi is run in master mode, the pointer value to the SPiCMD register can be checked by reading the SPECM bit of the SPiSSR register.

If an overrun error occurs and the OVRF bit is set to "1", normal reception operations cannot be performed until such time as the OVRF bit is cleared. The OVRF bit is cleared to "0" under the following conditions:

- After reading the SPiSR register in a condition in which the OVRF bit is set to "1", the CPU writes a "0" to the OVRF bit.
- Hardware reset

(2) Mode Fault Error

The RSPIi operates in multi-master mode when the MSTR bit in the RSPIi control register (SPiCR) is "1", the SPMS bit is "0", and the MODFEN bit is "1". If the active level is input with respect to the SSLi0 input signal of the RSPIi in multi-master mode, the RSPIi detects a mode fault error irrespective of the status of the serial transfer, and sets the MODF bit in the RSPIi status register (SPiSR) to "1". Upon detecting the mode fault error, the RSPIi copies the value of the pointer to the RSPIi command register (SPiCMD) to the SPECM bit in the RSPIi sequence status register (SPiSSR). The active level of the SSL0 signal is determined by the SSL0P bit in the RSPIi slave select polarity register (SPiSSLP).

When the MSTR bit is cleared to "0", RSPIi operates in slave mode. While in slave mode, with the MODFEN bit set to "1" and the SPMS bit cleared to "0", RSPIi detects a mode fault error when the SSL0 input signal is negated during the serial transfer period (from the time the driving of valid data is started to the time the final unit of valid data is fetched).

Upon detecting a mode fault error, RSPIi stops driving the output signals and clears the SPE bit in the SPiCR register to "0". When the SPE bit is cleared to "0", the RSPI function is disabled. (See section 24.4.8, Initializing RSPI.) In a multi-master configuration, it is possible to release the master rights by using a mode fault error to stop the driving of output signals and the RSPI function.

The occurrence of a mode fault error can be checked either by reading the SPiSR register or by using an RSPIi error interrupt and reading the SPiSR register. To use an RSPIi error interrupt, set the SPEIE bit in the RSPIi control register (SPiCR) to "1". To detect a mode fault error without using an RSPIi error interrupt, it is necessary to poll the SPiSR register. When using the RSPI in master mode, one can read the SPECM bit in the SPiSSR register to verify the value of the pointer to the SPiCMD register when an error occurs.

When the MODF bit is "1", the RSPIi ignores the writing of the value "1" to the SPE bit by the CPU. To enable the RSPIi function after the detection of a mode fault error, the MODF bit must be set to "0". The MODF bit is cleared to "0" under the following conditions:

- After reading the SPSR register in a condition where the MODF bit has turned "1", the CPU writes a "0" to the MODF bit.
- Hardware reset

24.4.8 Initializing RSPI

If the CPU writes a "0" to the SPE bit in the RSPIi control register (SPiCR) or the RSPIi clears the SPE bit to "0" because of the detection of a mode fault error, the RSPIi disables the RSPI function, and initializes a part of the module function. During a hardware reset, the RSPIi initializes all of the module function. An explanation follows of initialization by the clearing of the SPE bit and initialization by hardware reset.

(1) Initialization by Clearing SPE Bit

When the SPE bit in the SPiCR register is cleared, the RSPIi performs the following initialization:

- Suspending any serial transfer that is being executed
- Stopping the driving of output signals only in slave mode (Hi-Z)
- Initializing the internal state of the RSPIi
- Initializing the SPTEF bit in the RSPIi status register (SPiSR)

Initialization by the clearing of the SPE bit does not initialize the control bits of the RSPIi. For this reason, the RSPI can be started in the same transfer mode as prior to the initialization if the CPU resets the value "1" to the SPE bit.

The SPRF, OVRF, and MODF bits in the SPiSR register are not initialized, nor is the value of the RSPIi sequence status register (SPiSSR) initialized. For this reason, even after the RSPI is initialized, data from the receive buffer can be read in order to check the status of error occurrence during an RSPIi transfer.

The SPTEF bit in the SPiSR register is initialized to "1". Therefore, if the SPTIE bit in the SPiCR register is set to "1" after RSPIi initialization, an RSPIi transmit interrupt is generated. When the RSPIi is initialized by the CPU, in order to disable any RSPIi transmit interrupt, a "0" should be written to the SPTIE bit simultaneously with the writing of a "0" to the SPE bit. To disable any RSPIi transmit interrupt after a mode fault error is detected, use an error handling routine to write a "0" to the SPTIE bit.

(2) Hardware Reset

Initialization by a hardware reset completely initializes RSPIi by initializing all the bits used to control it, the status bits, and the data registers, in addition to the items listed in section 24.4.8 (1), Initialization by Clearing SPE Bit.

24.4.9 SPI Operation

(1) Master Mode Operation

The only difference between single-master mode operation and multi-master mode operation lies in mode fault error detection (see section 24.4.7, Error Detection). When operating in single-master mode, the RSPIi does not detect mode fault errors whereas the RSPIi running in multi-master mode does detect mode fault errors. This section explains operations that are common to single-/multi-master modes.

(a) Starting Serial Transfer

The RSPIi updates the data in the transmit buffer when the SPTEF bit in the RSPIi status register (SPiSR) is "1" and when either the CPU or the DMAC has written data to the RSPIi data register (SPiDR). When the shift register is empty in a condition where the SPTEF bit has been cleared to "0" by a write to the SPiDR register or by the CPU reading the SPTEF bit as "1" and then writing "0" to it, RSPIi copies the data in the transmit buffer to the shift register and starts a serial transfer. Upon copying transmit data to the shift register, the RSPIi 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 from the CPU.

For details on the RSPIi transfer format, see section 24.4.4, Transfer Format. The SSLi output signal polarity is dependent on the setting value of the RSPIi slave select polarity register (SPiSSLP).

(b) Terminating a Serial Transfer

Irrespective of the CPHA bit in the RSPIi command register (SPiCMD), the RSPIi terminates the serial transfer after transmitting an RSPCKi edge corresponding to the final sampling timing. If the SPRF bit in the RSPIi status register (SPiSR) is "0" and free space is available in the receive buffer, upon termination of serial transfer the RSPIi copies data from the shift register to the receive buffer of the RSPIi data register (SPiDR).

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 settings in the SPB bit in the RSPi command register (SPiCMD). The SSLi output signal polarity depends on the setting value of the RSPIi slave select polarity register (SPiSSLP). For details on the RSPIi transfer format, see section 24.4.4, Transfer Format.

(c) Sequence Control

The transfer format that is employed in master mode is determined by the RSPIi sequence control register (SPiSCR), RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3), the RSPIi bit rate register (SPiBR), the RSPIi clock delay register (SPiCKD), the RSPIi slave select negation delay register (SPiSSLND), and the RSPIi next-access delay register (SPiND).

The SPiSCR register is used to determine the sequence configuration for serial transfers that are executed by a master mode RSPIi. The following items are set in RSPI command registers SPiCMD0 to SPiCMD3: SSLi output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether the SPiCKD register is to be referenced, whether the SPiSSLND register is to be referenced, and whether the SPiND register is to be referenced. The SPiBR register holds some of the bit rate settings; the SPiCKD register, an RSPI clock delay value; the SPiSSLND register, an SSL negation delay; and the SPiND register, a next-access delay value.

According to the sequence length that is assigned to SPiSCR, the RSPIi makes up a sequence comprised of a part or all of SPiCMD0 to SPiCMD3. The RSPIi contains a pointer to the SPiCMD register that makes up the sequence. The value of this pointer can be checked by reading SPCP bit in the RSPIi sequence status register (SPiSSR). When the SPE bit in the RSPIi control register (SPiCR) is set to "1" and the RSPIi function is enabled, the RSPIi loads the pointer to the commands in the SPiCMD0 register, and incorporates the SPiCMD0 register settings into the transfer format at the beginning of serial transfer. The RSPIi 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 RSPIi sets the pointer in the SPiCMD0 register, and in this manner the sequence is executed repeatedly.

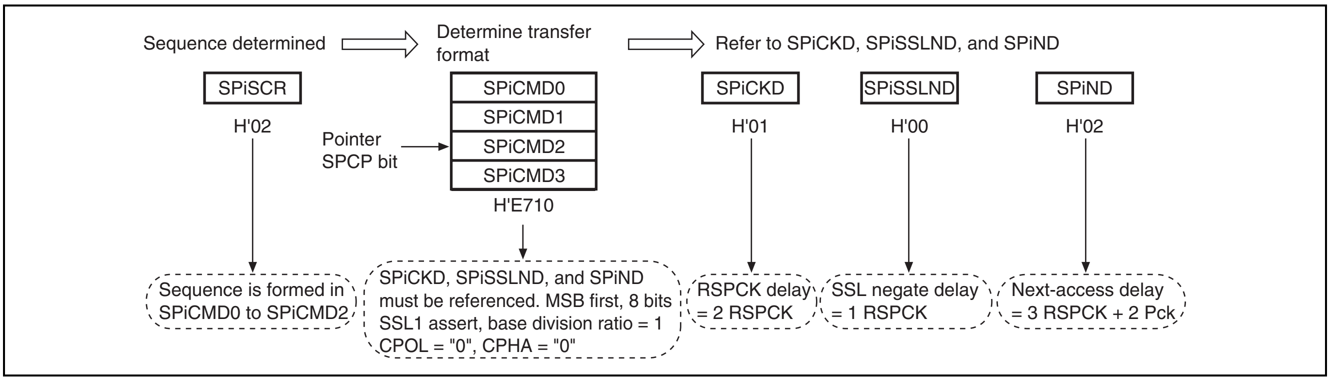

Sequence determined
Determine transfer format
Refer to SPiCKD, SPiSSLND, and SPiND
SPiSCR
H'02
SPiCMD0
SPiCMD1
SPiCMD2
SPiCMD3
H'E710
Pointer SPCP bit
SPiCKD
H'01
SPiSSLND
H'00
SPiND
H'02
Sequence is formed in SPiCMD0 to SPiCMD2
SPiCKD, SPiSSLND, and SPiND must be referenced. MSB first, 8 bits SSL1 assert, base division ratio = 1 CPOL = "0", CPHA = "0"
RSPCK delay = 2 RSPCK
SSL negate delay = 1 RSPCK
Next-access delay = 3 RSPCK + 2 Pck

Figure 24.19 Determination Procedure of Serial Transfer Mode in Master Mode

(d) Burst Transfer

If the SSLKP bit in the RSPIi command register (SPiCMD) that the RSPI references during the current serial transfer is "1", the RSPIi keeps the SSLi signal level during the serial transfer until the beginning of the SSLi signal assertion for the next serial transfer. If the SSLi signal level for the next serial transfer is the same as the SSLi signal level for the current serial transfer, the RSPIi can execute continuous serial transfers while keeping the SSLi signal assertion status (burst transfer).

Figure 24.20 shows an example of an SSL signal operation for the case where a burst transfer is implemented using SPiCMD0 and SPiCMD1 registers settings. The text below explains the RSPIi operations (1) to (7) as depicted in figure 24.20. It should be noted that the polarity of the SSLi output signal depends on the settings in the RSPIi slave select polarity register (SPiSSLP).

1. Based on the SPiCMD0 register, the RSPI asserts the SSL signal and inserts RSPCK delays.
2. The RSPIi executes serial transfers according to the SPiCMD0 register.
3. The RSPIi inserts SSL negation delays.
4. Because the SSLKP bit in SPiCMD0 is "1", the RSPIi keeps the SSL signal value on the SPiCMD0 register. This period is sustained, at the shortest, for a period equal to the next-access delay + 2 Pck of the SPiCMD0 register. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
5. Based on the SPiCMD1 register, the RSPIi asserts the SSL signal and inserts RSPCK delays.
6. The RSPIi executes serial transfers according to the SPiCMD1 register.
7. Because the SSLKP bit in the SPiCMD1 register is "0", the RSPIi negates the SSLi signal. In addition, a next-access delay is inserted according to the SPiCMD1 register.

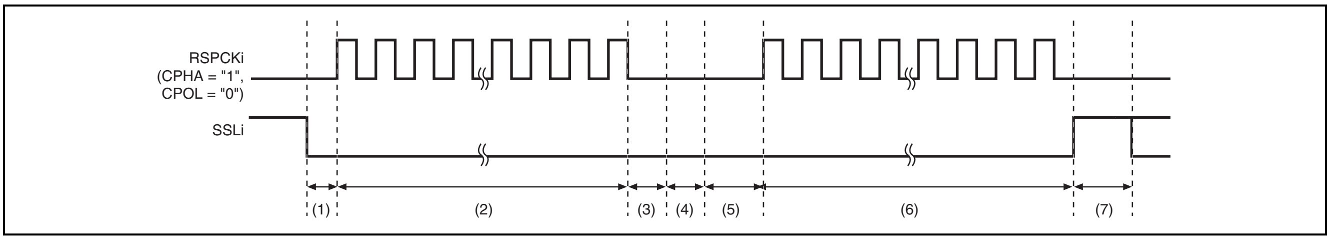


Figure 24.20 Example of Burst Transfer Operation using SSLKP Bit

If the SSL signal settings in the SPiCMD register in which "1" is assigned to the SSLKP bit are different from the SSLi signal output settings in the SPiCMD register to be used in the next transfer, the RSPIi switches the SSLi signal status to SSLi signal assertion ((5) in figure 24.20) corresponding to the command for the next transfer. Notice that if such an SSLi signal switching occurs, the slaves that drive the MISOi signal compete, and the possibility arises of the collision of signal levels.

The RSPIi in master mode references within the module the SSL signal operation for the case where the SSLKP bit is not used. Even when the CPHA bit in the SPiCMD register is "0", the RSPIi can accurately start serial transfers by asserting the SSLi signal for the next transfer. For this reason, burst transfers in master mode can be executed irrespective of CPHA bit settings (see section 24.4.9 (2), Slave Mode Operation).

(e) RSPCK Delay (t1)

The RSPCK delay value of the RSPIi in master mode depends on SCKDEN bit settings in the RSPIi command register (SPiCMD) and on RSPIi clock delay register (SPiCKD) settings. The RSPIi determines the SPiCMD register to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SCKDEN bit in the selected the SPiCMD and SPiCKD registers, as shown in table 24.9. For a definition of RSPCK delay, see section 24.4.4, Transfer Format.

Table 24.9 Relationship among SCKDEN Bit and SPiCKD Register Settings, and RSPCK Delay Values

SCKDEN Bits	SCKDL 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

(f) SSL Negation Delay (t2)

The SSL negation delay value of the RSPIi in master mode depends on SLNDEN bit settings in the RSPIi command register (SPiCMD) and on RSPi slave select negation delay register (SPiSSLND) settings. The RSPIi determines the SPiCMD register to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SLNDEN bit in the selected the SPiCMD register and the SPiSSLND register, as shown in table 24.10. For a definition of SSL negation delay, see section 24.4.4, Transfer Format.

Table 24.10 Relationship among SLNDEN Bit and SPiSSLND Regsiter Settings, and SSL Negation Delay Values

SLNDEN Bits	SLNDL 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

(g) Next-Access Delay (t3)

The next-access delay value of the RSPIi in master mode depends on SPNDEN bit settings in the RSPIi command register (SPiCMD) and on the RSPi next-access delay register (SPiND) settings. The RSPIi determines the SPiCMD register to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPNDEN bit in the selected the SPiCMD and SPiND registers, as shown in table 24.11. For a definition of next-access delay, see section 24.4.4, Transfer Format.

Table 24.11 Relationship among SPNDEN Bit and SPiND Register Settings, and Next-Access Delay Values

SPNDEN Bits	SPiND Registers	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2Pck
1	000	1 RSPCK + 2Pck
	001	2 RSPCK + 2Pck
	010	3 RSPCK + 2Pck
	011	4 RSPCK + 2Pck
	100	5 RSPCK + 2Pck
	101	6 RSPCK + 2Pck
	110	7 RSPCK + 2Pck
	111	8 RSPCK + 2Pck

(h) Initialization Flowchart

Figure 24.21 shows an example initialization flowchart during SPI operation when using RSPIi in master mode. For a description of how to set up an interrupt controller, the DMAC, and input/output ports, see the descriptions given in the individual blocks.

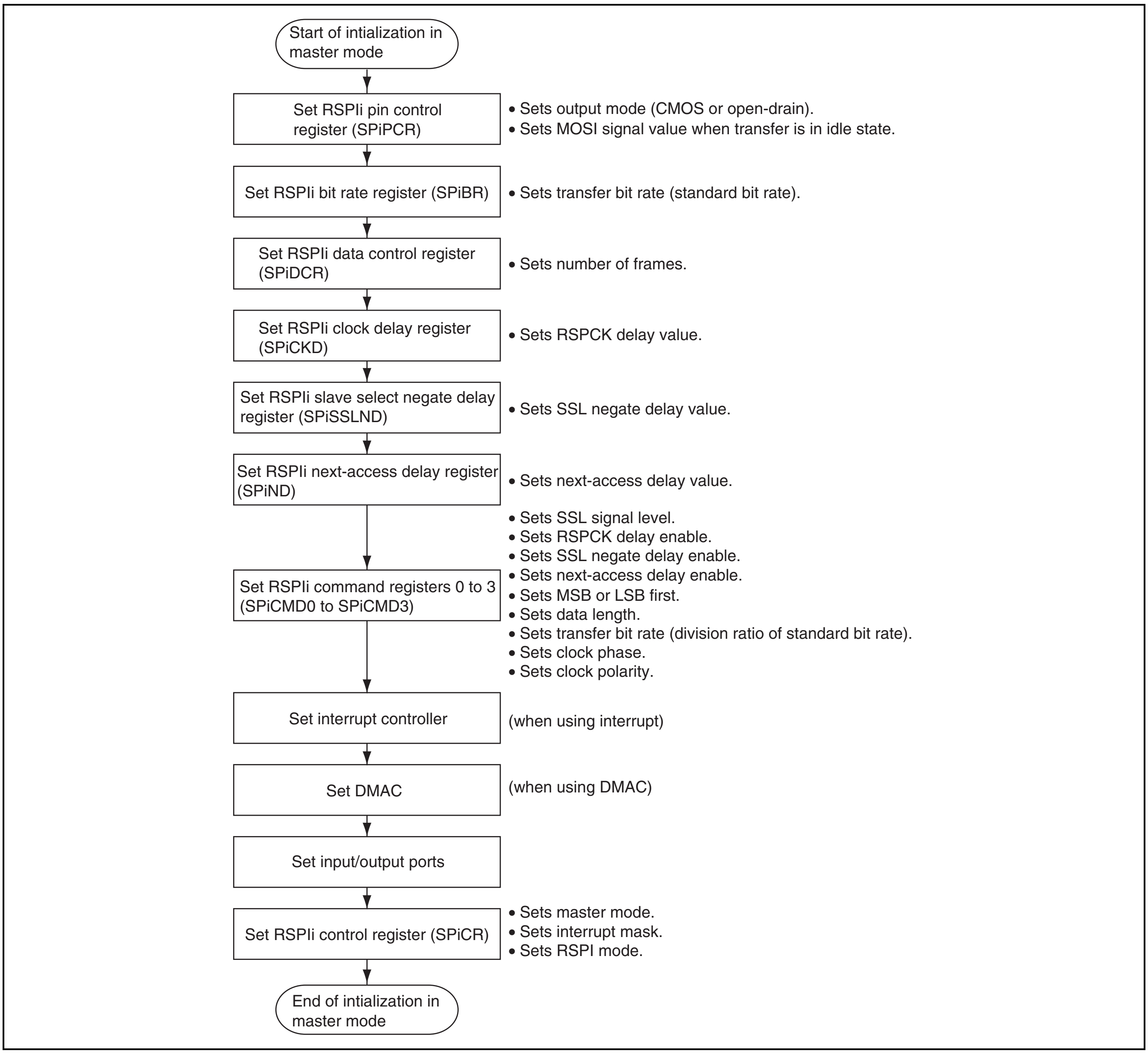
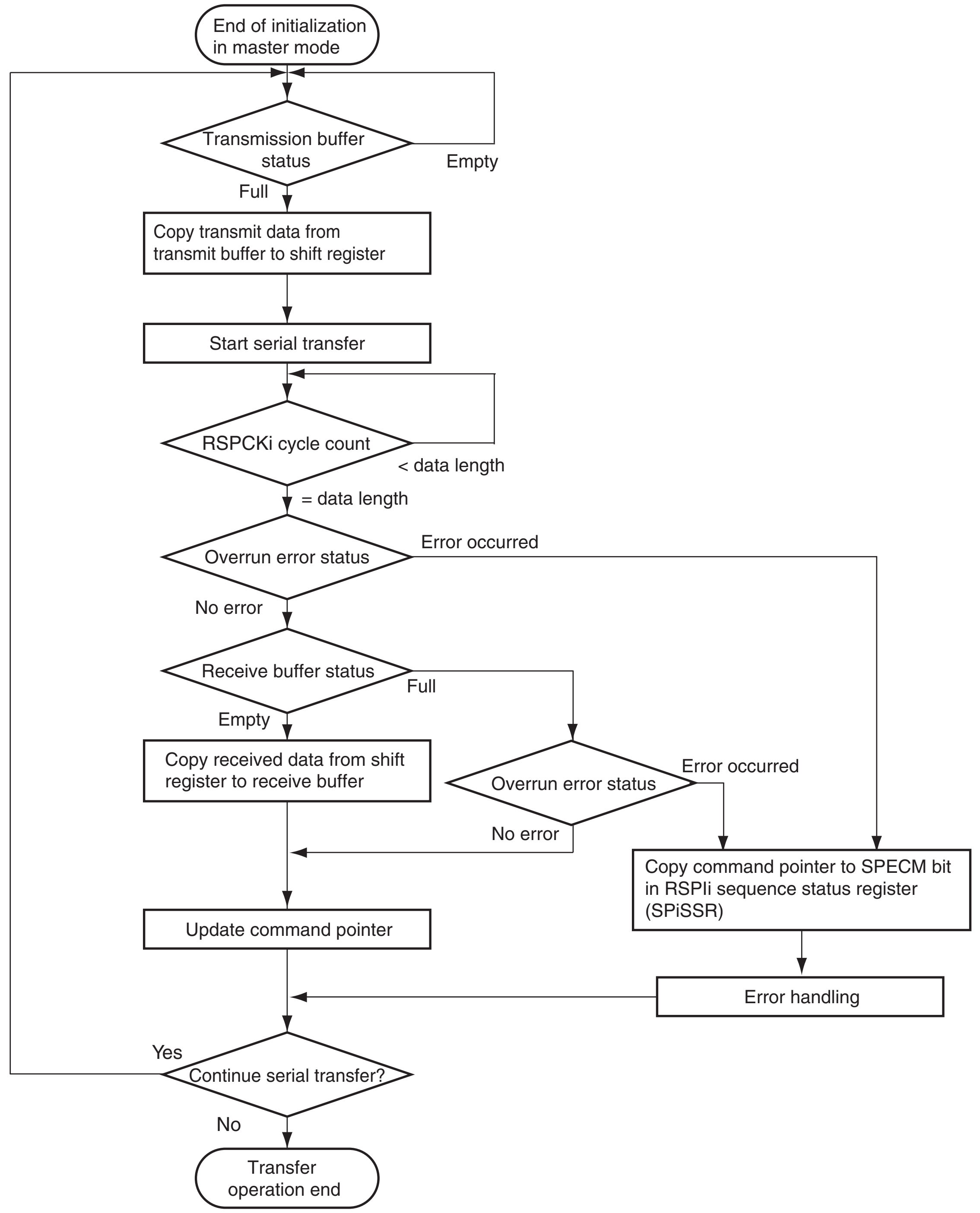

Start of intialization in master mode
Set RSPIi pin control register (SPiPCR)
• Sets output mode (CMOS or open-drain).
• Sets MOSI signal value when transfer is in idle state.
Set RSPIi bit rate register (SPiBR)
• Sets transfer bit rate (standard bit rate).
Set RSPIi data control register (SPiDCR)
• Sets number of frames.
Set RSPIi clock delay register (SPiCKD)
• Sets RSPCK delay value.
Set RSPIi slave select negate delay register (SPiSSLND)
• Sets SSL negate delay value.
Set RSPIi next-access delay register (SPiND)
• Sets next-access delay value.
Set RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3)
• Sets SSL signal level.
• Sets RSPCK delay enable.
• Sets SSL negate delay enable.
• Sets next-access delay enable.
• Sets MSB or LSB first.
• Sets data length.
• Sets transfer bit rate (division ratio of standard bit rate).
• Sets clock phase.
• Sets clock polarity.
Set interrupt controller
(when using interrupt)
Set DMAC
(when using DMAC)
Set input/output ports
Set RSPIi control register (SPiCR)
• Sets master mode.
• Sets interrupt mask.
• Sets RSPI mode.
End of intialization in master mode

Figure 24.21 Example of Initialization Flowchart in Master Mode

(i) Transfer Operation Flowchart

Figure 24.22 shows an example transfer flowchart during SPI operation when using RSPIi in master mode.


End of initialization in master mode
Transmission buffer status
Empty
Full
Copy transmit data from transmit buffer to shift register
Start serial transfer
RSPCKi cycle count
< data length
= data length
Overrun error status
Error occurred
No error
Receive buffer status
Full
Empty
Copy received data from shift register to receive buffer
Overrun error status
Error occurred
No error
Copy command pointer to SPECM bit in RSPIi sequence status register (SPiSSR)
Update command pointer
Error handling
Yes
Continue serial transfer?
No
Transfer operation end

Note: • When the SSLi0 input signal is asserted in multi-master mode, the RSPIi detects a mode fault error irrespective of the hardware status. When detecting the error, the RSPIi copies the command pointer to the SPECM bit in the RSPIi sequence status register (SPiSSR).

Figure 24.22 Example of Transfer Operation Flowchart in Master Mode

(2) Slave Mode Operation

(a) Starting a Serial Transfer

If the CPHA bit in RSPIi command register 0 (SPiCMD0) is "0", when detecting an SSLi0 input signal assertion, the RSPIi needs to start driving valid data to the MISOi output signal. For this reason, when the CPHA bit is "0", the asserting of the SSLi0 input signal triggers the start of a serial transfer.

If the CPHA bit is "1", when detecting the first RSPCKi edge in an SSLi0 signal asserted condition, the RSPIi needs to start driving valid data to the MSOi signal. For this reason, when the CPHA bit is "1", the first RSPCKi edge in an SSLi0 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 RSPIi changes the status of the shift register to "full", so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, the RSPIi leaves the status of the shift register intact, in the full state.

Irrespective of CPHA bit settings, the timing at which the RSPIi starts driving MISOi output signals is the SSLi0 signal assertion timing. The data which is output by the RSPIi is either valid or invalid, depending on CPHA bit settings.

For details on the RSPIi transfer format, see section 24.4.4, Transfer Format. The polarity of the SSLi0 input signal depends on the setting of the SSL0P bit in the RSPIi slave select polarity register (SPiSSLP).

(b) Terminating a Serial Transfer

Irrespective of the CPHA bit in RSPIi command register 0 (SPiCMD0), the RSPIi terminates the serial transfer after detecting an RSPCKi edge corresponding to the final sampling timing. When the SPRF bit in the RSPIi status register (SPiSR) is "0" and free space is available in the receive buffer, upon termination of serial transfer the RSPIi copies received data from the shift register to the receive buffer of the RSPIi data register (SPiDR). Irrespective of the value of the SPRF bit, upon termination of a serial transfer the RSPIi changes the status of the shift register to "empty". A mode fault error occurs if the RSPIi detects an SSLi0 input signal negation between the beginning and end of serial transfer. (see section 24.4.7, Error Detection).

The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the RSPIi data length depends on the settings in the SPB bit in the SPiCMD0 register. The polarity of the SSLi0 input signal depends on the setting in the SSL0P bit in the RSPIi slave select polarity register (SPiSSLP). For details on the RSPIi transfer format, see section 24.4.4, Transfer Format.

(c) Notes on Single-Slave Operations

If the CPHA bit in RSPIi command register 0 (SPiCMD0) is "0", the RSPIi starts serial transfers when it detects the assertion edge for an SSLi0 input signal. In the type of configuration shown in figure 24.5 as an example, if the RSPIi is used in single-slave mode, the SSLi0 signal is always fixed at active state. Therefore, when the CPHA bit is set to "0", the RSPIi cannot correctly start a serial transfer. To correctly execute send/receive operation by the RSPIi in a configuration in which the SSLi0 input signal is fixed at active state, the CPHA bit should be set to "1". If there is a need for setting the CPHA bit to "0", the SSLi0 input signal should not be fixed.

(d) Burst Transfer

If the CPHA bit in RSPIi command register 0 (SPiCMD0) is "1", continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSLi0 input signal. If the CPHA bit is "1", the period from the first RSPCKi edge to the sampling timing for the reception of the final bit in an SSLi0 signal active state corresponds to a serial transfer period. Even when the SSLi0 input signal remains at the active level, the RSPIi can accommodate burst transfers because it can detect the start of access.

If the CPHA bit is "0", for the reason given in (c), Notes on Single-Slave Operations of section 24.4.9 (2), Slave Mode Operation, second and subsequent serial transfers during the burst transfer cannot be executed correctly.

(e) Initialization Flowchart

Figure 24.23 shows an example initialization flowchart during SPI operation when using the RSPIi in slave mode. For a description of how to set up an interrupt controller, the DMAC, and input/output ports, see the descriptions given in the individual blocks.

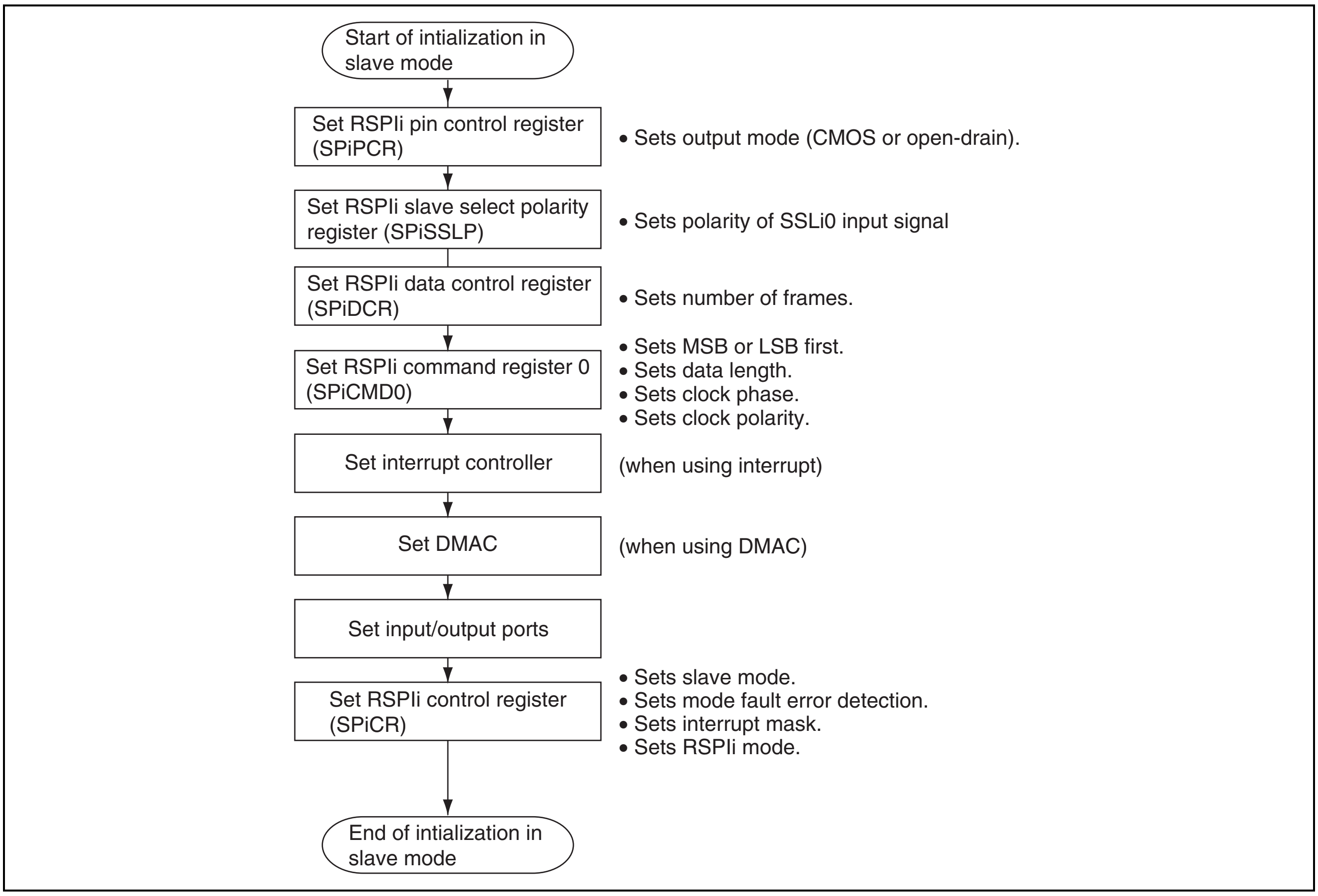

Start of intialization in slave mode
Set RSPIi pin control register (SPiPCR)
• Sets output mode (CMOS or open-drain).
Set RSPIi slave select polarity register (SPiSSLP)
• Sets polarity of SSLi0 input signal
Set RSPIi data control register (SPiDCR)
• Sets number of frames.
Set RSPIi command register 0 (SPiCMD0)
• Sets MSB or LSB first.
• Sets data length.
• Sets clock phase.
• Sets clock polarity.
Set interrupt controller
(when using interrupt)
Set DMAC
(when using DMAC)
Set input/output ports
Set RSPIi control register (SPiCR)
• Sets slave mode.
• Sets mode fault error detection.
• Sets interrupt mask.
• Sets RSPIi mode.
End of intialization in slave mode

Figure 24.23 Example of Initialization Flowchart in Slave Mode

(f) Transfer Operation Flowchart (CPHA = "0")

Figure 24.24 shows an example transfer flowchart during SPI operation when using RSPIi in slave mode with the CPHA bit in RSPIi command register 0 (SPiCMD0) cleared to "0".

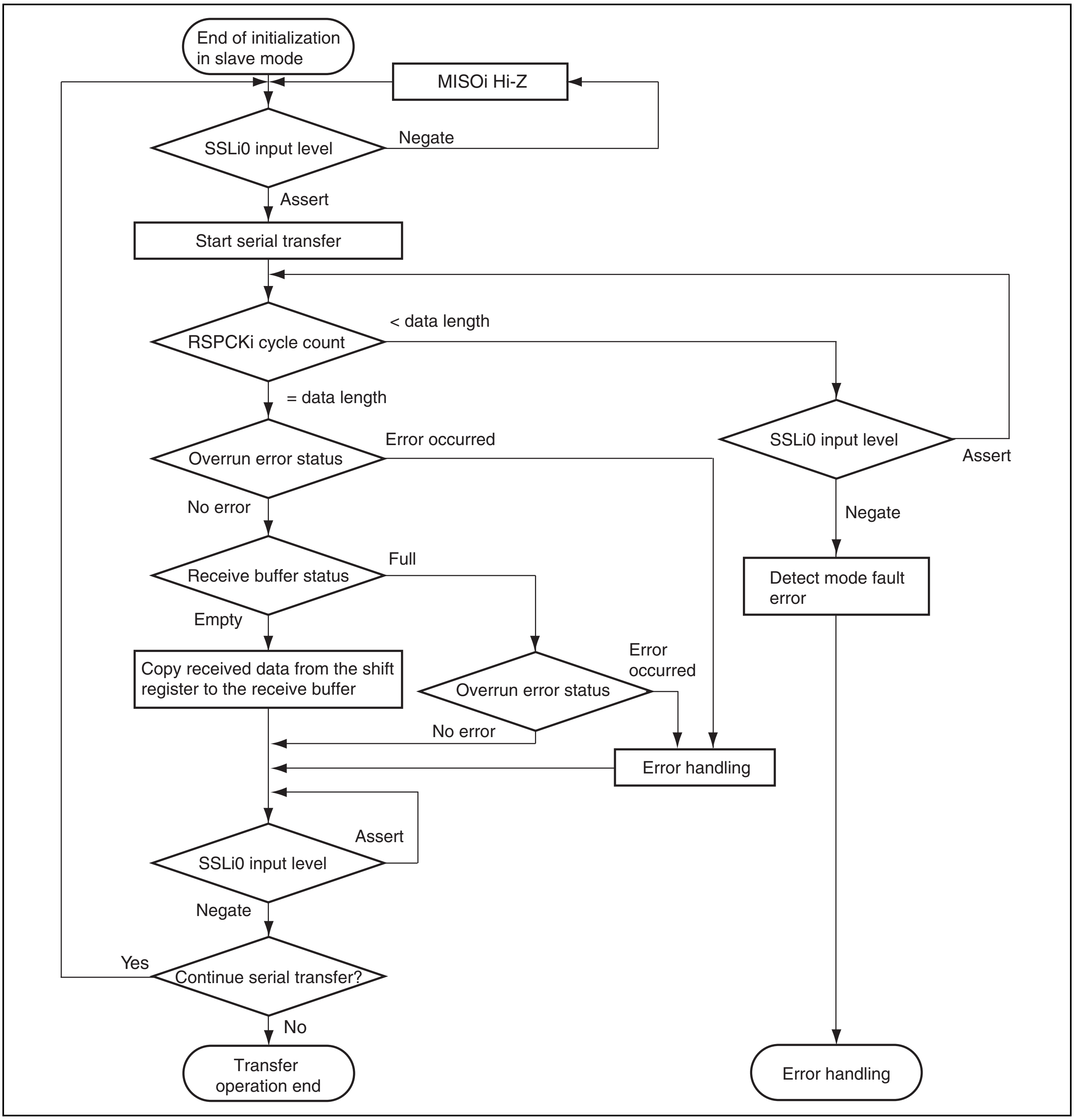


Figure 24.24 Example of Transfer Operation Flowchart in Slave Mode (CPHA = "0")

(g) Transfer Operation Flowchart (CPHA = "1")

Figure 24.25 shows an example transfer flowchart during SPI operation when using RSPIi in slave mode with the CPHA bit in RSPIi command register 0 (SPiCMD0) set to "1".

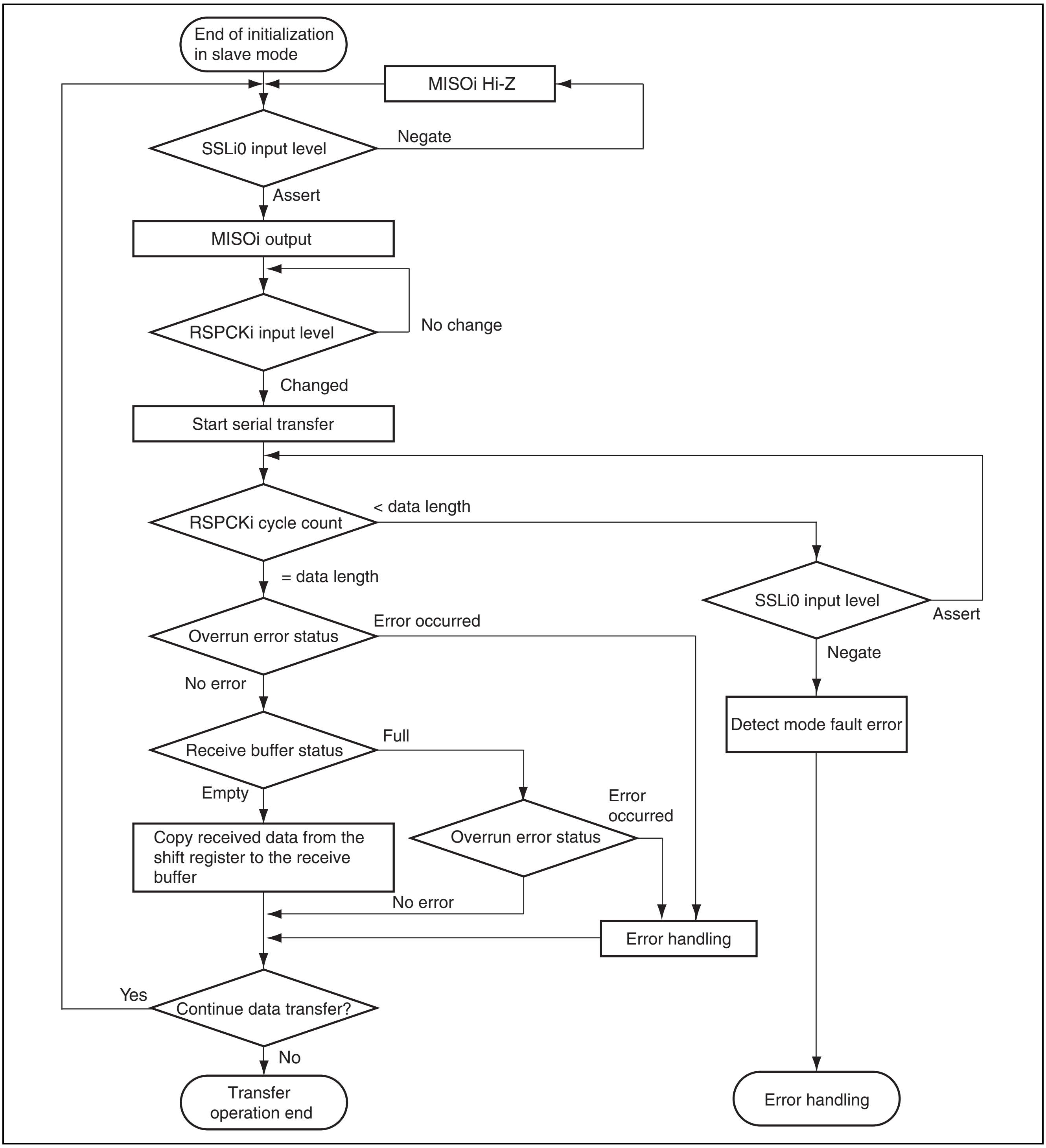


Figure 24.25 Transfer Operation Flowchart in Slave Mode (CPHA = "1")

24.4.10 Clock Synchronous Operation

RSPIi operates in clock-synchronous mode when the SPMS bit in the RSPIi control register (SPiCR) is set to "1". During clock-synchronous operation, the SSLi pins are unused and three pins, RSPCKi, MOSIi, and MISOi, are used for communication. This means the SSLi pins can be used as I/O ports.

The SSLi pins are not used for communication in clock-synchronous operation, but the internal operation of the RSPI module is the same as during SPI operation. In both master and slave mode, communication takes place using the same sequence as during SPI operation. However, mode fault error detection does not take place because the SSLi pins are not used.

Also note that clock-synchronous operation cannot be guaranteed when the CPHA bit in the RSPIi command register (SPiCMD) is cleared to "0".

(1) Master Mode Operation

(a) Starting a Serial Transfer

When the CPU or DMAC writes data to the RSPIi data register (SPiDR) while the SPTEF bit in the RSPIi status register (SPiSR) is set to "1", RSPIi updates the data in the SPiDR register transmit buffer. When the shift register is empty in a condition where the SPTEF bit has been cleared to "0" by a write to the SPiDR register or by the CPU reading the SPTEF bit as "1" and then writing "0" to it, RSPIi copies the data in the transmit buffer to the shift register and starts a serial transfer. The status of the shift register changes to full when RSPIi copies the transmit data to it and then back to empty when the serial transfer finishes. It is not possible for the CPU to reference the status of the shift register.

For details on the RSPIi transfer format, see section 24.4.4, Transfer Format. Note that the SSLi0 output signal is not used during clock-synchronous operation.

(b) Terminating a Serial Transfer

RSPIi terminates the serial transfer when an RSPCKi edge corresponding to the sampling timing is transmitted. When the SPRF bit in the RSPIi status register (SPiSR) is cleared to "0" and free space is available in the receive buffer, RSPIi copies data from the shift register to the receive buffer of the RSPIi data register (SPiDR) after the serial transfer terminates.

Note that the final sampling timing varies depending on the bit length of the transfer data. In master mode, the RSPIi data length depends on the setting of the SPB bits in the RSPIi command register (SPiCMD). For details on the RSPIi transfer format, see section 24.4.4, Transfer Format. Note that the SSLi0 output signal is not used during clock-synchronous operation.

(c) Sequence Control

The transfer format employed in master mode is determined by the RSPIi sequence control register (SPiSCR), RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3), RSPIi bit rate register (SPiBR), RSPIi clock delay register (SPiCKD), RSPIi slave select negation delay register (SPiSSLND), and RSPIi next-access delay register (SPiND). The SSLi signals are not output during clock-synchronous operation, but the above settings are valid.

The SPiSCR register is used to determine the sequence configuration for serial transfers executed by RSPIi in master mode. The settings of registers SPiCMD0 to SPiCMD3 specify the SSLi output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCKi polarity/phase, whether or not the SPiCKD register is referenced, whether or not the SPiSSLND register is referenced, and whether or not the SPiND register is referenced. The SPiBR register specifies some of the bit rate settings, the SPiCKD register specifies the RSPIi clock delay value, the SPiSSLND register specifies the SSL negation delay, and the SPiND register specifies the next-access delay value.

According to the sequence length specified in the SPiSCR register, RSPIi composes a sequence comprising some or all of registers SPiCMD0 to SPiCMD3. RSPIi has a pointer to the SPiCMD registers that compose the sequence. The CPU can check the pointer value by reading the SPCP bits in the RSPIi sequence status register (SPiSSR). When the SPE bit in the RSPIi control register (SPiCR) is set to "1" and the RSPI function is enabled, RSPIi sets the command pointer to the SPiCMD0 register, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. RSPIi 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 in the sequence, RSPIi sets the pointer to the SPiCMD0 register, and in this manner the sequence is executed repeatedly.

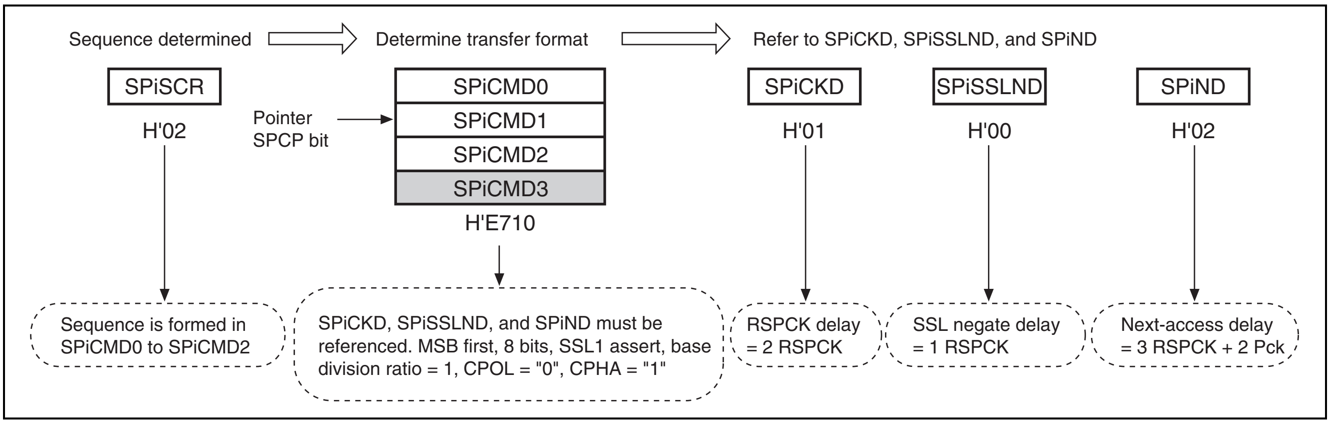
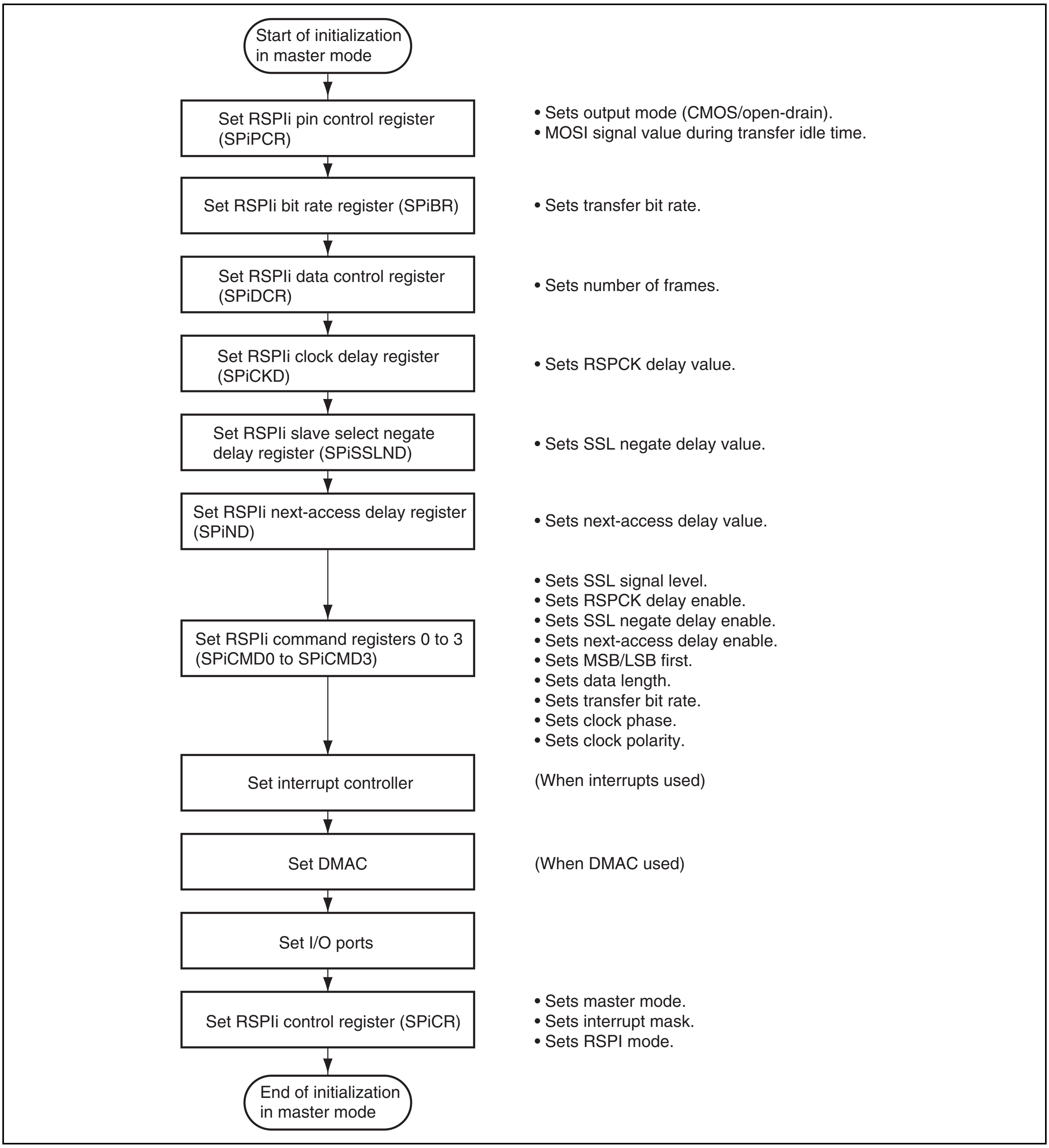

Sequence determined
Determine transfer format
Refer to SPiCKD, SPiSSLND, and SPiND
SPiSCR
H'02
Pointer
SPCP bit
SPiCMD0
SPiCMD1
SPiCMD2
SPiCMD3
H'E710
SPiCKD
H'01
SPiSSLND
H'00
SPiND
H'02
Sequence is formed in SPiCMD0 to SPiCMD2
SPiCKD, SPiSSLND, and SPiND must be referenced. MSB first, 8 bits, SSL1 assert, base division ratio = 1, CPOL = "0", CPHA = "1"
RSPCK delay = 2 RSPCK
SSL negate delay = 1 RSPCK
Next-access delay = 3 RSPCK + 2 Pck

Figure 24.26 Determination Procedure of Serial Transfer Mode in Master Mode

(d) Initialization Flowchart

Figure 24.27 shows an example initialization flowchart during clock-synchronous operation when using RSPIi in master mode. For information on how to make settings for the interrupt controller, DMAC, and input/output ports, see the descriptions of the individual blocks.


Start of initialization in master mode
Set RSPIi pin control register (SPiPCR)
Set RSPIi bit rate register (SPiBR)
Set RSPIi data control register (SPiDCR)
Set RSPIi clock delay register (SPiCKD)
Set RSPIi slave select negate delay register (SPiSSLND)
Set RSPIi next-access delay register (SPiND)
Set RSPIi command registers 0 to 3 (SPiCMD0 to SPiCMD3)
Set interrupt controller
Set DMAC
Set I/O ports
Set RSPIi control register (SPiCR)
End of initialization in master mode

- Sets output mode (CMOS/open-drain).
- MOSI signal value during transfer idle time.

- Sets transfer bit rate.

- Sets number of frames.

- Sets RSPCK delay value.

- Sets SSL negate delay value.

- Sets next-access delay value.

- Sets SSL signal level.
- Sets RSPCK delay enable.
- Sets SSL negate delay enable.
- Sets next-access delay enable.
- Sets MSB/LSB first.
- Sets data length.
- Sets transfer bit rate.
- Sets clock phase.
- Sets clock polarity.

(When interrupts used)

(When DMAC used)

- Sets master mode.
- Sets interrupt mask.
- Sets RSPI mode.

Figure 24.27 Flowchart of Initialization in Master Mode

(e) Transfer Operation Flowchart

Figure 24.28 shows a transfer flowchart during clock-synchronous operation in master mode.

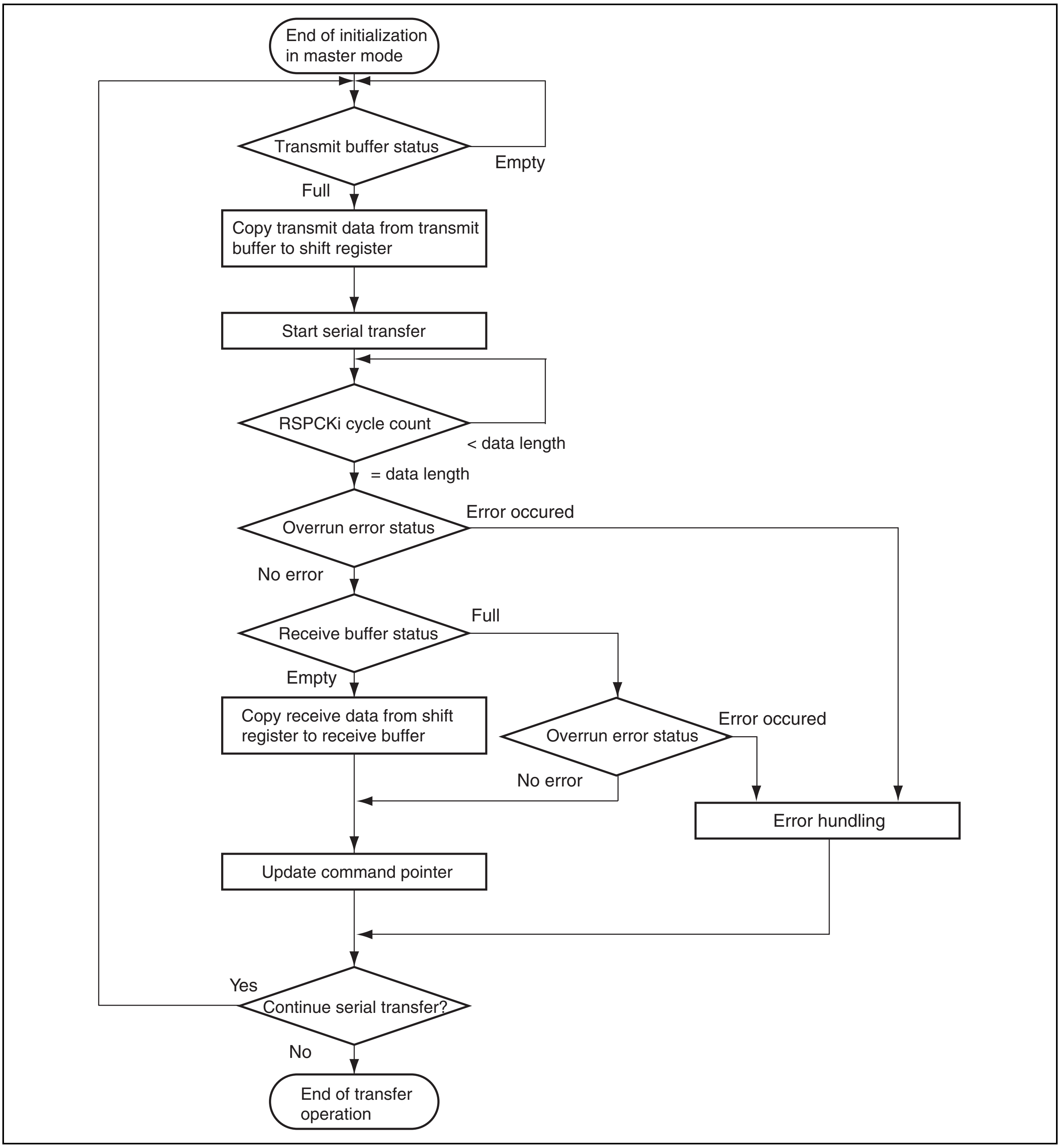


Figure 24.28 Flowchart of Transfer Operation in Master Mode

(2) Slave Mode Operation

(a) Starting a Serial Transfer

When the SPMS bit in the RSPIi control register (SPiCR) is set to "1", the first RSPCKi edge triggers the start of a serial transfer by RSPIi.

When RSPIi detects the start of a serial transfer when the shift register is empty, it changes the shift register status to full so that data cannot be copied from the transmit buffer to the shift register while the serial transfer is in progress. If the shift register was full before the serial transfer started, RSPI leaves the status of the shift register as full.

When the SPMS bit is set to "1", RSPIi drives the MISOi output signal constantly.

For details on the RSPIi transfer format, see section 24.4.4, Transfer Format. Note that the SSLi0 output signal is not used during clock-synchronous operation.

(b) Terminating a Serial Transfer

RSPIi terminates the serial transfer after detecting an RSPCKi edge corresponding to the final sampling timing. When the SPRF bit in the RSPIi status register (SPiSR) is cleared to "0" and free space is available in the receive buffer, RSPIi copies receive data from the shift register to the receive buffer of the RSPIi data register (SPiDR) after the serial transfer terminates. Irrespective of the value of the SPRF bit, upon termination of a serial transfer RSPIi changes the status of the shift register to empty. Note that the final sampling timing varies depending on the bit length of the transfer data. In slave mode, the RSPIi data length depends on the setting of the SPB bits in the SPiCMD0 register (SPiCMD). For details on the RSPIi transfer format, see section 24.4.4, Transfer Format.

(c) Initialization Flowchart

Figure 24.29 shows an example initialization flowchart during clock-synchronous operation when using RSPIi in slave mode. For information on how to make settings for the interrupt controller, DMAC, and input/output ports, see the descriptions of the individual blocks.

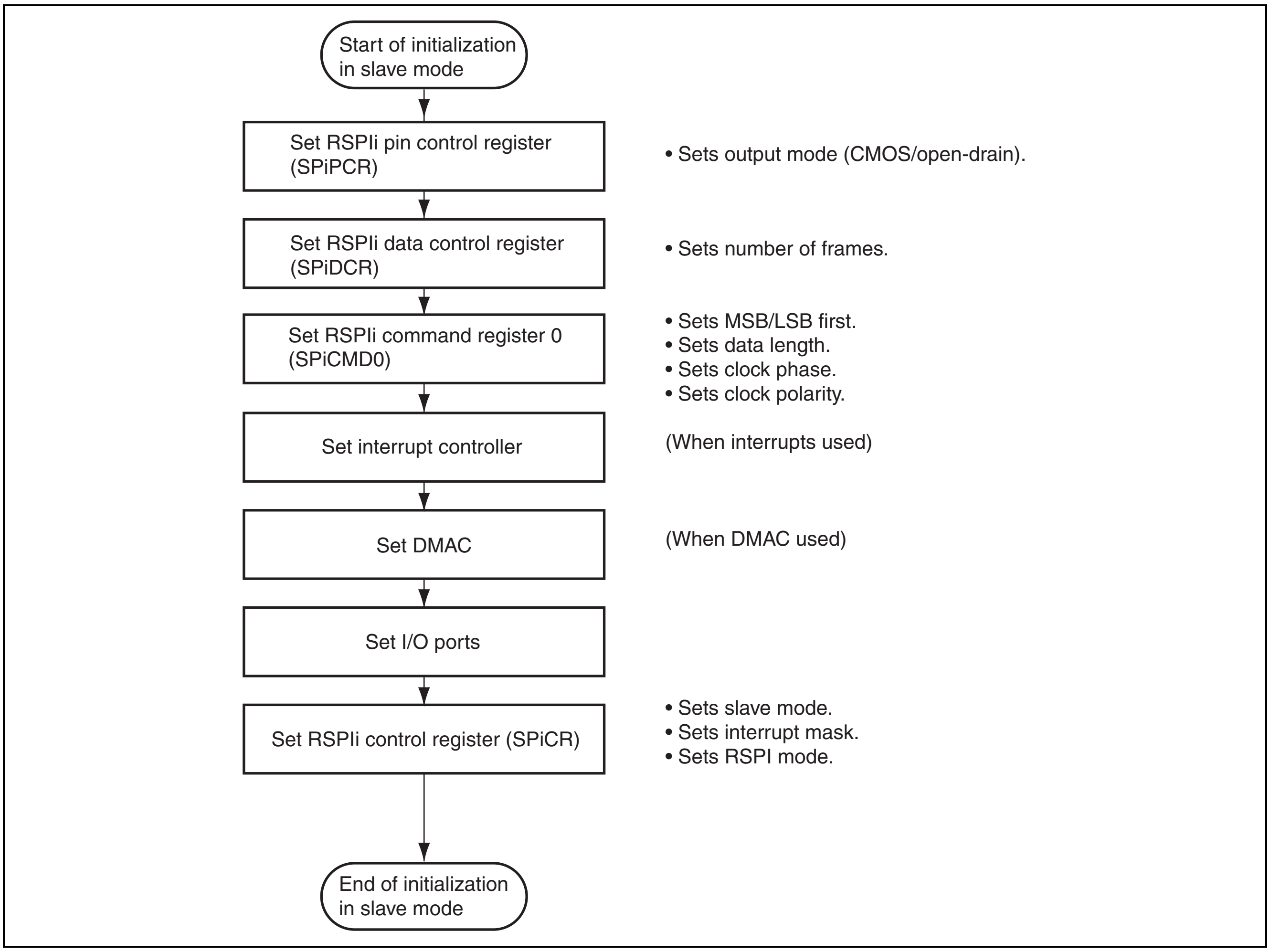

Start of initialization in slave mode
Set RSPIi pin control register (SPiPCR)
• Sets output mode (CMOS/open-drain).
Set RSPIi data control register (SPiDCR)
• Sets number of frames.
Set RSPIi command register 0 (SPiCMD0)
• Sets MSB/LSB first.
• Sets data length.
• Sets clock phase.
• Sets clock polarity.
Set interrupt controller
(When interrupts used)
Set DMAC
(When DMAC used)
Set I/O ports
Set RSPIi control register (SPiCR)
• Sets slave mode.
• Sets interrupt mask.
• Sets RSPI mode.
End of initialization in slave mode

Figure 24.29 Flowchart of Initialization in Slave Mode

(d) Transfer Operation Flowchart

Figure 24.30 shows an RSPIi transfer flowchart during clock-synchronous operation.

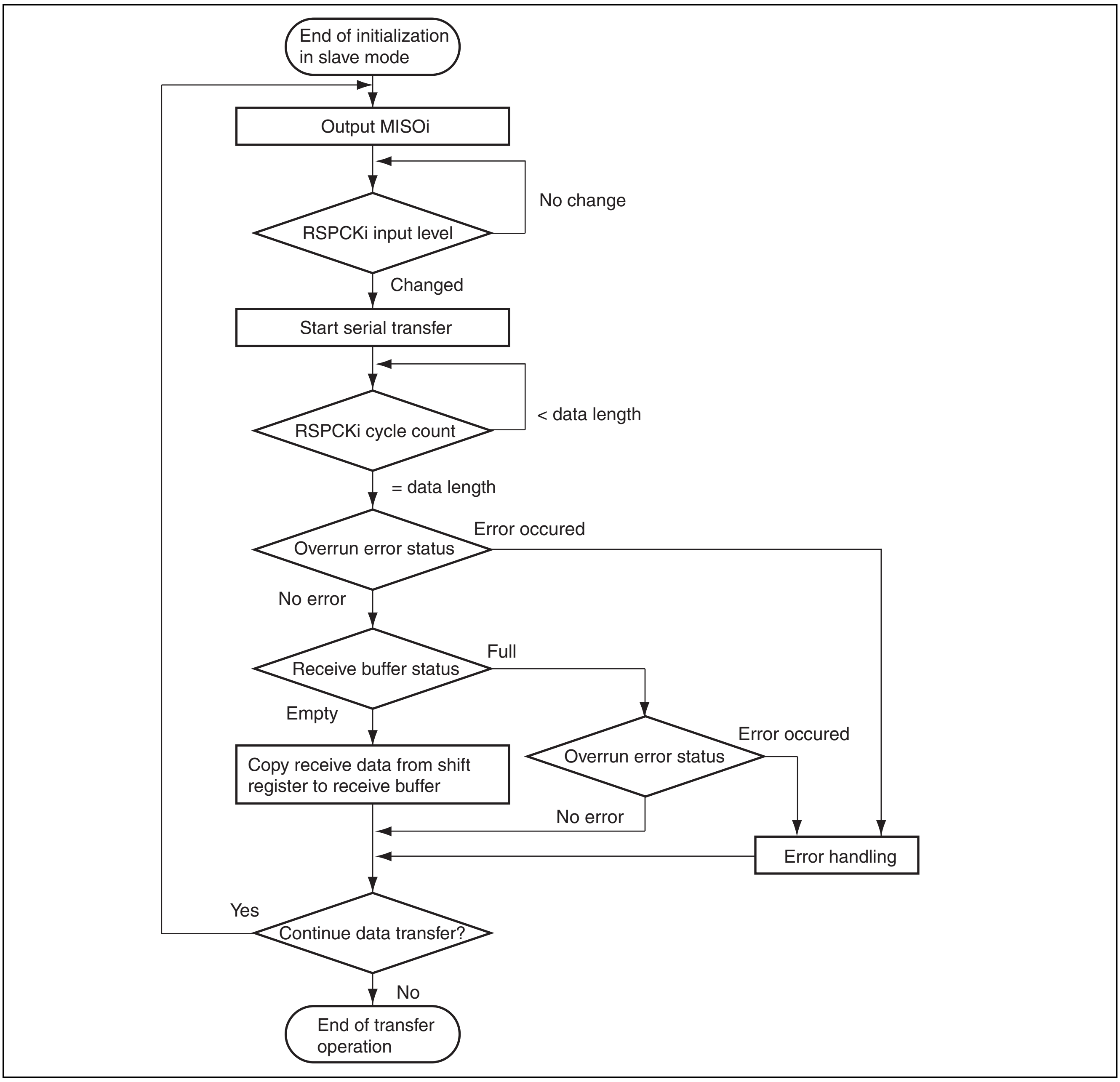


Figure 24.30 Flowchart of Transfer Operation in Slave Mode (CPHA = "1")

24.4.11 Error Handling

Figures 24.31 and 24.32 illustrate error handling by the RSPI. The following error handling routines can be used to recover from errors occurring in master mode or slave mode.

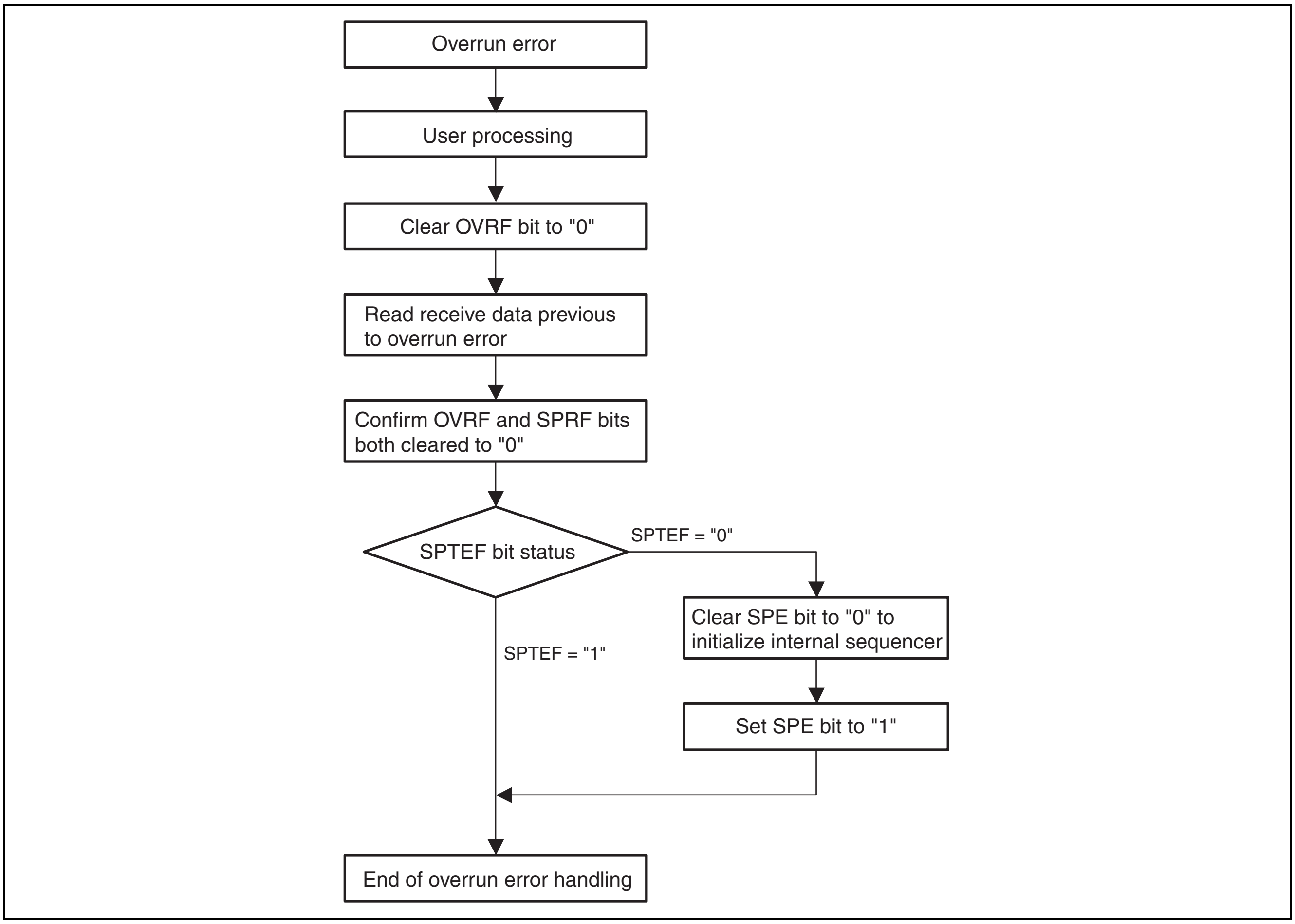

Overrun error
User processing
Clear OVRF bit to "0"
Read receive data previous to overrun error
Confirm OVRF and SPRF bits both cleared to "0"
SPTEF bit status
SPTEF = "0"
Clear SPE bit to "0" to initialize internal sequencer
Set SPE bit to "1"
SPTEF = "1"
End of overrun error handling

Figure 24.31 Error Handling (Overrun Error)

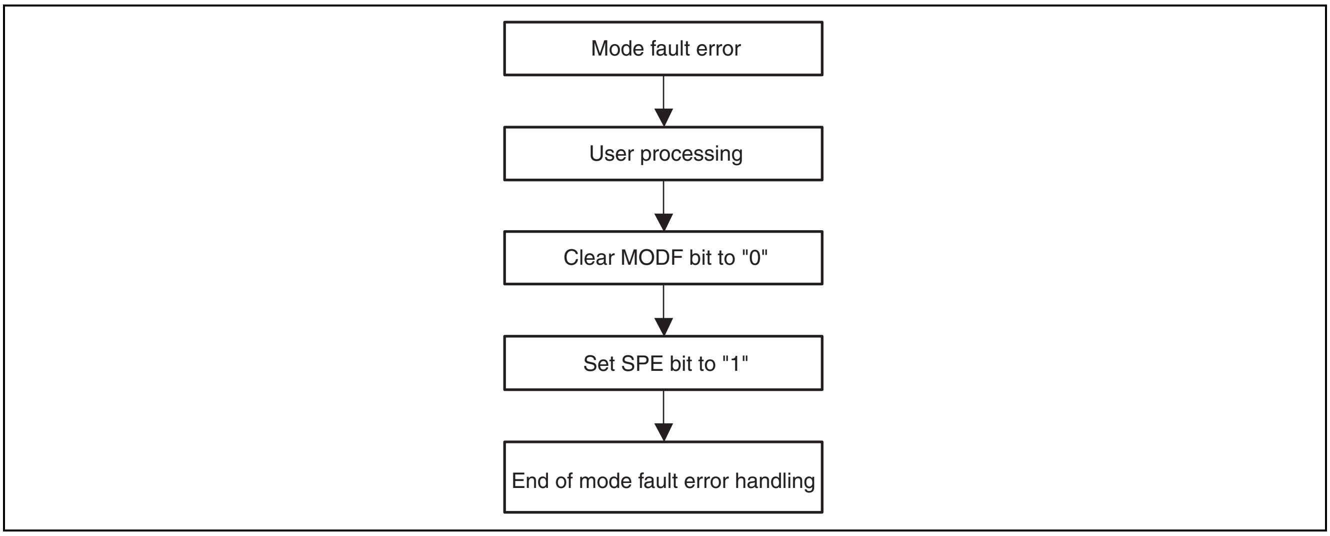


Figure 24.32 Error Handling (Mode Fault Error)

24.4.12 Loopback Mode

When the CPU writes "1" to the SPLP bit in the RSPIi pin control register (SPiPCR), the RSPIi shuts off the path between the MISOi pin and the shift register, and between the MOSIi pin and the shift register, and connects the input path and the output path (reversed) of the shift register. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data for the RSPIi becomes the received data for the RSPIi. Figure 24.33 shows the configuration of the shift register input/output paths for the case where the RSPIi in master mode is set in loopback mode.

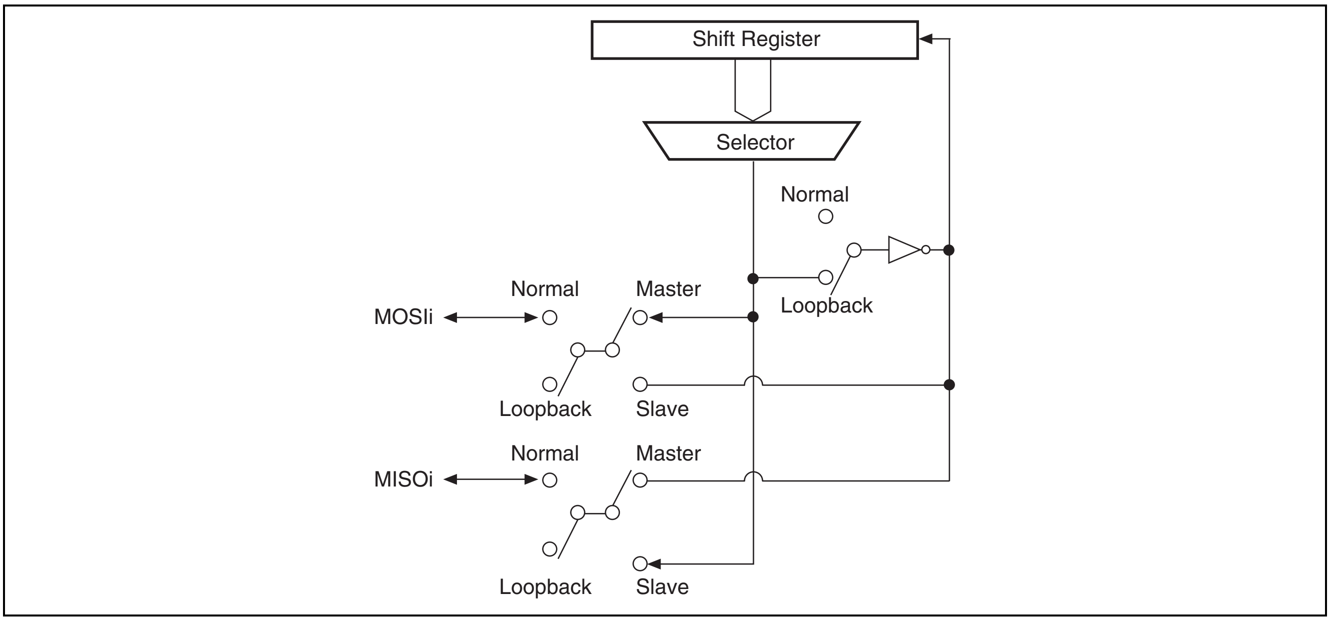


Figure 24.33 Configuration of Shift Register Input/Output Paths in Loopback Mode (Master Mode)

24.4.13 Interrupt Sources

The RSPIi interrupt sources are receive buffer full, transmit buffer empty, mode fault, and overrun.

The receive buffer full interrupt is assigned to SPRIn, the transmit buffer empty interrupt to SPTIn, and the mode fault and overrun interrupts to SPEIn. It is therefore necessary to use flags to identify interrupts. Table 24.12 lists the RSPIi interrupt sources.

An interrupt is generated when one of the interrupt conditions listed in table 24.12 is met. The interrupt sources should be cleared by a data transfer operation initiated by the CPU or DMAC.

Table 24.12 RSPIi Interrupt Sources Interrupt Condition

Interrupt Source	Interrupt Condition
Receive buffer full	(SPiCR.SPRIE = 1) • (SPiSR.SPRF = 1)
Transmit buffer empty	(SPiCR.SPTIE = 1) • (SPiSR.SPTEF = 1)
Mode fault	(SPiCR.SPEIE = 1) • (SPiSR.MODF = 1)
Overrun	(SPiCR.SPEIE = 1) • (SPiSR.OVRF = 1)

24.4.14 DMA Transfer Sources

The RSPI DMA transfer request sources are receive buffer full and transmit buffer empty. RSPIi issues a DMA transfer request when one of the DMA transfer request conditions listed in table 24.13 is met. If the appropriate DMAC startup settings have been made, the DMAC starts and data transfer can be performed. If the transmit data empty DMA transfer request is enabled in the DMAC startup settings, do not enable the corresponding interrupt request at the same time by setting the SPTIE bit to "1". In like manner, if the receive data full DMA transfer request is enabled, do not enable the corresponding interrupt request at the same time by setting the SPRIE bit to "1".

Table 24.13 RSPIi DMA Transfer Request Sources

DMA Transfer Request Source	DMA Transfer Request Condition
Receive buffer full	SPiSR.SPRF = 1
Transmit buffer empty	SPiSR.SPTEF = 1

Section 25 I^{2}C Bus Interface 3 (IIC3)

The I^{2}C bus interface 3 conforms to and provides a subset of the Philips I^{2}C (Inter-IC) bus interface functions. However, the configuration of the registers that control the I^{2}C bus differs partly from the Philips register configuration.

25.1 Overview

- Selection of I^{2}C format or clocked synchronous serial format
- Continuous transmission/reception

  Since the shift register, transmit data register, and receive data register are independent from each other, the continuous transmission/reception can be performed.

I^{2}C bus format:

- Start and stop conditions generated automatically in master mode
- Selection of acknowledge output levels when receiving
- Automatic loading of acknowledge bit when transmitting
- Bit synchronization function

  In master mode, the state of SCL is monitored per bit, and the timing is synchronized automatically. If transmission/reception is not yet possible, set the SCL to "L" level until preparations are completed.
- Six interrupt sources

  Transmit data empty (including slave-address match), transmit end, receive data full (including slave-address match), arbitration lost, NACK detection, and stop condition detection
- The direct memory access controller (DMAC) can be activated by a transmit-data-empty request or receive-data-full request to transfer data.
- Direct bus drive

  Two pins, SCL and SDA pins, function as NMOS open-drain outputs when the bus drive function is selected.

Clocked synchronous serial format:

- Four interrupt sources

  Transmit-data-empty, transmit-end, receive-data-full, and overrun error
- The direct memory access controller (DMAC) can be activated by a transmit-data-empty request or receive-data-full request to transfer data.

Figure 25.1 shows a block diagram of the I^{2}C bus interface 3.

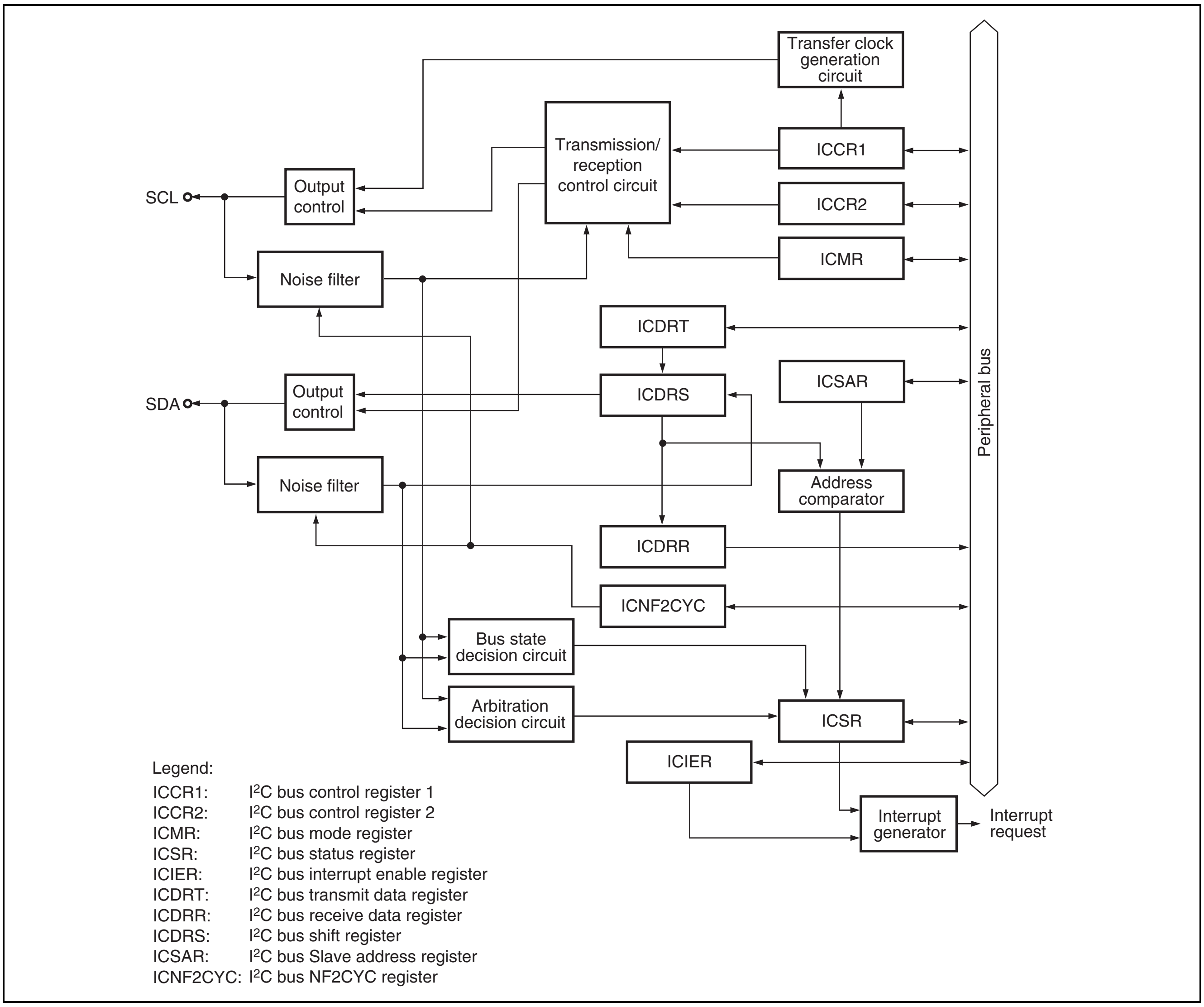

Transfer clock generation circuit
Transmission/ reception control circuit
ICCR1
ICCR2
ICMR
SCL
Output control
Noise filter
ICDRT
ICDRS
ICSAR
SDA
Output control
Noise filter
Address comparator
ICDRR
ICNF2CYC
Bus state decision circuit
Arbitration decision circuit
ICSR
ICIER
Interrupt generator
Interrupt request
Peripheral bus
Legend:
ICCR1: I2C bus control register 1
ICCR2: I2C bus control register 2
ICMR: I2C bus mode register
ICSR: I2C bus status register
ICIER: I2C bus interrupt enable register
ICDRT: I2C bus transmit data register
ICDRR: I2C bus receive data register
ICDRS: I2C bus shift register
ICSAR: I2C bus Slave address register
ICNF2CYC: I2C bus NF2CYC register

Figure 25.1 Block Diagram of I^{2}C Bus Interface 3

25.2 Input/Output Pins

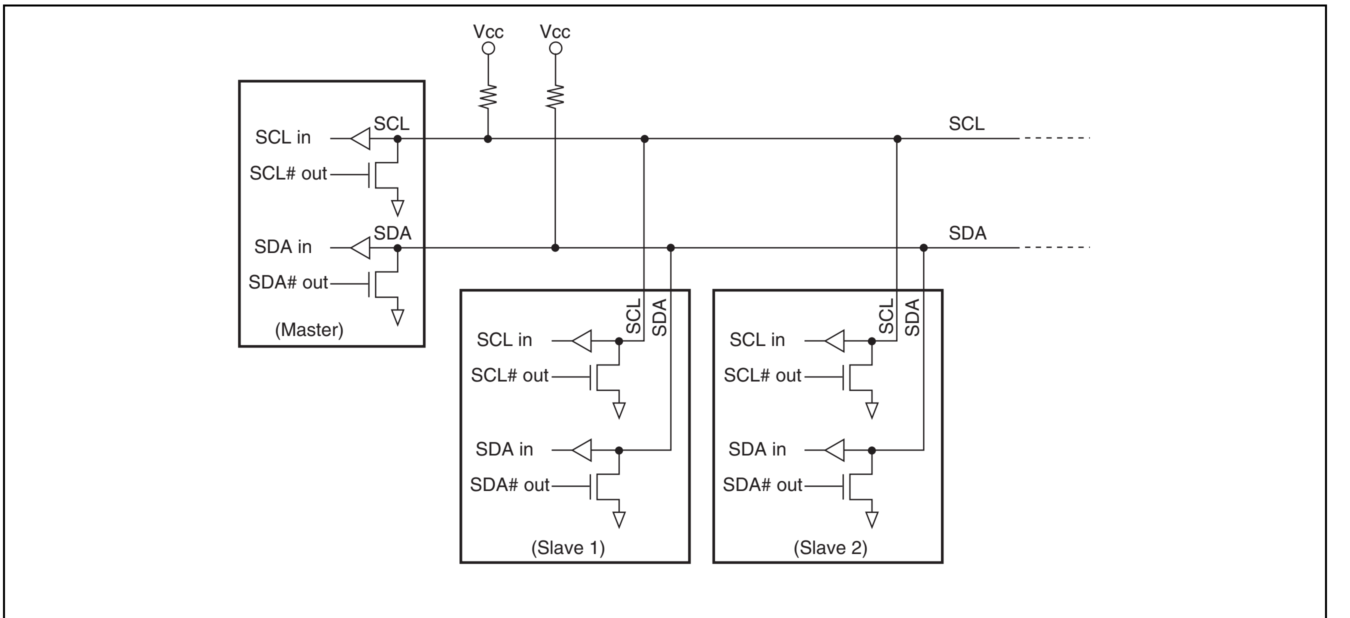
Table 25.1 shows the pin configuration of the I^{2}C bus interface 3.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 25.1 Pin Configuration

Pin Name	I/O	Function
SCL	I/O	I^{2}C serial clock input/output
SDA	I/O	I^{2}C serial data input/output

Figure 25.2 shows an example of I/O pin connections to external circuits.



Note: • Turn on/off Vcc for the I^{2}C bus power supply and for this MCU simultaneously.

Figure 25.2 External Circuit Connections of I/O Pins

25.3 Register Descriptions

The I^{2}C bus interface 3 has the following registers.

Table 25.2 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
I^{2}C bus control register 1	ICCR1	H'00	H'FFFE E000	8	25-4
I^{2}C bus control register 2	ICCR2	H'7D	H'FFFE E001	8	25-6
I^{2}C bus mode register	ICMR	H'38	H'FFFE E002	8	25-7
I^{2}C bus interrupt enable register	ICIER	H'00	H'FFFE E003	8	25-8
I^{2}C bus status register	ICSR	H'00	H'FFFE E004	8	25-10
I^{2}C bus Slave address register	ICSAR	H'00	H'FFFE E005	8	25-12
I^{2}C bus transmit data register	ICDRT	H'FF	H'FFFE E006	8	25-12
I^{2}C bus receive data register	ICDRR	H'FF	H'FFFE E007	8	25-13
I^{2}C bus NF2CYC register	ICNF2CYC	H'00	H'FFFE E008	8	25-14

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

25.3.1 I^{2}C Bus Control Register 1 (ICCR1)

The ICCR1 register controls the I²C bus interface 3 operate/stop state and transmit/receive operations and selects master or slave mode, transmit or receive, and the master mode transfer clock frequency.

I^{2}C Bus Control Register 1 (ICCR1)

<P4 address: location H'FFFE E000>

Bit:	7	6	5	4	3	2	1	0
	ICE	RCVD	MST	TRS	CKS			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	ICE	0	R	W	I^{2}C Bus Interface 3 Enable Bit 0: SCL and SDA output is prohibited (input to SCL and SDA is allowed). 1: This bit is enabled for transfer operations. (SCL and SDA pins are bus drive state.)
6	RCVD	0	R	W	Reception Disable Bit Enables or disables the next operation when TRS is "0" and the ICDRR register is read. 0: Enables next reception 1: Disables next reception

Bit	Abbreviation	After Reset	R	W	Description
5	MST	0	R	W	Master/Slave Select Bit
4	TRS	0	R	W	Transmit/Receive Select Bit In master mode with the I²C bus format, when arbitration is lost, MST and TRS bits are both reset by hardware, causing a transition to slave receive mode. Modification of the TRS bit should be made between transfer frames. When seven bits after the start condition is issued in slave receive mode match the slave address set to ICSAR and the 8th bit is set to "1", TRS bit is automatically set to "1". If an overrun error occurs in master receive mode with the clocked synchronous serial format, MST bit is cleared and the mode changes to slave receive mode. Operating modes are described below according to MST and TRS bits combination. When clocked synchronous serial format is selected and if MST = "1", clock is output. 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode
3 to 0	CKS	0000	R	W	Transfer Clock Select Bits These bits should be set according to the necessary transfer rate (table 25.3) in master mode. The transfer rate in master mode is determined by the combination of the CKS bit setting and the operating clock Pck provided to the IIC3 module. For this MCU, this document describes the case where Pck = 40 MHz.

Table 25.3 Transfer Rate

Bit 3	Bit 2	Bit 1	Bit 0		Transfer Rate (kHz)
CKS3	CKS2	CKS1	CKS0	Clock	Pck = 40 MHz
0	0	0	0	Pck/44	909
			1	Pck/52	769
		1	0	Pck/64	625
			1	Pck/72	556
	1	0	0	Pck/84	476
			1	Pck/92	434
		1	0	Pck/100	400
			1	Pck/108	370
1	0	0	0	Pck/176	227
			1	Pck/208	192
		1	0	Pck/256	156
			1	Pck/288	139
	1	0	0	Pck/336	119
			1	Pck/368	108
		1	0	Pck/400	100
			1	Pck/432	92.6

Note: • The settings should satisfy external specifications.

25.3.2 I^{2}C Bus Control Register 2 (ICCR2)

The ICCR2 register issues start/stop conditions, manipulates the SDA pin, monitors the SCL pin, and controls reset in the control part of the I^{2}C bus.

I^{2}C Bus Control Register 2 (ICCR2) <P4 address: location H'FFFE E001>

Bit:	7	6	5	4	3	2	1	0
	BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—
After Reset:	0	1	1	1	1	1	0	1

<After Reset: H'7D>

Bit	Abbreviation	After Reset	R	W	Description
7	BBSY	0	R	W	Bus Busy Flag Enables to confirm whether the I^{2}C bus is occupied or released and to issue start/stop conditions in master mode. With the clocked synchronous serial format, this bit is always read as "0". With the I^{2}C bus format, this bit is set to "1" when the SDA level changes from "H" to "L" under the condition of SCL = "H", assuming that the start condition has been issued. This bit is cleared to "0" when the SDA level changes from "L" to "H" under the condition of SCL = "H", assuming that the stop condition has been issued. Write "1" to BBSY flag and "0" to SCP flag to issue a start condition. Follow this procedure when also re-transmitting a start condition. Write "0" in BBSY flag and "0" in SCP bit to issue a stop condition.
6	SCP	1	R	W	Start/Stop Issue Condition Disable Bit Controls the issue of start/stop conditions in master mode. To issue a start condition, write "1" in BBSY flag and "0" in SCP bit. A retransmit start condition is issued in the same way. To issue a stop condition, write "0" in BBSY flag and "0" in SCP bit. This bit is always read as "1". Even if "1" is written to this bit, the data will not be stored.
5	SDAO	1	R	W	SDA Output Value Control Bit This bit is used with SDAOP bit when modifying output level of SDA bit. This bit should not be manipulated during transfer. 0: When reading, SDA pin outputs "L" level. When writing, SDA pin is changed to output "L" level. 1: When reading, SDA pin outputs "H" level. When writing, SDA pin is changed to output Hi-Z (outputs "H" level by external pull-up resistance).
4	SDAOP	1	R	W	SDAO Write Protect Bit Controls change of output level of the SDA pin by modifying the SDAO bit. To change the output level, clear SDAO and SDAOP bits to "0" or set SDAO bit to "1" and clear SDAOP bit to "0". This bit is always read as "1".
3	SCLO	1	R	0	SCL Output Level Flag Monitors SCL pin output level. 0: When read, SCL pin outputs "L" level. 1: When read, SCL pin outputs "H" level.
2	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".

Bit	Abbreviation	After Reset	R	W	Description
1	IICRST	0	R	W	IIC Control Part Reset Bit Resets the control part except for I^{2}C registers. If this bit is set to "1" when hang-up occurs because of communication failure during I^{2}C bus operation, some IIC3 registers and the control part can be reset.
0	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".

25.3.3 I^{2}C Bus Mode Register (ICMR)

The ICMR register selects whether the MSB or LSB is transferred first, and selects the transfer bit count. BC bit is initialized to H'0 by the IICRST bit in the ICCR2 register.

I^{2}C Bus Mode Register (ICMR) <P4 address: location H'FFFE E002>

Bit:	7	6	5	4	3	2	1	0
	MLS	—	—	—	BCWP	BC		
After Reset:	0	0	1	1	1	0	0	0

<After Reset: H'38>

Bit	Abbreviation	After Reset	R	W	Description
7	MLS	0	R	W	MSB-First/LSB-First Select Bit 0: MSB-first 1: LSB-first Set this bit to "0" when the I^{2}C bus format is used.
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	—	All 1	1	1	Reserved Bits These bits are always read as "1". The write value should always be "1".
3	BCWP	1	R	W	BC Write Protect Bit Controls the BC bit modifications. When modifying the BC bit, this bit should be cleared to "0". In clocked synchronous serial mode, the BC bit should not be modified. 0: When writing, values of the BC bit is set. 1: When reading, "1" is always read. When writing, settings of the BC bit is invalid.

Bit	Abbreviation	After Reset	R	W	Description
2 to 0	BC	000	R	W	Bit Counter Bits These bits specify the number of bits to be transferred next. When read, the remaining number of transfer bits is indicated. With the I^{2}C bus format, the data is transferred with one addition acknowledge bit. Should be made between transfer frames. If these bits are set to a value other than B'000, the setting should be made while the SCL pin is "L" level. The value returns to B'000 at the end of a data transfer, including the acknowledge bit. Also, after a stop condition is detected, this field will automatically be set to B'111. This field is cleared by a hardware reset and by setting the ICCR2 register IICRST bit to "1". With the clocked synchronous serial format, these bits should not be modified. I^{2}C Bus Format — Clocked Synchronous Serial Format 000: 9 bits — 000: 8 bits 001: 2 bits — 001: 1 bit 010: 3 bits — 010: 2 bits 011: 4 bits — 011: 3 bits 100: 5 bits — 100: 4 bits 101: 6 bits — 101: 5 bits 110: 7 bits — 110: 6 bits 111: 8 bits — 111: 7 bits

25.3.4 I^{2}C Bus Interrupt Enable Register (ICIER)

The ICIER register enables or disables interrupt sources and acknowledge bits, sets acknowledge bits to be transferred, and confirms acknowledge bits received.

I^{2}C Bus Interrupt Enable Register (ICIER) <P4 address: location H'FFFE E003>

Bit:	7	6	5	4	3	2	1	0
	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	TIE	0	R	W	Transmit Interrupt Enable Bit When the TDRE flag in the ICSR register is set, this bit enables or disables the transmit data empty interrupt (TXI). 0: Transmit data empty interrupt request (TXI) is disabled. 1: Transmit data empty interrupt request (TXI) is enabled.
6	TEIE	0	R	W	Transmit End Interrupt Enable Bit Enables or disables the transmit end interrupt (TEI) at the rising of the ninth clock while the TDRE flag in the ICSR register is "1". TEI can be canceled by clearing the TEND flag or the TEIE bit to "0". 0: Transmit end interrupt request (TEI) is disabled. 1: Transmit end interrupt request (TEI) is enabled.
5	RIE	0	R	W	Receive Interrupt Enable Bit Enables or disables the receive data full interrupt request (RXI) when receive data is transferred from the ICDRS register to the ICDRR register and the RDRF flag in the ICSR register is set to "1". RXI can be canceled by clearing the RDRF flag or RIE bit to "0". 0: Receive data full interrupt request (RXI) are disabled. 1: Receive data full interrupt request (RXI) are enabled.

Bit	Abbreviation	After Reset	R	W	Description
4	NAKIE	0	R	W	NACK Receive Interrupt Enable Bit When either the ICSR register NACKF or ALOVE flag is set, the NAKIE bit enables/disables the NACK detection and arbitration lost/overrun error interrupt request (NAKI). Note that the NAKI interrupt can be cleared either by clearing either the NACKF flag or the ALOVE flag to "0" or by clearing the NAKIE bit to "0". 0: NACK detection and arbitration lost/overrun error interrupt request (NAKI) is disabled. 1: NACK detection and arbitration lost/overrun error interrupt request (NAKI) is enabled.
3	STIE	0	R	W	Stop Condition Detection Interrupt Enable Bit Enables or disables the stop condition detection interrupt request (STPI) when the STOP flag in the ICSR register is set. 0: Stop condition detection interrupt request (STPI) is disabled. 1: Stop condition detection interrupt request (STPI) is enabled.
2	ACKE	0	R	W	Acknowledge Bit Judgment Select Bit 0: The value of the receive acknowledge bit is ignored, and continuous transfer is performed. 1: If the receive acknowledge bit is "1", continuous transfer is halted.
1	ACKBR	0	R	—	Receive Acknowledge Bit In transmit mode, this bit stores the acknowledge data that are returned by the receive device. This bit can be cleared by setting the BBSY flag in the ICCR2 register to "1". 0: Receive acknowledge = "0" 1: Receive acknowledge = "1"
0	ACKBT	0	R	W	Transmit Acknowledge Bit In receive mode, this bit specifies the bit to be sent at the acknowledge timing. 0: "0" is sent at the acknowledge timing. 1: "1" is sent at the acknowledge timing.

25.3.5 I^{2}C Bus Status Register (ICSR)

The ICSR register confirms interrupt request flags and their status.

I^{2}C Bus Status Register (ICSR) <P4 address: location H'FFFE E004>

Bit:	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	ALOVE	AAS	ADZ
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	TDRE	0	R	W	Transmit Data Register Empty Flag [Conditions for clearing to "0"] • When "0" is written in TDRE after reading TDRE = 1 • When data is written to ICDRT [Conditions for setting to "1"] • When data is transferred from the ICDRT register to the ICDRS register and the ICDRT register becomes empty • When the TRS bit in the ICCR1 register is set • When the start condition (including retransmission) is issued • When slave mode is changed from receive mode to transmit mode
6	TEND	0	R	W	Transmit End Flag [Conditions for clearing to "0"] • When "0" is written in TEND after reading TEND = "1" • When data is written to the ICDRT register [Conditions for setting to "1"] • When the ninth clock of SCL rises with the I^{2}C bus format while the TDRE flag is "1" • When the final bit of transmit frame is sent with the clocked synchronous serial format
5	RDRF	0	R	W	Receive Data Full Flag [Conditions for clearing to "0"] • When "0" is written in RDRF after reading RDRF = "1" • When the ICDRR register is read [Condition for setting to "1"] • When a receive data is transferred from the ICDRS register to the ICDRR register
4	NACKF	0	R	W	No Acknowledge Detection Flag [Condition for clearing to "0"] • When "0" is written in NACKF after reading NACKF = "1" [Condition for setting to "1"] • When no acknowledge is detected from the receive device in transmission while the ACKE bit in the ICIER register is "1"

Bit	Abbreviation	After Reset	R	W	Description
3	STOP	0	R	W	Stop Condition Detection Flag [Condition for clearing to "0"] • When "0" is written in STOP after reading STOP = "1" [Condition for setting to "1"] • When a stop condition is detected after frame transfer is completed
2	ALOVE	0	R	W	Arbitration Lost Flag/Overrun Error Flag Indicates that arbitration was lost in master mode with the I^{2}C bus format and that the final bit has been received while RDRF = "1" with the clocked synchronous format. When two or more master devices attempt to occupy the bus at nearly the same time, if the I^{2}C bus interface 3 detects SDA data differing from the data it sent, it sets ALOVE flag to "1" to indicate that the bus has been occupied by another master. [Condition for clearing to "0"] • When "0" is written in ALOVE after reading ALOVE = "1" [Conditions for setting to "1"] • If the internal SDA and SDA pin disagree at the rise of SCL in master transmit mode • When the SDA pin outputs "H" level in master mode while a start condition is detected • When the final bit is received with the clocked synchronous format while RDRF = "1"
1	AAS	0	R	W	Slave Address Recognition Flag In slave receive mode, this flag is set to "1" if the first frame following a start condition matches SVA bits in the ICSAR register. [Condition for clearing to "0"] • When "0" is written in AAS after reading AAS = "1" [Conditions for setting to "1"] • When the slave address is detected in slave receive mode • When the general call address is detected in slave receive mode.
0	ADZ	0	R	W	General Call Address Recognition Flag This bit is valid in slave receive mode with the I^{2}C bus format. [Condition for clearing to "0"] • When "0" is written in ADZ after reading ADZ = "1" [Condition for setting to "1"] • When the general call address is detected in slave receive mode

25.3.6 I^{2}C Bus Slave Address Register (ICSAR)

The ICSAR register selects the communications format and sets the slave address. In slave mode with the I^{2}C bus format, if the upper seven bits of ICSAR match the upper seven bits of the first frame received after a start condition, this module operates as the slave device.

I^{2}C Bus Slave Address Register (ICSAR) <P4 address: location H'FFFE E005>

Bit:	7	6	5	4	3	2	1	0
	SVA							FS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	SVA	All 0	R	W	Slave Address Bits These bits set a unique address in these bits, differing form the addresses of other slave devices connected to the I^{2}C bus.
0	FS	0	R	W	Format Select Bit 0: I^{2}C bus format is selected 1: Clocked synchronous serial format is selected

25.3.7 I^{2}C Bus Transmit Data Register (ICDRT)

When the ICDRT register detects the empty in the shift register (ICDRS), it transfers the transmit data which is written in the ICDRT register to the ICDRS register and starts transferring data. If the next transfer data is written to the ICDRT register while transferring data of the ICDRS register, continuous transfer is possible.

I^{2}C Bus Transmit Data Register (ICDRT) <P4 address: location H'FFFE E006>

Bit:	7	6	5	4	3	2	1	0
	ICDRT							
After Reset:	1	1	1	1	1	1	1	1

<After Reset: H'FF>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	ICDRT	All 1	R	W	Register to which the transmit data is written

25.3.8 I^{2}C Bus Receive Data Register (ICDRR)

When data of one byte is received, the ICDRR register transfers the receive data from the ICDRS register to the ICDRR register and the next data can be received. ICDRR register is a receive-only register, therefore the CPU cannot write to this register.

I^{2}C Bus Receive Data Register (ICDRR) <P4 address: location H'FFFE E007>

Bit:	7	6	5	4	3	2	1	0
	ICDRR							
After Reset:	1	1	1	1	1	1	1	1

<After Reset: H'FF>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	ICDRR	All 1	R	N	Register that stores the receive data

25.3.9 I^{2}C Bus Shift Register (ICDRS)

In transmission, data is transferred from the ICDRT register to the ICDRS register and the data is sent from the SDA pin. In reception, data is transferred from the ICDRS register to the ICDRR register after data of one byte is received. This register cannot be read directly from the CPU.

I^{2}C Bus Shift Register (ICDRS)

Bit:	7	6	5	4	3	2	1	0
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

25.3.10 I^{2}C Bus NF2CYC Register (ICNF2CYC)

The ICNF2CYC register selects the range of the noise filtering for the SCL and SDA pins. For details of the noise filter, see section 25.4.7, Noise Filter.

I^{2}C Bus NF2CYC Register (ICNF2CYC) <P4 address: location H'FFFE E008>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PRS	NF2 CYC
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	PRS	0	R	W	Pulse Width Ratio Select Bit Specifies the ratio of the "H" level period to the "L" level period for the SCL signal. 0: The ratio of "H" level to "L" level is 0.5 to 0.5. 1: The ratio of "H" level to "L" level is about 0.4 to 0.6.
0	NF2CYC	0	R	W	Noise Filtering Range Select Bit 0: The noise less than one cycle of the peripheral clock can be filtered out 1: The noise less than two cycles of the peripheral clock can be filtered out

25.4 Operation

The I²C bus interface 3 can communicate either in I²C bus mode or clocked synchronous serial mode by setting FS bit in the ICSAR register.

25.4.1 I²C Bus Format

Figure 25.3 shows the I²C bus formats. Figure 25.4 shows the I²C bus timing. The first frame following a start condition always consists of eight bits.

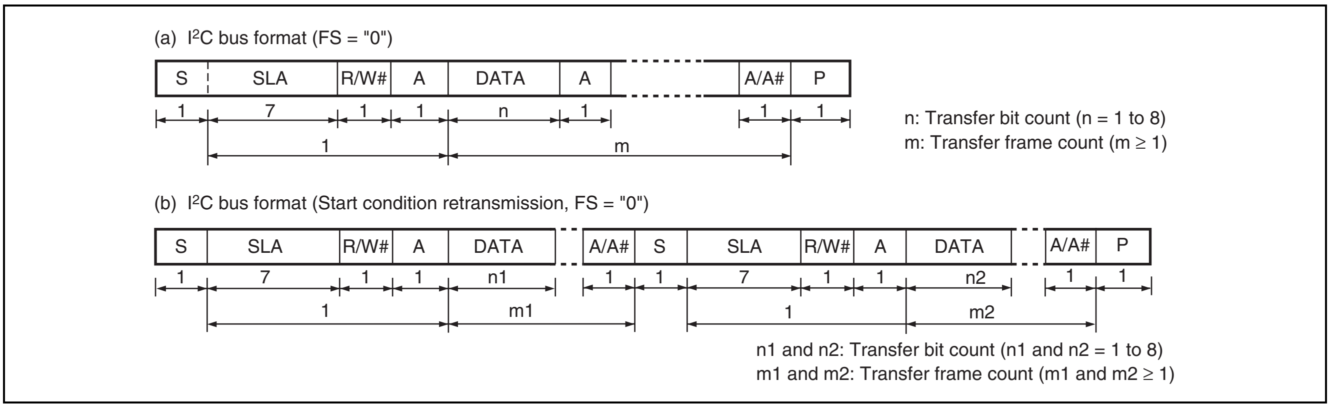

(a) I²C bus format (FS = "0")
S
SLA
R/W#
A
DATA
A
A/A#
P
1
7
1
1
n
1
1
1
1
m
n: Transfer bit count (n = 1 to 8)
m: Transfer frame count (m ≥ 1)
(b) I²C bus format (Start condition retransmission, FS = "0")
S
SLA
R/W#
A
DATA
A/A#
S
SLA
R/W#
A
DATA
A/A#
P
1
7
1
1
n1
1
1
7
1
1
n2
1
1
1
m1
1
m2
n1 and n2: Transfer bit count (n1 and n2 = 1 to 8)
m1 and m2: Transfer frame count (m1 and m2 ≥ 1)

Figure 25.3 I²C Bus Formats

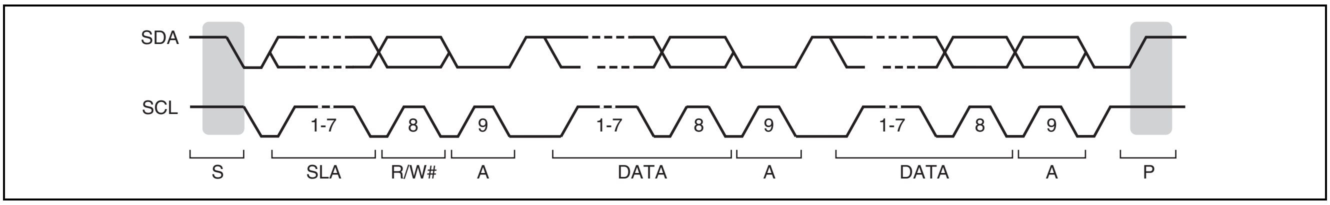


Figure 25.4 I²C Bus Timing

Legend:

S: Start condition. The master device drives SDA from "H" level to "L" level while SCL is "H" level.

SLA: Slave address

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 SDA to "L" level.

DATA: Transfer data

P: Stop condition. The master device drives SDA from "L" level to "H" level while SCL is "H" level.

25.4.2 Master Transmit Operation

In master transmit mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For master transmit mode operation timing, refer to figures 25.5 and 25.6. The transmission procedure and operations in master transmit mode are described below.

1. Set the ICE bit in the ICCR1 register to "1". Also, set CKS bit in ICCR1. (Initial setting)
2. Read the BBSY flag in the ICCR2 register to confirm that the bus is released. Set the MST and TRS bits in the ICCR1 register to select master transmit mode. Then, write "1" to BBSY and "0" to SCP. (Start condition issued) This generates the start condition.
3. After confirming that TDRE flag in the ICSR register has been set, write the transmit data (the first byte data show the slave address and R/W#) to the ICDRT register. At this time, TDRE flag is automatically cleared to "0", and data is transferred from the ICDRT register to the ICDRS register. TDRE flag is set again.
4. When transmission of one byte data is completed while TDRE flag is "1", TEND flag in the ICSR register is set to "1" at the rise of the 9th transmit clock pulse. Read the ACKBR bit in the ICIER register, and confirm that the slave device has been selected. Then, write second byte data to the ICDRT register. When ACKBR bit is "1", the slave device has not been acknowledged, so issue the stop condition. To issue the stop condition, write "0" to BBSY and SCP. SCL is fixed "L" level until the transmit data is prepared or the stop condition is issued.
5. The transmit data after the second byte is written to the ICDRT register every time TDRE flag is set.
6. Write the number of bytes to be transmitted to the ICDRT register. Wait until TEND flag is set (the end of last byte data transmission) while TDRE is "1", or wait for NACK (NACKF bit in the ICSR register = "1") from the receive device while ACKE bit in the ICIER register is "1". Then, issue the stop condition to clear TEND flag or NACKF flag.
7. When the STOP flag in the ICSR register is set to "1", the operation returns to the slave receive mode.

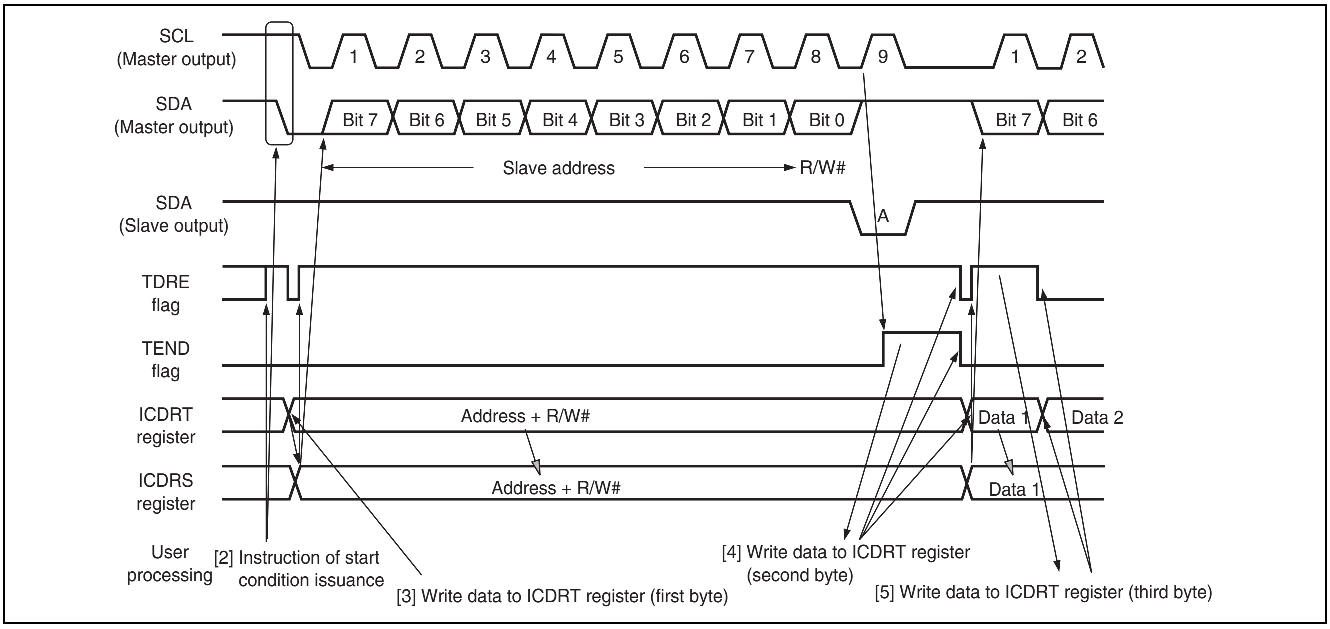


Figure 25.5 Master Transmit Mode Operation Timing (1)

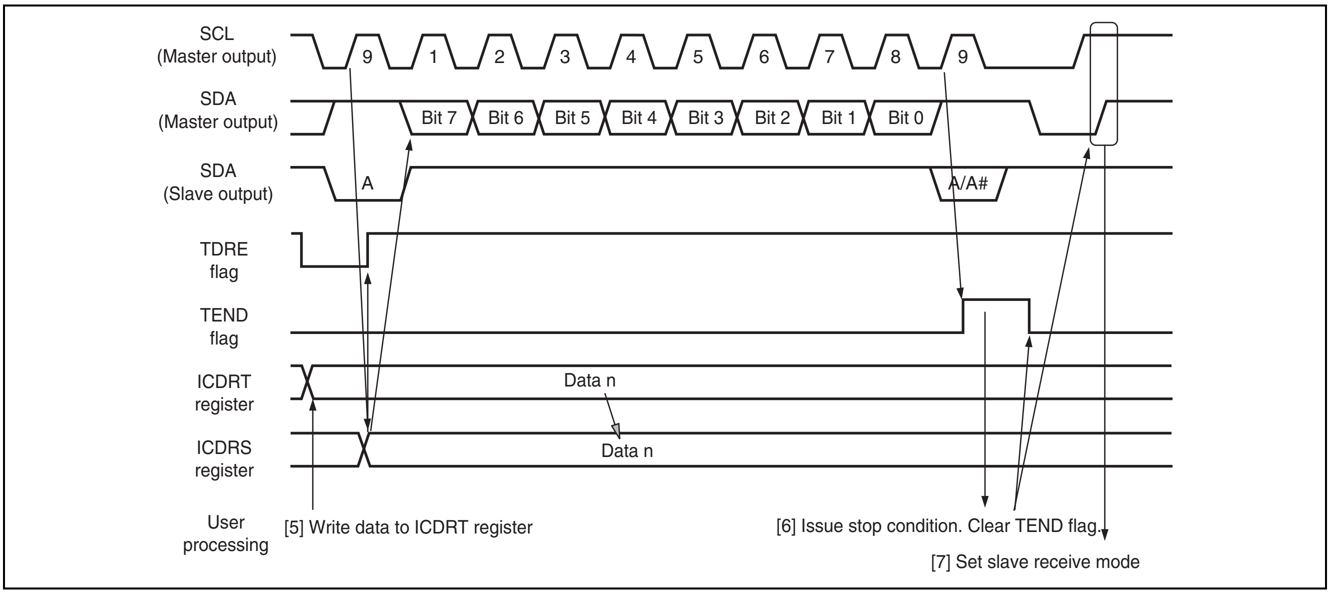


Figure 25.6 Master Transmit Mode Operation Timing (2)

25.4.3 Master Receive Operation

In master receive mode, the master device outputs the receive clock, receives data from the slave device, and returns an acknowledge signal. For master receive mode operation timing, refer to figures 25.7 and 25.8. The reception procedure and operations in master receive mode are shown below.

1. Clear the TEND flag bit in ICSR to "0", then clear the TRS bit in the ICCR1 register to "0" to switch from master transmit mode to master receive mode. Then, clear the TDRE flag to "0".
2. When the ICDRR register is read (dummy data read)*[1], reception is started, the receive clock is output, and data received, in synchronization with the internal clock. The master device outputs the level specified by ACKBT bit in the ICIER register to SDA, at the 9th receive clock pulse.
3. After the reception of first frame data is completed, the RDRF flag in the ICSR register is set to "1" at the rise of 9th receive clock pulse. At this time, the receive data is read by reading the ICDRR register, and RDRF flag is cleared to "0".
4. The continuous reception is performed by reading the ICDRR register every time RDRF flag is set. If 8th receive clock pulse falls after reading the ICDRR register by the other processing while RDRF flag is "1", SCL is fixed "L" level until the ICDRR register is read.
5. If next frame is the last receive data, set the RCVD bit in the ICCR1 register to "1" before reading the ICDRR register. This enables the issuance of the stop condition after the next reception.
6. When the RDRF flag is set to "1" at rise of the 9th receive clock pulse, issue the stop condition.
7. When the STOP flag in the ICSR register is set to "1", read the ICDRR register. Then clear the RCVD bit to "0".
8. The operation returns to the slave receive mode.

Note: *1 If only one byte is received, read the ICDRR register (dummy-read) after the RCVD bit in the ICCR1 register is set.

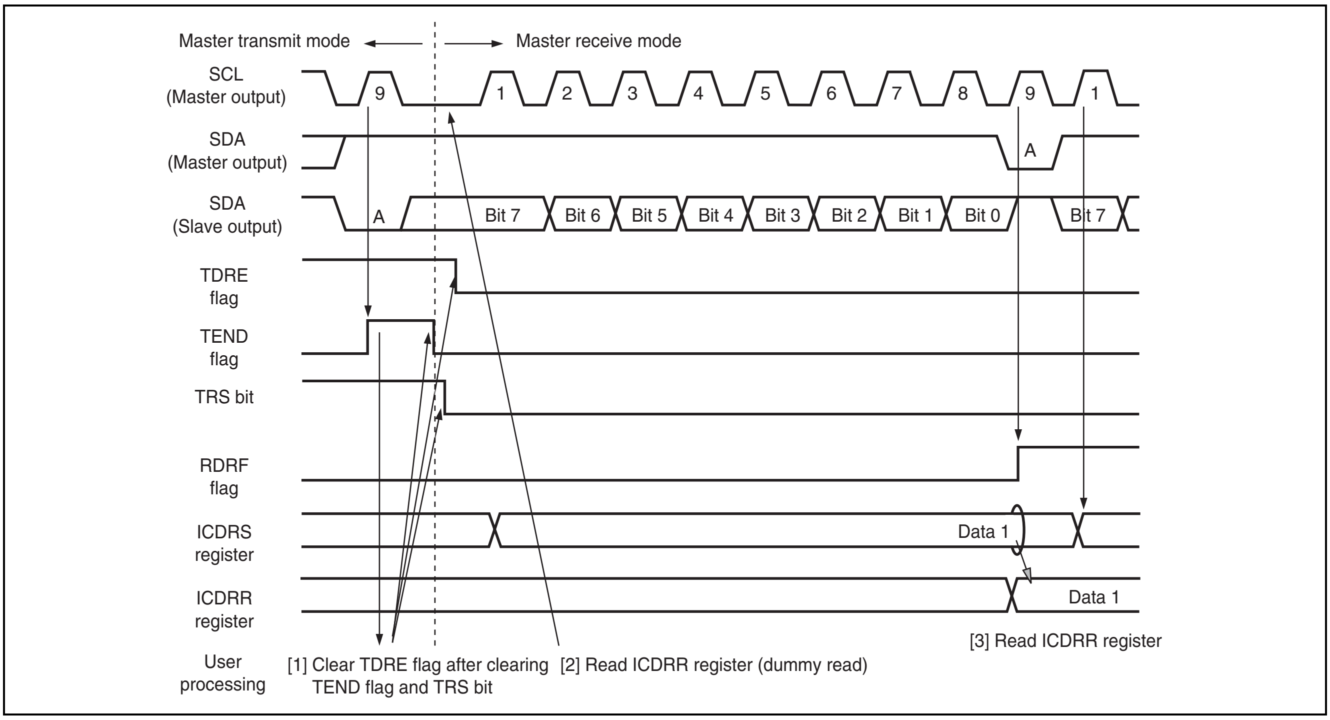


Figure 25.7 Master Receive Mode Operation Timing (1)

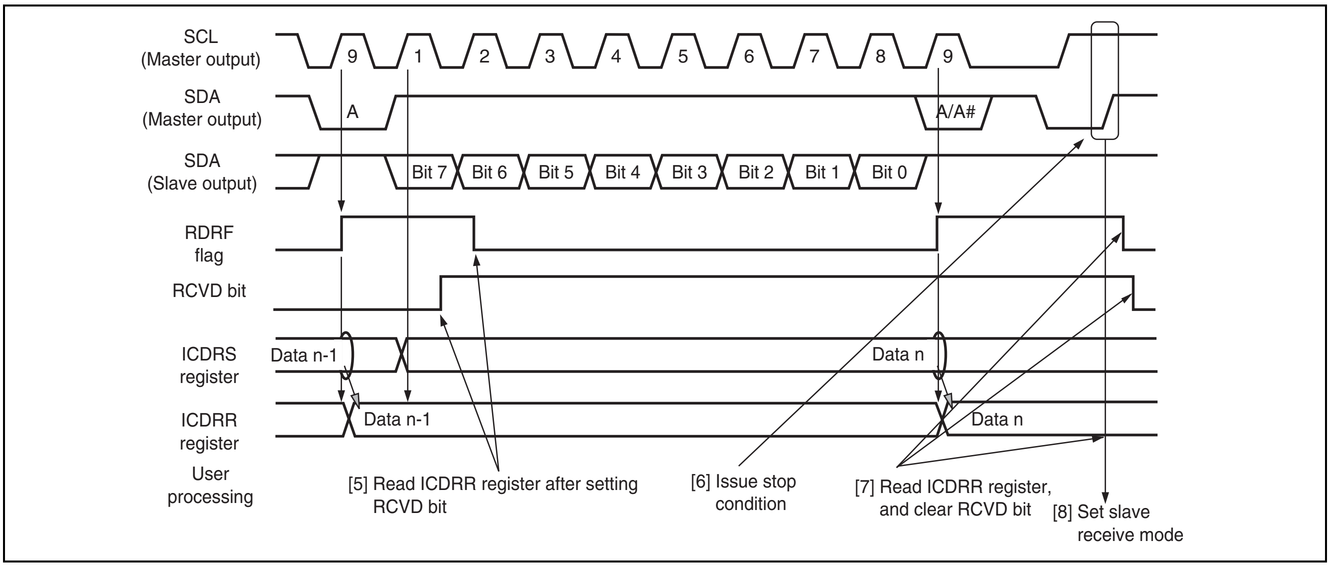


Figure 25.8 Master Receive Mode Operation Timing (2)

25.4.4 Slave Transmit Operation

In slave transmit mode, the slave device outputs the transmit data, while the master device outputs the receive clock and returns an acknowledge signal. For slave transmit mode operation timing, refer to figures 25.9 and 25.10.

The transmission procedure and operations in slave transmit mode are described below.

1. Set the ICE bit in the ICCR1 register to "1". Set CKS bit in the ICCR1 register. (Initial setting)
   Set the MST and TRS bits in the ICCR1 register to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT bit in the ICIER register to SDA, at the rise of the 9th clock pulse. At this time, if the 8th bit data (R/W) is "1", the TRS bit in the ICCR1 register and the TDRE flag in the ICSR register are set to "1", and the mode changes to slave transmit mode automatically. The continuous transmission is performed by writing transmit data to the ICDRT register every time TDRE flag is set.
3. If TDRE is set after writing last transmit data to the ICDRT register, wait until TEND flag in the ICSR register is set to "1", with TDRE flag = "1". When TEND flag is set, clear TEND flag.
4. Clear TRS bit for the end processing, and read the ICDRR register (dummy read). SCL is opened.
5. Clear TDRE flag.

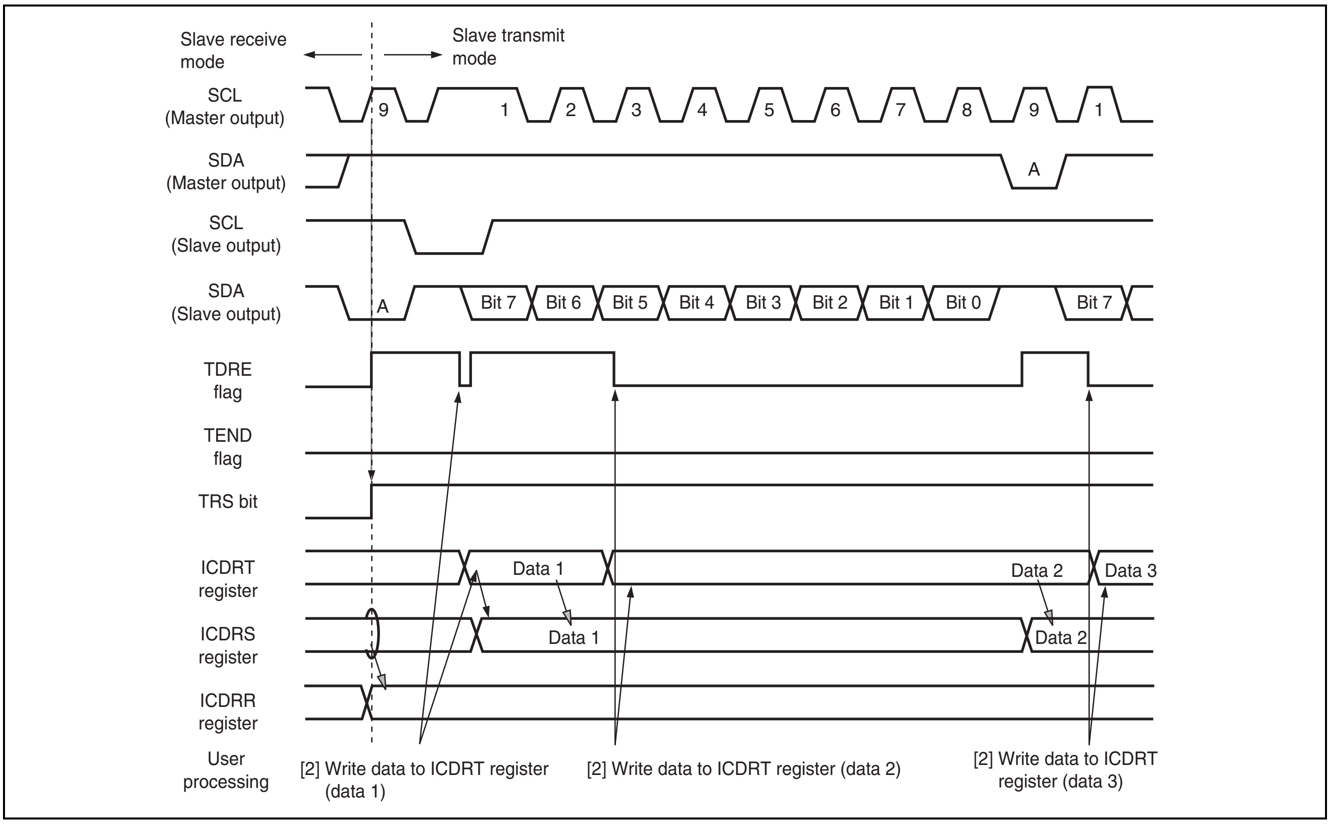


Figure 25.9 Slave Transmit Mode Operation Timing (1)

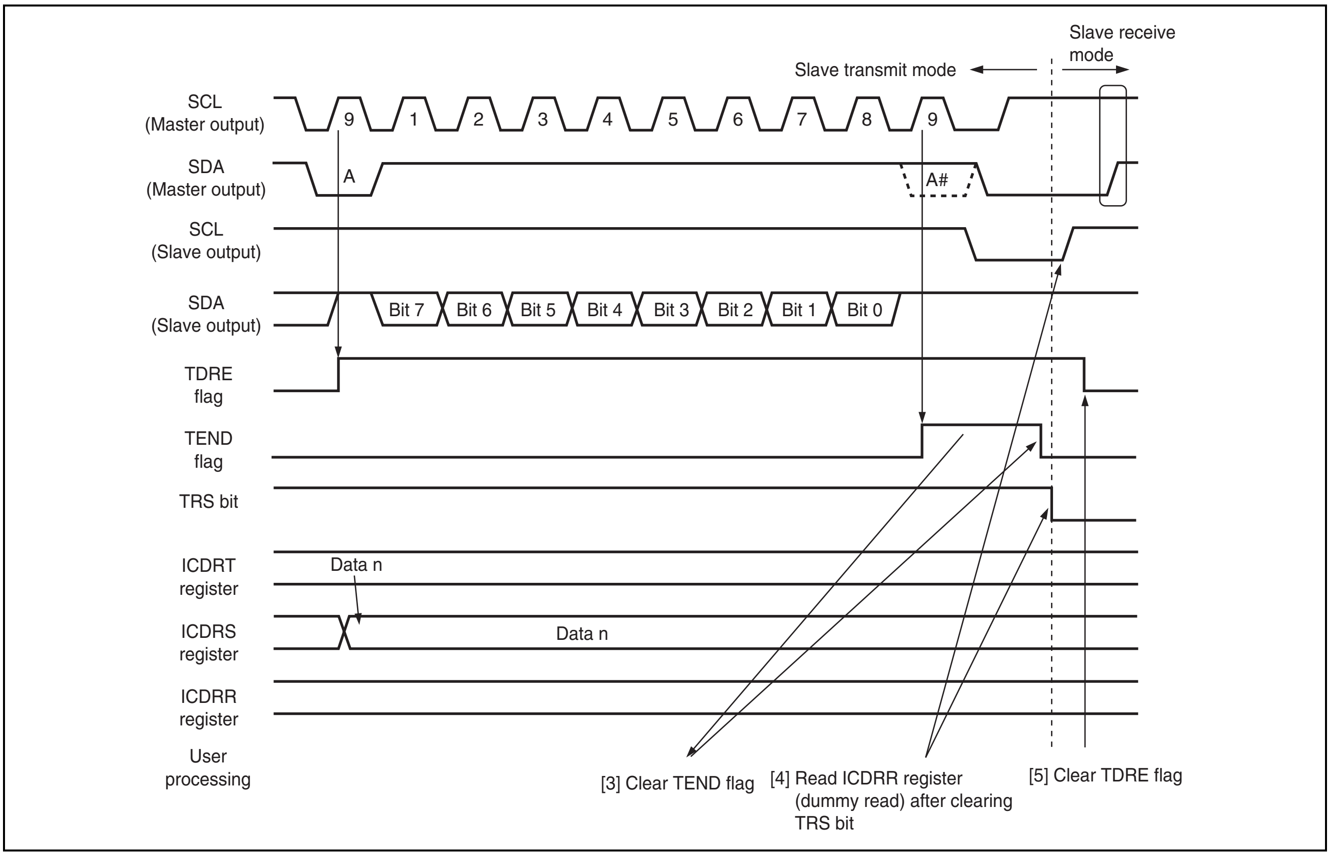


Figure 25.10 Slave Transmit Mode Operation Timing (2)

25.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For slave receive mode operation timing, refer to figures 25.11 and 25.12. The reception procedure and operations in slave receive mode are described below.

1. Set the ICE bit in the ICCR1 register to "1". Set CKS bit in the ICCR1 register. (Initial setting)
   Set the MST and TRS bits in the ICCR1 register to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT bit in the ICIER register to SDA, at the rise of the 9th clock pulse. At the same time, since RDRF flag in the ICSR register is set, read the ICDRR register (dummy read). (Since the read data show the slave address and R/W, it is not used.)
3. Read the ICDRR register every time RDRF flag is set. If 8th receive clock pulse falls while RDRF flag is "1", SCL is fixed "L" level until the ICDRR register is read. The change of the acknowledge before reading the ICDRR register, to be returned to the master device, is reflected to the next transmit frame.
4. The last byte data is read by reading the ICDRR register.

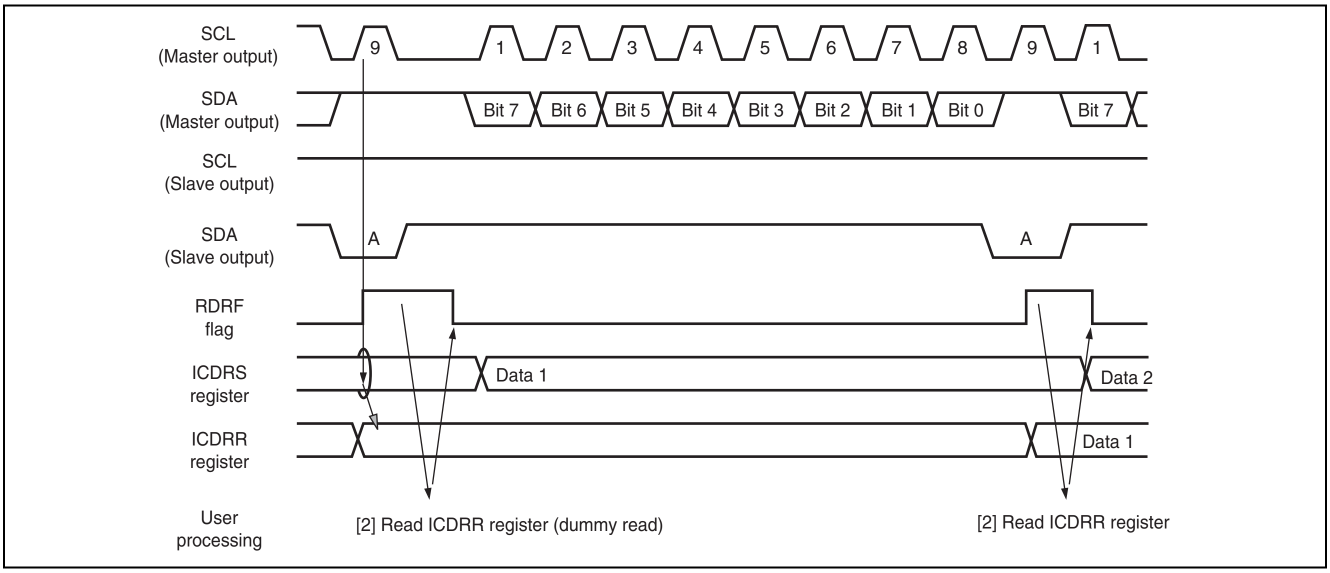


Figure 25.11 Slave Receive Mode Operation Timing (1)

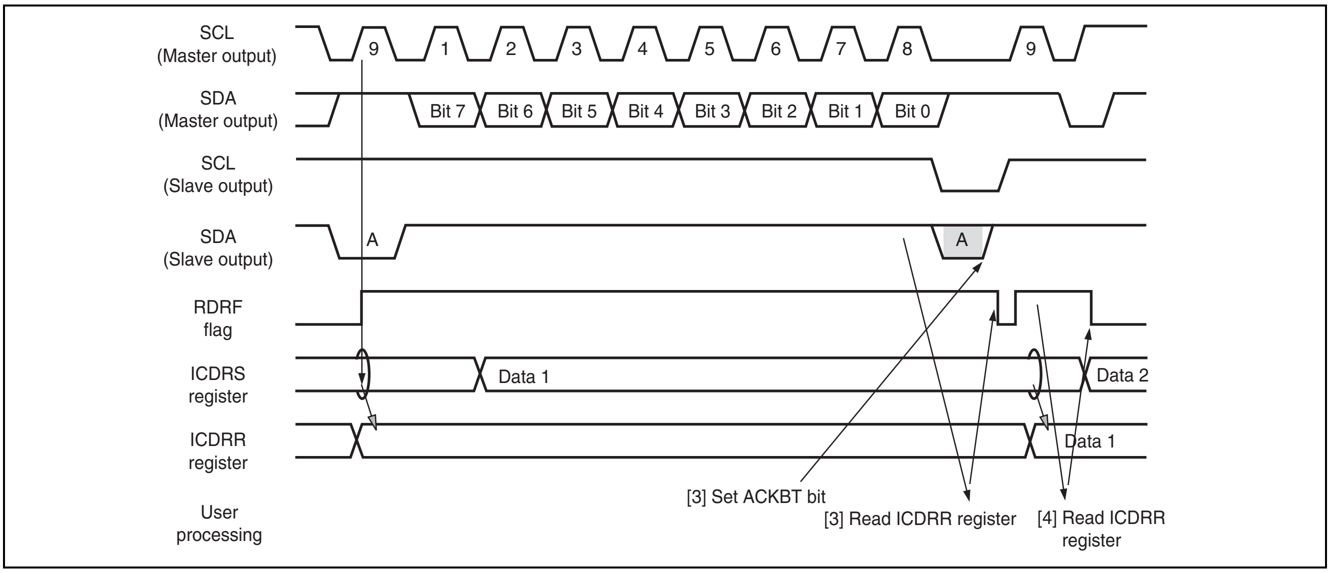


Figure 25.12 Slave Receive Mode Operation Timing (2)

25.4.6 Clocked Synchronous Serial Format

This module can be operated with the clocked synchronous serial format, by setting the FS bit in the ICSAR register to "1". When the MST bit in the ICCR1 register is "1", the transfer clock output from SCL is selected. When MST bit is "0", the external clock input is selected.

(1) Data Transfer Format

Figure 25.13 shows the clocked synchronous serial transfer format.

The transfer data is output from the fall to the fall of the SCL clock, and the data at the rising edge of the SCL clock is guaranteed. The MLS bit in the ICMR register sets the order of data transfer, in either the MSB first or LSB first. The output level of SDA can be changed during the transfer wait, by the SDAO bit in the ICCR2 register.

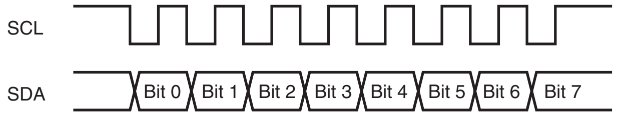


Figure 25.13 Clocked Synchronous Serial Transfer Format

(2) Transmit Operation

In transmit mode, transmit data is output from SDA, in synchronization with the fall of the transfer clock. The transfer clock is output when MST bit in the ICCR1 register is "1", and is input when MST bit is "0". For transmit mode operation timing, refer to figure 25.14. The transmission procedure and operations in transmit mode are described below.

1. Set the ICE bit in the ICCR1 register to 1. Set the MST and CKS bit in ICCR1 register. (Initial setting)
2. Set the TRS bit in the ICCR1 register to select the transmit mode. Then, TDRE flag in the ICSR register is set.
3. Confirm that TDRE flag has been set. Then, write the transmit data to the ICDRT register. The data is transferred from the ICDRT register to the ICDRS register, and TDRE flag is set automatically. The continuous transmission is performed by writing data to the ICDRT register every time TDRE flag is set. When changing from transmit mode to receive mode, clear TRS bit while TDRE flag is "1".

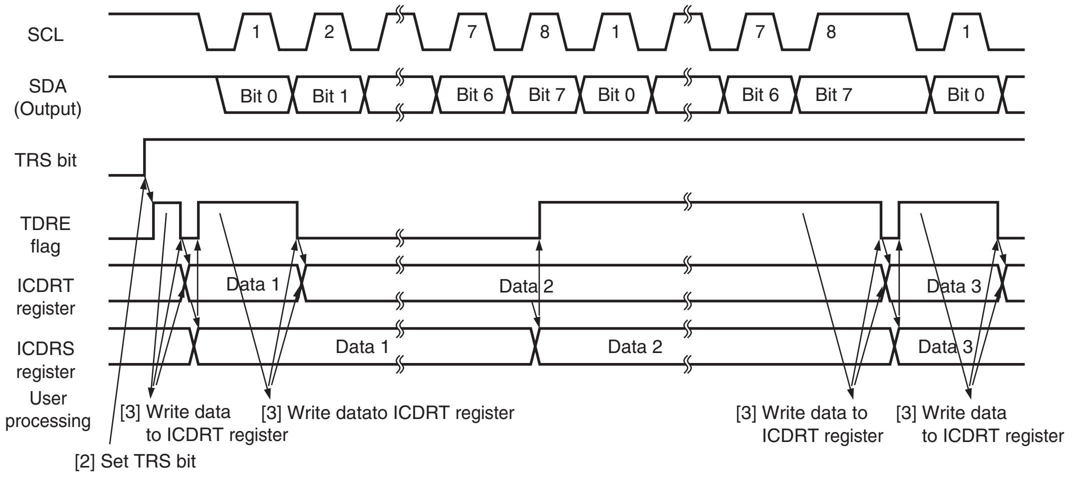


Figure 25.14 Transmit Mode Operation Timing

(3) Receive Operation

In receive mode, data is latched at the rise of the transfer clock. The transfer clock is output when MST in the ICCR1 register is "1", and is input when MST bit is "0". For receive mode operation timing, refer to figure 25.15. The reception procedure and operations in receive mode are described below.

1. Set the ICE bit in the ICCR1 register to "1". Set CKS bit in the ICCR1 register. (Initial setting)
2. When the transfer clock is output, set MST bit to "1" to start outputting the receive clock.
3. When the receive operation is completed, data is transferred from the ICDRS register to the ICDRR register and RDRF flag in the ICSR register is set. When MST bit = "1", the next byte can be received, so the clock is continually output. The continuous reception is performed by reading the ICDRR register every time RDRF flag is set. When the 8th clock is risen while RDRF flag is "1", the overrun is detected and ALOVE bit in the ICSR register is set. At this time, the previous reception data is retained in the ICDRR register.
4. To stop receiving when MST bit = "1xx", set RCVD bit in the ICCR1 register to "1", then read the ICDRR register. Then, SCL is fixed "H" level after receiving the next byte data.

Notes: • Follow the steps below to receive only one byte with MST = "1" specified. See figure 25.16 for the operation timing.

1. Set the ICE bit in the ICCR1 register to "1". Set CKS bit in the ICCR1 register. (Initial setting)
2. Set MST bit = 1 while the RCVD bit in the ICCR1 register is "0". This causes the receive clock to be output.
3. Check if the BC bit in the ICMR register is set to "1xx" and then set the RCVD bit in the ICCR1 register to "1". This causes the SCL to be fixed to the "H" level after outputting one byte of the receive clock.

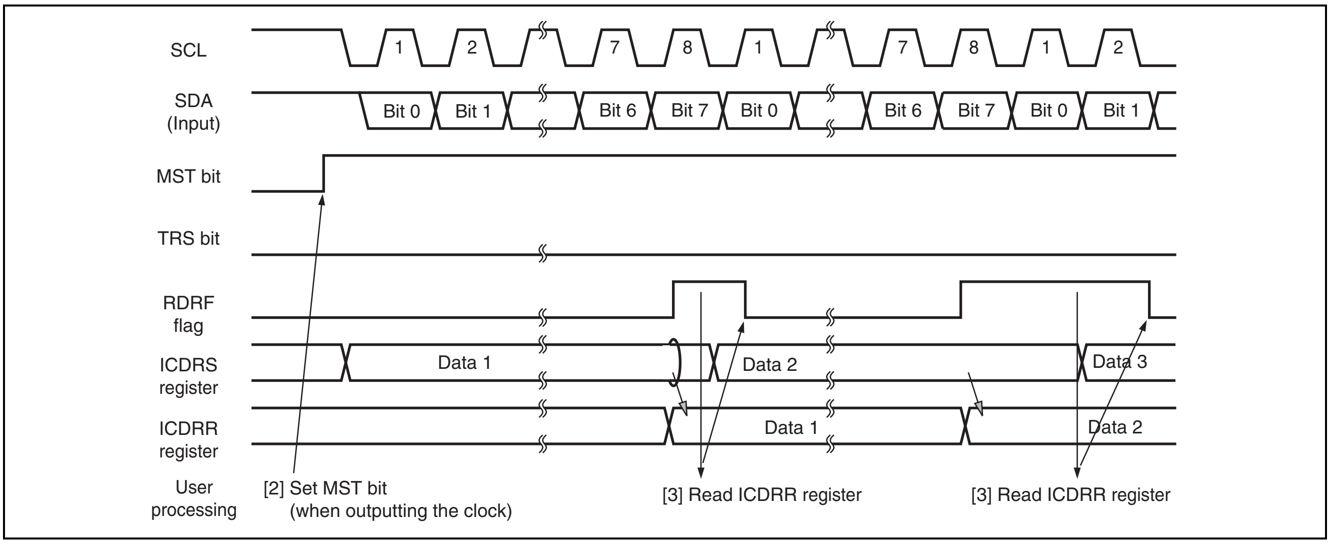


Figure 25.15 Receive Mode Operation Timing

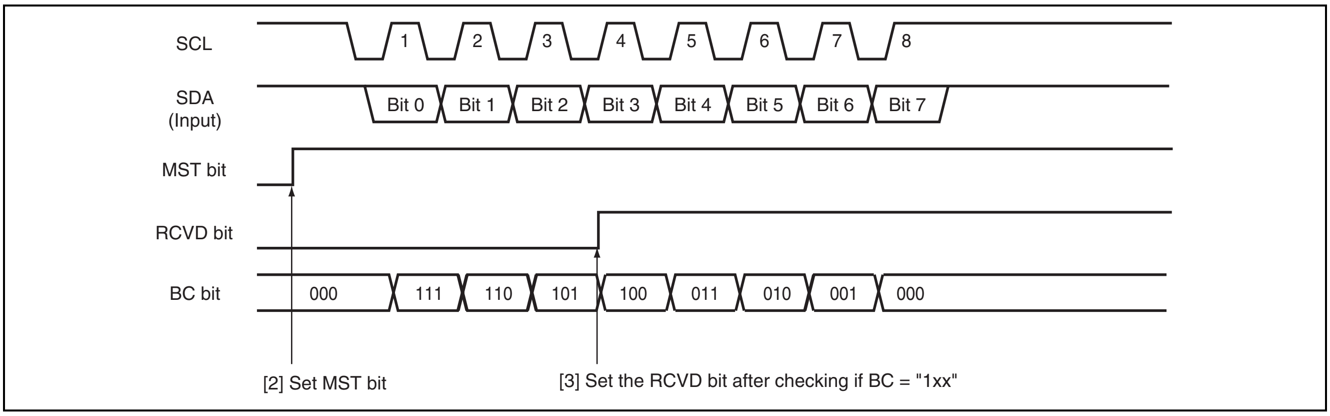


Figure 25.16 Operation Timing For Receiving One Byte (MST = "1")

25.4.7 Noise Filter

The logic levels at the SCL and SDA pins are routed through noise filters before being latched internally. Figure 25.17 shows a block diagram of the noise filter circuit.

The noise filter consists of three cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on the peripheral clock. When ICNF2CYC is set to "0", this signal is passed forward to the next circuit if the outputs of both latches agree. When ICNF2CYC is set to "1", this signal is passed forward to the next circuit if the outputs of three latches agree. If they do not agree, the previous value is held.

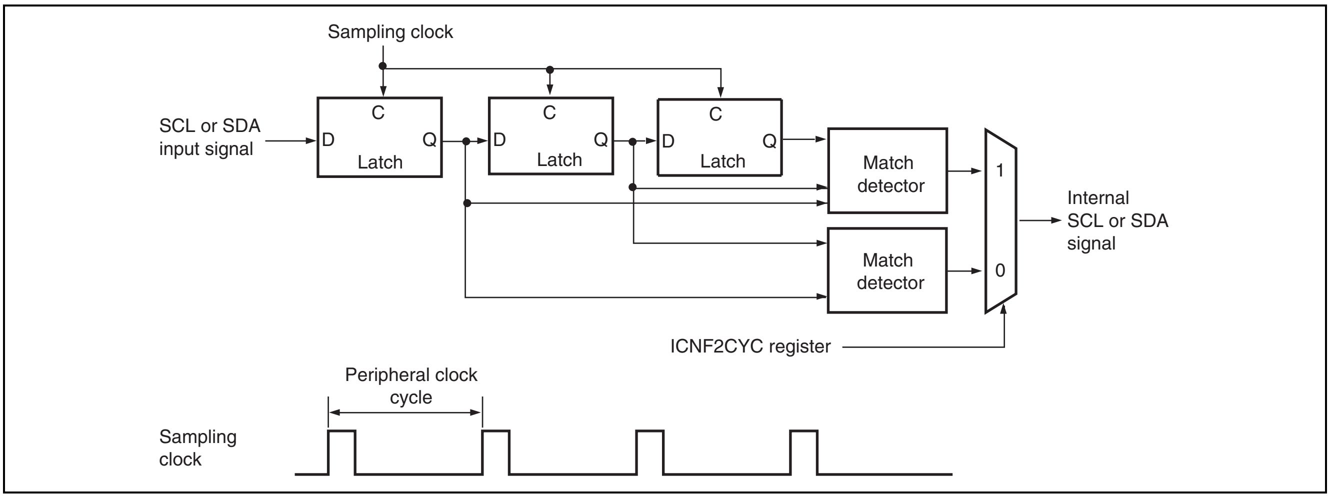
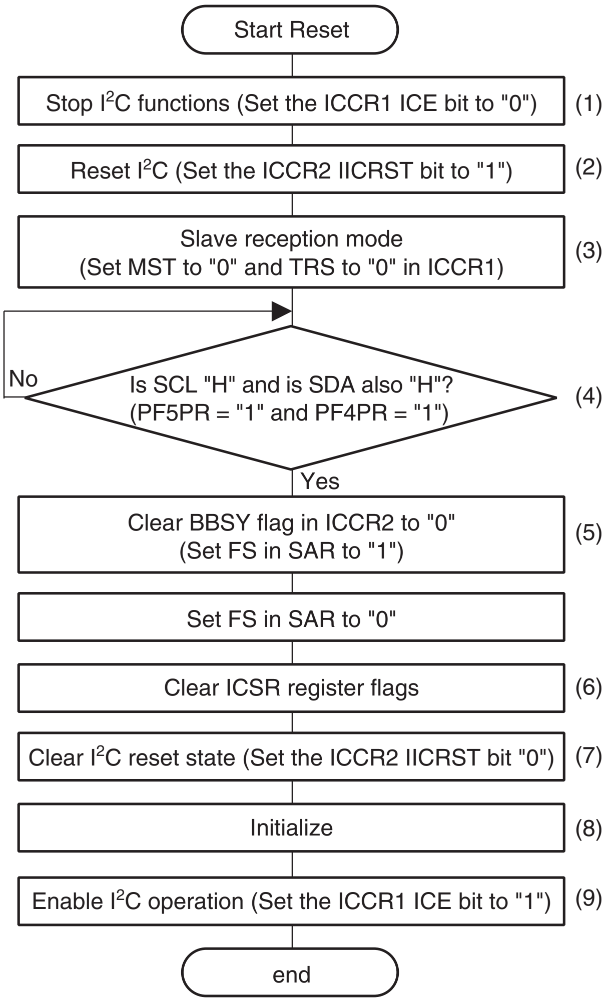


Figure 25.17 Block Diagram of Noise Filter

25.4.8 I²C Bus Interface 3 Reset by the IICRST Bit

For the I²C bus interface 3, certain of the I²C registers and the control block can be reset by writing "1" to the ICCR2 register IICRST bit. Figure 25.18 shows an example of a procedure for resetting I²C bus interface 3 using the IICRST bit.


Start Reset
Stop I^{2}C functions (Set the ICCR1 ICE bit to "0")
(1)
Reset I^{2}C (Set the ICCR2 IICRST bit to "1")
(2)
Slave reception mode
(Set MST to "0" and TRS to "0" in ICCR1)
(3)
No
Is SCL "H" and is SDA also "H"?
(PF5PR = "1" and PF4PR = "1")
(4)
Yes
Clear BBSY flag in ICCR2 to "0"
(Set FS in SAR to "1")
(5)
Set FS in SAR to "0"
Clear ICSR register flags
(6)
Clear I^{2}C reset state (Set the ICCR2 IICRST bit "0")
(7)
Initialize
(8)
Enable I^{2}C operation (Set the ICCR1 ICE bit to "1")
(9)
end

(1) Write "0" to the ICCR1 register ICE bit to stop the I^{2}C bus interface 3 functions.

(2) Write "1" to the ICCR2 register IICRST bit to reset certain of the registers and the control block in I^{2}C bus interface 3. The ICCR2 register BBSY flag goes to the undefined state at this time.

(3) Write "0" to the ICCR1 register MST bit and "0" to the TRS bit to switch to slave reception mode.

(4) Wait until the bus is in the bus released state. The bus released state can be recognized by a variety of means, including reading the I/O ports corresponding to SCL and SDA (the PF5PR and PF4PR bits in the PFPR register).

(5) Clear the ICCR2 register BBSY flag to "0" by writing "1" to the SAR register FS bit. After clearing the BBSY flag to "0", write 0 to the FS bit.

(6) Clear the ICSR register flags (TEND, RDRF, NACKF, STOP, ALOVE, AAS, and ADZ) to "0".

(7) Clear the I^{2}C reset state by writing "0" to the IICRST bit.

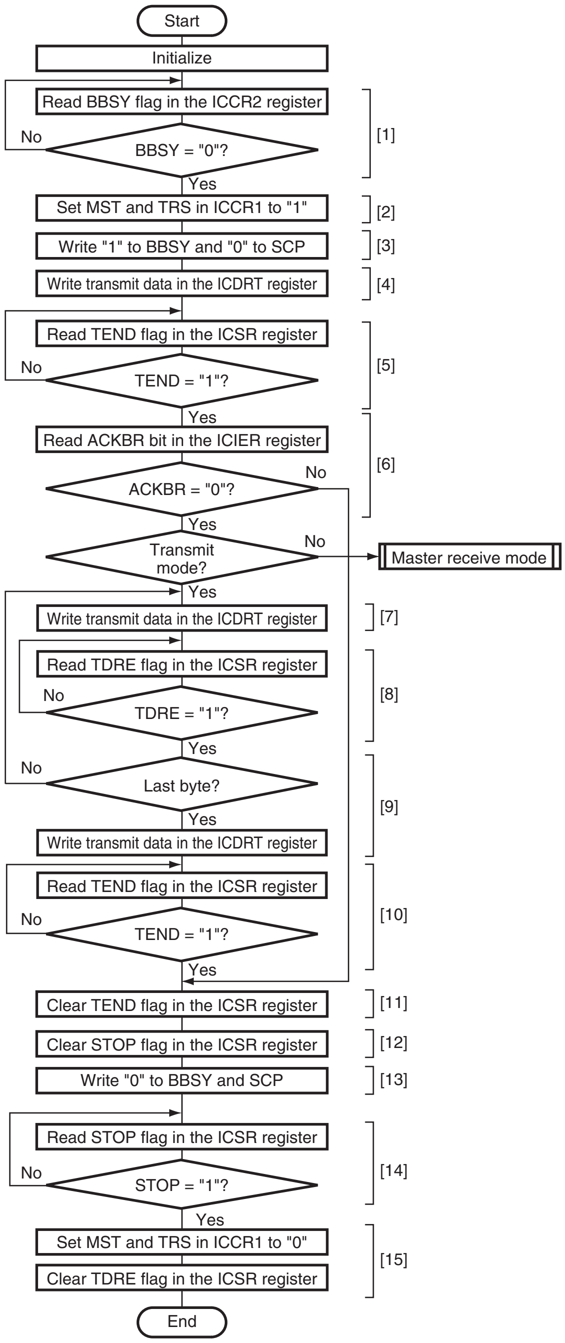
(8) Initialize the I^{2}C registers (ICCR1, ICCR2, ICMR, ICIER, SAR, and NF2CYC).

(9) Enable transfer operations by writing "1" to the ICCR1 register ICE bit.

Figure 25.18 I²C Bus Interface 3 Reset Procedure Example Using the IICRST BIT

25.4.9 Example of Use

Flowcharts in respective modes that use the I^{2}C bus interface 3 are shown in figures 25.19 to 25.22.


Start
Initialize
Read BBSY flag in the ICCR2 register
No
BBSY = "0"?
Yes
[1]
Set MST and TRS in ICCR1 to "1"
[2]
Write "1" to BBSY and "0" to SCP
[3]
Write transmit data in the ICDRT register
[4]
Read TEND flag in the ICSR register
No
TEND = "1"?
Yes
[5]
Read ACKBR bit in the ICIER register
ACKBR = "0"?
No
Yes
[6]
Transmit mode?
No
Master receive mode
Yes
Write transmit data in the ICDRT register
[7]
Read TDRE flag in the ICSR register
No
TDRE = "1"?
Yes
[8]
No
Last byte?
Yes
Write transmit data in the ICDRT register
[9]
Read TEND flag in the ICSR register
No
TEND = "1"?
Yes
[10]
Clear TEND flag in the ICSR register
[11]
Clear STOP flag in the ICSR register
[12]
Write "0" to BBSY and SCP
[13]
Read STOP flag in the ICSR register
No
STOP = "1"?
Yes
[14]
Set MST and TRS in ICCR1 to "0"
Clear TDRE flag in the ICSR register
[15]
End

[1] Test the status of the SCL and SDA lines.

[2] Set master transmit mode.

[3] Issue the start condition.

[4] Set the first byte (slave address + R/W#) of transmit data.

[5] Wait for 1 byte to be transmitted.

[6] Test the acknowledge transferred from the specified slave device.

[7] Set the second and subsequent bytes (except for the last byte) of transmit data.

[8] Wait for the ICDRT register empty.

[9] Set the last byte of transmit data.

[10] Wait for last byte to be transmitted.

[11] Clear the TEND flag.

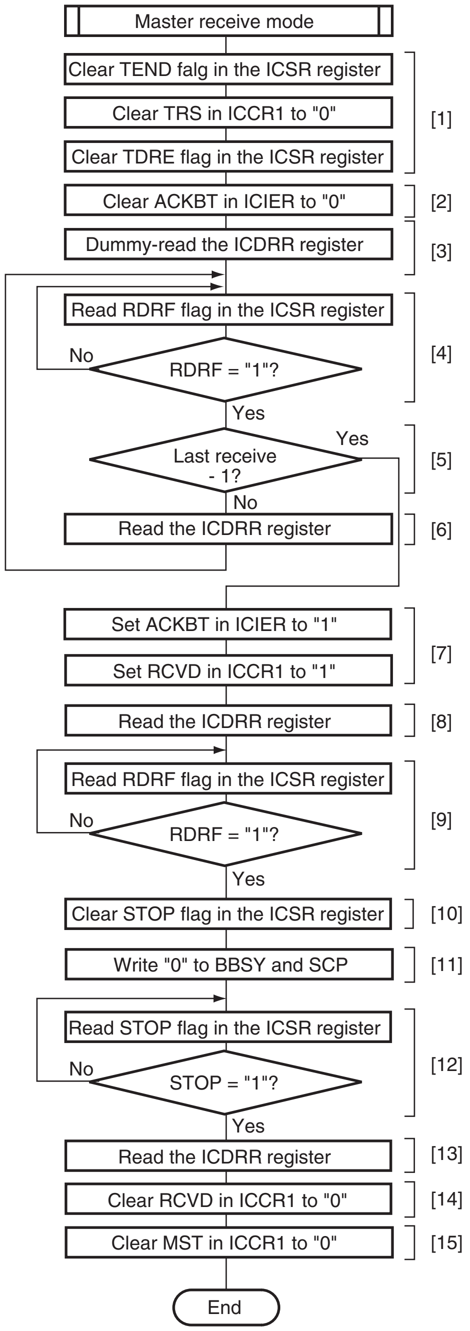
[12] Clear the STOP flag.

[13] Issue the stop condition.

[14] Wait for the creation of stop condition.

[15] Set slave receive mode. Clear TDRE flag.

Figure 25.19 Sample Flowchart for Master Transmit Mode


Master receive mode
Clear TEND falg in the ICSR register
Clear TRS in ICCR1 to "0"
Clear TDRE flag in the ICSR register
[1]
Clear ACKBT in ICIER to "0"
[2]
Dummy-read the ICDRR register
[3]
Read RDRF flag in the ICSR register
No
RDRF = "1"?
[4]
Yes
Yes
Last receive - 1?
[5]
No
Read the ICDRR register
[6]
Set ACKBT in ICIER to "1"
Set RCVD in ICCR1 to "1"
[7]
Read the ICDRR register
[8]
Read RDRF flag in the ICSR register
No
RDRF = "1"?
[9]
Yes
Clear STOP flag in the ICSR register
[10]
Write "0" to BBSY and SCP
[11]
Read STOP flag in the ICSR register
No
STOP = "1"?
[12]
Yes
Read the ICDRR register
[13]
Clear RCVD in ICCR1 to "0"
[14]
Clear MST in ICCR1 to "0"
[15]
End

[1] Clear TEND flag, select master receive mode, and then clear TDRE flag.*1

[2] Set acknowledge to the transmit device.*1

[3] Dummy-read the ICDDR register.*1

[4] Wait for 1 byte to be received*2

[5] Check whether it is the (last receive - 1).*2

[6] Read the receive data.

[7] Set acknowledge of the final byte. Disable continuous reception (RCVD = 1).*2

[8] Read the (last byte - 1) of received data.

[9] Wait for the last byte to be receive.

[10] Clear the STOP flag.

[11] Issue the stop condition.

[12] Wait for the creation of stop condition.

[13] Read the last byte of receive data.

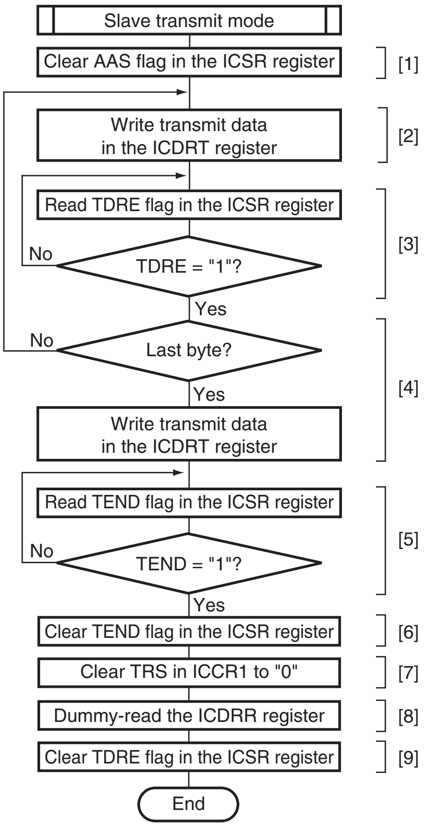
[14] Clear RCVD bit.

[15] Set slave receive mode.

Notes: *1 Make sure that no interrupt will be generated during steps [1] to [3].
*2 During the last receive - 1 (when condition [5] met), assure that no interrupts are generated during steps [4], [5], and [7].

- When the size of receive data is only one byte in reception, steps [2] to [6] are skipped after step [1], before jumping to step [7]. The step [8] is dummy-read in the ICDRR register.

Figure 25.20 Sample Flowchart for Master Receive Mode


Slave transmit mode
Clear AAS flag in the ICSR register
[1]
Write transmit data in the ICDRT register
[2]
Read TDRE flag in the ICSR register
No
TDRE = "1"?
Yes
[3]
No
Last byte?
Yes
Write transmit data in the ICDRT register
[4]
Read TEND flag in the ICSR register
No
TEND = "1"?
Yes
[5]
Clear TEND flag in the ICSR register
[6]
Clear TRS in ICCR1 to "0"
[7]
Dummy-read the ICDRR register
[8]
Clear TDRE flag in the ICSR register
[9]
End

[1] Clear the AAS flag.

[2] Set transmit data for the ICDRT register (except for the last byte).

[3] Wait for the ICDRT register empty.

[4] Set the last byte of transmit data.

[5] Wait for the last byte to be transmitted.

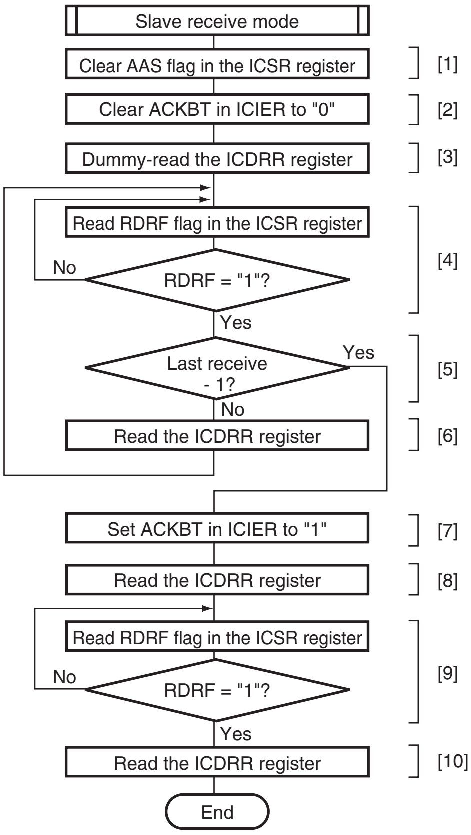
[6] Clear the TEND flag.

[7] Set slave receive mode.

[8] Dummy-read the ICDRR register to release the SCL.

[9] Clear the TDRE flag.

Figure 25.21 Sample Flowchart for Slave Transmit Mode


Slave receive mode
Clear AAS flag in the ICSR register
[1]
Clear ACKBT in ICIER to "0"
[2]
Dummy-read the ICDRR register
[3]
Read RDRF flag in the ICSR register
No
RDRF = "1"?
[4]
Yes
Last receive - 1?
Yes
[5]
No
Read the ICDRR register
[6]
Set ACKBT in ICIER to "1"
[7]
Read the ICDRR register
[8]
Read RDRF flag in the ICSR register
No
RDRF = "1"?
[9]
Yes
Read the ICDRR register
[10]
End

[1] Clear the AAS flag.

[2] Set acknowledge to the transmit device.

[3] Dummy-read the ICDRR register.

[4] Wait for 1 byte to be received.

[5] Check whether it is the (last receive - 1).

[6] Read the receive data.

[7] Set acknowledge of the last byte.

[8] Read the (last byte - 1) of receive data.

[9] Wait the last byte to be received.

[10] Read for the last byte of receive data.

Note: • When the size of receive data is only one byte in reception, steps [2] to [6] are skipped after step [1], before jumping to step [7]. The step [8] is dummy-read in the ICDRR register.

Figure 25.22 Sample Flowchart for Slave Receive Mode

25.5 Interrupt Requests

There are six interrupt requests in this module; transmit data empty, transmit end, receive data full, NACK detection, STOP recognition, and arbitration lost/overrun error. Table 25.4 shows the contents of each interrupt request.

Table 25.4 Interrupt Requests

Interrupt Request	Abbreviation	Interrupt Condition	I^{2}C Bus Format	Clocked Synchronous Serial Format
Transmit data Empty	TXI	(TDRE = 1) • (TIE = 1)	○	○
Transmit end	TEI	(TEND = 1) • (TEIE = 1)	○	○
Receive data full	RXI	(RDRF = 1) • (RIE = 1)	○	○
STOP recognition	STPI	(STOP = 1) • (STIE = 1)	○	×
NACK detection	NAKI	{(NACKF = 1) + (ALOVE = 1)} • (NAKIE = 1)	○	×
Arbitration lost			○	×
overrun error			×	○

When the interrupt condition in table 25.4 is "1", the IIC3 module will request interrupt handling to the CPU. The IIC3 module takes the logical OR of all the interrupt factors and issues the interrupt to the interrupt controller (INTC). During exception handling, the interrupt factor should be determined by reading the ICSR register. The handler should perform appropriate interrupt handling and clear the interrupt factor based on that determination. The TDRE and TEND flags are automatically cleared to "0" by writing the transmit data to the ICDRT register. The RDRF flag is automatically cleared to "0" by reading the ICDRR register. The TDRE flag is set to "1" again at the same time when the transmit data is written to the ICDRT register. Therefore, when the TDRE flag is cleared to "0", then an excessive data of one byte may be transmitted.

25.6 DMA Transfer Requests

This module issues two DMA transfer requests: the transmit data empty request and the receive data full request.

Table 25.5 lists these DMA transfer requests.

Table 25.5 DMA Transfer Requests

DMA Transfer Request	DMA Transfer Condition
Transmit data empty	TDRE = "1"
Receive data full	RDRF = "1"

When DMA transfer condition in table 25.5 is "1", the IIC3 module issues a DMA transfer request. If the DMAC activation settings have been made, the DMAC will start. If the transmit data empty DMA transfer request has been enabled by the DMAC activation settings, do not enable the interrupt request with the TIE bit. Similarly, if the receive data full DMA transfer request has been enabled, do not enable the interrupt request with the RIE bit.

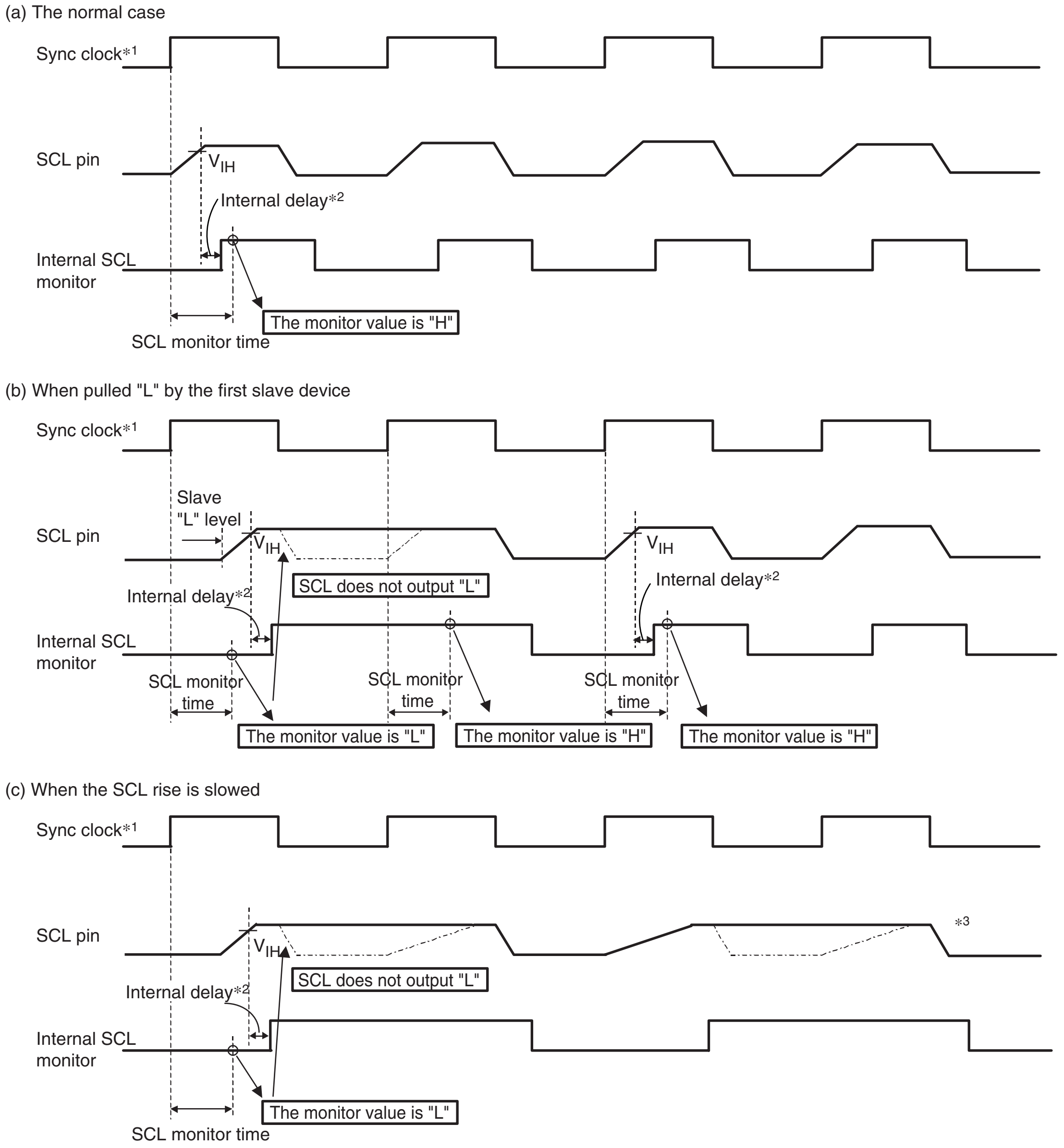
25.7 Bit Synchronous Circuit

In master mode, this module has a possibility that "H" level period may be short in the two states described below.

- When SCL is driven to "L" level by the slave device
- When the rising speed of SCL is lowered by the load of the SCL line (load capacitance or pull-up resistance)

Therefore, it monitors SCL and communicates by bit with synchronization.

Figure 25.23 shows the bit synchronization circuit timing, and table 25.6 shows the time from the point SCL switches from the "L" level to the high-impedance state until SCL is monitored.



Notes: *1 The clock for the transfer rate set with the I²C bus control register (ICCR1) CKS bit.
*2 This is 3 to 4 tpcyc when the I²C bus NF2CYC register (ICNF2CYC) NF2CYC bit is "0" and 4 to 5 tpcyc when that bit is "1".
*3 This differs from the set frequency.

Figure 25.23 Bit Synchronous Circuit Timing

Table 25.6 Time for Monitoring SCL

Bit 3 in CKS	Bit 2 in CKS	Time for Monitoring SCL*1
0	0	9 tpcyc*2
	1	21 tpcyc*2
1	0	39 tpcyc*2
	1	87 tpcyc*2

Notes: *1 Monitors the (on-board) SCL level after the time (pcyc) for monitoring SCL has passed since the rising edge of the SCL monitor timing reference clock.

*2 tpcyc indicates peripheral clock (Pck) cycle.

25.8 Usage Notes

25.8.1 Notes on Multi-Master Usage

When multi-master is used and this MCUs IIC transfer rate setting (the ICCR register CKS field) is slower than that for other masters, there are rare cases where an SCL with an unexpected width is output from SCL.

The transfer rate must be set to a rate that is faster than 1/1.8 times the rate of the other master with the highest transfer rate.

25.8.2 Notes on Master Reception Mode

There are cases where the receive data cannot be acquired if the ICDRR register is read at a time near the fall of the eighth clock cycle. Also, there are cases where it becomes impossible to issue a stop condition if RCVD is set to "1" with the receive buffer full near the fall of the eighth clock cycle.

Use either method 1 or 2 below to avoid this problem.

1. In master reception mode, perform the read of the ICDRR register before the rise of the eighth clock cycle.
2. In master reception mode, set the RCVD bit to "1" and perform the required processing using one byte per each communication.

25.8.3 Notes on ACKBT Settings in Master Reception Mode

When operating in master reception mode, set the ACKBT bit before the eighth fall of SCL for the last data item transferred continuously. Otherwise an overrun error could occur in the slave transmit device.

25.8.4 Notes on the MST and TRN Bit States at Arbitration Lost

When using this module in multi-master mode and setting up master transmission by manipulating the MST and TRS bits sequentially, there are cases where a contradictory state, in which the ICSR register ALOVE bit is "1" and furthermore master transmission mode (MST = "1" and TRS = "1") occurs. This phenomenon can be avoided by the following methods.

- In multi-master mode, use the MOV instruction to set the MST and TRS bits.
- If an arbitration lost occurs, verify that MST = "0" and that TRS = "0".

If a state other than MST = "0" and TRS = "0" occur, set MST = "0" and TRS = "0".

25.8.5 Notes on Setting up DMA Transfer Requests

When performing DMA transfers, set TIE and RIE so that interrupt requests due to the TDRE flag and the RDRF flag do not occur at the same time.

25.8.6 Notes on the I²C Bus Pull-up Voltage

Set the I²C bus pull-up voltage to be the same level as this MCUs Vcc pin. Do not apply a voltage higher than Vcc.

25.8.7 ICE and IICRST Bit Access During I²C Bus Operation

The ICCR2 register BBSY flag and the ICSR register STOP flag become undefined if, in any of the states 1 to 4 listed below, "0" is written to the ICCR1 register ICE bit or "1" is written to the ICCR2 register IICRST bit.

1. When this module has I²C bus rights in master transmit mode (MST = "1" and TRS = "1" in the ICCR1 register).
2. When this module has I²C bus rights in master receive mode (MST = "1" and TRS = "0" in the ICCR1 register).
3. When this module is receiving data in slave transmit mode (MST = "0" and TRS = "1" in the ICCR1 register).
4. When this module is transmitting and acknowledge in slave receive mode (MST = "0" and TRS = "0" in the ICCR1 register).

The undefined state of the ICCR2 register BBSY flag can be resolved in any of the following ways.

- When a start conditions is received (SCL is at the "H" level and the SDA signal falls from "H" to "L"), the BBSY flag is set to "1".
- When a stop conditions is received (SCL is at the "H" level and the SDA signal rises from "L" to "H"), the BSBY flag is cleared to "0".
- In master transmit mode and in the state where both SCL is at the "H" level and SDA is at the "H" level, a start condition is issued by writing "1" to the ICCR2 register BBSY flag and writing "0" to the SCP bit. When the start condition (SCL is at the "H" level and the SDA signal falls from "H" to "L") is output, the BBSY flag is set to "1".
- In either master transmit mode or master receive mode, and furthermore in the state where both SDL is at the "L" level and there is no device other than this module that is setting the SCL line to the "L" level, a stop condition is issued by writing "0" to the ICCR2 register BBSY flag and writing "0" to the SCP bit. When the stop condition (SCL is at the "H" level and the SDA signal rises from "L" to "H") is output, the BBSY flag is cleared to "0".
- The BBSY flag is cleared to "0" when "1" is written to the ICSAR register FS bit.

25.8.8 Register Initialization with the IICRST Bit

- The SDAO and SCLO bits in the ICCR2 register are set to "1" when "1" is written to the IICRST bit.
- In master transmit mode and slave transmit mode, the ICSR register TDRE flag is set to "1" when "1" is written to the IICRST bit.
- During the reset period due to setting the IICRST bit to "1", writes to the ICCR2 register BBSY flag, the SDAO bit, and the SCP bit are invalid.
- Even during the reset period due to setting the IICRST bit to "1", if either a start condition (SCL is at the "H" level and the SDA signal falls from "H" to "L") or a stop condition (SCL is at the "H" level and the SDA signal rises from "L" to "H") is received, the BBSY flag is set to "1" or cleared to "0" respectively.

25.8.9 I²C Bus Interface 3 Operation when ICE = "0"

If "0" is written to the ICCR1 register ICE bit, SCL and SDA output are prohibited. Note, however, that inputs to SCL and SDA remain valid. That is, this module operates according to the signal input to SCL and SDA in this state.

25.8.10 Notes on Master Reception Mode of I^{2}C Bus Interface Mode

The SCL signal extra outputs a clock after the ninth clock, if issue of stop condition or re-issue of start condition is overlapped with the falling of the ninth clock.

After the completion of a master reception, confirm the falling of the ninth clock on SCL signal, and then issue a stop condition or re-issue a start condition.

The falling of the ninth clock is confirmed as follow.

- Confirm that the RDRF flag (receive data full flag) in the ICSR register is set to "1", and then the SCLO flag (SCL output level flag) in the ICCR2 register is set to "0" (pins SCL outputs "L" level).

25.8.11 Notes on the Time of Stop Condition Generation in Master Transmit Mode

When a stop condition is issued in master transmit mode while the ACKE bit in the I^{2}C bus interrupt enable register (ICIER) is 1, the stop condition may not be normally output depending on the issued timing.

In master transmit mode while the ACKE bit in the I^{2}C bus interrupt enable register (ICIER) is 1, recognize the falling edge of the ninth clock before issuance of the stop condition.

The falling of the ninth clock is confirmed as follow.

- Confirm that the RDRF flag (receive data full flag) in the ICSR register is set to "1", and then the SCLO flag (SCL output level flag) in the ICCR2 register is set to "0" (pins SCL outputs "L" level).

Section 26 CAN Module

26.1 Overview

This MCU implements four channels (referred to as CAN0 to CAN3) of CAN (Controller Area Net-work) module that complies with the ISO11898-1 Specifications. The CAN module transmits and receives both formats of messages, namely the standard identifier (11 bits) (identifier hereafter referred to as ID) and extended ID (29 bits).

Table 26.1 lists the CAN module overview and figure 26.1 shows the CAN module block diagram.

Connect the CAN bus transceiver externally.

Table 26.1 CAN Module Overview

Item	Overview
Protocol	• ISO11898-1 compliant
Bit-rate	• Up to 1 Mbps
Message box	• 64 mailboxes: Two selectable mailbox mode Normal mailbox mode: Of the 64 mailboxes, 32 can be configured for either transmission or reception (and the other 32 are reception-only). FIFO mailbox mode: 24 mailboxes configurable as transmission or reception (and the other 32 are reception-only). 4 stages FIFO for transmission and 4 stages FIFO for reception
Reception	• Data frame and remote frame can be received. • Selectable receiving ID format (only standard ID, only extended ID or both ID) • Programmable one-shot reception function • Selectable overwrite mode (message overwritten) or overrun mode (message discarded) • The reception complete interrupt can be enabled or disabled for each mailbox.
Acceptance Filter	• 8 acceptance masks (one mask every 4 mailboxes) • 2 acceptance masks (one mask every 16 mailboxes) • The mask can be enabled or disabled for each mailbox.
Transmission	• Data frame and remote frame can be transmitted. • Selectable transmitting ID format (only standard ID, only extended ID or both ID) • Programmable one-shot transmission function (enable or disable) • Selectable ID priority mode or mailbox number priority mode • Transmission request can be aborted (The completion of abort can be confirmed with a flag.) • The transmission complete interrupt can be enabled or disabled for each mailbox.
Mode transition for bus-off recovery	• Mode transition for the recovery from the bus-off state can be selected: ISO11898-1 compliant Automatic entry to CAN halt mode at bus-off entry Automatic entry to CAN halt mode at bus-off end Entry to CAN halt mode by a program Transition into error-active state by a program

Item	Overview
Error status monitoring	• CAN bus errors (stuff error, form error, ACK error, CRC error, bit error, and ACK delimiter error) can be monitored. • Transition to error states can be detected (error-warning, error-passive, bus-off entry, and bus-off recovery). • The error counters can be read.
Time stamp function	• Time stamp function using a 16-bit counter • The reference clock can be selected from either 1-, 2-, 4- or 8-bit time periods.
Interrupt sources	• 5 types: Reception complete Transmission complete Receive FIFO Transmit FIFO Error
CAN sleep mode	• Current consumption can be reduced by stopping the CAN clock.
Software support unit	• 3 software support units: Acceptance filter support Mailbox search support (receive mailbox search, transmit mailbox search and message lost search) Channel search support
CAN clock source (fCAN)	• Selectable peripheral clock (Pck) or main clock
Test mode	• 3 test modes available for user evaluation: Listen-only mode Self-test mode 0 (external loop back) Self-test mode 1 (internal loop back)

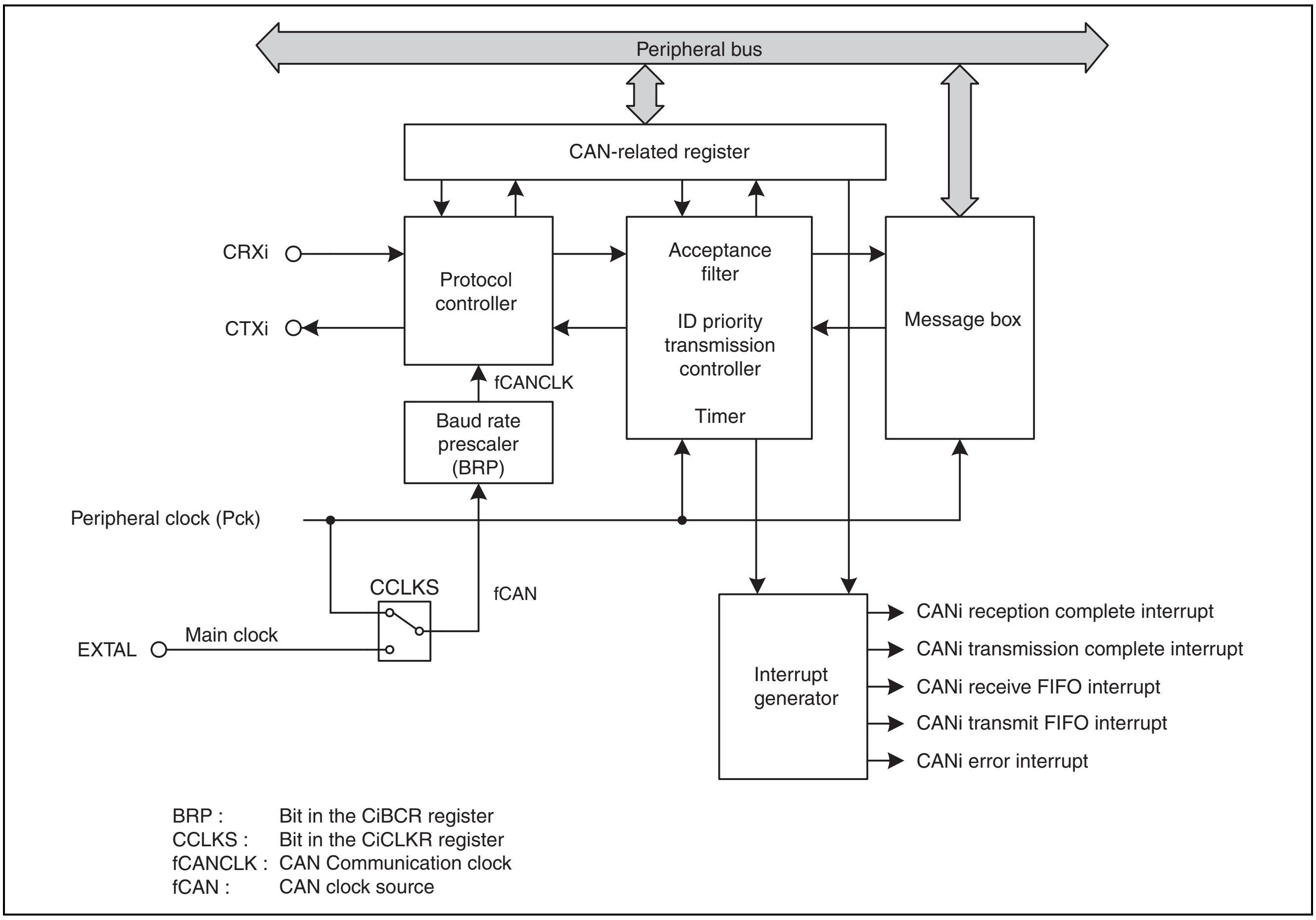

Peripheral bus
CAN-related register
CRXi
CTXi
Protocol controller
Acceptance filter
ID priority transmission controller
Timer
Message box
fCANCLK
Baud rate prescaler (BRP)
Peripheral clock (Pck)
CCLKS
fCAN
EXTAL
Main clock
Interrupt generator
CANi reception complete interrupt
CANi transmission complete interrupt
CANi receive FIFO interrupt
CANi transmit FIFO interrupt
CANi error interrupt
BRP : Bit in the CiBCR register
CCLKS : Bit in the CiCLKR register
fCANCLK : CAN Communication clock
fCAN : CAN clock source

Figure 26.1 Block Diagram of CAN Module (i = 0 to 3)

- CRXi/CTXi (i = 0 to 3):
  CAN input/output pins
- Protocol controller:
  Handles CAN protocol processing such as bus arbitration, bit timing at transmission and reception, stuffing, and error handling, etc.
- Message box:
  Consists of 64 mailboxes which can be configured as either transmit or receive mailboxes. Each mailbox has an individual ID, data length code, a data field (8 bytes), and a time stamp.
- Acceptance filter:
  Performs filtering of received messages. Registers CiMKR0 to CiMKR9 are used for the filtering process.
- Timer:
  Used for the time stamp function. The timer value when storing a message into the mailbox is written as the time stamp value.
- Interrupt generator:
  Generates the following five types of interrupts:
  CANi reception complete interrupt
  CANi transmission complete interrupt
  CANi receive FIFO interrupt
  CANi transmit FIFO interrupt
  CANi error interrupt

26.2 Input/Output Pins

Table 26.2 shows the CAN module pin.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 26.2 Pin Configuration

Pin Name	I/O	Function
CRXi	Input	Pins for receiving data
CTXi	Output	Pins for transmitting data

Legend: i = 0 to 3

26.3 Register Descriptions

Table 26.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
CAN0 Control Register	C0CTLR	H'0500	H'FFFF 6840	8, 16, 32	26-10
CAN0 Clock Select Register	C0CLKR	H'00	H'FFFF 6847	8, 16, 32	26-15
CAN0 Bit Configuration Register	C0BCR	H'00 0000	H'FFFF 6844	8, 16, 32	26-16
CAN0 Mask Register 0	C0MKR0	Undefined	H'FFFF 6430	8, 16, 32	26-18
CAN0 Mask Register 1	C0MKR1	Undefined	H'FFFF 6434	8, 16, 32	26-18
CAN0 Mask Register 2	C0MKR2	Undefined	H'FFFF 6400	8, 16, 32	26-18
CAN0 Mask Register 3	C0MKR3	Undefined	H'FFFF 6404	8, 16, 32	26-18
CAN0 Mask Register 4	C0MKR4	Undefined	H'FFFF 6408	8, 16, 32	26-18
CAN0 Mask Register 5	C0MKR5	Undefined	H'FFFF 640C	8, 16, 32	26-18
CAN0 Mask Register 6	C0MKR6	Undefined	H'FFFF 6410	8, 16, 32	26-18
CAN0 Mask Register 7	C0MKR7	Undefined	H'FFFF 6414	8, 16, 32	26-18
CAN0 Mask Register 8	C0MKR8	Undefined	H'FFFF 6418	8, 16, 32	26-18
CAN0 Mask Register 9	C0MKR9	Undefined	H'FFFF 641C	8, 16, 32	26-18
CAN0 FIFO Received ID Compare Register 0	C0FIDCR0	Undefined	H'FFFF 6420	8, 16, 32	26-20
CAN0 FIFO Received ID Compare Register 1	C0FIDCR1	Undefined	H'FFFF 6424	8, 16, 32	26-20
CAN0 Mask Invalid Register 0	C0MKIVLR0	Undefined	H'FFFF 6438	8, 16, 32	26-23
CAN0 Mask Invalid Register 1	C0MKIVLR1	Undefined	H'FFFF 6428	8, 16, 32	26-22
CAN0 Mailbox Register 0 to 63	C0MB0 to C0MB63	Undefined	H'FFFF 6000 to H'FFFF 63FF	8, 16, 32	26-24
CAN0 Mailbox Interrupt Enable Register 0	C0MIER0	Undefined	H'FFFF 643C	8, 16, 32	26-32
CAN0 Mailbox Interrupt Enable Register 1	C0MIER1	Undefined	H'FFFF 642C	8, 16, 32	26-30
CAN0 Message Control Register 0 to 63	C0MCTL0 to C0MCTL63	H'00	H'FFFF 6800 to H'FFFF 683F	8, 16, 32	26-33
CAN0 Receive FIFO Control Register	C0RFCR	H'80	H'FFFF 6848	8, 16, 32	26-38
CAN0 Receive FIFO Pointer Control Register	C0RFPCR	Undefined	H'FFFF 6849	8, 16, 32	26-41
CAN0 Transmit FIFO Control Register	C0TFCR	H'80	H'FFFF 684A	8, 16, 32	26-42
CAN0 Transmit FIFO Pointer Control Register	C0TFPCR	Undefined	H'FFFF 684B	8, 16, 32	26-45
CAN0 Status Register	C0STR	H'0500	H'FFFF 6842	8, 16, 32	26-46
CAN0 Mailbox Search Mode Register	C0MSMR	H'00	H'FFFF 6853	8, 16, 32	26-49
CAN0 Mailbox Search Status Register	C0MSSR	H'80	H'FFFF 6852	8, 16, 32	26-50
CAN0 Channel Search Support Register	C0CSSR	Undefined	H'FFFF 6851	8, 16, 32	26-51
CAN0 Acceptance Filter Support Register	C0AFSR	Undefined	H'FFFF 6856	8, 16, 32	26-53
CAN0 Error Interrupt Enable Register	C0EIER	H'00	H'FFFF 684C	8, 16, 32	26-54
CAN0 Error Interrupt Factor Judge Register	C0EIFR	H'00	H'FFFF 684D	8, 16, 32	26-56

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
CAN0 Receive Error Count Register	C0RECR	H'00	H'FFFF 684E	8, 16, 32	26-59
CAN0 Transmit Error Count Register	C0TECR	H'00	H'FFFF 684F	8, 16, 32	26-60
CAN0 Error Code Store Register	C0ECSR	H'00	H'FFFF 6850	8, 16, 32	26-61
CAN0 Time Stamp Register	C0TSR	H'0000	H'FFFF 6854	8, 16, 32	26-63
CAN0 Test Control Register	C0TCR	H'00	H'FFFF 6858	8	26-64
CAN0 Interrupt Enable Register	C0IER	H'00	H'FFFF 6860	8, 16	26-69
CAN0 Interrupt Status Register	C0ISR	H'00	H'FFFF 6861	8, 16	26-67
CAN0 Mailbox Search Mask Register	C0MBSMR	H'00	H'FFFF 6863	8, 16, 32	26-70
CAN1 Control Register	C1CTLR	H'0500	H'FFFF 7840	8, 16, 32	26-10
CAN1 Clock Select Register	C1CLKR	H'00	H'FFFF 7847	8, 16, 32	26-15
CAN1 Bit Configuration Register	C1BCR	H'00 0000	H'FFFF 7844	8, 16, 32	26-16
CAN1 Mask Register 0	C1MKR0	Undefined	H'FFFF 7430	8, 16, 32	26-18
CAN1 Mask Register 1	C1MKR1	Undefined	H'FFFF 7434	8, 16, 32	26-18
CAN1 Mask Register 2	C1MKR2	Undefined	H'FFFF 7400	8, 16, 32	26-18
CAN1 Mask Register 3	C1MKR3	Undefined	H'FFFF 7404	8, 16, 32	26-18
CAN1 Mask Register 4	C1MKR4	Undefined	H'FFFF 7408	8, 16, 32	26-18
CAN1 Mask Register 5	C1MKR5	Undefined	H'FFFF 740C	8, 16, 32	26-18
CAN1 Mask Register 6	C1MKR6	Undefined	H'FFFF 7410	8, 16, 32	26-18
CAN1 Mask Register 7	C1MKR7	Undefined	H'FFFF 7414	8, 16, 32	26-18
CAN1 Mask Register 8	C1MKR8	Undefined	H'FFFF 7418	8, 16, 32	26-18
CAN1 Mask Register 9	C1MKR9	Undefined	H'FFFF 741C	8, 16, 32	26-18
CAN1 FIFO Received ID Compare Register 0	C1FIDCR0	Undefined	H'FFFF 7420	8, 16, 32	26-20
CAN1 FIFO Received ID Compare Register 1	C1FIDCR1	Undefined	H'FFFF 7424	8, 16, 32	26-20
CAN1 Mask Invalid Register 0	C1MKIVLR0	Undefined	H'FFFF 7438	8, 16, 32	26-23
CAN1 Mask Invalid Register 1	C1MKIVLR1	Undefined	H'FFFF 7428	8, 16, 32	26-22
CAN1 Mailbox Register 0 to 63	C1MB0 to C1MB63	Undefined	H'FFFF 7000 to H'FFFF 73FF	8, 16, 32	26-24
CAN1 Mailbox Interrupt Enable Register 0	C1MIER0	Undefined	H'FFFF 743C	8, 16, 32	26-32
CAN1 Mailbox Interrupt Enable Register 1	C1MIER1	Undefined	H'FFFF 742C	8, 16, 32	26-30
CAN1 Message Control Register 0 to 63	C1MCTL0 to C1MCTL63	H'00	H'FFFF 7800 to H'FFFF 783F	8, 16, 32	26-33
CAN1 Receive FIFO Control Register	C1RFCR	H'80	H'FFFF 7848	8, 16, 32	26-38
CAN1 Receive FIFO Pointer Control Register	C1RFPCR	Undefined	H'FFFF 7849	8, 16, 32	26-41
CAN1 Transmit FIFO Control Register	C1TFCR	H'80	H'FFFF 784A	8, 16, 32	26-42
CAN1 Transmit FIFO Pointer Control Register	C1TFPCR	Undefined	H'FFFF 784B	8, 16, 32	26-45
CAN1 Status Register	C1STR	H'0500	H'FFFF 7842	8, 16, 32	26-46
CAN1 Mailbox Search Mode Register	C1MSMR	H'00	H'FFFF 7853	8, 16, 32	26-49
CAN1 Mailbox Search Status Register	C1MSSR	H'80	H'FFFF 7852	8, 16, 32	26-50

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
CAN1 Channel Search Support Register	C1CSSR	Undefined	H'FFFF 7851	8, 16, 32	26-51
CAN1 Acceptance Filter Support Register	C1AFSR	Undefined	H'FFFF 7856	8, 16, 32	26-53
CAN1 Error Interrupt Enable Register	C1EIER	H'00	H'FFFF 784C	8, 16, 32	26-54
CAN1 Error Interrupt Factor Judge Register	C1EIFR	H'00	H'FFFF 784D	8, 16, 32	26-56
CAN1 Receive Error Count Register	C1RECR	H'00	H'FFFF 784E	8, 16, 32	26-59
CAN1 Transmit Error Count Register	C1TECR	H'00	H'FFFF 784F	8, 16, 32	26-60
CAN1 Error Code Store Register	C1ECSR	H'00	H'FFFF 7850	8, 16, 32	26-61
CAN1 Time Stamp Register	C1TSR	H'0000	H'FFFF 7854	8, 16, 32	26-63
CAN1 Test Control Register	C1TCR	H'00	H'FFFF 7858	8	26-64
CAN1 Interrupt Enable Register	C1IER	H'00	H'FFFF 7860	8, 16	26-69
CAN1 Interrupt Status Register	C1ISR	H'00	H'FFFF 7861	8, 16	26-67
CAN1 Mailbox Search Mask Register	C1MBSMR	H'00	H'FFFF 7863	8, 16, 32	26-70
CAN2 Control Register	C2CTLR	H'0500	H'FFFF 8840	8, 16, 32	26-10
CAN2 Clock Select Register	C2CLKR	H'00	H'FFFF 8847	8, 16, 32	26-15
CAN2 Bit Configuration Register	C2BCR	H'00 0000	H'FFFF 8844	8, 16, 32	26-16
CAN2 Mask Register 0	C2MKR0	Undefined	H'FFFF 8430	8, 16, 32	26-18
CAN2 Mask Register 1	C2MKR1	Undefined	H'FFFF 8434	8, 16, 32	26-18
CAN2 Mask Register 2	C2MKR2	Undefined	H'FFFF 8400	8, 16, 32	26-18
CAN2 Mask Register 3	C2MKR3	Undefined	H'FFFF 8404	8, 16, 32	26-18
CAN2 Mask Register 4	C2MKR4	Undefined	H'FFFF 8408	8, 16, 32	26-18
CAN2 Mask Register 5	C2MKR5	Undefined	H'FFFF 840C	8, 16, 32	26-18
CAN2 Mask Register 6	C2MKR6	Undefined	H'FFFF 8410	8, 16, 32	26-18
CAN2 Mask Register 7	C2MKR7	Undefined	H'FFFF 8414	8, 16, 32	26-18
CAN2 Mask Register 8	C2MKR8	Undefined	H'FFFF 8418	8, 16, 32	26-18
CAN2 Mask Register 9	C2MKR9	Undefined	H'FFFF 841C	8, 16, 32	26-18
CAN2 FIFO Received ID Compare Register 0	C2FIDCR0	Undefined	H'FFFF 8420	8, 16, 32	26-20
CAN2 FIFO Received ID Compare Register 1	C2FIDCR1	Undefined	H'FFFF 8424	8, 16, 32	26-20
CAN2 Mask Invalid Register 0	C2MKIVLR0	Undefined	H'FFFF 8438	8, 16, 32	26-23
CAN2 Mask Invalid Register 1	C2MKIVLR1	Undefined	H'FFFF 8428	8, 16, 32	26-22
CAN2 Mailbox Register 0 to 63	C2MB0 to C2MB63	Undefined	H'FFFF 8000 to H'FFFF 83FF	8, 16, 32	26-24
CAN2 Mailbox Interrupt Enable Register 0	C2MIER0	Undefined	H'FFFF 843C	8, 16, 32	26-32
CAN2 Mailbox Interrupt Enable Register 1	C2MIER1	Undefined	H'FFFF 842C	8, 16, 32	26-30
CAN2 Message Control Register 0 to 63	C2MCTL0 to C2MCTL63	H'00	H'FFFF 8800 to H'FFFF 883F	8, 16, 32	26-33
CAN2 Receive FIFO Control Register	C2RFCR	H'80	H'FFFF 8848	8, 16, 32	26-38
CAN2 Receive FIFO Pointer Control Register	C2RFPCR	Undefined	H'FFFF 8849	8, 16, 32	26-41
CAN2 Transmit FIFO Control Register	C2TFCR	H'80	H'FFFF 884A	8, 16, 32	26-42

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
CAN2 Transmit FIFO Pointer Control Register	C2TFPCR	Undefined	H'FFFF 884B	8, 16, 32	26-45
CAN2 Status Register	C2STR	H'0500	H'FFFF 8842	8, 16, 32	26-46
CAN2 Mailbox Search Mode Register	C2MSMR	H'00	H'FFFF 8853	8, 16, 32	26-49
CAN2 Mailbox Search Status Register	C2MSSR	H'80	H'FFFF 8852	8, 16, 32	26-50
CAN2 Channel Search Support Register	C2CSSR	Undefined	H'FFFF 8851	8, 16, 32	26-51
CAN2 Acceptance Filter Support Register	C2AFSR	Undefined	H'FFFF 8856	8, 16, 32	26-53
CAN2 Error Interrupt Enable Register	C2EIER	H'00	H'FFFF 884C	8, 16, 32	26-54
CAN2 Error Interrupt Factor Judge Register	C2EIFR	H'00	H'FFFF 884D	8, 16, 32	26-56
CAN2 Receive Error Count Register	C2RECR	H'00	H'FFFF 884E	8, 16, 32	26-59
CAN2 Transmit Error Count Register	C2TECR	H'00	H'FFFF 884F	8, 16, 32	26-60
CAN2 Error Code Store Register	C2ECSR	H'00	H'FFFF 8850	8, 16, 32	26-61
CAN2 Time Stamp Register	C2TSR	H'0000	H'FFFF 8854	8, 16, 32	26-63
CAN2 Test Control Register	C2TCR	H'00	H'FFFF 8858	8	26-64
CAN2 Interrupt Enable Register	C2IER	H'00	H'FFFF 8860	8, 16	26-69
CAN2 Interrupt Status Register	C2ISR	H'00	H'FFFF 8861	8, 16	26-67
CAN2 Mailbox Search Mask Register	C2MBSMR	H'00	H'FFFF 8863	8, 16, 32	26-70
CAN3 Control Register	C3CTLR	H'0500	H'FFFF 9840	8, 16, 32	26-10
CAN3 Clock Select Register	C3CLKR	H'00	H'FFFF 9847	8, 16, 32	26-15
CAN3 Bit Configuration Register	C3BCR	H'00 0000	H'FFFF 9844	8, 16, 32	26-16
CAN3 Mask Register 0	C3MKR0	Undefined	H'FFFF 9430	8, 16, 32	26-18
CAN3 Mask Register 1	C3MKR1	Undefined	H'FFFF 9434	8, 16, 32	26-18
CAN3 Mask Register 2	C3MKR2	Undefined	H'FFFF 9400	8, 16, 32	26-18
CAN3 Mask Register 3	C3MKR3	Undefined	H'FFFF 9404	8, 16, 32	26-18
CAN3 Mask Register 4	C3MKR4	Undefined	H'FFFF 9408	8, 16, 32	26-18
CAN3 Mask Register 5	C3MKR5	Undefined	H'FFFF 940C	8, 16, 32	26-18
CAN3 Mask Register 6	C3MKR6	Undefined	H'FFFF 9410	8, 16, 32	26-18
CAN3 Mask Register 7	C3MKR7	Undefined	H'FFFF 9414	8, 16, 32	26-18
CAN3 Mask Register 8	C3MKR8	Undefined	H'FFFF 9418	8, 16, 32	26-18
CAN3 Mask Register 9	C3MKR9	Undefined	H'FFFF 941C	8, 16, 32	26-18
CAN3 FIFO Received ID Compare Register 0	C3FIDCR0	Undefined	H'FFFF 9420	8, 16, 32	26-20
CAN3 FIFO Received ID Compare Register 1	C3FIDCR1	Undefined	H'FFFF 9424	8, 16, 32	26-20
CAN3 Mask Invalid Register 0	C3MKIVLR0	Undefined	H'FFFF 9438	8, 16, 32	26-23
CAN3 Mask Invalid Register 1	C3MKIVLR1	Undefined	H'FFFF 9428	8, 16, 32	26-22
CAN3 Mailbox Register 0 to 63	C3MB0 to C3MB63	Undefined	H'FFFF 9000 to H'FFFF 93FF	8, 16, 32	26-24
CAN3 Mailbox Interrupt Enable Register 0	C3MIER0	Undefined	H'FFFF 943C	8, 16, 32	26-32
CAN3 Mailbox Interrupt Enable Register 1	C3MIER1	Undefined	H'FFFF 942C	8, 16, 32	26-30

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
CAN3 Message Control Register 0 to 63	C3MCTL0 to C3MCTL63	H'00	H'FFFF 9800 to H'FFFF 983F	8, 16, 32	26-33
CAN3 Receive FIFO Control Register	C3RFCR	H'80	H'FFFF 9848	8, 16, 32	26-38
CAN3 Receive FIFO Pointer Control Register	C3RFPCR	Undefined	H'FFFF 9849	8, 16, 32	26-41
CAN3 Transmit FIFO Control Register	C3TFCR	H'80	H'FFFF 984A	8, 16, 32	26-42
CAN3 Transmit FIFO Pointer Control Register	C3TFPCR	Undefined	H'FFFF 984B	8, 16, 32	26-45
CAN3 Status Register	C3STR	H'0500	H'FFFF 9842	8, 16, 32	26-46
CAN3 Mailbox Search Mode Register	C3MSMR	H'00	H'FFFF 9853	8, 16, 32	26-49
CAN3 Mailbox Search Status Register	C3MSSR	H'80	H'FFFF 9852	8, 16, 32	26-50
CAN3 Channel Search Support Register	C3CSSR	Undefined	H'FFFF 9851	8, 16, 32	26-51
CAN3 Acceptance Filter Support Register	C3AFSR	Undefined	H'FFFF 9856	8, 16, 32	26-53
CAN3 Error Interrupt Enable Register	C3EIER	H'00	H'FFFF 984C	8, 16, 32	26-54
CAN3 Error Interrupt Factor Judge Register	C3EIFR	H'00	H'FFFF 984D	8, 16, 32	26-56
CAN3 Receive Error Count Register	C3RECR	H'00	H'FFFF 984E	8, 16, 32	26-59
CAN3 Transmit Error Count Register	C3TECR	H'00	H'FFFF 984F	8, 16, 32	26-60
CAN3 Error Code Store Register	C3ECSR	H'00	H'FFFF 9850	8, 16, 32	26-61
CAN3 Time Stamp Register	C3TSR	H'0000	H'FFFF 9854	8, 16, 32	26-63
CAN3 Test Control Register	C3TCR	H'00	H'FFFF 9858	8	26-64
CAN3 Interrupt Enable Register	C3IER	H'00	H'FFFF 9860	8, 16	26-69
CAN3 Interrupt Status Register	C3ISR	H'00	H'FFFF 9861	8, 16	26-67
CAN3 Mailbox Search Mask Register	C3MBSMR	H'00	H'FFFF 9863	8, 16, 32	26-70

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

26.3.1 CANi Control Register (CiCTLR) (i = 0 to 3)

CAN0 Control Register (C0CTLR)	<P4 address: location H'FFFF 6840>
CAN1 Control Register (C1CTLR)	<P4 address: location H'FFFF 7840>
CAN2 Control Register (C2CTLR)	<P4 address: location H'FFFF 8840>
CAN3 Control Register (C3CTLR)	<P4 address: location H'FFFF 9840>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RBOC	BOM		SLPM	CANM		TSPS		TSRC	TPM	MLM	IDFM		MBM
After Reset:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0

<After Reset: H'0500>

Bit	Abbreviation	After Reset	R	W	Description
15, 14	—	All 0	0	0	Reserved Bits Should be written with "0".
13	RBOC	0	R	W	Forcible Return From Bus-OFF Bit*[1] When the RBOC bit is set to "1" (forcible return from bus-off) in bus-off state, the CAN module forcibly returns from the bus-off state. This bit is automatically set to 0. The error state changes from bus-off to error-active. When the RBOC bit is set to "1", registers CiRECR and CiTECR are set to "H'00" and the BOST bit in the CiSTR register is set to "0" (the CAN module is not in bus-off state). The other registers remain unchanged. No bus-off recovery interrupt request is generated by this recovery from the bus-off state. Use the RBOC bit only when the BOM bit = "00" (normal mode). 0: Nothing occurred 1: Forcible return from bus-off*[2]

Bit	Abbreviation	After Reset	R	W	Description
12, 11	BOM	All 0	R	W	Bus-Off Recovery Mode Bit*3 The BOM bit is used to select bus-off recovery mode. When the BOM bit is "00", the recovery from bus-off is compliant with ISO11898-1, i.e. the CAN module re-enters CAN communication (error-active state) after detecting 11 consecutive recessive bits 128 times. A bus-off recovery interrupt request is generated when recovering from bus-off. When the BOM bit is "01", as soon as the CAN reaches the bus-off state, the CANM bit in the CiCTLR register is set to "10" (CAN halt mode) and the CAN enters CAN halt mode. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to "H'00". When the BOM bit is "10", the CANM bit is set to 10b as soon as the CAN module reaches the bus-off state. The CAN module enters CAN halt mode after the recovery from the bus-off state, i.e. after detecting 11 consecutive recessive bits 128 times. A bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to "H'00". When the BOM bit is "11", the CAN module enters CAN halt mode by setting the CANM bit to "10" while the CAN module is still in bus-off state. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to "H'00". However, if the CAN module recovers from bus-off after detecting 11 consecutive recessive bits 128 times before the CANM bit is set to "10", a bus-off recovery interrupt request is generated. If the CPU requests an entry to CAN reset mode at the same time as the CAN module attempts to enter CAN halt mode (at bus-off entry when the BOM bit is "01", or at bus-off end when the BOM bit is "10"), then the CPU request to enter CAN reset mode has higher priority. 00: Normal mode (ISO11898-1 compliant) 01: Entry to CAN halt mode automatically at bus-off entry 10: Entry to CAN halt mode automatically at bus-off end 11: Entry to CAN halt mode (during bus-off recovery period) by a program request
10	SLPM	1	R	W	CAN Sleep Mode Bit*4*5 When the SLPM bit is set to "1", the CAN module enters CAN sleep mode. When the SLPM bit is set to "0", the CAN module exits CAN sleep mode. Refer to section 26.4, Operating Mode for detail. 0: Other than CAN sleep mode 1: CAN sleep mode

Bit	Abbreviation	After Reset	R	W	Description
9, 8	CANM	01	R	W	CAN Operating Mode Select Bit*4 The CANM bit selects one of the following modes for the CAN module: CAN operation mode, CAN reset mode or CAN halt mode. Refer to section 26.4, Operating Mode for detail. CAN sleep mode is set by the SLPM bit. When the CAN module enters CAN halt mode according to the setting of the BOM bit, the CANM bit is automatically set to "10". 00: CAN operation mode 01: CAN reset mode 10: CAN halt mode 11: CAN reset mode (forcible transition)
7, 6	TSPS	All 0	R	W	Time Stamp Prescaler Select Bit*3 The TSPS bit selects the prescaler for the time stamp. The reference clock for the time stamp can be selected to be either 1-, 2-, 4- or 8-bit time periods. 00: Every bit time 01: Every 2-bit time 10: Every 4-bit time 11: Every 8-bit time
5	TSRC	0	R	W	Time Stamp Counter Reset Command Bit*6 The TSRC bit is used to reset the time stamp counter. When the TSRC bit is set to "1", the CiTSR register is set to H'0000. It is automatically set to 0. 0: Nothing occurred 1: Reset*2
4	TPM	0	R	W	Transmission Priority Mode Select Bit*3 The TPM bit specifies the priority of modes when transmitting messages. ID priority transmit mode or mailbox number transmit mode can be selected. All mailboxes are set for either ID priority transmission or mailbox number priority transmission. When the TPM bit is 0, ID priority transmit mode is selected and transmission priority complies with the CAN bus arbitration rule, as defined in the ISO 11898-1 Specifications. In ID priority transmit mode, mailboxes [0] to [63] (in normal mailbox mode), and mailboxes [0] to [55] (in FIFO mailbox mode), and the transmit FIFO are compared for the IDs of mailboxes configured for transmission. If two or more mailbox IDs are the same, the mailbox with the smaller number has higher priority. Only the next message to be transmitted from the transmit FIFO is included in the transmission arbitration. If a transmit FIFO message is being transmitted, the next pending message within the transmit FIFO is included in the transmission arbitration. When the TPM bit is 1, mailbox number transmit mode is selected and the transmit mailbox with the smallest mailbox number has the highest priority. In FIFO mailbox mode, the transmit FIFO has lower priority than normal mailboxes (mailboxes [0] to [55]). 0: ID priority transmit mode 1: Mailbox number priority transmit mode

Bit	Abbreviation	After Reset	R	W	Description
3	MLM	0	R	W	Message Lost Mode Select Bit*3 The MLM bit specifies the operation when a new message is captured in the unread mailbox. Overwrite mode or overrun mode can be selected. All mailboxes (including the receive FIFO) are set to either overwrite mode or overrun mode. When the MLM bit is "0", all mailboxes are set to overwrite mode and the new message is overwriting the old message. When this bit is "1", all mailboxes are set to overrun mode and the new message is discarded. 0: Overwrite mode 1: Overrun mode
2, 1	IDFM	All 0	R	W	ID Format Mode Select Bit*3 The IDFM bit specifies the ID format. 00: Standard ID mode All mailboxes (including FIFO mailboxes) handle only standard IDs. 01: Extended ID mode All mailboxes (including FIFO mailboxes) handle only extended IDs. 10: Mixed ID mode All mailboxes (including FIFO mailboxes) handle both standard IDs and extended IDs. Standard IDs or extended IDs are specified by using the IDE bit in the corresponding mailbox in normal mail box mode. In FIFO mailbox mode, the IDE bit in the corresponding mailbox is used for mailboxes [0] to [55], the IDE bit in registers CiFIDCR0 and CiFIDCR1 is used for the receive FIFO, and the IDE bit in mailbox [56] is used for the transmit FIFO. 11: Setting prohibited
0	MBM	0	R	W	CAN Mailbox Mode Select Bit*3 When the MBM bit is "0" (normal mailbox mode), mailboxes [0] to [63] are configured as transmit or receive mailboxes. When the MBM bit is "1" (FIFO mailbox mode), mailboxes [0] to [55] are configured as transmit or receive mailboxes. Mailboxes [56] to [59] are configured as a transmit FIFO and mailboxes [60] to [63] as a receive FIFO. Transmit data is written into mailbox [56] (mailbox [56] is a window mailbox for the transmit FIFO). Receive data is read from mailbox [60] (mailbox [60] is a window mailbox for the receive FIFO). Table 26.4 lists the Mailbox Configuration. 0: Normal mailbox mode 1: FIFO mailbox mode

Notes: *1 Set the RBOC bit to "1" in bus-off state.

*2 Bits RBOC and TSRC are automatically set back to "0" after being set to "1". It should be read as "0".

*3 Write to bits BOM, MBM, IDFM, MLM, TPM, and TSPS in CAN reset mode.

*4 When bits CANM and SLPM are changed, check the CiSTR register to ensure that the mode has been switched.

*5 Write to the SLPM bit in CAN reset mode or CAN halt mode. When rewriting the SLPM bit, set only this bit to "0" or "1".

*6 Set the TSRC bit to "1" in CAN operation mode.

Table 26.4 Mailbox Configuration

Mailbox	MBM Bit = "0" (Normal mailbox mode)	MBM Bit = "1" (FIFO mailbox mode)
Mailboxes [0] to [55]	Normal mailbox	Normal mailbox
Mailboxes [56] to [59]		Transmit FIFO
Mailboxes [60] to [63]		Receive FIFO

Notes: • Points 1 to 5 below should be considered when the MBM bit is set to "1".

1. Transmit FIFO is controlled by the CiTFCR register.
   The CiMCTLj register of mailboxes [56] to [59] are disabled.
   Registers CiMCTL56 to CiMCTL59 cannot be used.
2. Receive FIFO is controlled by the CiRFCR register.
   The CiMCTLj register of mailboxes [60] to [63] are disabled.
   Registers CiMCTL60 to CiMCTL63 cannot be used.
3. Refer to the CiMIER1 register about the FIFO interrupts.
4. The corresponding bits in the CiMKIVLR1 register for mailboxes [56] to [63] are disabled. Set 0 to these bits.
5. Transmit/receive FIFOs can be used for both data frames and remote frames.

26.3.2 CANi Clock Select Register (CiCLKR) (i = 0 to 3)

CAN0 Clock Select Register (C0CLKR)	<P4 address: location H'FFFF 6847>
CAN1 Clock Select Register (C1CLKR)	<P4 address: location H'FFFF 7847>
CAN2 Clock Select Register (C2CLKR)	<P4 address: location H'FFFF 8847>
CAN3 Clock Select Register (C3CLKR)	<P4 address: location H'FFFF 9847>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CCLKS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	—	0	0	0	Reserved Bit Should be written with "0".
6, 5	—	All 0	0	0	No Register Bit Should be written with "0" and read as "0".
4	—	0	?	0	Reserved Bit Should be written with "0" and read as undefined value.
3	—	0	0	0	Reserved Bit Should be written with "0".
2	—	0	0	0	No Register Bit Should be written with "0" and read as "0".
1	—	0	0	0	Reserved Bit Should be written with "0".
0	CCLKS	0	R	W	CAN Clock Source Select Bit*1*2 When the CCLKS bit is set to "0", peripheral clock (Pck) generated by the PLL frequency synthesizer is used as the CAN clock source (fCAN). When the CCLKS bit is set to "1", the main clock directly input from the external EXTAL pin bypassing the PLL frequency synthesizer is used as fCAN. 0: Peripheral clock (Pck) 1: Main clock

Notes: *1 Write to the CCLKS bit in CAN reset mode.

*2 To set the CCLKS bit to "1", the frequency of the peripheral clock (Pck) should be equal to or higher than the frequency of the main clock.

26.3.3 CANi Bit Configuration Register (CiBCR) (i = 0 to 3)

Refer to section 26.5, CAN Communication Speed Configuration about bit timing configuration rule.

Set the CiBCR register before entering CAN halt mode from CAN reset mode or CAN operation mode from CAN reset mode. After the setting is made once, this register can be written to in CAN reset mode or CAN halt mode.

The CiBCR register consists of 24 bits. A 32-bit read/write access should be performed carefully not to rewrite the CiCLKR register.

CAN0 Bit Configuration Register (C0BCR)	<P4 address: location H'FFFF 6844>
CAN1 Bit Configuration Register (C1BCR)	<P4 address: location H'FFFF 7844>
CAN2 Bit Configuration Register (C2BCR)	<P4 address: location H'FFFF 8844>
CAN3 Bit Configuration Register (C3BCR)	<P4 address: location H'FFFF 9844>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSEG1				—	—	BRP									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8
	—	—	SJW		—	TSEG2		
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'000000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 28	TSEG1	All 0	R	W	Time Segment 1 Control Bits The TSEG1 bit is used to specify the total length of the propagation time segment (PROP_SEG) and phase buffer segment 1 (PHASE_SEG1) with the value of Tq. A value from 4 to 16 time quanta can be set. 0000: Setting prohibited 0001: Setting prohibited 0010: Setting prohibited 0011: 4 Tq 0100: 5 Tq 0101: 6 Tq 0110: 7 Tq 0111: 8 Tq 1000: 9 Tq 1001: 10 Tq 1010: 11 Tq 1011: 12 Tq 1100: 13 Tq 1101: 14 Tq 1110: 15 Tq 1111: 16 Tq
27	—	0	0	0	No Register Bit Should be written with "0" and read as "0".
26	—	0	0	0	Reserved Bit Should be written with "0".

Bit	Abbreviation	After Reset	R	W	Description
25 to 16	BRP	All 0	R	W	Prescaler Division Ratio Set Bit The BRP bit is used to set the peripheral bus clock periods contained in a Time Quantum. If the setting value is P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.
15, 14	—	All 0	0	0	No Register Bits Should be written with "0" and read as "0".
13, 12	SJW	All 0	R	W	Resynchronization Jump Width Control Bit The SJW bit is used to specify the resynchronization jump width with the value of Tq. A value from 1 to 4 time quanta can be set. Set the value smaller than or equal to that of the TSEG2 bit. 00: 1 Tq 01: 2 Tq 10: 3 Tq 11: 4 Tq
11	—	0	0	0	No Register Bit Should be written with "0" and read as "0".
10 to 8	TSEG2	All 0	R	W	Time Segment 2 Control Bit The TSEG2 bit is used to specify the length of phase buffer segment 2 (PHASE_SEG2) with the value of Tq. A value from 2 to 8 time quanta can be set. Set the value smaller than that of the TSEG1 bit. 000: Setting prohibited 001: 2 Tq 010: 3 Tq 011: 4 Tq 100: 5 Tq 101: 6 Tq 110: 7 Tq 111: 8 Tq

26.3.4 CANi Mask Register k (CiMKRk) (i = 0 to 3; k = 0 to 9)

For register configuration, refer to section 26.6, Mailbox and Mask Register Structure.

Refer to section 26.7, Acceptance Filtering and Masking Function about mask function in FIFO mailbox mode.

Write to registers CiMKR0 to CiMKR9 in CAN reset mode or CAN halt mode.

CAN0 Mask Register 0 (C0MKR0)	<P4 address: location H'FFFF 6430>
CAN0 Mask Register 1 (C0MKR1)	<P4 address: location H'FFFF 6434>
CAN0 Mask Register 2 (C0MKR2)	<P4 address: location H'FFFF 6400>
:	:
CAN0 Mask Register 9 (C0MKR9)	<P4 address: location H'FFFF 641C>
CAN1 Mask Register 0 (C1MKR0)	<P4 address: location H'FFFF 7430>
CAN1 Mask Register 1 (C1MKR1)	<P4 address: location H'FFFF 7434>
CAN1 Mask Register 2 (C1MKR2)	<P4 address: location H'FFFF 7400>
:	:
CAN1 Mask Register 9 (C1MKR9)	<P4 address: location H'FFFF 741C>
CAN2 Mask Register 0 (C2MKR0)	<P4 address: location H'FFFF 8430>
CAN2 Mask Register 1 (C2MKR1)	<P4 address: location H'FFFF 8434>
CAN2 Mask Register 2 (C2MKR2)	<P4 address: location H'FFFF 8400>
:	:
CAN2 Mask Register 9 (C2MKR9)	<P4 address: location H'FFFF 841C>
CAN3 Mask Register 0 (C3MKR0)	<P4 address: location H'FFFF 9430>
CAN3 Mask Register 1 (C3MKR1)	<P4 address: location H'FFFF 9434>
CAN3 Mask Register 2 (C3MKR2)	<P4 address: location H'FFFF 9400>
:	:
CAN3 Mask Register 9 (C3MKR9)	<P4 address: location H'FFFF 941C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SID											EID	
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EID															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	Undefined	0	0	The reset value is undefined. The write value should be "0". These bits are read as "0" after "0" is written to.
28 to 18	SID	Undefined	R	W	Standard ID Bit The SID bit is the filter mask bit corresponding to the CAN standard ID bit. The SID bit is used to receive both standard ID and extended ID messages. When the SID bit is set to "0", the corresponding SID bit is not compared for the received ID and the mailbox ID. When the SID bit is set to "1", corresponding SID bit compares received ID with mailbox ID. 0: Corresponding SID bit is not compared 1: Corresponding SID bit is compared
17 to 0	EID	Undefined	R	W	Extended ID Bit The EID bit is the filter mask bit for CAN extended ID bit. This bit is used to receive extended ID messages. When the EID bit is set to "0", corresponding EID bit does not compare received ID with mailbox ID. When the EID bit is set to "1", corresponding EID bit compares received ID with mailbox ID. 0: Corresponding EID bit is not compared 1: Corresponding EID bit is compared

26.3.5 CANi FIFO Received ID Compare Registers (CiFIDCR0 and CiFIDCR1) (i = 0 to 3)

Registers CiFIDCR0 and CiFIDCR1 are enabled when the MBM bit in the CiCTLR register is set to "1" (FIFO mailbox mode). Bits EID, SID, RTR, and IDE in registers CiMB60 to CiMB63 are disabled.

For register configuration, refer to section 26.6, Mailbox and Mask Register Structure.

Refer to section 26.7, Acceptance Filtering and Masking Function about the usage of these registers.

Write to registers CiFIDCR0 and CiFIDCR1 in CAN reset mode or CAN halt mode.

CAN0 FIFO Received ID Compare Register 0 (C0FIDCR0)	<P4 address: location H'FFFF 6420>
CAN0 FIFO Received ID Compare Register 1 (C0FIDCR1)	<P4 address: location H'FFFF 6424>
CAN1 FIFO Received ID Compare Register 0 (C1FIDCR0)	<P4 address: location H'FFFF 7420>
CAN1 FIFO Received ID Compare Register 1 (C1FIDCR1)	<P4 address: location H'FFFF 7424>
CAN2 FIFO Received ID Compare Register 0 (C2FIDCR0)	<P4 address: location H'FFFF 8420>
CAN2 FIFO Received ID Compare Register 1 (C2FIDCR1)	<P4 address: location H'FFFF 8424>
CAN3 FIFO Received ID Compare Register 0 (C3FIDCR0)	<P4 address: location H'FFFF 9420>
CAN3 FIFO Received ID Compare Register 1 (C3FIDCR1)	<P4 address: location H'FFFF 9424>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IDE	RTR	—	SID											EID	
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EID															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31	IDE	Undefined	R	W	ID Extension Bit*[1] The IDE bit sets the ID format to standard ID or extended ID. The IDE bit is enabled when the IDFM bit in the CiCTLR register is "10" (mixed ID mode). When the IDFM bit is "10", the IDE bit specifies the following operation. • When both IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to "0", only standard ID frames can be received. • When both IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to "1", only extended ID frames can be received. • When the IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to "0" or "1" individually, both standard ID and extended ID frames can be received. 0: Standard ID 1: Extended ID

Bit	Abbreviation	After Reset	R	W	Description
30	RTR	Undefined	R	W	Remote Transmission Request Bit The RTR bit sets the specified frames format of data frame or remote frames. The RTR bit specifies the following operation. • When both RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to "0", only data frames can be received. • When both RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to "1", only remote frames can be received. • When the RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to "0" or "1" individually, both data frames and remote frames can be received. 0: Data frame 1: Remote frame
29	—	Undefined	0	0	The reset value is undefined. The write value should be "0". This bit is read as "0" after "0" is written to.
28 to 18	SID	Undefined	R	W	Standard ID Bit The SID bit sets the standard ID of data frames and remote frames. The SID bit is used to receive both standard ID and extended ID messages. 0: Corresponding SID bit is "0" 1: Corresponding SID bit is "1"
17 to 0	EID	Undefined	R	W	Extended ID Bit The EID bit sets the extended ID of data frames and remote frames. The EID bit is used to receive extended ID messages. 0: Corresponding EID bit is "0" 1: Corresponding EID bit is "1"

Note: *1 When the IDFM bit is not "10", the IDE bit should be written with "0".

26.3.6 CANi Mask Invalid Registers (CiMKIVLR0 and CiMKIVLR1) (i = 0 to 3)

Each bit in registers CiMKIVLR0 and CiMKIVLR1 corresponds to a mailbox. The correspondence between the bits and mailboxes is as follows:

- Bit 0 in the CiMKIVLR0 register corresponds to mailbox 0 (MB0).
- Bit 31 in the CiMKIVLR0 register corresponds to mailbox 31 (MB31).
- Bit 0 in the CiMKIVLR1 register corresponds to mailbox 32 (MB32).
- Bit 31 in the CiMKIVLR1 register corresponds to mailbox 63 (MB63).

When each bit is "1", the acceptance mask for the mailbox corresponding to the bit number is disabled. In this case, a receiving message is stored into mailbox only its ID matches bits SID and EID in the CiMBj register.

Write to registers CiMKIVLR0 and CiMKIVLR1 either in CAN reset mode or CAN halt mode.

CAN0 Mas Invalid Register 1 (C0MKIVLR1)	<P4 address: location H'FFFF 6428>
CAN1 Mas Invalid Register 1 (C1MKIVLR1)	<P4 address: location H'FFFF 7428>
CAN2 Mas Invalid Register 1 (C2MKIVLR1)	<P4 address: location H'FFFF 8428>
CAN3 Mas Invalid Register 1 (C3MKIVLR1)	<P4 address: location H'FFFF 9428>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB63	MB62	MB61	MB60	MB59	MB58	MB57	MB56	MB55	MB54	MB53	MB52	MB51	MB50	MB49	MB48
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB47	MB46	MB45	MB44	MB43	MB42	MB41	MB40	MB39	MB38	MB37	MB36	MB35	MB34	MB33	MB32
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MB63 to 32	Undefined	R	W	Bit 31 corresponds to mailbox 63 (MB63), and bit 0 corresponds to mailbox 32 (MB32)*[1]. 0: Mask valid 1: Mask invalid

Note: *1 In FIFO mailbox mode, write "0" to bits 24 to 31.

CAN0 Mas Invalid Register 0 (C0MKIVLR0)	<P4 address: location H'FFFF 6438>
CAN1 Mas Invalid Register 0 (C1MKIVLR0)	<P4 address: location H'FFFF 7438>
CAN2 Mas Invalid Register 0 (C2MKIVLR0)	<P4 address: location H'FFFF 8438>
CAN3 Mas Invalid Register 0 (C3MKIVLR0)	<P4 address: location H'FFFF 9438>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB31	MB30	MB29	MB28	MB27	MB26	MB25	MB24	MB23	MB22	MB21	MB20	MB19	MB18	MB17	MB16
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB15	MB14	MB13	MB12	MB11	MB10	MB9	MB8	MB7	MB6	MB5	MB4	MB3	MB2	MB1	MB0
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MB31 to 0	Undefined	R	W	Bit 31 corresponds to mailbox 31 (MB31), and bit 0 corresponds to mailbox 0 (MB0). 0: Mask valid 1: Mask invalid

26.3.7 CANi Mailbox Register j (CiMBj) (i = 0 to 3; j = 0 to 63)

Table 26.5 lists the CANi mailbox memory mapping and table 26.6 lists the CAN data frame construction.

The value after reset of CANi Mailbox is undefined.

Write to the CiMBj register only when the associated CiMCTLj register is "H00" and the corresponding mailbox is not processing an abort request.

Refer to table 26.5 for detailed addresses.

For register configuration, refer to section 26.6, Mailbox and Mask Register Structure.

Table 26.5 CANi Mailbox Memory Mapping (i = 0 to 3)

Address				Message Content
CAN0	**CAN1**	**CAN2**	**CAN3**	**Memory Mapping**
H'FFFF 6000 + 16 × j + 0	H'FFFF 7000 + 16 × j + 0	H'FFFF 8000 + 16 × j + 0	H'FFFF 9000 + 16 × j + 0	IDE,RTR, SID10 to SID6
H'FFFF 6000 + 16 × j + 1	H'FFFF 7000 + 16 × j + 1	H'FFFF 8000 + 16 × j + 1	H'FFFF 9000 + 16 × j + 1	SID5 to SID0, EID17, EID16
H'FFFF 6000 + 16 × j + 2	H'FFFF 7000 + 16 × j + 2	H'FFFF 8000 + 16 × j + 2	H'FFFF 9000 + 16 × j + 2	EID15 to EID8
H'FFFF 6000 + 16 × j + 3	H'FFFF 7000 + 16 × j + 3	H'FFFF 8000 + 16 × j + 3	H'FFFF 9000 + 16 × j + 3	EID7 to EID0
H'FFFF 6000 + 16 × j + 4	H'FFFF 7000 + 16 × j + 4	H'FFFF 8000 + 16 × j + 4	H'FFFF 9000 + 16 × j + 4	—
H'FFFF 6000 + 16 × j + 5	H'FFFF 7000 + 16 × j + 5	H'FFFF 8000 + 16 × j + 5	H'FFFF 9000 + 16 × j + 5	Data length code (DLC)
H'FFFF 6000 + 16 × j + 6	H'FFFF 7000 + 16 × j + 6	H'FFFF 8000 + 16 × j + 6	H'FFFF 9000 + 16 × j + 6	Data byte 0
H'FFFF 6000 + 16 × j + 7	H'FFFF 7000 + 16 × j + 7	H'FFFF 8000 + 16 × j + 7	H'FFFF 9000 + 16 × j + 7	Data byte 1
:	:	:	:	:
H'FFFF 6000 + 16 × j + 13	H'FFFF 7000 + 16 × j + 13	H'FFFF 8000 + 16 × j + 13	H'FFFF 9000 + 16 × j + 13	Data byte 7
H'FFFF 6000 + 16 × j + 14	H'FFFF 7000 + 16 × j + 14	H'FFFF 8000 + 16 × j + 14	H'FFFF 9000 + 16 × j + 14	Time stamp upper byte
H'FFFF 6000 + 16 × j + 15	H'FFFF 7000 + 16 × j + 15	H'FFFF 8000 + 16 × j + 15	H'FFFF 9000 + 16 × j + 15	Time stamp lower byte

Table 26.6 CAN Data Frame Construction

SID10 to SID6	SID5 to SID0	EID17 to EID16	EID15 to EID8	EID7 to EID0	DLC3 to DLC0	DATA0	DATA1	…	DATA7

The previous value of each mailbox is retained unless a new message is received.

CAN0 Mailbox Register 0 to 63 (C0MB0 to 63)	<P4 address: location H'FFFF 6000 to H'FFFF 63FF>
CAN1 Mailbox Register 0 to 63 (C1MB0 to 63)	<P4 address: location H'FFFF 7000 to H'FFFF 73FF>
CAN2 Mailbox Register 0 to 63 (C2MB0 to 63)	<P4 address: location H'FFFF 8000 to H'FFFF 83FF>
CAN3 Mailbox Register 0 to 63 (C3MB0 to 63)	<P4 address: location H'FFFF 9000 to H'FFFF 93FF>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IDE	RTR	—	SID											EID	
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EID															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31	IDE	Undefined	R	W	ID Extension Bit*[1] The IDE bit sets the ID format of standard IDs or extended IDs. The IDE bit is enabled when the IDFM bit in the CiCTLR register is "10" (mixed ID mode). When the IDFM bit is "10", the IDE bit specifies the following operation. • Receive mailbox receives only ID format specified by the IDE bit. • Transmit mailbox transmits with ID format specified by the IDE bit. • Receive FIFO mailbox receives messages with the standard ID, extended ID, or both IDs specified by the IDE bit in registers CiFIDCR0 and CiFIDCR1. • Transmit FIFO mailbox transmits messages with the standard ID or extended ID specified by the IDE bit in the relevant transmitting message. 0: Standard ID 1: Extended ID
30	RTR	Undefined	R	W	Remote Frame Request Bit The RTR bit sets the frame format of data frames or remote frames. This bit specifies the following operation: • Receive mailbox receives only frames with the format specified by the RTR bit. • Transmit mailbox transmits according to the frame format specified by the RTR bit. • Receive FIFO mailbox receives the data frame, remote frame, or both frames specified by the RTR bit in registers CiFIDCR0 and CiFIDCR1. • Transmit FIFO mailbox transmits the data frame or remote frame specified by the RTR bit in the relevant transmitting message. 0: Data frame 1: Remote frame

Bit	Abbreviation	After Reset	R	W	Description
29	—	Undefined	0	0	The reset value is undefined. The write value should be "0". This bit is read as "0" after "0" is written to.
28 to 18	SID	Undefined	R	W	Standard ID Bits The SID bit sets the standard ID of data frames and remote frames. The SID bit is used to transmit or receive both standard ID and extended ID messages. 0: Corresponding SID bit is "0" 1: Corresponding SID bit is "1"
17 to 0	EID	Undefined	R	W	Extended ID Bits*[2] The EID bit sets the extended ID of data frames and remote frames. The EID bit is used to transmit or receive extended ID messages. 0: Corresponding EID bit is "0" 1: Corresponding EID bit is "1"

Notes: *1 When the IDFM bit is not "10", it should be written with "0".

*2 If the mailbox has received a standard ID message, the EID bit in the mailbox is undefined.

- For register configuration, refer to section 26.6, Mailbox and Mask Register Structure.

CAN0 Mailbox Register 0 to 63 (C0MB0 to 63) <P4 address: location H'FFFF 6000 to H'FFFF 63FF>
CAN1 Mailbox Register 0 to 63 (C1MB0 to 63) <P4 address: location H'FFFF 7000 to H'FFFF 73FF>
CAN2 Mailbox Register 0 to 63 (C2MB0 to 63) <P4 address: location H'FFFF 8000 to H'FFFF 83FF>
CAN3 Mailbox Register 0 to 63 (C3MB0 to 63) <P4 address: location H'FFFF 9000 to H'FFFF 93FF>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	DLC			
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
15 to 4	—	Undefined	0	0	The reset value is undefined. The write value should be "0". These bits are read as "0" after "0" is written to.
3 to 0	DLC	Undefined	R	W	Data Length Code Bits*1 The DLC is used to set the number of data bytes to be transmitted in a data frame. When data is requested using a remote frame, the number of data bytes to be requested is set. When a data frame is received, the number of received data bytes is stored. When a remote frame is received, the number of requested data bytes is stored. 0000: Data length = 0 byte 0001: Data length = 1 byte 0010: Data length = 2 bytes 0011: Data length = 3 bytes 0100: Data length = 4 bytes 0101: Data length = 5 bytes 0110: Data length = 6 bytes 0111: Data length = 7 bytes 1xxx: Data length = 8 bytes Legend: *x* represents any value.

Note: *1 If the mailbox has received a message with n bytes less than 8 bytes, the values of DATAn to DATA7 in the mailbox are undefined.

CAN0 Mailbox Register 0 to 63 (C0MB0 to 63)	<P4 address: location H'FFFF 6000 to H'FFFF 63FF>
CAN1 Mailbox Register 0 to 63 (C1MB0 to 63)	<P4 address: location H'FFFF 7000 to H'FFFF 73FF>
CAN2 Mailbox Register 0 to 63 (C2MB0 to 63)	<P4 address: location H'FFFF 8000 to H'FFFF 83FF>
CAN3 Mailbox Register 0 to 63 (C3MB0 to 63)	<P4 address: location H'FFFF 9000 to H'FFFF 93FF>

Bit:	7	6	5	4	3	2	1	0
	DATA0							
	DATA1							
	DATA2							
	DATA3							
	:							
	DATA7							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	DATA0 to 7	Undefined	R	W	Data Bytes 0 to 7*[1]*[2] DATA0 to DATA7 store the transmitted or received CAN message data. Transmission or reception starts from DATA0. The bit order on the CAN bus is MSB first, and transmission or reception starts from bit 7.

Notes: *1 If the mailbox has received a message with n bytes less than 8 bytes, the values of DATAn to DATA7 in the mailbox are undefined.

*2 If the mailbox has received a remote frame, the previous values of DATA0 to DATA7 in the mailbox are retained.

CAN0 Mailbox Register 0 to 63 (C0MB0 to 63) <P4 address: location H'FFFF 6000 to H'FFFF 63FF>
CAN1 Mailbox Register 0 to 63 (C1MB0 to 63) <P4 address: location H'FFFF 7000 to H'FFFF 73FF>
CAN2 Mailbox Register 0 to 63 (C2MB0 to 63) <P4 address: location H'FFFF 8000 to H'FFFF 83FF>
CAN3 Mailbox Register 0 to 63 (C3MB0 to 63) <P4 address: location H'FFFF 9000 to H'FFFF 93FF>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSH								TSL							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	TSH	Undefined	R	W	Time Stamp Higher Byte TSL and TSH store the counter value of the time stamp when received messages are stored in the mailbox.
7 to 0	TSL	Undefined	R	W	Time Stamp Lower Byte TSL and TSH store the counter value of the time stamp when received messages are stored in the mailbox.

26.3.8 CANi Mailbox Interrupt Enable Registers (CiMIER0 and CiMIER1) (i = 0 to 3)

Interrupts can enabled individually for each mailbox.

In normal mailbox mode (all bits) and in FIFO mailbox mode (bits 23 to 0 in the CiMIER1 register and all bits in the CiMIER0 register), each bit corresponds to the mailbox with the related number. These bits enable or disable transmission/reception complete interrupts for the corresponding mailboxes.

- Bit 0 in the CiMIER0 register corresponds to mailbox 0 (MB0).
- Bit 31 in the CiMIER0 register corresponds to mailbox 31 (MB31).
- Bit 0 in the CiMIER1 register corresponds to mailbox 32 (MB32).
- Bit 31 in the CiMIER1 register corresponds to mailbox 63 (MB63).

In FIFO mailbox mode, bits 29, 28, 25, and 24 of the CiMIER1 register specify whether transmit/receive FIFO interrupts are enabled/disabled and timing when interrupt requests are generated.

Write to registers CiMIER0 and CiMIER1 only when the associated CiMCTLj register (i = 0 to 4) (j = 0 to 63) is "H'00" and the corresponding mailbox is not processing a transmission or reception abort request. In FIFO mailbox mode, change the bits in the CiMIER1 register for the associated FIFO only when:

— The TFE bit in the CiTFCR register is 0 and the TFEST bit is 1, and

— The RFE bit in the CiRFCR register is 0 and the RFEST bit is 1.

CAN0 Mailbox Interrupt Enable Register 1 (C0MIER1) <P4 address: location H'FFFF 642C>
CAN1 Mailbox Interrupt Enable Register 1 (C1MIER1) <P4 address: location H'FFFF 742C>
CAN2 Mailbox Interrupt Enable Register 1 (C2MIER1) <P4 address: location H'FFFF 842C>
CAN3 Mailbox Interrupt Enable Register 1 (C3MIER1) <P4 address: location H'FFFF 942C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB63	MB62	MB61	MB60	MB59	MB58	MB57	MB56	MB55	MB54	MB53	MB52	MB51	MB50	MB49	MB48
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB47	MB46	MB45	MB44	MB43	MB42	MB41	MB40	MB39	MB38	MB37	MB36	MB35	MB34	MB33	MB32
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

- Normal mailbox mode

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MB63 to 32	Undefined	R	W	Interrupt Enable Bits Bit 31 corresponds to mailbox 63 (MB63), and bit 0 corresponds to mailbox 32 (MB32). 0: Interrupt disabled 1: Interrupt enabled

- FIFO mailbox mode (CiMIER1 only)

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31, 30	MB63, 62	Undefined	0	0	Reserved Bits Should be written with "0".
29	MB61	Undefined	R	W	Receive FIFO Interrupt Generation Timing Control Bit*[1] Receive FIFO interrupt request is generated 0: Every time reception is completed 1:When receive FIFO becomes buffer warning by completion of reception
28	MB60	Undefined	R	W	Receive FIFO Interrupt Enable Bit 0: Interrupt disabled 1: Interrupt enabled
27, 26	MB59, 58	Undefined	0	0	Reserved Bits Should be written with "0".
25	MB57	Undefined	R	W	Transmit FIFO Interrupt Generation Timing Control Bit Transmit FIFO interrupt request is generated 0: Every time transmission is completed 1:When transmit FIFO becomes empty due to completion of transmission
24	MB56	Undefined	R	W	Transmit FIFO Interrupt Enable Bit 0: Interrupt disabled 1: Interrupt enabled
23 to 0	MB55 to 32	Undefined	R	W	Interrupt Enable Bits Bit 23 corresponds to mailbox 55 (MB55), and bit 0 corresponds to mailbox 32 (MB32). 0: Interrupt disabled 1: Interrupt enabled

Note: *1 No interrupt request is generated when the receive FIFO becomes buffer warning from full. “Buffer warning” indicates a state in which the third unread message is stored in the receive FIFO.

CAN0 Mailbox Interrupt Enable Register 0 (C0MIER0) <P4 address: location H'FFFF 643C>
CAN1 Mailbox Interrupt Enable Register 0 (C1MIER0) <P4 address: location H'FFFF 743C>
CAN2 Mailbox Interrupt Enable Register 0 (C2MIER0) <P4 address: location H'FFFF 843C>
CAN3 Mailbox Interrupt Enable Register 0 (C3MIER0) <P4 address: location H'FFFF 943C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB31	MB30	MB29	MB28	MB27	MB26	MB25	MB24	MB23	MB22	MB21	MB20	MB19	MB18	MB17	MB16
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB15	MB14	MB13	MB12	MB11	MB10	MB9	MB8	MB7	MB6	MB5	MB4	MB3	MB2	MB1	MB0
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MB31 to 0	Undefined	R	W	Interrupt Enable Bits Bit 31 corresponds to mailbox 31 (MB31), and bit 0 corresponds to mailbox 0 (MB0). 0: Interrupt disabled 1: Interrupt enabled

26.3.9 CANi Message Control Register j (CiMCTLj) (i = 0 to 3; j = 0 to 63)

Write to the CiMCTLj register in CAN operation mode or CAN halt mode.

Do not use registers CiMCTL56 to CiMCTL63 in FIFO mailbox mode.

CAN0 Message Control Register 0 to 63 (C0MCTL0 to C0MCTL63)	<P4 address: location H'FFFF 6800 to H'FFFF 683F>
CAN1 Message Control Register 0 to 63 (C1MCTL0 to C1MCTL63)	<P4 address: location H'FFFF 7800 to H'FFFF 783F>
CAN2 Message Control Register 0 to 63 (C2MCTL0 to C2MCTL63)	<P4 address: location H'FFFF 8800 to H'FFFF 883F>
CAN3 Message Control Register 0 to 63 (C3MCTL0 to C3MCTL63)	<P4 address: location H'FFFF 9800 to H'FFFF 983F>

Registers CiMCTL32 to CiMCTL63

- Transmit mailbox setting enabled (When the TRMREQ bit is "1" and the RECREQ bit is "0")

Bit:	7	6	5	4	3	2	1	0
	TRM REQ	REC REQ	—	ONE SHOT	—	TRM ABT	TRM ACTIVE	SENT DATA
After Reset:	0	0	0	0	0	0	0	0

- Receive mailbox setting enabled (When the TRMREQ bit is "0" and the RECREQ bit is "1")

Bit:	7	6	5	4	3	2	1	0
	TRM REQ	REC REQ	—	ONE SHOT	—	MSG LOST	INVAL DATA	NEW DATA
After Reset:	0	0	0	0	0	0	0	0

Registers CiMCTL0 to CiMCTL31

Bit:	7	6	5	4	3	2	1	0
	—	REC REQ	—	—	—	MSG LOST	INVAL DATA	NEW DATA
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	TRMREQ	0	R	W	Transmit Mailbox Request Bit*2*4 The TRMREQ bit selects transmit modes shown in Table 26.11. When TRMREQ bit is set to "1", the corresponding mailbox is configured for transmission of a data frame or a remote frame. When TRMREQ bit is set to "0", the corresponding mailbox is not configured for transmission of a data frame or a remote frame. If the TRMREQ bit is changed from "1" to "0" to cancel the corresponding transmission request, either the TRMABT or SENTDATA bit is set to "1". When setting the TRMREQ bit to "1", do not set the RECREQ bit to "1". To change the configuration of a mailbox from reception to transmission, first abort the reception and then set bits NEWDATA and MSGLOST to "0" before changing to transmission. 0: Not configured for transmission 1: Configured for transmission
	—	0	0	0	No Register Bit (Registers CiMCTL0 to CiMCTL31) Should be written with "0" and read as "0".
6	RECREQ	0	R	W	Receive Mailbox Request Bit*2*4*5 The RECREQ bit selects receive modes shown in table 26.11. When the RECREQ bit is set to "1", the corresponding mailbox is configured for reception of a data frame or a remote frame. When the RECREQ bit is set to "0", the corresponding mailbox is not configured for reception of a data frame or a remote frame. Due to hardware protection the RECREQ bit cannot be set to "0" by writing "0" by a program during the following period. • Hardware protection is started From the acceptance filter procedure. (the beginning of CRC field) • Hardware protection is released — For the mailbox that is specified to receive the incoming message, after the received data is stored into the mailbox or a CAN bus error occurs. (i.e. a maximum period of hardware protection is from the beginning of CRC field to the end of the 7th bit of EOF.) — For the other mailboxes, after the acceptance filter procedure. — If no mailbox is specified to receive the message, after the acceptance filter procedure. When setting the RECREQ bit to "1", do not set "1" to the TRMREQ bit. To change the configuration of a mailbox from transmission to reception, first abort the transmission and then set bits SENTDATA and TRMABT to "0" before changing to reception. 0: Not configured for reception 1: Configured for reception

Bit	Abbreviation	After Reset	R	W	Description
5	—	0	0	0	No Register Bit Should be written with "0" and read as "0".
4	ONESHOT	0	R	W	One-shot Enable Bit*[3] The ONESHOT bit can be used in the following two ways, receive mode and transmit mode: • One-Shot Receive Mode When the ONESHOT bit is set to "1" in receive mode (RECREQ bit = "1" and TRMREQ bit = "0"), the mailbox receives a message only one time. The mailbox does not behave as a receive mailbox after having received a message one time. The behavior of bits NEWDATA and INVALDATA is the same as in normal reception mode. In one-shot receive mode, the MSGLOST bit is not set to "1". To set the ONESHOT bit to "0", first write "0" to the RECREQ bit and ensure that it has been set to "0". • One-Shot Transmit Mode When the ONESHOT bit is set to "1" in transmit mode (RECREQ bit = "0" and TRMREQ bit = "1"), the CAN module transmits a message only one time. The CAN module does not transmit the message again if a CAN bus error or CAN bus arbitration lost occurs. When transmission is completed, the SENTDATA bit is set to "1". If transmission is not completed due to a CAN bus error or CAN bus arbitration lost, the TRMABT bit is set to "1". Set the ONESHOT bit to "0" after the SENTDATA or TRMABT bit is set to "1". 0: One-shot reception or one-shot transmission disabled 1: One-shot reception or one-shot transmission enabled
	—	0	0	0	No Register Bit (Registers CiMCTL0 to CiMCTL31) Should be written with "0" and read as "0".
3	—	0	0	0	No Register Bit Should be written with "0" and read as "0".

Bit	Abbreviation	After Reset	R	W	Description
2	TRMABT	0	R	W	Transmission Abort Complete Flag (Transmit mailbox setting enabled)*1*2 The TRMABT bit is set to "1" in the following cases: • Following a transmission abort request, when the transmission abort is completed before starting transmission. • Following a transmission abort request, when the CAN module detects CAN bus arbitration lost or a CAN bus error. • In one-shot transmission mode (RECREQ bit = "0", TRMREQ bit = "1", and ONESHOT bit = "1"), when the CAN module detects CAN bus arbitration lost or a CAN bus error. The TRMABT bit is not set to "1" when data transmission is completed. In this case, the SENTDATA bit is set to "1". The TRMABT bit is set to "0" by writing "0" by a program. 0: Transmission has started, transmission abort failed because transmission is completed, or transmission abort is not requested 1: Transmission abort is completed
	MSGLOST	0	R	W	Message Lost Flag (Receive mailbox setting enabled)*1*2 The MSGLOST bit is set to "1" when the mailbox is overwritten or overrun by a new received message while the NEWDATA bit is "1". The MSGLOST bit is set to "1" at the end of the 6th bit of EOF. The MSGLOST bit is set to "0" by writing "0" by a program. In both overwrite and overrun modes, the MSGLOST bit is not set to "0" by writing "0" by a program during the 5 peripheral clock (Pck) cycles following the 6th bit of EOF. 0: Message is not overwritten or overrun 1: Message is overwritten or overrun
1	TRMACTIVE	0	R	0	Transmission-in-Progress Status Flag (Transmit mailbox setting enabled) The TRMACTIVE bit is set to "1" when the corresponding mailbox of the CAN module begins transmitting a message. The TRMACTIVE is set to "0" when the CAN module has lost CAN bus arbitration, a CAN bus error occurs, or data transmission is completed. 0: Transmission is pending or transmission is not requested 1: From acceptance of transmission request to completion of transmission, or error/arbitration lost
	INVALDATA	0	R	0	Reception-in-Progress Status Flag (Receive mailbox setting enabled) After the completion of a message reception, the INVALDATA bit is set to "1" while the received message is being updated into the corresponding mailbox. The INVALDATA bit is set to "0" immediately after the message has been stored. If the mailbox is read while the INVALDATA bit is "1", the data is undefined. 0: Message valid 1: Message being updated

Bit	Abbreviation	After Reset	R	W	Description
0	SENTDATA	0	R	W	Transmission Complete Flag (Transmit mailbox setting enabled)*[1]*[2] The SENTDATA bit is set to "1" when data transmission from the corresponding mailbox is completed. The SENTDATA bit is set to "0" by writing "0" by a program. To set the SENTDATA bit to "0", first set the TRMREQ bit to "0". Bits SENTDATA and TRMREQ cannot be set to "0" simultaneously. To transmit a new message from the corresponding mailbox, set the SENTDATA bit to "0". 0: Transmission is not completed (pending) 1: Transmission is completed (success)
	NEWDATA	0	R	W	Reception Complete Flag (Receive mailbox setting enabled)*[1]*[2] The NEWDATA bit is set to "1 when a new message is being stored or has been stored to the mailbox. The timing for setting this bit to "1" is simultaneous with the INVALDATA bit. The NEWDATA bit is set to "0" by writing "0" by a program. The NEWDATA bit is not set to "0" by writing "0" by a program while the related INVALDATA bit is "1". 0: No data has been received or "0" is written to the NEWDATA bit 1: A new message is being stored or has been stored to the mailbox

Notes: *1 Write "0" only. Writing "1" has no effect.

*2 When writing "0" to bits NEWDATA, SENTDATA, MSGLOST, TRMABT, RECREQ, and TRMREQ by a program, use the MOV instruction to ensure that only the specified bit is set to "0" and the other bits are set to "1".

*3 To enter one-shot receive mode, write "1" to the ONESHOT bit at the same time as setting the RECREQ bit to "1".

To exit one-shot receive mode, write "0" to the ONESHOT bit after writing "0" to the the RECREQ bit and confirming it has been set to "0".

To enter one-shot transmit mode, write "1" to the ONESHOT bit at the same time as setting the TRMREQ bit to "1".

To exit one-shot transmit mode, write "0" to the ONESHOT bit after the message has been transmitted or aborted.

*4 Do not set both the RECREQ and TRMREQ bits to "1".

*5 When setting the RECREQ bit to "0", set bits MSGLOST, NEWDATA, RECREQ to "0" simultaneously.

26.3.10 CANi Receive FIFO Control Register (CiRFCR) (i = 0 to 3)

Write to the CiRFCR registers in CAN operation mode or CAN halt mode.

CAN0 Receive FIFO Control Register (C0RFCR)	<P4 address: location H'FFFF 6848>
CAN1 Receive FIFO Control Register (C1RFCR)	<P4 address: location H'FFFF 7848>
CAN2 Receive FIFO Control Register (C2RFCR)	<P4 address: location H'FFFF 8848>
CAN3 Receive FIFO Control Register (C3RFCR)	<P4 address: location H'FFFF 9848>

Bit:	7	6	5	4	3	2	1	0
	RFEST	RFWST	RFFST	RFMLF	RFUST			RFE
After Reset:	1	0	0	0	0	0	0	0

<After Reset: H'80>

Bit	Abbreviation	After Reset	R	W	Description
7	RFEST	1	R	0	Receive FIFO Empty Status Flag The RFEST bit is set to "1" (no unread message in receive FIFO) when the number of unread messages in the receive FIFO is "0". The RFEST bit is set to "1" when the RFE bit is set to "0". The RFEST bit is set to "0" (unread message in receive FIFO) when the number of unread messages in the receive FIFO is one or more. 0: Unread message in receive FIFO 1: No unread message in receive FIFO
6	RFWST	0	R	0	Receive FIFO Buffer Warning Status Flag The RFWST bit is set to "1" (receive FIFO is buffer warning) when the number of unread messages in the receive FIFO is 3. The RFWST bit is "0" (receive FIFO is not buffer warning) when the number of unread messages in the receive FIFO is less than 3 or equal to 4. The RFWST bit is set to "0" when the RFE bit is "0". 0: Receive FIFO is not buffer warning 1: Receive FIFO is buffer warning (3 unread messages)
5	RFFST	0	R	0	Receive FIFO Full Status Flag The RFFST bit is set to "1" (receive FIFO is full) when the number of unread messages in the receive FIFO is 4. The RFFST bit is "0" (receive FIFO is not full) when the number of unread messages in the receive FIFO is less than 4. The RFFST bit is set to "0" when the RFE bit is "0". 0: Receive FIFO is not full 1: Receive FIFO full (4 unread messages)
4	RFMLF	0	R	0	Receive FIFO Message Lost Flag The RFMLF bit is set to "1" (receive FIFO message lost has occurred) when the receive FIFO receives a new message while the receive FIFO is full. The timing for setting this bit to "1" is at the end of the 6th bit of EOF. The RFMLF bit is set to "0" by writing "0" by a program (writing "1" has no effect). In both overwrite and overrun modes, the RFMLF bit cannot be set to "0" (receive FIFO message lost has not occurred) by writing "0" by a program due to hardware protection during the five cycles of peripheral clock (Pck) following the 6th bit of EOF, if the receive FIFO is full and determined to receive the message. 0: No receive FIFO message lost has occurred 1: Receive FIFO message lost has occurred

Bit	Abbreviation	After Reset	R	W	Description
3 to 1	RFUST	All 0	R	0	Receive FIFO Unread Message Number Status Flag The RFUST bit indicates the number of unread messages in the receive FIFO. The value of the RFUST bit is initialized to "000" when the RFE bit is set to "0". 000: No unread message 001: 1 unread message 010: 2 unread messages 011: 3 unread messages 100: 4 unread messages 101: Reserved 110: Reserved 111: Reserved
0	RFE	0	R	W	Receive FIFO Enable Bit When the RFE bit is set to "1", the receive FIFO is enabled. When this bit is set to "0", the receive FIFO is disabled for reception and becomes empty (RFEST bit = "1"). Write "0" to RFMLF and RFE bits respectively. Do not set this bit to "1" in normal mailbox mode (MBM bit in the CiCTLR register = "0"). Due to hardware protection, the RFE bit is not set to "0" by writing "0" by a program during the following period: • The hardware protection is started — From the acceptance filter procedure (the beginning of CRC field) • The hardware protection is released — If the receive FIFO is specified to receive the incoming message, after the received data is stored into the receive FIFO or a CAN bus error occurs. (i.e. maximum period of hardware protection is from the beginning of CRC field to the end of 7th bit of EOF.) — If the receive FIFO is not specified to receive the message, after the acceptance filter procedure. 0: Receive FIFO disabled 1: Receive FIFO enabled

Figure 26.2 shows the receive FIFO mailbox operation.

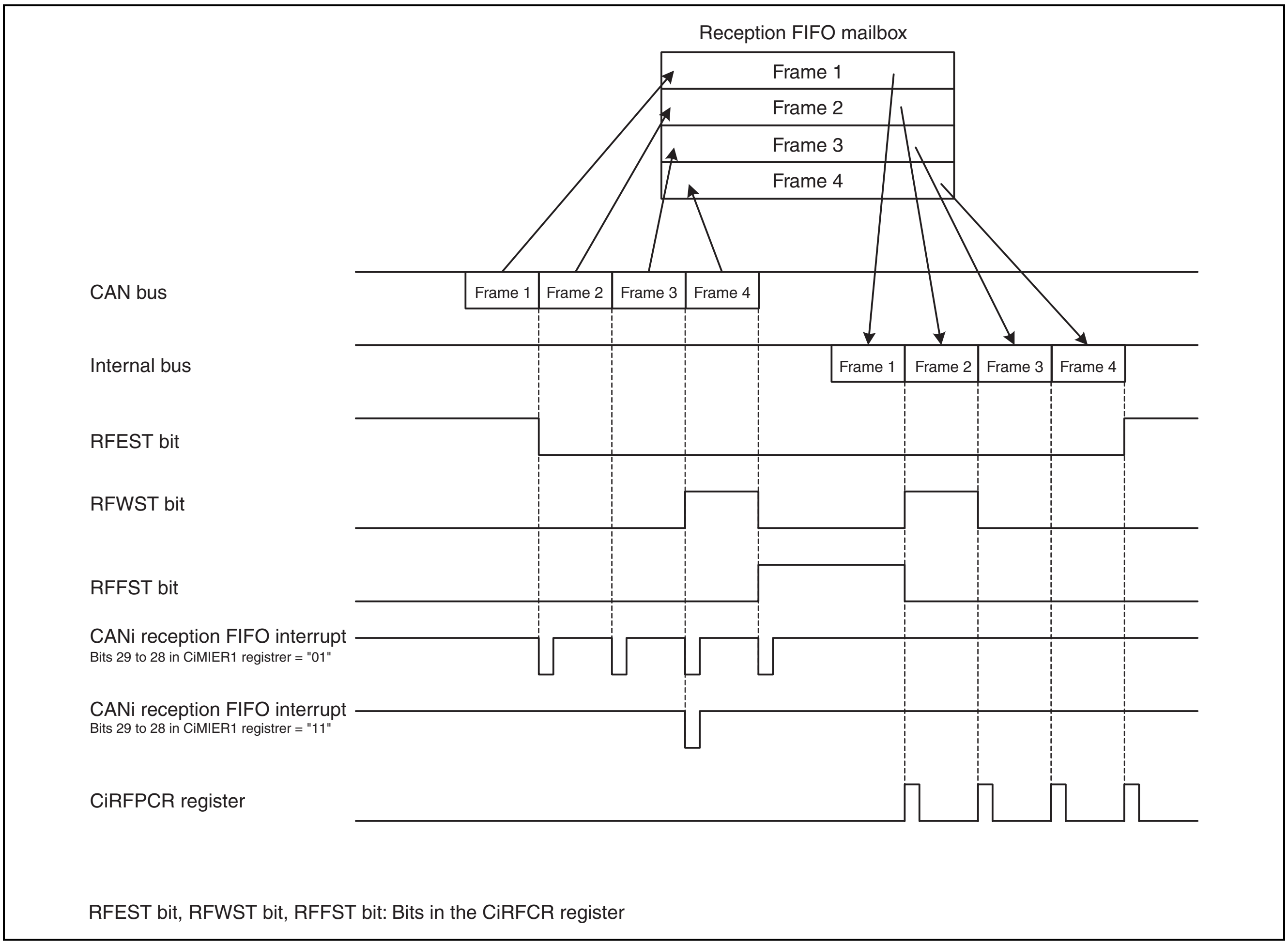

Reception FIFO mailbox
Frame 1
Frame 2
Frame 3
Frame 4
CAN bus
Frame 1
Frame 2
Frame 3
Frame 4
Internal bus
Frame 1
Frame 2
Frame 3
Frame 4
RFEST bit
RFWST bit
RFFST bit
CANi reception FIFO interrupt
Bits 29 to 28 in CiMIER1 registrer = "01"
CANi reception FIFO interrupt
Bits 29 to 28 in CiMIER1 registrer = "11"
CiRFPCR register
RFEST bit, RFWST bit, RFFST bit: Bits in the CiRFCR register

Figure 26.2 Receive FIFO Mailbox Operation (Bits 29 and 28 in CiMIER1 Register = "01" and "11") (i = 0 to 3)

26.3.11 CANi Receive FIFO Pointer Control Register (CiRFPCR) (i = 0 to 3)

When the receive FIFO is not empty, write "H'FF" to the CiRFPCR register by a program to increment the CPU-side pointer for the receive FIFO to the next mailbox location.

Do not write to the CiRFPCR register when the RFE bit in the CiRFCR register is "0" (receive FIFO disabled).

Both the CAN-side pointer and the CPU-side pointer are incremented when a new message is received while the RFFST bit is "1" (receive FIFO is full) in overwrite mode. When the RFMLF bit is "1" in this condition, the CPU-side pointer cannot be incremented by writing to the CiRFPCR register by a program.

CAN0 Receive FIFO Pointer Control Register (C0RFPCR)	<P4 address: location H'FFFF 6849>
CAN1 Receive FIFO Pointer Control Register (C1RFPCR)	<P4 address: location H'FFFF 7849>
CAN2 Receive FIFO Pointer Control Register (C2RFPCR)	<P4 address: location H'FFFF 8849>
CAN3 Receive FIFO Pointer Control Register (C3RFPCR)	<P4 address: location H'FFFF 9849>

Bit:	7	6	5	4	3	2	1	0
	CiRFPCR							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	CiRFPCR	Undefined	R	W	The CPU-side pointer for the receive FIFO is incremented by writing "H'FF"

26.3.12 CANi Transmit FIFO Control Register (CiTFCR) (i = 0 to 3)

Write to the CiTFCR register in CAN operation mode or CAN halt mode.

CAN0 Transmit FIFO Control Register (C0TFCR)	<P4 address: location H'FFFF 684A>
CAN1 Transmit FIFO Control Register (C1TFCR)	<P4 address: location H'FFFF 784A>
CAN2 Transmit FIFO Control Register (C2TFCR)	<P4 address: location H'FFFF 884A>
CAN3 Transmit FIFO Control Register (C3TFCR)	<P4 address: location H'FFFF 984A>

Bit:	7	6	5	4	3	2	1	0
	TFEST	TFFST	—	—	TFUST			TFE
After Reset:	1	0	0	0	0	0	0	0

<After Reset: H'80>

Bit	Abbreviation	After Reset	R	W	Description
7	TFEST	1	R	0	Transmit FIFO Empty Status Bit The TFEST bit is set to "1" (no message in transmit FIFO) when the number of unsent messages in the transmit FIFO is "0". The TFEST bit is set to "1" when transmission from the transmit FIFO has been aborted. The TFEST bit is set to "0" (message in transmit FIFO) when the number of unsent messages in the transmit FIFO is not "0". 0: Unsent message in transmit FIFO 1: No unsent message in transmit FIFO
6	TFFST	0	R	0	Transmit FIFO Full Status Bit The TFFST bit is set to "1" (transmit FIFO is full) when the number of unsent messages in the transmit FIFO is 4. The TFFST bit is set to "0" (transmit FIFO is not full) when the number of unsent messages in the transmit FIFO is less than 4. The TFFST bit is set to "0" when transmission from the transmit FIFO has been aborted. 0: Transmit FIFO is not full 1: Transmit FIFO is full (4 unsent messages)
5	—	0	?	0	Reserved Bit Should be written with "0" and read as undefined value.
4	—	0	0	0	No Register Bit Should be written with "0" and read as "0".

Bit	Abbreviation	After Reset	R	W	Description
3 to 1	TFUST	All 0	R	0	Transmit FIFO Unsent Message Number Status Bit The TFUST bit indicates the number of unsent messages in the transmit FIFO. After the TFE bit is set to "0", the value of the TFUST bit is initialized to "000" when transmission abort or transmission is completed. 000: No unsent message 001: 1 unsent message 010: 2 unsent messages 011: 3 unsent messages 100: 4 unsent messages 101: Reserved 110: Reserved 111: Reserved
0	TFE	0	R	W	Transmit FIFO Enable Bit When the TFE bit is set to "1", the transmit FIFO is enabled. When the TFE bit is set to "0", the transmit FIFO becomes empty (TFEST bit = "1") and then unsent messages from the transmit FIFO are lost as described below: • If a message from the transmit FIFO is not scheduled for the next transmission or during transmission. • Following the completion of transmission, a CAN bus error, CAN bus arbitration lost, or entry to CAN halt mode if a message from the transmit FIFO is scheduled for the next transmission or already during transmission. Before setting the TFE bit to set to "1" again, ensure that the TFEST bit has been set to "1". After setting the TFE bit to "1", write transmit data into the CiMB56 register. Do not set the TFE bit to "1" in normal mailbox mode (MBM bit in the CiCTLR register = "0"). 0: Transmit FIFO disabled 1: Transmit FIFO enabled

Figure 26.3 shows the transmit FIFO mailbox operation.

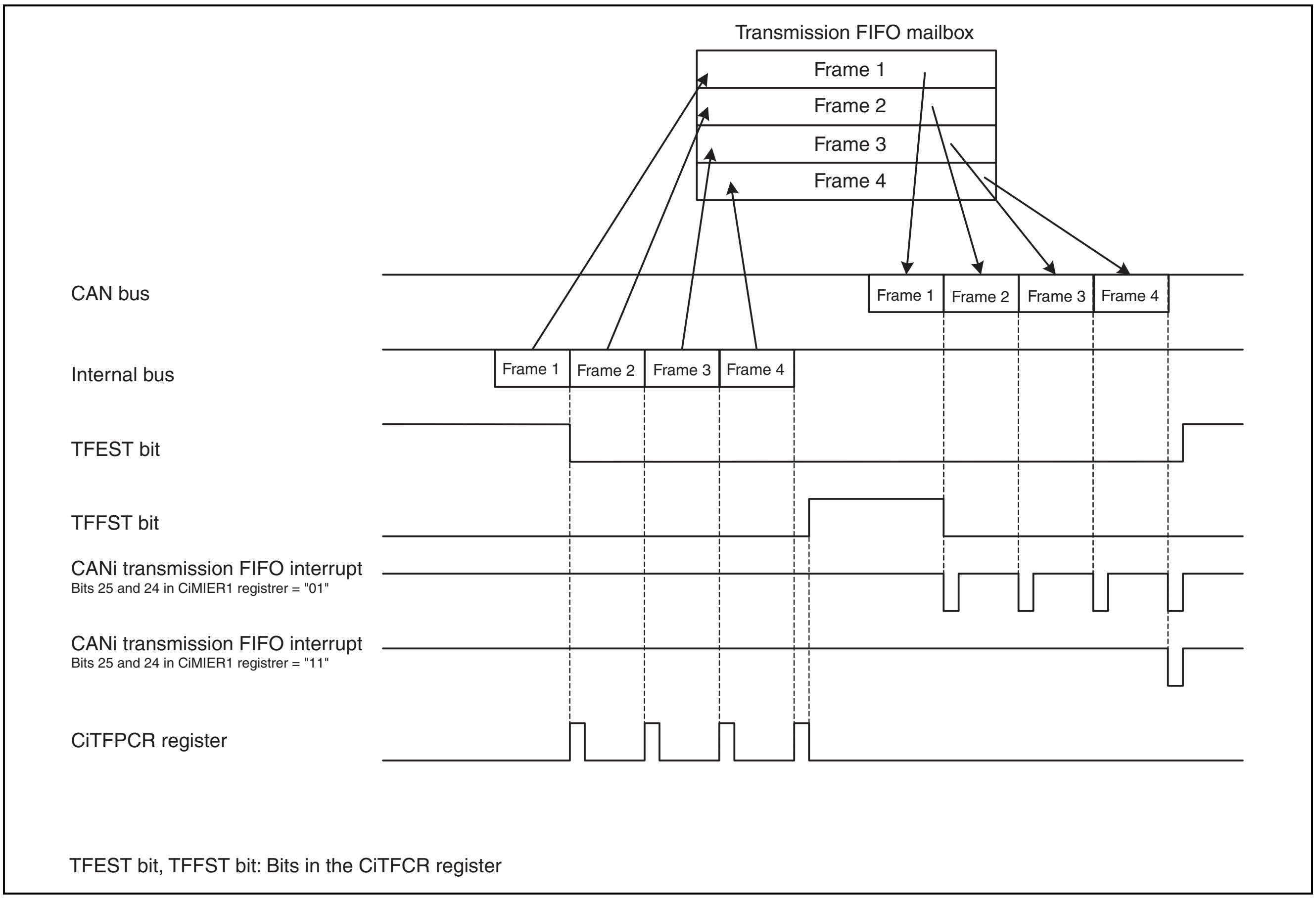

Transmission FIFO mailbox
Frame 1
Frame 2
Frame 3
Frame 4
CAN bus
Frame 1
Frame 2
Frame 3
Frame 4
Internal bus
Frame 1
Frame 2
Frame 3
Frame 4
TFEST bit
TFFST bit
CANi transmission FIFO interrupt
Bits 25 and 24 in CiMIER1 registrer = "01"
CANi transmission FIFO interrupt
Bits 25 and 24 in CiMIER1 registrer = "11"
CiTFPCR register
TFEST bit, TFFST bit: Bits in the CiTFCR register

Figure 26.3 Transmit FIFO Mailbox Operation (Bits 25 and 24 in CiMIER1 Register = "01" and "11") (i = 0 to 3)

26.3.13 CANi Transmit FIFO Pointer Control Register (CiTFPCR) (i = 0 to 3)

When the transmit FIFO is not full, write "H'FF" to the CiTFPCR register by a program to increment the CPU-side pointer for the transmit FIFO to the next mailbox location.

Do not write to the CiTFPCR register when the TFE bit in the CiTFPCR register is "0" (transmit FIFO disabled).

CAN0 Transmit FIFO Pointer Control Register (C0TFPCR)	<P4 address: location H'FFFF 684B>
CAN1 Transmit FIFO Pointer Control Register (C1TFPCR)	<P4 address: location H'FFFF 784B>
CAN2 Transmit FIFO Pointer Control Register (C2TFPCR)	<P4 address: location H'FFFF 884B>
CAN3 Transmit FIFO Pointer Control Register (C3TFPCR)	<P4 address: location H'FFFF 984B>

Bit:	7	6	5	4	3	2	1	0
	CiTFPCR							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	CiTFPCR	Undefined	R	W	The CPU-side pointer for the transmit FIFO is incremented by writing "H'FF"

26.3.14 CANi Status Register (CiSTR) (i = 0 to 3)

CAN0 Status Register (C0STR)	<P4 address: location H'FFFF 6842>
CAN1 Status Register (C1STR)	<P4 address: location H'FFFF 7842>
CAN2 Status Register (C2STR)	<P4 address: location H'FFFF 8842>
CAN3 Status Register (C3STR)	<P4 address: location H'FFFF 9842>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RECST	TRMST	BOST	EPST	SLPST	HLTST	RSTST	EST	TABST	FMLST	NMLST	TFST	RFST	SDST	NDST
After Reset:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0

<After Reset: H'0500>

Bit	Abbreviation	After Reset	R	W	Description
15	—	0	R	N	No Register Bit This bit is read as "0".
14	RECST	0	R	N	Receive Status Flag (receiver) The RECST bit is set to "1" when the CAN module performs as a receiver node. The RECST bit is set to "0" when the CAN module performs as a transmitter node or is in bus-idle state. 0: Bus idle or transmission in progress 1: Reception in progress
13	TRMST	0	R	N	Transmit Status Flag (transmitter) The TRMST bit is set to "1" when the CAN module performs as a transmitter node or is in the bus-off state. The TRMST bit is set to "0" when the CAN module performs as a receiver node or is in bus-idle state. 0: Bus idle or reception in progress 1: Transmission in progress or in bus-off state
12	BOST	0	R	N	Bus-Off Status Flag The BOST bit is set to "1" when the value of the CiTECR register exceeds 255 and the CAN module is in the bus-off state (TEC ≥ 256). The BOST bit is set to "0" when the CAN module is not in the bus-off state. 0: Not in bus-off state 1: In bus-off state
11	EPST	0	R	N	Error-Passive Status Flag The EPST bit is set to "1" when the value of the CiTECR or CiRECR register exceeds 127 and the CAN module is in error-passive state (128 ≤ TEC < 256 or 128 ≤ REC < 256). The EPST bit is set to "0" when the CAN module is not in the error-passive state. TEC indicates the value of the transmit error counter (CiTECR register) and REC indicates the value of the receive error counter (CiRECR register). 0: Not in error-passive state 1: In error-passive state
10	SLPST	1	R	N	CAN Sleep Status Flag The SLPST bit is set to "1" when the CAN module is in CAN sleep mode. The SLPST bit is set to "0" when the CAN module is not in CAN sleep mode. 0: Not in CAN sleep mode 1: In CAN sleep mode

Bit	Abbreviation	After Reset	R	W	Description
9	HLTST	0	R	N	CAN Halt Status Flag The HLTST bit is set to "1" when the CAN module is in CAN halt mode. The HLTST bit is set to "0" when the CAN module is not in CAN halt mode. Even when the state is changed from CAN halt mode to CAN sleep mode, the HLTST bit remains "1". 0: Not in CAN halt mode 1: In CAN halt mode
8	RSTST	1	R	N	CAN Reset Status Flag The RSTST bit is set to "1" when the CAN module is in CAN reset mode. The RSTST bit is "0" when the CAN module is not in CAN reset mode. Even when the state is changed from CAN reset mode to CAN sleep mode, the RSTST bit remains "1". 0: Not in CAN reset mode 1: In CAN reset mode
7	EST	0	R	N	Error Status Flag The EST bit is "1" when at least one error is detected by the CiEIFR register regardless of the value of the CiEIER register. The EST bit is set to "0" when no error is detected by the CiEIFR register. 0: No error occurred 1: Error occurred
6	TABST	0	R	N	Transmission Abort Status Flag The TABST bit is set to "1" when at least one TRMABT bit in the CiMCTLj register is "1" regardless of the value of the CiMIER register. The TABST bit is set to "0" when all TRMABT bits are "0". 0: No mailbox with TRMABT bit = "1" 1: Mailbox(es) with TRMABT bit = "1"
5	FMLST	0	R	N	FIFO Mailbox Message Lost Status Flag The FMLST bit is set to "1" when the RFMLF bit in the CiRFCR register is "1" regardless of the value of the CiMIER register. The FMLST bit is set to "0" when the RFMLF bit is "0". 0: RFMLF bit = "0" 1: RFMLF bit = "1"
4	NMLST	0	R	N	Normal Mailbox Message Lost Status Flag The NMLST bit is set to "1" when at least one MSGLOST bit in the CiMCTLj register is "1" regardless of the value of the CiMIER register. The NMLST bit is set to "0" when all MSGLOST bit is "0". 0: No mailbox with MSGLOST bit = "1" 1: Mailbox(es) with MSGLOST bit = "1"
3	TFST	0	R	N	Transmit FIFO Status Flag The TFST bit is set to "1" when the transmit FIFO is not full. The TFST bit is set to "0" when the transmit FIFO is full. The TFST bit is set to "0" when normal mailbox mode is selected. 0: Transmit FIFO is full 1: Transmit FIFO is not full

Bit	Abbreviation	After Reset	R	W	Description
2	RFST	0	R	N	Receive FIFO Status Flag The RFST bit is set to "1" when the receive FIFO is not empty. The RFST bit is set to "0" when the receive FIFO is empty. The RFST bit is set to "0" when normal mailbox mode is selected. 0: No message in receive FIFO 1: Message in receive FIFO
1	SDST	0	R	N	SENTDATA Status Flag The SDST bit is set to "1" when at least one SENTDATA bit in the CiMCTLj (j = 32 to 63) register is "1" regardless of the value of the CiMIER register. The SDST bit is set to "0" when all SENTDATA bits are "0". 0: No mailbox with SENTDATA bit = "1" 1: Mailbox(es) with SENTDATA bit = "1"
0	NDST	0	R	N	NEWDATA Status Flag The NDST bit is set to "1" when at least one NEWDATA bit in the CiMCTLj (j = 0 to 63) register is "1" regardless of the value of the CiMIER register. The NDST bit is set to "0" when all NEWDATA bits are "0". 0: No mailbox with NEWDATA bit = "1" 1: Mailbox(es) with NEWDATA bit = "1"

26.3.15 CANi Mailbox Search Mode Register (CiMSMR) (i = 0 to 3)

Write to the CiMSMR register in CAN operation mode or CAN halt mode.

CAN0 Mailbox Search Mode Register (C0MSMR)	<P4 address: location H'FFFF 6853>
CAN1 Mailbox Search Mode Register (C1MSMR)	<P4 address: location H'FFFF 7853>
CAN2 Mailbox Search Mode Register (C2MSMR)	<P4 address: location H'FFFF 8853>
CAN3 Mailbox Search Mode Register (C3MSMR)	<P4 address: location H'FFFF 9853>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MBSM	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	No Register Bits Should be written with "0" and read as "0".
1, 0	MBSM	All 0	R	W	Mailbox Search Mode Select Bits The MBSM bit selects the search mode for the mailbox search function. When the MBSM bit is "00", receive mailbox search mode is selected. In this mode, the search targets are the NEWDATA bit in the CiMCTLj register (j = 0 to 63) for the normal mailbox and the RFEST bit in the CiRFCR register. When the MBSM bit is "01", transmit mailbox search mode is selected. In this mode, targets the SENTDATA bit in the CiMCTLj register. When the MBSM bit is "10", message lost search mode is selected. In this mode, targets the MSGLOST bit in the CiMCTLj register for the normal mailbox and the RFMLF bit in the CiRFCR register. When the MBSM bit is "11", channel search mode is selected. In this mode, the search target is the CiCSSR register. Refer to section 26.3.17, CANi Channel Search Support Register (CiCSSR) (i = 0 to 3). 00: Receive mailbox search mode 01: Transmit mailbox search mode 10: Message lost search mode 11: Channel search mode

26.3.16 CANi Mailbox Search Status Register (CiMSSR) (i = 0 to 3)

CAN0 Mailbox Search Status Register (C0MSSR)	<P4 address: location H'FFFF 6852>
CAN1 Mailbox Search Status Register (C1MSSR)	<P4 address: location H'FFFF 7852>
CAN2 Mailbox Search Status Register (C2MSSR)	<P4 address: location H'FFFF 8852>
CAN3 Mailbox Search Status Register (C3MSSR)	<P4 address: location H'FFFF 9852>

Bit:	7	6	5	4	3	2	1	0
	SEST	—	MBNST					
After Reset:	1	0	0	0	0	0	0	0

<After Reset: H'80>

Bit	Abbreviation	After Reset	R	W	Description
7	SEST	1	R	0	Search Result Status Bit The SEST bit is set to "1" when no corresponding mailbox is found after searching all mailboxes. For example, in transmit mailbox search mode, the SEST bit is set to "1" when no SENTDATA bit for mailboxes is "1". The SEST bit is set to "0" when at least one SENTDATA bit is "1". When the SEST bit is "1", the value of the MBNST bits is undefined. 0: Search result found 1: No search result
6	—	0	0	0	No Register Bit Should be written with "0" and read as "0".
5 to 0	MBNST	All 0	R	0	Search Result Mailbox Number Status Bit The MBNST bit outputs the smallest mailbox number that is searched in each mode of the CiMSMR register. In receive mailbox, transmit mailbox, and message lost search modes, the value of the mailbox i.e., the search result to be output, is updated as described below: • When the NEWDATA, SENTDATA or MSGLOST bit for the output mailbox is set to "0". • When the NEWDATA, SENTDATA or MSGLOST bit for a higher-priority mailbox is set to "1". In receive mailbox search and message lost search modes, the receive FIFO (mailbox [60]) is output when the receive FIFO is not empty and there are no unread received messages or no lost messages in any of the normal mailboxes (mailboxes [0] to [55]). In transmit mailbox search mode, the transmit FIFO (mailbox [56]) is not output. Table 26.7 lists the behavior of MBNST bit in FIFO mailbox mode. In channel search mode, the MBNST bit outputs the corresponding channel number. After the CiMSSR register is read by a program, the next target channel number is output.

Table 26.7 Behavior of MBNST Bit in FIFO Mailbox Mode

MBSM Bit	Mailbox [56] (Transmit FIFO)	Mailbox [60] (Receive FIFO)
"00"	Mailbox [56] is not output.	Mailbox [60] is output when no NEWDATA bit for the normal mailbox is set to "1" and the receive FIFO is not empty.
"01"		Mailbox [60] is not output.
"10"		Mailbox [60] is output when no MSGLOST bit for the normal mailbox is set to "1" and the RFMLF bit is set to "1" (receive FIFO message lost has occurred) in the receive FIFO.
"11"		Mailbox [60] is not output.

26.3.17 CANi Channel Search Support Register (CiCSSR) (i = 0 to 3)

The bits in the CiCSSR register, which are set to "1", are encoded by an 8/3 encoder (the lower bit position, the higher priority) and output to the MBNST bits in the CiMSSR register.

The CiMSSR register outputs the updated value whenever the CiMSSR register is read by a program.

Write to the CiCSSR register only when the MBSM bit in the CiMSMR register is "11" (channel search mode). Write to this register in CAN operation mode or CAN halt mode.

CAN0 Channel Search Support Register (C0CSSR)	<P4 address: location H'FFFF 6851>
CAN1 Channel Search Support Register (C1CSSR)	<P4 address: location H'FFFF 7851>
CAN2 Channel Search Support Register (C2CSSR)	<P4 address: location H'FFFF 8851>
CAN3 Channel Search Support Register (C3CSSR)	<P4 address: location H'FFFF 9851>

Bit:	7	6	5	4	3	2	1	0
	CiCSSR							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	CiCSSR	Undefined	R	W	When the value for the channel search is input, the channel number is output to the CiMSSR register.

Figure 26.4 shows the write and read of registers CiCSSR and CiMSSR.

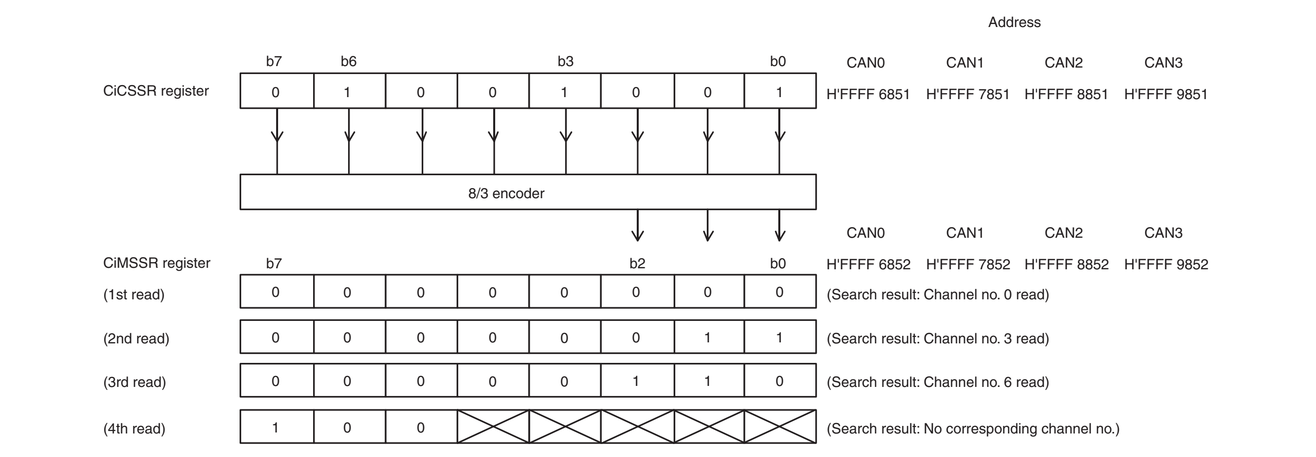


Figure 26.4 Write and Read of Registers CiCSSR and CiMSSR (i = 0 to 3)

The value of the CiCSSR register is also updated whenever the CiMSSR register is read. When the CiCSSR register is read, the value before the 8/3 encoder conversion is read.

26.3.18 CANi Acceptance Filter Support Register (CiAFSR) (i = 0 to 3)

The acceptance filter support unit (ASU) can be used for data table (8 bits × 256) search. In the data table, all standard IDs created by the user are set to be valid/invalid in bit units. When the CAFSR register is written with the 16-bit unit data including the SID bit in the CiMBj register (j = 0 to 63), in which a received ID is stored, a decoded row (byte offset) position and column (bit) position for data table search can be read. The ASU can be used for standard (11-bit) IDs only.

The ASU is enabled in the following cases:

- When the ID to receive cannot be masked by the acceptance filter.
  (Example) IDs to receive: H'078, H'087, H'111
- When there are too many IDs to receive and software filtering time is expected to be shortened.

Write to the CiAFSR register in CAN operation mode or CAN halt mode.

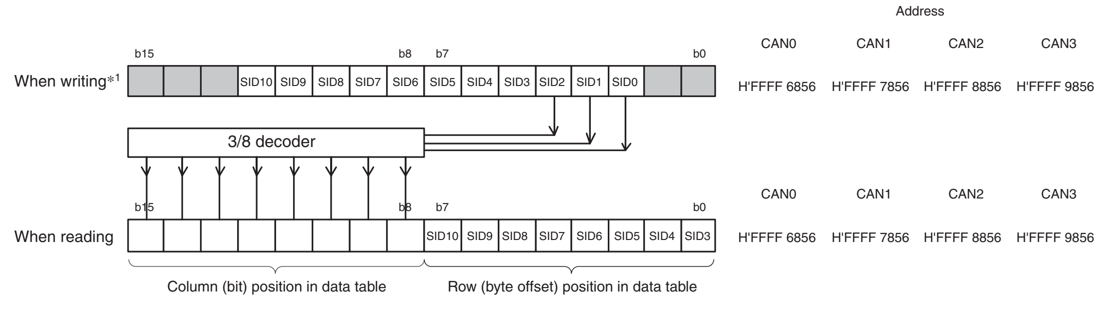
CAN0 Acceptance Filter Support Register (C0AFSR)	<P4 address: location H'FFFF 6856>
CAN1 Acceptance Filter Support Register (C1AFSR)	<P4 address: location H'FFFF 7856>
CAN2 Acceptance Filter Support Register (C2AFSR)	<P4 address: location H'FFFF 8856>
CAN3 Acceptance Filter Support Register (C3AFSR)	<P4 address: location H'FFFF 9856>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CiAFSR															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	CiAFSR	Undefined	R	W	After the standard ID of a received message is written, the value converted for data table search can be read.

Figure 26.5 shows the write and read of CiAFSR register.



Note: *1 Write the same value as the 16-bit unit data including the SID bit in the CiMBj (j = 0 to 63) register.

Figure 26.5 Write and Read of CiAFSR Register (i = 0 to 3)

26.3.19 CANi Error Interrupt Enable Register (CiEIER) (i = 0 to 3)

The CiEIER register is used to set the error interrupt enabled/disabled individually for each error interrupt source in the CiEIFR register.

Write to the CiEIER register in CAN reset mode.

CAN0 Error Interrupt Enable Register (C0EIER)	<P4 address: location H'FFFF 684C>
CAN1 Error Interrupt Enable Register (C1EIER)	<P4 address: location H'FFFF 784C>
CAN2 Error Interrupt Enable Register (C2EIER)	<P4 address: location H'FFFF 884C>
CAN3 Error Interrupt Enable Register (C3EIER)	<P4 address: location H'FFFF 984C>

Bit:	7	6	5	4	3	2	1	0
	BLIE	OLIE	ORIE	BORIE	BOEIE	EPIE	EWIE	BEIE
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	BLIE	0	R	W	Bus Lock Interrupt Enable Bit When the BLIE bit is "0", no error interrupt request is generated even if the BLIF bit in the CiEIFR register is set to "1". When the BLIE bit is "1", an error interrupt request is generated if the BLIF bit is set to "1". 0: Bus lock interrupt disabled 1: Bus lock interrupt enabled
6	OLIE	0	R	W	Overload Frame Transmit Interrupt Enable Bit When the OLIE bit is "0", no error interrupt request is generated even if the OLIF bit in the CiEIFR register is set to "1". When the OLIE bit is "1", an error interrupt request is generated if the OLIF bit is set to "1". 0: Overload frame transmit interrupt disabled 1: Overload frame transmit interrupt enabled
5	ORIE	0	R	W	Receive Overrun Interrupt Enable Bit When the ORIE bit is "0", an error interrupt request is not generated even if the ORIF bit in the CiEIFR register is set to "1". When the ORIE bit is "1", an error interrupt request is generated if the ORIF bit is set to "1". 0: Receive overrun interrupt disabled 1: Receive overrun interrupt enabled
4	BORIE	0	R	W	Bus-Off Recovery Interrupt Enable Bit When the BORIE bit is "0", an error interrupt request is not generated even if the BORIF bit in the CiEIFR register is set to "1". When the BORIE bit is set to "1", an error interrupt request is generated if the BORIF bit is set to "1". 0: Bus-off recovery interrupt disabled 1: Bus-off recovery interrupt enabled

Bit	Abbreviation	After Reset	R	W	Description
3	BOEIE	0	R	W	Bus-Off Entry Interrupt Enable Bit When the BOEIE bit is "0", no error interrupt request is generated even if the BOEIF bit in the CiEIFR register is set to "1". When the BOEIE bit is "1", an error interrupt request is generated if the BOEIF bit is set to "1". 0: Bus-off entry interrupt disabled 1: Bus-off entry interrupt enabled
2	EPIE	0	R	W	Error-Passive Interrupt Enable Bit When the EPIE bit is "0", no error interrupt request is generated even if the EPIF bit in the CiEIFR register is set to "1". When the EPIE bit is "1", an error interrupt request is generated if the EPIF bit is set to "1". 0: Error-passive interrupt disabled 1: Error-passive interrupt enabled
1	EWIE	0	R	W	Error-Warning Interrupt Enable Bit When the EWIE bit is "0", no error interrupt request is generated even if the EWIF bit in the CiEIFR register is set to "1". When the EWIE bit is "1", an error interrupt request is generated if the EWIF bit is set to "1". 0: Error-warning interrupt disabled 1: Error-warning interrupt enabled
0	BEIE	0	R	W	Bus Error Interrupt Enable Bit When the BEIE bit is "0", no error interrupt request is generated even if the BEIF bit in the CiEIFR register is set to "1". When the BEIE bit is "1", an error interrupt request is generated if the BEIF bit is set to "1". 0: Bus error interrupt disabled 1: Bus error interrupt enabled

26.3.20 CANi Error Interrupt Factor Judge Register (CiEIFR) (i = 0 to 3)

If an event corresponding to each bit occurs, the corresponding bit in the CiEIFR register is set to "1" regardless of the setting of the CiEIER register.

To set each bit to "0", write "0" by a program. If the set timing occurs simultaneously with the clear timing by the program, the bit becomes "1".

When writing "0" to a single bit by a program, use the MOV instruction to ensure that only the specified bit is set to "0" and the other bits are set to "1". Writing "1" has no effect to these bit values.

CAN0 Error Interrupt Factor Judge Register (C0EIFR)	<P4 address: location H'FFFF 684D>
CAN1 Error Interrupt Factor Judge Register (C1EIFR)	<P4 address: location H'FFFF 784D>
CAN2 Error Interrupt Factor Judge Register (C2EIFR)	<P4 address: location H'FFFF 884D>
CAN3 Error Interrupt Factor Judge Register (C3EIFR)	<P4 address: location H'FFFF 984D>

Bit:	7	6	5	4	3	2	1	0
	BLIF	OLIF	ORIF	BORIF	BOEIF	EPIF	EWIF	BEIF
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	BLIF	0	R	W	Bus Lock Detect Flag*[1] The BLIF bit is set to "1" if 32 consecutive dominant bits are detected on the CAN bus while the CAN module is in CAN operation mode. After the bit is set to "1", redetection takes place under either of the following conditions: • After this bit is set to "0" from "1", recessive bits are detected. • After this bit is set to "0" from "1", the CAN module enters CAN reset mode or CAN halt mode and then enters CAN operation mode again. 0: No bus lock detected 1: Bus lock detected
6	OLIF	0	R	W	Overload Frame Transmission Detect Flag*[1] The OLIF bit is set to "1" if the transmitting condition of an overload frame is detected when the CAN module performs transmission or reception. 0: No overload frame transmission detected 1: Overload frame transmission detected
5	ORIF	0	R	W	Receive Overrun Detect Flag*[1] The ORIF bit is set to "1" when a receive overrun occurs. This bit is not to set to "1" in overwrite mode. In overwrite mode, a reception complete interrupt request is generated if an overwrite condition occurs, and the ORIF bit is not set to "1". In normal mailbox mode, if an overrun occurs in any of mailboxes [0] to [63] in overrun mode, this bit is set to "1". In FIFO mailbox mode, if an overrun occurs in any of mailboxes [0] to [55] or the receive FIFO in overrun mode, this bit is set to "1". 0: No receive overrun detected 1: Receive overrun detected

Bit	Abbreviation	After Reset	R	W	Description
4	BORIF	0	R	W	Bus-Off Recovery Detect Flag*1 The BORIF bit is set to "1" when the CAN module recovers from the bus-off state normally by detecting 11 consecutive bits 128 times in the following conditions: • When the BOM bit in the CiCTLR register is "00". • When the BOM bit is "10". • When the BOM bit is "11". The BORIF bit is not set to "1" if the CAN module recovers from the bus-off state in the following conditions: • When the CANM bit in the CiCTLR register is set to "01" or "11" (CAN reset mode). • When the RBOC bit in the CiCTLR register is set to "1" (forcible return from bus-off). • When the BOM bit is "01". • When the BOM bit is "11" and the CANM bit is set to "10" (CAN halt mode) before normal recovery occurs. 0: No bus-off recovery detected 1: Bus-off recovery detected
3	BOEIF	0	R	W	Bus-Off Entry Detect Flag*1 The BOEIF bit is set to "1" when the CAN error state becomes bus-off (the TEC value exceeds 255). The BOEIF bit is also set to "1" when the BOM bit in the CiCTLR register is "01" (entry to CAN halt mode automatically at bus-off entry) and the CAN module becomes the bus-off state. 0: No bus-off entry detected 1: Bus-off entry detected
2	EPIF	0	R	W	Error Passive Detect Flag*1 The EPIF bit is set to "1" when the CAN error state becomes error-passive (the REC or TEC value exceeds 127). The EPIF bit is set to "1" only when the REC or TEC initially exceeds 127. Thus, if "0" is written to the EPIF bit by a program while the REC or TEC remains greater than 127, the EPIF bit is not set to "1" until the REC and TEC goes below 127 and then exceeds 127 again. 0: No error passive detected 1: Error passive detected
1	EWIF	0	R	W	Error Warning Detect Flag*1 The EWIF bit is set to "1" when the value of the receive error counter (REC) or transmit error counter (TEC) exceeds 95. The EWIF bit is set to "1" only when the REC or TEC initially exceeds 95. Thus, if "0" is written to the EWIF bit by a program while the REC or TEC remains greater than 95, the EWIF bit is not set to "1" until the REC and TEC goes below 95 and then exceeds 95 again. 0: No error warning detected 1: Error warning detected

Bit	Abbreviation	After Reset	R	W	Description
0	BEIF	0	R	W	Bus Error Detect Flag*[1] The BEIF bit is set to "1" when a bus error is detected. 0: No bus error detected 1: Bus error detected

Note: *1 Only "0" may be written to this bit. (Writing "1" has no effect.) When writing "0" to specific bits in software, use the MOV instruction and write "0" to each bit to be cleared to "0" and "1" to all other bits.

Table 26.8 lists the behavior of bits BOEIF and BORIF according to BOM bit setting value.

Table 26.8 Behavior of Bits BOEIF and BORIF according to BOM Bit Setting Value

BOM Bit	BOEIF Bit	BORIF Bit
00	Set to "1" on entry to the bus-off state.	Set to "1" on exit from the bus-off state.
01		Do not set to "1".
10		Set to "1" on exit from the bus-off state.
11		Set to "1" if normal bus-off recovery occurs before the CANM bit is set to "10" (CAN halt mode).

26.3.21 CANi Receive Error Count Register (CiRECR) (i = 0 to 3)

The CiRECR register indicates the value of the receive error counter.

Refer to the CAN Specifications (ISO11898-1) about the increment/decrement conditions of the receive error counter.

The value in bus-off state is undefined.

CAN0 Receive Error Count Register (C0RECR)	<P4 address: location H'FFFF 684E>
CAN1 Receive Error Count Register (C1RECR)	<P4 address: location H'FFFF 784E>
CAN2 Receive Error Count Register (C2RECR)	<P4 address: location H'FFFF 884E>
CAN3 Receive Error Count Register (C3RECR)	<P4 address: location H'FFFF 984E>

Bit:	7	6	5	4	3	2	1	0
	CiRECR							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	CiRECR	All 0	R	N	Receive Error Count Function The CiRECR register increments or decrements the counter value according to error status of the CAN module during reception.

26.3.22 CANi Transmit Error Count Register (CiTECR) (i = 0 to 3)

The CiTECR register indicates the value of the TEC error counter.

Refer to the CAN Specifications (ISO11898-1) about the increment/decrement conditions of the transmit error counter.

The value in bus-off state is undefined.

CAN0 Transmit Error Count Register (C0TECR)	<P4 address: location H'FFFF 684F>
CAN1 Transmit Error Count Register (C1TECR)	<P4 address: location H'FFFF 784F>
CAN2 Transmit Error Count Register (C2TECR)	<P4 address: location H'FFFF 884F>
CAN3 Transmit Error Count Register (C3TECR)	<P4 address: location H'FFFF 984F>

Bit:	7	6	5	4	3	2	1	0
	CiTECR							
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	CiTECR	All 0	R	N	Transmit Error Count Function The CiTECR register increments or decrements the counter value according to error status of the CAN module during transmission.

26.3.23 CANi Error Code Store Register (CiECSR) (i = 0 to 3)

The CiECSR register can be used to monitor whether an error has occurred on the CAN bus. Refer to the CAN Specifications (ISO11898-1) to check the generation conditions of each error.

To set each bit except the EDPM bit to "0", write "0" by a program. If the timing at which each bit is set to "1" and the timing at which "0" is written by a program are the same, the relevant bit is set to "1".

CAN0 Error Code Store Register (C0ECSR)	<P4 address: location H'FFFF 6850>
CAN1 Error Code Store Register (C1ECSR)	<P4 address: location H'FFFF 7850>
CAN2 Error Code Store Register (C2ECSR)	<P4 address: location H'FFFF 8850>
CAN3 Error Code Store Register (C3ECSR)	<P4 address: location H'FFFF 9850>

Bit:	7	6	5	4	3	2	1	0
	EDPM	ADEF	BE0F	BE1F	CEF	AEF	FEF	SEF
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	EDPM	0	R	W	Error Display Mode Select Bit*1*2 The EDPM bit selects the output mode of the CiECSR register. When the EDPM bit is set to "0", the CiECSR register outputs the first error code. When the EDPM bit is set to "1", the CiECSR register outputs the accumulated error code. 0: Output of first detected error code 1: Output of accumulated error code
6	ADEF	0	R	W	ACK Delimiter Error Flag*3*4 The ADEF bit is set to "1" when a form error is detected with the ACK delimiter during transmission. 0: No ACK delimiter error detected 1: ACK delimiter error detected
5	BE0F	0	R	W	Bit Error (dominant) Flag*3*4 The BE0F bit is set to "1" when a dominant bit error is detected. 0: No bit error (dominant) detected 1: Bit error (dominant) detected
4	BE1F	0	R	W	Bit Error (recessive) Flag*3*4 The BE1F bit is set to "1" when a recessive bit error is detected. 0: No bit error (recessive) detected 1: Bit error (recessive) detected
3	CEF	0	R	W	CRC Error Flag*3*4 The CEF bit is set to "1" when a CRC error is detected. 0: No CRC error detected 1: CRC error detected

Bit	Abbreviation	After Reset	R	W	Description
2	AEF	0	R	W	ACK Error Flag[*3][*4] The AEF bit is set to "1" when an ACK error is detected. 0: No ACK error detected 1: ACK error detected
1	FEF	0	R	W	Form Error Flag[*3][*4] The FEF bit is set to "1" when a form error is detected. 0: No form error detected 1: Form error detected
0	SEF	0	R	W	Stuff Error Flag[*3][*4] The SEF bit is set to "1" when a stuff error is detected. 0: No stuff error detected 1: Stuff error detected

Notes: *1 Write to the EDPM bit in CAN reset mode or CAN halt mode.

*2 If more than one error condition is detected simultaneously, all related bits are set to "1".

*3 Writing "1" has no effect to these bit values.

*4 When writing "0" to bits SEF, FEF, AEF, CEF, BE1F, BE0F, and ADEF by a program, use the MOV instruction to ensure that only the specified bit is set to "0" and the other bits are set to "1".

26.3.24 CANi Time Stamp Register (CiTSR) (i = 0 to 3)

When the CiTSR register is read, the value of the time stamp counter (16-bit free-running counter) at that moment is read.

The value of the time stamp counter reference clock is a multiple of 1 bit time, as configured by the TSPS bit in the CiCTLR register.

The time stamp counter stops in CAN sleep mode and CAN halt mode, and is initialized in CAN reset mode.

The time stamp counter value is stored to TSL and TSH in the CiMBj register when a received message is stored in a receive mailbox.

CAN0 Time Stamp Register (C0TSR)	<P4 address: location H'FFFF 6854>
CAN1 Time Stamp Register (C1TSR)	<P4 address: location H'FFFF 7854>
CAN2 Time Stamp Register (C2TSR)	<P4 address: location H'FFFF 8854>
CAN3 Time Stamp Register (C3TSR)	<P4 address: location H'FFFF 9854>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CiTSR															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	CiTSR	All 0	R	N	Free-running counter value for the time stamp function

Note: • Read the CiTSR register in 16-bit units.

26.3.25 CANi Test Control Register (CiTCR) (i = 0 to 3)

Write to the CiTCR register in CAN halt mode only.

CAN0 Test Control Register (C0TCR)	<P4 address: location H'FFFF 6858>
CAN1 Test Control Register (C1TCR)	<P4 address: location H'FFFF 7858>
CAN2 Test Control Register (C2TCR)	<P4 address: location H'FFFF 8858>
CAN3 Test Control Register (C3TCR)	<P4 address: location H'FFFF 9858>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	TSTM		TSTE
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits Should be written with "0".
2, 1	TSTM	All 0	R	W	CAN Test Mode Select Bits The TSTM bit selects the CAN test mode. For details on the CAN test modes, see section 26.3.25 (1), Listen-Only Mode, section 26.3.25 (2), Self-Test Mode 0 (External Loop Back), and section 26.3.25 (3), Self-Test Mode 1 (Internal Loop Back). 00: Other than CAN test mode 01: Listen-only mode 10: Self-test mode 0 (external loop back) 11: Self-test mode 1 (internal loop back)
0	TSTE	0	R	W	CAN Test Mode Enable Bit When the TSTE bit is set to "0", CAN test mode is disabled. When the TSTE bit is set to "1", CAN test mode is enabled. 0: CAN test mode disabled 1: CAN test mode enabled

(1) Listen-Only Mode

The ISO 11898-1 recommends an optional bus monitoring mode. In listen-only mode, the CAN node is able to receive valid data frames and valid remote frames. It sends only recessive bits on the CAN bus, and the protocol controller is not required to send the ACK bit, overload flag, or active error flag.

Listen-only mode can be used for baud rate detection.

Do not request transmission from any mailboxes in this mode.

Figure 26.6 shows the connection when listen-only mode is selected.

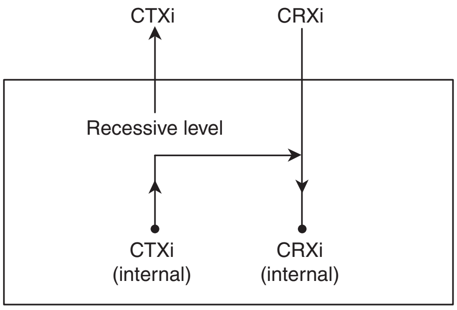


Figure 26.6 Connection when Listen-Only Mode is Selected

(2) Self-Test Mode 0 (External Loop Back)

Self-test mode 0 is provided for CAN transceiver tests.

In this mode, the protocol controller treats its own transmitted messages as messages received via the CAN transceiver and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit.

Connect the CTXi/CRXi pins to the transceiver.

Figure 26.7 shows the connection when self-test mode 0 is selected.

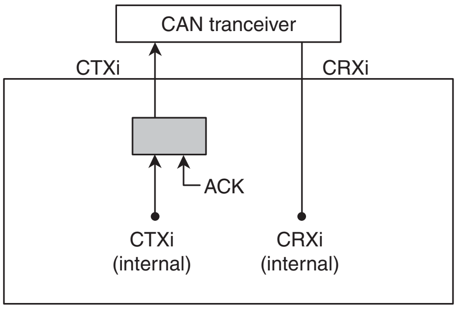


Figure 26.7 Connection when Self-Test Mode 0 is Selected

(3) Self-Test Mode 1 (Internal Loop Back)

Self-test mode 1 is provided for self-test functions.

In this mode, the protocol controller treats its transmitted messages as received messages and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit.

In self-test mode 1, the protocol controller performs an internal feedback from the internal CTXi pin to the internal CRXi pin. The input value of the external CRXi pin is ignored. The external CTXi pin outputs only recessive bits. The CTXi/CRXi pins do not need to be connected to the CAN bus or any external device.

Figure 26.8 shows the connection when self-test mode 1 is selected.

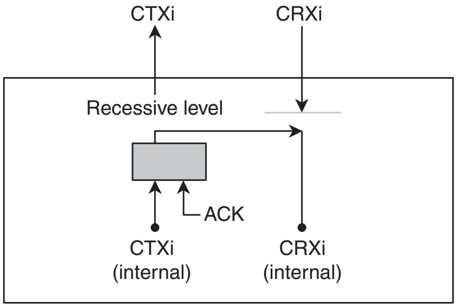


Figure 26.8 Connection when Self-Test Mode 1 is Selected

26.3.26 CANi Interrupt Status Register (CiISR) (i = 0 to 3)

The CiISR register shows interrupt sources before masking by the CiIER register.

CAN0 Interrupt Status Register (C0ISR)	<P4 address: location H'FFFF 6861>
CAN1 Interrupt Status Register (C1ISR)	<P4 address: location H'FFFF 7861>
CAN2 Interrupt Status Register (C2ISR)	<P4 address: location H'FFFF 8861>
CAN3 Interrupt Status Register (C3ISR)	<P4 address: location H'FFFF 9861>

Bit:	7	6	5	4	3	2	1	0
	—	—	ERSF	RXFF	TXFF	RXM0F	RXM1F	TXMF
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	No Register Bits Should be written with "0" and read as "0".
5	ERSF	0	R	0	Error (ERS) Interrupt Status Bit*[1] The ERSF bit shows the error interrupt source status. 0: ERS interrupt source not detected 1: ERS interrupt source detected
4	RXFF	0	R	W	Reception FIFO (RXF) Interrupt Status Bit*[2] The RXFF bit shows the FIFO receive interrupt source status. 0: RXF interrupt source not detected 1: RXF interrupt source detected The RXFF is cleared to "0" by writing "0" to it by software. (writing "1" has no effect)
3	TXFF	0	R	W	Transmission FIFO (TXF) Interrupt Status Bit*[3] The TXFF bit shows the FIFO transmit interrupt source status. 0: TXF interrupt source not detected 1: TXF interrupt source detected The TXFF is cleared to "0" by writing "0" to it by software. (writing "1" has no effect)
2	RXM0F	0	R	0	Mailbox 0 Successful Reception (RXM0) Interrupt Status Bit*[4] The RXM0F bit shows the successful reception interrupt source status for mailbox 0. 0: RXM0 interrupt source not detected 1: RXM0 interrupt source detected
1	RXM1F	0	R	0	Mailbox 1 to 63 Successful Reception (RXM1) Interrupt Status Bit*[5] The RXM1F bit shows the successful reception interrupt source status for mailboxes 1 to 63. 0: RXM1 interrupt source not detected 1: RXM1 interrupt source detected
0	TXMF	0	R	0	Mailbox 32 to 63 Successful Transmission (TXM) Interrupt Status Bit*[6] The TXMF bit shows the successful transmission interrupt source status for mailboxes 32 to 63. 0: TXM interrupt source not detected 1: TXM interrupt source detected

Notes: *1 The ERSF bit is set to "1" when a bit in one of the CiEIFR[j] registers is set to "1" due to a communication error while the corresponding bit in the CiEIER[j] register is set to "1" (j = 0 to 7).

*2 The RXFF bit is set to "1" when bit 6 or 5 in the CiRFCR register is set to "1" because of a reception FIFO full or warning condition due to the setting of CiMIER[61].

*3 The TXFF bit is set to "1" when the transmission FIFO message count reaches the specified value due to the setting of CiMIER[57].

*4 After the NEWDATA bit in CiMCTL0 is set to "1" at the end of a receive operation, the RXM0F bit is set to "1" if storing of the receive data is complete (corresponding INVALDATA bit value changed from "1" to "0") and the CiMIER0[0] bit has the been set to "1".

*5 After the NEWDATA bit in CiMCTLj is set to "1" at the end of a receive operation, the RXM1F bit is set to "1" if storing of the receive data is complete (corresponding INVALDATA bit value changed from "1" to "0") and the bit in CiMIER0 or CiMIER1 corresponding to mailbox j has the been set to "1" (j = 1 to 63).

*6 The TXMF bit is set to "1" when the bit in the CiMIER1 register corresponding to mailbox j is set to "1" while the value of the SENTDATA bit in the CiMCTLj register is "1" following a successful reception (j = 32 to 63).

26.3.27 CANi Interrupt Enable Register (CiIER) (i = 0 to 3)

The CiIER register can be used by an application to cause some interrupts to be ignored while processing by an interrupt service routine is taking place. Each bit affects the individual interrupt source corresponding to it.

CAN0 Interrupt Enable Register (C0IER)	<P4 address: location H'FFFF 6860>
CAN1 Interrupt Enable Register (C1IER)	<P4 address: location H'FFFF 7860>
CAN2 Interrupt Enable Register (C2IER)	<P4 address: location H'FFFF 8860>
CAN3 Interrupt Enable Register (C3IER)	<P4 address: location H'FFFF 9860>

Bit:	7	6	5	4	3	2	1	0
	—	—	ERS IE	RXF IE	TXF IE	RXM0 IE	RXM1 IE	TXM IE
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	No Register Bits Should be written with "0" and read as "0".
5	ERSIE	0	R	W	Error (ERS) Interrupt Enable Bit The ERSIE bit enables or disables the ERS interrupt controller. 0: ERS interrupt disabled 1: ERS interrupt enabled
4	RXFIE	0	R	W	Reception FIFO (RXF) Interrupt Enable Bit The RXFIE bit enables or disables the RXF interrupt controller. 0: RXF interrupt disabled 1: RXF interrupt enabled
3	TXFIE	0	R	W	Transmission FIFO (TXF) Interrupt Enable Bit The TXFIE bit enables or disables the TXF interrupt controller. 0: TXF interrupt disabled 1: TXF interrupt enabled
2	RXM0IE	0	R	W	Mailbox 0 Successful Reception (RXM0) Interrupt Enable Bit The RXM0IE bit enables or disables the RXM0 interrupt controller. 0: RXM0 interrupt disabled 1: RXM0 interrupt enabled
1	RXM1IE	0	R	W	Mailbox 1 to 63 Successful Reception (RXM1) Interrupt Enable Bit The RXM1IE bit enables or disables the RXM1 interrupt controller. 0: RXM1 interrupt disabled 1: RXM1 interrupt enabled
0	TXMIE	0	R	W	Mailbox 32 to 63 Successful Transmission (TXM) Interrupt Enable Bit The TXMIE bit enables or disables the TXM interrupt controller. 0: TXM interrupt disabled 1: TXM interrupt enabled

26.3.28 CANi Mailbox Search Mask Register (CiMBSMR) (i = 0 to 3)

Write to the CiMBSMR register in CAN halt mode only.

CAN0 Mailbox Search Mask Register (C0MBSMR)	<P4 address: location H'FFFF 6863>
CAN1 Mailbox Search Mask Register (C1MBSMR)	<P4 address: location H'FFFF 7863>
CAN2 Mailbox Search Mask Register (C2MBSMR)	<P4 address: location H'FFFF 8863>
CAN3 Mailbox Search Mask Register (C3MBSMR)	<P4 address: location H'FFFF 9863>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MB0SM
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 6	—	All 0	0	0	No Register Bits Should be written with "0" and read as "0".
0	MB0SM	0	R	W	Mailbox 0 Search Mask Bit*[1] When the MB0SM bit is set to "1", message box 0 is excluded from the search target for the CANi mailbox search status register.

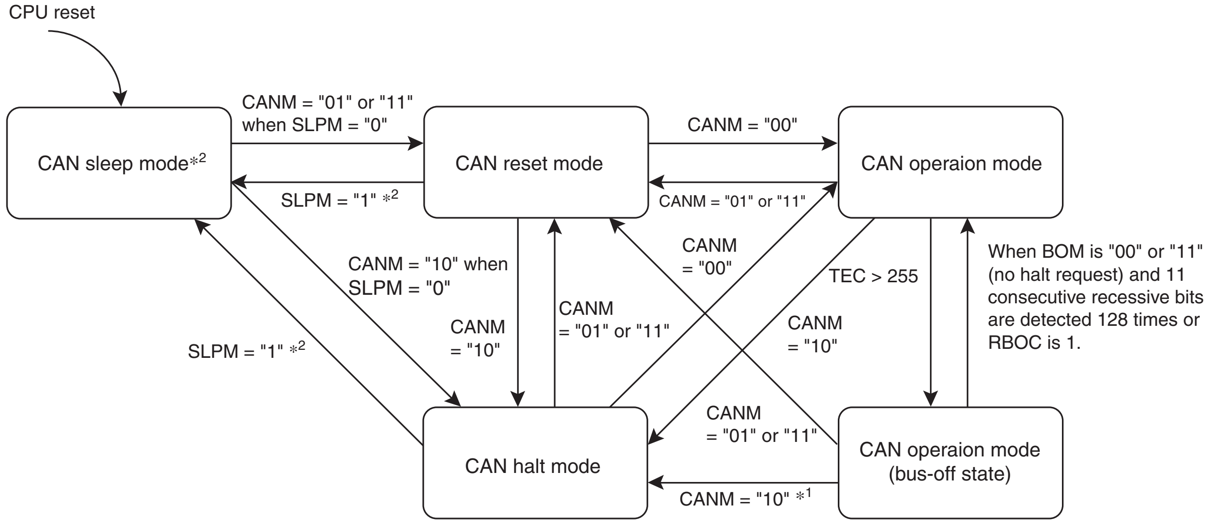
Note: *1 The MB0SM bit is enabled in the search modes except channel search mode. In the RXM1 interrupt handling, this bit is usable to exclude message box 0 from the search target for the CiMSSR register in receive mailbox search mode.

26.4 Operating Mode

The CAN module has the following four operating modes.

- CAN reset mode
- CAN halt mode
- CAN operation mode
- CAN sleep mode

Figure 26.9 shows the transition between CAN operating modes.



CANM, SLPM, BOM, RBOC: Bits in the CiCTLR register

Notes: *1 The transition timing from the bus-off state to CAN halt mode depends on the setting of the BOM bit.
— When the BOM bit is "01", the state transition timing is immediately after entering the bus-off state.
— When the BOM bit is "10", the state transition timing is at the end of the bus-off state.
— When the BOM bit is "11", the state transition timing is at the setting of the CANM bit to "10" (CAN halt mode).
*2 Write to the SLPM bit in CAN reset mode or CAN halt mode. When rewriting the SLPM bit, set only this bit to "0" or "1".

Figure 26.9 Transition between CAN Operating Modes (i = 0 to 3)

26.4.1 CAN Reset Mode

CAN reset mode is provided for CAN communication configuration.

When the CANM bit in the CiCTLR register is set to "01" or "11", the CAN module enters CAN reset mode. Then the RSTST bit in the CiSTR register is set to "1". Do not change the CANM bit until the RSTST bit is set to "1". Configure the CiBCR register before exiting CAN reset mode to any other modes.

The following registers are initialized to their reset values after entering CAN reset mode and their initialized values are retained during CAN reset mode:

- CiMCTLj register
- CiSTR register (except bits SLPST and TFST)
- CiEIFR register
- CiRECR register
- CiTECR register
- CiTSR register
- CiMSSR register
- CiMSMR register
- CiRFCR register
- CiTFCR register
- CiTCR register
- CiECSR register (except EDPM bit)
- CiISR register
- CiMBSMR register

The following registers retain their values after entering CAN reset mode.

- CiCLKR register
- CiCTLR register
- CiSTR register (bits SLPST and TFST)
- Registers CiMIER0 and CiMIER1
- CiEIER register
- CiBCR register
- CiCSSR register
- CiECSR register (EDPM bit only)
- CiMBj register
- Registers CiMKR0 to CiMKR9
- Registers CiFIDCR0 and CiFIDCR1
- Registers CiMKIVLR0 and CiMKIVLR1
- CiAFSR register
- CiRFPCR register
- CiTFPCR register
- CiIER register

26.4.2 CAN Halt Mode

CAN halt mode is used for mailbox configuration and test mode setting.

When the CANM bit in the CiCTLR register is set to "10", CAN halt mode is selected. Then the HLTST bit in the CiSTR register is set to "1". Do not change the CANM bit until the HLTST bit is set to "1".

Refer to table 26.9, Operation in CAN Reset Mode and CAN Halt Mode regarding the state transition conditions when transmitting or receiving.

All registers except bits RSTST, HLTST, and SLPST in the CiSTR register remain unchanged when the CAN module enters CAN halt mode.

Do not change registers CiCLKR, CiCTLR (except bits CANM and SLPM), and CiEIER in CAN halt mode. The CiBCR register can be changed in CAN halt mode only when listen-only mode is selected to use for automatic bit rate detection.

Table 26.9 Operation in CAN Reset Mode and CAN Halt Mode

Mode	Receiver	Transmitter	Bus-Off
CAN reset mode (forcible transition) CANM = "11"	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode without waiting for the end of message transmission.	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN reset mode CANM = "01"	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode after waiting for the end of message transmission*[1]*[4].	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN halt mode	CAN module enters CAN halt mode after waiting for the end of message reception.*[2]*[3].	CAN module enters CAN halt mode after waiting for the end of message transmission.*[1]*[4].	[When the BOM bit is "00"] A halt request from a program will be acknowledged only after bus-off recovery. [When the BOM bit is "01"] CAN module enters automatically to CAN halt mode without waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bit is "10"] CAN module enters automatically to CAN halt mode after waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bit is "11"] CAN module enters CAN halt mode (without waiting for the end of bus-off recovery) if a halt is requested by a program during bus-off.

Legend: BOM bit: Bit in CiCTLR register (i = 0 to 3)

Notes: *1 If several messages are requested to be transmitted, mode transition occurs after the completion of the first transmission. In a case that the CAN reset mode is being requested during suspend transmission, mode transition occurs when the bus is idle, the next transmission ends, or the CAN module becomes a receiver.

*2 If the CAN bus is locked at the dominant level, the program can detect this state by monitoring the BLIF bit in the CiEIFR register.

*3 If a CAN bus error occurs during reception after CAN halt mode is requested, the CAN mode transits to CAN halt mode.

*4 If a CAN bus error or arbitration lost occurs during transmission after CAN reset mode or CAN halt mode is requested, the CAN mode transits to the requested CAN mode.

26.4.3 CAN Sleep Mode

CAN sleep mode is used for reducing current consumption by stopping the clock supply to the CAN module. After MCU hardware reset or software reset, the CAN module starts from CAN sleep mode.

When the SLPM bit in the CiCTLR register is set to "1", the CAN module enters CAN sleep mode. Then the SLPST bit in the CiSTR register is set to "1". Do not change the value of the SLPM bit until the SLPST bit is set to "1". The other registers remain unchanged when the MCU enters CAN sleep mode.

Write to the SLPM bit in CAN reset mode and CAN halt mode. Do not change any registers (except the SLPM bit) during CAN sleep mode. Read operation is still allowed.
When the SLPM bit is set to "0", the CAN module is released from CAN sleep mode. When the CAN module exits CAN sleep mode, the other registers remain unchanged.

26.4.4 CAN Operation Mode (Excluding Bus-Off State)

CAN operation mode is used for CAN communication.

When the CANM bit in the CiCTLR register is set to "00", the CAN module enters CAN operation mode.

Then bits RSTST and HLTST in the CiSTR register are set to "0". Do not change the value of the CANM bit until these bits are set to "0".

If 11 consecutive recessive bits are detected after entering CAN operation mode, the CAN module is in the following states:

- The CAN module becomes an active node on the network that enables transmission and reception of CAN messages.
- Error monitoring of the CAN bus, such as receive and transmit error counters, is performed.

During CAN operation mode, the CAN module may be in one of the following three sub-modes, depending on the status of the CAN bus:

- Idle mode: Transmission or reception is not being performed.
- Receive mode: A CAN message sent by another node is being received.
- Transmit mode: A CAN message is being transmitted. The CAN module may receive its own message simultaneously when self-test mode 0 (TSTM bit in the CiSTR register = "10") or self-test mode 1 (TSTM bit = "11") is selected.

Figure 26.10 shows the sub mode in CAN operation mode.

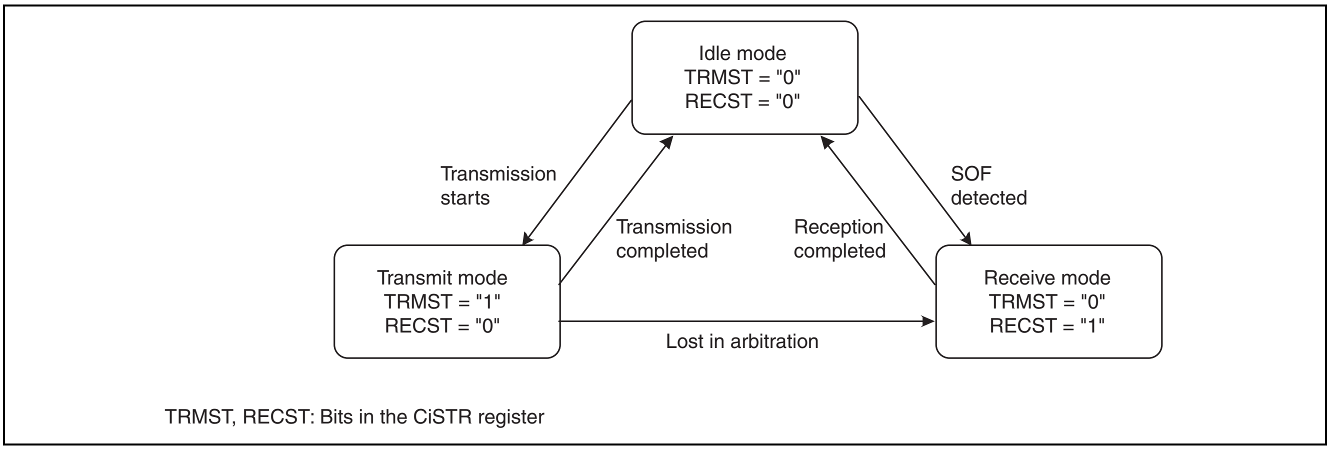


Figure 26.10 Sub Mode in CAN Operation Mode (i = 0 to 3)

26.4.5 CAN Operation Mode (Bus-Off State)

The CAN module enters the bus-off state according to the increment/decrement rules for the transmit/error counters in the CAN Specifications.

The following cases apply when recovering from the bus-off state. When the CAN module is in bus-off state, the values of the associated registers, except registers CiSTR, CiEIFR, CiRECR, CiTECR and CiTSR, remain unchanged.

1. When the BOM bit in the CiCTLR register is "00" (normal mode)

   The CAN module enters the error-active state after it has completed the recovery from the bus-off state and CAN communication is enabled instantly. The BORIF bit in the CiEIFR register is set to "1" (bus-off recovery detected) at this time.
2. When the RBOC bit in the CiCTLR register is set to "1" (forcible return from bus-off)

   The CAN module enters the error-active state when it is in bus-off state and the RBOC bit is set to "1". CAN communication is enabled again after 11 consecutive recessive bits are detected. The BORIF bit is not set to "1" at this time.
3. When the BOM bit is "01" (entry to CAN halt mode automatically at bus-off entry)

   The CAN module enters CAN halt mode when it reaches the bus-off state. The BORIF bit is not set to "1" at this time.
4. When the BOM bit is "10" (entry to CAN halt mode automatically at bus-off end)

   The CAN module enters CAN halt mode when it has completed the recovery from bus-off. The BORIF bit is set to "1" at this time.
5. When the BOM bit is "11" (entry to CAN halt mode by a program) and the CANM bit in the CiCTLR register is set to "10" (CAN halt mode) during the bus-off state

   The CAN module enters CAN halt mode when it is in bus-off state and the CANM bit is set to "10" (CAN halt mode). The BORIF bit is not set to "1" at this time.

   If the CANM bit is not set to "10" during bus-off, the same behavior as (1) applies.

26.5 CAN Communication Speed Configuration

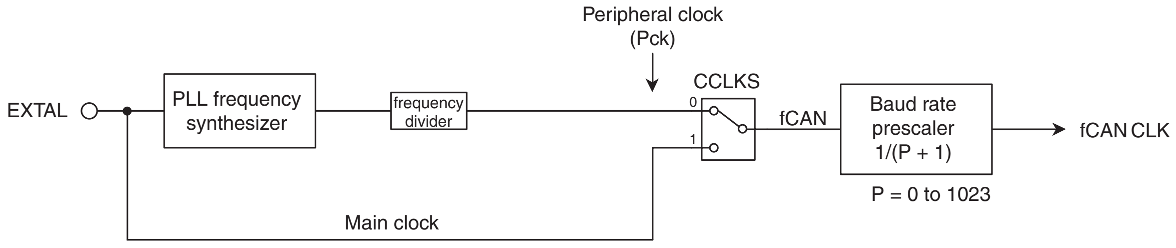
The following description explains about the CAN communication speed configuration.

26.5.1 CAN Clock Configuration

This MCU has a CAN clock selector.

The CAN clock can be configured by setting the CCLKS bit in the CiCLKR register and the BRP bit in the CiBCR register.

Figure 26.11 shows the block diagram of CAN clock generator.



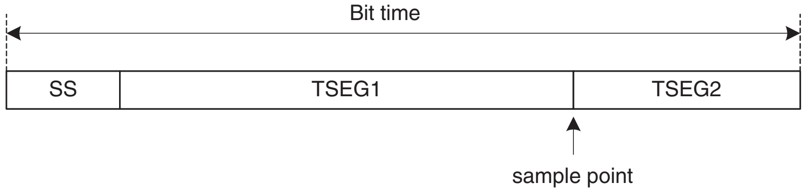
CCLKS: Bit in the CiCLKR register
fCAN: CAN clock source
P: Value selected using BRP bit in the CiBCR register
fCANCLK: CAN communication clock, fCANCLK = fCAN/(P + 1)

Figure 26.11 Block Diagram of CAN Clock Generator (i = 0 to 3)

26.5.2 Bit Timing Configuration

The bit time consists of the following three segments.

Figure 26.12 shows the bit timing.



The range of each segment: Bit time = 8 Tq to 25 Tq
SS = 1 Tq
TSEG1 = 4 Tq to 16 Tq
TSEG2 = 2 Tq to 8 Tq
SJW = 1 Tq to 4 Tq
Setting of TSEG1 and TSEG2: TSEG1 > TSEG2 ≥ SJW

Figure 26.12 Bit Timing

26.5.3 Bit-rate

The bit rate depends on the division value of the fCAN (CAN clock source), the division value of the baud rate prescaler, and the number of Tq of one bit time.

$$\text{Bit rate [bps]} = \frac{fCAN}{\text{Baud rate prescaler division value}^{*1} \times \text{number of Tq of one bit}} = \frac{fCANCLK}{\text{Number of Tq of one bit time}}$$

Note: *1 Division value of the baud rate prescaler = P + 1 (P: 0 to 1023)
P: Setting value of the BRP bit in the CiBCR register (i = 0 to 3)

Table 26.10 lists bit rate examples.

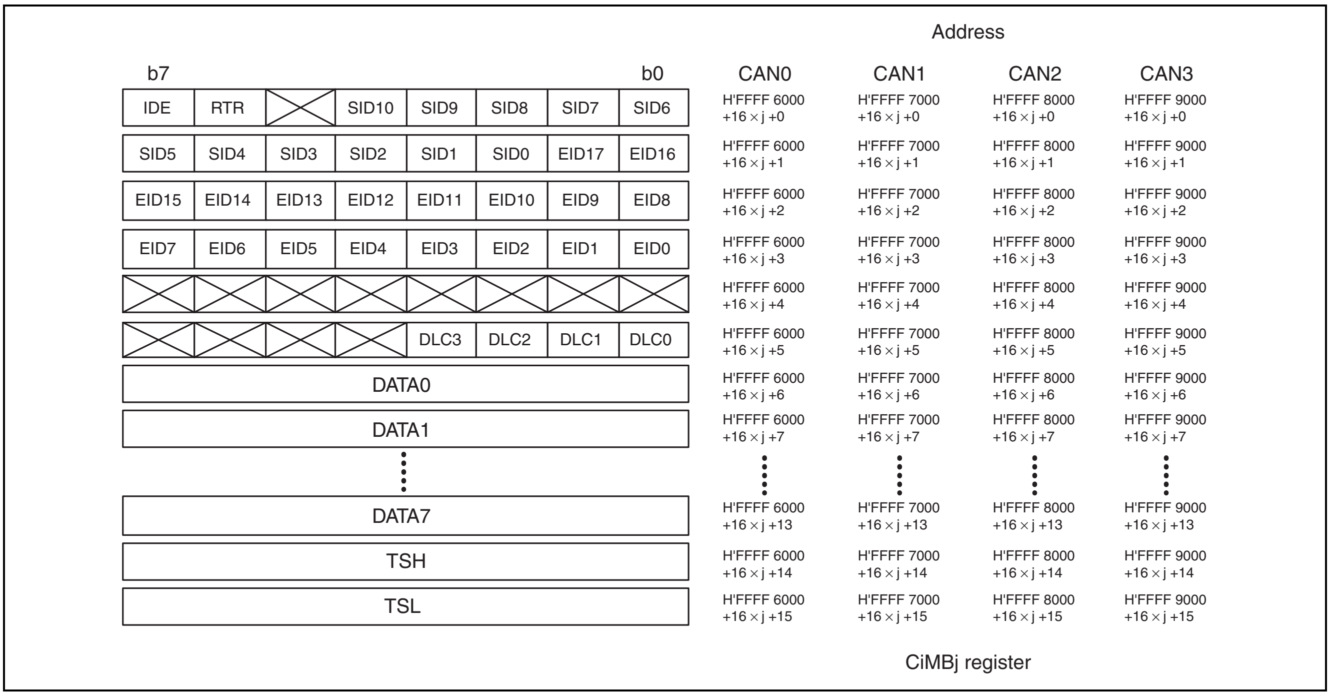
Table 26.10 Example of Bit-rate

fCAN	40MHz		32MHz		20MHz		16MHz	
Bit-rate	**No. of Tq**	**P + 1**	**No. of Tq**	**P + 1**	**No. of Tq**	**P + 1**	**No. of Tq**	**P + 1**
1 Mbps	10Tq	4	8Tq	4	10Tq	2	8Tq	2
	20Tq	2	16Tq	2	20Tq	1	16Tq	1
500 kbps	10Tq	8	8Tq	8	10Tq	4	8Tq	4
	20Tq	4	16Tq	4	20Tq	2	16Tq	2
250 kbps	10Tq	16	8Tq	16	10Tq	8	8Tq	8
	20Tq	8	16Tq	8	20Tq	4	16Tq	4
83.3 kbps	8Tq	60	8Tq	48	8Tq	30	8Tq	24
	10Tq	48	16Tq	24	10Tq	24	16Tq	12
	16Tq	30			16Tq	15		
	20Tq	24			20Tq	12		
33.3 kbps	8Tq	150	8Tq	120	8Tq	75	8Tq	60
	10Tq	120	10Tq	96	10Tq	60	10Tq	48
	20Tq	60	16Tq	60	20Tq	30	16Tq	30
			20Tq	48			20Tq	24

26.6 Mailbox and Mask Register Structure

Figure 26.13 shows the structure of the CiMBj register.

There are 64 mailboxes with the same structure.



Address

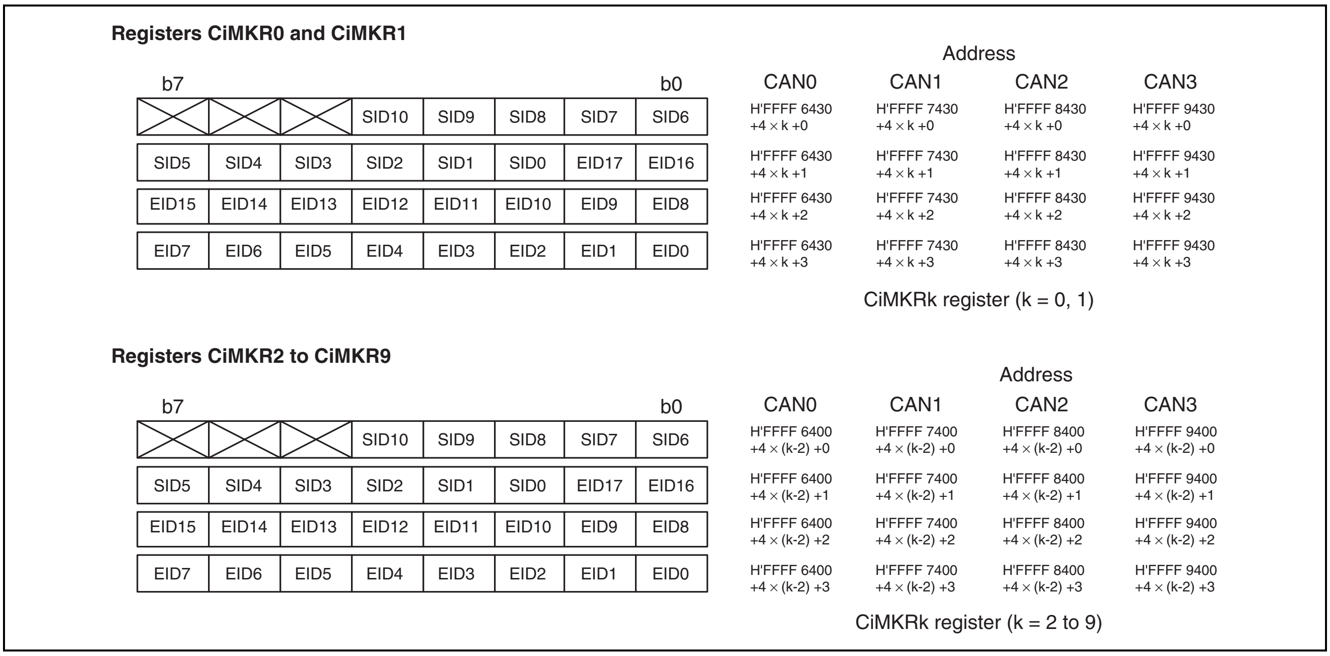
b7							b0	CAN0	CAN1	CAN2	CAN3
IDE	RTR	☒	SID10	SID9	SID8	SID7	SID6	H'FFFF 6000 +16 × j +0	H'FFFF 7000 +16 × j +0	H'FFFF 8000 +16 × j +0	H'FFFF 9000 +16 × j +0
SID5	SID4	SID3	SID2	SID1	SID0	EID17	EID16	H'FFFF 6000 +16 × j +1	H'FFFF 7000 +16 × j +1	H'FFFF 8000 +16 × j +1	H'FFFF 9000 +16 × j +1
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	H'FFFF 6000 +16 × j +2	H'FFFF 7000 +16 × j +2	H'FFFF 8000 +16 × j +2	H'FFFF 9000 +16 × j +2
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	H'FFFF 6000 +16 × j +3	H'FFFF 7000 +16 × j +3	H'FFFF 8000 +16 × j +3	H'FFFF 9000 +16 × j +3
☒	☒	☒	☒	☒	☒	☒	☒	H'FFFF 6000 +16 × j +4	H'FFFF 7000 +16 × j +4	H'FFFF 8000 +16 × j +4	H'FFFF 9000 +16 × j +4
☒	☒	☒	☒	DLC3	DLC2	DLC1	DLC0	H'FFFF 6000 +16 × j +5	H'FFFF 7000 +16 × j +5	H'FFFF 8000 +16 × j +5	H'FFFF 9000 +16 × j +5
DATA0								H'FFFF 6000 +16 × j +6	H'FFFF 7000 +16 × j +6	H'FFFF 8000 +16 × j +6	H'FFFF 9000 +16 × j +6
DATA1								H'FFFF 6000 +16 × j +7	H'FFFF 7000 +16 × j +7	H'FFFF 8000 +16 × j +7	H'FFFF 9000 +16 × j +7
⋮								⋮	⋮	⋮	⋮
DATA7								H'FFFF 6000 +16 × j +13	H'FFFF 7000 +16 × j +13	H'FFFF 8000 +16 × j +13	H'FFFF 9000 +16 × j +13
TSH								H'FFFF 6000 +16 × j +14	H'FFFF 7000 +16 × j +14	H'FFFF 8000 +16 × j +14	H'FFFF 9000 +16 × j +14
TSL								H'FFFF 6000 +16 × j +15	H'FFFF 7000 +16 × j +15	H'FFFF 8000 +16 × j +15	H'FFFF 9000 +16 × j +15

CiMBj register

Figure 26.13 Structure of CiMBj Register (i = 0 to 3; j = 0 to 63)

Figure 26.14 shows the structure of registers CiMKR0, CiMKR1, and CiMKR2 to CiMKR9.

There are 10 mask registers with the same structure.



Registers CiMKR0 and CiMKR1

Address

b7							b0	CAN0	CAN1	CAN2	CAN3
☒	☒	☒	SID10	SID9	SID8	SID7	SID6	H'FFFF 6430 +4 × k +0	H'FFFF 7430 +4 × k +0	H'FFFF 8430 +4 × k +0	H'FFFF 9430 +4 × k +0
SID5	SID4	SID3	SID2	SID1	SID0	EID17	EID16	H'FFFF 6430 +4 × k +1	H'FFFF 7430 +4 × k +1	H'FFFF 8430 +4 × k +1	H'FFFF 9430 +4 × k +1
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	H'FFFF 6430 +4 × k +2	H'FFFF 7430 +4 × k +2	H'FFFF 8430 +4 × k +2	H'FFFF 9430 +4 × k +2
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	H'FFFF 6430 +4 × k +3	H'FFFF 7430 +4 × k +3	H'FFFF 8430 +4 × k +3	H'FFFF 9430 +4 × k +3

CiMKRk register (k = 0, 1)

Registers CiMKR2 to CiMKR9

Address

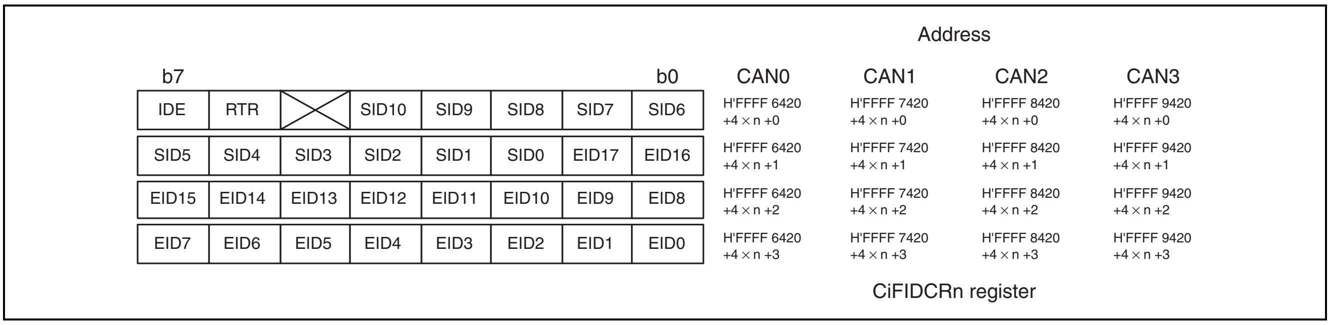
b7							b0	CAN0	CAN1	CAN2	CAN3
☒	☒	☒	SID10	SID9	SID8	SID7	SID6	H'FFFF 6400 +4 × (k-2) +0	H'FFFF 7400 +4 × (k-2) +0	H'FFFF 8400 +4 × (k-2) +0	H'FFFF 9400 +4 × (k-2) +0
SID5	SID4	SID3	SID2	SID1	SID0	EID17	EID16	H'FFFF 6400 +4 × (k-2) +1	H'FFFF 7400 +4 × (k-2) +1	H'FFFF 8400 +4 × (k-2) +1	H'FFFF 9400 +4 × (k-2) +1
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	H'FFFF 6400 +4 × (k-2) +2	H'FFFF 7400 +4 × (k-2) +2	H'FFFF 8400 +4 × (k-2) +2	H'FFFF 9400 +4 × (k-2) +2
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	H'FFFF 6400 +4 × (k-2) +3	H'FFFF 7400 +4 × (k-2) +3	H'FFFF 8400 +4 × (k-2) +3	H'FFFF 9400 +4 × (k-2) +3

CiMKRk register (k = 2 to 9)

Figure 26.14 Structure of CiMKRk Register (i = 0 to 3; k = 0 to 9)

Figure 26.15 shows the structure of the CiFIDCRn register.

There are 2 FIFO received ID compare registers with the same structure.



b7							b0	CAN0	CAN1	CAN2	CAN3
IDE	RTR	☒	SID10	SID9	SID8	SID7	SID6	H'FFFF 6420 +4 × n +0	H'FFFF 7420 +4 × n +0	H'FFFF 8420 +4 × n +0	H'FFFF 9420 +4 × n +0
SID5	SID4	SID3	SID2	SID1	SID0	EID17	EID16	H'FFFF 6420 +4 × n +1	H'FFFF 7420 +4 × n +1	H'FFFF 8420 +4 × n +1	H'FFFF 9420 +4 × n +1
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	H'FFFF 6420 +4 × n +2	H'FFFF 7420 +4 × n +2	H'FFFF 8420 +4 × n +2	H'FFFF 9420 +4 × n +2
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	H'FFFF 6420 +4 × n +3	H'FFFF 7420 +4 × n +3	H'FFFF 8420 +4 × n +3	H'FFFF 9420 +4 × n +3

Figure 26.15 Structure of CiFIDCRn register (i = 0 to 3; n = 0, 1)

26.7 Acceptance Filtering and Masking Function

Acceptance filtering allows the user to receive messages with a specified range of multiple IDs for mailboxes. Registers CiMKR0 to CiMKR9 can perform masking of the standard ID and the extended ID of 29 bits.

- The CiMKR0 register corresponds to mailboxes [0] to [15].
- The CiMKR1 register corresponds to mailboxes [16] to [31].
- The CiMKR2 register corresponds to mailboxes [32] to [35].
- The CiMKR3 register corresponds to mailboxes [36] to [39].
- The CiMKR4 register corresponds to mailboxes [40] to [43].
- The CiMKR5 register corresponds to mailboxes [44] to [47].
- The CiMKR6 register corresponds to mailboxes [48] to [51].
- The CiMKR7 register corresponds to mailboxes [52] to [55].
- The CiMKR8 register corresponds to mailboxes [56] to [59] in normal mailbox mode and the receive FIFO mailboxes [60] to [63] in FIFO mailbox mode.
- The CiMKR9 register corresponds to mailboxes [60] to [63] in normal mailbox mode and the receive FIFO mailboxes [60] to [63] in FIFO mailbox mode.

Registers CiMKIVLR0 and CiMKIVLR1 disable acceptance filtering individually for each mailbox.

The IDE bit in the CiMBj register is enabled when the IDFM bit in the CiCTLR register is "10" (mixed ID mode).

The RTR bit in the CiMBj register selects a data frame or a remote frame.

In FIFO mailbox mode, normal mailboxes (mailboxes [0] to [55]) use the single corresponding register among registers CiMKR0 to CiMKR7 for acceptance filtering. Receive FIFO mailboxes (mailboxes [60] to [63]) use two registers CiMKR8 and CiMKR9 for the acceptance filtering.
Also, the receive FIFO uses two registers CiFIDCR0 and CiFIDCR1 for ID comparison. Bits EID, SID, RTR, and IDE in registers CiMB60 to CiMB63 for the receive FIFO are disabled. As acceptance filtering depends on the result of two ID-mask sets, two ranges of IDs can be received into the receive FIFO.
Registers CiMKIVLR0 and CiMKIVLR1 are disabled for the receive FIFO.
If both setting of standard ID and extended ID are set in the IDE bits in registers CiFIDCR0 and CiFIDCR1 individually, both ID formats are received.
If both setting of data frame and remote frame are set in the RTR bits in registers CiFIDCR0 and CiFIDCR1 individually, both data and remote frames are received.
When combination with two ranges of IDs is not necessary, set the same mask value and the same ID into both of the FIFO ID/mask register sets.

Figure 26.16 shows the correspondence of mask registers to mailboxes. Figure 26.17 shows the acceptance filtering.

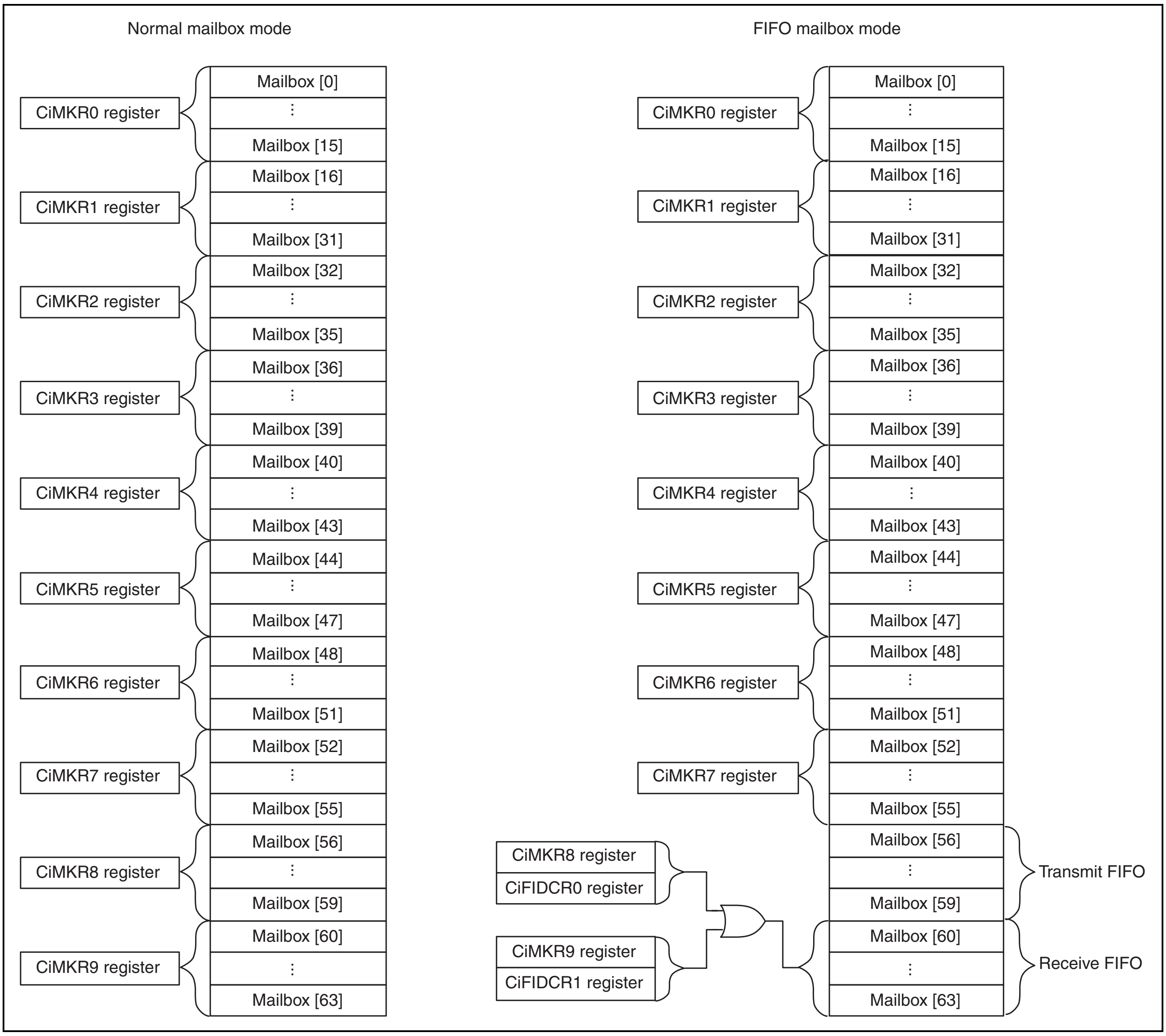
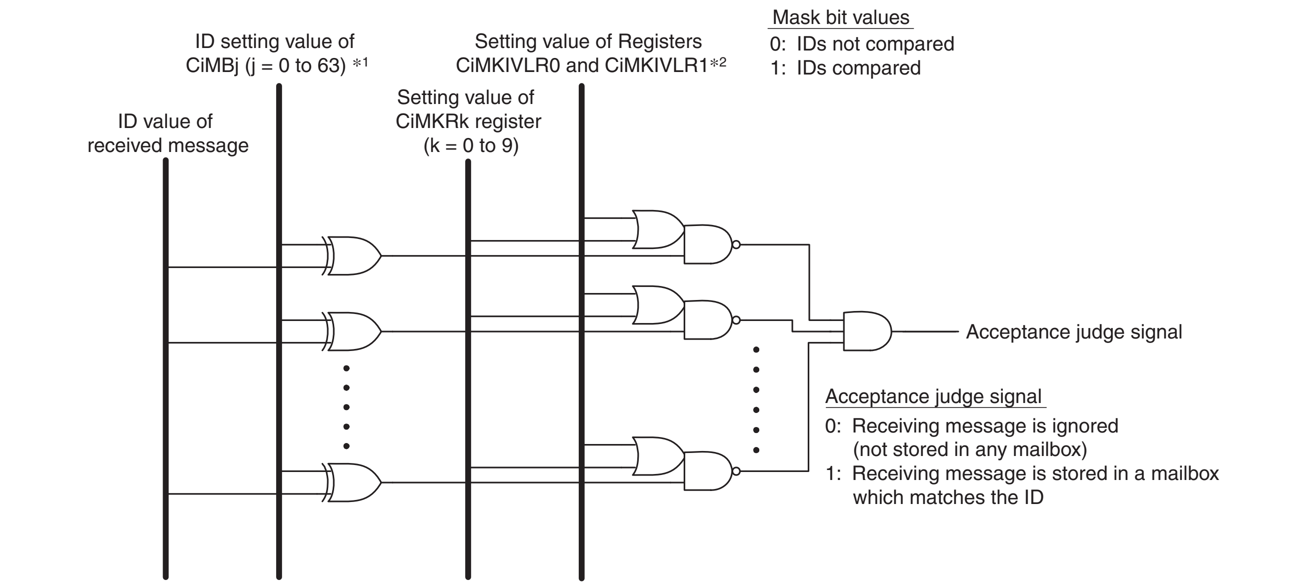


Figure 26.16   Correspondence of Mask Registers to Mailboxes (i = 0 to 3)


Mask bit values
0: IDs not compared
1: IDs compared
ID setting value of CiMBj (j = 0 to 63) *1
Setting value of Registers CiMKIVLR0 and CiMKIVLR1*2
Setting value of CiMKRk register (k = 0 to 9)
ID value of received message
Acceptance judge signal
Acceptance judge signal
0: Receiving message is ignored (not stored in any mailbox)
1: Receiving message is stored in a mailbox which matches the ID

Notes: *1 The values set in registers CiFIDCR0 and CiFIDCR1 are used in FIFO mailbox mode.
*2 Invalid in FIFO mailboxes.

Figure 26.17 Acceptance Filtering (i = 0 to 3)

26.8 Reception and Transmission

Table 26.11 list the CAN communication mode configuration.

Table 26.11 Configuration for CAN Reception Mode and Transmission Mode

TRMREQ	RECREQ	ONESHOT	Communication Mode of Mailbox
0	0	0	Mailbox disabled or transmission being aborted
0	0	1	Configurable only when transmission or reception from a mailbox (programmed in one-shot mode) is aborted.
0	1	0	Configured as a receive mailbox for a data frame or a remote frame.
0	1	1	Configured as a one-shot receive mailbox for a data frame or a remote frame.
1	0	0	Configured as a transmit mailbox for a data frame or a remote frame.
1	0	1	Configured as a one-shot transmit mailbox for a data frame or a remote frame.
1	1	0	Do not set.
1	1	1	Do not set.

Note: TRMREQ, RECREQ, ONESHOT: Bits in CiMCTLj register (i = 0 to 3; j = 32 to 63)

When a mailbox is configured as a receive mailbox or a one-shot receive mailbox, note the following:

1. Before a mailbox is configured as a receive mailbox or a one-shot receive mailbox, set the CiMCTLj register to "H'00".
2. A received message is stored into the first mailbox that matches the condition according to the result of receive mode configuration and acceptance filtering. Upon deciding a mailbox which stores the received message, the mailbox with the smaller number has higher priority.
3. In CAN operation mode, when a CAN module transmits a message whose ID matches with the ID/mask set of a mailbox configured to receive messages, the CAN module never receives the transmitted data. In self-test mode, however, the CAN module may receive its transmitted data. In this case, the CAN module sends an ACK.

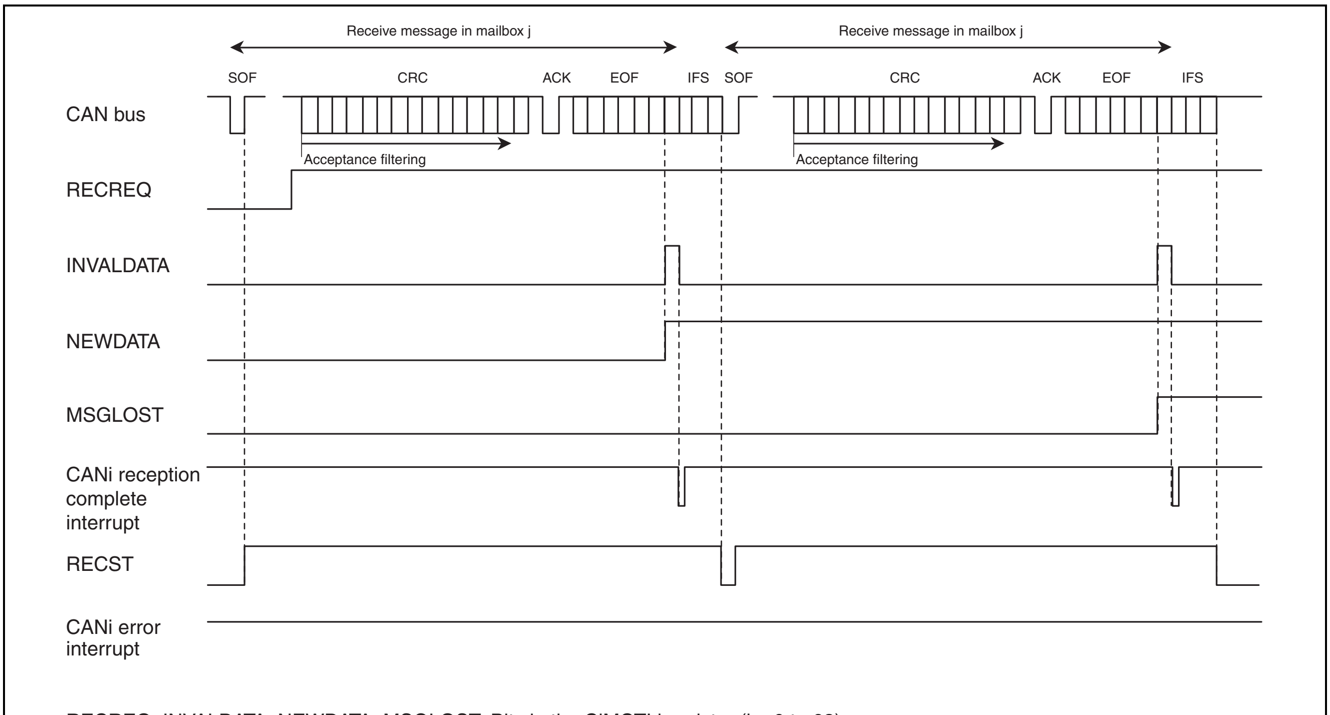
When configuring a mailbox as a transmit mailbox or a one-shot transmit mailbox, note the following:

1. Before a mailbox is configured as a transmit mailbox or one-shot transmit mailbox, ensure that the CiMCTLj register is "H'00" and that there is no pending abort process.

26.8.1 Reception

Figure 26.18 shows an operation example of data frame reception in overwrite mode.

This example shows the operation of overwriting the first message when the CAN module receives two consecutive CAN messages that matches the receiving conditions of the CiMCTLj register.

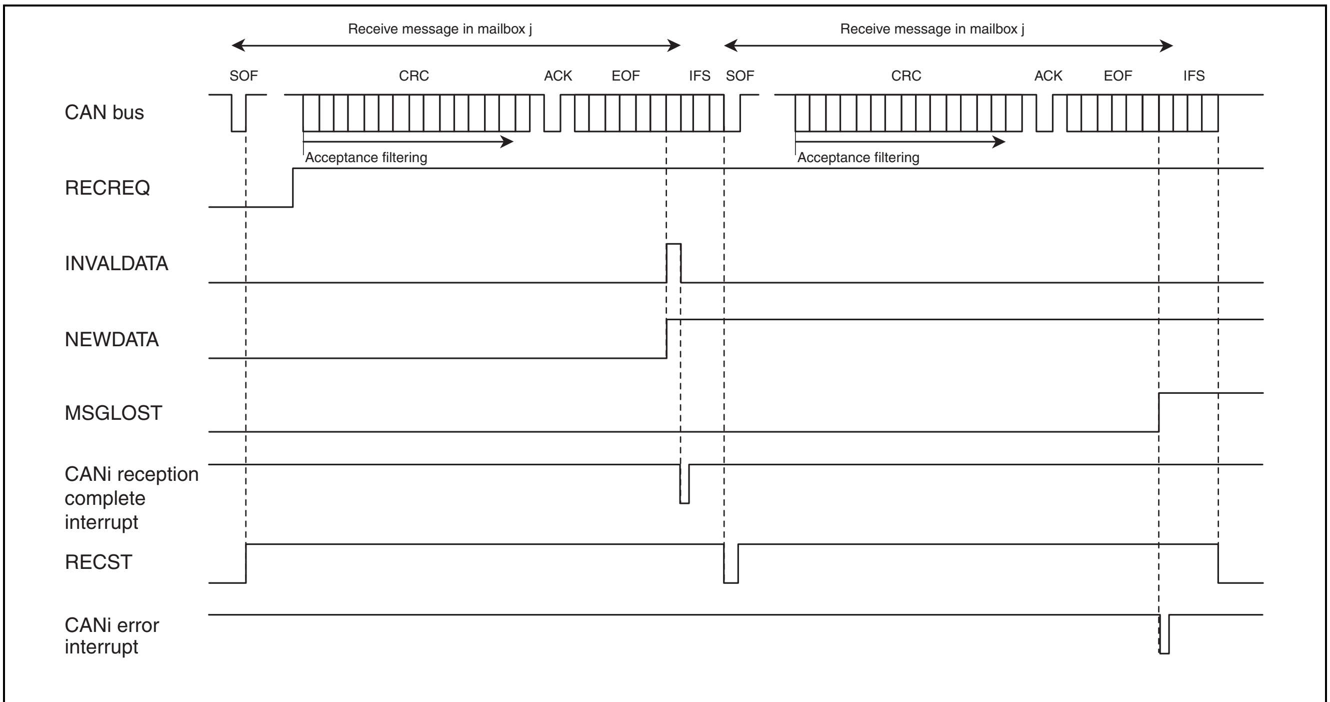


RECREQ, INVALDATA, NEWDATA, MSGLOST: Bits in the CiMCTLj register (j = 0 to 63)
RECST: Bit in the CiSTR register

Figure 26.18 Operation Example of Data Frame Reception in Overwrite Mode (i = 0 to 3)

1. When a SOF is detected on the CAN bus, the RECST bit in the CiSTR register is set to "1" (reception in progress) if the CAN module has no message ready to start transmission.
2. The acceptance filter procedure starts at the beginning of the CRC field to select the receive mailbox.
3. After a message has been received, the NEWDATA bit in the CiMCTLj register for the receive mailbox is set to "1" (new data being updated/stored in the mailbox). The INVALDATA bit in the CiMCTLj register is set to "1" (message is being updated) at the same time, and then the INVALDATA bit is set to "0" (message valid) again after the complete message is transferred to the mailbox.
4. When the interrupt enable bit in the CiMIER register for the receive mailbox is "1" (interrupt enabled), the CANi reception complete interrupt request is generated. This interrupt is generated when the INVALDATA bit is set to "0".
5. After reading the message from the mailbox, the NEWDATA bit needs to be set to "0" by a program.
6. In overwrite mode, if the next CAN message has been received into a mailbox whose NEWDATA bit is still set to "1", the MSGLOST bit in the CiMCTLj register is set to "1" (message has been overwritten). The new received message is transferred to the mailbox. The CANi reception complete interrupt request is generated the same as in 4.

Figure 26.19 shows the operation example of data frame reception in overrun mode. This example shows the operation of overruning the second message when the CAN module receives two consecutive CAN messages that matches the receiving conditions of the CiMCTLj register.



RECREQ, INVALDATA, NEWDATA, MSGLOST: Bits in the CiMCTLj register (j = 0 to 63)
RECST: Bit in the CiSTR register

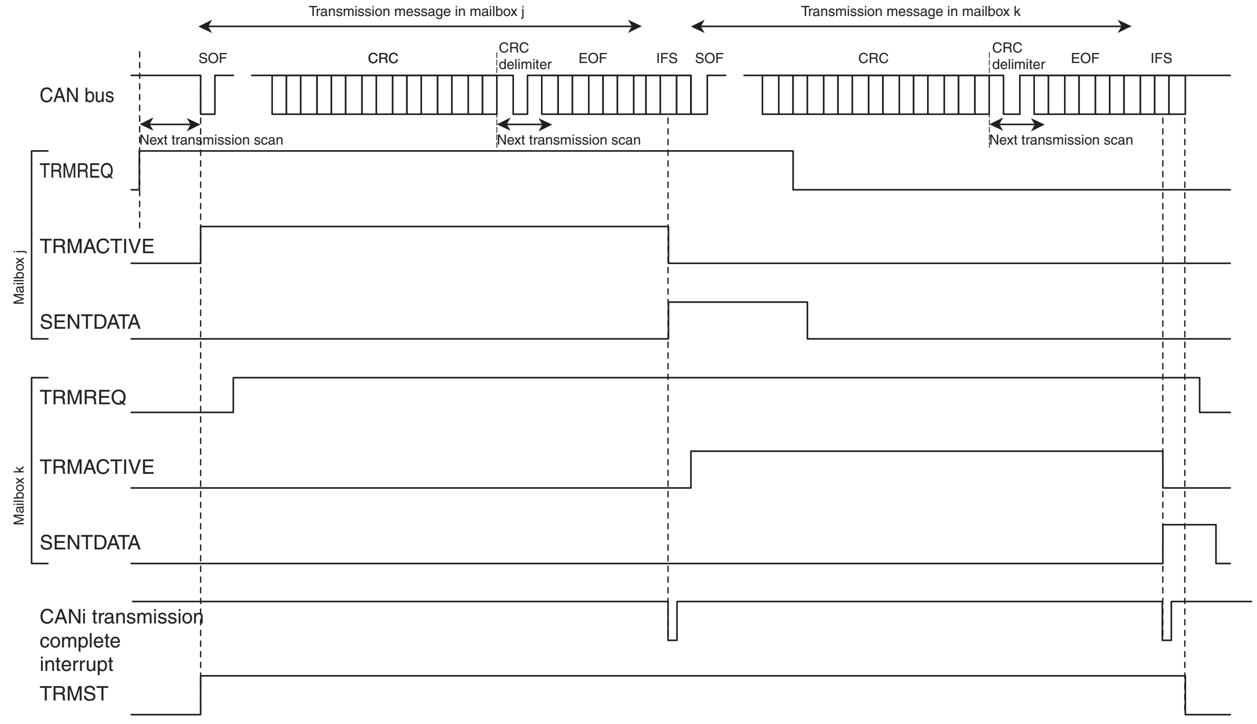
Figure 26.19 Operation Example of Data Frame Reception in Overrun Mode (i = 0 to 3)

1. to 5. are the same as overwrite mode.

6. In overrun mode, if the next message has been received before the NEWDATA bit is set to "0", the MSGLOST bit in the CiMCTLj register is set to "1" (message has been overrun). The new received message is discarded and a CANi error interrupt request is generated if the corresponding interrupt enable bit in the CiEIER register is set to "1" (interrupt enabled).

26.8.2 Transmission

Figure 26.20 shows an operation example of data frame transmission.



TRMREQ, TRMACTIVE, SENTDATA: Bits in the CiMCTLj and CiMCTLk register (j, k = 32 to 63, j ≠ k)
TRMST: Bit in the CiSTR register

Figure 26.20 Operation Example of Data Frame Transmission (i = 0 to 3)

1. When a TRMREQ bit in the CiMCTLj register is set to 1 (transmit mailbox) in bus-idle state, the mailbox scan procedure starts to decide the highest-priority mailbox for transmission. Once the transmit mailbox is decided, the TRMACTIVE bit in the CiMCTLj register is set to "1" (from when a transmission request is received until transmission is completed, or an error/arbitration lost has occurred), the TRMST bit in the CiSTR register is set to "1" (transmission in progress), and the CAN module starts transmission. *[1]
2. If other TRMREQ bits are set, the transmission scan procedure starts with the CRC delimiter for the next transmission.
3. If transmission is completed without losing arbitration, the SENTDATA bit in the CiMCTLj register is set to "1" (transmission completed) and the TRMACTIVE bit is set to "0" (transmission is pending, or no transmission request). If the interrupt enable bit in the CiMIER register is "1" (interrupt enabled), the CANi transmission complete interrupt request is generated.
4. When requesting the next transmission from the same mailbox, set bits SENDTDATA and TRMREQ to "0", then set the TRMREQ bit to "1" after checking that bits SENDTDATA and TRMREQ have been set to "0".

Note: *1 If arbitration is lost after the CAN module starts transmission, the TRMACTIVE bit is set to "0". The transmission scan procedure is performed again to search for the highest-priority transmit mailbox from the beginning of the CRC delimiter. If an error occurs either during transmission or following the arbitration lost the transmission scan procedure is performed again from the start of the error delimiter to search for the highest-priority transmit mailbox.

26.9 CAN Interrupt

The CAN module provides the following CAN interrupts for each channel. Table 26.12 lists CAN interrupts.

- CANi reception complete interrupt (mailbox 0) [RXM0i]
- CANi reception complete interrupt (mailbox 1 to 63) [RXM1i]
- CANi transmission complete interrupt (mailbox 32 to 63) [TXMi]
- CANi reception FIFO interrupt [RXFi]
- CANi transmission FIFO interrupt [TXFi]
- CANi error interrupt [ERSi]

  There are eight types of interrupt sources for the CANi error interrupts. These sources can be determined by checking the CiEIFR register.

  — Bus error
  — Error-warning
  — Error-passive
  — Bus-off entry
  — Bus-off recovery
  — Receive overrun
  — Overload frame transmission
  — Bus lock

Table 26.12 CAN Interrupts

Module	Interrupt Abbreviation	Interrupt Source	Source Flag
CANi	ERSi	Bus lock detected	CiEIFR.BLIF
		Overload frame transmission detected	CiEIFR.OLIF
		Overrun detected	CiEIFR.ORIF
		Bus-off recovery detected	CiEIFR.BORIF
		Bus-off entry detected	CiEIFR.BOEIF
		Error-passive detected	CiEIFR.EPIF
		Error-warning detected	CiEIFR.EWIF
		Bus error detected	CiEIFR.BEIF
	RXFi	Receive FIFO message received (CiMIER1[29] = 0)	CiISR.RXFF
		Receive FIFO warning (CiMIER1[29] = 1)	
	TXFi	Transmit FIFO message transmission completed (CiMIER1[25] = 0)	CiISR.TXFF
		FIFO last message transmission completed (CiMIER1[25] = 1)	
	RXM0i	Mailbox 0 message received	CiMCTL0.NEWDATA
	RXM1i	Mailbox 1 to 63 message received	CiMCTL1.NEWDATA to CiMCTL63.NEWDATA
	TXMi	Mailbox 32 to 63 message transmission completed	CiMCTL32.SENTDATA to CiMCTL63.SENTDATA

Legend: i = 0 to 3

(1) CANi reception complete interrupt (mailbox 0) [RXM0i]

After the CiMCTL0.NEWDATA bit is set by the completion of reception, if received data has been stored (the corresponding INVALDATA bit changes from "1" to "0"), the CiISR.RXM0F bit is set to "1" when CiMIER0[0] has been set to "1". When the mailbox 0 reception complete (RXM0) interrupt has been enabled, the RXM0 interrupt is requested to the interrupt controller.

To clear the RXM0 interrupt, clear the CiMCTL0.NEWDATA bit in the RXM0 interrupt handling routine. To change the CiIER.RXM0IE bit to disabled after having set the bit, make the change while no RXM0 interrupt is generated or during the RXM0 interrupt handling routine. This also applies to CiMIER0[0].

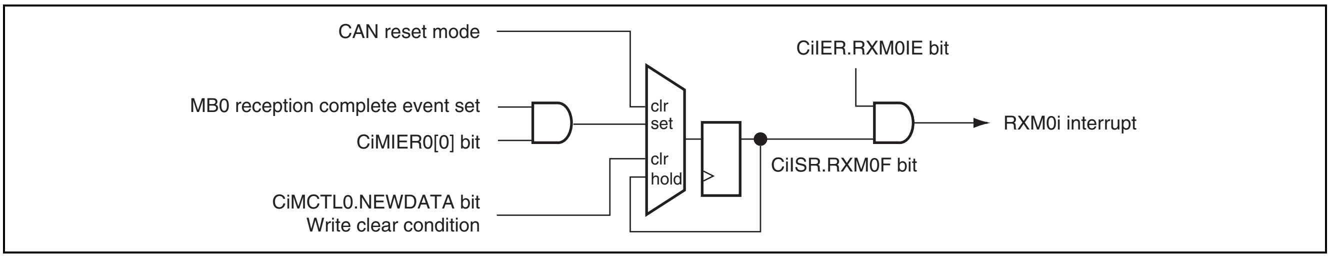


Figure 26.21 Block Diagram of CANi Reception Complete Interrupt (Mailbox 0) [RXM0i]

(2) CANi reception complete interrupt (mailbox 1 to 63) [RXM1i]

After the CiMCTLj.NEWDATA bit is set by the completion of reception, if received data has been stored (the corresponding INVALDATA bit changes from "1" to "0"), the CiISR.RXM1F bit is set to "1" when the CiMIER0 or CiMIER1 register bit corresponding to mailbox j has been set to "1". When the mailbox 1 to 63 reception complete (RXM1) interrupt has been enabled, the RXM1 interrupt is requested to the interrupt controller.

To clear the RXM1 interrupt, clear the CiMCTLj.NEWDATA bit in the RXM1 interrupt handling routine. To change the CiIER.RXM1IE bit to disabled after having set the bit, make the change while no RXM1 interrupt is generated or during the RXM1 interrupt handling routine. This also applies to CiMIER0[j] (j = 1 to 31) or CiMIER1[j-32] (j = 32 to 63).

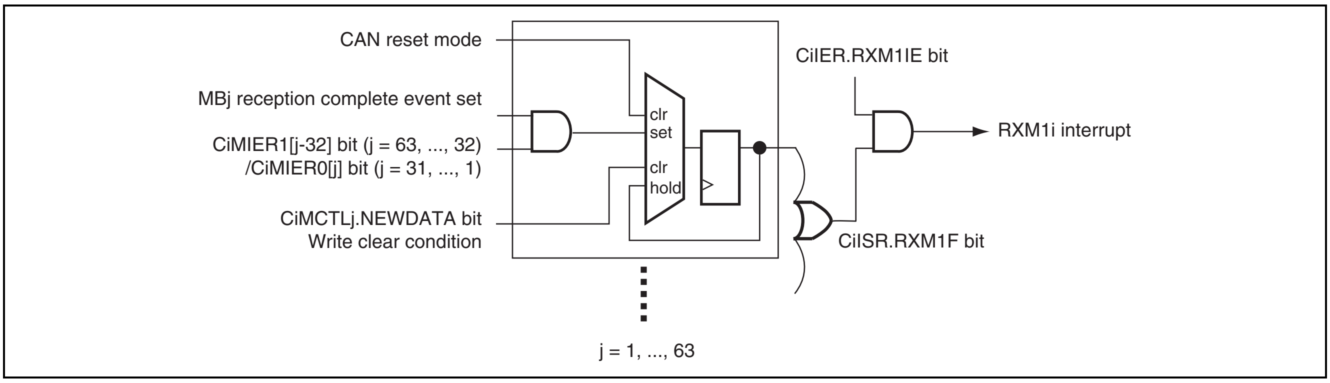


Figure 26.22 Block Diagram of CANi Reception Complete Interrupt (Mailbox 1 to 63) [RXM1i]

(3) CANi transmission complete interrupt (mailbox 32 to 63) [TXMi]

If the CiMCTLj.SENTDATA bit is set by the completion of transmission, the CiISR.TXMF bit is set to "1" when the CiMIER1 register bit corresponding to mailbox j has been set to "1". When the mailbox 32 to 63 transmission complete (TXM) interrupt has been enabled, the TXM interrupt is requested to the interrupt controller.

To clear the TXM interrupt, clear the CiMCTLj.SENTDATA bit in the TXM interrupt handling routine. To change the CiIER.TXMIE bit to disabled after having set the bit, make the change while no TXM interrupt is generated or during the TXM interrupt handling routine. This also applies to CiMIER1[j-32] (j = 32 to 63).

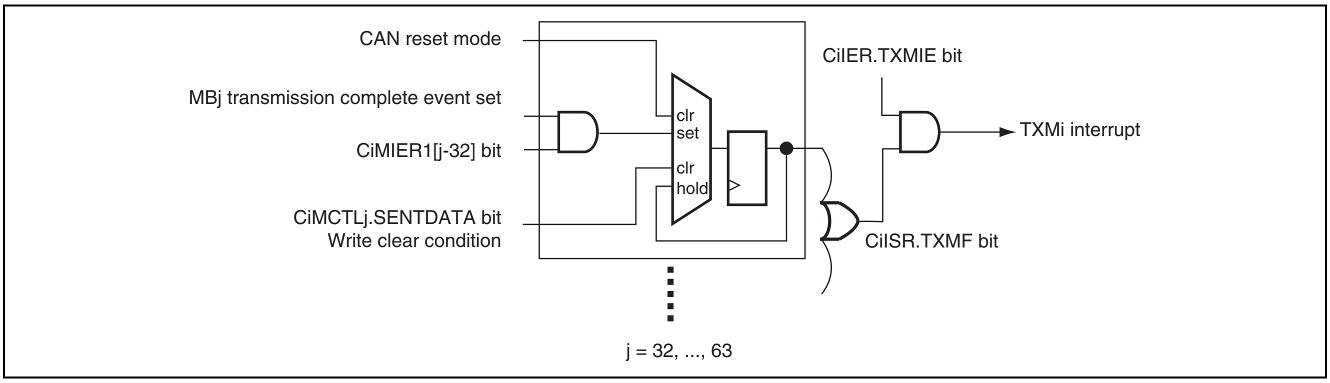


Figure 26.23 Block Diagram of CANi Transmission Complete Interrupt (Mailbox 32 to 63) [TXMi]

(4) CANi receive FIFO interrupt [RXFi]

If CiRFCR[6:5] are set by the reception of a receive FIFO message or by a warning with the settings of CiMIER1[29:28], the CiISR.RXFF bit is set to "1". When the receive FIFO (RXF) interrupt has been enabled with the CiIER.RXFIE bit, the RXF interrupt is requested to the interrupt controller.

To clear the RXF interrupt, clear the CiISR.RXFF bit in the RXF interrupt handling routine. To change the CiIER.RXFIE bit to disabled after having set the bit, make the change while no RXF interrupt is generated or during the RXF interrupt handling routine. This also applies to CiMIER1[28].

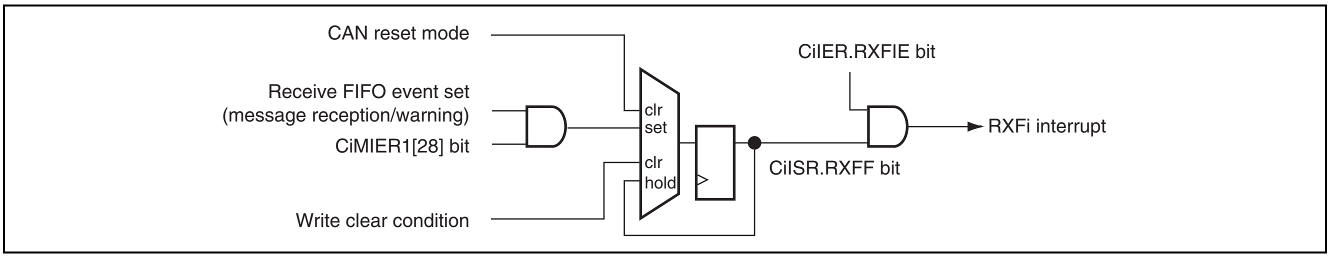


Figure 26.24 Block Diagram of CANi Receive FIFO Interrupt [RXFi]

(5) CANi transmit FIFO interrupt [TXFi]

When the transmission of a transmit FIFO message is counted for the specified number of times with the settings of CiMIER1 [25:24], the CiISR.TXFF bit is set to "1". When the transmit FIFO (TXF) interrupt has been enabled with the CiIER.TXFIE bit, the TXF interrupt is requested to the interrupt controller.

To clear the TXF interrupt, clear the CiISR.TXF bit in the TXF interrupt handling routine. To change the CiIER.TXFIE bit to disabled after having set the bit, make the change while no TXF interrupt is generated or during the TXF interrupt handling routine. This also applies to CiMIER1[24].

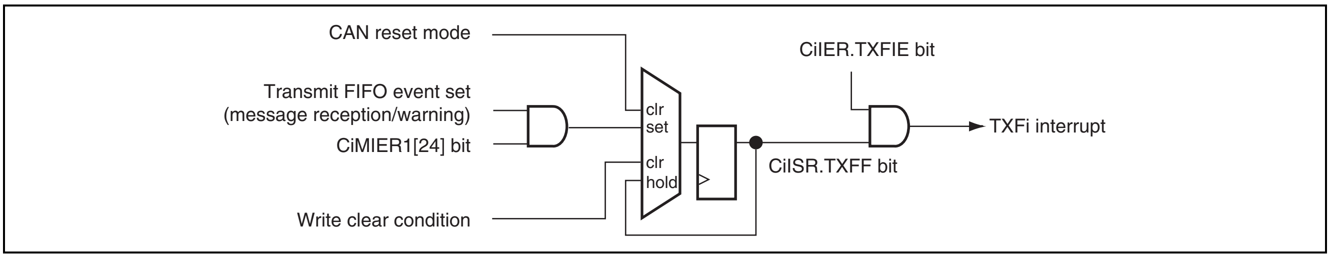


Figure 26.25 Block Diagram of CANi Transmit FIFO Interrrupt [TXFi]

(6) CANi error interrupt [ERSi]

If CiEIFR[j] is set by a communication error, the CiISR.ERSF bit is set to "1" when the corresponding CiEIER[j] has been set to "1". When the error (ERS) interrupt has been enabled with the CiIER.ERSIE bit, the ERS interrupt is requested to the interrupt controller.

To clear the the ERS interrupt, clear each CiEIFR[j] register bit in the ERS interrupt handling routine. To change the CiIER.ERSIE bit to disabled after having set the bit, make the change while no ERS interrupt is generated or during the ERS interrupt handling routine. This also applies to CiEIER[j] (j = 7 to 0).

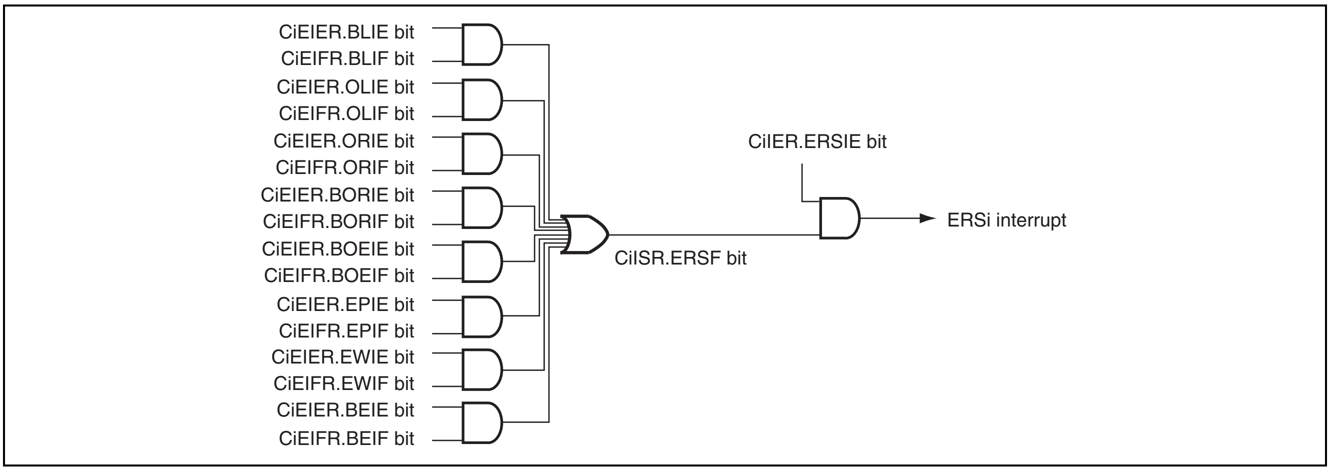


Figure 26.26 Block Diagram of CANi Error Interrrupt [ERSi]

Section 27 A/D Converter (ADC)

27.1 Overview

This MCU includes a 12-bit successive approximation A/D converter, which consists of two independent units (AD0 and AD1). Up to 16 channel analog inputs can be selected by software. Note that in this section the i used in the register names ADi, pin names, and signal names is 0 or 1, that m is 0, 2, 4, 6, 8 to 15, and that n is 0, 1, 4, 5. (For pin specifications, see table 27.2.)

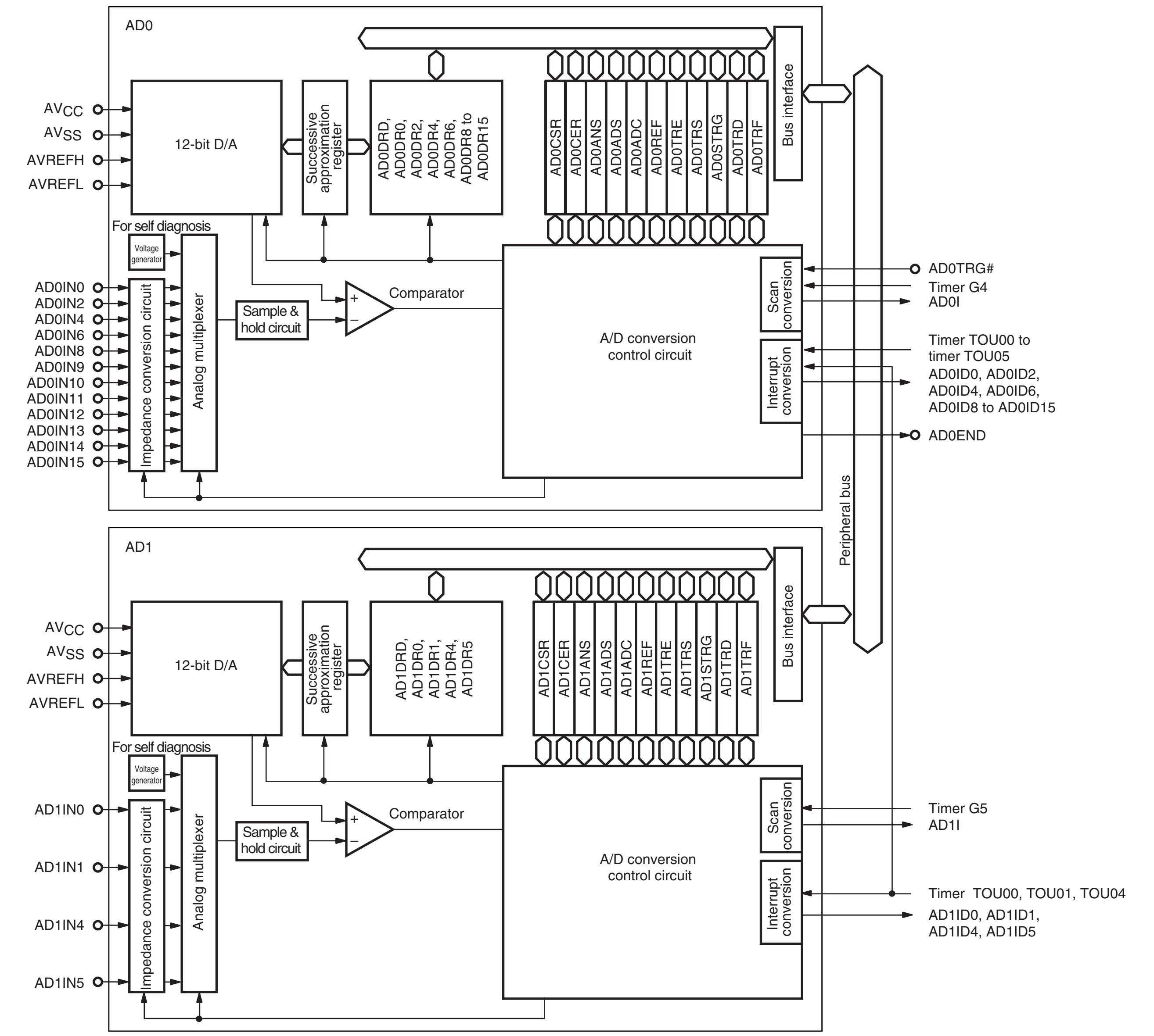
Table 27.1 lists the overview of the ADC.

Table 27.1 Overview of the ADC

Item	Description
Resolution	• 12 bits
Input channels	• 16 channels AD0: 12 channels (AD0INm (m = 0, 2, 4, 6, 8 to 15)), AD1: 4 channels (AD1INn (n = 0, 1, 4, 5))
Minimum conversion time	• 1.25 µs/channel (operating at Pck = 40 MHz, 50 conversion states)
Scan conversion modes	• 2 modes Single cycle scan mode: Scanning performed only once Continuous scan mode: Scanning performed repeatedly The channels subject to scanning are selectable. A/D conversion proceeds in order from lower-numbered to higher-numbered channels (AD0IN0 to AD0IN15 for AD0 and AD1IN0 to AD1IN5 for AD1).
A/D-converted value addition mode	The same channel is A/D converted two to four times in succession, and the sum of the converted values is stored in the A/D data registers. (Only channels AD0IN0, AD0IN2, AD0IN4, AD0IN6, AD1IN0, AD1IN1, AD1IN4, and AD1IN5 are supported.)
Registers	• Sixteen 12-bit A/D data registers
Sample and hold function	Each A/D converter module (AD0 and AD1) includes a sample and hold circuit.
Two ways of starting scan conversion	• AD0: Selectable software trigger (ADST bit in AD0CSR register), external trigger (AD0TRG#), or ATU-IIIS timer trigger (timer G4) • AD1: Selectable software trigger (ADST bit in AD1CSR register) or ATU-IIIS timer trigger (timer G5)

Item	Description
Interrupt-triggered conversion	Independently from scan conversion, it is possible to preferentially process channels requested by an ATU-IIIS timer trigger or software trigger for A/D conversion. This function supports channels AD0INm and AD1INn. When an interrupt conversion and a scan conversion conflict, the scan conversion is suspended and A/D conversion is executed preferentially on the channel for which the interrupt conversion was requested. On completion of the interrupt conversion, the scan conversion is resumed on the channel there A/D conversion was interrupted.
Support for scan conversion end interrupt (ADI), interrupt conversion end interrupt, and DMA transfer function	On completion of scanning for scan conversion, a scan conversion end interrupt request (AD0I, AD1I) can be generated or the DMAC can be started. On completion of interrupt conversion on channels AD0INm and AD1INn, an interrupt conversion end interrupt request (AD0IDm and AD1IDn) can be generated, or the DMAC (AD0ID0, AD0ID2, and AD0ID15) can be started.
Programmable analog input voltage range	The analog conversion voltage range is programmable by using the AVREFH pin.
ADEND output	When the AD0IN0 channel is used for scan conversion, the conversion timing signal is output on the AD0END pin.
A/D converter self-diagnosis function	This function A/D converts the internally generated voltage values of AVREF (AVREFH) × 0, AVREF × 1/2, and AVREF × 1, and returns the A/D converted values and the converted voltage information to the AD0DRD register and the AD1DRD register, respectively. Then the function reads the AD0DRD and AD1DRD registers by software to determine whether or not the A/D converted values are within the normal range, and detects an error in the A/D converter.

Figure 27.1 shows a block diagram of the A/D converter.



Legend:

ADiDR:	A/D data register
ADiDRD:	A/D data register DIAGi
ADiCSR:	A/D control register
ADiREF:	A/D conversion status register
ADiTRE:	A/D interrupt trigger enable register
ADiTRF:	A/D interrupt trigger conversion end flag register
ADiTRS:	A/D interrupt trigger source select register
ADiSTRG:	A/D interrupt software trigger register
ADiTRD:	A/Di Interrupt Trigger Conversion End Interrupt Enable Register
ADiADS:	A/D converted value addition mode select register
ADiADC:	A/D converted value addition count select register
ADiANS:	A/D channel select register
ADiCER:	A/D control extended register

Note: i = 0, 1

Figure 27.1 Block Diagram of A/D Converter

27.2 Input/Output Pins

Table 27.2 shows the pin configuration of the A/D converter.

To maintain reliable operation of this MCU, ensure the appropriate relation between AVcc and AVss to Vcc and Vss. For details, see section 27.8, Usage Notes.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 27.2 Pin Configuration

Pin Name	I/O	Function
AV_{CC}	Input	Analog power supply
AV_{SS}	Input	Analog ground
AVREFL	Input	Input pin for analog reference voltage (AVREFL < AVREFH)
AVREFH	Input	Input pin for analog reference voltage (AVREFL < AVREFH)
AD0IN0	Input	AD0 analog input pin 0
AD0IN2	Input	AD0 analog input pin 2
AD0IN4	Input	AD0 analog input pin 4
AD0IN6	Input	AD0 analog input pin 6
AD0IN8	Input	AD0 analog input pin 8
AD0IN9	Input	AD0 analog input pin 9
AD0IN10	Input	AD0 analog input pin 10
AD0IN11	Input	AD0 analog input pin 11
AD0IN12	Input	AD0 analog input pin 12
AD0IN13	Input	AD0 analog input pin 13
AD0IN14	Input	AD0 analog input pin 14
AD0IN15	Input	AD0 analog input pin 15
AD1IN0	Input	AD1 analog input pin 0
AD1IN1	Input	AD1 analog input pin 1
AD1IN4	Input	AD1 analog input pin 4
AD1IN5	Input	AD1 analog input pin 5
AD0TRG#	Input	Input pin for scan conversion trigger of AD0
AD0END	Output	Output pin for monitoring AD0 conversion timing of AD0IN0

27.3 Functions Assigned to Each Channel

Table 27.3 lists the functions assigned to each channel.

Table 27.3 List of Functions Assigned to Each Channel

A/D Converter	A/D Conversion Time/Channel	A/D Conversion Channel		Scan Conversion								A/D Data Register (Storage Register for A/D-Converted Value)	A/D-Converted Value Addition Mode	A/D Conversion Timing Monitor Pin (AD0END)	Interrupt Conversion								
				Trigger Source			Channel Selection	Scan Conversion Mode	End Interrupt/DMA Transfer Request Signal	DMA Transfer Function (DMAC)	End Flag (at Scan Completion)				Trigger Source			End Interrupt/DMA Transfer Request Signal	DMA Transfer Function	Enable (Channel Selection)	Trigger Source Selection	End Flag	End Interrupt Enable
				Source 1	Source 2	Source 3									Source 1	Source 2	Source 3						
		Pin Name	Symbol	Software	Pin	ATU-IIIS	ADiCER and ADiANS						ADiADS		ATU-IIIS		Software (ADiSTRG)		DMAC	ADiTRE	ADiTRS	ADiTRF	ADiTRD
A/D converter 0 (AD0)	1.25 µs	For self diagnosis 0	DIAG0	ADST(AD0CSR)	AD0 external trigger (AD0TRG)	Timer G4	DIAGM (AD0CER)	Single cycle scanning or continuous scanning	AD0I	○	ADF(AD0REF)	AD0DRD	×	×	×	×	×	×	×	×	×	×	×
		AD0 analog input pin 0	AD0IN0				AD0ANS0					AD0DR0	AD0ADS0	○	Timer TOU00	Timer TOU00	AD0STRG0	AD0ID0	○	AD0TRGE0	AD0TRS0	AD0TF0	AD0IDE0
		AD0 analog input pin 2	AD0IN2				AD0ANS2					AD0DR2	AD0ADS2	×	Timer TOU00	Timer TOU02	AD0STRG2	AD0ID2	○	AD0TRGE2	AD0TRS2	AD0TF2	AD0IDE2
		AD0 analog input pin 4	AD0IN4				AD0ANS4					AD0DR4	AD0ADS4	×	Timer TOU01	Timer TOU04	AD0STRG4	AD0ID4	×	AD0TRGE4	AD0TRS4	AD0TF4	AD0IDE4
		AD0 analog input pin 6	AD0IN6				AD0ANS6					AD0DR6	AD0ADS6	×	Timer TOU02	Timer TOU04	AD0STRG6	AD0ID6	×	AD0TRGE6	AD0TRS6	AD0TF6	AD0IDE6
		AD0 analog input pin 8	AD0IN8				AD0ANS8					AD0DR8	×	×	Timer TOU03	Timer TOU05	AD0STRG8	AD0ID8	×	AD0TRGE8	AD0TRS8	AD0TF8	AD0IDE8
		AD0 analog input pin 9	AD0IN9				AD0ANS9					AD0DR9	×	×	Timer TOU03	Timer TOU05	AD0STRG9	AD0ID9	×	AD0TRGE9	AD0TRS9	AD0TF9	AD0IDE9
		AD0 analog input pin 10	AD0IN10				AD0ANS10					AD0DR10	×	×	Timer TOU04	Timer TOU05	AD0STRG10	AD0ID10	×	AD0TRGE10	AD0TRS10	AD0TF10	AD0IDE10
		AD0 analog input pin 11	AD0IN11				AD0ANS11					AD0DR11	×	×	Timer TOU04	Timer TOU05	AD0STRG11	AD0ID11	×	AD0TRGE11	AD0TRS11	AD0TF11	AD0IDE11
		AD0 analog input pin 12	AD0IN12				AD0ANS12					AD0DR12	×	×	Timer TOU05	Timer TOU05	AD0STRG12	AD0ID12	×	AD0TRGE12	AD0TRS12	AD0TF12	AD0IDE12
		AD0 analog input pin 13	AD0IN13				AD0ANS13					AD0DR13	×	×	Timer TOU05	Timer TOU05	AD0STRG13	AD0ID13	×	AD0TRGE13	AD0TRS13	AD0TF13	AD0IDE13
		AD0 analog input pin 14	AD0IN14				AD0ANS14					AD0DR14	×	×	Timer TOU05	Timer TOU05	AD0STRG14	AD0ID14	×	AD0TRGE14	AD0TRS14	AD0TF14	AD0IDE14
		AD0 analog input pin 15	AD0IN15				AD0ANS15					AD0DR15	×	×	Timer TOU05	Timer TOU05	AD0STRG15	AD0ID15	○	AD0TRGE15	AD0TRS15	AD0TF15	AD0IDE15
A/D converter 1 (AD1)	1.25 µs	For self diagnosis 1	DIAG1	ADST(AD1CSR)	None	Timer G5	DIAGM (AD1CER)		AD1I	○	ADF(AD1REF)	×	×	×	×	×	×	×	×	×	×	×	×
		AD1 analog input pin 0	AD1IN0				AD1ANS0					AD1DR0	AD1ADS0	×	Timer TOU00	Timer TOU00	AD1STRG0	AD1ID0	×	AD1TRGE0	AD1TRS0	AD1TF0	AD1IDE0
		AD1 analog input pin 1	AD1IN1				AD1ANS1					AD1DR1	AD1ADS1	×	Timer TOU00	Timer TOU01	AD1STRG1	AD1ID1	×	AD1TRGE1	AD1TRS1	AD1TF1	AD1IDE1
		AD1 analog input pin 4	AD1IN4				AD1ANS4					AD1DR4	AD1ADS4	×	Timer TOU01	Timer TOU04	AD1STRG4	AD1ID4	×	AD1TRGE4	AD1TRS4	AD1TF4	AD1IDE4
		AD1 analog input pin 5	AD1IN5				AD1ANS5					AD1DR5	AD1ADS5	×	Timer TOU01	Timer TOU04	AD1STRG5	AD1ID5	×	AD1TRGE5	AD1TRS5	AD1TF5	AD1IDE5

27.4 Register Descriptions

Table 27.4 lists the register configuration of the A/D converter.

Table 27.4 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size*[1]	Page
A/D0 data register DIAG0	AD0DRD	H'0000	H'FFFF 403E	16	27-11
A/D0 data register 0	AD0DR0	H'0000	H'FFFF 4040	16	27-8
A/D0 data register 2	AD0DR2	H'0000	H'FFFF 4044	16	27-8
A/D0 data register 4	AD0DR4	H'0000	H'FFFF 4048	16	27-8
A/D0 data register 6	AD0DR6	H'0000	H'FFFF 404C	16	27-8
A/D0 data register 8	AD0DR8	H'0000	H'FFFF 4050	16	27-10
A/D0 data register 9	AD0DR9	H'0000	H'FFFF 4052	16	27-10
A/D0 data register 10	AD0DR10	H'0000	H'FFFF 4054	16	27-10
A/D0 data register 11	AD0DR11	H'0000	H'FFFF 4056	16	27-10
A/D0 data register 12	AD0DR12	H'0000	H'FFFF 4058	16	27-10
A/D0 data register 13	AD0DR13	H'0000	H'FFFF 405A	16	27-10
A/D0 data register 14	AD0DR14	H'0000	H'FFFF 405C	16	27-10
A/D0 data register 15	AD0DR15	H'0000	H'FFFF 405E	16	27-10
A/D1 data register DIAG1	AD1DRD	H'0000	H'FFFF 443E	16	27-11
A/D1 data register 0	AD1DR0	H'0000	H'FFFF 4440	16	27-8
A/D1 data register 1	AD1DR1	H'0000	H'FFFF 4442	16	27-8
A/D1 data register 4	AD1DR4	H'0000	H'FFFF 4448	16	27-8
A/D1 data register 5	AD1DR5	H'0000	H'FFFF 444A	16	27-8
A/D0 control register	AD0CSR	H'00	H'FFFF 4000	8	27-12
A/D1 control register	AD1CSR	H'00	H'FFFF 4400	8	27-12
A/D0 conversion status register	AD0REF	H'00	H'FFFF 4002	8	27-17
A/D1 conversion status register	AD1REF	H'00	H'FFFF 4402	8	27-17
A/D0 interrupt trigger enable register	AD0TRE	H'0000	H'FFFF 4004	8, 16	27-21
A/D1 interrupt trigger enable register	AD1TRE	H'00	H'FFFF 4410	8	27-21
A/D0 interrupt trigger conversion end flag register	AD0TRF	H'0000	H'FFFF 4006	8, 16	27-26
A/D1 interrupt trigger conversion end flag register	AD1TRF	H'00	H'FFFF 4412	8	27-27
A/D0 interrupt trigger source select register	AD0TRS	H'0000	H'FFFF 4008	8, 16	27-22
A/D1 interrupt trigger source select register	AD1TRS	H'00	H'FFFF 4414	8	27-23
A/D0 interrupt software trigger register	AD0STRG	H'0000	H'FFFF 400A	8, 16	27-24
A/D1 interrupt software trigger register	AD1STRG	H'00	H'FFFF 4416	8	27-25
A/D0 interrupt trigger conversion end interrupt enable register	AD0TRD	H'0000	H'FFFF 400C	8, 16	27-28
A/D1 interrupt trigger conversion end interrupt enable register	AD1TRD	H'00	H'FFFF 4418	8	27-29
A/D0 converted value addition mode select register	AD0ADS	H'00	H'FFFF 401C	8	27-18

Register Name	Abbreviation	After Reset	P4 Address	Size*1	Page
A/D1 converted value addition mode select register	AD1ADS	H'00	H'FFFF 441C	8	27-19
A/D0 converted value addition count select register	AD0ADC	H'00	H'FFFF 401E	8	27-20
A/D1 converted value addition count select register	AD1ADC	H'00	H'FFFF 441E	8	27-20
A/D0 channel select register	AD0ANS	H'0000	H'FFFF 4020	8, 16	27-16
A/D1 channel select register	AD1ANS	H'0000	H'FFFF 4420	8, 16	27-16
A/D0 control extended register	AD0CER	H'0000	H'FFFF 4030	8, 16	27-14
A/D1 control extended register	AD1CER	H'0000	H'FFFF 4430	8, 16	27-14

Notes: *1 16-bit access can be made only at word boundaries.

- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

27.4.1 A/D0 Data Registers m and DIAG0 (AD0DRm and AD0DRD) A/D1 Data Registers n and DIAG1 (AD1DRn and AD1DRD)

The AD0DRm and AD1DRn registers are read-only registers that hold the results of A/D conversion of the AD0INm and AD1INn. The registers should be read in words. See table 27.3 for the correspondence between the AD0DRm registers and the AD0INm and between the AD1DRn registers and the AD1INn. The registers should be read in words.

The AD0DRD and AD1DRD registers are read-only registers that store the self-diagnosed A/D-converted results of channels AD0 and AD1, respectively.

The AD0DRm, AD1DRn, AD0DRD, and AD1DRD registers use different formats depending on the settings of the A/D data register format select bit (ADRFMT) and the A/D-converted value addition channel select bits (AD0ADS0 to 7 and AD1ADS0 to 7). Note that A/D-converted value addition mode can be selected only for registers AD0DR0, 2, 4, 6, AD1DR0, 1, 4, and 5. The self-diagnosis status bit is added to AD0DRD and AD1DRD.

(1) A/D0 Data Registers 0, 2, 4, 6, A/D1 Data Registers 0, 1, 4, and 5 (AD0DR0, 2, 4, 6, AD1DR0, 1, 4, and 5)

When A/D-converted value addition mode is not selected, left-shift or right-shift format can be selected by setting the ADRFMT bit in the A/D control extended register. At this time, bits AD11 to AD0 show the 12-bit A/D-converted value. The other bits are reserved. They are always read as "0".

When A/D-converted value addition mode is selected, the setting of the ADRFMT bit is invalid. At this time, bits AD13 to AD0 show the sum of all the values added in A/D converted value addition mode. The other bits are reserved. They are always read as "0".

The following minimum and maximum values apply to channels for which A/D-converted value addition mode is selected:

1st conversion: H'0000 ≤ AD0DR0, 2, 4, 6, AD1DR0, 1, 4, 5 ≤ H'3FFC
2nd conversion: H'0000 ≤ AD0DR0, 2, 4, 6, AD1DR0, 1, 4, 5 ≤ H'7FF8
3rd conversion: H'0000 ≤ AD0DR0, 2, 4, 6, AD1DR0, 1, 4, 5 ≤ H'BFF4
4th conversion: H'0000 ≤ AD0DR0, 2, 4, 6, AD1DR0, 1, 4, 5 ≤ H'FFF0

- When A/D-converted value addition mode is not selected

A/D0 Data Register 0 (AD0DR0)	<P4 address: H'FFFF 4040>
A/D0 Data Register 2 (AD0DR2)	<P4 address: H'FFFF 4044>
A/D0 Data Register 4 (AD0DR4)	<P4 address: H'FFFF 4048>
A/D0 Data Register 6 (AD0DR6)	<P4 address: H'FFFF 404C>
A/D1 Data Register 0 (AD1DR0)	<P4 address: H'FFFF 4440>
A/D1 Data Register 1 (AD1DR1)	<P4 address: H'FFFF 4442>
A/D1 Data Register 4 (AD1DR4)	<P4 address: H'FFFF 4448>
A/D1 Data Register 5 (AD1DR5)	<P4 address: H'FFFF 444A>

- When left-shift format is selected

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 4	AD11 to AD0	All 0	R	—	12-Bit A/D-Converted Value
3 to 0	—	All 0	0	—	Reserved Bits These bits are always read as "0".

- When right-shift format is selected

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	—	Reserved Bits These bits are always read as "0".
11 to 0	AD11 to AD0	All 0	R	—	12-Bit A/D-Converted Value

- When A/D-converted value addition mode is selected

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD13	AD12	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 2	AD13 to AD0	All 0	R	—	Sum of All Values Added in A/D-Converted Value Addition Mode
1, 0	—	All 0	0	—	Reserved Bits These bits are always read as "0".

(2) A/D0 Data Registers 8 to 15 (AD0DR8 to AD0DR15)

Left-shift or right-shift format can be selected by setting the ADRFMT bit in the AD0CER register.

Bits AD11 to AD0 show the 12-bit A/D converted value. The other bits are reserved. They are always read as "0".

Note that registers AD0DR8 to AD0DR15 cannot be set to A/D-converted value addition mode.

A/D0 Data Registers 8 to 15 (AD0DR8 to AD0DR15) <P4 address: location H'FFFF 4050 to H'FFFF 405E>

- When left-shift format is selected

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 4	AD11 to AD0	All 0	R	—	12-Bit A/D-Converted Value
3 to 0	—	All 0	0	—	Reserved Bits These bits are always read as "0".

- When right-shift format is selected

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	—	Reserved Bits These bits are always read as "0".
11 to 0	AD11 to AD0	All 0	R	—	12-Bit A/D-Converted Value

(3) A/D0 Data Register DIAG0 and A/D1 Data Register DIAG1 (AD0DRD and AD1DRD)

Left-shift or right-shift format can be selected by setting the ADRFMT bit in the ADiCER register. At this time, bits AD11 to AD0 show the 12-bit A/D-converted value. The other bits are reserved. They are always read as "0".

Note that registers AD0DRD and AD1DRD cannot be set to A/D-converted value addition mode.

A/D0 Data Registers DIAG0 (AD0DRD) <P4 address: location H'FFFF 403E>
A/D1 Data Registers DIAG1 (AD1DRD) <P4 address: location H'FFFF 443E>

- When left-shift format is selected

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	DIAGST	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 4	AD11 to AD0	All 0	R	—	12-Bit A/D-Converted Value
3, 2	—	All 0	0	—	Reserved Bits These bits are always read as "0".
1, 0	DIAGST	All 0	R	—	Self-Diagnosis Status These bits show the converted voltage based on self diagnosis. For details on self diagnosis, see section 27.4.3, A/Di Control Extended Register (ADiCER). 00: Indicate not a single self diagnosis is performed after hardware reset 01: Indicate the AVREF × 0 voltage value is self-diagnosed 10: Indicate the AVREF × 1/2 voltage value is self-diagnosed 11: Indicate the AVREF × 1 voltage value is self-diagnosed

- When right-shift format is selected

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DIAGST		—	—	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15, 14	DIAGST	All 0	R	—	Self-Diagnosis Status These bits show the converted voltage based on self diagnosis. For details on self diagnosis, see section 27.4.3, A/Di Control Extended Register (ADiCER). 00: Indicate not a single self diagnosis is performed after hardware reset 01: Indicate the AVREF × 0 voltage value is self-diagnosed 10: Indicate the AVREF × 1/2 voltage value is self-diagnosed 11: Indicate the AVREF × 1 voltage value is self-diagnosed
13, 12	—	All 0	0	—	Reserved Bits These bits are always read as "0".
11 to 0	AD11 to AD0	All 0	R	—	12-Bit A/D-Converted Value

27.4.2 A/Di Control Register (ADiCSR)

The ADiCSR register is used to make settings for scan conversion mode.

A/D0 Control Register (AD0CSR)	<P4 address: location H'FFFF 4000>
A/D1 Control Register (AD1CSR)	<P4 address: location H'FFFF 4400>

Bit:	7	6	5	4	3	2	1	0
	ADST	ADCS	—	ADIE	—	—	TRGE	EXTRG
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	ADST	0	R	W	Scan Conversion Start Bit Starts or stops scan conversion. When the ADST bit is set from "0" to "1", the A/D converter detects the rising edge of the ADST bit and then starts scan conversion. When the ADST bit is cleared from "1" to "0", the A/D converter detects the falling edge of the ADST bit and then stops scan conversion. The ADST bit does not affect interrupt conversion. To check whether a scan conversion is being performed, read the ADSCACT bit in the ADiREF register. 0: Stops a scan conversion process. 1: Starts a scan conversion process.
6	ADCS	0	R	W	Scan Conversion Mode Select Bit Selects scan conversion mode. To prevent incorrect operation, the value of the ADCS bit must be changed while the ADSCACT bit in the ADiREF register is cleared to "0". In single-cycle scan mode, scanning is performed once and, upon completion, scan conversion ends. In continuous scan mode, scanning is repeated indefinitely. Scan conversion can be stopped by writing "0" to the ADST bit when the bit is set to "1". In scan conversion, A/D conversion proceeds in order from lower-numbered to higher-numbered channels (AD0IN0 to AD0IN15 for AD0 and AD1IN0 to AD1IN5 for AD1). In continuous scan mode, the conversion process returns to the first channel when all the selected channels have been converted. 0: Single-cycle scan mode 1: Continuous scan mode
5	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
4	ADIE	0	R	W	Interrupt Enable Bit Enables or disables generation of the A/D scan conversion end interrupt (ADI). To prevent incorrect operation, the value of the ADIE bit must be changed while the ADSCACT bit in the ADiREF register is cleared to "0". When the ADF bit in the ADiREF register is set to "1" upon completion of each scan in the scan conversion process, an ADI interrupt is generated when the ADIE bit is set to "1". The ADI interrupt can be cleared by clearing the ADF bit to "0" or clearing the ADIE bit to "0". 0: Disables ADI interrupt generation upon scanning completion. 1: Enables ADI interrupt generation upon scanning completion.

Bit	Abbreviation	After Reset	R	W	Description
3, 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	TRGE	0	R	W	Trigger Enable Bit Enables or disables scan conversion to be started by an external trigger (AD0TGR) or ATU-IIIS timer trigger (AD0: timer G4, AD1: timer G5). 0: Disables scan conversion to be started by an external trigger or ATU-IIIS timer trigger. 1: Enables scan conversion to be started by an external trigger or ATU-IIIS timer trigger.
0	EXTRG	0	R	W	Trigger Select Bit Selects a trigger source for scan conversion. Either the external trigger (AD0TGR) or the ATU-IIIS timer trigger (AD0: timer G4, AD1: timer G5) is selected. 0: Scan conversion is started by an ATU-IIIS timer trigger. 1: Scan conversion is started by an external trigger. Note: • For the AD1CSR register, always write "0".

Notes:

- Starting AD0 and AD1 scan conversion simultaneously

  AD0 and AD1 scan conversion can be started simultaneously by writing "1" to the TRGE bit and "0" to the EXTRG bit in both AD0 and AD1, and by inputting the timer G4 trigger and the timer G5 trigger from the ATU-IIIS simultaneously. For details on the timer G settings, see section 21, Advanced Timer Unit IIIS (ATU-IIIS).

- Starting AD0 and AD1 scan conversion with different timing from each other

  AD0 and AD1 scan conversion can be started with different timings by writing "1" to the TRGE bit and "0" to the EXTRG bit for AD0 and AD1, and by inputting the timer G4 trigger and timer G5 trigger from the ATU-IIIS with different timings. For details on the timer G settings, see section 21, Advanced Timer Unit IIIS (ATU-IIIS).

- Staring an interrupt conversion and a scan conversion simultaneously

  If "1" is written to the TRGE bit and "0" is written to the EXTRG bit in AD0, and "1" is written to the AD0TRGE4 bit and the AD0TRGE6 bit in the AD0TRE register, and then the timer G4 trigger and the timer TOU01 trigger are simultaneously input from the ATU-IIIS, the AD0 executes the following operations in the indicated sequence: AD0IN4 interrupt conversion → AD0IN6 interrupt conversion → scan conversion. To execute a scan conversion only, clear both the AD0TRGE4 and AD0TRGE6 bits to "0". Either AD0IN4 or AD0IN6 can also execute a single-channel interrupt conversion. Similar operations can also be accomplished through combinations of an AD1 scan conversion and an AD1IN4 interrupt conversion by using the timer G5 trigger and the timer TOU01 trigger from the ATU-IIIS.

- Starting a scan conversion using an external trigger

  If "1" is written to both the TRGE and EXTRG bits when "H" level signals are input to the external trigger pins (AD0TRG#), and then if a "L" level pulse is input to either the AD0TRG# pin, either AD0 detects a pulse falling edge and starts the scan conversion process. In this case, the "L" pulse width must be 1.5Pck clock or more.

  The required high pulse width depends on the setting of the CKS bit in the ADiCER register.

  When CKS = "0" : 2Pck clock or more

  When CKS = "1" : 4Pck clock or more

- Independent of the ADST bit, external triggers and ATU-IIIS timer triggers, startup of a scan conversion is enabled when the ADSCACT bit in the ADiREF register is cleared to "0". The startup source for a scan conversion is not retained.

- Regarding the startup cycle time for scan conversion and interrupt conversion initiated by the ATU-IIIS timer trigger, specify an ATU-IIIS timer trigger cycle time that exceeds the scan conversion time (for example, 56 states when the CKS bit is cleared to "0" and 112 states when the CKS bit is set to "1", respectively, for conversion on a single channel) and the interrupt conversion time (for example, 50 states when the CKS bit is cleared to "0" and 100 states when the CKS bit is set to "1", for conversion on a single channel using one trigger source). For details on the timer trigger cycle setting, see section 21, Advanced Timer Unit IIIS (ATU-IIIS).

27.4.3 A/Di Control Extended Register (ADiCER)

The ADiCER register is used to make settings such as self-diagnosis mode, data format, and clock selection.

A/D0 Control Extended Register (AD0CER) <P4 address: location H'FFFF 4030>
A/D1 Control Extended Register (AD1CER) <P4 address: location H'FFFF 4430>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR FMT	—	—	—	DIAG M	DIAG LD	DIAGVAL		CKS	—	—	—	—	—	—	IT TRGS
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	ADRFMT	0	R	W	A/D Data Register Format Select Bit The format of the A/D data register corresponding to the channel with A/D-converted value addition mode selected is left-shift, regardless of the ADRFMT bit. For details on the A/D data register formats, see section 27.4.1, A/D0 Data Registers m and DIAG0 (AD0DRm and AD0DRD) A/D1 Data Registers n and DIAG1 (AD1DRn and AD1DRD). 0: A/D data register format is left-shift 1: A/D data register format is right-shift
14 to 12	—	All 0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
11	DIAGM	0	R	W	Self-Diagnosis Enable Bit Self diagnosis is a function to detect an error in the A/D converter (AD0, AD1). The internal generated three voltage values of AVREF (AVREFH) × 0, AVREF × 1/2, and AVREF × 1 are converted by AD1 and AD1. Then, the function reads AD0DRD and AD1DRD by software to determine whether the converted values are within the normal range (normal) or not (error). Self diagnosis is performed before scan converting the channel with the lowest number. The self diagnosis execution time is the same as the A/D conversion time for one channel. To prevent incorrect operation, the DIAGM bit must be switched only while the ADSCACT bit in the ADiREF register is set to "0". 0: A/D converter is not self-diagnosed 1: A/D converter is self-diagnosed

Bit	Abbreviation	After Reset	R	W	Description
10	DIAGLD	0	R	W	Self-Diagnosis Mode Select Bit Selects whether to rotate or fix the three voltage values that to be converted based on self diagnosis. When the DIAGLD bit is set to "0", the voltage values are rotated in order of AVREF × 0 to AVREF × 1/2 to AVREF × 1. When self diagnosis is performed from AVREF × 0 after a hardware reset, the rotation does not return to AVREF × 0 even scan conversion is completed. If scan conversion is performed again, the rotation begins from the previous order. When the DIAGLD bit is set to "1", the conversion is performed on the fixed voltage selected with the DIAGVAL bit in the ADiCER register (no automatic rotation). When the the DIAGLD bit is also set to "0" again, the rotation begins from the fixed voltage value (load function). 0: Self diagnosis is performed by automatically rotating the values 1: Self diagnosis is performed by fixing the values with the DIAGVAL bit setting
9, 8	DIAGVAL	00	R	W	Self-Diagnosis Voltage Select Bits For details, refer to the description of the DIAGLD bit. While these bits are "B'00", self diagnosis must not be performed by setting the DIAGLD bit to "1". 00: Reserved 01: AVREF × 0 voltage value is self-diagnosed 10: AVREF × 1/2 voltage value is self-diagnosed 11: AVREF × 1 voltage value is self-diagnosed
7	CKS	0	R	W	Clock Select Bit Selects A/D conversion time. To prevent incorrect operation, the values of the ADSCACT and ADITACT bits in the ADiREF register must be "0" when the value of the CKS bit is changed. 0: A/D conversion time = 50 states (Pck conversion) 1: A/D conversion time = 100 states (Pck conversion)
6 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	ITTRGS	0	R	W	Expanded Interrupt Conversion Trigger Source Select Bit Selects either the source 1 timer or the source 2 timer as the interrupt conversion trigger source for AD0INm and AD1INn. The setting of the ITTRGS bit has effect only when the AD0TRGEm or AD1TRGEn bit in the ADiTRE register is set to "1" and the AD0TRSm and AD1TRSn bit in the ADiTRS register is cleared to "0". See table 27.3. 0: AD0INm and AD1INn interrupt conversion is triggered by the source 1 timer 1: AD0INm and AD1INn interrupt conversion is triggered by the source 2 timer

Legend: m = 0, 2, 4, 6, 8 to 15, n = 0, 1, 4, 5.

27.4.4 A/Di Channel Select Register (ADiANS)

The ADiANS register is used to select channels that are subject to scan conversion. To prevent incorrect operation, the ADSCACT bit in the ADiREF register must be cleared to "0" while the ADiANS register values are changed.

Note: • The ADiANS register selects scan conversion channels; it is not used to select interrupt conversion channels. An interrupt conversion channel is selected by the ADiTRE register.
If a channel is selected by both the ADiANS and ADiTRE registers, it is subject to conversion in both scan conversion and interrupt conversion. A channel that is selected only by the ADiTRE register is excluded from the list of channels eligible for scan conversion, and only receives an interrupt conversion.

(1) A/D0 Channel Select Register (AD0ANS)

A/D0 Channel Select Register (AD0ANS) <P4 address: location H'FFFF 4020>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD0 ANS15	AD0 ANS14	AD0 ANS13	AD0 ANS12	AD0 ANS11	AD0 ANS10	AD0 ANS9	AD0 ANS8	AD0 ANS7	AD0 ANS6	AD0 ANS5	AD0 ANS4	AD0 ANS3	AD0 ANS2	AD0 ANS1	AD0 ANS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	AD0ANS15 to AD0ANS0	All 0	R	W	Setting the AD0ANSm bit to "1" selects AD0INm. The correspondence between AD0INm and the AD0ANSm bit is shown in table 27.3. 0: AD0INm is not selected. 1: AD0INm is selected.

Note: • Set "0" to AD0INm (m = 1, 3, 5, 7).
Legend: m = 0 to 15

(2) A/D1 Channel Select Register (AD1ANS)

A/D1 Channel Select Register (AD1ANS) <P4 address: location H'FFFF 4420>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	AD1 ANS7	AD1 ANS6	AD1 ANS5	AD1 ANS4	AD1 ANS3	AD1 ANS2	AD1 ANS1	AD1 ANS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7 to 0	AD1ANS7 to AD1ANS0	All 0	R	W	Setting the AD1ANSn bit to "1" selects AD1INn. The correspondence between AD1INn and the AD1ANSn bit is shown in table 27.3. 0: AD1INn is not selected. 1: AD1INn is selected.

Note: • Set "0" to AD1INn (n = 1, 3, 5, 7).
Legend: n = 0 to 7

27.4.5 A/Di Conversion Status Register (ADiREF)

The ADiREF register indicates the status of the A/D converter.

A/D0 Conversion Status Register (AD0REF) <P4 address: location H'FFFF 4002>
A/D1 Conversion Status Register (AD1REF) <P4 address: location H'FFFF 4402>

Bit:	7	6	5	4	3	2	1	0
	AD SCACT	AD ITACT	—	—	—	—	—	ADF
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	ADSCACT	0	R	—	Scan Conversion Status Bit Indicates whether the scan conversion process is in the idle state or it is being executed. If an interrupt conversion is started during a scan conversion, the A/D converter stops the scan conversion process and preferentially executes the interrupt conversion. However, until such time that all scan conversion is completed, the ADSCACT bit maintains to be set to 1 and is not cleared to "0". 0: Scan conversion process is in idle state. 1: Scan conversion process is being executed.
6	ADITACT	0	R	—	Interrupt Conversion Status Bit Indicates whether the interrupt conversion process is in the idle state or it is being executed. The ADSCACT and ADITACT bits can indicate the status of the AD0 and AD1. For details, see table 27.5. 0: Interrupt conversion process is in idle state. 1: Interrupt conversion process is being executed.
5 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	ADF	0	R	*1	Single Scan End Flag This bit is set to 1 each time scanning ends in the scan conversion process (when all selected channels are converted). 1 cannot be written to this bit. When the ADF bit is set to 1, either a scan conversion end interrupt or a DMA transfer request to the DMAC can be generated. In this manner, processing such as storing the contents of the A/D data register to the SHwyRAM can be implemented by means of either software or the DMAC. 0: Scan conversion process is in idle state. 1: Single scan has been completed and the A/D-converted values on all selected AD0INm and AD1INn have been transferred to the A/D data register. [Conditions for clearing to "0"] • 0 is written to this bit after reading 1. • A DMA transfer request from the ADI is accepted by the DMAC. [Condition for setting to "1"] • All analog conversion has been completed during each scanning in scan conversion process.

Note: *1 Do not write "1" to the ADF bit. To clear the flag, write "0" to it after reading its state as "1".

Table 27.5 Relationship between AD0 and AD1 Status and ADSCACT and ADITACT

ADSCACT Bit	ADITACT Bit	AD0 and AD1 Status	Source of Scan Conversion	Source of Interrupt Conversion
0	0	Idle state	No	No
	1	Interrupt conversion	No	Yes
1	0	Scan conversion	Yes	No
	1	Interrupt conversion	Yes	Yes

27.4.6 A/Di-Converted Value Addition Mode Select Register (ADiADS)

The ADiADS register selects channels AD0INm (m = 0, 2, 4, 6) and AD1INn (n = 0, 1, 4, 5) on which A/D conversion is performed successively two to four times, after which the converted values are added (integrated).

(1) A/D0-Converted Value Addition Mode Select Register (AD0ADS)

A/D0-Converted Value Addition Mode Select Register (AD0ADS) <P4 address: location H'FFFF 401C>

Bit:	7	6	5	4	3	2	1	0
	AD0 ADS7	AD0 ADS6	AD0 ADS5	AD0 ADS4	AD0 ADS3	AD0 ADS2	AD0 ADS1	AD0 ADS0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AD0ADS7 to AD0ADS0	All 0	R	W	A/D-Converted Value Addition Channel Select Bits When the AD0ADSn bit is set to "1", the A/D converter performs conversion on AD0INn successively 2 to 4 times and returns the added (integrated) conversion results to the AD0DRn register. If the AD0ADSn bit is cleared to "0", the A/D converter performs a normal 1-time conversion of AD0INn and returns the conversion result to the AD0DRn register. Irrespective of whether a scan conversion or an interrupt conversion is to be performed, the A/D converter determines whether or not to perform an addition according to the AD0ADSn value. To prevent incorrect operation, both the ADSCACT and ADITACT bits in AD0REF must be cleared to "0" while the AD0ADSn bit value is changed. The correspondence between AD0INn and the AD0ADSn bit is shown in table 27.3. How to select the addition count is described in section 27.4.7, A/Di-Converted Value Addition Count Select Register (ADiADC). 0: A/D-converted value addition mode is not selected. 1: A/D-converted value addition mode (successively 2 to 4 times of addition) is selected.

Note: • Set "0" to AD0ADSn (n = 1, 3, 5, 7).

Legend: n = 0 to 7

(2) A/D1-Converted Value Addition Mode Select Register (AD1ADS)

A/D1-Converted Value Addition Mode Select Register (AD1ADS) <P4 address: location H'FFFF 441C>

Bit:	7	6	5	4	3	2	1	0
	AD1 ADS7	AD1 ADS6	AD1 ADS5	AD1 ADS4	AD1 ADS3	AD1 ADS2	AD1 ADS1	AD1 ADS0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AD1ADS7 to AD1ADS0	All 0	R	W	A/D-Converted Value Addition Channel Select Bits When the AD1ADSn bit is set to "1", the A/D converter performs conversion on AD1INn successively 2 to 4 times and returns the added (integrated) conversion results to the AD1DRn register. If the AD1ADSn bit is cleared to "0", the A/D converter performs a normal 1-time conversion of AD1INn and returns the conversion result to the AD1DRn register. Irrespective of whether a scan conversion or an interrupt conversion is to be performed, the A/D converter determines whether or not to perform an addition according to the AD1ADSn bit value. To prevent incorrect operation, both the ADSCACT and ADITACT bits in the AD1REF register must be cleared to "0" while the AD1ADSn bit value is changed. The correspondence between AD1INn and the AD1ADSn bit is shown in table 27.3. How to select the addition count is described in section 27.4.7, A/Di-Converted Value Addition Count Select Register (ADiADC). 0: A/D-converted value addition mode is not selected. 1: A/D-converted value addition mode (successively 2 to 4 times of addition) is selected.

Note: • Set "0" to AD1ADSn (n = 2, 3, 6, 7).

Legend: n = 0 to 7

Figure 27.2 shows a scan conversion sequence in which both the AD0ADS2 and AD0ADS6 bits are set to "1", based on the assumption that the addition count is set to 4, and that channels AD0IN0, AD0IN2, AD0IN4, and AD0IN6, are selected. The conversion process begins with AD0IN. The AD0IN2 conversion is performed successively 4 times, and the addition (integration) value is returned to the data register, after which the AD0IN4 conversion process is started. If an interrupt conversion is requested in the midst of a scan conversion, the scan conversion process is stopped and an A/D conversion is preferentially executed on the channel in which an interrupt conversion was requested. Upon completion of the interrupt conversion, the scan conversion process is resumed from the A/D conversion on the interrupted channel. However, if the AD0ADSn bit of the interrupted channel (AD0INn) is set to 1, even if addition has been performed at least once (two to four times), the conversion is restarted from the 1st conversion.

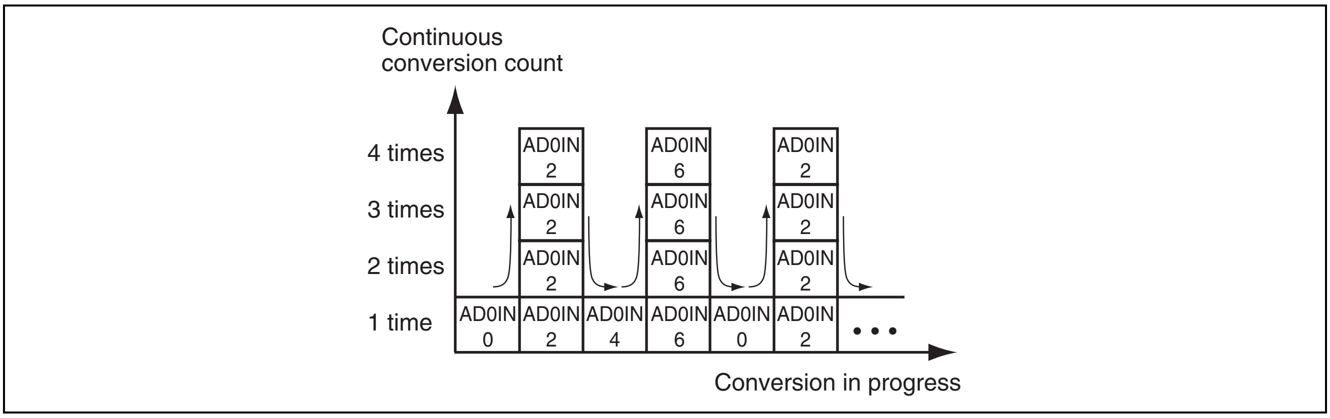


Figure 27.2 Scan Conversion Sequence with Bits AD0ADS2 and AD0ADS6 Set to "1"

27.4.7 A/Di-Converted Value Addition Count Select Register (ADiADC)

The ADiADC register sets the addition count for channels for which A/D-converted value addition mode is selected.

A/D0-Converted Value Addition Count Select Register (AD0ADC) <P4 address: location H'FFFF 401E>
A/D1-Converted Value Addition Count Select Register (AD1ADC) <P4 address: location H'FFFF 441E>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ADC	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1, 0	ADC	00	R	W	Addition Count Select Bits These bits select the number of additions to be performed in A/D-converted value addition mode. These bits have no effect on the A/D conversion of channels for which A/D-converted value addition mode is not selected. To prevent incorrect operation, both the ADSCACT and ADITACT bits in the ADiREF register must be cleared to "0" while the ADC1 and ADC0 bit values are changed. 00: 1-time conversion (normal conversion) 01: 2-time conversion 10: 3-time conversion 11: 4-time conversion

27.4.8 A/Di Interrupt Trigger Enable Register (ADiTRE)

The ADiTRE register enables or disables an interrupt conversion request for AD0INm and AD1INn. Channels for which interrupt conversion is enabled are subjected to an interrupt conversion when a corresponding interrupt conversion request is input.

(1) A/D0 Interrupt Trigger Enable Register (AD0TRE)

A/D0 Interrupt Trigger Enable Register (AD0TRE) <P4 address: location H'FFFF 4004>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD0TRGE15	AD0TRGE14	AD0TRGE13	AD0TRGE12	AD0TRGE11	AD0TRGE10	AD0TRGE9	AD0TRGE8	AD0TRGE7	AD0TRGE6	AD0TRGE5	AD0TRGE4	AD0TRGE3	AD0TRGE2	AD0TRGE1	AD0TRGE0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	AD0TRGE15 to AD0TRGE0	All 0	R	W	Interrupt Conversion Request Enable Bits Setting the AD0TRGEm bit to "1" enables the interrupt conversion request to the corresponding AD0INm channel. The correspondence among the AD0TRGEm bit, AD0INm, and the interrupt request trigger source is shown in table 27.3. 0: Disables an interrupt conversion request to AD0INm by ATU-IIIS timer or software trigger (AD0STRGm). 1: Enables an interrupt conversion request to AD0INm by ATU-IIIS timer or software trigger (AD0STRGm).

Note: • Set "0" to AD0TRGEm (m = 1, 3, 5, 7).

Legend: m = 0 to 15

(2) A/D1 Interrupt Trigger Enable Register (AD1TRE)

A/D1 Interrupt Trigger Enable Register (AD1TRE) <P4 address: location H'FFFF 4410>

Bit:	7	6	5	4	3	2	1	0
	AD1TRGE7	AD1TRGE6	AD1TRGE5	AD1TRGE4	AD1TRGE3	AD1TRGE2	AD1TRGE1	AD1TRGE0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AD1TRGE7 to AD1TRGE0	All 0	R	W	Interrupt Conversion Request Enable Bits Setting the AD1TRGEn bit to "1" enables the interrupt conversion request to the corresponding AD1INn channel. The correspondence among the AD1TRGEn bit, AD1INn, and the interrupt request trigger source is shown in table 27.3. 0: Disables an interrupt conversion request to AD1INn by ATU-IIIS timer or software trigger (AD1STRGn). 1: Enables an interrupt conversion request to AD1INn by ATU-IIIS timer or software trigger (AD1STRGn).

Note: • Set "0" to AD1TRGEn (n = 2, 3, 6, 7).

Legend: n = 0 to 7

27.4.9 A/Di Interrupt Trigger Source Select Register (ADiTRS)

The ADiTRS register selects the trigger source for interrupt conversion. Either an ATU-IIIS timer trigger or a software trigger caused by writing to the ADiTRS register can be selected.

(1) A/D0 Interrupt Trigger Source Select Register (AD0TRS)

A/D0 Interrupt Trigger Source Select Register (AD0TRS) <P4 address: location H'FFFF 4008>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD0TRS 15	AD0TRS 14	AD0TRS 13	AD0TRS 12	AD0TRS 11	AD0TRS 10	AD0TRS 9	AD0TRS 8	AD0TRS 7	AD0TRS 6	AD0TRS 5	AD0TRS 4	AD0TRS 3	AD0TRS 2	AD0TRS 1	AD0TRS 0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	AD0TRS15 to AD0TRS0	All 0	R	W	Interrupt Conversion Trigger Source Select Bits If the AD0TRSm bit is cleared to "0" and the AD0TREm bit in the AD0TRE register is set to "1", the A/D converter performs edge detection and begins interrupt conversion on AD0INm when a trigger source 1 or trigger source 2 timer trigger is input. If the AD0TRSm bit is set to "1", the A/D converter performs edge detection and begins interrupt conversion on AD0INm when "1" is written to the AD0STRGm bit in the AD0STRG register. Selection of trigger source 1 or trigger source 2 is accomplished by setting the ITTRGS bit in the AD0CER register. The correspondence among the AD0TRSm bit, AD0INm, and the interrupt request trigger source is shown in table 27.3. 0: Trigger source 1 or trigger source 2 used as AD0INm interrupt conversion request source 1: Software trigger (AD0STRGm) used as AD0INm interrupt conversion request source

Note: • Set "0" to AD0TRSm (m = 1, 3, 5, 7).

Legend: m = 0 to 15

(2) A/D1 Interrupt Trigger Source Select Register (AD1TRS)

A/D1 Interrupt Trigger Source Select Register (AD1TRS) <P4 address: location H'FFFF 4414>

Bit:	7	6	5	4	3	2	1	0
	AD1TRS7	AD1TRS6	AD1TRS5	AD1TRS4	AD1TRS3	AD1TRS2	AD1TRS1	AD1TRS0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AD1TRS7 to AD1TRS0	All 0	R	W	Interrupt Conversion Trigger Source Select Bits If the AD1TRSn bit is cleared to "0" and the AD1TREn bit in the AD1TRE register is set to "1", the A/D converter performs edge detection and begins AD1INn interrupt conversion when a trigger source 1 or trigger source 2 timer trigger is input. If the AD1TRSn bit is set to "1", the A/D converter performs edge detection and begins interrupt conversion on AD1INn when "1" is written to the AD1STRGn bit in the AD1STRG register. Selection of trigger source 1 or trigger source 2 is accomplished by setting the ITTRGS bit in the AD1CER register. The correspondence among the AD1TRSn bit, AD1INn, and the interrupt request trigger sources is shown in table 27.3. 0: Trigger source 1 or trigger source 2 used as AD1INn interrupt conversion request source 1: Software trigger (AD1STRGn) used as AD1INn interrupt conversion request source

Note: • Set "0" to AD1TRSn (n = 2, 3, 6, 7).

Legend: n = 0 to 7

27.4.10 A/Di Interrupt Software Trigger Register (ADiSTRG)

The ADiSTRG register starts an interrupt conversion by software. The ADiSTRG register is write-only register and it is always read as 0s.

(1) A/D0 Interrupt Software Trigger Register (AD0STRG)

A/D0 Interrupt Software Trigger Register (AD0STRG) <P4 address: location H'FFFF 400A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD0STRG 15	AD0STRG 14	AD0STRG 13	AD0STRG 12	AD0STRG 11	AD0STRG 10	AD0STRG 9	AD0STRG 8	AD0STRG 7	AD0STRG 6	AD0STRG 5	AD0STRG 4	AD0STRG 3	AD0STRG 2	AD0STRG 1	AD0STRG 0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	AD0STRG15 to AD0STRG0	All 0	0	W	Interrupt Conversion Software Trigger Bits If the AD0TRSm bit in the AD0TRS register corresponding to AD0INm is set to "1" and the AD0TREm bit in the AD0TRE register is set to "1", the A/D converter performs edge detection and begins interrupt conversion on AD0INm when "1" is written to the AD0STRGm bit. Write "0" to the AD0STRGm bit corresponding to each channel (AD0INm) not subject to interrupt conversion requests. Any AD0INm channel for which "0" is written to the corresponding AD0STRGm bit is not affected by any of these operations. When interrupt conversion requests are issued, the interrupt sources are stored in internal circuits in AD0INm units. When an interrupt conversion on a channel (AD0INm) with a stored source is performed and completed, the source associated with the AD0INm is cleared. Consequently, writing "1" to the AD0STRGm bit and subsequently writing "0" to it does not clear the source associated with AD0INm, so interrupt conversion is executed. However, writing "1" to the AD0STRGm bit when a source is already pending does not cause interrupt conversion to be performed on AD0INm twice. There is one source per channel. This also applies to the execution of an interrupt conversion in response to a request from an ATU-IIIS timer trigger. See table 27.3 for the correspondence between each AD0STRGm bit and AD0INm channel. 0: No interrupt conversion request (software trigger) on AD0INm 1: Interrupt conversion request (software trigger) on AD0INm

Note: • Set "0" to AD0STRGm (m = 1, 3, 5, 7).

Legend: m = 0 to 15

(2) A/D1 Interrupt Software Trigger Register (AD1STRG)

A/D1 Interrupt Software Trigger Register (AD1STRG) <P4 address: location H'FFFF 4416>

Bit:	7	6	5	4	3	2	1	0
	AD1STRG7	AD1STRG6	AD1STRG5	AD1STRG4	AD1STRG3	AD1STRG2	AD1STRG1	AD1STRG0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AD1STRG7 to AD1STRG0	All 0	0	W	Interrupt Conversion Software Trigger Bits If the AD1TRSn bit in the AD1TRS register corresponding to AD1Nn is set to "1" and the AD1TREn bit in the AD1TRE register is set to "1", the A/D converter performs edge detection and begins interrupt conversion on AD1INn when "1" is written to the AD1STRGn bit. Write "0" to the AD1STRGn bit corresponding to each channel (AD1INn) not subject to interrupt conversion requests. Any AD1INn channel for which "0" is written to the corresponding AD1STRGn bit is not affected by any of these operations. When interrupt conversion requests are issued, the interrupt sources are stored in internal circuits in AD1INn units. When an interrupt conversion on a channel (AD1INn) with a stored source is performed and completed, the source associated with the AD1INn is cleared. Consequently, writing "1" to the AD1STRGn bit and subsequently writing "0" to it does not clear the source associated with AD1INn, so interrupt conversion is executed. However, writing "1" to the AD1STRGn bit when a source is already pending does not cause interrupt conversion to be performed on AD1INn twice. There is one source per channel. This also applies to the execution of an interrupt conversion in response to a request from an ATU-IIIS timer trigger. See table 27.3 for the correspondence between each AD1STRGn bit and AD1INn channel. 0: No interrupt conversion request (software trigger) on AD1INn 1: Interrupt conversion request (software trigger) on AD1INn

Note: • Set "0" to AD1STRGn (n = 2, 3, 6, 7).

Legend: n = 0 to 7

27.4.11 A/Di Interrupt Trigger Conversion End Flag Register (ADiTRF)

The ADiTRF register indicates that an interrupt conversion has been completed. When an interrupt conversion has been completed, the interrupt conversion end flag (AD0TFm and AD1TFn) corresponding to the channel (AD0INm and AD1INn) is set to "1".

(1) A/D0 Interrupt Trigger Conversion End Flag Register (AD0TRF)

A/D0 Interrupt Trigger Conversion End Flag Register (AD0TRF) <P4 address: location H'FFFF 4006>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD0TF15	AD0TF14	AD0TF13	AD0TF12	AD0TF11	AD0TF10	AD0TF9	AD0TF8	AD0TF7	AD0TF6	AD0TF5	AD0TF4	AD0TF3	AD0TF2	AD0TF1	AD0TF0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
14 to 4	AD0TF14 to AD0TF4	All 0	R	*[1]	Interrupt Conversion End Flag AD0TFp is a status flag bit which indicates that an interrupt conversion has been completed. "1" must not be written to AD0TFp. When AD0TFp is set to "1", an AD0INp interrupt conversion end interrupt (AD0IDp) can be generated. See table 27.3 for correspondence between AD0TFp and AD0INp. 0: Interrupt conversion process on AD0INp is in idle state. 1: Interrupt conversion process on AD0INp has been completed and the conversion result has been transferred to the AD0DRp register. [Condition for clearing to "0"] • Writing "0" to the AD0TFp bit after reading it as "1". [Condition for setting to "1"] • An interrupt conversion process on AD0INp has been completed.
15, 3 to 0	AD0TF15, AD0TF3 to AD0TF0	All 0	R	*[1]	Interrupt Conversion End Flag AD0TFq is a status flag bit that indicates that an interrupt conversion has been completed. Writing "1" to AD0TFq is prohibited. When AD0TFq is set to "1", an AD0INq interrupt conversion end interrupt (AD0IDq) can be generated. See table 27.3 for the correspondence between AD0TFq and AD0INq. 0: Interrupt conversion on AD0INq is in idle state 1: Interrupt conversion on AD0INq has completed and the A/D-converted value transferred to the AD0DRq register [Conditions for clearing to "0"] • Writing "0" to the AD0TFq bit after reading it as "1". • Receipt by DMAC of DMA transfer request at AD0IDq. [Condition for setting to "1"] • Completion of interrupt conversion on AD0INq.

Notes: *1 Do not write "1" to the AD0TFp or AD0TFq bits. To clear a flag, write "0" to an AD0TFp or AD0TFq bit after reading its state as "1".

- Even when AD0TFp or AD0TFq is not cleared to "0", an interrupt conversion request for AD0INp or AD0INq can be accepted. Exercise care regarding the timing of storing values in the AD0DRm register.
- Set "0" to AD0TFm (m = 1, 3, 5, 7).

Legend: p = 4 to 14, q = 0 to 3, 15

(2) A/D1 Interrupt Trigger Conversion End Flag Register (AD1TRF)

A/D1 Interrupt Trigger Conversion End Flag Register (AD1TRF) <P4 address: location H'FFFF 4412>

Bit:	7	6	5	4	3	2	1	0
	AD1TF7	AD1TF6	AD1TF5	AD1TF4	AD1TF3	AD1TF2	AD1TF1	AD1TF0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AD1TF7 to AD1TF0	All 0	R	*[1]	Interrupt Conversion End Flag AD1TFn is a status flag bit which indicates that an interrupt conversion has been completed. "1" must not be written to AD1TFn. When AD1TFn is set to "1", an AD1INn interrupt conversion end interrupt (AD1IDn) can be generated. See table 27.3 for correspondence between ADTFn and AD1INn. 0: Interrupt conversion process on AD1INn is in idle state. 1: Interrupt conversion process on AD1INn has been completed and the conversion result has been transferred to AD1DRn. [Condition for clearing to "0"] • Writing "0" to the AD1TFn bit after reading it as "1". [Condition for setting to "1"] • An interrupt conversion process on AD1INn has been completed.

Notes: *1 Do not write "1" to the AD1TFn bits. To clear a flag, write "0" to an AD1TFn bit after reading its state as "1".

- Even when the AD1TFn is not cleared to "0", an interrupt conversion request on AD1INn can be accepted. Storing timing on the AD1DRn register should be provided with care.
- Set "0" to AD1TFm (m = 2, 3, 6, 7).

Legend: n = 0 to 7

27.4.12 A/Di Interrupt Trigger Conversion End Interrupt Enable Register (ADiTRD)

The ADiTRD register enables or disables an A/D interrupt conversion end interrupt generation when the interrupt conversion end flag (AD0TFm and AD1TFn) in the ADiTRF register is set to "1".

(1) A/D0 Interrupt Trigger Conversion End Interrupt Enable Register (AD0TRD)

A/D0 Interrupt Trigger Conversion End Interrupt Enable Register (AD0TRD) <P4 address: location H'FFFF 400C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD0IDE 15	AD0IDE 14	AD0IDE 13	AD0IDE 12	AD0IDE 11	AD0IDE 10	AD0IDE 9	AD0IDE 8	AD0IDE 7	AD0IDE 6	AD0IDE 5	AD0IDE 4	AD0IDE 3	AD0IDE 2	AD0IDE 1	AD0IDE 0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
14 to 4	AD0IDE14 to AD0IDE4	All 0	R	W	Interrupt Conversion End Interrupt Enable Bits The AD0IDEp bit enables or disables an AD0INp interrupt conversion end interrupt (AD0IDp) to be generated. To prevent incorrect operation, the ADITACT bit in the AD0REF register must be cleared to "0" while the AD0IDEp bit value is changed. If the AD0IDEp bit is "1" when the AD0TFp bit in the interrupt conversion end flag register is set to "1" upon completion of AD0INp interrupt conversion, the AD0IDp signal is generated. The AD0IDp signal can be cleared by clearing the AD0TFp bit or the AD0IDEp bit to "0". The correspondence among the AD0IDEp bit, AD0INp, and AD0IDp is shown in table 27.3. 0: Disables an interrupt request upon completion of AD0INp interrupt conversion (AD0IDp). 1: Enables an interrupt request upon completion of AD0INp interrupt conversion (AD0IDp).
15, 3 to 0	AD0IDE15, AD0IDE3 to AD0IDE0	All 0	R	W	Interrupt Conversion End Interrupt Enable Bits The AD0IDEq bit enables or disables an AD0INq interrupt conversion end interrupt (AD0IDq) to be generated. To prevent incorrect operation, the ADITACT bit in AD0REF register must be "0" while the AD0IDEq bit is changed. If the AD0IDEq bit is "1" when the AD0TFq bit in the interrupt conversion end flag register is set to "1" upon completion of AD0INq interrupt conversion, the AD0IDq signal is generated. The AD0IDq signal can be cleared by clearing AD0TFq or AD0IDEq to "0". See table 27.3 for correspondence between AD0IDEq, AD0INq, and AD0IDq. 0: Disables an interrupt request upon completion of AD0INq interrupt conversion (AD0IDq) or a DMA transfer request. 1: Enables an interrupt request upon completion of AD0INq interrupt conversion (AD0IDq) or a DMA transfer request.

Note: • Set "0" to AD0IDEm (m = 1, 3, 5, 7).

Legend: p = 4 to 14, q = 0 to 3, 15

(2) A/D1 Interrupt Trigger Conversion End Interrupt Enable Register (AD1TRD)

A/D1 Interrupt Trigger Conversion End Interrupt Enable Register (AD1TRD) <P4 address: location H'FFFF 4418>

Bit:	7	6	5	4	3	2	1	0
	AD1IDE7	AD1IDE6	AD1IDE5	AD1IDE4	AD1IDE3	AD1IDE2	AD1IDE1	AD1IDE0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AD1IDE7 to AD1IDE0	All 0	R	W	Interrupt Conversion End Interrupt Enable Bits The AD1IDEn bit enables or disables an AD1INn interrupt conversion end interrupt (AD1IDn) to be generated. To prevent incorrect operation, the ADITACT bit in AD1REF1 must be "0" while the AD1IDEn bit is changed. If the AD1IDEn bit is "1" when the AD1TFn bit in the interrupt conversion end flag register is set to "1" upon completion of AD1INn interrupt conversion, the AD1IDn signal is generated. The AD1IDn signal can be cleared by clearing AD1TFn or AD1IDEn to "0". See table 27.3 for correspondence between AD1IDEn, AD1INn, and AD1IDn. 0: Disables an interrupt request upon completion of AD1INn interrupt conversion (AD1IDn). 1: Enables an interrupt request upon completion of AD1INn interrupt conversion (AD1IDn).

Note: • Set "0" to AD0IDEn (n = 2, 3, 6, 7).

Legend: n = 0 to 7

27.4.13 Interface with CPU

The AD0DRm and AD1DRn registers are a 16-bit register that is connected to the CPU via the 32-bit peripheral bus. The AD0DRm and AD1DRn registers must be read in units of words (16 bits). If the A/D data register is read in byte (8 bits) units by dividing a word into upper and lower bytes and performing read operations twice on it, the A/D converted value read in the first read operation and that read in the second read operation may change. To avoid this error, the A/D data register should not be read in byte (8 bits) units.

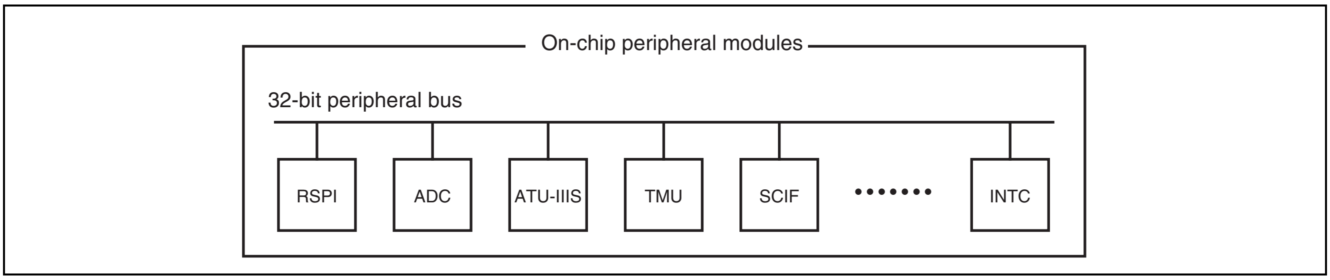


Figure 27.3 Interface between CPU and A/D Converter (ADC)

27.5 Operation

27.5.1 Scan Conversion

A scan conversion is performed in two operating modes: single-cycle scan mode and continuous scan mode. In single-cycle scan mode, one or more specified channels are scanned once. In continuous scan mode, one or more specified channels are scanned until the ADST bit is cleared to "0" (changed from "1" to "0") by software.

Single-cycle scan mode is selected by clearing the ADCS bit in the ADiCSR register to "0", while continuous scan mode is selected by setting the ADCS bit to "1". Scan conversion proceeds in order from lower-numbered to higher-numbered channels: AD0INn (n = 0, 2, 4, 6, and 8 to 15) for AD0 and AD1INn (n = 0, 1, 4, and 5) for AD1.

In single-cycle scan mode, after scanning all selected channels once, the A/D converter sets the ADF bit in the ADiREF register to "1" and then clears the ADSCACT bit in ADiREF to "0" to completes the scan conversion. In continuous scan mode, after scanning all selected channels once, the A/D converter sets the ADF bit to "1" and then continues to scanning. The ADF bit is set to "1" each time A/D conversion (scanning) on all the specified channels is completed.

To stop the scanning, write "0" to the ADST bit when it is "1". Writing "0" to the ADST bit when it is "0" does not affect the A/D converter. Similarly, writing "1" to the ADST bit when it is "1" does not affect the A/D converter. Therefore, to stop a scan conversion started by a request other than the ADST bit, first write "1" to the ADST bit and then write "0" to it.

When the ADF bit is set to "1" while the ADIE bit in the ADiCSR register is set to "1", an ADI interrupt request is generated. To clear the ADF bit to "0", write "0" to the ADF bit after reading it as "1". When the DMAC is started by an ADI interrupt, the ADF bit is automatically cleared to "0" and the ADI interrupt is also cleared.

27.5.2 Single-Cycle Scan Conversion Mode

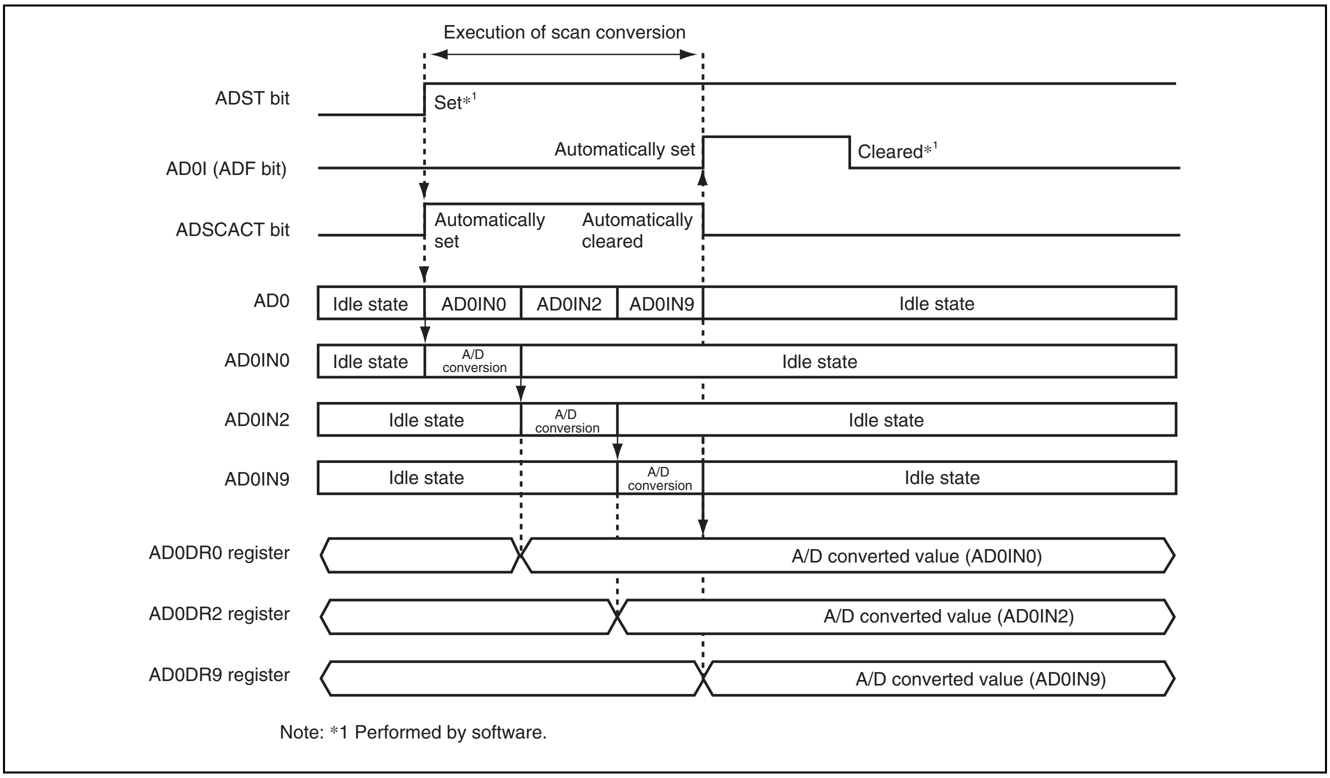
The following is an example operation of single-scan conversion where three channels AD0IN0, AD0IN2, and AD0IN9 are selected and an ADI0 interrupt is enabled. The same operations can also apply to AD1.

1. Clear the ADCS bit in the AD0CSR register to "0" and set the ADIE bit in the AD0CSR register to "1".
2. Set bits AD0ANS0, AD0ANS2, and AD0ANS9 in the AD0ANS register.
3. Set the ADST bit in the AD0CSR register to "1" to start scan conversion. If the ADST bit is already set to "1", write "1" to it after clearing it to "0".

In this case, write "1" to the ADST bit after the interval of a specified period*[1] or more.

Note: *1 When CKS = "0": 2Pck clock
When CKS = "1": 4Pck clock

4. Starting the scan conversion sets the ADSCACT bit to "1". Then, the A/D conversion on channel AD0IN0 is started. On completion of the A/D conversion, the A/D converted value is transferred to the AD0DR0 register. After that, channels AD0IN2 and AD0IN9 are scanned in the order in the same way as in AD0IN0.
5. When the A/D converted values of all the selected channels (AD0IN0, AD0IN2, and AD0IN9) have been transferred to the AD0DR0, AD0DR2, and AD0DR9 registers, the ADF bit is set to "1". At this time, an AD0I interrupt is generated since the ADIE bit is set to "1". The ADSCACT bit is cleared to "0" and the scan conversion is completed.
6. Next, the AD0I interrupt handler is started. In the interrupt handler, clear the AD0I bit by writing "0" to the ADF bit after reading it as "1". After that, read the contents of AD0DR0, AD0DR2, and AD0DR9 registers.
7. Complete the AD0I interrupt handler.



Note: *1 Performed by software.

Figure 27.4 Example Operation in Single-Cycle Scan Mode

27.5.3 Continuous Scan Conversion Mode

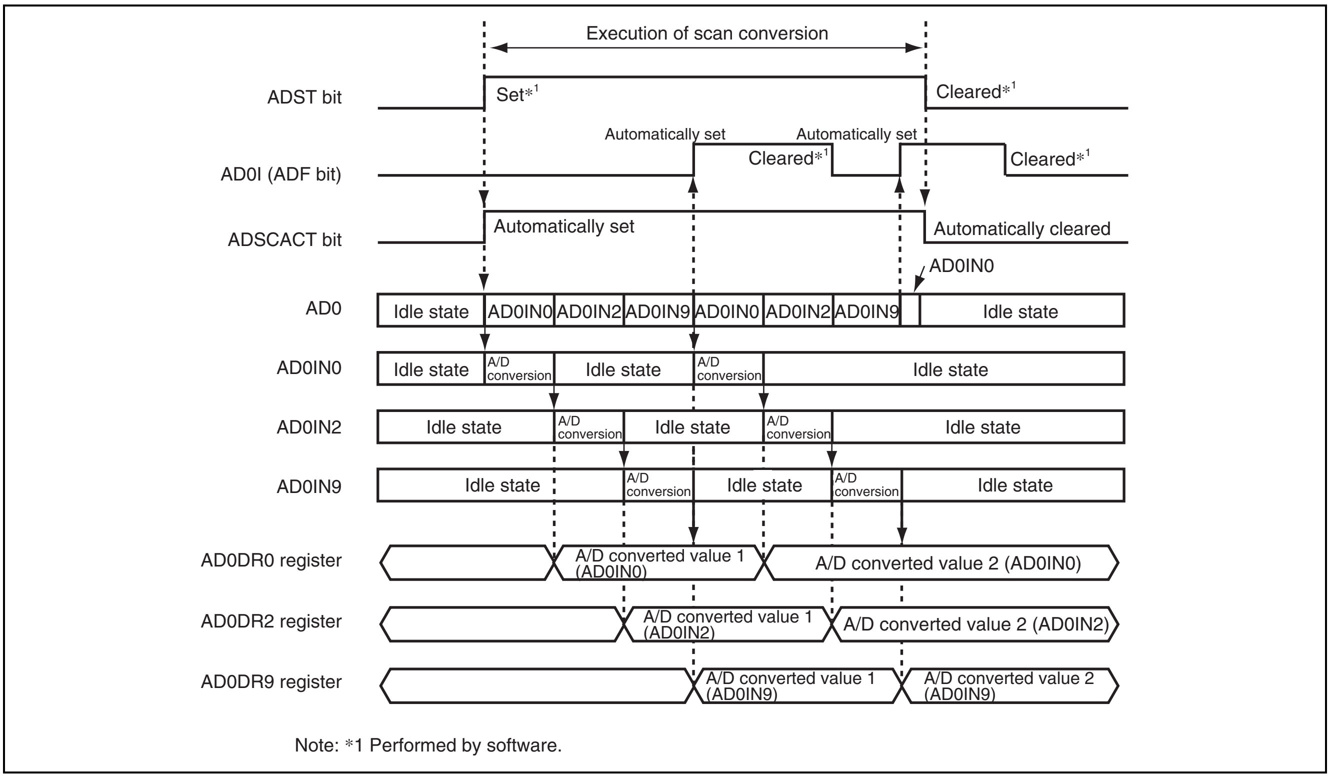
The following is an example operation of continuous scan conversion where three channels AD0IN0, AD0IN2, and AD0IN9 are selected and an ADI0 interrupt is enabled. The same operations can also apply to AD1.

1. Set the ADCS bit and ADIE bit in the AD0CSR register to "1".
2. Set bits AD0ANS0, AD0ANS2, and AD0ANS9 in the AD0ANS register to "1".
3. Set the ADST bit in AD0CSR register to "1" to start scan conversion. If the ADST bit is already "1", write "1" to it after clearing it to "0".

In this case, write "1" to the ADST bit after the interval of a specified period*[1] or more.

Note: *1 When CKS = "0": 2Pck clock
When CKS = "1": 4Pck clock

4. Starting the scan conversion sets the ADSCACT bit to "1". Then, the A/D conversion on channel AD0IN0 is started. On completion of the A/D conversion, the A/D converted value is transferred to the AD0DR0 register. After that, channels AD0IN2 and AD0IN9 are scanned in the order in the same way as in AD0IN0.
5. When the A/D converted values of all the selected channels (AD0IN0, AD0IN2, and AD0IN9) have been transferred to the AD0DR0, AD0DR2, and AD0DR9 registers, the ADF bit is set to "1". At this time, an AD0I interrupt is generated since the ADIE bit is set to "1". Also, the scan conversion returns to the start.
6. The AD0I interrupt handler is started simultaneously. In the interrupt handler, clear the AD0I bit by writing "0" to the ADF bit after reading it as "1". After that, read the contents of the AD0DR0, AD0DR2, and AD0DR9 registers.
7. Complete the AD0I interrupt handler.
8. Steps 4 to 7 are repeated as long as the ADST bit is "1". Clearing the ADST bit to "0" clears the ADSCACT bit to "0", and completes the scan conversion. Setting the ADST bit to "1" initiates scan conversion.



Note: *1 Performed by software.

Figure 27.5 Example Operation in Continuous Scan Mode

27.5.4 Interrupt Conversion

When an ATU-IIIS timer trigger or software trigger is requested on channels AD0INm and AD1INn, A/D conversion is performed on the requested channels. In scan conversion all selected channels are converted when a request is received, but in interrupt conversion channels are converted one at a time in response to individual requests.

To perform interrupt conversion, set the AD0TRGEm and AD1TRGEn bits in the ADiTRE register to "1" and select the trigger source by the AD0TRSm and AD1TRSn bits in the ADiTRS register. When interrupt conversion is requested by the selected trigger source, A/D conversion is performed on the corresponding AD0INm and AD1INn channels. On completion of the interrupt conversion on the AD0INm and AD1INn channels, the AD0TFm and AD1TFn bits in the ADiTRF register is set to "1". The AD0TFm and AD1TFn bits is set to "1" each time an interrupt conversion is performed on AD0INm and AD1INn channels. Furthermore, if any interrupt conversion is performed, the ADITACT bit in the ADiREF register is set to "1". When A/D conversion has been completed on all AD0INm and AD1INn channels to which interrupt conversion is requested, the ADITACT bit is cleared to "0".

When interrupt conversion requests conflict, A/D conversion is performed according to the priority. AD0 is prioritized as AD0IN0 > AD0IN2 > … AD0IN14 > AD0IN15, that is, the lower channel number corresponds to the higher priority. AD1 is prioritized as AD1IN0 > AD1IN1 > AD1IN4 > AD1IN5, thus channel AD1IN0 is given the highest priority, and AD1IN5 the lowest. When interrupt conversion is requested on another channel (AD0INj or AD0INk) during the interrupt conversion on channel AD0INi, the A/D conversion is not interrupted during the conversion regardless of the priority. In this case, on completion of the A/D conversion on channel AD0INi, A/D conversion is performed according to the priority on remaining channels (in this case, AD0INj and AD0INk) in which interrupt conversion requests are pending. Therefore, the priority on interrupt conversion determines which channel is to be converted for the next operation. When a single trigger source generates interrupt conversion requests on two channels or multiple trigger sources simultaneously generate interrupt conversion requests, A/D conversion is performed according to this priority.

When interrupt conversion is requested during scan conversion, the scan conversion on channel AD0INi is suspended, and A/D conversion on the other channel (AD0ANj) in which the interrupt conversion was requested is performed. On completion of the interrupt conversion on channel AD0ANj, the scan conversion is resumed from the interrupted channel (AD0INi). This scheme ensures that the length of time required from the initiation of an interrupt conversion request to the completion of it is always constant. This makes it possible, for example, to perform A/D conversion in pin-point accuracy by synchronizing them with the operation of A/D conversion sources that are external to the MCU.

When the AD0TFm and AD1TFn bits is set to "1" while the AD0IDEm and AD1IDEn bits in the ADiTRD register is set to "1", AD0IDm and AD1IDn interrupts are requested. To clear the AD0TFm and AD1TFn bits to "0", write "0" to the AD0TFm and AD1TFn bits after reading it as "1". If the DMAC is started by AD0IDm and AD1IDn interrupts, note that the AD0TFm and AD1TFn bits are automatically cleared to "0" and AD0IDm and AD1IDn interrupts are also cleared. The DMA transfer of the DMAC is supported on channels AD0IN0 (AD0ID0), AD0IN2 (AD0ID2), and AD0IN15 (AD0ID15).

27.5.5 Example Operation of Interrupt Conversion

The following is an example of interrupt conversion operation where timer TOU00 is selected as the trigger source for channel AD1IN0 and timer TOU01 is selected as the trigger source for channels AD1IN4 and AD1IN5.

1. Set bits AD1TRGE0, AD1TRGE4, and AD1TRE register to "1".
2. Clear the ITTRGS bit in the AD1CER register to "0". Clear bits AD1TRS0, AD1TRS4, and AD1TRS5 in the AD1TRS register to "0".
3. Subsequently, interrupt conversion requests are generated by timers TOU00 and TOU01 at intervals specified by the ATU-IIIS registers. For details on the ATU-IIIS registers, see section 21, Advanced Timer Unit IIIS (ATU-IIIS).
4. When interrupt conversion is requested by timer TOU00, the ADITACT bit in the AD1REF register is set to "1" and interrupt conversion on channel AD1IN0 is performed. On completion of A/D conversion on AD1IN0, the A/D converted value of AD1IN0 is transferred to the AD1DR0 and the AD1TF0 bit in the AD1TRF register is set to "1". The ADITACT bit is cleared to "0" and the interrupt conversion is completed. Furthermore, if the AD1IDE0 bit in the AD1TRD register is "1", an AD1ID0 interrupt is requested to the CPU.
5. When interrupt conversion is requested by timer TOU01, the ADITACT bit in the AD1REF register is set to "1", and interrupt conversion on channels AD1IN4 and AD1IN5 is performed. Then A/D conversion on channel AD1IN4 is performed. On completion of the conversion, the A/D converted value of AD1IN4 is transferred to the AD1DR4 register, and the AD1TF4 bit in the AD1TRF register is set to "1". An A/D conversion on channel AD1IN5 is then performed. On completion of the conversion, the A/D converted value of AD1IN5 is transferred to the AD1DR5 register, and the AD1TF5 bit in the AD1TRF register is set to "1". The ADITACT bit is cleared to "0" and the interrupt conversion is completed. Further, if bits AD1IDE4 and AD1IDE5 in the AD1TRD register are set to "1" when either the AD1TF4 or AD1TF5 bit is set to "1", the A/D converter requests an AD1ID4 or an AD1ID5 interrupt to the CPU.
6. Subsequently, steps 4 to 5 are repeated. The following is an example operation when requests by timers TOU00 and TOU01 conflict.

(1) Example Operation 1

When a timer TOU00 interrupt conversion request is input during the A/D conversion on channel AD1IN4 due to a timer TOU01 interrupt conversion request, the request is processed as follows.

The timer TOU00 interrupt source is retained in the A/D converter until conversion on channel AD1IN4 completes. When A/D conversion on channel AD1IN4 finishes, A/D conversion is performed on channels AD1IN0, AD1IN5, in that order, according to their priority.

(2) Example Operation 2

When interrupt conversion requests by timers TOU00 and TOU01 are input simultaneously, the requests are processed as follows.

The timer TOU00 and timer TOU01 interrupt sources are retained in the A/D converter. Then A/D conversion is performed on channels AD1IN0, AD1IN4, and AD1IN5, in that order, according to their priority.

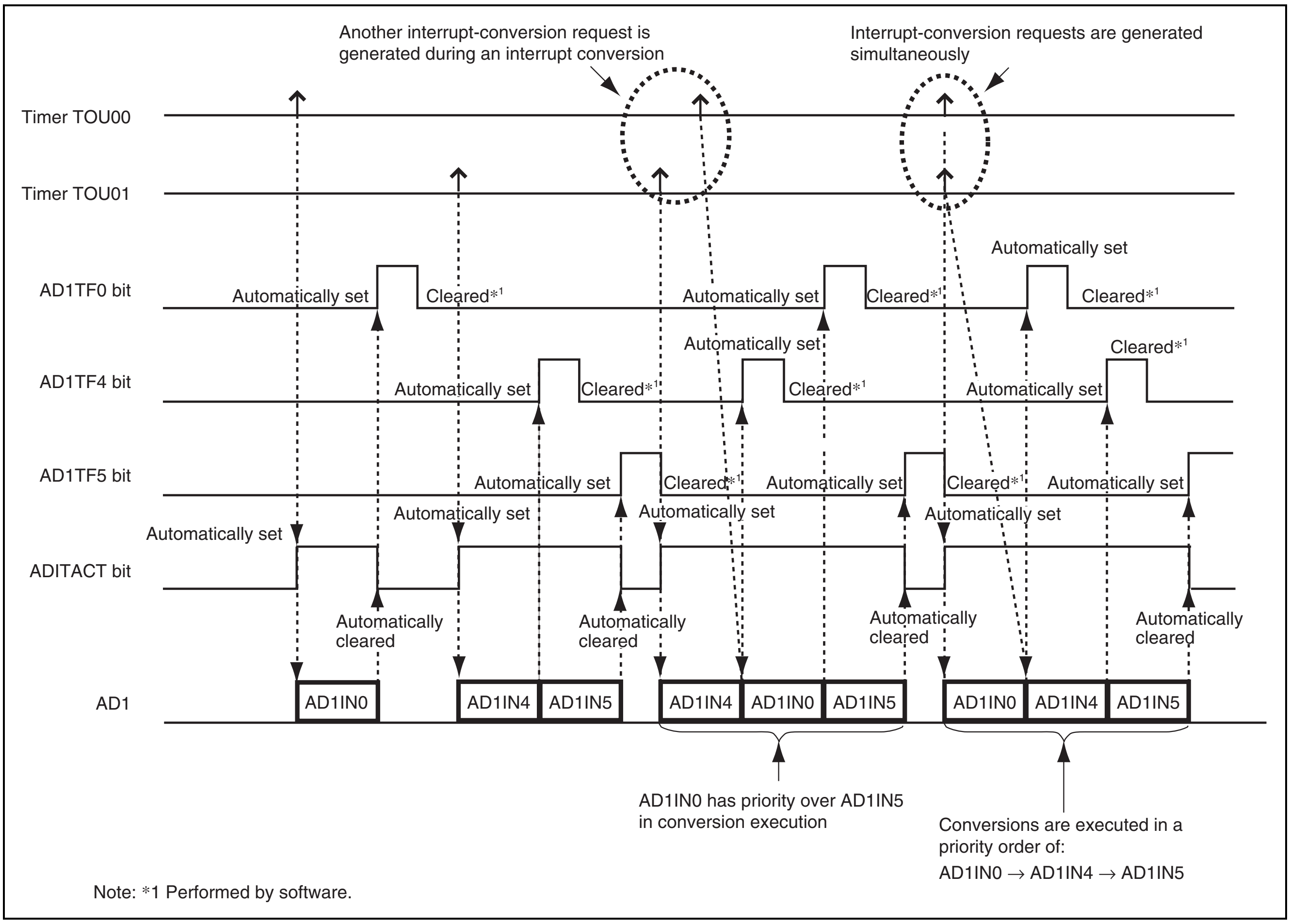

Another interrupt-conversion request is generated during an interrupt conversion
Interrupt-conversion requests are generated simultaneously
Timer TOU00
Timer TOU01
AD1TF0 bit
AD1TF4 bit
AD1TF5 bit
ADITACT bit
AD1
Automatically set
Cleared*1
Automatically cleared
AD1IN0
AD1IN4
AD1IN5
AD1IN0 has priority over AD1IN5 in conversion execution
Conversions are executed in a priority order of:
AD1IN0 → AD1IN4 → AD1IN5
Note: *1 Performed by software.

Figure 27.6 Example Operation of Interrupt Conversion

27.5.6 Interrupt Conversion during Scan Conversion

The following is an example operation where single scan conversion on three channels AD0IN0, AD0IN2, and AD0IN9 is started by a scan conversion request from timer G4 and then an interrupt conversion on channel AD0IN6 is started by a interrupt conversion request from timer TOU02.

1. Clear the ADCS and EXTRG bits in the AD0CSR register to "0", and set the TRGE bit in AD0CSR to "1".
2. Set the AD0ANS0 bits, AD0ANS2 and AD0ANS9 bits in the AD0ANS register to "1".
3. Set the AD0TRGE6 bit in the AD0TRE register to "1".
4. Clear the ITTRGS bit in the AD0CER register to "0". Clear the AD0TRS6 bit in the AD0TRS register to "0".
5. Subsequently, a scan conversion request is generated by timer G4 and an interrupt conversion request is generated by timer TOU02 at intervals specified by the ATU-IIIS registers. For details on the ATU-IIIS registers, see section 21, Advanced Timer Unit IIIS (ATU-IIIS).
6. When scan conversion is requested by timer G4, the ADSCACT bit is set to "1". Then, A/D conversion on channels AD0IN0, AD0IN2, and AD0IN9 is performed in the order. On completion of the conversion, the ADF bit is set to "1" and the ADSCACT bit is cleared to "0", indicating that the scan conversion is completed.
7. When interrupt conversion is requested by timer TOU02, the ADITACT bit is set to "1" and interrupt conversion on channel AD0IN6 is performed. On completion of the A/D conversion on channel AD0IN6, the AD0TF6 bit in the AD0TRF register is set to "1" and the ADITACT bit is cleared to "0", indicating that the interrupt conversion is completed.
8. Subsequently, steps 6 to 7 are repeated. The following is an example operation where a scan conversion and an interrupt conversion conflict.

(1) Example Operation

When a timer TOU02 interrupt conversion request is input during the A/D conversion on channel AD0IN2 in the scan conversion due to a timer G4 scan conversion request, the request is processed as follows.

The timer TOU02 interrupt source is retained in the A/D converter, and the scan conversion on channel AD0IN2 is suspended. The priority is applied to channels AD0IN2 and AD0IN9 on which scan conversion is pending, and is applied to AD0IN6 which is the current request. In this case, the A/D conversion on channels AD0IN6, AD0IN2, and AD0IN9 in the order.

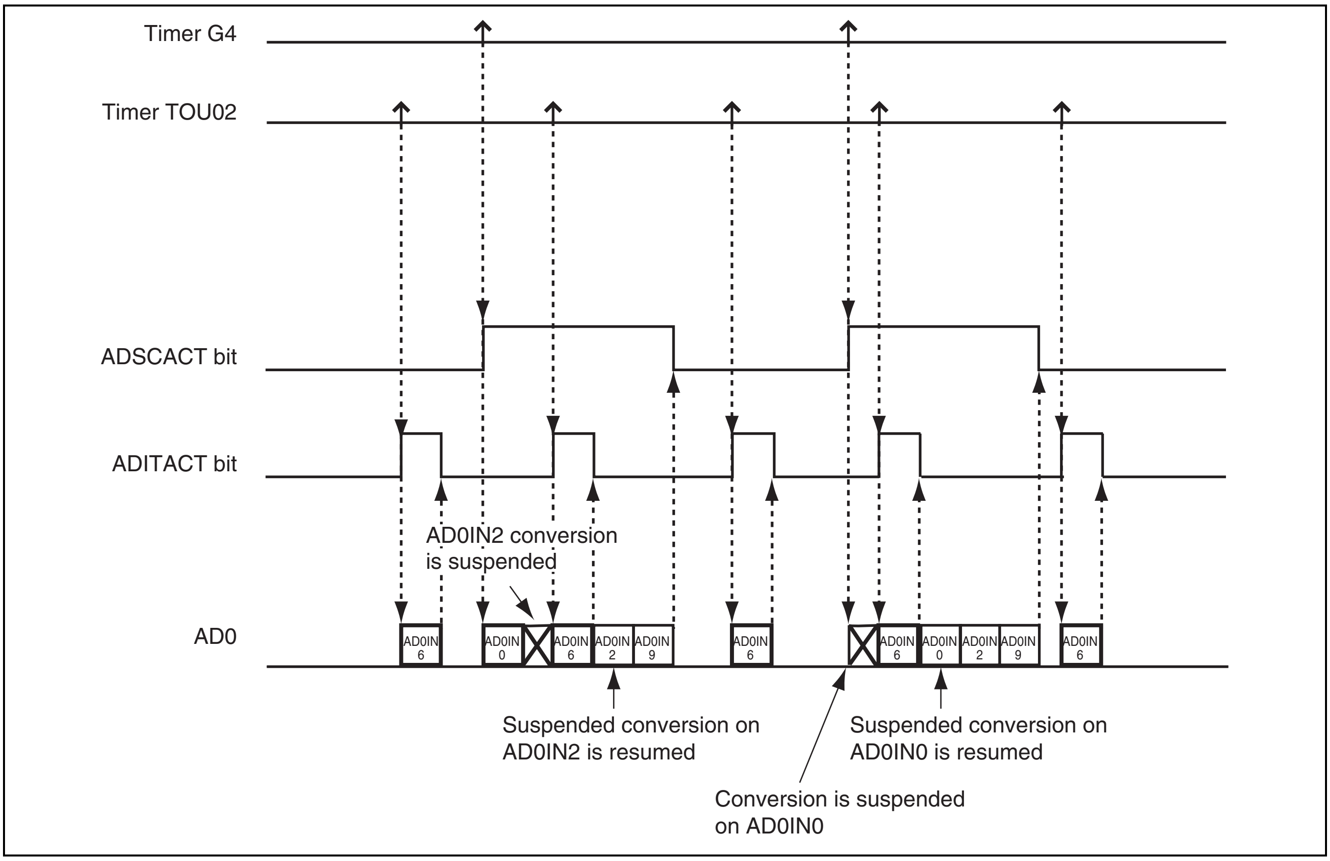

Timer G4
Timer TOU02
ADSCACT bit
ADITACT bit
AD0
AD0IN2 conversion is suspended
AD0IN 6
AD0IN 0
AD0IN 6
AD0IN 2
AD0IN 9
AD0IN 6
AD0IN 6
AD0IN 0
AD0IN 2
AD0IN 9
AD0IN 6
Suspended conversion on AD0IN2 is resumed
Suspended conversion on AD0IN0 is resumed
Conversion is suspended on AD0IN0

Figure 27.7 Operation Example of Interrupt Conversion during Scan Conversion

27.5.7 Analog Input Sampling and Scan Conversion Time

The A/D converter includes the sample and hold circuit. When start-of-scan-conversion delay time (t_D) have passed after the ADST bit in the ADiCSR register is set to "1", the A/D converter samples the analog input, and then begins the conversion process.

Figure 27.8 shows a timing chart for a scan conversion on one channel in single-cycle scan mode. Scan conversion time (t_{SCAN}) includes start-of-scan-conversion delay time (t_D), analog input sampling time (t_{SPL}), A/D conversion processing time (t_{CONV}) and end-of-scan-conversion delay time (t_{ED}). The scan conversion time is shown in table 27.6.

The scan conversion time (t_{SCAN}) in single-cycle scan mode for which the number of selected channels is n can be determined according to the following equation:

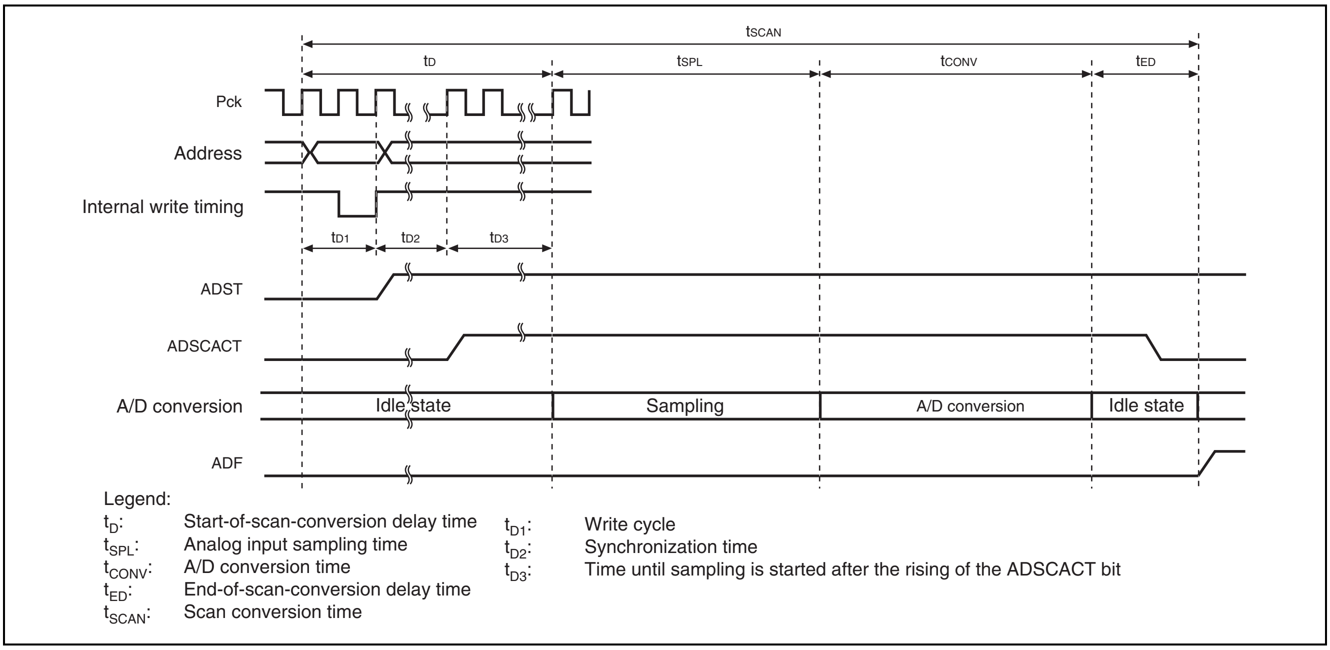
$$t_{SCAN} = t_D + \{(t_{SPL} + t_{CONV}) \times n\} + t_{ED}$$

The scan conversion time for the first cycle in continuous scan mode is t_{SCAN} for single-cycle scan minus t_{ED}.

The scan conversion time for the second and subsequent cycles in continuous scan mode is a fixed time, which is equal to $\{(t_{SPL} + t_{CONV}) \times n\}$.

Table 27.6 Scan Conversion Time

		Pck = 40 MHz (Pck conversion)		
Item	**Symbol**	**CKS = "0"**	**CKS = "1"**	**Unit**
Start-of-scan-conversion delay time	t_D	7	11, 12	State
Write cycle	t_{D1}	2	2	
Synchronization time	t_{D2}	2	3, 4	
Time until sampling is started after the rising of the ADSCACT bit	t_{D3}	3	6	
Analog input sampling time	t_{SPL}	20	40	
A/D conversion processing time	t_{CONV}	30	60	
End-of-scan-conversion delay time	t_{ED}	4	7	
Scan conversion time	t_{SCAN}	61	118, 119	



Legend:

t_D:	Start-of-scan-conversion delay time	t_{D1}:	Write cycle
t_{SPL}:	Analog input sampling time	t_{D2}:	Synchronization time
t_{CONV}:	A/D conversion time	t_{D3}:	Time until sampling is started after the rising of the ADSCACT bit
t_{ED}:	End-of-scan-conversion delay time		
t_{SCAN}:	Scan conversion time		

Figure 27.8   Timing Diagram for Scan Conversion (Single Channel, Single Cycle)

27.5.8   Starting Scan Conversion with External Trigger

The A/D converter can be activated by the input of an external trigger. To start up the A/D converter by an external trigger, the pin function should be set up using the pin function unit. After applying a "H" level signal to the AD0TRG# pin, both the TRGE and EXTRG bits in the A/D0 control register (AD0CSR) should be set to "1". If a "L" level signal is then input to the AD0TRG# pin, the A/D converter detects a pulse fall edge and sets the ADSCACT bit to "1".

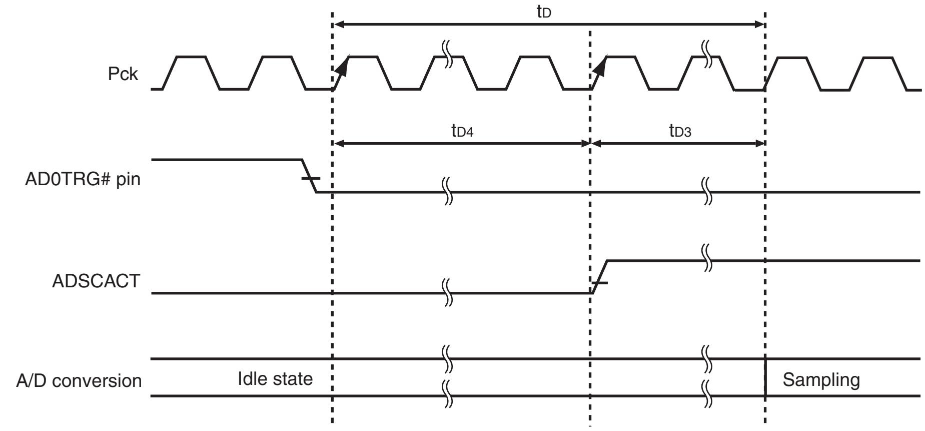
Figure 27.9 shows an external of trigger input timing. Table 27.7 lists scan conversion time for external trigger input.

The timing at which a scan conversion is started after the ADSCACT bit is set to "1" is the same as the case where the ADST bit is set to "1" from "0" by software. For details on pin function setting, see section 18, I/O Ports and Pin Function Unit.

To stop the scan conversion process while it is in progress, write "1" to the ADST bit and then write "0" to it.

Table 27.7   Scan Conversion Time for External Trigger Input

		Pck = 40 MHz (Pck conversion)		
Item	**Symbol**	**CKS = "0"**	**CKS = "1"**	**Unit**
Start-of-scan-conversion delay time	t_D	8	13, 14	State
Time until the rising of the ADSCACT bit after the falling of the AD0TRG# pin is sampled	t_{D4}	5	7, 8	
Time until sampling is started after the falling of the ADSCACT bit	t_{D3}	3	6	
Analog input sampling time	t_{SPL}	20	40	
A/D conversion processing time	t_{CONV}	30	60	
End-of-scan-conversion delay time	t_{ED}	4	7	
Scan conversion time	t_{SCAN}	62	120, 121	



Legend:
t_D: Start-of-scan-conversion delay time
t_{D3}: Time until sampling is started after the rising of the ADSCACT bit
t_{D4}: Time until the rising of the ADSCACT bit after the falling of AD0TRG# pin is sampled

Figure 27.9 External Trigger Input Timing

27.5.9 Starting Scan Conversion with ATU-IIIS Timer Trigger

A scan conversion can be activated by an ATU-IIIS timer trigger. To start up a scan conversion by an ATU-IIIS timer trigger, the TRGE bit in the ADiCSR register is set to "1", and the EXTRG bit is set to "0". If a timer trigger (timer G4 or timer G5) is entered in this situation, the ADSCACT bit is set to "1". The timing at which a scan conversion is started after the ADSCACT bit is set to "1" is the same as the case where the ADST bit is set to "1" from "0" by software.

To stop the scan conversion process while it is in progress, write "1" to the ADST bit and then write "0" to it.

27.5.10 Monitoring via AD0END Pin

The timing at which AD0IN0 is scan-converted can be monitored via the AD0END pin, respectively. For details on pin function setting, see section 18, I/O Ports and Pin Function Unit.

Figure 27.10 shows AD0END output example. If AD0END output is selected by the pin function unit, monitor signals are output, respectively, from the AD0END output pin during the conversion processing of channel AD0IN0. Upon completion of the AD0IN0 sampling, output is produced from the AD0END pin, respectively.

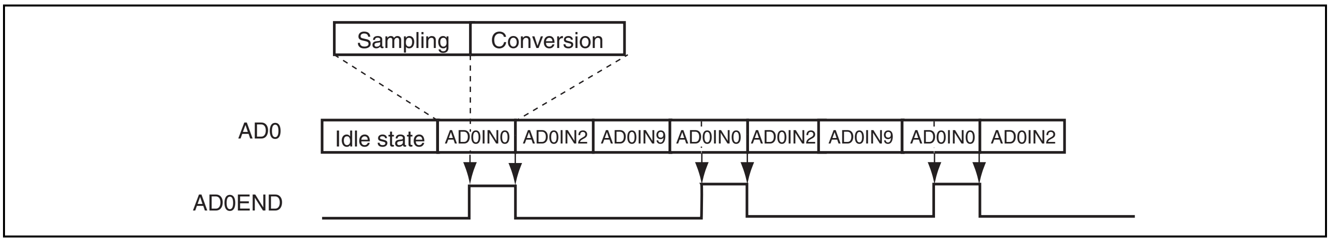


Figure 27.10 Example of AD0END Outputs

Note: • If an interrupt conversion is performed while "H" level signal is output from the AD0END pin, "L" level signals are output once from this pin. After that, because channel AD0IN0 is converted again in a scan conversion process, "H" level signal is output from the AD0END pin again. In addition, if channel AD0IN0 is converted in an interrupt conversion process, "H" level signals are also output from the AD0END pin. Further, if channel AD0IN0 is set to A/D-converted value addition mode, "H" level signal is output from the AD0END pin only during the final A/D conversion (in the 4th conversion if four-addition conversion is performed, for example).

27.6 Interrupt Sources and DMA Transfer Request

27.6.1 Interrupt Requests on Completion of Scan Conversion

The A/D converter can generate a scan conversion end interrupt request (ADI) to the CPU. By setting the ADIE bit in the ADiCSR register to "1", an ADI interrupt is enabled; by clearing the bit to "0", an ADI interrupt is disabled. In addition, the DMAC can be started up when an ADI interrupt is generated. In this case, interrupts are not generated to the CPU. If the DMAC is started upon an ADI interrupt, the ADF bit in the ADiREF register is automatically cleared to "0" when the data transfer request is accepted by the DMAC.

For details on DMAC settings, see section 20, Direct Memory Access Controller (DMAC).

Note: • The ADF bit is not cleared by an interrupt request to the CPU.

27.6.2 Interrupt Requests on Completion of Interrupt Conversion

The A/D converter can issue interrupt conversion end interrupt requests (AD0IDm and AD1IDn) to the CPU upon completion of an interrupt conversion. By setting the AD0IDEm and AD1IDEn bits in the ADiTRD register to "1", the AD0IDm and AD1IDn interrupts are enabled, respectively; by clearing the bits to "0", the AD0IDm and AD1IDn interrupts are disabled, respectively. If the DMAC is activated by AD0ID0, AD0ID2, or AD0ID15, the corresponding bit (AD0ID0, AD0ID2, or AD0ID15) in the AD0TRF register is automatically cleared to "0" when the DMAC accepts the data transfer request.

For details on DMAC settings, see section 20, Direct Memory Access Controller (DMAC).

Note: • The ADiTF bit is not cleared by an interrupt request to the CPU.

27.7 Definition of A/D Conversion Accuracy

The definition of A/D conversion accuracy is described below.

- Resolution

  This indicates the number of digital output codes in the A/D converter
- Quantization error

  This error, which is inherent to the A/D converter, is given as 1/2LSB (figure 27.11).
- Offset error

  This error, which is exclusive of quantization error, is a deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from a minimum voltage value B'000000000000 to B'000000000001 (figure 27.11).
- Full scale error

  This error, which is exclusive of quantization error, is a deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from B'111111111110 to B'111111111111 (figure 27.11).
- Nonlinearity error

  This error, which is exclusive of offset error, full scale error and quantization error, is a deviation from the ideal A/D conversion characteristics through the zero-scale and full-scale transitions (figure 27.11).
- Absolute accuracy

  This is a deviation of the digital value from the analog input value. This includes offset error, full scale error, quantization error, and nonlinearity error.

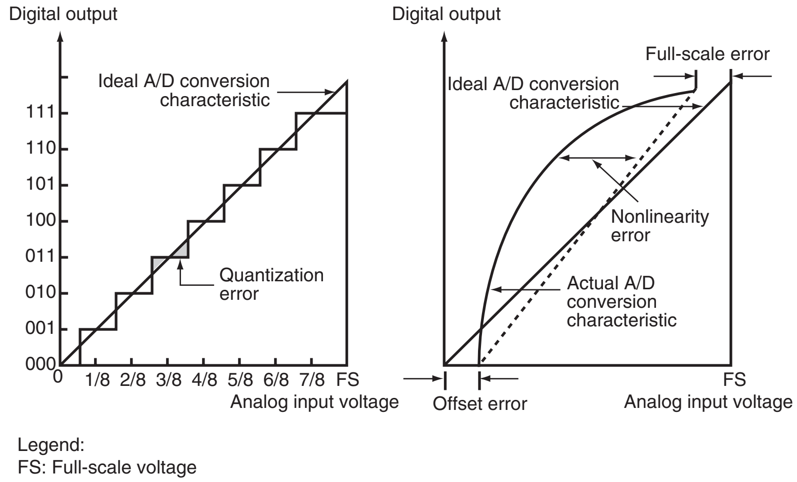


Figure 27.11 Definition of A/D Conversion Accuracy

27.8 Usage Notes

27.8.1 Analog Input Voltage Range

The voltage applied to the analog input pin (AD0INm and AD1INn) during A/D conversion should be within a range of AVREFL ≤ AD0INm (m = 0, 2, 4, 6, and 8 to 15)/AD1INn (n = 0, 1, 4, and 5) ≤ AVREFH.

27.8.2 Relationship among AV_{CC}, AV_{SS}, V_{CC}, and V_{SS}

When using the A/D converter, make sure that the following relationships are held among AV_{CC}, AV_{SS}, V_{CC} and V_{SS}:

AVcc ≥ Vcc, $AV_{SS} = V_{SS}$

When the A/D converter is not used, AV_{CC} pin must not be open. In this case, the following relationship should be held between AV_{SS} and V_{SS}:

$AV_{SS} = V_{SS}$

27.8.3 Allowable Settings for Pins AVREFH and AVREFL

The allowable settings for the AVREFH pin are as follows:

When the A/D converter is used: AVREFH = 4.5 V to AVcc when AVcc = 5.0 V

AVREFH = 3.0 V to AVcc when AVcc = 3.3 V

When the A/D converter is not used: AVREFH ≤ AV_{CC}

If any value outside the above range is set, it can adversely affect the reliability of the MCU. For the AVREFL pin, set AVREFL = AV_{SS} = V_{SS}.

27.8.4 Precautions on Board Design

For designing a board, to the maximum extent possible the digital circuits should be laid out separately from the analog circuits. Layouts involving the crossing of signal lines for digital circuits and signal lines for analog circuits, or placing them in proximity to each other, should be avoided. If the dissimilar signal lines are placed in close proximity to each other, the resulting induction can lead to a malfunction of the analog circuits or produce an adverse impact on A/D conversion values.

It should be noted that the analog input pins (AD0INm and AD1INn), the analog reference voltages (AVREFH, AVREFL), and the analog power supply (AV_{CC}) should be isolated from the digital circuits by means of analog grounding (AV_{SS}). In addition, the analog ground (AV_{SS}) should be connected in one point to a stable digital ground (V_{SS}) on the board.

27.8.5 Precautions on Noise Measures

A protection circuit to prevent the analog input pin (AD0INm and AD1INn) from damages, by such abnormal voltages as excessive surges, should be connected between AV_{CC} and AV_{SS}, as illustrated in figure 27.12. Also bypass capacitors connected between AVREFH and AVREFL, or a filter capacitor connected to an analog input pin (AD0INm and AD1INn), should be connected to the AV_{SS}. Since connecting a filter capacitor can cause an error by averaging the input currents to the analog input pins (AD0INm and AD1INn), care must be taken to choose appropriate circuit constants, as shown in figure 27.12.

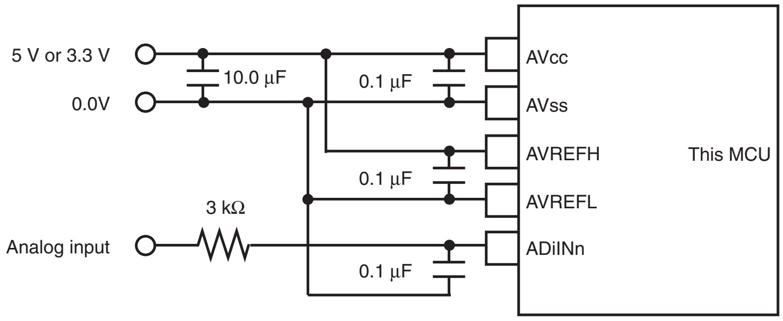


Figure 27.12 Example of Protection Circuit for Analog Input Pin

27.8.6 Notes on Use of Analog Input Pins as Digital Inputs

The pin function unit is used for function selection when using analog input pins as digital inputs, but the following should be kept in mind.

When multiple analog input pins are used as a mixture of analog and digital inputs, to not mix analog and digital inputs within the same module (for example, by using AD0IN0 as an analog input and AD0IN2 as a digital input).

Table 27.8 Use of Analog Input Pins as Digital Inputs

AD0	AD1	Setting
Mixed	Don't care	Prohibited
Don't care	Mixed	Prohibited
Digital only	Analog only	Allowed
Analog only	Digital only	Allowed
Digital only	Digital only	Allowed
Analog only	Analog only	Allowed

This page is blank for reasons of layout.

Section 28 Direct RAM Input Interface (DRI)

28.1 Overview

The direct RAM input interface (DRI) is a parallel interface that acquires parallel data input to this MCU in synchronization with a clock signal and stores that data in SHwyRAM. The DRIi modules acquire data without stopping CPU operation by using a separate dedicated DRI/DRO bus to write data to SHwyRAM. The DRIi modules also selectively acquire data through the decimal control function using an internal event counter. Note that the lower case "i" that appears in DRIi and the register names indicates the numbers 0 to 2. Also, the "j" used in the DINj and DDj pins indicates letters B and C. (For pin specifications, see table 28.3)

Table 28.1 lists the overview of the DRIi modules. Table 28.2 the DRIi module interrupt request and DMA transfer request generation function.

Table 28.1 DRIi Overview

Item	Description
Number of channels	3 channels
Operating frequency	80 MHz (when PAck = 80 MHz)
Transfer method	Clock synchronous parallel input
Access areas	All SHwy RAM areas (up to 256 Kbytes)
Maximum transfer rate	80 Mbytes/second (when the DRIi operating frequency is 80 MHz)
Minimum data acquisition period	The following are the minimum periods when the DRIi operating frequency is 80 MHz. 43.75 ns (special mode disabled and the input data bus width is 8 or 16 bits) 25 ns (special mode enabled)
Data acquisition bus width	8 or 16 bits
Event counter	16 bits × 6 counters (DEC5 to DEC0)
Bank switching function	Two banks can be specified as the data storage destination in SHwyRAM
Data acquisition edges	Either rising edges, falling edges, or both edges can be selected
Acquisition timing adjustment function	Sets the time between detection of the data acquisition edge and the acquisition operation
Decimation control function	Data can be acquired selectively using an event counter (DEC5 to DEC0)

Table 28.2 DRIi Interrupt Request and DMA Transfer Request Sources

Source	Interrupt Request Sourcs	DMA Transfer Request Source*1
DIN0 event detection	DRI event detection interrupt	DRI0 DIN0 event detection
DIN1 event detection		DRI0 DIN1 event detection
DIN2 event detection		DRI0 DIN2 event detection
DIN3 event detection		DRI0 DIN3 event detection
DIN4 event detection		DRI0 DIN4 event detection
DIN5 event detection		DRI0 DIN5 event detection
DEC0 underflow	DRI counter interrupt	DRI0 DEC0 underflow
DEC1 underflow		DRI0 DEC1 underflow
DEC2 underflow		DRI0 DEC2 underflow
DEC3 underflow		DRI0 DEC3 underflow
DEC4 underflow		DRI0 DEC4 underflow
DEC5 underflow		DRI0 DEC5 underflow
DRI address counter 0 transfer complete	DRI transfer complete interrupt	DRI0 DRI address counter 0 transfer complete
DRI address counter 1 transfer complete		DRI0 DRI address counter 1 transfer complete
Overrun error		—
Acquisition enable error		—
DRI acquisition event counter underflow		DRI0 DRI acquisition event counter underflow
DRI transfer counter underflow	—	DRI0 DRI transfer counter underflow

Note: *1 DMA transfers are only possible with DRI0. DRI1 and DRI2 do not support the DMA transfer request generation function.

See section 20, Direct Memory Access Controller (DMAC), for details.

Figure 28.1 shows the block diagram of the DRIi module.

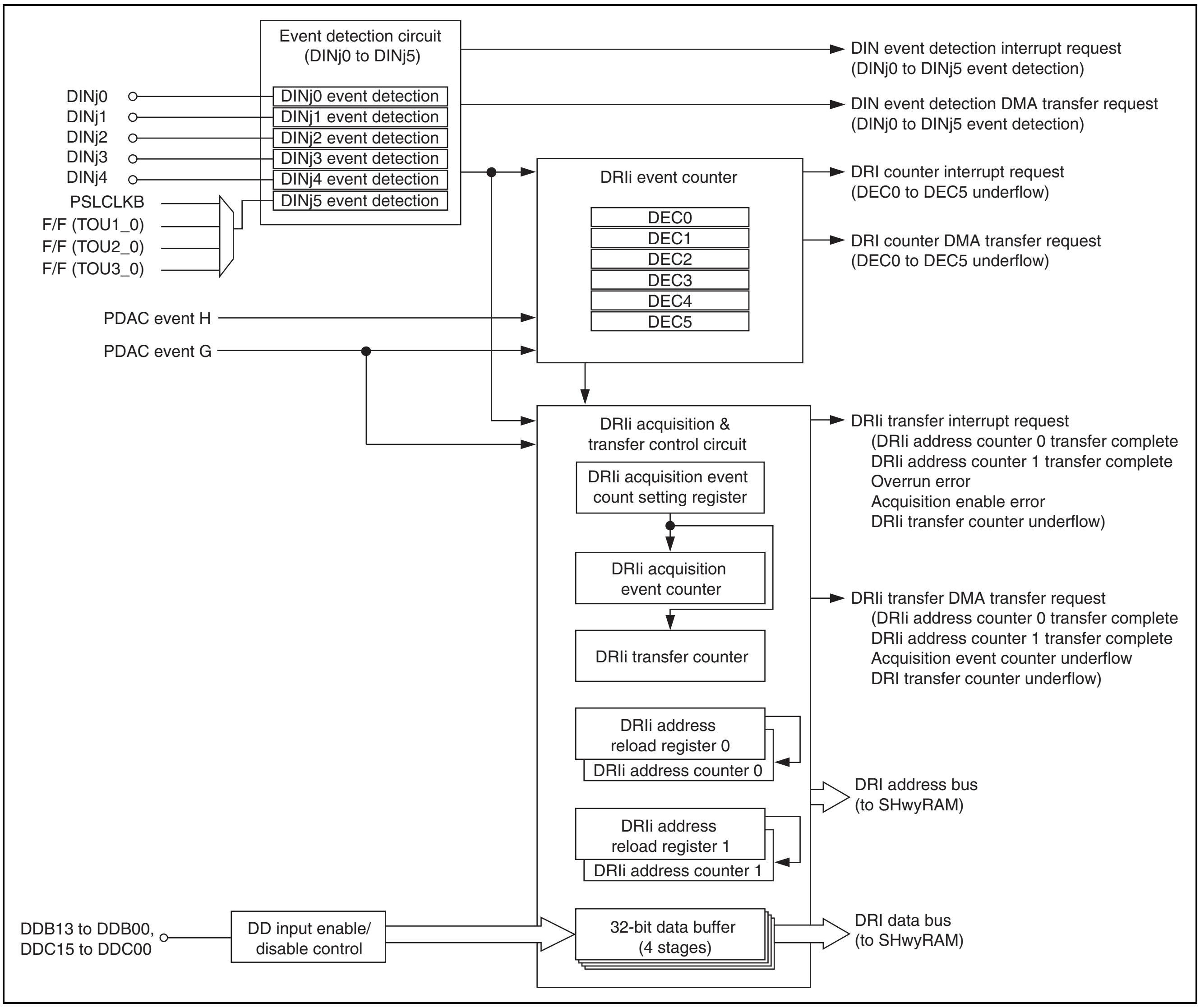


Figure 28.1 Block Diagram of DRIi

28.2 Input/Output Pins

Table 28.3 lists the DRIi pin configuration.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 28.3 Pin Configuration

Pin Name	I/O	Function
DDB13 to DDB00	Input	Input data
DDC15 to DDC00	Input	Input data
DINB3, DINB1, and DINB0	Input	Input event signal
DINC4	Input	Input event signal

In the remainder of this section, unless otherwise specified pin notations refer to the corresponding inputs selected from the pin groups shown above.

28.3 Register Descriptions

Table 28.4 lists the DRIi register configuration.

Table 28.4 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DRI0DIN interrupt request status register	DRI0DINIST	H'00	H'FFBF C000	8	28-12
DRI0DIN interrupt request enable register	DRI0DINIEN	H'00	H'FFBF C001	8	28-13
DRI0DIN DMA transfer request status register	DRI0DINDST	H'00	H'FFBF C002	8	28-14
DRI0DIN DMA transfer enable register	DRI0DINDEN	H'00	H'FFBF C003	8	28-16
DRI0DEC interrupt request status register	DRI0DECIST	H'00	H'FFBF C004	8	28-17
DRI0DEC interrupt request enable register	DRI0DECIEN	H'00	H'FFBF C005	8	28-18
DRI0DEC DMA transfer request status register	DRI0DECDST	H'00	H'FFBF C006	8	28-19
DRI0DEC DMA transfer enable register	DRI0DECDEN	H'00	H'FFBF C007	8	28-21
DRI0 transfer interrupt request status register	DRI0TRMIST	H'00	H'FFBF C008	8	28-22
RI0 transfer interrupt request enable register	DRI0TRMIEN	H'00	H'FFBF C009	8	28-24
DRI0 DMA transfer request status register	DRI0TRMDST	H'00	H'FFBF C00A	8	28-25
DRI0 DMA transfer enable register	DRI0TRMDEN	H'00	H'FFBF C00B	8	28-27
DRI0 transfer control register	DRI0TRMCNT	H'00	H'FFBF C00C	8	28-28
DRI0 special mode register	DRI0SPMOD	H'00	H'FFBF C00D	8	28-31
DRI0 data acquisition control register	DRI0DCAPCNT	H'0000	H'FFBF C00E	16	28-35
DRI0 data decimation control register	DRI0DSELCNT	H'00	H'FFBF C010	8	28-39
RI0 data decimation event selection register	DRI0DEVTCNT	H'00	H'FFBF C011	8	28-40
DRI0DIN input event selection register	DRI0DINSEL	H'00	H'FFBF C012	8	28-41
DRI0DD input enable register	DRI0DDEN	H'0000 0000	H'FFBF C014	32	28-42
DRI0 data acquisition event count setting register	DRI0DCAPNUM	H'0000 0000	H'FFBF C018	32	28-43
DRI0 acquisition event counter	DRI0DCAPCT	H'0000 0000	H'FFBF C01C	32	28-44
DRI0 transfer counter	DRI0TRMCT	H'0000 0000	H'FFBF C020	32	28-45
DRI0 address reload register 0	DRI0ADR0RLD	H'0000 0000	H'FFBF C024	32	28-46
DRI0 address counter 0	DRI0ADR0CT	H'0000 0000	H'FFBF C028	32	28-47
DRI0 address reload register 1	DRI0ADR1RLD	H'0000 0000	H'FFBF C02C	32	28-46
DRI0 address counter 1	DRI0ADR1CT	H'0000 0000	H'FFBF C030	32	28-47
DRI0 input processing control register	DRI0DINCNT	H'0000	H'FFBF C034	16	28-48
DRI0DEC0 control register	DRI0DEC0CNT	H'00	H'FFBF C036	8	28-49
DRI0DEC0 reload register	DRI0DEC0RLD	H'0000	H'FFBF C038	16	28-61
DRI0DEC0 counter	DRI0DEC0CT	H'0000	H'FFBF C03A	16	28-63
DRI0DEC1 control register	DRI0DEC1CNT	H'00	H'FFBF C03C	8	28-51
DRI0DEC1 reload register	DRI0DEC1RLD	H'0000	H'FFBF C03E	16	28-61
DRI0DEC1 counter	DRI0DEC1CT	H'0000	H'FFBF C040	16	28-63
DRI0DEC2 control register	DRI0DEC2CNT	H'00	H'FFBF C042	8	28-53

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DRI0DEC2 reload register	DRI0DEC2RLD	H'0000	H'FFBF C044	16	28-61
DRI0DEC2 counter	DRI0DEC2CT	H'0000	H'FFBF C046	16	28-63
DRI0DEC3 control register	DRI0DEC3CNT	H'00	H'FFBF C048	8	28-55
DRI0DEC3 reload register	DRI0DEC3RLD	H'0000	H'FFBF C04A	16	28-61
DRI0DEC3 counter	DRI0DEC3CT	H'0000	H'FFBF C04C	16	28-63
DRI0DEC4 control register	DRI0DEC4CNT	H'00	H'FFBF C04E	8	28-57
DRI0DEC4 reload register	DRI0DEC4RLD	H'0000	H'FFBF C050	16	28-61
DRI0DEC4 counter	DRI0DEC4CT	H'0000	H'FFBF C052	16	28-63
DRI0DEC5 control register	DRI0DEC5CNT	H'00	H'FFBF C054	8	28-59
DRI0DEC5 reload register	DRI0DEC5RLD	H'0000	H'FFBF C056	16	28-62
DRI0DEC5 counter	DRI0DEC5CT	H'0000	H'FFBF C058	16	28-64
DRI1DIN interrupt request status register	DRI1DINIST	H'00	H'FFBF D000	8	28-12
DRI1DIN interrupt request enable register	DRI1DINIEN	H'00	H'FFBF D001	8	28-13
DRI1DEC interrupt request status register	DRI1DECIST	H'00	H'FFBF D004	8	28-17
DRI1DEC interrupt request enable register	DRI1DECIEN	H'00	H'FFBF D005	8	28-18
DRI1 transfer interrupt request status register	DRI1TRMIST	H'00	H'FFBF D008	8	28-22
DRI1 transfer interrupt request enable register	DRI1TRMIEN	H'00	H'FFBF D009	8	28-24
DRI1 transfer control register	DRI1TRMCNT	H'00	H'FFBF D00C	8	28-28
DRI1 special mode register	DRI1SPMOD	H'00	H'FFBF D00D	8	28-31
DRI1 data acquisition control register	DRI1DCAPCNT	H'0000	H'FFBF D00E	16	28-35
DRI1 data decimation control register	DRI1DSELCNT	H'00	H'FFBF D010	8	28-39
DRI1 data decimation event selection register	DRI1DEVTCNT	H'00	H'FFBF D011	8	28-40
DRI1DIN input event selection register	DRI1DINSEL	H'00	H'FFBF D012	8	28-41
DRI1DD input enable register	DRI1DDEN	H'0000 0000	H'FFBF D014	32	28-42
DRI1 data acquisition event count setting register	DRI1DCAPNUM	H'0000 0000	H'FFBF D018	32	28-43
DRI1 acquisition event counter	DRI1DCAPCT	H'0000 0000	H'FFBF D01C	32	28-44
DRI1 transfer counter	DRI1TRMCT	H'0000 0000	H'FFBF D020	32	28-45
DRI1 address reload register 0	DRI1ADR0RLD	H'0000 0000	H'FFBF D024	32	28-46
DRI1 address counter 0	DRI1ADR0CT	H'0000 0000	H'FFBF D028	32	28-47
DRI1 address reload register 1	DRI1ADR1RLD	H'0000 0000	H'FFBF D02C	32	28-46
DRI1 address counter 1	DRI1ADR1CT	H'0000 0000	H'FFBF D030	32	28-47
DRI1 input processing control register	DRI1DINCNT	H'0000	H'FFBF D034	16	28-48
DRI1DEC0 control register	DRI1DEC0CNT	H'00	H'FFBF D036	8	28-49
DRI1DEC0 reload register	DRI1DEC0RLD	H'0000	H'FFBF D038	16	28-61
DRI1DEC0 counter	DRI1DEC0CT	H'0000	H'FFBF D03A	16	28-63
DRI1DEC1 control register	DRI1DEC1CNT	H'00	H'FFBF D03C	8	28-51
DRI1DEC1 reload register	DRI1DEC1RLD	H'0000	H'FFBF D03E	16	28-61
DRI1DEC1 counter	DRI1DEC1CT	H'0000	H'FFBF D040	16	28-63
DRI1DEC2 control register	DRI1DEC2CNT	H'00	H'FFBF D042	8	28-53

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DRI1DEC2 reload register	DRI1DEC2RLD	H'0000	H'FFBF D044	16	28-61
DRI1DEC2 counter	DRI1DEC2CT	H'0000	H'FFBF D046	16	28-63
DRI1DEC3 control register	DRI1DEC3CNT	H'00	H'FFBF D048	8	28-55
DRI1DEC3 reload register	DRI1DEC3RLD	H'0000	H'FFBF D04A	16	28-61
DRI1DEC3 counter	DRI1DEC3CT	H'0000	H'FFBF D04C	16	28-63
DRI1DEC4 control register	DRI1DEC4CNT	H'00	H'FFBF D04E	8	28-57
DRI1DEC4 reload register	DRI1DEC4RLD	H'0000	H'FFBF D050	16	28-61
DRI1DEC4 counter	DRI1DEC4CT	H'0000	H'FFBF D052	16	28-63
DRI1DEC5 control register	DRI1DEC5CNT	H'00	H'FFBF D054	8	28-59
DRI1DEC5 reload register	DRI1DEC5RLD	H'0000	H'FFBF D056	16	28-62
DRI1DEC5 counter	DRI1DEC5CT	H'0000	H'FFBF D058	16	28-64
DRI2DIN interrupt request status register	DRI2DINIST	H'00	H'FFBF E000	8	28-12
DRI2DIN interrupt request enable register	DRI2DINIEN	H'00	H'FFBF E001	8	28-13
DRI2DEC interrupt request status register	DRI2DECIST	H'00	H'FFBF E004	8	28-17
DRI2DEC interrupt request enable register	DRI2DECIEN	H'00	H'FFBF E005	8	28-18
DRI2 transfer interrupt request status register	DRI2TRMIST	H'00	H'FFBF E008	8	28-22
DRI2 transfer interrupt request enable register	DRI2TRMIEN	H'00	H'FFBF E009	8	28-24
DRI2 transfer control register	DRI2TRMCNT	H'00	H'FFBF E00C	8	28-28
DRI2 special mode register	DRI2SPMOD	H'00	H'FFBF E00D	8	28-31
DRI2 data acquisition control register	DRI2DCAPCNT	H'0000	H'FFBF E00E	16	28-35
DRI2 data decimation control register	DRI2DSELCNT	H'00	H'FFBF E010	8	28-39
DRI2 data decimation event selection register	DRI2DEVTCNT	H'00	H'FFBF E011	8	28-40
DRI2DIN input event selection register	DRI2DINSEL	H'00	H'FFBF E012	8	28-41
DRI2DD input enable register	DRI2DDEN	H'0000 0000	H'FFBF E014	32	28-42
DRI2 data acquisition event count setting register	DRI2DCAPNUM	H'0000 0000	H'FFBF E018	32	28-43
DRI2 acquisition event counter	DRI2DCAPCT	H'0000 0000	H'FFBF E01C	32	28-44
DRI2 transfer counter	DRI2TRMCT	H'0000 0000	H'FFBF E020	32	28-45
DRI2 address reload register 0	DRI2ADR0RLD	H'0000 0000	H'FFBF E024	32	28-46
DRI2 address counter 0	DRI2ADR0CT	H'0000 0000	H'FFBF E028	32	28-47
DRI2 address reload register 1	DRI2ADR1RLD	H'0000 0000	H'FFBF E02C	32	28-46
DRI2 address counter 1	DRI2ADR1CT	H'0000 0000	H'FFBF E030	32	28-47
DRI2 input processing control register	DRI2DINCNT	H'0000	H'FFBF E034	16	28-48
DRI2DEC0 control register	DRI2DEC0CNT	H'00	H'FFBF E036	8	28-49
DRI2DEC0 reload register	DRI2DEC0RLD	H'0000	H'FFBF E038	16	28-61
DRI2DEC0 counter	DRI2DEC0CT	H'0000	H'FFBF E03A	16	28-63
DRI2DEC1 control register	DRI2DEC1CNT	H'00	H'FFBF E03C	8	28-51
DRI2DEC1 reload register	DRI2DEC1RLD	H'0000	H'FFBF E03E	16	28-61
DRI2DEC1 counter	DRI2DEC1CT	H'0000	H'FFBF E040	16	28-63
DRI2DEC2 control register	DRI2DEC2CNT	H'00	H'FFBF E042	8	28-53

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DRI2DEC2 reload register	DRI2DEC2RLD	H'0000	H'FFBF E044	16	28-61
DRI2DEC2 counter	DRI2DEC2CT	H'0000	H'FFBF E046	16	28-63
DRI2DEC3 control register	DRI2DEC3CNT	H'00	H'FFBF E048	8	28-55
DRI2DEC3 reload register	DRI2DEC3RLD	H'0000	H'FFBF E04A	16	28-61
DRI2DEC3 counter	DRI2DEC3CT	H'0000	H'FFBF E04C	16	28-63
DRI2DEC4 control register	DRI2DEC4CNT	H'00	H'FFBF E04E	8	28-57
DRI2DEC4 reload register	DRI2DEC4RLD	H'0000	H'FFBF E050	16	28-61
DRI2DEC4 counter	DRI2DEC4CT	H'0000	H'FFBF E052	16	28-63
DRI2DEC5 control register	DRI2DEC5CNT	H'00	H'FFBF E054	8	28-59
DRI2DEC5 reload register	DRI2DEC5RLD	H'0000	H'FFBF E056	16	28-62
DRI2DEC5 counter	DRI2DEC5CT	H'0000	H'FFBF E058	16	28-64

Note:
- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

The DRIi interrupt related registers (DRIiDINIST, DRIiDINIEN, DRIiDECIST, DRIiDECIEN, DRIiTRMIST, and DRIiTRMIEN) control the interrupt request signals output from the DRIi module to the interrupt controller.

- Interrupt request status bits

  These status bits are used to determine the interrupt request that occurred and are set to "1" when the corresponding interrupt request occurs. These bits cannot be set to "1" in software. These status bits are cleared by writing a "0" and once a "1" has been written the state of the status bit is retained. Note that since this operation is not influenced by the interrupt request enable bits, they can also be used to verify the operation of peripheral functions. When handling an interrupt, the handler must only clear those status bits that correspond to the interrupts it is actually processing. If an interrupt handler clears status bits for interrupts it is not handling, interrupts that have not been processed will be cleared.

- Interrupt request enable bits

  These flags are used to enable interrupt requests. To enable an interrupt request, set the corresponding flag to "1" and to disable an request, set the flag to "0".

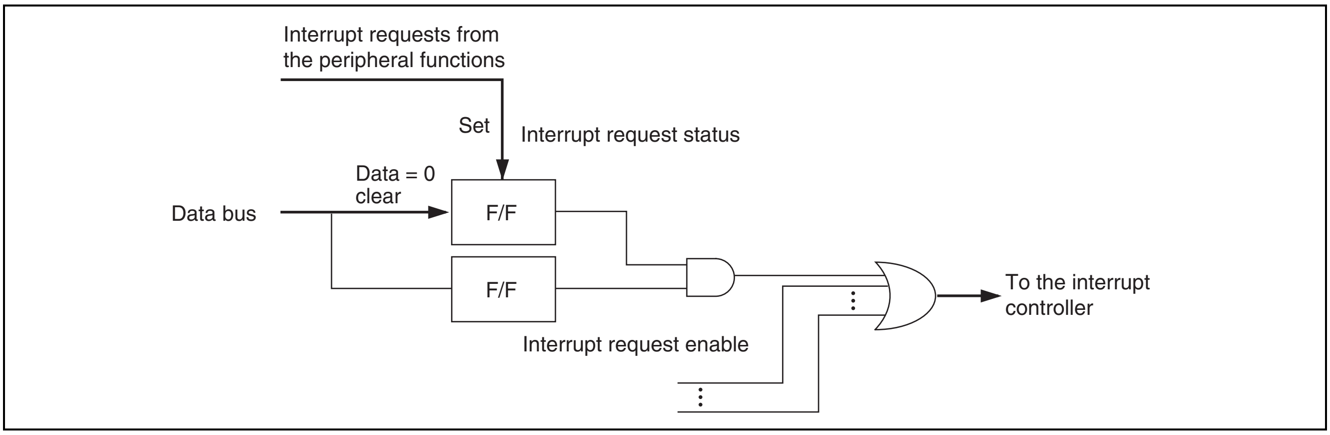
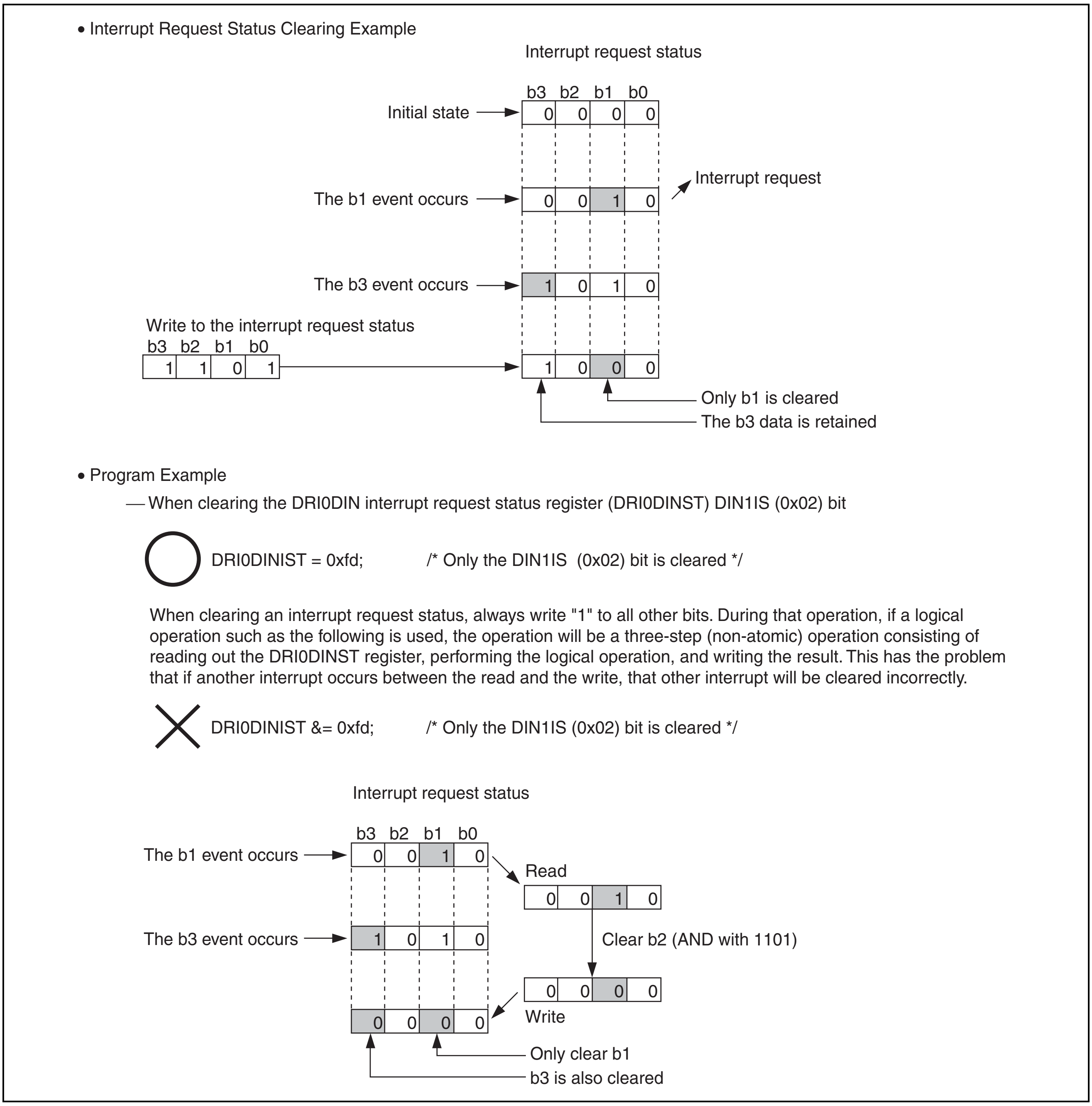


Figure 28.2 Interrupt Request Status Register and Interrupt Request Enable Register



• Program Example

— When clearing the DRI0DIN interrupt request status register (DRI0DINST) DIN1IS (0x02) bit

○
```
DRI0DINIST = 0xfd;            /* Only the DIN1IS  (0x02) bit is cleared */
```

When clearing an interrupt request status, always write "1" to all other bits. During that operation, if a logical operation such as the following is used, the operation will be a three-step (non-atomic) operation consisting of reading out the DRI0DINST register, performing the logical operation, and writing the result. This has the problem that if another interrupt occurs between the read and the write, that other interrupt will be cleared incorrectly.

✕
```
DRI0DINIST &= 0xfd;           /* Only the DIN1IS (0x02) bit is cleared */
```

Figure 28.3 Interrupt Request Status Clearing Examples

The DRIi interrupt related registers (DRI0DINDST, DRI0DINDEN, DRI0DECDST, DRI0DECDEN, DRI0TRMDST, DRI0TRMDEN) control the DMA request signals output from the DRIi module to the DMAC module.

- DMA transfer request status bits

  These status bits are used to determine whether or not a DMA transfer request is outstanding. While the DMA transfer request enable bits are "1", they are set to "1" when a DMA transfer request occurs and are automatically cleared to "0" when the DMA controller accepts the transfer request. While the DMA transfer request enable bits are "0", a DMA transfer request does not occur. Also note that when the DMA transfer request enable bits are changed from "1" to "0", no DMA transfer request will be issued after that point.

  If a DMA transfer request has already been issued, the "1" state will be retained until the DMA transfer request is accepted, and it will be cleared to "0" when accepted.

An outstanding DMA transfer request can be forcibly cancelled by clearing the corresponding bit to "0". It is not possible to write "1" to these bits.

- DMA transfer request enable bits

  These flags are used to enable DMA transfer requests. Set these flags to "1" to enable a DMA transfer request, and set them to "0" to disable a request.

  To prevent incorrect DMA operation, only rewrite these bits from the DMA transfer masked state to the DMA transfer enabled state when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"). Do not rewrite from the DMA transfer enabled state to the DMA transfer masked state when DRI acquisition is enabled, since that can result in a DMA request not being handled.

28.3.1 DRIiDIN Interrupt Request Status Register (DRIiDINIST)

When a DINn event is detected according to the settings of the DRIi input processing control register (DRIiDINCNT), the status bit corresponding to that DINn is set to "1".

If a status bit is set by an interrupt request and that status bit is cleared by software at the same time, the set of the status bit by the interrupt request takes precedence.

Legend: n = 0 to 5

DRI0DIN Interrupt Request Status Register (DRI0DINIST) <P4 address: location H'FFBF C000>
DRI1DIN Interrupt Request Status Register (DRI1DINIST) <P4 address: location H'FFBF D000>
DRI2DIN Interrupt Request Status Register (DRI2DINIST) <P4 address: location H'FFBF E000>

Bit:	7	6	5	4	3	2	1	0
	—	—	DIN5 IS	DIN4 IS	DIN3 IS	DIN2 IS	DIN1 IS	DIN0 IS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DIN5IS	0	R	*[1]	DIN5 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
4	DIN4IS	0	R	*[1]	DIN4 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
3	DIN3IS	0	R	*[1]	DIN3 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
2	DIN2IS	0	R	*[1]	DIN2 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
1	DIN1IS	0	R	*[1]	DIN1 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
0	DIN0IS	0	R	*[1]	DIN0 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred

Note: *1 Only "0" is valid on write. The previous value is retained if a "1" is written.

28.3.2 DRIiDIN Interrupt Request Enable Register (DRIiDINIEN)

Controls the enabled/disabled states for interrupts due to DINn event detection. When one of these bits is set to "1", the interrupt request for the corresponding DINn event detection is enabled.

Legend: n = 0 to 5

DRI0DIN Interrupt Request Enable Register (DRI0DINIEN) <P4 address: location H'FFBF C001>
DRI1DIN Interrupt Request Enable Register (DRI1DINIEN) <P4 address: location H'FFBF D001>
DRI2DIN Interrupt Request Enable Register (DRI2DINIEN) <P4 address: location H'FFBF E001>

Bit:	7	6	5	4	3	2	1	0
	—	—	DIN5 IEN	DIN4 IEN	DIN3 IEN	DIN2 IEN	DIN1 IEN	DIN0 IEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DIN5IEN	0	R	W	DIN5 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
4	DIN4IEN	0	R	W	DIN4 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
3	DIN3IEN	0	R	W	DIN3 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
2	DIN2IEN	0	R	W	DIN2 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
1	DIN1IEN	0	R	W	DIN1 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
0	DIN0IEN	0	R	W	DIN0 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled

28.3.3 DRI0DIN DMA Transfer Request Status Register (DRI0DINDST)

These bits indicate whether or not there is a DMA transfer request due to a DINn event being detected according to the settings of the DRI0 input processing control register (DRI0DINCNT). These bits are set by the occurrence of a DMA transfer request in the DMA transfer request enabled state only for bits that are set in the DRI0DIN DMA transfer enable register (DRI0DINDEN). If a status bit is set by a DMA transfer request and that status bit is cleared by software at the same time, the set of the status bit by the DMA transfer request takes precedence.

Legend: n = 0 to 5

DRI0DIN DMA Transfer Request Status Register (DRI0DINDST) <P4 address: location H'FFBF C002>

Bit:	7	6	5	4	3	2	1	0
	—	—	DIN5 DS	DIN4 DS	DIN3 DS	DIN2 DS	DIN1 DS	DIN0 DS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DIN5DS	0	R	*1	DIN5 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state
4	DIN4DS	0	R	*1	DIN4 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state

Bit	Abbreviation	After Reset	R	W	Description
3	DIN3DS	0	R	*[1]	DIN3 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state
2	DIN2DS	0	R	*[1]	DIN2 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state
1	DIN1DS	0	R	*[1]	DIN1 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state
0	DIN0DS	0	R	*[1]	DIN0 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state

Note: *1 Only "0" is valid on write. The previous value is retained if a "1" is written.

28.3.4 DRI0DIN DMA Transfer Enable Register (DRI0DINDEN)

Controls the enabled/disabled states for DMA transfer requests due to DINn event detection. Setting one of these bits to "1" enables the corresponding DMA transfer request output due to DINn event detection. If a DMA transfer request mask (disable) setting and an internal DMA transfer request occur at the same time, the DMA transfer request mask (disable) setting takes precedence.

Also note that it is only possible to rewrite the DRI0DINDEN register bits from the transfer masked state to the transfer enabled state when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"). Do not rewrite from the transfer enabled state to the transfer masked state when DRI acquisition is enabled.

Legend: n = 0 to 5

DRI0DIN DMA Transfer Enable Register (DRI0DINDEN)

<P4 address: location H'FFBF C003>

Bit:	7	6	5	4	3	2	1	0
	—	—	DIN5 DEN	DIN4 DEN	DIN3 DEN	DIN2 DEN	DIN1 DEN	DIN0 DEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DIN5DEN	0	R	W	DIN5 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
4	DIN4DEN	0	R	W	DIN4 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
3	DIN3DEN	0	R	W	DIN3 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
2	DIN2DEN	0	R	W	DIN2 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
1	DIN1DEN	0	R	W	DIN1 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
0	DIN0DEN	0	R	W	DIN0 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled

28.3.5 DRIiDEC Interrupt Request Status Register (DRIiDECIST)

The corresponding status bit in this register is set to "1" by the underflow of one of the six event counters (DEC5 to DEC0) built into the DRIi module. If a status bit is set by an interrupt request and cleared by software at the same time, the set of the status bit by the interrupt takes precedence.

DRI0DEC Interrupt Request Status Register (DRI0DECIST)	<P4 address: location H'FFBF C004>
DRI1DEC Interrupt Request Status Register (DRI1DECIST)	<P4 address: location H'FFBF D004>
DRI2DEC Interrupt Request Status Register (DRI2DECIST)	<P4 address: location H'FFBF E004>

Bit:	7	6	5	4	3	2	1	0
	—	—	DEC5 IS	DEC4 IS	DEC3 IS	DEC2 IS	DEC1 IS	DEC0 IS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DEC5IS	0	R	*[1]	DEC5 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
4	DEC4IS	0	R	*[1]	DEC4 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
3	DEC3IS	0	R	*[1]	DEC3 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
2	DEC2IS	0	R	*[1]	DEC2 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
1	DEC1IS	0	R	*[1]	DEC1 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred
0	DEC0IS	0	R	*[1]	DEC0 Interrupt Request Status Bit 0: No interrupt occurred 1: An interrupt occurred

Note: *1 Only "0" is valid on write. The previous value is retained if a "1" is written.

28.3.6 DRIiDEC Interrupt Request Enable Register (DRIiDECIEN)

Controls the enabled/disabled states of interrupts due to event counter underflow. If one of these bits is set to "1", the corresponding event counter underflow interrupt is enabled.

DRI0DEC Interrupt Request Enable Register (DRI0DECIEN)	<P4 address: location H'FFBF C005>
DRI1DEC Interrupt Request Enable Register (DRI1DECIEN)	<P4 address: location H'FFBF D005>
DRI2DEC Interrupt Request Enable Register (DRI2DECIEN)	<P4 address: location H'FFBF E005>

Bit:	7	6	5	4	3	2	1	0
	—	—	DEC5 IEN	DEC4 IEN	DEC3 IEN	DEC2 IEN	DEC1 IEN	DEC0 IEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DEC5IEN	0	R	W	DEC5 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
4	DEC4IEN	0	R	W	DEC4 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
3	DEC3IEN	0	R	W	DEC3 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
2	DEC2IEN	0	R	W	DEC2 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
1	DEC1IEN	0	R	W	DEC1 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
0	DEC0IEN	0	R	W	DEC0 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled

28.3.7 DRI0DEC DMA Transfer Request Status Register (DRI0DECDST)

These bits indicate whether or not there is a DMA transfer request due to an event counter underflow. These bits are set by the occurrence of a DMA transfer request in the DMA transfer request enabled state only for bits that are set in the DRI0DEC DMA transfer enable register (DRI0DECDEN). If a status bit is set by a DMA transfer request and that status bit is cleared by software at the same time, the set of the status bit by the DMA transfer request takes precedence.

DRI0DEC DMA Transfer Request Status Register (DRI0DECDST) <P4 address: location H'FFBF C006>

Bit:	7	6	5	4	3	2	1	0
	—	—	DEC5DS	DEC4DS	DEC3DS	DEC2DS	DEC1DS	DEC0DS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DEC5DS	0	R	*[1]	DEC5 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state
4	DEC4DS	0	R	*[1]	DEC4 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] A DMA transfer request occurred in the DMA transfer request enabled state
3	DEC3DS	0	R	*[1]	DEC3 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state

Bit	Abbreviation	After Reset	R	W	Description
2	DEC2DS	0	R	*[1]	DEC2 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state
1	DEC1DS	0	R	*[1]	DEC1 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state
0	DEC0DS	0	R	*[1]	DEC0 DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • A DMA transfer request was accepted by the DMAC • A "0" was written by software (forcible request clear) [Condition for setting to "1"] • A DMA transfer request occurred in the DMA transfer request enabled state

Note: *1 Only "0" is valid on write. The previous value is retained if a "1" is written.

28.3.8 DRI0DEC DMA Transfer Enable Register (DRI0DECDEN)

Controls the enabled/disabled states for DMA transfer requests due to event counter underflow events. Setting one of these bits to "1" enables the corresponding DMA transfer request output due to event counter underflow.

If a DMA transfer request mask (disable) setting and an internal DMA transfer request occur at the same time, the DMA transfer request mask (disable) setting takes precedence. Also note that it is only possible to rewrite the DRI0DECDEN register bits from the transfer masked state to the transfer enabled state when DEC counter operation is enabled (DRIiDECnCNT.DECnEN bit = "1"). Do not rewrite from the transfer enabled state to the transfer masked state when DEC counter operation is enabled.

DRI0DEC DMA transfer enable register (DRI0DECDEN) <P4 address: location H'FFBF C007>

Bit:	7	6	5	4	3	2	1	0
	—	—	DEC5 DEN	DEC4 DEN	DEC3 DEN	DEC2 DEN	DEC1 DEN	DEC0 DEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DEC5DEN	0	R	W	DEC5 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
4	DEC4DEN	0	R	W	DEC4 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
3	DEC3DEN	0	R	W	DEC3 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
2	DEC2DEN	0	R	W	DEC2 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
1	DEC1DEN	0	R	W	DEC1 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
0	DEC0DEN	0	R	W	DEC0 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled

28.3.9 DRIi Transfer Interrupt Request Status Register (DRIiTRMIST)

The bits in this register indicate the presence or absence of an interrupt request issued by a DRI transfer. If a status bit is set by an interrupt request and that status bit is cleared by software at the same time, the set of the status bit by the interrupt request takes precedence.

DRI0 Transfer Interrupt Request Status Register (DRI0TRMIST)	<P4 address: location H'FFBF C008>
DRI1 Transfer Interrupt Request Status Register (DRI1TRMIST)	<P4 address: location H'FFBF D008>
DRI2 Transfer Interrupt Request Status Register (DRI2TRMIST)	<P4 address: location H'FFBF E008>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	DTRFIS	DCPEIS	OVREIS	ADR1IS	ADR0IS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	DTRFIS	0	R	*[1]	DRIi Transfer Counter Interrupt Request Status Bit This bit is set when the DRIi transfer counter (DRIiTRMCT) underflows (the counter becomes H'0000 0000 and stops). 0: No interrupt request occurred 1: An interrupt request occurred
3	DCPEIS	0	R	*[1]	Interrupt Enable Error Interrupt Request Status Bit This bit is set to "1" if, before either the DRIi acquisition event counter (DRIiDCAPCT) or the DRIi transfer counter (DRIiTRMCT) underflows (the counter becomes H'0000 0000 and stops), either the DRIi data acquisition control register (DRIiDCAPCNT) DCPEN (acquisition enable) bit changes from "0" to "1" or an external event is detected. 0: No interrupt request occurred 1: An interrupt request occurred [Conditions for setting to "1"] • When acquisition by an external event is enabled with the DRIi data acquisition control register (DRIiDCAPCNT) DEXL (acquisition enable external factor selection) bits 1. If the selected external event is detected in the state where the DCPEN (acquisition enable) bit enables data acquisition.*[2] 2. If the selected external event is detected before the DRIi transfer counter (DRIiTRMCT) underflows (the counter becomes H'0000 0000 and stops)*[3] • When software changes the DCPEN (acquisition enable) bit from "0" to "1" before the DRIi transfer counter (DRIiTRMCT) underflows (the counter becomes H'0000 0000 and stops)*[3]

Bit	Abbreviation	After Reset	R	W	Description
2	OVREIS	0	R	*[1]	Overflow Error Interrupt Request Status Bit To prevent acquired data from being lost due to a SHwyRAM access conflict with another bus master, the DRIi module includes an internal 32 bit by 4 stage intermediate buffer. This bit, however, is set to "1" when a data acquisition event is detected in the state when all four stages are full. The data acquisition event detected in the buffer full state is ignored. 0: No interrupt request occurred 1: An interrupt request occurred
1	ADR1IS	0	R	*[1]	DRIi Address Counter 1 Interrupt Request Status Bit This bit is set to "1" when the DRIi transfer counter (DRIiTRMCT) underflows (the counter becomes H'0000 0000 and stops) in the state where DRIi address counter 1 (DRIiADR1CT) is enabled as the acquired data transfer destination. 0: No interrupt request occurred 1: An interrupt request occurred
0	ADR0IS	0	R	*[1]	DRIi Address Counter 0 Interrupt Request Status Bit This bit is set to "1" when the DRIi transfer counter (DRIiTRMCT) underflows (the counter becomes H'0000 0000 and stops) in the state where DRIi address counter 0 (DRIiADR0CT) is enabled as the acquired data transfer destination. 0: No interrupt request occurred 1: An interrupt request occurred

Notes: *1 Only "0" is valid on write. The previous value is retained if a "1" is written.

*2 The acquisition event will be ignored.

*3 It will be necessary to clear to "0" both the DRIi transfer control register (DRIiTRMCNT) and the DRIi data acquisition control register (DRIiDCAPCNT) and initialize the interrupt control block.

28.3.10 DRIi Transfer Interrupt Request Enable Register (DRIiTRMIEN)

Controls the enabled/disabled states for DRIi transfer related interrupt requests. If one of these bits is set to "1", the corresponding interrupt is enabled.

DRI0 Transfer Interrupt Request Enable Register (DRI0TRMIEN)	<P4 address: location H'FFBF C009>
DRI1 Transfer Interrupt Request Enable Register (DRI1TRMIEN)	<P4 address: location H'FFBF D009>
DRI2 Transfer Interrupt Request Enable Register (DRI2TRMIEN)	<P4 address: location H'FFBF E009>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	DTRF IEN	DCPE IEN	OVRE IEN	ADR1 IEN	ADR0 IEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	DTRFIEN	0	R	W	DRIi Transfer Counter Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
3	DCPEIEN	0	R	W	Acquisition Enable Error Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
2	OVREIEN	0	R	W	Overrun Error Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
1	ADR1IEN	0	R	W	DRIi Address Counter 1 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled
0	ADR0IEN	0	R	W	DRIi Address Counter 0 Interrupt Request Enable Bit 0: Interrupt request masked (disabled) 1: Interrupt request enabled

28.3.11 DRI0 DMA Transfer Request Status Register (DRI0TRMDST)

The bits in this register indicate the presence or absence of a DMA transfer request issued by a DRI transfer.

If a status bit is set by a DMA transfer request and that status bit is cleared by software at the same time, the set of the status bit by the DMA transfer request takes precedence.

DRI0 DMA Transfer Request Status Register (DRI0TRMDST) <P4 address: location H'FFBF C00A>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	DTRFDS	DUDFDS	—	ADR1DS	ADR0DS
After Reset:	0	0	0	0	0		0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	DTRFDS	0	R	*[1]	DRI0 Transfer Counter DMA Transfer Request Status Bit 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • When the DMA transfer request is accepted by the DMAC • When "0" is written by software (forcible request clear) [Condition for setting to "1"] • When the DRI0 transfer counter (DRI0TRMCT) underflows (the counter becomes H'0000 0000 and stops) in the state where DMA transfer requests are enabled.
3	DUDFDS	0	R	*[1]	DRI0 Acquisition Event Counter Underflow DMA Transfer Request Status Bit This bit is set at the point the DRI0 acquisition event counter (DRI0DCAPCT) underflows (the counter becomes H'0000 0000 and stops). 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • When the DMA transfer request is accepted by the DMAC • When "0" is written by software (forcible request clear) [Condition for setting to "1"] • When the DRI0 acquisition event counter (DRI0DCAPCT) underflows (the counter becomes H'0000 0000 and stops) in the state where DMA transfer requests are enabled.
2	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
1	ADR1DS	0	R	*[1]	DRI0 Address Counter 1 DMA Transfer Request Status Bit The ADR1DS bit is set under the same conditions as the DRI0 transfer interrupt request status register (DRI0TRMIST). 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • When the DMA transfer request is accepted by the DMAC • When "0" is written by software (forcible request clear) [Condition for setting to "1"] • When a DMA transfer request occurs in the state where DMA transfer requests are enabled.
0	ADR0DS	0	R	*[1]	DRI0 Address Counter 0 DMA Transfer Request Status Bit The ADR0DS bit is set under the same conditions as the DRI0 transfer interrupt request status register (DRI0TRMIST). 0: No DMA transfer request occurred 1: A DMA transfer request occurred [Conditions for clearing to "0"] • When the DMA transfer request is accepted by the DMAC • When "0" is written by software (forcible request clear) [Condition for setting to "1"] • When a DMA transfer request occurs in the state where DMA transfer requests are enabled.

Note: *1 Only "0" is valid on write. The previous value is retained if a "1" is written.

28.3.12 DRI0 DMA Transfer Enable Register (DRI0TRMDEN)

Controls the enabled/disabled states for DRI0 transfer related DMA transfer requests. If one of these bits is set to "1", the corresponding DMA transfer request signal output is enabled. If a DMA transfer mask (disable) is set at the same time as an internal DMA transfer request, the DMA transfer mask (disable) takes precedence. Also note that when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"), the DRI0TRMDEN register may only be rewritten from the transfer masked state to the transfer enabled state. Do not rewrite any bits in this register from the transfer enabled state to the transfer masked state when DRI acquisition is enabled.

DRI0 DMA Transfer Enable Register (DRI0TRMDEN) <P4 address: location H'FFBF C00B>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	DTRF DEN	DUDF DEN	—	ADR1 DEN	ADR0 DEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	DTRFDEN	0	R	W	DRI0 Transfer Counter DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
3	DUDFDEN	0	R	W	DRI0 Acquisition Event Counter Underflow DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
2	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
1	ADR1DEN	0	R	W	DRI0 Address Counter 1 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled
0	ADR0DEN	0	R	W	DRI0 Address Counter 0 DMA Transfer Request Enable Bit 0: DMA transfer request masked (disabled) 1: DMA transfer request enabled

28.3.13 DRIi Transfer Control Register (DRIiTRMCNT)

The only rewrite of the DRIiTRMCNT that may be performed when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1") is to change the DRST bit from "1" to "0". Other changes to that bit or changes to any other bit may not be performed when DRI acquisition is enabled.

DRI0 Transfer Control Register (DRI0TRMCNT)	<P4 address: location H'FFBF C00C>
DRI1 Transfer Control Register (DRI1TRMCNT)	<P4 address: location H'FFBF D00C>
DRI2 Transfer Control Register (DRI2TRMCNT)	<P4 address: location H'FFBF E00C>

Bit:	7	6	5	4	3	2	1	0
	ADEV	—	ADSL		ADMD	ADST	DBST	DRST
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	ADEV	0	R	W	Address Counter Switching Selection Bit The DRIi module includes two address counters that specify the address in SHwyRAM which is the transfer destination, and applications can select which address counter is used. This bit is only valid when the ADSL bit setting is set to "DRIi address counter 0/1 alternation" and it selects the event that switches between DRIi address counter 0 (DRIiADR0CT) and DRIi address counter 1 (DRIiADR1CT), which specify the address in SHwyRAM that is the transfer destination for acquired data. 0: DRIi transfer counter underflow 1: DEC4 underflow Note: • When DEC4 underflow is selected as the address counter switching event, DIN1 event detection/acquisition event may not be selected as the DEC4 count event.
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	ADSL	00	R	W	Address Counter Selection Bits • DRIi address counter 0 selected Data is transferred to SHwyRAM specified by DRIi address counter 0 (DRIiADR0CT). • DRIi address counter 1 selected Data is transferred to SHwyRAM specified by DRIi address counter 1 (DRIiADR1CT). • DRIi address counter 0/1 alternation The DRIi address counter is switched in hardware by the event selected by the ADEV (address counter switching selection) bit. After a microcontroller reset is cleared, DRIi address counter 0 (DRIiADR0CT) is active. Also, the active DRIi address counter is initialized to be DRIi address counter 0 (DRIiADR0CT) when DRST (DRIi reset) is cleared to "0". 00: DRIi address counter 0 selected 01: DRIi address counter 1 selected 10: DRIi address counter 0/1 alternation 11: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
3	ADMD	0	R	W	Address Counter Operating Mode Selection Bit Selects the operating mode for DRIi address counter 0 (DRIiADR0CT) and DRIi address counter 1 (DRIiADR1CT). Both DRIi address counters operate with the same operating mode. • In continuous mode The active DRIi address counter is incremented by +4 each time a DRIi transfer completes. In continuous mode, the DRIi address reload register is not used. • In reload mode When the DRIi data acquisition control register (DRIiDCAPCNT) DCPEN (acquisition enable) bit changes from acquisition disabled to acquisition enabled, the value is loaded from the DRIi address reload register corresponding to the DRIi address counter and after that, the active DRIi address counter is incremented by +4 each time a DRIi transfer completes. 0: Continuous mode 1: Reload mode Note: • Each time a data acquisition event for 32 bits of data input externally occurs (each time 4 data acquisition events occur when 8 bits is selected as the input bus width, each time 2 data acquisition events occur when 16 bits is selected as the input bus width), a DRIi transfer is performed. If the set count in the DRIi data acquisition event count setting register (DRIiDCAPNUM) cannot be divided evenly by the 32-bit data unit size (a value other than 4n when the bus width is 8 bits, and a value other than 2n when the bus with is 16 bits), a DRIi transfer will be performed for the last acquisition event occurrence. Even if the set count cannot be evenly divided into 32-bit units, the DRIi transfers are performed in 32-bit units. The part that does not fill the 32 bits is filled with "0" and a 32-bit unit is transferred.
2	ADST	0	R	0	Address Counter Status Bit Indicates whether the DRIi transfer destination address specification is performed by DRIi address counter 0 or by DRIi address counter 1. 0: DRIi address counter 0 is active 1: DRIi address counter 1 is active
1	DBST	0	R	0	DRIi Buffer Status Bit Indicates whether or not there is data for which DRI transfer has not completed in the DRI module. To prevent loss of DRI transfer data within the DRI module, the module includes a 32 bits by 4 stage intermediate buffer, and this DBST bit will be "1" in the state where there is data present in this intermediate buffer. On the other hand, the DBST bit is a value "0" when there is no data in the intermediate buffer. Also note that this bit will be cleared to "0" when the DRST bit is cleared to "0". 0: There is no data in the DRIi buffer 1: There is data in the DRIi buffer

Bit	Abbreviation	After Reset	R	W	Description
0	DRST	0	R	W	DRIi Reset Bit This is the DRIi control block software reset bit; data acquisition and DRIi transfers are not performed in the state where this bit is "0". This bit must be set to "1" to operate the DRIi module. If this bit is cleared to "0" when DRI acquisition is enabled (DRIiDCAPCNT.DCPEN bit = "1"), the DRIi acquisition control block and the DRIi transfer control block will be initialized and, if there is DRIi transfer incomplete data in the DRIi module, all those transfers will be cancelled and at the same time the module will become unable to acquired data. The following registers and bits are affected by this bit. • ADST bit When DRIi address counter 0/1 alternation is selected with the ADSL bits, setting the DRST bit to "0" will make the DRIi address counter 0 (DRIiADR0CT) active and clear the ADST bit to "0". • DBST (DRIi buffer status) bit Setting the DRST bit to "0" will initialize DBST to "0". • DRIi transfer counter (DRIiTRMCT) Setting the DRST bit to "0" will initialize the DRIi transfer counter (DRIiTRMCT) to "0". 0: DRIi reset 1: Operation enabled Notes: • The DRST bit has no influence on DIN input processing control or DEC5 to DEC0 operation. • A period of 4 PAck is required for the change to become valid after changing the value of the DRST bit. The DRST bit must not be changed again during that period. • After manipulation the DRST bit, a period of 1 PAck is required until the ADST bit and DBST bits are initialized. • The values of the ADMD (address counter operating mode selection) bit, ADSL (address counter selection) bit, and ADEV (address counter switching) bit must not be changed when the DRST bit is in the "1" state.

28.3.14 DRIi Special Mode Register (DRIiSPMOD)

Faster data acquisition is possible if special mode is selected. However, the input data bus width becomes either 8 or 16 bits when special mode is used. The acquisition clock can be selected from DINj4 and DINj3. Furthermore, signals whose transfer rate has been halved external are passed to the DRIi event detection block and data acquisition block as shown in figure 28.6.

DRI0 Special Mode Register (DRI0SPMOD)	<P4 address: location H'FFBF C00D>
DRI1 Special Mode Register (DRI1SPMOD)	<P4 address: location H'FFBF D00D>
DRI2 Special Mode Register (DRI2SPMOD)	<P4 address: location H'FFBF E00D>

Bit:	7	6	5	4	3	2	1	0
	SPCPSL		—	—	SPMEN	SPISL	SPRSM	SPSSL
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	SPCPSL	00	R	W	Special Mode Control Block Acquistion Clock Selection Bits These bits select the pins that input the acquisition clock when special mode is selected. 00: DINj3 01: DINj4 10: Setting prohibited 11: Setting prohibited
5, 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
3	SPMEN	0	R	W	Special Mode Enable Bit Enables/disables special mode operation. The following limitations apply when special mode operation enabled is selected. • DRIi data acquisition control register (DRIiDCAPCNT) 1. DWDSL (input data bus width) bits The data widths that can be output in special mode are either 8 bits or 16 bits. Set the DWDSL bits as shown below according to the data width input. When the input data width is 8 bits: Set the DWDSL bits to "16 bits". When the input data width is 16 bits: Set the DWDSL bits to "32 bits". 2. DCPSL (acquisition event selection) bit The same acquisition event as that selected with the SPCPSL bits must be selected with DCPSL. 3. DTMSL (acquisition timing selection) bit Select the default setting. • DRIi input processing control register (DRIiDINCNT) 1. DINnED (DINn event detection control) bit Select falling edge detection for the DIN selected with the DCPSL (acquisition event selection) bit and SPCPSL (special mode control block acquisition clock selection) bits. 0: Special mode disabled 1: Special mode enabled Note: • This register must only be set when the DRIi transfer control register (DRIiTRMCNT) DRST (DRIi reset) bit is in the "0" (disabled) state. Legend: n = 0 to 5

Bit	Abbreviation	After Reset	R	W	Description
2	SPISL	0	R	W	Special Mode Control Block Initialization DIN1 Level Selection Bit The special mode control circuit block can be initialized by the signal input to DIN1. This bit selects at what DIN1 level this initialization is performed. When DIN1 goes to the initialization level, all of the output signals to the event detection block and output signals to the data acquisition block go to the "L" level and the data acquisition operation is not performed. Inversely, when DIN1 is not at the initialization level, the data acquisition operation is performed and the signals shown in figure 28.6 are passed to the event detection and data acquisition blocks. The initialization timing is controlled by the SPRSM (special mode control block initialization method selection) bit: when SPRSM is set to "0", initialization is performed when DIN1 reaches the initialization level, and when SPRSM is set to "1", initialization is performed 4 cycles of the acquisition clock after DIN1 reaches the initialization level. Note that the DIN1 special mode control circuit block initialization function is not influenced by the setting of the DRIi input processing control register (DRIiDINCNT) DIN1ED bit. Also, this bit must only be changed when the DRIi transfer control register (DRIiTRMCNT) DRST (DRIi reset) bit is in the "0" state. 0: "L" level 1: "H" level Note: • In the state where the DRIi data acquisition control register (DRIiDCAPCNT) DCPEN (acquisition enable) bit is "1", the following phenomena may occur when DIN1 changes to the initialization level. 1. The DRIi module incorrectly acquires data. 2. The acquisitions for the 8 data items prior to the change to the reset state are not performed. • The above phenomena can be avoided by selecting the delayed reset method with the SPRSM (special mode control block initialization method selection) bit and inputting the special mode control block acquisition clock during the period the initialization level is applied to DIN1.
1	SPRSM	0	R	W	Special Mode Control Block Initialization Method Selection Bit Selects the special mode control block initialization method. When this bit is "0", the DIN1 input is used directly as the special mode control block initialization signal. When this bit is "1", a signal that is the DIN1 signal delayed by 4 cycles of the special mode control block acquisition clock is used as the initialization signal. 0: DIN1 direct reset 1: DIN1 delayed reset

Bit	Abbreviation	After Reset	R	W	Description
0	SPSSL	0	R	W	Acquisition Edge Selection Bit Select the falling edge as the acquisition edge if the transfer method shown in figure 28.4 is used and select the rising edge as the acquisition edge if the transfer method shown in figure 28.5 is used. Only change the value of this bit when the DRIi transfer control register (DRIiTRMCNT) DRST (DRIi reset) bit is "0". Note that in special mode only one of DIN4 to DIN3 may be selected as the acquisition clock with the SPCSPL setting. Also, in special mode, the signal controlled by the DRIi transfer control register (DRIiTRMCNT) DINiED (DINi event detection control) bit is not the signal input to DINn (the signal selected by SPCSPL) but rather is the "signal output to the event detection block" shown in figure 28.6. 0: Rising edge 1: Falling edge Legend: n = 2 to 4

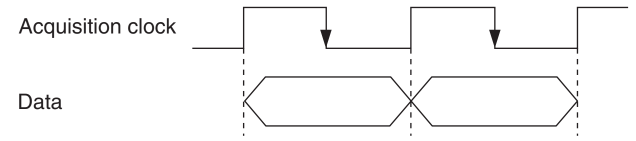


Figure 28.4 Data Transfer Method 1

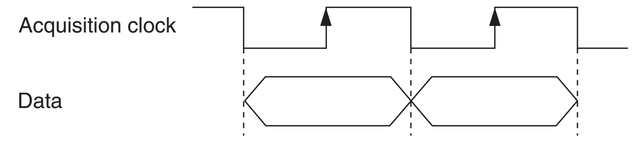
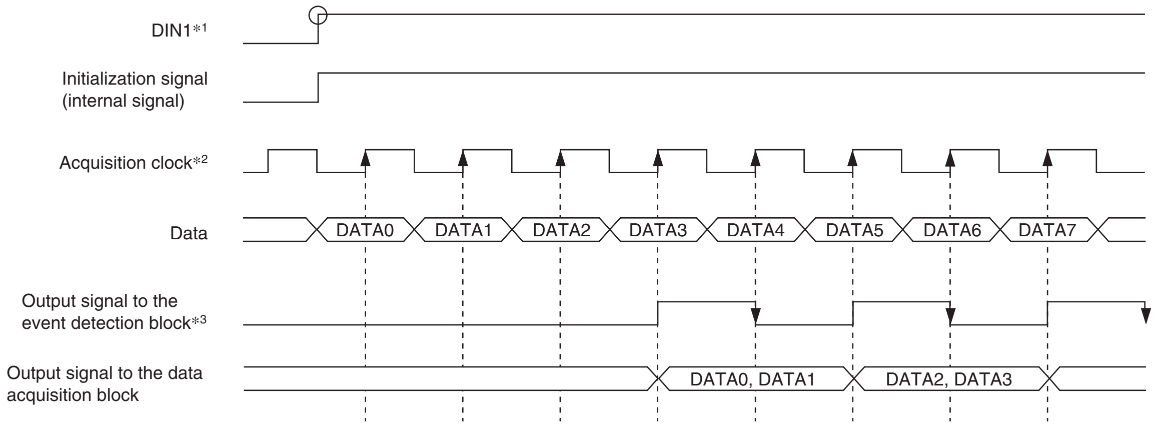


Figure 28.5 Data Transfer Method 2


DIN1*1
Initialization signal (internal signal)
Acquisition clock*2
Data
DATA0
DATA1
DATA2
DATA3
DATA4
DATA5
DATA6
DATA7
Output signal to the event detection block*3
Output signal to the data acquisition block
DATA0, DATA1
DATA2, DATA3

Notes: *1 When the "L" level is selected by the SPISL bit in the DRIi special mode register (DRIiSPMOD).
*2 The acquisition clock shows the DIN selected by the SPCPSL (special mode control block acquisition clock selection) bits.
*3 Select falling edge detection by the DINnED (DINn event detection control) bit in the DRIi input processing control register (DRIiDINCNT) for the DIN selected by the SPCPSL (special mode control block acquisition clock selection) bits.

Figure 28.6 Timing Chart when Special Mode is Enabled

28.3.15 DRIi Data Acquisition Control Register (DRIiDCAPCNT)

This register sets items related to acquisition of data input in synchronization with the acquisition clock. This register must only be set after the DRIi transfer control register (DRIiTRMCNT) DRST (DRIi reset) bit is set to "1". Also, if the DRST bit is cleared to "0", this register must also be cleared to "0".

The only rewrite of the DRIiDCAPCNT that may be performed when DRI acquisition is enabled (DCPEN bit = "1") is to change the DCPEN (acquisition enable) bit from "1" to "0". Do not change other bit except DCPEN bit when DRI acquisition is enabled.

DRI0 Data Acquisition Control Register (DRI0DCAPCNT)	<P4 address: location H'FFBF C00E>
DRI1 Data Acquisition Control Register (DRI1DCAPCNT)	<P4 address: location H'FFBF D00E>
DRI2 Data Acquisition Control Register (DRI2DCAPCNT)	<P4 address: location H'FFBF E00E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTMSL					DWR PR	DCPSL		DWDSL		DDSSL		DEXSL			DCP EN
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	DTMSL	00000	R	W	Acquisition Timing Selection Bits These bits select the time from the detection of a data acquisition event to the point the data is acquired. The DRIi module performs event detection on each PAck rising edge, and the default selection time is to acquire the data on the PAck rising edge at the time the event was detected. Times from 1 PAck cycle to 31 PAck cycles later, referenced to that point, can be selected. Figure 28.7 shows the timing chart for data acquisition. 00000: Default 10000: 16 × PAck 00001: 1 × PAck 10001: 17 × PAck 00010: 2 × PAck 10010: 18 × PAck 00011: 3 × PAck 10011: 19 × PAck 00100: 4 × PAck 10100: 20 × PAck 00101: 5 × PAck 10101: 21 × PAck 00110: 6 × PAck 10110: 22 × PAck 00111: 7 × PAck 10111: 23 × PAck 01000: 8 × PAck 11000: 24 × PAck 01001: 9 × PAck 11001: 25 × PAck 01010: 10 × PAck 11010: 26 × PAck 01011: 11 × PAck 11011: 27 × PAck 01100: 12 × PAck 11100: 28 × PAck 01101: 13 × PAck 11101: 29 × PAck 01110: 14 × PAck 11110: 30 × PAck 01111: 15 × PAck 11111: 31 × PAck Note: • The default setting must be selected when special mode is selected.

Bit	Abbreviation	After Reset	R	W	Description
10	DWRPR	0	R	W	Acquisition Control Write Protect Bit This bit sets the write enabled/disabled state of the DCPEN (acquisition enable) bit and the DEXSL (acquisition enable external factor selection) bits during writes to this register. If this bit is "0" during a write, writes to those bits are enabled. If this bit is "1", writes to those bits will be ignored. (Note that this bit always reads as "0", regardless of the value that was written.) 0: Writing to the DCPEN bit and the DEXSL bits enabled 1: Writing to the DCPEN bit and the DEXSL bits disabled
9, 8	DCPSL	00	R	W	Acquisition Event Selection Bits These bits select the data acquisition event. In the state where the DRIi transfer control register (DRIiTRMCNT) DRST (DRIi reset) bit is "operation enabled" and the DCPEN (acquisition enable) bit is "data acquisition enabled", and furthermore decimation control is used, if the acquisition event conditions are met, data acquisition will be performed when the selected event is detected. Note that if the DCPEN (acquisition enable) bit is set at the same time as a data acquisition event is detected, the data acquisition will be performed. 00: Setting prohibited 01: DIN3 event detection 10: DIN4 event detection 11: DIN5 event detection Note: • When special mode is selected, the selected DIN must also be selected with the SPCPSL (special mode control block acquisition clock selection) bit.
7, 6	DWDSL	00	R	W	Input Data Bus Width Selection Bits These bits select the bus width for the externally input data. A DRIi transfer is performed after each four data acquisition event occurrences when the 8-bit width is selected, after each two data acquisition event occurrences when the 16-bit width is selected, and after each data acquisition event occurrence when the 32-bit width is selected. Note, however, that if the set count in the DRIi data acquisition event count setting register (DRIiDCAPNUM) cannot be divided evenly by the 32-bit data unit size (a value other than 4n when the bus width is 8 bits, and a value other than 2n when the bus with is 16 bits), a DRIi transfer will be performed for the last acquisition event occurrence. Figure 28.5 shows the bits acquired as data when each data bus width is set. 00: 8 bits 01: 16 bits 10: 32 bits (selectable in special mode only) 11: Setting prohibited Note: • Certain limitations apply on the input data bus width selection setting when special mode is selected. See section 28.3.14, DRIi Special Mode Register (DRIiSPMOD) for details.

Bit	Abbreviation	After Reset	R	W	Description
5, 4	DDSSL	00	R	W	Acquisition External Control Disable Factor Selection Bits These bits select an event that clears to "0" the acquisition enable external factor selection bits. 00: No disable factor selected. 01: DRIi acquisition event counter underflow 10: DEC3 underflow 11: DEC4 underflow
3 to 1	DEXSL	000	R	W	Acquisition Enable External Factor Selection Bits These bits select an external factor that sets the DCPEN (acquisition enable) bit to data acquisition enabled. If the event selected here is detected, the acquisition enable bit is set to "1". When no factor is selected, the enable bit will not be set by any external factor. Note that it is also possible to clear that bit to "0" in hardware by setting the DDSSL (acquisition external control disable factor selection) bits. 000: No external factor selected 001: DIN0 event detection 010: DIN1 event detection 011: DIN2 event detection 100: DIN5 event detection 101: DEC0 underflow 110: DEC5 underflow 111: PDAC event G
0	DCPEN	0	R	W	Acquisition Enable Bit Data acquisition is enabled when this bit is "1". 0: Data acquisition disabled 1: Data acquisition enabled [Conditions for clearing to "0"] • When the software writes "0" • When an underflow (counter stopped at H'0000 0000) occurs in the DRIi acquisition event counter (DRIiDCAPCNT) [Conditions for setting to "1"] • When the software writes "1" • When the event selected with the DEXSL (acquisition enable external factor selection) bits Notes: • When an external factor is selected with the DEXSL (acquisition enable external factor selection) bits, it is illegal for the software to set this bit to "1". • When setting this bit to "1" in software, always read the DRIi transfer counter (DRIiTRMCT) and verify that it is in the underflow state (counter stopped at H'0000 0000). • If contention between a set condition and a clear condition occurs, the clear operation will always take precedence.

Table 28.5 Acquisition Data Position

	DD31 to DD24	DD23 to DD16	DD15 to DD08	DD07 to DD00
When the 8-bit width is selected	Don't care			Acquisition data
When the 16-bit width is selected	Don't care		Acquisition data	
When the 32-bit width is selected	Acquisition data			

Notes:
- The relationship between the actual data bus width and the input data bus width changes when operating in special mode. See section 28.3.14, DRIi Special Mode Register (DRIiSPMOD) for details.
- "DD31" is the MSB and "DD00" is the LSB.

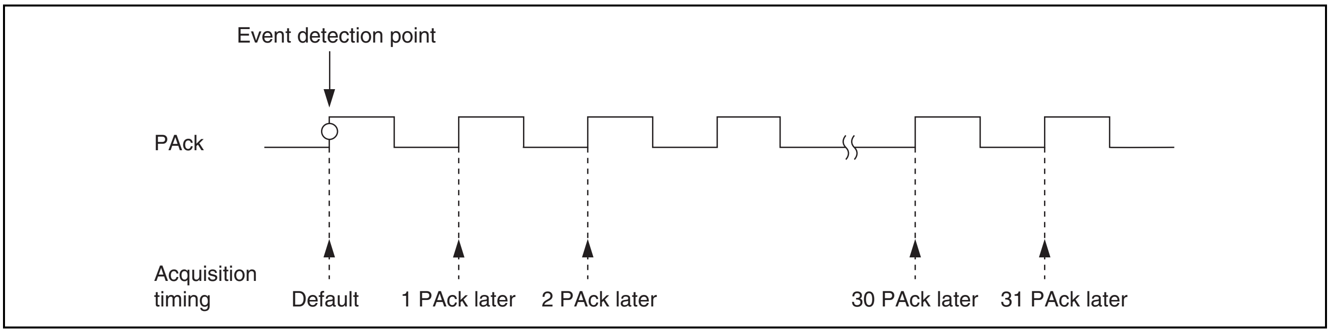


Figure 28.7 Data Acquisition Timing

28.3.16 DRIi Data Decimation Control Register (DRIiDSELCNT)

The DRIi module can acquire data while decimating that data in hardware using the module's 6 internal counters. This register sets items relating to control of that decimation operation.

If one of these bits is set to "0", data decimation control using the corresponding DEC counter will not be performed. When one of these bits is set to "1", data will only be acquired when the corresponding DEC counter is in the state set by the DRIi data decimation selection register (DRIiDEVTCNT).

If decimation control is enabled for multiple event counters. data will only be acquired on data acquisition events input when all of the DECn counters for which the decimation control bit is set to "1" are in the state set by the DSETVn (DECn decimation event selection) bits.

Events are valid as acquisition events starting with the next event after the counter goes to the state set with the DSETVn (DECn decimation event selection) bits.

Legend: n = 0 to 5

DRI0 Data Decimation Control Register (DRI0DSELCNT) <P4 address: location H'FFBF C010>
DRI1 Data Decimation Control Register (DRI1DSELCNT) <P4 address: location H'FFBF D010>
DRI2 Data Decimation Control Register (DRI2DSELCNT) <P4 address: location H'FFBF E010>

Bit:	7	6	5	4	3	2	1	0
	—	—	DSD5	DSD4	DSD3	DSD2	DSD1	DSD0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DSD5	0	R	W	DEC5 Data Decimation Control Bit 0: No decimation 1: Decimation performed according to DEC5CT
4	DSD4	0	R	W	DEC4 Data Decimation Control Bit 0: No decimation 1: Decimation performed according to DEC4CT
3	DSD3	0	R	W	DEC3 Data Decimation Control Bit 0: No decimation 1: Decimation performed according to DEC3CT
2	DSD2	0	R	W	DEC2 Data Decimation Control Bit 0: No decimation 1: Decimation performed according to DEC2CT
1	DSD1	0	R	W	DEC1 Data Decimation Control Bit 0: No decimation 1: Decimation performed according to DEC1CT
0	DSD0	0	R	W	DEC0 Data Decimation Control Bit 0: No decimation 1: Decimation performed according to DEC0CT

28.3.17 DRIi Data Decimation Event Selection Register (DRIiDEVTCNT)

Sets the data acquisition conditions for decimation control. If one of these bits is set to "0", data acquisition is only performed when the corresponding DEC counter underflows (counter value = H'FFFF). When set to "1", data is only acquired in states other than when a counter underflow has not occurred.

DRI0 Data Decimation Event Selection Register (DRI0DEVTCNT)	<P4 address: location H'FFBF C011>
DRI1 Data Decimation Event Selection Register (DRI1DEVTCNT)	<P4 address: location H'FFBF D011>
DRI2 Data Decimation Event Selection Register (DRI2DEVTCNT)	<P4 address: location H'FFBF E011>

Bit:	7	6	5	4	3	2	1	0
	—	—	DSEVT5	DSEVT4	DSEVT3	DSEVT2	DSEVT1	DSEVT0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	DSEVT5	0	R	W	DEC5 Data Decimation Event Selection Bit 0: Data is only acquired when a DEC5 underflow has occurred 1: Data is acquired at times other than when a DEC5 underflow has occurred
4	DSEVT4	0	R	W	DEC4 Data Decimation Event Selection Bit 0: Data is only acquired when a DEC4 underflow has occurred 1: Data is acquired at times other than when a DEC4 underflow has occurred
3	DSEVT3	0	R	W	DEC3 Data Decimation Event Selection Bit 0: Data is only acquired when a DEC3 underflow has occurred 1: Data is acquired at times other than when a DEC3 underflow has occurred
2	DSEVT2	0	R	W	DEC2 Data Decimation Event Selection Bit 0: Data is only acquired when a DEC2 underflow has occurred 1: Data is acquired at times other than when a DEC2 underflow has occurred
1	DSEVT1	0	R	W	DEC1 Data Decimation Event Selection Bit 0: Data is only acquired when a DEC1 underflow has occurred 1: Data is acquired at times other than when a DEC1 underflow has occurred
0	DSEVT0	0	R	W	DEC0 Data Decimation Event Selection Bit 0: Data is only acquired when a DEC0 underflow has occurred 1: Data is acquired at times other than when a DEC0 underflow has occurred

28.3.18 DRIiDIN Input Event Selection Register (DRIiDINSEL)

The event selected by the DIN5SL bits is input as the DIN5 input processing circuit input signal.

DRI0DIN Input Event Selection Register (DRI0DINSEL)	<P4 address: location H'FFBF C012>
DRI1DIN Input Event Selection Register (DRI1DINSEL)	<P4 address: location H'FFBF D012>
DRI2DIN Input Event Selection Register (DRI2DINSEL)	<P4 address: location H'FFBF E012>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DIN5SL	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1, 0	DIN5SL	00	R	W	DIN5 Input Event Selection Bits These bits specify the DIN5 (internal signal) input signal. The clock output from PSEL or the F/F output from timer TOU of ATU-IIIS is input as an internal signal to the DIN5 signal. For example, when the DIN5SL setting "01" (F/F (TOU1_0)) is selected, the value output by timer TOU1_0 (the value of the FFDT10 bit in TO1FFDR) is input to DIN5 as an internal signal. 00: PSLCLKB 01: F/F (TOU1_0) 10: F/F (TOU2_0) 11: F/F (TOU3_0)

28.3.19 DRIiDD Input Enable Register (DRIiDDEN)

The DRIiDDEN register controls the enabled/disabled state of data input to the DRIi module.

When the DDn input enable bit is set to "0", the input to DRIi is held fixed at "0" regardless of the level applied to the corresponding input pin. When the DDn input enable bit is set to "1", data is input to DRIi according to the level applied to the corresponding input pin.

Legend: n = 0 to 15

DRI0DD Input Enable Register (DRI0DDEN)	<P4 address: location H'FFBF C014>
DRI1DD Input Enable Register (DRI1DDEN)	<P4 address: location H'FFBF D014>
DRI2DD Input Enable Register (DRI2DDEN)	<P4 address: location H'FFBF E014>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DD15 EN	DD14 EN	DD13 EN	DD12 EN	DD11 EN	DD10 EN	DD09 EN	DD08 EN	DD07 EN	DD06 EN	DD05 EN	DD04 EN	DD03 EN	DD02 EN	DD01 EN	DD00 EN
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 16	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
15 to 0	DD15EN to DD00EN	All 0	R	W	DD15 to DD00 Input Enable Bits 0: Input disabled 1: Input enabled

28.3.20 DRIi Data Acquisition Event Count Setting Register (DRIiDCAPNUM)

Sets the event count for performing data acquisition. The value set here is also used as the DRIi acquisition event counter (DRIiDCAPCT) and DRIi transfer counter (DRIiTRMCT) reload value. The DRIi module performs DRIi transfers in 32-bit units. Transfers are still performed in 32-bit units even if the set value does not complete a 32-bit unit. The excess portion required to complete the 32-bit unit is filled with "0" values and transferred. When special mode is selected, this register must be set to a value that meets the conditions listed in table 28.6 according to the setting of the DRIi data acquisition control register (DRIiDCAPCNT) DWDSL (input data bus width selection) bit.

Applications must assure that the total amount of acquired data does not exceed the capacity of the SHwyRAM area supported by the DRIi module. This register must only be written when the DRIi data acquisition control register (DRIiDCAPCNT) DCPEN (acquisition enable) bit is "0".

DRI0 Data Acquisition Event Count Setting Register (DRI0DCAPNUM)	<P4 address: location H'FFBF C018>
DRI1 Data Acquisition Event Count Setting Register (DRI1DCAPNUM)	<P4 address: location H'FFBF D018>
DRI2 Data Acquisition Event Count Setting Register (DRI2DCAPNUM)	<P4 address: location H'FFBF E018>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DCAPNUM		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCAPNUM															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18 to 0	DCAPNUM	All 0	R	W	Transfer Event Count Bits [When special mode is disabled] Set this field to any value. [When special mode is enabled] Set this field to the count values show below. When the 32-bit width is selected (in 16-bit acquisition mode): Any value (1 or larger) When the 16-bit width is selected (in 8-bit acquisition mode): A multiple of 2 (2 or larger) When the 8-bit width is selected: Using this setting is not allowed.

Table 28.6 Relationship between Bus Width, Acquisition Event Settings, and Acquisition Counts in Special Mode

DWDSL (input data bus width selection)	Acquisition data bus width	DRIiDCAPNUM (acquisition event count)	Number of external acquisitions
8 bits (00)	(Illegal setting)	(Illegal setting)	(Illegal setting)
16 bits (01)	8-bit acquisition	Multiple of 2 (2 or larger)	The DRIiDCAPNUM set value × 2
32 bits (10)	16-bit acquisition	Any value (1 or larger)	The DRIiDCAPNUM set value × 2

28.3.21 DRIi Acquisition Event Counter (DRIiDCAPCT)

The DRIiDCAPCT register is a 19-bit counter that counts data acquisition events. The value of this register is reloaded from the DRIi data acquisition event count setting register (DRIiDCAPNUM) when the DRIi data acquisition control register (DRIiDCAPCNT) DCPEN (acquisition enable) bit changes from the data acquisition disabled state to the enabled state. After that, the DRIi acquisition event counter is decremented by "1" each time a data acquisition is performed.

Counter operation stops at the point the DRIi acquisition event counter becomes H'0000 0000, and the DCPEN bit is cleared to "0".

DRI0 Acquisition Event Counter (DRI0DCAPCT) <P4 address: location H'FFBF C01C>
DRI1 Acquisition Event Counter (DRI1DCAPCT) <P4 address: location H'FFBF D01C>
DRI2 Acquisition Event Counter (DRI2DCAPCT) <P4 address: location H'FFBF E01C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DCAPCT		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCAPCT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	N	Reserved Bits These bits are always read as "0".
18 to 0	DCAPCT	H'0000	R	N	Acquisition Event Counter

28.3.22 DRIi Transfer Counter (DRIiTRMCT)

The DRIiTRMCT counter is a 19-bit counter that counts DRIi transfer data acquisitions. This register is reloaded by the count value specified by the set value of the DRIi data acquisition event count setting register (DRIiDCAPNUM) and the set value of the DRIi data acquisition control register (DRIiDCAPCNT) DWDSL (input data bus width selection) bits when the DRIiDCAPCNT register DCPEN (acquisition enable) bit changes from the data acquisition disabled state to the enabled state. Since the DRIi module performs transfers in 32-bit units, only the values shown below may be reloaded as the counter value.

- When the 8-bit width is selected: The DRIiDCAPNUM value divided by 4
- When the 16-bit width is selected: The DRIiDCAPNUM value divided by 2
- When the 32-bit width is selected: The DRIiDCAPNUM value

Note: • When the DRIi data acquisition event count setting register (DRIiDCAPNUM) set value cannot be divided by the 32-bit data unit size, the indivisible part is rounded up for the reload counter value and transfers are always performed in 32-bit units.

While special mode is disabled, a DRIi transfer is executed each time four data acquisition events occur when 8 bits is selected as the external input data bus width, each time two data acquisition events occur when 16 bits is selected. If the DRIi data acquisition event count setting register (DRIiDCAPNUM) cannot be divided by the 32-bit data unit size (that is, a value other than 4n when 8 bits is selected or a value other than 2n when 16 bits is selected) a DRIi transfer will be executed when the last acquisition event occurs. Since DRIi transfers are performed in 32-bit units, if the transfer count cannot be divided by the 32-bit unit size the excess portion that does not fill out to 32 bits is filled with zeros ("0"). This counter is decremented by "1" each time a DRIi transfer completes. At the point the counter underflows (H'0000 0000), the count operation stops and a DRIi transfer counter interrupt request is issued. The DRIi counter underflow occurs when the counter reaches H'0000 0000 (and the count stops).

DRI0 Transfer Counter (DRI0TRMCT) <P4 address: location H'FFBF C020>
DRI1 Transfer Counter (DRI1TRMCT) <P4 address: location H'FFBF D020>
DRI2 Transfer Counter (DRI2TRMCT) <P4 address: location H'FFBF E020>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DRICT		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRICT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	N	Reserved Bits These bits are always read as "0".
18 to 0	DRICT	All 0	R	N	DRIi Transfer Counter

28.3.23 DRIi Address Reload Registers 0 and 1 (DRIiADR0RLD and DRIiADR1RLD)

DRIiADR0CT and DRIiADR1CT are registers that hold counter reload values. When reload mode is selected with the DRIi transfer control register (DRIiTRMCNT) ADMD (address counter operating mode selection) bit, the corresponding DRIi address counters are reloaded with the values set in these registers when the DRIi data acquisition control register (DRIiDCAPCNT) DCPEN (acquisition enable) bit changes from "0" to "1".

Note: • These registers may only be rewritten when the DRIi data acquisition control register (DRIiDCAPCNT) DCPEN (acquisition enable) bit is in the "0" state.

DRI0 Address Reload Register 0 (DRI0ADR0RLD) <P4 address: location H'FFBF C024>
DRI1 Address Reload Register 0 (DRI1ADR0RLD) <P4 address: location H'FFBF D024>
DRI2 Address Reload Register 0 (DRI2ADR0RLD) <P4 address: location H'FFBF E024>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DRIAD0RLD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRIAD0RLD														—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0 Address Reload Register 1 (DRI0ADR1RLD) <P4 address: location H'FFBF C02C>
DRI1 Address Reload Register 1 (DRI1ADR1RLD) <P4 address: location H'FFBF D02C>
DRI2 Address Reload Register 1 (DRI2ADR1RLD) <P4 address: location H'FFBF E02C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DRIAD1RLD		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRIAD1RLD														—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18 to 2	DRIADmRLD	All 0	R	W	Address Bits 18 to 2 Reload Value (256-Kbyte area)
1, 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: m = 0 or 1

28.3.24 DRIi Address Counters 0 and 1 (DRIiADR0CT and DRIiADR1CT)

The DRIiADR0CT and DRIiADR1CT counters are provided to specify bits A18 to A2 of the address in SHwyRAM that is the DRIi module transfer destination. Bits A31 to A19 are fixed at "0". These counters are incremented by "4" each time a DRIi transfer completes. There are two DRIi address counter operating modes, and applications can select the mode with the DRIi transfer control register (DRIiTRMCNT) ADMD bit. See the documentation of the DRIi transfer control register (DRIiTRMCNT) for details.

Notes:
- If a DRIi address counter value is a value other than an area in which SHwyRAM is located, the DRIi module will behave as though the DRIi transfers complete, but no writes of the acquired data will be performed whatsoever.
- A DRIi address counter is incremented by "4" when a DRIi transfer completed. This is performed for the one that is active at that time according to the setting of the DRIi transfer control register (DRIiTRMCNT) ADSL (address counter selection) bit.
- These registers must only be rewritten in the state where a DRIi transfer counter (DRIiTRMCT) underflow has occurred (the counter is stopped at the value H'0000 0000).

DRI0 Address Counters 0 (DRI0ADR0CT) <P4 address: location H'FFBF C028>
DRI1 Address Counters 0 (DRI1ADR0CT) <P4 address: location H'FFBF D028>
DRI2 Address Counters 0 (DRI2ADR0CT) <P4 address: location H'FFBF E028>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DRIAD0		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRIAD0														—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0 Address Counters 1 (DRI0ADR1CT) <P4 address: location H'FFBF C030>
DRI1 Address Counters 1 (DRI1ADR1CT) <P4 address: location H'FFBF D030>
DRI2 Address Counters 1 (DRI2ADR1CT) <P4 address: location H'FFBF E030>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DRIAD1		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRIAD1														—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18 to 2	DRIADn	All 0	R	W	Destination Address Bits 18 to 2 (256-Kbyte area)
1, 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Legend: n = 0 or 1

28.3.25 DRIi Input Processing Control Register (DRIiDINCNT)

This register selects the event detection method for signals input from sources external to the DRIi module. Event detection can be set to be on rising edges, falling edges, or both edges. If "input disabled" is selected, event detection is not performed.

DRI0 Input Processing Control Register (DRI0DINCNT)	<P4 address: location H'FFBF C034>
DRI1 Input Processing Control Register (DRI1DINCNT)	<P4 address: location H'FFBF D034>
DRI2 Input Processing Control Register (DRI2DINCNT)	<P4 address: location H'FFBF E034>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DIN5ED		DIN4ED		DIN3ED		DIN2ED		DIN1ED		DIN0ED	
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11, 10	DIN5ED	00	R	W	DIN5 Event Detection Control Bit 00: Input disabled 01: Rising edge detection 10: Falling edge detection 11: Both rising and falling edge detection
9, 8	DIN4ED	00	R	W	DIN4 Event Detection Control Bit 00: Input disabled 01: Rising edge detection 10: Falling edge detection 11: Both rising and falling edge detection
7, 6	DIN3ED	00	R	W	DIN3 Event Detection Control Bit 00: Input disabled 01: Rising edge detection 10: Falling edge detection 11: Both rising and falling edge detection
5, 4	DIN2ED	00	R	W	DIN2 Event Detection Control Bit 00: Input disabled 01: Rising edge detection 10: Falling edge detection 11: Both rising and falling edge detection
3, 2	DIN1ED	00	R	W	DIN1 Event Detection Control Bit 00: Input disabled 01: Rising edge detection 10: Falling edge detection 11: Both rising and falling edge detection
1, 0	DIN0ED	00	R	W	DIN0 Event Detection Control Bit 00: Input disabled 01: Rising edge detection 10: Falling edge detection 11: Both rising and falling edge detection

28.3.26 DRIiDEC0 Control Register (DRIiDEC0CNT)

Controls the DRIi module DEC0 internal event counter.

DRI0DEC0 Control Register (DRI0DEC0CNT)	<P4 address: location H'FFBF C036>
DRI1DEC0 Control Register (DRI1DEC0CNT)	<P4 address: location H'FFBF D036>
DRI2DEC0 Control Register (DRI2DEC0CNT)	<P4 address: location H'FFBF E036>

Bit:	7	6	5	4	3	2	1	0
	DEC0 MOD	—	DEC0 CS		DEC0 EXT			DEC0 EN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
7	DEC0MOD	0	R	W	DEC0 Operating Mode Selection Bit Selects the DRIiDEC0 counter (DRIiDEC0CT) operating mode. 0: One-shot mode 1: Continuous operating mode
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	DEC0CS	00	R	W	DEC0 Counter Event Selection Bits Selects the event that will be used as the DRIiDEC0CT counter count source. If an event is detected from the factor selected in the state where the DEC0EN bit is "1", the value of the DEC0 counter (DEC0CT) is decremented by "1". 00: DIN0 event detection 01: DIN1 event detection 10: Setting prohibited 11: DRIi acquisition event counter underflow
3 to 1	DEC0EXT	000	R	W	DEC0 Counter Enable Factor Selection Bits To enable counting the DRIiDEC0CT counter by external events, use this field to select the count enable factor. When an event is detected from the selected factor, the DEC0EN bit is set to "1". 000: External factors disabled 001: DIN0 event detection 010: DIN1 event detection 011: Setting prohibited 100: Acquisition enable 101: PDAC event H 110: Setting prohibited 111: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
0	DEC0EN	0	R	W	DEC0 Counter Enable Bit Controls the enabled/disabled state of DEC0 counter operation. This bit can be set to "1" by an external event. Note also that if one-shot mode is selected as the operating mode, this bit will be cleared to "0" when the DEC0 counter underflows. 0: Counting disabled 1: Counting enabled [Conditions for clearing to "0"] • When the software writes "0" • When one-shot mode is selected and the DEC0 counter underflows [Conditions for setting to "1"] • When the software writes "1" • When the event selected with the DEC0EXT bits occurs Notes: • It is illegal for the software to set this bit to "1" when an external factor is selected with the DEC0EXT bits. • When DEC operation is enabled, the only write operation that may be performed is a write of "0" to the DEC0EN bit to clear that bit. Do not rewrite the values of any other bits when DEC operation is enabled.

28.3.27 DRIiDEC1 Control Register (DRIiDEC1CNT)

Controls the DRIi module DEC1 internal event counter.

DRI0DEC1 Control Register (DRI0DEC1CNT)	<P4 address: location H'FFBF C03C>
DRI1DEC1 Control Register (DRI1DEC1CNT)	<P4 address: location H'FFBF D03C>
DRI2DEC1 Control Register (DRI2DEC1CNT)	<P4 address: location H'FFBF E03C>

Bit:	7	6	5	4	3	2	1	0
	DEC1 MOD	—	DEC1 CS		DEC1 EXT			DEC1 EN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	DEC1MOD	0	R	W	DEC1 Operating Mode Selection Bit Selects the DRIiDEC1 counter (DRIiDEC1CT) operating mode. 0: One-shot mode 1: Continuous operating mode
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	DEC1CS	00	R	W	DEC1 Counter Event Selection Bits Selects the event that will be used as the DRIiDEC1CT counter count source. If an event is detected from the factor selected in the state where the DEC1EN bit is "1", the value of the DEC1 counter (DEC1CT) is decremented by "1". 00: DIN1 event detection 01: Setting prohibited 10: DIN3 event detection 11: DEC0 underflow
3 to 1	DEC1EXT	000	R	W	DEC1 Counter Enable Factor Selection Bits To enable counting the DRIiDEC1 counter (DRIiDEC1CT) by external events, use this field to select the count enable factor. When an event is detected from the selected factor, the DEC1EN (DEC1 counter enable) bit is set to "1". 000: External factors disabled 001: DIN0 event detection 010: DIN1 event detection 011: DEC0 underflow 100: Acquisition enable 101: PDAC event H 11x: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
0	DEC1EN	0	R	W	DEC1 Counter Enable Bit Controls the enabled/disabled state of DEC1 counter operation. This bit can be set to "1" by an external event. Note also that if one-shot mode is selected as the operating mode, this bit will be cleared to "0" when the DEC1 counter underflows. 0: Counting disabled 1: Counting enabled [Conditions for clearing to "0"] • When the software writes "0" • When one-shot mode is selected and the DEC1 counter underflows [Conditions for setting to "1"] • When the software writes "1" • When the event selected with the DEC1EXT bits occurs Notes: • It is illegal for the software to set this bit to "1" when an external factor is selected with the DEC1EXT bits. • When DEC operation is enabled, the only write operation that may be performed is a write of "0" to the DEC1EN bit to clear that bit. Do not rewrite the values of any other bits when DEC operation is enabled.

28.3.28 DRIiDEC2 Control Register (DRIiDEC2CNT)

Controls the DRIi module DEC2 internal event counter.

DRI0DEC2 Control Register (DRI0DEC2CNT)	<P4 address: location H'FFBF C042>
DRI1DEC2 Control Register (DRI1DEC2CNT)	<P4 address: location H'FFBF D042>
DRI2DEC2 Control Register (DRI2DEC2CNT)	<P4 address: location H'FFBF E042>

Bit:	7	6	5	4	3	2	1	0
	DEC2 MOD	—	DEC2 CS		DEC2 EXT			DEC2 EN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	DEC2MOD	0	R	W	DEC2 Operating Mode Selection Bit Selects the DRIiDEC2 counter (DRIiDEC2CT) operating mode. 0: One-shot mode 1: Continuous operating mode
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	DEC2CS	00	R	W	DEC2 Counter Event Selection Bits Selects the event that will be used as the DRIiDEC2 counter (DRIiDEC2CT) count source. If an event is detected from the factor selected in the state where the DEC2EN (DEC2 counter enable) bit is "1", the value of the DEC2 counter (DEC2CT) is decremented by "1". 00: DIN1 event detection 01: Setting prohibited 10: DIN3 event detection 11: Acquisition event
3 to 1	DEC2EXT	000	R	W	DEC2 Counter Enable Factor Selection Bits To enable counting the DRIiDEC2 counter (DRIiDEC2CT) by external events, use this field to select the count enable factor. When an event is detected from the selected factor, the DEC2EN (DEC2 counter enable) bit is set to "1". 000: External factors disabled 001: DIN0 event detection 010: DIN1 event detection 011: Setting prohibited 100: Acquisition enable 101: PDAC event G 11x: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
0	DEC2EN	0	R	W	DEC2 Counter Enable Bit Controls the enabled/disabled state of DEC2 counter operation. This bit can be set to "1" by an external event. Note also that if one-shot mode is selected as the operating mode, this bit will be cleared to "0" when the DEC2 counter underflows. 0: Counting disabled 1: Counting enabled [Conditions for clearing to "0"] • When the software writes "0" • When one-shot mode is selected and the DEC2 counter underflows. [Conditions for setting to "1"] • When the software writes "1" • When the event selected with the DEC2EXT bits occurs Notes: • It is illegal for the software to set this bit to "1" when an external factor is selected with the DEC2EXT bits. • When DEC operation is enabled, the only write operation that may be performed is a write of "0" to the DEC2EN bit to clear that bit. Do not rewrite the values of any other bits when DEC operation is enabled.

28.3.29 DRIiDEC3 Control Register (DRIiDEC3CNT)

Controls the DRIi module DEC3 internal event counter.

DRI0DEC3 Control Register (DRI0DEC3CNT)	<P4 address: location H'FFBF C048>
DRI1DEC3 Control Register (DRI1DEC3CNT)	<P4 address: location H'FFBF D048>
DRI2DEC3 Control Register (DRI2DEC3CNT)	<P4 address: location H'FFBF E048>

Bit:	7	6	5	4	3	2	1	0
	DEC3 MOD	—	DEC3 CS		DEC3 EXT			DEC3 EN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	DEC3MOD	0	R	W	DEC3 Operating Mode Selection Bit Selects the DRIiDEC3 counter (DRIiDEC3CT) operating mode. 0: One-shot mode 1: Continuous operating mode
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	DEC3CS	00	R	W	DEC3 Counter Event Selection Bits Selects the event that will be used as the DRIiDEC3 counter (DRIiDEC3CT) count source. If an event is detected from the factor selected in the state where the DEC3EN (DEC3 counter enable) bit is "1", the value of the DEC3 counter (DEC3CT) is decremented by "1". 00: Setting prohibited 01: DIN3 event detection 10: DIN4 event detection 11: DIN5 event detection
3 to 1	DEC3EXT	000	R	W	DEC3 Counter Enable Factor Selection Bits To enable counting the DRIiDEC3 counter (DRIiDEC3CT) by external events, use this field to select the count enable factor. When an event is detected from the selected factor, the DEC3EN (DEC3 counter enable) bit is set to "1". 000: External factors disabled 001: DIN0 event detection 010: DIN1 event detection 011: DEC2 underflow 100: Acquisition enable 101: Setting prohibited 11x: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
0	DEC3EN	0	R	W	DEC3 Counter Enable Bit Controls the enabled/disabled state of DEC3 counter operation. This bit can be set to "1" by an external event. Note also that if one-shot mode is selected as the operating mode, this bit will be cleared to "0" when the DEC3 counter underflows. 0: Counting disabled 1: Counting enabled [Conditions for clearing to "0"] • When the software writes "0" • When one-shot mode is selected and the DEC3 counter underflows [Conditions for setting to "1"] • When the software writes "1" • When the event selected with the DEC3EXT bits occurs Notes: • It is illegal for the software to set this bit to "1" when an external factor is selected with the DEC3EXT bits. • When DEC operation is enabled, the only write operation that may be performed is a write of "0" to the DEC3EN bit to clear that bit. Do not rewrite the values of any other bits when DEC operation is enabled.

28.3.30 DRIiDEC4 Control Register (DRIiDEC4CNT)

Controls the DRIi module DEC4 internal event counter.

DRI0DEC4 Control Register (DRI0DEC4CNT)	<P4 address: location H'FFBF C04E>
DRI1DEC4 Control Register (DRI1DEC4CNT)	<P4 address: location H'FFBF D04E>
DRI2DEC4 Control Register (DRI2DEC4CNT)	<P4 address: location H'FFBF E04E>

Bit:	7	6	5	4	3	2	1	0
	DEC4 MOD	—	DEC4 CS		DEC4 EXT			DEC4 EN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	DEC4MOD	0	R	W	DEC4 Operating Mode Selection Bit Selects the DRIiDEC4 counter (DRIiDEC4CT) operating mode. 0: One-shot mode 1: Continuous operating mode
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	DEC4CS	00	R	W	DEC4 Counter Event Selection Bits Selects the event that will be used as the DRIiDEC4 counter (DRIiDEC4CT) count source. If an event is detected from the factor selected in the state where the DEC4EN (DEC4 counter enable) bit is "1", the value of the DEC4 counter (DEC4CT) is decremented by "1". 00: DIN1 event detection 01: Acquisition event 10: DRIi single transfer complete 11: DRIi acquisition event counter underflow
3 to 1	DEC4EXT	000	R	W	DEC4 Counter Enable Factor Selection Bits To enable counting the DRIiDEC4 counter (DRIiDEC4CT) by external events, use this field to select the count enable factor. When an event is detected from the selected factor, the DEC4EN (DEC4 counter enable) bit is set to "1". 000: External factors disabled 001: DIN0 event detection 010: DIN1 event detection 011: DEC3 underflow 100: Acquisition enable 101: PDAC event H 11x: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
0	DEC4EN	0	R	W	DEC4 Counter Enable Bit Controls the enabled/disabled state of DEC4 counter operation. This bit can be set to "1" by an external event. Note also that if one-shot mode is selected as the operating mode, this bit will be cleared to "0" when the DEC4 counter underflows. 0: Counting disabled 1: Counting enabled [Conditions for clearing to "0"] • When the software writes "0" • When one-shot mode is selected and the DEC4 counter underflows [Conditions for setting to "1"] • When the software writes "1" • When the event selected with the DEC4EXT bits occurs Notes: • It is illegal for the software to set this bit to "1" when an external factor is selected with the DEC4EXT bits. • When DEC operation is enabled, the only write operation that may be performed is a write of "0" to the DEC4EN bit to clear that bit. Do not rewrite the values of any other bits when DEC operation is enabled.

28.3.31 DRIiDEC5 Control Register (DRIiDEC5CNT)

Controls the DRIi module DEC5 internal event counter.

DRI0DEC5 Control Register (DRI0DEC5CNT)	<P4 address: location H'FFBF C054>
DRI1DEC5 Control Register (DRI1DEC5CNT)	<P4 address: location H'FFBF D054>
DRI2DEC5 Control Register (DRI2DEC5CNT)	<P4 address: location H'FFBF E054>

Bit:	7	6	5	4	3	2	1	0
	DEC5 MOD	—	DEC5 CS		DEC5 EXT			DEC5 EN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	DEC5MOD	0	R	W	DEC5 Operating Mode Selection Bit Selects the DRIiDEC5 counter (DRIiDEC5CT) operating mode. 0: One-shot mode 1: Continuous operating mode
6	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
5, 4	DEC5CS	00	R	W	DEC5 Counter Event Selection Bits Selects the event that will be used as the DRIiDEC5 counter (DRIiDEC5CT) count source. If an event is detected from the factor selected in the state where the DEC5EN (DEC5 counter enable) bit is "1", the value of the DEC5 counter (DEC5CT) is decremented by "1". 00: DIN0 event detection 01: DIN1 event detection 10: DIN3 event detection 11: DIN4 event detection
3 to 1	DEC5EXT	000	R	W	DEC5 Counter Enable Factor Selection Bits To enable counting the DRIiDEC5 counter (DRIiDEC5CT) by external events, use this field to select the count enable factor. When an event is detected from the selected factor, the DEC5EN (DEC5 counter enable) bit is set to "1". 000: External factors disabled 001: DIN0 event detection 010: DIN1 event detection 011: DEC2 underflow 100: Acquisition enable 101: Setting prohibited 11x: Setting prohibited

Bit	Abbreviation	After Reset	R	W	Description
0	DEC5EN	0	R	W	DEC5 Counter Enable Bit Controls the enabled/disabled state of DEC5 counter operation. This bit can be set to "1" by an external event. Note also that if one-shot mode is selected as the operating mode, this bit will be cleared to "0" when the DEC5 counter underflows. 0: Counting disabled 1: Counting enabled [Conditions for clearing to "0"] • When the software writes "0" • When one-shot mode is selected and the DEC5 counter underflows [Conditions for setting to "1"] • When the software writes "1" • When the event selected with the DEC5EXT bits occurs Notes: • It is illegal for the software to set this bit to "1" when an external factor is selected with the DEC5EXT bits. • When DEC operation is enabled, the only write operation that may be performed is a write of "0" to the DEC5EN bit to clear that bit. Do not rewrite the values of any other bits when DEC operation is enabled.

28.3.32 DRIiDEC0 to DRIiDEC5 Reload Registers (DRIiDEC0RLD to DRIiDEC5RLD)

The DRIiDEC0 to DRIiDEC5 reload registers are provided to reload data into the DECm counters. The contents of a reload register are reloaded into a counter under the following conditions.

- In one-shot mode when the state changes from count disabled to count enabled.
- In continuous mode when the DECm counter underflows.

Legend: m = 0 to 5

DRI0DEC0 Reload Register (DRI0DEC0RLD)	<P4 address: location H'FFBF C038>
DRI1DEC0 Reload Register (DRI1DEC0RLD)	<P4 address: location H'FFBF D038>
DRI2DEC0 Reload Register (DRI2DEC0RLD)	<P4 address: location H'FFBF E038>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC0RLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC1 Reload Register (DRI0DEC1RLD)	<P4 address: location H'FFBF C03E>
DRI1DEC1 Reload Register (DRI1DEC1RLD)	<P4 address: location H'FFBF D03E>
DRI2DEC1 Reload Register (DRI2DEC1RLD)	<P4 address: location H'FFBF E03E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC1RLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC2 Reload Register (DRI0DEC2RLD)	<P4 address: location H'FFBF C044>
DRI1DEC2 Reload Register (DRI1DEC2RLD)	<P4 address: location H'FFBF D044>
DRI2DEC2 Reload Register (DRI2DEC2RLD)	<P4 address: location H'FFBF E044>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC2RLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC3 Reload Register (DRI0DEC3RLD)	<P4 address: location H'FFBF C04A>
DRI1DEC3 Reload Register (DRI1DEC3RLD)	<P4 address: location H'FFBF D04A>
DRI2DEC3 Reload Register (DRI2DEC3RLD)	<P4 address: location H'FFBF E04A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC3RLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC4 Reload Register (DRI0DEC4RLD)	<P4 address: location H'FFBF C050>
DRI1DEC4 Reload Register (DRI1DEC4RLD)	<P4 address: location H'FFBF D050>
DRI2DEC4 Reload Register (DRI2DEC4RLD)	<P4 address: location H'FFBF E050>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC4RLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC5 Reload Register (DRI0DEC5RLD) <P4 address: location H'FFBF C056>
DRI1DEC5 Reload Register (DRI1DEC5RLD) <P4 address: location H'FFBF D056>
DRI2DEC5 Reload Register (DRI2DEC5RLD) <P4 address: location H'FFBF E056>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC5RLD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	DECmRLD	H'0000	R	W	DECm Reload Value

Note: • This register must only be accessed in word units from a word boundary.

Legend: m = 0 to 5

28.3.33 DRIiDEC0 to DRIiDEC5 Counters (DRIiDEC0CT to DRIiDEC5CT)

The DRIiDEC0 to DRIiDEC5 counters are 16-bit down counters that count in synchronization with event detection after counting is enabled. When a DECm counter used in one-shot mode, do not write to the DRIiDEC0CT to DRIiDEC5CT counters in the state where the corresponding DECm counter is in the count enabled state.

Legend: m = 0 to 5

DRI0DEC0 Counter (DRI0DEC0CT) <P4 address: location H'FFBF C03A>
DRI1DEC0 Counter (DRI1DEC0CT) <P4 address: location H'FFBF D03A>
DRI2DEC0 Counter (DRI2DEC0CT) <P4 address: location H'FFBF E03A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC0CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC1 Counter (DRI0DEC1CT) <P4 address: location H'FFBF C040>
DRI1DEC1 Counter (DRI1DEC1CT) <P4 address: location H'FFBF D040>
DRI2DEC1 Counter (DRI2DEC1CT) <P4 address: location H'FFBF E040>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC1CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC2 Counter (DRI0DEC2CT) <P4 address: location H'FFBF C046>
DRI1DEC2 Counter (DRI1DEC2CT) <P4 address: location H'FFBF D046>
DRI2DEC2 Counter (DRI2DEC2CT) <P4 address: location H'FFBF E046>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC2CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC3 Counter (DRI0DEC3CT) <P4 address: location H'FFBF C04C>
DRI1DEC3 Counter (DRI1DEC3CT) <P4 address: location H'FFBF D04C>
DRI2DEC3 Counter (DRI2DEC3CT) <P4 address: location H'FFBF E04C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC3CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC4 Counter (DRI0DEC4CT) <P4 address: location H'FFBF C052>
DRI1DEC4 Counter (DRI1DEC4CT) <P4 address: location H'FFBF D052>
DRI2DEC4 Counter (DRI2DEC4CT) <P4 address: location H'FFBF E052>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC4CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DRI0DEC5 Counter (DRI0DEC5CT) <P4 address: location H'FFBF C058>
DRI1DEC5 Counter (DRI1DEC5CT) <P4 address: location H'FFBF D058>
DRI2DEC5 Counter (DRI2DEC5CT) <P4 address: location H'FFBF E058>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEC5CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	DECmCT	H'0000	R	W	DECm Counter

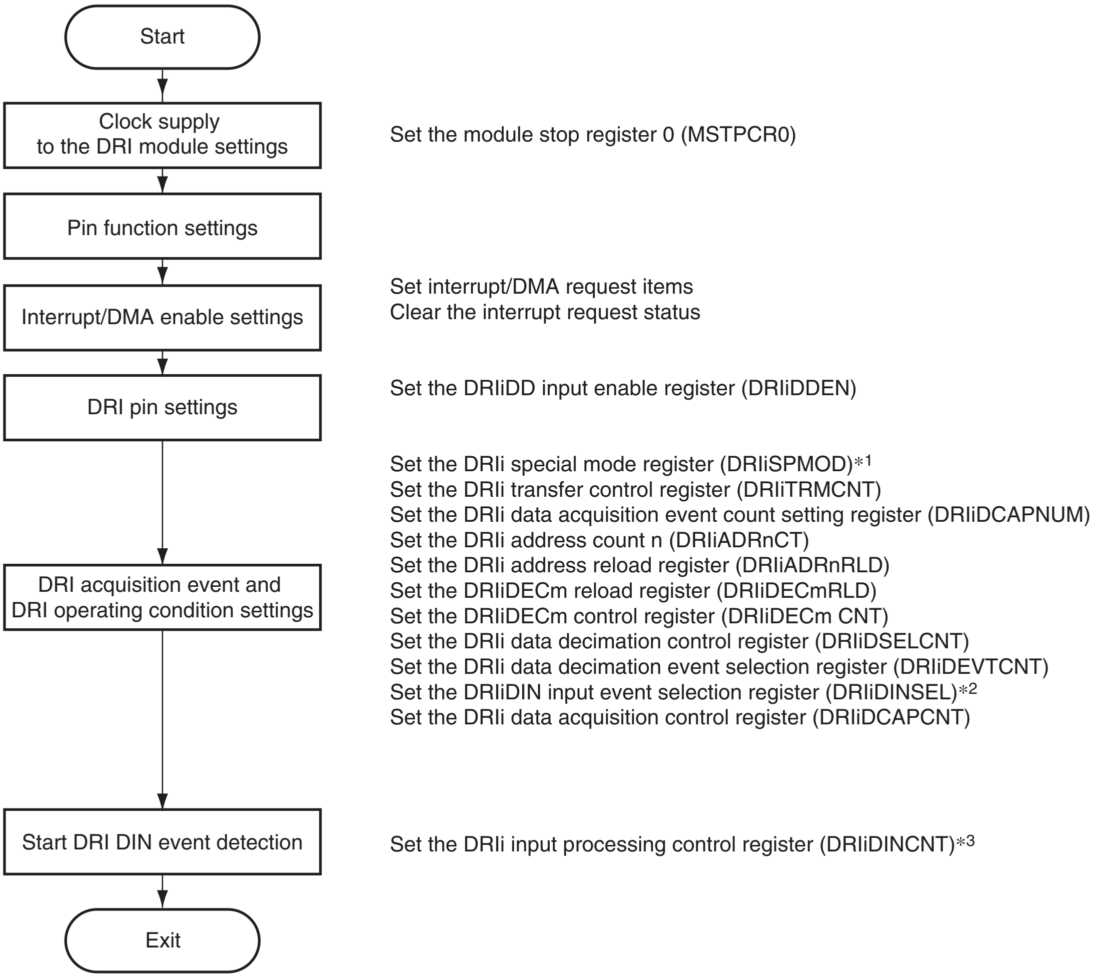
Note: • This register must only be accessed in word units from a word boundary.

Legend: m = 0 to 5

28.4 Operation

28.4.1 DRI Initialization Flowchart

Figure 28.8 shows the DRI initialization flowchart.


Start
Clock supply to the DRI module settings
Pin function settings
Interrupt/DMA enable settings
DRI pin settings
DRI acquisition event and DRI operating condition settings
Start DRI DIN event detection
Exit

Set the module stop register 0 (MSTPCR0)

Set interrupt/DMA request items
Clear the interrupt request status

Set the DRIiDD input enable register (DRIiDDEN)

Set the DRIi special mode register (DRIiSPMOD)*1
Set the DRIi transfer control register (DRIiTRMCNT)
Set the DRIi data acquisition event count setting register (DRIiDCAPNUM)
Set the DRIi address count n (DRIiADRnCT)
Set the DRIi address reload register (DRIiADRnRLD)
Set the DRIiDECm reload register (DRIiDECmRLD)
Set the DRIiDECm control register (DRIiDECm CNT)
Set the DRIi data decimation control register (DRIiDSELCNT)
Set the DRIi data decimation event selection register (DRIiDEVTCNT)
Set the DRIiDIN input event selection register (DRIiDINSEL)*2
Set the DRIi data acquisition control register (DRIiDCAPCNT)

Set the DRIi input processing control register (DRIiDINCNT)*3

Notes: *1 Only set when used in special mode.
*2 Only set when used in DIN5.
*3 The DRIiDINCNT register is set last.
Legend: n = 0 or 1, m = 0 to 5

Figure 28.8 DRI Initialization Flowchart

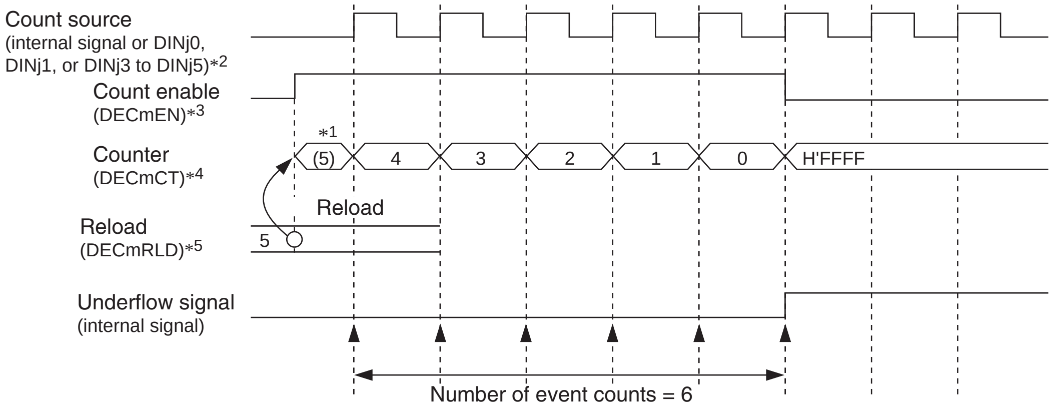
28.4.2 Event Counter (DEC) Operating Modes

(1) One-shot mode

In this mode, the contents of the DRIiDECm reload register (DRIiDECmRLD) are loaded into the DRIiDECm counter (DRIiDECmCT) when the DECmEN (DECm count enable) bit in the DRIiDECm control registers (DRIiDECmCNT) changes from disabled to enabled. From that point on, the counter is decremented each time the event selected by the DECmCS (DECm count event selection) bits occurs. When exactly 1 plus the DECm reload register set value (DRIiEDCmRLD) events have been counted, the count operation stops in the underflow state (counter value = H'FFFF) and the DECmEN (DECm count enable) bit is cleared to "0".

Notes:
- The reload value that is reloaded into the counter when counting is enabled cannot be read out. If read, the value returned will be count value before the reload.
- If the count is enabled by an external event and the count source occurs at the same time, while the set to "1" of the count enable bit by the external event will be performed, the count operation will not be performed.
- If a counter stop due to underflow and a count enable by an external event occur at the same time, the count stop by underflow takes precedence.
- If the count is enabled by an external event and a write to stop the count occur at the same time, the count stop operation takes precedence.

Legend: m = 0 to 5



Notes: *1 The reload value that is reloaded into the counter when counting is enabled cannot be read out. If read, the value returned will be count value before the reload.

*2 Count source indicates the event selected by the DECmCS bits (DECm count event selection) in the DRIiDECm control register (DRIiDECmCNT). Events DIN0 to DIN4 can be input via the DINj0 to DINj4 pins. For the relations between the signals and the pins, refer to section 28.6, Relations between DRI Pins and DRI Module Signals. Use the port DRI input channel switching register (PDRIR) (described in section 18.3.22) to appropriately select the DRI pins.

*3 Count enable indicates the DECmEN bit (DECm count enable) in the DRIiDECm control register (DRIiDECmCNT).

*4 Counter indicates the DECmCT bits (DECm counter) in the DRIiDECm counter (DRIiDECmCT).

*5 Reload indicates the DECmRLD bits (DECm reload value) in the DRIiDECm reload register (DRIiDECmRLD).

- This figure does not include detailed timings.

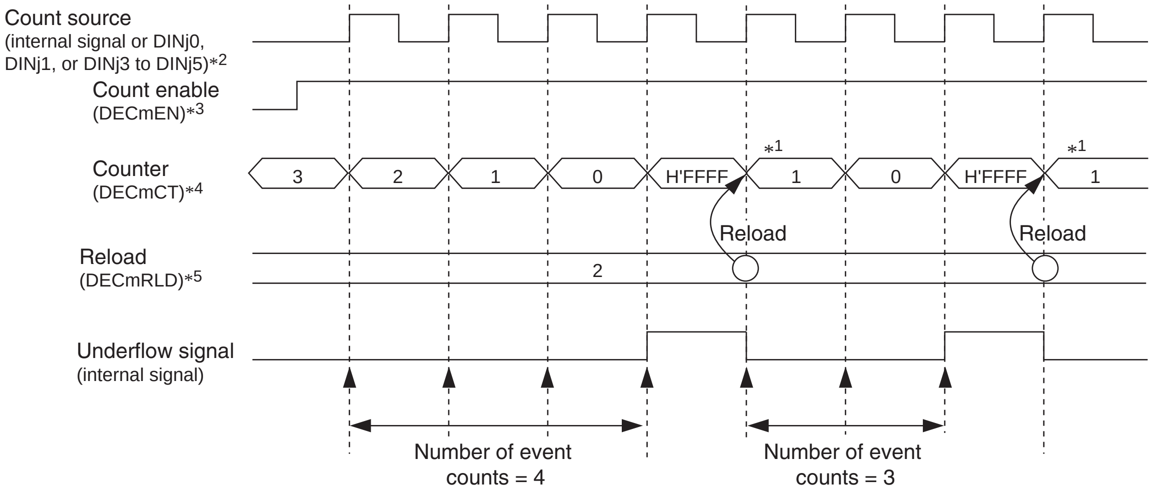
Figure 28.9 DEC One-Shot Mode Count Example

(2) Continuous mode

When the DECmEN (DECm count enable) bit is enabled, each time the event selected with the DECmCS (DECm count event selection) bits occurs the DRIiDECm counter (DRIiDECmCT) set value is decremented and the DRIiDECm reload register (DRIiDECmRLD) value is reloaded on the DECm counter underflow (counter value = H'FFFF). From that point on, this operation is repeated on each DRIiDECm counter (DRIiDECmCT) underflow.

Notes:
- If the count is enabled by an external event and the count source occurs at the same time, while the set to "1" of the count enable bit by the external event will be performed, the count operation will not be performed.
- If a count enable by an external event and a count disable write to the DECmEN (count enable) bit occur at the same time, the count disable operation takes precedence.
- If a reload and a write to the counter occur at the same time, the write to the counter takes precedence. At this time, an interrupt due to a DECm counter underflow will not occur.
- If a counter source and a write to the counter occur at the same time, the write to the counter takes precedence. At this time, the counter source is ignored.

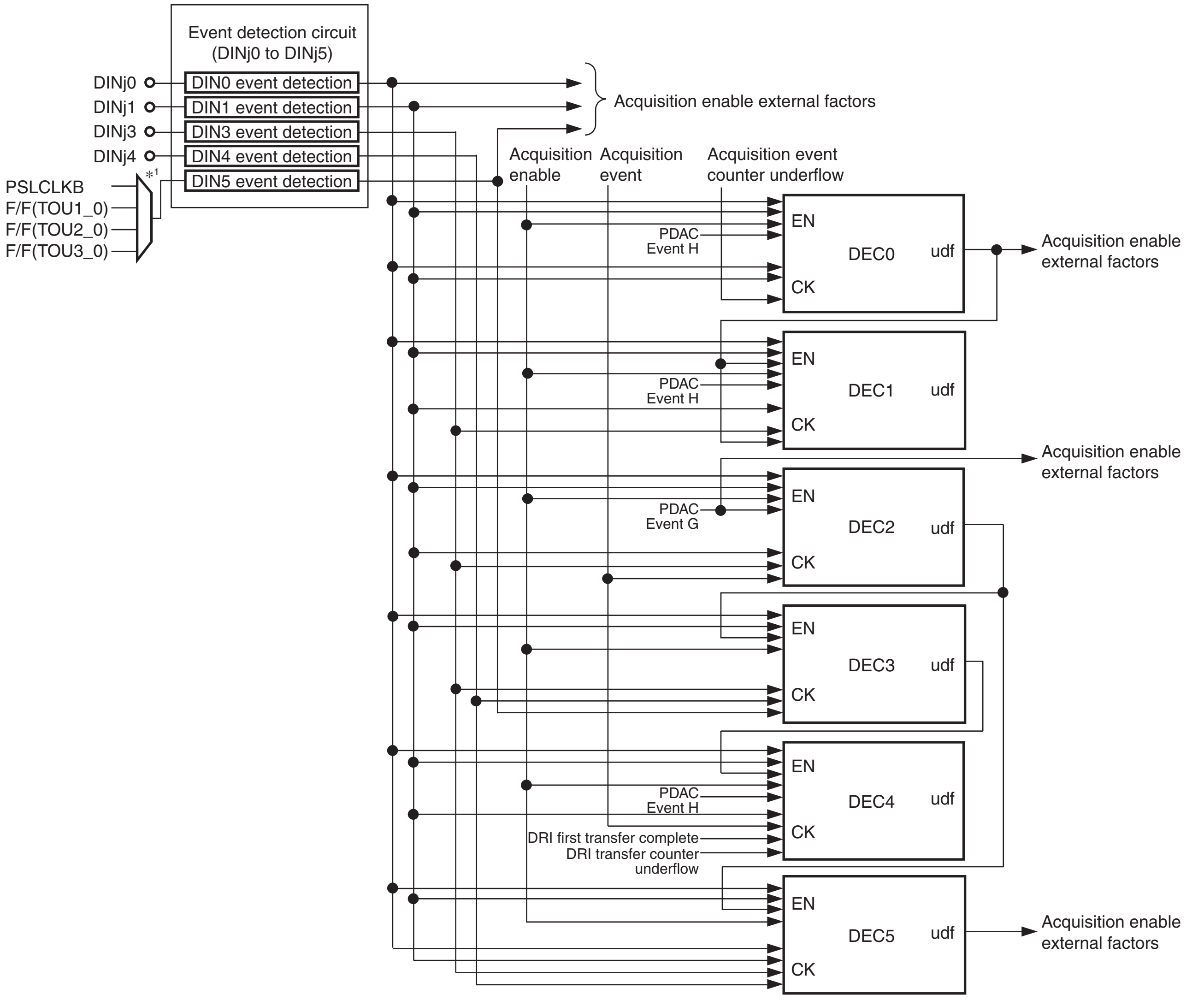
Legend: m = 0 to 5



Notes: *1 The reload value minus 1 is reloaded to the counter using the next count clock following the DEC underflow.
*2 Count source indicates the event selected by the DECmCS bits (DEC count event selection) in the DRIiDECm control register (DRIiDECmCNT). Events DIN0 to DIN4 can be input via the DINj0 to DINj4 pins. For the relations between the signals and the pins, refer to section 28.6, Relations between DRI Pins and DRI Module Signals. Use the port DRI input channel switching register (PDRIR) (described in section 18.3.22) to appropriately select the DRI pins.
*3 Count enable indicates the DECmEN bit (DECm count enable) in the DRIiDECm control register (DRIiDECmCNT).
*4 Counter indicates the DECmCT bits (DECm counter) in the DRIiDECm counter (DRIiDECmCT).
*5 Reload indicates the DECmRLD bits (DECm reload value) in the DRIiDECm reload register (DRIiDECmRLD).
This figure does not include detailed timings.

Figure 28.10 DEC Continuous Mode Count Example

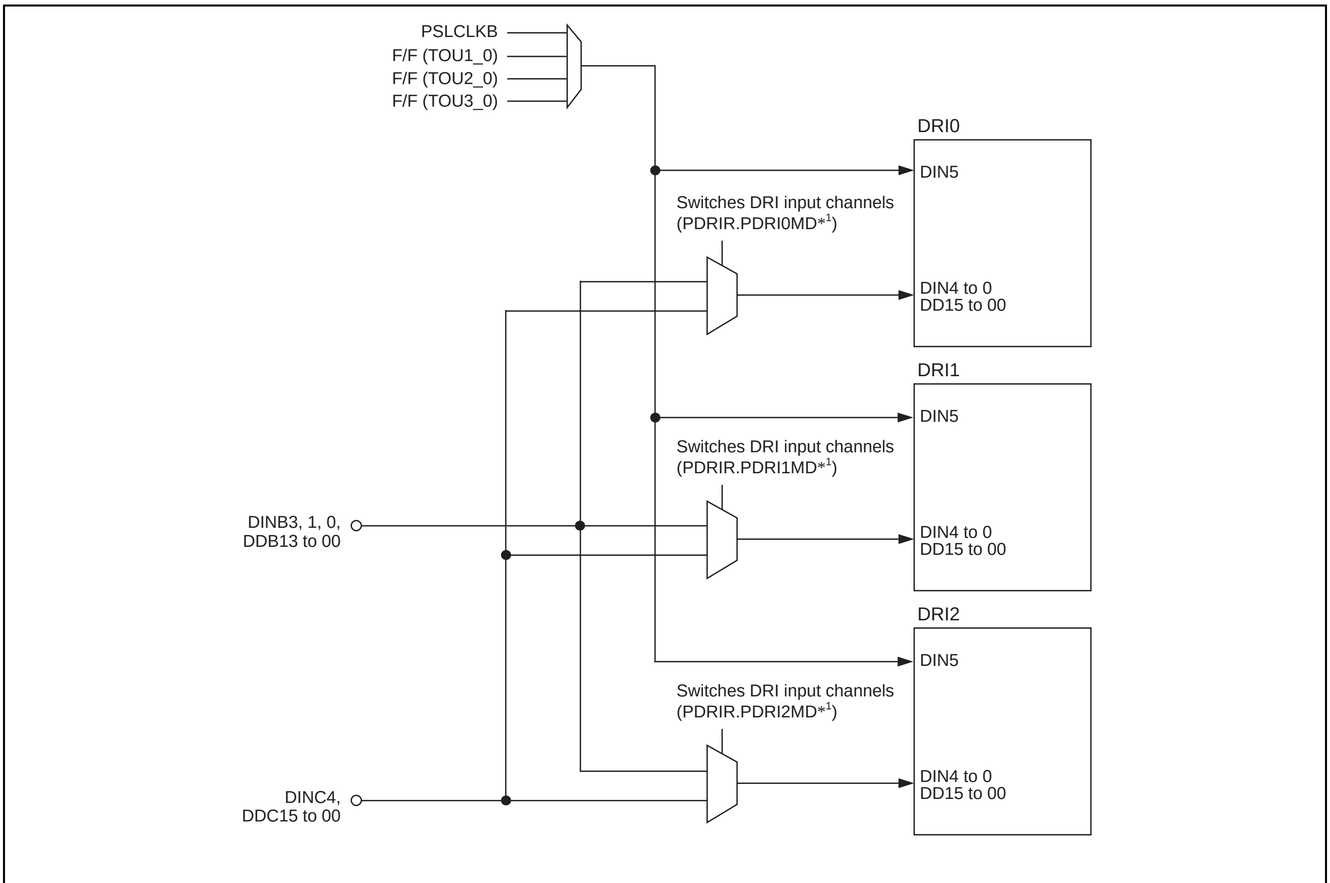
28.5 DEC Connection Diagram



Note: *1 The clock output from PSEL or the F/F output from timer TOU of ATU-IIIS is input as an internal signal to the DIN5 signal. For example, when F/F (TOU1_0) is selected in the DRIiDINSEL register, the value output by timer TOU1_0 (the value of the FFDT10 bit in TO1FFDR) is input to DIN5 as an internal signal.

Figure 28.11 DEC Connection Diagram

28.6 Relations between DRI Pins and DRI Module Signals



Note: *1 Use the port DRI input channel switching register (PDRIR) (described in section 18.3.22) to appropriately select the DRI pins.

Figure 28.12 Connections between DRI Pins and DRI Module Signals

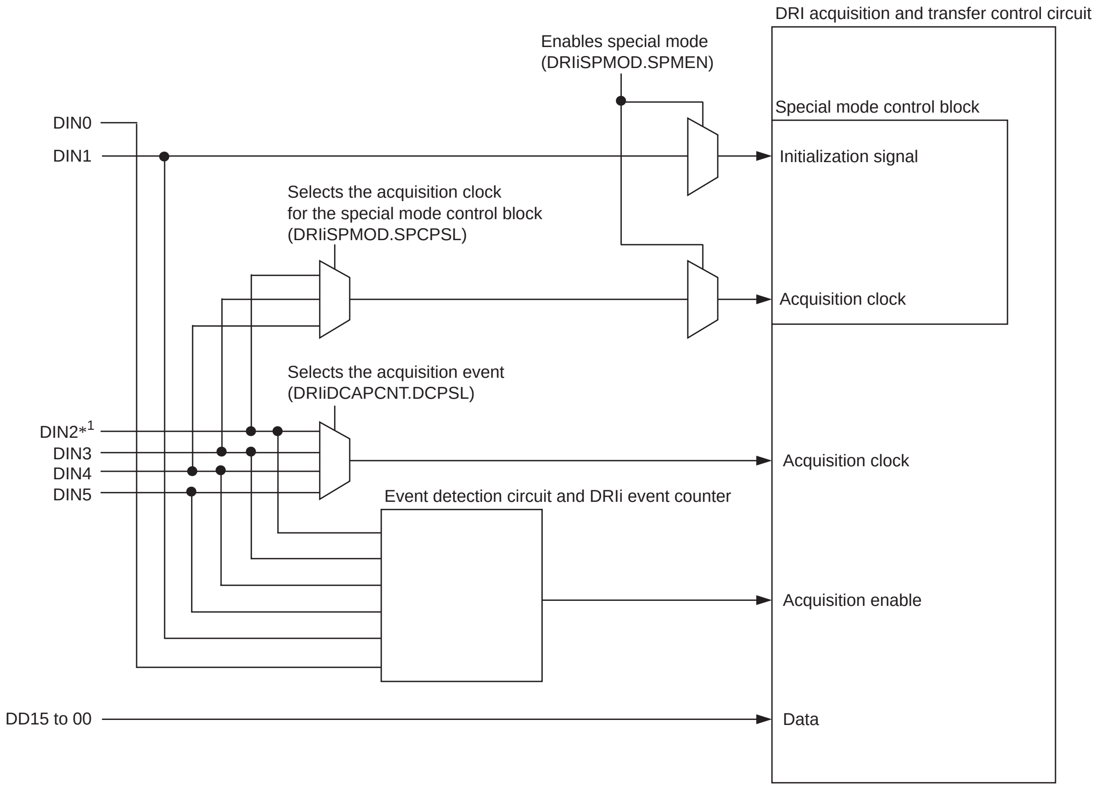
28.7 DRIi Special Mode

Faster data acquisition is possible if DRIi special mode is turned on (enabled) through the DRIi special mode register (DRIiSPMOD). Table 28.7 compares the operation when special mode is on to the operation when off. Figure 28.13 shows the signal connections to turn on/off the special mode.

Table 28.7 Comparison between Operation when Special Mode is On to Operation when Off

Item	Special Mode is On	Special Mode is Off
Data acquisition clock	Either DIN3 or DIN4 selected*[1]	Either DIN3, DIN4, or DIN5 selected
Special mode control block initialization signal	DIN1*[1]	—
Data acquisition edge	Either rising or falling edges selected	Either rising edges, falling edges, or both edges selected
Minimum data acquisition period	2 tc (PAck)	3.5 tc (PAck)

Note: *1 The DIN1, DIN3, or DIN4 clock signal can be input via the DINj1, DINj3, or DINj4 pin, respectively. For the relations between the signals and the pins, refer to section 28.6, Relations between DRI Pins and DRI Module Signals. Use the port DRI input channel switching register (PDRIR) (described in section 18.3.22) to appropriately select the DRI pins.


DRI acquisition and transfer control circuit
Enables special mode
(DRIiSPMOD.SPMEN)
Special mode control block
DIN0
DIN1
Initialization signal
Selects the acquisition clock
for the special mode control block
(DRIiSPMOD.SPCPSL)
Acquisition clock
Selects the acquisition event
(DRIiDCAPCNT.DCPSL)
DIN2*1
DIN3
DIN4
DIN5
Acquisition clock
Event detection circuit and DRIi event counter
Acquisition enable
DD15 to 00
Data

Note: *1 DIN2 is not available because no DRI pins provide input to DIN2.

Figure 28.13 Signal Connections to Turn on/off Special Mode

28.8 Usage Notes

28.8.1 Module Stop Function Setting before Using the DRI

To use the DRI channel i, set the DRIi bit in the module stop register 0 (MSTPCR0) to "0" to enable the DRIi operation, and then set the DRIi related register. Otherwise, the clocks are not supplied to the DRIi module and DRIi operation is disabled even though the DRIi related register is set.

28.8.2 Contention between the DRO/DRI Module and the SuperHyway Bus Master

DRI0 to DRI2 and DRO use (and share) a dedicated DRO/DRI bus separate from the SuperHyway bus to access SHwyRAM. Access contention occurs when accesses from the DRI0 to DRI2 and DRO occur at the same time or overlap.

When contention occurs on the DRI0 to DRI2 and DRO, DRI0 has the highest priority and the rest is fixed in the following order.

- DRI0 > DRI1 > DRI2 > DRO

Note also that since a dedicated DRO/DRI bus connection is used to connect DRI0 to DRI2 and DRO, access contention with the SuperHyway bus master (CPU, DMA, and other units), normally will not occur. However, access contention does occur for the same 64-Kbyte area in SHwyRAM when accesses (read/write) from a dedicated DRI/DRO bus and the SuperHyway bus occur at the same time or overlap.

When access contention occurs, arbitration is performed in the following order.

- SuperHyway bus > dedicated DRO/DRI bus

28.8.3 Number of Acquisition Events in Special Mode

Although an arbitrary number of data acquisition events can be specified when normal mode is used, in special mode, there are limitations on the values that can be set depending on the setting of the DRIi data acquisition control register (DRIiDCAPCNT) DWDSL (input data bus width selection) bits.

See section 28.3.20, DRIi Data Acquisition Event Count Setting Register (DRIiDCAPNUM), and set an appropriate value.

28.8.4 Registers that May Not Be Rewritten During Operation

To prevent incorrect operation, applications must not rewrite the registers listed in table 28.8 during DRIi module operation.

Table 28.8 Registers that May Not Be Rewritten During Operation

Register Name	Abbreviation	Operating State	Notes
DRI0DIN DMA transfer enable register	DRI0DINDEN	DRI	*[1]
DRI0DEC DMA transfer enable register	DRI0DECDEN	DEC	*[1]
DRI0DMA transfer enable register	DRI0TRMDEN	DRI	*[1]
DRIi transfer control register	DRIiTRMCNT	DRI	*[2]
DRIi special mode register	DRIiSPMOD	DRI/DEC	
DRIi data acquisition control register	DRIiDCAPCNT	DRI/DEC	*[3]
DRIi data decimation control register	DRIiDSELCNT	DEC	
DRIi data decimation event selection register	DRIiDEVTCNT	DEC	
DRIiDIN input event selection register	DRIiDINSEL	DRI/DEC	
DRIiDD input enable register	DRIiDDEN	DRI	
DRIi data acquisition event count setting register	DRIiDCAPNUM	DRI	
DRIi address reload register 0	DRIiADR0RLD	DRI	
DRIi address counter 0	DRIiADR0CT	DRI	
DRIi address reload register 1	DRIiADR1RLD	DRI	
DRIi address counter 1	DRIiADR1CT	DRI	
DRIiDIN input processing control register	DRIiDINCNT	DRI/DEC	
DRIiDEC0 control register	DRIiDEC0CNT	DEC	*[4]
DRIiDEC1 control register	DRIiDEC1CNT	DEC	*[4]
DRIiDEC2 control register	DRIiDEC2CNT	DEC	*[4]
DRIiDEC3 control register	DRIiDEC3CNT	DEC	*[4]
DRIiDEC4 control register	DRIiDEC4CNT	DEC	*[4]
DRIiDEC5 control register	DRIiDEC5CNT	DEC	*[4]

Notes: *1 When DMA is not used (DE = "0"), only setting DE to "1" is allowed.

*2 Only a rewrite that changes the DRST (DRI reset) bit from "1" to "0" is allowed.

*3 Only a rewrite that changes the DCPEN (acquisition enable) bit from "1" to "0" is allowed.

*4 Rewrite is only allowed in continuous mode. Changes are illegal in one-shot mode.

- Registers marked DRI in the operating state column may not be rewritten when DRI acquisition is enabled, and register marked DEC may not be rewritten when DEC operation is enabled.
- The DRI acquisition enabled state is the state where the DRIiDCAPCNT register DCPEN bit is "1", and the DEC operation enabled state is the state where the DRIiDECnCNT register DECnEN bit is "1".

Legend: n = 0 to 5

Section 29 Direct RAM Output Interface (DRO)

29.1 Overview

The direct RAM output interface (DRO) module provides a parallel interface for the output of RAM data to external systems. Along with the data, the DRO module also outputs a strobe signal that indicates the sampling timing. Since data readout from SuperHyway is performed with a dedicated bus used only by the DRI and DRO modules independently of the CPU and DMAC, this module can output data efficiently without interfering with CPU or DMAC operation.

Table 29.1 lists the overview of the DRO module.

Table 29.1 DRO Module Overview

Item	Description
Transfer method	Parallel strobed output
Access area	SHwyRAM area (256 Kbytes)
Output data width	Either 8-bits or 16-bits
Maximum transfer clock	10 MHz
Maximum transfer rate	20 Mbytes/s (when 16 bits is selected, Pck = 40 MHz)
Strobe polarity	Either "H" active or "L" active may be selected.
Timing adjustment function	The setup and hold times can be programmed in 1Pck units relative to the strobe signal edge.
Interrupt request	An interrupt request is generated after a prespecified number of data items have been output.

Figure 29.1 shows the block diagram of the DRO module.

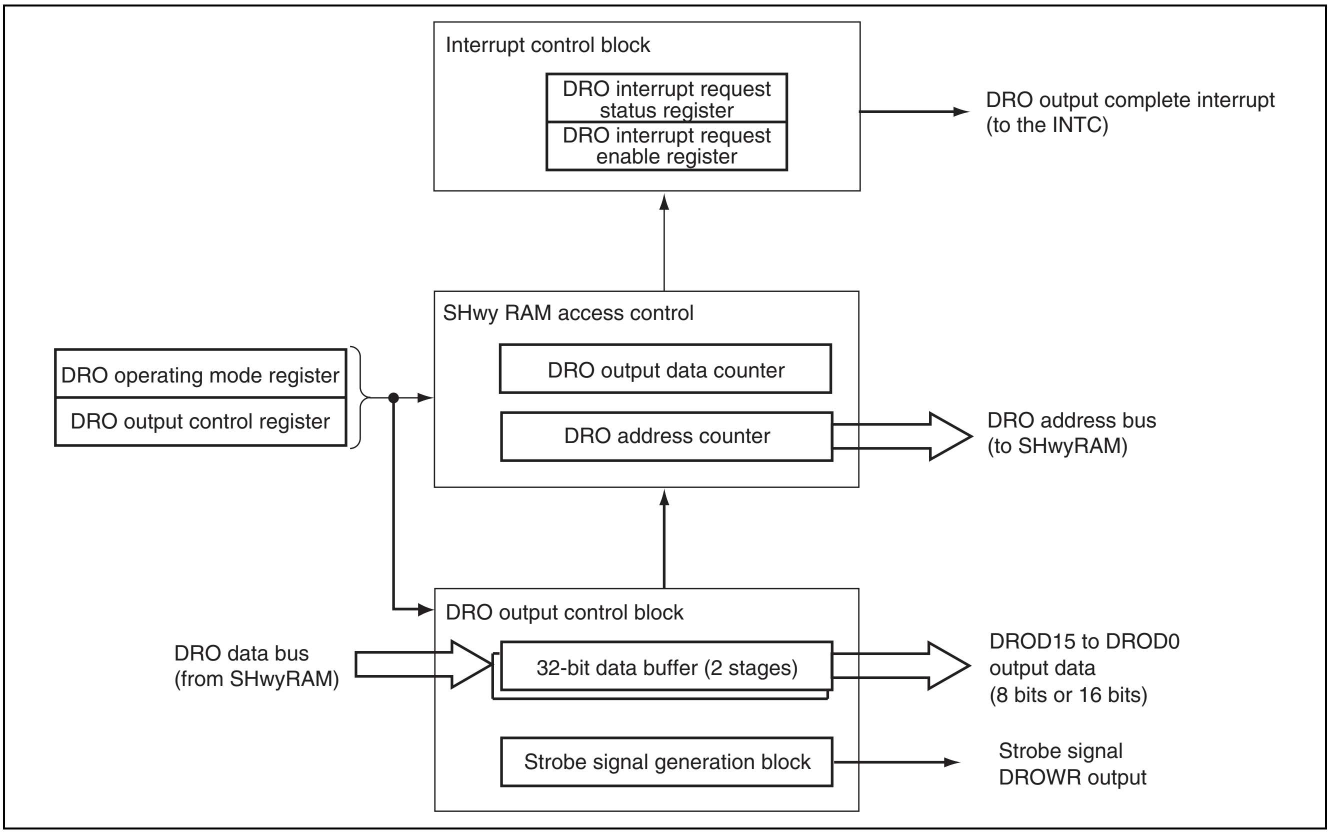

Interrupt control block
DRO interrupt request status register
DRO interrupt request enable register
DRO output complete interrupt (to the INTC)
SHwy RAM access control
DRO output data counter
DRO address counter
DRO address bus (to SHwyRAM)
DRO operating mode register
DRO output control register
DRO output control block
DRO data bus (from SHwyRAM)
32-bit data buffer (2 stages)
DROD15 to DROD0 output data (8 bits or 16 bits)
Strobe signal generation block
Strobe signal DROWR output

Figure 29.1 Block Diagram of DRO

29.2 Input/Output Pins

Table 29.2 lists the DRO module pin configuration.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 29.2 Pin Configuration

Pin Name	I/O	Function
DROD15 to DROD0	Output	Output data bus
DROWR	Output	Output data strobe

29.3 Register Descriptions

Table 29.3 lists the DRO module registers.

Table 29.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
DRO interrupt request status register	DROIST	H'00	H'FFFE F000	8, 16	29-4
DRO interrupt request enable register	DROIEN	H'00	H'FFFE F001	8, 16	29-4
DRO operating mode register	DROMOD	H'0000	H'FFFE F004	16	29-5
DRO output control register	DROCNT	H'00	H'FFFE F006	8	29-7
DRO output data counter	DRODCT	H'0000 0000	H'FFFE F008	32	29-8
DRO address counter	DROADRCT	H'0000 0000	H'FFFE F00C	32	29-9

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

The DRO control registers (registers DROIST and DROIEN) control the interrupt request signal to be output to the interrupt controller from the DRO module.

- Interrupt request status bits

  These status bits are used to determine the interrupt request that occurred and are set to "1" when the corresponding interrupt request occurs. These bits cannot be set to "1" in software. These status bits are cleared by writing a "0" and once a "1" has been written the state of the status bit is retained. Note that since this operation is not influenced by the interrupt request enable bits, they can also be used to verify the operation of peripheral functions. When handling an interrupt, of the grouped interrupt request status bits, the handler must only clear those status bits that correspond to the interrupts it is actually processing. If an interrupt handler clears status bits for interrupts it is not handling, interrupts that have not been processed will be cleared.

- Interrupt request enable bits

  These flags are used to disable the interrupts in a grouped set of interrupts that are not used. To enable an interrupt request, set the corresponding flag to "1" and to disable an interrupt request, set the flag to "0".

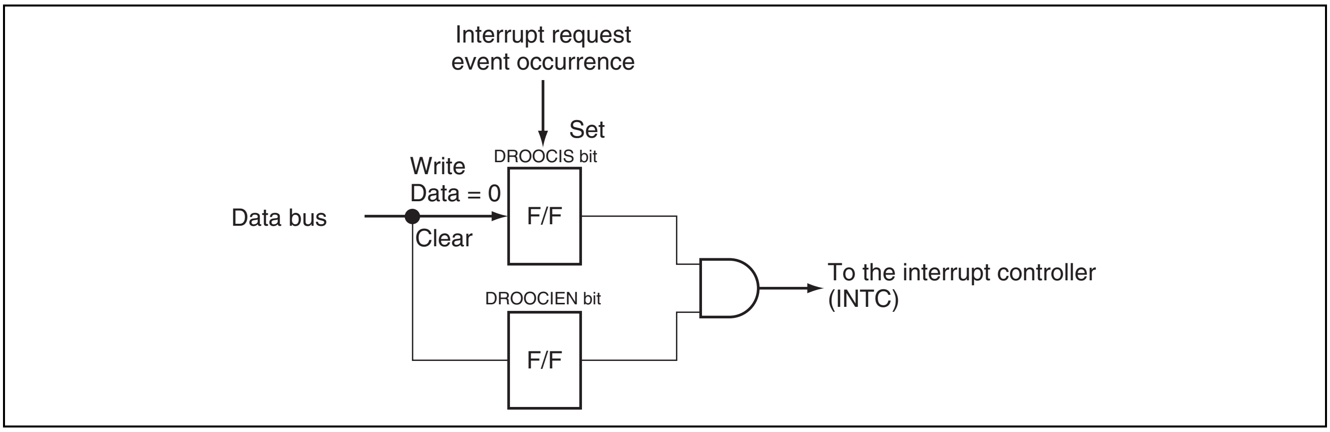

Interrupt request event occurrence
Set
DROOCIS bit
Write Data = 0
Data bus
Clear
F/F
DROOCIEN bit
F/F
To the interrupt controller (INTC)

Figure 29.2 Relationship Between DROIST and DROIEN Registers

29.3.1 DRO Interrupt Request Status Register (DROIST)

DRO Interrupt Request Status Register (DROIST) <P4 address: location H'FFFE F000>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DRO OCIS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	DROOCIS	0	R	*[1]	DRO output complete interrupt request status bit This bit is set to "1" when the data count set in the DRO output data counter has completed (and the value of the DRO output data counter is "1" (DRODCT) in all positions). Clear this bit by writing "0". Note that if "1" is written, this bit will retain its previous value. This bit is not influenced by DRO interrupt request enable register (DROIEN) and is always set to "1" when a data transfer completes. If the DRO output request interrupt enable bit (DROOCIEN) is set to "1", an interrupt request will be issued to the interrupt controller when a data transfer completes. If this bit is set to "1" by an all data output complete event and is cleared to "0" by software at the same time, the set to "1" operation will take precedence. 0: No interrupt has occurred 1: An interrupt has occurred

Note: *1 Only writing "0" is valid. If "1" is written, the previous value will be retained.

29.3.2 DRO Interrupt Request Enable Register (DROIEN)

DRO Interrupt Request Enable Register (DROIEN) <P4 address: location H'FFFE F001>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DRO OCIEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	DROOCIEN	0	R	W	DRO Output Complete Interrupt Request Enable Bit This bit controls the enabled/disabled state of the interrupt request to the INTC when the data count set in the DRO output data counter (DRODCT) has completed. If this bit is set to "1", the interrupt request to the INTC is enabled. 0: The interrupt request is masked (disabled) 1: The interrupt request is enabled

29.3.3 DRO Operating Mode Register (DROMOD)

DRO Operating Mode Register (DROMOD) <P4 address: location H'FFFE F004>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DROSU				DROHD				—	—	—	—	—	DRO SSL	DRO ODS	DRO RST
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	DROSU	0000	R	W	DRO Setup Time Setting Bits This field selects the output data (DROD15 to DROD0) setup time relative to the strobe signal rising edge (when "L" active is selected with the DRO strobe polarity selection bit (DROSSL)) or falling edge (when "H" active is selected with the DROSSL bit). The setup time can be set to a value in the range 1 to 16 times Pck in Pck units. 0000: 1Pck 0001: 2Pck 0010: 3Pck 0011: 4Pck 0100: 5Pck 0101: 6Pck 0110: 7Pck 0111: 8Pck 1000: 9Pck 1001: 10Pck 1010: 11Pck 1011: 12Pck 1100: 13Pck 1101: 14Pck 1110: 15Pck 1111: 16Pck Note: The setting value of the setup time must meet the following conditions: 4 ≤ (setup time by DROSU bit) + (hold time by DROHD bit)

Bit	Abbreviation	After Reset	R	W	Description
11 to 8	DROHD	0000	R	W	DRO Hold Time Setting Bits This field selects the output data (DROD15 to DROD0) hold time relative to the strobe signal rising edge (when "L" active is selected with the DRO strobe polarity selection bit (DROSSL)) or falling edge (when "H" active is selected with the DROSSL bit). The hold time can be set to a value in the range 1 to 16 times Pck in Pck units. 0000: 1Pck 0001: 2Pck 0010: 3Pck 0011: 4Pck 0100: 5Pck 0101: 6Pck 0110: 7Pck 0111: 8Pck 1000: 9Pck 1001: 10Pck 1010: 11Pck 1011: 12Pck 1100: 13Pck 1101: 14Pck 1110: 15Pck 1111: 16Pck Note: The setting value of the hold time must meet the following conditions: 4 ≤ (setup time by DROSU bit) + (hold time by DROHD bit)
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	DROSSL	0	R	W	DRO Strobe Polarity Selection Bit This bit sets the polarity of the DROWR signal, which is output by the DRO module along with the output data. 0: "L" active In this mode, the DROWR signal is at the "H" level when data is not output. Valid data is output at the same time the DROWR signal falls. The DROWR signal then goes to the "H" level the number of cycles later set in the DRO Setup Time Setting bits (DROSU). After that, the data output is held for the number of cycles later set in the DROHD bits. 1: "H" active In this mode, the DROWR signal is at the "L" level when data is not output. Valid data is output at the same time the DROWR signal rises. The DROWR signal then goes to the "L" level the number of cycles later set in the DROSU bits. After that, the data output is held for the number of cycles later set in the DROHD bits.

Bit	Abbreviation	After Reset	R	W	Description
1	DROODS	0	R	W	DRO Output Data Width Selection Bit Selects whether the output data width is 8 bits or 16 bits. When 8 bits is selected, DROD15 to DROD8 and DROD7 to DROD0 output the same values. 0: 8 bits 1: 16 bits
0	DRORST	0	R	W	DRO Reset Bit This bit selects whether the DRO output control block is reset or the DRO module operation is enabled. when this bit is "0", the DRO module cannot output data. If this bit is cleared to "0" during DRO module operation, the DRO control block is initialized and: • If there is any data stored in the DRO module that has not been output, that output is all cancelled. • If this bit is cleared to "0" during a data output operation, the DROWR signal is switched to the inactive state and the data outputs DROD15 to DROD0 are negated. • If the DRORST bit is cleared to "0", the DRO output enable bit (DROOEN) in the DRO output control register (DROCNT) is cleared to "0". 0: DRO reset 1: DRO operation enabled

29.3.4 DRO Output Control Register (DROCNT)

DRO Output Control Register (DROCNT) <P4 address: location H'FFFE F006>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DRO OEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	DROOEN	0	R	*1	DRO Output Enable Bit After setting the DRO operating mode register (DROMOD), the DRO output data counter (DRODCT), and the DRO address counter (DROADRCT), a data output operation from the DRO module is started by setting this bit to "1". This bit is cleared to "0" when all the specified data has been output and the DRODCT counter has become "1" in all positions, or if the DRO reset bit (DRORST) in the DROMOD register is cleared to "0". This bit cannot be directly cleared to "0" in software. Note that write to this bit is only valid when the DRORST bit is "1". If this bit is cleared to "0" (when the DRODCT counter is "1" in all positions"0" or "0" is written to the DRORST bit) and this bit is set to "1" in software at the same time, the clear to "0" event takes precedence. 0: Output disabled 1: Output enabled

Note: *1 Only writing "1" is valid. If "0" is written, the previous value will be retained.

29.3.5 DRO Output Data Counter (DRODCT)

DRO Output Data Counter (DRODCT) <P4 address: location H'FFFE F008>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DRODNUM		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRODNUM															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18 to 0	DRODNUM	All 0	R	W	DRO Output Data Count Bits This is a 19-bit counter that sets the number of times data is output. The number of data output operations is the set value plus 1. This counter is written in the state where the DRO output enable bit (DROOEN) in the DRO output control register (DROCNT) is "0". Writes to this field during DRO operation (when the DROOEN bit is "1") are ignored. After setting this counter and after the DROOEN bit is set to "1", this counter will be decremented by 1 each time data is output from the DRO module. When the value of this counter becomes "1" in all positions, the DROOEN bit is cleared to "0", and operation stops at the same time. When this counter is read during DRO operation, the number of remaining data outputs can be read.

Note: • This counter and the DRO address counter (DROADRCT) must be set to a value such that they do not exceed the SHwyRAM address range. If that range is exceeded, the values output from the DRO module will be undefined.

29.3.6 DRO Address Counter (DROADRCT)

DRO Address Counter (DROADRCT) <P4 address: location H'FFFE F00C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	DROADR		
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DROADR															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18 to 2	DROADR	All 0	R	W	DRO Address Counter Bits This counter specifies a the lower 19 bits of an address of data in SHwyRAM. Set the SHwyRAM start address that will be the transfer source while the DRO output enable bit (DROOEN) in the DRO output control register (DROCNT) is "0". The start address must be aligned on a long word boundary. Also, writes to this counter when the DROOEN bit is "1" are ignored. After this counter is set, when the DROOEN bit is set to "1", data is read out from SHwyRAM in 32-bit units and the DRO address counter is incremented by 4. When this counter is read during DRO operation, the SHwyRAM address that will be the next transfer source can be read.
1, 0			0	0	These bits are always set to "0". The write value must always be "0".

Note: • This counter and the DRO output data counter (DRODCT) must be set to a value such that they does not exceed the SHwyRAM address range. If that range is exceeded, the values output from the DRO module will be undefined.

29.4 Initial Setup Example

29.4.1 DRO Setup Example

Figure 29.3 shows an example for setting up the DRO module.

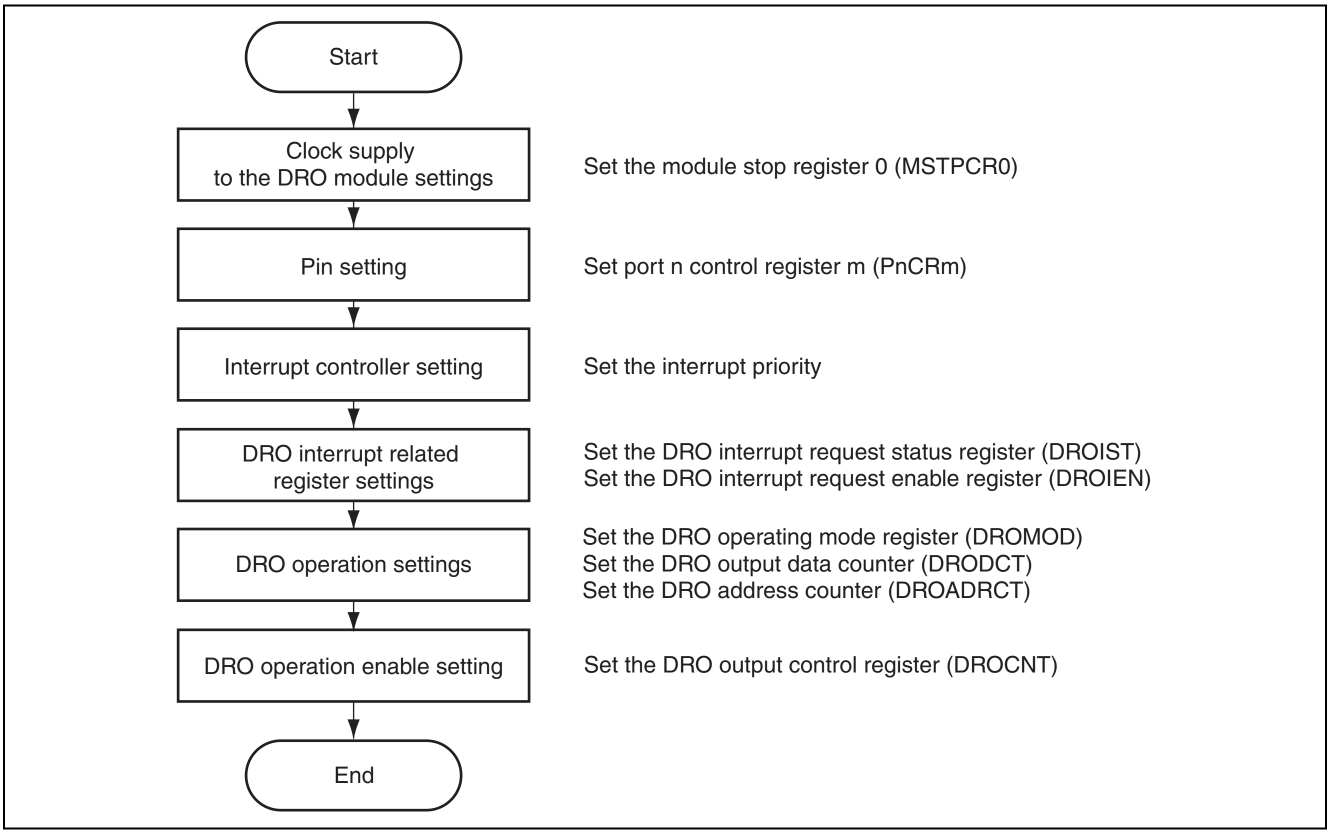

Start
Clock supply to the DRO module settings
Set the module stop register 0 (MSTPCR0)
Pin setting
Set port n control register m (PnCRm)
Interrupt controller setting
Set the interrupt priority
DRO interrupt related register settings
Set the DRO interrupt request status register (DROIST)
Set the DRO interrupt request enable register (DROIEN)
DRO operation settings
Set the DRO operating mode register (DROMOD)
Set the DRO output data counter (DRODCT)
Set the DRO address counter (DROADRCT)
DRO operation enable setting
Set the DRO output control register (DROCNT)
End

Figure 29.3 DRO Setup Example

29.4.2 Output Data Format

Figure 29.4 shows the relationship between the data arrangement in SHwyRAM and the output format.

The DRO module uses a cycle stealing method to read data from SHwyRAM for output. As a result, there are cases where the output will not be strictly periodic depending on the internal bus status.

In cases where the output is not periodic, the data output format will have an extended hold period.

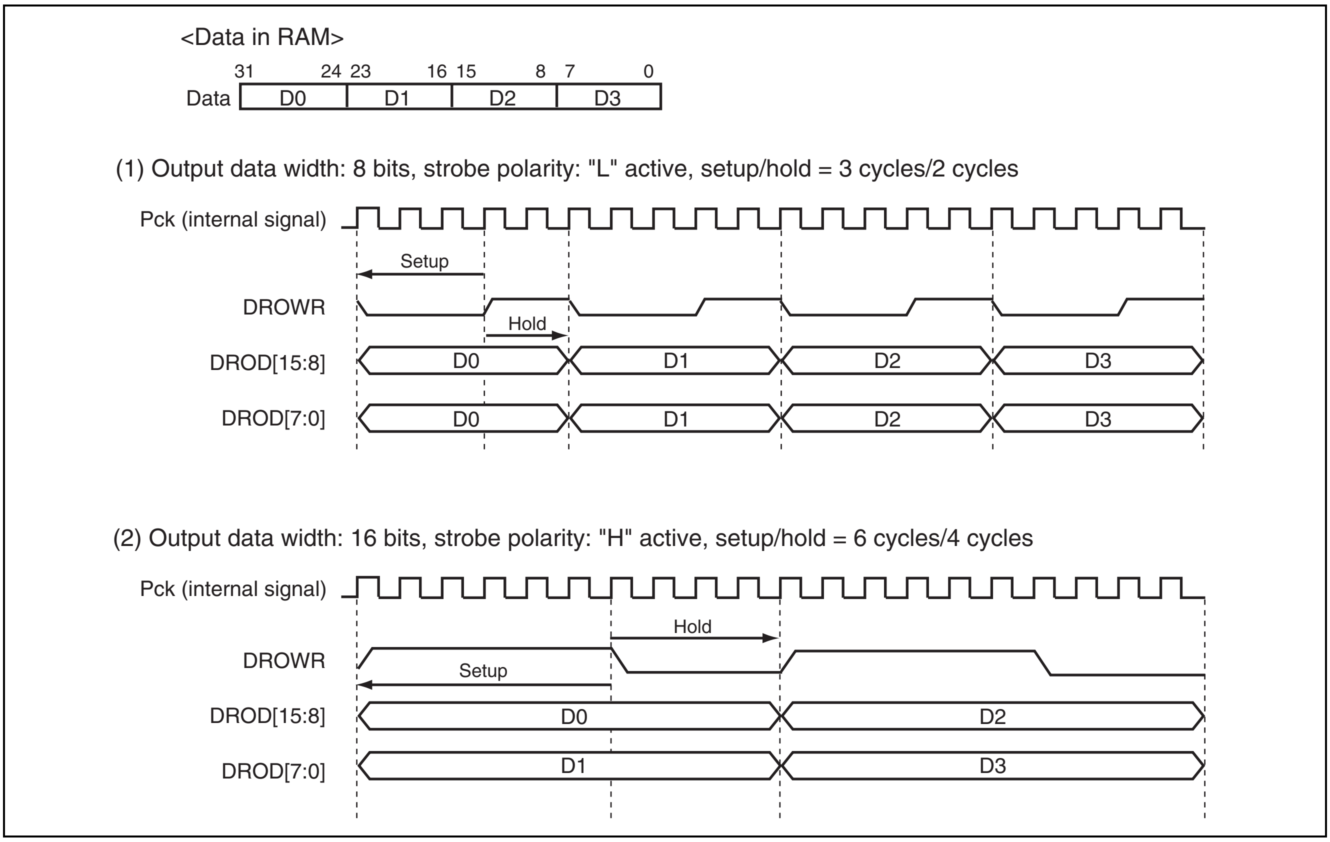

<Data in RAM>
31
24 23
16 15
8 7
0
Data
D0
D1
D2
D3
(1) Output data width: 8 bits, strobe polarity: "L" active, setup/hold = 3 cycles/2 cycles
Pck (internal signal)
Setup
DROWR
Hold
DROD[15:8]
D0
D1
D2
D3
DROD[7:0]
D0
D1
D2
D3
(2) Output data width: 16 bits, strobe polarity: "H" active, setup/hold = 6 cycles/4 cycles
Pck (internal signal)
Hold
DROWR
Setup
DROD[15:8]
D0
D2
DROD[7:0]
D1
D3

Figure 29.4 Data Format Example for Periodic Output

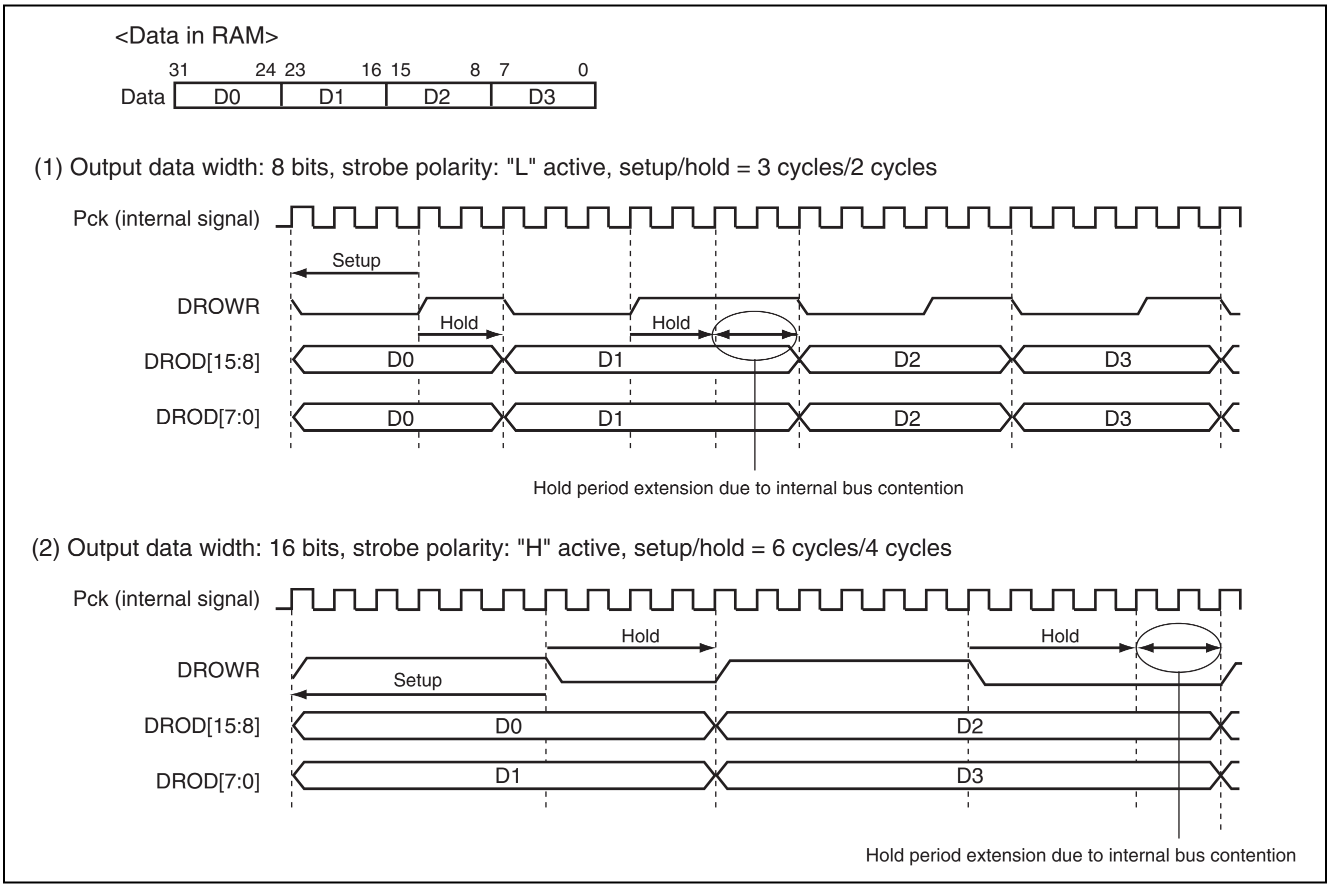

<Data in RAM>
31
24 23
16 15
8 7
0
Data
D0
D1
D2
D3
(1) Output data width: 8 bits, strobe polarity: "L" active, setup/hold = 3 cycles/2 cycles
Pck (internal signal)
Setup
DROWR
Hold
Hold
DROD[15:8]
D0
D1
D2
D3
DROD[7:0]
D0
D1
D2
D3
Hold period extension due to internal bus contention
(2) Output data width: 16 bits, strobe polarity: "H" active, setup/hold = 6 cycles/4 cycles
Pck (internal signal)
Hold
Hold
DROWR
Setup
DROD[15:8]
D0
D2
DROD[7:0]
D1
D3
Hold period extension due to internal bus contention

Figure 29.5 Data Format Example for Nonperiodic Output

29.5 Usage Notes

29.5.1 Module Stop Function Setting before Using the DRO

To use the DRO, set the DRO bit in the module stop register 0 (MSTPCR0) to "0" to enable the DRO operation, and then set the DRO related register. Otherwise, the clocks are not supplied to the DRO module and DRO operation is disabled even though the DRO related register is set.

29.5.2 Contention Between the DRO/DRI Module and the SuperHyway Bus Master

DRI0 to DRI2 and DRO use (and share) a dedicated DRO/DRI bus separate from the SuperHyway bus to access SHwyRAM. Access contention occurs when accesses from the DRI0 to DRI2 and DRO occur at the same time or overlap.

When contention occurs on the DRI0 to DRI2 and DRO, DRI0 has the highest priority and the rest is fixed in the following order.

- DRI0 > DRI1 > DRI2 > DRO

Note also that since a dedicated DRO/DRI bus connection is used to connect DRI0 to DRI2 and DRO, access contention with the SuperHyway bus master (CPU, DMA, and other units), normally will not occur. However, access contention does occur for the same 64-Kbyte area in SHwyRAM when accesses (read/write) from a dedicated DRI/DRO bus and the SuperHyway bus occur at the same time or overlap.

When access contention occurs, arbitration is performed in the following order.

- SuperHyway bus > dedicated DRI/DRO bus

This page is blank for reasons of layout.

Section 30 Parallel DAC Control (PDAC)

30.1 Overview

This MCU provides a parallel D/A converter controller circuit (the PDAC module) for controlling a 10-bit D/A converter. The PDAC module manages how the output waveforms, modulation A, modulation B, and modulation C, are output by the D/A converter.

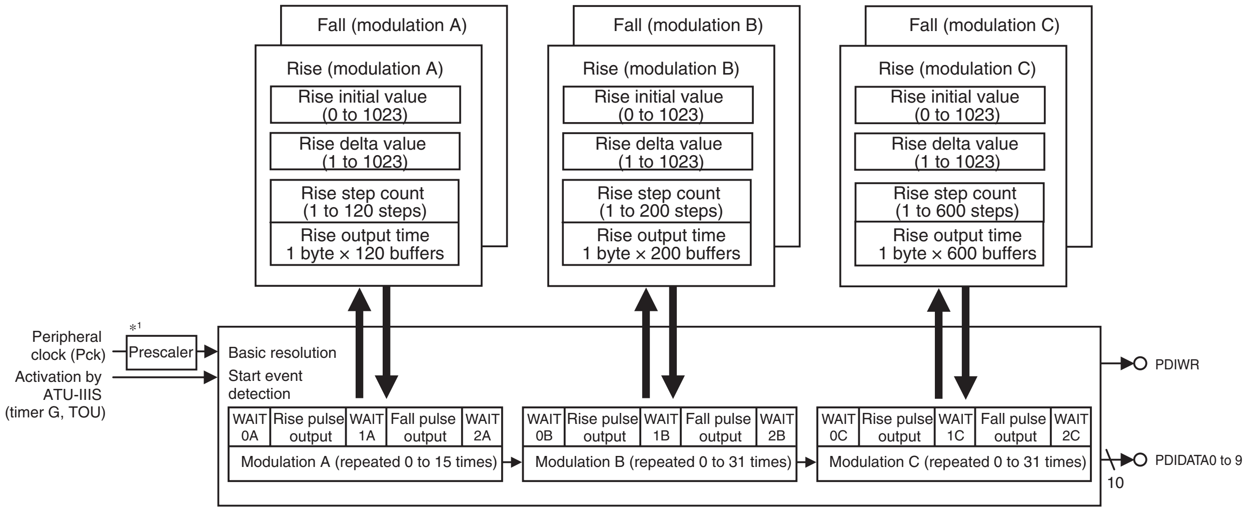
Table 30.1 lists the overview of the PDAC module.

Table 30.1 PDAC Overview

Item	Description
Modulated waveform output control	• Basic resolution: Set by the peripheral clock (Pck) and a prescaler. When Pck is operating at 40 MHz, can control with periods of 50 ns, 75 ns, 100 ns, ... 375 ns. • The time management for the three waveform outputs (modulation A, modulation B, and modulation C) can be controlled individually. The number of steps that can be set in each modulation segment are as follows. Modulation A: 240 steps, modulation B: 400 steps, modulation C: 1200 steps • Built-in registers to manage the output waveforms for each modulation The transition time, step count, initial value, and delta value can be set for the rise and fall. • Wait times can be set up for before the start of waveform output, after the rising edge, and after the falling edge. • Start events The start events can be selected from the ATU-IIIS module timer TOU2_7, timer TOU3_7, timer G channel 4, and timer G channel 5. • Operation can be stopped by software.
Event output	• Event output to ATU-IIIS, PSEL, and DRI with timing that corresponds to the waveform output • Event outputs: 4 systems to ATU-IIIS, 2 systems to PSEL, and 2 systems to DRI • Event outputs can occur at the start of the control period, at the rise of each waveform, at the fall of each waveform, at the completion of each modulation, and at the completion of the last modulation.
Write signal (PDIWR signal)	• The setup period, enable period, and polarity.
Interrupt sources	• An interrupt request can be generated after the completion of modulation waveform output.

30.2 Block Diagram

Figure 30.1 shows the PDAC block diagram.


Fall (modulation A)
Rise (modulation A)
Rise initial value (0 to 1023)
Rise delta value (1 to 1023)
Rise step count (1 to 120 steps)
Rise output time 1 byte × 120 buffers
Fall (modulation B)
Rise (modulation B)
Rise initial value (0 to 1023)
Rise delta value (1 to 1023)
Rise step count (1 to 200 steps)
Rise output time 1 byte × 200 buffers
Fall (modulation C)
Rise (modulation C)
Rise initial value (0 to 1023)
Rise delta value (1 to 1023)
Rise step count (1 to 600 steps)
Rise output time 1 byte × 600 buffers
Peripheral clock (Pck)
*1
Prescaler
Activation by ATU-IIIS (timer G, TOU)
Basic resolution
Start event detection
WAIT 0A
Rise pulse output
WAIT 1A
Fall pulse output
WAIT 2A
Modulation A (repeated 0 to 15 times)
WAIT 0B
Rise pulse output
WAIT 1B
Fall pulse output
WAIT 2B
Modulation B (repeated 0 to 31 times)
WAIT 0C
Rise pulse output
WAIT 1C
Fall pulse output
WAIT 2C
Modulation C (repeated 0 to 31 times)
PDIWR
PDIDATA0 to 9
10

Note: *1 The basic resolution is set with the divisor due to the prescaler.
The PDAC output time, wait time, and write signal (PDIWR signal) setup and enable valid period are determined by multiples (n) of the basic resolution.
When Pck is 40 MHz, the basic resolution values that can be set are 50 ns, 75 ns, 100 ns, ... 375 ns.

Figure 30.1 Block Diagram of PDAC

30.3 Input/Output Pins

Table 30.2 lists the PDAC module pins. The PDIDATA0 to PDIDATA9 pins output setting data to the D/A converter, and the PDIWR pin outputs the data write signal to the D/A converter.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 30.2 Pin Configuration

Pin Name	I/O	Function
PDIDATA0 to 9	Output	Setting data output to the D/A converter
PDIWR	Output	Write signal output to the D/A converter

30.4 Registers Descriptions

Table 30.3 lists the PDAC module registers.

Table 30.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
PDAC forced stop register	PDISTOP	H'00	H'FFFF 3400	8, 16	30-5
PDAC basic resolution setting register	PDIPRE	H'0F	H'FFFF 3401	8, 16	30-6
PDAC control period event selection register	PDICPT	H'00	H'FFFF 3402	8	30-7
PDAC status register	PDISTATUS	H'00	H'FFFF 3404	8, 16, 32	30-8
PDAC status register A	PDISTAA	H'00	H'FFFF 3405	8, 16, 32	30-9
PDAC status register B	PDISTAB	H'00	H'FFFF 3406	8, 16, 32	30-9
PDAC status register C	PDISTAC	H'00	H'FFFF 3407	8, 16, 32	30-10
PDAC interrupt control register	PDIINT	H'00	H'FFFF 3408	8	30-10
PDAC write signal period adjustment register	PDIWRC	H'0101	H'FFFF 340A	8, 16	30-11
PDAC wait time control register	PDIWTEN	H'0000	H'FFFF 340C	16	30-12
PDAC output event selection A register	PDISELA	H'0000	H'FFFF 3410	16, 32	30-13
PDAC output event selection B register	PDISELB	H'0000	H'FFFF 3412	16, 32	30-14
PDAC output event selection C register	PDISELC	H'0000	H'FFFF 3414	16, 32	30-15
PDAC output event selection D register	PDISELD	H'0000	H'FFFF 3416	16, 32	30-16
PDAC output event selection E register	PDISELE	H'0000	H'FFFF 3418	16, 32	30-17
PDAC output event selection F register	PDISELF	H'0000	H'FFFF 341A	16, 32	30-18
PDAC output event selection G register	PDISELG	H'0000	H'FFFF 341C	16, 32	30-19
PDAC output event selection H register	PDISELH	H'0000	H'FFFF 341E	16, 32	30-20
PDAC modulation A rise step count register	PDIRSA	H'01	H'FFFF 3430	8, 16	30-21
PDAC modulation A fall step count register	PDIFSA	H'01	H'FFFF 3431	8, 16	30-22
PDAC modulation A rise initial value register	PDIRIA	H'0000	H'FFFF 3434	16, 32	30-22
PDAC modulation A fall initial value register	PDIFIA	H'0000	H'FFFF 3436	16, 32	30-23
PDAC modulation A rise delta value register	PDIRDA	H'0001	H'FFFF 3438	16, 32	30-23
PDAC modulation A fall delta value register	PDIFDA	H'0001	H'FFFF 343A	16, 32	30-24
PDAC modulation A output start wait time register	PDIWT0A	H'0000	H'FFFF 343C	16, 32	30-24
PDAC modulation A post-rise wait time register	PDIWT1A	H'0000	H'FFFF 343E	16, 32	30-25
PDAC modulation A post-fall wait time register	PDIWT2A	H'0000	H'FFFF 3440	16	30-25
PDAC modulation A repeat count register	PDIREPA	H'00	H'FFFF 3442	8	30-26
PDAC modulation B rise step count register	PDIRSB	H'01	H'FFFF 3450	8, 16	30-26
PDAC modulation B fall step count register	PDIFSB	H'01	H'FFFF 3451	8, 16	30-27
PDAC modulation B rise initial value register	PDIRIB	H'0000	H'FFFF 3454	16, 32	30-27
PDAC modulation B fall initial value register	PDIFIB	H'0000	H'FFFF 3456	16, 32	30-28
PDAC modulation B rise delta value register	PDIRDB	H'0001	H'FFFF 3458	16, 32	30-28
PDAC modulation B fall delta value register	PDIFDB	H'0001	H'FFFF 345A	16, 32	30-29

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
PDAC modulation B output start wait time register	PDIWT0B	H'0000	H'FFFF 345C	16, 32	30-29
PDAC modulation B post-rise wait time register	PDIWT1B	H'0000	H'FFFF 345E	16, 32	30-30
PDAC modulation B post-fall wait time register	PDIWT2B	H'0000	H'FFFF 3460	16	30-30
PDAC modulation B repeat count register	PDIREPB	H'00	H'FFFF 3462	8	30-31
PDAC modulation C rise step count register	PDIRSC	H'0001	H'FFFF 3470	16, 32	30-31
PDAC modulation C fall step count register	PDIFSC	H'0001	H'FFFF 3472	16, 32	30-32
PDAC modulation C rise initial value register	PDIRIC	H'0000	H'FFFF 3474	16, 32	30-32
PDAC modulation C fall initial value register	PDIFIC	H'0000	H'FFFF 3476	16, 32	30-33
PDAC modulation C rise delta value register	PDIRDC	H'0001	H'FFFF 3478	16, 32	30-33
PDAC modulation C fall delta value register	PDIFDC	H'0001	H'FFFF 347A	16, 32	30-34
PDAC modulation C output start wait time register	PDIWT0C	H'0000	H'FFFF 347C	16, 32	30-34
PDAC modulation C post-rise wait time register	PDIWT1C	H'0000	H'FFFF 347E	16, 32	30-35
PDAC modulation C post-fall wait time register	PDIWT2C	H'0000	H'FFFF 3480	16	30-35
PDAC modulation C repeat count register	PDIREPC	H'00	H'FFFF 3482	8	30-36
PDAC modulation A rise output time registers 1 to 120	PDIRTA1 to PDIRTA120	Undefined	H'FFFF 3800 to H'FFFF 3877	8, 16, 32	30-36
PDAC modulation A fall output time registers 1 to 120	PDIFTA1 to PDIFTA120	Undefined	H'FFFF 3880 to H'FFFF 38F7	8, 16, 32	30-37
PDAC modulation B rise output time registers 1 to 200	PDIRTB1 to PDIRTB200	Undefined	H'FFFF 3900 to H'FFFF 39C7	8, 16, 32	30-38
PDAC modulation B fall output time registers 1 to 200	PDIFTB1 to PDIFTB200	Undefined	H'FFFF 3A00 to H'FFFF 3AC7	8, 16, 32	30-39
PDAC modulation C rise output time registers 1 to 600	PDIRTC1 to PDIRTC600	Undefined	H'FFFF 3B00 to H'FFFF 3D57	8, 16, 32	30-40
PDAC modulation C fall output time registers 1 to 600	PDIFTC1 to PDIFTC600	Undefined	H'FFFF 3D80 to H'FFFF 3FD7	8, 16, 32	30-41

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

30.4.1 PDAC Forced Stop Register (PDISTOP)

The PDISTOP register is used to forcibly stop operation during PDAC waveform output.

Forcible stop processing is performed when "1" is written to the STOP bit. (Note that "0" is returned if the written value is read out.)

When a forced stop is performed, all operations are stopped. The PDICPT register enable bit (ENB), the PDISTATUS register modulation waveform output bit (DWOUT), and the PDISTATUS register waveform output status monitor bits (DWMON) become "0". The output values immediately prior to the forced stop are retained as the output data (PDIDATA9 to 0). The write control signal (PDIWR) also becomes inactive.

Applications should set up all PDAC registers before setting the ENB bit in the PDICPT register to "1" the next time. Although the PDISTOP and PDICPT registers can be accessed at the same time if a 32-bit access is used, do not set both the STOP bit and ENB bit to "1" at the same time.

If writing to "1" the STOP bit and a start event occur at the same time, the forced stop takes precedence. The STOP bit should be always read as "0".

PDAC Forced Stop Register (PDISTOP) <P4 address: location H'FFFF 3400>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	STOP
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	STOP	0	R	W	Forced Stop Bit 0: Normal operation 1: Specifies a forced stop (Since the data written to this bit is not saved, the read value will always be "0".)

30.4.2 PDAC Basic Resolution Setting Register (PDIPRE)

The PDIPRE register sets the prescaler base used to generate the basic resolution. Change the values of the PRE bits according to the Pck frequency to select the PDAC operating clock (PDAC clock). (Values in the range 2 to 15 may be specified.) The following formula gives the PDAC clock frequency.

PDAC clock = 1/(PRE setting) × Pck, Basic resolution = 1/PDAC clock

PDAC Basic Resolution Setting Register (PDIPRE) <P4 address: location H'FFFF 3401>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	PRE			
After Reset:	0	0	0	0	1	1	1	1

<After Reset: H'0F>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3 to 0	PRE	1111	R	W	Basic Resolution Setting Bits See table 30.4.

Table 30.4 lists the relationship between the setting value and the basic resolution according to the Pck frequency.

Table 30.4 Relationship between Setting Values and Basic Resolution According to the Pck Frequency (When Pck = 40 MHz)

PRE Bits Set Value	Basic Resolution
0000	Setting prohibited
0001	Setting prohibited
0010	50 ns
0011	75 ns
0100	100 ns
0101	125 ns
0110	150 ns
0111	175 ns
1000	200 ns
1001	225 ns
1010	250 ns
1011	275 ns
1100	300 ns
1101	325 ns
1110	350 ns
1111	375 ns

30.4.3 PDAC Control Period Event Selection Register (PDICPT)

The PDICPT register enables whether or not start events can be acquired and selects from which ATU-IIIS timers start events are accepted.

There are four timers that are the object of this selection: the ATU-IIIS timer TOU2_7, timer TOU3_7, timer G channel 4, and timer G channel 5.

Since the ENB bit is controlled from the software, the PDAC module becomes unable to acquire start events from the ATU-IIIS module in the future when the ENB bit is set to "0". (Waveform output, however, is not stopped.) The operation performed is that if the enable bit is set to "0" during waveform output, PDAC operation stops after waveform output. Applications should reference the status register to determine if operation is in progress.

Also note that if "1" is written to the forced stop bit, the ENB bit becomes "0". If a start event occurs during waveform output (the PDISTATUS register DWOUT bit is "1"), that start event will be ignored (the start event is not accepted).

PDAC Control Period Event Selection Register (PDICPT) <P4 address: location H'FFFF 3402>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	ENB	—	—	CPT	
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	ENB	0	R	W	Enable Bit 0: Stopped state or stop requested state (start events are not accepted) 1: Operation enabled states (start events are accepted)
3, 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1, 0	CPT	00	R	W	Start Event Selection Bits These bits select the source of the start events. 00: Underflow of the ATU-IIIS timer TOU2_7 01: Underflow of the ATU-IIIS timer TOU3_7 10: Compare match in the ATU-IIIS timer G channel 4 11: Compare match in the ATU-IIIS timer G channel 5

30.4.4 PDAC Status Register (PDISTATUS)

The PDISTATUS register indicates the PDAC module operating state.

Figure 30.6 shows the detailed timing of the DWMON bit.

PDAC Status Register (PDISTATUS) <P4 address: location H'FFFF 3404>

Bit:	7	6	5	4	3	2	1	0
	DW OUT	DWMON			—	—	—	DO END
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	DWOUT	0	R	—	Modulated Waveform Output Bit Indicates whether modulated waveform output is disabled or enabled. 0: Modulated waveform output is disabled. 1: Modulated waveform output is enabled (for all of modulation A, modulation B, modulation C, and the wait time).
6 to 4	DWMON	000	R	—	Waveform Output Status Monitor Bits These bits indicate the waveform output status. 000: Waveform output stopped state 010: Modulation A waveform output wait 011: Modulation A waveform output in progress 100: Modulation B waveform output wait 101: Modulation B waveform output in progress 110: Modulation C waveform output wait 111: Modulation C waveform output in progress
3 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	DOEND	0	R	0	Final Modulation Complete Interrupt Request Status Flag Indicates whether or not all waveform output has completed. This flag is set to "1" when all waveform output completes (when the specified modulated waveform output (including wait time) has completed). This flag can be cleared by writing "0". Writing "1" has no effect. The set to "1" operation takes precedence over writing "0". 0: There is no final modulation complete interrupt request 1: There is a final modulation complete interrupt request

30.4.5 PDAC Status Register A (PDISTAA)

The PDISTAA register monitors the number of times waveform output has been performed for modulation A in the control period.

The value of the output count counter is retained until the next valid start event occurs.

Figure 30.8 shows a relationship between the STATUSA bits and modulated waveforms.

PDAC Status Register A (PDISTAA) <P4 address: location H'FFFF 3405>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	STATUSA			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	N	Reserved Bits These bits are always read as "0".
3 to 0	STATUSA	0000	R	N	Modulation A Waveform Output Count Monitor Bits These bits indicates the number of times (0 to 15) a waveform has been output for the modulation A currently being output.

30.4.6 PDAC Status Register B (PDISTAB)

The PDISTAB register monitors the number of times waveform output has been performed for modulation B in the control period.

The value of the output count counter is retained until the next valid start event occurs.

Figure 30.8 shows a relationship between the STATUSB bits and modulated waveforms.

PDAC Status Register B (PDISTAB) <P4 address: location H'FFFF 3406>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	STATUSB				
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	N	Reserved Bits These bits are always read as "0".
4 to 0	STATUSB	00000	R	N	Modulation B Waveform Output Count Monitor Bits These bits indicates the number of times (0 to 31) a waveform has been output for the modulation B currently being output.

30.4.7 PDAC Status Register C (PDISTAC)

The PDISTAC register monitors the number of times waveform output has been performed for modulation C in the control period.

The value of the output count counter is retained until the next valid start event occurs.

Figure 30.8 shows a relationship between the STATUSC bits and modulated waveforms.

PDAC Status Register C (PDISTAC) <P4 address: location H'FFFF 3407>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	STATUSC				
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	N	Reserved Bits These bits are always read as "0".
4 to 0	STATUSC	00000	R	N	Modulation C Waveform Output Count Monitor Bits These bits indicates the number of times (0 to 31) a waveform has been output for the modulation C currently being output.

30.4.8 PDAC Interrupt Control Register (PDIINT)

The PDIINT register controls whether or not an interrupt request is generated when all waveform output completes.

PDAC Interrupt Control Register (PDIINT) <P4 address: location H'FFFF 3408>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DO ENDE
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	DOENDE	0	R	W	Final Modulation Complete Interrupt Enable Bit Controls whether or not an interrupt request is generated when waveform output completes. 0: The output complete interrupt request is disabled. 1: The output complete interrupt request is enabled.

30.4.9 PDAC Write Signal Period Adjustment Register (PDIWRC)

The PDIWRC register sets the enable period for the write signal to the D/A converter.

This is a 16-bit register, but the upper and lower 8 bits can be accessed in byte unit individually.

The PDIWR signal polarity can be changed with the WRPOL bit.

After the polarity change, the changed polarity is reflected on the PDIWR signal after the next Pck cycle. Writing to the PDIWRC register is prohibited during waveform output (while the PDISTATUS register modulation waveform output bit (DWOUT) is "1").

PDAC Write Signal Period Adjustment Register (PDIWRC) <P4 address: location H'FFFF 340A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WR POL	—	—	ENABLE					—	—	—	SETUP				
After Reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

<After Reset: H'0101>

Bit	Abbreviation	After Reset	R	W	Description
15	WRPOL	0	R	W	Write Control Signal Polarity Sets the polarity of the PDIWR signal. 0: "L" active 1: "H" active
14, 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12 to 8	ENABLE	00001	R	W	Enable Period Setting Bits These bits set the enable period (active period) for the PDIWR signal (1 to 31). Do not set these bits to all zeros. Enable period = basic resolution × ENABLE
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4 to 0	SETUP	00001	R	W	Setup Period Setting Bits These bits set the setup period for the PDIWR signal (1 to 31). Do not set these bits to all zeros. Setup period = basic resolution × SETUP

Figure 30.2 shows the write timing to the D/A converter.

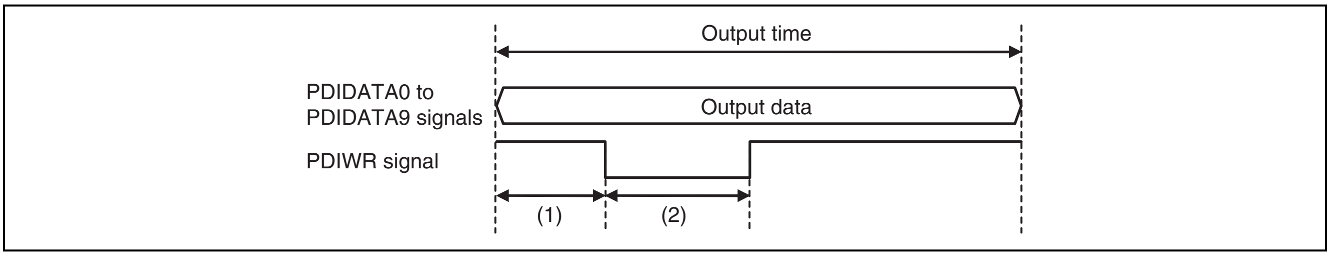


Figure 30.2 D/A Converter Write Timing

The write signal can be modified with the PDIWRC register setup period and enable period settings. (The PDIWRC register data items are multiplied by the basic resolution.)

Setup period ((1) in the figure): A "H" level period specified by the PDIWRC register SETUP bits.

Enable period ((2) in the figure): A "L" level period specified by the PDIWRC register ENABLE bits.

The output data output time is indicated by WTjkT (j = 0 to 2, k = A, B, or C), PDIRTnm and PDIFTnm (n = A, B, or C, m = 1 or higher). These times must be set to values that meet the following conditions.

- SETUP + ENABLE < WTjkT
- SETUP + ENABLE < PDIRTnm
- SETUP + ENABLE < PDIFTnm

30.4.10 PDAC Wait Time Control Register (PDIWTEN)

The PDIWTEN register controls the enabled/disabled states of the wait times during waveform output.

The PDIWTEN register must also be set when setting the various modulation wait time setting registers.

PDAC Wait Time Control Register (PDIWTEN) <P4 address: location H'FFFF 340C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	WT0AE	WT1AE	WT2AE	—	WT0BE	WT1BE	WT2BE	—	WT0CE	WT1CE	WT2CE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10	WT0AE	0	R	W	WT0A Enable Bit 0: No wait time inserted 1: Wait time inserted
9	WT1AE	0	R	W	WT1A Enable Bit 0: No wait time inserted 1: Wait time inserted
8	WT2AE	0	R	W	WT2A Enable Bit 0: No wait time inserted 1: Wait time inserted
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6	WT0BE	0	R	W	WT0B Enable Bit 0: No wait time inserted 1: Wait time inserted
5	WT1BE	0	R	W	WT1B Enable Bit 0: No wait time inserted 1: Wait time inserted

Bit	Abbreviation	After Reset	R	W	Description
4	WT2BE	0	R	W	WT2B Enable Bit 0: No wait time inserted 1: Wait time inserted
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2	WT0CE	0	R	W	WT0C Enable Bit 0: No wait time inserted 1: Wait time inserted
1	WT1CE	0	R	W	WT1C Enable Bit 0: No wait time inserted 1: Wait time inserted
0	WT2CE	0	R	W	WT2C Enable Bit 0: No wait time inserted 1: Wait time inserted

30.4.11 PDAC Output Event Selection A Register (PDISELA)

The PDISELA register selects output events for event A, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection A Register (PDISELA) <P4 address: location H'FFFF 3410>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELA										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELA	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.12 PDAC Output Event Selection B Register (PDISELB)

The PDISELB register selects output events for event B, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection B Register (PDISELB) <P4 address: location H'FFFF 3412>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELB										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELB	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.13 PDAC Output Event Selection C Register (PDISELC)

The PDISELC register selects output events for event C, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection C Register (PDISELC) <P4 address: location H'FFFF 3414>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELC										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELC	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.14 PDAC Output Event Selection D Register (PDISELD)

The PDISELD register selects output events for event D, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection D Register (PDISELD) <P4 address: location H'FFFF 3416>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELD										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELD	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.15 PDAC Output Event Selection E Register (PDISELE)

The PDISELE register selects output events for event E, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection E Register (PDISELE) <P4 address: location H'FFFF 3418>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELE										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELE	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.16 PDAC Output Event Selection F Register (PDISELF)

The PDISELF register selects output events for event F, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection F Register (PDISELF) <P4 address: location H'FFFF 341A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELF										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELF	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.17 PDAC Output Event Selection G Register (PDISELG)

The PDISELG register selects output events for event G, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection G Register (PDISELG) <P4 address: location H'FFFF 341C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELG										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELG	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.18 PDAC Output Event Selection H Register (PDISELH)

The PDISELH register selects output events for event H, which is reported to circuits external to the module. The events that can be selected are start, completion for each of the modulations, final modulation complete, and the rise and fall waveform output starts within each of the modulations.

Figure 30.12 shows the output event generation timing.

PDAC Output Event Selection H Register (PDISELH) <P4 address: location H'FFFF 341E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SELH										
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	SELH	H'000	R	W	Output Event Selection Bits Bit 10: Final modulation completion Bit 9: Modulation C completion Bit 8: Modulation B completion Bit 7: Modulation A completion Bit 6: Modulation C fall start Bit 5: Modulation C rise start Bit 4: Modulation B fall start Bit 3: Modulation B rise start Bit 2: Modulation A fall start Bit 1: Modulation A rise start Bit 0: Start (the start of the control period)

30.4.19 PDAC Modulation A Rise Step Count Register (PDIRSA)

The PDIRSA sets the number of steps required for the modulation rise.

The number of referenced rise output time registers (PDIRTA) is determined by specifying the number of steps. The elapsed time for each step is determined by the PDAC modulation A rise output time registers 1 to 120 (PDIRTA1 to 120).

PDAC Modulation A Rise Step Count Register (PDIRSA) <P4 address: location H'FFFF 3430>

Bit:	7	6	5	4	3	2	1	0
	—	RSA						
After Reset:	0	0	0	0	0	0	0	1

<After Reset: H'01>

Bit	Abbreviation	After Reset	R	W	Description
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 0	RSA	H'01	R	W	Modulation A Rise Step Count Bits These bits set the number of steps for the rise period in modulation A. This field must be set to a value in the range 1 to 120. Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.) The following is an example of RSA usage. • If RSA is set to 5, PDIRTA1 to PDIRTA5 will be valid during the modulation A rise waveform period.

30.4.20 PDAC Modulation A Fall Step Count Register (PDIFSA)

The PDIFSA sets the number of steps required for the modulation fall.

The number of referenced fall output time registers (PDIFTA) is determined by specifying the number of steps. The elapsed time for each step is determined by the PDAC modulation A fall output time registers 1 to 120 (PDIFTA1 to 120).

PDAC Modulation A Fall Step Count Register (PDIFSA) <P4 address: location H'FFFF 3431>

Bit:	7	6	5	4	3	2	1	0
	—	FSA						
After Reset:	0	0	0	0	0	0	0	1

<After Reset: H'01>

Bit	Abbreviation	After Reset	R	W	Description
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 0	FSA	H'01	R	W	Modulation A Fall Step Count Bits These bits set the number of steps for the fall period in modulation A. This field must be set to a value in the range 1 to 120. Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.) The following is an example of FSA usage. • If FSA is set to 5, PDIFTA1 to PDIFTA5 will be valid during the modulation A fall waveform period.

30.4.21 PDAC Modulation A Rise Initial Value Register (PDIRIA)

The PDIRIA register sets the value that will be the output origin during waveform output.

The value set here is output at modulation A rise and after that, the modulation A rise delta value (PDIRDA) will be added from time to time.

PDAC Modulation A Rise Initial Value Register (PDIRIA) <P4 address: location H'FFFF 3434>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RIA									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	RIA	H'000	R	W	Modulation A Rise Initial Value Bits These bits set the initial value at rise for modulation A (0 or larger).

30.4.22 PDAC Modulation A Fall Initial Value Register (PDIFIA)

The PDIFIA register sets the value that will be the fall origin during waveform output.

The value set here is output at modulation A fall and after that, the modulation A fall delta value (PDIFDA) will be subtracted from time to time.

PDAC Modulation A Fall Initial Value Register (PDIFIA) <P4 address: location H'FFFF 3436>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FIA									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	FIA	H'000	R	W	Modulation A Rise Initial Value Bits These bits set the initial value at fall for modulation A (0 or larger).

30.4.23 PDAC Modulation A Rise Delta Value Register (PDIRDA)

The PDIRDA register sets the change value (amount added) for each time transition during waveform output.

During modulation A rising waveform output, the value set here is added to the output value after the time (the basic resolution × PDIRTAn) set with the PDAC modulation A rise output time register 1 to 120 (PDIRTA1 to 120) has elapsed.

Since a normal addition operation is used even if the total of the added values (PDIRIA + the result of adding PDIRDA up to this point) overflows, applications must be designed so that the maximum value is not exceeded.

PDAC Modulation A Rise Delta Value Register (PDIRDA) <P4 address: location H'FFFF 3438>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RDA									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	RDA	H'001	R	W	Modulation A Rise Delta Value Bits Sets the change value (amount added) for the rise time in modulation A (1 or larger).

30.4.24 PDAC Modulation A Fall Delta Value Register (PDIFDA)

The PDIFDA register sets the change value (amount subtracted) for each time transition during waveform output.

During modulation A falling waveform output, the value set here is subtracted from the output value after the time (the basic resolution × PDIFTAn) set with the PDAC modulation A fall output time registers (PDIFTAn) has elapsed.

Since a normal subtraction operation is used even if the total of the summed values (PDIFIA + the result of subtracting PDIFDA up to this point) underflows, applications must be designed so that the minimum value is not exceeded.

PDAC Modulation A Fall Delta Value Register (PDIFDA) <P4 address: location H'FFFF 343A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FDA									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	FDA	H'001	R	W	Modulation A Fall Delta Value Bits Sets the change value (amount subtracted) for the fall time in modulation A (1 or larger).

30.4.25 PDAC Modulation A Output Start Wait Time Register (PDIWT0A)

The PDIWT0A register sets the wait time after start until the modulation A waveform output starts to be <basic resolution> × WT0A. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT0A wait time inserted state.

PDAC Modulation A Output Start Wait Time Register (PDIWT0A) <P4 address: location H'FFFF 343C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT0AT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT0AT	H'0000	R	W	Modulation A Output Start Wait Time Bits Set these bits to the wait time after start until the modulation A waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT0AE bit is "1" (wait time inserted). See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.26 PDAC Modulation A Post-Rise Wait Time Register (PDIWT1A)

The PDIWT1A register sets the wait time after modulation A rise waveform output to be <basic resolution> × WT1A. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT1A wait time inserted state.

PDAC Modulation A Post-Rise Wait Time Register (PDIWT1A) <P4 address: location H'FFFF 343E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT1AT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT1AT	H'0000	R	W	Modulation A Post-Rise Wait Time Bits Set these bits to the wait time after modulation A rise waveform output and before the fall waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT1AE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.27 PDAC Modulation A Post-Fall Wait Time Register (PDIWT2A)

The PDIWT2A register sets the wait time after modulation A fall waveform output to be <basic resolution> × WT2A. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT2A wait time inserted state.

PDAC Modulation A Post-Fall Wait Time Register (PDIWT2A) <P4 address: location H'FFFF 3440>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT2AT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT2AT	H'0000	R	W	Modulation A Post-Fall Wait Time Bits Set these bits to the wait time after modulation A fall waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT2AE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.28 PDAC Modulation A Repeat Count Register (PDIREPA)

The PDIREPA register sets the number of times for modulation A waveform output in the control period.

PDAC Modulation A Repeat Count Register (PDIREPA) <P4 address: location H'FFFF 3442>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	REPA			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3 to 0	REPA	0000	R	W	Modulation A Repeat Count Setting Bits Set these bits to the number of waveforms to be output in the modulation A control period (to a value in the range 0 to 15). Do not set the modulation A, B, and C repeat counts to all be zero.

30.4.29 PDAC Modulation B Rise Step Count Register (PDIRSB)

The PDIRSB sets the number of steps required for the modulation rise.

The number of referenced rise output time registers (PDIRTB) is determined by specifying the number of steps. The elapsed time for each step is determined by the PDAC modulation B rise output time registers 1 to 200 (PDIRTB1 to 200).

PDAC Modulation B Rise Step Count Register (PDIRSB) <P4 address: location H'FFFF 3450>

Bit:	7	6	5	4	3	2	1	0
	RSB							
After Reset:	0	0	0	0	0	0	0	1

<After Reset: H'01>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	RSB	H'01	R	W	Modulation B Rise Step Count Bits These bits set the number of steps for the rise period in modulation B. This field must be set to a value in the range 1 to 200. Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.) The following is an example of RSB usage. • If RSB is set to 5, PDIRTB1 to PDIRTB5 will be valid during the modulation B rise waveform period.

30.4.30 PDAC Modulation B Fall Step Count Register (PDIFSB)

The PDIFSB sets the number of steps required for the modulation fall.

The number of referenced fall output time registers (PDIFTB) is determined by specifying the number of steps. The elapsed time for each step is determined by the PDAC modulation B fall output time registers 1 to 200 (PDIFTB1 to 200).

PDAC Modulation B Fall Step Count Register (PDIFSB) <P4 address: location H'FFFF 3451>

Bit:	7	6	5	4	3	2	1	0
	FSB							
After Reset:	0	0	0	0	0	0	0	1

<After Reset: H'01>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	FSB	H'01	R	W	Modulation B Fall Step Count Bits These bits set the number of steps for the fall period in modulation B. This field must be set to a value in the range 1 to 200. Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.) The following is an example of FSB usage. • If FSB is set to 5, PDIFTB1 to PDIFTB5 will be valid during the modulation B fall waveform period.

30.4.31 PDAC Modulation B Rise Initial Value Register (PDIRIB)

The PDIRIB register sets the value that will be the output origin during waveform output.

The value set here is output at modulation B rise and after that, the modulation B rise delta value (PDIRDB) will be added from time to time.

PDAC Modulation B Rise Initial Value Register (PDIRIB) <P4 address: location H'FFFF 3454>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RIB									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	RIB	H'000	R	W	Modulation B Rise Initial Value Bits These bits set the initial value at rise for modulation B (0 or larger).

30.4.32 PDAC Modulation B Fall Initial Value Register (PDIFIB)

The PDIFIB register sets the value that will be the output origin during waveform output.

The value set here is output at modulation B fall and after that, the modulation B fall delta value (PDIRDB) will be subtracted from time to time.

PDAC Modulation B Fall Initial Value Register (PDIFIB) <P4 address: location H'FFFF 3456>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FIB									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	FIB	H'000	R	W	Modulation B Fall Initial Value Bits These bits set the initial value at fall for modulation B (0 or larger).

30.4.33 PDAC Modulation B Rise Delta Value Register (PDIRDB)

The PDIRDB register sets the change value (amount added) for each time transition during waveform output.

During modulation B rising waveform output, the value set here is added to the output value after the time (the basic resolution × PDIRTBn) set with the PDAC modulation B rise output time register 1 to 200 (PDIRTB1 to 200) has elapsed.

Since a normal addition operation is used even if the total of the added values (PDIRIB + the result of adding PDIRDB up to this point) overflows, applications must be designed so that the maximum value is not exceeded.

PDAC Modulation B Rise Delta Value Register (PDIRDB) <P4 address: location H'FFFF 3458>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RDB									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	RDB	H'001	R	W	Modulation B Rise Delta Value Bits Sets the change value (amount added) for the rise time in modulation B (1 or larger).

30.4.34 PDAC Modulation B Fall Delta Value Register (PDIFDB)

The PDIFDB register sets the change value (amount subtracted) for each time transition during waveform output.

During modulation B falling waveform output, the value set here is subtracted from the output value after the time (the basic resolution × PDIFTBn) set with the PDAC modulation B fall output time register 1 to 200 (PDIFTB1 to 200) has elapsed.

Since a normal subtraction operation is used even if the total of the summed values (PDIFIB + the result of subtracting PDIFDB up to this point) underflows, applications must be designed so that the minimum value is not exceeded.

PDAC Modulation B Fall Delta Value Register (PDIFDB) <P4 address: location H'FFFF 345A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FDB									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	FDB	H'001	R	W	Modulation B Fall Delta Value Bits Sets the change value (amount subtracted) for the fall time in modulation B (1 or larger).

30.4.35 PDAC Modulation B Output Start Wait Time Register (PDIWT0B)

The PDIWT0B register sets the wait time after start until the modulation B waveform output starts to be <basic resolution> × WT0B. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT0B wait time inserted state.

PDAC Modulation B Output Start Wait Time Register (PDIWT0B) <P4 address: location H'FFFF 345C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT0BT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT0BT	H'0000	R	W	Modulation B Output Start Wait Time Bits Set these bits to the wait time after start until the modulation B waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT0BE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.36 PDAC Modulation B Post-Rise Wait Time Register (PDIWT1B)

The PDIWT1B register sets the wait time after modulation B rise waveform output to be <basic resolution> × WT1B. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT1B wait time inserted state.

PDAC Modulation B Post-Rise Wait Time Register (PDIWT1B) <P4 address: location H'FFFF 345E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT1BT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT1BT	H'0000	R	W	Modulation B Post-Rise Wait Time Bits Set these bits to the wait time after modulation B rise waveform output and before the fall waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT1BE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.37 PDAC Modulation B Post-Fall Wait Time Register (PDIWT2B)

The PDIWT2B register sets the wait time after modulation B fall waveform output to be <basic resolution> × WT2B. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT2B wait time inserted state.

PDAC Modulation B Post-Fall Wait Time Register (PDIWT2B) <P4 address: location H'FFFF 3460>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT2BT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT2BT	H'0000	R	W	Modulation B Post-Fall Wait Time Bits Set these bits to the wait time after modulation B fall waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT2BE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.38 PDAC Modulation B Repeat Count Register (PDIREPB)

The PDIREPB register sets the number of times for modulation B waveform output in the control period.

PDAC Modulation B Repeat Count Register (PDIREPB) <P4 address: location H'FFFF 3462>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	REPB				
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4 to 0	REPB	00000	R	W	Modulation B Repeat Count Setting Bits Set these bits to the number of waveforms to be output in the modulation B control period (to a value in the range 0 to 31). Do not set the modulation A, B, and C repeat counts to all be zero.

30.4.39 PDAC Modulation C Rise Step Count Register (PDIRSC)

The PDIRSC sets the number of steps required for the modulation rise.

The number of referenced rise output time registers (PDIRTC) is determined by specifying the number of steps. The elapsed time for each step is determined by the PDAC modulation C rise output time registers 1 to 600 (PDIRTC1 to 600).

PDAC Modulation C Rise Step Count Register (PDIRSC) <P4 address: location H'FFFF 3470>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RSC									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	RSC	H'001	R	W	Modulation C Rise Step Count Bits These bits set the number of steps for the rise period in modulation C. This field must be set to a value in the range 1 to 600. Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.) The following is an example of RSC usage. • If RSC is set to 5, PDIRTC1 to PDIRTC5 will be valid during the modulation C rise waveform period.

30.4.40 PDAC Modulation C Fall Step Count Register (PDIFSC)

The PDIFSC sets the number of steps required for the modulation fall.

The number of referenced fall output time registers (PDIFTC) is determined by specifying the number of steps. The elapsed time for each step is determined by the PDAC modulation C fall output time registers 1 to 600 (PDIFTC1 to 600).

PDAC Modulation C Fall Step Count Register (PDIFSC) <P4 address: location H'FFFF 3472>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FSC									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	FSC	H'001	R	W	Modulation C Fall Step Count Bits These bits set the number of steps for the fall period in modulation C. This field must be set to a value in the range 1 to 600. Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.) The following is an example of FSC usage. • If FSC is set to 5, PDIFTC1 to PDIFTC5 will be valid during the modulation C fall waveform period.

30.4.41 PDAC Modulation C Rise Initial Value Register (PDIRIC)

The PDIRIC register sets the value that will be the output origin during waveform output.

The value set here is output at modulation C rise and after that, the modulation C rise delta value (PDIRDC) will be added from time to time.

PDAC Modulation C Rise Initial Value Register (PDIRIC) <P4 address: location H'FFFF 3474>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RIC									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	RIC	H'000	R	W	Modulation C Rise Initial Value Bits These bits set the initial value at rise for modulation C (0 or larger).

30.4.42 PDAC Modulation C Fall Initial Value Register (PDIFIC)

The PDIFIC register sets the value that will be the output origin during waveform output.

The value set here is output at modulation C fall and after that, the modulation C fall delta value (PDIFDC) will be subtracted from time to time.

PDAC Modulation C Fall Initial Value Register (PDIFIC) <P4 address: location H'FFFF 3476>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FIC									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	FIC	H'000	R	W	Modulation C Fall Initial Value Bits These bits set the initial value at fall for modulation C (0 or larger).

30.4.43 PDAC Modulation C Rise Delta Value Register (PDIRDC)

The PDIRDC register sets the change value (amount added) for each time transition during waveform output.

During modulation C rising waveform output, the value set here is added to the output value after the time (the basic resolution × PDIRTCn) set with the PDAC modulation C rise output time register 1 to 600 (PDIRTC1 to 600) has elapsed.

Since a normal addition operation is used even if the total of the added values (PDIRIC + the result of adding PDIRDC up to this point) overflows, applications must be designed so that the maximum value is not exceeded.

PDAC Modulation C Rise Delta Value Register (PDIRDC) <P4 address: location H'FFFF 3478>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RDC									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	RDC	H'001	R	W	Modulation C Rise Delta Value Bits Sets the change value (amount added) for the rise time in modulation C (1 or larger).

30.4.44 PDAC Modulation C Fall Delta Value Register (PDIFDC)

The PDIFDC register sets the change value (amount subtracted) for each time transition during waveform output.

During modulation C falling waveform output, the value set here is subtracted from the output value after the time (the basic resolution × PDIFTCn) set with the PDAC modulation C fall output time register 1 to 600 (PDIFTC1 to 600) has elapsed.

Since a normal subtraction operation is used even if the total of the summed values (PDIFIC + the result of subtracting PDIFDC up to this point) underflows, applications must be designed so that the minimum value is not exceeded.

PDAC Modulation C Fall Delta Value Register (PDIFDC) <P4 address: location H'FFFF 347A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FDC									
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	FDC	H'001	R	W	Modulation C Fall Delta Value Bits Sets the change value (amount subtracted) for the fall time in modulation C (1 or larger).

30.4.45 PDAC Modulation C Output Start Wait Time Register (PDIWT0C)

The PDIWT0C register sets the wait time after start until the modulation C waveform output starts to be <basic resolution> × WT0C. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT0C wait time inserted state.

PDAC Modulation C Output Start Wait Time Register (PDIWT0C) <P4 address: location H'FFFF 347C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT0CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT0CT	H'0000	R	W	Modulation C Output Start Wait Time Bits Set these bits to the wait time after start until the modulation C waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT0CE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.46 PDAC Modulation C Post-Rise Wait Time Register (PDIWT1C)

The PDIWT1C register sets the wait time after modulation C rise waveform output to be <basic resolution> × WT1C. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT1C wait time inserted state.

PDAC Modulation C Post-Rise Wait Time Register (PDIWT1C) <P4 address: location H'FFFF 347E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT1CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT1CT	H'0000	R	W	Modulation C Post-Rise Wait Time Bits Set these bits to the wait time after modulation C rise waveform output and before the fall waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT1CE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.47 PDAC Modulation C Post-Fall Wait Time Register (PDIWT2C)

The PDIWT2C register sets the wait time after modulation C fall waveform output to be <basic resolution> × WT2C. Note that this register is valid when the PDAC wait time control register (PDIWTEN) is set to the WT2C wait time inserted state.

PDAC Modulation C Post-Fall Wait Time Register (PDIWT2C) <P4 address: location H'FFFF 3480>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WT2CT															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	WT2CT	H'0000	R	W	Modulation C Post-Fall Wait Time Bits Set these bits to the wait time after modulation C fall waveform output (1 or larger). Do not set these bits to all zeros when the PDIWTEN register WT2CE bit is "1" (wait time inserted). There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

30.4.48 PDAC Modulation C Repeat Count Register (PDIREPC)

The PDIREPC register sets the number of times for modulation C waveform output in the control period.

PDAC Modulation C Repeat Count Register (PDIREPC) <P4 address: location H'FFFF 3482>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	REPC				
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4 to 0	REPC	00000	R	W	Modulation C Repeat Count Setting Bits Set these bits to the number of waveforms to be output in the modulation C control period (to a value in the range 0 to 31). Do not set the modulation A, B, and C repeat counts to all be zero.

30.4.49 PDAC Modulation A Rise Output Time Registers 1 to 120 (PDIRTA1 to 120)

The PDIRTA1 to 120 registers set the transition times for the rise time during modulation A waveform output (up to a maximum of 120 steps). The current output value is held for the times specified with these registers (<basic resolution> × PDIRTAn).

After that, the modulation A rise delta value (PDIRDA) is added, and after outputting that as the next output value, that output value is held for the time in the next output time register (example: if the current point is PDIRTA1, then the next one will be PDIRTA2).

The PDIRTA1 to 120 registers specify the hold times (transition times) from the rise first step until the 120th step. When these registers are read during operation, the read values are "0".

There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

PDAC Modulation A Rise Output Time Registers 1 to 120 (PDIRTA1 to 120) <P4 address: location H'FFFF 3800 to H'FFFF 3877>

Bit:	7	6	5	4	3	2	1	0
	RTAn							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	RTAn	Undefined	R	W	Modulation A Rise Time Setting Bits Set these bits to the count value for holding the output to the D/A converter (1 to 255). Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.)

Legend: n = 1 to 120

30.4.50 PDAC Modulation A Fall Output Time Registers 1 to 120 (PDIFTA1 to 120)

The PDIFTA1 to 120 registers set the transition times for the fall time during modulation A waveform output (up to a maximum of 120 steps). The current output value is held for the times specified with these registers (<basic resolution> × PDIFTAn).

After that, the modulation A fall delta value (PDIFDA) is added, and after outputting that as the next output value, that output value is held for the time in the next output time register (example: if the current point is PDIFTA1, then the next one will be PDIFTA2).

The PDIFTA1 to 120 registers specify the hold times (transition times) from the fall first step until the 120th step. When these registers are read during operation, the read values are "0".

There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

PDAC Modulation A Fall Output Time Registers 1 to 120
(PDIFTA1 to 120)

<P4 address: location H'FFFF 3880 to H'FFFF 38F7>

Bit:	7	6	5	4	3	2	1	0
	FTAn							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	FTAn	Undefined	R	W	Modulation A Fall Time Setting Bits Set these bits to the count value for holding the output to the D/A converter (1 to 255). Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.)

Legend: n = 1 to 120

30.4.51 PDAC Modulation B Rise Output Time Registers 1 to 200 (PDIRTB1 to 200)

The PDIRTB1 to 200 registers set the transition times for the rise time during modulation B waveform output (up to a maximum of 200 steps). The current output value is held for the times specified with these registers (<basic resolution> × PDIRTBn).

After that, the modulation B rise delta value (PDIRDB) is added, and after outputting that as the next output value, that output value is held for the time in the next output time register (example: if the current point is PDIRTB1, then the next one will be PDIRTB2).

The PDIRTB1 to 200 registers specify the hold times (transition times) from the rise first step until the 200th step. When these registers are read during operation, the read values are "0".

There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

PDAC Modulation B Rise Output Time Registers 1 to 200 (PDIRTB1 to 200) <P4 address: location H'FFFF 3900 to H'FFFF 39C7>

Bit:	7	6	5	4	3	2	1	0
	RTBn							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	RTBn	Undefined	R	W	Modulation B Rise Time Setting Bits Set these bits to the count value for holding the output to the D/A converter (1 to 255). Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.)

Legend: n = 1 to 200

30.4.52 PDAC Modulation B Fall Output Time Registers 1 to 200 (PDIFTB1 to 200)

The PDIFTB1 to 200 registers set the transition times for the fall time during modulation B waveform output (up to a maximum of 200 steps). The current output value is held for the times specified with these registers (<basic resolution> × PDIFTBn).

After that, the modulation B fall delta value (PDIFDB) is added, and after outputting that as the next output value, that output value is held for the time in the next output time register (example: if the current point is PDIFTB1, then the next one will be PDIFTB2).

The PDIFTB1 to 200 registers specify the hold times (transition times) from the fall first step until the 200th step. When these registers are read during operation, the read values are "0".

There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

PDAC Modulation B Fall Output Time Registers 1 to 200
(PDIFTB1 to 200)

<P4 address: location H'FFFF 3A00 to H'FFFF 3AC7>

Bit:	7	6	5	4	3	2	1	0
	FTBn							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	FTBn	Undefined	R	W	Modulation B Fall Time Setting Bits Set these bits to the count value for holding the output to the D/A converter (1 to 255). Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.)

Legend: n = 1 to 200

30.4.53 PDAC Modulation C Rise Output Time Registers 1 to 600 (PDIRTC1 to 600)

The PDIRTC1 to 600 registers set the transition times for the rise time during modulation C waveform output (up to a maximum of 600 steps). The current output value is held for the times specified with these registers (<basic resolution> × PDIRTCn).

After that, the modulation C rise delta value (PDIRDC) is added, and after outputting that as the next output value, that output value is held for the time in the next output time register (example: if the current point is PDIRTC1, then the next one will be PDIRTC2).

The PDIRTC1 to 600 registers specify the hold times (transition times) from the rise first step until the 600th step. When these registers are read during operation, the read values are "0".

There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

PDAC Modulation C Rise Output Time Registers 1 to 600
(PDIRTC1 to 600)

<P4 address: location H'FFFF 3B00 to H'FFFF 3D57>

Bit:	7	6	5	4	3	2	1	0
	RTCn							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	RTCn	Undefined	R	W	Modulation C Rise Time Setting Bits Set these bits to the count value for holding the output to the D/A converter (1 to 255). Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.)

Legend: n = 1 to 600

30.4.54 PDAC Modulation C Fall Output Time Registers 1 to 600 (PDIFTC1 to 600)

The PDIFTC1 to 600 registers set the transition times for the fall time during modulation C waveform output (up to a maximum of 600 steps). The current output value is held for the times specified with these registers (<basic resolution> × PDIFTCn).

After that, the modulation C fall delta value (PDIFDC) is added, and after outputting that as the next output value, that output value is held for the time in the next output time register (example: if the current point is PDIFTC1, then the next one will be PDIFTC2).

The PDIFTC1 to 600 registers specify the hold times (transition times) from the fall first step until the 600th step. When these registers are read during operation, the read values are "0".

There are conditions on the values to which these bits may be set. See section 30.4.9, PDAC Write Signal Period Adjustment Register (PDIWRC) for details.

PDAC Modulation C Fall Output Time Registers 1 to 600
(PDIFTC1 to 600)

<P4 address: location H'FFFF 3D80 to H'FFFF 3FD7>

Bit:	7	6	5	4	3	2	1	0
	FTCn							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	FTCn	Undefined	R	W	Modulation C Fall Time Setting Bits Set these bits to the count value for holding the output to the D/A converter (1 to 255). Values other than the above are illegal. (Operation is not guaranteed if an illegal value is set.)

Legend: n = 1 to 600

30.5 Operation

30.5.1 Overview

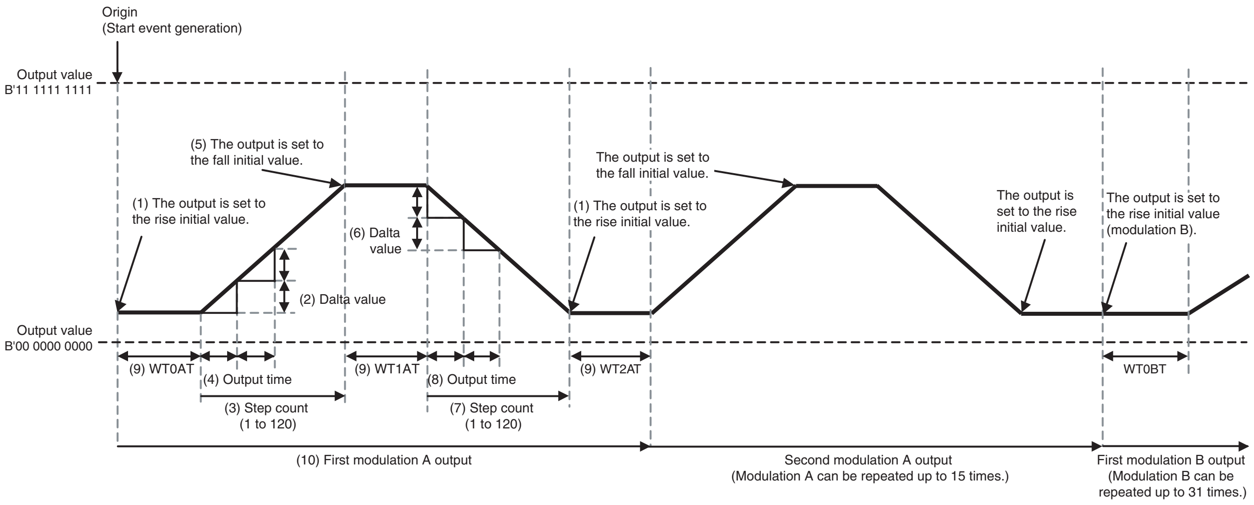
The PDAC module is started by a start event from the timer TOU2_7, the timer TOU3_7, a channel 4 in the timer G, and a channel 5 in the timer G, and output, as a time series, data provided to a D/A converter to generate the waveforms for modulation A, modulation B, and modulation C in the control period.

A period with a fixed wait time can be set for the period from activation/start to the start of modulation A, the period from the end of modulation A to the start of modulation B, and the period from the end of modulation B to the start of modulation C.

Since an interrupt is generated after the last modulation output, the parameters can be changed at that time. The parameters must not be modified before that point (operation is not guaranteed in that case).

Since start events that occur during waveform output are ignored (not accepted), adequate care must be observed with respect to the setting values.

Figure 30.3 presents an overview of PDAC module operation.

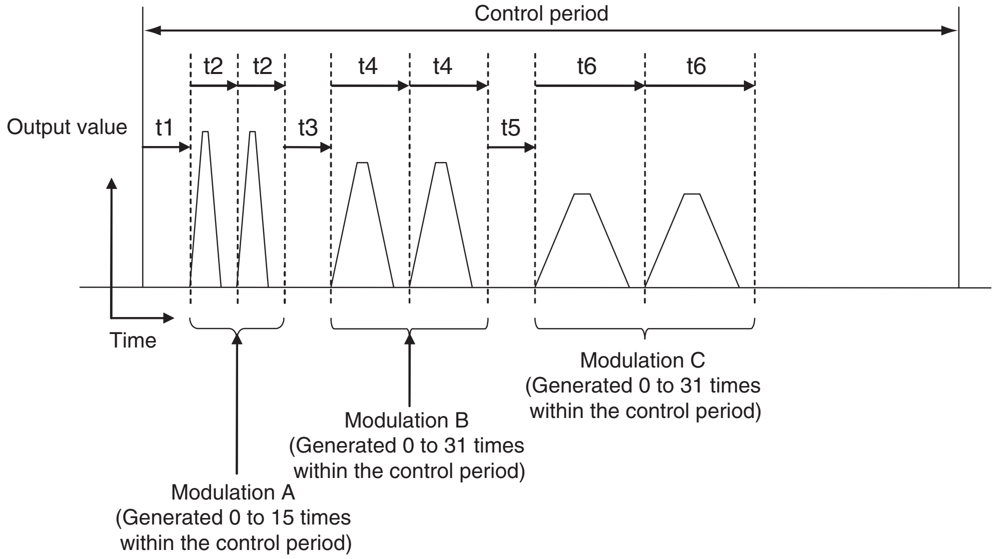

Origin
(Start event generation)
Output value B'11 1111 1111
(5) The output is set to the fall initial value.
The output is set to the fall initial value.
(1) The output is set to the rise initial value.
(6) Dalta value
(1) The output is set to the rise initial value.
The output is set to the rise initial value.
The output is set to the rise initial value (modulation B).
(2) Dalta value
Output value B'00 0000 0000
(9) WT0AT
(4) Output time
(9) WT1AT
(8) Output time
(9) WT2AT
WT0BT
(3) Step count (1 to 120)
(7) Step count (1 to 120)
(10) First modulation A output
Second modulation A output (Modulation A can be repeated up to 15 times.)
First modulation B output (Modulation B can be repeated up to 31 times.)

[Values that can be set]
(1) Rise initial value, (2) Rise delta value, (3) Rise step count, (4) Rise output time,
(5) Fall initial value, (6) Fall delta value, (7) Fall step count, (8) Fall output time,
(9) Wait values (three types), (10) Output count

The allowable values for (3) Rise step count, (7) Fall step count, and (10) Output count differ between modulation A, modulation B, and modulation C. Also, it is not legal to set all the output counts to 0.
Modulation A: (3) Rise step count: 1 to 120, (7) Fall step count 1 to 120, (10) Output count: 0 to 15
Modulation B: (3) Rise step count: 1 to 200, (7) Fall step count 1 to 200, (10) Output count: 0 to 31
Modulation C: (3) Rise step count: 1 to 600, (7) Fall step count 1 to 600, (10) Output count: 0 to 31

Figure 30.3 PDAC Operation Overview

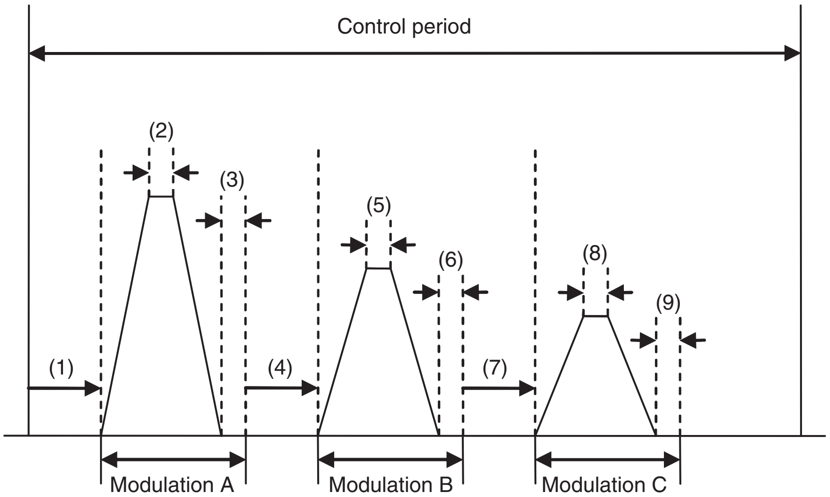
Figure 30.4 presents an operational overview of the PDAC module control period.



Legend:
t1: Wait time from the start event until the start of modulation A
t2: Modulation A output cycle (modulation waveform output and the wait time following that output)
t3: Wait time from the end of modulation A to the start of modulation B
t4: Modulation B output cycle (modulation waveform output and the wait time following that output)
t5: Wait time from the end of modulation B to the start of modulation C
t6: Modulation C output cycle (modulation waveform output and the wait time following that output)

Figure 30.4 PDAC Module Control Period Operation Overview

Figure 30.5 shows the relationship between the output waveform within the PDAC control period and the wait time setting registers.



Legend:
(1) PDIWT0A: Wait time from the start event until the start of modulation A
(2) PDIWT1A: Wait time after the modulation A rise and before the modulation A fall
(3) PDIWT2A: Wait time after the modulation A fall and before the modulation A end
(4) PDIWT0B: Wait time after the modulation A end and before the modulation B start
(5) PDIWT1B: Wait time after the modulation B rise and before the modulation B fall
(6) PDIWT2B: Wait time after the modulation B fall and before the modulation B end
(7) PDIWT0C: Wait time after the modulation B end and before the modulation C start
(8) PDIWT1C: Wait time after the modulation C rise and before the modulation C fall
(9) PDIWT2C: Wait time after the modulation C fall and before the modulation C end

Figure 30.5 Relationship between Output Waveform and Wait Time Setting Registers in the PDAC Control Period

Figure 30.6 shows the relationship between the PDAC start event and both operation enable the operation monitor (during normal operation).

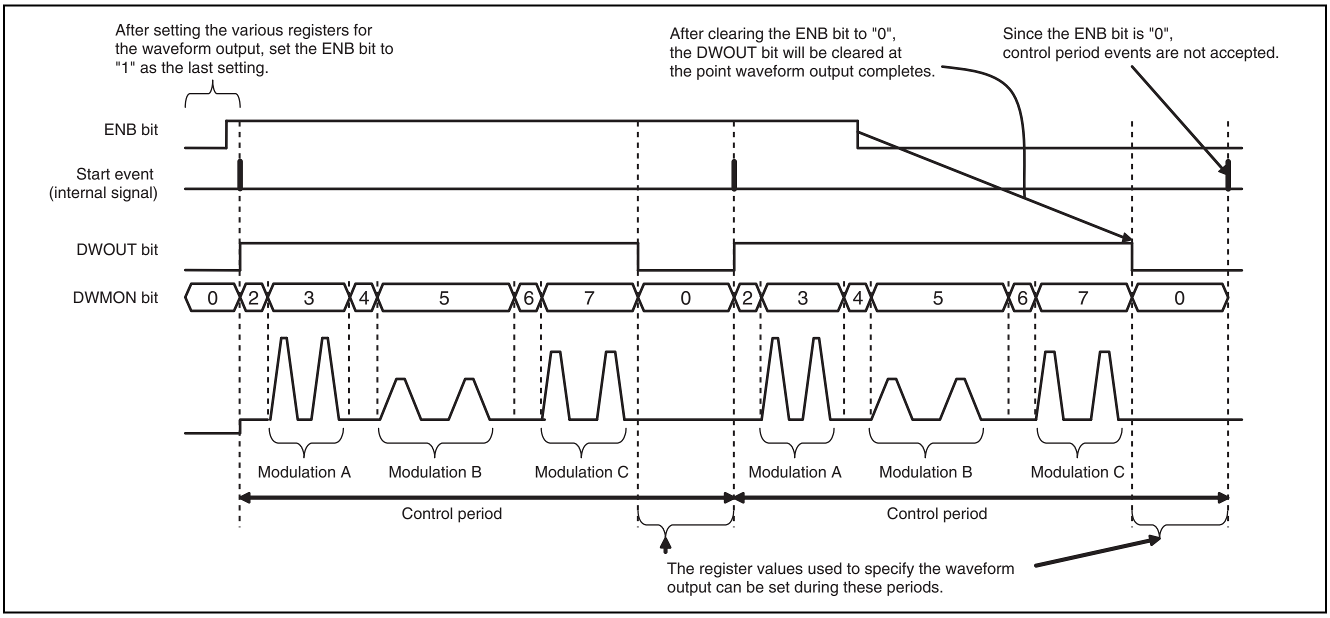

After setting the various registers for the waveform output, set the ENB bit to "1" as the last setting.
After clearing the ENB bit to "0", the DWOUT bit will be cleared at the point waveform output completes.
Since the ENB bit is "0", control period events are not accepted.
ENB bit
Start event (internal signal)
DWOUT bit
DWMON bit
0
2
3
4
5
6
7
0
2
3
4
5
6
7
0
Modulation A
Modulation B
Modulation C
Modulation A
Modulation B
Modulation C
Control period
Control period
The register values used to specify the waveform output can be set during these periods.

Figure 30.6 Relationship between PDAC Start Events and both Operation Enable and Operation Monitor (Normal Operation)

Figure 30.7 shows the relationship between the PDAC start event and both operation enable the operation monitor (during forced stop processing).

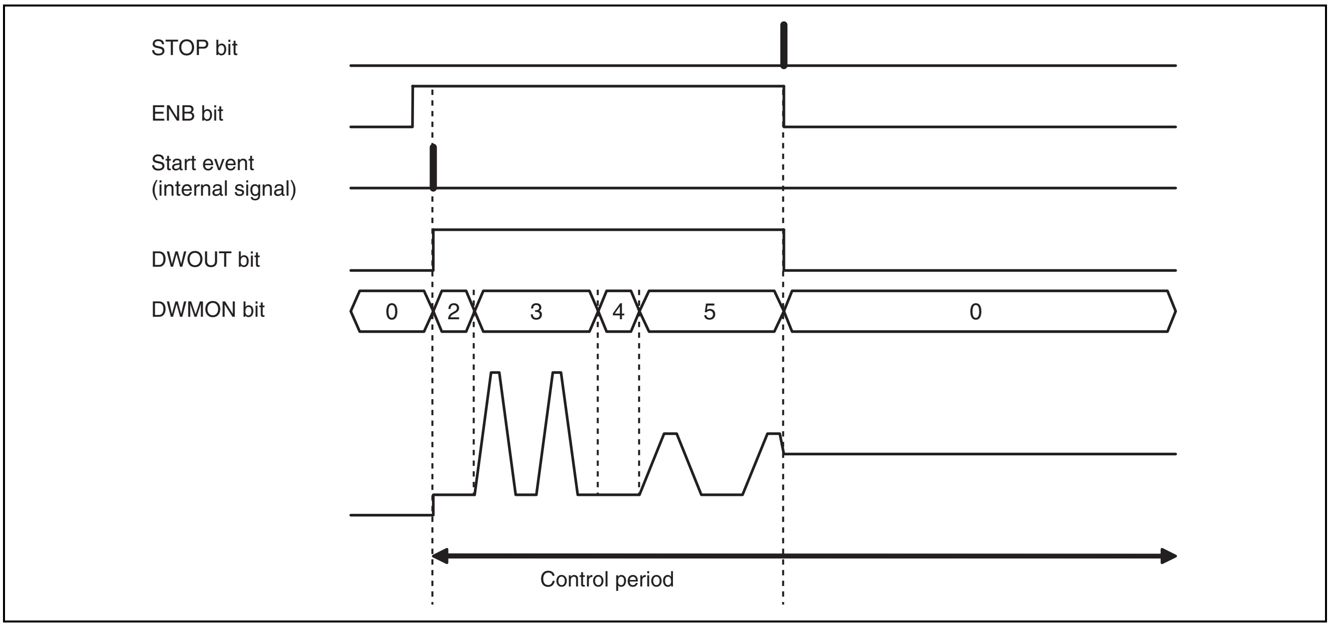


Figure 30.7 Relationship between PDAC Start Events and both Operation Enable and Operation Monitor (Forced Stop Processing)

When "1" is written to the forced stop bit (STOP), the PDICPT register ENB bit and the PDISTATUS register DWOUT bit will become "0". Do not set the ENB bit and the STOP bit to "1" at the same time.

Figure 30.8 shows the various modulation status information within the PDAC control period.

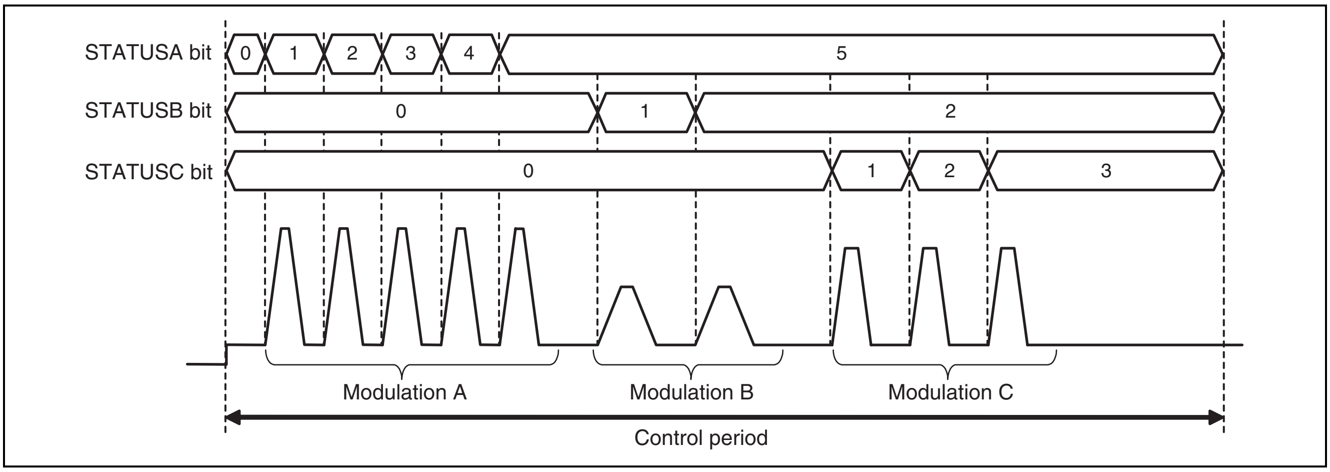


Figure 30.8 Modulation Status Information Within the PDAC Control Period

30.5.2 Modulation A Output Processing

In modulation A, the sequence of operations (1) to (4) can be repeated up to 15 times.

(1) Modulation A rise initial value output

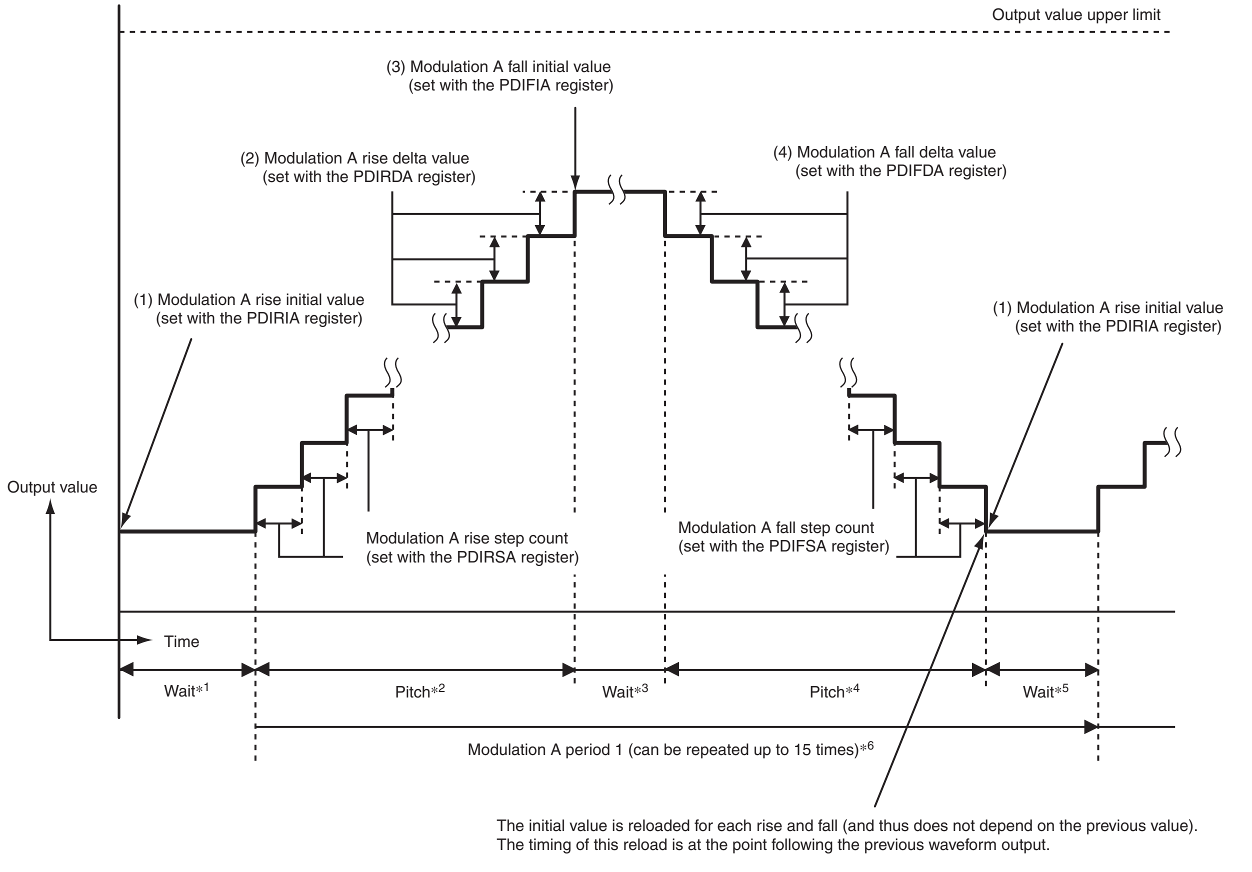
(2) Modulation A rise delta value output

(3) Modulation A fall initial value output

(4) Modulation A fall delta value output

Figure 30.9 shows an overview of modulation A waveform output operation.

See section 30.6, Timing Charts, for details on the set times.


Output value upper limit
(3) Modulation A fall initial value (set with the PDIFIA register)
(2) Modulation A rise delta value (set with the PDIRDA register)
(4) Modulation A fall delta value (set with the PDIFDA register)
(1) Modulation A rise initial value (set with the PDIRIA register)
(1) Modulation A rise initial value (set with the PDIRIA register)
Output value
Modulation A rise step count (set with the PDIRSA register)
Modulation A fall step count (set with the PDIFSA register)
Time
Wait*1
Pitch*2
Wait*3
Pitch*4
Wait*5
Modulation A period 1 (can be repeated up to 15 times)*6

The initial value is reloaded for each rise and fall (and thus does not depend on the previous value).
The timing of this reload is at the point following the previous waveform output.

Notes: *1 Wait time from the start event until the start of modulation A rise (set with the PDIWT0A register)
*2 Time required for the modulation A rise (set with the PDIRSA pitch counts (1 to 120 times) and the individual elapsed times)
*3 Wait time after the modulation A rise and before the modulation A fall (set with the PDIWT1A register)
*4 Time required for the modulation A fall (set with the PDIFSA pitch counts (1 to 120 times) and the individual elapsed times)
*5 Wait time after the modulation A fall and before the modulation A end (set with the PDIWT2A register)
*6 The number of waveforms output in the modulation A control period can be set to be up to 15 waveforms (repeated up to 15 times) with the PDIREPA register.

Figure 30.9 Operational Overview of the Modulation A Waveform Output

30.5.3 Modulation B Output Processing

In modulation B, the sequence of operations (1) to (4) can be repeated up to 31 times.

(1) Modulation B rise initial value output

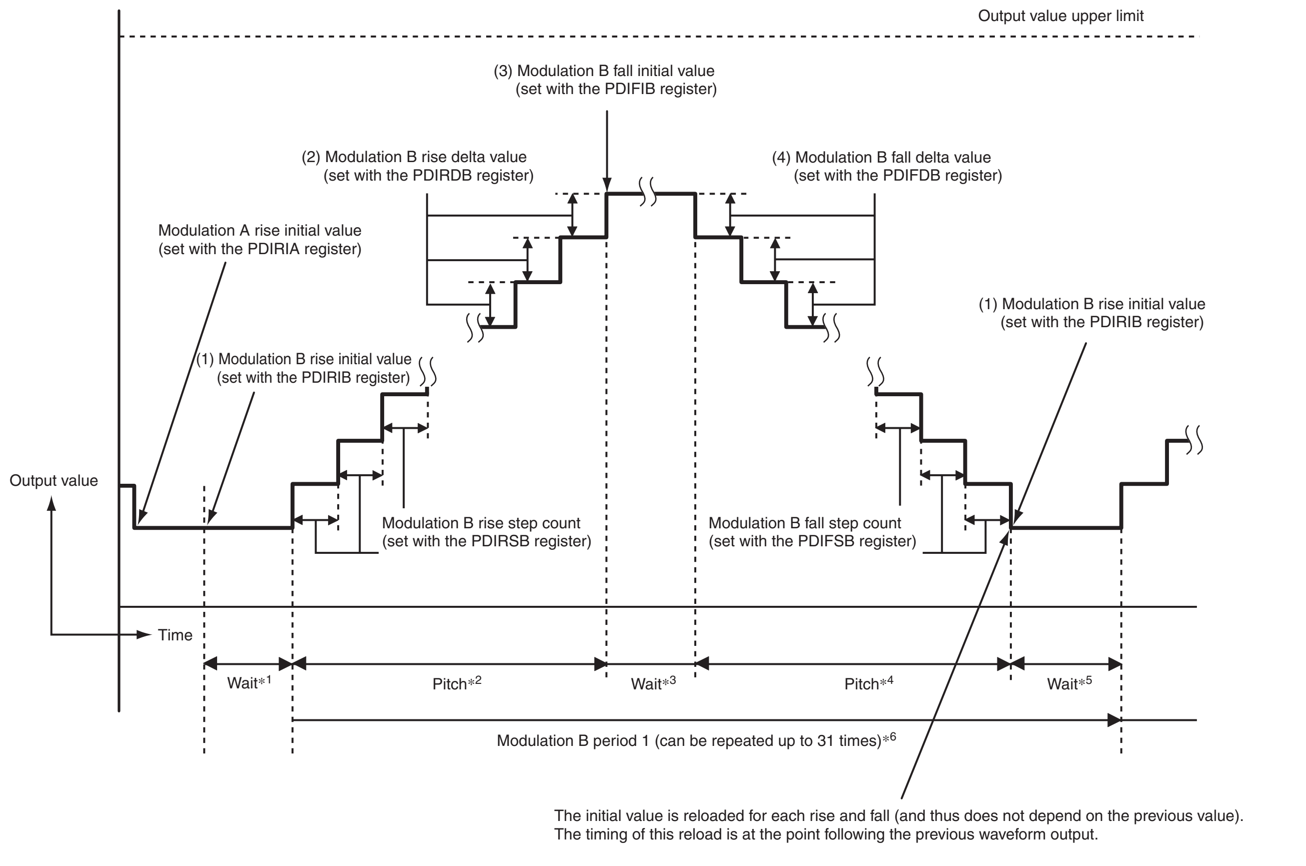
(2) Modulation B rise delta value output

(3) Modulation B fall initial value output

(4) Modulation B fall delta value output

Figure 30.10 shows an overview of modulation B waveform output operation.

See section 30.6, Timing Charts, for details on the set times.


Output value upper limit
(3) Modulation B fall initial value
(set with the PDIFIB register)
(2) Modulation B rise delta value
(set with the PDIRDB register)
(4) Modulation B fall delta value
(set with the PDIFDB register)
Modulation A rise initial value
(set with the PDIRIA register)
(1) Modulation B rise initial value
(set with the PDIRIB register)
(1) Modulation B rise initial value
(set with the PDIRIB register)
Output value
Modulation B rise step count
(set with the PDIRSB register)
Modulation B fall step count
(set with the PDIFSB register)
Time
Wait*1
Pitch*2
Wait*3
Pitch*4
Wait*5
Modulation B period 1 (can be repeated up to 31 times)*6
The initial value is reloaded for each rise and fall (and thus does not depend on the previous value).
The timing of this reload is at the point following the previous waveform output.

Notes: *1 Wait time from the end of modulation A fall waveform output until the start of modulation B rise (set with the PDIWT0B register)
*2 Time required for the modulation B rise (set with the PDIRSB pitch counts (1 to 200 times) and the individual elapsed times)
*3 Wait time after the modulation B rise and before the modulation B fall (set with the PDIWT1B register)
*4 Time required for the modulation B fall (set with the PDIFSB pitch counts (1 to 200 times) and the individual elapsed times)
*5 Wait time after the modulation B fall and before the modulation B end (set with the PDIWT2B register)
*6 The number of waveforms output in the modulation B control period can be set to be up to 31 waveforms (repeated up to 31 times) with the PDIREPB register.

Figure 30.10 Operational Overview of the Modulation B Waveform Output

30.5.4 Modulation C Output Processing

In modulation C, the sequence of operations (1) to (4) can be repeated up to 31 times.

(1) Modulation C rise initial value output

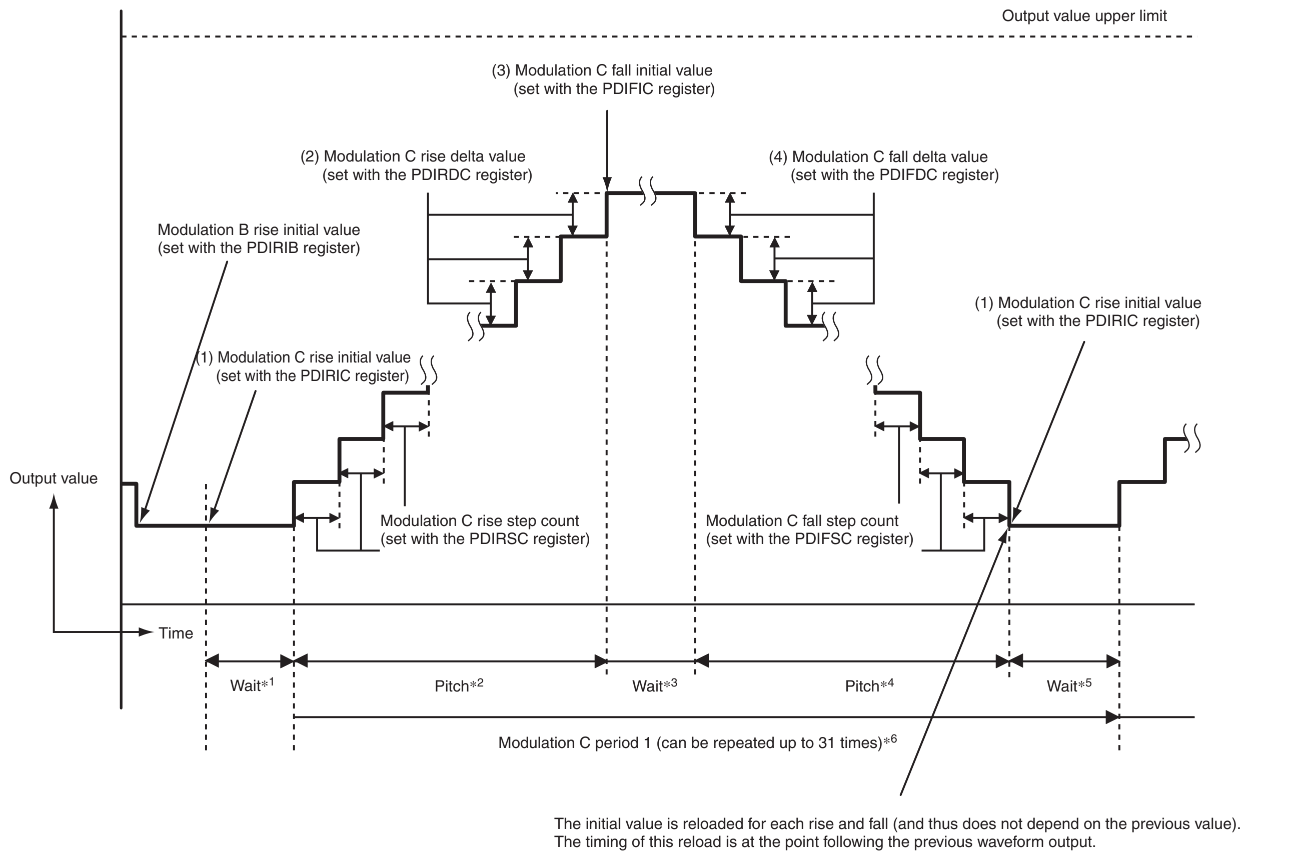
(2) Modulation C rise delta value output

(3) Modulation C fall initial value output

(4) Modulation C fall delta value output

Figure 30.11 shows an overview of modulation C waveform output operation.

See section 30.6, Timing Charts, for details on the set times.


Output value upper limit
(3) Modulation C fall initial value (set with the PDIFIC register)
(2) Modulation C rise delta value (set with the PDIRDC register)
(4) Modulation C fall delta value (set with the PDIFDC register)
Modulation B rise initial value (set with the PDIRIB register)
(1) Modulation C rise initial value (set with the PDIRIC register)
(1) Modulation C rise initial value (set with the PDIRIC register)
Output value
Modulation C rise step count (set with the PDIRSC register)
Modulation C fall step count (set with the PDIFSC register)
Time
Wait*1
Pitch*2
Wait*3
Pitch*4
Wait*5
Modulation C period 1 (can be repeated up to 31 times)*6

The initial value is reloaded for each rise and fall (and thus does not depend on the previous value). The timing of this reload is at the point following the previous waveform output.

Notes: *1 Wait time from the end of modulation B fall waveform output until the start of modulation C rise (set with the PDIWT0C register)
*2 Time required for the modulation C rise (set with the PDIRSC pitch counts (1 to 600 times) and the individual elapsed times)
*3 Wait time after the modulation C rise and before the modulation C fall (set with the PDIWT1C register)
*4 Time required for the modulation C fall (set with the PDIFSC pitch counts (1 to 600 times) and the individual elapsed times)
*5 Wait time after the modulation C fall and before the modulation C end (set with the PDIWT2C register)
*6 The number of waveforms output in the modulation C control period can be set to be up to 31 waveforms (repeated up to 31 times) with the PDIREPC register.

Figure 30.11 Operational Overview of the Modulation C Waveform Output

30.5.5 Coordination with Other Modules

The PDAC module generates output events to coordinate other modules (DRI, ATU-IIIS, and PSEL) with each of the waveform outputs.

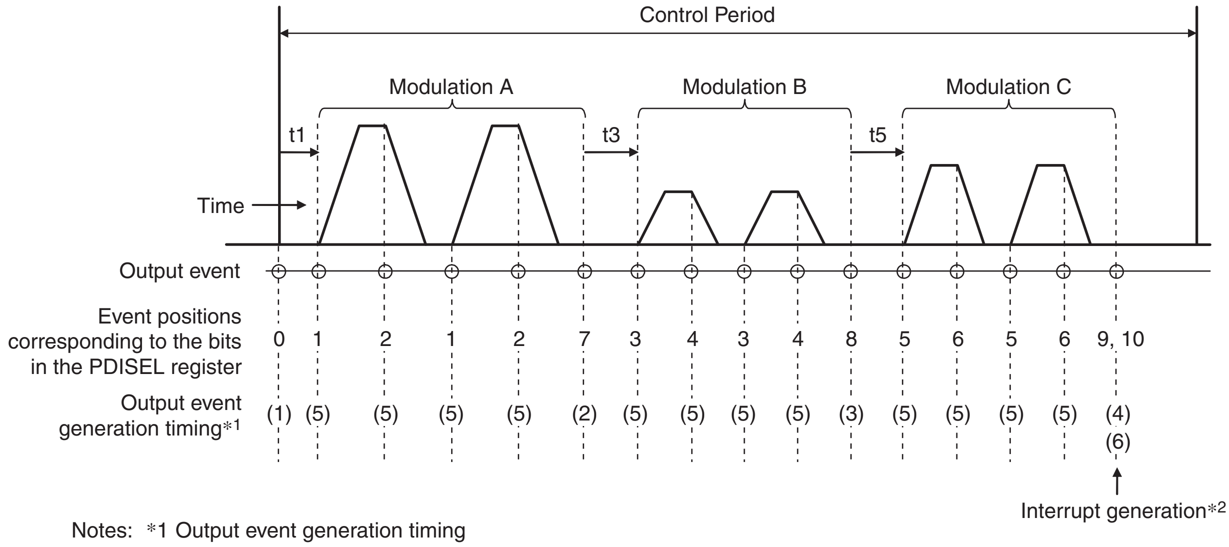
These events are generated immediately after operation starts, after each modulation (modulations A, B, and C), and for the rise and fall of the output waveform for each modulation, for a total of 10 sources plus a final modulation termination event, for a grand total of 11 sources.

It is possible to select whether these start sources are reported or not reported to other modules with the register corresponding to each signal (internal signal) (see the description of the output event selection register).

If a given modulation has no output (the repeat count for the corresponding modulation is 0), the output event following that waveform output will not be generated. However, if the waveform output for any of the modulations is used, the output event for final modulation completion will be generated.

When modulation output completes, the information (settings) for the next output waveform can be written by generating an interrupt. (Applications must not change the settings during waveform output.)

Figure 30.12 shows the timings of the output events and interrupt generation.



Notes: *1 Output event generation timing
(1) At start
(2) At modulation A output termination (including repeats), i.e. immediately before t3. This event is not generated if the repeat count is 0.
(3) At modulation B output termination (including repeats), i.e. immediately before t5. This event is not generated if the repeat count is 0.
(4) At modulation C output termination (including repeats). This event is not generated if the repeat count is 0.
(5) At rise and at fall of the modulation A, B, and C output waveforms
(6) Modulation output termination
*2 Interrupt generation conditions
Generated when all modulation waveform output (including wait times) has completed.

Figure 30.12 Output Event and Interrupt Generation Timing

30.5.6 PDAC Initialization Procedure

Figure 30.13 shows the initialization procedure used to start PDAC operation.

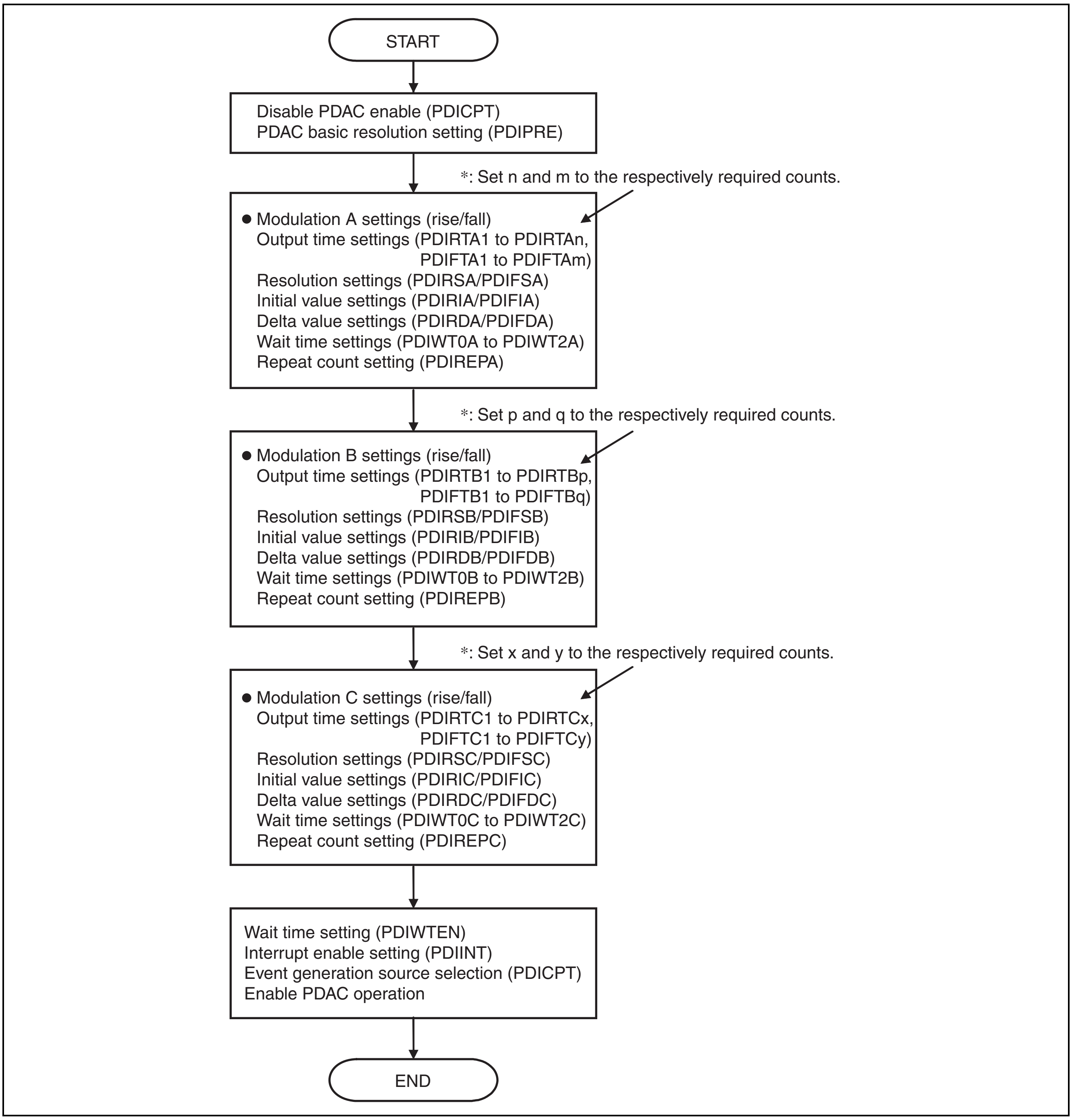
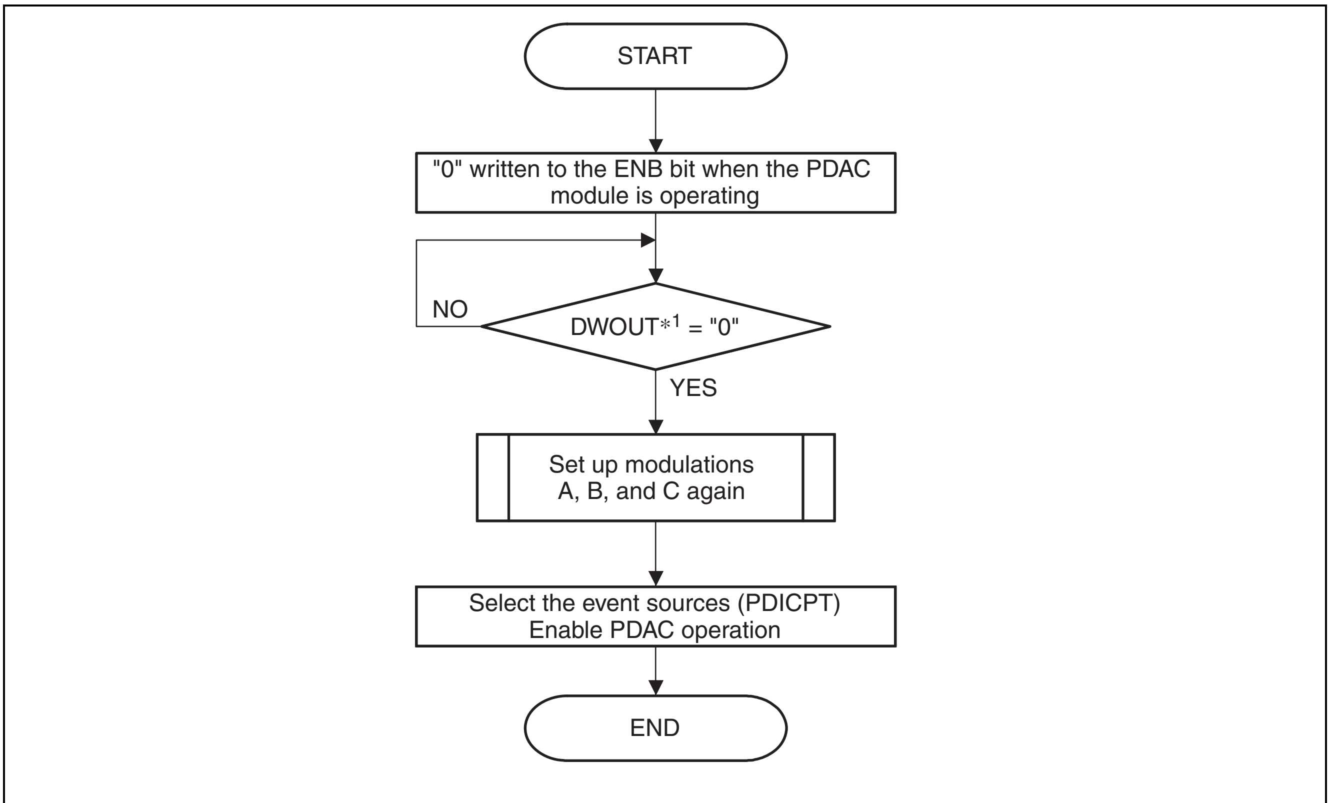

START
Disable PDAC enable (PDICPT)
PDAC basic resolution setting (PDIPRE)
*: Set n and m to the respectively required counts.
● Modulation A settings (rise/fall)
Output time settings (PDIRTA1 to PDIRTAn, PDIFTA1 to PDIFTAm)
Resolution settings (PDIRSA/PDIFSA)
Initial value settings (PDIRIA/PDIFIA)
Delta value settings (PDIRDA/PDIFDA)
Wait time settings (PDIWT0A to PDIWT2A)
Repeat count setting (PDIREPA)
*: Set p and q to the respectively required counts.
● Modulation B settings (rise/fall)
Output time settings (PDIRTB1 to PDIRTBp, PDIFTB1 to PDIFTBq)
Resolution settings (PDIRSB/PDIFSB)
Initial value settings (PDIRIB/PDIFIB)
Delta value settings (PDIRDB/PDIFDB)
Wait time settings (PDIWT0B to PDIWT2B)
Repeat count setting (PDIREPB)
*: Set x and y to the respectively required counts.
● Modulation C settings (rise/fall)
Output time settings (PDIRTC1 to PDIRTCx, PDIFTC1 to PDIFTCy)
Resolution settings (PDIRSC/PDIFSC)
Initial value settings (PDIRIC/PDIFIC)
Delta value settings (PDIRDC/PDIFDC)
Wait time settings (PDIWT0C to PDIWT2C)
Repeat count setting (PDIREPC)
Wait time setting (PDIWTEN)
Interrupt enable setting (PDIINT)
Event generation source selection (PDICPT)
Enable PDAC operation
END

Figure 30.13 Initialization Procedure (When starting for the first time)

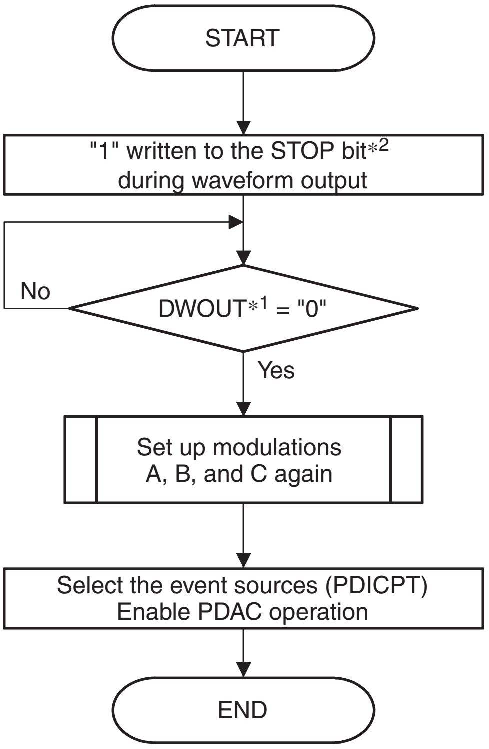
Figure 30.14 shows an example of the procedure for modifying the registers when the PDAC module is operating (with the PDICPT register ENB bit in the "1" state).


START
"0" written to the ENB bit when the PDAC module is operating
NO
DWOUT*1 = "0"
YES
Set up modulations A, B, and C again
Select the event sources (PDICPT) Enable PDAC operation
END

Note: *1 PDAC status register (PDISTATUS) modulation waveform output bit

Figure 30.14 Example Procedure for Register Modification During PDAC Operation

Figure 30.15 shows an example of the recovery procedure following a forced stop.



Notes: *1 PDAC status register (PDISTATUS) modulation waveform output bit
*2 Forced stop bit in the PDAC forced stop register (PDISTOP)

Figure 30.15 Example of Forced Stop and the Following Recovery Procedure

30.6 Timing Charts

Figure 30.16 shows the relationship between the registers and the waveform output.

- When the PDIRSA register is set to "H'03", the PDIRTA1 to PDIRTA3 registers will be valid.
- When the PDIFSA register is set to "H'04", the PDIFTA1 to PDIFTA4 registers will be valid.

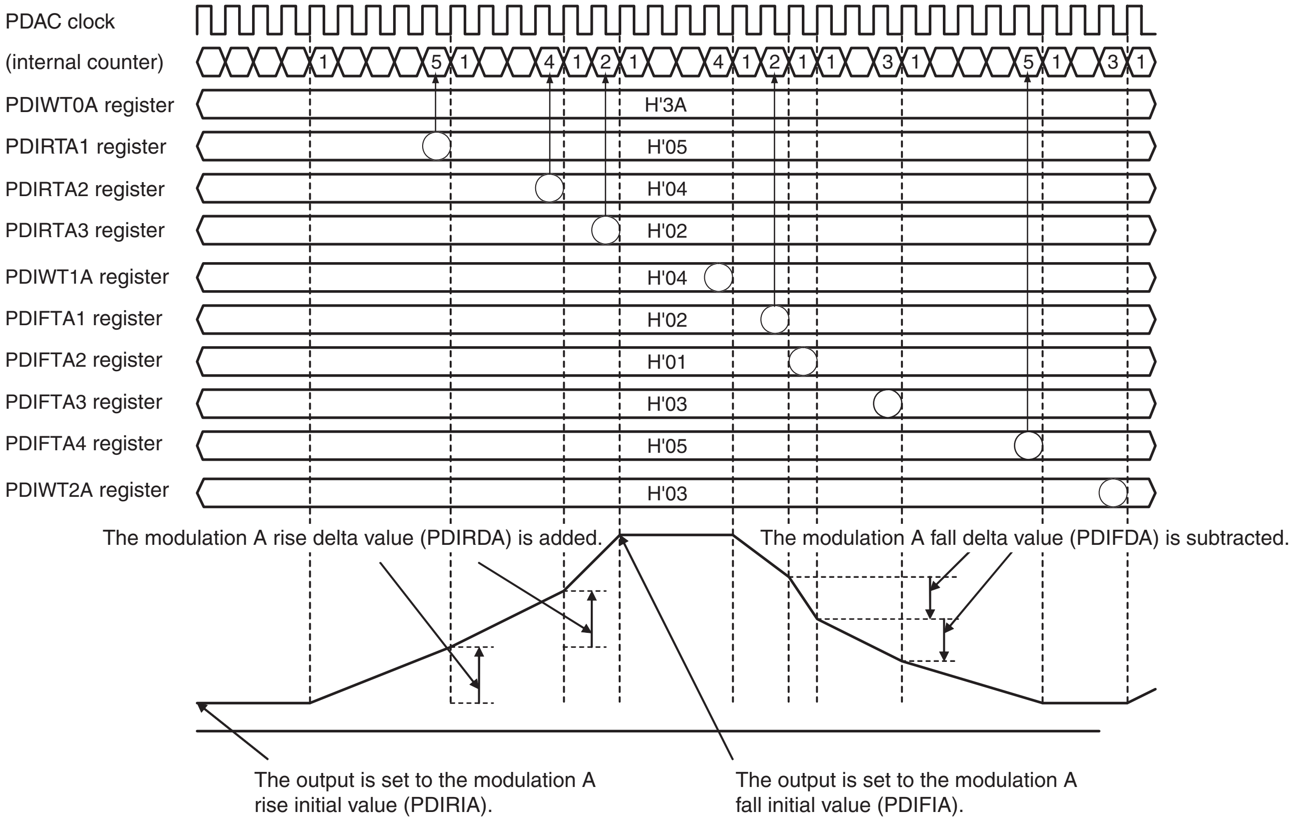


Figure 30.16 Modulation Waveform Output Timing (Example: Modulation A)

- Rise operation

  Based on the step count specified with the PDIRSA register (3 in this example), the PDIRTA1 to PDIRTA3 registers are processed as the rise time setting.

  Prior to rise waveform output, the initial value set with the PDIRIA register is applied.

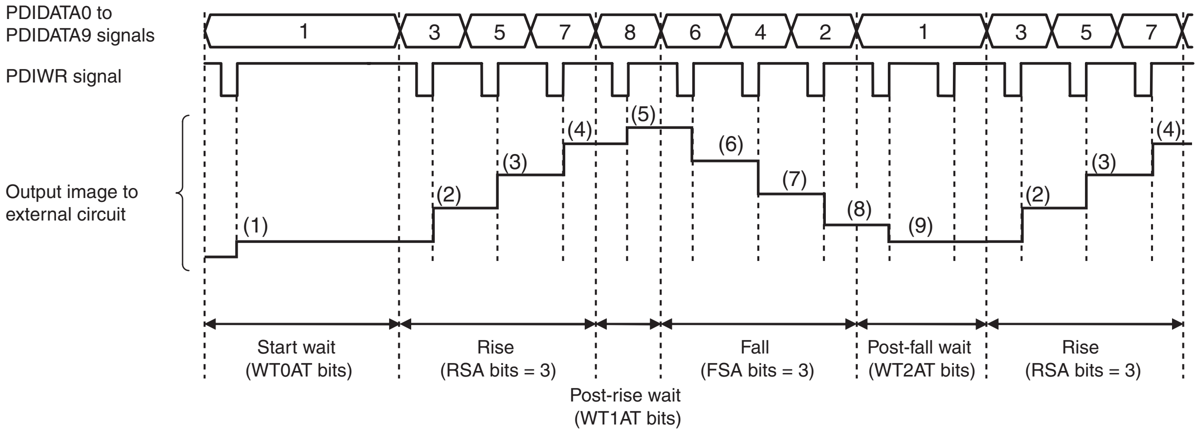
- Fall operation

  Based on the step count specified with the PDIFSA register (4 in this example), the PDIFTA1 to PDIFTA4 registers are processed as the fall time setting.

  Prior to fall waveform output, the initial value set with the PDIFIA register is applied.

Figure 30.17 shows the timing with which the initial value and delta value are reflected when there are wait times (when the WT0AE to WT2AE bits are "1").

- When two or more repetitions of modulation A are selected
- When "L" active is selected for the PDIWR signal

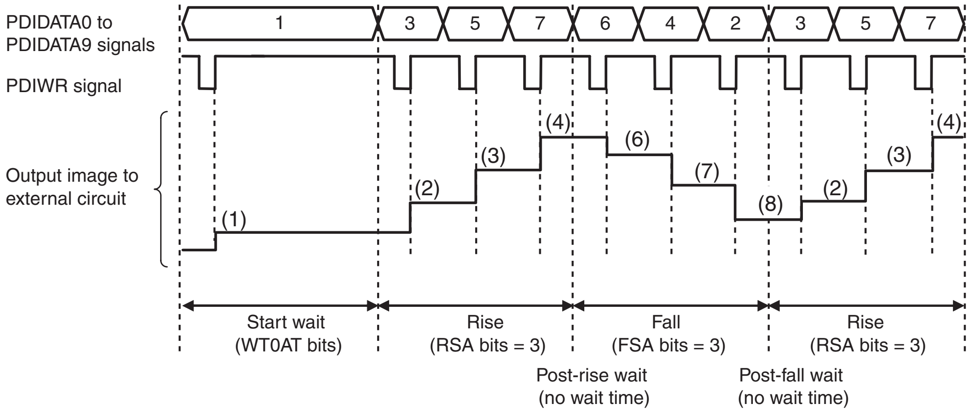


(1): "1" is set as the initial value.
(2) to (4): The output is set to 3, 5, and 7 by addition of the delta value (PDIRDn: in this case, 2)
(5): The output is set to 8 as the initial value.
(6) to (8): The output is set to 6, 4, and 2 by subtraction of the delta value (PDIFDn: in this case, 2)
(9) "1" is set as the initial value.

Figure 30.17 Initial Value and Delta Value Timing for Each Modulation (When wait times are used)

Figure 30.18 shows the timing with which the initial value and delta value are reflected when there are no wait times (when the WT1AE and WT2AE bits are "0").

- When two or more repetitions of modulation A are selected
- When "L" active is selected for the PDIWR signal



(1): "1" is set as the initial value.
(2) to (4): The output is set to 3, 5, and 7 by addition of the delta value (PDIRDn: in this case, 2)
(5): Since the WT1AE bit is "0" (no wait), this period is not present in the output waveform.
(6) The first subtraction is performed on the initial value 8, resulting in 6, to which the output is set.
(7) to (8): The output is set to 4 and 2 by subtraction of the delta value (PDIFDn: in this case, 2)
(9): Since the WT2AE bit is "0" (no wait), this period is not present in the output waveform.

Figure 30.18 Initial Value and Delta Value Timing for Each Modulation (When no wait times are used)

- When there are no wait times

  While there are no particular problems when, after the output is set to the initial value, the output is held for set wait time and then the fall (or rise) starts, if there is no wait time, the initial value setting and addition (or subtraction) processing must be performed at the same time.

  The processing performed is as follows. At the first waveform output, if the immediately preceding wait time setting is "0", the output is set to the combined value of the initial value and the added (or subtracted) value as the initial value.

30.7 Event Flag Wiring

The PDAC module outputs 8 event reporting signals, and of these, 4 are distributed to the ATU-IIIS module, 2 to PSEL, and 2 to DRI.

Four event signals are input to the PDAC and PSEL modules from the ATU-IIIS module.

Figure 30.19 shows the event flag wiring diagram.

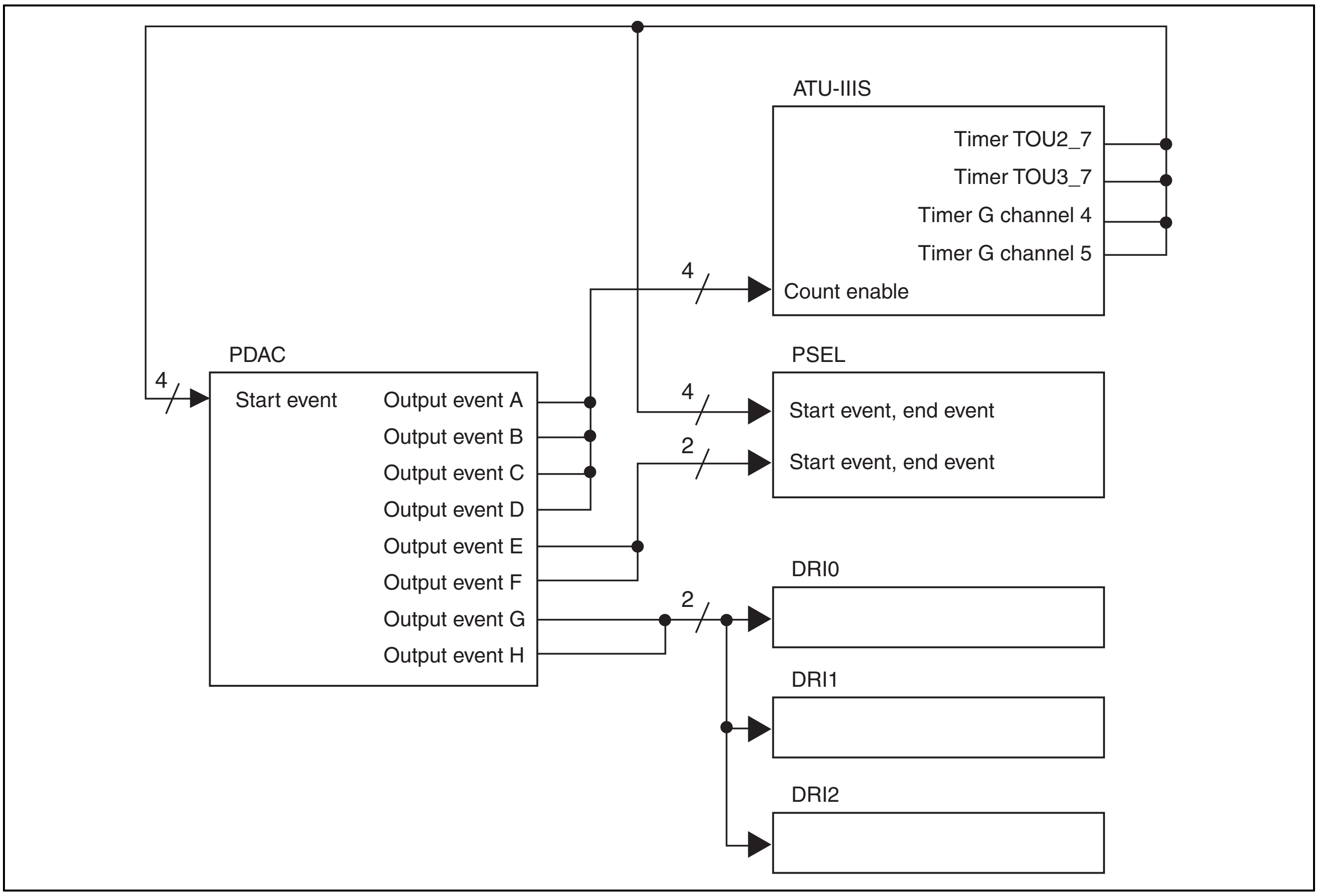


Figure 30.19 Event Flag Wiring Diagram

30.8 Usage Notes

- To use the PDAC, set the PDAC bit in the module stop register 0 (MSTPCR0) to "0" to enable PDAC and PSEL operations, and then set the PDAC related register. Otherwise, the clocks are not supplied to the PDAC module and PDAC operation is disabled even though the PDAC related register is set.
- Certain registers (PDIRTn, PDIFTn (n = A, B, C)) have no initial value stored. These registers must be written to before being used. Operation is not guaranteed if this module is used without writing these registers.
- If a start event occurs during waveform output (the PDISTATUS register DWOUT bit is "1"), that start event will be ignored (the start event is not accepted).
- Access to the following registers is restricted during waveform output (the PDISTATUS register DWOUT bit is "1").

  Except for the PDISTOP, PDICPT, and PDISTATUS registers, write accesses to all PDAC registers are illegal. However, only the PDICPT register ENB bit may be modified (but the CPT bit must be set to the same value it had the previous time).

  Also note that both read and write accesses to the following registers are illegal.

  Registers PDIRTA1 to PDIRTA120

  Registers PDIFTA1 to PDIFTA120

  Registers PDIRTB1 to PDIRTB200

  Registers PDIFTB1 to PDIFTB200

  Registers PDIRTC1 to PDIRTC600

  Registers PDIFTC1 to PDIFTC600

Section 31 Parallel Selector (PSEL)

31.1 Overview

This MCU includes a parallel selector circuit (the PSEL module) that can periodically modify an external selector or similar circuit. The PSEL module is activated by a start event and then periodically outputs the specified number of selector data items (output data values that are specified with register settings). The output can also be stopped by a stop event.

Table 31.1 lists the overview of the PSEL module.

Table 31.1 PSEL Module Overview

Item	Description
Selection data output	Selection data can be output to a selector or other circuit external to this MCU. Sixteen arbitrary values can be set as the selection data. This output can be iterated a number of times specified separately. The initial value can also be set separately.
Clock output	This module can output two clock systems (PSLCLKA and PSLCLKB) that are generated by dividing the peripheral clock (Pck) with a prescaler. After the start event, output can be started after a delay. The output polarity can also be selected.
Event output	The module is started by a start event, and then output of selection data for the specified number of channels continues. Events from software, the ATU-IIIS module (timers G or TOU), and the PDAC module can be used as the start event. After the PSEL module is started, it can be stopped by a stop event. Events from the ATU-IIIS module (timers G or TOU), the DRI module, and the PDAC module can be used as the end event.
Clear signal output	After the start event, a delayed clear signal can be output. Also, the clear signal active period, the output method (single or continuous), and the output polarity can be set.
Other functions	The PSEL module operating state can be determined from the PSEL status register (PSLSTATUS).

Figure 31.1 shows the block diagram of the PSEL module.

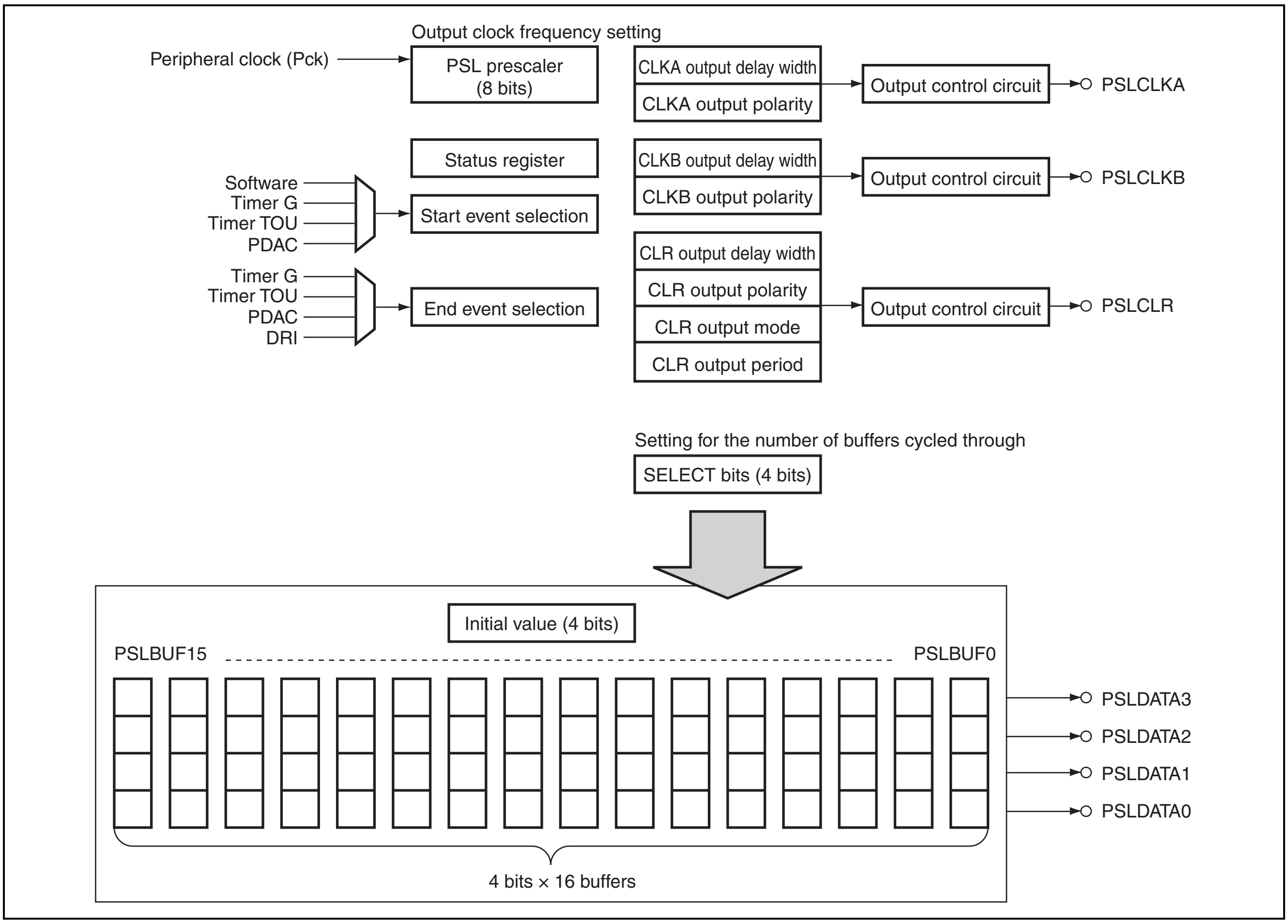


Figure 31.1 Block Diagram of PSEL

31.2 Input/Output Pins

Table 31.2 lists the PSEL module input and output pins.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 31.2 Pin Configuration

Pin Name	I/O	Function
PSLCLKA	Output	PSEL clock A output
PSLCLKB	Output	PSEL clock B output
PSLDATA3 to PSLDATA0	Output	PSEL select data output
PSLCLR	Output	PSEL clear pulse output

31.3 Register Descriptions

Table 31.3 lists the PSEL registers.

Table 31.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
PSEL event selection register	PSLCTRL	H'00	H'FFFF 3000	8, 16, 32	31-4
PSEL output clock divisor setting register	PSLPRE	H'FF	H'FFFF 3001	8, 16, 32	31-5
PSEL channel count selection register	PSLSEL	H'00	H'FFFF 3002	8, 16, 32	31-6
PSEL output polarity control register	PSLPOL	H'06	H'FFFF 3003	8, 16, 32	31-7
PSEL trigger register	PSLTRIG	H'00	H'FFFF 3004	8	31-8
PSEL status register	PSLSTATUS	H'00	H'FFFF 3006	8	31-8
PSEL clock A delay register	PSLDLYA	H'0001	H'FFFF 3008	16, 32	31-9
PSEL clock B delay register	PSLDLYB	H'0001	H'FFFF 300A	16, 32	31-9
PSEL clear delay period register	PSLCLRD	H'0001	H'FFFF 300C	16, 32	31-10
PSEL clear control register	PSLCLRC	H'0101	H'FFFF 300E	16, 32	31-11
PSEL data buffer 0/1 register	PSLDT0001	H'00	H'FFFF 3010	8	31-12
PSEL data buffer 2/3 register	PSLDT0203	H'00	H'FFFF 3011	8	31-12
PSEL data buffer 4/5 register	PSLDT0405	H'00	H'FFFF 3012	8	31-13
PSEL data buffer 6/7 register	PSLDT0607	H'00	H'FFFF 3013	8	31-13
PSEL data buffer 8/9 register	PSLDT0809	H'00	H'FFFF 3014	8	31-14
PSEL data buffer 10/11 register	PSLDT1011	H'00	H'FFFF 3015	8	31-14
PSEL data buffer 12/13 register	PSLDT1213	H'00	H'FFFF 3016	8	31-15
PSEL data buffer 14/15 register	PSLDT1415	H'00	H'FFFF 3017	8	31-15
PSEL data initial value register	PSLINIT	H'00	H'FFFF 3018	8	31-16

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

31.3.1 PSEL Event Selection Register (PSLCTRL)

The PSLCTRL register selects the PSEL module operation enabled/disabled state and the start and end events.

Do not change the event conditions (the START bits and END bits) during PSEL module operation (although PSEL module operation may be stopped by writing to the ENB bit). Note that if a start event and end event arrive at the same time, the end event takes precedence.

PSEL Event Selection Register (PSLCTRL) <P4 address: location H'FFFF 3000>

Bit:	7	6	5	4	3	2	1	0
	ENB	START			END			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7	ENB	0	R	W	Enable Bit 0: Selects the operation disabled state (The PSEL module is initialized and start and end events are not accepted.) 1: Selects the operation enabled state (Start and end events are accepted.) Note: • This setting takes effect after the next Pck cycle.
6 to 4	START	000	R	W	Start Event Selection Bits This field selects the source that generates start events. 000: Software control (See the description of the PSLTRIG register.) 001: ATU-IIIS timer TOU2_7 underflow 010: ATU-IIIS timer TOU3_7 underflow 011: ATU-IIIS timer G channel 4 compare match 100: ATU-IIIS timer G channel 5 compare match 101: PDAC event E 110: PDAC event F Other than above: Setting prohibited
3 to 0	END	0000	R	W	End Event Selection Bits This field selects the source that generates end events. 0000: No end event selected 0001: ATU-IIIS timer TOU2_7 underflow 0010: ATU-IIIS timer TOU3_7 underflow 0011: ATU-IIIS timer G channel 4 compare match 0100: ATU-IIIS timer G channel 5 compare match 0101: PDAC event E 0110: PDAC event F 1000: The DRI channel 0 DRI data acquisition disabled condition (When the DCPEN bit in the DRI0DCAPCNT register falls) 1001: The DRI channel 1 DRI data acquisition disabled condition (When the DCPEN bit in the DRI1DCAPCNT register falls) 1010: The DRI channel 2 DRI data acquisition disabled condition (When the DCPEN bit in the DRI2DCAPCNT register falls) Other than above: Setting prohibited

31.3.2 PSEL Output Clock Divisor Setting Register (PSLPRE)

The PSLPRE register sets the divisor for the PSLCLKA and PSLCLKB output clocks.

PSEL Output Clock Divisor Setting Register (PSLPRE) <P4 address: location H'FFFF 3001>

Bit:	7	6	5	4	3	2	1	0
	PRE							
After Reset:	1	1	1	1	1	1	1	1

<After Reset: H'FF>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	PRE	H'FF	R	W	Divisor Setting Bits This field sets the output clock divisor (1 to 255). The output clock frequency will be a peripheral clock (Pck) frequency/(divisor × 2). See table 31.4 for output examples for different setting values. The PRE bits must have a value in the range H'01 to H'FF. Operation is not guaranteed if the PRE field is set to "0".

Table 31.4 Output Frequency Setting Examples (When the peripheral clock frequency is 40 MHz)

PRE Field Setting	Output frequency
1	20 MHz
2	10 MHz
3	6.67 MHz
4	5 MHz
5	4 MHz
6	3.33 MHz
7	2.86 MHz
8	2.5 MHz
9	2.22 MHz
10	2 MHz
11	1.82 MHz
12	1.67 MHz
13	1.54 MHz
14	1.43 MHz
15	1.33 MHz
16	1.25 MHz
17	1.18 MHz
18	1.11 MHz
19	1.05 MHz
20	1 MHz
⋮	⋮
⋮	⋮
242	83 kHz
243	82 kHz
244	82 kHz
245	82 kHz
246	81 kHz
247	81 kHz
248	81 kHz
249	80 kHz
250	80 kHz
251	80 kHz
252	79 kHz
253	79 kHz
254	79 kHz
255	78 kHz

31.3.3 PSEL Channel Count Selection Register (PSLSEL)

The PSLSEL register sets the number of selection data output buffers. When the ENB bit in the PSLCTRL register is in the operation enabled state and a start event is accepted, the number of selection data items specified with the SELECT bits are output periodically (the output data values are set with the PSEL data buffer n registers).

PSEL Channel Count Selection Register (PSLSEL) <P4 address: location H'FFFF 3002>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	SELECT			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3 to 0	SELECT	0000	R	W	Channel Count Specification Bits This field specifies the number of channels output (0 to 15). Table 31.5 presents an overview of the channel data output operation.

Table 31.5 Channel Data Output Operation

SELECT Field Setting	Output Data Buffer
0	0
1	0→1 (→0→1…)
2	0→1→2 (→0→1…)
3	0→1→2→3 (→0→1…)
4	0→1→2→3→4 (→0→1…)
5	0→1→2→3→4→5 (→0→1…)
6	0→1→2→3→4→5→6 (→0→1…)
7	0→1→2→3→4→5→6→7 (→0→1…)
8	0→1→2→3→4→5→6→7→8 (→0→1…)
9	0→1→2→3→4→5→6→7→8→9 (→0→1…)
10	0→1→2→3→4→5→6→7→8→9→10 (→0→1…)
11	0→1→2→3→4→5→6→7→8→9→10→11 (→0→1…)
12	0→1→2→3→4→5→6→7→8→9→10→11→12 (→0→1…)
13	0→1→2→3→4→5→6→7→8→9→10→11→12→13 (→0→1…)
14	0→1→2→3→4→5→6→7→8→9→10→11→12→13→14 (→0→1…)
15	0→1→2→3→4→5→6→7→8→9→10→11→12→13→14→15 (→0→1…)

31.3.4 PSEL Output Polarity Control Register (PSLPOL)

The PSLPOL register sets the polarity of the PSLCLKA, PSLCLKB, and PSLCLR output signals.

The PSLPOL register should be set in the PSEL operation disabled state (when the ENB bit in the PSLCTRL register is "0"). The setting of the PSLPOL register is reflected at the next Pck cycle after the cycle in which the register is set.

PSEL Output Polarity Control Register (PSLPOL) <P4 address: location H'FFFF 3003>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	CLKA POL	CLKB POL	CLR POL
After Reset:	0	0	0	0	0	1	1	0

<After Reset: H'06>

Bit	Abbreviation	After Reset	R	W	Description
7 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	CLKAPOL	1	R	W	PSLCLKA Output Polarity Bit 0: Output with negative polarity 1: Output with positive polarity
1	CLKBPOL	1	R	W	PSLCLKB Output Polarity Bit 0: Output with negative polarity 1: Output with positive polarity
0	CLRPOL	0	R	W	PSLCLR Signal Output Polarity Bit 0: Output with negative polarity 1: Output with positive polarity

The polarities of the clock and clear outputs can be controlled with the PSEL output polarity control register (PSLPOL). Positive polarity refers to the state in which the signal is "L" after a reset and "H" when enabled (active). Similarly, negative polarity refers to the state in which the signal is "H" after a reset and "L" when enabled (active).

Figure 31.2 shows the PSEL module output signal polarity.

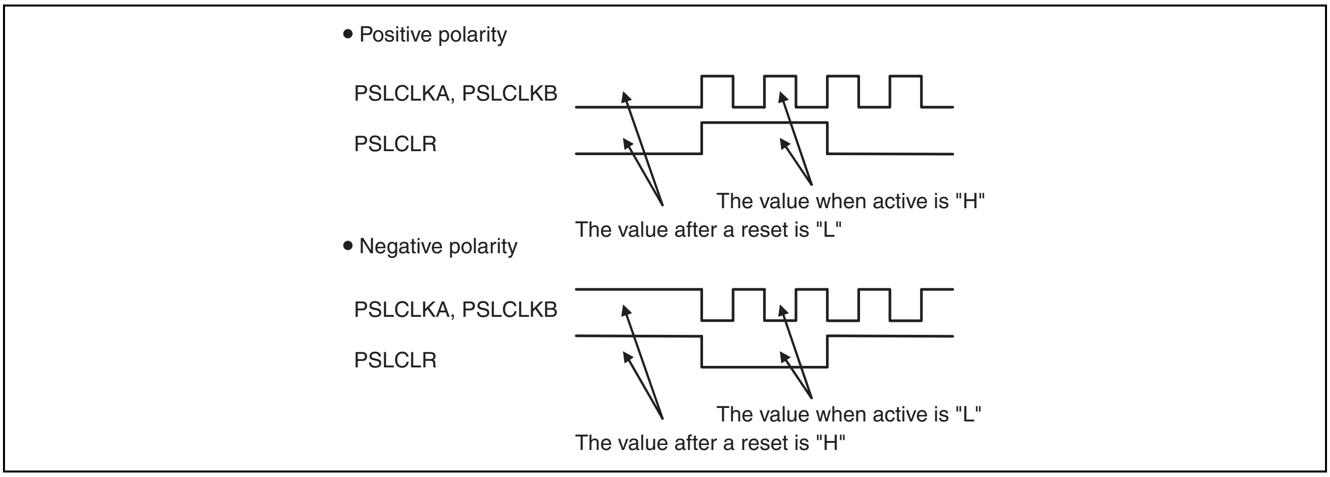

• Positive polarity
PSLCLKA, PSLCLKB
PSLCLR
The value when active is "H"
The value after a reset is "L"
• Negative polarity
PSLCLKA, PSLCLKB
PSLCLR
The value when active is "L"
The value after a reset is "H"

Figure 31.2 PSEL Output Signal Polarity

31.3.5 PSEL Trigger Register (PSLTRIG)

The PSLTRIG register generates a start event when written by software. To perform a software start, select software control (START = "B'000") with the PSLCTRL register START bits. Writing "1" to the ST bit is illegal when any mode other than software control is set. Note that the ST bit always reads out as "0", regardless of the value written.

PSEL Trigger Register (PSLTRIG) <P4 address: location H'FFFF 3004>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ST
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	ST	0	0	W	Software Trigger Bit A start event is generated when "1" is written to this bit. Always set the PSLCTRL register ENB bit to "1" before writing "1" to this bit. Writing "0" to this bit is invalid.

31.3.6 PSEL Status Register (PSLSTATUS)

The PSLSTATUS register is used to determine whether or not the PSEL module is operating. The STATUS bit is set to "1" when the PSLCTRL register ENB bit is set to the operation enabled state and a start event is accepted. The STATUS bit will be "0" if the ENB bit is set to the operation disabled state or an end event has been accepted.

PSEL Status Register (PSLSTATUS) <P4 address: location H'FFFF 3006>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	STATUS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 1	—	All 0	0	N	Reserved Bits These bits are always read as "0".
0	STATUS	0	R	N	Monitor Bit 0: The PSEL module is stopped 1: The PSEL module is operating (including the delay period until output starts)

31.3.7 PSEL Clock A Delay Register (PSLDLYA)

The PSLDLYA register sets the delay time from the start event until PSLCLKA output and channel data output start.

PSEL Clock A Delay Register (PSLDLYA) <P4 address: location H'FFFF 3008>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DLYA															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	DLYA	H'0001	R	W	PSLCLKA Delay Time Setting Bits This field sets the delay time from the acceptance of a start event until PSLCLKA and the channel data are output. Do not set this field to "H'0000". The delay time is given by 1 Pck × DLYA. Note that this time is not affected by the prescaler divisor setting.

31.3.8 PSEL Clock B Delay Register (PSLDLYB)

The PSLDLYB register sets the delay time from the start event until the start of PSLCLKB output.

PSEL Clock B Delay Register (PSLDLYB) <P4 address: location H'FFFF 300A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DLYB															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	DLYB	H'0001	R	W	PSLCLKB Delay Time Setting Bits This field sets the delay time from the acceptance of a start event until PSLCLKB is output. Do not set this field to "H'0000". The delay time is given by 1 Pck × DLYB. Note that this time is not affected by the prescaler divisor setting.

31.3.9 PSEL Clear Delay Period Register (PSLCLRD)

The PSLCLRD register sets the delay time from the start event until the PSLCLR signal active period.

PSEL Clear Delay Period Register (PSLCLRD) <P4 address: location H'FFFF 300C>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLRD															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

<After Reset: H'0001>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	CLRD	H'0001	R	W	PSLCLR Delay Time Setting Bits This field sets the delay time from the acceptance of a start event until the PSLCLR signal active period. Do not set this field to "H'0000". The delay time is given by 1 Pck × CLRD. Note that this time is not affected by the prescaler divisor setting.

31.3.10 PSEL Clear Control Register (PSLCLRC)

The PSLCLRC register sets the PSLCLR signal active period, single or continuous control, and the period used in continuous mode. The PSLCLR signal single mode operation consists of issuing the specified PSLCLR signal (pulse waveform) exactly once after the start event is accepted and waiting a delay period. Continuous mode operation consists of accepting a start event, waiting a delay period, and then iterating the signal output until the module is set to the operation disabled state (setting the ENB bit in the PSLCTRL register to "0") or an end event arrives. For details, see section 31.4.2, Timing Charts.

PSEL clear control register (PSLCLRC) <P4 address: location H'FFFF 300E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOD SEL	—	CYCLE						—	—	—	ACTIVE				
After Reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

<After Reset: H'0101>

Bit	Abbreviation	After Reset	R	W	Description
15	MODSEL	0	R	W	Single Mode/Continuous Mode Switching Bit 0: The PSLCLR signal is output as a single event. 1: The PSLCLR signal is output continuously with a certain period.
14	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
13 to 8	CYCLE	000001	R	W	Continuous Period Setting Bits This field specifies the generation period for PSLCLR signal continuous output. Set this field to a value in the range "1" to "63". Do not set this field to "0". When the MODSEL bit is "1", set this field to a value larger than the ACTIVE bits value. (Operation is not guaranteed for any relationship other than CYCLE > ACTIVE.) The CYCLE setting is ignored when the MODSEL bit is "0". The generation period is given by the formula CYCLE/clock A output frequency. Note that the output frequency is affected by the divisor set with the PSLPRE register.
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4 to 0	ACTIVE	00001	R	W	PSLCLR Active Period Setting Bits This field specifies the PSLCLR signal active period after a start event has been accepted and the PSLCLR signal wait time has elapsed. Set this field to a value in the range "1" to "31". Operation is not guaranteed if this field is set to "0". The active period is given by the formula ACTIVE/clock A output frequency. Note that the output frequency is affected by the divisor set with the PSLPRE register.

31.3.11 PSEL Data Buffer 0/1 Register (PSLDT0001)

The PSLDT0001 register sets the output channel data values.

PSEL Data Buffer 0/1 Register (PSLDT0001) <P4 address: location H'FFFF 3010>

Bit:	7	6	5	4	3	2	1	0
	DATA0				DATA1			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA0	0000	R	W	Data Buffer 0 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA1	0000	R	W	Data Buffer 1 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.12 PSEL Data Buffer 2/3 Register (PSLDT0203)

The PSLDT0203 register sets the output channel data values.

PSEL Data Buffer 2/3 Register (PSLDT0203) <P4 address: location H'FFFF 3011>

Bit:	7	6	5	4	3	2	1	0
	DATA2				DATA3			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA2	0000	R	W	Data Buffer 2 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA3	0000	R	W	Data Buffer 3 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.13 PSEL Data Buffer 4/5 Register (PSLDT0405)

The PSLDT0405 register sets the output channel data values.

PSEL Data Buffer 4/5 Register (PSLDT0405) <P4 address: location H'FFFF 3012>

Bit:	7	6	5	4	3	2	1	0
	DATA4				DATA5			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA4	0000	R	W	Data Buffer 4 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA5	0000	R	W	Data Buffer 5 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.14 PSEL Data Buffer 6/7 Register (PSLDT0607)

The PSLDT0607 register sets the output channel data values.

PSEL Data Buffer 6/7 Register (PSLDT0607) <P4 address: location H'FFFF 3013>

Bit:	7	6	5	4	3	2	1	0
	DATA6				DATA7			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA6	0000	R	W	Data Buffer 6 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA7	0000	R	W	Data Buffer 7 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.15 PSEL Data Buffer 8/9 Register (PSLDT0809)

The PSLDT0809 register sets the output channel data values.

PSEL Data Buffer 8/9 Register (PSLDT0809) <P4 address: location H'FFFF 3014>

Bit:	7	6	5	4	3	2	1	0
	DATA8				DATA9			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA8	0000	R	W	Data Buffer 8 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA9	0000	R	W	Data Buffer 9 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.16 PSEL Data Buffer 10/11 Register (PSLDT1011)

The PSLDT1011 register sets the output channel data values.

PSEL Data Buffer 10/11 Register (PSLDT1011) <P4 address: location H'FFFF 3015>

Bit:	7	6	5	4	3	2	1	0
	DATA10				DATA11			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA10	0000	R	W	Data Buffer 10 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA11	0000	R	W	Data Buffer 11 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.17 PSEL Data Buffer 12/13 Register (PSLDT1213)

The PSLDT1213 register sets the output channel data values.

PSEL Data Buffer 12/13 Register (PSLDT1213) <P4 address: location H'FFFF 3016>

Bit:	7	6	5	4	3	2	1	0
	DATA12				DATA13			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA12	0000	R	W	Data Buffer 12 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA13	0000	R	W	Data Buffer 13 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.18 PSEL Data Buffer 14/15 Register (PSLDT1415)

The PSLDT1415 register sets the output channel data values.

PSEL Data Buffer 14/15 Register (PSLDT1415) <P4 address: location H'FFFF 3017>

Bit:	7	6	5	4	3	2	1	0
	DATA14				DATA15			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	DATA14	0000	R	W	Data Buffer 14 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15
3 to 0	DATA15	0000	R	W	Data Buffer 15 This field specifies the output channel data. 0000: 0 0001: 1 0010: 2 : 1101: 13 1110: 14 1111: 15

31.3.19 PSEL Data Initial Value Register (PSLINIT)

The PSLINIT register sets the initial values of the selection data output in the state prior to the start of PSEL operation. This value is selected as the channel data during the period from the point the PSEL module is set to the operation enabled state until a start event is accepted. After PSEL start, the initial value set with the INIT bits is also output from the selection data output pins when the end event selected with the END bits in the PSLCTRL register occurs.

The value of the INIT bits will be reflected in the module output at the following timings.

- Write accesses to the PSLINIT register at times other than during the output operation period
- Start trigger acceptance
- End trigger acceptance
- When "0" (the operation disabled state) is written to the ENB bit in the PSLCTRL register during PSEL operation

PSEL Data Initial Value Register (PSLINIT) <P4 address: location H'FFFF 3018>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	INIT			
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3 to 0	INIT	0000	R	W	Selection data initial value specification bits Set this field to the initial value of the selection data to be output (a value in the range 0 to 15).

31.4 Operation

31.4.1 Overview

PSEL module operation is started by a start event and the module continues to output the selected channel data during the period until the enable state is invalidated or an end event arrives. (For the channel data, the values set in data buffer registers 0 to 15 are output in sequence.)

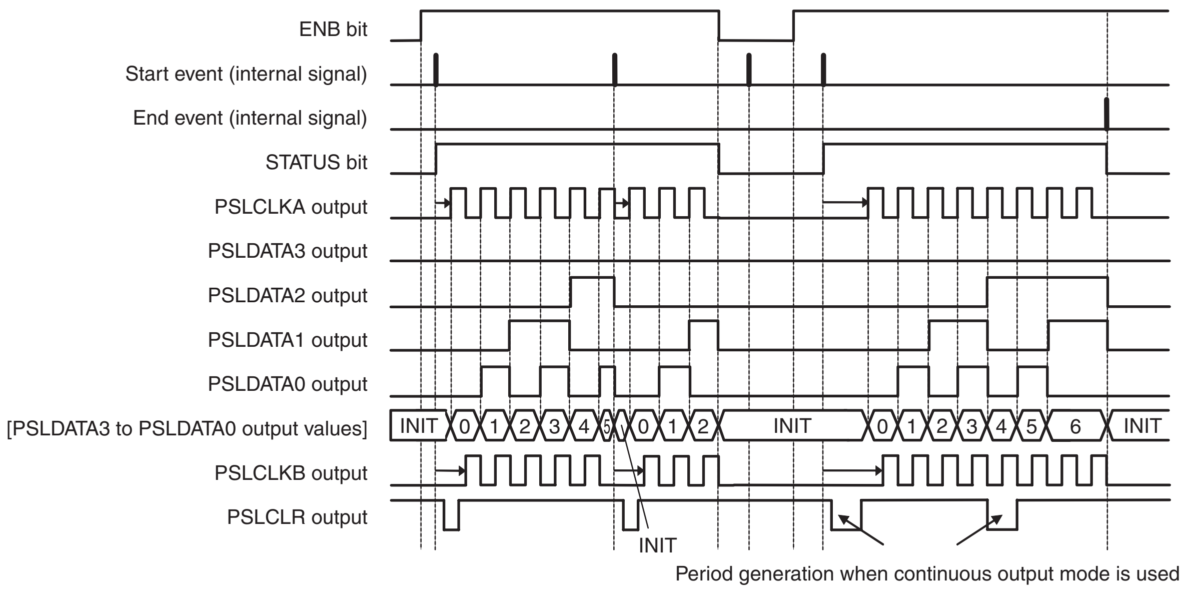
For the PSLCLKA pin and the PSLDATA3 to PSLDATA0 pins, the PSLCLKA output from the start event and the delay time until channel data output start are controlled by the PSEL clock A delay register (PSLDLYA).

For the PSLCLKB pin, the delay time from the start event until PSLCLKB output start is controlled by the PSEL clock B delay register (PSLDLYB). For the PSLCLR pin, the delay time from the start event is controlled by the PSEL clear delay period register (PSLCLRD) and the active period, the single/continuous control, and the generation period in continuous mode are controlled by the PSEL clear control register (PSLCLRC).

If another start event arrives during operation, the channel data being output is reset and output is restarted from 0. (The clock divisor position is also initialized.) The clear signal being asserted is immediately negated, and then reasserted after the delay time set by the PSEL clear delay period register (PSLCLRD). Note that if a start event or the operation enabled setting of the ENB bit in the PSLCTRL register, and an end event arrive at the same time, the end event takes precedence. (The PSEL module does not operate).

Figure 31.3 shows an overview of PSEL module operation.

- The channel count is set to 6 or higher
- PSLCLKA is set to positive polarity
- PSLCLKB is set to positive polarity



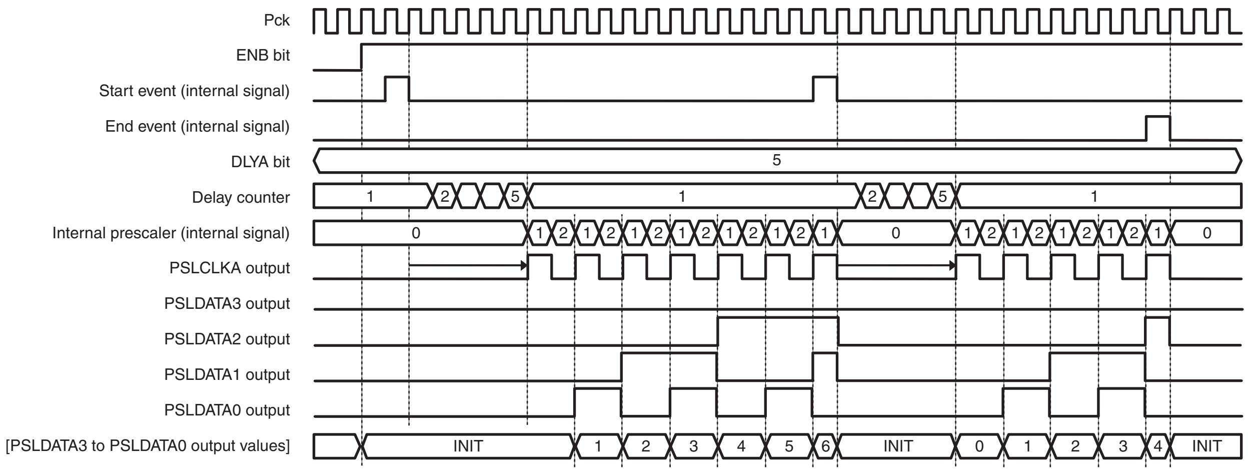
Notes:
- The arrows shown for the PSLCLKA and PSLCLKB signals indicate the effects of delays.
- The places where a number is shown indicate that the value of the corresponding data buffer n register is output.

Figure 31.3 PSEL Operation Overview

31.4.2 Timing Charts

Figure 31.4 shows the PSEL module data output timing.

- PSLCLKA and PSLCLKB are set to Pck/2
- The channel count is set to 6 or higher
- PSLCLKA is set to positive polarity
- The PSLCLKA delay time is set to 5Pck



Notes: • The arrows shown for the PSLCLKA signal indicate the effects of delays.
• The places where a number is shown indicate that the value of the corresponding data buffer n register is output.

Figure 31.4 PSEL Data Output Timing

Figure 31.5 shows the PSLCLR signal output timing.

- PSLCLKA and PSLCLKB are set to Pck/2
- The PSLCLR signal is set to output with a continuous period
- The continuous period setting is set to 8
- The PSLCLR delay time is set to 5Pck
- The PSLCLR signal active period setting is set to 3
- PSLCLR is set to negative polarity

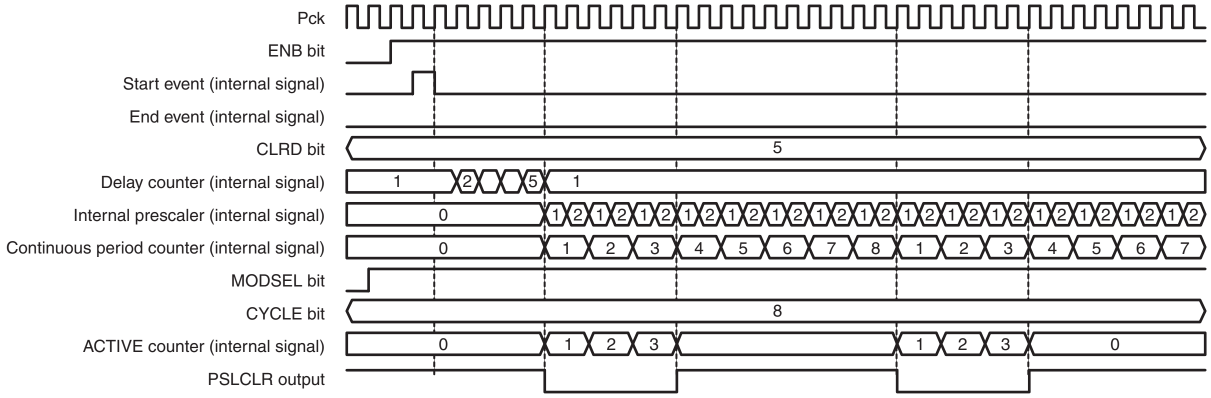

Pck
ENB bit
Start event (internal signal)
End event (internal signal)
CLRD bit
5
Delay counter (internal signal)
1 2 5 1
Internal prescaler (internal signal)
0 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2
Continuous period counter (internal signal)
0 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7
MODSEL bit
CYCLE bit
8
ACTIVE counter (internal signal)
0 1 2 3 1 2 3 0
PSLCLR output

Figure 31.5 PSLCLR Signal Output Timing

31.5 Usage Notes

31.5.1 Module Stop Function Setting before Using the PSEL

To use the PSEL, set the PDAC bit in the module stop register 0 (MSTPCR0) to "0" to enable PDAC and PSEL operations, and then set the PSEL related register. Otherwise, the clocks are not supplied to the PSEL module and PSEL operation is disabled even though the PSEL related register is set.

31.5.2 Notes on Register Access During PSEL Module Operation

The following restrictions on register access apply during PSEL module operation (when the STATUS bit in the PSLSTATUS register is "1").

Except for the PSLCTRL, and PSLTRIG registers, write access to all PSEL module registers is illegal. However, only the ENB bit in the PSLCTRL register may be changed (the START bit and the END bit must be set to their previous values).

This page is blank for reasons of layout.

Section 32 FlexRay Module

32.1 Overview

The FlexRay module of the SH7455 Group, FlexRay protocol specification v2.1-compliant, consists 2 channels (for Channels A and B). Table 32.1 lists the specifications of the FlexRay module.

The SH7456 Group has no FlexRay module. Addresses H'FFBF F000 to H'FFBF FFFF of the FlexRay-related registers are reserved areas. The read value should be undefined. Do not write to these registers.

Table 32.1 FlexRay Module Specifications*[1]

Item	Specification
Protocol	Compliant with FlexRay protocol specification v2.1
Channels	2 channels (for channels A and B)
Message RAM	8 Kbytes of Message RAM for storage of e.g. 128 message buffers with max. 48 byte data section or up to 30 message buffers with 254 byte data section
Receive FIFO	Up to 128 message buffers configurable (sharing message RAM with receive buffer)
Message filtering	Filtering for slot counter, cycle counter, and channel ID Configurable in transmit/receive buffer
NM data transmission/reception	Up to 12 bytes of NM vector supported • Interrupts generated by the change of NM vector
Timers	Timer 0: Absolute timer Timer 1: Relative timer Stop watch timer: Capture of cycle counter and MT counter values • MT is supported for timer configuration
Operating clock*[2]	• Peripheral A clock (PAck) Used for any units other than the protocol controllers • FlexRay clock (FRck) Used for the protocol controllers
Bit rate	Bit Rate = $\frac{1}{\text{Sample clock} \times 8}$ Sample clock divided by 1 to 2 of FRck according to bit settings of the BRP0 bit in the FRPRTC1 register
Forced reset	FlexRay module is forcibly reset

Notes: *1 FlexRay is a trademark of Daimler AG in Japan and other countries.

*2 Clock setting should be done in DEFAULT_CONFIG state.

Figure 32.1 shows a block diagram of the FlexRay module.

The CPU can access to message RAM via Input Buffer (IBF) or Output Buffer (OBF).

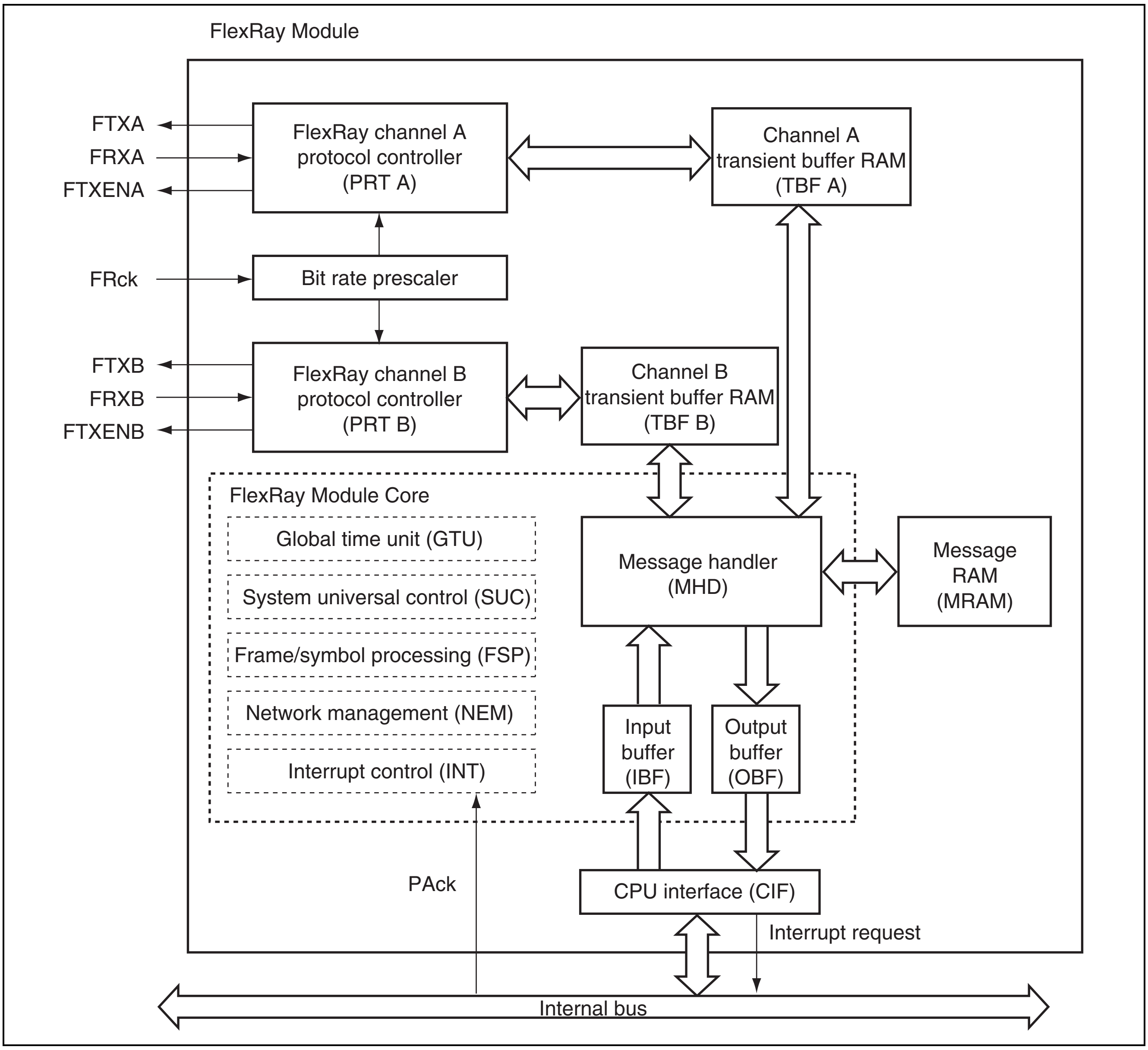


Figure 32.1   Block Diagram of FlexRay Module

Table 32.2 lists the FlexRay module pins.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 18, I/O Ports and Pin Function Unit.

Table 32.2 Pin Configuration

Pin	Description
FRXA	Channel A receive data input pin
FTXA	Channel A transmit data output pin
FTXENA	Channel A transmit enable pin High: transmit disabled Low: transmit enabled
FRXB	Channel B receive data input pin
FTXB	Channel B transmit data output pin
FTXENB	Channel B transmit enable pin High: transmit disabled Low: transmit enabled

32.2 Register Descriptions

Table 32.3 lists the register configuration of the FlexRay module.

The SH7456 Group has no FlexRay module. Addresses H'FFBF F000 to H'FFBF FFFF of the FlexRay-related registers are reserved areas. The read value should be undefined. Do not write to these registers.

Table 32.3 Register Configration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
FlexRay Operation Control Register	FXROC	H'04	H'FFBF F004	8	32-12
FlexRay Timer Interrupt Request Status Register	FXRTISR	H'00	H'FFBF F00C	8	32-50
FlexRay Timer Interrupt Enable Register	FXRTIER	H'00	H'FFBF F00D	8	32-51
FlexRay Lock Register	FRLCK	H'00	H'FFBF F01F	8	32-15
FlexRay Error Interrupt Register	FREIR	H'0000 0000	H'FFBF F020	32	32-17
FlexRay Status Interrupt Register	FRSIR	H'0000 0000	H'FFBF F024	32	32-21
FlexRay Error Interrupt Line Select Register	FREILS	H'0000 0000	H'FFBF F028	32	32-24
FlexRay Status Interrupt Line Select Register	FRSILS	H'0303 FFFF	H'FFBF F02C	32	32-26
FlexRay Error Interrupt Enable Set Register	FREIES	H'0000 0000	H'FFBF F030	32	32-28
FlexRay Error Interrupt Enable Reset Register	FREIER	H'0000 0000	H'FFBF F034	32	32-32
FlexRay Status Interrupt Enable Set Register	FRSIES	H'0000 0000	H'FFBF F038	32	32-36
FlexRay Status Interrupt Enable Reset Register	FRSIER	H'0000 0000	H'FFBF F03C	32	32-40
FlexRay Interrupt Line Enable Register	FRILE	H'00	H'FFBF F043	8	32-44
FlexRay Timer0 Configuration Register	FRT0C	H'0000 0000	H'FFBF F044	32	32-45
FlexRay Timer 1 Configuration Register	FRT1C	H'0002 0000	H'FFBF F048	32	32-46
FlexRay Stop Watch Register 1	FRSTPW1	H'0000 0000	H'FFBF F04C	32	32-47
FlexRay Stop Watch Register 2	FRSTPW2	H'0000 0000	H'FFBF F050	32	32-49
FlexRay SUC Configuration Register 1	FRSUCC1	H'0C40 1080	H'FFBF F080	32	32-52
FlexRay SUC Configuration Register 2	FRSUCC2	H'0100 0504	H'FFBF F084	32	32-58
FlexRay SUC Configuration Register 3	FRSUCC3	H'11	H'FFBF F08B	8	32-59
FlexRay NEM Configuration Register	FRNEMC	H'00	H'FFBF F08F	8	32-60
FlexRay PRT Configuration Register 1	FRPRTC1	H'084C 0633	H'FFBF F090	32	32-61
FlexRay PRT Configuration Register 2	FRPRTC2	H'0F2D 0A0E	H'FFBF F094	32	32-63
FlexRay MHD Configuration Register	FRMHDC	H'0000 0000	H'FFBF F098	32	32-64
FlexRay GTU Configuration Register 1	FRGTUC1	H'0000 0280	H'FFBF F0A0	32	32-65
FlexRay GTU Configuration Register 2	FRGTUC2	H'0002 000A	H'FFBF F0A4	32	32-66
FlexRay GTU Configuration Register 3	FRGTUC3	H'0202 0000	H'FFBF F0A8	32	32-67
FlexRay GTU Configuration Register 4	FRGTUC4	H'0008 0007	H'FFBF F0AC	32	32-68
FlexRay GTU Configuration Register 5	FRGTUC5	H'0E00 0000	H'FFBF F0B0	32	32-69

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
FlexRay GTU Configuration Register 6	FRGTUC6	H'0002 0000	H'FFBF F0B4	32	32-70
FlexRay GTU Configuration Register 7	FRGTUC7	H'0002 0004	H'FFBF F0B8	32	32-71
FlexRay GTU Configuration Register 8	FRGTUC8	H'0000 0002	H'FFBF F0BC	32	32-72
FlexRay GTU Configuration Register 9	FRGTUC9	H'0000 0101	H'FFBF F0C0	32	32-73
FlexRay GTU Configuration Register 10	FRGTUC10	H'0002 0005	H'FFBF F0C4	32	32-74
FlexRay GTU Configuration Register 11	FRGTUC11	H'0000 0000	H'FFBF F0C8	32	32-75
FlexRay CC Status Vector Register	FRCCSV	Undefined	H'FFBF F100	32	32-76
FlexRay CC Error Vector Register	FRCCEV	H'0000	H'FFBF F106	16	32-80
FlexRay Slot Counter Value Register	FRSCV	H'0000 0000	H'FFBF F110	32	32-81
FlexRay Macrotick and Cycle Counter Value Register	FRMTCCV	H'0000 0000	H'FFBF F114	32	32-82
FlexRay Rate Correction Value Register	FRRCV	H'0000	H'FFBF F11A	16	32-83
FlexRay Offset Correction Value	FROCV	H'0000 0000	H'FFBF F11C	32	32-83
FlexRay Sync Frame Status Register	FRSFS	H'0000 0000	H'FFBF F120	32	32-84
FlexRay Symbol Window and NIT Status Register	FRSWNIT	H'0000	H'FFBF F126	16	32-86
FlexRay Aggregated Channel Status Register	FRACS	H'0000	H'FFBF F12A	16	32-88
FlexRay Even Sync ID 1 Register	FRESID1	H'0000	H'FFBF F132	16	32-90
FlexRay Even Sync ID 2 Register	FRESID2	H'0000	H'FFBF F136	16	32-90
FlexRay Even Sync ID 3 Register	FRESID3	H'0000	H'FFBF F13A	16	32-90
FlexRay Even Sync ID 4 Register	FRESID4	H'0000	H'FFBF F13E	16	32-90
FlexRay Even Sync ID 5 Register	FRESID5	H'0000	H'FFBF F142	16	32-90
FlexRay Even Sync ID 6 Register	FRESID6	H'0000	H'FFBF F146	16	32-90
FlexRay Even Sync ID 7 Register	FRESID7	H'0000	H'FFBF F14A	16	32-90
FlexRay Even Sync ID 8 Register	FRESID8	H'0000	H'FFBF F14E	16	32-90
FlexRay Even Sync ID 9 Register	FRESID9	H'0000	H'FFBF F152	16	32-90
FlexRay Even Sync ID 10 Register	FRESID10	H'0000	H'FFBF F156	16	32-90
FlexRay Even Sync ID 11 Register	FRESID11	H'0000	H'FFBF F15A	16	32-90
FlexRay Even Sync ID 12 Register	FRESID12	H'0000	H'FFBF F15E	16	32-90
FlexRay Even Sync ID 13 Register	FRESID13	H'0000	H'FFBF F162	16	32-90
FlexRay Even Sync ID 14 Register	FRESID14	H'0000	H'FFBF F166	16	32-90
FlexRay Even Sync ID 15 Register	FRESID15	H'0000	H'FFBF F16A	16	32-90
FlexRay Odd Sync ID 1 Register	FROSID1	H'0000	H'FFBF F172	16	32-91
FlexRay Odd Sync ID 2 Register	FROSID2	H'0000	H'FFBF F176	16	32-91
FlexRay Odd Sync ID 3 Register	FROSID3	H'0000	H'FFBF F17A	16	32-91
FlexRay Odd Sync ID 4 Register	FROSID4	H'0000	H'FFBF F17E	16	32-91
FlexRay Odd Sync ID 5 Register	FROSID5	H'0000	H'FFBF F182	16	32-91
FlexRay Odd Sync ID 6 Register	FROSID6	H'0000	H'FFBF F186	16	32-91
FlexRay Odd Sync ID 7 Register	FROSID7	H'0000	H'FFBF F18A	16	32-91
FlexRay Odd Sync ID 8 Register	FROSID8	H'0000	H'FFBF F18E	16	32-91
FlexRay Odd Sync ID 9 Register	FROSID9	H'0000	H'FFBF F192	16	32-91
FlexRay Odd Sync ID 10 Register	FROSID10	H'0000	H'FFBF F196	16	32-91

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
FlexRay Odd Sync ID 11 Register	FROSID11	H'0000	H'FFBF F19A	16	32-91
FlexRay Odd Sync ID 12 Register	FROSID12	H'0000	H'FFBF F19E	16	32-91
FlexRay Odd Sync ID 13 Register	FROSID13	H'0000	H'FFBF F1A2	16	32-91
FlexRay Odd Sync ID 14 Register	FROSID14	H'0000	H'FFBF F1A6	16	32-91
FlexRay Odd Sync ID 15 Register	FROSID15	H'0000	H'FFBF F1AA	16	32-91
FlexRay Network Management Vector 1 Register	FRNMV1	H'0000 0000	H'FFBF F1B0	32	32-92
FlexRay Network Management Vector 2 Register	FRNMV2	H'0000 0000	H'FFBF F1B4	32	32-92
FlexRay Network Management Vector 3 Register	FRNMV3	H'0000 0000	H'FFBF F1B8	32	32-92
FlexRay Message RAM Configuration Register	FRMRC	H'0180 0000	H'FFBF F300	32	32-94
FlexRay FIFO Rejection Filter Register	FRFRF	H'0180 0000	H'FFBF F304	32	32-97
FlexRay FIFO Rejection Filter Mask Register	FRFRFM	H'0000	H'FFBF F30A	16	32-98
FlexRay FIFO Critical Level Register	FRFCL	H'80	H'FFBF F30F	8	32-99
FlexRay Message Handler Status Register	FRMHDS	H'0000 0080	H'FFBF F310	32	32-100
FlexRay Last Dynamic Transmit Slot Register	FRLDTS	H'0000 0000	H'FFBF F314	32	32-102
FlexRay FIFO Status Register	FRFSR	H'0000	H'FFBF F31A	16	32-103
FlexRay Message Handler Constraints Flags Register	FRMHDF	H'0000	H'FFBF F31E	16	32-104
FlexRay Transmission Request Register 1	FRTXRQ1	H'0000 0000	H'FFBF F320	32	32-106
FlexRay Transmission Request Register 2	FRTXRQ2	H'0000 0000	H'FFBF F324	32	32-107
FlexRay Transmission Request Register 3	FRTXRQ3	H'0000 0000	H'FFBF F328	32	32-108
FlexRay Transmission Request Register 4	FRTXRQ4	H'0000 0000	H'FFBF F32C	32	32-109
FlexRay New Data Register 1	FRNDAT1	H'0000 0000	H'FFBF F330	32	32-110
FlexRay New Data Register 2	FRNDAT2	H'0000 0000	H'FFBF F334	32	32-111
FlexRay New Data Register 3	FRNDAT3	H'0000 0000	H'FFBF F338	32	32-112
FlexRay New Data Register 4	FRNDAT4	H'0000 0000	H'FFBF F33C	32	32-113
FlexRay Message Buffer Status Changed Register 1	FRMBSC1	H'0000 0000	H'FFBF F340	32	32-114
FlexRay Message Buffer Status Changed Register 2	FRMBSC2	H'0000 0000	H'FFBF F344	32	32-115
FlexRay Message Buffer Status Changed Register 3	FRMBSC3	H'0000 0000	H'FFBF F348	32	32-116
FlexRay Message Buffer Status Changed Register 4	FRMBSC4	H'0000 0000	H'FFBF F34C	32	32-117
FlexRay Write Data Section 1 Register	FRWRDS1	H'0000 0000	H'FFBF F400	32	32-119
FlexRay Write Data Section 2 Register	FRWRDS2	H'0000 0000	H'FFBF F404	32	32-119
FlexRay Write Data Section 3 Register	FRWRDS3	H'0000 0000	H'FFBF F408	32	32-119

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
FlexRay Write Data Section 4 Register	FRWRDS4	H'0000 0000	H'FFBF F40C	32	32-119
FlexRay Write Data Section 5 Register	FRWRDS5	H'0000 0000	H'FFBF F410	32	32-119
FlexRay Write Data Section 6 Register	FRWRDS6	H'0000 0000	H'FFBF F414	32	32-119
FlexRay Write Data Section 7 Register	FRWRDS7	H'0000 0000	H'FFBF F418	32	32-119
FlexRay Write Data Section 8 Register	FRWRDS8	H'0000 0000	H'FFBF F41C	32	32-119
FlexRay Write Data Section 9 Register	FRWRDS9	H'0000 0000	H'FFBF F420	32	32-119
FlexRay Write Data Section 10 Register	FRWRDS10	H'0000 0000	H'FFBF F424	32	32-119
FlexRay Write Data Section 11 Register	FRWRDS11	H'0000 0000	H'FFBF F428	32	32-119
FlexRay Write Data Section 12 Register	FRWRDS12	H'0000 0000	H'FFBF F42C	32	32-119
FlexRay Write Data Section 13 Register	FRWRDS13	H'0000 0000	H'FFBF F430	32	32-119
FlexRay Write Data Section 14 Register	FRWRDS14	H'0000 0000	H'FFBF F434	32	32-119
FlexRay Write Data Section 15 Register	FRWRDS15	H'0000 0000	H'FFBF F438	32	32-119
FlexRay Write Data Section 16 Register	FRWRDS16	H'0000 0000	H'FFBF F43C	32	32-119
FlexRay Write Data Section 17 Register	FRWRDS17	H'0000 0000	H'FFBF F440	32	32-119
FlexRay Write Data Section 18 Register	FRWRDS18	H'0000 0000	H'FFBF F444	32	32-119
FlexRay Write Data Section 19 Register	FRWRDS19	H'0000 0000	H'FFBF F448	32	32-119
FlexRay Write Data Section 20 Register	FRWRDS20	H'0000 0000	H'FFBF F44C	32	32-119
FlexRay Write Data Section 21 Register	FRWRDS21	H'0000 0000	H'FFBF F450	32	32-119
FlexRay Write Data Section 22 Register	FRWRDS22	H'0000 0000	H'FFBF F454	32	32-119
FlexRay Write Data Section 23 Register	FRWRDS23	H'0000 0000	H'FFBF F458	32	32-119
FlexRay Write Data Section 24 Register	FRWRDS24	H'0000 0000	H'FFBF F45C	32	32-119
FlexRay Write Data Section 25 Register	FRWRDS25	H'0000 0000	H'FFBF F460	32	32-119
FlexRay Write Data Section 26 Register	FRWRDS26	H'0000 0000	H'FFBF F464	32	32-119
FlexRay Write Data Section 27 Register	FRWRDS27	H'0000 0000	H'FFBF F468	32	32-119
FlexRay Write Data Section 28 Register	FRWRDS28	H'0000 0000	H'FFBF F46C	32	32-119
FlexRay Write Data Section 29 Register	FRWRDS29	H'0000 0000	H'FFBF F470	32	32-119
FlexRay Write Data Section 30 Register	FRWRDS30	H'0000 0000	H'FFBF F474	32	32-119
FlexRay Write Data Section 31 Register	FRWRDS31	H'0000 0000	H'FFBF F478	32	32-119
FlexRay Write Data Section 32 Register	FRWRDS32	H'0000 0000	H'FFBF F47C	32	32-119
FlexRay Write Data Section 33 Register	FRWRDS33	H'0000 0000	H'FFBF F480	32	32-119
FlexRay Write Data Section 34 Register	FRWRDS34	H'0000 0000	H'FFBF F484	32	32-119
FlexRay Write Data Section 35 Register	FRWRDS35	H'0000 0000	H'FFBF F488	32	32-119
FlexRay Write Data Section 36 Register	FRWRDS36	H'0000 0000	H'FFBF F48C	32	32-119
FlexRay Write Data Section 37 Register	FRWRDS37	H'0000 0000	H'FFBF F490	32	32-119
FlexRay Write Data Section 38 Register	FRWRDS38	H'0000 0000	H'FFBF F494	32	32-119
FlexRay Write Data Section 39 Register	FRWRDS39	H'0000 0000	H'FFBF F498	32	32-119
FlexRay Write Data Section 40 Register	FRWRDS40	H'0000 0000	H'FFBF F49C	32	32-119
FlexRay Write Data Section 41 Register	FRWRDS41	H'0000 0000	H'FFBF F4A0	32	32-119
FlexRay Write Data Section 42 Register	FRWRDS42	H'0000 0000	H'FFBF F4A4	32	32-119
FlexRay Write Data Section 43 Register	FRWRDS43	H'0000 0000	H'FFBF F4A8	32	32-119
FlexRay Write Data Section 44 Register	FRWRDS44	H'0000 0000	H'FFBF F4AC	32	32-119
FlexRay Write Data Section 45 Register	FRWRDS45	H'0000 0000	H'FFBF F4B0	32	32-119

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
FlexRay Write Data Section 46 Register	FRWRDS46	H'0000 0000	H'FFBF F4B4	32	32-119
FlexRay Write Data Section 47 Register	FRWRDS47	H'0000 0000	H'FFBF F4B8	32	32-119
FlexRay Write Data Section 48 Register	FRWRDS48	H'0000 0000	H'FFBF F4BC	32	32-119
FlexRay Write Data Section 49 Register	FRWRDS49	H'0000 0000	H'FFBF F4C0	32	32-119
FlexRay Write Data Section 50 Register	FRWRDS50	H'0000 0000	H'FFBF F4C4	32	32-119
FlexRay Write Data Section 51 Register	FRWRDS51	H'0000 0000	H'FFBF F4C8	32	32-119
FlexRay Write Data Section 52 Register	FRWRDS52	H'0000 0000	H'FFBF F4CC	32	32-119
FlexRay Write Data Section 53 Register	FRWRDS53	H'0000 0000	H'FFBF F4D0	32	32-119
FlexRay Write Data Section 54 Register	FRWRDS54	H'0000 0000	H'FFBF F4D4	32	32-119
FlexRay Write Data Section 55 Register	FRWRDS55	H'0000 0000	H'FFBF F4D8	32	32-119
FlexRay Write Data Section 56 Register	FRWRDS56	H'0000 0000	H'FFBF F4DC	32	32-119
FlexRay Write Data Section 57 Register	FRWRDS57	H'0000 0000	H'FFBF F4E0	32	32-119
FlexRay Write Data Section 58 Register	FRWRDS58	H'0000 0000	H'FFBF F4E4	32	32-119
FlexRay Write Data Section 59 Register	FRWRDS59	H'0000 0000	H'FFBF F4E8	32	32-119
FlexRay Write Data Section 60 Register	FRWRDS60	H'0000 0000	H'FFBF F4EC	32	32-119
FlexRay Write Data Section 61 Register	FRWRDS61	H'0000 0000	H'FFBF F4F0	32	32-119
FlexRay Write Data Section 62 Register	FRWRDS62	H'0000 0000	H'FFBF F4F4	32	32-119
FlexRay Write Data Section 63 Register	FRWRDS63	H'0000 0000	H'FFBF F4F8	32	32-119
FlexRay Write Data Section 64 Register	FRWRDS64	H'0000 0000	H'FFBF F4FC	32	32-119
FlexRay Write Header Section Register 1	FRWRHS1	H'0000 0000	H'FFBF F500	32	32-120
FlexRay Write Header Section Register 2	FRWRHS2	H'0000 0000	H'FFBF F504	32	32-122
FlexRay Write Header Section Register 3	FRWRHS3	H'0000	H'FFBF F50A	16	32-123
FlexRay Input Buffer Command Mask Register	FRIBCM	H'0000 0000	H'FFBF F510	32	32-124
FlexRay Input Buffer Command Request Register	FRIBCR	H'0000 0000	H'FFBF F514	32	32-125
FlexRay Read Data Section Register 1	FRRDDS1	H'0000 0000	H'FFBF F600	32	32-127
FlexRay Read Data Section Register 2	FRRDDS2	H'0000 0000	H'FFBF F604	32	32-127
FlexRay Read Data Section Register 3	FRRDDS3	H'0000 0000	H'FFBF F608	32	32-127
FlexRay Read Data Section Register 4	FRRDDS4	H'0000 0000	H'FFBF F60C	32	32-127
FlexRay Read Data Section Register 5	FRRDDS5	H'0000 0000	H'FFBF F610	32	32-127
FlexRay Read Data Section Register 6	FRRDDS6	H'0000 0000	H'FFBF F614	32	32-127
FlexRay Read Data Section Register 7	FRRDDS7	H'0000 0000	H'FFBF F618	32	32-127
FlexRay Read Data Section Register 8	FRRDDS8	H'0000 0000	H'FFBF F61C	32	32-127
FlexRay Read Data Section Register 9	FRRDDS9	H'0000 0000	H'FFBF F620	32	32-127
FlexRay Read Data Section Register 10	FRRDDS10	H'0000 0000	H'FFBF F624	32	32-127
FlexRay Read Data Section Register 11	FRRDDS11	H'0000 0000	H'FFBF F628	32	32-127
FlexRay Read Data Section Register 12	FRRDDS12	H'0000 0000	H'FFBF F62C	32	32-127
FlexRay Read Data Section Register 13	FRRDDS13	H'0000 0000	H'FFBF F630	32	32-127
FlexRay Read Data Section Register 14	FRRDDS14	H'0000 0000	H'FFBF F634	32	32-127

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
FlexRay Read Data Section Register 15	FRRDDS15	H'0000 0000	H'FFBF F638	32	32-127
FlexRay Read Data Section Register 16	FRRDDS16	H'0000 0000	H'FFBF F63C	32	32-127
FlexRay Read Data Section Register 17	FRRDDS17	H'0000 0000	H'FFBF F640	32	32-127
FlexRay Read Data Section Register 18	FRRDDS18	H'0000 0000	H'FFBF F644	32	32-127
FlexRay Read Data Section Register 19	FRRDDS19	H'0000 0000	H'FFBF F648	32	32-127
FlexRay Read Data Section Register 20	FRRDDS20	H'0000 0000	H'FFBF F64C	32	32-127
FlexRay Read Data Section Register 21	FRRDDS21	H'0000 0000	H'FFBF F650	32	32-127
FlexRay Read Data Section Register 22	FRRDDS22	H'0000 0000	H'FFBF F654	32	32-127
FlexRay Read Data Section Register 23	FRRDDS23	H'0000 0000	H'FFBF F658	32	32-127
FlexRay Read Data Section Register 24	FRRDDS24	H'0000 0000	H'FFBF F65C	32	32-127
FlexRay Read Data Section Register 25	FRRDDS25	H'0000 0000	H'FFBF F660	32	32-127
FlexRay Read Data Section Register 26	FRRDDS26	H'0000 0000	H'FFBF F664	32	32-127
FlexRay Read Data Section Register 27	FRRDDS27	H'0000 0000	H'FFBF F668	32	32-127
FlexRay Read Data Section Register 28	FRRDDS28	H'0000 0000	H'FFBF F66C	32	32-127
FlexRay Read Data Section Register 29	FRRDDS29	H'0000 0000	H'FFBF F670	32	32-127
FlexRay Read Data Section Register 30	FRRDDS30	H'0000 0000	H'FFBF F674	32	32-127
FlexRay Read Data Section Register 31	FRRDDS31	H'0000 0000	H'FFBF F678	32	32-127
FlexRay Read Data Section Register 32	FRRDDS32	H'0000 0000	H'FFBF F67C	32	32-127
FlexRay Read Data Section Register 33	FRRDDS33	H'0000 0000	H'FFBF F680	32	32-127
FlexRay Read Data Section Register 34	FRRDDS34	H'0000 0000	H'FFBF F684	32	32-127
FlexRay Read Data Section Register 35	FRRDDS35	H'0000 0000	H'FFBF F688	32	32-127
FlexRay Read Data Section Register 36	FRRDDS36	H'0000 0000	H'FFBF F68C	32	32-127
FlexRay Read Data Section Register 37	FRRDDS37	H'0000 0000	H'FFBF F690	32	32-127
FlexRay Read Data Section Register 38	FRRDDS38	H'0000 0000	H'FFBF F694	32	32-127
FlexRay Read Data Section Register 39	FRRDDS39	H'0000 0000	H'FFBF F698	32	32-127
FlexRay Read Data Section Register 40	FRRDDS40	H'0000 0000	H'FFBF F69C	32	32-127
FlexRay Read Data Section Register 41	FRRDDS41	H'0000 0000	H'FFBF F6A0	32	32-127
FlexRay Read Data Section Register 42	FRRDDS42	H'0000 0000	H'FFBF F6A4	32	32-127
FlexRay Read Data Section Register 43	FRRDDS43	H'0000 0000	H'FFBF F6A8	32	32-127
FlexRay Read Data Section Register 44	FRRDDS44	H'0000 0000	H'FFBF F6AC	32	32-127
FlexRay Read Data Section Register 45	FRRDDS45	H'0000 0000	H'FFBF F6B0	32	32-127
FlexRay Read Data Section Register 46	FRRDDS46	H'0000 0000	H'FFBF F6B4	32	32-127
FlexRay Read Data Section Register 47	FRRDDS47	H'0000 0000	H'FFBF F6B8	32	32-127
FlexRay Read Data Section Register 48	FRRDDS48	H'0000 0000	H'FFBF F6BC	32	32-127
FlexRay Read Data Section Register 49	FRRDDS49	H'0000 0000	H'FFBF F6C0	32	32-127
FlexRay Read Data Section Register 50	FRRDDS50	H'0000 0000	H'FFBF F6C4	32	32-127
FlexRay Read Data Section Register 51	FRRDDS51	H'0000 0000	H'FFBF F6C8	32	32-127
FlexRay Read Data Section Register 52	FRRDDS52	H'0000 0000	H'FFBF F6CC	32	32-127
FlexRay Read Data Section Register 53	FRRDDS53	H'0000 0000	H'FFBF F6D0	32	32-127
FlexRay Read Data Section Register 54	FRRDDS54	H'0000 0000	H'FFBF F6D4	32	32-127
FlexRay Read Data Section Register 55	FRRDDS55	H'0000 0000	H'FFBF F6D8	32	32-127
FlexRay Read Data Section Register 56	FRRDDS56	H'0000 0000	H'FFBF F6DC	32	32-127

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
FlexRay Read Data Section Register 57	FRRDDS57	H'0000 0000	H'FFBF F6E0	32	32-127
FlexRay Read Data Section Register 58	FRRDDS58	H'0000 0000	H'FFBF F6E4	32	32-127
FlexRay Read Data Section Register 59	FRRDDS59	H'0000 0000	H'FFBF F6E8	32	32-127
FlexRay Read Data Section Register 60	FRRDDS60	H'0000 0000	H'FFBF F6EC	32	32-127
FlexRay Read Data Section Register 61	FRRDDS61	H'0000 0000	H'FFBF F6F0	32	32-127
FlexRay Read Data Section Register 62	FRRDDS62	H'0000 0000	H'FFBF F6F4	32	32-127
FlexRay Read Data Section Register 63	FRRDDS63	H'0000 0000	H'FFBF F6F8	32	32-127
FlexRay Read Data Section Register 64	FRRDDS64	H'0000 0000	H'FFBF F6FC	32	32-127
FlexRay Read Header Section Register 1	FRRDHS1	H'0000 0000	H'FFBF F700	32	32-128
FlexRay Read Header Section Register 2	FRRDHS2	H'0000 0000	H'FFBF F704	32	32-129
FlexRay Read Header Section Register 3	FRRDHS3	H'0000 0000	H'FFBF F708	32	32-130
FlexRay Message Buffer Status Register	FRMBS	H'0000 0000	H'FFBF F70C	32	32-131
FlexRay Output Buffer Command Mask Register	FROBCM	H'0000 0000	H'FFBF F710	32	32-134
FlexRay Output Buffer Command Request Register	FROBCR	H'0000 0000	H'FFBF F714	32	32-135

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

32.3 Terms and Abbreviations

Terms and abbreviations used in this chapter are listed in Table 32.4.

Table 32.4 Terms and Abbreviations

Term	Meaning
CAS	Collision Avoidance Symbol
CC	Communication Controller
CHI	Controller Host Interface
FSM	Finite State Machine
FTM	Fault Tolerant Midpoint
GTU	Global Time Unit Block
IBF	Input Buffer
MHD	Message Handler Block
MT	Macrotick
µT	Microtick
MTS	Media Access Test Symbol
NCT	Network Communication Time
NEM	Network Management Block
NIT	Network Idle Time
NM	Network Management
OBF	Output Buffer
POC	Protocol Operation Control
PRT	Protocol Controller Block
SUC	System Universal Control Block
TBF	Transient Buffer
TDMA	Time Division Multiple Access
TT-D	Time Triggered Distributed Synchronization
WUP	Wakeup Pattern
WUS	Wakeup Symbol

32.4 Special Registers

32.4.1 FlexRay Operation Control Register (FXROC)

FlexRay Operation Control Register (FXROC) <P4 address: location H'FFBF F004>

Bit:	7	6	5	4	3	2	1	0
	FOPC	—	—	—	—	FBSEN	FRSTAT	FOPEN
After Reset:	0	0	0	0	0	1	0	0

<After Reset: H'04>

Bit	Abbreviation	After Reset	R	W	Description
7	FOPC	0	R	W	FlexRay Operation Control Protection Bit This bits protects against unintended write access to the FOPEN bit in the FXROC register. 0: Unprotected Write access to the FOPEN bit in the FXROC register is enabled. 1: Protected Write access to the FOPEN bit in the FXROC register is disabled.
6 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
2	FBSEN	1	R	W	FlexRay Byte Swap Enable Bit

This bit controls the byte order on reading and writing the FlexRay Network Management Vector register (FRNMVn), FlexRay Write Data Section (FRWRDSn) and FlexRay Read Data Section (FRRDDSn).

Reference in FRNMV, FRWRDSn and FRRDDSn section to this section.

0: Disabled

Byte alignment in FRNMVn, FRWRDSn and FRRDDSn is in little endian style.

- FRNVMn

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NMV1	Data3								Data2								Data1								Data0							
NMV2	Data7								Data6								Data5								Data4							
NMV3	Data11								Data10								Data9								Data8							

- FRWRDSn
  FRWRDSn.MD[7:0]=DWn, byten-1
  FRWRDSn.MD[15:8]=DWn, byten
  FRWRDSn.MD[23:16]=DWn+1, byten+1
  FRWRDSn.MD[31:24]=DWn+1, byten+2
- FRRDDSn
  FRRDDSn.MD[7:0]=DWn, byten-1
  FRRDDSn.MD[15:8]=DWn, byten
  FRRDDSn.MD[23:16]=DWn+1, byten+1
  FRRDDSn.MD[31:24]=DWn+1, byten+2

1: Enabled

Byte alignment in FRNMVn, FRWRDSn and FRRDDSn is in big endian style.

- FRNVMn

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NMV1	Data0								Data1								Data2								Data3							
NMV2	Data4								Data5								Data6								Data7							
NMV3	Data8								Data9								Data10								Data11							

- FRWRDSn
  FRWRDSn.MD[7:0]=DWn+1, byten+2
  FRWRDSn.MD[15:8]=DWn+1, byten+1
  FRWRDSn.MD[23:16]=DWn, byten
  FRWRDSn.MD[31:24]=DWn, byten-1
- FRRDDSn
  FRRDDSn.MD[7:0]=DWn, byten-1
  FRRDDSn.MD[15:8]=DWn, byten
  FRRDDSn.MD[23:16]=DWn+1, byten+1
  FRRDDSn.MD[31:24]=DWn+1, byten+2

Note: • In the case of FRNMVn, n is in the range 0 to 11.

Bit	Abbreviation	After Reset	R	W	Description
1	FRSTAT	0	R	—	FlexRay Reset Status Bit This bit indicates that the FlexRay module is in the reset state or operation state. 0: The FlexRay module is in the reset state. In this state, it is not possible to access to registers mapped to the FlexRay address area (H'FFBF F010 to H'FFBF FFFF) and the FlexRay module gets reset. When the FOPEN bit in the FXROC register is set to 'Operation disabled', it takes up to 24 cycles of the peripheral A clock (PAck) until the FRSTAT bit goes to 'Reset state'. 1: The FlexRay module is in the operating state In this state, it is possible to access to registers mapped to the FlexRay address area (H'FFBF F010 to H'FFBF FFFF). When the FOPEN bit in the FXROC register is set to 'Operation enabled' it takes up to 24 cycles of the peripheral A clock (PAck) until the FRSTAT bit goes to 'Operating state'.
0	FOPEN	0	R	W	FlexRay Enable Bit This bit controls the operation/ reset of the FlexRay module. Write access to this bit is only possible if the FOPC bit in the FXROC register was set to 'Unprotected' in a previous write access to this register bit. Any write access to this register bit is ignored as long as the FOPC bit in the FXROC register is set to 'Protected'. 0: Operation disabled When the FOPEN bit is set to 'Operation disabled', the FlexRay module is forcibly moved to the reset state, whatever the state of the FlexRay module is. In 'Operation disabled' state, all registers mapped to the FlexRay address area (H'FFBF F010 to H'FFBF FFFF) are initialised. It takes up to the maximum of 24 cycles of the peripheral A clock (PAck) from a change of the setting (from 'Operation enabled' to 'Operation disabled') to actual transition of the FlexRay Module to Reset state. The FlexRay Reset Status Bit (FRSTAT) indicates whether the FlexRay Module is in Reset state or not. If the setting is changed from 'Operation enabled' to 'Operation disabled', it is prohibited to change it again to operation enabled, before the FRSTAT bit in the FRXROC register indicates 'Reset state'. In operation disabled state, do not access registers mapped to the FlexRay address area (H'FFBF F010 to H'FFBF FFFF). 1: Operation enabled When the FOPEN bit is set to 'Operation enabled', the reset state of the FlexRay Module is released; access to registers mapped to the FlexRay address area (H'FFBF F010 to H'FFBF FFFF) becomes possible. In order to perform FlexRay communication, this bit must be set to 'Operation enabled'. It takes up to the maximum of 24 cycles of the peripheral A clock (PAck) from a change of the setting (from operation disabled to enabled) to actual release of the FlexRay Module's reset state. If the setting is changed from 'Operation disabled' to 'Operation enabled', it is prohibited to change it again to 'Operation disabled', before the FRSTAT bit in the FXROC register indicates 'Operating state'.

32.4.2 FlexRay Lock Register (FRLCK)

The FRLCK register unlocks the protection from an unexpected write access to the FRSUCC1 register by which the CC would transit to READY state from CONFIG state.

FlexRay Lock Register (FRLCK) <P4 address: location H'FFBF F01F>

Bit:	7	6	5	4	3	2	1	0
	CLK7	CLK6	CLK5	CLK4	CLK3	CLK2	CLK1	CLK0
After Reset:	0	0	0	0	0	0	0	0

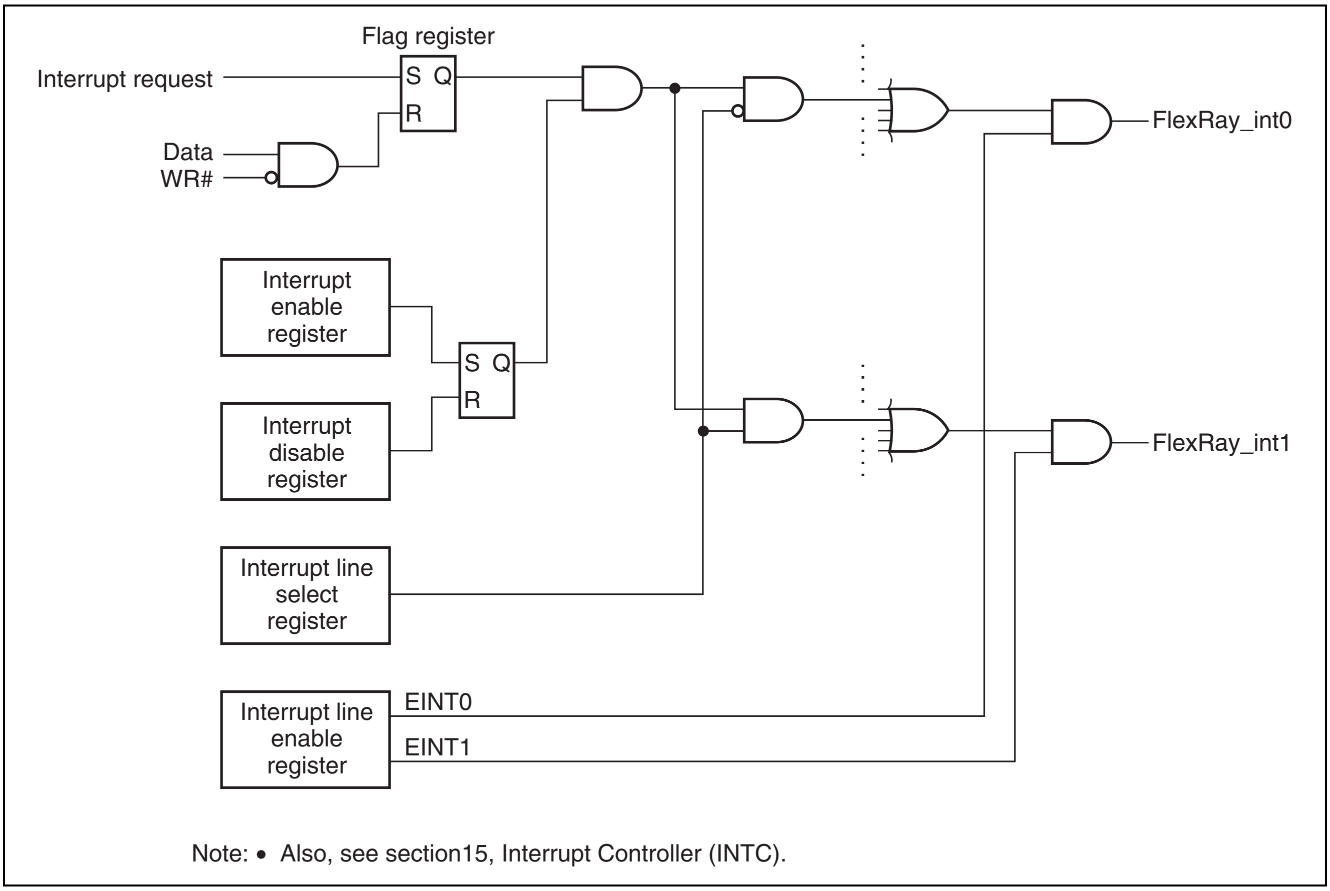
<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	CLK7 to CLK0	All 0	0	W	Configuration Lock Key Bit To leave CONFIG state by writing bits CMD3 to CMD0 in the FRSUCC1 register (commands READY), the write operation has to be directly preceded by two write accesses to the Configuration Lock Key (unlock sequence). If the write sequence below is interrupted by other write accesses between the second write to the Configuration Lock Key and the write access to the FRSUCC1 register, the CC remains in CONFIG state and the sequence has to be repeated. First write: bits CLK7 to CLK0 in the FRLCK register = H'CE Second write: bits CLK7 to CLK0 in the FRLCK register = H'31 Third write: bits CMD3 to CMD0 in the FRSUCC1 Setting Value: "H'00" to "H'FF"

32.5 Interrupt Registers

Interrupt registers control interrupt requests from the FlexRay module.

These interrupt requests are generated by two sources: errors and status change. According to the setting of interrupt enable/disable registers and interrupt line select registers, they are merged into FlexRay_int0 or FlexRay_int1.


Flag register
Interrupt request
S Q
R
Data
WR#
Interrupt enable register
Interrupt disable register
S Q
R
Interrupt line select register
Interrupt line enable register
EINT0
EINT1
FlexRay_int0
FlexRay_int1

Note: • Also, see section15, Interrupt Controller (INTC).

Figure 32.2 Block Diagram of FlexRay Interrupt Requests 0/1

In addition to the FlexRay_int0, FlexRay_int1 above, the FlexRay module contains FlexRay_tint0, FlexRay_tint1, which is a part of the FlexRay_int0, FlexRay_int1.

32.5.1 FlexRay Error Interrupt Register (FREIR)

The flags are set to "1" when the CC detects one of the listed error conditions. The flags remain set to "1" until the program sets them to "0". A flag is set to "0" by writing a "1" to the corresponding bit position. Writing a "0" has no effect on the flag.

FlexRay Error Interrupt Register (FREIR) <P4 address: location H'FFBF F020>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABB	LTVB	EDB	—	—	—	—	—	TABA	LTVA	EDA
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHF	IOBA	IIBA	EFA	RFO	PERR	CCL	CCF	SFO	SFBM	CNA	PEMC
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
32 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26	TABB	0	R	W	Transmission Across Boundary Channel B Flag The flag signals to the CPU that a transmission across a slot boundary occurred for channel B. 0: No transmission across slot boundary detected on channel B 1: Transmission across slot boundary detected on channel B
25	LTVB	0	R	W	Latest Transmit Violation Channel B Flag The flag signals a latest transmit violation on channel B to the CPU. 0: No latest transmit violation detected on channel B 1: Latest transmit violation detected on channel B
24	EDB	0	R	W	Error Detected on Channel B Flag This flag signals an error detected on channel B. This bit is set to "1" whenever one of the flags in the FRACS register, SEDB, CEDB, CIB, or SBVB changes from "0" to "1". 0: No error detected on channel B 1: Error detected on channel B
23 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18	TABA	0	R	W	Transmission Across Boundary Channel A Flag The flag signals to the CPU that a transmission across a slot boundary occurred for channel A. 0: No transmission across slot boundary detected on channel A 1: Transmission across slot boundary detected on channel A
17	LTVA	0	R	W	Latest Transmit Violation Channel A Flag The flag signals a latest transmit violation on channel A to the CPU. 0: No latest transmit violation detected on channel A 1: Latest transmit violation detected on channel A

Bit	Abbreviation	After Reset	R	W	Description
16	EDA	0	R	W	Error Detected on Channel A Flag This flag signals an error detected on channel A. This bit is set to "1" whenever one of the flags in the FRACS register, SEDA, CEDA, CIA, or SBVA changes from "0" to "1". 0: No error detected on channel A 1: Error detected on channel A
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	MHF	0	R	W	Message Handler Constraints Flag The flag is set to "1" whenever one of the flags in the FRMHDF register, SNUA, SNUB, FNFA, FNFB, TBFA, TBFB, or WAHP changes from "0" to "1". 0: No Message Handler failure detected 1: Message Handler failure detected
10	IOBA	0	R	W	Illegal Output buffer Access Flag This flag signals an illegal access to Output Buffer. This flag is set to "1" by the CC when the CPU requests the transfer of a message buffer from the Message RAM to the Output Buffer while the OBSYS bit in the FROBCR register is set to "1". 0: No illegal CPU access to Output Buffer occurred 1: Illegal CPU access to Output Buffer occurred
9	IIBA	0	R	W	Illegal Input Buffer Access Flag This flag is set to "1" by the CC when the CPU wants to modify a message buffer via Input Buffer and one of the following conditions applies: (1) The CC is not in CONFIG or DEFAULT_CONFIG state and the CPU writes to FRIBCR register to modify the • Header section of message buffer 0, 1 if configured for transmission in key slot • Header section of static message buffers with buffer number < bits FDB7 to FDB0 in the FRMRC register while bits SEC1 to SEC0 in the FRMRC register = "01" • Header section of any static or dynamic message buffer while bits SEC1 to SEC0 in the FRMRC register = "10" or "11" • Header and / or data section of any message buffer belonging to the receive FIFO (2) The CPU writes to any register of the Input Buffer while the IBSYH bit in the FRIBCR register is set to "1". 0: No illegal CPU access to Input Buffer occurred 1: Illegal CPU access to Input Buffer occurred
8	EFA	0	R	W	Empty FIFO Access Flag This flag is set to "1" by the CC when the CPU requests the transfer of a message from the receive FIFO via Output Buffer while the receive FIFO is empty. 0: No CPU access to empty FIFO occurred 1: CPU access to empty FIFO occurred

Bit	Abbreviation	After Reset	R	W	Description
7	RFO	0	R	W	Receive FIFO Overrun Flag The flag is set to "1" by the CC when a receive FIFO overrun is detected. When a receive FIFO overrun occurs, the oldest message is overwritten with the actual received message. The actual state of the FIFO is monitored in the FRFSR register. 0: No receive FIFO overrun detected 1: A receive FIFO overrun has been detected
6	PERR	0	R	W	Parity Error Flag The flag signals a parity error to the CPU. It is set to "1" whenever one of the flags in the FRMHDS register, PIBF, POBF, PMR, PTBF1, or PTBF2 changes from "0" to "1". 0: No parity error detected 1: Parity error detected
5	CCL	0	R	W	CHI Command Locked Flag The flag signals that the write access to the CHI command vector (bits CMD3 to CMD0 in the FRSUCC1 register) was not successful because the execution of the previous CHI command has not yet completed. In this case the CNA bit is also set to "1". 0: CHI command accepted 1: CHI command not accepted*[5]
4	CCF	0	R	W	Clock Correction Failure Flag This flag is set to "1" at the end of the cycle whenever one of the following errors occurred: • Missing offset and / or rate correction • Clock correction limit reached The clock correction status is monitored in registers FRCCEV and FRSFS. 0: No clock correction error 1: Clock correction failed*[4]
3	SFO	0	R	W	Sync Frame Overflow Flag Set to "1" when either the number of sync frames received during the last communication cycle or the total number of different sync frame IDs received during the last double cycle exceeds the maximum number of sync frames as defined by bits SNM3 to SNM0 in the FRGTUC2 register. 0: Number of received sync frames ≤ bits SNM3 to SNM0 in the FRGTUC2 register 1: More sync frames received than configured by bits SNM3 to SNM0 in the FRGTUC2 register
2	SFBM	0	R	W	Sync Frames Below Minimum Flag This flag signals that the number of sync frames received during the last communication cycle was below the limit required by the FlexRay protocol. May be set to "1" during startup and therefore should be set to "0" by the program after the CC entered NORMAL_ACTIVE state. 0: Sync node: 1 or more sync frames received Non-sync node: 2 or more sync frames received 1: Less than the required minimum of sync frames received*[3]*[4]

Bit	Abbreviation	After Reset	R	W	Description
1	CNA	0	R	W	Command Not Accepted Flag The flag signals that the write access to the CHI command vector (bits CMD3 to CMD0 in the FRSUCC1 register) was not successful because the requested command was not valid in the actual POC state, or because the CHI command was locked (the CCL bit = 1). 0: CHI command accepted 1: CHI command not accepted*[2]
0	PEMC	0	R	W	POC Error Mode Changed Flag This flag is set to "1" whenever the error mode signalled by bits ERRM1 to ERRM0 in the FRCCEV register has changed. 0: Error mode has not changed 1: Error mode has changed*[1]

Notes: *1 This flag is set to "1" whenever the error mode signalled by bits ERRM1 to ERRM0 in the FRCCEV register has changed.

*2 The CCL bit is also set to "1" when the CHI command is executed during state transitions.

*3 This flag signals that the number of sync frames received during the last communication cycle was below the limit required by the FlexRay protocol. Sync node: 1 or more sync frames received, non-sync node: 2 or more sync frames received

*4 May be set to "1" during startup and therefore should be set to "0" by the program after the CC entered NORMAL_ACTIVE state.

*5 The CNA flag is also set to "1" when this flag is set to "1".

32.5.2 FlexRay Status Interrupt Register (FRSIR)

The flags are set to "1" when the CC detects one of the listed events. The flags remain set to "1" until the program sets them to "0". A flag is set to "0" by writing a "1" to the corresponding bit position. Writing a "0" has no effect on the flag.

FlexRay Status Interrupt Register (FRSIR) <P4 address: location H'FFBF F024>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSB	WUPB	—	—	—	—	—	—	MTSA	WUPA
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS	MBSI	SUCS	SWE	TOBC	TIBC	TI1	TI0	NMVC	RFCL	RFNE	RXI	TXI	CYCS	CAS	WST
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 26	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
25	MTSB	0	R	W	MTS Received on Channel B Flag (vSS!ValidMTSB) Media Access Test symbol received on channel B during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window. 0: No MTS received on channel B 1: MTS received on channel B
24	WUPB	0	R	W	Wakeup Pattern Channel B Flag This flag is set to "1" by the CC when a wakeup pattern was received on channel B. Only set to "1" when the CC is in WAKEUP, READY, or STARTUP state. 0: No wakeup pattern received on channel B 1: Wakeup pattern received on channel B*[1]
23 to 18	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
17	MTSA	0	R	W	MTS Received on Channel A Flag (vSS!ValidMTSA) Media Access Test symbol received on channel A during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window. 0: No MTS received on channel A 1: MTS received on channel A
16	WUPA	0	R	W	Wakeup Pattern Channel A Flag This flag is set to "1" by the CC when a wakeup pattern was received on channel A. Only set to "1" when the CC is in WAKEUP, READY, or STARTUP state. 0: No wakeup pattern received on channel A 1: Wakeup pattern received on channel A*[1]
15	SDS	0	R	W	Start of Dynamic Segment Flag 0: Dynamic segment not yet started 1: Dynamic segment started

Bit	Abbreviation	After Reset	R	W	Description
14	MBSI	0	R	W	Message Buffer Status Interrupt Flag This flag is set to "1" by the CC when the message buffer status (MBS) has changed and if the MBI bit of that message buffer is set to "1". 0: No message buffer status change of message buffer with MBI = 1 1: Message buffer status of at least one message buffer with MBI = 1 has changed
13	SUCS	0	R	W	Startup Completed Successfully Flag This flag is set to "1" whenever a startup completed successfully and the CC entered NORMAL_ACTIVE state. 0: No startup completed successfully 1: Startup completed successfully
12	SWE	0	R	W	Stop Watch Event Flag This flag is set to "1" after a stop watch activation when the actual cycle counter and macrotick value are stored in FRSTPW1 register. 0: No Stop Watch Event 1: Stop Watch Event occurred
11	TOBC	0	R	W	Transfer Output Buffer Completed Flag This flag is set to "1" whenever a transfer from Message RAM to the Output Buffer has completed and the OBSYS bit in the FROBCR register has been set to "0" by the Message Handler. 0: No transfer completed 1: Transfer between Message RAM and Output Buffer completed
10	TIBC	0	R	W	Transfer Input Buffer Completed Flag This flag is set to "1" whenever a transfer from Input Buffer to the Message RAM has completed and the IBSYS bit in the FRIBCR register has been set to "0" by the Message Handler. 0: No transfer completed 1: Transfer between Input Buffer and Message RAM completed
9	TI1	0	R	W	Timer Interrupt 1 Flag This flag is set to "1" whenever timer 1 matches the conditions configured in the FRT1C register. A Timer Interrupt 1 is also signalled on the FlexRay_tint1 line. 0: No Timer Interrupt 1 1: Timer Interrupt 1 occurred
8	TI0	0	R	W	Timer Interrupt 0 Flag This flag is set to "1" whenever timer 0 matches the conditions configured in the FRT0C register. A Timer Interrupt 0 is also signalled on the FlexRay_tint0 line. 0: No Timer Interrupt 0 1: Timer Interrupt 0 occurred
7	NMVC	0	R	W	Network Management Vector Changed Flag This interrupt flag signals a change in the Network Management Vector visible to the CPU. 0: No change in the network management vector 1: Network management vector changed

Bit	Abbreviation	After Reset	R	W	Description
6	RFCL	0	R	W	Receive FIFO Critical Level Flag This flag is set to "1" when the receive FIFO fill level (bits RFFL7 to RFFL0 in the FRFSR register) is equal or greater than the critical level as configured by bits CL7 to CL0 in the FRFCL register. 0: Receive FIFO below critical level 1: Receive FIFO critical level reached
5	RFNE	0	R	W	Receive FIFO Not Empty Flag This flag is set to "1" by the CC when a received valid frame was stored into the empty receive FIFO. The actual state of the receive FIFO is monitored in the empty FRFSR register. 0: Receive FIFO is empty 1: Receive FIFO is not empty
4	RXI	0	R	W	Transmit Interrupt Flag This flag is set to "1" by the CC whenever the set condition of a message buffers ND flag is fulfilled, and if the MBI bit of that message buffer is set to "1". 0: No ND flag of a receive buffer with MBI = 1 has been set to "1" 1: At least one ND flag of a receive buffer with MBI = ‘1’ has been set to "1"
3	TXI	0	R	W	Transmit Interrupt Flag This flag is set to "1" by the CC at the end of frame transmission if the MBI bit in the respective message buffer is set to "1". 0: No frame transmitted from a transmit buffer with MBI = 1 1: At least one frame was transmitted from a transmit buffer with MBI = 1
2	CYCS	0	R	W	Cycle Start Interrupt Flag This flag is set to "1" by the CC when a communication cycle starts. 0: No communication cycle started 1: Communication cycle started
1	CAS	0	R	W	Collision Avoidance Symbol Flag This flag is set to "1" by the CC during STARTUP state when a CAS or a potential CAS was received. 0: No bit pattern matching the CAS symbol received 1: Bit pattern matching the CAS symbol received
0	WST	0	R	W	Wakeup Status Flag This flag is set to "1" when bits WSV2 to WSV0 in the FRCCSV register is changed. 0: Wakeup status unchanged 1: Wakeup status changed

Note: *1 Only set to "1" when the CC is in WAKEUP, READY, or STARTUP state.

32.5.3 FlexRay Error Interrupt Line Select Register (FREILS)

The Error Interrupt Line Select register assigns an interrupt generated by a specific error interrupt flag from the FREIR register to one of the two module interrupt lines: FlexRay_int1 and FlexRay_int0.

FlexRay Error Interrupt Line Select Register (FREILS) <P4 address: location H'FFBF F028>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABBL	LTVBL	EDBL	—	—	—	—	—	TABAL	LTVAL	EDAL
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHFL	IOBAL	IIBAL	EFAL	RFOL	PERRL	CCLL	CCFL	SFOL	SFBML	CNAL	PEMCL
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26	TABBL	0	R	W	Transmission Across Boundary Channel B Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
25	LTVBL	0	R	W	Latest Transmit Violation Channel B Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
24	EDBL	0	R	W	Error Detected on Channel B Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
23 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18	TABAL	0	R	W	Transmission Across Boundary Channel A Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
17	LTVAL	0	R	W	Latest Transmit Violation Channel A Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
16	EDAL	0	R	W	Error Detected on Channel A Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	MHFL	0	R	W	Message Handler Constraints Flag Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
10	IOBAL	0	R	W	Illegal Output Buffer Access Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1

Bit	Abbreviation	After Reset	R	W	Description
9	IIBAL	0	R	W	Illegal Input Buffer Access Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
8	EFAL	0	R	W	Empty FIFO Access Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
7	RFOL	0	R	W	Receive FIFO Overrun Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
6	PERRL	0	R	W	Parity Error Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
5	CCLL	0	R	W	CHI Command Locked Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
4	CCFL	0	R	W	Clock Correction Failure Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
3	SFOL	0	R	W	Sync Frame Overflow Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
2	SFBML	0	R	W	Sync Frames Below Minimum Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
1	CNAL	0	R	W	Command Not Accepted Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
0	PEMCL	0	R	W	POC Error Mode Changed PEMCL Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1

32.5.4 FlexRay Status Interrupt Line Select Register (FRSILS)

The Status Interrupt Line Select register assigns an interrupt generated by a specific status interrupt flag from the FRSIR register to one of the two module interrupt lines: FlexRay_int1 and FlexRay_int0.

FlexRay Status Interrupt Line Select Register (FRSILS) <P4 address: location H'FFBF F02C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSBL	WUPBL	—	—	—	—	—	—	MTSAL	WUPAL
After Reset:	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDSL	MBSIL	SUCSL	SWEL	TOBCL	TIBCL	TI1L	TI0L	NMVCL	RFCLL	RFNEL	RXIL	TXIL	CYCSL	CASL	WSTL
After Reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

<After Reset: H'0303 FFFF>

Bit	Abbreviation	After Reset	R	W	Description
31 to 26	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
25	MTSBL	1	R	W	Media Access Test Symbol Channel B Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
24	WUPBL	1	R	W	Wakeup Pattern Channel B Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
23 to 18	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
17	MTSAL	1	R	W	Media Access Test Symbol Channel A Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
16	WUPAL	1	R	W	Wakeup Pattern Channel A Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
15	SDSL	1	R	W	Start of Dynamic Segment Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
14	MBSIL	1	R	W	Message Buffer Status Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
13	SUCSL	1	R	W	Startup Completed Successfully Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
12	SWEL	1	R	W	Stop Watch Event Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
11	TOBCL	1	R	W	Transfer Output Buffer Completed Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1

Bit	Abbreviation	After Reset	R	W	Description
10	TIBCL	1	R	W	Transfer Input Buffer Completed Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
9	TI1L	1	R	W	Timer Interrupt 1 Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
8	TI0L	1	R	W	Timer Interrupt 0 Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
7	NMVCL	1	R	W	Network Management Vector Changed Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
6	RFCLL	1	R	W	Receive FIFO Critical Level Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
5	RFNEL	1	R	W	Receive FIFO Not Empty Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
4	RXIL	1	R	W	Receive Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
3	TXIL	1	R	W	Transmit Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
2	CYCSL	1	R	W	Cycle Start Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
1	CASL	1	R	W	Collision Avoidance Symbol Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1
0	WSTL	1	R	W	Wakeup Status Interrupt Line Bit 0: Interrupt assigned to interrupt line FlexRay_int0 1: Interrupt assigned to interrupt line FlexRay_int1

32.5.5 FlexRay Error Interrupt Enable Set Register (FREIES)

The settings in the Error Interrupt Enable register determine which status changes in the Error Interrupt Register will result in an interrupt.

Writing a "1" sets the specific enable bit to "0" in the FREIER register, writing a "0" has no effect.

FlexRay Error Interrupt Enable Set Register (FREIES) <P4 address: location H'FFBF F030>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABBE	LTVBE	EDBE	—	—	—	—	—	TABAE	LTVAE	EDAE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHFE	IOBAE	IIBAE	EFAE	RFOE	PERRE	CCLE	CCFE	SFOE	SFBME	CNAE	PEMCE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26	TABBE	0	R	W	Transmission Across Boundary Channel B Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
25	LTVBE	0	R	W	Latest Transmit Violation Channel B Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
24	EDBE	0	R	W	Error Detected on Channel B Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
23 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
18	TABAE	0	R	W	Transmission Across Boundary Channel A Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
17	LTVAE	0	R	W	Last Transmit Violation Channel A Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
16	EDAE	0	R	W	Error Detected on Channel A Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	MHFE	0	R	W	Message Handler Constraints Flag Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
10	IOBAE	0	R	W	Illegal Output Buffer Access Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
9	IIBAE	0	R	W	Illegal Input Buffer Access Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled

Bit	Abbreviation	After Reset	R	W	Description
8	EFAE	0	R	W	Empty FIFO Access Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
7	RFOE	0	R	W	Receive FIFO Overrun Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
6	PERRE	0	R	W	Parity Error Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
5	CCLE	0	R	W	CHI Command Locked Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
4	CCFE	0	R	W	Clock Correction Failure Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
3	SFOE	0	R	W	Sync Frame Overflow Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled

Bit	Abbreviation	After Reset	R	W	Description
2	SFBME	0	R	W	Sync Frames Below Minimum Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
1	CNAE	0	R	W	Command Not Accepted Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
0	PEMCE	0	R	W	POC Error Mode Changed PEMCE Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled

32.5.6 FlexRay Error Interrupt Enable Reset Register (FREIER)

The settings in the Error Interrupt Enable register determine which status changes in the Error Interrupt Register will result in an interrupt.

Writing a "1" sets the specific enable bit to "0" in the FREIER register, writing a "0" has no effect.

FlexRay Error Interrupt Enable Reset Register (FREIER) <P4 address: location H'FFBF F034>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABBE	LTVBE	EDBE	—	—	—	—	—	TABAE	LTVAE	EDAE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHFE	IOBAE	IIBAE	EFAE	RFOE	PERRE	CCLE	CCFE	SFOE	SFBME	CNAE	PEMCE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26	TABBE	0	R	W	Transmission Across Boundary Channel B Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
25	LTVBE	0	R	W	Latest Transmit Violation Channel B Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
24	EDBE	0	R	W	Error Detected on Channel B Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
23 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
18	TABAE	0	R	W	Transmission Across Boundary Channel A Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
17	LTVAE	0	R	W	Last Transmit Violation Channel A Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
16	EDAE	0	R	W	Error Detected on Channel A Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
15 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11	MHFE	0	R	W	Message Handler Constraints Flag Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
10	IOBAE	0	R	W	Illegal Output Buffer Access Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
9	IIBAE	0	R	W	Illegal Input Buffer Access Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled

Bit	Abbreviation	After Reset	R	W	Description
8	EFAE	0	R	W	Empty FIFO Access Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
7	RFOE	0	R	W	Receive FIFO Overrun Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
6	PERRE	0	R	W	Parity Error Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
5	CCLE	0	R	W	CHI Command Locked Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
4	CCFE	0	R	W	Clock Correction Failure Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
3	SFOE	0	R	W	Sync Frame Overflow Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled

Bit	Abbreviation	After Reset	R	W	Description
2	SFBME	0	R	W	Sync Frames Below Minimum Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
1	CNAE	0	R	W	Command Not Accepted Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
0	PEMCE	0	R	W	POC Error Mode Changed PEMCE Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled

32.5.7 FlexRay Status Interrupt Enable Set Register (FRSIES)

The settings in the Status Interrupt Enable register determine which status changes in the Status Interrupt Register will result in an interrupt.

Writing a "1" sets the specific enable bit to "1" in the FREIES register, writing a "0" has no effect.

FlexRay Status Interrupt Enable Set Register (FRSIES) <P4 address: location H'FFBF F038>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSBE	WUPBE	—	—	—	—	—	—	MTSAE	WUPAE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDSE	MBSIE	SUCSE	SWEE	TOBCE	TIBCE	TI1E	TI0E	NMVCE	RFCLE	RFNEE	RXIE	TXIE	CYCSE	CASE	WSTE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 26	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
25	MTSBE	0	R	W	MTS Received on Channel B Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
24	WUPBE	0	R	W	Wakeup Pattern Channel B Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
23 to 18	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
17	MTSAE	0	R	W	MTS Received on Channel A Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled

Bit	Abbreviation	After Reset	R	W	Description
16	WUPAE	0	R	W	Wakeup Pattern Channel A Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
15	SDSE	0	R	W	Status of Dynamic Segment Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
14	MBSIE	0	R	W	Message Buffer Status Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
13	SUCSE	0	R	W	Startup Completed Successfully Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
12	SWEE	0	R	W	Stop Watch Event Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
11	TOBCE	0	R	W	Transfer Output Buffer Completed Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled

Bit	Abbreviation	After Reset	R	W	Description
10	TIBCE	0	R	W	Transfer Input Buffer Completed Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
9	TI1E	0	R	W	Timer Interrupt 1 Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
8	TI0E	0	R	W	Timer Interrupt 0 Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
7	NMVCE	0	R	W	Network Management Vector Changed Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
6	RFCLE	0	R	W	Receive FIFO Critical Level Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
5	RFNEE	0	R	W	Receive FIFO Not Empty Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled

Bit	Abbreviation	After Reset	R	W	Description
4	RXIE	0	R	W	Receive Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
3	TXIE	0	R	W	Transmit Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
2	CYCSE	0	R	W	Cycle Start Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
1	CASE	0	R	W	Collision Avoidance Symbol Interrupt Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled
0	WSTE	0	R	W	Wakeup Status Interrupt WSTE Enable Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt enabled

32.5.8 FlexRay Status Interrupt Enable Reset Register (FRSIER)

The settings in the Status Interrupt Enable register determine which status changes in the Status Interrupt Register will result in an interrupt.

Writing a "1" sets the specific enable bit to "0" in the FREIER register, writing a "0" has no effect.

FlexRay Status Interrupt Enable Reset Register (FRSIER) <P4 address: location H'FFBF F03C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSBE	WUPBE	—	—	—	—	—	—	MTSAE	WUPAE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDSE	MBSIE	SUCSE	SWEE	TOBCE	TIBCE	TI1E	TI0E	NMVCE	RFCLE	RFNEE	RXIE	TXIE	CYCSE	CASE	WSTE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 26	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
25	MTSBE	0	R	W	MTS Received on Channel B Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
24	WUPBE	0	R	W	Wakeup Pattern Channel B Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
23 to 18	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
17	MTSAE	0	R	W	MTS Received on Channel A Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled

Bit	Abbreviation	After Reset	R	W	Description
16	WUPAE	0	R	W	Wakeup Pattern Channel A Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
15	SDSE	0	R	W	Status of Dynamic Segment Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
14	MBSIE	0	R	W	Message Buffer Status Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
13	SUCSE	0	R	W	Startup Completed Successfully Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
12	SWEE	0	R	W	Stop Watch Event Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
11	TOBCE	0	R	W	Transfer Output Buffer Completed Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled

Bit	Abbreviation	After Reset	R	W	Description
10	TIBCE	0	R	W	Transfer Input Buffer Completed Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
9	TI1E	0	R	W	Timer Interrupt 1 Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
8	TI0E	0	R	W	Timer Interrupt 0 Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
7	NMVCE	0	R	W	Network Management Vector Changed Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
6	RFCLE	0	R	W	Receive FIFO Critical Level Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
5	RFNEE	0	R	W	Receive FIFO Not Empty Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled

Bit	Abbreviation	After Reset	R	W	Description
4	RXIE	0	R	W	Receive Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
3	TXIE	0	R	W	Transmit Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
2	CYCSE	0	R	W	Cycle Start Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
1	CASE	0	R	W	Collision Avoidance Symbol Interrupt Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled
0	WSTE	0	R	W	Wakeup Status Interrupt WSTE Bit • In reading 0: Interrupt disabled 1: Interrupt enabled • In writing 0: Ignored 1: Interrupt disabled

32.5.9 FlexRay Interrupt Line Enable Register (FRILE)

Each of the two interrupt lines to the CPU (FlexRay_int0, FlexRay_int1) can be enabled / disabled separately by programming bits EINT0 and EINT1.

FlexRay Interrupt Line Enable Register (FRILE) <P4 address: location H'FFBF F043>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	EINT1	EINT0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	EINT1	0	R	W	Enable Interrupt Line 1 Bit 0: Interrupt line FlexRay_int1 disabled 1: Interrupt line FlexRay_int1 enabled
0	EINT0	0	R	W	Enable Interrupt Line 0 Bit 0: Interrupt line FlexRay_int0 disabled 1: Interrupt line FlexRay_int0 enabled

32.5.10 FlexRay Timer0 Configuration Register (FRT0C)

Absolute timer. Specifies in terms of cycle count and macrotick the point in time when the timer 0 interrupt occurs. When the timer 0 interrupt is asserted, output signal (FlexRay_tint0) is set to "1" for the duration of one macrotick and the TI0 bit in the FRSIR register is set to "1".

Timer 0 can be activated as long as the POC is either in NORMAL_ACTIVE state or in NORMAL_PASSIVE state. Timer 0 is deactivated when leaving NORMAL_ACTIVE state or NORMAL_PASSIVE state except for transitions between the two states.

Before reconfiguration of the timer, the timer has to be halted first by writing the T0RC bit to "0".

Note: • The configuration of timer 0 is compared against the macrotick counter value, there is no separate counter for timer 0.

FlexRay Timer0 Configuration Register (FRT0C) <P4 address: location H'FFBF F044>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	T0MO13	T0MO12	T0MO11	T0MO10	T0MO9	T0MO8	T0MO7	T0MO6	T0MO5	T0MO4	T0MO3	T0MO2	T0MO1	T0MO0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	T0CC6	T0CC5	T0CC4	T0CC3	T0CC2	T0CC1	T0CC0	—	—	—	—	—	—	T0MS	T0RC
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 30	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
29 to 16	T0MO13 to T0MO0	All 0	R	W	Timer 0 Macrotick Offset Bit*[1] Configures the macrotick offset from the beginning of the cycle where the interrupt is to occur. The Timer 0 Interrupt occurs at this offset for each cycle of the cycle set.
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 8	T0CC6 to T0CC0	All 0	R	W	Timer 0 Cycle Code Bit*[1] The 7-bit timer 0 cycle code determines the cycle set used for generation of the timer 0 interrupt. For details about the configuration of the cycle code see section 32.18.2, Cycle Counter Filtering.
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	T0MS	0	R	W	Timer 0 Mode Select Bit 0: Single-shot mode 1: Continuous mode
0	T0RC	0	R	W	Timer 0 Run Control Bit 0: Timer 0 halted 1: Timer 0 running

Note: *1 Before reconfiguration of the timer, the timer has to be halted first by writing the T0RC bit to "0".

32.5.11 FlexRay Timer 1 Configuration Register (FRT1C)

Relative timer. After the specified number of macroticks has expired, the timer 1 interrupt is asserted, output signal (FlexRay_tint1) is set to "1" for the duration of one macrotick and the TI1 bit in the FRSIR register is set to "1".

Timer 1 can be activated as long as the POC is either in NORMAL_ACTIVE state or in NORMAL_PASSIVE state. Timer 1 is deactivated when leaving NORMAL_ACTIVE state or NORMAL_PASSIVE state except for transitions between the two states.

Before reconfiguration of the timer, the timer has to be halted first by writing the T1RC bit to "0".

FlexRay Timer 1 Configuration Register (FRT1C) <P4 address: location H'FFBF F048>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	T1MC13	T1MC12	T1MC11	T1MC10	T1MC9	T1MC8	T1MC7	T1MC6	T1MC5	T1MC4	T1MC3	T1MC2	T1MC1	T1MC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	T1MS	T1RC
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0002 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 30	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
29 to 16	T1MC13 to T1MC0	H'0002	R	W	Timer 1 Macrotick Count Bit*[1] When the configured macrotick count reached the timer 1 interrupt is generated. Valid values are: 2 to 16383 MT in continuous mode 1 to 16383 MT in single-shot mode
15 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	T1MS	0	R	W	Timer 1 Mode Select Bit 0: Single-shot mode 1: Continuous mode
0	T1RC	0	R	W	Timer 1 Run Control Bit 0: Timer 1 halted 1: Timer 1 running

Note: *1 Before reconfiguration of the timer, the timer has to be halted first by writing the T1RC bit to "0".

32.5.12 FlexRay Stop Watch Register 1 (FRSTPW1)

The stop watch is activated by a FlexRay_int0 event or FlexRay_int1 event or by the program by writing the SSWT bit to "1". With the macrotick counter increment following next to the stop watch activation the actual cycle counter and macrotick values are captured in the FRSTPW1 register while the slot counter values for channel A and B are captured in the FRSTPW2 register.

FlexRay Stop Watch Register 1 (FRSTPW1) <P4 address: location H'FFBF F04C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	SMTV13	SMTV12	SMTV11	SMTV10	SMTV9	SMTV8	SMTV7	SMTV6	SMTV5	SMTV4	SMTV3	SMTV2	SMTV1	SMTV0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SCCV5	SCCV4	SCCV3	SCCV2	SCCV1	SCCV0	—	EINT1	EINT0	—	SSWT	—	SWMS	ESWT
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 30	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
29 to 16	SMTV13 to SMTV0	All 0	R	0	Stop Watch Captured Macrotick Value State of the macrotick counter when the stop watch event occurred. Valid values are 0 to 15999.
15 to 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13 to 8	SCCV5 to SCCV0	All 0	R	0	Stop Watch Captured Cycle Counter Value State of the cycle counter when the stop watch event occurred. Valid values are 0 to 63.
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6	EINT1	0	R	W	Enable Interrupt 1 Trigger Bit*4 0: Stop watch trigger by FlexRay_int1 disabled 1: FlexRay_int1 event triggers stop watch
5	EINT0	0	R	W	Enable Interrupt 0 Trigger Bit*4 0: Stop watch trigger by FlexRay_int0 disabled 1: FlexRay_int0 event triggers stop watch
4	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
3	SSWT	0	R	W	Software Stop Watch Trigger Bit*1*3 0: Software trigger reset 1: Stop watch activated by software trigger
2	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
1	SWMS	0	R	W	Stop Watch Mode Select Bit When the CPU writes this bit to "1" the stop watch is activated. After the actual cycle counter and macrotick value are stored in the Stop Watch register this bit is set to 0. 0: Single-shot mode 1: Continuous mode
0	ESWT	0	R	W	Enable Stop Watch Trigger Bit*1*2 If enabled FlexRay_int0 event or FlexRay_int1 event activates the stop watch. 0: Stop watch trigger disabled 1: Stop watch trigger enabled

Notes: *1 Bits ESWT and SSWT cannot be set to "1" simultaneously. In this case the write access to the register is ignored, and both bits keep their previous values. Either the FlexRay_int0 event or FlexRay_int1 event event or the software stop watch trigger may be used.

*2 In single-slot mode this bit is set to "0" after the actual cycle counter and macrotick value are stored in the Stop Watch register.

*3 This bit is only writable while the ESWT bit = "0".

*4 Enables stop watch trigger by FlexRay_int0 event or FlexRay_int1 event only if the ESWT bit = "1".

32.5.13 FlexRay Stop Watch Register 2 (FRSTPW2)

FlexRay Stop Watch Register 2 (FRSTPW2) <P4 address: location H'FFBF F050>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SSCVB10	SSCVB9	SSCVB8	SSCVB7	SSCVB6	SSCVB5	SSCVB4	SSCVB3	SSCVB2	SSCVB1	SSCVB0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SSCVA10	SSCVA9	SSCVA8	SSCVA7	SSCVA6	SSCVA5	SSCVA4	SSCVA3	SSCVA2	SSCVA1	SSCVA0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	N	Reserved Bits These bits are always read as "0".
26 to 16	SSCVB10 to SSCVB0	All 0	R	N	Stop Watch Captured Slot Counter Value Channel B State of the slot counter for channel B when the stop watch event occurred. Valid values are 0 to 2047.
15 to 11	—	All 0	0	N	Reserved Bits These bits are always read as "0".
10 to 0	SSCVA10 to SSCVA0	All 0	R	N	Stop Watch Captured Slot Counter Value Channel A State of the slot counter for channel A when the stop watch event occurred. Valid values are 0 to 2047.

32.5.14 FlexRay Timer Interrupt Request Status Register (FXRTISR)

Each bit in this register is set if the corresponding FlexRay Timer interrupt event occurs regardless of the setting of the FlexRay timer interrupt enable register (FXRTIER). An interrupt will be only generated if the corresponding bit in the FXRTIER register is enabled.

Writing '1' to the corresponding bit position in the FXRTIER register clears the status. Writing '0' has no effect on the flag. Only bits that are set to '1' are allowed to clear.

FlexRay Timer Interrupt Request Status Register (FXRTISR) <P4 address: location H'FFBF F00C>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FT1IS	FT0IS
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	FT1IS	0	R	W	FlexRay Timer 1 Interrupt Status Bit This flag is set whenever FlexRay Timer 1 matches the condition configured in the FlexRay Timer 1 configuration register (FRT1C). 0: No interrupt 1: FlexRay Timer 1 interrupt occurred
0	FT0IS	0	R	W	FlexRay Timer 0 Interrupt Status Bit This flag is set whenever FlexRay Timer 0 matches the condition configured in the FlexRay Timer 1 configuration register (FRT0C). 0: No interrupt 1: FlexRay Timer 0 interrupt occurred

32.5.15 FlexRay Timer Interrupt Enable Register (FXRTIER)

It is possible to configure individually the timer interrupt enable for each FlexRay timer (T0 and T1).

FlexRay Timer Interrupt Enable Register (FXRTIER) <P4 address: location H'FFBF F00D>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FT1IEN	FT0IEN
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	FT1IEN	0	R	W	FlexRay Timer 1 Interrupt Enable Bit 0: Disabled 1: Enabled
0	FT0IEN	0	R	W	FlexRay Timer 0 Interrupt Enable Bit 0: Disabled 1: Enabled

32.6 CC Control Registers

This section describes the registers provided by the CC to allow the CPU to control the operation of the CC. The FlexRay protocol specification requires the CPU to write application configuration data in CONFIG state only. Please consider that the configuration registers are not locked for writing in DEFAULT_CONFIG state.

The configuration data is initialized when DEFAULT_CONFIG state is entered from hard reset. To change POC state from DEFAULT_CONFIG to CONFIG state, the CPU has to apply CHI command CONFIG. If the CPU wants the CC to leave CONFIG state, the CPU has to proceed as described in section 32.4.2, FlexRay Lock Register (FRLCK).

32.6.1 FlexRay SUC Configuration Register 1 (FRSUCC1)

FlexRay SUC Configuration Register 1 (FRSUCC1) <P4 address: location H'FFBF F080>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	CCHB	CCHA	MTSB	MTSA	HCSE	TSM	WUCS	PTA4	PTA3	PTA2	PTA1	PTA0
After Reset:	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSA4	CSA3	CSA2	CSA1	CSA0	—	TXSY	TXST	PBSY	—	—	—	CMD3	CMD2	CMD1	CMD0
After Reset:	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0

<After Reset: H'0C40 1080>

Bit	Abbreviation	After Reset	R	W	Description
31 to 28	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
27	CCHB	1	R	W	Connected to Channel B Bit (pChannels) 0: Not connected to channel B 1: Node connected to channel B (default after hard reset)
26	CCHA	1	R	W	Connected to Channel A Bit (pChannels) 0: Not connected to channel A 1: Node connected to channel A (default after hard reset)
25	MTSB	0	R	W	Select Channel B for MTS Transmission Bit*2*5*6 0: Channel B disabled for MTS transmission 1: Channel B selected for MTS transmission
24	MTSA	0	R	W	Select Channel A for MTS Transmission Bit*2*5*6 0: Channel A disabled for MTS transmission 1: Channel A selected for MTS transmission
23	HCSE	0	R	W	Halt due to Clock Sync Error Bit (pAllowHaltDueToClock)*2 0: CC will enter / remain in NORMAL_PASSIVE 1: CC will enter HALT state

Bit	Abbreviation	After Reset	R	W	Description
22	TSM	1	R	W	Transmission Slot Mode Bit (pSingleSlotEnabled)*[2] Selects the initial transmission slot mode. In SINGLE slot mode the CC may only transmit in the preconfigured key slot. The key slot ID is configured in the header section of message buffer 0 respectively message buffers 0 and 1 depending on the SPLM bit in the FRMRCR register. In case the TSM bit = 1, message buffer 0 respectively message buffers 0, 1 can be (re)configured in CONFIG state only. In ALL slot mode the CC may transmit in all slots. The TSM bit is a configuration bit which can only be set to 1 / 0 by the program. The bit can be written in CONFIG state only. The CC changes to ALL slot mode when the CPU successfully applied the ALL_SLOTS command by writing bits CMD3 to CMD0 = B'0101 in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. The actual slot mode is monitored by bits SLM1 to SLM0 in the FRCCSV register. 0: ALL Slot Mode 1: SINGLE Slot Mode (default after hard reset)
21	WUCS	0	R	W	Wakeup Channel Select Bit (pWakeupChannel)*[2] 0: Send wakeup pattern on channel A 1: Send wakeup pattern on channel B
20 to 16	PTA4 to PTA0	All 0	R	W	Passive to Active Bit (pAllowPassiveToActive)*[2] Defines the number of consecutive even / odd cycle pairs that must have valid clock correction terms before the CC is allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state. If set to 00000b the CC is not allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state. It can be modified in CONFIG state only. Valid values are 0 to 31 even / odd cycle pairs.
15 to 11	CSA4 to CSA0	H'02	R	W	Cold Start Attempts Bit (gColdStartAttempts)*[2]*[4] Configures the maximum number of attempts that a cold starting node is permitted to try to start up the network without receiving any valid response from another node. It can be modified in CONFIG state only. Must be identical in all nodes of a cluster. Valid values are 2 to 31.
10	—	0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9	TXSY	0	R	W	Transmit Sync Frame in Key Slot Bit (pKeySlotUsedForSync)*[2]*[3] 0: No sync frame transmission in key slot, node is neither sync nor coldstart node 1: Key slot used to transmit sync frame, node is sync node
8	TXST	0	R	W	Transmit Startup Frame in Key Slot Bit (pKeySlotUsedForStartup)*[2]*[3] 0: No startup frame transmission in key slot, node is non-coldstarter 1: Key slot used to transmit startup frame, node is leading or following coldstarter
7	PBSY	1	R	—	POC Busy Flag*[1] 0: POC not busy, bits CMD0 to CMD3 writable 1: POC is busy, bits CMD0 to CMD3 locked
6 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
3 to 0	CMD3 to CMD0	All 0	R	W	CHI Command Vector The CPU may write any CHI command at any time, but certain commands are enabled only in certain POC states. If a command is not enabled, it will not be executed, the CHI command vector (bits CMD3 to CMD0) will be set to B'0000 = command_not_accepted, and the CNA bit in the FREIR register will be set to "1". In case the previous CHI command has not yet completed, the CCL bit in the FREIR register is set to "1" together with the CNA bit; the CHI command needs to be repeated. Except for HALT state, a POC state change command applied while the CC is already in the requested POC state neither causes a state change nor will the CNA bit in the FREIR register be set to "1". Reading bits CMD3 to CMD0 shows whether the last CHI command was accepted. The actual POC state is monitored by bits POCS5 to POCS0 in the FRCCSV register. Each command are described below this table. 0000: command_not_accepted 0001: CONFIG 0010: READY 0011: WAKEUP 0100: RUN 0101: ALL_SLOTS 0110: HALT 0111: FREEZE 1000: SEND_MTS 1001: ALLOW_COLDSTART 1010: RESET_STATUS INDICATORS 1011: reserved 1100: CLEAR_RAMS 1101 to 1111: reserved

Notes: *1 Set to "1" after hard reset during initialization of internal RAM block.

*2 The bit can be modified in CONFIG state only.

*3 The protocol requires that both bits TXST and TXSY are set for coldstart nodes.

*4 Must be identical in all nodes of a cluster.

*5 If both bits MTSA and MTSB are set to "1", an MTS symbol will be transmitted on both channels when requested by writing bits CMD3 to CMD0 = "B'1000".

*6 Bits MTSA and MTSB may also be changed outside DEFAULT_CONFIG or CONFIG state when the write to FRSUCC1 register is directly preceded by the unlock sequence for the Configuration Lock Key as described in the FRLCK register. This may be combined with CHI command SEND_MTS.

(1) command_not_accepted (CMD3 to 0 = "B'0000")

Bits CMD3 to CMD0 are set to 0000b due to one of the following conditions:

- Illegal command applied by the CPU
- CPU applied command to leave CONFIG state without preceding config lock key
- CPU applied new command while execution of the previous CPU command has not completed
- CPU writes command_not_accepted

When bits CMD3 to CMD0 are set to 0000b, the CNA bit in the FREIR register is set to "1", and - if enabled - an interrupt is generated.

Commands which are not accepted are not executed.

(2) CONFIG (CMD3 to 0 = "B'0001")

Go to POC state CONFIG when called in POC states DEFAULT_CONFIG, READY. When called in HALT state the CC transits to POC state DEFAULT_CONFIG. When called in any other state, bits CMD3 to CMD0 will be set to 0000b = command_not_accepted.

(3) READY (CMD3 to 0 = "B'0010")

Go to POC state READY when called in POC states CONFIG, NORMAL_ACTIVE, NORMAL_PASSIVE, STARTUP, or WAKEUP. When called in any other state, bits CMD3 to CMD0 will be set to 0000b = command_not_accepted.

(4) WAKEUP (CMD3 to 0 = "B'0011")

Go to POC state WAKEUP when called in POC state READY. When called in any other state, bits CMD3 to CMD0 will be set to 0000b = command_not_accepted.

(5) RUN (CMD3 to 0 = "B'0100")

Go to POC state STARTUP when called in POC state READY. When called in any other state, bits CMD3 to CMD0 will be set to 0000b = command_not_accepted.

(6) ALL_SLOTS (CMD3 to 0 = "B'0101")

Leave SINGLE slot mode after successful startup / integration at the next end of cycle when called in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. When called in any other state, bits CMD3 to CMD0 will be set to 0000b = command_not_accepted.

(7) HALT (CMD3 to 0 = "B'0110")

Set halt request (the HRQ bit in the FRCCSV register) to "1" and go to POC state HALT at the end of cycle when called in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. When called in any other state, bits CMD3 to CMD0 will be set to 0000b = command_not_accepted.

(8) FREEZE (CMD3 to 0 = "B'0111")

Set the freeze status indicator (the FSI bit in the FRCCSV register) and go to POC state HALT immediately. Can be called from any state.

(9) SEND_MTS (CMD3 to 0 = "B'1000")

Send single MTS symbol during the symbol window on the channel already configured by bits MTSA and MTSB in POC state NORMAL_ACTIVE after CC entered ALL slot mode (bits SLM1 to SLM0 in the FRCCSV register = B'11) when called by one Macrotick before starting the symbol window. When called one Macrotic before that symbol window or later, the MTS symbol is sent during the following symbol window. When called in any other state, or when called while a previously requested MTS has not yet been transmitted, bits CMD3 to CMD0 will be set to 0000b = command_not_accepted.

(10) ALLOW_COLDSTART (CMD3 to 0 = "B'1001")

The command sets the CSI bit in the FRCCSV register to "0" to enable the node to become leading coldstarter. When called in states DEFAULT_CONFIG, CONFIG, HALT, bits CMD3 to CMD0 will be set to "B'0000" = command_not_accepted. To become leading coldstarter, it is also required that both bits TXST and TXSY are set to "1".

(11) RESET_STATUS_INDICATORS (CMD3 to 0 = "B'1010")

Reset status flags CSNI, CSAI, and WSV2 to WSV0 in the FRCCSV register to their default values. May be called in POC states READY and STARTUP. When called in any other state, bits CMD3 to CMD0 will be reset to B'0000 = command_not_accepted.

(12) CLEAR_RAMS (CMD3 to 0 = "B'1100")

Sets the CRAM bit in the FRMHDS register to "1" when called in DEFAULT_CONFIG or CONFIG state. When called in any other state, bits CMD3 to CMD0 will be set to "B'0000" = command_not_accepted. The CRAM bit in the FRMHDS register is also set to "1" when the CC leaves hard reset. By setting the CRAM bit in the FRMHDS register all internal RAM blocks are initialized to "0". During the initialization of the RAMs, the PBSY bit will show POC busy. Access to the configuration and status registers is possible during execution of CHI command CLEAR_RAMS.

The initialization of the FlexRay internal RAM blocks requires 2048 PAck cycles. There should be no CPU access to IBF or OBF during initialization of the internal RAM blocks after hard reset or after assertion of CHI command CLEAR_RAMS. Before asserting CHI command CLEAR_RAMS the CPU should make sure that no transfer between Message RAM and IBF / OBF is ongoing. This command also resets the Message Buffer Status registers (FRMHDS, FRLDTS, FRFSR, FRMHDF, FRTXRQi (i = 1 to 4), FRNDATi (i = 1 to 4), and FRMBSCi (i = 1 to 4).

Note: • All accepted commands with exception of CLEAR_RAMS and SEND_MTS will cause a change of the POC state in the FRck domain after at most 8 cycles of the slower of the two clocks (PAck and FRck), assumed that POC was not busy when the command was applied and that no POC state change was forced by bus activity in that time frame. Reading the FRCCSV register will show data that is additionally delayed by synchronization from FRck to PAck domain and by the Host-specific CPU interface. The maximum additional delay is 12 cycles of the slower of the two clocks (PAck and FRck).

Table 32.5 below references the CHI commands from the FlexRay Protocol Specification v2.1 to the FlexRay CHI command vector (bits CMD3 to CMD0).

Table 32.5 Reference to CHI Command Summary from FlexRay Protocol Specification

CHI command	Where processed (POC States)	CHI Command Vector Bits CMD3 to CMD0
ALL_SLOTS	POC: normal active, POC: normal passive	ALL_SLOTS
ALLOW_COLDSTART	All except POC: default config, POC: config, POC: halt	ALLOW_COLDSTART
CONFIG	POC: default config, POC: ready	CONFIG
CONFIG_COMPLETE	POC: config	Unlock sequence & READY
DEFAULT_CONFIG	POC: halt	CONFIG
FREEZE	All	FREEZE
HALT	POC: normal active, POC: normal passive	HALT
READY	All except POC: default config, POC: config, POC: ready, POC: halt	READY
RUN	POC: ready	RUN
WAKEUP	POC: ready	WAKEUP

32.6.2 FlexRay SUC Configuration Register 2 (FRSUCC2)

Note: • The wakeup / startup noise timeout is calculated as follows:
LT × (LTN + 1)

FlexRay SUC Configuration Register 2 (FRSUCC2) <P4 address: location H'FFBF F084>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	LTN3	LTN2	LTN1	LTN0	—	—	—	LT20	LT19	LT18	LT17	LT16
After Reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LT15	LT14	LT13	LT12	LT11	LT10	LT9	LT8	LT7	LT6	LT5	LT4	LT3	LT2	LT1	LT0
After Reset:	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0

<After Reset: H'0100 0504>

Bit	Abbreviation	After Reset	R	W	Description
31 to 28	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
27 to 24	LTN3 to LTN0	0001	R	W	Listen Timeout Noise Bit (gListenNoise - 1)*1 Configures the upper limit for startup and wakeup listen timeout in the presence of noise expressed as a multiple of pdListenTimeout. The range for gListenNoise is 2 to 16.
23 to 21	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
20 to 0	LT20 to LT0	H'000504	R	W	Listen Timeout Bit (pdListenTimeout) Configures wakeup / startup listen timeout in µT. The range for pdListenTimeout is 1284 to 1283846 µT.

Notes: *1 Must be configured identical in all nodes of a cluster.
- The CC accepts modifications of the register in CONFIG state only.

32.6.3 FlexRay SUC Configuration Register 3 (FRSUCC3)

FlexRay SUC Configuration Register 3 (FRSUCC3) <P4 address: location H'FFBF F08B>

Bit:	7	6	5	4	3	2	1	0
	WCF3	WCF2	WCF1	WCF0	WCP3	WCP2	WCP1	WCP0
After Reset:	0	0	0	1	0	0	0	1

<After Reset: H'11>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	WCF3 to WCF0	0001	R	W	Maximum Without Clock Correction Fatal Bit (gMaxWithoutClockCorrectionFatal)*1 Defines the number of consecutive even / odd cycle pairs with missing clock correction terms that will cause a transition from NORMAL_ACTIVE or NORMAL_PASSIVE to HALT state.
3 to 0	WCP3 to WCP0	0001	R	W	Maximum Without Clock Correction Passive Bit (gMaxWithoutClockCorrectionPassive) Defines the number of consecutive even / odd cycle pairs with missing clock correction terms that will cause a transition from NORMAL_ACTIVE to NORMAL_PASSIVE state. Valid values are 1 to 15 cycle pairs.

Notes: *1 The transition to HALT state is prevented if the HCSE bit in the FRSUCC1 register is not set to "1".

- The CC accepts modifications of the register in CONFIG state only.
- Must be configured identical in all nodes of a cluster.

32.6.4 FlexRay NEM Configuration Register (FRNEMC)

FlexRay NEM Configuration Register (FRNEMC) <P4 address: location H'FFBF F08F>

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	NML3	NML2	NML1	NML0
After Reset:	0	0	0	0	0	0	0	0

<After Reset: H'00>

Bit	Abbreviation	After Reset	R	W	Description
7 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3 to 0	NML3 to NML0	All 0	R	W	Network Management Vector Length Bit (gNetworkManagementVectorLength) These bits configure the length of the NM vector. Valid values are 0 to 12 bytes.

Notes:
- The CC accepts modifications of the register in CONFIG state only.
- Must be configured identical in all nodes of a cluster.

32.6.5 FlexRay PRT Configuration Register 1 (FRPRTC1)

FlexRay PRT Configuration Register 1 (FRPRTC1) <P4 address: location H'FFBF F090>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RWP5	RWP4	RWP3	RWP2	RWP1	RWP0	—	RXW8	RXW7	RXW6	RXW5	RXW4	RXW3	RXW2	RXW1	RXW0
After Reset:	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BRP0	SPP1	SPP0	—	CASM6	CASM5	CASM4	CASM3	CASM2	CASM1	CASM0	TSST3	TSST2	TSST1	TSST0
After Reset:	0	0	0	0	0	1	1	0	0	0	1	1	0	0	1	1

<After Reset: H'084C 0633>

Bit	Abbreviation	After Reset	R	W	Description
31 to 26	RWP5 to RWP0	H'02	R	W	Repetitions of Tx Wakeup Pattern Bit (pWakeupPattern) Configures the number of repetitions (sequences) of the Tx wakeup symbol. Valid values are 2 to 63.
25	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
24 to 16	RXW8 to RXW0	H'04C	R	W	Wakeup Symbol Receive Window Length Bit (gdWakeupSymbolRxWindow)*1 Configures the number of bit times used by the node to test the duration of the received wakeup pattern. Valid values are 76 to 301 bit times.
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14	BRP0	0	R	W	Baud Rate Prescaler Bit (gdSampleClockPeriod, pSamplesPerMicrotick) The Baud Rate Prescaler configures the baud rate on the FlexRay bus. One bit time always consists of 8 samples independent of the configured baud rate. 0: 10 MBit/s (default) gdSampleClockPeriod = 12.5 ns = 1 × FRck pSamplePerMicrotick = 2 (1 µT = 25 ns) 1: 5 MBit/s gdSampleClockPeriod = 25 ns= 2 × FRck pSamplePerMicrotick = 1 (1 µT = 25 ns)
13 to 12	SPP1 to SPP0	00	R	W	Strobe Point Position Bit*4 Defines the sample count value for strobing. The strobed bit value is set to the voted value when the sample count is incremented to the value configured by bits SPP1 to SPP0. 00: Sample 5 (default) 01: Sample 4 10: Sample 6 11: Sample 5
11	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
10	CASM6	1	R	—	Collision Avoidance Symbol Max Bit (gdCASRxLowMax)*3
9 to 4	CASM5 to CASM0	100011	R	W	Configures the upper limit of the acceptance window for a collision avoidance symbol (CAS). Valid values are 67 to 99 bit times.

Bit	Abbreviation	After Reset	R	W	Description
3 to 0	TSST3 to TSST0	0011	R	W	Transmission Start Sequence Transmitter Bit (gdTSSTransmitter)*[1] Configures the duration of the Transmission Start Sequence (TSS) in terms of bit times.*[2] Valid values are 3 to 15 bit times.

Notes: *1 Must be identical in all nodes of a cluster.

*2 One bit time = 4 μT = 100 ns@10 Mbps

*3 The CASM6 bit is fixed to "1".

*4 The current revision 2.1 of the FlexRay protocol requires that bits SPP1 to SPP0 = "B'00". The alternate strobe point positions could be used to compensate for asymmetries in the physical layer.

- The CC accepts modifications of the register in CONFIG state only.

32.6.6 FlexRay PRT Configuration Register 2 (FRPRTC2)

FlexRay PRT Configuration Register 2 (FRPRTC2) <P4 address: location H'FFBF F094>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TXL5	TXL4	TXL3	TXL2	TXL1	TXL0	TXI7	TXI6	TXI5	TXI4	TXI3	TXI2	TXI1	TXI0
After Reset:	0	0	0	0	1	1	1	1	0	0	1	0	1	1	0	1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RXL5	RXL4	RXL3	RXL2	RXL1	RXL0	—	—	RXI5	RXI4	RXI3	RXI2	RXI1	RXI0
After Reset:	0	0	0	0	1	0	1	0	0	0	0	0	1	1	1	0

<After Reset: H'0F2D 0A0E>

Bit	Abbreviation	After Reset	R	W	Description
31, 30	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
29 to 24	TXL5 to TXL0	H'0F	R	W	Wakeup Symbol Transmit Low Bit (gdWakeupSymbolTxLow) Configures the number of bit times used by the node to transmit the low phase of the wakeup symbol. Valid values are 15 to 60 bit times.
23 to 16	TXI7 to TXI0	H'2D	R	W	Wakeup Symbol Transmit Idle Bit (gdWakeupSymbolTxIdle) Configures the number of bit times used by the node to transmit the idle phase of the wakeup symbol. Valid values are 45 to 180 bit times.
15, 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13 to 8	RXL5 to RXL0	H'0A	R	W	Wakeup Symbol Receive Low Bit (gdWakeupSymbolRxLow) Configures the number of bit times used by the node to test the duration of the low phase of the received wakeup symbol. Valid values are 10 to 55 bit times.
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5 to 0	RXI5 to RXI0	H'0E	R	W	Wakeup Symbol Receive Idle Bit (gdWakeupSymbolRxIdle) Configures the number of bit times used by the node to test the duration of the idle phase of the received wakeup symbol. Valid values are 14 to 59 bit times.

Notes:
- The CC accepts modifications of the register in CONFIG state only.
- Must be identical in all nodes of a cluster.

32.6.7 FlexRay MHD Configuration Register (FRMHDC)

FlexRay MHD Configuration Register (FRMHDC) <P4 address: location H'FFBF F098>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SLT12	SLT11	SLT10	SLT9	SLT8	SLT7	SLT6	SLT5	SLT4	SLT3	SLT2	SLT1	SLT0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SFDL6	SFDL5	SFDL4	SFDL3	SFDL2	SFDL1	SFDL0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 16	SLT12 to SLT0	All 0	R	W	Start of Latest Transmit Bit (pLatestTx) *[2] Configures the maximum minislot value allowed before inhibiting frame transmission in the dynamic segment of the cycle. Valid values are 0 to 7981 minislots.
15 to 7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6 to 0	SFDL6 to SFDL0	All 0	R	W	Static Frame Data Length Bit (gPayloadLengthStatic) *[1] Configures the cluster-wide payload length for all frames sent in the static segment in double bytes. Valid values are 0 to 127.

Notes: *1 Must be identical in all nodes of a cluster.

*2 There is no transmission in dynamic segment if bits SLT12 to SLT0 are set to "0".

- The CC accepts modifications of the register in CONFIG state only.

32.6.8 FlexRay GTU Configuration Register 1 (FRGTUC1)

FlexRay GTU Configuration Register 1 (FRGTUC1) <P4 address: location H'FFBF F0A0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UT19	UT18	UT17	UT16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UT15	UT14	UT13	UT12	UT11	UT10	UT9	UT8	UT7	UT6	UT5	UT4	UT3	UT2	UT1	UT0
After Reset:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0

<After Reset: H'0000 0280>

Bit	Abbreviation	After Reset	R	W	Description
31 to 20	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
19 to 0	UT19 to UT0	H'00280	R	W	Microtick per Cycle Bit (pMicroPerCycle) Configures the duration of the communication cycle in microticks. Valid value are 640 to 640000 µT.

Note: • The CC accepts modifications of the register in CONFIG state only.

32.6.9 FlexRay GTU Configuration Register 2 (FRGTUC2)

FlexRay GTU Configuration Register 2 (FRGTUC2) <P4 address: location H'FFBF F0A4>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	SNM3	SNM2	SNM1	SNM0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MPC13	MPC12	MPC11	MPC10	MPC9	MPC8	MPC7	MPC6	MPC5	MPC4	MPC3	MPC2	MPC1	MPC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

<After Reset: H'0002 000A>

Bit	Abbreviation	After Reset	R	W	Description
31 to 20	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
19 to 16	SNM3 to SNM0	0010	R	W	Sync Node Max Bit (gSyncNodeMax) Maximum number of frames within a cluster with sync frame indicator bit (the SYN bit) set to "1". Valid values are 2 to 15.
15, 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13 to 0	MPC13 to MPC0	H'000A	R	W	Macrotick Per Cycle Bit (gMacroPerCycle) Configures the duration of one communication cycle in macroticks. Valid values are 10 to 16000 MT.

Notes:
- The CC accepts modifications of the register in CONFIG state only.
- Must be identical in all nodes of a cluster.

32.6.10 FlexRay GTU Configuration Register 3 (FRGTUC3)

FlexRay GTU Configuration Register 3 (FRGTUC3) <P4 address: location H'FFBF F0A8>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MIOB6	MIOB5	MIOB4	MIOB3	MIOB2	MIOB1	MIOB0	—	MIOA6	MIOA5	MIOA4	MIOA3	MIOA2	MIOA1	MIOA0
After Reset:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UIOB7	UIOB6	UIOB5	UIOB4	UIOB3	UIOB2	UIOB1	UIOB0	UIOA7	UIOA6	UIOA5	UIOA4	UIOA3	UIOA2	UIOA1	UIOA0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0202 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
30 to 24	MIOB6 to MIOB0	H'02	R	W	Macrotick Initial Offset Channel B Bit (pMacroInitialOffset[B])*1 Configures the number of macroticks between the static slot boundary and the subsequent macrotick boundary of the secondary time reference point based on the nominal macrotick duration Valid values are 2 to 72 MT.
23	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
22 to 16	MIOA6 to MIOA0	H'02	R	W	Macrotick Initial Offset Channel A Bit (pMacroInitialOffset[A])*1 Configures the number of macroticks between the static slot boundary and the subsequent macrotick boundary of the secondary time reference point based on the nominal macrotick duration.*2 Valid values are 2 to 72 MT.
15 to 8	UIOB7 to UIOB0	All 0	R	W	Microtick Initial Offset Channel B Bit (pMicroInitialOffset[B]) Configures the number of microticks between the actual time reference point on channel B and the subsequent macrotick boundary of the secondary time reference point. The parameter depends on pDelayCompensation[B] and therefore has to be set for each channel independently. Valid values are 0 to 240 µT.
7 to 0	UIOA7 to UIOA0	All 0	R	W	Microtick Initial Offset Channel A Bit (pMicroInitialOffset[A]) Configures the number of microticks between the actual time reference point on channel A and the subsequent macrotick boundary of the secondary time reference point. The parameter depends on pDelayCompensation[A] and therefore has to be set for each channel independently. Valid values are 0 to 240 µT.

Notes: *1 Must be identical in all nodes of a cluster.

*2 Nominal macroticks represent values before offset or rate correction.

- The CC accepts modifications of the register in CONFIG state only.

32.6.11 FlexRay GTU Configuration Register 4 (FRGTUC4)

For details about configuration of bits NIT13 to NIT0 and OCS13 to OCS0, see section 32.12.5, Configuration of NIT Start and Offset Correction Start.

FlexRay GTU Configuration Register 4 (FRGTUC4) <P4 address: location H'FFBF F0AC>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	OCS13	OCS12	OCS11	OCS10	OCS9	OCS8	OCS7	OCS6	OCS5	OCS4	OCS3	OCS2	OCS1	OCS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	NIT13	NIT12	NIT11	NIT10	NIT9	NIT8	NIT7	NIT6	NIT5	NIT4	NIT3	NIT2	NIT1	NIT0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

<After Reset: H'0008 0007>

Bit	Abbreviation	After Reset	R	W	Description
31, 20	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
29 to 16	OCS13 to OCS0	H'0008	R	W	Offset Correction Start Bit (gOffsetCorrectionStart - 1) Determines the start of the offset correction within the NIT phase, calculated from start of cycle. Valid values are 8 to 15998 MT.
15, 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13 to 0	NIT13 to NIT0	H'0007	R	W	Network Idle Time Start Bit (gMacroPerCycle - gdNIT - 1) Configures the starting point of the Network Idle Time NIT at the end of the communication cycle expressed in terms of macroticks from the beginning of the cycle. The start of NIT is recognized if Macrotick = gMacroPerCycle - gdNIT -1 and the increment pulse of Macrotick is set Valid values are 7 to 15997 MT.

Notes:
- The CC accepts modifications of the register in CONFIG state only.
- Must be identical in all nodes of a cluster.

32.6.12 FlexRay GTU Configuration Register 5 (FRGTUC5)

FlexRay GTU Configuration Register 5 (FRGTUC5) <P4 address: location H'FFBF F0B0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEC7	DEC6	DEC5	DEC4	DEC3	DEC2	DEC1	DEC0	—	—	—	CDD4	CDD3	CDD2	CDD1	CDD0
After Reset:	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCB7	DCB6	DCB5	DCB4	DCB3	DCB2	DCB1	DCB0	DCA7	DCA6	DCA5	DCA4	DCA3	DCA2	DCA1	DCA0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0E00 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	DEC7 to DEC0	H'0E	R	W	Decoding Correction Bit (pDecodingCorrection) Configures the decoding correction value used to determine the primary time reference point. Valid values are 14 to 143 µT.
23 to 21	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
20 to 16	CDD4 to CDD0	All 0	R	W	Cluster Drift Damping Bit (pClusterDriftDamping) Configures the cluster drift damping value used in clock synchronization to minimize accumulation of rounding errors. Valid values are 0 to 20 µT.
15 to 8	DCB7 to DCB0	All 0	R	W	Delay Compensation Channel B Bit (pDelayCompensation[B]) Used to compensate for reception delays on the indicated channel. This covers assumed propagation delay up to cPropagationDelayMax for microticks in the range of 0.0125 to 0.05 µs. In practice, the minimum of the propagation delays of all sync nodes should be applied. Valid values are 0 to 200 µT.
7 to 0	DCA7 to DCA0	All 0	R	W	Delay Compensation Channel A Bit (pDelayCompensation[A]) Used to compensate for reception delays on the indicated channel. This covers assumed propagation delay up to cPropagationDelayMax for microticks in the range of 0.0125 to 0.05 µs. In practice, the minimum of the propagation delays of all sync nodes should be applied. Valid values are 0 to 200 µT.

Note: • The CC accepts modifications of the register in CONFIG state only.

32.6.13 FlexRay GTU Configuration Register 6 (FRGTUC6)

FlexRay GTU Configuration Register 6 (FRGTUC6) <P4 address: location H'FFBF F0B4>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MOD10	MOD9	MOD8	MOD7	MOD6	MOD5	MOD4	MOD3	MOD2	MOD1	MOD0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ASR10	ASR9	ASR8	ASR7	ASR6	ASR5	ASR4	ASR3	ASR2	ASR1	ASR0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0002 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26 to 16	MOD10 to MOD0	H'002	R	W	Maximum Oscillator Drift Bit (pdMaxDrift) Maximum drift offset between two nodes that operate with unsynchronized clocks over one communication cycle in µT. Valid values are 2 to 1923 µT.
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	ASR10 to ASR0	All 0	R	W	Accepted Startup Range Bit (pdAcceptedStartupRange) Number of microticks constituting the expanded range of measured deviation for startup frames during integration. Valid values are 0 to 1875 µT.

Note: • The CC accepts modifications of the register in CONFIG state only.

32.6.14 FlexRay GTU Configuration Register 7 (FRGTUC7)

FlexRay GTU Configuration Register 7 (FRGTUC7) <P4 address: location H'FFBF F0B8>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NSS9	NSS8	NSS7	NSS6	NSS5	NSS4	NSS3	NSS2	NSS1	NSS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SSL9	SSL8	SSL7	SSL6	SSL5	SSL4	SSL3	SSL2	SSL1	SSL0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

<After Reset: H'0002 0004>

Bit	Abbreviation	After Reset	R	W	Description
31 to 26	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
25 to 16	NSS9 to NSS0	H'002	R	W	Number of Static Slots Bit (gNumberOfStaticSlots)*1 Configures the number of static slots in a cycle. At least 2 coldstart nodes must be configured to startup a FlexRay network. Valid values are 2 to 1023 MT.
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9 to 0	SSL9 to SSL0	H'004	R	W	Static Slot Length Bit (gdStaticSlot) Configures the duration of a static slot in macroticks. Valid values are 4 to 659 MT.

Notes: *1 At least 2 coldstart nodes must be configured to startup a FlexRay network.

- The CC accepts modifications of the register in CONFIG state only.
- Must be identical in all nodes of a cluster.

32.6.15 FlexRay GTU Configuration Register 8 (FRGTUC8)

FlexRay GTU Configuration Register 8 (FRGTUC8) <P4 address: location H'FFBF F0BC>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	NMS12	NMS11	NMS10	NMS9	NMS8	NMS7	NMS6	NMS5	NMS4	NMS3	NMS2	NMS1	NMS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	MSL5	MSL4	MSL3	MSL2	MSL1	MSL0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

<After Reset: H'0000 0002>

Bit	Abbreviation	After Reset	R	W	Description
31 to 29	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
28 to 16	NMS12 to NMS0	All 0	R	W	Number of Minislots Bit (gNumberOfMinislots) Configures the number of minislots within the dynamic segment of a cycle. Valid values are 0 to 7986 MT.
15 to 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5 to 0	MSL5 to MSL0	H'02	R	W	Minislot Length Bit (gdMinislot) Configures the duration of a minislot in macroticks. Valid values are 2 to 63 MT.

Notes:
- The CC accepts modifications of the register in CONFIG state only.
- Must be identical in all nodes of a cluster.

32.6.16 FlexRay GTU Configuration Register 9 (FRGTUC9)

FlexRay GTU Configuration Register 9 (FRGTUC9) <P4 address: location H'FFBF F0C0>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DSI1	DSI0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MAPO4	MAPO3	MAPO2	MAPO1	MAPO0	—	—	APO5	APO4	APO3	APO2	APO1	APO0
After Reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

<After Reset: H'0000 0101>

Bit	Abbreviation	After Reset	R	W	Description
31 to 18	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
17, 16	DSI1, DSI0	00	R	W	Dynamic Slot Idle Phase Bit (gdDynamicSlotIdlePhase)*[1] Configures the duration of the dynamic slot idle phase in minislots. Valid values are 0 to 2 Minislot.
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12 to 8	MAPO4 to MAPO0	H'01	R	W	Minislot Action Point Offset Bit (gdMinislotActionPointOffset) Configures the action point offset in macroticks within the minislots of the dynamic segment. Valid values are 1 to 31 MT.
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5 to 0	APO5 to APO0	H'01	R	W	Action Point Offset Bit (gdActionPointOffset) Configures the action point offset in macroticks within static slots and symbol window. Valid values are 1 to 63 MT.

Notes: *1 The duration of the dynamic slot idle phase has to be greater or equal than the idle detection time.

- The CC accepts modifications of the register in CONFIG state only.
- Must be identical in all nodes of a cluster.

32.6.17 FlexRay GTU Configuration Register 10 (FRGTUC10)

FlexRay GTU Configuration Register 10 (FRGTUC10) <P4 address: location H'FFBF F0C4>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MRC10	MRC9	MRC8	MRC7	MRC6	MRC5	MRC4	MRC3	MRC2	MRC1	MRC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MOC13	MOC12	MOC11	MOC10	MOC9	MOC8	MOC7	MOC6	MOC5	MOC4	MOC3	MOC2	MOC1	MOC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

<After Reset: H'0002 0005>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26 to 16	MRC10 to MRC0	H'002	R	W	Maximum Rate Correction Bit (pRateCorrectionOut) Holds the maximum permitted rate correction value to be applied by the internal clock synchronization algorithm. The CC checks only the internal rate correction value against the maximum rate correction value (absolute value). Valid values are 2 to 1923 µT.
15, 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13 to 0	MOC13 to MOC0	H'0005	R	W	Maximum Offset Correction Bit (pOffsetCorrectionOut) Holds the maximum permitted offset correction value to be applied by the internal clock synchronization algorithm (absolute value). The CC checks only the internal offset correction value against the maximum offset correction value. Valid values are 5 to 15266 µT.

Note: • The CC accepts modifications of the register in CONFIG state only.

32.6.18 FlexRay GTU Configuration Register 11 (FRGTUC11)

FlexRay GTU Configuration Register 11 (FRGTUC11) <P4 address: location H'FFBF F0C8>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ERC2	ERC1	ERC0	—	—	—	—	—	EOC2	EOC1	EOC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ERCC1	ERCC0	—	—	—	—	—	—	EOCC1	EOCC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26 to 24	ERC2 to ERC0	All 0	R	W	External Rate Correction Bit (pExternRateCorrection)*2*3 Holds the external rate correction value in microticks to be applied by the internal clock synchronization algorithm. Valid values are 0 to 7 µT.
23 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18 to 16	EOC2 to EOC0	All 0	R	W	External Offset Correction Bit (pExternOffsetCorrection)*2*3 Holds the external offset correction value in microticks to be applied by the internal clock synchronization algorithm. The value is subtracted / added from / to the calculated offset correction value. Valid values are 0 to 7 µT.
15 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
9, 8	ERCC1 to ERCC0	00	R	W	External Rate Correction Control Bit (vExternRateControl)*1 0X: No external rate correction 10: External rate correction value subtracted from calculated rate correction value 11: External rate correction value added to calculated rate correction value
7 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1, 0	EOCC1, EOCC0	00	R	W	External Offset Correction Control Bit (vExternOffsetControl)*1 0X: No external offset correction 10: External offset correction value subtracted from calculated offset correction value 11: External offset correction value added to calculated offset correction value

Notes: *1 Should be modified only outside NIT.

*2 May be modified in CONFIG state only.

*3 The value is applied during NIT.

32.7 CC Status Registers

During 8/16-bit accesses to status variables coded with more than 8/16-bit, the variable might be updated by the CC between two accesses (non-atomic read accesses). The status vector may change faster than the CPU can poll the status vector, depending on PAck frequency.

32.7.1 FlexRay CC Status Vector Register (FRCCSV)

FlexRay CC Status Vector Register (FRCCSV) <P4 address: location H'FFBF F100>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	PSL5	PSL4	PSL3	PSL2	PSL1	PSL0	RCA4	RCA3	RCA2	RCA1	RCA0	WSV2	WSV1	WSV0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	Undefined	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CSI	CSAI	CSNI	—	—	SLM1	SLM0	HRQ	FSI	POCS5	POCS4	POCS3	POCS2	POCS1	POCS0
After Reset:	0	Undefined	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31, 30	—	All 0	0	N	Reserved Bits These bits are always read as "0".
29 to 24	PSL5 to PSL0	All 0	R	N	POC Status Log Status of bits POCS5 to POCS0 immediately before entering HALT state. Set when entering HALT state. Set to HALT when FREEZE command is applied during HALT state and the FSI bit is not already set i.e. the HALT state was not reached by FREEZE command. Set to B'00 0000 when leaving HALT state.
23 to 19	RCA4 to RCA0	Undefined	R	N	Remaining Coldstart Attempts (vRemainingColdstartAttempts) Indicates the number of remaining coldstart attempts. The RUN command initializes this counter to the maximum number of coldstart attempts as configured by bits CSA4 to CSA0 in the FRSUCC1 register. The initial value of bits RCA4 to RCA0 during CONFIG state is also bits CSA4 to CSA0 in the FRSUCC1 register.

Bit	Abbreviation	After Reset	R	W	Description
18 to 16	WSV2 to WSV0	All 0	R	N	Wakeup Status (vPOC!WakeupStatus)*[6] Indicates the status of the current wakeup attempt. 000 = UNDEFINED. Wakeup not yet executed by the CC. 001 = RECEIVED_HEADER. Set when the CC finishes wakeup due to the reception of a frame header without coding violation on either channel in WAKEUP_LISTEN state. 010 = RECEIVED_WUP. Set when the CC finishes wakeup due to the reception of a valid wakeup pattern on the configured wakeup channel in WAKEUP_LISTEN state. 011 = COLLISION_HEADER. Set when the CC stops wakeup due to a detected collision during wakeup pattern transmission by receiving a valid header on either channel. 100 = COLLISION_WUP. Set when the CC stops wakeup due to a detected collision during wakeup pattern transmission by receiving a valid wakeup pattern on the configured wakeup channel. 101 = COLLISION_UNKNOWN. Set when the CC stops wakeup by leaving WAKEUP_DETECT state after expiration of the wakeup timer without receiving a valid wakeup pattern or a valid frame header. 110 = TRANSMITTED. Set when the CC has successfully completed the transmission of the wakeup pattern. 111 = reserved
15	—	0	0	N	Reserved Bit This bit is always read as "0".
14	CSI	Undefined	R	N	Cold Start Inhibit Flag (vColdStartInhibit)*[5] Indicates that the node is disabled from cold starting. The flag is set to "1" whenever the POC enters READY state due to CHI command READY. The flag has to be set to "0" under control of the CPU by CHI command ALLOW_COLDSTART (bits CMD3 to CMD0 in the FRSUCC1 register = B'1001). 0: Cold starting of node enabled 1: Cold starting of node disabled
13	CSAI	0	R	N	Coldstart Abort Indicator Flag*[4] Coldstart aborted. 0: No state change 1: Coldstart aborted
12	CSNI	0	R	N	Coldstart Noise Indicator Flag (vPOC!ColdstartNoise)*[4] Indicates that the cold start procedure occurred under noisy conditions. 0: No state change 1: The cold start procedure occurred under noisy conditions.
11, 10	—	All 0	0	N	Reserved Bits These bits are always read as "0".

Bit	Abbreviation	After Reset	R	W	Description
9, 8	SLM1, SLM0	00	R	N	Slot Mode Flag (vPOC!SlotMode)*3 Indicates the actual slot mode of the POC in states READY, WAKEUP, STARTUP, NORMAL_ACTIVE, and NORMAL_PASSIVE. Default is SINGLE. Changes to ALL, depending on the TSM bit in the FRSUCC1 register. In NORMAL_ACTIVE or NORMAL_PASSIVE state, the CHI command ALL_SLOTS will change the slot mode from SINGLE over ALL_PENDING to ALL. Set to SINGLE in all other states. 00: SINGLE 01: reserved 10: ALL_PENDING 11: ALL
7	HRQ	0	R	N	Halt Request Flag (vPOC!CHIHaltRequest)*2 0: No state change 1: A request from the CPU has been received to halt the POC at the end of the communication cycle.
6	FSI	0	R	N	Freeze Status Indicator Flag (vPOC!Freeze)*1 0: No state change 1: The POC has entered the HALT state due to CHI command FREEZE or due to an error condition requiring an immediate POC halt.
5 to 0	POCS5 to POCS0	All 0	R	N	Protocol Operation Control Status Flag Indicates the actual state of operation of the CC Protocol Operation Control, with bits POCS5 to POCS4 the POC in the wakeup path, or the POC in the startup path with bits POCS3 to POCS0, as shown in Table 32.6. 00: The actual state of operation of the CC Protocol Operation Control 01: The actual state of operation of the POC in the wakeup path 10: The actual state of operation of the POC in the startup path 11: Reserved

Notes: *1 Set to "0" by transition from HALT to DEFAULT_CONFIG state.

*2 Set to "0" by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

*3 In all states other than NORMAL_ACTIVE or NORMAL_PASSIVE, the CHI command RESET_STATUS_INDICATOR will change the value set with the TSM bit in the FRSUCC1 register to "0".

*4 Set to "0" by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or from READY to STARTUP state.

*5 Set to "1" whenever the POC enters READY state due to CHI command READY. The flag has to be set to "0" under control of the CPU by CHI command ALLOW_COLDSTART (bits CMD3 to CMD0 in the FRSUCC1 register = "B'1001").

*6 Set to "0" by CHI command RESET_STATUS_INDICATORS (bits CMD3 to CMD0 in the FRSUCC1 register = "B'1010") or by transition from HALT to DEFAULT_CONFIG state or from READY to STARTUP state.

- Some WAKEUP states and STARTUP states are active for a short time only before a transition to the next state occurs. Therefore it can not be guaranteed that all states are read by the application.

Table 32.6 POC Status Flags

Bits POCS5 to POCS0	POC Status	
B'00 0000	The actual state of operation of the CC Protocol Operation Control	DEFAULT_CONFIG state
B'00 0001		READY state
B'00 0010		NORMAL_ACTIVE state
B'00 0011		NORMAL_PASSIVE state
B'00 0100		HALT state
B'00 0101		MONITOR_MODE state
B'00 0110 to B'00 1110		Reserved
B'00 0101 to B'00 1110		Reserved
B'00 1111		CONFIG state
B'01 0000	The actual state of operation of the POC in the wakeup path	WAKEUP_STANDBY state
B'01 0001		WAKEUP_LISTEN state
B'01 0010		WAKEUP_SEND state
B'01 0011		WAKEUP_DETECT state
B'01 0100 to B'01 1111		Reserved
B'10 0000	The actual state of operation of the POC in the startup path	STARTUP_PREPARE state
B'10 0001		COLDSTART_LISTEN state
B'10 0010		COLDSTART_COLLISION_RESOLUTION state
B'10 0011		COLDSTART_CONSISTENCY_CHECK state
B'10 0100		COLDSTART_GAP state
B'10 0101		COLDSTART_JOIN state
B'10 0110		INTEGRATION_COLDSTART_CHECK state
B'10 0111		INTEGRATION_LISTEN state
B'10 1000		INTEGRATION_CONSISTENCY_CHECK state
B'10 1001		INITIALIZE_SCHEDULE state
B'10 1010		ABORT_STARTUP state
B'10 1011		STARTUP_SUCCESS state
B'10 1100 to B'11 1111		Reserved

32.7.2 FlexRay CC Error Vector Register (FRCCEV)

FlexRay CC Error Vector Register (FRCCEV) <P4 address: location H'FFBF F106>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PTAC4	PTAC3	PTAC2	PTAC1	PTAC0	ERRM1	ERRM0	—	—	CCFC3	CCFC2	CCFC1	CCFC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	N	Reserved Bits These bits are always read as "0".
12 to 8	PTAC4 to PTAC0	All 0	R	N	Passive to Active Count (vAllowPassiveToActive) Indicates the number of consecutive even / odd cycle pairs that have passed with valid rate and offset correction terms, while the node is waiting to transit from NORMAL_PASSIVE state to NORMAL_ACTIVE state. The transition takes place when bits PTAC4 to PTAC0 equal bits PTA4 to PTA0 in the FRSUCC1 -1.
7, 6	ERRM1, ERRM0	00	R	N	Error Mode Flag (vPOC!ErrorMode) 00: ACTIVE (green) 01: PASSIVE (yellow) 10: COMM_HALT (red) 11: reserved
5, 4	—	All 0	0	N	Reserved Bits These bits are always read as "0".
3 to 0	CCFC3 to CCFC0	All 0	R	N	Clock Correction Failed Counter (vClockCorrectionFailed) Indicates the value (0 to 15) of the clock correction failed counter in the CC. The Clock Correction Failed Counter is incremented by one at the end of any odd communication cycle where either the missing offset correction error or missing rate correction error are active. The Clock Correction Failed Counter is set to "0" at the end of an odd communication cycle if neither the offset correction failed nor the rate correction failed errors are active. The Clock Correction Failed Counter stops at 15.

Note: • Initialized to "H'0000" by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

32.7.3 FlexRay Slot Counter Value Register (FRSCV)

FlexRay Slot Counter Value Register (FRSCV) <P4 address: location H'FFBF F110>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SCCB10	SCCB9	SCCB8	SCCB7	SCCB6	SCCB5	SCCB4	SCCB3	SCCB2	SCCB1	SCCB0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCCA10	SCCA9	SCCA8	SCCA7	SCCA6	SCCA5	SCCA4	SCCA3	SCCA2	SCCA1	SCCA0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	N	Reserved Bits These bits are always read as "0".
26 to 16	SCCB10 to SCCB0	All 0	R	N	Slot Counter Channel B (vSlotCounter[B]) Current slot counter value on channel B. The value is incremented by the CC and set to "0" at the start of a communication cycle. Valid values are 0 to 2047.
15 to 12	—	All 0	0	N	Reserved Bits These bits are always read as "0".
11 to 0	SCCA10 to SCCA0	All 0	R	N	Slot Counter Channel A (vSlotCounter[A]) Current slot counter value on channel A. The value is incremented by the CC and set to "0" at the start of a communication cycle. Valid values are 0 to 2047.

32.7.4 FlexRay Macrotick and Cycle Counter Value Register (FRMTCCV)

FlexRay Macrotick and Cycle Counter Value Register (FRMTCCV) <P4 address: location H'FFBF F114>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CCV5	CCV4	CCV3	CCV2	CCV1	CCV0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MTV13	MTV12	MTV11	MTV10	MTV9	MTV8	MTV7	MTV6	MTV5	MTV4	MTV3	MTV2	MTV1	MTV0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 22	—	All 0	0	N	Reserved Bits These bits are always read as "0".
21 to 16	CCV5 to CCV0	All 0	R	N	Cycle Counter Value (vCycleCounter) Current cycle counter value. The value is incremented by the CC at the start of a communication cycle. Valid values are 0 to 63.
15, 14	—	All 0	0	N	Reserved Bits These bits are always read as "0".
13 to 0	MTV13 to MTV0	All 0	R	N	Macrotick Value (vMacrotick) Current macrotick value. The value is incremented by the CC and set to "0" at the start of a communication cycle. Valid values are 0 to 15999.

Note: • The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.5 FlexRay Rate Correction Value Register (FRRCV)

FlexRay Rate Correction Value Register (FRRCV) <P4 address: location H'FFBF F11A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RCV11	RCV10	RCV9	RCV8	RCV7	RCV6	RCV5	RCV4	RCV3	RCV2	RCV1	RCV0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	N	Reserved Bits These bits are always read as "0".
11 to 0	RCV11 to RCV0	All 0	R	N	Rate Correction Value (vRateCorrection) Rate correction value (two's complement). Calculated internal rate correction value before limitation.*[1] If the RCV value exceeds the limits defined by bits MRC10 to MRC0 in the FRGTUC10 register, the RCLR flag in the FRSFS register is set to "1".

Notes: *1 The external rate correction value is added to the limited rate correction value.

- The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.6 FlexRay Offset Correction Value (FROCV)

FlexRay Offset Correction Value (FROCV) <P4 address: location H'FFBF F11C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	OCV18	OCV17	OCV16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCV15	OCV14	OCV13	OCV12	OCV11	OCV10	OCV9	OCV8	OCV7	OCV6	OCV5	OCV4	OCV3	OCV2	OCV1	OCV0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
32 to 19	—	All 0	0	N	Reserved Bits These bits are always read as "0".
18 to 0	OCV18 to OCV0	All 0	R	N	Offset Correction Value (vOffsetCorrection) Offset correction value (two's complement).*[1] Calculated internal offset correction value before limitation. If the OCV value exceeds the limits defined by bits MOC13 to MOC0 in the FRGTUC10 register, the OCLR flag in the FRSFS register is set to "1".

Notes: *1 The external offset correction value is added to the limited offset correction value.

- The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.7 FlexRay Sync Frame Status Register (FRSFS)

The maximum number of valid sync frames in a communication cycle is 15.

FlexRay Sync Frame Status Register (FRSFS) <P4 address: location H'FFBF F120>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	RCLR	MRCS	OCLR	MOCS
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSBO3	VSBO2	VSBO1	VSBO0	VSBE3	VSBE2	VSBE1	VSBE0	VSAO3	VSAO2	VSAO1	VSAO0	VSAE3	VSAE2	VSAE1	VSAE0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 20	—	All 0	0	N	Reserved Bits These bits are always read as "0".
19	RCLR	0	R	N	Rate Correction Limit Reached Flag The Rate Correction Limit Reached flag signals to the CPU, that the rate correction value has exceeded its limit as defined by bits MRC10 to MRC0 in the FRGTUC10 register. The flag is updated by the CC at start of offset correction phase. 0: Rate correction below limit 1: Rate correction limit reached
18	MRCS	0	R	N	Missing Rate Correction Signal Flag The Missing Rate Correction flag signals to the CPU, that no rate correction calculation can be performed because no pairs of even / odd sync frames were received. The flag is updated by the CC at start of offset correction phase. 0: Rate correction signal valid 1: Missing rate correction signal
17	OCLR	0	R	N	Offset Correction Limit Reached Flag The Offset Correction Limit Reached flag signals to the CPU, that the offset correction value has exceeded its limit as defined by bits MOC13 to MOC0 in the FRGTUC10 register. The flag is updated by the CC at start of offset correction phase. 0: Offset correction below limit 1: Offset correction limit reached
16	MOCS	0	R	N	Missing Offset Correction Signal Flag The Missing Offset Correction flag signals to the CPU, that no offset correction calculation can be performed because no sync frames were received. The flag is updated by the CC at start of offset correction phase. 0: Offset correction signal valid 1: Missing offset correction signal
15 to 12	VSBO3 to VSBO0	All 0	R	N	Valid Sync Frames Channel B, odd communication cycle[*1][*3] Holds the number of valid sync frames received on channel B in the odd communication cycle. The value is updated during the NIT of each odd communication cycle.

Bit	Abbreviation	After Reset	R	W	Description
11 to 8	VSBE3 to VSBE0	All 0	R	N	Valid Sync Frames Channel B, even communication cycle*[1]*[3] Holds the number of valid sync frames received on channel B in the even communication cycle. The value is updated during the NIT of each even communication cycle.
7 to 4	VSAO3 to VSAO0	All 0	R	N	Valid Sync Frames Channel A, odd communication cycle*[1]*[2] Holds the number of valid sync frames received on channel A in the odd communication cycle. The value is updated during the NIT of each odd communication cycle.
3 to 0	VSAE3 to VSAE0	All 0	R	N	Valid Sync Frames Channel A, even communication cycle*[1]*[2] Holds the number of valid sync frames received on channel A in the even communication cycle. The value is updated during the NIT of each even communication cycle.

Notes: *1 If transmission of sync frames is enabled by the TXSY bit in the FRSUCC1 register the value is incremented by one.

*2 The bit fields above are only valid if the respective channel is assigned to the CC by the CCHA bit in the FRSUCC1 register.

*3 The bit fields above are only valid if the respective channel is assigned to the CC by the CCHB bit in the FRSUCC1 register.

- The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.8 FlexRay Symbol Window and NIT Status Register (FRSWNIT)

Bits 7 to 0 reflect symbol window related status information. Updated by the CC at the end of the symbol window for each channel. During startup the status data is not updated.

Bits 11 to 8 reflect NIT related status information. Updated by the CC at the end of the NIT for each channel.

FlexRay Symbol Window and NIT Status Register (FRSWNIT) <P4 address: location H'FFBF F126>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SBNB	SENB	SBNA	SENA	MTSB	MTSA	TCSB	SBSB	SESB	TCSA	SBSA	SESA
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 12	—	All 0	0	N	Reserved Bits These bits are always read as "0".
11	SBNB	0	R	N	Slot Boundary Violation during NIT Channel B Flag (vSS!BViolationB) 0: No slot boundary violation detected 1: Slot boundary violation during NIT detected on channel B
10	SENB	0	R	N	Syntax Error during NIT Channel B Flag (vSS!SyntaxErrorB) 0: No syntax error detected 1: Syntax error during NIT detected on channel B
9	SBNA	0	R	N	Slot Boundary Violation during NIT Channel A Flag (vSS!BViolationA) 0: No slot boundary violation detected 1: Slot boundary violation during NIT detected on channel A
8	SENA	0	R	N	Syntax Error during NIT Channel A Flag (vSS!SyntaxErrorA) 0: No syntax error detected 1: Syntax error during NIT detected on channel A
7	MTSB	0	R	N	MTS Received on Channel B Flag (vSS!ValidMTSB)*2 Media Access Test symbol received on channel B during the preceding symbol window. 0: No MTS symbol received on channel B 1: MTS symbol received on channel B
6	MTSA	0	R	N	MTS Received on Channel A Flag (vSS!ValidMTSA)*1 Media Access Test symbol received on channel A during the preceding symbol window. 0: No MTS symbol received on channel A 1: MTS symbol received on channel A
5	TCSB	0	R	N	Transmission Conflict in Symbol Window Channel B Flag (vSS!TxConflictB) 0: No transmission conflict detected 1: Transmission conflict in symbol window detected on channel B
4	SBSB	0	R	N	Slot Boundary Violation in Symbol Window Channel B Flag (vSS!BViolationB) 0: No slot boundary violation detected 1: Slot boundary violation during symbol window detected on channel B

Bit	Abbreviation	After Reset	R	W	Description
3	SESB	0	R	N	Syntax Error in Symbol Window Channel B Flag (vSS!SyntaxErrorB) 0: No syntax error detected 1: Syntax error during symbol window detected on channel B
2	TCSA	0	R	N	Transmission Conflict in Symbol Window Channel A Flag (vSS!TxConflictA) 0: No transmission conflict detected 1: Transmission conflict in symbol window detected on channel A
1	SBSA	0	R	N	Slot Boundary Violation in Symbol Window Channel A Flag (vSS!BViolationA) 0: No slot boundary violation detected 1: Slot boundary violation during symbol window detected on channel A
0	SESA	0	R	N	Syntax Error in Symbol Window Channel A Flag (vSS!SyntaxErrorA) 0: No syntax error detected 1: Syntax error during symbol window detected on channel A

Notes: *1 When this bit is set to "1", also interrupt flag (the MTSA bit in the FRSIR register) is set to "1".

*2 When this bit is set to "1", also interrupt flag (the MTSB bit in the FRSIR register) is set to "1".

- The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.9 FlexRay Aggregated Channel Status Register (FRACS)

The aggregated channel status register provides the CPU with an accrued status of channel activity for all communication slots regardless of whether they are assigned for transmission or subscribed for reception. The aggregated channel status register also includes status data from the symbol window and the network idle time. The status data is updated (set to "1") after each slot and aggregated until it is set to "0" by the program. During startup the status data is not updated. A flag is set to "0" by writing a "1" to the corresponding bit position. Writing a "0" has no effect on the flag. The register is reset when the CC leaves CONFIG state or enters STARTUP state.

FlexRay Aggregated Channel Status Register (FRACS) <P4 address: location H'FFBF F12A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SBVB	CIB	CEDB	SEDB	VFRB	—	—	—	SBVA	CIA	CEDA	SEDA	VFRA
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12	SBVB	0	R	W	Slot Boundary Violation on Channel B Flag (vSS!BViolationB)*1 One or more slot boundary violations were observed on channel B at any time during the observation period (static or dynamic slots, symbol window, and NIT). 0: No slot boundary violation observed 1: Slot boundary violation(s) observed on channel B
11	CIB	0	R	W	Communication Indicator Channel B Flag*1*2 One or more valid frames were received on channel B in slots that also contained any additional communication during the observation period, i.e. one or more slots received a valid frame AND had any combination of either syntax error OR content error OR content error OR slot boundary violation. 0: No valid frame(s) received in slots containing any additional communication 1: Valid frame(s) received on channel B in slots containing any additional communication
10	CEDB	0	R	W	Content Error Detected on Channel B Flag (vSS!ContentErrorB)*1 One or more frames with a content error were received on channel B in any static or dynamic slot during the observation period. 0: No frame with content error received 1: Frame(s) with content error received on channel B
9	SEDB	0	R	W	Syntax Error Detected on Channel B Flag (vSS!SyntaxErrorB)*1 One or more syntax errors in static or dynamic slots, symbol window, and NIT were observed on channel B. 0: No syntax error observed 1: Syntax error(s) observed on channel B
8	VFRB	0	R	W	Valid Frame Received on Channel B Flag (vSS!ValidFrameB) One or more valid frames were received on channel B in any static or dynamic slot during the observation period. Set to "0" under control of the CPU. 0: No valid frame received 1: Valid frame(s) received on channel B

Bit	Abbreviation	After Reset	R	W	Description
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
4	SBVA	0	R	W	Slot Boundary Violation on Channel A Flag (vSS!BViolationA)*1 One or more slot boundary violations were observed on channel A at any time during the observation period (static or dynamic slots, symbol window, and NIT). 0: No slot boundary violation observed 1: Slot boundary violation(s) observed on channel A
3	CIA	0	R	W	Communication Indicator Channel A Flag*1*2 One or more valid frames were received on channel A in slots that also contained any additional communication during the observation period, i.e. one or more slots received a valid frame AND had any combination of either syntax error OR content error OR slot boundary violation. 0: No valid frame(s) received in slots containing any additional communication 1: Valid frame(s) received on channel A in slots containing any additional communication
2	CEDA	0	R	W	Content Error Detected on Channel A Flag (vSS!ContentErrorA)*1 One or more frames with a content error were received on channel A in any static or dynamic slot during the observation period. 0: No frame with content error received 1: Frame(s) with content error received on channel A
1	SEDA	0	R	W	Syntax Error Detected on Channel A Flag (vSS!SyntaxErrorA)*1 One or more syntax errors in static or dynamic slots, symbol window, and NIT were observed on channel A. 0: No syntax error observed 1: Syntax error(s) observed on channel A
0	VFRA	0	R	W	Valid Frame Received on Channel A Flag (vSS!ValidFrameA) One or more valid frames were received on channel A in any static or dynamic slot during the observation period. 0: No valid frame received 1: Valid frame(s) received on channel A

Notes: *1 When one of the flags SEDA, CEDA, CIA, SBVA changes from 0 to "1", interrupt flag (the EDA bit in the FREIR register) is set to "1". When one of the flags SEDB, CEDB, CIB, SBVB changes from 0 to "1", interrupt flag (the EDB bit in the FREIR register) is set to "1".

*2 The set condition of flags CIA and CIB is also fulfilled if there is only one single frame in the slot and the slot boundary at the end of the slot is reached during the frames channel idle recognition phase.

- The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.10 FlexRay Even Sync ID i Register (FRESIDi) (i = 1 to 15)

Registers FRESID1 to FRESID15 hold the frame IDs of the sync frames received in even communication cycles used for clock synchronisation up to the limit of gSyncNodeMax. The values are sorted in ascending order, with the FRESID1 register holding the lowest received sync frame ID. If the node itself transmits a sync frame in an even communication cycle, the FRESID1 register holds the respective sync frame ID as configured in message buffer 0 and flags RXEA, RXEB are set to "1". The value is updated during the NIT of each even communication cycle.

FlexRay Even Sync ID 1 Register (FRESID1)	<P4 address: location H'FFBF F132>
FlexRay Even Sync ID 2 Register (FRESID2)	<P4 address: location H'FFBF F136>
FlexRay Even Sync ID 3 Register (FRESID3)	<P4 address: location H'FFBF F13A>
:	:
FlexRay Even Sync ID 13 Register (FRESID13)	<P4 address: location H'FFBF F162>
FlexRay Even Sync ID 14 Register (FRESID14)	<P4 address: location H'FFBF F166>
FlexRay Even Sync ID 15 Register (FRESID15)	<P4 address: location H'FFBF F16A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXEB	RXEA	—	—	—	—	EID9	EID8	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	RXEB	0	R	N	Even Sync ID Signals that a sync frame corresponding to the stored even sync ID was received on channel B or that the node is configured to be a sync node with key slot = bits EID9 to EID0 (the FRESID1 register only). 0: No sync frame received on channel B / node not configured to transmit sync frames 1: Sync frame received on channel B/ node configured to transmit sync frames
14	RXEA	0	R	N	Received / Configured Even Sync ID on Channel A Flag Signals that a sync frame corresponding to the stored even sync ID was received on channel A or that the node is configured to be a sync node with key slot = bits EID9 to EID0 (the FRESID1 register only). 0: No sync frame received on channel A / node not configured to transmit sync frames 1: Sync frame received on channel A / node configured to transmit sync frames
13 to 10	—	All 0	0	N	Reserved Bits These bits are always read as "0".
9 to 0	EID9 to EID0	All 0	R	N	Even Sync ID (vsSyncIDListA,B even)*[1] Sync frame ID even communication cycle.

Notes: *1 Sync frames are limited to the static segment. The maximum number of the static slots is 1023.

- The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.11 FlexRay Odd Sync ID i Register (FROSIDi) (i = 1 to 15)

Registers FROSID1 to FROSID15 hold the frame IDs of the sync frames received in odd communication cycles used for clock synchronisation up to the limit of gSyncNodeMax. The values are sorted in ascending order, with the FROSID1 register holding the lowest received sync frame ID. If the node itself transmits a sync frame in an odd communication cycle, the FROSID1 register holds the respective sync frame ID as configured in message buffer 0 and flags RXOA, RXOB are set to "1". The value is updated during the NIT of each odd communication cycle.

FlexRay Odd Sync ID 1 Register (FROSID1)	<P4 address: location H'FFBF F172>
FlexRay Odd Sync ID 2 Register (FROSID2)	<P4 address: location H'FFBF F176>
FlexRay Odd Sync ID 3 Register (FROSID3)	<P4 address: location H'FFBF F17A>
:	:
FlexRay Odd Sync ID 13 Register (FROSID13)	<P4 address: location H'FFBF F1A2>
FlexRay Odd Sync ID 14 Register (FROSID14)	<P4 address: location H'FFBF F1A6>
FlexRay Odd Sync ID 15 Register (FROSID15)	<P4 address: location H'FFBF F1AA>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXOB	RXOA	—	—	—	—	OID9	OID8	OID7	OID6	OID5	OID4	OID3	OID2	OID1	OID0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15	RXOB	0	R	N	Received/Configured Odd Sync ID on Channel B Flag Signals that a sync frame corresponding to the stored odd sync ID was received on channel B or that the node is configured to be a sync node with key slot = bits OID9 to OID0 (the FROSID1 register only). 0: No sync frame received on channel B / node not configured to transmit sync frames 1: Sync frame received on channel B/ node configured to transmit sync frames
14	RXOA	0	R	N	Received/Configured Odd Sync ID on Channel A Flag Signals that a sync frame corresponding to the stored odd sync ID was received on channel A or that the node is configured to be a sync node with key slot = bits OID9 to OID0 (the FROSID1 register only). 0: No sync frame received on channel A / node not configured to transmit sync frames 1: Sync frame received on channel A/ node configured to transmit sync frames
13 to 10	—	All 0	0	N	Reserved Bits These bits are always read as "0".
9 to 0	OID9 to OID0	All 0	R	N	Odd Sync ID (vsSyncIDListA,B odd)*1 Sync frame ID odd communication cycle.

Notes: *1 Sync frames are limited to the static segment. The maximum number of the static slots is 1023.

- The register is reset when the CC leaves CONFIG state or enters STARTUP state.

32.7.12 FlexRay Network Management Vector i Register (FRNMVi) (i = 1 to 3)

The three network management registers hold the accrued NM vector (configurable 0 to 12 bytes).

The accrued NM vector is generated by the CC by bit-wise ORing each NM vector received (valid static frames with PPI = 1) on each channel (see section 32.17, Network Management).

The CC updates the NM vector at the end of each communication cycle as long as the CC is either in NORMAL_ACTIVE or NORMAL_PASSIVE state.

NMVn-bytes exceeding the configured NM vector length are not valid.

FlexRay Network Management Vector 1 Register (FRNMV1)	<P4 address: location H'FFBF F1B0>
FlexRay Network Management Vector 2 Register (FRNMV2)	<P4 address: location H'FFBF F1B4>
FlexRay Network Management Vector 3 Register (FRNMV3)	<P4 address: location H'FFBF F1B8>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NM31	NM30	NM29	NM28	NM27	NM26	NM25	NM24	NM23	NM22	NM21	NM20	NM19	NM18	NM17	NM16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NM15	NM14	NM13	NM12	NM11	NM10	NM9	NM8	NM7	NM6	NM5	NM4	NM3	NM2	NM1	NM0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	NM31 to NM24	All 0	R	N	The three network management registers hold the accrued NM vector (configurable 0 to 12 bytes).*[1] The accrued NM vector is generated by the CC by bit-wise ORing each NM vector received (valid static frames with PPI = 1) on each channel.
23 to 16	NM23 to NM16	All 0	R	N	
15 to 8	NM15 to NM8	All 0	R	N	
7 to 0	NM7 to NM0	All 0	R	N	

Note: *1 The following shows the assignment of the received payload's data bytes to the network management vector.

Assignment of Data Bytes to Network Management Vector

Register	NM31 to NM24	NM23 to NM16	NM15 to NM8	NM7 to NM0
FRNMV1	Data3	Data2	Data1	Data0
FRNMV2	Data7	Data6	Data5	Data4
FRNMV3	Data11	Data10	Data9	Data8

By setting the FBSEN bit in the FXROC register, the bit order on reading this register can be selected to be little endian style or big endian style.

- When the FBSEN bit in the FXROC register is set to "1" (big endian style)

Register	NM31 to NM24	NM23 to NM16	NM15 to NM8	NM7 to NM0
FRNMV1	Data0	Data1	Data2	Data3
FRNMV2	Data4	Data5	Data6	Data7
FRNMV3	Data8	Data9	Data10	Data11

- When the FBSEN bit in the FXROC register is set to "0" (little endian style)

Register	NM31 to NM24	NM23 to NM16	NM15 to NM8	NM7 to NM0
FRNMV1	Data3	Data2	Data1	Data0
FRNMV2	Data7	Data6	Data5	Data4
FRNMV3	Data11	Data10	Data9	Data8

32.8 Message Buffer Control Registers

32.8.1 FlexRay Message RAM Configuration Register (FRMRC)

The Message RAM Configuration register defines the number of message buffers assigned to the static segment, dynamic segment, and FIFO.

FlexRay Message RAM Configuration Register (FRMRC) <P4 address: location H'FFBF F300>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SPLM	SEC1	SEC0	LCB7	LCB6	LCB5	LCB4	LCB3	LCB2	LCB1	LCB0
After Reset:	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FFB7	FFB6	FFB5	FFB4	FFB3	FFB2	FFB1	FFB0	FDB7	FDB6	FDB5	FDB4	FDB3	FDB2	FDB1	FDB0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0180 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
26	SPLM	0	R	W	Sync Frame Payload Multiplex Bit This bit is only evaluated if the node is configured as sync node (the TXSY bit in the FRSUCC1 register = 1) or for single slot mode operation (the TSM bit in the FRSUCC1 register = 1). When this bit is set to "1" message buffers 0 and 1 are dedicated for sync frame transmission with different payload data on channels A and B. When this bit is set to "0", sync frames are transmitted from message buffer 0 with the same payload data on both channels. Note that the channel filter configuration for message buffer 0 resp. message buffer 1 has to be chosen accordingly. 0: Only message buffer 0 locked against reconfiguration 1: Both message buffers 0 and 1 are locked against reconfiguration Note: • In case the node is configured as sync node (the TXSY bit in the FRSUCC1 register = 1) or for single slot mode operation (the TSM bit in the FRSUCC1 register = 1), message buffer 0 resp. 1 is reserved for sync frames or single slot frames and have to be configured with the node-specific key slot ID. In case the node is neither configured as sync node nor for single slot operation message buffer 0 resp. 1 is treated like all other message buffers.
25	SEC1	0	R	W	Secure Buffers Bit*[1]
24	SEC0	1	R	W	00: Reconfiguration of message buffers enabled with numbers < FFB bit enabled*[2] 01: Reconfiguration of message buffers with numbers < FDB and with numbers ≥ FFB bit locked and transmission of message buffers for static segment with numbers ≥ FDB bit disabled 10: Reconfiguration of all message buffers locked 11: Reconfiguration of all message buffers locked and transmission of message buffers for static segment with numbers ≥ FDB bit disabled
23 to 16	LCB7 to LCB0	H'80	R	W	Last Configured Buffer Bit 0 to 127: Number of message buffers is LCB + 1 ≥ 128: No message buffer configured

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	FFB7 to FFB0	All 0	R	W	First Buffer of FIFO Bit 0: All message buffers assigned to the FIFO 1 to 127: Message buffers from FFB bit to LCB bit assigned to the FIFO ≥ 128: No message buffer assigned to the FIFO
7 to 0	FDB7 to FDB0	All 0	R	W	First Dynamic Buffer Bit 0: No group of message buffers exclusively for the static segment configured 1 to 127: Message buffers 0 to FDB bit - 1 reserved for static segment ≥ 128: No dynamic message buffers configured

Notes: *1 Not evaluated when the CC is in DEFAULT_CONFIG or CONFIG state.

*2 n nodes configured for sync frame transmission or for single slot mode operation message buffer 0 (and if the SPLM bit = 1, also message buffer 1) is always locked.

- In case the node is configured as sync node (the TXSY bit in the FRSUCC1 register = 1) or for single slot mode operation (the TSM bit in the FRSUCC1 register = 1), message buffer 0 resp. 1 is reserved for sync frames or single slot frames and have to be configured with the node-specific key slot ID. In case the node is neither configured as sync node nor for single slot operation message buffer 0 resp. 1 is treated like all other message buffers.
- The register can be written during CONFIG state only.

Figure 32.3 shows an example of the message buffer structure configured by bits FDB, FFB, and LCB.

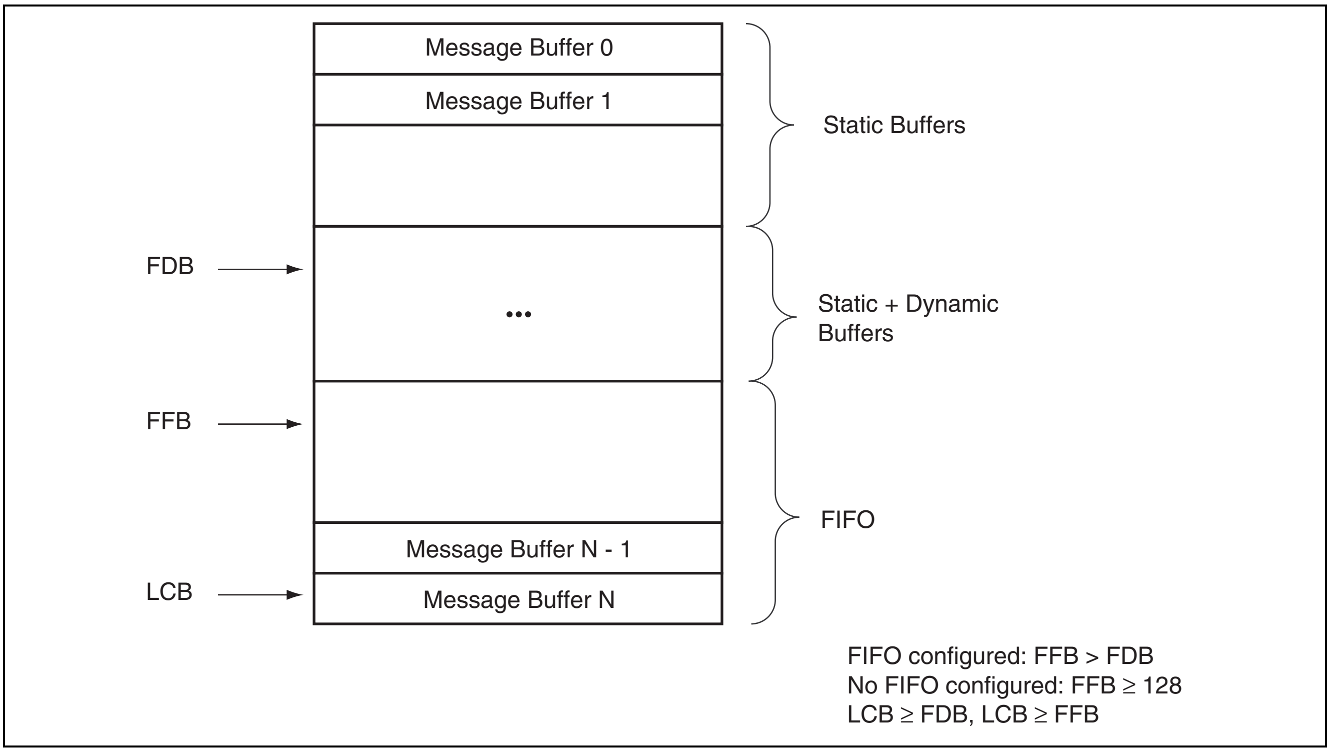


Figure 32.3 Message Buffer Structure

Notes:
- The programmer has to ensure that the configuration defined by bits FDB7 to FDB0, FFB7 to FFB0, and LCB7 to LCB0 is valid. The CC does not check for erroneous configurations!
- The maximum number of header sections is 128. This means a maximum of 128 message buffers can be configured. The maximum length of a data section is 254 bytes. The length of the data section may be configured differently for each message buffer. For details, see section 32.23, Message RAM.
- In case two or more message buffers are assigned to slot 1 by use of cycle filtering, all of them must be located either in the “Static Buffers” or at the beginning of the “Static + Dynamic Buffers” section.
- The payload length configured and the length of the data section need to be configured identical for all message buffers belonging to the FIFO via bits PLC6 to PLC0 in the FRWRHS2 register and bits DP10 to DP0 in the FRWRHS3 register. When the CC is not in CONFIG state reconfiguration of message buffers belonging to the FIFO is locked.
- The FlexRay protocol specification requires that each node has to send a frame in its key slot. Therefore at least message buffer 0 is reserved for transmission in the key slot. Due to this requirement a maximum number of 127 message buffers can be assigned to the FIFO. Nevertheless, a non protocol conform configuration without a transmission slot in the static segment would still be operational.

32.8.2 FlexRay FIFO Rejection Filter Register (FRFRF)

The FIFO Rejection Filter defines a user specified sequence of bits to which channel, frame ID, and cycle count of the incoming frames are compared. Together with the FIFO Rejection Filter Mask this register determines whether a message is rejected by the FIFO.

FlexRay FIFO Rejection Filter Register (FRFRF) <P4 address: location H'FFBF F304>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	RNF	RSS	CYF6	CYF5	CYF4	CYF3	CYF2	CYF1	CYF0
After Reset:	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FID10	FID9	FID8	FID7	FID6	FID5	FID4	FID3	FID2	FID1	FID0	CH1	CH0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0180 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 25	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
24	RNF	1	R	W	Reject Null Frames Bit 0: Null frames are stored in the FIFO 1: Reject all null frames
23	RSS	1	R	W	Reject in Static Segment Bit 0: FIFO also used for static segment 1: Reject messages in static segment
22 to 16	CYF6 to CYF0	All 0	R	W	Cycle Counter Filter Bit The 7-bit cycle counter filter determines the cycle set to which frame ID and channel rejection filter are applied. In cycles not belonging to the cycle set specified by bits CYF6 to CYF0, all frames are rejected. For details about the configuration of the cycle counter filter, see section 32.18.2, Cycle Counter Filtering.
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12 to 2	FID10 to FID0	All 0	R	W	Frame ID Filter Bit*2*3 Determines the frame ID to be rejected by the FIFO. 0 to 2047 = Frame ID filter values
1	CH1	0	R	W	Channel Filter Bit*1
0	CH0	0	R	W	00: Receive on both channels 01: Receive only on channel B 10: Receive only on channel A 11: No reception

Notes: *1 If reception on both channels is configured, also in static segment always both frames (from channel A and B) are stored in the FIFO, even if they are identical.

*2 With the additional configuration of the FRFRFM register, the corresponding frame ID filter bits are ignored, which results in further rejected frame IDs.

*3 When bits MFID10 to MFID0 in the FRFRFM register are "0", a frame ID filter value of zero means that no frame ID is rejected.

- The FRFRF register can be written during CONFIG state only.

32.8.3 FlexRay FIFO Rejection Filter Mask Register (FRFRFM)

The FIFO Rejection Filter Mask specifies which of the corresponding frame ID filter bits are relevant for rejection filtering. If a bit is set to "1", it indicates that the corresponding bit in the FRFRF register will not be considered for rejection filtering.

FlexRay FIFO Rejection Filter Mask Register (FRFRFM) <P4 address: location H'FFBF F30A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MFID10	MFID9	MFID8	MFID7	MFID6	MFID5	MFID4	MFID3	MFID2	MFID1	MFID0	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 13	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
12 to 2	MFID10 to MFID0	All 0	R	W	Mask Frame ID Filter Bit 0: Corresponding frame ID filter bit is used for rejection filtering 1: Ignore corresponding frame ID filter bit
1, 0	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Note: • The FRFRFM register can be written during CONFIG state only.

32.8.4 FlexRay FIFO Critical Level Register (FRFCL)

FlexRay FIFO Critical Level Register (FRFCL) <P4 address: location H'FFBF F30F>

Bit:	7	6	5	4	3	2	1	0
	CL7	CL6	CL5	CL4	CL3	CL2	CL1	CL0
After Reset:	1	0	0	0	0	0	0	0

<After Reset: H'80>

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	CL7 to CL0	H'80	R	W	Critical Level Bit When the receive FIFO fill level (bits RFFL7 to RFFL0 in the FRFSR register) is equal or greater than the critical level configured by bits CL7 to CL0, the receive FIFO critical level flag (the RFCL bit in the FRFSR register) is set to "1". If bits CL7 to CL0 are programmed to values > 128, the RFCL bit in the FRFSR register is never set to "1". When the RFCL bit in the FRFSR register changes from 0 to 1 the RFCL bit in the FRSIR register is set to "1", and if enabled, an interrupt is generated.

Note: • The CC accepts modifications of the register in CONFIG state only.

32.9 Message Buffer Status Registers

32.9.1 FlexRay Message Handler Status Register (FRMHDS)

A flag is set to "0" by writing a "1" to the corresponding bit position. Writing a "0" has no effect on the flag. The register will also be set to "0" by hard reset or by CHI command CLEAR_RAMS.

FlexRay Message Handler Status Register (FRMHDS) <P4 address: location H'FFBF F310>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MBU6	MBU5	MBU4	MBU3	MBU2	MBU1	MBU0	—	MBT6	MBT5	MBT4	MBT3	MBT2	MBT1	MBT0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	FMB6	FMB5	FMB4	FMB3	FMB2	FMB1	FMB0	CRAM	MFMB	FMBD	PTBF2	PTBF1	PMR	POBF	PIBF
After Reset:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

<After Reset: H'0000 0080>

Bit	Abbreviation	After Reset	R	W	Description
31	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
30 to 24	MBU6 to MBU0	All 0	R	0	Message Buffer Updated*3 Number of message buffer that was updated last by the CC.*6
23	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
22 to 16	MBT6 to MBT0	All 0	R	0	Message Buffer Transmitted*4 Number of last successfully transmitted message buffer.*5
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
14 to 8	FMB6 to FMB0	All 0	R	0	Faulty Message Buffer Parity error occurred when reading from the message buffer or when transferring data from Input Buffer or Transient Buffer 1,2 to the message buffer referenced by bits FMB6 to FMB0.*2*3
7	CRAM	1	R	0	Clear all internal RAM's Flag Signals that execution of the CHI command CLEAR_RAMS is ongoing (all bits of all internal RAM blocks are written to "0"). The bit is set to "1" by hard reset or by CHI command CLEAR_RAMS. 0: No execution of the CHI command CLEAR_RAMS 1: Execution of the CHI command CLEAR_RAMS ongoing
6	MFMB	0	R	W	Multiple Faulty Message Buffers detected Flag 0: No additional faulty message buffer 1: Another faulty message buffer was detected while the FMBD flag is set to "1"
5	FMBD	0	R	W	Faulty Message Buffer Detected Flag Message buffer referenced by bits FMB6 to FMB0 holds faulty data due to a parity error. 0: No faulty message buffer 1: Message buffer referenced by bits FMB6 to FMB0 holds faulty data due to a parity error

Bit	Abbreviation	After Reset	R	W	Description
4	PTBF2	0	R	W	Parity Error Transient Buffer RAM B Flag 0: No parity error 1: Parity error occurred when reading Transient Buffer RAM B*[1]
3	PTBF1	0	R	W	Parity Error Transient Buffer RAM A Flag 0: No parity error 1: Parity error occurred when reading Transient Buffer RAM A*[1]
2	PMR	0	R	W	Parity Error Message RAM Flag 0: No parity error 1: Parity error occurred when reading the Message RAM*[1]
1	POBF	0	R	W	Parity Error Output Buffer RAM 1, 2 Flag 0: No parity error 1: Parity error occurred when reading Output Buffer RAM 1, 2*[1]
0	PIBF	0	R	W	Parity Error Input Buffer RAM 1, 2 Flag 0: No parity error 1: Parity error occurred when reading Input Buffer RAM 1, 2*[1]

Notes: *1 When one of the flags PIBF, POBF, PMR, PTBF1, PTBF2 changes from 0 to 1, the PERR bit in the FREIR register is set to "1".

*2 Value only valid when one of the flags PIBF, PMR, PTBF1, PTBF2, and the FMBD bit is set to "1".

*3 Is not updated while flag FMBD is set.

*4 Bits MBT6 to MBT0 and MBU6 to MBU0 are set to "0" when the CC leaves CONFIG state or enters STARTUP state.

*5 f the message buffer is configured for singleshot mode, the respective TXR flag in the FRTXRQi register (i = 1 to 4) was set to "0".

*6 For this message buffer the respective ND and/or MBC flag in the FRNDATi register (i = 1 to 4) and the FRMBSCi register (i = 1 to 4) are also set to "1".

32.9.2 FlexRay Last Dynamic Transmit Slot Register (FRLDTS)

FlexRay Last Dynamic Transmit Slot Register (FRLDTS) <P4 address: location H'FFBF F314>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	LDTB10	LDTB9	LDTB8	LDTB7	LDTB6	LDTB5	LDTB4	LDTB3	LDTB2	LDTB1	LDTB0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	LDTA10	LDTA9	LDTA8	LDTA7	LDTA6	LDTA5	LDTA4	LDTA3	LDTA2	LDTA1	LDTA0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 27	—	All 0	0	N	Reserved Bits These bits are always read as "0".
26 to 16	LDTB10 to LDTB0	All 0	R	N	Last Dynamic Transmission Channel B*[1] Value of vSlotCounter[B] at the time of the last frame transmission on channel B in the dynamic segment of this node.
15 to 11	—	All 0	0	N	Reserved Bits These bits are always read as "0".
10 to 0	LDTA10 to LDTA0	All 0	R	N	Last Dynamic Transmission Channel A*[1] Value of vSlotCounter[A] at the time of the last frame transmission on channel A in the dynamic segment of this node.

Notes: *1 It is updated at the end of the dynamic segment and is set to "0" if no frame was transmitted during the dynamic segment.

- The register is initialized when the CC leaves CONFIG state or enters STARTUP state.

32.9.3 FlexRay FIFO Status Register (FRFSR)

FlexRay FIFO Status Register (FRFSR) <P4 address: location H'FFBF F31A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFFL7	RFFL6	RFFL5	RFFL4	RFFL3	RFFL2	RFFL1	RFFL0	—	—	—	—	—	RFO	RFCL	RFNE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	RFFL7 to RFFL0	All 0	R	N	Receive FIFO Fill Level Number of FIFO buffers filled with new data not yet read by the CPU. Maximum value is 128.
7 to 3	—	All 0	0	N	Reserved Bits These bits are always read as "0".
2	RFO	0	R	N	Receive FIFO Overrun Flag The flag is set to "1" by the CC when a receive FIFO overrun is detected. When a receive FIFO overrun occurs, the oldest message is overwritten with the actual received message. In addition, interrupt flag (the RFO bit in the FREIR register) is set to "1". The flag is set to "0" by the next FIFO read access issued by the CPU. 0: No receive FIFO overrun detected 1: A receive FIFO overrun has been detected
1	RFCL	0	R	N	Receive FIFO Critical Level Flag This flag is set to "1" when the receive FIFO fill level (bits RFFL7 to RFF0) is equal or greater than the critical level as configured by bits CL7 to CL0 in the FRFCL register. The flag is set to "0" by the CC as soon as bits RFFL7 to RFFL0 drop below bits CL7 to CL0 in the FRFCL register. When the RFCL bit changes from 0 to 1 the RFCL bit in the FRSIR register is set to "1", and if enabled, an interrupt is generated. 0: Receive FIFO below critical level 1: Receive FIFO critical level reached
0	RFNE	0	R	N	Receive FIFO Not Empty Flag This flag is set to "1" by the CC when a received valid frame (data or null frame depending on rejection mask) was stored in the FIFO. In addition, interrupt flag (the RFNE flag in the FRSIR register) is set to "1". The bit is set to "0" after the CPU has read all message from the FIFO. 0: Receive FIFO is empty 1: Receive FIFO is not empty

Note: • The register is initialized when the CC leaves CONFIG state or enters STARTUP state.

32.9.4 FlexRay Message Handler Constraints Flags Register (FRMHDF)

Some constraints exist for the Message Handler regarding PAck frequency, Message RAM configuration, and FlexRay bus traffic. To simplify software development, constraints violations are reported by setting flags in the FRMHDF register.

A flag is set to "0" by writing a "1" to the corresponding bit position. Writing a "0" has no effect on the flag. An initialisation of the FlexRay module will also set the register to "0".

FlexRay Message Handler Constraints Flags Register (FRMHDF) <P4 address: location H'FFBF F31E>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WAHP	—	—	TBFB	TBFA	FNFB	FNFA	SNUB	SNUA
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 9	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
8	WAHP	0	R	W	Write Attempt to Header Partition Flag*[1] Outside DEFAULT_CONFIG and CONFIG state this flag is set to "1" by the CC when the message handler tries to write message data into the header partition of the Message RAM due to faulty configuration of a message buffer. The write attempt is not executed, to protect the header partition from unintended write accesses. 0: No write attempt to header partition 1: Write attempt to header partition
7, 6	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
5	TBFB	0	R	W	Transient Buffer Access Failure B Flag*[1] This flag is set to "1" by the CC when a read or write access to TBF B requested by PRT B could not complete within the available time. 0: No TBF B access failure 1: TBF B access failure
4	TBFA	0	R	W	Transient Buffer Access Failure A Flag*[1] This flag is set to "1" by the CC when a read or write access to TBF A requested by PRT A could not complete within the available time. 0: No TBF A access failure 1: TBF A access failure
3	FNFB	0	R	W	Find Sequence Not Finished for Channel B Flag*[1]*[2] This flag is set to "1" by the CC when the Message Handler, due to overload condition, was not able to finish a find sequence with respect to channel B. 0: No find sequence not finished for channel B 1: Find sequence not finished for channel B
2	FNFA	0	R	W	Find Sequence Not Finished for Channel A Flag*[1]*[2] This flag is set to "1" by the CC when the Message Handler, due to overload condition, was not able to finish a find sequence with respect to channel A. 0: No find sequence not finished for channel A 1: Find sequence not finished for channel A

Bit	Abbreviation	After Reset	R	W	Description
1	SNUB	0	R	W	Status Not Updated Channel B Flag*[1] This flag is set to "1" by the CC when the Message Handler, due to overload condition, was not able to update a message buffer's status (MBS) with respect to channel B. 0: No overload condition occurred when updating MBS for channel B 1: MBS for channel A not updated
0	SNUA	0	R	W	Status Not Updated Channel A Flag*[1] This flag is set to "1" by the CC when the Message Handler, due to overload condition, was not able to update a message buffer's status (MBS) with respect to channel A. 0: No overload condition occurred when updating MBS for channel A 1: MBS for channel A not updated

Notes: *1 When one of the flags SNUA, SNUB, FNFA, FNFB, TBFA, TBFB, WAHP changes from 0 to 1, interrupt flag (the MHF bit in the FREIR register) is set to "1".

*2 Sequence: scan of Message RAM for matching message buffer

- The register is initialized when the CC leaves CONFIG state or enters STARTUP state.

32.9.5 FlexRay Transmission Request Register i (FRTXRQi) (i = 1 to 4)

The four registers reflect the state of the TXR flags of all configured message buffers. The flags are evaluated for transmit buffers only. If the number of configured message buffers is less than 128, the remaining TXR flags have no meaning.

(1) FlexRay Transmission Request Register 1 (FRTXRQ1)

FlexRay Transmission Request Register 1 (FRTXRQ1) <P4 address: location H'FFBF F320>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TXR 31	TXR 30	TXR 29	TXR 28	TXR 27	TXR 26	TXR 25	TXR 24	TXR 23	TXR 22	TXR 21	TXR 20	TXR 19	TXR 18	TXR 17	TXR 16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXR 15	TXR 14	TXR 13	TXR 12	TXR 11	TXR 10	TXR 9	TXR 8	TXR 7	TXR 6	TXR 5	TXR 4	TXR 3	TXR 2	TXR 1	TXR 0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TXR31 to TXR0	All 0	R	N	Transmission Request Flag*[1] If the flag is set to "1", the respective message buffer is ready for transmission respectively transmission of this message buffer is in progress. In single-shot mode the flags are reset after transmission has completed.

Note: *1 If the number of configured message buffers is less than 128, the remaining TXR flags have no meaning.

(2) FlexRay Transmission Request Register 2 (FRTXRQ2)

FlexRay Transmission Request Register 2 (FRTXRQ2) <P4 address: location H'FFBF F324>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TXR 63	TXR 62	TXR 61	TXR 60	TXR 59	TXR 58	TXR 57	TXR 56	TXR 55	TXR 54	TXR 53	TXR 52	TXR 51	TXR 50	TXR 49	TXR 48
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXR 47	TXR 46	TXR 45	TXR 44	TXR 43	TXR 42	TXR 41	TXR 40	TXR 39	TXR 38	TXR 37	TXR 36	TXR 35	TXR 34	TXR 33	TXR 32
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TXR63 to TXR32	All 0	R	N	Transmission Request Flag*[1] If the flag is set to "1", the respective message buffer is ready for transmission respectively transmission of this message buffer is in progress. In single-shot mode the flags are reset after transmission has completed.

Note: *1 If the number of configured message buffers is less than 128, the remaining TXR flags have no meaning.

(3) FlexRay Transmission Request Register 3 (FRTXRQ3)

FlexRay Transmission Request Register 3 (FRTXRQ3) <P4 address: location H'FFBF F328>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TXR 95	TXR 94	TXR 93	TXR 92	TXR 91	TXR 90	TXR 89	TXR 88	TXR 87	TXR 86	TXR 85	TXR 84	TXR 83	TXR 82	TXR 81	TXR 80
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXR 79	TXR 78	TXR 77	TXR 76	TXR 75	TXR 74	TXR 73	TXR 72	TXR 71	TXR 70	TXR 69	TXR 68	TXR 67	TXR 66	TXR 65	TXR 64
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TXR95 to TXR64	All 0	R	N	Transmission Request Flag*[1] If the flag is set to "1", the respective message buffer is ready for transmission respectively transmission of this message buffer is in progress. In single-shot mode the flags are reset after transmission has completed.

Note: *1 If the number of configured message buffers is less than 128, the remaining TXR flags have no meaning.

(4) FlexRay Transmission Request Register 4 (FRTXRQ4)

FlexRay Transmission Request Register 4 (FRTXRQ4) <P4 address: location H'FFBF F32C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TXR 127	TXR 126	TXR 125	TXR 124	TXR 123	TXR 122	TXR 121	TXR 120	TXR 119	TXR 118	TXR 117	TXR 116	TXR 115	TXR 114	TXR 113	TXR 112
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXR 111	TXR 110	TXR 109	TXR 108	TXR 107	TXR 106	TXR 105	TXR 104	TXR 103	TXR 102	TXR 101	TXR 100	TXR 99	TXR 98	TXR 97	TXR 96
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	TXR127 to TXR96	All 0	R	N	Transmission Request Flag*[1] If the flag is set to "1", the respective message buffer is ready for transmission respectively transmission of this message buffer is in progress. In single-shot mode the flags are reset after transmission has completed.

Note: *1 If the number of configured message buffers is less than 128, the remaining TXR flags have no meaning.

32.9.6 FlexRay New Data Register i (FRNDATi) (i = 1 to 4)

The four registers reflect the state of the ND flags of all configured message buffers. ND flags belonging to transmit buffers have no meaning. If the number of configured message buffers is less than 128, the remaining ND flags have no meaning.

(1) FlexRay New Data Register 1 (FRNDAT1)

FlexRay New Data Register 1 (FRNDAT1) <P4 address: location H'FFBF F330>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ND 31	ND 30	ND 29	ND 28	ND 27	ND 26	ND 25	ND 24	ND 23	ND 22	ND 21	ND 20	ND 19	ND 18	ND 17	ND 16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ND 15	ND 14	ND 13	ND 12	ND 11	ND 10	ND 9	ND 8	ND 7	ND 6	ND 5	ND 4	ND 3	ND 2	ND 1	ND 0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	ND31 to ND0	All 0	R	N	New Data Flag*[1] The flags are set to "1" when a valid received data frame matches the message buffer’s filter configuration, independent of the payload length received or the payload length configured for that message buffer. The flags are not set to "1" after reception of null frames except for message buffers belonging to the receive FIFO. An ND flag is set to "0" when the header section of the corresponding message buffer is reconfigured or when the data section has been transferred to the Output Buffer.

Notes: *1 If the number of configured message buffers is less than 128, the remaining ND flags have no meaning. ND flags belonging to transmit buffers have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

(2) FlexRay New Data Register 2 (FRNDAT2)

FlexRay New Data Register 2 (FRNDAT2) <P4 address: location H'FFBF F334>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ND63	ND62	ND61	ND60	ND59	ND58	ND57	ND56	ND55	ND54	ND53	ND52	ND51	ND50	ND49	ND48
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ND47	ND46	ND45	ND44	ND43	ND42	ND41	ND40	ND39	ND38	ND37	ND36	ND35	ND34	ND33	ND32
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	ND63 to ND32	All 0	R	N	New Data Flag*[1] The flags are set to "1" when a valid received data frame matches the message buffer's filter configuration, independent of the payload length received or the payload length configured for that message buffer. The flags are not set to "1" after reception of null frames except for message buffers belonging to the receive FIFO. An ND flag is set to "0" when the header section of the corresponding message buffer is reconfigured or when the data section has been transferred to the Output Buffer.

Notes: *1 If the number of configured message buffers is less than 128, the remaining ND flags have no meaning. ND flags belonging to transmit buffers have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

(3) FlexRay New Data Register 3 (FRNDAT3)

FlexRay New Data Register 3 (FRNDAT3) <P4 address: location H'FFBF F338>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ND 95	ND 94	ND 93	ND 92	ND 91	ND 90	ND 89	ND 88	ND 87	ND 86	ND 85	ND 84	ND 83	ND 82	ND 81	ND 80
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ND 79	ND 78	ND 77	ND 76	ND 75	ND 74	ND 73	ND 72	ND 71	ND 70	ND 69	ND 68	ND 67	ND 66	ND 65	ND 64
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	ND95 to ND64	All 0	R	N	New Data Flag*[1] The flags are set to "1" when a valid received data frame matches the message buffer's filter configuration, independent of the payload length received or the payload length configured for that message buffer. The flags are not set to "1" after reception of null frames except for message buffers belonging to the receive FIFO. An ND flag is set to "0" when the header section of the corresponding message buffer is reconfigured or when the data section has been transferred to the Output Buffer.

Notes: *1 If the number of configured message buffers is less than 128, the remaining ND flags have no meaning. ND flags belonging to transmit buffers have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

(4) FlexRay New Data Register 4 (FRNDAT4)

FlexRay New Data Register 4 (FRNDAT4) <P4 address: location H'FFBF F33C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ND 127	ND 126	ND 125	ND 124	ND 123	ND 122	ND 121	ND 120	ND 119	ND 118	ND 117	ND 116	ND 115	ND 114	ND 113	ND 112
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ND 111	ND 110	ND 109	ND 108	ND 107	ND 106	ND 105	ND 104	ND 103	ND 102	ND 101	ND 100	ND 99	ND 98	ND 97	ND 96
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	ND127 to ND96	All 0	R	N	New Data Flag*[1] The flags are set to "1" when a valid received data frame matches the message buffer’s filter configuration, independent of the payload length received or the payload length configured for that message buffer. The flags are not set to "1" after reception of null frames except for message buffers belonging to the receive FIFO. An ND flag is set to "0" when the header section of the corresponding message buffer is reconfigured or when the data section has been transferred to the Output Buffer.

Notes: *1 If the number of configured message buffers is less than 128, the remaining ND flags have no meaning. ND flags belonging to transmit buffers have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

32.9.7 FlexRay Message Buffer Status Changed Register i (FRMBSCi) (i = 1 to 4)

The four registers reflect the state of the MBC flags of all configured message buffers. If the number of configured message buffers is less than 128, the remaining MBC flags have no meaning.

(1) FlexRay Message Buffer Status Changed Register 1 (FRMBSC1)

FlexRay Message Buffer Status Changed Register 1 (FRMBSC1) <P4 address: location H'FFBF F340>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MBC 31	MBC 30	MBC 29	MBC 28	MBC 27	MBC 26	MBC 25	MBC 24	MBC 23	MBC 22	MBC 21	MBC 20	MBC 19	MBC 18	MBC 17	MBC 16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MBC 15	MBC 14	MBC 13	MBC 12	MBC 11	MBC 10	MBC 9	MBC 8	MBC 7	MBC 6	MBC 5	MBC 4	MBC 3	MBC 2	MBC 1	MBC 0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MBC31 to MBC0	All 0	R	N	Message Buffer Status Changed Flag*[1] An MBC flag is set to "1" whenever the Message Handler changes one of the status flags VFRA,VFRB, SEOA, SEOB, CEOA, CEOB, SVOA, SVOB, TCIA, TCIB, ESA, ESB, MLST,FTA, FTB in the header section of the respective message buffer. 0: No change 1: Changed

Notes: *1 If the number of configured message buffers is less than 128, the remaining MBC flags have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

(2) FlexRay Message Buffer Status Changed Register 2 (FRMBSC2)

FlexRay Message Buffer Status Changed Register 2 (FRMBSC2) <P4 address: location H'FFBF F344>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MBC 63	MBC 62	MBC 61	MBC 60	MBC 59	MBC 58	MBC 57	MBC 56	MBC 55	MBC 54	MBC 53	MBC 52	MBC 51	MBC 50	MBC 49	MBC 48
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MBC 47	MBC 46	MBC 45	MBC 44	MBC 43	MBC 42	MBC 41	MBC 40	MBC 39	MBC 38	MBC 37	MBC 36	MBC 35	MBC 34	MBC 33	MBC 32
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MBC63 to MBC32	All 0	R	N	Message Buffer Status Changed Flag*[1] An MBC flag is set to "1" whenever the Message Handler changes one of the status flags VFRA,VFRB, SEOA, SEOB, CEOA, CEOB, SVOA, SVOB, TCIA, TCIB, ESA, ESB, MLST,FTA, FTB in the header section of the respective message buffer. 0: No change 1: Changed

Notes: *1 If the number of configured message buffers is less than 128, the remaining MBC flags have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

(3) FlexRay Message Buffer Status Changed Register 3 (FRMBSC3)

FlexRay Message Buffer Status Changed Register 3 (FRMBSC3) <P4 address: location H'FFBF F348>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MBC 95	MBC 94	MBC 93	MBC 92	MBC 91	MBC 90	MBC 89	MBC 88	MBC 87	MBC 86	MBC 85	MBC 84	MBC 83	MBC 82	MBC 81	MBC 80
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MBC 79	MBC 78	MBC 77	MBC 76	MBC 75	MBC 74	MBC 73	MBC 72	MBC 71	MBC 70	MBC 69	MBC 68	MBC 67	MBC 66	MBC 65	MBC 64
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MBC95 to MBC64	All 0	R	N	Message Buffer Status Changed Flag*[1] An MBC flag is set to "1" whenever the Message Handler changes one of the status flags VFRA,VFRB, SEOA, SEOB, CEOA, CEOB, SVOA, SVOB, TCIA, TCIB, ESA, ESB, MLST,FTA, FTB in the header section of the respective message buffer. 0: No change 1: Changed

Notes: *1 If the number of configured message buffers is less than 128, the remaining MBC flags have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

(4) FlexRay Message Buffer Status Changed Register 4 (FRMBSC4)

FlexRay Message Buffer Status Changed Register 4 (FRMBSC4) <P4 address: location H'FFBF F34C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MBC 127	MBC 126	MBC 125	MBC 124	MBC 123	MBC 122	MBC 121	MBC 120	MBC 119	MBC 118	MBC 117	MBC 116	MBC 115	MBC 114	MBC 113	MBC 112
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MBC 111	MBC 110	MBC 109	MBC 108	MBC 107	MBC 106	MBC 105	MBC 104	MBC 103	MBC 102	MBC 101	MBC 100	MBC 99	MBC 98	MBC 97	MBC 96
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MBC127 to MBC96	All 0	R	N	Message Buffer Status Changed Flag*1 An MBC flag is set to "1" whenever the Message Handler changes one of the status flags VFRA,VFRB, SEOA, SEOB, CEOA, CEOB, SVOA, SVOB, TCIA, TCIB, ESA, ESB, MLST,FTA, FTB in the header section of the respective message buffer. 0: No change 1: Changed

Notes: *1 If the number of configured message buffers is less than 128, the remaining MBC flags have no meaning.

- The registers are initialized when the CC leaves CONFIG state or enters STARTUP state.

32.10 Input Buffer

Double buffer structure consisting of Input Buffer Host and Input Buffer Shadow. While the CPU can write to Input Buffer Host, the transfer to the Message RAM is done from Input Buffer Shadow. The Input Buffer holds the header and data sections to be transferred to the selected message buffer in the Message RAM. It is used to configure the message buffers in the Message RAM and to update the data sections of transmit buffers.

When updating the header section of a message buffer in the Message RAM from the Input Buffer, the Message Buffer Status as described in section 32.11.5, FlexRay Message Buffer Status Register (FRMBS) is automatically set to "0".

The header sections of message buffers belonging to the receive FIFO can only be (re)configured when the CC is in CONFIG state. For those message buffers only the payload length configured and the data pointer need to be configured via bits PLC6 to PLC0 in the FRWRHS2 register and bits DP10 to DP0 in the FRWRHS3 register. All information required for acceptance filtering is taken from the FIFO rejection filter and the FIFO rejection filter mask.

The data transfer between Input Buffer (IBF) and Message RAM is described in detail in section 32.22.2 (1), Data Transfer from Input Buffer to Message RAM.

32.10.1 FlexRay Write Data Section i Register (FRWRDSi) (i = 1 to 64)

Holds the data words to be transferred to the data section of the addressed message buffer. The data words (DWi) are written to the Message RAM in transmission order from DW1 (byte0, byte1) to DWPL (PL = number of data words as defined by the payload length configured by bits PLC6 to PLC0 in the FRWRHS2 register).

FlexRay Write Data Section 1 Register (FRWRDS1)	<P4 address: location H'FFBF F400>
FlexRay Write Data Section 2 Register (FRWRDS2)	<P4 address: location H'FFBF F404>
FlexRay Write Data Section 3 Register (FRWRDS3)	<P4 address: location H'FFBF F408>
:	:
FlexRay Write Data Section 62 Register (FRWRDS62)	<P4 address: location H'FFBF F4F4>
FlexRay Write Data Section 63 Register (FRWRDS63)	<P4 address: location H'FFBF F4F8>
FlexRay Write Data Section 64 Register (FRWRDS64)	<P4 address: location H'FFBF F4FC>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD31	MD30	MD29	MD28	MD27	MD26	MD25	MD24	MD23	MD22	MD21	MD20	MD19	MD18	MD17	MD16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MD15	MD14	MD13	MD12	MD11	MD10	MD9	MD8	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MD31 to MD0	All 0	R	W	Message Data MD7 to MD0: DW_{n+1}, $byte_{n+2}$ MD15 to MD8: DW_{n+1}, $byte_{n+1}$ MD23 to MD16: DW_{n}, $byte_{n}$ MD31 to MD24: DW_{n}, $byte_{n-1}$

Note: • DW127 is located on bits MD15 to MD0 in the FRWRDS64 register. In this case bits MD 31 to MD16 in the FRWRDS64 register are unused (no valid data). The Input Buffer RAMs are initialized to "0" when leaving hard reset or by CHI command CLEAR_RAMS.

By setting the FBSEN bit in the FXROC register, the bit order on reading and writing this register can be selected to be little endian style or big endian style.

- When the FBSEN bit in the FXROC register is set to "1" (big endian style)
  MD7 to MD0 in the FRWRDSn register = DW_{n+1}, $byte_{n+2}$
  MD15 to MD8 in the FRWRDSn register = DW_{n+1}, $byte_{n+1}$
  MD23 to MD16 in the FRWRDSn register = DW_{n}, $byte_{n}$
  MD31 to MD24 in the FRWRDSn register = DW_{n}, $byte_{n-1}$
- When the FBSEN bit in the FXROC register is set to "0" (little endian style)
  MD7 to MD0 in the FRWRDSn register = DW_{n}, $byte_{n-1}$
  MD15 to MD8 in the FRWRDSn register = DW_{n}, $byte_{n}$
  MD23 to MD16 in the FRWRDSn register = DW_{n+1}, $byte_{n+1}$
  MD31 to MD24 in the FRWRDSn register = DW_{n+1}, $byte_{n+2}$

Legend: n = 0 to 11

32.10.2 FlexRay Write Header Section Register 1 (FRWRHS1)

FlexRay Write Header Section Register 1 (FRWRHS1) <P4 address: location H'FFBF F500>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MBI	TXM	PPIT	CFG	CHB	CHA	—	CYC6	CYC5	CYC4	CYC3	CYC2	CYC1	CYC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	FID10	FID9	FID8	FID7	FID6	FID5	FID4	FID3	FID2	FID1	FID0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31, 30	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
29	MBI	0	R	W	Message Buffer Interrupt Bit This bit enables the receive / transmit interrupt for the corresponding message buffer. After a dedicated receive buffer has been updated by the Message Handler, flags RXI and / or MBSI in the FRSIR register are set to "1". After a transmission has completed the TXI flag in the FRSIR register is set to "1". 0: The corresponding message buffer interrupt is disabled 1: The corresponding message buffer interrupt is enabled
28	TXM	0	R	W	Transmission Mode Bit This bit is used to select the transmission mode (see section 32.19.3, Transmit Buffers). 0: Continuous mode 1: Single-shot mode
27	PPIT	0	R	W	Payload Preamble Indicator Transmit Bit This bit is used to control the state of the Payload Preamble Indicator in transmit frames. If the bit is set to "1" in a static message buffer, the respective message buffer holds network management information. If the bit is set to "1" in a dynamic message buffer the first two bytes of the payload segment may be used for message ID filtering by the receiver. Message ID filtering of received FlexRay frames is not supported by the FlexRay module, but can be done by the CPU. 0: Payload Preamble Indicator not set to "1" 1: Payload Preamble Indicator set to "1"
26	CFG	0	R	W	Message Buffer Direction Configuration Bit*[3] 0: The corresponding buffer is configured as Receive Buffer 1: The corresponding buffer is configured as Transmit Buffer

Bit	Abbreviation	After Reset	R	W	Description
25	CHB	0	R	W	Cycle Code Bit*[2]
24	CHA	0	R	W	• In transmit buffer 00: no transmission 01: channel A 10: channel B 11: both channels • In receive buffer 00: ignore frame 01: channel A 10: channel B 11: channel A or B (store first semantically valid frame)
23	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
22 to 16	CYC6 to CYC0	All 0	R	W	Cycle Code Bit The 7-bit cycle code determines the cycle set used for cycle counter filtering. For details about the configuration of the cycle code see section 32.18.2, Cycle Counter Filtering.
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	FID10 to FID0	All 0	R	W	Frame ID Bit*[1] Frame ID of the selected message buffer. The frame ID defines the slot number for transmission / reception of the respective message.

Notes: *1 Message buffers with frame ID = 0 are considered as not valid.

*2 If a message buffer is configured for the dynamic segment and both bits of the channel filtering field are set to "1", no frames are transmitted resp. received frames are ignored (same function as CHA = CHB = 0)

*3 For message buffers belonging to the receive FIFO the bit is not evaluated.

32.10.3 FlexRay Write Header Section Register 2 (FRWRHS2)

FlexRay Write Header Section Register 2 (FRWRHS2) <P4 address: location H'FFBF F504>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PLC6	PLC5	PLC4	PLC3	PLC2	PLC1	PLC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CRC10	CRC9	CRC8	CRC7	CRC6	CRC5	CRC4	CRC3	CRC2	CRC1	CRC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 23	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
22 to 16	PLC6 to PLC0	All 0	R	W	Payload Length Configured Bit Length of data section (number of 2-byte words) as configured by the program. During static segment the static frame payload length as configured by bits SFDL6 to SFDL0 in the FRMHDC register defines the payload length for all static frames. If the payload length configured by bits PLC6 to PLC0 is shorter than this value padding bytes are inserted to ensure that frames have proper physical length. The padding pattern is logical "0".
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	CRC10 to CRC0	All 0	R	W	Header CRC Bit (vRF!Header!HeaderCRC)*[1] Receive Buffer: Configuration not required Transmit Buffer: Header CRC calculated and configured by the program For calculation of the header CRC the payload length of the frame send on the bus has to be considered. In static segment the payload length of all frames is configured by bits SFDL6 to SFDL0 in the FRMHDC register.

Note: *1 Receive Buffer: Configuration not required.

32.10.4 FlexRay Write Header Section Register 3 (FRWRHS3)

FlexRay Write Header Section Register 3 (FRWRHS3) <P4 address: location H'FFBF F50A>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DP10	DP9	DP8	DP7	DP6	DP5	DP4	DP3	DP2	DP1	DP0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 11	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
10 to 0	DP10 to DP0	All 0	R	W	Data Pointer Bit Pointer to the first 32-bit word of the data section of the addressed message buffer in the Message RAM.

32.10.5 FlexRay Input Buffer Command Mask Register (FRIBCM)

Configures how the message buffer in the Message RAM selected by the FRIBCR register is updated. When IBF Host and IBF Shadow are swapped, also mask bits LHSH, LDSH, and STXRH are swapped with bits LHSS, LDSS, and STXRS to keep them attached to the respective Input Buffer transfer.

FlexRay Input Buffer Command Mask Register (FRIBCM) <P4 address: location H'FFBF F510>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	STXRS	LDSS	LHSS
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	STXRH	LDSH	LHSH
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 19	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
18	STXRS	0	R	0	Set Transmission Request Shadow Bit 0: Set TXR flag to "0" 1: Set TXR flag to "1", transmit buffer released for transmission (operation ongoing or finished)
17	LDSS	0	R	0	Load Data Section Shadow Bit 0: Data section is not updated 1: Data section selected for transfer from Input Buffer to the Message RAM (transfer ongoing or finished)
16	LHSS	0	R	0	Load Header Section Shadow Bit 0: Header section is not updated 1: Header section selected for transfer from Input Buffer to the Message RAM (transfer ongoing or finished)
15 to 3	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
2	STXRH	0	R	W	Set Transmission Request Host Bit*[1]*[2] 0: Set TXR flag to "0" 1: Set TXR flag to "1", transmit buffer released for transmission
1	LDSH	0	R	W	Load Data Section Host Bit 0: Data section is not updated 1: Data section selected for transfer from Input Buffer to the Message RAM
0	LHSH	0	R	W	Load Header Section Host Bit 0: Header section is not updated 1: Header section selected for transfer from Input Buffer to the Message RAM

Notes: *1 If this bit is set to "1", the TXR flag for the selected message buffer is set to "1" in the TXRQi registers (i = 1 to 4) to release the message buffer for transmission. In single-shot mode the flag is set to "0" by the CC after transmission has completed.

*2 When the STXRH bit is set to "1", the TXR bit is set to "1" regardless of receive or transmit buffers. For transmission, only the TXR bit for transmit buffers is checked; no transmission is performed for receive buffers even if the TXR bit is set to "1".

- When IBF Host and IBF Shadow are swapped, also mask bits LHSH, LDSH, and STXRH are swapped with bits LHSS, LDSS, and STXRS to keep them attached to the respective Input Buffer transfer.

32.10.6 FlexRay Input Buffer Command Request Register (FRIBCR)

When the CPU writes the number of the target message buffer in the Message RAM to bits IBRH6 to IBRH0, IBF Host and IBF Shadow are swapped. In addition the message buffer numbers stored under bits IBRH6 to IBRH0 and IBRS6 to IBRS0 are also swapped (see also section 32.22.2 (1), Data Transfer from Input Buffer to Message RAM).

With this write operation the IBSYS bit is set to "1". The Message Handler then starts to transfer the contents of IBF Shadow to the message buffer in the Message RAM selected by bits IBRS6 to IBRS0.

While the Message Handler transfers the data from IBF Shadow to the target message buffer in the Message RAM, the CPU may write the next message into the IBF Host. After the transfer between IBF Shadow and the Message RAM has completed, the IBSYS bit is set back to "0" and the next transfer to the Message RAM may be started by the CPU by writing the respective target message buffer number to bits IBRH6 to IBRH0.

If a write access to bits IBRH6 to IBRH0 occurs while the IBSYS bit is "1", the IBSYH bit is set to "1". After completion of the ongoing data transfer from IBF Shadow to the Message RAM, IBF Host and IBF Shadow are swapped, the IBSYH bit is set to "0". The IBSYS bit remains set to "1", and the next transfer to the Message RAM is started. In addition the message buffer numbers stored under bits IBRH6 to IBRH0 and IBRS6 to IBRS0 are also swapped.

Any write access to an Input Buffer register while both bits IBSYS and IBSYH are set to "1" will cause the error flag (the IIBA bit in the FREIR register) to be set to "1". In this case the Input Buffer will not be changed.

FlexRay Input Buffer Command Request Register (FRIBCR) <P4 address: location H'FFBF F514>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IBSYS	—	—	—	—	—	—	—	—	IBRS6	IBRS5	IBRS4	IBRS3	IBRS2	IBRS1	IBRS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IBSYH	—	—	—	—	—	—	—	—	IBRH6	IBRH5	IBRH4	IBRH3	IBRH2	IBRH1	IBRH0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	IBSYS	0	R	0	Input Buffer Busy Shadow Flag Set to "1" after writing bits IBRH6 to IBRH0. When the transfer between IBF Shadow and the Message RAM has completed, the IBSYS bit is set back to "0". 0: Transfer between IBF Shadow and Message RAM completed 1: Transfer beteween IBF Shadow and Message RAM in progress
30 to 23	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
22 to 16	IBRS6 to IBRS0	All 0	R	0	Input Buffer Request Shadow Bit*[1] Number of the target message buffer actually updated / lately updated. Valid values are 0x00 to 0x7F (0...127).
15	IBSYH	0	R	0	Input Buffer Busy Host Flag Set to "1" by writing bits IBRH6 to IBRH0 while the IBSYS bit is still "1". After the ongoing transfer between IBF Shadow and the Message RAM has completed, the IBSYH bit is set back to "0". 0: No request pending 1: Request while transfer between IBF Shadow and Message RAM in progress
14 to 7	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
6 to 0	IBRH6 to IBRH0	All 0	R	W	Input Buffer Request Host Bit*[1] Selects the target message buffer in the Message RAM for data transfer from Input Buffer. Valid values are 0x00 to 0x7F (0...127).

Note: *1 The values of bits IBRH and IBRS are swapped at the same time as a transfer is started.

32.11 Output Buffer

Double buffer structure consisting of Output Buffer Host and Output Buffer Shadow. Used to read out message buffers from the Message RAM. While the CPU can read from Output Buffer Host, the Message Handler transfers the selected message buffer from Message RAM to Output Buffer Shadow. The data transfer between Message RAM and Output Buffer (OBF) is described in section 32.22.2 (2), Data Transfer from Message RAM to Output Buffer.

32.11.1 FlexRay Read Data Section Register i (FRRDDSi) (i = 1 to 64)

FlexRay Read Data Section Register 1 (FRRDDS1)	<P4 address: location H'FFBF F600>
FlexRay Read Data Section Register 2 (FRRDDS2)	<P4 address: location H'FFBF F604>
FlexRay Read Data Section Register 3 (FRRDDS3)	<P4 address: location H'FFBF F608>
:	:
FlexRay Read Data Section Register 62 (FRRDDS62)	<P4 address: location H'FFBF F6F4>
FlexRay Read Data Section Register 63 (FRRDDS63)	<P4 address: location H'FFBF F6F8>
FlexRay Read Data Section Register 64 (FRRDDS64)	<P4 address: location H'FFBF F6FC>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD31	MD30	MD29	MD28	MD27	MD26	MD25	MD24	MD23	MD22	MD21	MD20	MD19	MD18	MD17	MD16
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MD15	MD14	MD13	MD12	MD11	MD10	MD9	MD8	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	MD31 to MD0	All 0	R	N	Message Data Holds the data words read from the data section of the addressed message buffer. The data words (DWi) are read from the Message RAM in reception order from DW1 (byte0, byte1) to DW_{PL} (PL = number of data words as defined by the payload length configured by bits PLC6 to PLC0 in the FRRDHS2 register). MD7 to MD0: DW_{n+1}, $byte_{n+2}$ MD15 to MD8: DW_{n+1}, $byte_{n+1}$ MD23 to MD16: DW_{n}, $byte_{n}$ MD31 to MD24: DW_{n}, $byte_{n-1}$

Note: • DW127 is located on bits MD15 to MD0 in the FRRDDS64 register. In this case bits MD31 to MD16 in the FRRDDS64 register are unused (no valid data). The Output Buffer RAMs are initialized to "0" when leaving hard reset or by CHI command CLEAR_RAMS.

By setting the FBSEN bit in the FXROC register, the bit order on reading this register can be selected to be little endian style or big endian style.

- When the FBSEN bit in the FXROC register is set to "1" (big endian style)

MD7 to MD0 in the FRRDDSn register = DW_{n+1}, $byte_{n+2}$

MD15 to MD8 in the FRRDDSn register = DW_{n+1}, $byte_{n+1}$

MD23 to MD16 in the FRRDDSn register = DW_{n}, $byte_{n}$

MD31 to MD24 in the FRRDDSn register = DW_{n}, $byte_{n-1}$

- When the FBSEN bit in the FXROC register is set to "0" (little endian style)

MD7 to MD0 in the FRRDDSn register = DW_{n}, $byte_{n-1}$

MD15 to MD8 in the FRRDDSn register = DW_{n}, $byte_{n}$

MD23 to MD16 in the FRRDDSn register = DW_{n+1}, $byte_{n+1}$

MD31 to MD24 in the FRRDDSn register = DW_{n+1}, $byte_{n+2}$

32.11.2 FlexRay Read Header Section Register 1 (FRRDHS1)

FlexRay Read Header Section Register 1 (FRRDHS1) <P4 address: location H'FFBF F700>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MBI	TXM	PPIT	CFG	CHB	CHA	—	CYC6	CYC5	CYC4	CYC3	CYC2	CYC1	CYC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	FID10	FID9	FID8	FID7	FID6	FID5	FID4	FID3	FID2	FID1	FID0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 30	—	All 0	0	N	Reserved Bits These bits are always read as "0".
29	MBI	0	R	N	Message Buffer Interrupt Bit Values as configured by the program via FRWRHS1 register.
28	TXM	0	R	N	Transmission Mode Bit Values as configured by the program via FRWRHS1 register.
27	PPIT	0	R	N	Payload Preamble Indicator Transmit Bit Values as configured by the program via FRWRHS1 register.
26	CFG	0	R	N	Message Buffer Direction Configuration Bit Values as configured by the program via FRWRHS1 register.
25	CHB	0	R	N	Channel Filter Control Bit Values as configured by the program via FRWRHS1 register.
24	CHA	0	R	N	
23	—	0	0	N	Reserved Bits This bit is always read as "0".
22 to 16	CYC6 to CYC0	All 0	R	N	Cycle Code Values as configured by the program via FRWRHS1 register.
15 to 11	—	All 0	0	N	Reserved Bits These bits are always read as "0".
10 to 0	FID10 to FID0	All 0	R	N	Frame ID Values as configured by the program via FRWRHS1 register.

Note: • In case that the message buffer read from the Message RAM belongs to the receive FIFO. Bits FID10 to FID0 hold the received frame ID, while bits CYC6 to CYC0, CHA, CHB, CFG, PPIT, TXM, and MBI are set to "0".

32.11.3 FlexRay Read Header Section Register 2 (FRRDHS2)

The FRRDHS2 register is updated from data frames only.

FlexRay Read Header Section Register 2 (FRRDHS2) <P4 address: location H'FFBF F704>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	PLR6	PLR5	PLR4	PLR3	PLR2	PLR1	PLR0	—	PLC6	PLC5	PLC4	PLC3	PLC2	PLC1	PLC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CRC10	CRC9	CRC8	CRC7	CRC6	CRC5	CRC4	CRC3	CRC2	CRC1	CRC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	—	0	0	N	Reserved Bit This bit is always read as "0".
30 to 24	PLR6 to PLR0	All 0	R	N	Payload Length Received (vRF!Header!Length)*[1] Payload length value updated from received data frames.
23	—	0	0	N	Reserved Bit This bit is always read as "0".
22 to 16	PLC6 to PLC0	All 0	R	N	Payload Length Configured Length of data section (number of 2-byte words) as configured by the program.
15 to 11	—	All 0	0	N	Reserved Bits These bits are always read as "0".
10 to 0	CRC10 to CRC0	All 0	R	N	Header CRC (vRF!Header!HeaderCRC) Receive Buffer: Header CRC updated from received data frames Transmit Buffer: Header CRC calculated and configured by the program

Note: *1 If message buffer belongs to the receive FIFO (bits PLR6 to PLR0) is also updated from received null frames.

When a message is stored into a message buffer the following behaviour with respect to payload length received and payload length configured is implemented:

- Bits PLR6 to PLR0 > Bits PLC6 to PLC0:

  The payload data stored in the message buffer is truncated to the payload length configured if bits PLC 6 to PLC0 are even or else truncated to bits PLC6 to PLC0 + 1.
- PLR6 to PLR0 ≤ Bits PLC6 to PLC0:

  The received payload data is stored into the message buffers data section. The remaining data bytes of the data section as configured by bits PLC6 to PLC0 are filled with undefined data
- Bits PLR6 to PLR0 = 0:

  The message buffer's data section is filled with undefined data
- Bits PLC6 to PLC0 = 0:

  Message buffer has no data section configured. No data is stored into the message buffer's data section.

Note: • The Message RAM is organized in 4-byte words. When received data is stored into a message buffer's data section, the number of 2-byte data words written into the message buffer is bits PLC6 to PLC0 rounded to the next even value. Bits PLC6 to PLC0 should be configured identical for all message buffers belonging to the receive FIFO. Header 2 is updated from data frames only.

32.11.4 FlexRay Read Header Section Register 3 (FRRDHS3)

The FRRDHS3 register is updated from data frames only.

FlexRay Read Header Section Register 3 (FRRDHS3) <P4 address: location H'FFBF F708>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RES	PPI	NFI	SYN	SFI	RCI	—	—	RCC5	RCC4	RCC3	RCC2	RCC1	RCC0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DP10	DP9	DP8	DP7	DP6	DP5	DP4	DP3	DP2	DP1	DP0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 30	—	All 0	0	N	Reserved Bits These bits are always read as "0".
29	RES	0	R	N	Reserved Bit (vRF!Header!Reserved) Reflects the state of the received reserved bit. The reserved bit is transmitted as "0"*[1].
28	PPI	0	R	N	Payload Preamble Indicator Flag (vRF!Header!PPIndicator) 0: The payload segment of the received frame does not contain a network management vector nor a message ID 1: Static segment: Network management vector in the first part of the payload Dynamic segment: Message ID in the first part of the payload
27	NFI	0	R	N	Null Frame Indicator Flag (vRF!Header!NFIndicator) 0: Up to now no data frame has been stored into the respective message buffer 1: At least one data frame has been stored into the respective message buffer
26	SYN	0	R	N	Sync Frame Indicator Flag (vRF!Header!SyFIndicator) 0: The received frame is not a sync frame 1: The received frame is a sync frame
25	SFI	0	R	N	Startup Frame Indicator Flag (vRF!Header!SuFIndicator) 0: The received frame is not a startup frame 1: The received frame is a startup frame
24	RCI	0	R	N	Received on Channel Indicator Flag (vSS!Channel) 0: Frame received on channel B 1: Frame received on channel A
23 to 22	—	All 0	0	N	Reserved Bits These bits are always read as "0".
21 to 16	RCC5 to RCC0	All 0	R	N	Receive Cycle Count (vRF!Header!CycleCount) Cycle counter value updated from received data frame
15 to 11	—	All 0	0	N	Reserved Bits These bits are always read as "0".
10 to 0	DP10 to DP0	All 0	R	N	Data Pointer Pointer to the first 32-bit word of the data section of the addressed message buffer in the Message RAM

Note: *1 The reserved bit is transmitted as "0".

32.11.5 FlexRay Message Buffer Status Register (FRMBS)

The message buffer status is updated by the CC with respect to the assigned channel(s) latest at the end of the slot following the slot assigned to the message buffer. The flags are updated only when the CC is in NORMAL_ACTIVE or NORMAL_PASSIVE state.

If only one channel (A or B) is assigned to a message buffer, the channel-specific status flags of the other channel are written to "0". If both channels are assigned to a message buffer, the channel-specific status flags of both channels are updated. The message buffer status is updated only when the slot counter reached the configured frame ID and when the cycle counter filter matched. When the CPU updates a message buffer via Input Buffer, all flags in the FRMBS register are set to "0" independent of which bits in the FRIBCM register are set to "1" or not.

For details about receive / transmit filtering see section 32.18, Filtering and Masking, section 32.19, Transmit Process, and section 32.20, Receive Process. Whenever the Message Handler changes one of the flags VFRA, VFRB, SEOA, SEOB, CEOA, CEOB, SVOA, SVOB, TCIA, TCIB, ESA, ESB, MLST, FTA, and FTB the respective message buffer's MBC flag in the FRMBSCi register (i = 1 to 4) is set to 1.

FlexRay Message Buffer Status Register (FRMBS) <P4 address: location H'FFBF F70C>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RESS	PPIS	NFIS	SYNS	SFIS	RCIS	—	—	CCS5	CCS4	CCS3	CCS2	CCS1	CCS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FTB	FTA	—	MLST	ESB	ESA	TCIB	TCIA	SVOB	SVOA	CEOB	CEOA	SEOB	SEOA	VFRB	VFRA
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31, 30	—	All 0	0	N	Reserved Bits These bits are always read as "0".
29	RESS	0	R	N	Reserved Bit Status Flag (vRF!Header!Reserved)*4 Reflects the state of the received reserved bit.*6
28	PPIS	0	R	N	Payload Preamble Indicator Status Flag (vRF!Header!PPIndicator)*4 0: The payload segment of the received frame does not contain a network management vector or a message ID 1: Static segment: Network management vector at the beginning of the payload, Dynamic segment: Message ID at the beginning of the payload
27	NFIS	0	R	N	Null Frame Indicator Status Flag (vRF!Header!NFIndicator)*4 0: Received frame is a null frame*5 1: Received frame is not a null frame
26	SYNS	0	R	N	Sync Frame Indicator Status Flag (vRF!Header!SyFIndicator)*4 0: No sync frame received 1: The received frame is a sync frame
25	SFIS	0	R	N	Startup Frame Indicator Status Flag (vRF!Header!SuFIndicator)*4 0: No startup frame received 1: The received frame is a startup frame

Bit	Abbreviation	After Reset	R	W	Description
24	RCIS	0	R	N	Received on Channel Indicator Status Flag (vSS!Channel)*[4] 0: Frame received on channel B 1: Frame received on channel A
23, 22	—	All 0	0	N	Reserved Bits These bits are always read as "0".
21 to 16	CCS5 to CCS0	All 0	R	N	Cycle Count Status Actual cycle count when status was updated
15	FTB	0	R	N	Frame Transmitted on Channel B Flag*[3] 0: No data frame transmitted on channel B 1: Data frame transmitted on channel B
14	FTA	0	R	N	Frame Transmitted on Channel A Flag*[3] 0: No data frame transmitted on channel A 1: Data frame transmitted on channel A
13	—	0	0	N	Reserved Bit This bit is always read as "0".
12	MLST	0	R	N	Message Lost Flag The flag is set to "1" in case the CPU did not read the message before the message buffer was updated from a received data frame. Not affected by reception of null frames except for message buffers belonging to the receive FIFO. The flag is set to "0" by a CPU write to the message buffer via IBF or when a new message is stored into the message buffer after the message buffers ND flag was set to "0" by reading out the message buffer via OBF. 0: No message lost 1: Unprocessed message was overwritten
11	ESB	0	R	N	Empty Slot Channel B Flag*[2] 0: Bus activity detected in the assigned slot on channel B 1: No bus activity detected in the assigned slot on channel B
10	ESA	0	R	N	Empty Slot Channel A Flag*[2] 0: Bus activity detected in the assigned slot on channel A 1: No bus activity detected in the assigned slot on channel A
9	TCIB	0	R	N	Transmission Conflict Indication Channel B Flag (vSS!TxConflictB) 0: No transmission conflict occurred on channel B 1: Transmission conflict occurred on channel B
8	TCIA	0	R	N	Transmission Conflict Indication Channel A Flag (vSS!TxConflictA) 0: No transmission conflict occurred on channel A 1: Transmission conflict occurred on channel A
7	SVOB	0	R	N	Slot Boundary Violation Observed on Channel B Flag (vSS!BViolationB)*[1] 0: No slot boundary violation observed on channel B 1: Slot boundary violation observed on channel B
6	SVOA	0	R	N	Slot Boundary Violation Observed on Channel A Flag (vSS!BViolationA)*[1] 0: No slot boundary violation observed on channel A 1: Slot boundary violation observed on channel A

Bit	Abbreviation	After Reset	R	W	Description
5	CEOB	0	R	N	Content Error Observed on Channel B Flag (vSS!ContentErrorB) 0: No content error observed on channel B 1: Content error observed on channel B
4	CEOA	0	R	N	Content Error Observed on Channel A Flag (vSS!ContentErrorA) 0: No content error observed on channel A 1: Content error observed on channel A
3	SEOB	0	R	N	Syntax Error Observed on Channel B Flag (vSS!SyntaxErrorB) 0: No syntax error observed on channel B 1: Syntax error observed on channel B
2	SEOA	0	R	N	Syntax Error Observed on Channel A Flag (vSS!SyntaxErrorA) 0: No syntax error observed on channel A 1: Syntax error observed on channel A
1	VFRB	0	R	N	Valid Frame Received on Channel B Flag (vSS!ValidFrameB) 0: No valid frame received on channel B 1: Valid frame received on channel B
0	VFRA	0	R	N	Valid Frame Received on Channel A Flag (vSS!ValidFrameA) 0: No valid frame received on channel A 1: Valid frame received on channel A

Notes: *1 A slot boundary violation: Channel active at the start or at the end of the assigned slot.

*2 The condition is checked in static and dynamic slots.

*3 The FlexRay protocol specification requires that bits FTA, and FTB can only be set to "0" by the program. Therefore the Cycle Count Status (bits CCS5 to CCS0) for these bits is only valid for the cycle where the bits are set to "1".

*4 For receive buffers (the CFG bit = 0) the following status bits are updated from both valid data and null frames. If no valid frame was received, the previous value is maintained. For transmit buffers the flags have no meaning and should be ignored.

*5 If set to "0" the payload segment of the received frame contains no usable data.

*6 The reserved bit is transmitted as "0".

32.11.6 FlexRay Output Buffer Command Mask Register (FROBCM)

Configures how the Output Buffer is updated from the message buffer in the Message RAM selected by the FROBCR register. Mask bits RDSS and RHSS are copied to the register internal storage when a Message RAM transfer is requested by the REQ bit in the FROBCR register. When OBF Host and OBF Shadow are swapped, mask bits RDSH and RHSH are swapped with the register internal storage to keep them attached to the respective Output Buffer transfer.

FlexRay Output Buffer Command Mask Register (FROBCM) <P4 address: location H'FFBF F710>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDSH	RHSH
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDSS	RHSS
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 18	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
17	RDSH	0	R	0	Read Data Section Host Flag 0: Data section is not read 1: Data section selected for transfer from Message RAM to Output Buffer
16	RHSH	0	R	0	Read Header Section Host Flag 0: Header section is not read 1: Header section selected for transfer from Message RAM to Output Buffer
15 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	RDSS	0	R	W	Read Data Section Shadow Bit 0: Data section is not read 1: Data section selected for transfer from Message RAM to Output Buffer
0	RHSS	0	R	W	Read Header Section Shadow Bit 0: Header section is not read 1: Header section selected for transfer from Message RAM to Output Buffer

Note: • After the transfer of the header section from the Message RAM to OBF Shadow has completed, the message buffer status changed flag (the MBC bit) of the selected message buffer in the FRMBSCi register (i = 1 to 4) is set to "0". After the transfer of the data section from the Message RAM to OBF Shadow has completed, the new data flag (the ND bit) of the selected message buffer in the FRNDATi register (i = 1 to 4) is set to "0".

32.11.7 FlexRay Output Buffer Command Request Register (FROBCR)

After setting the REQ bit to "1" while the OBSYS bit is "0", the OBSYS bit is automatically set to "1", bits OBRS6 to OBRS0 is copied to the register internal storage, mask bits RDSS and RHSS in the FROBCM register are copied to FROBCM register internal storage, and the transfer of the message buffer selected by bits OBRS6 to OBRS0 from the Message RAM to OBF Shadow is started. When the transfer between the Message RAM and OBF Shadow has completed, this is signalled by setting the OBSYS bit back to "0".

By setting VIEW bit to "1" while the OBSYS bit is "0", OBF Host and OBF Shadow are swapped. Additionally mask bits the RDSH and RHSH in the FROBCM register are swapped with the FROBCM register internal storage to keep them attached to the respective Output Buffer transfer. Bits OBRH6 to OBRH0 signals the number of the message buffer currently accessible by the CPU.

If bits REQ and VIEW are set to "1" with the same write access while the OBSYS bit is "0", the OBSYS bit is automatically set to "1" and OBF Shadow and OBF Host are swapped. Additionally mask bits RDSH and RHSH in the FROBCM register are swapped with the registers internal storage to keep them attached to the respective Output Buffer transfer. Afterwards bits OBRS6 to OBRS0 is copied to the register internal storage, and the transfer of the selected message buffer from the Message RAM to OBF Shadow is started. While the transfer is ongoing the Host can read the message buffer transferred by the previous transfer from OBF Host. When the current transfer between Message RAM and OBF Shadow has completed, this is signalled by setting the OBSYS bit back to "0".

Any write access to bits OBCR15 to OBCR8 while the OBSYS bit is set will cause the error flag (the IOBA bit in the FREIR register) to be set to "1". In this case the Output Buffer will not be changed.

FlexRay Output Buffer Command Request Register (FROBCR) <P4 address: location H'FFBF F714>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	OBRH6	OBRH5	OBRH4	OBRH3	OBRH2	OBRH1	OBRH0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OBSYS	—	—	—	—	—	REQ	VIEW	—	OBRS6	OBRS5	OBRS4	OBRS3	OBRS2	OBRS1	OBRS0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 23	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
22 to 16	OBRH6 to OBRH0	All 0	R	0	Output Buffer Request Host Number of message buffer currently accessible by the CPU. Valid values are 0x00 to 0x7F (0 to 127).
15	OBSYS	0	R	0	Output Buffer Busy Shadow Flag Set to "1" after setting the REQ bit. When the transfer between the Message RAM and OBF Shadow has completed, the OBSYS bit is set back to "0". 0: No transfer in progress 1: Transfer between Message RAM and OBF Shadow in progress
14 to 10	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
9	REQ	0	R	W	Request Message RAM Transfer Bit*[1]*[2] 0: No request 1: Transfer to OBF Shadow requested
8	VIEW	0	R	W	View Shadow Buffer Bit*[1]*[2] 0: No action 1: Swap OBF Shadow and OBF Host
7	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
6 to 0	OBRS6 to OBRS0	All 0	R	W	Output Buffer Request Shadow Bit Number of source message buffer to be transferred from the Message RAM to OBF Shadow. Valid values are 0x00 to 0x7F (0 to127). If the number of the first message buffer of the receive FIFO is written to this register the Message Handler transfers the message buffer addressed by the GET Index bit (GIDX, see section 32.21, FIFO Function) to OBF Shadow.

Notes: *1 Only writable while the OBSYS bit = 0.

*2 Bits VIEW and REQ are not set back to "0" automatically. To set either of these bits to "1", another bit should be set to "0" simultaneously. To set bits OBRS6 to OBRS0 only, both bits VIEW and REQ should be set to "0".

32.12 Communication Cycle

Communication on FlexRay networks is based on frames and symbols. The wakeup symbol (WUS) and the collision avoidance symbol (CAS) are transmitted outside the communication cycle to setup the time schedule. Frames and media access test symbols (MTS) are transmitted inside the communication cycle.

A FlexRay communication cycle consists of the following elements:

- Static Segment
- Dynamic Segment (optional)
- Symbol Window (optional)
- Network Idle Time (NIT)

Static segment, dynamic segment, and symbol window form the Network Communication Time (NCT). For each communication channel the slot counter starts at 1 and counts up until the end of the dynamic segment is reached. Both channels use the same synchronized macrotick.

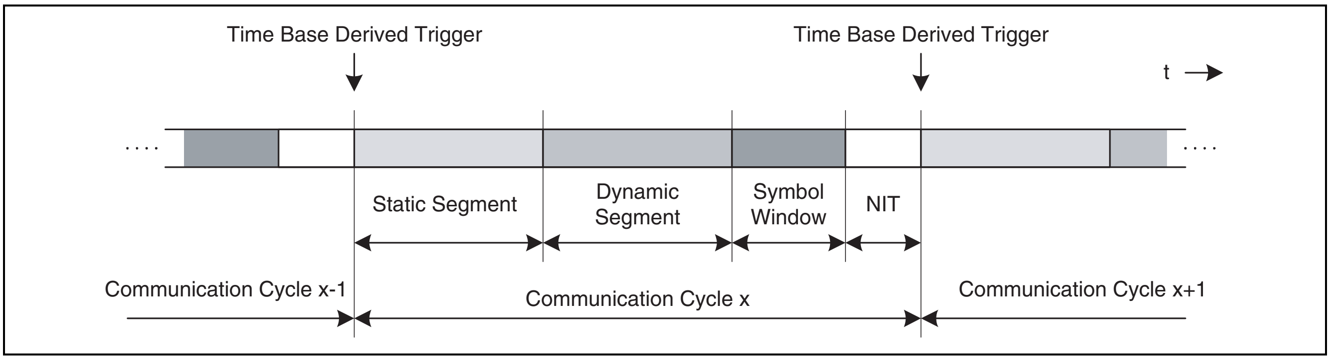


Figure 32.4 Structure of Communication Cycle

32.12.1 Static Segment

The Static Segment is characterized by the following features:

- Time slots of fixed length (optionally protected by bus guardian)
- Start of frame transmission at action point of the respective static slot
- Payload length same for all frames on both channels

Parameters: Number of Static Slots (bits NSS9 to NSS0 in the FRGTUC7 register), Static Slot Length (bits SSL9 to SSL0 in the FRGTUC7 register), Payload Length Static (bits SFDL6 to SFDL0 in the FRMHDC register), Action Point Offset (bits APO5 to APO0 in the FRGTUC9 register)

32.12.2 Dynamic Segment

The Dynamic Segment is characterized by the following features:

- All controllers have bus access (no bus guardian protection supported)
- Variable payload length and duration of slots, different for both channels
- Start of transmission at minislot action point

Parameters: Number of Minislots (bits NMS12 to NMS0 in the FRGTUC8 register), Minislot Length (bits MSL5 to MSL0 in the FRGTUC8 register), Minislot Action Point Offset (bits MAPO4 to MAPO0 in the FRGTUC9 register), Start of Latest Transmit (last minislot) (bits SLT12 to SLT0 in the FRMHDC register)

32.12.3 Symbol Window

During the symbol window only one media access test symbol (MTS) may be transmitted per channel. MTS symbols are send in NORMAL_ACTIVE state.

The symbol window is characterized by the following features:

- Send single symbol
- Transmission of the MTS symbol starts at the symbol windows action point

Parameters: Symbol Window Action Point Offset (bits APO4 to APO0 in the FRGTUC9 register) (same as for static slots), Network Idle Time Start (bits NIT13 to NIT0 in the FRGTUC4 register)

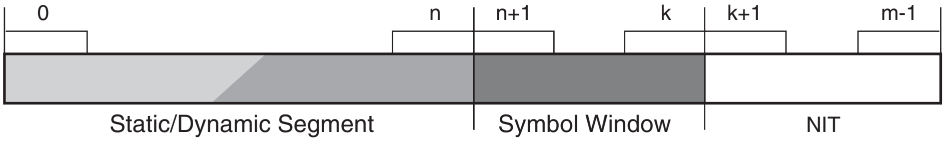
32.12.4 Network Idle Time (NIT)

During network idle time the CC has to perform the following tasks:

- Calculate clock correction terms (offset and rate)
- Distribute offset correction over multiple macroticks after offset correction start
- Perform cluster cycle related tasks

Parameters: Network Idle Time Start (bits NIT13 to NIT0 in the FRGTUC4 register), Offset Correction Start (bits OCS13 to OCS0 in the FRGTUC4 register)

32.12.5 Configuration of NIT Start and Offset Correction Start



m = MPC bit in FRGTUC2 register
k = NIT bit in FRGTUC4 register
NIT + 1 = OCS bit in FRGTUC4 register

Figure 32.5 Configuration of NIT Start and Offset Correction Start

The number of macroticks per cycle gMacroPerCycle is assumed to be m. It is configured by programming the MPC bit in the FRGTUC2 register = m.

The static / dynamic segment starts with macrotick 0 and ends with macrotick n:
n = static segment length + dynamic segment offset + dynamic segment length - 1MT
n = gNumberOfStaticSlots × gdStaticSlot + dynamic segment offset
+ gNumberOfMinislots × gdMinislot - 1 MT

The static segment length is configured by bits SSL and NSS in the FRGTUC7 register.
The dynamic segment length is configured by bits MSL and NMS in the FRGTUC8 register.

The dynamic segment offset is:

- If gdActionPointOffset ≤ gdMinislotActionPointOffset:
  dynamic segment offset = 0 MT
- Else if gdActionPointOffset > gdMinislotActionPointOffset:
  dynamic segment offset = gdActionPointOffset – gdMinislotActionPointOffset

The NIT starts with macrotick k+1 and ends with the last macrotick of cycle m-1. It has to be configured by setting the NIT bit in the FRGTUC4 register = k.

For the FlexRay module the offset correction start is required to be the OCS bit in the FRGTUC4 register σ the NIT bit int the FRGTUC4 register + 1 = k+1.

The length of symbol window results from the number of macroticks between the end of the static / dynamic segment and the beginning of the NIT. It can be calculated by k - n.

32.13 Communication Modes

The FlexRay Protocol Specification v2.1 defines the Time-Triggered Distributed (TT-D) mode.

32.13.1 Time-triggered Distributed (TT-D)

In TT-D mode the following configurations are possible:

- Pure static: Minimum 2 static slots + symbol window (optional)
- Mixed static/dynamic: Minimum 2 static slots + dynamic segment + symbol window (optional)

A minimum of two coldstart nodes needs to be configured for distributed time-triggered operation.
Two fault-free coldstart nodes are necessary for the cluster startup. Each startup frame must be a sync frame, therefore all coldstart nodes are sync nodes.

32.14 Clock Synchronization

In TT-D mode a distributed clock synchronization is used. Each node individually synchronizes itself to the cluster by observing the timing of received sync frames from other nodes.

32.14.1 Global Time

Activities in a FlexRay node, including communication, are based on the concept of a global time,. It is the clock synchronization mechanism that differentiates the FlexRay cluster from other node collections with independent clock mechanisms. The global time is a vector of two values; the cycle (cycle counter) and the cycle time (macrotick counter).

Cluster specific:

- Macrotick (MT) = basic unit of time measurement in a FlexRay network,
  a macrotick consists of an integer number of microticks (µT)
- Cycle length = duration of a communication cycle in units of macroticks (MT)

32.14.2 Local Time

Internally, nodes time their behaviour with microtick resolution. Microticks are time units derived from the oscillator clock tick of the specific node. Therefore microticks are controller-specific units. They may have different duration in different controllers. The precision of a node’s local time difference measurements is a microtick (µT).

Node specific:

- Oscillator clock -> prescaler -> microtick (µT)
- µT = basic unit of time measurement in a CC, clock correction is done in units of µTs
  Cycle counter + macrotick counter = nodes local view of the global time

32.14.3 Synchronization Process

Clock synchronization is performed by means of sync frames. Only preconfigured nodes (sync nodes) are allowed to send sync frames. In a two-channel cluster a sync node has to send its sync frame on both channels.

For synchronization in FlexRay the following constraints have to be considered:

- Max. one sync frame per node in one communication cycle
- Max. 15 sync frames per cluster in one communication cycle
- Every node has to use a preconfigured number of sync frames (bits SNM3 to SNM0 in the FRGTUC2 register) for clock synchronization

- Minimum of two sync nodes required for clock synchronization and startup

For clock synchronization the time difference between expected and observed arrival time of sync frames received during the static segment is measured. In a two channel cluster the sync node has to be configured to send sync frames on both channels. The calculation of correction terms is done during NIT (offset: every cycle, rate: every odd cycle) by using an FTM algorithm. For details see FlexRay protocol specification v2.1, chapter 8.

(1) Offset (phase) Correction

- Only deviation values measured and stored in the current cycle used
- For a two channel node the smaller value will be taken
- Calculation during NIT of every communication cycle
- Offset correction value calculated in even cycles used for error checking only
- Checked against limit values
- Correction value is a signed integer number of µTs
- Correction done in odd numbered cycles, distributed over the macroticks beginning at offset correction start up to cycle end (end of NIT) to shift nodes next start of cycle (MTs lengthened / shortened)

(2) Rate (frequency) Correction

- Pairs of deviation values measured and stored in even / odd cycle pair used
- For a two channel node the average of the differences from the two channels is used
- Calculated during NIT of odd numbered cycles
- Cluster drift damping is performed using global damping value
- Checked against limit values
- Correction value is a signed integer number of µTs
- Distributed over macroticks comprising the next even / odd cycle pair (MTs lengthened / shortened)

(3) Sync Frame Transmission

Sync frame transmission is only possible from buffer 0 and 1. Message buffer 1 may be used for sync frame transmission in case that sync frames should have different payloads on the two channels. In this case the SPLM bit in the FRMRC register has to be programmed to "1".

Message buffers used for sync frame transmission have to be configured with the key slot ID and can be (re)configured in CONFIG state only. For nodes transmitting sync frames the TXSY bit in the FRSUCC1 register must be set to "1".

32.14.4 External Clock Synchronization

During normal operation, independent clusters can drift significantly. If synchronous operation across independent clusters is desired, external synchronization is necessary; even though the nodes within each cluster are synchronized. This can be accomplished with synchronous application of CPU-deduced rate and offset correction terms to the clusters.

- External offset / rate correction value is a signed integer
- External offset / rate correction value is added to calculated offset / rate correction value
- Aggregated offset / rate correction term (external + internal) is not checked against configured limits

32.15 Error Handling

The implemented error handling concept is intended to ensure that, in case of a lower layer protocol error in one single node, communication between non-affected nodes can be maintained. In some cases, higher layer program activity is required for the CC to resume normal operation. A change of the error handling state will set the PEMC bit in the FREIR register to "1" and may trigger an interrupt to the CPU if enabled. The actual error mode is signalled by bits ERRM1 to ERRM0 in the FRCCEV register.

Table 32.7 Error Modes of the POC (degradation model)

Error Mode	Activity
ACTIVE (green)	Full operation, State: NORMAL_ACTIVE The CC is fully synchronized and supports the cluster wide clock synchronization. The CPU is informed of any error condition(s) or status change by interrupt (if enabled) or by reading the error and status interrupt flags from registers FREIR and FRSIR.
PASSIVE (yellow)	Reduced operation, State: NORMAL_PASSIVE, CC self rescue allowed The CC stops transmitting frames and symbols, but received frames are still processed. Clock synchronization mechanisms are continued based on received frames. No active contribution to the cluster wide clock synchronization. The CPU is informed of any error condition(s) or status change by interrupt (if enabled) or by reading the error and status interrupt flags from registers FREIR and FRSIR.
COMM_HALT (red)	Operation halted, State: HALT, CC self rescue not allowed The CC stops frame and symbol processing, clock synchronization processing, and the macrotick generation. The CPU has still access to error and status information by reading the error and status interrupt flags from registers FREIR and FRSIR. The bus drivers are disabled.

32.15.1 Clock Correction Failed Counter

When the Clock Correction Failed Counter reaches the “maximum without clock correction passive” limit defined by bits WCP3 to WCP0 in the FRSUCC3 register, the POC transits from NORMAL_ACTIVE to NORMAL_PASSIVE state. When it reaches the “maximum without clock correction fatal” limit defined by bits WCF3 to WCF0 in the FRSUCC3 register, it transits from NORMAL_ACTIVE or NORMAL_PASSIVE to HALT state.

The Clock Correction Failed Counter (bits CCFC3 to CCFC0 in the FRCCEV register) allows the CPU to monitor the duration of the inability of a node to compute clock correction terms after the CC passed protocol startup phase.

It will be incremented by one at the end of any odd communication cycle during which either the missing offset correction (the MOCS bit in the FRSFS register) or the missing rate correction (the MRCS bit in the FRSFS register) is set to "1".

The Clock Correction Failed Counter is set to "0" at the end of an odd communication cycle if neither the missing offset correction (the MOCS bit in the FRSFS register) nor the missing rate correction (the MRCS flag in the FRSFS register) is set to "1".

The Clock Correction Failed Counter stops incrementing when the “maximum without clock correction fatal” value (bits WCF3 to WCF0 in the FRSUCC3 register) is reached (i.e. incrementing the counter at its maximum value will not cause it to wrap around back to "0"). The Clock Correction Failed Counter is initialized to "0" when the CC enters READY state or when NORMAL_ACTIVE state is entered.

Note: • The transition to HALT state is prevented if the HCSE bit in the FRSUCC1 register is not set.

32.15.2 Passive to Active Counter

The passive to active counter controls the transition of the POC from NORMAL_PASSIVE to NORMAL_ACTIVE state. Bits PTA4 to PTA0 in the FRSUCC1 register define the number of consecutive even / odd cycle pairs that must have valid clock correction terms before the CC is allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state. If bits PTA4 to PTA0 in the FRSUCC1 register is set to "0" the CC is not allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state.

32.15.3 HALT Command

In case the CPU wants to stop FlexRay communication of the local node it can bring the CC into HALT state by asserting the HALT command. This can be done by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0110. In order to shut down communication on an entire FlexRay network, a higher layer protocol is required to assure that all nodes apply the HALT command at the same time.

The POC state from which the transition to HALT state took place can be read from bits PSL5 to PSL0 in the FRCCSV register.

When called in NORMAL_ACTIVE or NORMAL_PASSIVE state the POC transits to HALT state at the end of the current cycle. When called in any other state bits CMD3 to CMD0 in the FRSUCC1 register will be set to B'0000 = command_not_accepted and the CNA bit in the FREIR register is set to "1". If enabled an interrupt to the CPU is generated.

32.15.4 FREEZE Command

In case the CPU detects a severe error condition it can bring the CC into HALT state by asserting the FREEZE command. This can be done by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0111. The FREEZE command triggers the entry of the HALT state immediately regardless of the actual POC state.

The POC state from which the transition to HALT state took place can be read from bits PSL5 to PSL0 in the FRCCSV register.

32.16 Communication Controller Status

32.16.1 Communication Controller State Diagram

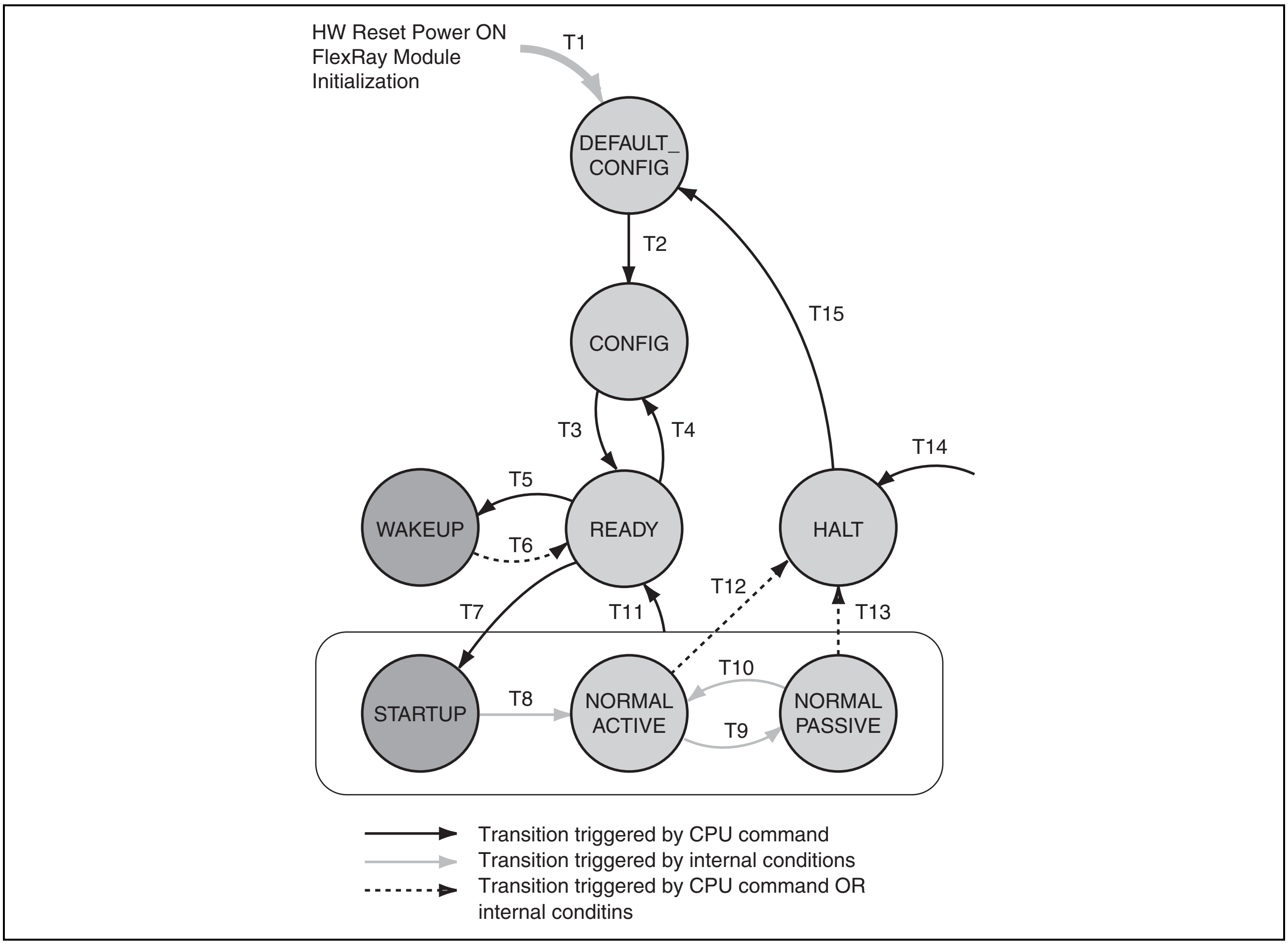


Figure 32.6 Overall State Diagram of FlexRay Communication Controller

State transitions are controlled by externals pins RESET, FRXA, and FRXB by the POC state machine, and by the CHI Command Vector (bits CMD3 to CMD0 in the FRSUCC1 register).

The CC exits from all states to HALT state after application of the FREEZE command (bits CMD3 to CMD0 in the FRSUCC1 register = B'0111).

Table 32.8 State Transitions of FlexRay overall state Machine

T#	Condition	From	To
T1	Hard reset	All states	DEFALT_CONFIG
T2	Command CONFIG, bits CMD3 to CMD0 in the FRSUCC1 register = B'0001	DEFALT_CONFIG	CONFIG
T3	Unlock sequence followed by command READY, bits CMD3 to CMD0 in the FRSUCC1 register = B'0010	CONFIG	READY
T4	Command CONFIG, bits CMD3 to CMD0 in the FRSUCC1 register = B'0001	READY	CONFIG
T5	Command WAKEUP, bits CMD3 to CMD0 in the FRSUCC1 register = B'0011	READY	WAKEUP
T6	Complete, non-aborted transmission of wakeup pattern OR received WUP OR received frame header OR wakeup collision OR command READY, bits CMD3 to CMD0 in the FRSUCC1 register = B'0010	WAKEUP	READY
T7	Command RUN, bits CMD3 to CMD0 in the FRSUCC1 register = B'0100	READY	STARTUP
T8	Successful startup	STARTUP	NORMAL_ACTIVE
T9	Clock Correction Failed counter reached Maximum Without Clock Correction Passive limit configured by bits WCP3 to WCP0 in FRSUCC3 register	NORMAL_ACTIVE	NORMAL_PASSIVE
T10	Number of valid correction terms reached the Passive to Active limit configured by bits PTA4 to PTA0 in the FRSUCC1 register	NORMAL_PASSIVE	NORMAL_ACTIVE
T11	Command READY, bits CMD3 to CMD0 in the FRSUCC1 register = B'0010	STARTUP NORMAL_ACTIVE NORMAL_PASSIVE	READY
T12	Clock Correction Failed counter reached Maximum Without Clock Correction Fatal limit configured by bits CF3 to CF0 in the FRSUCC3 register AND the HCSE bit in the FRSUCC1 register is set to 1 OR command HALT, bits CMD3 to CMD0 in the FRSUCC1 register = B'0110	NORMAL_ACTIVE	HALT
T13	Clock Correction Failed counter reached Maximum Without Clock Correction Fatal limit configured by bits WCF3 to WCF0 in the FRSUCC3 register AND the HCSE bit in the FRSUCC1 register set to 1 OR command HALT, bits CMD3 to CMD9 in the FRSUCC1 = B'0110	NORMAL_PASSIVE	HALT
T14	Command FREEZE, bits CMD3 to CMD0 in the FRSUCC1 register = B'0111	All States	HALT
T15	Command CONFIG, bits CMD3 to CMD0 in the FRSUCC1 register = B'0001	HALT	DEFALT_CONFIG

32.16.2 DEFAULT_CONFIG State

In DEFAULT_CONFIG state, the CC is stopped. All configuration registers are accessible and the pins to the physical layer are in their inactive state.

The CC enters this state

- When leaving hard reset (external reset signal (RESET) is deactivated)
- When exiting from HALT state

To leave DEFAULT_CONFIG state the CPU has to write bits CMD3 to CMD0 in the FRSUCC1 register = B'0001. The CC then transits to CONFIG state.

32.16.3 CONFIG State

In CONFIG state, the CC is stopped. All configuration registers are accessible and the pins to the physical layer are in their inactive state. This state is used to initialize the CC configuration.

The CC enters this state

- When exiting from DEFAULT_CONFIG state
- When exiting from READY state

When the state has been entered via HALT and DEFAULT_CONFIG state, the CPU can analyse status information and configuration. Before leaving CONFIG state the CPU has to assure that the configuration is fault-free.

To leave CONFIG state, the CPU has to perform the unlock sequence as described in section 32.4.2, FlexRay Lock Register (FRLCK), Directly after unlocking the CONFIG state the CPU has to write bits CMD3 to CMD0 in the FRSUCC1 register to enter the next state.

Note: • Status bits FRMHDS14 to FRMHDS0, registers FRTXRQ1 to FRTXR4, and status data stored in the Message RAM are not affected by the transition of the POC from CONFIG to READY state.

When the CC is in CONFIG state it is also possible to bring the CC into a power saving mode by halting the module clocks (FRck, PAck). To do this the CPU has to assure that all Message RAM transfers have finished before turning off the clocks.

32.16.4 READY State

After unlocking CONFIG state and writing bits CMD3 to CMD0 in the FRSUCC1 register = 0010b the CC enters READY state.

From this state the CC can transit to WAKEUP state and perform a cluster wakeup or to STARTUP state to perform a coldstart or to integrate into a running cluster.

The CC enters this state

- When exiting from CONFIG, WAKEUP, STARTUP, NORMAL_ACTIVE, or NORMAL_PASSIVE state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0010 (READY command).

The CC exits from this state

- To CONFIG state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0001 (CONFIG command)
- To WAKEUP state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0011 (WAKEUP command)
- To STARTUP state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0100 (RUN command)

Internal counters and the CC status flags are set to 0 when the CC enters STARTUP state.

Note: • Status bits FRMHDS14 to FRMHDS0, registers FRTXRQ1 to FRTXRQ4, and status data stored in the Message RAM are not affected by the transition of the POC from READY to STARTUP state.

32.16.5 WAKEUP State

The description below is intended to help configuring wakeup for the FlexRay module. A detailed description of the wakeup procedure together with the respective SDL diagrams can be found in the FlexRay protocol specification v2.1, section 7.1.

The CC enters this state

- When exiting from READY state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0011 (WAKEUP command).

The CC exits from this state to READY state

- After complete non-aborted transmission of wakeup pattern
- After WUP reception
- After detecting a WUP collision
- After reception of a frame header
- By writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0010 (READY command)

The cluster wakeup must precede the communication startup in order to ensure that all nodes in a cluster are awake. The minimum requirement for a cluster wakeup is that all bus drivers are supplied with power. A bus driver has the ability to wake up the other components of its node when it receives a wakeup pattern on its channel. At least one node in the cluster needs an external wakeup source.

The CPU completely controls the wakeup procedure. It is informed about the state of the cluster by the bus driver and the CC and configures bus guardian (if available) and CC to perform the cluster wakeup. The CC provides to the CPU the ability to transmit a special wakeup pattern on each of its available channels separately.

Wakeup may be performed on only one channel at a time. The CPU has to configure the wakeup channel while the CC is in CONFIG state by writing the WUCS bit in the FRSUCC1 register. The CC ensures that ongoing communication on this channel is not disturbed. The CC cannot guarantee that all nodes connected to the configured channel awake upon the transmission of the wakeup pattern, since these nodes cannot give feedback until the startup phase. The wakeup procedure enables single-channel devices in a two-channel system to trigger the wakeup, by only transmitting the wakeup pattern on the single channel to which they are connected. Any coldstart node that deems a system startup necessary will then wake the remaining channel before initiating communication startup.

The wakeup procedure tolerates any number of nodes simultaneously trying to wakeup a single channel and resolves this situation such that only one node transmits the pattern. Additionally the wakeup pattern is collision resilient, so even in the presence of a fault causing two nodes to simultaneously transmit a wakeup pattern, the resulting collided signal can still wake the other nodes.

After wakeup the CC returns to READY state and signals the change of the wakeup status to the CPU by setting the WST flag in the FRSIR register to "1". The wakeup status vector can be read from bits WSV2 to WSV0 in the FRCCSV register. If a valid wakeup pattern was received also either the WUPA flag in the FRSIR register or the WUPB flag in the FRSIR register is set to "1".

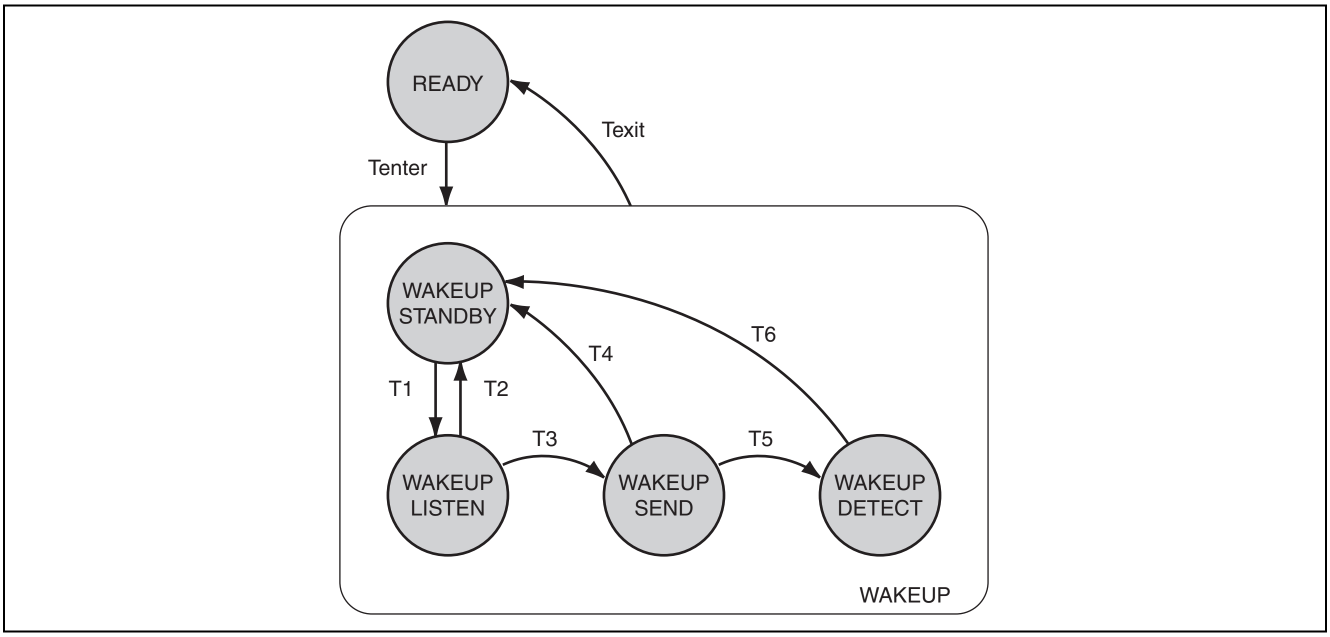


Figure 32.7 Structure of POC State WAKEUP

Table 32.9 State Transitions WAKEUP

T#	Condition	From	To
Tenter	CPU commands change to WAKEUP state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0011 (WAKEUP command)	READY	WAKEUP
T1	CHI command WAKEUP triggers wakeup FSM to transit to WAKEUP_LISTEN state	WAKEUP_STANDBY	WAKEUP_LISTEN
T2	Received WUP on wakeup channel selected by the WUCS bit in the FRSUCC1 register OR frame header on either available channel	WAKEUP_LISTEN	WAKEUP_STANDBY
T3	Timer event	WAKEUP_LISTEN	WAKEUP_SEND
T4	Complete, non-aborted transmission of wakeup pattern	WAKEUP_SEND	WAKEUP_STANDBY
T5	Collision detected	WAKEUP_SEND	WAKEUP_DETECT
T6	Wakeup timer expired OR WUP detected on wakeup channel selected by the WUCS bit in the FRSUCC1 register OR frame header received on either available channel	WAKEUP_DETECT	WAKEUP_STANDBY
Texit	Wakeup completed (after T2 or T4 or T6) OR CPU commands change to READY state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0010 (READY command). This command also resets the wakeup FSM to WAKEUP_STANDBY state	WAKEUP	READY

The WAKEUP_LISTEN state is controlled by the wakeup timer and the wakeup noise timer. The two timers are controlled by the parameters listen timeout (bits LT20 to LT0 in the FRSUCC2 register) and listen timeout noise (bits LTN3 to LTN0 in the FRSUCC2 register). Listen timeout enables a fast cluster wakeup in case of a noise free environment, while listen timeout noise enables wakeup under more difficult conditions regarding noise interference.

In WAKEUP_SEND state the CC transmits the wakeup pattern on the configured channel and checks for collisions. After return from wakeup the CPU has to bring the CC into STARTUP state by CHI command RUN.

In WAKEUP_DETECT state the CC attempts to identify the reason for the wakeup collision detected in WAKEUP_SEND state. The monitoring is bounded by the expiration of listen timeout as configured by bits LT20 to LT0 in the FRSUCC2 register. Either the detection of a wakeup pattern indicating a wakeup attempt by another node or the reception of a frame header indicating ongoing communication, causes the direct transition to READY state. Otherwise WAKEUP_DETECT is left after expiration of listen timeout; in this case the reason for wakeup collision is unknown.

The CPU has to be aware of possible failures of the wakeup and act accordingly. It is advisable to delay any potential startup attempt of the node having instigated the wakeup by the minimal time it takes another coldstart node to become awake and to be configured.

The FlexRay Protocol Specification v2.1 recommends that two different CCs shall awake the two channels.

(1) CPU activities

The CPU must coordinate the wakeup of the two channels and must decide whether, or not, to wake a specific channel. The sending of the wakeup pattern is initiated by the program. The wakeup pattern is detected by the remote BDs and signalled to their local Host.

Wakeup procedure controlled by program (single-channel wakeup):

- Configure the CC in CONFIG state
  - Select wakeup channel by programming the WUCS bit in the FRSUCC1 register
- Check local BDs whether a WUP was received
- Activate BD of selected wakeup channel
- Command CC to enter READY state
- Command CC to start wakeup on the configured channel by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0011
  - CC enters WAKEUP
  - CC returns to READY state and signals status of wakeup attempt to the CPU
- Wait predefined time to allow the other nodes to wakeup and configure themselves
- Coldstart node:
  - In a dual channel cluster wait for WUP on the other channel
  - Set coldstart inhibit flag (the CSI bit in the FRCCSV register) to "0" by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'1001 (ALLOW_COLDSTART command)
- Command CC to enter startup by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0100 (RUN command)

Wakeup procedure triggered by BD:

- Wakeup recognized by BD
- BD triggers power-up of CPU (if required)
- BD signals wakeup event to CPU
- CPU configures its local CC
- If necessary, CPU commands wakeup of second channel and waits predefined time to allow the other nodes to wakeup and configure themselves
- CPU commands CC to enter STARTUP state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0100 (RUN command)

(2) Wakeup Pattern (WUP)

The wakeup pattern (WUP) is composed of at least two wakeup symbols (WUS). Wakeup symbol and wakeup pattern are configured by registers FRPRTC1 and FRPRTC2.

- Single channel wakeup, wakeup symbol may not be sent on both channels at the same time
- Wakeup symbol collision resilient for at least two sending nodes (two overlapping wakeup symbols always recognizable)
- Wakeup symbol must be configured identical in all nodes of a cluster
- Wakeup symbol transmit low time configured by bits TXL5 to TXL0 in the FRPRTC2 register
- Wakeup symbol idle time used to listen for activity on the bus, configured by bits TXI7 to TXI0 in the FRPRTC2 register
- A wakeup pattern composed of at least two Tx-wakeup symbols needed for wakeup
- Number of repetitions configurable by bits RWP5 to RWP0 in the FRPRTC1 register (2 to 63 repetitions)
- Wakeup symbol receive window length configured by bits RXW8 to RXW0 in the FRPRTC1 register
- Wakeup symbol receive low time configured by bits RXL5 to RXL0 in the FRPRTC2 register
- Wakeup symbol receive idle time configured by bits RXI5 to RXI0 in the FRPRTC2 register

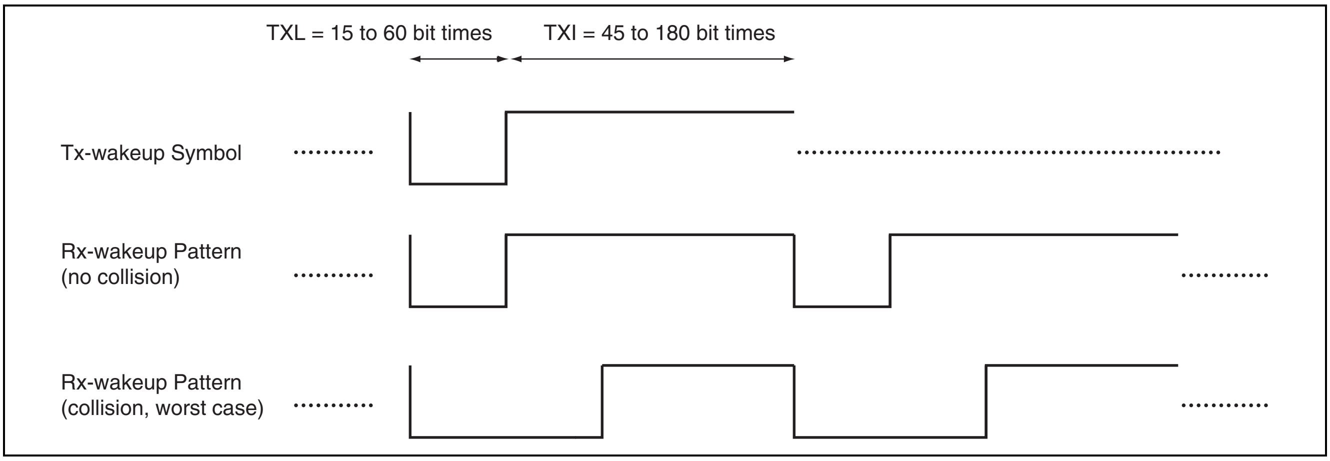

TXL = 15 to 60 bit times
TXI = 45 to 180 bit times
Tx-wakeup Symbol
Rx-wakeup Pattern (no collision)
Rx-wakeup Pattern (collision, worst case)

Figure 32.8 Timing of Wakeup Pattern

32.16.6 STARTUP State

The description below is intended to help configuring startup for the FlexRay module. A detailed description of the startup procedure together with the respective SDL diagrams can be found in the FlexRay protocol specification v2.1, section 7.2.

Any node entering STARTUP state that has coldstart capability should assure that both channels attached have been awakened before initiating coldstart.

It cannot be assumed that all nodes and stars need the same amount of time to become completely awake and to be configured. Since at least two nodes are necessary to start up the cluster communication, it is advisable to delay any potential startup attempt of the node having instigated the wakeup by the minimal amount of time it takes another coldstart node to become awake, to be configured and to enter startup. It may require several hundred milliseconds (depending on the hardware used) before all nodes and stars are completely awakened and configured.

Startup is performed on all channels synchronously. During startup, a node only transmits startup frames. Startup frames are both sync frames and null frames during startup.

A fault-tolerant, distributed startup strategy is specified for initial synchronization of all nodes. In general, anode may enter NORMAL_ACTIVE state via (see Figure 32.9):

- Coldstart path initiating the schedule synchronization (leading coldstart node)
- Coldstart path joining other coldstart nodes (following coldstart node)
- Integration path integrating into an existing communication schedule (all other nodes)

A coldstart attempt begins with the transmission of a collision avoidance symbol (CAS). Only a coldstart node that had transmitted the CAS transmits frames in the first four cycles after the CAS, it is then joined firstly by the other coldstart nodes and afterwards by all other nodes.

A coldstart node has bits TXST and TXSY in the FRSUCC1 register set to "1". Message buffer 0 holds the key slot ID which defines the slot number where the startup frame is send. In the frame header of the startup frame the startup frame indicator bit is set to "1".

In clusters consisting of three or more nodes, at least three nodes shall be configured to be coldstart nodes. In clusters consisting of two nodes, both nodes must be coldstart nodes. At least two fault-free coldstart nodes are necessary for the cluster to startup.

Each startup frame must also be a sync frame; therefore each coldstart node will also be a sync node. The number of coldstart attempts is configured by bits CSA4 to CSA0 in the FRSUCC1 register.

A non-coldstart node requires at least two startup frames from distinct nodes for integration. It may start integration before the coldstart nodes have finished their startup. It will not finish its startup until at least two coldstart nodes have finished their startup.

Both non-coldstart nodes and coldstart nodes start passive integration via the integration path as soon as they receive sync frames from which to derive the TDMA schedule information. During integration, the node has to adapt its own clock to the global clock (rate and offset) and has to make its cycle time consistent with the global schedule observable at the network. Afterwards, these settings are checked for consistency with all available network nodes. The node can only leave the integration phase and actively participate in communication when these checks are passed.

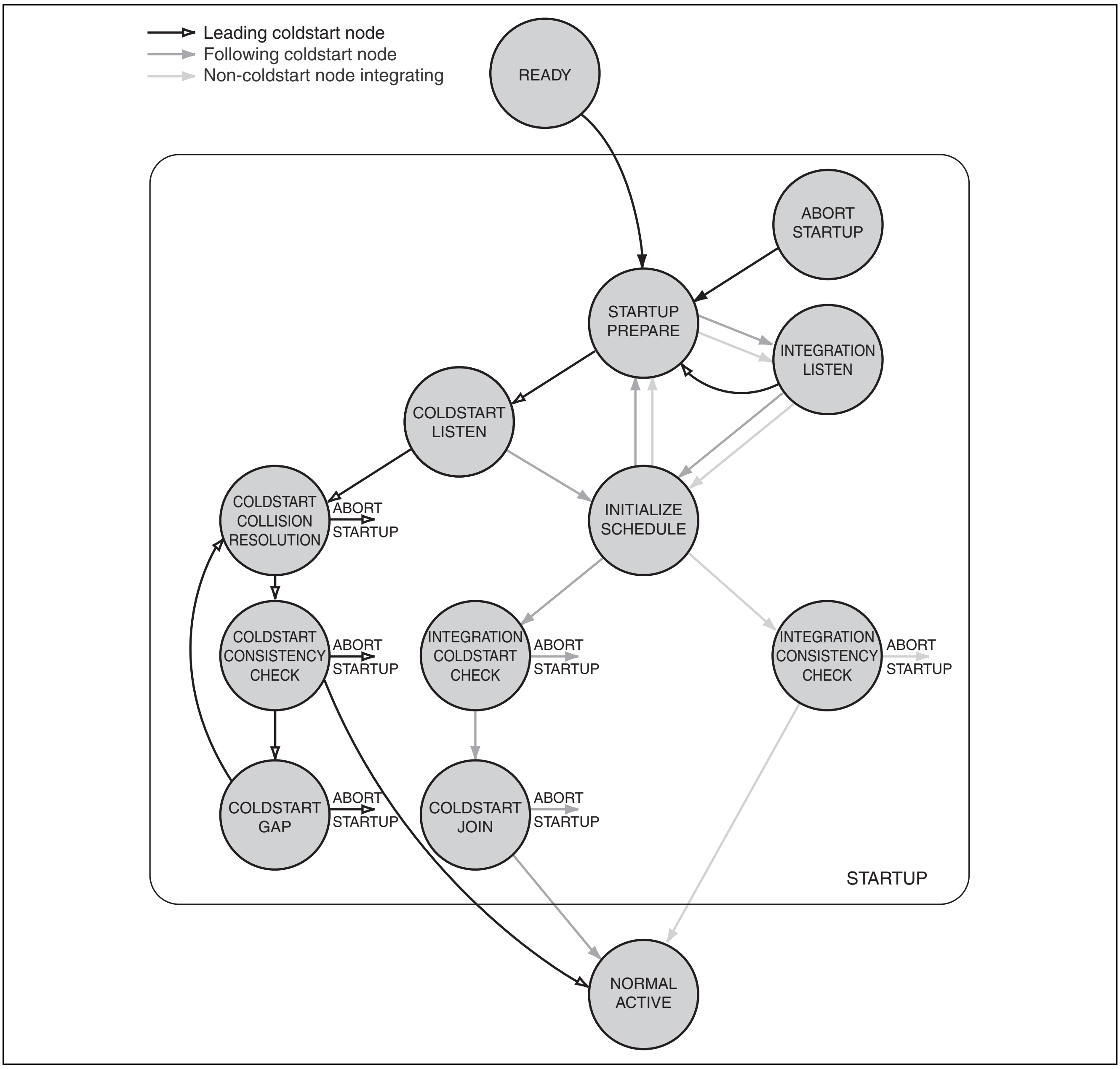


Figure 32.9 State Diagram Time-Triggered Startup

(1) Coldstart Inhibit Mode

In coldstart inhibit mode the node is prevented from initializing the TDMA communication schedule. If the CSI bit in the FRCCSV register is set to "1", the node is not allowed to initialize the cluster communication, i.e. entering the coldstart path is prohibited. The node is allowed to integrate to a running cluster or to transmit startup frames after another coldstart node started the initialization of the cluster communication.

The coldstart inhibit bit (the CSI bit in the FRCCSV register) is set to "1" whenever the POC enters READY state. The bit has to be set to "0" under control of the CPU by CHI command ALLOW_COLDSTART (bits CMD3 to CMD0 in the FRSUCC1 = B'1001).

(2) Startup Timeouts

The CC supplies two different µT timers supporting two timeout values, startup timeout and startup noise timeout. The two timers are started when the CC enters the COLDSTART_LISTEN state. That expiration of either of these timers causes the node to leave the initial sensing phase (COLDSTART_LISTEN state) with the intention of starting up communication.

Note: • The startup and startup noise timers are identical with the wakeup and wakeup noise timers and use the same configuration values (bits LT20 to LT0 in the FRSUCC2 register and LTN3 to LTN0 bit in the FRSUCC2 register)

Startup Timeout

The startup timeout limits the listen time used by a node to determine if there is already communication between other nodes or at least one coldstart node actively requesting the integration of others. The startup timer is configured by programming bits LT20 to LT0 in the FRSUCC2 register.

The startup timeout is: pdListenTimeout = bits LT20 to LT0 in the FRSUCC2 register

The startup timer is restarted upon:

- Entering the COLDSTART_LISTEN state
- Both channels reaching idle state while in COLDSTART_LISTEN state

The startup timer is stopped:

- If communication channel activity is detected on one of the configured channels while the node is in the COLDSTART_LISTEN state.
- When the COLDSTART_LISTEN state is left

Once the startup timeout expires, neither an overflow nor a cyclic restart of the timer is performed.
The timer status is kept for further processing by the startup state machine.

Startup Noise Timeout

At the same time the startup timer is started for the first time (transition from STARTUP_PREPARE state to COLDSTART_LISTEN state), the startup noise timer is started. This additional timeout is used to improve reliability of the startup procedure in the presence of noise. The startup noise timeout is configured by programming bits LTN3 to LTN0 in the FRSUCC2 register.

The startup noise timeout is:
pdListenTimeout × gListenNoise = bits LT20 to LT0 in the FRSUCC2 register × (bits LTN3 to LTN0 in the FRSUCC2 register + 1)

The startup noise timer is restarted upon:

- Entering the COLDSTART_LISTEN state
- Reception of correctly decoded headers or CAS symbols while the node is in COLDSTART_LISTEN state

The startup noise timer is stopped when the COLDSTART_LISTEN state is left.

Once the startup noise timeout expires, neither an overflow nor a cyclic restart of the timer is performed. The status is kept for further processing by the startup state machine. Since the startup noise timer won't be restarted when random channel activity is sensed, this timeout defines the fall-back solution that guarantees that a node will try to start up the communication cluster even in the presence of noise.

(3) Path of Leading Coldstart Node (Initiating Coldstart)

When a coldstart node enters COLDSTART_LISTEN, it listens to its attached channels.

If no communication is detected, the node enters the COLDSTART_COLLISION_RESOLUTION state and commences a coldstart attempt. The initial transmission of a CAS symbol is succeeded by the first regular cycle. This cycle has the number zero.

From cycle zero on, the node transmits its startup frame. Since each coldstart node may perform a coldstart attempt, it may occur that several nodes simultaneously transmit the CAS symbol and enter the coldstart path. This situation is resolved during the first four cycles after CAS transmission.

As soon as a node that initiates a coldstart attempt receives a CAS symbol or a frame header during these four cycles, it re-enters the COLDSTART_LISTEN state. Thereby, only one node remains in this path. In cycle four, other coldstart nodes begin to transmit their startup frames.

After four cycles in COLDSTART_COLLISION_RESOLUTION state, the node that initiated the coldstart enters the COLDSTART_CONSISTENCY_CHECK state. It collects all startup frames from cycle four and five and performs the clock correction. If the clock correction does not deliver any errors and it has received at least one valid startup frame pair, the node leaves COLDSTART_CONSISTENCY_CHECK and enters NORMAL_ACTIVE state.

The number of coldstart attempts that a node is allowed to perform is configured by bits CSA4 to CSA0 in the FRSUCC1 register. The number of remaining coldstarts attempts can be read from bits RCA4 to RCA0 in the FRCCSV register. The number of remaining coldstart attempts is reduced by one for each attempted coldstart. A node may enter the COLDSTART_LISTEN state only if this value is larger than one and it may enter the COLDSTART_COLLISION_RESOLUTION state only if this value is larger than zero. If the number of coldstart attempts is one, coldstart is inhibited but integration is still possible.

(4) Path of Following Coldstart Node (responding to leading Coldstart Node)

When a coldstart node enters the COLDSTART_LISTEN state, it tries to receive a valid pair of startup frames to derive its schedule and clock correction from the leading coldstart node.

As soon as a valid startup frame has been received the INITIALIZE_SCHEDULE state is entered. If the clock synchronization can successfully receive a matching second valid startup frame and derive a schedule from this, the INTEGRATION_COLDSTART_CHECK state is entered.

In INTEGRATION_COLDSTART_CHECK state it is assured that the clock correction can be performed correctly and that the coldstart node from which this node has initialized its schedule is still available. The node collects all sync frames and performs clock correction in the following double cycle. If clock correction does not signal any errors and if the node continues to receive sufficient frames from the same node it has integrated on, the COLDSTART_JOIN state is entered.

In COLDSTART_JOIN state following coldstart nodes begin to transmit their own startup frames and continue to do so in subsequent cycles. Thereby, the leading coldstart node and the nodes joining it can check if their schedules agree with each other. If the clock correction signals any error, the node aborts the integration attempt. If a node in this state sees at least one valid startup frame during all even cycles in this state and at least one valid startup frame pair during all double cycles in this state, the node leaves COLDSTART_JOIN state and enters NORMAL_ACTIVE state. Thereby it leaves STARTUP at least one cycle after the node that initiated the coldstart.

(5) Path of Non-Coldstart Node

When a non-coldstart node enters the INTEGRATION_LISTEN state, it listens to its attached channels.

As soon as a valid startup frame has been received, the INITIALIZE_SCHEDULE state is entered. If the clock synchronization can successfully receive a matching second valid startup frame and derive a schedule from this, the INTEGRATION_CONSISTENCY_CHECK state is entered.

In INTEGRATION_CONSISTENCY_CHECK state the node verifies that the clock correction can be performed correctly and that enough coldstart nodes (at least 2) are sending startup frames that agree with the node's own schedule. Clock correction is activated, and if any errors are signalled, the integration attempt is aborted.

During the first even cycle in this state, either two valid startup frames or the startup frame of the node that this node has integrated on must be received; otherwise the node aborts the integration attempt.

During the first double-cycle in this state, either two valid startup frame pairs or the startup frame pair of the node that this node has integrated on must be received; otherwise the node aborts the integration attempt.

If after the first double-cycle less than two valid startup frames are received within an even cycle, or less than two valid startup frame pairs are received within a double-cycle, the startup attempt is aborted.

Nodes in this state need to see two valid startup frame pairs for two consecutive double-cycles each to be allowed to leave STARTUP and enter NORMAL_ACTIVE. Consequently, they leave startup at least one double-cycle after the node that initiated the coldstart and only at the end of a cycle with an odd cycle number.

32.16.7 NORMAL_ACTIVE State

As soon as the node that transmitted the first CAS symbol (resolving the potential access conflict and entering STARTUP via coldstart path) and one additional node have entered the NORMAL_ACTIVE state, the startup phase for the cluster has finished. In the NORMAL_ACTIVE state, all configured messages are scheduled for transmission. This includes all data frames as well as the sync frames. Rate and offset measurement is started in all even cycles (even / odd cycle pairs required).

In NORMAL_ACTIVE state the CC supports regular communication functions

- The CC performs transmission and reception on the FlexRay bus as configured
- Clock synchronization is running
- The CPU interface is operational

The CC exits from that state to

- HALT state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0110 (HALT command, at the end of the current cycle)
- HALT state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0111 (FREEZE command, immediately)
- HALT state due to change of the error state from ACTIVE to COMM_HALT
- NORMAL_PASSIVE state due to change of the error state from ACTIVE to PASSIVE
- READY state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0010 (READY command)

32.16.8 NORMAL_PASSIVE State

NORMAL_PASSIVE state is entered from NORMAL_ACTIVE state when the error state changes from ACTIVE to PASSIVE.

In NORMAL_PASSIVE state, the node is able to receive all frames (node is fully synchronized and performs clock synchronization). Contrary to the NORMAL_ACTIVE state, the node does not actively participate in communication, i.e. neither symbols nor frames are transmitted.

In NORMAL_PASSIVE state

- The CC performs reception on the FlexRay bus
- The CC does not transmit any frames or symbols on the FlexRay bus
- Clock synchronization is running
- The CPU interface is operational

The CC exits from this state to

- HALT state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0110 (HALT command, at the end of the current cycle)
- HALT state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0111 (FREEZE command, immediately)
- HALT state due to change of the error state from PASSIVE to COMM_HALT
- NORMAL_ACTIVE state due to change of the error state from PASSIVE to ACTIVE. The transition takes place when bits PTAC4 to PTAC0 in the FRCCEV register equal bits PTA4 to PTA0 in the FRSUCC1 register - 1
- To READY state by writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0010 (READY command)

32.16.9 HALT State

In this state all communication (reception and transmission) is stopped.

The CC enters this state

- By writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0110 (HALT command) while the CC is in NORMAL_ACTIVE or NORMAL_PASSIVE state
- By writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0111 (FREEZE command) from all states
- When exiting from NORMAL_ACTIVE state because the clock correction failed counter reached the “maximum without clock correction fatal” limit and the HCSE bit in the FRSUCC1 register is set to "1".
- When exiting from NORMAL_PASSIVE state because the clock correction failed counter reached the “maximum without clock correction fatal” limit and the HCSE bit in the FRSUCC1 register is set to "1".
- The CC exits from this state to DEFAULT_CONFIG state
- By writing bits CMD3 to CMD0 in the FRSUCC1 register = B'0001 (CONFIG command)

When the CC enters HALT state, all configuration and status data is maintained for analysing purposes.

When the CPU writes bits CMD3 to CMD0 in the FRSUCC1 register = B'0110 (HALT command), the CC sets the HRQ bit in the FRCCSV register to "1" and enters HALT state at the next end of cycle.

When the CPU writes bits CMD3 to CMD0 in the FRSUCC1 register = B'0111 (FREEZE command), the CC enters HALT state immediately and sets the FSI bit in the FRCCSV register to "1".

The POC state from which the transition to HALT state took place can be read from bits PSL5 to PSL0 in the FRCCSV register.

32.17 Network Management

The accrued Network Management (NM) vector can be read from registers FRNMV1 to FRNMV3. The CC performs a bit-wise OR operation over all NM vectors out of all received valid NM frames with the Payload Preamble Indicator (PPI) bit which is set to "1". Only static frames may be configured to hold NM information. The CC updates the NM vector at the end of each cycle.

The length of the NM vector can be configured from 0 to 12 bytes by bits NML3 to NML0 in the FRNEMC register. The NM vector length must be configured identically in all nodes of a cluster.

To configure a transmit buffer to send FlexRay frames with the PPI bit set to "1", the PPIT bit in the header section of the respective transmit buffer has to be set to "1" via the PPIT bit in the FRWRHS1 register. In addition the CPU has to write the NM information to the data section of the respective transmit buffer.

The evaluation of the NM vector has to be done by the application running on the CPU.

Notes:
- In case a message buffer is configured for transmission / reception of network management frames, the payload length configured in header 2 of that message buffer should be equal or greater than the length of the NM vector configured by bits NML3 to MML0 in the FRNEMC register.
- When the CC transits to HALT state, the cycle count is not incremented and therefore the NM vector is not updated. In this case registers NMV1 to NMV3 hold the value from the cycle before.

32.18 Filtering and Masking

Filtering is done by comparison of the configuration of assigned message buffers against actual slot and cycle counter values and channel ID (channel A, B). A message buffer is only updated / transmitted if the required matches occur.

Filtering is done on:

- Slot Counter
- Cycle Counter
- Channel ID

The following filter combinations for acceptance / transmit filtering are allowed:

- Slot Counter + Channel ID
- Slot Counter + Cycle Counter + Channel ID

All configured filters must match in order to store a received message in a message buffer.

Note: • For the FIFO the acceptance filter is configured by the FIFO Rejection Filter and the FIFO Rejection Filter Mask.

A message will be transmitted in the time slot corresponding to the configured frame ID on the configured channel(s). If cycle counter filtering is enabled the configured cycle filter value must also match.

32.18.1 Slot Counter Filtering

Every transmit and receive buffer contains a frame ID stored in the header section. This frame ID is compared against the actual slot counter value in order to assign receive and transmit buffers to the corresponding slot.

If two or more message buffers are configured with the same frame ID, and if they have a matching cycle counter filter value for the same slot, then the message buffer with the lowest message buffer number is used.

32.18.2 Cycle Counter Filtering

Cycle counter filtering is based on the notion of a cycle set. For filtering purposes, a match is detected if any one of the elements of the cycle set is matched. The cycle set is defined by the cycle code field in header section 1 of each message buffer.

If message buffer 0 resp. 1 is configured to hold the startup / sync frame or the single slot frame by bits TXST, TXSY, and TSM in the FRSUCC1 register, cycle counter filtering for message buffer 0 resp. 1 must be disabled.

Note: • Sharing of a static time slot via cycle counter filtering between different nodes of a FlexRay network is not allowed.

The set of cycle numbers belonging to a cycle set is determined as described in table 32.10.

Table 32.10 Definition of Cycle Set

Cycle Code	Matching Cycle Counter Values
B'000000x	all Cycles
B'000001c	every second Cycle at (Cycle Count)mod2 = c
B'00001cc	every fourth Cycle at (Cycle Count)mod4 = cc
B'0001ccc	every eighth Cycle at (Cycle Count)mod8 = ccc
B'001cccc	every sixteenth Cycle at (Cycle Count)mod16 = cccc
B'01ccccc	every thirty-second Cycle at (Cycle Count)mod32 = ccccc
B'1cccccc	every sixty-fourth Cycle at (Cycle Count)mod64 = cccccc

Table 32.11 below gives some examples for valid cycle sets to be used for cycle counter filtering:

Table 32.11 Examples for Valid Cycle Sets

Cycle Code	Matching Cycle Counter Values
B'0000011	1, 3, 5, 7, ... 63
B'0000100	0, 4, 8, 12, ... 60
B'0001110	6, 14, 22, 30, ... 62
B'0011000	8 ,24, 40, 56
B'0100011	3, 35
B'1001001	9

The received message is stored only if the cycle counter value of the cycle during which the message is received matches an element of the receive buffer’s cycle set. Other filter criteria must also be met.

The content of a transmit buffer is transmitted on the configured channel(s) when an element of the cycle set matches the current cycle counter value. Other filter criteria must also be met.

32.18.3 Channel ID Filtering

There is a 2-bit channel filtering field (CHA, CHB) located in the header section of each message buffer in the Message RAM. It serves as a filter for receive buffers, and as a control field for transmit buffers (see table 32.12).

Table 32.12 Channel Filtering Configuration

CHA	CHB	Transmit Buffer(transmit frame)	Receive Buffer(store valid receive frame)
1	1	on both channels (static segment only)	received on channel A or B (store first semantically valid frame, static segment only)
1	0	on channel A	received on channel A
0	1	on channel B	received on channel B
0	0	no transmission	ignore frame

The contents of a transmit buffer is transmitted on the channels specified in the channel filtering field when the slot counter filtering and cycle counter filtering criteria are also met. Only in static segment a transmit buffer may be set up for transmission on both channels (CHA and CHB set to "1").

Valid received frames are stored if they are received on the channels specified in the channel filtering field when the slot counter filtering and cycle counter filtering criteria are also met. Only in static segment a receive buffer may be setup for reception on both channels (CHA and CHB set to "1").

Note: • If a message buffer is configured for the dynamic segment and both bits of the channel filtering field are set to "1", no frames are transmitted resp. received frames are ignored (same function as the CHA bit = the CHB bit = 0).

32.18.4 FIFO Filtering

For FIFO filtering there is one rejection filter and one rejection filter mask available. The FIFO filter consists of channel filter (bits CH1 to CH0 in the FRFRF register), frame ID filter (bits FID10 to FID0 in the FRFRF register), and cycle counter filter (bits CYF6 to CYF0 in the FRFRF register). Registers FRFRF and FRFRFM can be configured in CONFIG state only. The filter configuration in the header section of message buffers belonging to the FIFO is ignored.

The 7-bit cycle counter filter determines the cycle set to which frame ID and channel rejection filter are applied. In cycles not belonging to the cycle set specified by bits CYF6 to CYF0 in the FRFRF register, all frames are rejected.

A valid received frame is stored in the FIFO if channel ID, frame ID, and cycle counter are not rejected by the configured rejection filter and rejection filter mask, and if there is no matching dedicated receive buffer.

32.19 Transmit Process

32.19.1 Static Segment

For the static segment, if there are several messages pending for transmission, the message with the frame ID corresponding to the next sending slot is selected for transmission.

The data section of transmit buffers assigned to the static segment can be updated until the end of the preceding time slot. This means that a transfer from the Input Buffer has to be started by writing to the Input Buffer Command Request register latest at this time.

32.19.2 Dynamic Segment

In the dynamic segment, if several messages are pending, the message with the highest priority (lowest frame ID) is selected next. In the dynamic segment different slot counter sequences on channel A and channel B are possible (concurrent sending of different frame IDs on both channels).

The data section of transmit buffers assigned to the dynamic segment can be updated until the end of the preceding slot. This means that a transfer from the Input Buffer has to be started by writing to the Input Buffer Command Request register latest at this time.

The start of latest transmit configured by bits SLT12 to SLT0 in the FRMHDC register defines the maximum minislot value allowed before inhibiting new frame transmission in the dynamic segment of the current cycle.

32.19.3 Transmit Buffers

FlexRay message buffers can be configured as transmit buffers by programming the CFG bit in the header section of the respective message buffer to "1" via the FRWRHS1 register.

There exist the following possibilities to assign a transmit buffer to the CC channels:

- Static segment: channel A or channel B, channel A and channel B
- Dynamic segment: channel A or channel B

Message buffer 0 resp. 1 is dedicated to hold the startup frame, the sync frame, or the designated single slot frame as configured by bits TXST, TXSY, and TSM in the FRSUCC1 register. In this case, it can be reconfigured in CONFIG state only. This ensures that any node transmits at most one startup / sync frame per communication cycle. Transmission of startup / sync frames from other message buffers is not possible.

All other message buffers configured for transmission in static or dynamic segment are reconfigurable during runtime depending on the configuration of bits SEC1 to SEC0 in the FRMRC register (see section 32.22.1, Reconfiguration of Message Buffers). Due to the organization of the data partition in the Message RAM (reference by data pointer), reconfiguration of the configured payload length and the data pointer in the header section of a message buffer may lead to erroneous configurations.

If a message buffer is reconfigured (header section updated) during runtime, it may happen that this message buffer is not send out in the respective communication cycle.

The CC does not have the capability to calculate the header CRC. The CPU is supposed to provide the header CRCs for all transmit buffers. If network management is required, the CPU has to set the PPIT bit in the header section of the respective message buffer to "1" and write the network management information to the data section of the message buffer (see section 32.17, Network Management).

The payload length field configures the payload length in 2-byte words. If the configured payload length of a static transmit buffer is shorter than the payload length configured for the static segment by bits SFDL6 to SFDL0 in the

FRMHDC register, the CC generates padding bytes to ensure that frames have proper physical length. The padding pattern is logical "0".

Note: • In case of an odd payload length (PLC = 1, 3, 5,...) the application has to write "0" to the last 16 bit of the message buffers data section to ensure that the padding pattern is all "0".

Each transmit buffer provides a transmission mode flag (the TXM bit) that allows the CPU to configure the transmission mode for the transmit buffer. If this bit is set to "1", the transmitter operates in the single-shot mode. If this bit is set to "0", the transmitter operates in the continuous mode.

In single-shot mode the CC sets the respective TXR flag to "0" after transmission has completed. Now the CPU may update the transmit buffer.

In continuous mode, the CC does not set the respective transmission request flag (the TXR bit) to "0" after successful transmission. In this case a frame is sent out each time the filter criteria match. The TXR flag can be set to "0" by the CPU by writing the respective message buffer number to the FRIBCR register while the STXRH bit in the FRIBCM register is set to "0".

If two or more transmit buffers meet the filter criteria simultaneously, the transmit buffer with the lowest message buffer number will be transmitted in the respective slot.

32.19.4 Frame Transmission

The following steps are required to prepare a message buffer for transmission:

- Configure the transmit buffer in the Message RAM via registers FRWRHS1, FRWRHS2, and FRWRHS3
- Write the data section of the transmit buffer via the FRWRDSi register
- Transfer the configuration and message data from Input Buffer to the Message RAM by writing the number of the target message buffer to the FRIBCR register
- If configured in the FRIBCM register, the transmission request flag (the TXR bit) for the respective message buffer will be set to "1" as soon as the transfer has completed, and the message buffer is ready for transmission.
- Check whether the message buffer has been transmitted by checking the respective TXR bit (TXR = 0) in registers FRTRXQ1 to FRTRXQ4 (single-shot mode only).

After transmission has completed, the respective TXR flag in registers FRTXRQ1 to FRTXRQ4 is set to "0" (single-shot mode), and, if the MBI bit in the header section of the message buffer is set to "1", the TXI bit in the FRSIR register is set to "1". If enabled, an interrupt is generated.

32.19.5 Null Frame Transmission

If in static segment the CPU does not set the transmission request flag to "1" before transmit time, and if there is no other transmit buffer with matching filter criteria, the CC transmits a null frame with the null frame indication bit set to "0" and the payload data set to "0".

In the following cases the CC transmits a null frame:

- If the message buffer with the lowest message buffer number matching the filter criteria does not have its transmission request flag which is set to1 (the TXR bit = 0).
- No transmit buffer configured for the slot has a cycle counter filter that matches the current cycle. In this case, no message buffer status (MBS) is updated.

Null frames are not transmitted in the dynamic segment.

32.20 Receive Process

32.20.1 Dedicated Receive Buffers

A portion of the FlexRay message buffers can be configured as dedicated receive buffers by programming the CFG bit in the header section of the respective message buffer to "0" via the FRWRHS1 register.

The following possibilities exist to assign a receive buffer to the CC channels:

- Static segment: channel A or channel B, channel A and channel B (the CC stores the first semantically valid frame)
- Dynamic segment: channel A or channel B

The CC transfers the payload data of valid received messages from the shift register of the FlexRay channel protocol controller (channel A or B) to the receive buffer with the matching filter configuration. A receive buffer stores all frame elements except the frame CRC.

All message buffers configured for reception in static or dynamic segment are reconfigurable during runtime depending on the configuration of bits SEC1 to SEC0 in the FRMRC register (see section 32.22.1, Reconfiguration of Message Buffers). If a message buffer is reconfigured (header section updated) during runtime it may happen that in the respective communication cycle a received message is lost.

If two or more receive buffers meet the filter criteria simultaneously, the receive buffer with the lowest message buffer number is updated with the received message.

32.20.2 Frame Reception

The following steps are required to prepare a dedicated message buffer for reception:

- Configure the receive buffer in the Message RAM via registers FRWRHS1, FRWRHS2, and FRWRHS3
- Transfer the configuration from Input Buffer to the Message RAM by writing the number of the target message buffer to the FRIBCR register

Once these steps are performed, the message buffer functions as an active receive buffer and participates in the internal acceptance filtering process which takes place every time the CC receives a message. The first matching receive buffer is updated from the received message.

If a valid payload segment was stored in the data section of a message buffer, the respective ND flag in registers FRNDAT1 to FRNDAT4 is set to "1", and, if the MBI bit in the header section of that message buffer is set to "1", the RXI bit in the FRSIR register is set to "1". If enabled, an interrupt is generated.

In case that the ND bit was already set to "1" when the Message Handler updates the message buffer, (MBS) of the respective message buffer is set to "1" and the unprocessed message data is lost.

If no frame, a null frame, or a corrupted frame was received in a slot, the data section of the message buffer configured for this slot is not updated. In this case only the respective message buffer status (MBS) is updated.

When the Message Handler changed the message buffer status (MBS) in the header section of a message buffer, the respective MBC flag in registers FRMBSC1 to FRMBSC4 is set to "1", and if the MBI bit in the header section of that message buffer is set to "1", the MBSI bit in the FRSIR register is set to "1". If enabled an interrupt is generated.

If the payload length of a received frame (bits PLR6 to PLR0) is longer than the value programmed by bits PLC6 to PLC0 in the header section of the respective message buffer, the data field stored in the message buffer is truncated to that length.

To read a receive buffer from the Message RAM via the Output Buffer, proceed as described in section 32.22.2 (2), Data Transfer from Message RAM to Output Buffer.

Note: • Bits ND and MBC are automatically set to "0" by the Message Handler when the payload data and the header of a received message have been transferred to the Output Buffer, respectively.

32.20.3 Null Frame Reception

The payload segment of a received null frame is not copied into the matching dedicated receive buffer. If a null frame has been received, only the message buffer status (MBS) of the matching message buffer is updated from the received null frame. All bits in header 2 and 3 of the matching message buffer remain unchanged. They are updated from received data frames only.

When the Message Handler changed the message buffer status (MBS) in the header section of a message buffer, the respective MBC flag in registers FRMBSC1 to FRMBSC4 is set to "1", and if the MBI bit in the header section of that message buffer is set to "1", the MBSI flag in the FRSIR register is set to "1". If enabled, an interrupt is generated.

32.21 FIFO Function

32.21.1 Description

A group of the message buffers can be configured as a cyclic First-In-First-Out (FIFO) buffer. The group of message buffers belonging to the FIFO is contiguous in the register map starting with the message buffer referenced by bits FFB7 to FFB0 in the FRMRC register and ending with the message buffer referenced by bits LCB7 to LCB0 in the FRMRC register. Up to 127 message buffers can be assigned to the FIFO.

Every valid incoming message not matching with any dedicated receive buffer but passing the programmable FIFO filter is stored into the FIFO. In this case frame ID, payload length, receive cycle count, and the message buffer status (MBS) of the addressed FIFO message buffer are overwritten with frame ID, payload length, receive cycle count, and the status from the received frame. The RFNE bit in the FRSIR register with 1 shows that the FIFO is not empty, the RFCL bit in the FRSIR register is set to "1" when the receive FIFO fill level (bits RFFL7 to RFFL0 in the FRFSR register) is equal or greater than the critical level as configured by bits CL7 to CL0 in the FRFCL register, the RFO bit in the FREIR register with 1 shows that a FIFO overrun has been detected. If enabled, interrupts are generated.

If null frames are not rejected by the FIFO rejection filter, the null frames will be treated like data frames when they are stored into the FIFO.

There are two index registers associated with the FIFO. The PUT Index Register (the PIDX register) is an index to the next available location in the FIFO. When a new message has been received it is written into the message buffer addressed by the PIDX register. The PIDX register is then incremented and addresses the next available message buffer. If the PIDX register is incremented past the highest numbered message buffer of the FIFO, the PIDX register is loaded with the number of the first (lowest numbered) message buffer in the FIFO chain. The GET Index Register (the GIDX register) is used to address the next message buffer of the FIFO to be read. The GIDX register is incremented after transfer of the contents of a message buffer belonging to the FIFO to the Output Buffer. The PUT Index Register and the GET Index Register are not accessible by the CPU.

The FIFO is completely filled when the PUT index (the PIDX register) reaches the value of the GET index (the GIDX register). When the next message is written to the FIFO before the oldest message has been read, both registers PUT index and GET index are incremented and the new message overwrites the oldest message in the FIFO. This will set the FIFO overrun flag (the RFO bit in the FREIR register) to "1".

A FIFO non empty status is detected when the PUT index (the PIDX register) differs from the GET index (the GIDX register). In this case the RFNE flag in the FRSIR register is set to "1". This indicates that there is at least one received message in the FIFO. The FIFO empty, FIFO not empty, and the FIFO overrun states are explained in figure 32.10 for a three message buffer FIFO.

The programmable FIFO Rejection Filter (the FRFRF register) defines a filter pattern for messages to be rejected. The FIFO filter consists of channel filter, frame ID filter, and cycle counter filter. If the RSS bit in the FRFRF register is set to "1" (default), all messages received in the static segment are rejected by the FIFO. If the RFN bit in the FRFRF register is set to "1" (default), received null frames are not stored in the FIFO.

The FIFO Rejection Filter Mask (the FRFRFM register) specifies which bits of the frame ID filter in the FIFO Rejection Filter register are marked “don't care” for rejection filtering.

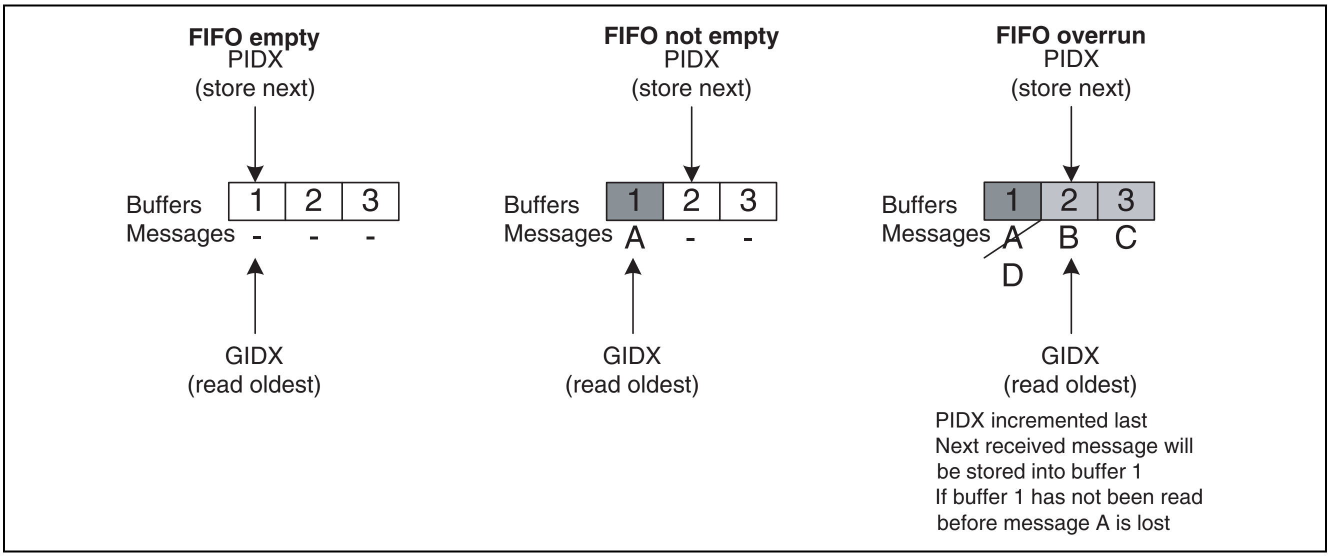


Figure 32.10 FIFO Status: Empty, Not Empty, Overrun

32.21.2 Configuration of the FIFO

(Re)configuration of message buffers belonging to the FIFO is only possible when the CC is in CONFIG state. While the CC is in CONFIG state, the FIFO function is not available.

For all message buffers belonging to the FIFO the payload length configured should be programmed to the same value via bits PLC6 to PLC0 in the FRWRHS2 register. The data pointer to the first 32-bit word of the data section of the respective message buffer in the Message RAM has to be configured via bits DP10 to DP0 in the FRWRHS3 register.

All information required for acceptance filtering is taken from the FIFO rejection filter and the FIFO rejection filter mask. The values configured in the header sections of the message buffers belonging to the FIFO are, with exception of bits DP and PLC, irrelevant.

Notes:
- It is recommended to program the MBI bits of the message buffers belonging to the FIFO to "0" via the MBI bit in the FRWRHS1 register to avoid generation of RX interrupts.
- If the payload length of a received frame is longer than the value programmed by bits PLC6 to PLC0 in the FRWRHS2 register in the header section of the respective message buffer, the data field stored in a message buffer of the FIFO is truncated to that length.

32.21.3 Access to the FIFO

For FIFO access outside DEFAULT_CONFIG and CONFIG state, the CPU has to trigger a transfer from the Message RAM to the Output Buffer by writing the number of the first message buffer of the FIFO (referenced by bits FFB7 to FFB0 in the FRMRC register) to the FROBCR register. The Message Handler then transfers the message buffer addressed by the GET Index Register (the GIDX register) to the Output Buffer. After this transfer the GET Index Register (the GIDX register) is incremented.

32.22 Message Handling

The Message Handler controls data transfers between the Input / Output Buffer and the Message RAM and between the Message RAM and the two Transient Buffer RAMs. All accesses to the internal RAMs are 32+1 bit accesses. The additional bit is used for parity checking.

Access to the message buffers stored in the Message RAM is done under control of the Message Handler state machine. This avoids conflicts between accesses of the two FlexRay channel protocol controllers and the CPU to the Message RAM.

Frame IDs of message buffers assigned to the static segment have to be in the range from 1 to bits NSS9 to NSS0 in the FRGTUC7 register. Frame IDs of message buffers assigned to the dynamic segment have to be in the range from bits NSS9 to NSS0 in the FRGTUC7 register + 1 to 2047.

Received messages with no matching dedicated receive buffer (static or dynamic segment) are stored in the receive FIFO (if configured) if they pass the FIFO rejection filter.

32.22.1 Reconfiguration of Message Buffers

In case that an application needs to operate with more than 128 different messages, static and dynamic message buffers may be reconfigured during FlexRay operation. This is done by updating the header section of the respective message buffer via Input Buffer registers (FRWRHS1 to FRWRHS3).

Reconfiguration has to be enabled via control bits SEC1 to SEC0 in the FRMRC register in the Message RAM Configuration register.

If a message buffer has not been transmitted / updated from a received frame before reconfiguration starts, the respective message is lost.

The point in time when a reconfigured message buffer is ready for transmission / reception according to the reconfigured frame ID depends on the actual state of the slot counter when the update of the header section has completed. Therefore it may happen that a reconfigured message buffer is not transmitted / updated from a received frame in the cycle where it was reconfigured.

The Message RAM is scanned according to table 32.13 below:

Table 32.13 Scan of Message RAM

Start of Scan in Slot	Scan for Slots
1	2 ... 15, 1 (next cycle)
8	16 ... 23, 1 (next cycle)
16	24 ... 31, 1 (next cycle)
24	32 ... 39, 1 (next cycle)
...	...

A Message RAM scan is terminated with the start of NIT regardless whether it has completed or not. The scan of the Message RAM for slots 2 to 15 starts at the beginning of slot 1 of the actual cycle.
The scan of the Message RAM for slot 1 is done in the cycle before by checking in parallel to each scan of the Message RAM whether there is a message buffer configured for slot 1 of the next cycle.

The number of the first dynamic message buffer is configured by bits FDB7 to FDB0 in the FRMRC register. In case a Message RAM scan starts while the CC is in dynamic segment, the scan starts with the message buffer number configured by bits FDB7 to FDB0 in the FRMRC register.

In case a message buffer should be reconfigured to be used in slot 1 of the next cycle, the following has to be considered:

- If the message buffer to be reconfigured for slot 1 is part of the “Static Buffers”, it will only be found if it is reconfigured before the last Message RAM scan in the static segment of the actual cycle evaluates this message buffer.
- If the message buffer to be reconfigured for slot 1 is part of the “Static + Dynamic Buffers”, it will be found if it is reconfigured before the last Message RAM scan in the actual cycle evaluates this message buffer.
- The start of NIT terminates the Message RAM scan. In case the Message RAM scan has not evaluated the reconfigured message buffer until this point in time, the message buffer will not be considered for the next cycle.

Note: • Reconfiguration of message buffers may lead to the loss of messages and therefore has to be used very carefully. In worst case (reconfiguration in consecutive cycles) it may happen that a message buffer is never transmitted / updated from a received frame.

32.22.2 CPU Access to Message RAM

The message transfer between Input Buffer and Message RAM as well as between Message RAM and Output Buffer is triggered by the CPU by writing the number of the target / source message buffer to be accessed to the FRIBCR or FROBCR register.

Registers FRIBCM and FROBCM can be used to write / read header and data section of the selected message buffer separately.

If the STXR bit in the FRIBCM register is set to = 1, the transmission request flag (the TXR bit) of the selected message buffer is automatically set to "1" after the message buffer has been updated. If the STXR bit in the FRIBCM register is set to "0", the transmission request flag (the TXR bit) of the selected message buffer is set to "0". This can be used to stop transmission from message buffers operated in continuous mode.

Input Buffer (IBF) and Output Buffer (OBF) are build up as a double buffer structure. One half of this double buffer structure is accessible by the CPU (IBF Host / OBF Host), while the other half (IBF Shadow / OBF Shadow) is accessed by the Message Handler for data transfers between IBF / OBF and Message RAM.

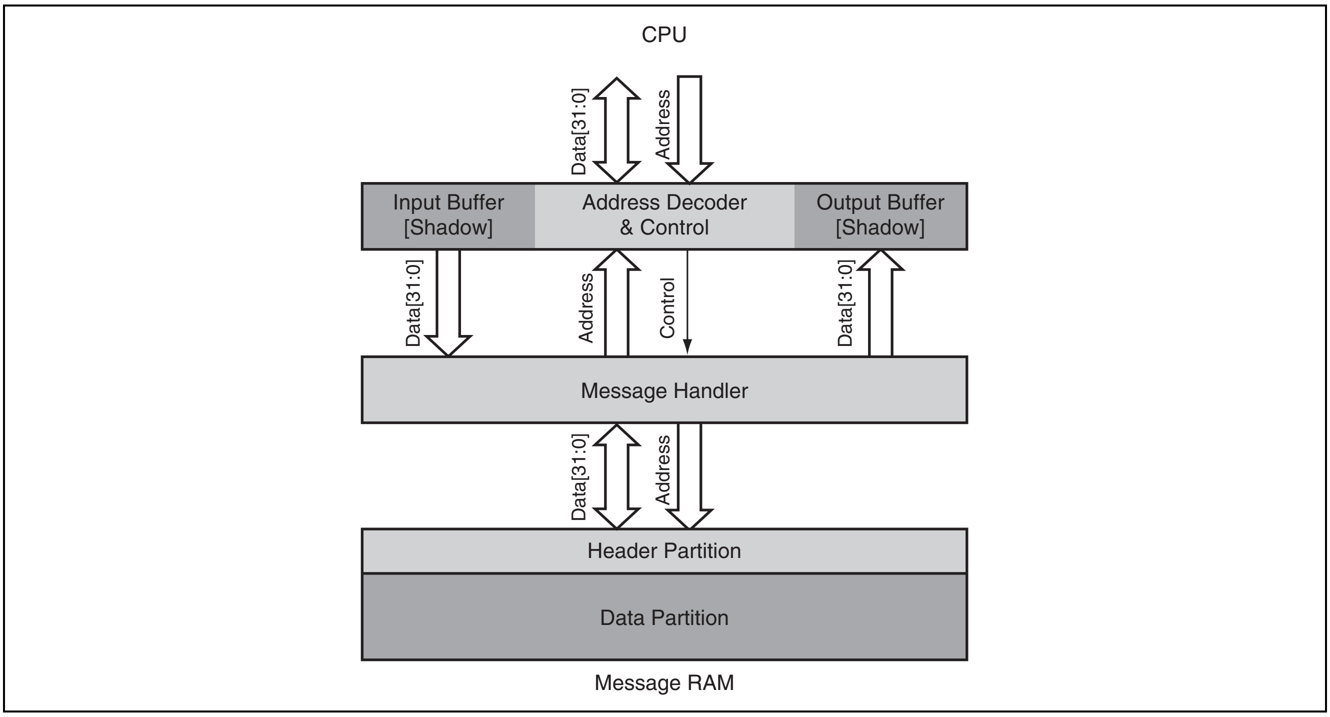


Figure 32.11 CPU Access to Message RAM

(1) Data Transfer from Input Buffer to Message RAM

To configure / update a message buffer in the Message RAM, the CPU has to write the data to the FRWRDSi register and the header to registers FRWRHS1 to FRWRHS3. The specific action is selected by configuring the Input Buffer Command Mask (the FRIBCM register).

When the CPU writes the number of the target message buffer in the Message RAM to bits IBRH6 to IBRH0 in the FRIBCR register, IBF Host and IBF Shadow are swapped (see figure 32.12).

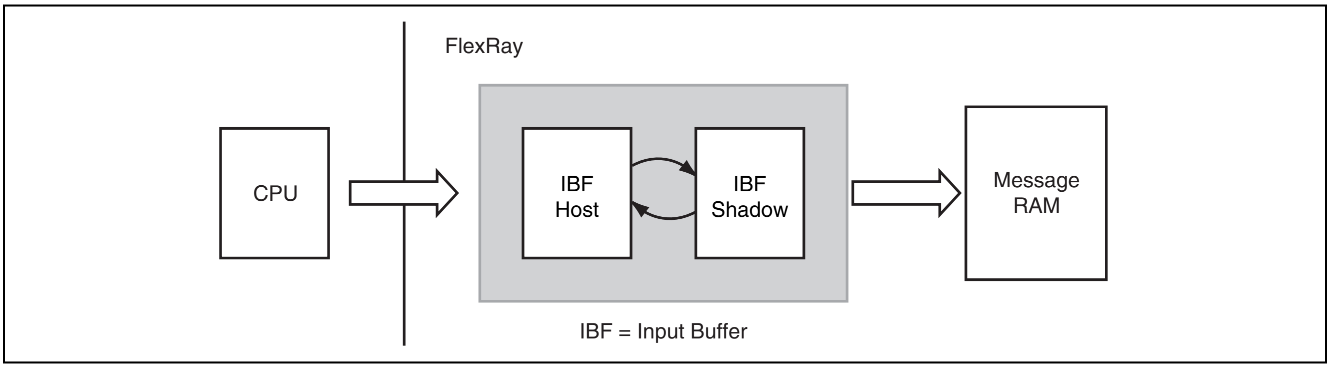


Figure 32.12 Double Buffer Structure Input Buffer

In addition the bits in the FRIBCM and FRIBCR registers are also swapped to keep them attached to the respective IBF section (see figure 32.13).

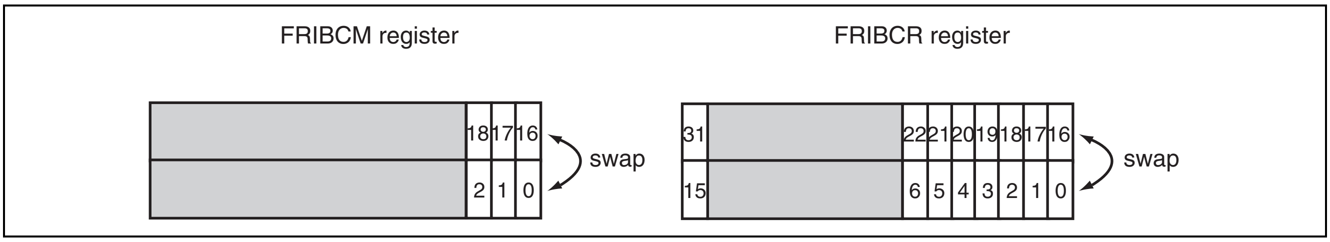


Figure 32.13 Swapping of Bits in Registers FRIBCM and FRIBCR

With this write operation the IBSYS bit in the FRIBCR register is set to "1". The Message Handler then starts to transfer the contents of IBF Shadow to the message buffer in the Message RAM selected by bits IBRS6 to IBRS0 in the FRIBCR register.

While the Message Handler transfers the data from IBF Shadow to the target message buffer in the Message RAM, the CPU may write the next message to IBF Host. After the transfer between IBF Shadow and the Message RAM has completed, the IBSYS bit in the FRIBCR register is set back to "0" and the next transfer to the Message RAM may be started by the CPU by writing the respective target message buffer number to bits IBRH6 to IBRH0 in the FRIBCR register.

If a write access to bits IBRH6 to IBRH0 in the FRIBCR register occurs while the IBSYS bit in the FRIBCR register is "1", the IBSYH bit in the FRIBCR register is set to "1".
After completion of the ongoing data transfer from IBF Shadow to the Message RAM, IBF Host and IBF Shadow are swapped, the IBSYH bit in the FRIBCR register is set to "0", the IBSYS bit in the FRIBCR register remains set to "1", and the next transfer to the Message RAM is started. In addition the message buffer numbers stored under bits IBRH6 to IBRH0 in the FRIBCR register and bits IBRS6 to IBRS0 in the FRIBCR register and the command mask flags are also swapped.

Example of a CPU access sequence:
Configure / update n-th message buffer via IBF

- Wait until the IBSYH bit in the FRIBCR register is set to "0"
- Write data section to the FRWRDSi register
- Write header section to registers FRWRHS1 to FRWRHS3
- Write Command Mask: write bits STXRH, LDSH, and LHSH in the FRIBCM register
- Demand data transfer to target message buffer: write bits IBRH6 to IBRH0 in the FRIBCR register

Configure / update (n+1)th message buffer via IBF

- Wait until the IBSYH bit in the FRIBCR register is set to "0"
- Write data section to the FRWRDSi register
- Write header section to registers FRWRHS1 to FRWRHS3
- Write Command Mask: write bits STXRH, LDSH, and LHSH in the FRIBCM register
- Demand data transfer to target message buffer: write bits IBRH6 to IBRH0 in the FRIBCR register

Note: • Any write access to IBF while the IBSYH bit in the FRIBCR register is "1" will set error flag (IIBA bit in the FREIR register) to "1". In this case the write access has no effect.

Table 32.14 Assignment of Bits in the FRIBCM Register

Pos.	Access	Bit	Function
18	R	STXRS	Set Transmission Request Shadow ongoing or finished
17	R	LDSS	Load Data Section Shadow ongoing or finished
16	R	LHSS	Load Header Section Shadow ongoing or finished
2	R/W	STXRH	Set Transmission Request Host
1	R/W	LDSH	Load Data Section Host
0	R/W	LHSH	Load Header Section Host

Table 32.15 Assignment of Bits in the FRIBCR Register

Pos.	Access	Bit	Function
31	R	IBSYS	IBF Busy Shadow, signals ongoing transfer from IBF Shadow to Message RAM
22 to 16	R	IBRS6 to IBRS0	IBF Request Shadow, number of message buffer currently / lately updated
15	R	IBSYH	IBF Busy Host, transfer request pending for message buffer referenced by bits IBRH6 to IBRH0
6 to 0	R/W	IBRH6 to IBRH0	IBF Request Host, number of message buffer to be updated next

(2) Data Transfer from Message RAM to Output Buffer

To read a message buffer from the Message RAM, the CPU has to write to the FROBCR register to trigger the data transfer as configured in FROBCM register. After the transfer has completed, the CPU can read the transferred data from registers FRRDDSn, FRRDHS1 to FRRDHS3, and FRMBS.

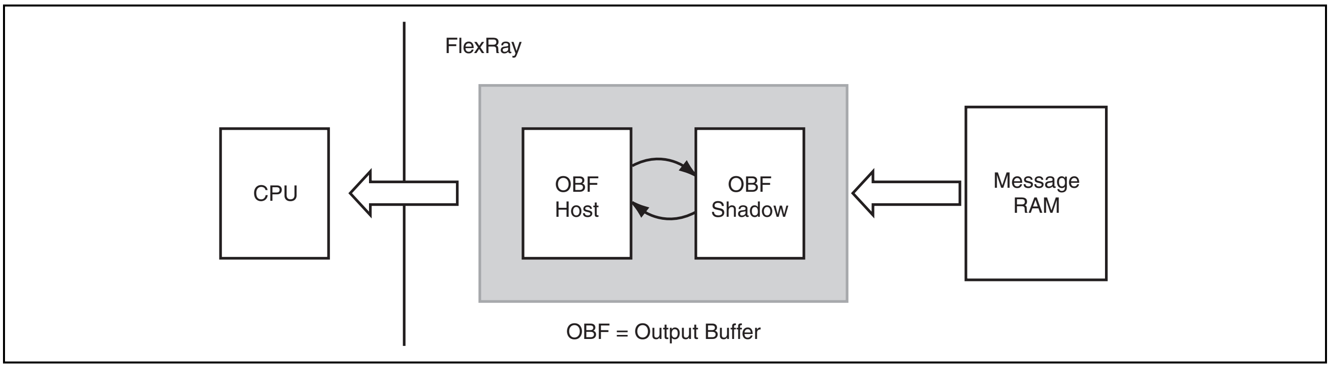


Figure 32.14 Double Buffer Structure Output Buffer

OBF Host and OBF Shadow as well as bits RHSS, RDSS, RHSH, and RDSH in the FROBCM register and bits OBRS6 to OBRS0, OBRH6 to OBRH0 in the FROBCR register are swapped under control of bits VIEW and REQ in the FROBCR register.

Writing the REQ bit in the FROBCR register to "1" copies bits RHSS, RDSS in the FROBCR register, and bits OBRS6 to OBRS0 in the FROBCR register to an internal storage (see figure 32.15).

After setting the REQ bit in the FROBCR register to "1", the OBSYS bit in the FROBCR register is set to "1", and the transfer of the message buffer selected by bits OBRS6 to OBRS0 in the FROBCR register from the Message RAM to OBF Shadow is started. After the transfer between the Message RAM and OBF Shadow has completed, the OBSYS bit in the FROBCR register is set back to "0". Bits REQ and VIEW in the FROBCR register can only be set to "1" while the OBSYS bit in the FROBCR register is "0".

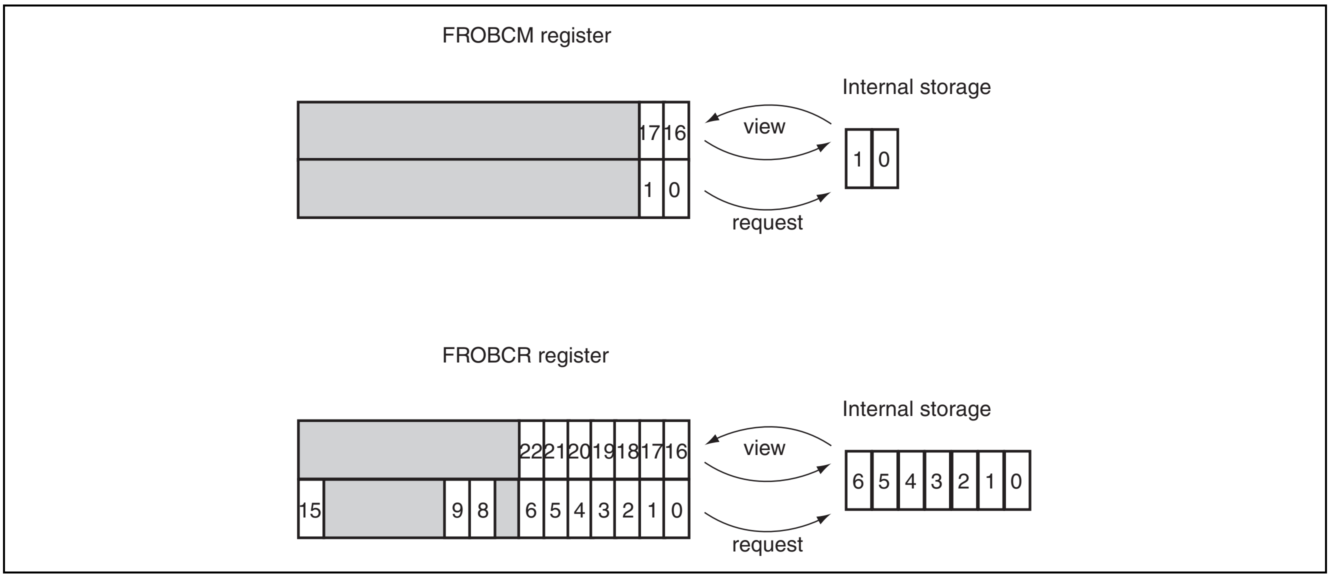


Figure 32.15 Swapping of Bits in Registers FROBCM and FROBCR

OBF Host and OBF Shadow are swapped by setting the VIEW bit in the FROBCR register to "1" while the OBSYS bit in the FROBCR register is "0" (see figure 32.14).

In addition bits OBRH6 to OBRH0 in the FROBCR register and bits RHSH and RDSH in the FROBCM register are swapped with the registers internal storage thus assuring that the message buffer number stored in bits OBRH6 to OBRH0 in the FROBCR register and the mask configuration stored in bits RHSH and RDSH in the FROBCM register matches the transferred data stored in OBF Host (see figure 32.15).

Now the CPU can read the transferred message buffer from OBF Host while the Message Handler may transfer the next message from the Message RAM to OBF Shadow.

If bits REQ and VIEW are set to "1" with the same write access while the OBSYS bit is "0", the OBSYS bit is automatically set to "1" and OBF Shadow and OBF Host are swapped. Additionally, mask bits RDSH and RHSH in the FROBCM register are swapped with the register internal storage to keep them attached to the respective Output Buffer transfer. Afterwards, bits OBRS6 to OBRS0 are copied to the register internal storage, mask bits RDSS and RHSS in the FROBCM register are copied to the FROBCM register internal storage, and the transfer of the selected message buffer from the Message RAM to OBF Shadow is started.
While the transfer is ongoing, the Host can read the message buffer transferred by the previous transfer from OBF Host. When the current transfer between the Message RAM and OBF Shadow is completed, this is signalled by setting the OBSYS bit back to "0".

Example of a CPU access to a single message buffer:
If a single message buffer has to be read out, two separate write accesses to bits REQ and VIEW in the FROBCR register are necessary:

- Wait until the OBSYS bit in the FROBCR register is set to "0"
- Write Output Buffer Command Mask (bits RHSS and RDSS in the FROBCM register)
- Request transfer of message buffer to OBF Shadow by writing bits OBRS6 to OBRS0 in the FROBCR register and the REQ bit in the FROBCR register
- Wait until the OBSYS bit in the FROBCR register is set to "0"
- Toggle OBF Shadow and OBF Host by writing the VIEW bit in the FROBCR register = 1, and the REQ bit = 0

- Read out transferred message buffer by reading registers FRRDDSn, FRRDHS1 to FRRDHS3, and FRMBS

Example of a CPU access sequence:
Request transfer of 1st message buffer to OBF Shadow

- Wait until the OBSYS bit in the FROBCR register is set to "0"
- Write Output Buffer Command Mask (bits RHSS and RDSS in the FROBCM register) for 1st message buffer
- Request transfer of 1st message buffer to OBF Shadow by writing bits OBRS6 to OBRS0 in the FROBCR register and the REQ bit in the FROBCR register.

Toggle OBF Shadow and OBF Host to read out 1st transferred message buffer and request transfer of 2nd message buffer:

- Wait until the OBSYS bit in the FROBCR register is set to "0"
- Write Output Buffer Command Mask (bits RHSS and RDSS in the FROBCM register) for 2nd message buffer
- Toggle OBF Shadow and OBF Host and start transfer of 2nd message buffer to OBF Shadow simultaneously by writing bits OBRS6 to OBRS0 in the FROBCR register of 2nd message buffer, the REQ bit in the FROBCR register, and the ViEW bit in the FROBCR register
- Read out 1st transferred message buffer by reading registers FRRDDSn, FRRDHS1 to FRRDHS3, and FRMBS

Demand access to last requested message buffer without request of another message buffer:

- Wait until the OBSYS bit in the FROBCR register is set to "0"
- Demand access to last transferred message buffer by writing the VIEW bit in the FROBCR register = 1, and the REQ bit = "0"
- Read out last transferred message buffer by reading registers FRRDDSn, FRRDHS1 to FRRDHS3, and FRMBS

Table 32.16 Assignment of Bits in the FROBCM Register

Pos.	Access	Bit	Function
17	R	RDSH	Data Section available for CPU access
16	R	RHSH	Header Section available for CPU access
1	R/W	RDSS	Read Data Section Shadow
0	R/W	RHSS	Read Header Section Shadow

Table 32.17 Assignment of Bits in the FROBCR Register

Pos.	Access	Bit	Function
22 to 16	R	OBRH6 to OBRH0	OBF Request Host, number of message buffer available for CPU access
15	R	OBSYS	OBF Busy Shadow, signals ongoing transfer from Message RAM to OBF Shadow
9	R/W	REQ	Request Transfer from Message RAM to OBF Shadow
8	R/W	VIEW	View OBF Shadow, swap OBF Shadow and OBF Host
6 to 0	R/W	OBRS6 to OBRS0	OBF Request Shadow, number of message buffer for next request

32.22.3 FlexRay Protocol Controller Access to Message RAM

The two Transient Buffer RAMs (TBF A, B) are used to buffer the data for transfer between the two FlexRay Protocol Controllers and the Message RAM.

Each Transient Buffer RAM is build up as a double buffer, able to store two complete FlexRay messages. There is always one buffer assigned to the corresponding Protocol Controller while the other one is accessible by the Message Handler.

If e.g. the Message Handler writes the next message to be send to Transient Buffer Tx, the FlexRay Channel Protocol Controller can access Transient Buffer Rx to store the message it is actually receiving. During transmission of the message stored in Transient Buffer Tx, the Message Handler transfers the last received message stored in Transient Buffer Rx to the Message RAM (if it passes acceptance filtering) and updates the respective message buffer.

Data transfers between the Transient Buffer RAMs and the shift registers of the FlexRay Channel Protocol Controllers are done in words of 32 bit. This enables the use of a 32 bit shift register independent of the length of the FlexRay messages.

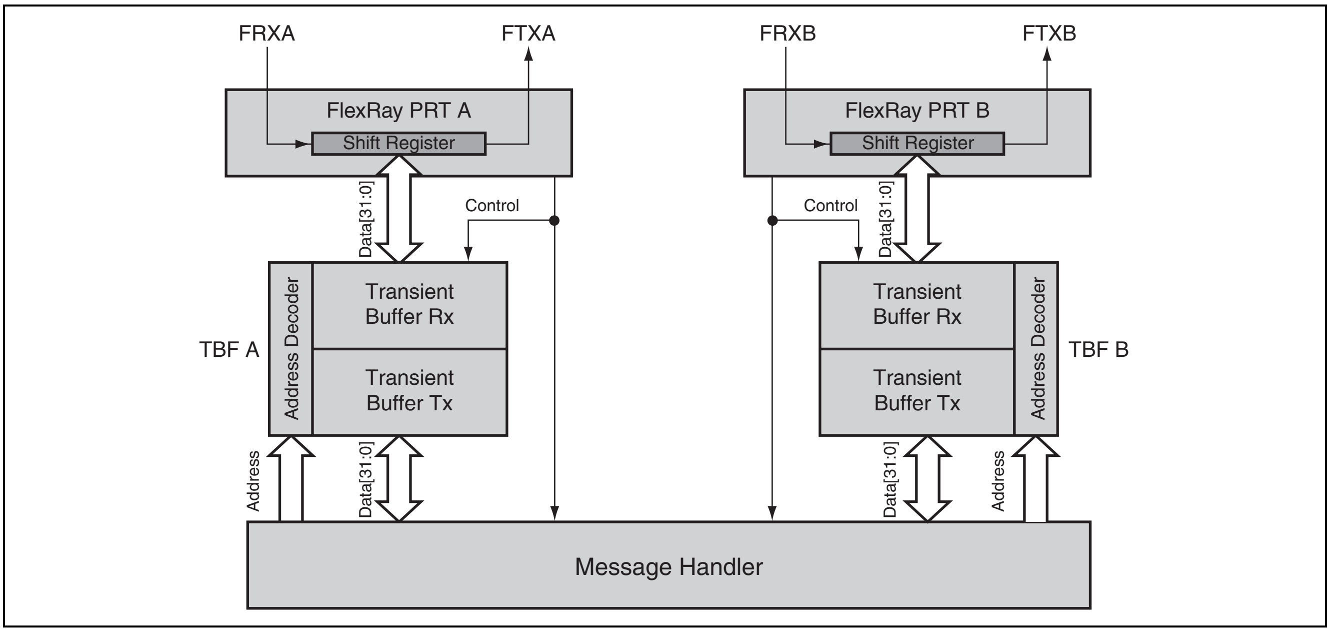


Figure 32.16 Access to Transient Buffer RAMs

32.23 Message RAM

To avoid conflicts between CPU access to the Message RAM and FlexRay message reception / transmission, the CPU cannot directly access the message buffers in the Message RAM. These accesses are handled via the Input and Output Buffers. The Message RAM is able to store up to 128 message buffers depending on the configured payload length.

The Message RAM is organized 2048 × 33 = 67,584 bits. Each 32-bit word is protected by a parity bit.
To achieve the required flexibility with respect to different numbers of data bytes per FlexRay frame (0 to 254), the Message RAM has a structure as shown in figure 32.17.

The data partition is allowed to start at Message RAM word number: (the LCB bit in the FRMRC register + 1) × 4

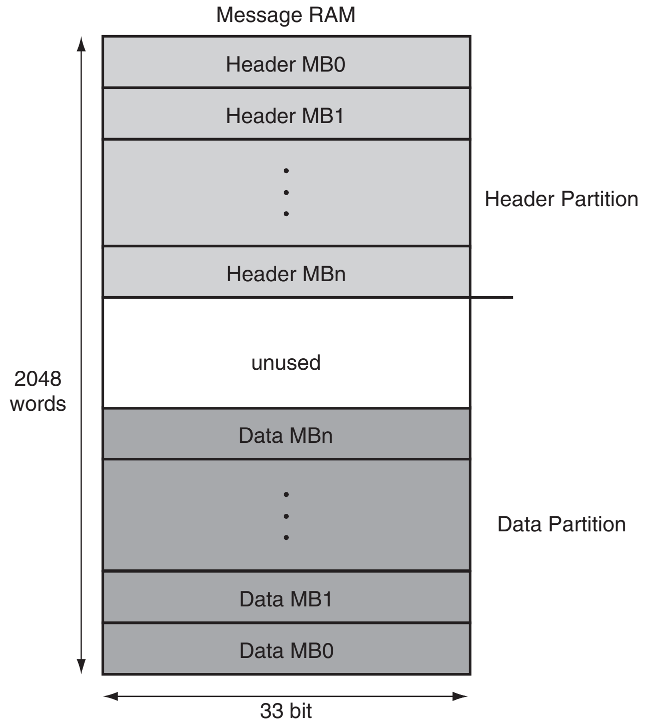


Figure 32.17 Configuration Example of Message Buffers in the Message RAM

Header Partition
Stores header sections of the configured message buffers:

- Supports a maximum of 128 message buffers
- Each message buffer has a header section of four 32+1 bit words
- Header 3 of each message buffer holds the 11-bit data pointer to the respective data section in the data partition

Data Partition
Flexible storage of data sections with different length. Some maximum values are:

- 30 message buffers with 254 byte data section each
- Or 56 message buffers with 128 byte data section each
- Or 128 message buffers with 48 byte data section each

Note: • Header partition + data partition may not occupy more than 2048 33-bit words.

32.23.1 Header Partition

The elements used for configuration of a message buffer as well as the actual message buffer status are stored in the header partition of the Message RAM as listed in figure 32.18 below. Configuration of the header sections of the message buffers is done via IBF (registers FRWRHS1 to FRWRHS3). Read access to the header sections is done via OBF (registers FRRDHS1 to FRRDHS3 + FRMBS). The data pointer has to be calculated by the programmer to define the starting point of the data section for the respective message buffer in the data partition of the Message RAM. The data pointer should not be modified during runtime. For message buffers belonging to the receive FIFO (re)configuration is possible in CONFIG state only.

The header section of each message buffer occupies four 33-bit words in the header partition of the Message RAM. The header of message buffer 0 starts with the first word in the Message RAM.

For transmit buffers the Header CRC has to be calculated by the program.

Payload Length Received (bits PLR6 to PLR0), Receive Cycle Count (bits RCC5 to RCC0), Received on Channel Indicator (the RCI bit), Startup Frame Indicator (the SFI bit), Sync Frame Indicator (the SYN bit), Null Frame Indicator (the NFI bit), Payload Preamble Indicator (the PPI bit), and Reserved Bit (the RES bit) are updated from received valid data frames only.

Header word 3 of each configured message buffer holds the respective Message Buffer Status (MBS).

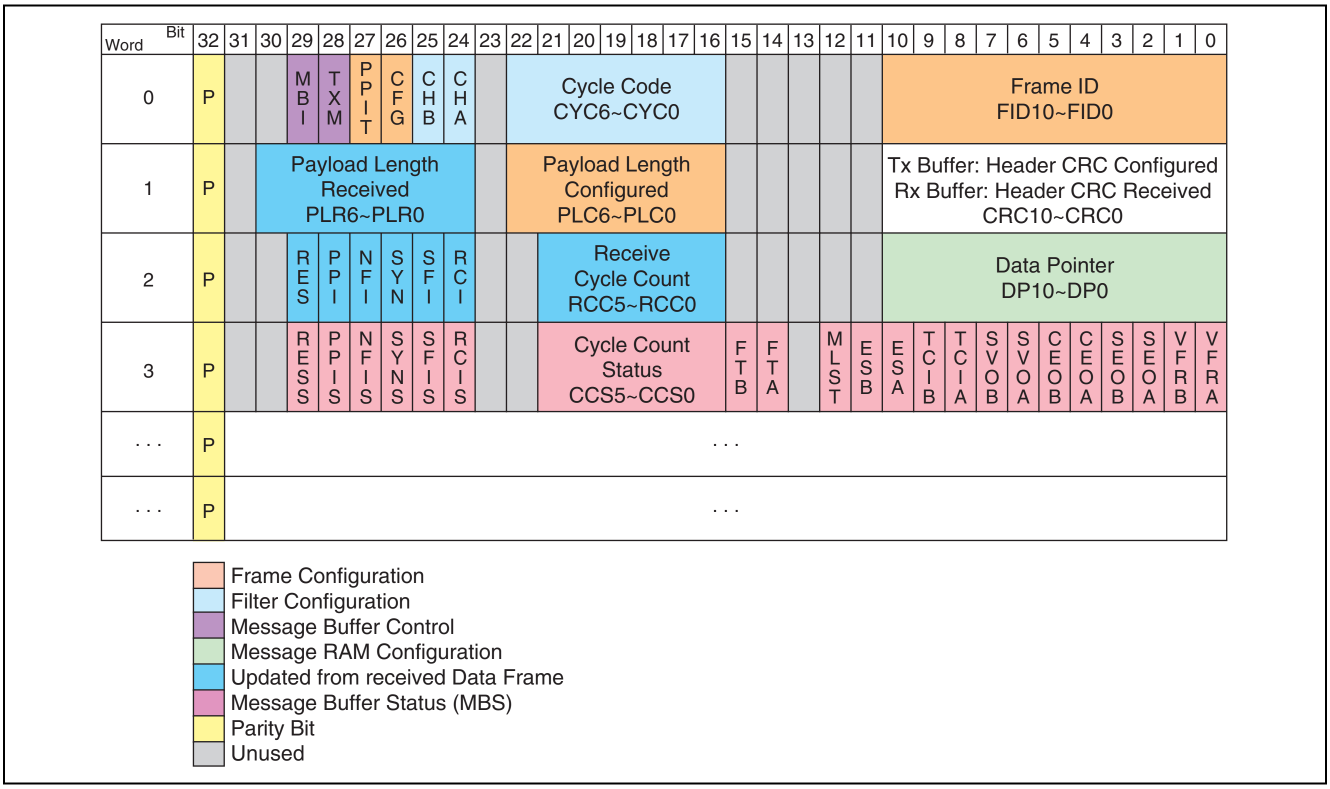


Figure 32.18 Header Section of a Message Buffer in the Message RAM

Header 1 (word 0)

Write access via the FRWRHS1 register, read access via the FRRDHS1 register:

- Frame ID - Slot counter filtering configuration
- Cycle Code - Cycle counter filtering configuration
- CHA, CHB - Channel filtering configuration
- CFG - Message buffer direction configuration: receive / transmit
- PPIT - Payload Preamble Indicator Transmit
- TXM - Transmit mode configuration: single-shot / continuous
- MBI - Message buffer receive / transmit interrupt enable

Header 2 (word 1)

Write access via the FRWRHS2 register, read access via the FRRDHS2 register:

- Header CRC - Transmit Buffer: Configured by the program (calculated from frame header)
  - Receive Buffer: Updated from received frame
- Payload Length Configured - Length of data section (2-byte words) as configured by the program
- Payload Length Received - Length of payload segment (2-byte words) stored from received frame

Header 3 (word 2)

Write access via the FRWRHS3 register, read access via the FRRDHS3 register:

- Data Pointer - Pointer to the beginning of the corresponding data section in the data partition

Read access via the FRRDHS3 register, valid for receive buffers only, updated from received frames:

- Receive Cycle Count - Cycle count from received frame
- RCI - Received on Channel Indicator
- SFI - Startup Frame Indicator
- SYN - Sync Frame Indicator
- NFI - Null Frame Indicator
- PPI - Payload Preamble Indicator
- RES - Reserved bit

Message Buffer Status (MBS) (word 3)

Read access via the FRMBS register, updated by the CC at the end of the configured slot.

- VFRA - Valid Frame Received on channel A
- VFRB - Valid Frame Received on channel B
- SEOA - Syntax Error Observed on channel A
- SEOB - Syntax Error Observed on channel B
- CEOA - Content Error Observed on channel A
- CEOB - Content Error Observed on channel B
- SVOA - Slot boundary Violation Observed on channel A
- SVOB - Slot boundary Violation Observed on channel B
- TCIA - Transmission Conflict Indication channel A
- TCIB - Transmission Conflict Indication channel B
- ESA - Empty Slot Channel A
- ESB - Empty Slot Channel B
- MLST - Message LoST
- FTA - Frame Transmitted on Channel A

- FTB - Frame Transmitted on Channel B
- Cycle Count Status- Actual cycle count when status was updated
- RCIS - Received on Channel Indicator Status
- SFIS - Startup Frame Indicator Status
- SYNS - Sync Frame Indicator Status
- NFIS - Null Frame Indicator Status
- PPIS - Payload Preamble Indicator Status
- RESS - Reserved bit Status

32.23.2 Data Partition

The data partition of the Message RAM stores the data sections of the message buffers configured for reception / transmission as defined in the header partition. The number of data bytes for each message buffer can vary from 0 to 254. To optimize the data transfer between the shift registers of the two FlexRay Protocol Controllers and the Message RAM as well as between the CPU interface and the Message RAM, the physical width of the Message RAM is set to 4 bytes plus one parity bit.

The data partition starts after the last word of the header partition. When configuring the message buffers in the Message RAM the programmer has to assure that the data pointers point to addresses within the data partition. Figure 32.19 below shows an example how the data sections of the configured message buffers can be stored in the data partition of the Message RAM.

The beginning and the end of a message buffer's data section is determined by the data pointer and the payload length configured in the message buffer's header section, respectively. This enables a flexible usage of the available RAM space for storage of message buffers with different data length.

If the size of the data section is an odd number of 2-byte words, the remaining 16 bits in the last 32-bit word are unused (see figure 32.19 below).

Word / Bit	32	31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
• • •	P	Unused	Unused	Unused	Unused
• • •	P	Unused	Unused	Unused	Unused
• • •	P	MBn Data3	MBn Data2	MBn Data1	MBn Data0
• • •	P	• • •	• • •	• • •	• • •
• • •	P	• • •	• • •	• • •	• • •
• • •	P	MBn Data (m)	MBn Data (m-1)	MBn Data (m-2)	MBn Data (m-3)
• • •	P	• • •	• • •	• • •	• • •
• • •	P	• • •	• • •	• • •	• • •
• • •	P	• • •	• • •	• • •	• • •
• • •	P	MB1 Data3	MB1 Data2	MB1 Data1	MB1 Data0
• • •	P	• • •	• • •	• • •	• • •
• • •	P	MB1 Data (k)	MB1 Data (k-1)	MB1 Data (k-2)	MB1 Data (k-3)
2046	P	MB0 Data3	MB0 Data2	MB0 Data1	MB0 Data0
2047	P	Unused	Unused	MB0 Data5	MB0 Data4

Figure 32.19 Example for Structure of the Data Partition in the Message RAM

32.23.3 Parity Check

There is a parity checking mechanism implemented in the FlexRay core to assure the integrity of the data stored in the seven RAM blocks. The RAM blocks have a parity generator / checker attached as shown in figure 32.20. When data is written to a RAM block, the local parity generator generates the parity bit. The FlexRay core uses an even parity (with an even number of ones in the 32-bit data word a zero parity bit is generated). The parity bit is stored together with the respective data word. The parity is checked each time a data word is read from any of the RAM blocks. The FlexRay core's internal data buses have a width of 32 bits.

If a parity error is detected, the respective error flag is set to "1". The parity error flags (bits PIBF, POBF, PMR, PTBF1, and PTBF2 in the FRMHDS register), and the faulty message buffer indicators (bits FMBD, MFMB, FMB6 to FMB0 in the FRMHDS register) are located in the Message Handler Status register. These single error flags control the error interrupt flag (the PERR bit in the FREIR register).

Figure 32.20 shows the data paths between the RAM blocks and the parity generators / checkers.

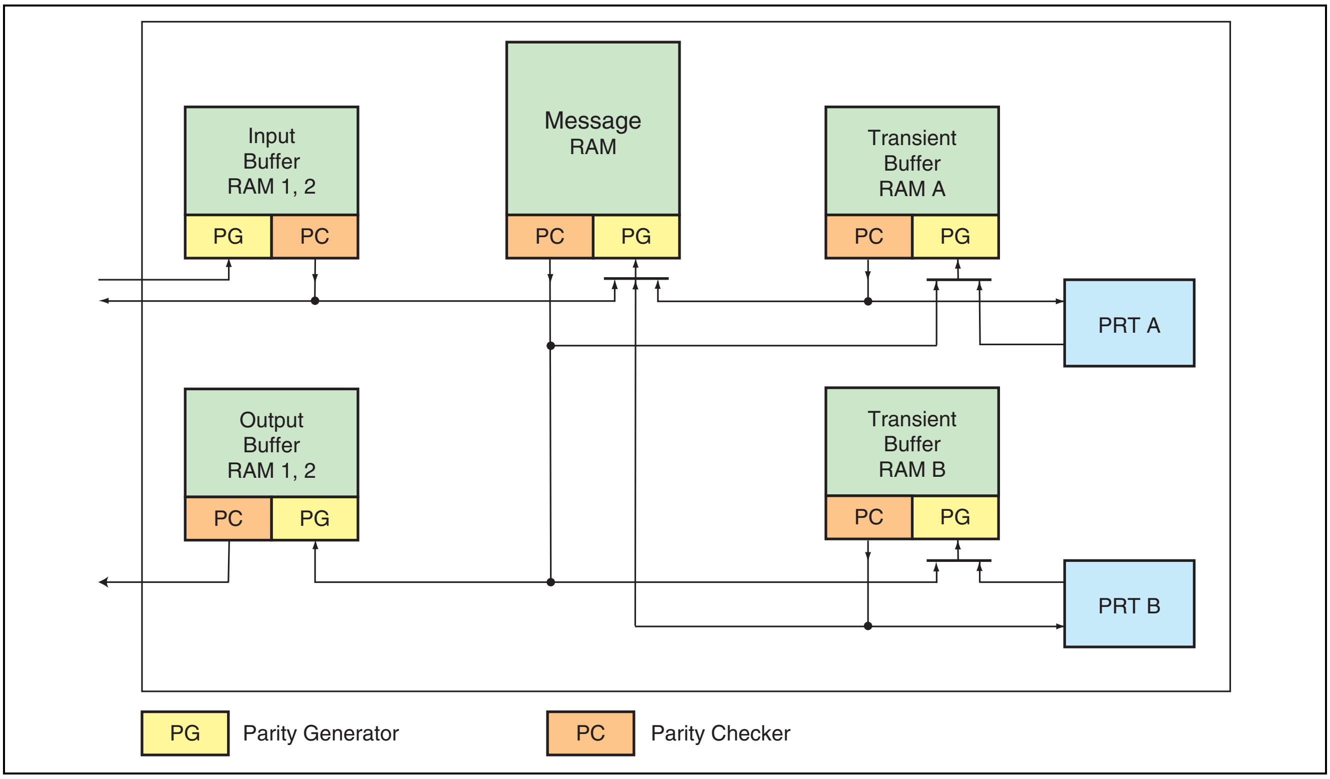


Figure 32.20 Parity Generation and Check

When a parity error has been detected the following actions will be performed:
In all cases

- The respective parity error flag in the FRMHDS register is set to "1"
- The parity error flag (the PERR bit in the FREIR register) is set to "1" and, if enabled, a module interrupt to the CPU will be generated.

Additionally in specific cases

(1) Parity error during data transfer from Input Buffer RAM 1, 2 -> Message RAM

(a) Transfer of header and data section:

- The PIBF bit in the FRMHDS register is set to "1"
- The FMBD in the FRMHDS register is set to "1" to indicate that bits FMB6 to FMB0 in the FRMHDS register point to a faulty message buffer
- Bits FMB6 to FMB0 in the FRMHDS register indicate the number of the faulty message buffer
- Transmit buffer: Transmission request for the respective message buffer is not set

(b) Transfer of data section only:

Parity error when reading header section of respective message buffer from Message RAM.

- The PRM bit in the FRMHDS register is set to "1"
- The FMBD bit in the FRMHDS register is set to "1" to indicate that bits FMB6 to FMB0 in the FRMHDS register point to a faulty message buffer
- Bits FMB6 to FMB0 in the FRMHDS register indicate the number of the faulty message buffer
- The data section of the respective message buffer is not updated
- Transmit buffer: Transmission request for the respective message buffer is not set to "1"

(2) Parity error during CPU reading Input Buffer RAM 1, 2

- The PIBF bit in the FRMHDS register is set to "1"

(3) Parity error during scan of header sections in Message RAM

- The PMR bit in the FRMHDS register is set to "1"
- The FMBD bit in the FRMHDS register is set to "1" to indicate that bits FMB6 to FMB0 in the FRMHDS register point to a faulty message buffer
- Bits FMB6 to FMB0 in the FRMHDS register indicate the number of the faulty message buffer
- Ignore message buffer (message buffer is skipped)

(4) Parity error during data transfer from Message RAM -> Transient Buffer RAM 1, 2

- The PRM bit in the FRMHDS register is set to "1"
- The FMBD bit in the FRMHDS register is set to "1" to indicate that bits FMB6 to FMB0 in the FRMHDS register point to a faulty message buffer
- Bits FMB6 to FMB0 in the FRMHDS register indicate the number of the faulty message buffer
- Frame not transmitted, frames already in transmission are invalidated by setting the frame CRC to "0"

(5) Parity error during data transfer from Transient Buffer RAM 1, 2 -> Protocol Controller 1, 2

- Bits PTBF1 and PTBF2 in the FRMHDS register is set to "1"
- Frames already in transmission are invalidated by setting the frame CRC to "0"

(6) Parity error during data transfer from Transient Buffer RAM 1, 2 -> Message RAM

(a) Parity error when reading header section of respective message buffer from Message RAM:

- The PMR bit in the FRMHDS register is set to "1"

- The FMBD bit in the FRMHDS register is set to "1" to indicate that bits FMB6 to FMB0 in the FRMHDS register point to a faulty message buffer
- Bits FMB6 to FMB0 in the FRMHDS register indicate the number of the faulty message buffer
- The data section of the respective message buffer is not updated

(b) Parity error when reading Transient Buffer RAM 1, 2:

- Bits PTBF1 and PTBF2 in the FRMHDS register is set to "1"
- The FMBD bit in the FRMHDS register is set to "1" to indicate bits FMB6 to FMB0 in the FRMHDS register point to a faulty message buffer
- Bits FMB6 to FMB0 in the FRMHDS register indicate the number of the faulty message buffer

(7) Parity error during data transfer from Message RAM -> Output Buffer RAM

- The PMR bit in the FRMHDS register is set to "1"
- The FMBD bit in the FRMHDS register is set to "1" to indicate that bits FMB6 to FMB0 in the FRMHDS register point to a faulty message buffer
- Bits FMB6 to FMB0 in the FRMHDS register indicate the number of the faulty message buffer

(8) Parity error during CPU reading Output Buffer RAM 1, 2

- The POBF bit in the FRMHDS register is set to "1"

(9) Parity error during data read of Transient Buffer RAM 1, 2

When a parity error occurs when the Message Handler reads a frame with network management information (PPI = 1) from the Transient Buffer RAM 1, 2 the corresponding network management vector registers (FRNMV1 to FRNMV3) are not updated from that frame.

32.24 Module Interrupt

32.24.1 Module Interrupt FlexRay interrupt 0, 1

In general, interrupts provide a close link to the protocol timing as they are triggered almost immediately when an error or status change is detected by the CC, a frame is received or transmitted, a configured timer interrupt is activated, or a stop watch event occurred. This enables the CPU to react very quickly on specific error conditions, status changes, or timer events. On the other hand too many interrupts can cause the CPU to miss deadlines required for the application. Therefore the CC supports enable / disable controls for each individual interrupt source separately.

An interrupt may be triggered when

- An error was detected
- A status flag is set to "1"
- A timer reaches a preconfigured value
- A message transfer from Input Buffer to Message RAM or from Message RAM to Output Buffer has completed
- A stop watch event occurred

Tracking status and generating interrupts when a status change or an error occurs are two independent tasks. Regardless of whether an interrupt is enabled or not, the corresponding status is tracked and indicated by the CC. The CPU has access to the actual status and error information by reading registers FREIR and FRSIR.

Table 32.18 Module Interrupt Flags and Interrupt Line Enable

Register	Bit	Function
FREIR	PEMC	Protocol Error Mode Changed
	CNA	Command Not Void
	SFBM	Sync Frames Below Minimum
	SFO	Sync Frame Overflow
	CCF	Clock Correction Failure
	CCL	CHI Command Locked
	PERR	Parity Error
	RFO	Receive FIFO Overrun
	EFA	Empty FIFO Access
	IIBA	Illegal Input Buffer Access
	IOBA	Illegal Output Buffer Access
	MHF	Message Handler Constraints Flag
	EDA	Error Detected on Channel A
	LTVA	Latest Transmit Violation Channel A
	TABA	Transmission Across Boundary Channel A
	EDB	Error Detected on Channel B
	LTVB	Latest Transmit Violation Channel B
	TABB	Transmission Across Boundary Channel B

Register	Bit	Function
FRSIR	WST	Wakeup Status
	CAS	Collision Avoidance Symbol
	CYCS	Cycle Start Interrupt
	TXI	Transmit Interrupt
	RXI	Receive Interrupt
	RFNE	Receive FIFO not Empty
	RFCL	Receive FIFO Critical Level
	NMVC	Network Management Vector Changed
	TI0	Timer Interrupt 0
	TI1	Timer Interrupt 1
	TIBC	Transfer Input Buffer Completed
	TOBC	Transfer Output Buffer Completed
	SWE	Stop Watch Event
	SUCS	Startup Completed Successfully
	MBSI	Message Buffer Status Interrupt
	SDS	Start of Dynamic Segment
	WUPA	Wakeup Pattern Channel A
	MTSA	MTS Received on Channel A
	WUPB	Wakeup Pattern Channel B
	MTSB	MTS Received on Channel B
FRILE	EINT0	Enable Interrupt Line 0
	EINT1	Enable Interrupt Line 1

The interrupt lines to the CPU, FlexRay_int0 and FlexRay_int1, are controlled by the enabled interrupts. In addition each of the two interrupt lines can be enabled / disabled separately by programming bits EINT0 and EINT1 in the FRILE register.

When a transfer between IBF / OBF and the Message RAM has completed the TIBC or TOBC bit in the FRSIR register is set to "1".

32.24.2 FlexRay timer interrupt 0, 1

The two timer interrupts generated by interrupt timer 0 and 1 are available on lines FlexRay_tint0 and FlexRray_tint1. They can be configured via registers FRT0C and FRT1C.

32.25 Assignment of FlexRay Configuration Parameters

Table 32.19 FlexRay Configuration Parameters

Parameter	Bit (field)	Page
pKeySlotUsedForStartup	TXST bit in FRSUCC1 register	32-53
pKeySlotUsedForSync	TXSY bit in FRSUCC1 register	32-53
gColdStartAttempts	Bits CSA4 to CSA0 in FRSUCC1 register	32-53
pAllowPassiveToActive	Bits PTA4 to PTA0 in FRSUCC1 register	32-53
pWakeupChannel	WUCS bit in FRSUCC1 register	32-53
pSingleSlotEnabled	TSM bit in FRSUCC1 register	32-53
pAllowHaltDueToClock	HCSE bit in FRSUCC1 register	32-52
pChannels	Bits CCHA and CCHB in FRSUCC1 register	32-52
pdListenTimeout	Bits LT20 to LT0 in FRSUCC2 register	32-58
gListenNoise	Bits LTN3 to LTN0 in FRSUCC2 register	32-58
gMaxWithoutClockCorrectionPassive	Bits WCP3 to WCP0 in FRSUCC3 register	32-59
gMaxWithoutClockCorrectionFatal	Bits WCF3 to WCF0 in FRSUCC3 register	32-59
gNetworkManagementVectorLength	Bits NML3 to NML0 in FRNEMC register	32-60
gdTSSTransmitter	Bits TSST3 to TSST0 in FRPRTC1 register	32-62
gdCASRxLowMax	Bits CASM6 to CASM0 in FRPRTC1 register	32-61
gdSampleClockPeriod	BRP0 bit in FRPRTC1 register	32-61
pSamplePerMicrotick	BRP0 bit in FRPRTC1 register	32-61
gdWakeupSymbolRxWindow	Bits RXW8 to RSW0 in FRPRTC1 register	32-61
pWakeupPattern	Bits RWP5 to RWP0 in FRPRTC1 register	32-61
gdWakeupSymbolRxIdle	Bits RXI5 to RXI0 in FRPRTC2 register	32-63
gdWakeupSymbolRxLow	Bits RXL5 to RXL0 in FRPRTC2 register	32-63
gdWakeupSymbolTxIdle	Bits TXI7 to TXI0 in FRPRTC2 register	32-63
gdWakeupSymbolTxLow	Bits TXL5 to TXL0 in FRPRTC2 register	32-63
gPayloadLengthStatic	Bits SFDL6 to SFDL0 in FRMHDC register	32-64
pLatestTx	Bits SLT12 to SLT0 in FRMHDC register	32-64
pMicroPerCycle	Bits UT19 to UT0 in FRGTUC1 register	32-65
gMacroPerCycle	Bits MPC13 to MPC0 in FRGTUC2 register	32-66
gSyncNodeMax	Bits SNM3 to SNM0 in FRGTUC2 register	32-66
pMicroInitialOffset[A]	Bits UIOA7 to UIOA0 in FRGTUC3 register	32-67
pMicroInitialOffset[B]	Bits UIOB7 to UIOB0 in FRGTUC3 register	32-67
pMacroInitialOffset[A]	Bits MIOA6 to MIOA0 in FRGTUC3 register	32-67
pMacroInitialOffset[B]	Bits MIOB6 to MIOBA0 in FRGTUC3 register	32-67
gdNIT	Bits NIT13 to NIT0 in FRGTUC4 register	32-68
gOffsetCorrectionStart	Bits OCS13 to OCS0 in FRGTUC4 register	32-68
pDelayCompensation[A]	Bits DCA7 to DCA0 in FRGTUC5 register	32-69
pDelayCompensation[B]	Bits DCB7 to DCB0 in FRGTUC5 register	32-69
pClusterDriftDamping	Bits CDD4 to CDD0 in FRGTUC5 register	32-69
pDecodingCorrection	Bits DEC7 to DEC0 in FRGTUC5 register	32-69
pdAcceptedStartupRange	Bits ASR10 to ASR0 in FRGTUC6 register	32-70

Parameter	Bit (field)	Page
pdMaxDrift	Bits MOD10 to MOD0 in FRGTUC6 register	32-70
gdStaticSlot	Bits SSL9 to SSL0 in FRGTUC7 register	32-71
gNumberOfStaticSlots	Bits NSS9 to NSS0 in FRGTUC7 register	32-71
gdMinislot	Bits MSL5 to MSL0 in FRGTUC8 register	32-72
gNumberOfMinislots	Bits NMS12 to NMS0 in FRGTUC8 register	32-72
gdActionPointOffset	Bits APO5 to APO0 in FRGTUC9 register	32-73
gdMinislotActionPointOffset	Bits MAPO4 to MAPO0 in FRGTUC9 register	32-73
gdDynamicSlotIdlePhase	Bits DSI1 to DSI0 in FRGTUC9 register	32-73
pOffsetCorrectionOut	Bits MOC13 to MOC0 in FRGTUC10 register	32-74
pRateCorrectionOut	Bits MRC10 to MRC0 in FRGTUC10 register	32-74
pExternOffsetCorrection	Bits EOC2 to EOC0 in FRGTUC11 register	32-75
pExternRateCorrection	Bits ERC2 to ERC0 in FRGTUC11 register	32-75

This page is blank for reasons of layout.

Section 33 Module Stop Function

33.1 Overview

This MCU provides a module stop function which allows certain of the modules to be stopped.

33.1.1 Module Stop Function

For on-chip peripheral modules that can be stopped, this function can stop the operation of the module by stopping the clock supply. Clock supply to the corresponding module can be controlled individually with the bits in module stop register 0 (MSTPCR0). The module stop function can be used with the following modules.

- Parallel DAC control (PDAC)
- Parallel selector (PSEL)
- Direct RAM input interface (DRI0, DRI1, and DRI2)
- Direct RAM output interface (DRO)

33.2 Register Descriptions

Table 33.1 lists the registers used in conjunction with the module stop function.

Table 33.1 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Module stop register 0	MSTPCR0	H'001F	H'FFFF 2800	8, 16	33-2

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

33.2.1 Module Stop Register 0 (MSTPCR0)

The MSTPCR0 register controls enabling/disabling operation for the allocated modules.

Writes to MSTPCR0 must be performed in word units. When writing a value to the low-order byte, the value H'3C must be written to the high-order byte (MSTPCR0KEY) at the same time. If any value other than H'3C is written to the MSTPCR0KEY field, or if the low-order byte is written as a byte unit, the write to the low-order byte will be ignored.

The MSTPCR0KEY field can be read in either byte or word units. However, since the data written to the MSTPCR0KEY field is not saved, the value read from the MSTPCR0KEY field is always H'00.

Note: • This register uses a different write method from ordinary registers to prevent it from being overwritten inadvertently. See section 33.2.2, Register Access Notes, for details.

Module stop register 0 (MSTPCR0) <P4 address: location H'FFFF 2800>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTPCR0KEY								—	—	—	DRO	DRI2	DRI1	DRI0	PDAC
After Reset:	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

<After Reset: H'001F >

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	MSTPCR0KEY	All 0	0	W	MSTPCR0 Register Write Key Code Bits Controls whether or not the low-order byte can be written. The data written to these bits are not retained. These bits are always read as "0". H'3C: The low-order byte can be written. Values other than H'3C: The low-order byte cannot be written.
7 to 5	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0"
4	DRO	1	R	W	DRO Module Stop Bit Setting this bit to "1" stops clock supply to the DRO module. When this bit is cleared to "0: clock supply to the DRO module will resume. Note, however, that the DRO registers are not initialized by stopping clock supply. 0: DRO operates 1: Clock supply to DRO is stopped
3	DRI2	1	R	W	DRI2 Module Stop Bit Setting this bit to "1" stops clock supply to the DRI2 module. When this bit is cleared to "0: clock supply to the DRI2 module will resume. Note, however, that the DRI2 registers are not initialized by stopping clock supply. 0: DRI2 operates 1: Clock supply to DRI2 is stopped
2	DRI1	1	R	W	DRI1 Module Stop Bit Setting this bit to "1" stops clock supply to the DRI1 module. When this bit is cleared to "0: clock supply to the DRI1 module will resume. Note, however, that the DRI1 registers are not initialized by stopping clock supply. 0: DRI1 operates 1: Clock supply to DRI1 is stopped

Bit	Abbreviation	After Reset	R	W	Description
1	DRI0	1	R	W	DRI0 Module Stop Bit Setting this bit to "1" stops clock supply to the DRI0 module. When this bit is cleared to "0: clock supply to the DRI0 module will resume. Note, however, that the DRI0 registers are not initialized by stopping clock supply. 0: DRI0 operates 1: Clock supply to DRI0 is stopped
0	PDAC	1	R	W	PDAC and PSEL Module Stop Bit Setting this bit to "1" stops clock supply to the PDAC module. When this bit is cleared to "0: clock supply to the PDAC module will resume. Note, however, that the PDAC registers are not initialized by stopping clock supply. 0: PDAC module operates 1: Clock supply to PDAC module is stopped

33.2.2 Register Access Notes

Module stop register 0 (MSTPCR0) uses a different write method from ordinary registers to prevent it from being overwritten inadvertently. Use the methods described below to write and read this register.

Word transfer instructions must be used when writing to MSTPCR0. This register cannot be written using byte instructions. As shown in figure 33.1, when writing to MSTPCR0 set the high-order byte to H'3C and the low-order byte to the write data. The same methods as those used for reading ordinary registers can be used to read MSTPCR0.

MSTPCR0 is allocated to location H'FFFF 2800, and can be read using either byte transfer instructions or word transfer instructions.

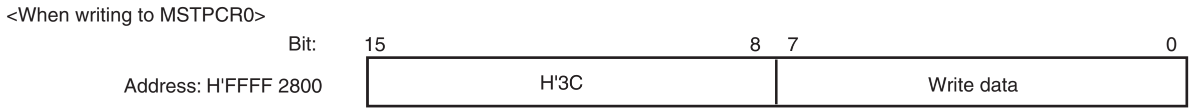


Figure 33.1 Writing the MSTPCR0 Register

33.3 Operation

33.3.1 Overview

Clock supply to the an on-chip peripheral module can be started by writing "0" to the corresponding module stop bit in the module stop register (MSTPCR0).

33.4 Usage Notes

Do not use the SLEEP instruction because sleep mode is not available in this MCU.

Read and write accesses to the register area of a module in the module stopped state are illegal. Execute read and write accesses to the register area of a module in the module stopped state after supplying the clock to the corresponding module.

This page is blank for reasons of layout.

Section 34 Power Supply Circuit

34.1 Structure of the Power Supply Circuit

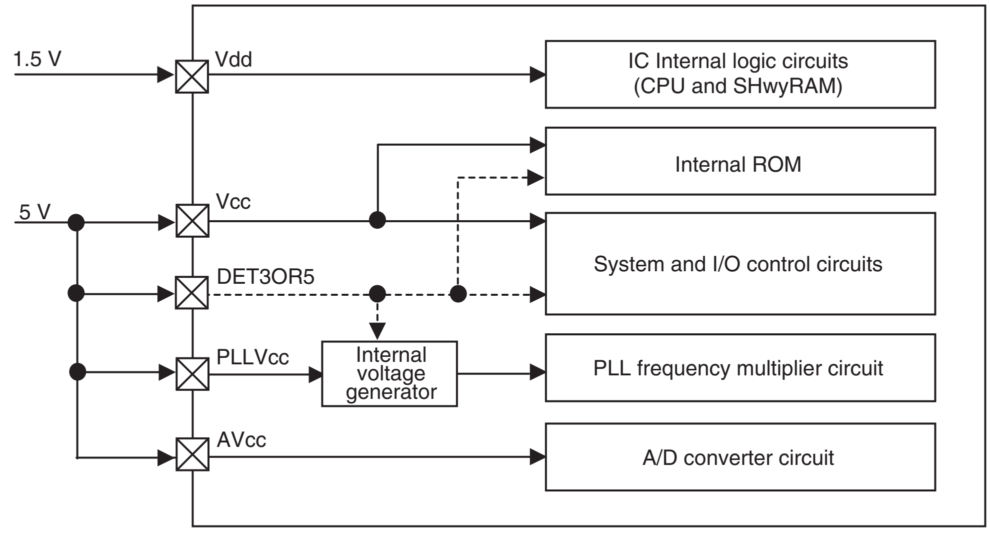
This MCU operates using the following supply voltages: 5 V ±0.5 V, 3.3 ±0.3 V, and 1.5 V +0.15/-0.1 V.

Unless noted otherwise, 5 V ±0.5 V is indicated by 5 V, 3.3 ±0.3 V is indicated by 3.3 V, and 1.5 V +0.15/-0.1 V is indicated by 1.5 V.

Table 34.1 Power Supply Functions

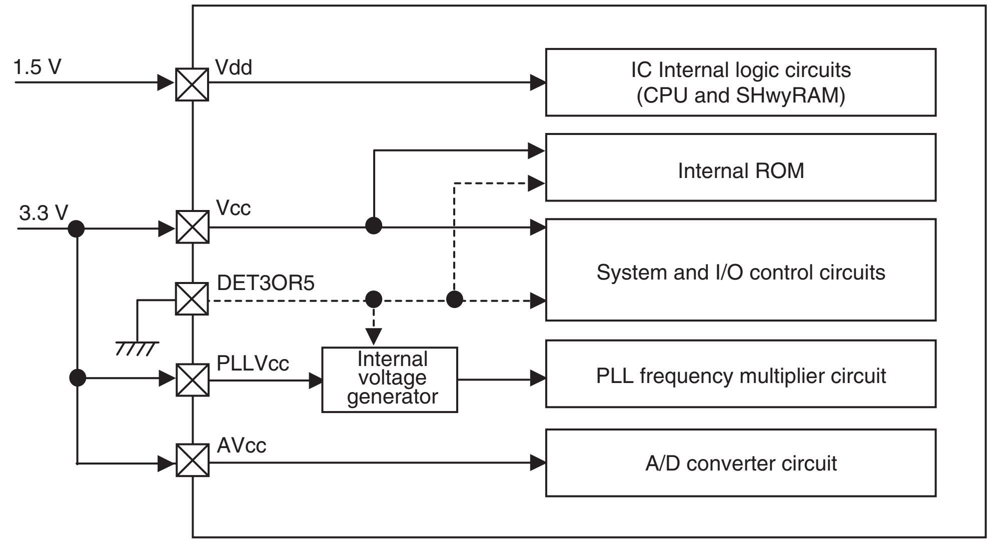
Pin Name	Function	Supply Type
Vdd	IC internal logic circuit power supply	1.5 V
Vcc	System and I/O port power supply	3.3 V or 5 V
DET3OR5	Vcc voltage level specification pin	When Vcc = 3.3 V: connect to Vss (pull down). When Vcc = 5 V: connect to Vcc (pull up).
PLLVcc	PLL frequency multiplier circuit power supply	3.3 V or 5 V*1
AVcc	A/D converter power supply	3.3 V or 5 V*1
Vss	Ground pins. All these pins must be connected to ground (GND).	—
PLLVss	Ground pin for the PLL frequency multiplier circuit	—
AVss	Ground pin for the A/D converter	—

Note: *1 When Vcc = 5 V, Vcc = PLLVcc = AVcc = 5 V
When Vcc = 3.3 V, Vcc = PLLVcc = 3.3 V


1.5 V
Vdd
IC Internal logic circuits (CPU and SHwyRAM)
Internal ROM
5 V
Vcc
System and I/O control circuits
DET3OR5
PLLVcc
Internal voltage generator
PLL frequency multiplier circuit
AVcc
A/D converter circuit

Note: • When Vcc = 5 V, connect the DET3OR5 pin to Vcc (i.e. pull up).

Figure 34.1 Power Supply Circuit Structure (when Vcc = 5 V)


1.5 V
Vdd
IC Internal logic circuits
(CPU and SHwyRAM)
Internal ROM
3.3 V
Vcc
DET3OR5
System and I/O control circuits
PLLVcc
Internal voltage generator
PLL frequency multiplier circuit
AVcc
A/D converter circuit

Note: • When Vcc = 3.3 V, connect the DET3OR5 pin to Vss (i.e. pull down).

Figure 34.2 Power Supply Circuit Structure (when Vcc = 3.3 V)

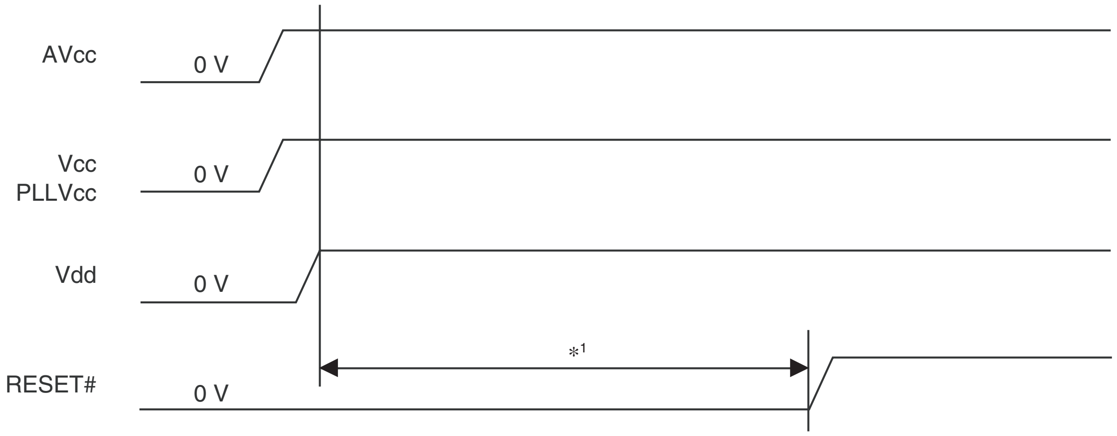
Table 34.2 Combination of Power Supply Voltages

Item	Vdd (Core Power Supply)	Vcc (Bus and System Power Supply)	PLLVcc (PLL Power Supply)	AVcc (ADC Power Supply)
Case 1	1.5 V	3.3 V	3.3 V	3.3 V
Case 2	1.5 V	3.3 V	3.3 V	5.0 V
Case 3	1.5 V	5.0 V	5.0 V	5.0 V

Notes:
- Use the voltages according to the above combination.
- For a correspondence between power supply names and pins, see section 1.5, Pin Functions.

34.2 Power On Sequence

Figure 34.3 shows the power on sequence.



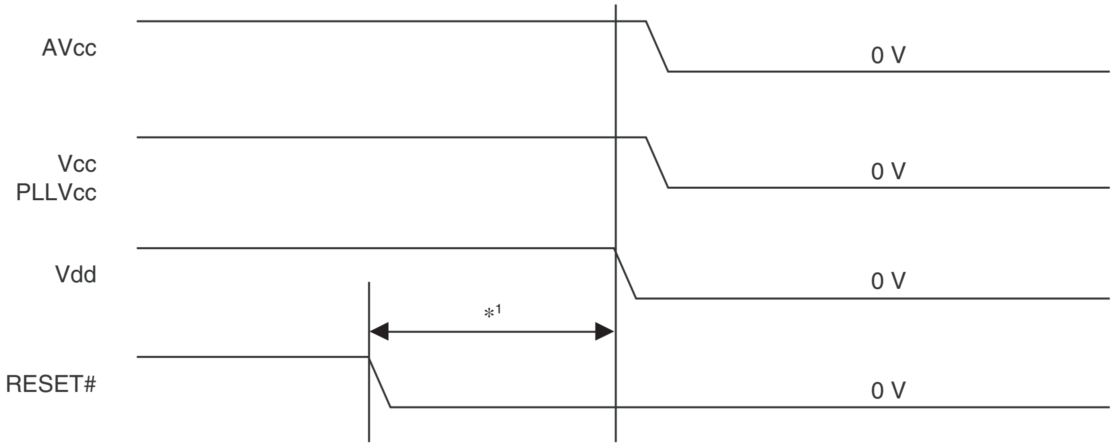
Notes: *1 After applying all power supply levels and holding the RESET# pin at the "L" level for the oscillator stabilization time, apply a "H" level to the RESET# pin (reset clear).

- Power supply operating conditions
  Vdd = 1.5 V + 0.15 V, -0.1 V
  Vcc = 5 V ±0.5 V or 3.3 V ±0.3 V
- Constraints at power on
  Design applications so that individual power supply voltages are not turned on or off at any time other than power on and power off.
  (Assure that either all the supply voltages are on or that all are off.)

Figure 34.3 Power On Sequence

34.3 Power Off Sequence

Figure 34.4 shows the power off sequence.



Notes: *1 First set the RESET# pin to the "L" level and the turn of the power supply voltages.

- Power supply operating conditions
  Vdd = 1.5 V + 0.15 V, -0.1 V
  Vcc = 5 V ±0.5 V or 3.3 V ±0.3 V
- Constraints at power off
  Design applications so that individual power supply voltages are not turned on or off at any time other than power on and power off.
  (Assure that either all the supply voltages are on or that all are off.)

Figure 34.4 Power Off Sequence

Section 35 User Break Controller (UBC)

The user break controller (UBC) provides versatile functions to facilitate program debugging. These functions help to ease creation of a self-monitor/debugger, which allows easy program debugging using this MCU alone, without using the in-circuit emulator. Various break conditions can be set in the UBC: instruction fetch or read/write access of an operand, operand size, data contents, address value, and program stop timing for instruction fetch.

35.1 Overview

1. The following break conditions can be set.

   Break channels: Two (channels 0 and 1)

   User break conditions can be set independently for channels 0 and 1, and can also be set as a single sequential condition for the two channels, that is, a sequential break. (Sequential break involves two cases such that the channel 0 break condition is satisfied in a certain bus cycle and then the channel 1 break condition is satisfied in a different bus cycle, and vice versa.)

- Address

  When 40 bits containing ASID and 32-bit address are compared with the specified value, all the ASID bits can be compared or masked.

  32-bit address can be masked bit by bit, allowing the user to mask the address in desired page sizes such as lower 12 bits (4-Kbyte page) and lower 10 bits (1-Kbyte page).

- Data

  32 bits can be masked only for channel 1.

- Bus cycle

  The program can break either for instruction fetch (PC break) or operand access.

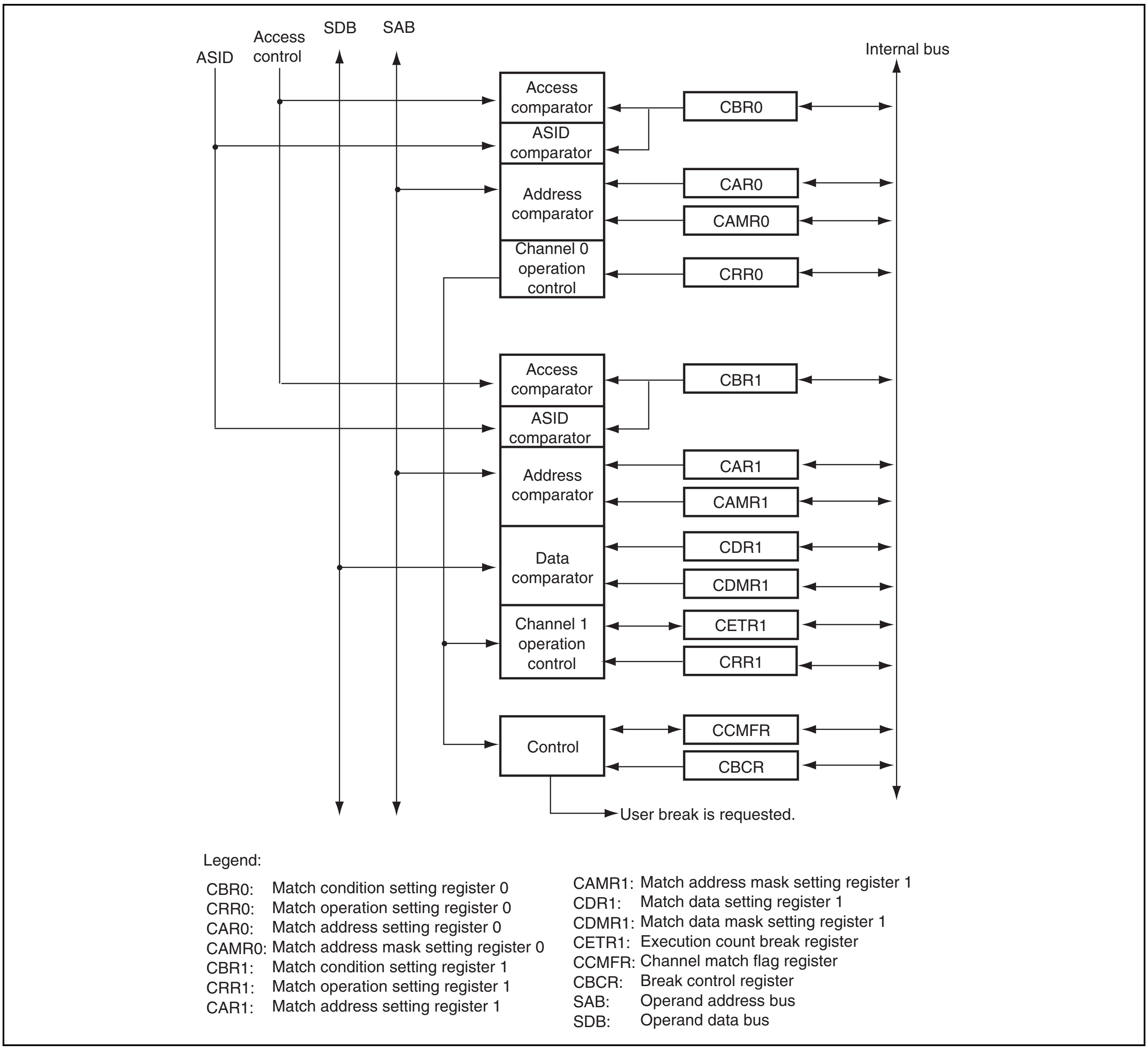
- Read or write access

- Operand sizes

  Byte, word, longword, and quadword are supported.

2. The user-designated exception handling routine for the user break condition can be executed.
3. Pre-instruction-execution or post-instruction-execution can be selected as the PC break timing.
4. A maximum of $2^{12} – 1$ repetition counts can be specified as the break condition (available only for channel 1).

Figure 35.1 shows the UBC block diagram.



Legend:

CBR0:	Match condition setting register 0
CRR0:	Match operation setting register 0
CAR0:	Match address setting register 0
CAMR0:	Match address mask setting register 0
CBR1:	Match condition setting register 1
CRR1:	Match operation setting register 1
CAR1:	Match address setting register 1
CAMR1:	Match address mask setting register 1
CDR1:	Match data setting register 1
CDMR1:	Match data mask setting register 1
CETR1:	Execution count break register
CCMFR:	Channel match flag register
CBCR:	Break control register
SAB:	Operand address bus
SDB:	Operand data bus

Figure 35.1 Block Diagram of UBC

35.2 Register Descriptions

Table 35.1 shows the register configuration of the UBC.

Table 35.1 Register Configuration

Name	Abbreviation	After Reset	P4 Address	Size	Page
Match condition setting register 0	CBR0	H'2000 0000	H'FF20 0000	32	35-4
Match operation setting register 0	CRR0	H'0000 2000	H'FF20 0004	32	35-9
Match address setting register 0	CAR0	Undefined	H'FF20 0008	32	35-11
Match address mask setting register 0	CAMR0	Undefined	H'FF20 000C	32	35-12
Match condition setting register 1	CBR1	H'2000 0000	H'FF20 0020	32	35-6
Match operation setting register 1	CRR1	H'0000 2000	H'FF20 0024	32	35-10
Match address setting register 1	CAR1	Undefined	H'FF20 0028	32	35-11
Match address mask setting register 1	CAMR1	Undefined	H'FF20 002C	32	35-12
Match data setting register 1	CDR1	Undefined	H'FF20 0030	32	35-13
Match data mask setting register 1	CDMR1	Undefined	H'FF20 0034	32	35-14
Execution count break register 1	CETR1	Undefined	H'FF20 0038	32	35-14
Channel match flag register	CCMFR	H'0000 0000	H'FF20 0600	32	35-15
Break control register	CBCR	H'0000 0000	H'FF20 0620	32	35-16

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

The access size must be the same as the control register size. If the size is different, the register is not written to if attempted, and reading the register returns the undefined value. A desired break may not occur between the time when the instruction for rewriting the control register is executed and the time when the written value is actually reflected on the register. In order to confirm the exact timing when the control register is updated, read the data which has been written most recently. The subsequent instructions are valid for the most recently written register value.

35.2.1 Match Condition Setting Registers 0 and 1 (CBR0 and CBR1)

The CBR0 and CBR1 registers specify the break conditions for channels 0 and 1, respectively. The following break conditions can be set in the CBR0 and CBR1: (1) whether or not to include the match flag in the conditions, (2) whether or not to include the ASID, and the ASID value when included, (3) whether or not to include the data value, (4) operand size, (5) whether or not to include the execution count, (6) bus type, (7) instruction fetch cycle or operand access cycle, and (8) read or write access cycle.

(1) Match Condition Setting Register 0 (CBR0)

Match Condition Setting Register 0 (CBR0) <P4 address: location H'FF20 0000>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MFE	AIE	MFI						AIV							
After Reset:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SZ			—	—	—	—	CD		ID		—	RW		CE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'2000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	MFE	0	R	W	Match Flag Enable Bit Specifies whether or not to include the match flag value specified by the MFI bit of this register in the match conditions. When the specified match flag value is "1", the condition is determined to be satisfied. 0: The match flag is not included in the match conditions; thus, not checked 1: The match flag is included in the match conditions
30	AIE	0	R	W	ASID Enable Bit Specifies whether or not to include the ASID specified by the AIV bit of this register in the match conditions. 0: The ASID is not included in the match conditions; thus, not checked 1: The ASID is included in the match conditions
29 to 24	MFI	100000	R	W	Match Flag Specify Bits Specifies the match flag to be included in the match conditions. 000000: MF0 bit of the CCMFR register 000001: MF1 bit of the CCMFR register Others: Reserved (setting prohibited) Note: • The value after a reset is the reserved state value, but when "1" is written into CBR0[0], MFI must be set to "000000" or "000001". And note that the channel 0 is not hit when MFE bit of this register is "1" and MFI bits are 000000 in the condition of CCMFR.MF0 = "0".
23 to 16	AIV	All 0	R	W	ASID Specify Bits Specifies the ASID value to be included in the match conditions.
15	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".

Bit	Abbreviation	After Reset	R	W	Description
14 to 12	SZ	All 0	R	W	Operand Size Select Bits Specifies the operand size to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 000: The operand size is not included in the match conditions; thus, not checked (any operand size specifies the match condition).*1 001: Byte access 010: Word access 011: Longword access 100: Quadword access*2 Others: Reserved (setting prohibited)
11 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7, 6	CD	00	R	W	Bus Select Bits Specifies the bus to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 00: Operand bus for operand access Others: Reserved (setting prohibited)
5, 4	ID	00	R	W	Instruction Fetch/Operand Access Select Bits Specifies the instruction fetch cycle or operand access cycle as the match condition. 00: Instruction fetch cycle or operand access cycle 01: Instruction fetch cycle 10: Operand access cycle 11: Instruction fetch cycle or operand access cycle
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2, 1	RW	00	R	W	Bus Command Select Bits Specifies the read/write cycle as the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 00: Read cycle or write cycle 01: Read cycle 10: Write cycle 11: Read cycle or write cycle
0	CE	0	R	W	Channel Enable Bit Validates/invalidates the channel. If this bit is "0", all the other bits of this register are invalid. 0: Invalidates the channel 1: Validates the channel

Notes: *1 If the data value is included in match conditions, be sure to specify the operand size.

*2 If the quad word access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and the match data mask setting register.

(2) Match Condition Setting Register 1 (CBR1)

Match Condition Setting Register 1 (CBR1) <P4 address: location H'FF20 0020>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MFE	AIE	MFI						AIV							
After Reset:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBE	SZ			ETBE	—	—	—	CD		ID		—	RW		CE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'2000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31	MFE	0	R	W	Match Flag Enable Bit Specifies whether or not to include the match flag value specified by the MFI bit of this register in the match conditions. When the specified match flag value is "1", the condition is determined to be satisfied. 0: The match flag is not included in the match conditions; thus, not checked 1: The match flag is included in the match conditions
30	AIE	0	R	W	ASID Enable Bit Specifies whether or not to include the ASID specified by the AIV bit of this register in the match conditions. 0: The ASID is not included in the match conditions; thus, not checked 1: The ASID is included in the match conditions
29 to 24	MFI	100000	R	W	Match Flag Specify Bits Specifies the match flag to be included in the match conditions. 000000: The MF0 bit of the CCMFR register 000001: The MF1 bit of the CCMFR register Others: Reserved (setting prohibited) Note: • The value after a reset is the reserved state value, but when "1" is written into CBR1[0], MFI must be set to 000000 or 000001. And note that the channel 1 is not hit when MFE bit of this register is "1" and MFI bits are 000001 in the condition of CCMFR.MF1 = "0".
23 to 16	AIV	All 0	R	W	ASID Specify Bits Specifies the ASID value to be included in the match conditions.
15	DBE	0	R	W	Data Value Enable Bit*[2] Specifies whether or not to include the data value in the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 0: The data value is not included in the match conditions; thus, not checked 1: The data value is included in the match conditions

Bit	Abbreviation	After Reset	R	W	Description
14 to 12	SZ	000	R	W	Operand Size Select Bits Specifies the operand size to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 000: The operand size is not included in the match condition; thus, not checked (any operand size specifies the match condition).*1 001: Byte access 010: Word access 011: Longword access 100: Quadword access*3 Others: Reserved (setting prohibited)
11	ETBE	0	R	W	Execution Count Value Enable Bit Specifies whether or not to include the execution count value in the match conditions. If this bit is "1" and the match condition satisfaction count matches the value specified by the CETR1 register, the operation specified by the CRR1 register is performed. 0: The execution count value is not included in the match conditions; thus, not checked 1: The execution count value is included in the match conditions
10 to 8	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
7, 6	CD	00	R	W	Bus Select Bits Specifies the bus to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 00: Operand bus for operand access Others: Reserved (setting prohibited)
5, 4	ID	00	R	W	Instruction Fetch/Operand Access Select Bits Specifies the instruction fetch cycle or operand access cycle as the match condition. 00: Instruction fetch cycle or operand access cycle 01: Instruction fetch cycle 10: Operand access cycle 11: Instruction fetch cycle or operand access cycle
3	—	0	0	0	Reserved Bit This bit is always read as "0". The write value should always be "0".
2, 1	RW	00	R	W	Bus Command Select Bits Specifies the read/write cycle as the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 00: Read cycle or write cycle 01: Read cycle 10: Write cycle 11: Read cycle or write cycle
0	CE	0	R	W	Channel Enable Bit Validates/invalidates the channel. If this bit is "0", all the other bits in this register are invalid. 0: Invalidates the channel 1: Validates the channel

Notes: *1 If the data value is included in the match conditions, be sure to specify the operand size.

*2 The OCBI instruction is handled as longword write access without the data value, and the PREF, OCBP, and OCBWB instructions are handled as longword read access without the data value. Therefore, do not include the data value in the match conditions for these instructions.

*3 If the quadword access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and the match data mask setting register.

35.2.2 Match Operation Setting Registers 0 and 1 (CRR0 and CRR1)

The CRR0 and CRR1 registers specify the operation to be executed when channels 0 and 1 satisfy the match condition, respectively. The following operations can be set in the CRR0 and CRR1 registers: (1) breaking at a desired timing for the instruction fetch cycle and (2) requesting a break.

(1) Match Operation Setting Register 0 (CRR0)

Match Operation Setting Register 0 (CRR0) <P4 address: location H'FF20 0004>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCB	BIE
After Reset:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 2000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
12 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	PCB	0	R	W	PC Break Select Bit Specifies either before or after instruction execution as the break timing for the instruction fetch cycle. This bit is invalid for breaks other than the ones for the instruction fetch cycle. 0: Sets the PC break before instruction execution 1: Sets the PC break after instruction execution
0	BIE	0	R	W	Break Enable Bit Specifies whether or not to request a break when the match condition is satisfied for the channel. 0: Does not request a break 1: Requests a break

(2) Match Operation Setting Register 1 (CRR1)

Match Operation Setting Register 1 (CRR1) <P4 address: location H'FF20 0024>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCB	BIE
After Reset:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 2000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 14	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
13	—	1	1	1	Reserved Bit This bit is always read as "1". The write value should always be "1".
12 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	PCB	0	R	W	PC Break Select Bit Specifies either before or after instruction execution as the break timing for the instruction fetch cycle. This bit is invalid for breaks other than ones for the instruction fetch cycle. 0: Sets the PC break before instruction execution 1: Sets the PC break after instruction execution
0	BIE	0	R	W	Break Enable Bit Specifies whether or not to request a break when the match condition is satisfied for the channel. 0: Does not request a break 1: Requests a break

35.2.3 Match Address Setting Registers 0 and 1 (CAR0 and CAR1)

The CAR0 and CAR1 registers specify the virtual address to be included in the break conditions for channels 0 and 1, respectively.

(1) Match Address Setting Register 0 (CAR0)

Match Address Setting Register 0 (CAR0) <P4 address: location H'FF20 0008>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CA															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CA															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	CA	Undefined	R	W	Compare Address Bits Specifies the address to be included in the break conditions. When the operand bus has been specified using the CBR0 register, specify the SAB address in CA[31:0].

(2) Match Address Setting Register 1 (CAR1)

Match Address Setting Register 1 (CAR1) <P4 address: location H'FF20 0028>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CA															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CA															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	CA	Undefined	R	W	Compare Address Bits Specifies the address to be included in the break conditions. When the operand bus has been specified using the CBR1 register, specify the SAB address in CA[31:0].

35.2.4 Match Address Mask Setting Registers 0 and 1 (CAMR0 and CAMR1)

The CMAR0 and CMAR1 registers specify the bits to be masked among the address bits specified by using the match address setting register of the corresponding channel. (Set the bits to be masked to "1".)

(1) Match Address Mask Setting Register 0 (CAMR0)

Match Address Mask Setting Register 0 (CAMR0) <P4 address: location H'FF20 000C>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAM															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAM															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	CAM	Undefined	R	W	Compare Address Mask Bits Specifies the bits to be masked among the address bits which are specified using the CAR0 register. (Set the bits to be masked to "1".) 0: Address bits CA[n] are included in the break condition 1: Address bits CA[n] are masked and not included in the break condition Legend: n = 31 to 0

(2) Match Address Mask Setting Register 1 (CAMR1)

Match Address Mask Setting Register 1 (CAMR1) <P4 address: location H'FF20 002C>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAM															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAM															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	CAM	Undefined	R	W	Compare Address Mask Bits Specifies the bits to be masked among the address bits which are specified using the CAR1 register. (Set the bits to be masked to "1".) 0: Address bits CA[n] are included in the break condition 1: Address bits CA[n] are masked and not included in the break condition Legend: n = 31 to 0

35.2.5 Match Data Setting Register 1 (CDR1)

The CDR1 register specifies the data value to be included in the break conditions for channel 1.

Match Data Setting Register 1 (CDR1) <P4 address: location H'FF20 0030>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CD															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CD															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	CD	Undefined	R	W	Compare Data Value Bits Specifies the data value to be included in the break conditions. When the operand bus has been specified using the CBR1 register, specify the SDB data value in CD[31:0].

Table 35.2 Settings for Match Data Setting Register

Bus and Size Selected Using CBR1	CD[31:24]	CD[23:16]	CD[15:8]	CD[7:0]
Operand bus (byte)	Don't care			SDB7 to SDB0
Operand bus (word)	Don't care		SDB15 to SDB0	
Operand bus (longword)	SDB31 to SDB0			

Notes:
- If the data value is included in the match conditions, be sure to specify the operand size.
- The OCBI instruction is handled as longword write access without the data value, and the PREF, OCBP, and OCBWB instructions are handled as longword read access without the data value. Therefore, do not include the data value in the match conditions for these instructions.
- If the quadword access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and match data mask setting register.

35.2.6 Match Data Mask Setting Register 1 (CDMR1)

The CDMR1 register specifies the bits to be masked among the data value bits specified using the match data setting register. (Set the bits to be masked to "1".)

Match Data Mask Setting Register 1 (CDMR1) <P4 address: location H'FF20 0034>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDM															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDM															
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 0	CDM	Undefined	R	W	Compare Data Value Mask Bits Specifies the bits to be masked among the data value bits specified using the CDR1 register. (Set the bits to be masked to "1".) 0: Data value bits CD[n] are included in the break condition 1: Data value bits CD[n] are masked and not included in the break condition Legend: n = 31 to 0

35.2.7 Execution Count Break Register 1 (CETR1)

The CETR1 register specifies the number of the channel hits before a break occurs. A maximum value of $2^{12} – 1$ can be specified. When the execution count value is included in the match conditions by using the match condition setting register, the value of this register is decremented by one every time the channel is hit. When the channel is hit after the register value reaches H'001, a break occurs.

Execution Count Break Register 1 (CETR1) <P4 address: location H'FF20 0038>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CET											
After Reset:	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 12	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
11 to 0	CET	Undefined	R	W	Execution Count Bits Specifies the execution count to be included in the break conditions.

35.2.8 Channel Match Flag Register (CCMFR)

The CCMFR register indicates whether or not the match conditions have been satisfied for each channel. When a channel match condition has been satisfied, the corresponding flag bit is set to "1". To clear the flags, write the data containing value "0" for the bits to be cleared and value "1" for the other bits to this register. (The logical AND between the value which has been written and the current register value is actually written to the register.) Sequential operation using multiple channels is available by using these match flags.

Channel Match Flag Register (CCMFR) <P4 address: location H'FF20 0600>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MF1	MF0
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 2	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
1	MF1	0	R	W	Channel 1 Condition Match Flag This flag is set to "1" when the channel 1 match condition has been satisfied. To clear the flag, write "0" to this bit. 0: Channel 1 match condition has not been satisfied 1: Channel 1 match condition has been satisfied
0	MF0	0	R	W	Channel 0 Condition Match Flag This flag is set to "1" when the channel 0 match condition has been satisfied. To clear the flag, write "0" to this bit. 0: Channel 0 match condition has not been satisfied 1: Channel 0 match condition has been satisfied

35.2.9 Break Control Register (CBCR)

The CBCR register specifies whether or not to use the user break debugging support function. For details on the user break debugging support function, refer to section 35.4, User Break Debugging Support Function.

Break Control Register (CBCR) <P4 address: location H'FF20 0620>

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UBDE
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000 0000>

Bit	Abbreviation	After Reset	R	W	Description
31 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	UBDE	0	R	W	User Break Debugging Support Function Enable Bit Specifies whether or not to use the user break debugging support function. 0: Does not use the user break debugging support function 1: Uses the user break debugging support function

35.3 Operation Description

35.3.1 Definition of Words Related to Accesses

"Instruction fetch" refers to an access in which an instruction is fetched. For example, fetching the instruction located at the branch destination after executing a branch instruction is an instruction access. "Operand access" refers to any memory access accompanying execution of an instruction. For example, accessing an address (PC + disp × 2 + 4) in the instruction MOV.W@(disp,PC),Rn is an operand access. "Data" is used in contrast to "address".

All types of operand access are classified into read or write access. Special care must be taken in using the following instructions.

- PREF, OCBP, and OCBWB: Instructions for a read access
- MOVCA.L and OCBI: Instructions for a write access
- TAS.B: Instruction for a single read access or a single write access

The operand access accompanying the PREF, OCBP, OCBWB, and OCBI instructions is access without the data value; therefore, do not include the data value in the match conditions for these instructions.

The operand size should be defined for all types of operand access. Available operand sizes are byte, word, longword, and quadword. For operand access accompanying the PREF, OCBP, OCBWB, MOVCA.L, and OCBI instructions, the operand size is defined as longword.

35.3.2 User Break Operation Sequence

The following describes the sequence from when the break condition is set until the user break exception handling is initiated.

1. Specify the operand size, bus, instruction fetch/operand access, and read/write as the match conditions using the match condition setting register (CBR0 or CBR1). Specify the break address using the match address setting register (CAR0 or CAR1), and specify the address mask condition using the match address mask setting register (CAMR0 or CAMR1). To include the ASID in the match conditions, set the AIE bit in the match condition setting register and specify the ASID value by the AIV bit in the same register. To include the data value in the match conditions, set the DBE bit in the match condition setting register; specify the break data using the match data setting register (CDR1); and specify the data mask condition using the match data mask setting register (CDMR1). To include the execution count in the match conditions, set the ETBE bit of the match condition setting register; and specify the execution count using the execution count break register (CETR1). To use the sequential break, set the MFE bit of the match condition setting register; and specify the number of the first channel using the MFI bit.
2. Specify whether or not to request a break when the match condition is satisfied and the break timing when the match condition is satisfied as a result of fetching the instruction using the match operation setting register (CRR0 or CRR1). After having set all the bits in the match condition setting register except the CE bit and the other necessary registers, set the CE bit and read the match condition setting register again. This ensures that the set values in the control registers are valid for the subsequent instructions immediately after reading the register. Setting the CE bit of the match condition setting register in the initial state after reset via the control registers may cause an undesired break.
3. When the match condition has been satisfied, the corresponding condition match flag (MF1 or MF0) in the channel match flag register (CCMFR) is set. A break is also requested to the CPU according to the set values in the match operation setting register (CRR0 or CRR1). The CPU operates differently according to the BL bit value of the SR register: when the BL bit is "0", the CPU accepts the break request and executes the specified exception handling; and when the BL bit is "1", the CPU does not execute the exception handling.
4. The match flags (MF1 and MF0) can be used to confirm whether or not the corresponding match condition has been satisfied. Although the flag is set when the condition is satisfied, it is not cleared automatically; therefore, write "0"

to the flag bit by issuing a memory store instruction to the channel match flag register (CCMFR) in order to use the flag again.

5. Breaks may occur virtually at the same time for channels 0 and 1. In this case, only one break request is sent to the CPU; however, the two condition match flags corresponding to these breaks may be set.
6. While the BL bit in the SR register is "1", no break requests are accepted. However, whether or not the condition has been satisfied is determined. When the condition is determined to be satisfied, the corresponding condition match flag is set.
7. If the sequential break conditions are set, the condition match flag is set every time the match conditions are satisfied for each channel. When the conditions have been satisfied for the first channel in the sequence but not for the second channel in the sequence, clear the condition match flag for the first channel in the sequence in order to release the first channel in the sequence from the match state.

35.3.3 Instruction Fetch Cycle Break

1. If the instruction fetch cycle is set in the match condition setting register (CBR0 or CBR1), the instruction fetch cycle is handled as a match condition. To request a break upon satisfying the match condition, set the BIE bit in the match operation setting register (CRR0 or CRR1) of the corresponding channel. Either before or after executing the instruction can be selected as the break timing according to the PCB bit value. If the instruction fetch cycle is specified as a match condition, be sure to clear the LSB to "0" in the match address setting register (CAR0 or CAR1); otherwise, no break occurs.
2. If pre-instruction-execution break is specified for the instruction fetch cycle, the break is requested when the instruction is fetched and determined to be executed. Therefore, this function cannot be used for the instructions which are fetched through overrun (i.e., the instructions fetched during branching or making transition to the interrupt routine but not executed). For priorities of pre-instruction-execution break and the other exceptions, refer to section 5, Exception Handling. If pre-instruction-execution break is specified for the delayed slot of the delayed branch instruction, the break is requested before the delayed branch instruction is executed. However, do not specify pre-instruction-execution break for the delayed slot of the RTE instruction.
3. If post-instruction-execution break is specified for the instruction fetch cycle, the break is requested after the instruction which satisfied the match condition has been executed and before the next instruction is executed. Similar to pre-instruction-execution break, this function cannot be used for the instructions which are fetched through overrun. For priorities of post-instruction-execution break and the other exceptions, refer to section 5, Exception Handling. If post-instruction-execution break is specified for the delayed branch instruction and its delayed slot, the break does not occur until the first instruction at the branch destination.
4. If the instruction fetch cycle is specified as the channel 1 match condition, the DBE bit of match condition setting register (CBR1) becomes invalid, the settings of match data setting register (CDR1) and match data mask setting register (CDMR1) are ignored. Therefore, the data value cannot be specified for the instruction fetch cycle break.

35.3.4 Operand Access Cycle Break

1. Table 35.3 shows the relation between the operand sizes specified using the match condition setting register (CBR0 or CBR1) and the address bits to be compared for the operand access cycle break.

Table 35.3 Relation between Operand Sizes and Address Bits to be Compared

Selected Operand Size	Address Bits to be Compared
Quadword	Address bits A31 to A3
Longword	Address bits A31 to A2
Word	Address bits A31 to A1
Byte	Address bits A31 to A0
Operand size is not included in the match conditions	Address bits A31 to A3 for quadword access Address bits A31 to A2 for longword access Address bits A31 to A1 for word access Address bits A31 to A0 for byte access

The above table means that if address H'0000 1003 is set in the match address setting register (CAR0 or CAR1), for example, the match condition is satisfied for the following access cycles (assuming that all the other conditions are satisfied):

— Longword access to address H'0000 1000

— Word access to address H'0000 1002

— Byte access to address H'0000 1003

2. When the data value is included in the channel 1 match conditions:

   If the data value is included in the match conditions, be sure to select quadword, longword, word, or byte as the operand size using the operand size select bit (SZ) of the match condition setting register (CBR1), and also set the match data setting register (CDR1) and the match data mask setting register (CDMR1). With these settings, the match condition is satisfied when both of the address and data conditions are satisfied. The data value and mask control for byte access, word access, and longword access should be set in bits 7 to 0, 15 to 0, and 31 to 0 in the bits CDR1 and CDMR1, respectively. For quadword access, 64-bit data is divided into the upper and lower 32-bit data units, and each unit is independently compared with the specified condition. When either the upper or lower 32-bit data unit satisfies the match condition, the match condition for the 64-bit data is determined to be satisfied.

3. The operand access accompanying the PREF, OCBP, OCBWB, and OCBI instructions are access without the data value; therefore, if the data value is included in the match conditions for these instructions, the match conditions will never be satisfied.

4. If the operand bus is selected, a break occurs after executing the instruction which has satisfied the conditions and immediately before executing the next instruction. However, if the data value is included in the match conditions, a break may occur after executing several instructions after the instruction which has satisfied the conditions; therefore, it is impossible to identify the instruction causing the break. If such a break has occurred for the delayed branch instruction or its delayed slot, the break does not occur until the first instruction at the branch destination.

   However, do not specify the operand break for the delayed slot of the RTE instruction. And if the data value is included in the match conditions, it is not allowed to set the break for the preceding the RTE instruction by one to six instructions.

35.3.5 Sequential Break

1. Sequential break conditions can be specified by setting the MFE and MFI bits in the match condition setting registers (CBR0 and CBR1). (Sequential break involves two cases such that channel 0 break condition is satisfied then channel 1 break condition is satisfied, and vice versa.) To use the sequential break function, clear the MFE bit of the match condition setting register and the BIE bit of the match operation setting register of the first channel in the sequence, and set the MFE bit and specify the number of the second channel in the sequence using the MFI bit in the match condition setting register of the second channel in the sequence. If the sequential break condition is set, the condition match flag is set every time the match condition is satisfied for each channel. When the condition has been satisfied for the first channel in the sequence but not for the second channel in the sequence, clear the condition match flag for the first channel in the sequence in order to release the first channel in the sequence from the match state.
2. For channel 1, the execution count break condition can also be included in the sequential break conditions.
3. If the match conditions for the first and second channels in the sequence are satisfied within a significantly short time, sequential operation may not be guaranteed in some cases, as shown below.

- When the match condition is satisfied at the instruction fetch cycle for both the first and second channels in the sequence:

Instruction B is 0 instruction after instruction A	Equivalent to setting the same addresses; do not use this setting.
Instruction B is one instruction after instruction A	Sequential operation is not guaranteed.
Instruction B is two or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the instruction fetch cycle for the first channel in the sequence whereas the match condition is satisfied at the operand access cycle for the second channel in the sequence:

Instruction B is 0 or one instruction after instruction A	Sequential operation is not guaranteed.
Instruction B is two or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the operand access cycle for the first channel in the sequence whereas the match condition is satisfied at the instruction fetch cycle for the second channel in the sequence:

Instruction B is 0 to five instructions after instruction A	Sequential operation is not guaranteed.
Instruction B is six or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the operand access cycle for both the first and second channels in the sequence:

Instruction B is 0 to five instructions after instruction A	Sequential operation is not guaranteed.
Instruction B is six or more instructions after instruction A	Sequential operation is guaranteed.

35.3.6 Program Counter Value to be Saved

When a break has occurred, the address of the instruction to be executed when the program restarts is saved in the SPC then the exception handling state is initiated. A unique instruction causing a break can be identified unless the data value is included in the match conditions.

1. When the instruction fetch cycle (before instruction execution) is specified as the match condition:

   The address of the instruction which has satisfied the match conditions is saved in the SPC. The instruction which has satisfied the match conditions is not executed, but a break occurs instead. However, if the match conditions are satisfied for the delayed slot instruction, the address of the delayed branch instruction is saved in the SPC.
2. When the instruction fetch cycle (after instruction execution) is specified as the match condition:

   The address of the instruction immediately after the instruction which has satisfied the match conditions is saved in the SPC. The instruction which has satisfied the match conditions is executed, then a break occurs before the next instruction. If the match conditions are satisfied for the delayed branch instruction or its delayed slot, these instructions are executed and the address of the branch destination is saved in the SPC.
3. When the operand access (address only) is specified as the match condition:

   The address of the instruction immediately after the instruction which has satisfied the break conditions is saved in the SPC. The instruction which has satisfied the match conditions are executed, then a break occurs before the next instruction. However, if the conditions are satisfied for the delayed slot, the address of the branch destination is saved in the SPC.
4. When the operand access (address and data) is specified as the match condition:

   If the data value is added to the match conditions, the instruction which has satisfied the match conditions is executed. A user break occurs before executing an instruction that is one through six instructions after the instruction which has satisfied the match conditions. The address of the instruction is saved in the SPC; thus, it is impossible to identify exactly where a break will occur. If the conditions are satisfied for the delayed slot instruction, the address of the branch destination is saved in the SPC. If a branch instruction follows the instruction which has satisfied the match conditions, a break may occur after the delayed instruction and delayed slot are executed. In this case, the address of the branch destination is also saved in the SPC.

35.4 User Break Debugging Support Function

By using the user break debugging support function, the branch destination address can be modified when the CPU accepts the user break request. Specifically, setting the UBDE bit of break control register CBCR to "1" allows branching to the address indicated by DBR instead of branching to the address indicated by the [VBR + offset].

Figure 35.2 shows the flowchart of the user break debugging support function.

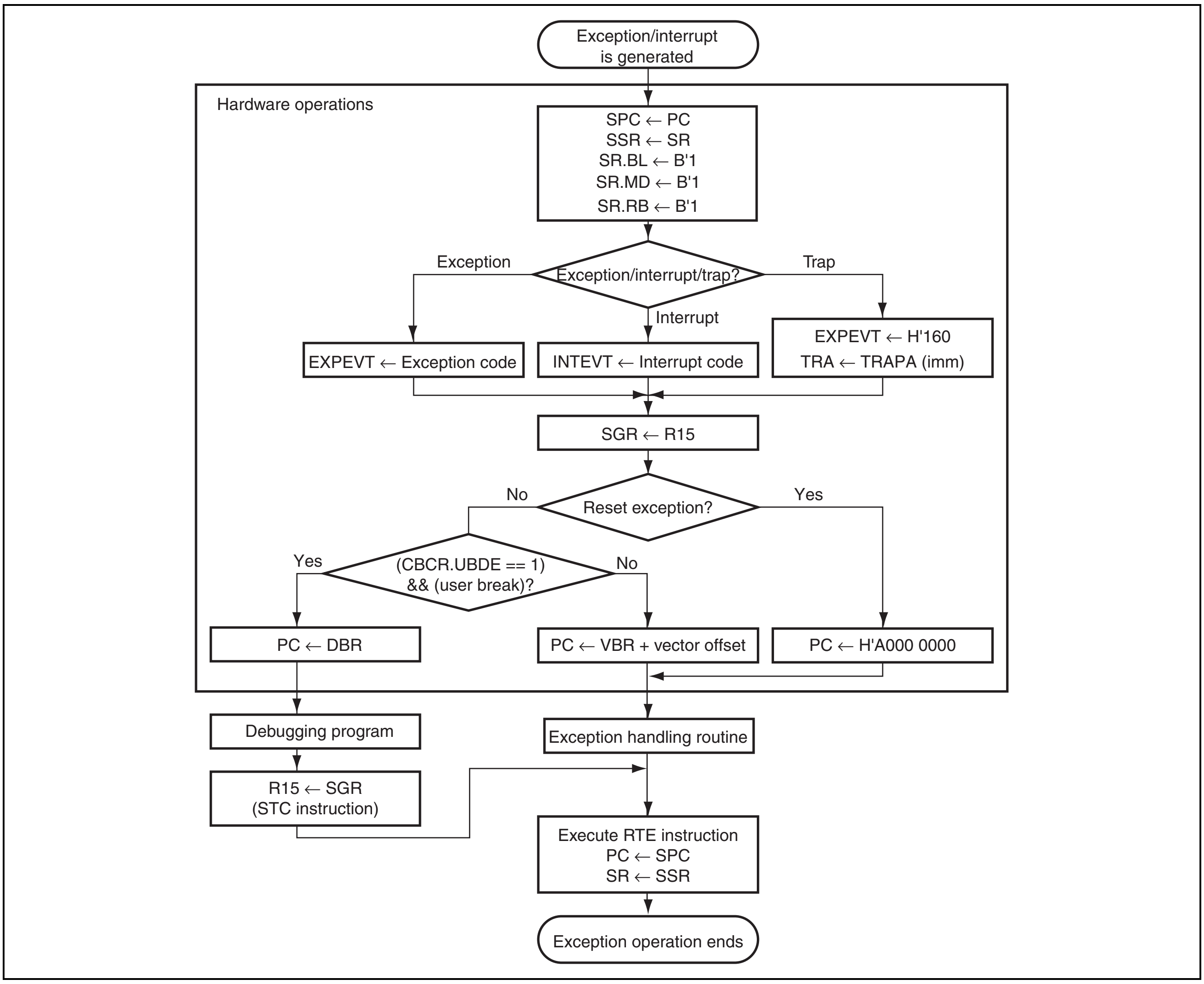


Figure 35.2 Flowchart of User Break Debugging Support Function

35.5 User Break Examples

(1) Match Conditions are Specified for an Instruction Fetch Cycle

- Example 1-1

  Register settings: CBR0 = H'0000 0013 / CRR0 = H'0000 2003 / CAR0 = H'0000 0404 / CAMR0 = H'0000 0000 / CBR1 = H'0000 0013 / CRR1 = H'0000 2001 / CAR1 = H'0000 8010 / CAMR1 = H'0000 0006 / CDR1 = H'0000 0000 / CDMR1 = H'0000 0000 / CETR1 = H'0000 0000 / CBCR = H'0000 0000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'0000 0404 / Address mask: H'0000 0000

  Bus cycle: Instruction fetch (after executing the instruction)

  ASID is not included in the conditions.

  — Channel 1

  Address: H'0000 8010 / Address mask: H'0000 0006

  Data: H'0000 0000 / Data mask: H'0000 0000 / Execution count: H'0000 0000

  Bus cycle: Instruction fetch (before executing instruction)

  ASID, data values, and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'0000 0404 or before executing the instruction at address H'0000 8010 to H'0000 8016.

- Example 1-2

  Register settings: CBR0 = H'4080 0013 / CRR0 = H'0000 2000 / CAR0 = H'0003 7226 / CAMR0 = H'0000 0000 / CBR1 = H'C070 0013 / CRR1 = H'0000 2001 / CAR1 = H'0003 722E / CAMR1 = H'0000 0000 / CDR1 = H'0000 0000 / CDMR1 = H'0000 0000 / CETR1 = H'0000 0000 / CBCR = H'0000 0000

  Specified conditions: Channel 0 → Channel1 sequential mode

  — Channel 0

  Address: H'0003 7226 / Address mask: H'0000 0000 / ASID: H'80

  Bus cycle: Instruction fetch (before executing the instruction)

  — Channel 1

  Address: H'0003 722E / Address mask: H'0000 0000 / ASID: H'70

  Data: H'0000 0000 / Data mask: H'0000 0000 / Execution count: H'0000 0000

  Bus cycle: Instruction fetch (before executing the instruction)

  Data values and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'0003 7226 where ASID is H'80 before executing the instruction at address H'0003 722E where ASID is H'70.

- Example 1-3

  Register settings: CBR0 = H'0000 0013 / CRR0 = H'0000 2001 / CAR0 = H'0002 7128 / CAMR0 = H'0000 0000 / CBR1 = H'0000 0013 / CRR1 = H'0000 2001 / CAR1 = H'0003 1415 / CAMR1 = H'0000 0000 / CDR1 = H'0000 0000 / CDMR1 = H'0000 0000 / CETR1 = H'0000 0000 / CBCR = H'0000 0000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'0002 7128 / Address mask: H'0000 0000

  Bus cycle: Instruction fetch (before executing the instruction)

  ASID is not included in the conditions.

— Channel 1

Address: H'0003 1415 / Address mask: H'0000 0000

Data: H'0000 0000 / Data mask: H'0000 0000 / Execution count: H'0000 0000

Bus cycle: Instruction fetch (before executing the instruction)

ASID, data values, and execution count are not included in the conditions.

With the above settings, the user break occurs for channel 0 before executing the instruction at address H'0002 7128. No user break occurs for channel 1 since the instruction fetch is executed only at even addresses.

- Example 1-4

  Register settings: CBR0 = H'4080 0013 / CRR0 = H'0000 2000 / CAR0 = H'0003 7226 / CAMR0 = H'0000 0000 / CBR1 = H'C070 0013 / CRR1 = H'0000 2001 / CAR1 = H'0003 722E / CAMR1 = H'0000 0000 / CDR1 = H'0000 0000 / CDMR1 = H'0000 0000 / CETR1 = H'0000 0000 / CBCR = H'0000 0000

  Specified conditions: Channel 0 → Channel 1 sequential mode

  — Channel 0

  Address: H'0003 7226 / Address mask: H'0000 0000 / ASID: H'80

  Bus cycle: Instruction fetch (before executing the instruction)

  — Channel 1

  Address: H'0003 722E / Address mask: H'0000 0000 / ASID: H'70

  Data: H'0000 0000 / Data mask: H'0000 0000 / Execution count: H'0000 0000

  Bus cycle: Instruction fetch (before executing the instruction)

  Data values and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'0003 7226 where ASID is H'80 and before executing the instruction at address H'0003 722E where ASID is H'70.

- Example 1-5

  Register settings: CBR0 = H'0000 0013 / CRR0 = H'0000 2001 / CAR0 = H'0000 0500 / CAMR0 = H'0000 0000 / CBR1 = H'0000 0813 / CRR1 = H'0000 2001 / CAR1 = H'0000 1000 / CAMR1 = H'0000 0000 / CDR1 = H'0000 0000 / CDMR1 = H'0000 0000 / CETR1 = H'0000 0005 / CBCR = H'0000 0000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'0000 0500 / Address mask: H'0000 0000

  Bus cycle: Instruction fetch (before executing the instruction)

  ASID is not included in the conditions.

  — Channel 1

  Address: H'0000 1000 / Address mask: H'0000 0000

  Data: H'0000 0000 / Data mask: H'0000 0000 / Execution count: H'0000 0005

  Bus cycle: Instruction fetch (before executing the instruction)

  Execution count: 5

  ASID and data values are not included in the conditions.

  With the above settings, the user break occurs for channel 0 before executing the instruction at address H'0000 0500. The user break occurs for channel 1 after executing the instruction at address H'0000 1000 four times; before executing the instruction five times.

- Example 1-6

  Register settings: CBR0 = H'4080 0013 / CRR0 = H'0000 2003 / CAR0 = H'0000 8404 / CAMR0 = H'0000 0FFF / CBR1 = H'4070 0013 / CRR1 = H'0000 2001 / CAR1 = H'0000 8010 / CAMR1 = H'0000 0006 / CDR1 = H'0000 0000 / CDMR1 = H'0000 0000 / CETR1 = H'0000 0000 / CBCR = H'0000 0000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'0000 8404 / Address mask: H'0000 0FFF / ASID: H'80

  Bus cycle: Instruction fetch (after executing the instruction)

  — Channel 1

  Address: H'0000 8010 / Address mask: H'0000 0006 / ASID: H'70

  Data: H'0000 0000 / Data mask: H'0000 0000 / Execution count: H'0000 0000

  Bus cycle: Instruction fetch (before executing the instruction)

  Data values and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'0000 8000 to H'0000 8FFE where ASID is H'80 or before executing the instruction at address H'0000 8010 to H'0000 8016 where ASID is H'70.

(2) Match Conditions are Specified for an Operand Access Cycle

- Example 2-1

  Register settings: CBR0 = H'4080 0023 / CRR0 = H'0000 2001 / CAR0 = H'0012 3456 / CAMR0 = H'0000 0000 / CBR1 = H'4070 A025 / CRR1 = H'0000 2001 / CAR1 = H'000A BCDE / CAMR1 = H'0000 00FF / CDR1 = H'0000 A512 / CDMR1 = H'0000 0000 / CETR1 = H'0000 0000 / CBCR = H'0000 0000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'0012 3456 / Address mask: H'0000 0000 / ASID: H'80

  Bus cycle: Operand bus, operand access, and read (operand size is not included in the conditions.)

  — Channel 1

  Address: H'000A BCDE / Address mask: H'0000 00FF / ASID: H'70

  Data: H'0000 A512 / Data mask: H'0000 0000 / Execution count: H'0000 0000

  Bus cycle: Operand bus, operand access, write, and word size

  Execution count is not included in the conditions.

  With these settings, the user break occurs for channel 0 for the following accesses: longword read access to address H'0012 3454, word read access to address H'0012 3456, byte read access to address H'0012 3456 where ASID is H'80. The user break occurs for channel 1 when word H'A512 is written to address H'000A BC00 to H'000A BCFE where ASID is H'70.

35.6 Usage Notes

1. A desired break may not occur between the time when the instruction for rewriting the UBC register is executed and the time when the written value is actually reflected on the register.

   After the UBC register is updated, execute one of the following three methods.

   A. Read the updated UBC register, and execute a branch using the RTE instruction.
   (It is not necessary that a branch using the RTE instruction is next to a reading UBC register.)

   B. Read the updated UBC register, and execute the ICBI instruction for any address (including non-cacheable area).
   (It is not necessary that the ICBI instruction is next to a reading UBC register.)

   C. Set 0 (the value after a reset) to IRMCR.R1 before updating the UBC register and update with following sequence.

   a. Write the UBC register.

   b. Read the UBC register which is updated at a.

   c. Write the value which is read at b to the UBC register.

Note: When two or more UBC registers are updated, executing these methods at each updating the UBC registers is not necessary. At only last updating the UBC register, execute one of these methods.

2. The PCB bit of the CRR0 and CRR1 registers is valid only when the instruction fetch is specified as the match condition.
3. If the sequential break conditions are set, the sequential break conditions are satisfied when the conditions for the first and second channels in the sequence are satisfied in this order. Therefore, if the conditions are set so that the conditions for channels 0 and 1 should be satisfied simultaneously for the same bus cycle, the sequential break conditions will not be satisfied, causing no break.
4. If the user break and other exceptions occur for the same instruction, they are determined according to the specified priority. For the priority, refer to section 5, Exception Handling. If the exception having the higher priority occurs, the user break does not occur.

   — The pre-instruction-execution break is accepted prior to any other exception.

   — If the post-instruction-execution break and data access break have occurred simultaneously with the re-execution type exception (including the pre-instruction-execution break) having a higher priority, only the re-execution type exception is accepted, and no condition match flags are set. When the exception handling has finished thus clearing the exception source, and when the same instruction has been executed again, the break occurs setting the corresponding flag.

   — If the post-instruction-execution break or operand access break has occurred simultaneously with the completion-type exception (TRAPA) having a higher priority, then no user break occurs; however, the condition match flag is set.
5. When conditions have been satisfied simultaneously and independently for channels 0 and 1, resulting in identical SPC values for both of the breaks, the user break occurs only once. However, the condition match flags are set for both channels. For example,

   Instruction at address 110 (post-instruction-execution break for instruction fetch for channel 0) → SPC = 112, CCMFR.MF0 = "1"

   Instruction at address 112 (pre-instruction-execution break for instruction fetch for channel 1) → SPC = 112, CCMFR.MF1 = "1"
6. It is not allowed to set the pre-instruction-execution break or the operand break in the delayed slot instruction of the RTE instruction. And if the data value is included in the match conditions of the operand break, do not set the break for the preceding the RTE instruction by one to six instructions.
7. If the re-execution type exception and the post-instruction-execution break are in conflict for the instruction requiring two or more execution states, then the re-execution type exception occurs. Here, the CCMFR.MF0 (or CCMFR.MF1) bit may or may not be set to "1" when the break conditions have been satisfied.

Section 36 AUD RAM Monitor (AUDR)

36.1 Overview

The AUD RAM monitor (AUDR) module provides RAM monitor and event output functions. The AUDR module includes a dedicated DMA circuit and accesses internal RAM and other areas by using this DMA circuit.

Table 36.1 lists the overview of the AUDR module.

Table 36.1 AUDR Module Overview

Item	Overview
Transfer method	Clock synchronous parallel interface (4 bits)
Transfer clock generation	Generated by the external host
Access area	Physical address area on the SuperHyway bus
Access size	8, 16, and 32 bits
Maximum transfer rate	12.5 MHz
I/O pins	7 pins (AUDRD3 to AUDRD0, AUDRCLK, AUDRSYN#, and AUDREVT#)
Functions	• RAM monitor function This function performs reads and writes for physical address areas that can be accessed from the SuperHyway bus. This function can reference and modify RAM data. It also supports access to peripheral modules over the peripheral bus and peripheral A bus. Also, the AUDR module can access ILRAM and OLRAM in the SH-4A core. Since the SH-4A internal cache memory and TLB are dedicated CPU resources, they cannot be accessed by AUDR. • Event output function This function outputs a "L" level from the AUDREVT# pin to report the occurrence of write accesses to the AUDREVNT register. • Configuration information retention (startup communication) function This function retains the values of the AUDRD3 to AUDRD0 pins at reset. It is used for communication with a RAM monitor tool. • Synchronous communication (message board) function This function is a flag register that is used by the firmware running on the CPU to communicate with a RAM monitor tool.

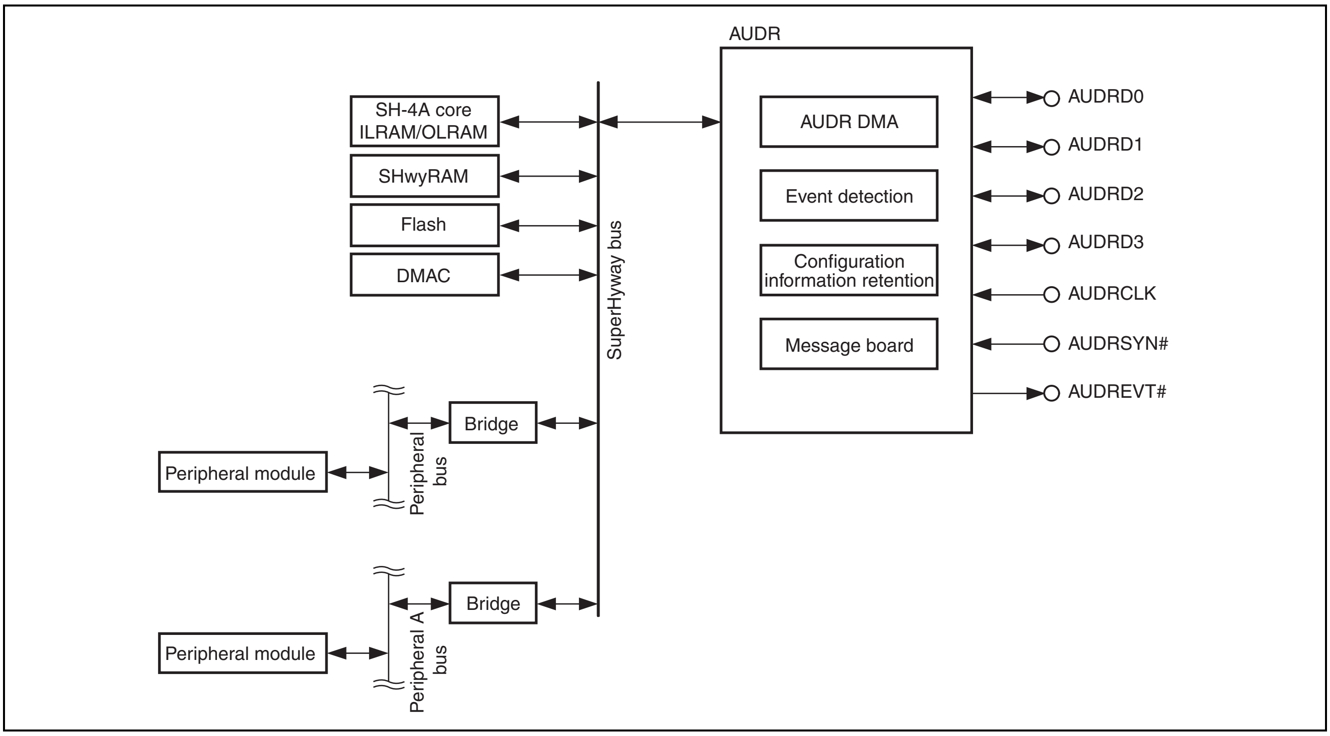


Figure 36.1 Block Diagram of AUDR

36.2 AUDR Module Usage Example

36.2.1 Example 1: RAM Monitor/Calibration

Figure 36.2 shows a simplified timing chart for RAM monitor/calibration operation.

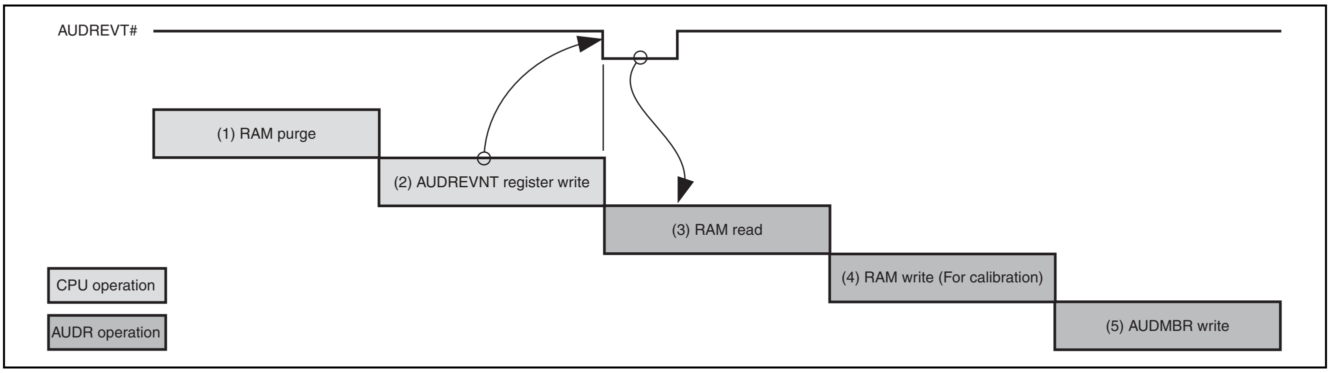


Figure 36.2 AUDR Module Usage Example (RAM monitor/calibration procedure)

1. When the AUDR module reads RAM data, it first performs a RAM purge, since it is not possible to determine if the data matches the cached data.
2. The RAM monitor tool is informed that step 1 has completed by event generation.
3. After the event is detected, the AUDR module performs the RAM read.
4. The AUDR module then performs a RAM write (calibration).
5. The CPU is informed, using the AUDRMBR register, that the AUDR module calibration has completed.

36.2.2 Example 2: Flash Memory Write

Figure 36.3 shows a simplified timing chart for the flash memory write operation.

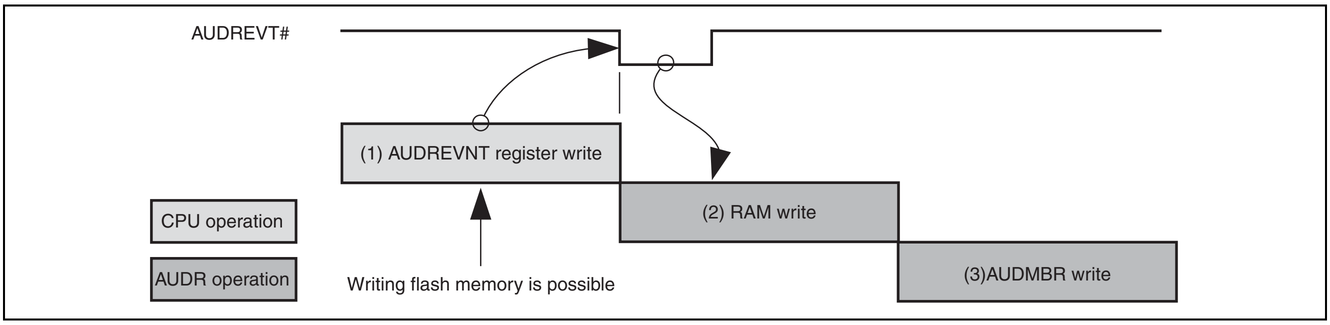


Figure 36.3 AUDR Module Usage Example (AUDR module flash memory write operation)

1. The CPU generates an event to indicate that writing flash memory is possible.
2. The AUDR module writes to flash memory.
3. When the flash memory write completes, the AUDR module reports that to the CPU using the AUDMBR register.

36.3 Input/Output Pins

Table 36.2 lists the AUDR module pins.

When using each AUDR pin, the pin settings are required using the pin function unit. For details, see section 18, I/O Ports and Pin Function Unit. Do not access any pin until the settings have been completed by a program.

Table 36.2 Pin Configuration

Pin Name	I/O	Function
AUDRCLK	Input	Synchronization clock input Inputs an external clock signal. Input the clock to be used for debugging to this pin. Frequencies up to 12.5 MHz can be used.
AUDRSYN#	Input	Data start position recognition signal input 1: Read data output 0: Write address, data, and DIR field command input, ready flag output Note: • Do not assert this pin from the external input of the command to the AUDRD pin until the required data has been output. See the protocol description later in this section for details.
AUDRD3 to AUDRD0	I/O	Command, address, and data I/O The following data is input and output in a time multiplexed manner. Command Addresses Data When an external command is input, data is output after a ready transmission. The output starts after the AUDRSYN pin is negated. See the protocol description later in this section for details. This pin must be pulled up externally.
AUDREVT#	Output	Event output (This pin outputs a low level with a 2Pck width when an event occurs.)

36.4 Register Descriptions

Table 36.3 lists the AUDR module related registers.

Table 36.3 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
AUDR enable register	AUDRENB	H'0000	H'FE40 0000	16	36-4
AUDR event generation register	AUDREVNT	Undefined	H'FE40 0008	8	36-10
AUDR configuration information retention register	AUDISR	Undefined*[1]	H'FE40 0010	16	36-11
AUDR message board register	AUDMBR	H'0000	H'FE40 0018	16	36-12

Notes: *1 The AUDISR register retains the values of the AUDRD3 to AUDRD0 pins when a reset is cleared. See section 36.7.2, AUDR Configuration Information Retention Register (AUDISR), for details.

- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

36.4.1 AUDR Enable Register (AUDRENB)

Since the AUDRENB register has this write key, writes must be performed in word units. To rewrite the value of the AUDRENB bit, the value H'17 must be written to the AUDREKEY field at the same time. Write operations are ignored if performed in byte units or if any value other than H'17 is written to the AUDREKEY field. The AUDRENB register can be read out in either word or byte units. Note, however, that since write data is not stored in the AUDREKEY field, this field always returns H'00 when read.

AUDR enable register (AUDRENB) <P4 address: location H'FE40 0000>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AUDREKEY								—	—	—	—	—	—	—	AUDR EN
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	AUDREKEY	All 0	0	W	AUDRENB Write Key Code Bit These bits enable or disable AUDREN bit modification. The data written to these bits are not retained. These bits are always read as "0". H'17: Enable AUDREN bit modification Other than H'17: Disable AUDREN bit modification
7 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	AUDREN	0	R	W	AUDR Operation Enable Bit Switches the AUDR module functions between the enabled and disabled states. When the AUDREN bit is set to "0", the AUDR module goes to the reset state and the registers are returned to their reset cleared values. When using the AUDR module functions, this bit must first be set to "1" before setting any other AUDR module registers. When the AUDREN bit is "0", access to AUDR module registers other than the AUDRENB register is disabled. 0: AUDR module operation is disabled 1: AUDR module operation is enabled

Note: • When changing the value of the AUDREN bit from the disabled to the enabled state, a wait period of 6 AUDRCLK cycles is required before the AUDR module can operate.

36.5 RAM Monitor Functions

36.5.1 Communication Protocol

When the AUDRSYN# pin is asserted, the AUDR module acquires the values input to the AUDRD pin. Use the format shown in figure 36.4 to input a command, address, and data to the AUDRD pin.

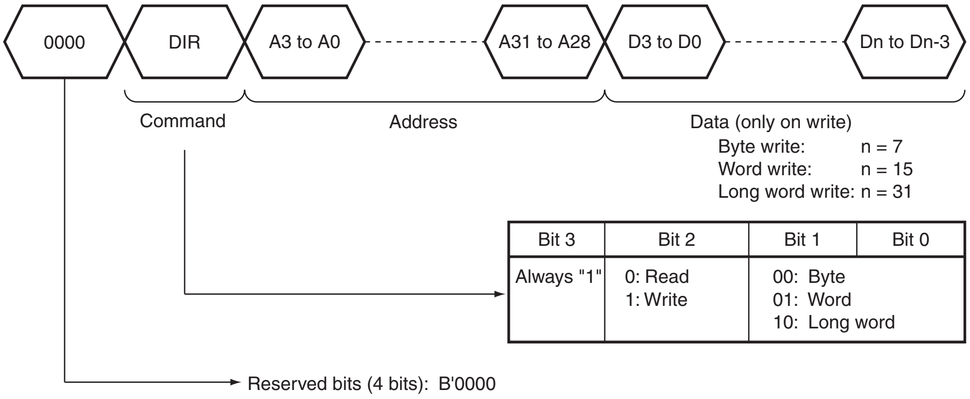


Figure 36.4 AUDRD Input Format

36.5.2 Operation

Figure 36.5 shows an example of a read operation and figure 36.6 shows an example of a write operation.

When the AUDRSYN# pin is asserted, the AUDR module starts acquiring input from the AUDRD pin. When the command, address, and data (only for write) have been input in the format shown in figure 36.4, execution of a read or write for the specified address starts. During internal execution, the AUDR module will return "not ready" ("0000"). When execution completes, the ready flags ("0001") are returned. (See figures 36.5 and 36.6.) Table 36.4 shows the format of the ready flags.

When a read is performed, after detection of this flag, the specified size data will be output when the AUDRSYN# pin is negated. (See figure 36.5.) If a command other than one of the above is input as the DIR field, the AUDR module will cancel the processing as a command error and set the CFLG bit in the ready flags to "1". Also, if the read or write operation due to the command specified in the DIR field causes a bus error, the AUDR module will cancel the processing and set the BFLG bit in the ready flags to "1". (See figure 36.7.)

Table 36.4 Ready Flag Format Bit Positions

Bit Position	Bit Name	Function	Content
AUDRD3	0	—	—
AUDRD2	BFLG	Indicates that a bus error occurred.	0: Normal state 1: A bus error occurred
AUDRD1	CFLG	Indicates that a command error occurred.	0: Normal state 1: A command error occurred
AUDRD0	RFLG	Indicates that an AUDR module operation completed.	0: Not Ready 1: Ready

(1) Command Error Conditions

Table 36.5 lists the command error conditions.

Table 36.5 Command (DIR) Conditions

Bit 3 (IT)	Bit 2 (RW)	Bit 1 (SIZ1)	Bit 0 (SIZ0)	Content
0	x	x	x	Command error
1	0	0	0	Read (byte)
1	0	0	1	Read (word)
1	0	1	0	Read (long word)
1	1	0	0	Write (byte)
1	1	0	1	Write (word)
1	1	1	0	Write (long word)
x	x	1	1	Command error

(2) Bus Error Conditions

1. Word access to a location of the form 4n + 1 or 4n + 3
2. Long word access to a location of the form 4n + 1, 4n + 2, or 4n + 3
3. An error response was received from the SuperHyway bus.

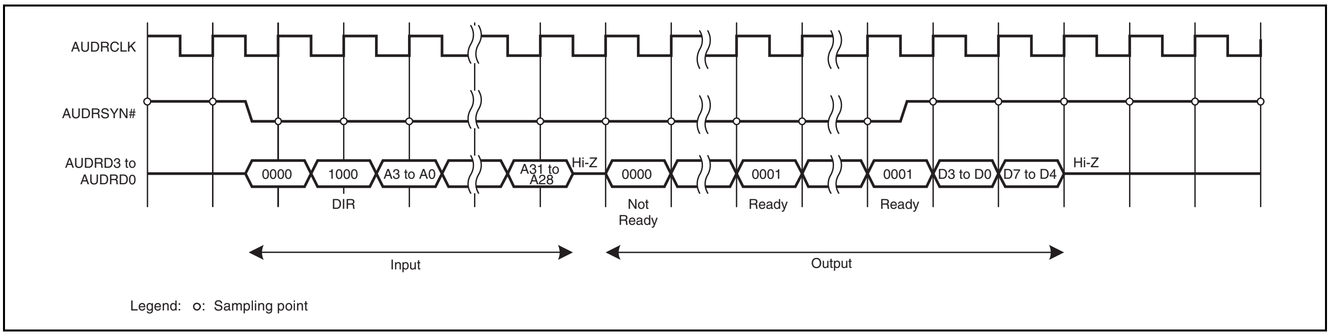


Figure 36.5 Read Operation Example (byte read)

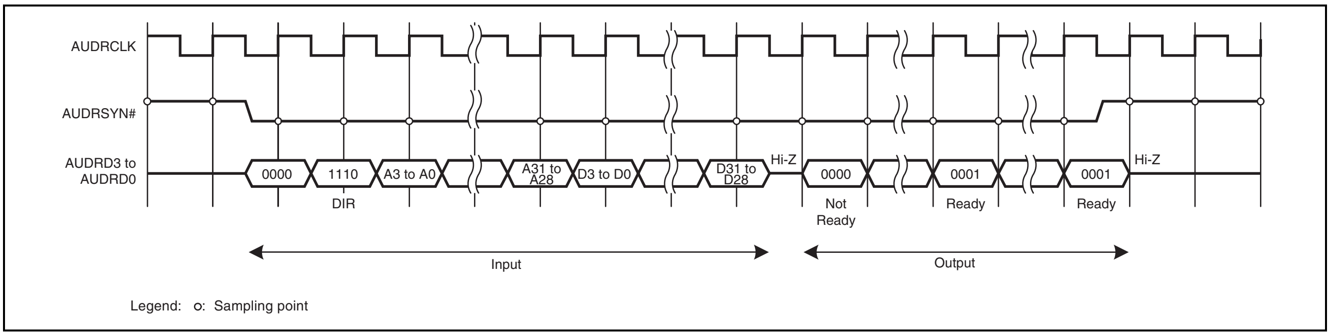


Figure 36.6 Write Operation Example (long word write)

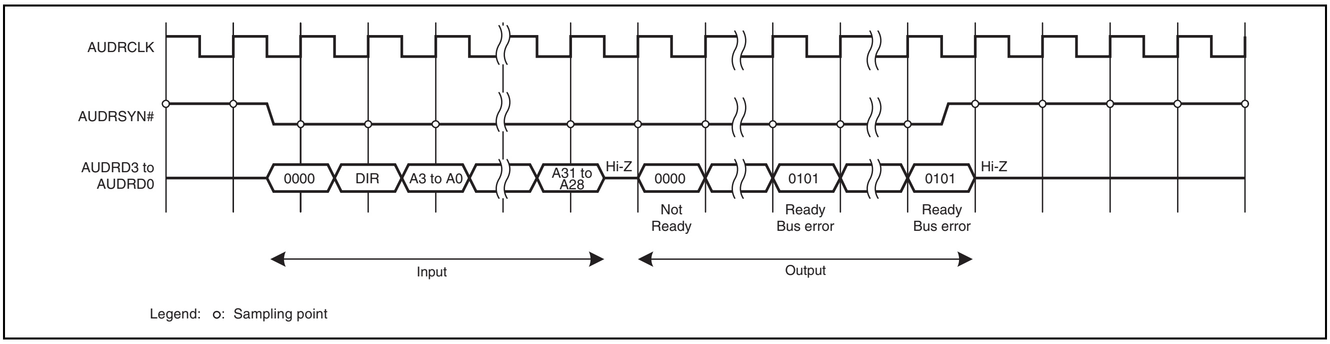


Figure 36.7 Error Example (long word read)

36.5.3 AUDRD Data Format

This section describes the format of the data input to the AUDRD pin.

(1) Input Format

Table 36.6 Input Format Bit Positions

		Bit Position				O: Required, —: Not required			
Input Order	Format Name	AUDRD3	AUDRD2	AUDRD1	AUDRD0	Read	Byte write	Word write	Long word write
First	Reserved bits	aux3	aux2	aux1	aux0	O	O	O	O
	DIR command	IT	RW	SIZ1	SIZ0	O	O	O	O
	Address	A3	A2	A1	A0	O	O	O	O
		A7	A6	A5	A4	O	O	O	O
		A11	A10	A9	A8	O	O	O	O
		A15	A14	A13	A12	O	O	O	O
		A19	A18	A17	A16	O	O	O	O
		A23	A22	A21	A20	O	O	O	O
		A27	A26	A25	A24	O	O	O	O
		A31	A30	A29	A28	O	O	O	O
	Data (only for write)	D3	D2	D1	D0	—	O	O	O
		D7	D6	D5	D4	—	O	O	O
		D11	D10	D9	D8	—	—	O	O
		D15	D14	D13	D12	—	—	O	O
		D19	D18	D17	D16	—	—	—	O
		D23	D22	D21	D20	—	—	—	O
		D27	D26	D25	D24	—	—	—	O
Last		D31	D30	D29	D28	—	—	—	O

- Reserved bits

Bit Name	Function	Description
aux3	Future expansion	This bit must be set to "0".
aux2	Future expansion	This bit must be set to "0".
aux1	Future expansion	This bit must be set to "0".
aux0	Future expansion	This bit must be set to "0".

Note: • Operation is not guaranteed if any other values are used.

- DIR command

Bit Name	Function	Description
IT	Access space specification	This bit must be set to "1".*[1]
RW	Read/write specification	0: Read 1: Write
SIZ[1:0]	Access size specification	00: Byte (8 bits) 01: Word (16 bits) 10: Long word (32 bits) 11: Setting prohibited

Note: *1 Operation is not guaranteed if any other value is used.

- Address format

Bit Name	Function	Description
A31 to A0	Address specification	Specifies the address to be accessed.

- Data format (only for write)

Bit Name	Function	Description
D31 to D0	Write data specification	The number of required bits changes depending on the control field RW bit and SIZ[1:0] field specifications. (See table 36.6 for details.)

(2) Ready Flag Format

See table 36.4 for the ready flag format.

(3) Read Data Format (output)

Table 36.7 Read Data Format Bit Positions

Output Order	Bit Positions				O: Required, — : Not required			
	AUDRD3	AUDRD2	AUDRD1	AUDRD0	Byte read	Word read	Long word read	Write
First	D3	D2	D1	D0	O	O	O	—
	D7	D6	D5	D4	O	O	O	—
	D11	D10	D9	D8	—	O	O	—
	D15	D14	D13	D12	—	O	O	—
	D19	D18	D17	D16	—	—	O	—
	D23	D22	D21	D20	—	—	O	—
	D27	D26	D25	D24	—	—	O	—
Last	D31	D30	D29	D28	—	—	O	—

Bit Name	Function	Description
D31 to D0	Read data output	The number of bits output changes depending on the control field RW bit and SIZ[1:0] field specifications. (See table 36.7 for details.)

36.5.4 RAM Monitor Function Usage Notes

- Do not negate the AUDRSYN# pin from the start of AUDRD pin command input and until ready is returned.

36.6 Event Detection Function

The AUDR module provides an event output function that detects the occurrence of write accesses to the AUDREVNT register. The "L" level is the valid level in the event output, and the valid period is 2Pck.

36.6.1 AUDR Event Generation Register (AUDREVNT)

AUDR event generation register (AUDREVNT) <P4 address: location H'FE40 0008>

Bit:	7	6	5	4	3	2	1	0
	AUDREVNT							
After Reset:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

<After Reset: Undefined >

Bit	Abbreviation	After Reset	R	W	Description
7 to 0	AUDREVNT	Undefined	?	W	When an arbitrary value is written to this register, a 2Pck width "L" level is output from the AUDREVT# pin. Undefined data values are returned when this field is read.

Note: • When multiple events occur close together temporally, there are cases when only a 2Pck width "L" level (the amount for one event) will be output from the AUDREVT# pin. There are also cases when 2Pck width "L" level periods will be output consecutively due to the event occurrence conditions.

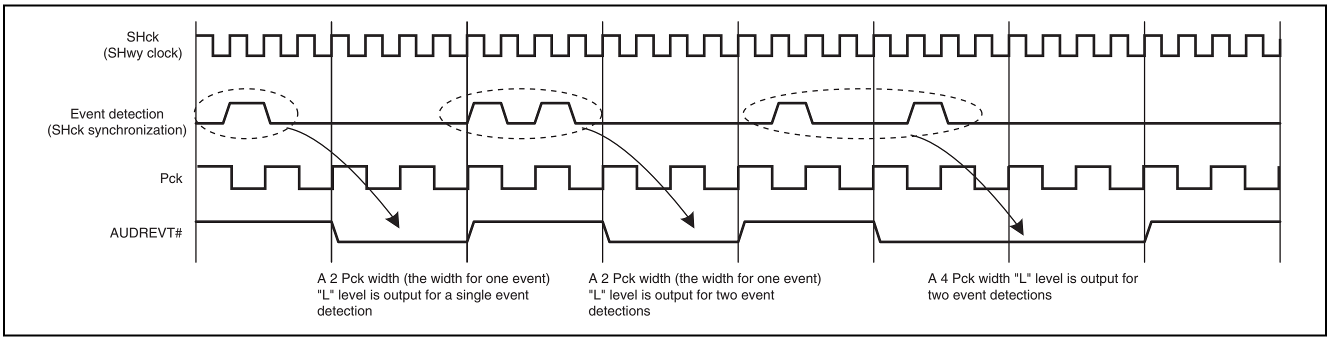

SHck
(SHwy clock)
Event detection
(SHck synchronization)
Pck
AUDREVT#
A 2 Pck width (the width for one event) "L" level is output for a single event detection
A 2 Pck width (the width for one event) "L" level is output for two event detections
A 4 Pck width "L" level is output for two event detections

Figure 36.8 AUDR Event Detection and AUDREVT# Pin Operation

36.7 Configuration Information Retention Function

This function stores the values at reset of the AUDRD3 to AUDRD0 pins.

36.7.1 Block Diagram

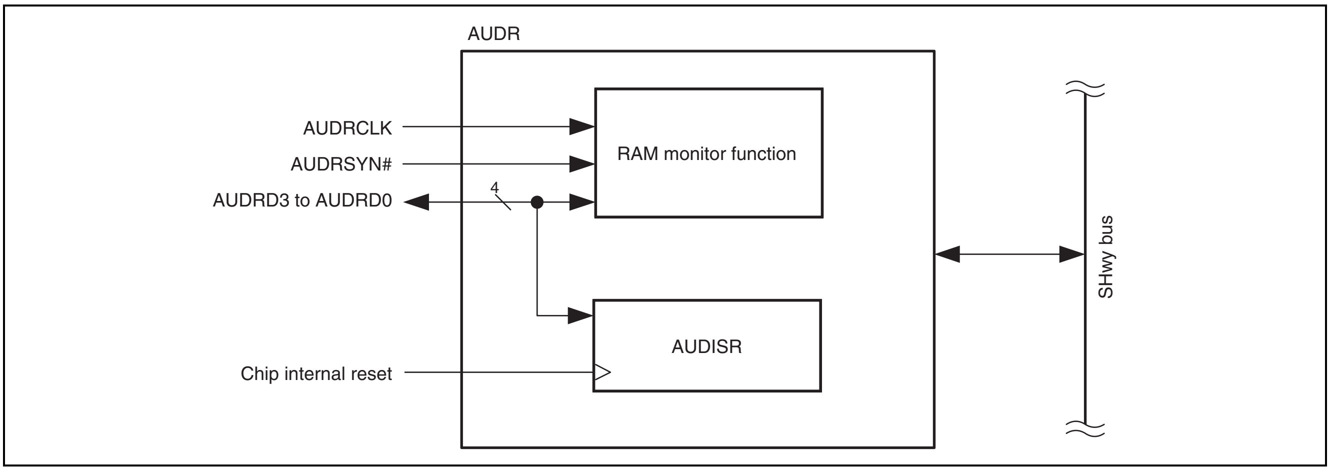


Figure 36.9 Configuration Information Retention Function Block Diagram

36.7.2 AUDR Configuration Information Retention Register (AUDISR)

AUDR configuration information retention register (AUDISR) <P4 address: location H'FE40 0010>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	DATA			
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	Undefined*1	Undefined*1	Undefined*1	Undefined*1

<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
15 to 4	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
3 to 0	DATA	Undefined*1	R	—	This field stores the values of the AUDRD3 to AUDRD0 pins after a reset is cleared. The AUDRD3 to AUDRD0 pins must be pulled up externally when a debugging or other tool is not connected.

Note: *1 When exiting the reset state, the values of these bits are determined by the states of the corresponding pins.

36.7.3 Operation

The values of the AUDRD pins after a reset are stored in the AUDISR register.

36.8 Synchronous Communication (message board) Function

The message board is a flag register that the firmware running on the CPU can use to communicate with a RAM monitor tool. The register used by this function can be accessed by both the CPU and the AUDR RAM monitor function.

36.8.1 Block Diagram

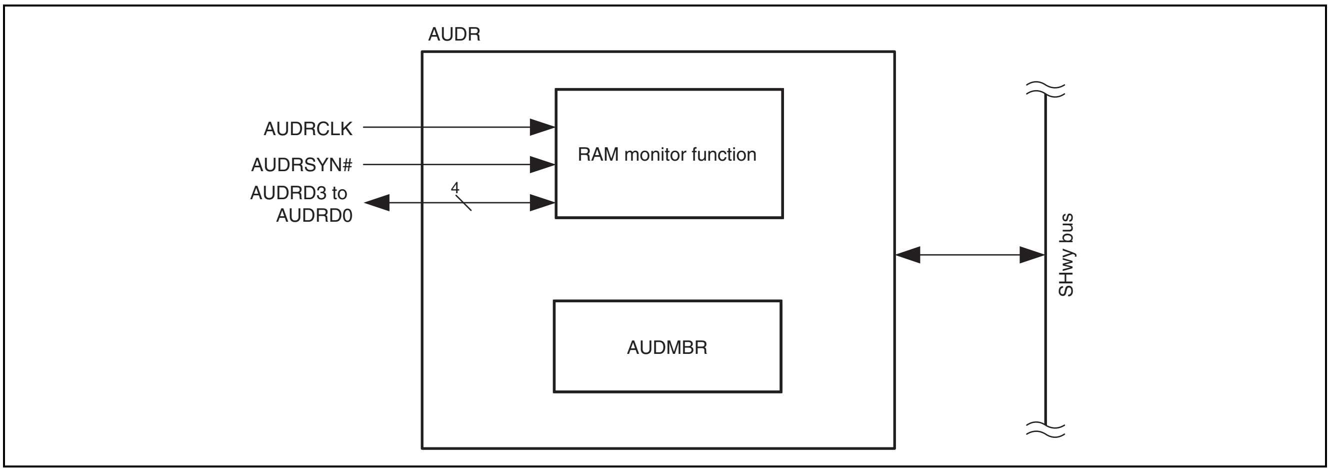


Figure 36.10 Synchronous Communication (AUDR message board register) Block Diagram

36.8.2 AUDR Message Board Register (AUDMBR)

All bits in the AUDMBR register are cleared when it is read by the AUDR module. It is not cleared by being read by the CPU.

This register can be written by both the AUDR module and the CPU. However, only the value "1" can be written to the bits in this register; writes of "0" are ignored. If read by the AUDR module with a size other than word (16 bits), the AUDMBR register is not cleared.

AUDR message board register (AUDMBR) <P4 address: location H'FE40 0018>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AUDMBR															
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 0	AUDMBR	All 0	R	W	Flags for communication between the RAM monitor and CPU*[1]

Note: *1 The access methods are listed in table 36.8.

Table 36.8 AUDMBR Register Access Methods

Host	Access Direction	Accessible or not	Notes
AUDR	Write	Only "1" can be written. Writes of "0" are ignored.	—
	Read	Can be read	When the AUDMBR register is read with a word size access, it is cleared after the read.
CPU	Write	Only "1" can be written. Writes of "0" are ignored.	—
	Read	Can be read	There is no clear operation.

36.8.3 Synchronous Communication Function Usage Notes

(1) AUDR and CPU Access Contention

The AUDMBR register can be access from the CPU and the AUDR module (RAM monitor function) over the SuperHyway bus. Therefore exclusive control for accesses from the CPU and the AUDR module is implemented by the SuperHyway bus functions.

This page is blank for reasons of layout.

Section 37 User Debugging Interface (H-UDI)

The H-UDI is a serial interface which conforms to the JTAG (IEEE 1149.1: IEEE Standard Test Access Port and Boundary-Scan Architecture) standard. The H-UDI is also used for emulator connection.

37.1 Overview

The H-UDI is a serial interface which conforms to the JTAG standard. The H-UDI is also used for emulator connection. Refer to the appropriate emulator users manual for the method of connecting the emulator.

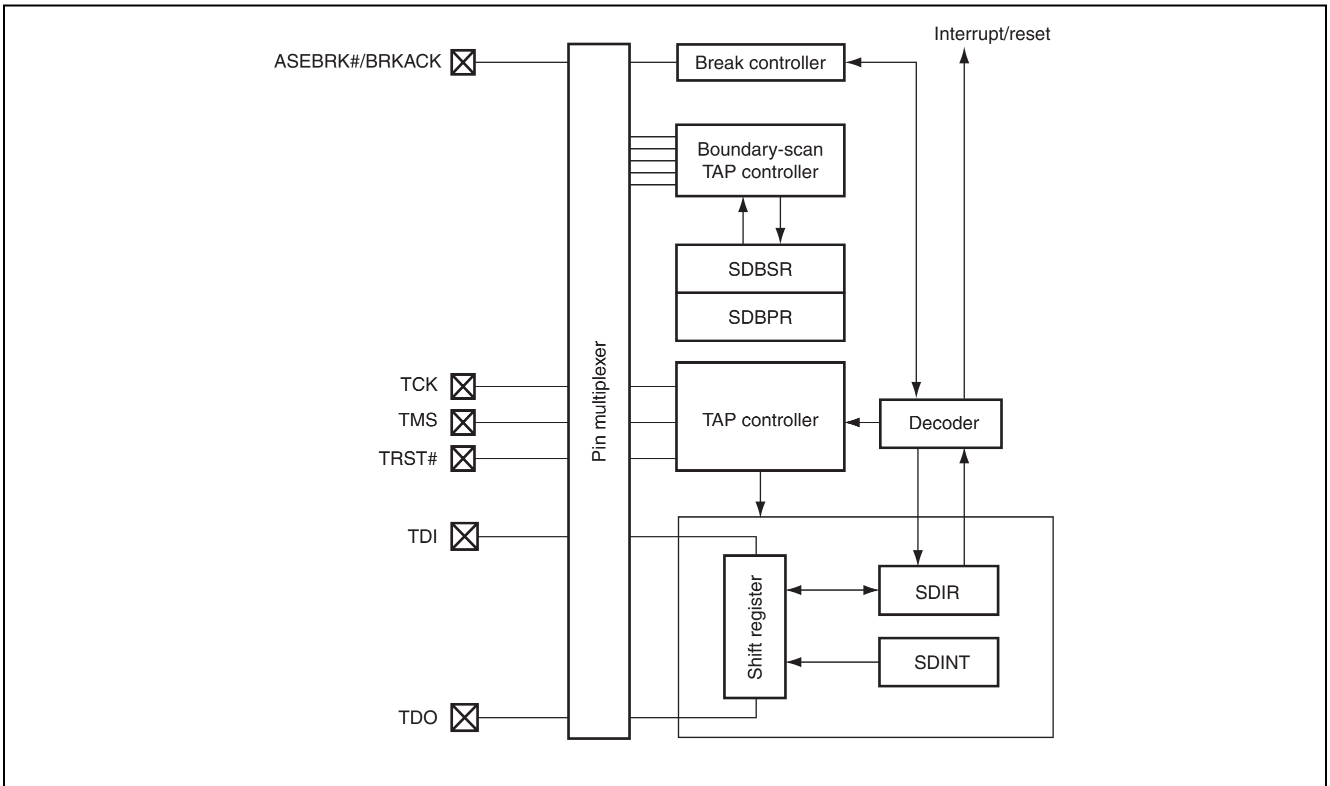
The H-UDI has six pins: TCK, TMS, TDI, TDO, TRST#, and ASEBRK#/BRKACK. The pin functions except ASEBRK#/BRKACK and serial communications protocol conform to the JTAG standard. Additionally, this MCU also has a single signal (MPMD) for use as a chip mode specification pin.

The H-UDI has two TAP controller blocks; one is for the boundary-scan test and another is H-UDI function except the boundary-scan test. Since the boundary scan TAP controller is selected by a "L" level input to the TRST# pin, including when power is first applied, it is necessary to input a switching command to use the H-UDI functions. And the CPU cannot access the boundary scan TAP controller.

The H-UDI has the TAP (Test Access Port) controller and four registers (SDBPR, SDBSR, SDIR, and SDINT). The SDBPR register supports the JTAG bypass mode, the SDBSR register supports the JTAG boundary scan mode, SDIR is used for commands, and the SDINT register is used for H-UDI interrupts. SDIR is directly accessed from the TDI and TDO pins.

The TAP (Test Access Port) controller and control registers, as well as the boundary scan TAP controller, are independent of the reset pin. These circuits are reset by setting either the TRST# pin to the "L" level or the TMS pin to the "H" level and hold that state for at least 5 cycles of the TCK signal. Other circuits are initialized by the reset applied during the normal reset period.

Figure 37.1 shows the block diagram of the H-UDI module.



Legend:

SDBPR:	Bypass register
SDBSR:	Boundary scan register
SDINT:	Interrupt source register
SDIR:	Instruction register

Figure 37.1 Block Diagram of H-UDI

37.2 Input/Output Pins

Table 37.1 shows the pin configuration for the H-UDI.

These pins are not multiplexed with other functions.

Table 37.1 Pin Configuration

Pin Name	I/O	Description	When Not in Use
TCK	Input	Functions as the serial clock input pin stipulated in the JTAG standard. Data input to the H-UDI via the TDI pin or data Output via the TDO pin is performed in synchronization with this signal.	Pull-up
TMS	Input	Mode Select Input Changing this signal in synchronization with the TCK signal determines the significance of data input via the TDI pin. Its protocol conforms to the JTAG standard (IEEE standard 1149.1).	Pull-up
TRST#*2	Input	H-UDI Reset Input This signal is received asynchronously with a TCK signal. Asserting this signal resets the JTAG interface circuit. Regardless of whether or not JTAG is used, the TRST# pin must be set to the "L" level for a fixed period when power is first applied. Note that this differs from the IEEE JTAG stipulations.	*3
TDI	Input	Data Input Data is sent to the H-UDI by changing this signal in synchronization with the TCK signal.	Pull-up
TDO	Output	Data Output Data is read from the H-UDI in synchronization with the TCK signal.	Pull-up or open
ASEBRK#/ BRKACK	I/O	Pins for an emulator	Pull-up
MPMD	Input	Selects the operation mode of this MCU, whether emulation support mode (Low level) or MCU operation mode (High level).	Open*1

Notes: *1 This pin is pulled up in this MCU. When using interrupts or resets via the H-UDI or emulator, the use of external pull-up resistors will not cause any problem.

*2 When using interrupts or resets via the H-UDI or emulator, the TRST# pin should be designed so that it can be controlled independently and can be controlled to retain "L" level while the RESET# pin is asserted at a hardware reset.

*3 Pull each pin low to Vss via a 0 to 100 kΩ resistor or connect to RESET#.

Set the TCK (and TMS, TDI, and TDO) frequencies to be lower than this MCU peripheral clock frequency. See section 38, Electrical Characteristics, for the maximum operating frequency for which this product is guaranteed.

37.3 Boundary Scan TAP Controllers (IDCODE, EXTEST, SAMPLE/PRELOAD, BYPASS, CLAMP, and HIGHZ)

The H-UDI contains two separate TAP controllers: one for controlling the boundary-scan function and another for controlling the H-UDI reset and interrupt functions. Assertion of TRST#, for example at hardware reset, activates the boundary-scan TAP controller and enables the boundary-scan function prescribed in the JTAG standards. Executing a switchover command to the H-UDI allows usage of the H-UDI reset and H-UDI interrupts. This MCU, however, has the following limitations:

- Clock-related pins (EXTAL and XTAL) are out of the scope of the boundary-scan test.
- Reset-related pin (RESET#) is out of the scope of the boundary-scan test.
- H-UDI-related pins (TCK, TDI, TDO, TMS, TRST# and MPMD) are out of the scope of the boundary-scan test.
- During the boundary scan (IDCODE, EXTEST, SAMPLE/PRELOAD, BYPASS, CLAMP, HIGHZ and H-UDI switchover command), the maximum TCK signal frequency is 2 MHz.
- The external controller has 4-bit access to the boundary-scan TAP controller via the H-UDI.

Note: • During the boundary scan, the RESET# pin should be fixed "L" level. Figure 37.2 shows a sequence for switching from boundary-scan TAP controller to H-UDI.

Table 37.2 Commands Supported by Boundary-Scan TAP Controller

Bit 3	Bit 2	Bit 1	Bit 0	Description
1	1	1	1	BYPASS
0	0	0	0	EXTEST
0	0	0	1	SAMPLE/PRELOAD
0	1	0	0	IDCODE
0	1	1	0	CLAMP
0	1	1	1	HIGHZ
0	0	1	1	H-UDI (switchover command)
Other than above				Reserved (BYPASS)

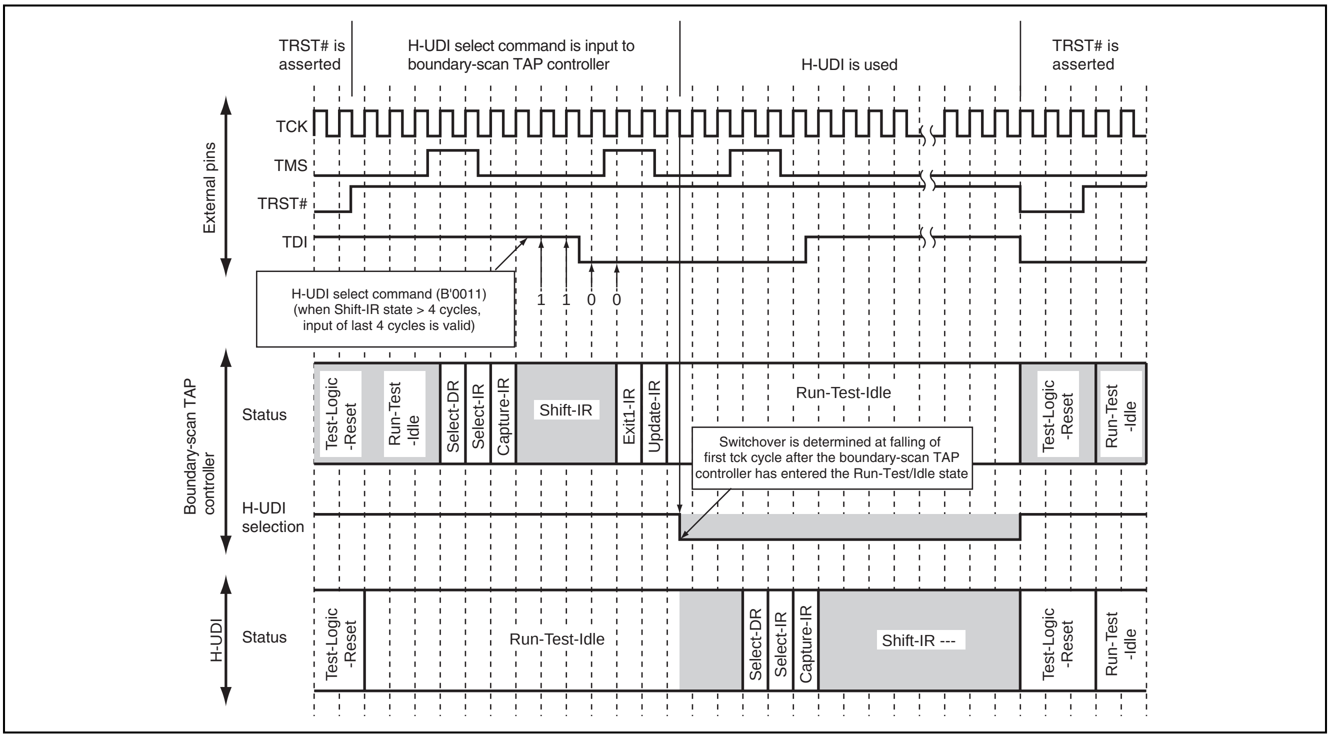


Figure 37.2 Sequence for Switching from Boundary-Scan TAP Controller to H-UDI

37.4 Register Descriptions

Tables 37.3 and 37.4 show the pin configuration of the H-UDI.

Table 37.3 Register Configuration (1)

		CPU Side			
Register Name	**Abbreviation**	**After Reset*[1]**	**P4 Address**	**Size**	**Page**
Instruction register	SDIR	H'0EFF	H'FC11 0000	16	37-7
Interrupt source register	SDINT	H'0000	H'FC11 0018	16	37-8
Boundary scan register	SDBSR	—	—	—	37-9
Bypass register	SDBPR	Undefined	—	—	37-8

Notes: *1 The "L" level of the TRST# pin or the Test-Logic-Reset state of the TAP controller initializes to these values.

- The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

Table 37.4 Register Configuration (2)

		H-UDI Side	
Register Name	**Abbreviation**	**After Reset*[1]**	**Size**
Instruction register	SDIR	H'FFFF FFFD (fixed value*[2])	32
Interrupt source register	SDINT	H'0000 0000	32
Boundary scan register	SDBSR	—	—
Bypass register	SDBPR	Undefined	1

Notes: *1 The "L" level of the TRST# pin or the Test-Logic-Reset state of the TAP controller initializes to these values.

*2 When reading via the H-UDI, the value is always H'FFFF FFFD.

37.4.1 Instruction Register (SDIR)

Commands are set via the serial input (TDI). SDIR is initialized by TRST# or in the Test-Logic-Reset state and can be written by the H-UDI irrespective of the CPU mode. Operation is not guaranteed when a reserved command is set to this register.

Instruction Register (SDIR) <P4 address: location H'FC11 0000>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	T1								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1

<After Reset: H'0EFF>

Bit	Abbreviation	After Reset	R	W	Description
15 to 8	TI	0000 1110	R	—	Test Instruction Bits 0110 xxxx: Negate H-UDI reset 0111 xxxx: Assert H-UDI reset 101x xxxx: H-UDI interrupt 0000 1110: Initial state Other than above: Setting prohibited
7 to 0	—	All 1	1	1	Reserved Bits These bits are always read as "1". The write value should always be "1".

Note: • The clock generator (CPG) and watchdog timer (WDT) modules are not initialized by an H-UDI reset.

37.4.2 Interrupt Source Register (SDINT)

Specifying an H-UDI interrupt command in SDIR via H-UDI pin (Update-IR) sets the INTREQ bit to "1". While an H-UDI interrupt command is set in SDIR, SDINT which is connected between the TDI and TDO pins can be read as a 32-bit register. In this case, the upper 16 bits will be "0" and the lower 16 bits represent the SDINT value.

Only "0" can be written to the INTREQ bit by the CPU. While this bit is set to "1", an interrupt request will continue to be generated. This bit, therefore, should be cleared by the interrupt handling routine. It is initialized by TRST# or in the Test-Logic-Reset state.

Interrupt Source Register (SDINT) <P4 address: location H'FC11 0018>

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTREQ
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

<After Reset: H'0000>

Bit	Abbreviation	After Reset	R	W	Description
15 to 1	—	All 0	0	0	Reserved Bits These bits are always read as "0". The write value should always be "0".
0	INTREQ	0	R	W	Interrupt Request Bit Indicates whether or not an interrupt by an H-UDI interrupt command has occurred. Clearing this bit to "0" by the CPU cancels an interrupt request. When writing "1" to this bit, the previous value is maintained.

37.4.3 Bypass Register (SDBPR)

The SDBPR register is a one-bit register that supports the J-TAG bypass mode. When the BYPASS command is set to the boundary scan TAP controller, the TDI and TDO are connected by way of SDBPR. This register cannot be accessed from the CPU with all operating modes. Though this register is not initialized by a hardware reset and the TRST# pin asserted, initialized to "0" in the Capture-DR state.

37.4.4 Boundary Scan Register (SDBSR)

The SDBSR register is a shift register, located on the PAD, for controlling the input/Output pins, which supports the boundary scan mode of the JTAG standard.

Using the EXTEST and SAMPLE/PRELOAD commands, a boundary-scan test complying with the JTAG standards (IEEE1149.1) can be carried out.

This register cannot be accessed from the CPU with all operating modes.

This register is not initialized by a hardware reset, nor is it initialized by a "L" level input to the TRST# pin.

37.5 Operation

37.5.1 TAP Control

Figure 37.3 shows the internal states of the TAP controller. The state transitions basically conform to the JTAG standard.

- State transitions occur according to the TMS value at the rising edge of the TCK signal.
- The TDI value is sampled at the rising edge of the TCK signal and shifted at the falling edge of the TCK signal.
- The TDO value is changed at the falling edge of the TCK signal. The TDO signal is in a Hi-Z state other than in the Shift-DR or Shift-IR state.
- A transition to the Test-Logic-Reset by clearing TRST# to "0" is performed asynchronously with the TCK signal.

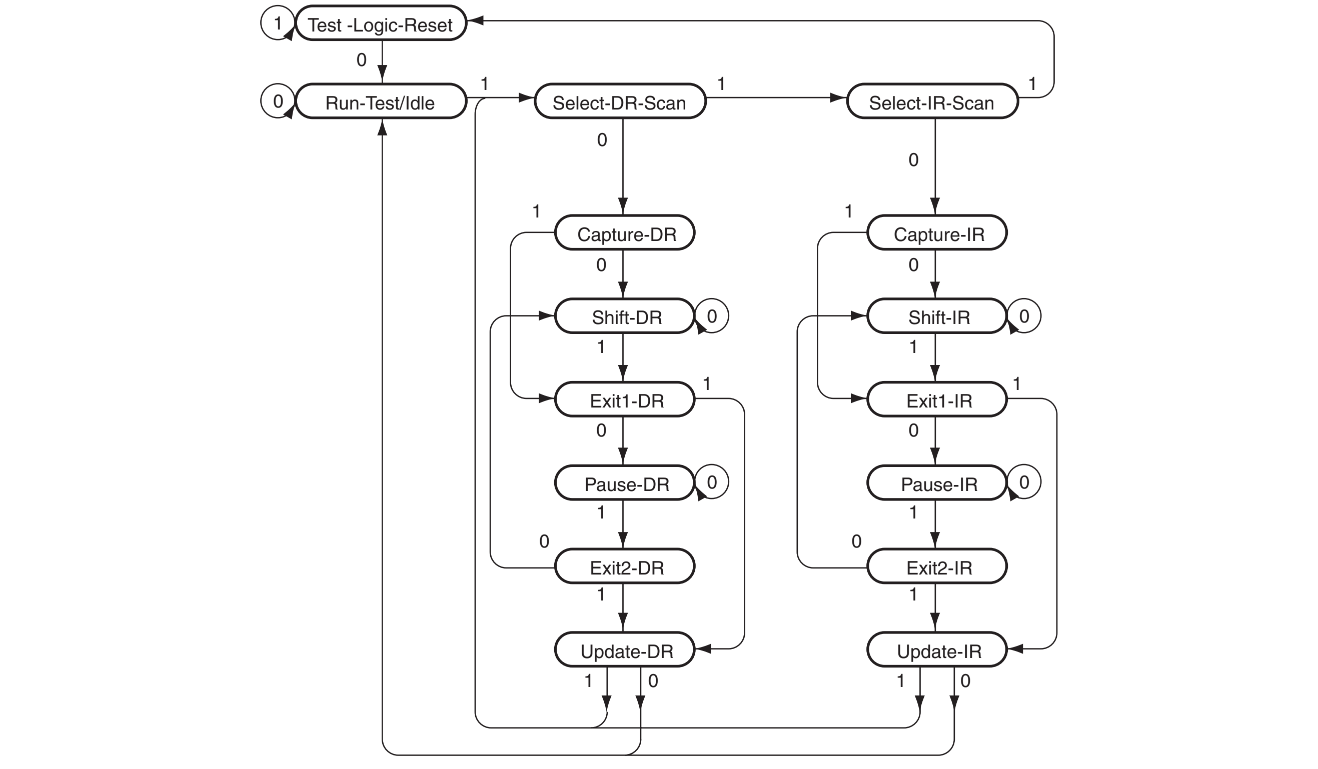


Figure 37.3 TAP Controller State Transitions

37.5.2 H-UDI Reset

A hardware reset is generated by the H-UDI (SDIR) command. After the H-UDI reset assert command has been sent from the H-UDI pin, sending the H-UDI reset negate command resets the CPU (see figure 37.4). The required time between the H-UDI reset assert and H-UDI reset negate commands is the same as the time for holding the reset pin low in order to reset this MCU by a hardware reset.

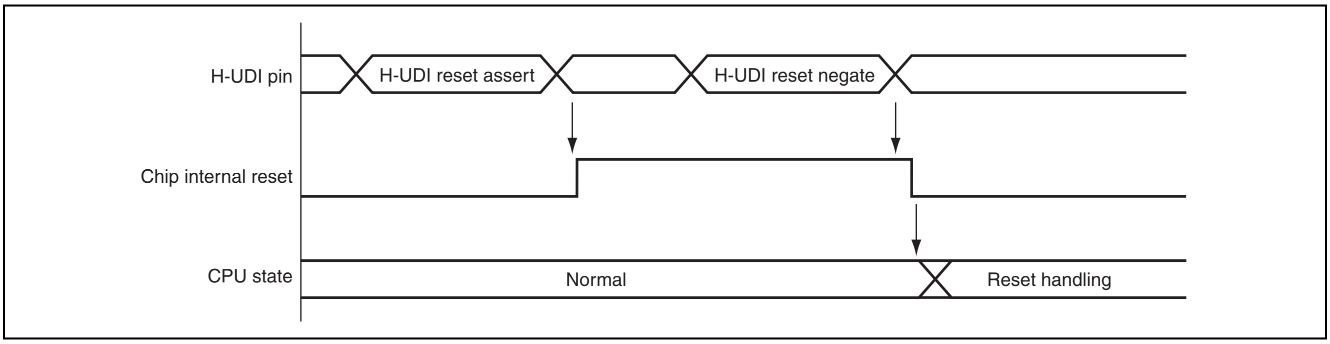


Figure 37.4 H-UDI Reset

37.5.3 H-UDI Interrupt

The H-UDI interrupt function generates an interrupt by setting the appropriate command in the SDIR register from the H-UDI.

An H-UDI interrupt request signal is asserted when the INTREQ bit in the SDINT register is set to "1" by setting the appropriate command. Since the interrupt request is not cleared until the INTREQ bit is cleared to "0" by software, it is not possible to lose the interrupt request. While an H-UDI interrupt command is set in the SDIR register, the SDINT register is connected between the TDI and TDO pins.

37.6 Usage Notes

Once an SDIR command has been set, other than an initialization either by a "L" level input to the TRST# pin or by setting TAP to the Test Logic Reset state, the command will not change unless the H-UDI writes another command.

The H-UDI is used for emulator connection. Therefore, H-UDI functions cannot be used when using an emulator.

This page is blank for reasons of layout.

Section 38 Electrical Characteristics

38.1 Absolute Maximum Ratings

Table 38.1 shows the absolute maximum ratings.

Table 38.1 Absolute Maximum Ratings

Item		Symbol	Rating	Unit	Remarks
Power supply voltage	Vdd	Vdd	–0.3 to +2.0	V	
	Vcc, PLLVcc	Vcc	–0.3 to +6.5	V	
Input voltage	Vcc power supply related pins	Vin	–0.3 to Vcc +0.3	V	
Analog supply voltage		AVcc	–0.3 to +6.5	V	
Analog reference voltage		AVREFH	–0.3 to AVcc +0.3	V	AVREFH > AVREFL
		AVREFL	–0.3 to AVss +0.3	V	
Analog input voltage		VAN	–0.3 to AVcc +0.3	V	
Vss differential voltage		Vss – PLLVss	–0.1 to +0.1	V	
		Vss – AVss	–0.1 to +0.1	V	
		PLLVss – AVss	–0.1 to +0.1	V	
Maximum input current per pin*2 (per pin)	Digital input pins	Imax	–20 to +20	mA	
	Analog input pins	Imax	–20 to +20	mA	
Power dissipation		Pd	1000	mW	Ta = –40°C to +125°C
Operating temperature*1		topr	–40 to +125	°C	
Storage temperature		tstg	–55 to +125	°C	Before assembly

[Usage Notes]

Operating the MCU in excess of the absolute maximum ratings may result in permanent damage. Be sure to use the MCU in compliance with the connection of power pins, combination conditions of applicable power supply voltages, voltage applicable to each pin, and conditions of output voltage, as specified in the manual. Connecting a non-specified power supply or using the MCU at an incorrect voltage may result in permanent damage of the MCU or the system that contains the MCU.

Notes: *1 This does not guarantee that the microcomputer can operate continuously at 85°C-plus. Consult Renesas if the microcomputer is going to be used for 85°C-plus applications.

*2 Ensure that the current input duration does not exceed 10 ms and that the total current input does not exceed 100 mA.

38.2 DC Characteristics

Tables 38.2 to 38.15 show the DC characteristics.

Table 38.2 DC Characteristics - Power Supply Voltage

Symbol	Min.	Typ.	Max.	Unit
Vdd	1.4	1.5	1.65	V
Vcc	3.0	3.3	3.6	V
	4.5	5.0	5.5	V
PLLVcc	3.0	3.3	3.6	V
	4.5	5.0	5.5	V
AVcc	3.0	3.3	3.6	V
	4.5	5.0	5.5	V
AVREFH*[1]	3.0	3.3	3.6	V
	4.5	5.0	5.5	V

Notes: *1 Set to a value that does not exceed AVcc.

- See table 34.2 for combination of power supply voltages.

Table 38.3 DC Characteristics - Input Level Voltage: When 5 V is Used

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V, AVcc = 5.0 V ±0.5 V

					Rating				Measurement Conditions
Item				Symbol	Min.	Typ.	Max.	Unit	
High-level input voltage	Pins with threshold value switching function*1	Vcc power supply pin CMOS input selected	Threshold value selection: 0.35 Vcc	V_{IH}	0.45 Vcc		Vcc	V	
			Threshold value selection: 0.5 Vcc		0.6 Vcc		Vcc	V	
			Threshold value selection: 0.7 Vcc		0.8 Vcc		Vcc	V	
		Vcc power supply pin schmitt input selected	VT+/VT-: 0.5 Vcc/0.35 Vcc		0.6 Vcc		Vcc	V	
			VT+/VT-: 0.7 Vcc/0.35 Vcc		0.8 Vcc		Vcc	V	
			VT+/VT-: 0.7 Vcc/0.5 Vcc		0.8 Vcc		Vcc	V	
	Pins without threshold value switching function	(Vcc power supply pin) PG0 to PG3, PJ1, PJ3 to PJ5, TCK			0.6 Vcc		Vcc	V	
		(AVcc power supply pin) PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5			0.6 AVcc		AVcc	V	
		NMI, FWE, MD0 to MD2, MPMD, DET3OR5, TRST#, TMS, TDI, ASEBRK#/BRKACK, RESET#			0.8 Vcc		Vcc	V	
		EXTAL			0.7 Vcc		Vcc	V	
Low-level input voltage	Pins with threshold value switching function*1	Vcc power supply pin CMOS input selected	Threshold value selection: 0.35 Vcc	V_{IL}	0		0.25 Vcc	V	
			Threshold value selection: 0.5 Vcc		0		0.4 Vcc	V	
			Threshold value selection: 0.7 Vcc		0		0.6 Vcc	V	
		Vcc power supply pin schmitt input selected	VT+/VT-: 0.5 Vcc /0.35 Vcc		0		0.25 Vcc	V	
			VT+/VT-: 0.7 Vcc /0.35 Vcc		0		0.25 Vcc	V	
			VT+/VT-: 0.7 Vcc /0.5 Vcc		0		0.4 Vcc	V	
	Pins without threshold value switching function	(Vcc power supply pin) PG0 to PG3, PJ1, PJ3 to PJ5, TCK			0		0.4 Vcc	V	
		(AVcc power supply pin) PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5			0		0.4 AVcc	V	
		NMI, FWE, MD0 to MD2, MPMD, DET3OR5, TRST#, TMS, TDI, ASEBRK#/BRKACK, RESET#			0		0.25 Vcc	V	
		EXTAL			0		0.125 Vcc	V	

Note: *1 The V_{IH} and the V_{IL} are fixed at 0.7 Vcc and 0.3 Vcc respectively when SDA or SCL is selected by the PF4 or the PF5.

Table 38.4 DC Characteristics - Output Level Voltage: When 5 V is Used

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V, AVcc = 5.0 V ±0.5 V

			Rating				
Item		**Symbol**	**Min.**	**Typ.**	**Max.**	**Unit**	**Measurement Conditions**
Output high-level voltage (normal output and driving ability "increased")*[1]	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10,PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9	V_{OH}	Vcc –0.5	—	—	V	I_{OH} = 200 µA
			Vcc –1.0	—	—	V	I_{OH} = 1 mA
	WDTOVF#, ASEBRK#/BRKACK		Vcc –0.5	—	—	V	I_{OH} = 200 µA
			Vcc –1.0	—	—	V	I_{OH} = 1 mA
	TDO		Vcc –0.5	—	—	V	I_{OH} = 200 µA
			Vcc –1.0	—	—	V	I_{OH} =1 mA
Output low-level voltage (normal output and driving ability "increased")*[1]	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10,PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA
			—	—	1.2	V	I_{OL} = 4 mA
	WDTOVF#, ASEBRK#/BRKACK		—	—	0.4	V	I_{OL} = 1.6 mA
			—	—	1.2	V	I_{OL} = 4 mA
	TDO		—	—	0.4	V	I_{OL} = 1.6 mA
			—	—	1.2	V	I_{OL} = 4 mA

Note: *1 When the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR) is set to "0" or "1".

Table 38.5 DC Characteristics - Input Level Voltage: When 3.3 V is Used

Recommended Operating Conditions: Vcc = PLLVcc = 3.3 V ±0.3 V, AVcc = 3.3 V ±0.3 V

Item				Symbol	Rating			Unit	Measurement Conditions
					Min.	Typ.	Max.		
High-level input voltage	Pins with threshold value switching function*1	Vcc power supply pin CMOS input selected	Threshold value selection: 0.35 Vcc	V_{IH}	0.5 Vcc		Vcc	V	
			Threshold value selection: 0.5 Vcc		0.65 Vcc		Vcc	V	
			Threshold value selection: 0.7 Vcc		0.8 Vcc		Vcc	V	
		Vcc power supply pin schmitt input selected	VT+/VT-: 0.5 Vcc /0.35 Vcc		0.65 Vcc		Vcc	V	
			VT+/VT-: 0.7 Vcc /0.35 Vcc		0.8 Vcc		Vcc	V	
			VT+/VT-: 0.7 Vcc /0.5 Vcc		0.8 Vcc		Vcc	V	
	Pins without threshold value switching function	(Vcc power supply pin) PG0 to PG3, PJ1, PJ3 to PJ5, TCK			0.65 Vcc		Vcc	V	
		(AVcc power supply pin) PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5			0.65 AVcc		AVcc	V	
		NMI, FWE, MD0 to MD2, MPMD, DET3OR5, TRST#, TMS, TDI, ASEBRK#/BRKACK, RESET#			0.8 Vcc		Vcc	V	
		EXTAL			0.7 Vcc		Vcc	V	
Low-level input voltage	Pins with threshold value switching function*1	Vcc power supply pin CMOS input selected	Threshold value selection: 0.35 Vcc	V_{IL}	0		0.2 Vcc	V	
			Threshold value selection: 0.5 Vcc		0		0.35 Vcc	V	
			Threshold value selection: 0.7 Vcc		0		0.5 Vcc	V	
		Vcc power supply pin schmitt input selected	VT+/VT-: 0.5 Vcc /0.35 Vcc		0		0.2 Vcc	V	
			VT+/VT-: 0.7 Vcc /0.35 Vcc		0		0.2 Vcc	V	
			VT+/VT-: 0.7 Vcc /0.5 Vcc		0		0.35 Vcc	V	
	Pins without threshold value switching function	(Vcc power supply pin) PG0 to PG3, PJ1, PJ3 to PJ5, TCK			0		0.35 Vcc	V	
		(AVcc power supply pin) PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5			0		0.35 AVcc	V	
		NMI, FWE, MD0 to MD2, MPMD, DET3OR5, TRST#, TMS, TDI, ASEBRK#/BRKACK, RESET#			0		0.2 Vcc	V	
		EXTAL			0		0.2 Vcc	V	

Note: *1 The V_{IH} and the V_{IL} are fixed at 0.7 Vcc and 0.3 Vcc respectively when SDA or SCL is selected by the PF4 or the PF5.

Table 38.6 DC Characteristics - Output Level Voltage: When 3.3 V is Used with Driving Ability Set to "Increased"

Recommended Operating Conditions: Vcc = PLLVcc = 3.3 V ±0.3 V, AVcc = 3.3 V ±0.3 V

Item		Symbol	Rating			Unit	Measurement Conditions
			Min.	Typ.	Max.		
Output high-level voltage (normal output and driving ability)*[1]	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9	V_{OH}	Vcc –1.1	—	—	V	I_{OH}= 200 µA
Output low-level voltage (normal output and driving ability)*[1]	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9	V_{OL}	—	—	0.9	V	I_{OL}= 1.6 mA

Note: *1 When the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR) is set to "1".

Table 38.7 DC Characteristics - Output Voltage: When 3.3 V is Used with Driving Ability Set to "Normal Output"

Recommended Operating Conditions: Vcc = PLLVcc = 3.3 V ±0.3 V, AVcc = 3.3 V ±0.3 V

Item		Symbol	Rating			Unit	Measurement Conditions
			Min.	Typ.	Max.		
Output high-level voltage (normal output)*[1]	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9	V_{OH}	Vcc –1.1	—	—	V	I_{OH}= 200 µA
	WDTOVF#, ASEBRK#/BRKACK		Vcc –1.1	—	—	V	I_{OH}= 200 µA
	TDO		Vcc –0.5	—	—	V	I_{OH} = 200 µA
Output low-level voltage (normal output)*[1]	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9	V_{OL}	—	—	0.9	V	I_{OL}= 1.6 mA
	WDTOVF#, ASEBRK#/BRKACK		—	—	0.9	V	I_{OL} = 1.6 mA
	TDO		—	—	0.4	V	I_{OL} = 1.6 mA

Note: *1 When the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR) is set to "0".

Table 38.8 DC Characteristics - Input Leak Current

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V, AVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input leak current	RESET#, NMI, EXTAL, TRST#, TCK, TMS, TDI, ASEBRK#/BRKACK, PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10,PE15, PF0, PF1, PF4, PF5, PG0 to PG4, PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9	| Iin |	—	—	2.0	µA	Vin = 0.3V to Vcc –0.3V
	PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5 (Function 1: When a port is selected)		—	—	2.0	µA	Vin = 0.3V to AVcc –0.3V
	PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5 (Function 2: When analog input is selected)		—	—	2.0	µA	Vin = 0.3V to AVcc –0.3V

Table 38.9 DC Characteristics - Pull-up/Pull-down MOS Current- Input Voltage: When 5 V is Used

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input pull-up MOS current	MPMD, DET3OR5	-Ipu	50	—	300	µA	Vin = 0 V
Input pull-down MOS current	MD0 to MD2, FWE	Ipd	50	—	300	µA	Vin = Vcc

Note: • Only the pins listed in the table above have pull-up/pull-down functions.

Table 38.10 DC Characteristics - Pull-up/Pull-down MOS Current- Input Voltage: When 3.3 V is Used

Recommended Operating Conditions: Vcc = PLLVcc = 3.3 V ±0.3 V

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input pull-up MOS current	MPMD, DET3OR5	-Ipu	10	—	150	µA	Vin = 0 V
Input pull-down MOS current	MD0 to MD2, FWE	Ipd	10	—	150	µA	Vin = Vcc

Note: • Only the pins listed in the table above have pull-up/pull-down functions.

Table 38.11 DC Characteristics - Permissible Output Current Values

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V, AVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V

Item		Symbol	Min.	Typ.	Max.	Unit
Output low-level permissible current (per pin)		I_{OL}	—	—	4.0	mA
Output low-level permissible current	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4	ΣI_{OL}	—	—	52.4	mA
	PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9		—	—	27.6	
Output high-level permissible current (per pin)		I_{OH}	—	—	2.0	mA
Output high-level permissible current	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4	ΣI_{OH}	—	—	16.4	mA
	PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9		—	—	8.6	

Table 38.12 DC Characteristics - Injection Current Values

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V, AVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V

Item		Symbol	Min.	Typ.	Max.	Unit
DC Injection current (per pin)	logic pin	I_{IC}	–1.0	—	1.0	mA
	Analog pin		–0.1	—	0.1	mA
DC Injection current	PA0 to PA13, PB0, PB1, PB3, PC0 to PC3, PC5, PC6, PC14, PD0 to PD10, PE15, PF0, PF1, PF4, PF5, PG0 to PG4	$\Sigma\|I_{IC}\|$	—	—	25.6	mA
	PH0 to PH15, PJ0 to PJ7, PJ10 to PJ15, PK0, PK5, PK6, PK8 to PK14, PL2 to PL6, PL8, PL9		—	—	17.2	
	PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, PN5		—	—	7.2	

Table 38.13 DC Characteristics - Input Capacitance

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V, AVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input capacitance	Cin	—	10	20	pF	Vin = 0 V, f = 1 MHz, Ta = 25°C

Note: • For details on an analog input capacitance, see table 38.36.

Table 38.14 DC Characteristics - Supply Current

Recommended Operating Conditions: Vcc = PLLVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V, AVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Core supply current (Vdd power supply)		I_{DD}	—	—	480	mA	Ick = 160 MHz
System consumption current (Vcc power supply)*[1] (Including flash memory programming and erasure)		I_{CC}	—	—	90	mA	Pck = 40 MHz
PLL supply current (PLLVcc power supply)		I_{PLL}	—	—	10	mA	
Analog supply current (AVcc power supply)	During A/D conversion	I_{AVcc}	—	—	10	mA	2 modules, Pck = 40MHz
	Awaiting A/D conversion		—	—	1	mA	
ADC reference power supply current (AVREF)	During A/D conversion	I_{AVREF}	—	—	4	mA	2 modules, Pck = 40MHz
	Awaiting A/D conversion		—	—	3.5	mA	

Notes: *1 An inrush current of about 100 mA will be caused at power on.

- When the A/D converter is not used, do not leave the AVcc, AVref, and AVss pins open.
- The supply current is measured when V_{IH}min = Vcc – 0.5 V, V_{IL} = 0.5 V, with all output pins unloaded.

Table 38.15 DC Characteristics - Output Load Capacitance

Item	Symbol	Min.	Max.	Unit
Output Load Capacitance	CL	15	50	pF

38.3 AC Characteristics

- The timing conditions without specifications are the following:

  Vdd = 1.5 V + 0.15 V, -0.1 V, Vcc = PLLVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V, AVcc = 5.0 V ±0.5 V/3.3 V ±0.3 V, AVREFH = 4.5 V to AVcc/3.0 V to AVcc,
  Vss = PLLVss = AVss = AVREFL = 0 V, Ta = -40°C to +125°C

  When not otherwise specified, the input threshold value is the value under conditions where all module input pins for the same channel are set to the same characteristics. When not otherwise specified, the output driving ability is the value under conditions where all module output pins for the same channel are set to the same characteristics.
- Standard values are guaranteed when the output load capacity of the measurement pin is 15 pF to 50 pF. Note that the output load capacity of the CLKOUT pin is 15pF to 30pF.

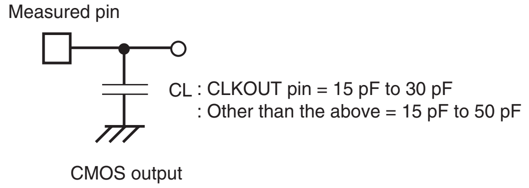

Measured pin
CL : CLKOUT pin = 15 pF to 30 pF
: Other than the above = 15 pF to 50 pF
CMOS output

Figure 38.1 Measurement Circuit for Output Switching Characteristics

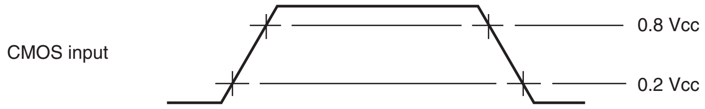
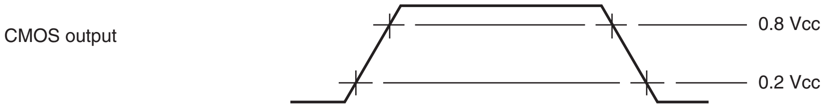


Figure 38.2 Input Waveform and Timing Check Points at Characteristics Measurement



Note: • For details on CLKOUT output timing check points at characteristics measurement, refer to each figure in this section.

Figure 38.3 Output Timing Check Points at Characteristics Measurement

38.3.1 Power-On/Off Timing

Table 38.16 shows the power-on/off timing.

Table 38.16 Power-On/Off Timing

Item	Symbol	Min.	Max.	Unit	Figures
Vdd power-on time	t_{VDDS}	10	—	μs	38.4
Vcc holding time at Vdd shutdown	t_{VDDH}	—	0	μs	
Vcc voltage at power off	VCCL	0	1.0	V	
Vdd voltage at power on	VDDL	0	0.5	V	

Notes: • AVcc ≥ Vcc must be satisfied. If not, an electric current may flow.

• AVREFH ≤ AVcc + 0.3 must always be satisfied even at power-on/off.

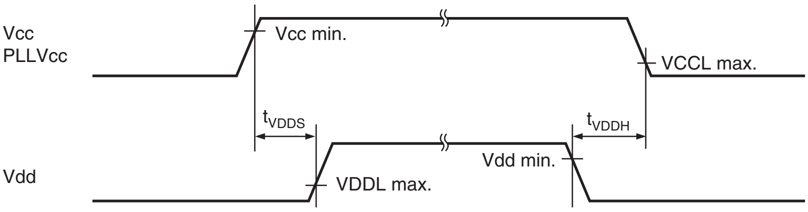


Figure 38.4 Power-On/Off Timing

38.3.2 Operation Mode and Oscillation Timing

Table 38.17 shows the operation mode and oscillation timing.

Table 38.17 Operation Mode and Oscillation Timing

Item	Symbol	Min.	Max.	Unit	Figures
Oscillation settling time*[1]	t_{OSC1}	10	—	ms	38.5
Operation mode set up time when start-up	t_{MDS1}	10	—	ms	
Operation mode set up time during operation	t_{MDS2}	10	—	µs	
Operation mode hold time after reset is inactive	t_{MDH1}	30	—	µs	
Operation mode hold time when power-off	t_{MDH2}	0	—	ms	
Vcc hold time after reset*[2]	$t_{RES\text{-}VccH}$	1	—	ms	
Vdd hold time after reset*[2]	$t_{RES\text{-}VDDH}$	1	—	ms	
RESET# pulse width	t_{RESW}	100	—	µs	38.6

Notes: *1 The oscillation settling time (tosc1) is specified as the value that includes the PLL oscillator settling time only, not including the settling time of the oscillator circuit. Consult with the resonator or oscillator manufacturer to decide the settling time of the oscillator circuit including an external resonator or oscillator. Apply RESET# during the total period of the oscillator circuit settling time and PLL oscillator settling time when power-on.

*2 $t_{RES\text{-}VCCH}$ and $t_{RES\text{-}VDDH}$ = 1 ms (min.) is specified for internal flash memory programming/erasing. Otherwise, $t_{RES\text{-}VCCH}$ and $t_{RES\text{-}VDDH}$ = 0 ms (min.).

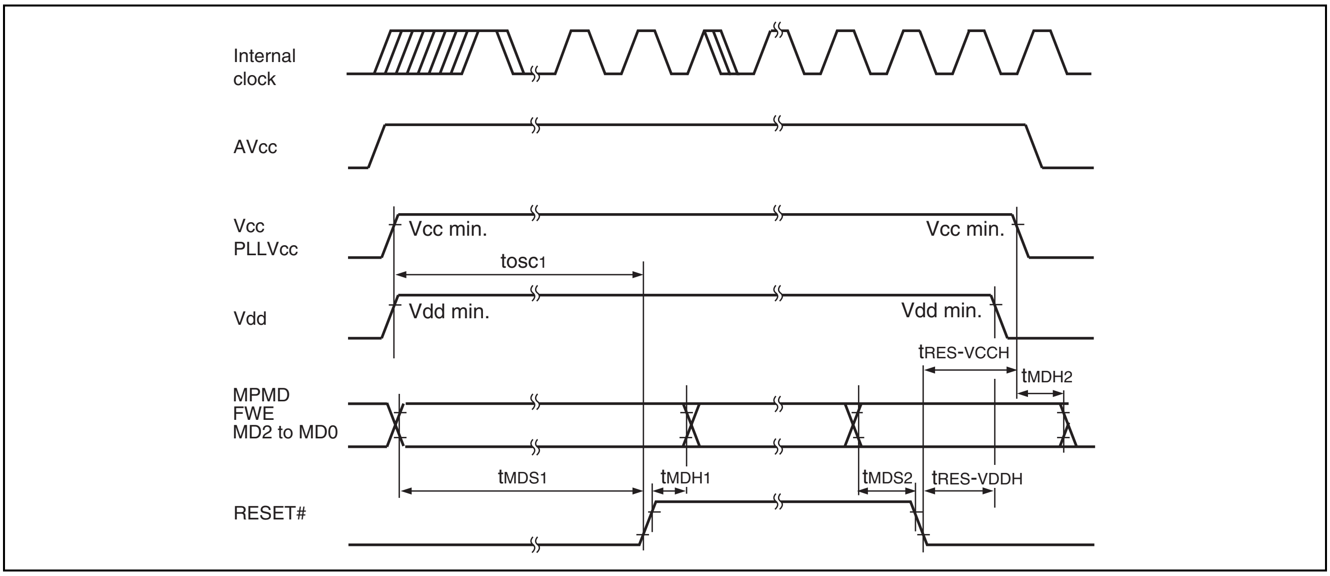


Figure 38.5 Operation Mode and Oscillation Timing at Power-On/Off

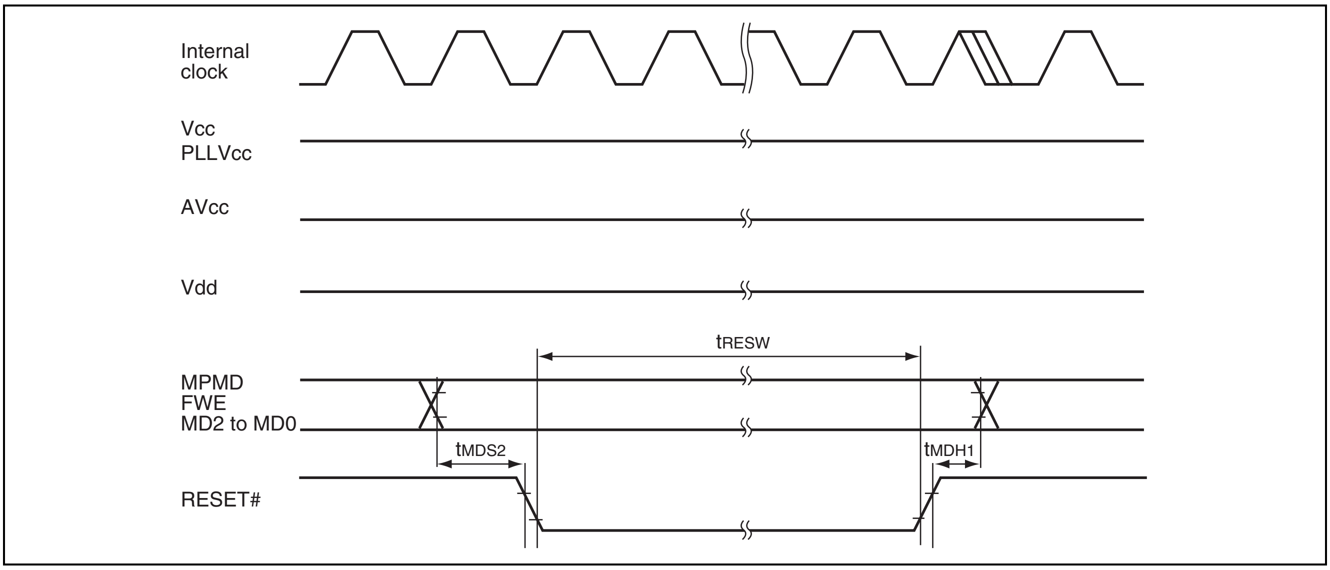


Figure 38.6 Operation Mode and Oscillation Timing during Operation

38.3.3 Clock Timing

Table 38.18 shows the clock timing.

Table 38.18 Clock Timing

Item	Symbol	Min.	Max.	Unit	Figures
EXTAL clock input frequency	f_{EX}	16	20	MHz	38.7
EXTAL clock input cycle time	t_{EXcyc}	50	62.5	ns	
EXTAL clock input low-level pulse width	t_{EXL}	15	—	ns	
EXTAL clock input high-level pulse width	t_{EXH}	15	—	ns	
EXTAL clock input rise time	t_{EXR}	—	4	ns	
EXTAL clock input fall time	t_{EXF}	—	4	ns	
Clock frequency*[1]	f_{op}	—	40	MHz	38.8
Clock cycle time	t_{cyc}	25	—	ns	
Clock low-level pulse width	t_{CL}	4	—	ns	
Clock high-level pulse width	t_{CH}	4	—	ns	

Note: *1 The CLKOUT pin outputs the peripheral clock signal (Pck).

[Usage Notes]

The EXTAL, XTAL, and CLKOUT pins constitute a circuit requiring a power supply voltage of Vcc = 3.3 V ±0.3 V or 5.0 V ±0.5 V. Comply with the input and output voltages specified in the DC characteristics.

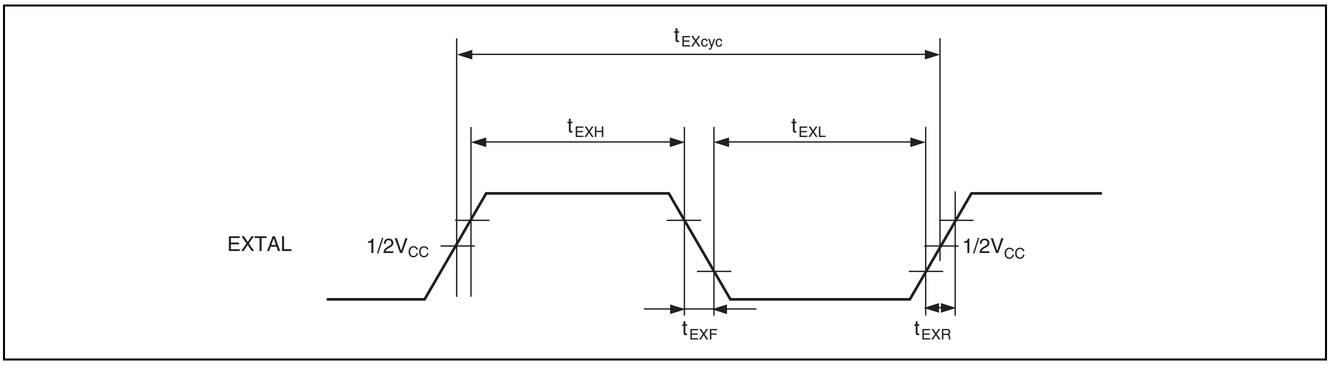


Figure 38.7 EXTAL Clock Input Timing

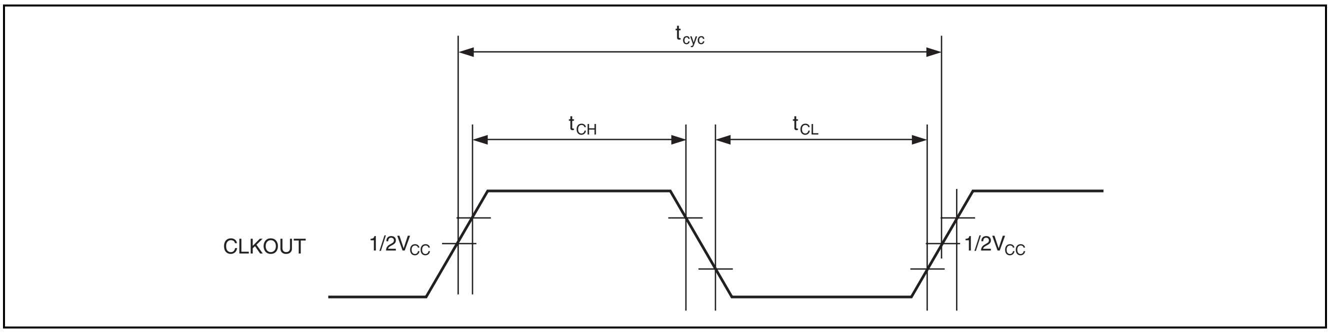


Figure 38.8 System Clock Timing

38.3.4 Control Signal Timing

Table 38.19 shows the control signal timing.

Table 38.19 Control Signal Timing

Item	Symbol	Min.	Max.	Unit	Figures
RESET# pulse width	t_{RESW}	100	—	μs	38.9
RESET# noise cancel width	t_{RESNCW}	25	300	ns	
NMI pulse width	t_{NMIW}	300 +6 tc(Pck)	—	ns	38.10
IRQn setup time (edge detection)	t_{IRQS}	23 +tc(Pck)	—	ns	38.11
IRQn hold time (edge detection)	t_{IRQH}	23	—	ns	
IRQn pulse width (level detection)	t_{IRQW}	6 tc(Pck)	—	ns	

Note: • tc(Pck) indicates the cycle of the peripheral clock (Pck).

[Usage Notes]

The RESET#, NMI, and IRQn signals are asynchronous inputs, but when the setup times shown here are provided, the signals are considered to have been changed at clock rise. If the setup times are not provided, recognition is delayed until the next clock rise.

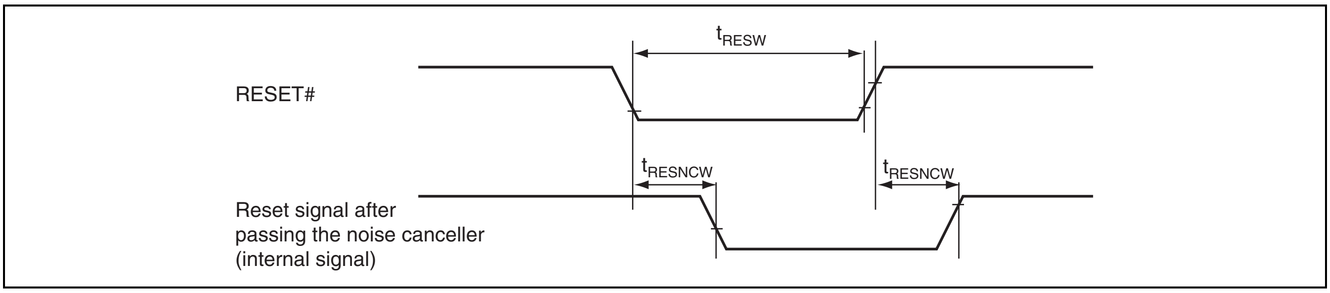


Figure 38.9 Reset Input Timing

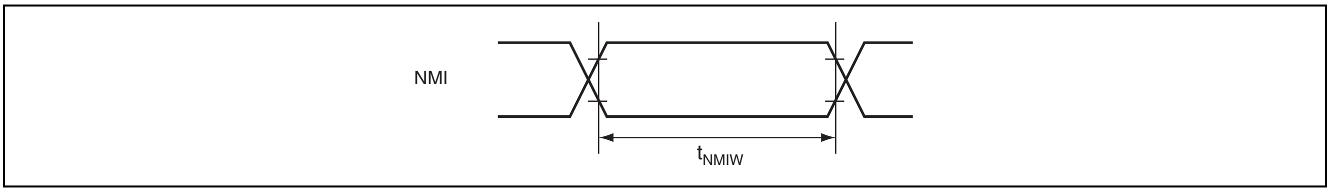


Figure 38.10 NMI Timing

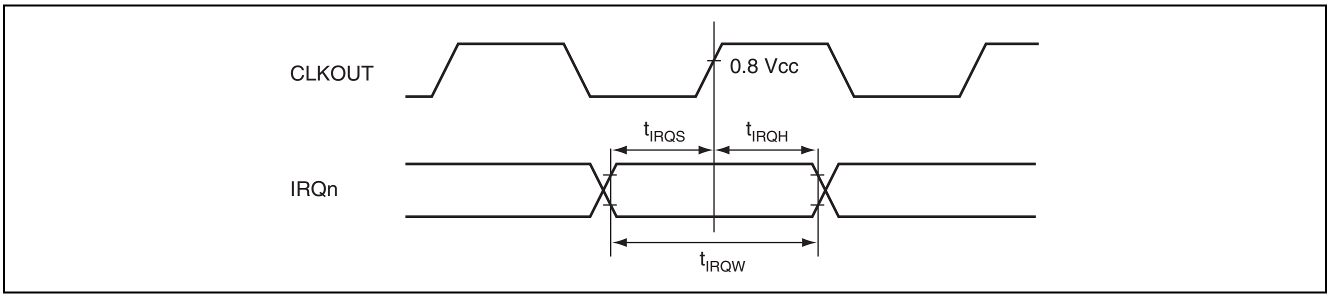


Figure 38.11 Interrupt Signal Input Timing

38.3.5 DMAC Timing

Table 38.20 shows the DMAC timing.

Table 38.20 DMAC Timing

Condition: All values are for setting conditions that the driving ability is increased.*[1]

Item	Symbol	Min.	Max.	Unit	Figures
DREQn setup time	t_{DRQS}	20	—	ns	38.12
DREQn hold time	t_{DRQH}	20	—	ns	

Note: *1 To increase a driving ability, set "1" to the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR).

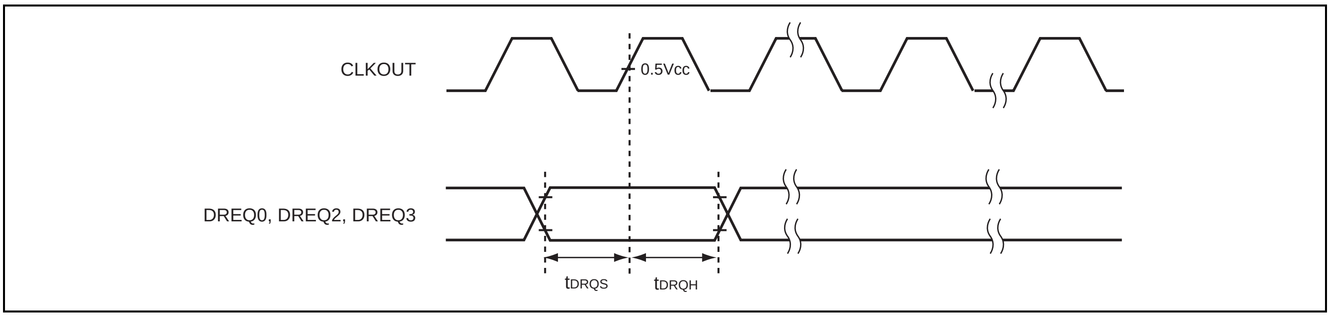


Figure 38.12 DMAC Timing

38.3.6 ATU-IIIS Module Timing

Table 38.21 shows the ATU-IIIS module timing.

Table 38.21 ATU-IIIS Module Timing

Item	Symbol	Min.	Max.	Unit	Figures
Timer output delay time	t_{TOD}	—	100	ns	38.13
Timer input setup time	t_{TIS}	100	—	ns	
Timer clock input setup time	t_{TCKS}	100	—	ns	38.14
Timer clock pulse width (single edge specified)	t_{TCKWH}, t_{TCKWL}	1.5 tc(Pck)	—	ns	
Timer clock pulse width (both edges specified)	t_{TCKWH}, t_{TCKWL}	2.5 tc(Pck)	—	ns	

Note: • tc(Pck) indicates the cycle of the peripheral clock (Pck).

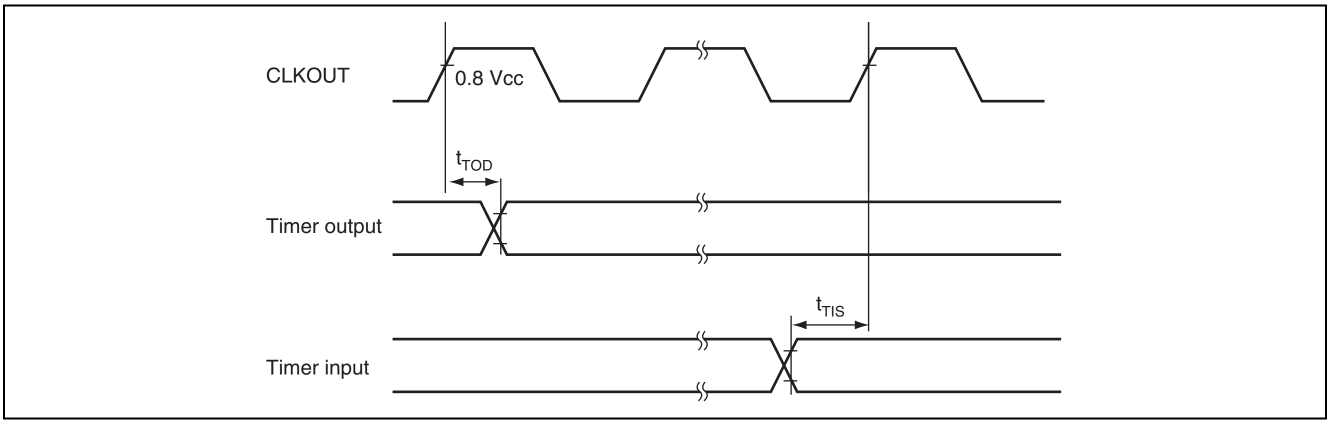


Figure 38.13 ATU-IIIS Input/Output Timing

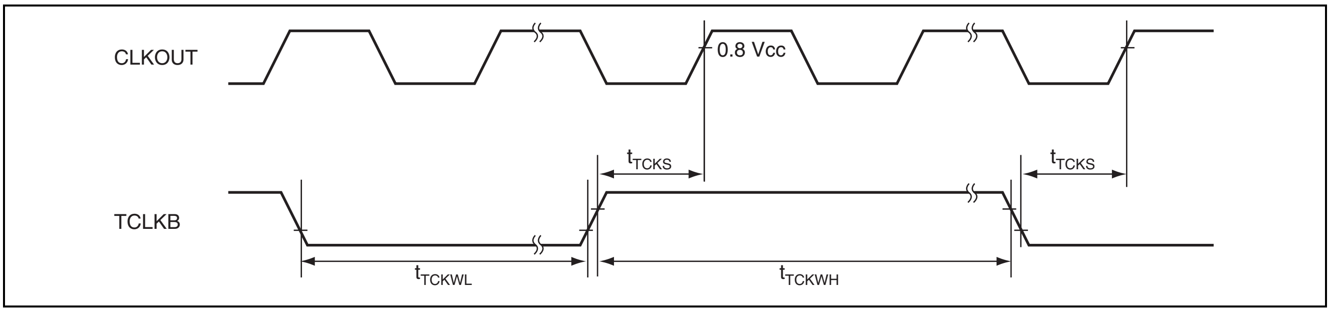


Figure 38.14 ATU-IIIS Clock Input Timing

38.3.7 I/O Port Timing

Table 38.22 shows the I/O port timing.

Table 38.22 I/O Port Timing

Item	Symbol	Min.	Max.	Unit	Figures
Port input setup time	tsu (P-E)	100	—	ns	38.15
Port input hold time	th (E-P)	0	—	ns	
Port output data delay time	td (E-P)	—	100	ns	

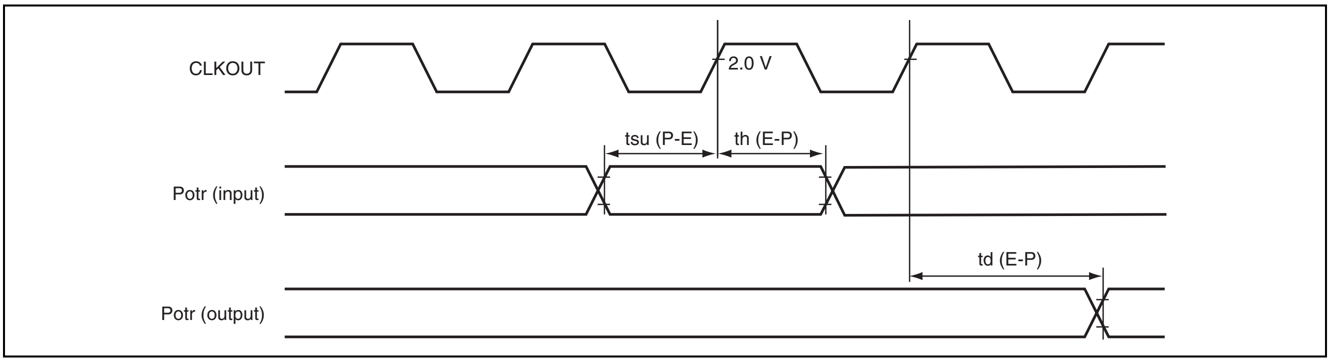


Figure 38.15 I/O Port Input/Output Timing

38.3.8 WDT Timing

Table 38.23 shows the WDT timing.

Table 38.23 WDT Timing

Item	Symbol	Min.	Max.	Unit	Figures
WDTOVF# delay time	t_{WOVD}	—	100	ns	38.16

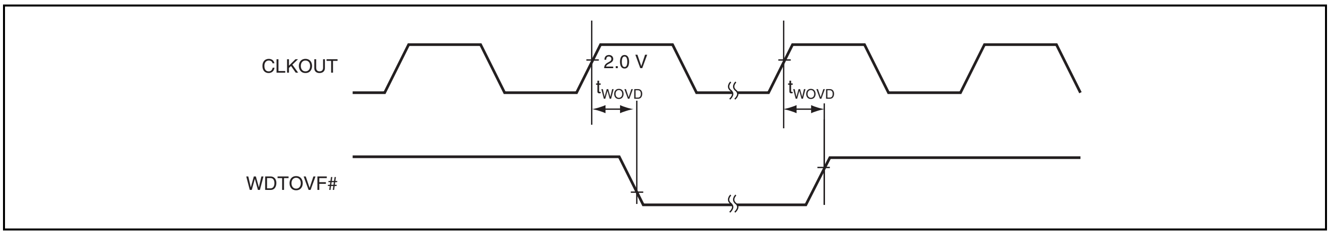


Figure 38.16 WDT Timing

38.3.9 SCIF Interface Timing

Table 38.24 shows the SCIF interface timing.

Table 38.24 SCIF Interface Timing

Condition: All values are for setting conditions that the driving ability is increased.*[1]

Item		Symbol	Min.	Max.	Unit	Figures
Clock cycle	Clock sync	t_{SCYC}	12 tc(Pck)	—	ns	38.17, 38.18
	Asynchronous		4 tc(Pck)	—	ns	
Input clock rise time		t_{SCKr}	—	1.5 tc(Pck)	ns	
Input clock fall time		t_{SCKf}	—	1.5 tc(Pck)	ns	
Input clock pulse width		t_{SCKW}	0.4 t_{scyc}	0.6 t_{scyc}	ns	
Transmit data delay time (clock sync)	Internal clock selected	t_{TXD}	–50	4 tc(Pck) +45	ns	
	External clock selected		0	4 tc(Pck) +45	ns	
Receive data setup time (clock sync)		t_{RXS}	4 tc(Pck) +70	—	ns	
Receive data hold time (clock sync)		t_{RXH}	tc(Pck) +15	—	ns	

Notes: *1 To increase a driving ability, set "1" to the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR).

- tc(Pck) indicates the cycle of the peripheral clock (Pck).

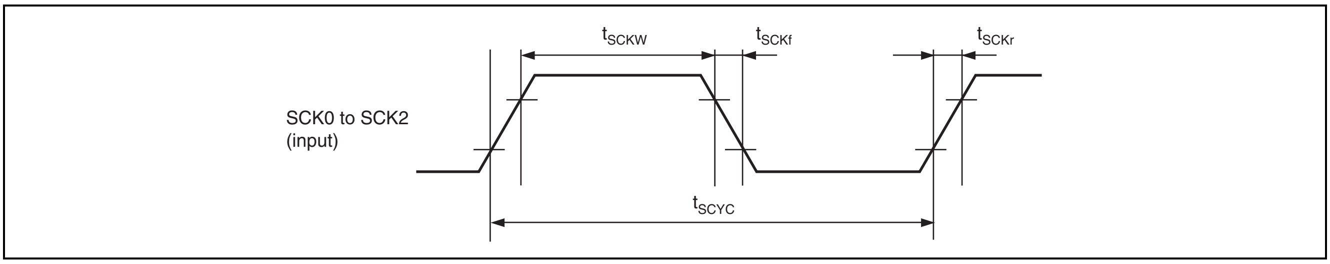


Figure 38.17 SCK Input Timing

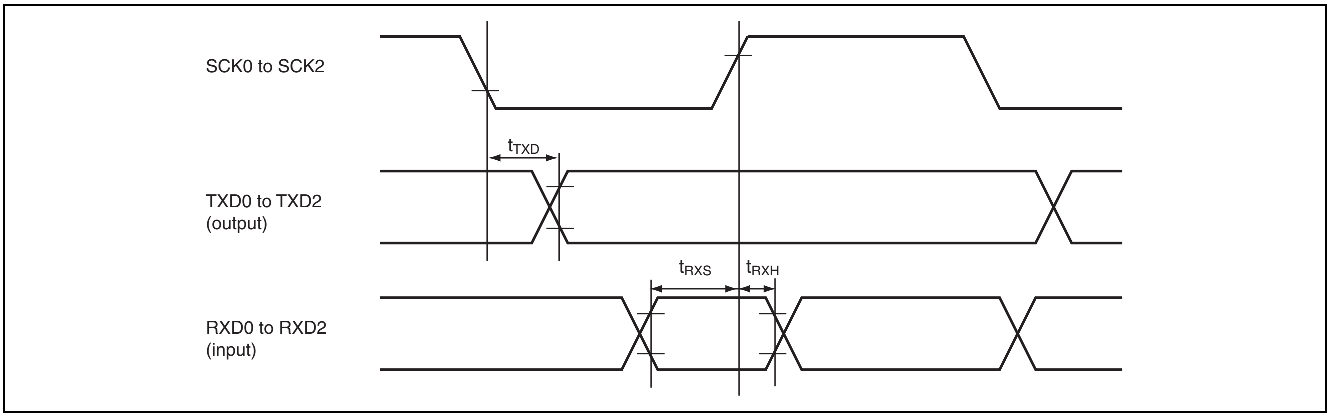


Figure 38.18 SCI Input/Output Timing

38.3.10 RSPI Timing

Table 38.25 shows the RSPI timing.

Table 38.25 RSPI Timing

Condition: All values are for setting conditions that the driving ability is increased.*[1]

Item		Symbol	Min.	Max.	Unit	Figures
RSPCK clock cycle	Master	t_{SPcyc}	4	4096	t_{cyc}	38.19
	Slave		16	4096		
RSPCK clock pulse width	Master	t_{SPCKW}	$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF})$ /2 - 5	—	ns	
	Slave		$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF})$ /2	—		
RSPCK clock input rise/fall time	Master	t_{SPCKR}, t_{SPCKF}	—	8	ns	
	Slave		—	1	µs	
Data input setup time	Master	t_{SU}	25	—	ns	38.20 to 38.23
	Slave		$25 + 2 \times t_{cyc}$	—		
Data input hold time	Master	t_{H}	0	—	ns	
	Slave		$20 + 2 \times t_{cyc}$	—		
SSL setup time	Master	t_{LEAD}	1	8	t_{SPcyc}	
	Slave		4	—	t_{cyc}	
SSL hold time	Master	t_{LAG}	1	8	t_{SPcyc}	
	Slave		4	—	t_{cyc}	
Data output delay time	Master	t_{OD}	—	20	ns	
	Slave		—	$3 \times t_{cyc} + 32$		
Data output hold time	Master	t_{OH}	–10	—	ns	
	Slave		0	—		
Continuous transmission delay time	Master	t_{TD}	$t_{SPcyc} + 2 \times t_{cyc}$	$8 \times t_{SPcyc} + 2 \times t_{cyc}$	ns	
	Slave		$4 \times t_{cyc}$	—		
MOSI, MISO rise/fall time	Output	t_{DR}, t_{DF}	—	8	ns	
	Input		—	1	µs	
SSL rise/fall time	Output	t_{SSLR}, t_{SSLF}	—	8	ns	
	Input		—	1	µs	
Slave access time		t_{SA}	—	4	t_{cyc}	38.22, 38.23
Slave out release time		t_{REL}	—	3	t_{cyc}	

Note: *1 To increase a driving ability, set "1" to the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR).

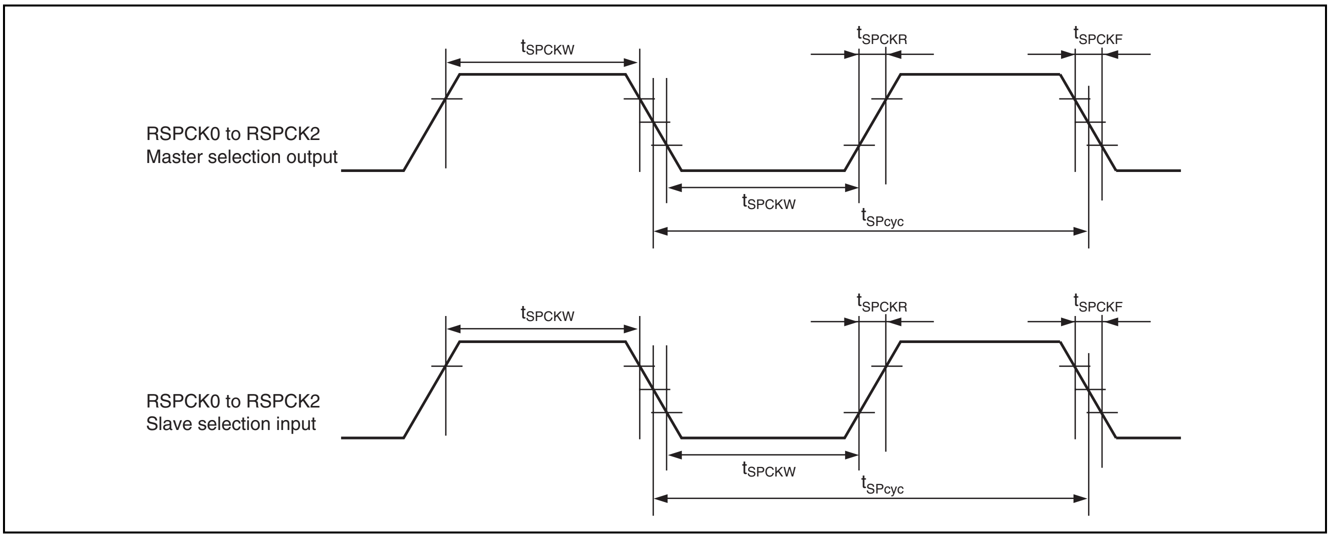


Figure 38.19 RSPI Clock Timing

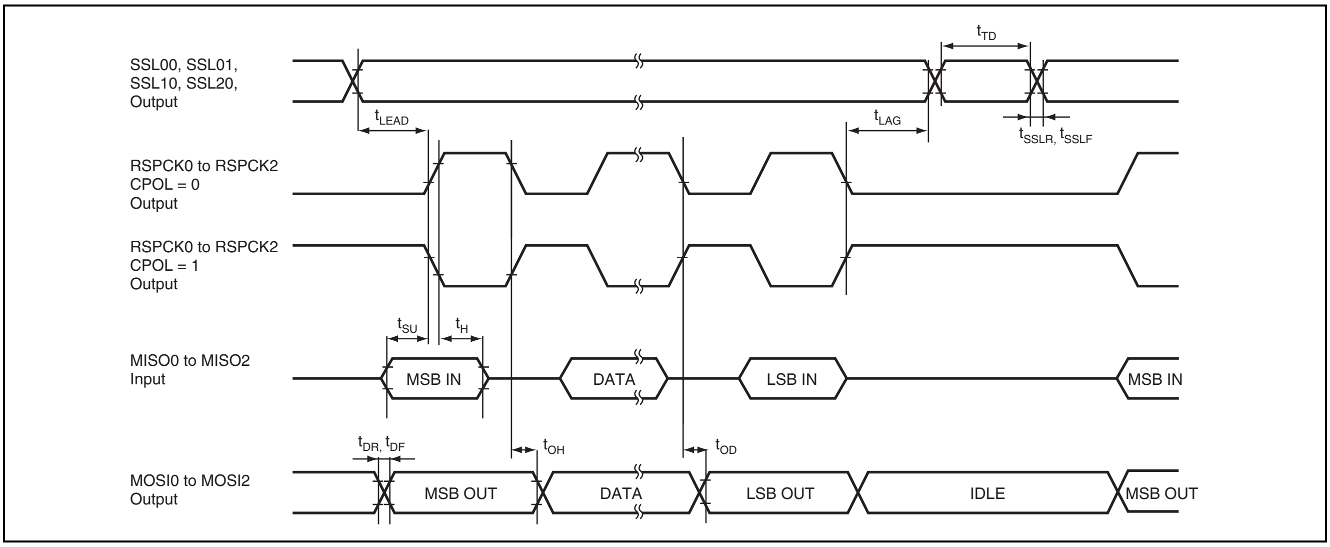


Figure 38.20 RSPI Timing (Master, CPHA = "0")

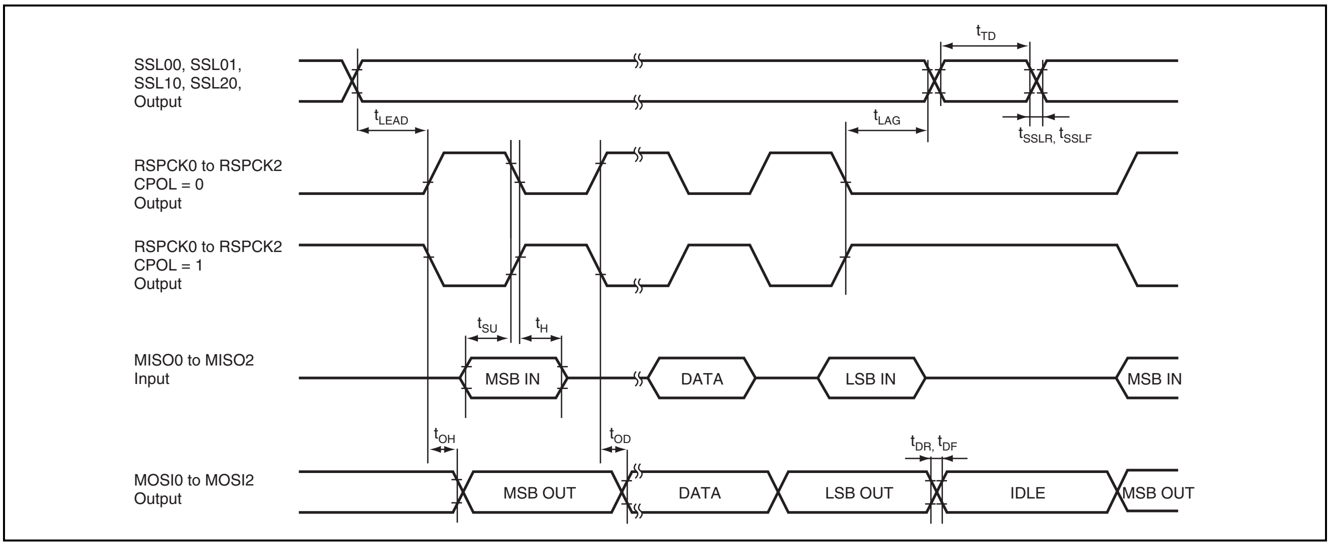


Figure 38.21 RSPI Timing (Master, CPHA = "1")

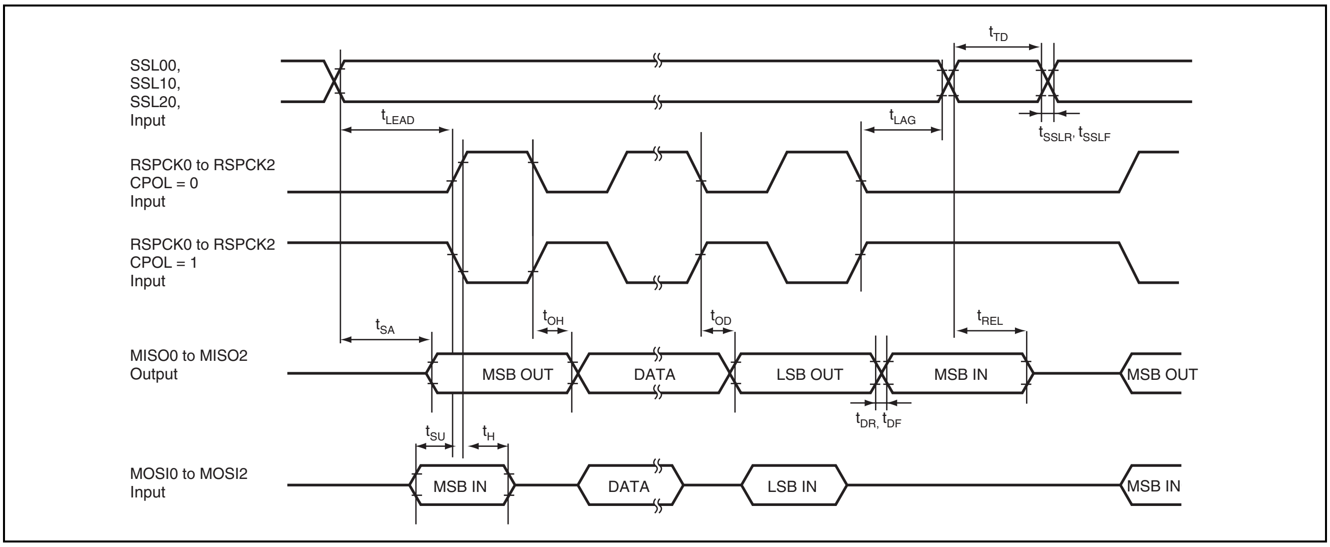


Figure 38.22 RSPI Timing (Slave, CPHA = "0")

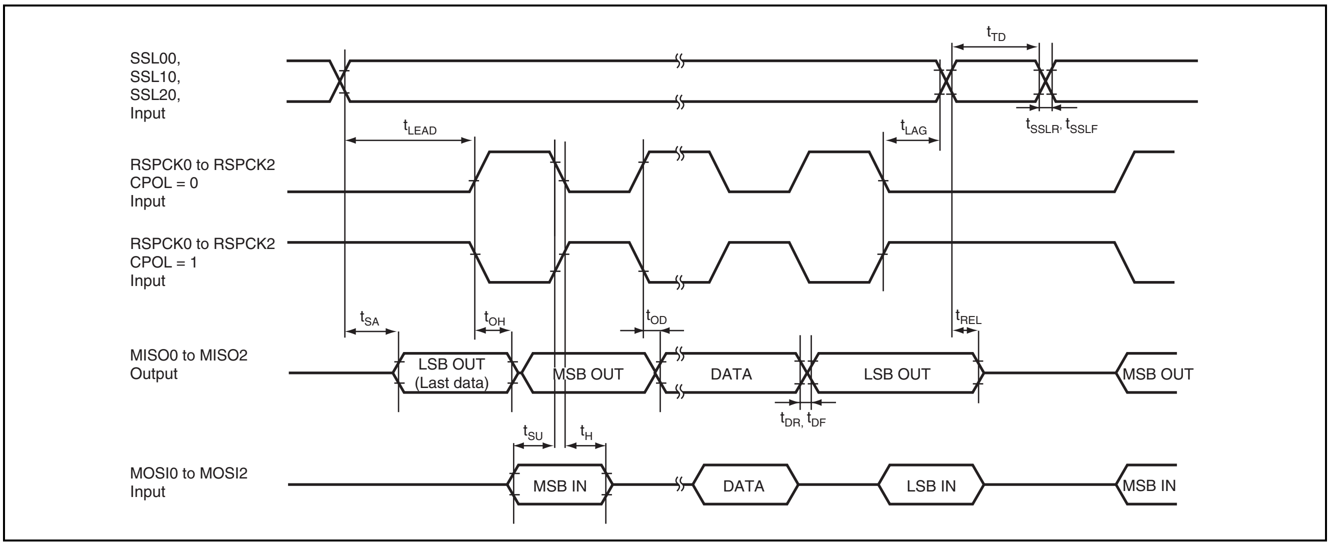


Figure 38.23 RSPI Timing (Slave, CPHA = "1")

38.3.11 IIC3 Timing

Table 38.26 shows the IIC3 timing.

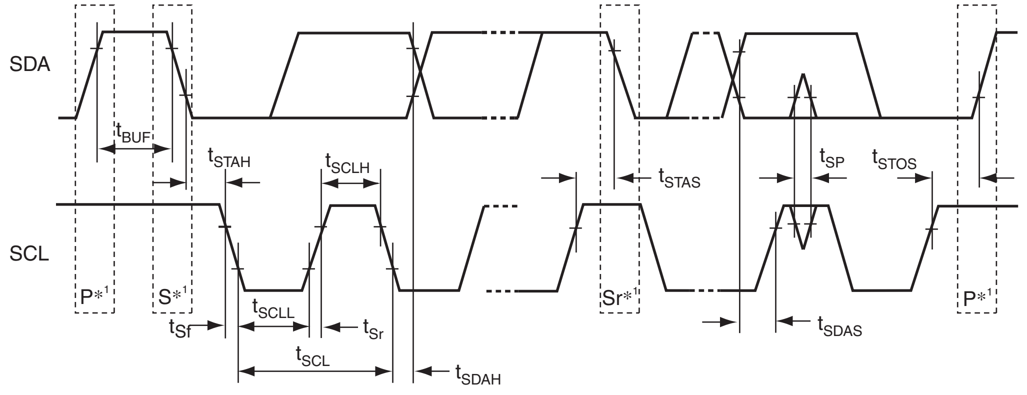
Table 38.26 IIC3 Timing

Input waveform timing check points "H" level (V_{IH}) : 0.7 Vcc, "L" level (V_{IL}) : 0.3 Vcc
Output reference level "H" level (V_{OH}) : 0.7 Vcc, "L" level (V_{OL}) : 0.3 Vcc.

Item		Symbol	Min.	Max.	Unit	Figures
SCL input cycle time		t_{SCL}	12 tc(Pck) + 600	—	ns	38.24
SCL input high-level pulse width		t_{SCLH}	3 tc(Pck) + 300	—	ns	
SCL input low-level pulse width		t_{SCLL}	5 tc(Pck) + 300	—	ns	
SCL, SDA input rise time		t_{Sr}	—	300	ns	
SCL, SDA input fall time		t_{Sf}	—	300	ns	
SCL and SDA input spike pulse elimination time	ICNF2CYC.NF2CYC = "0"	t_{SP}	—	tc(Pck)	ns	
	ICNF2CYC.NF2CYC = "1"		—	2 tc(Pck)	ns	
SDA input buss free time		t_{BUF}	5 tc(Pck)	—	ns	
Start condition input hold time		t_{STAH}	3 tc(Pck)	—	ns	
Retransmission start condition input setup time		t_{STAS}	3 tc(Pck)	—	ns	
Stop condition input setup time		t_{STOS}	3 tc(Pck)	—	ns	
Data input setup time		t_{SDAS}	tc(Pck) + 20	—	ns	
Data input hold time		t_{SDAH}	0	—	ns	
SCL, SDA capacitance load		Cb	0	400	pF	
SCL, SDA output fall time*[1]		t_{Sf}	—	250	ns	

Notes: *1 SCL and SDA output fall time indicates the characteristic of the I/O buffer.

- tc(Pck) indicates the cycle of the peripheral clock (Pck).



Note: *1 S, P, and Sr indicate the following conditions, respectively.
S: Start condition
P: Stop condition
Sr: Retransmission start condition

Figure 38.24 IIC3 Module Input/Output Timing

38.3.12 DRI Timing

Table 38.27 and table 38.28 show the DRI timing.

Table 38.27 DRI Timing (When Special Mode is Off)

Item	Symbol	Min.	Max.	Unit	Figures
DIN input pulse width	tw (DIN)	1.5 tc(PAck)	—	ns	38.25, 38.26
Acquisition period (8-bit width, 16-bit width)	tc (DCAP)	3.5 tc(PAck)	—	ns	
DD input-acquisition edge setup time (DIN3, DIN4)	tsu (DD-E)	15	—	ns	
DD input-acquisition edge setup time (DIN5)	tsu (DD-E)	60 – 2 tc(PAck)	—	ns	
Acquisition edge - DD input hold time	th (E-DD)	15 + tc(PAck)	—	ns	
DIN5 - DD input hold time	th (E-DD)	5 + 3 tc(PAck)	—	ns	
Interval to prevent simultaneous event detection	ts (E-E)	15 + tc(PAck)	—	ns	

Notes:
- Increases a driving ability when the DIN5 event detection (an event specified by the DRIiDINSEL register) is used. To increase a driving ability, set "1" to the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR).
- tc(PAck) indicates the cycle of the peripheral A clock.

Table 38.28 DRI Timing (When Special Mode is On)

Item	Symbol	Min.	Max.	Unit	Figures
DIN input pulse width (DIN0, DIN1, DIN3, DIN4)	tw (DIN)	1.5 tc(PAck)	—	ns	38.25 to 38.28
When DIN3, DIN4 acquisition pulse is selected	tw (DIN)	0.8 tc(PAck)	—	ns	
Acquisition period (8-bit width, 16-bit width)	tc (DCAP)	2 tc(PAck)	—	ns	
DD input-acquisition edge setup time (DIN3, DIN4)	tsu (DD-E)	8	—	ns	
Acquisition edge - DD input hold time	th (E-DD)	12	—	ns	
Interval to prevent simultaneous event detection	ts (E-E)	15 + tc(PAck)	—	ns	
DIN3, DIN4 sampling edge undefined time before DIN1 initialization level release (when direct reset is selected)	tar	8	—	ns	
DIN3, DIN4 sampling edge undefined time before DIN1 initialization level release	tbr	12	—	ns	
Minimum DIN edge count at DINI initialization level in delayed reset mode (minimum width at initialization level)	twDLYDIN1	8 tc(DCAP)*1	—	ns	

Notes: *1 In special mode, 8 tc(CAP) indicates the cycle of DINn selected as the acquisition event.
- tc(PAck) indicates the cycle of the peripheral A clock.

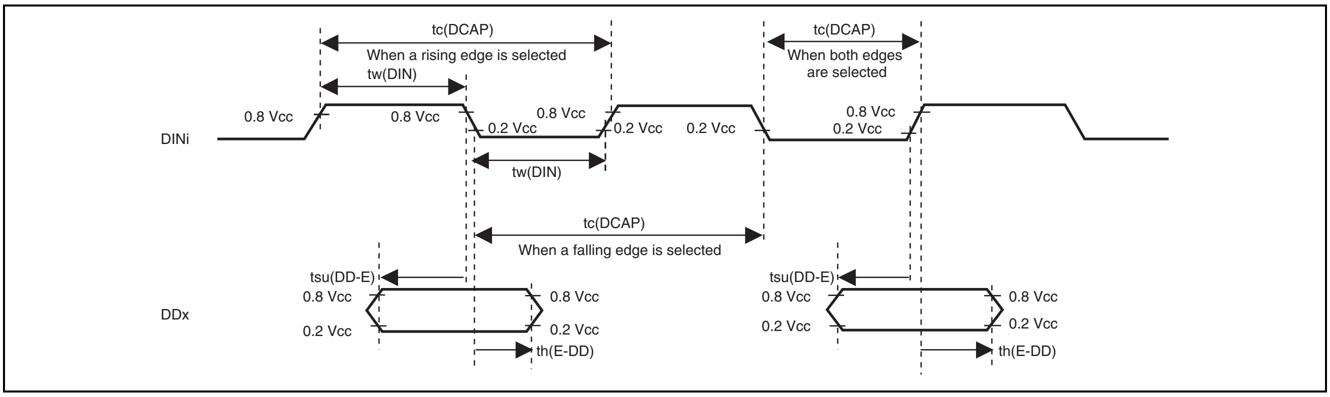


Figure 38.25 Data Acquisition Related Timing

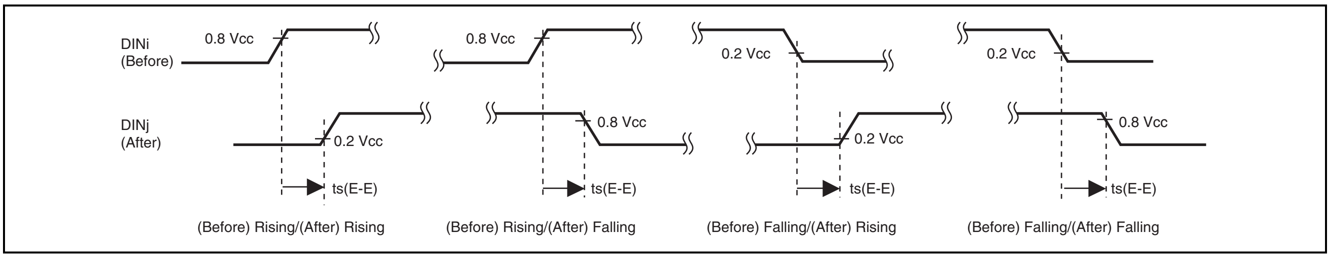

DINi
(Before)
DINj
(After)
0.8 Vcc
0.2 Vcc
ts(E-E)
(Before) Rising/(After) Rising
0.8 Vcc
0.8 Vcc
ts(E-E)
(Before) Rising/(After) Falling
0.2 Vcc
0.2 Vcc
ts(E-E)
(Before) Falling/(After) Rising
0.2 Vcc
0.8 Vcc
ts(E-E)
(Before) Falling/(After) Falling

Figure 38.26 Edge Detection Timing (Edge Interval to Prevent DRI Internal Simultaneous Edge Detection Timing)

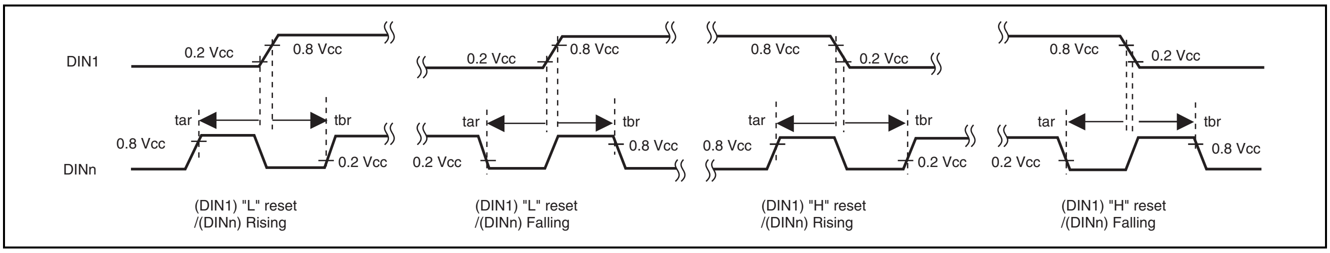


Figure 38.27 DIN1/DIN3, DIN4 Related Timing in Special Mode (DIN1 Reset Release Edge and Preceding or Subsequent Sampling Edge Interval)

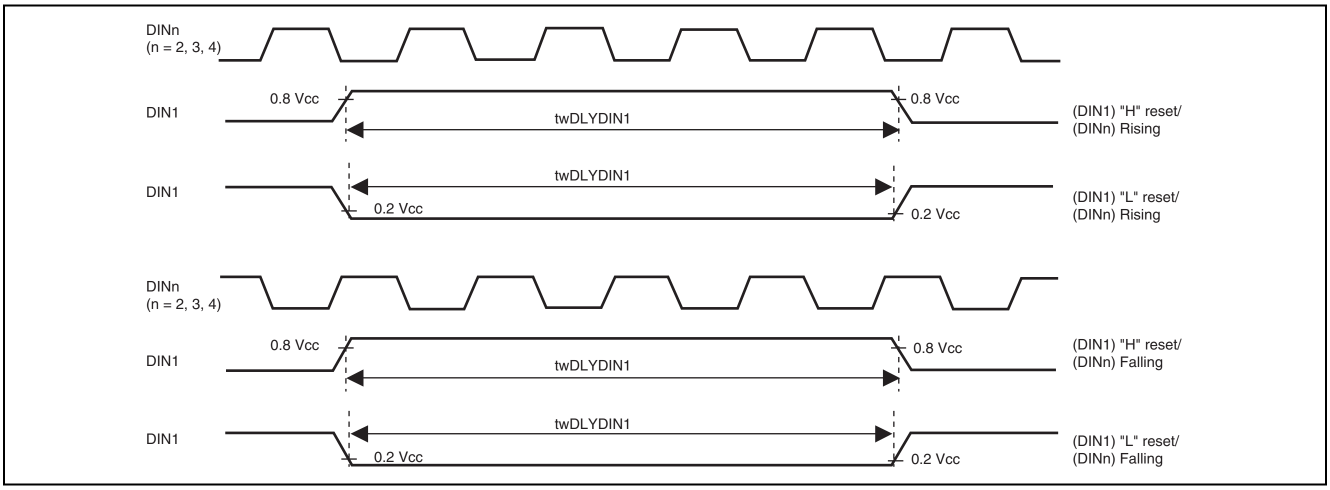


Figure 38.28 Minimum Edge Count at DIN1 Initialization Level in Delayed Reset Mode (Minimum Width at Initialization Level)

38.3.13 DRO Timing

Table 38.29 shows the DRO timing.

Table 38.29 DRO Timing

Condition: All values are for setting conditions that the driving ability is increased.*1

Item	Symbol	Min.	Max.	Unit	Figures
Output data strobe width	twDROWR	(DROSU + 1) × tc(Pck) – 20	—	ns	38.29, 38.30
Output data strobe disable width	twDROWROFF	(DROHD + 1) × tc(Pck) – 20	—	ns	
Data output delay time from output data strobe assertion	tdDROD	—	15	ns	
Data valid period from output data strobe negation	tvDROD	–15	—	ns	

Notes: *1 To increase a driving ability, set "1" to the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR).

- Refer to the DROSU bit in the DRO operating mode register for the details on DROSU.
- Refer to the DROHD bit in the DRO operating mode register for the details on DROHD.
- tc(Pck) indicates the cycle of the peripheral clock (Pck).

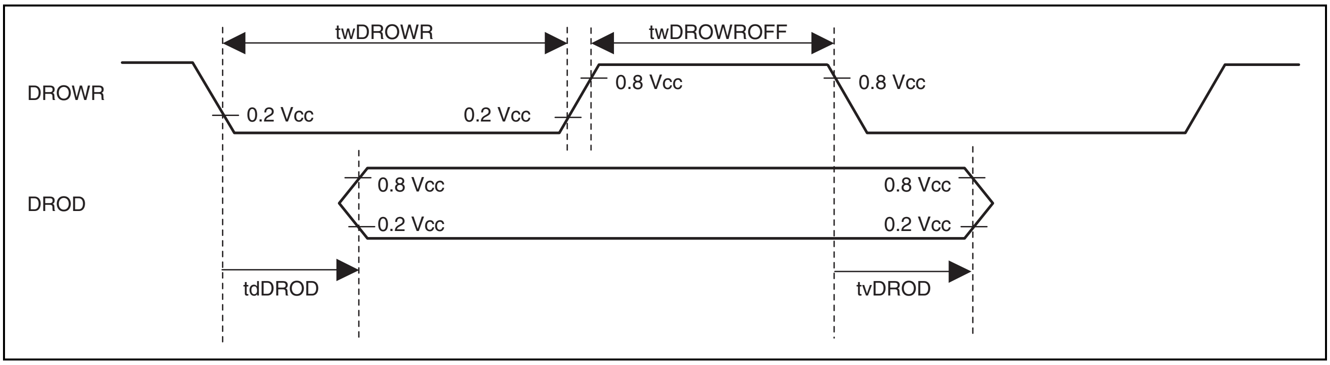


Figure 38.29 Strobe Polarity with "L" Active Selected

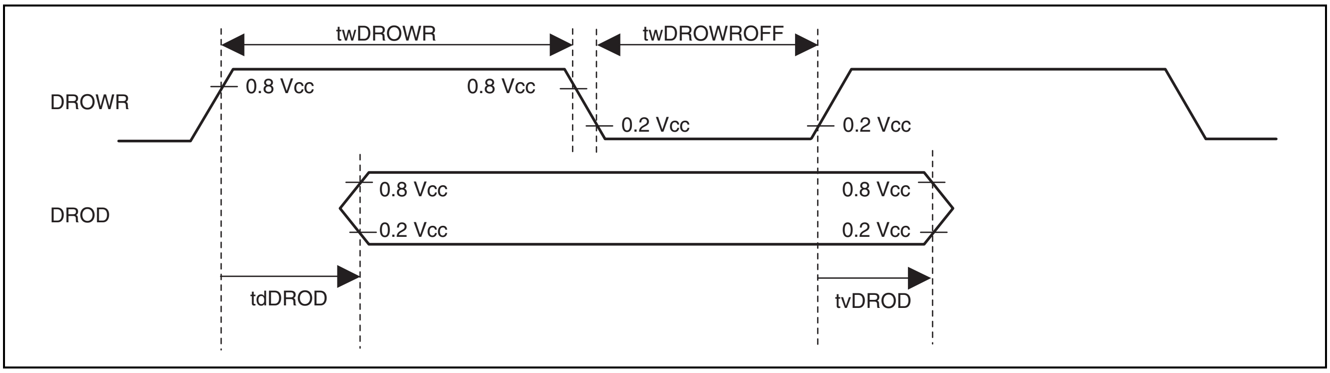


Figure 38.30 Strobe Polarity with "H" Active Selected

38.3.14 PDAC Timing

Table 38.30 shows the PDAC timing.

Table 38.30 PDAC Timing

Item	Symbol	Min.	Max.	Unit	Figures
Timing from set data (PDIDATA9 to 0) update until write signal (PDIWR) assertion	tdPDID	tc(Pck) × PDI_PRE × SETUP – 20	tc(Pck) × PDI_PRE × SETUP + 20	ns	38.31
The timing from write signal (PDIWR) assertion until its negation	twPDIWR	tc(Pck) × PDI_PRE × ENABLE – 20	tc(Pck) × PDI_PRE × ENABLE + 20	ns	

Note: • tc(Pck) indicates the cycle of the peripheral clock (Pck).

Legend: PDI_PRE: The setting value of the PDAC basic resolution setting register (setting "0" or "1" is prohibited)
SETUP: The setting value of the PDAC write signal period adjustment register (setting "0" is prohibited)
ENABLE: The setting value of the PDAC write signal period adjustment register (setting "0" is prohibited)

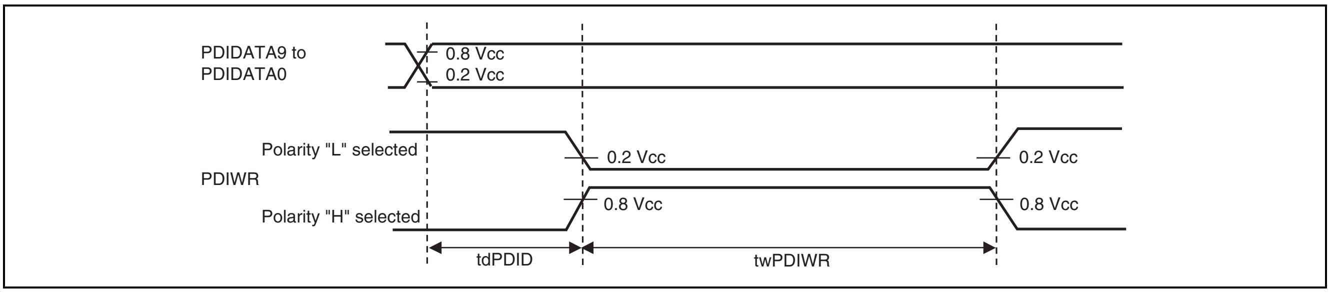


Figure 38.31 PDAC Output Timing

38.3.15 PSEL Timing

Table 38.31 shows the PSEL timing.

Table 38.31 PSEL Timing

Item	Symbol	Min.	Max.	Unit	Figures
PSLCLKA/PSLCLKB output pulse width	twPSLCLK	tc(Pck) × PSL_PRE – 20	—	ns	38.32
PSLCLKA—PSLDATA output delay time	tdPSLDATA	—	20	ns	
PSLCLKA—PSLDATA output hold time	thPSLDATA	–15	—	ns	
PSLCLKA—PSLCLR interval output delay time	tdPSLCLR	–15	25	ns	

Note: • tc(Pck) indicates the cycle of the peripheral clock (Pck).

Legend: PSL_PRE: The setting value of the PSEL output clock divisor setting register (setting "0" is prohibited)

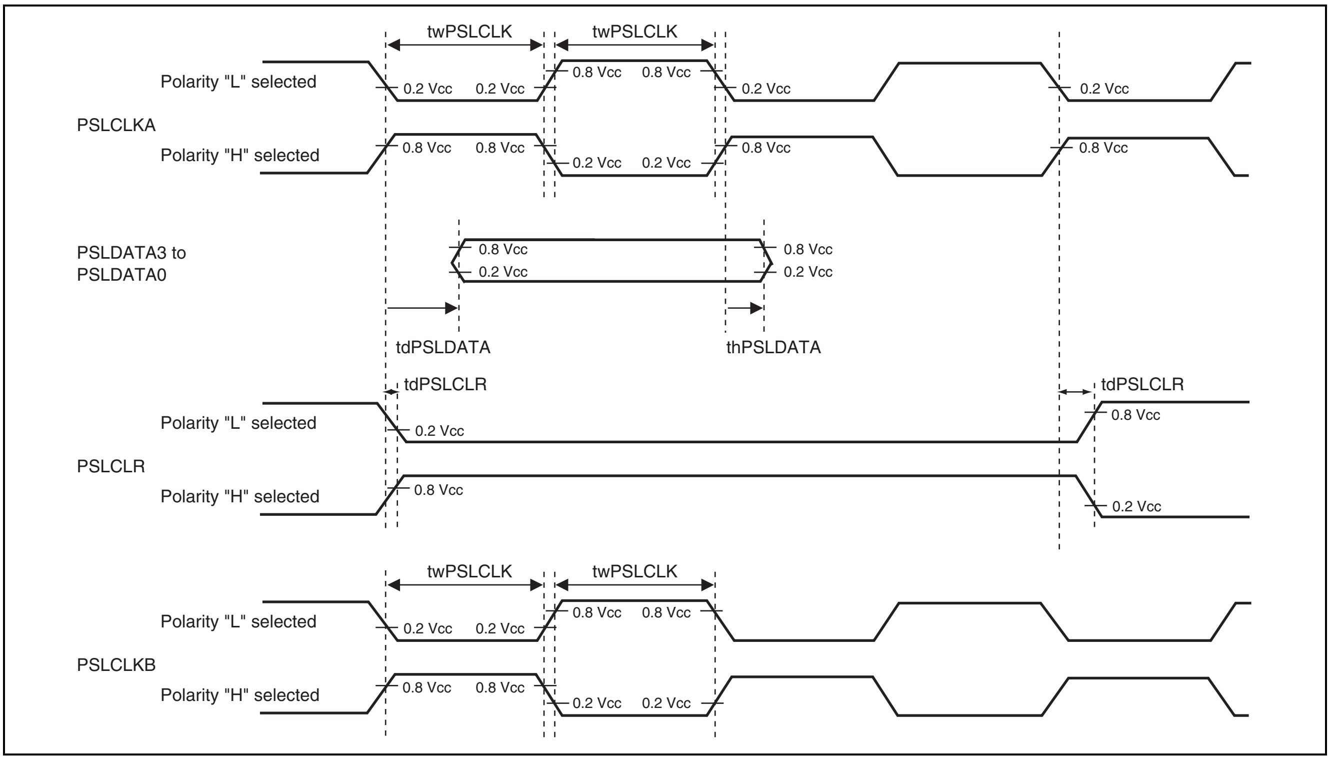


Figure 38.32 PSEL Output Timing

38.3.16 A/D Converter Timing

Table 38.32 shows the A/D converter timing.

Table 38.32 A/D Converter Timing

Item	Symbol	Min.	Max.	Unit	Figures
External trigger input start delay time	t_{TRGS}	50	—	ns	38.33
ADEND output delay time	t_{ADENDD}	—	100	ns	38.34

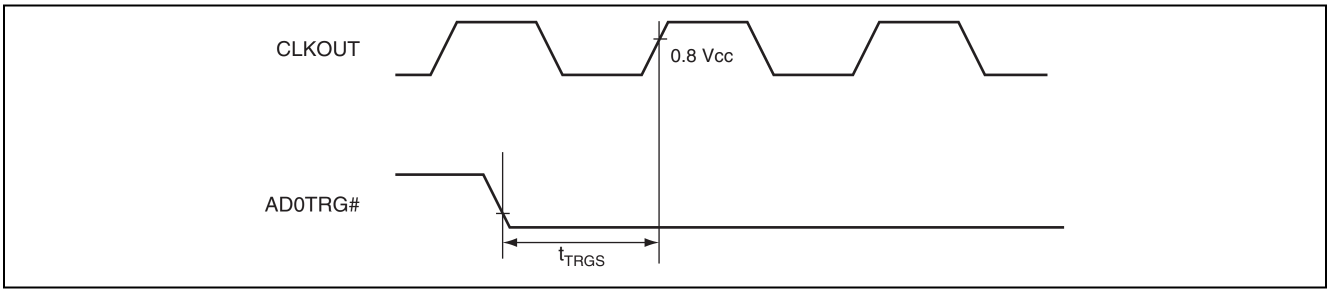


Figure 38.33 External Trigger Input Timing

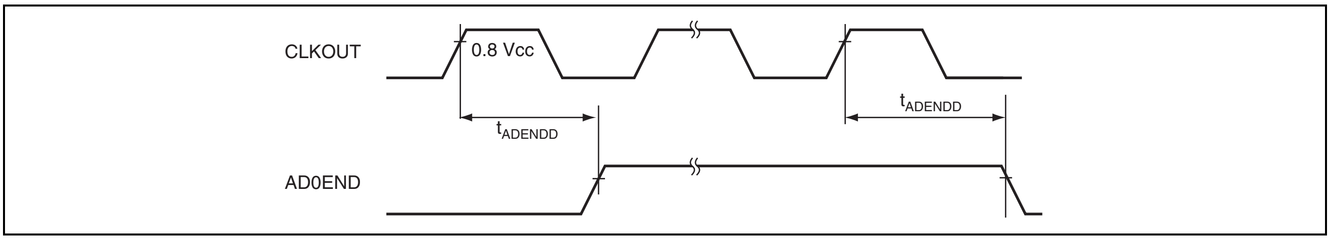


Figure 38.34 Analog Conversion Timing

38.3.17 H-UDI Interface Timing

Table 38.33 shows the H-UDI interface timing.

Table 38.33 H-UDI Interface Timing

Item	Symbol	Min.	Max.	Unit	Figures
TCK clock cycle	t_{TCKcyc}	2 tc(Pck)	—	ns	38.35
TCK clock pulse width	t_{TCKw}	0.4 tc(TCK)	0.6 tc(TCK)	ns	
TRST# pulse width	t_{TRSTw}	20 tc(TCK)	—	ns	38.36
TMS setup time	t_{TMSS}	15	—	ns	38.37
TMS hold time	t_{TMSH}	15	—	ns	
TDI setup time	t_{TDIS}	15	—	ns	
TDI hold time	t_{TDIH}	15	—	ns	
TDO data delay time	t_{TDOD}	—	40	ns	

Notes:
- tc(Pck) indicates the cycle of the peripheral clock (Pck).
- tc(TCK) indicates the cycle of the TCK.

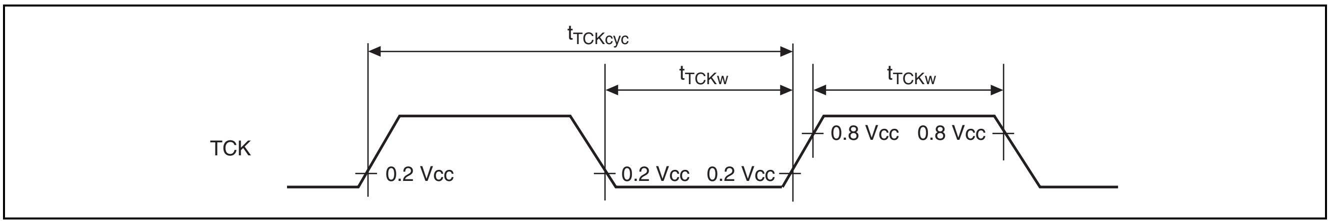


Figure 38.35 TCK Input Timing

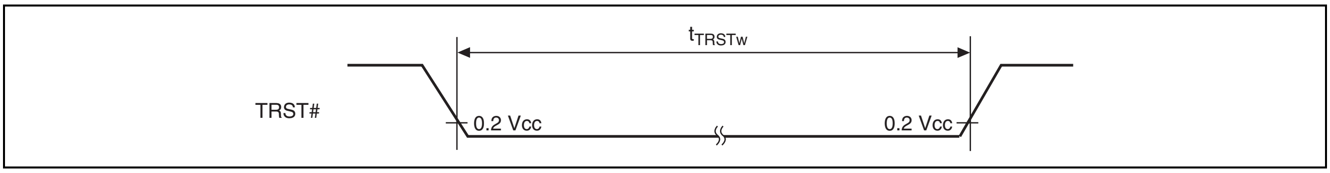


Figure 38.36 TRST# Input Timing

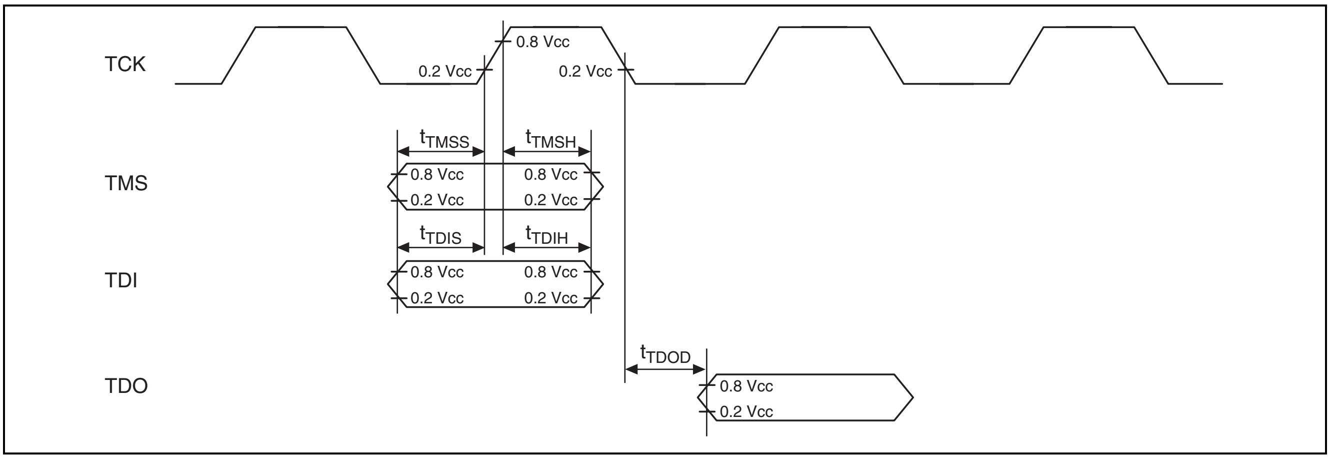


Figure 38.37 H-UDI Data Transfer Timing

38.3.18 AUDR Module Timing

Tables 38.34 and 38.35 show the AUDR module timing.

Table 38.34 AUDR Module Timing (Vcc = 5.0 V)

Condition: All values are for setting conditions that the driving ability is increased.*[1]
Vcc = 5.0 V ±0.5 V

Item	Symbol	Min.	Max.	Unit	Figures
AUDRCLK cycle time	tc(AUDRCLK)	80	—	ns	38.38
AUDRCLK input pulse width	tw(AUDRCLK)	35	—	ns	
AUDRD input setup time before AUDRCLK	tsu(AUDRD-AUDRCLKH)	20	—	ns	
AUDRD input hold time after AUDRCLK	th(AUDRCLKH-AUDRD)	7	—	ns	
AUDRSYN# input setup time before AUDRCLK	tsu(AUDRSYNL-AUDRCLKH)	20	—	ns	
AUDRSYN# input hold time after AUDRCLK	th(AUDRCLKH-AUDRSYNL)	7	—	ns	
AUDRD output delay time before AUDRCLK	td(AUDRCLKH-AUDRD)	—	35	ns	
AUDRD output enable time after AUDRCLK	tv(AUDRCLKH-AUDRD)	3	—	ns	
AUDREVT# output "L" pulse width	tw(AUDREVTL)	2 tc(Pck) –20	—	ns	

Notes: *1 To increase a driving ability, set "1" to the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR).

- tc(Pck) indicates the cycle of the peripheral clock (Pck).

Table 38.35 AUDR Module Timing (Vcc = 3.3 V)

Condition: All values are for setting conditions that the driving ability is increased.*[1]
Vcc = 3.3 V ±0.3 V

Item	Symbol	Min.	Max.	Unit	Figures
AUDRCLK cycle time	tc(AUDRCLK)	80	—	ns	38.38
AUDRCLK input pulse width	tw(AUDRCLK)	35	—	ns	
AUDRD input setup time before AUDRCLK	tsu(AUDRD-AUDRCLKH)	20	—	ns	
AUDRD input hold time after AUDRCLK	th(AUDRCLKH-AUDRD)	7	—	ns	
AUDRSYN# input setup time before AUDRCLK	tsu(AUDRSYNL-AUDRCLKH)	20	—	ns	
AUDRSYN# input hold time after AUDRCLK	th(AUDRCLKH-AUDRSYNL)	7	—	ns	
AUDRD output delay time before AUDRCLK	td(AUDRCLKH-AUDRD)	—	40	ns	
AUDRD output enable time after AUDRCLK	tv(AUDRCLKH-AUDRD)	3	—	ns	
AUDREVT# output "L" pulse width	tw(AUDREVTL)	2 tc(Pck) –20	—	ns	

Notes: *1 To increase a driving ability, set "1" to the corresponding bit in the port A to H and J to L driving ability setting registers (PADSR to PHDSR and PJDSR to PLDSR).

- tc(Pck) indicates the cycle of the peripheral clock (Pck).

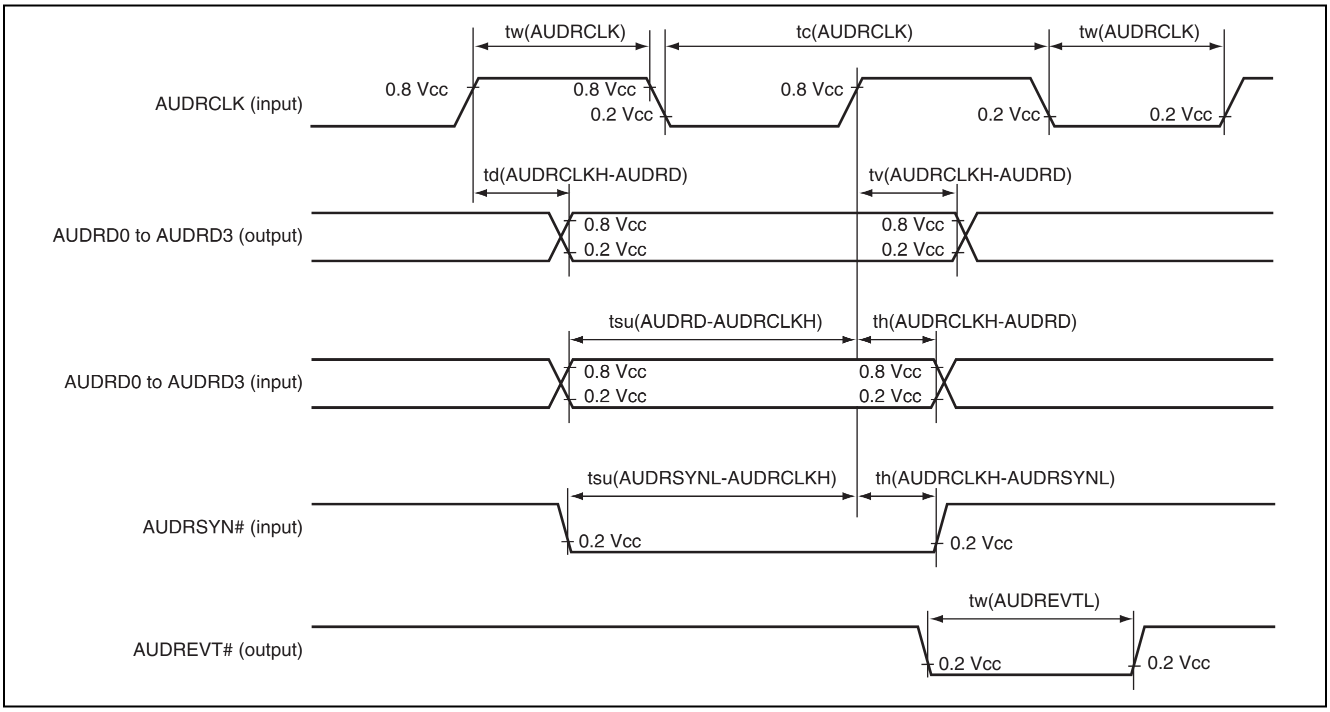


Figure 38.38 AUDR Module Timing

38.4 A/D Converter Characteristics

Table 38.36 shows the A/D converter characteristics.

Table 38.36 A/D Converter Characteristics

Item			Symbol	Min.	Typ.	Max.	Unit
Digit resolution			—	12	—	12	bit
Voltage resolution*[1]	AVcc = 5.0 V		—	1.10	—	1.34	mV
	AVcc = 3.3 V		—	0.73	—	0.88	mV
A/D conversion time*[2]	AVcc = 5.0 V	High-speed conversion	—	—	50 × tc(Pck)	—	ns
		Low-speed conversion	—	—	100 × tc(Pck)	—	ns
	AVcc = 3.3 V	High-speed conversion	—	—	50 × tc(Pck)	—	ns
		Low-speed conversion	—	—	100 × tc(Pck)	—	ns
Non-linear error	AVcc = 5.0 V	High-speed conversion	—	—	—	±8	LSB
		Low-speed conversion	—	—	—	±4	LSB
	AVcc = 3.3 V	High-speed conversion	—	—	—	±16	LSB
		Low-speed conversion	—	—	—	±16	LSB
Offset error	AVcc = 5.0 V	High-speed conversion	—	—	—	±15.5	LSB
		Low-speed conversion	—	—	—	±7.5	LSB
	AVcc = 3.3 V	High-speed conversion	—	—	—	±31.5	LSB
		Low-speed conversion	—	—	—	±31.5	LSB
Full-scale error	AVcc = 5.0 V	High-speed conversion	—	—	—	±15.5	LSB
		Low-speed conversion	—	—	—	±7.5	LSB
	AVcc = 3.3 V	High-speed conversion	—	—	—	±31.5	LSB
		Low-speed conversion	—	—	—	±31.5	LSB
Quantization error			—	0.5	0.5	0.5	LSB
Absolute error	AVcc = 5.0 V	High-speed conversion	—	—	—	±16	LSB
		Low-speed conversion	—	—	—	±8	LSB
	AVcc = 3.3 V	High-speed conversion	—	—	—	±32	LSB
		Low-speed conversion	—	—	—	±32	LSB
Self-diagnostic absolute error	AVcc = 5.0 V	High-speed conversion	—	—	—	±80	LSB
		Low-speed conversion	—	—	—	±40	LSB
	AVcc = 3.3 V	High-speed conversion	—	—	—	±160	LSB
		Low-speed conversion	—	—	—	±160	LSB
Analog input capacitance	Awaiting		—	—	—	20	pF
	Sampling		—	—	—	40	pF
Permitted analog signal source impedance			—	—	—	3	kΩ

Notes: *1 At AVREFH - AVREFL = 3.0 V, resolution is 0.73 mV. At AVREFH - AVREFL = 3.6 V, resolution is 0.88 mV. At AVREFH - AVREFL = 4.5 V, resolution is 1.10 mV. At AVREFH - AVREFL = 5.5 V, resolution is 1.34 mV.

*2 The A/D conversion time depends on the CKS bit settings in the AD0CER or AD1CER register.

- tc(Pck) indicates the cycle of the peripheral clock (Pck).

38.5 Flash Memory Characteristics

Table 38.37 shows the flash memory characteristics.

Table 38.37 Flash Memory Characteristics

Item		Symbol	Min.	Typ.	Max.	Unit
Programming time	256 bytes	t_{P256}	—	2	12	ms
Erase time	8 Kbytes	t_{E8K}	—	50	150	ms
	32 Kbytes*[1]	t_{E32K}	—	200	560	ms
	64 Kbytes	t_{E64K}	—	400	1120	ms
	128 Kbytes	t_{E128K}	—	800	2240	ms
Number of times for reprogramming/erasing*[2]		N_{PEC}	100*[3]	—	—	Times
FRESET "1" setting time		t_{RESW2}	100	—	—	µs

Notes: *1 User Boot Mat is 32 Kbytes.

*2 Definition of the number of times for reprogramming/erasing.

The number of times for reprogramming/erasing is the number of times for erase for each blocks. When the number of times is 100, it can be erased 100 times for each blocks. However, the same address must not be programmed more than once per erase operation (overwriting prohibited).

*3 Endurance to guarantee all characteristics after reprogramming/erasing (1 to minimum value can be guaranteed).

This page is blank for reasons of layout.

Appendix A CPU Operation Mode Register (CPUOPM)

The CPUOPM register is used to control the CPU operating mode. This register can be read from or written to the address H'FF2F 0000 in 32-bit size. The write value to the reserved bits must be the initial value. Operation is not guaranteed if the write value is other than the initial value.

The CPUOPM register should be updated by the MOV instruction of the CPU not the access from the SuperHyway bus master except the CPU. After the CPUOPM register is updated, read the CPUOPM register once, and execute one of the following two methods.

1. Execute a branch using the RTE instruction.
2. Execute the ICBI instruction for any address (including non-cacheable area).

The CPU is guaranteed to operate using the updated CPUOPM register value after either of the above methods has been executed.

CPU Operation Mode Register (CPUOPM) <P4 address: location H'FF2F 0000>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	RABD	—	INTMU	—	—	—
After Reset:	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0

<After Reset: H'0000 03C0>

Bit	Abbreviation	After Reset	R	W	Description
31 to 6	—	H'000000F	R	*1	Reserved Bits The write value must be the initial value.
5	RABD	0	R	W	Speculative Execution Bit For Subroutine Return Bit 0: Instruction fetch for subroutine return is issued speculatively. When this bit is set to "0", refer to appendix C, Speculative Execution for Subroutine Return. 1: Instruction fetch for subroutine return is not issued speculatively.
4	—	0	R	*1	Reserved Bit The write value must be the initial value.
3	INTMU	0	R	W	Interrupt Mode Switch Bit 0: SR.IMASK value is not changed when an interrupt is accepted. 1: SR.IMASK value is changed to the accepted interrupt level.
2 to 0	—	All 0	R	*1	Reserved Bits The write value must be the initial value.

Note: *1 When writing to these bits, always write the initial value.

This page is blank for reasons of layout.

Appendix B Instruction Prefetching and Its Side Effects

The SH-4A is provided with an internal buffer for holding pre-read instructions, and always performs pre-reading. Therefore, program must not be located in the last 64-byte area of any memory space. If program is located in these areas, a bus access for instruction prefetch may occur exceeding the memory areas boundary.

A case in which this is a problem is shown below.

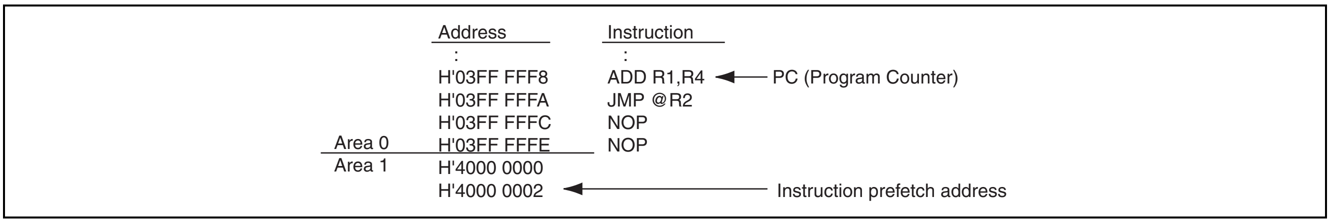


Figure B.1 Instruction Prefetch

Figure B.1 presupposes a case in which the instruction (ADD) indicated by the program counter (PC) and the address H'0400 0002 instruction prefetch are executed simultaneously. It is also assumed that the program branches to an area other than area 1 after executing the following JMP instruction and delay slot instruction.

In this case, a bus access (instruction prefetch) to area 1 may unintentionally occur from the programming flow.

(1) Instruction Prefetch Side Effects

1. It is possible that an external bus access caused by an instruction prefetch may result in misoperation of an external device, such as a FIFO, connected to the area concerned.
2. If there is no device to reply to an external bus request caused by an instruction prefetch, hang-up will occur.

(2) Remedies

1. These illegal instruction fetches can be avoided by using the MMU.
2. The problem can be avoided by not locating program in the last 64 bytes of any area.

This page is blank for reasons of layout.

Appendix C Speculative Execution for Subroutine Return

The SH-4A has the mechanism to issue an instruction fetch speculatively when returning from subroutine. By issuing an instruction fetch speculatively, the execution cycles to return from subroutine may be shortened.

This function is enabled by setting "0" to the bit 5 (RABD) of the CPU Operation Mode register (CPUOPM). But this speculative instruction fetch may issue the access to the address that should not be accessed from the program. Therefore, a bus access to an unexpected area or an internal instruction address error may cause a problem. As for the effect of this bus access to unexpected memory area, refer to appendix B (1), Instruction Prefetch Side Effects.

(1) Usage Condition

When the speculative execution for subroutine return is enabled, the RTS instruction should be used to return to the address set in PR by the JSR, BSR, or BSRF instructions. It can prevent the access to unexpected address and avoid the problem.

This page is blank for reasons of layout.

Appendix D Processor Version Register (PVR)

The SH-4A has the read-only register which show the version of a processor core. This makes it possible to implement highly scalable systems since the software can determine the processor version from the value of this register.

Note: • The bit 7 to bit 0 of the PVR register should be masked by the software.

Table D.1 Register Configuration

Register Name	Abbreviation	After Reset	P4 Address	Size	Page
Processor version register	PVR	Undefined	H'FF00 0030	32	D-7

Note: • The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

Processor Version Register (PVR) <P4 address: location H'FF00 0030>

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CHIP								VER							
After Reset:	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CUT								—	—	—	—	—	—	—	—
After Reset:	0	0	0	0	1	0	0	0	—	—	—	—	—	—	—	—

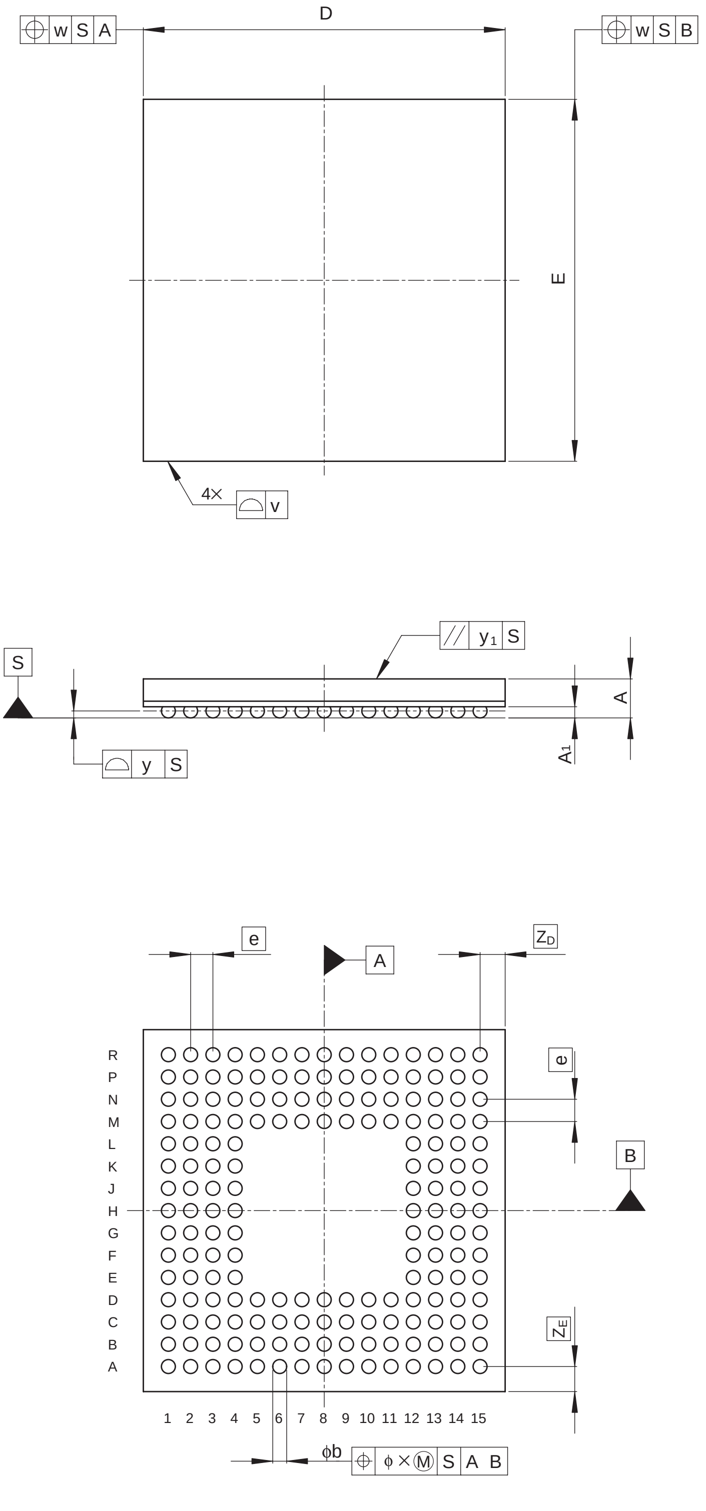
<After Reset: Undefined>

Bit	Abbreviation	After Reset	R	W	Description
31 to 24	CHIP	H'10	R	—	Processor Family The read value is always "H'10" in the SH-4A.
23 to 16	VER	H'30	R	—	Major Version This value is changed when performing major enhancement of the architecture.
15 to 8	CUT	H'08	R	—	Minor Version This value is changed when performing minor enhancement of the architecture.
7 to 0	—	Undefined	R	—	These bits read as undefined. It should be masked by software when using it.

This page is blank for reasons of layout.

Appendix E Package Dimensions

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-FBGA176-13x13-0.80	PRBG0176GA-A	—	0.5g



Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	——	13.0	——
E	——	13.0	——
v	——	——	0.15
w	——	——	0.20
A	——	——	1.90
A_1	0.35	0.4	0.45
e	——	0.8	——
b	0.45	0.50	0.55
x	——	——	0.08
y	——	——	0.10
y_1	——	——	0.20
S_D	——	——	——
S_E	——	——	——
Z_D	——	0.9	——
Z_E	——	0.9	——

Figure E.1 Package Dimensions

This page is blank for reasons of layout.

Appendix F Index Indication

Figure F.1 shows the location of the index indication on the top of the package. The information printed on the package differs depending on the customer’s product specifications. For details, please contact Renesas Electronics Corp.

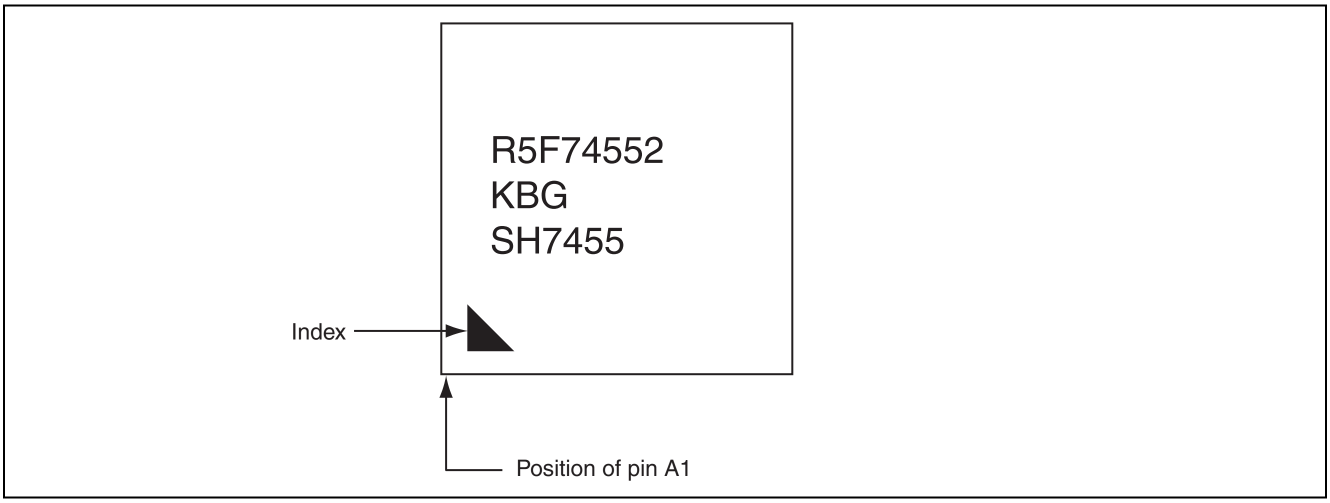


Figure F.1 Index Indication (Example: SH74552)

This page is blank for reasons of layout.

Appendix G Register Assignments

(1) Reading the Address Listings

The addresses in the address listings in table G.1 in this section are P4 addresses. The P4 address applies when the P4 area of the virtual address space is used. When the upper three bits of the 32-bit P4 area address are cleared to "0", the result is an area 7 address. This address is used when accessing area 7 of the physical address space using the TLB.

(2) Dummy Access Area

Address H'FFA0 0000 to H'FFA0 0003 are specified as the dummy access area (DUMMYHPB1) in the peripheral A bus (PAck). Address H'FFFF 5020 to H'FFFF 5023 are specified as the dummy access area (DUMMYHPB0) in the peripheral bus (Pck). For accesses to this area, the write value should be invalid and the read value should be undefined. Write or read operation to the dummy access area does not affect other register areas.

(3) Reading Access Size Column

The access size is shown by using 32-bit configuration per line.

8-bit access is indicated as "8", 16-bit access as "16", and 32-bit access as "32", respectively.

The access size for prohibited areas is indicated as "-".

For registers that allow multiple accesses, each access size is indicated with a slash (/).

If an access size is indicated without a slash (/), only the indicated size is allowed.

- For 32-bit registers that can be accessed using 32-bit and 16-bit accesses
  The access size is indicated as "16/32".
- For 8-bit registers that can be accessed using 8-bit access, and also using 16-bit access at the same time with the next aligned 8-bit register
  The access size is indicated as "8/16".
- When multiple registers are included in one line and the units of their access size are different
  For example, the access size is indicated as "8/16, 8, 8", separated with a comma ",".
- When multiple registers are included in one line and the units of their access size are the same
  For example, the access size is collectively indicated as "8", not separated as "8, 8, 8, 8".

(4) Notes on SH7456 Group Register Addresses

The SH7456 Group has no FlexRay module. Addresses H'FFBF F000 to H'FFBF FFFF of the FlexRay-related registers are reserved areas. The read value should be undefined. Do not write to these registers.

Table G.1 Register Assignments

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FC11 0000	Instruction Register (SDIR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FC11 0018	Interrupt Source Register (SDINT)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FDFF A800	Flash Pin Monitor Register (FPMON)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FDFF A810	Flash Access Status Register (FASTAT)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FDFF A820	ROM MAT Select Register (ROMMAT)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FDFF A900	Flash Status Register 0 (FSTATR0)	Flash Status Register 1 (FSTATR1)	Flash P/E Mode Entry Register (FENTRYR)		8/16
H'FDFF A904	Flash Protect Register (FPROTR)		Flash Reset Register (FRESETR)		8/16
:	(Reserved)				-
H'FDFF A91C	Flash P/E Status Register (FPESTAT)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FE40 0000	AUDR Enable Register (AUDRENB)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FE40 0008	AUDR Event Generation Register (AUDREVNT)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FE40 0010	AUDR Configuration Information Retention Register (AUDISR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FE40 0018	AUDR Message Board Register (AUDMBR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FF00 0000	Page Table Entry High Register (PTEH)				32
H'FF00 0004	Page Table Entry Low Register (PTEL)				32
H'FF00 0008	Translation Table Base Register (TTB)				32
H'FF00 000C	TLB Exception Address Register (TEA)				32
H'FF00 0010	MMU Control Register (MMUCR)				32
:	(Reserved)				-
H'FF00 001C	Cache Control Register (CCR)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FF00 0020	TRAPA Exception Register (TRA)				32
H'FF00 0024	Exception Event Register (EXPEVT)				32
H'FF00 0028	Interrupt Event Register (INTEVT)				32
:	(Reserved)				-
H'FF00 0030	Processor Version Register (PVR)				32
H'FF00 0034	Page Table Entry Assistance Register (PTEA)				32
H'FF00 0038	Queue Address Control Register 0 (QACR0)				32
H'FF00 003C	Queue Address Control Register 1 (QACR1)				32
:	(Reserved)				-
H'FF00 0050	OL memory Transfer Source Address Register 0 (LSA0)				32
H'FF00 0054	OL memory Transfer Source Address Register 1 (LSA1)				32
H'FF00 0058	OL memory Transfer Destination Address Register 0 (LDA0)				32
H'FF00 005C	OL memory Transfer Destination Address Register 1 (LDA1)				32
:	(Reserved)				-
H'FF00 0070	Physical Address Space Control Register (PASCR)				32
H'FF00 0074	On-Chip Memory Control Register (RAMCR)				32
H'FF00 0078	Instruction Re-Fetch Inhibit Control Register (IRMCR)				32
:	(Reserved)				-
H'FF20 0000	Match Condition Setting Register 0 (CBR0)				32
H'FF20 0004	Match Operation Setting Register 0 (CRR0)				32
H'FF20 0008	Match Address Setting Register 0 (CAR0)				32
H'FF20 000C	Match Address Mask Setting Register 0 (CAMR0)				32
:	(Reserved)				-
H'FF20 0020	Match Condition Setting Register 1 (CBR1)				32
H'FF20 0024	Match Operation Setting Register 1 (CRR1)				32
H'FF20 0028	Match Address Setting Register 1 (CAR1)				32
H'FF20 002C	Match Address Mask Setting Register 1 (CAMR1)				32
H'FF20 0030	Match Data Setting Register 1 (CDR1)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FF20 0034	Match Data Mask Setting Register 1 (CDMR1)				32
H'FF20 0038	Execution Count Break Register 1 (CETR1)				32
:	(Reserved)				-
H'FF20 0600	Channel Match Flag Register (CCMFR)				32
:	(Reserved)				-
H'FF20 0620	Break Control Register (CBCR)				32
:	(Reserved)				-
H'FF2F 0000	CPU Operation Mode Register (CPUOPM)				32
H'FF2F 0004	Unsupported Function Detection Exception Register (EXPMASK)				32
:	(Reserved)				-
H'FF60 8020	DMA0 Source Address Register (DM0SAR)				32
H'FF60 8024	DMA0 Destination Address Register (DM0DAR)				32
H'FF60 8028	DMA0 Transfer Count Register (DM0TCR)				32
H'FF60 802C	DMA0 Channel Control Register (DM0CHCR)				32
H'FF60 8030	DMA1 Source Address Register (DM1SAR)				32
H'FF60 8034	DMA1 Destination Address Register (DM1DAR)				32
H'FF60 8038	DMA1 Transfer Count Register (DM1TCR)				32
H'FF60 803C	DMA1 Channel Control Register (DM1CHCR)				32
H'FF60 8040	DMA2 Source Address Register (DM2SAR)				32
H'FF60 8044	DMA2 Destination Address Register (DM2DAR)				32
H'FF60 8048	DMA2 Transfer Count Register (DM2TCR)				32
H'FF60 804C	DMA2 Channel Control Register (DM2CHCR)				32
H'FF60 8050	DMA3 Source Address Register (DM3SAR)				32
H'FF60 8054	DMA3 Destination Address Register (DM3DAR)				32
H'FF60 8058	DMA3 Transfer Count Register (DM3TCR)				32
H'FF60 805C	DMA3 Channel Control Register (DM3CHCR)				32
H'FF60 8060	DMA05 Operation Register (DM05OR)		(Reserved)	(Reserved)	16, -, -
H'FF60 8070	DMA4 Source Address Register (DM4SAR)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FF60 8074	DMA4 Destination Address Register (DM4DAR)				32
H'FF60 8078	DMA4 Transfer Count Register (DM4TCR)				32
H'FF60 807C	DMA4 Channel Control Register (DM4CHCR)				32
H'FF60 8080	DMA5 Source Address Register (DM5SAR)				32
H'FF60 8084	DMA5 Destination Address Register (DM5DAR)				32
H'FF60 8088	DMA5 Transfer Count Register (DM5TCR)				32
H'FF60 808C	DMA5 Channel Control Register (DM5CHCR)				32
:	(Reserved)				-
H'FF60 8120	DMA0 Source Address Register B (DM0SARB)				32
H'FF60 8124	DMA0 Destination Address Register B (DM0DARB)				32
H'FF60 8128	DMA0 Transfer Count Register B (DM0TCRB)				32
:	(Reserved)				-
H'FF60 8130	DMA1 Source Address Register B (DM1SARB)				32
H'FF60 8134	DMA1 Destination Address Register B (DM1DARB)				32
H'FF60 8138	DMA1 Transfer Count Register B (DM1TCRB)				32
:	(Reserved)				-
H'FF60 8140	DMA2 Source Address Register B (DM2SARB)				32
H'FF60 8144	DMA2 Destination Address Register B (DM2DARB)				32
H'FF60 8148	DMA2 Transfer Count Register B (DM2TCRB)				32
:	(Reserved)				-
H'FF60 8150	DMA3 Source Address Register B (DM3SARB)				32
H'FF60 8154	DMA3 Destination Address Register B (DM3DARB)				32
H'FF60 8158	DMA3 Transfer Count Register B (DM3TCRB)				32
:	(Reserved)				-
H'FF60 9000	DMA01 Extended Resource Select Register (DM01ARS)		(Reserved)	(Reserved)	16, -, -
H'FF60 9004	DMA23 Extended Resource Select Register (DM23ARS)		(Reserved)	(Reserved)	16, -, -
H'FF60 9008	DMA45 Extended Resource Select Register (DM45ARS)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FF61 8020	DMA6 Source Address Register (DM6SAR)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FF61 8024	DMA6 Destination Address Register (DM6DAR)				32
H'FF61 8028	DMA6 Transfer Count Register (DM6TCR)				32
H'FF61 802C	DMA6 Channel Control Register (DM6CHCR)				32
H'FF61 8030	DMA7 Source Address Register (DM7SAR)				32
H'FF61 8034	DMA7 Destination Address Register (DM7DAR)				32
H'FF61 8038	DMA7 Transfer Count Register (DM7TCR)				32
H'FF61 803C	DMA7 Channel Control Register (DM7CHCR)				32
H'FF61 8040	DMA8 Source Address Register (DM8SAR)				32
H'FF61 8044	DMA8 Destination Address Register (DM8DAR)				32
H'FF61 8048	DMA8 Transfer Count Register (DM8TCR)				32
H'FF61 804C	DMA8 Channel Control Register (DM8CHCR)				32
H'FF61 8050	DMA9 Source Address Register (DM9SAR)				32
H'FF61 8054	DMA9 Destination Address Register (DM9DAR)				32
H'FF61 8058	DMA9 Transfer Count Register (DM9TCR)				32
H'FF61 805C	DMA9 Channel Control Register (DM9CHCR)				32
H'FF61 8060	DMA611 Operation Register (DM611OR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FF61 8070	DMA10 Source Address Register (DM10SAR)				32
H'FF61 8074	DMA10 Destination Address Register (DM10DAR)				32
H'FF61 8078	DMA10 Transfer Count Register (DM10TCR)				32
H'FF61 807C	DMA10 Channel Control Register (DM10CHCR)				32
H'FF61 8080	DMA11 Source Address Register (DM11SAR)				32
H'FF61 8084	DMA11 Destination Address Register (DM11DAR)				32
H'FF61 8088	DMA11 Transfer Count Register (DM11TCR)				32
H'FF61 808C	DMA11 Channel Control Register (DM11CHCR)				32
:	(Reserved)				-
H'FF61 8120	DMA6 Source Address Register B (DM6SARB)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FF61 8124	DMA6 Destination Address Register B (DM6DARB)				32
H'FF61 8128	DMA6 Transfer Count Register B (DM6TCRB)				32
:	(Reserved)				-
H'FF61 8130	DMA7 Source Address Register B (DM7SARB)				32
H'FF61 8134	DMA7 Destination Address Register B (DM7DARB)				32
H'FF61 8138	DMA7 Transfer Count Register B (DM7TCRB)				32
:	(Reserved)				-
H'FF61 8140	DMA8 Source Address Register B (DM8SARB)				32
H'FF61 8144	DMA8 Destination Address Register B (DM8DARB)				32
H'FF61 8148	DMA8 Transfer Count Register B (DM8TCRB)				32
:	(Reserved)				-
H'FF61 8150	DMA9 Source Address Register B (DM9SARB)				32
H'FF61 8154	DMA9 Destination Address Register B (DM9DARB)				32
H'FF61 8158	DMA9 Transfer Count Register B (DM9TCRB)				32
:	(Reserved)				-
H'FF61 9000	DMA67 Extended Resource Select Register (DM67ARS)		(Reserved)	(Reserved)	16, -, -
H'FF61 9004	DMA89 Extended Resource Select Register (DM89ARS)		(Reserved)	(Reserved)	16, -, -
H'FF61 9008	DMA1011 Extended Resource Select Register (DM1011ARS)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFA0 0000	Dummy Access Area (DUMMYHPB1)				8/16/32
:	(Reserved)				-
H'FFBF C000	DRI0DIN Interrupt Request Status Register (DRI0DINIST)	DRI0DIN Interrupt Request Enable Register (DRI0DINIEN)	DRI0DIN DMA Transfer Request Status Register (DRI0DINDST)	DRI0DIN DMA Transfer Enable Register (DRI0DINDEN)	8
H'FFBF C004	DRI0DEC Interrupt Request Status Register (DRI0DECIST)	DRI0DEC Interrupt Request Enable Register (DRI0DECIEN)	DRI0DEC DMA Transfer Request Status Register (DRI0DECDST)	DRI0DEC DMA Transfer Enable Register (DRI0DECDEN)	8
H'FFBF C008	DRI0 Transfer Interrupt Request Status Register (DRI0TRMIST)	DRI0 Transfer Interrupt Request Enable Register (DRI0TRMIEN)	DRI0DMA Transfer Request Status Register (DRI0TRMDST)	DRI0DMA Transfer Enable Register (DRI0TRMDEN)	8
H'FFBF C00C	DRI0I Transfer Control Register (DRI0TRMCNT)	DRI0 Special Mode Register (DRI0SPMOD)	DRI0 Data Acquisition Control Register (DRI0DCAPCNT)		8, 8, 16
H'FFBF C010	DRI0 Data Decimation Control Register (DRI0DSELCNT)	DRI0 Data Decimation Event Selection Register (DRI0DEVTCNT)	DRI0DIN Input Event Selection Register (DRI0DINSEL)	(Reserved)	8, 8, 8, -
H'FFBF C014	DRI0DD Input Enable Register (DRI0DDEN)				32
H'FFBF C018	DRI0 Data Acquisition Event Count Setting Register (DRI0DCAPNUM)				32
H'FFBF C01C	DRI0 Acquisition Event Counter (DRI0DCAPCT)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF C020	DRI0 Transfer Counter (DRI0TRMCT)				32
H'FFBF C024	DRI0 Address Reload Register 0 (DRI0ADR0RLD)				32
H'FFBF C028	DRI0 Address Counter 0 (DRI0ADR0CT)				32
H'FFBF C02C	DRI0 Address Reload Register 1 (DRI0ADR1RLD)				32
H'FFBF C030	DRI0 Address Counter 1 (DRI0ADR1CT)				32
H'FFBF C034	DRI0 Input Processing Control Register (DRI0DINCNT)		DRI0DEC0 Control Register (DRI0DEC0CNT)	(Reserved)	16, 8, -
H'FFBF C038	DRI0DEC0 Reload Register (DRI0DEC0RLD)		DRI0DEC0 Counter (DRI0DEC0CT)		16
H'FFBF C03C	DRI0DEC1 Control Register (DRI0DEC1CNT)	(Reserved)	DRI0DEC1 Reload Register (DRI0DEC1RLD)		8, -, 16
H'FFBF C040	DRI0DEC1 Counter (DRI0DEC1CT)		DRI0DEC2 Control Register (DRI0DEC2CNT)	(Reserved)	16, 8, -
H'FFBF C044	DRI0DEC2 Reload Register (DRI0DEC2RLD)		DRI0DEC2 Counter (DRI0DEC2CT)		16
H'FFBF C048	DRI0DEC3 Control Register (DRI0DEC3CNT)	(Reserved)	DRI0DEC3 Reload Register (DRI0DEC3RLD)		8, -, 16
H'FFBF C04C	DRI0DEC3 Counter (DRI0DEC3CT)		DRI0DEC4 Control Register (DRI0DEC4CNT)	(Reserved)	16, 8, -
H'FFBF C050	DRI0DEC4 Reload Register (DRI0DEC4RLD)		DRI0DEC4 Counter (DRI0DEC4CT)		16
H'FFBF C054	DRI0DEC5 Control Register (DRI0DEC5CNT)	(Reserved)	DRI0DEC5 Reload Register (DRI0DEC5RLD)		8, -, 16
H'FFBF C058	DRI0DEC5 Counter (DRI0DEC5CT)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFBF D000	DRI1DIN Interrupt Request Status Register (DRI1DINIST)	DRI1DIN Interrupt Request Enable Register (DRI1DINIEN)	(Reserved)	(Reserved)	8, 8, -, -
H'FFBF D004	DRI1DEC Interrupt Request Status Register (DRI1DECIST)	DRI1DEC Interrupt Request Enable Register (DRI1DECIEN)	(Reserved)	(Reserved)	8, 8, -, -
H'FFBF D008	DRI1 Transfer Interrupt Request Status Register (DRI1TRMIST)	DRI1 Transfer Interrupt Request Enable Register (DRI1TRMIEN)	(Reserved)	(Reserved)	8, 8, -, -
H'FFBF D00C	DRI1I Transfer Control Register (DRI1TRMCNT)	DRI1 Special Mode Register (DRI1SPMOD)	DRI1 Data Acquisition Control Register (DRI1DCAPCNT)		8, 8, 16
H'FFBF D010	DRI1 Data Decimation Control Register (DRI1DSELCNT)	DRI1 Data Decimation Event Selection Register (DRI1DEVTCNT)	DRI1DIN Input Event Selection Register (DRI1DINSEL)	(Reserved)	8, 8, 8, -
H'FFBF D014	DRI1DD Input Enable Register (DRI1DDEN)				32
H'FFBF D018	DRI1 Data Acquisition Event Count Setting Register (DRI1DCAPNUM)				32
H'FFBF D01C	DRI1 Acquisition Event Counter (DRI1DCAPCT)				32
H'FFBF D020	DRI1 Transfer Counter (DRI1TRMCT)				32
H'FFBF D024	DRI1 Address Reload Register 0 (DRI1ADR0RLD)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF D028	DRI1 Address Counter 0 (DRI1ADR0CT)				32
H'FFBF D02C	DRI1 Address Reload Register 1 (DRI1ADR1RLD)				32
H'FFBF D030	DRI1 Address Counter 1 (DRI1ADR1CT)				32
H'FFBF D034	DRI1 Input Processing Control Register (DRI1DINCNT)		DRI1DEC0 Control Register (DRI1DEC0CNT)	(Reserved)	16, 8, -
H'FFBF D038	DRI1DEC0 Reload Register (DRI1DEC0RLD)		DRI1DEC0 Counter (DRI1DEC0CT)		16
H'FFBF D03C	DRI1DEC1 Control Register (DRI1DEC1CNT)	(Reserved)	DRI1DEC1 Reload Register (DRI1DEC1RLD)		8, -, 16
H'FFBF D040	DRI1DEC1 Counter (DRI1DEC1CT)		DRI1DEC2 Control Register (DRI1DEC2CNT)	(Reserved)	16, 8, -
H'FFBF D044	DRI1DEC2 Reload Register (DRI1DEC2RLD)		DRI1DEC2 Counter (DRI1DEC2CT)		16
H'FFBF D048	DRI1DEC3 Control Register (DRI1DEC3CNT)	(Reserved)	DRI1DEC3 Reload Register (DRI1DEC3RLD)		8, -, 16
H'FFBF D04C	DRI1DEC3 Counter (DRI1DEC3CT)		DRI1DEC4 Control Register (DRI1DEC4CNT)	(Reserved)	16, 8, -
H'FFBF D050	DRI1DEC4 Reload Register (DRI1DEC4RLD)		DRI1DEC4 Counter (DRI1DEC4CT)		16
H'FFBF D054	DRI1DEC5 Control Register (DRI1DEC5CNT)	(Reserved)	DRI1DEC5 Reload Register (DRI1DEC5RLD)		8, -, 16
H'FFBF D058	DRI1DEC5 Counter (DRI1DEC5CT)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFBF E000	DRI2DIN Interrupt Request Status Register (DRI2DINIST)	DRI2DIN Interrupt Request Enable Register (DRI2DINIEN)	(Reserved)	(Reserved)	8, 8, -, -
H'FFBF E004	DRI2DEC Interrupt Request Status Register (DRI2DECIST)	DRI2DEC Interrupt Request Enable Register (DRI2DECIEN)	(Reserved)	(Reserved)	8, 8, -, -
H'FFBF E008	DRI2 Transfer Interrupt Request Status Register (DRI2TRMIST)	DR2 Tranfer Interrupt Request Enable Register (DRI2TRMIEN)	(Reserved)	(Reserved)	8, 8, -, -
H'FFBF E00C	DRI2 Transfer Control Register (DRI2TRMCNT)	DRI2 Special Mode Register (DRI2SPMOD)	DRI2 Data Acquisition Control Register (DRI2DCAPCNT)		8, 8, 16
H'FFBF E010	DRI2 Data Decimation Control Register (DRI2DSELCNT)	DRI2 Data Decimation Event Selection Register (DRI2DEVTCNT)	DRI2DIN Input Event Selection Register (DRI2DINSEL)	(Reserved)	8, 8, 8, -
H'FFBF E014	DRI2DD Input Enable Register (DRI2DDEN)				32
H'FFBF E018	DRI2 Data Acquisition Event Count Setting Register (DRI2DCAPNUM)				32
H'FFBF E01C	DRI2 Acquisition Event Counter (DRI2DCAPCT)				32
H'FFBF E020	DRI2 Transfer Counter (DRI2TRMCT)				32
H'FFBF E024	DRI2 Address Reload Register 0 (DRI2ADR0RLD)				32
H'FFBF E028	DRI2 Address Counter 0 (DRI2ADR0CT)				32
H'FFBF E02C	DRI2 Address Reload Register 1 (DRI2ADR1RLD)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF E030	DRI2 Address Counter 1 (DRI2ADR1CT)				32
H'FFBF E034	DRI2 Input Processing Control Register (DRI2DINCNT)		DRI2DEC0 Control Register (DRI2DEC0CNT)	(Reserved)	16, 8, -
H'FFBF E038	DRI2DEC0 Reload Register (DRI2DEC0RLD)		DRI2DEC0 Counter (DRI2DEC0CT)		16
H'FFBF E03C	DRI2DEC1 Control Register (DRI2DEC1CNT)	(Reserved)	DRI2DEC1 Reload Register (DRI2DEC1RLD)		8, -, 16
H'FFBF E040	DRI2DEC1 Counter (DRI2DEC1CT)		DRI2DEC2 Control Register (DRI2DEC2CNT)	(Reserved)	16, 8, -
H'FFBF E044	DRI2DEC2 Reload Register (DRI2DEC2RLD)		DRI2DEC2 Counter (DRI2DEC2CT)		16
H'FFBF E048	DRI2DEC3 Control Register (DRI2DEC3CNT)	(Reserved)	DRI2DEC3 Reload Register (DRI2DEC3RLD)		8, -, 16
H'FFBF E04C	DRI2DEC3 Counter (DRI2DEC3CT)		DRI2DEC4 Control Register (DRI2DEC4CNT)	(Reserved)	16, 8, -
H'FFBF E050	DRI2DEC4 Reload Register (DRI2DEC4RLD)		DRI2DEC4 Counter (DRI2DEC4CT)		16
H'FFBF E054	DRI2DEC5 Control Register (DRI2DEC5CNT)	(Reserved)	DRI2DEC5 Reload Register (DRI2DEC5RLD)		8, -, 16
H'FFBF E058	DRI2DEC5 Counter (DRI2DEC5CT)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFBF F004	FlexRay Operation Control Register (FXROC)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFBF F00C	FlexRay Timer Interrupt Request Status Register (FXRTISR)	FlexRay Timer Interrupt Enable Register (FXRTIER)	(Reserved)	(Reserved)	8, 8, -, -
:	(Reserved)				-
H'FFBF F01C	(Reserved)	(Reserved)	(Reserved)	FlexRay Lock Register (FRLCK)	-, -, -, 8
H'FFBF F020	FlexRay Error Interrupt Register (FREIR)				32
H'FFBF F024	FlexRay Status Interrupt Register (FRSIR)				32
H'FFBF F028	FlexRay Error Interrupt Line Select Register (FREILS)				32
H'FFBF F02C	FlexRay Status Interrupt Line Select Register (FRSILS)				32
H'FFBF F030	FlexRay Error Interrupt Enable Set Register (FREIES)				32
H'FFBF F034	FlexRay Error Interrupt Enable Reset Register (FREIER)				32
H'FFBF F038	FlexRay Status Interrupt Enable Set Register (FRSIES)				32
H'FFBF F03C	FlexRay Status Interrupt Enable Reset Register (FRSIER)				32
H'FFBF F040	(Reserved)	(Reserved)	(Reserved)	FlexRay Interrupt Line Enable Register (FRILE)	-, -, -, 8
H'FFBF F044	FlexRay Timer 0 Configuration Register (FRT0C)				32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFBF F048	FlexRay Timer 1 Configuration Register (FRT1C)								32
H'FFBF F04C	FlexRay Stop Watch Register 1 (FRSTPW1)								32
H'FFBF F050	FlexRay Stop Watch Register 2 (FRSTPW2)								32
:	(Reserved)								-
H'FFBF F080	FlexRay SUC Configuration Register 1 (FRSUCC1)								32
H'FFBF F084	FlexRay SUC Configuration Register 2 (FRSUCC2)								32
H'FFBF F088	(Reserved)		(Reserved)		(Reserved)		FlexRay SUC Configuration Register 3 (FRSUCC3)		-, -, -, 8
H'FFBF F08C	(Reserved)		(Reserved)		(Reserved)		FlexRay NEM Configuration Register (FRNEMC)		-, -, -, 8
H'FFBF F090	FlexRay PRT Configuration Register 1 (FRPRTC1)								32
H'FFBF F094	FlexRay PRT Configuration Register 2 (FRPRTC2)								32
H'FFBF F098	FlexRay MHD Configuration Register (FRMHDC)								32
:	(Reserved)								-
H'FFBF F0A0	FlexRay GTU Configuration Register 1 (FRGTUC1)								32
H'FFBF F0A4	FlexRay GTU Configuration Register 2 (FRGTUC2)								32
H'FFBF F0A8	FlexRay GTU Configuration Register 3 (FRGTUC3)								32
H'FFBF F0AC	FlexRay GTU Configuration Register 4 (FRGTUC4)								32
H'FFBF F0B0	FlexRay GTU Configuration Register 5 (FRGTUC5)								32
H'FFBF F0B4	FlexRay GTU Configuration Register 6 (FRGTUC6)								32
H'FFBF F0B8	FlexRay GTU Configuration Register 7 (FRGTUC7)								32
H'FFBF F0BC	FlexRay GTU Configuration Register 8 (FRGTUC8)								32
H'FFBF F0C0	FlexRay GTU Configuration Register 9 (FRGTUC9)								32
H'FFBF F0C4	FlexRay GTU Configuration Register 10 (FRGTUC10)								32
H'FFBF F0C8	FlexRay GTU Configuration Register 11 (FRGTUC11)								32
:	(Reserved)								-
H'FFBF F100	FlexRay CC Status Vector Register (FRCCSV)								32
H'FFBF F104	(Reserved)		(Reserved)		FlexRay CC Error Vector Register (FRCCEV)				-, -, 16
:	(Reserved)								-
H'FFBF F110	FlexRay Slot Counter Value Register (FRSCV)								32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF F114	FlexRay Macrotick and Cycle Counter Value Register (FRMTCCV)				32
H'FFBF F118	(Reserved)	(Reserved)	FlexRay Rate Correction Value Register (FRRCV)		-, -, 16
H'FFBF F11C	FlexRay Offset Correction Value (FROCV)				32
H'FFBF F120	FlexRay Sync Frame Status Register (FRSFS)				32
H'FFBF F124	(Reserved)	(Reserved)	FlexRay Symbol Window and NIT Status Register (FRSWNIT)		-, -, 16
H'FFBF F128	(Reserved)	(Reserved)	FlexRay Aggregated Channel Status Register (FRACS)		-, -, 16
:	(Reserved)				-
H'FFBF F130	(Reserved)	(Reserved)	FlexRay Even Sync ID 1 Register (FRESID1)		-, -, 16
H'FFBF F134	(Reserved)	(Reserved)	FlexRay Even Sync ID 2 Register (FRESID2)		-, -, 16
H'FFBF F138	(Reserved)	(Reserved)	FlexRay Even Sync ID 3 Register (FRESID3)		-, -, 16
H'FFBF F13C	(Reserved)	(Reserved)	FlexRay Even Sync ID 4 Register (FRESID4)		-, -, 16
H'FFBF F140	(Reserved)	(Reserved)	FlexRay Even Sync ID 5 Register (FRESID5)		-, -, 16
H'FFBF F144	(Reserved)	(Reserved)	FlexRay Even Sync ID 6 Register (FRESID6)		-, -, 16
H'FFBF F148	(Reserved)	(Reserved)	FlexRay Even Sync ID 7 Register (FRESID7)		-, -, 16
H'FFBF F14C	(Reserved)	(Reserved)	FlexRay Even Sync ID 8 Register (FRESID8)		-, -, 16
H'FFBF F150	(Reserved)	(Reserved)	FlexRay Even Sync ID 9 Register (FRESID9)		-, -, 16
H'FFBF F154	(Reserved)	(Reserved)	FlexRay Even Sync ID 10 Register (FRESID10)		-, -, 16
H'FFBF F158	(Reserved)	(Reserved)	FlexRay Even Sync ID 11 Register (FRESID11)		-, -, 16
H'FFBF F15C	(Reserved)	(Reserved)	FlexRay Even Sync ID 12 Register (FRESID12)		-, -, 16
H'FFBF F160	(Reserved)	(Reserved)	FlexRay Even Sync ID 13 Register (FRESID13)		-, -, 16
H'FFBF F164	(Reserved)	(Reserved)	FlexRay Even Sync ID 14 Register (FRESID14)		-, -, 16
H'FFBF F168	(Reserved)	(Reserved)	FlexRay Even Sync ID 15 Register (FRESID15)		-, -, 16
:	(Reserved)				-
H'FFBF F170	(Reserved)	(Reserved)	FlexRay Odd Sync ID 1 Register (FROSID1)		-, -, 16
H'FFBF F174	(Reserved)	(Reserved)	FlexRay Odd Sync ID 2 Register (FROSID2)		-, -, 16
H'FFBF F178	(Reserved)	(Reserved)	FlexRay Odd Sync ID 3 Register (FROSID3)		-, -, 16
H'FFBF F17C	(Reserved)	(Reserved)	FlexRay Odd Sync ID 4 Register (FROSID4)		-, -, 16

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF F180	(Reserved)	(Reserved)	FlexRay Odd Sync ID 5 Register (FROSID5)		-, -, 16
H'FFBF F184	(Reserved)	(Reserved)	FlexRay Odd Sync ID 6 Register (FROSID6)		-, -, 16
H'FFBF F188	(Reserved)	(Reserved)	FlexRay Odd Sync ID 7 Register (FROSID7)		-, -, 16
H'FFBF F18C	(Reserved)	(Reserved)	FlexRay Odd Sync ID 8 Register (FROSID8)		-, -, 16
H'FFBF F190	(Reserved)	(Reserved)	FlexRay Odd Sync ID 9 Register (FROSID9)		-, -, 16
H'FFBF F194	(Reserved)	(Reserved)	FlexRay Odd Sync ID 10 Register (FROSID10)		-, -, 16
H'FFBF F198	(Reserved)	(Reserved)	FlexRay Odd Sync ID 11 Register (FROSID11)		-, -, 16
H'FFBF F19C	(Reserved)	(Reserved)	FlexRay Odd Sync ID 12 Register (FROSID12)		-, -, 16
H'FFBF F1A0	(Reserved)	(Reserved)	FlexRay Odd Sync ID 13 Register (FROSID13)		-, -, 16
H'FFBF F1A4	(Reserved)	(Reserved)	FlexRay Odd Sync ID 14 Register (FROSID14)		-, -, 16
H'FFBF F1A8	(Reserved)	(Reserved)	FlexRay Odd Sync ID 15 Register (FROSID15)		-, -, 16
:	(Reserved)				-
H'FFBF F1B0	FlexRay Network Management Vector 1 Register (FRNMV1)				32
H'FFBF F1B4	FlexRay Network Management Vector 2 Register (FRNMV2)				32
H'FFBF F1B8	FlexRay Network Management Vector 3 Register (FRNMV3)				32
:	(Reserved)				-
H'FFBF F300	FlexRay Message RAM Configuration Register (FRMRC)				32
H'FFBF F304	FlexRay FIFO Rejection Filter Register (FRFRF)				32
H'FFBF F308	(Reserved)	(Reserved)	FlexRay FIFO Rejection Filter Mask Register (FRFRFM)		-, -, 16
H'FFBF F30C	(Reserved)	(Reserved)	(Reserved)	FlexRay FIFO Critical Level Register (FRFCL)	-, -, -, 8
H'FFBF F310	FlexRay Message Handler Status Register (FRMHDS)				32
H'FFBF F314	FlexRay Last Dynamic Transmit Slot Register (FRLDTS)				32
H'FFBF F318	(Reserved)	(Reserved)	FlexRay FIFO Status Register (FRFSR)		-, -, 16
H'FFBF F31C	(Reserved)	(Reserved)	FlexRay Message Handler Constraints Flags Register (FRMHDF)		-, -, 16
H'FFBF F320	FlexRay Transmission Request Register 1 (FRTXRQ1)				32
H'FFBF F324	FlexRay Transmission Request Register 2 (FRTXRQ2)				32
H'FFBF F328	FlexRay Transmission Request Register 3 (FRTXRQ3)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF F32C	FlexRay Transmission Request Register 4 (FRTXRQ4)				32
H'FFBF F330	FlexRay New Data Register 1 (FRNDAT1)				32
H'FFBF F334	FlexRay New Data Register 2 (FRNDAT2)				32
H'FFBF F338	FlexRay New Data Register 3 (FRNDAT3)				32
H'FFBF F33C	FlexRay New Data Register 4 (FRNDAT4)				32
H'FFBF F340	FlexRay Message Buffer Status Changed Register 1 (FRMBSC1)				32
H'FFBF F344	FlexRay Message Buffer Status Changed Register 2 (FRMBSC2)				32
H'FFBF F348	FlexRay Message Buffer Status Changed Register 3 (FRMBSC3)				32
H'FFBF F34C	FlexRay Message Buffer Status Changed Register 4 (FRMBSC4)				32
:	(Reserved)				-
H'FFBF F400	FlexRay Write Data Section 1 Register (FRWRDS1)				32
H'FFBF F404	FlexRay Write Data Section 2 Register (FRWRDS2)				32
H'FFBF F408	FlexRay Write Data Section 3 Register (FRWRDS3)				32
H'FFBF F40C	FlexRay Write Data Section 4 Register (FRWRDS4)				32
H'FFBF F410	FlexRay Write Data Section 5 Register (FRWRDS5)				32
H'FFBF F414	FlexRay Write Data Section 6 Register (FRWRDS6)				32
H'FFBF F418	FlexRay Write Data Section 7 Register (FRWRDS7)				32
H'FFBF F41C	FlexRay Write Data Section 8 Register (FRWRDS8)				32
H'FFBF F420	FlexRay Write Data Section 9 Register (FRWRDS9)				32
H'FFBF F424	FlexRay Write Data Section 10 Register (FRWRDS10)				32
H'FFBF F428	FlexRay Write Data Section 11 Register (FRWRDS11)				32
H'FFBF F42C	FlexRay Write Data Section 12 Register (FRWRDS12)				32
H'FFBF F430	FlexRay Write Data Section 13 Register (FRWRDS13)				32
H'FFBF F434	FlexRay Write Data Section 14 Register (FRWRDS14)				32
H'FFBF F438	FlexRay Write Data Section 15 Register (FRWRDS15)				32
H'FFBF F43C	FlexRay Write Data Section 16 Register (FRWRDS16)				32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFBF F440	FlexRay Write Data Section 17 Register (FRWRDS17)								32
H'FFBF F444	FlexRay Write Data Section 18 Register (FRWRDS18)								32
H'FFBF F448	FlexRay Write Data Section 19 Register (FRWRDS19)								32
H'FFBF F44C	FlexRay Write Data Section 20 Register (FRWRDS20)								32
H'FFBF F450	FlexRay Write Data Section 21 Register (FRWRDS21)								32
H'FFBF F454	FlexRay Write Data Section 22 Register (FRWRDS22)								32
H'FFBF F458	FlexRay Write Data Section 23 Register (FRWRDS23)								32
H'FFBF F45C	FlexRay Write Data Section 24 Register (FRWRDS24)								32
H'FFBF F460	FlexRay Write Data Section 25 Register (FRWRDS25)								32
H'FFBF F464	FlexRay Write Data Section 26 Register (FRWRDS26)								32
H'FFBF F468	FlexRay Write Data Section 27 Register (FRWRDS27)								32
H'FFBF F46C	FlexRay Write Data Section 28 Register (FRWRDS28)								32
H'FFBF F470	FlexRay Write Data Section 29 Register (FRWRDS29)								32
H'FFBF F474	FlexRay Write Data Section 30 Register (FRWRDS30)								32
H'FFBF F478	FlexRay Write Data Section 31 Register (FRWRDS31)								32
H'FFBF F47C	FlexRay Write Data Section 32 Register (FRWRDS32)								32
H'FFBF F480	FlexRay Write Data Section 33 Register (FRWRDS33)								32
H'FFBF F484	FlexRay Write Data Section 34 Register (FRWRDS34)								32
H'FFBF F488	FlexRay Write Data Section 35 Register (FRWRDS35)								32
H'FFBF F48C	FlexRay Write Data Section 36 Register (FRWRDS36)								32
H'FFBF F490	FlexRay Write Data Section 37 Register (FRWRDS37)								32
H'FFBF F494	FlexRay Write Data Section 38 Register (FRWRDS38)								32
H'FFBF F498	FlexRay Write Data Section 39 Register (FRWRDS39)								32
H'FFBF F49C	FlexRay Write Data Section 40 Register (FRWRDS40)								32
H'FFBF F4A0	FlexRay Write Data Section 41 Register (FRWRDS41)								32
H'FFBF F4A4	FlexRay Write Data Section 42 Register (FRWRDS42)								32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF F4A8	FlexRay Write Data Section 43 Register (FRWRDS43)				32
H'FFBF F4AC	FlexRay Write Data Section 44 Register (FRWRDS44)				32
H'FFBF F4B0	FlexRay Write Data Section 45 Register (FRWRDS45)				32
H'FFBF F4B4	FlexRay Write Data Section 46 Register (FRWRDS46)				32
H'FFBF F4B8	FlexRay Write Data Section 47 Register (FRWRDS47)				32
H'FFBF F4BC	FlexRay Write Data Section 48 Register (FRWRDS48)				32
H'FFBF F4C0	FlexRay Write Data Section 49 Register (FRWRDS49)				32
H'FFBF F4C4	FlexRay Write Data Section 50 Register (FRWRDS50)				32
H'FFBF F4C8	FlexRay Write Data Section 51 Register (FRWRDS51)				32
H'FFBF F4CC	FlexRay Write Data Section 52 Register (FRWRDS52)				32
H'FFBF F4D0	FlexRay Write Data Section 53 Register (FRWRDS53)				32
H'FFBF F4D4	FlexRay Write Data Section 54 Register (FRWRDS54)				32
H'FFBF F4D8	FlexRay Write Data Section 55 Register (FRWRDS55)				32
H'FFBF F4DC	FlexRay Write Data Section 56 Register (FRWRDS56)				32
H'FFBF F4E0	FlexRay Write Data Section 57 Register (FRWRDS57)				32
H'FFBF F4E4	FlexRay Write Data Section 58 Register (FRWRDS58)				32
H'FFBF F4E8	FlexRay Write Data Section 59 Register (FRWRDS59)				32
H'FFBF F4EC	FlexRay Write Data Section 60 Register (FRWRDS60)				32
H'FFBF F4F0	FlexRay Write Data Section 61 Register (FRWRDS61)				32
H'FFBF F4F4	FlexRay Write Data Section 62 Register (FRWRDS62)				32
H'FFBF F4F8	FlexRay Write Data Section 63 Register (FRWRDS63)				32
H'FFBF F4FC	FlexRay Write Data Section 64 Register (FRWRDS64)				32
H'FFBF F500	FlexRay Write Header Section Register 1 (FRWRHS1)				32
H'FFBF F504	FlexRay Write Header Section Register 2 (FRWRHS2)				32
H'FFBF F508	(Reserved)	(Reserved)	FlexRay Write Header Section Register 3 (FRWRHS3)		-, -, 16
H'FFBF F50C	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF F510	FlexRay Input Buffer Command Mask Register (FRIBCM)				32
H'FFBF F514	FlexRay Input Buffer Command Request Register (FRIBCR)				32
:	(Reserved)				-
H'FFBF F600	FlexRay Read Data Section Register 1 (FRRDDS1)				32
H'FFBF F604	FlexRay Read Data Section Register 2 (FRRDDS2)				32
H'FFBF F608	FlexRay Read Data Section Register 3 (FRRDDS3)				32
H'FFBF F60C	FlexRay Read Data Section Register 4 (FRRDDS4)				32
H'FFBF F610	FlexRay Read Data Section Register 5 (FRRDDS5)				32
H'FFBF F614	FlexRay Read Data Section Register 6 (FRRDDS6)				32
H'FFBF F618	FlexRay Read Data Section Register 7 (FRRDDS7)				32
H'FFBF F61C	FlexRay Read Data Section Register 8 (FRRDDS8)				32
H'FFBF F620	FlexRay Read Data Section Register 9 (FRRDDS9)				32
H'FFBF F624	FlexRay Read Data Section Register 10 (FRRDDS10)				32
H'FFBF F628	FlexRay Read Data Section Register 11 (FRRDDS11)				32
H'FFBF F62C	FlexRay Read Data Section Register 12 (FRRDDS12)				32
H'FFBF F630	FlexRay Read Data Section Register 13 (FRRDDS13)				32
H'FFBF F634	FlexRay Read Data Section Register 14 (FRRDDS14)				32
H'FFBF F638	FlexRay Read Data Section Register 15 (FRRDDS15)				32
H'FFBF F63C	FlexRay Read Data Section Register 16 (FRRDDS16)				32
H'FFBF F640	FlexRay Read Data Section Register 17 (FRRDDS17)				32
H'FFBF F644	FlexRay Read Data Section Register 18 (FRRDDS18)				32
H'FFBF F648	FlexRay Read Data Section Register 19 (FRRDDS19)				32
H'FFBF F64C	FlexRay Read Data Section Register 20 (FRRDDS20)				32
H'FFBF F650	FlexRay Read Data Section Register 21 (FRRDDS21)				32
H'FFBF F654	FlexRay Read Data Section Register 22 (FRRDDS22)				32
H'FFBF F658	FlexRay Read Data Section Register 23 (FRRDDS23)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFBF F65C	FlexRay Read Data Section Register 24 (FRRDDS24)				32
H'FFBF F660	FlexRay Read Data Section Register 25 (FRRDDS25)				32
H'FFBF F664	FlexRay Read Data Section Register 26 (FRRDDS26)				32
H'FFBF F668	FlexRay Read Data Section Register 27 (FRRDDS27)				32
H'FFBF F66C	FlexRay Read Data Section Register 28 (FRRDDS28)				32
H'FFBF F670	FlexRay Read Data Section Register 29 (FRRDDS29)				32
H'FFBF F674	FlexRay Read Data Section Register 30 (FRRDDS30)				32
H'FFBF F678	FlexRay Read Data Section Register 31 (FRRDDS31)				32
H'FFBF F67C	FlexRay Read Data Section Register 32 (FRRDDS32)				32
H'FFBF F680	FlexRay Read Data Section Register 33 (FRRDDS33)				32
H'FFBF F684	FlexRay Read Data Section Register 34 (FRRDDS34)				32
H'FFBF F688	FlexRay Read Data Section Register 35 (FRRDDS35)				32
H'FFBF F68C	FlexRay Read Data Section Register 36 (FRRDDS36)				32
H'FFBF F690	FlexRay Read Data Section Register 37 (FRRDDS37)				32
H'FFBF F694	FlexRay Read Data Section Register 38 (FRRDDS38)				32
H'FFBF F698	FlexRay Read Data Section Register 39 (FRRDDS39)				32
H'FFBF F69C	FlexRay Read Data Section Register 40 (FRRDDS40)				32
H'FFBF F6A0	FlexRay Read Data Section Register 41 (FRRDDS41)				32
H'FFBF F6A4	FlexRay Read Data Section Register 42 (FRRDDS42)				32
H'FFBF F6A8	FlexRay Read Data Section Register 43 (FRRDDS43)				32
H'FFBF F6AC	FlexRay Read Data Section Register 44 (FRRDDS44)				32
H'FFBF F6B0	FlexRay Read Data Section Register 45 (FRRDDS45)				32
H'FFBF F6B4	FlexRay Read Data Section Register 46 (FRRDDS46)				32
H'FFBF F6B8	FlexRay Read Data Section Register 47 (FRRDDS47)				32
H'FFBF F6BC	FlexRay Read Data Section Register 48 (FRRDDS48)				32
H'FFBF F6C0	FlexRay Read Data Section Register 49 (FRRDDS49)				32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFBF F6C4	FlexRay Read Data Section Register 50 (FRRDDS50)								32
H'FFBF F6C8	FlexRay Read Data Section Register 51 (FRRDDS51)								32
H'FFBF F6CC	FlexRay Read Data Section Register 52 (FRRDDS52)								32
H'FFBF F6D0	FlexRay Read Data Section Register 53 (FRRDDS53)								32
H'FFBF F6D4	FlexRay Read Data Section Register 54 (FRRDDS54)								32
H'FFBF F6D8	FlexRay Read Data Section Register 55 (FRRDDS55)								32
H'FFBF F6DC	FlexRay Read Data Section Register 56 (FRRDDS56)								32
H'FFBF F6E0	FlexRay Read Data Section Register 57 (FRRDDS57)								32
H'FFBF F6E4	FlexRay Read Data Section Register 58 (FRRDDS58)								32
H'FFBF F6E8	FlexRay Read Data Section Register 59 (FRRDDS59)								32
H'FFBF F6EC	FlexRay Read Data Section Register 60 (FRRDDS60)								32
H'FFBF F6F0	FlexRay Read Data Section Register 61 (FRRDDS61)								32
H'FFBF F6F4	FlexRay Read Data Section Register 62 (FRRDDS62)								32
H'FFBF F6F8	FlexRay Read Data Section Register 63 (FRRDDS63)								32
H'FFBF F6FC	FlexRay Read Data Section Register 64 (FRRDDS64)								32
H'FFBF F700	FlexRay Read Header Section Register 1 (FRRDHS1)								32
H'FFBF F704	FlexRay Read Header Section Register 2 (FRRDHS2)								32
H'FFBF F708	FlexRay Read Header Section Register 3 (FRRDHS3)								32
H'FFBF F70C	FlexRay Message Buffer Status Register (FRMBS)								32
H'FFBF F710	FlexRay Output Buffer Command Mask Register (FROBCM)								32
H'FFBF F714	FlexRay Output Buffer Command Request Register (FROBCR)								32
:	(Reserved)								-
H'FFFE E000	I^{2}C Bus Control Register 1 (ICCR1)		I^{2}C Bus Control Register 2 (ICCR2)		I^{2}C Bus Mode Register (ICMR)		I^{2}C Bus Interrupt Enable Register (ICIER)		8
H'FFFE E004	I^{2}C Bus Status Register (ICSR)		I^{2}C Bus Slave Address Register (ICSAR)		I^{2}C Bus Transmit Data Register (ICDRT)		I^{2}C Bus Receive Data Register (ICDRR)		8
H'FFFE E008	I^{2}C Bus NF2CYC Register (ICNF2CYC)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
:	(Reserved)								-
H'FFFE F000	DRO Interrupt Request Status Register (DROIST)		DRO Interrupt Request Enable Register (DROIEN)		(Reserved)		(Reserved)		8/16, 8/16, -, -

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFE F004	DRO Operating Mode Register (DROMOD)		DRO Output Control Register (DROCNT)	(Reserved)	16, 8, -
H'FFFE F008	DRO Output Data Counter (DRODCT)				32
H'FFFE F00C	DRO Address Counter (DROADRCT)				32
:	(Reserved)				-
H'FFFF 1000	Watchdog Timer Stop Time Register (WDTST)				32
H'FFFF 1004	Watchdog Timer Control/Status Register (WDTCSR)				32
H'FFFF 1008	Watchdog Timer Base Stop Time Register (WDTBST)				32
:	(Reserved)				-
H'FFFF 1010	Watchdog Timer Counter (WDTCNT)				32
:	(Reserved)				-
H'FFFF 1018	Watchdog Timer Base Counter (WDTBCNT)				32
:	(Reserved)				-
H'FFFF 2000	(Reserved)	Mode Control Register (MDCR)	(Reserved)	(Reserved)	-, 8, -, -
:	(Reserved)				-
H'FFFF 2800	Module Stop Register 0 (MSTPCR0)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 2810	Oscillator Status Register (OSCSR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF 2814	Oscillator Control Register (OSCCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF 3000	PSEL Event Selection Register (PSLCTRL)	PSEL Output Clock Divisor Setting Register (PSLPRE)	PSEL Channel Count Selection Register (PSLSEL)	PSEL Output Polarity Control Register (PSLPOL)	8/16/32
H'FFFF 3004	PSEL Trigger Register (PSLTRIG)	(Reserved)	PSEL Status Register (PSLSTATUS)	(Reserved)	8 -, 8, -
H'FFFF 3008	PSEL Clock A Delay Register (PSLDLYA)		PSEL Clock B Delay Register (PSLDLYB)		16/32
H'FFFF 300C	PSEL Clear Delay Period Register (PSLCLRD)		PSEL Clear Control Register (PSLCLRC)		16/32
H'FFFF 3010	PSEL Data Buffer 0/1 Register (PSLDT0001)	PSEL Data Buffer 2/3 Register (PSLDT0203)	PSEL Data Buffer 4/5 Register (PSLDT0405)	PSEL Data Buffer 6/7 Register (PSLDT0607)	8
H'FFFF 3014	PSEL Data Buffer 8/9 Register (PSLDT0809)	PSEL Data Buffer 10/11 Register (PSLDT1011)	PSEL Data Buffer 12/13 Register (PSLDT1213)	PSEL Data Buffer 14/15 Register (PSLDT1415)	8
H'FFFF 3018	PSEL Data Initial Value Register (PSLINIT)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF 3400	PDAC Forced Stop Register (PDISTOP)	PDAC Basic Resolution Setting Register (PDIPRE)	PDAC Control Period Event Selection Register (PDICPT)	(Reserved)	8/16, 8/16, 8, -
H'FFFF 3404	PDAC Status Register (PDISTATUS)	PDAC Status Register A (PDISTAA)	PDAC Status Register B (PDISTAB)	PDAC Status Register C (PDISTAC)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 3408	PDAC Interrupt Control Register (PDIINT)	(Reserved)	PDAC Write Signal Period Adjustment Register (PDIWRC)		8, -, 8/16
H'FFFF 340C	PDAC Wait Time Control Register (PDIWTEN)		(Reserved)	(Reserved)	16, -, -
H'FFFF 3410	PDAC Output Event Selection A Register (PDISELA)		PDAC Output Event Selection B Register (PDISELB)		16/32
H'FFFF 3414	PDAC Output Event Selection C Register (PDISELC)		PDAC Output Event Selection D Register (PDISELD)		16/32
H'FFFF 3418	PDAC Output Event Selection E Register (PDISELE)		PDAC Output Event Selection F Register (PDISELF)		16/32
H'FFFF 341C	PDAC Output Event Selection G Register (PDISELG)		PDAC Output Event Selection H Register (PDISELH)		16/32
:	(Reserved)				-
H'FFFF 3430	PDAC Modulation A Rise Step Count Register (PDIRSA)	PDAC Modulation A Fall Step Count Register (PDIFSA)	(Reserved)	(Reserved)	8/16, 8/16, -, -
H'FFFF 3434	PDAC Modulation A Rise Initial Value Register (PDIRIA)		PDAC Modulation A Fall Initial Value Register (PDIFIA)		16/32
H'FFFF 3438	PDAC Modulation A Rise Delta Value Register (PDIRDA)		PDAC Modulation A Fall Delta Value Register (PDIFDA)		16/32
H'FFFF 343C	PDAC Modulation A Output Start Wait Time Register (PDIWT0A)		PDAC Modulation A Post-Rise Wait Time Register (PDIWT1A)		16/32
H'FFFF 3440	PDAC Modulation A Post-Fall Wait Time Register r (PDIWT2A)		PDAC Modulation A Repeat Count Register (PDIREPA)	(Reserved)	16, 8/, -
:	(Reserved)				-
H'FFFF 3450	PDAC Modulation B Rise Step Count Register (PDIRSB)	PDAC Modulation B Fall Step Count Register (PDIFSB)	(Reserved)	(Reserved)	8/16, 8/16, -, -
H'FFFF 3454	PDAC Modulation B Rise Initial Value Register (PDIRIB)		PDAC Modulation B Fall Initial Value Register (PDIFIB)		16/32
H'FFFF 3458	PDAC Modulation B Rise Delta Value Register (PDIRDB)		PDAC Modulation B Fall Delta Value Register (PDIFDB)		16/32
H'FFFF 345C	PDAC Modulation B Output Start Wait Time Register (PDIWT0B)		PDAC Modulation B Post-Rise Wait Time Register (PDIWT1B)		16/32
H'FFFF 3460	PDAC Modulation B Post-Fall Wait Time Register (PDIWT2B)		PDAC Modulation B Repeat Count Register (PDIREPB)	(Reserved)	16, 8, -
:	(Reserved)				-
H'FFFF 3470	PDAC Modulation C Rise Step Count Register (PDIRSC)		PDAC Modulation C Fall Step Count Register (PDIFSC)		16/32
H'FFFF 3474	PDAC Modulation C Rise Initial Value Register (PDIRIC)		PDAC Modulation C Fall Initial Value Register (PDIFIC)		16/32
H'FFFF 3478	PDAC Modulation C Rise Delta Value Register (PDIRDC)		PDAC Modulation C Fall Delta Value Register (PDIFDC)		16/32
H'FFFF 347C	PDAC Modulation C Output Start Wait Time Register (PDIWT0C)		PDAC Modulation C Post-Rise Wait Time Register (PDIWT1C)		16/32
H'FFFF 3480	PDAC Modulation C Post-Fall Wait Time Register (PDIWT2C)		PDAC Modulation C Repeat Count Register (PDIREPC)	(Reserved)	16, 8, -
:	(Reserved)				-
H'FFFF 3800	PDAC Modulation A Rise Output Time Register 1 (PDIRTA1)	PDAC Modulation A Rise Output Time Register 2 (PDIRTA2)	PDAC Modulation A Rise Output Time Register 3 (PDIRTA3)	PDAC Modulation A Rise Output Time Register 4 (PDIRTA4)	8/16/32
H'FFFF 3804	PDAC Modulation A Rise Output Time Register 5 (PDIRTA5)	PDAC Modulation A Rise Output Time Register 6 (PDIRTA6)	PDAC Modulation A Rise Output Time Register 7 (PDIRTA7)	PDAC Modulation A Rise Output Time Register 8 (PDIRTA8)	8/16/32
H'FFFF 3808	PDAC Modulation A Rise Output Time Register 9 (PDIRTA9)	PDAC Modulation A Rise Output Time Register 10 (PDIRTA10)	PDAC Modulation A Rise Output Time Register 11 (PDIRTA11)	PDAC Modulation A Rise Output Time Register 12 (PDIRTA12)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 380C	PDAC Modulation A Rise Output Time Register 13 (PDIRTA13)	PDAC Modulation A Rise Output Time Register 14 (PDIRTA14)	PDAC Modulation A Rise Output Time Register 15 (PDIRTA15)	PDAC Modulation A Rise Output Time Register 16 (PDIRTA16)	8/16/32
H'FFFF 3810	PDAC Modulation A Rise Output Time Register 17 (PDIRTA17)	PDAC Modulation A Rise Output Time Register 18 (PDIRTA18)	PDAC Modulation A Rise Output Time Register 19 (PDIRTA19)	PDAC Modulation A Rise Output Time Register 20 (PDIRTA20)	8/16/32
H'FFFF 3814	PDAC Modulation A Rise Output Time Register 21 (PDIRTA21)	PDAC Modulation A Rise Output Time Register 22 (PDIRTA22)	PDAC Modulation A Rise Output Time Register 23 (PDIRTA23)	PDAC Modulation A Rise Output Time Register 24 (PDIRTA24)	8/16/32
H'FFFF 3818	PDAC Modulation A Rise Output Time Register 25 (PDIRTA25)	PDAC Modulation A Rise Output Time Register 26 (PDIRTA26)	PDAC Modulation A Rise Output Time Register 27 (PDIRTA27)	PDAC Modulation A Rise Output Time Register 28 (PDIRTA28)	8/16/32
H'FFFF 381C	PDAC Modulation A Rise Output Time Register 29 (PDIRTA29)	PDAC Modulation A Rise Output Time Register 30 (PDIRTA30)	PDAC Modulation A Rise Output Time Register 31 (PDIRTA31)	PDAC Modulation A Rise Output Time Register 32 (PDIRTA32)	8/16/32
H'FFFF 3820	PDAC Modulation A Rise Output Time Register 33 (PDIRTA33)	PDAC Modulation A Rise Output Time Register 34 (PDIRTA34)	PDAC Modulation A Rise Output Time Register 35 (PDIRTA35)	PDAC Modulation A Rise Output Time Register 36 (PDIRTA36)	8/16/32
H'FFFF 3824	PDAC Modulation A Rise Output Time Register 37 (PDIRTA37)	PDAC Modulation A Rise Output Time Register 38 (PDIRTA38)	PDAC Modulation A Rise Output Time Register 39 (PDIRTA39)	PDAC Modulation A Rise Output Time Register 40 (PDIRTA40)	8/16/32
H'FFFF 3828	PDAC Modulation A Rise Output Time Register 41 (PDIRTA41)	PDAC Modulation A Rise Output Time Register 42 (PDIRTA42)	PDAC Modulation A Rise Output Time Register 43 (PDIRTA43)	PDAC Modulation A Rise Output Time Register 44 (PDIRTA44)	8/16/32
H'FFFF 382C	PDAC Modulation A Rise Output Time Register 45 (PDIRTA45)	PDAC Modulation A Rise Output Time Register 46 (PDIRTA46)	PDAC Modulation A Rise Output Time Register 47 (PDIRTA47)	PDAC Modulation A Rise Output Time Register 48 (PDIRTA48)	8/16/32
H'FFFF 3830	PDAC Modulation A Rise Output Time Register 49 (PDIRTA49)	PDAC Modulation A Rise Output Time Register 50 (PDIRTA50)	PDAC Modulation A Rise Output Time Register 51 (PDIRTA51)	PDAC Modulation A Rise Output Time Register 52 (PDIRTA52)	8/16/32
H'FFFF 3834	PDAC Modulation A Rise Output Time Register 53 (PDIRTA53)	PDAC Modulation A Rise Output Time Register 54 (PDIRTA54)	PDAC Modulation A Rise Output Time Register 55 (PDIRTA55)	PDAC Modulation A Rise Output Time Register 56 (PDIRTA56)	8/16/32
H'FFFF 3838	PDAC Modulation A Rise Output Time Register 57 (PDIRTA57)	PDAC Modulation A Rise Output Time Register 58 (PDIRTA58)	PDAC Modulation A Rise Output Time Register 59 (PDIRTA59)	PDAC Modulation A Rise Output Time Register 60 (PDIRTA60)	8/16/32
H'FFFF 383C	PDAC Modulation A Rise Output Time Register 61 (PDIRTA61)	PDAC Modulation A Rise Output Time Register 62 (PDIRTA62)	PDAC Modulation A Rise Output Time Register 63 (PDIRTA63)	PDAC Modulation A Rise Output Time Register 64 (PDIRTA64)	8/16/32
H'FFFF 3840	PDAC Modulation A Rise Output Time Register 65 (PDIRTA65)	PDAC Modulation A Rise Output Time Register 66 (PDIRTA66)	PDAC Modulation A Rise Output Time Register 67 (PDIRTA67)	PDAC Modulation A Rise Output Time Register 68 (PDIRTA68)	8/16/32
H'FFFF 3844	PDAC Modulation A Rise Output Time Register 69 (PDIRTA69)	PDAC Modulation A Rise Output Time Register 70 (PDIRTA70)	PDAC Modulation A Rise Output Time Register 71 (PDIRTA71)	PDAC Modulation A Rise Output Time Register 72 (PDIRTA72)	8/16/32
H'FFFF 3848	PDAC Modulation A Rise Output Time Register 73 (PDIRTA73)	PDAC Modulation A Rise Output Time Register 74 (PDIRTA74)	PDAC Modulation A Rise Output Time Register 75 (PDIRTA75)	PDAC Modulation A Rise Output Time Register 76 (PDIRTA76)	8/16/32
H'FFFF 384C	PDAC Modulation A Rise Output Time Register 77 (PDIRTA77)	PDAC Modulation A Rise Output Time Register 78 (PDIRTA78)	PDAC Modulation A Rise Output Time Register 79 (PDIRTA79)	PDAC Modulation A Rise Output Time Register 80 (PDIRTA80)	8/16/32
H'FFFF 3850	PDAC Modulation A Rise Output Time Register 81 (PDIRTA81)	PDAC Modulation A Rise Output Time Register 82 (PDIRTA82)	PDAC Modulation A Rise Output Time Register 83 (PDIRTA83)	PDAC Modulation A Rise Output Time Register 84 (PDIRTA84)	8/16/32
H'FFFF 3854	PDAC Modulation A Rise Output Time Register 85 (PDIRTA85)	PDAC Modulation A Rise Output Time Register 86 (PDIRTA86)	PDAC Modulation A Rise Output Time Register 87 (PDIRTA87)	PDAC Modulation A Rise Output Time Register 88 (PDIRTA88)	8/16/32
H'FFFF 3858	PDAC Modulation A Rise Output Time Register 89 (PDIRTA89)	PDAC Modulation A Rise Output Time Register 90 (PDIRTA90)	PDAC Modulation A Rise Output Time Register 91 (PDIRTA91)	PDAC Modulation A Rise Output Time Register 92 (PDIRTA92)	8/16/32
H'FFFF 385C	PDAC Modulation A Rise Output Time Register 93 (PDIRTA93)	PDAC Modulation A Rise Output Time Register 94 (PDIRTA94)	PDAC Modulation A Rise Output Time Register 95 (PDIRTA95)	PDAC Modulation A Rise Output Time Register 96 (PDIRTA96)	8/16/32
H'FFFF 3860	PDAC Modulation A Rise Output Time Register 97 (PDIRTA97)	PDAC Modulation A Rise Output Time Register 98 (PDIRTA98)	PDAC Modulation A Rise Output Time Register 99 (PDIRTA99)	PDAC Modulation A Rise Output Time Register 100 (PDIRTA100)	8/16/32
H'FFFF 3864	PDAC Modulation A Rise Output Time Register 101 (PDIRTA101)	PDAC Modulation A Rise Output Time Register 102 (PDIRTA102)	PDAC Modulation A Rise Output Time Register 103 (PDIRTA103)	PDAC Modulation A Rise Output Time Register 104 (PDIRTA104)	8/16/32
H'FFFF 3868	PDAC Modulation A Rise Output Time Register 105 (PDIRTA105)	PDAC Modulation A Rise Output Time Register 106 (PDIRTA106)	PDAC Modulation A Rise Output Time Register 107 (PDIRTA107)	PDAC Modulation A Rise Output Time Register 108 (PDIRTA108)	8/16/32
H'FFFF 386C	PDAC Modulation A Rise Output Time Register 109 (PDIRTA109)	PDAC Modulation A Rise Output Time Register 110 (PDIRTA110)	PDAC Modulation A Rise Output Time Register 111 (PDIRTA111)	PDAC Modulation A Rise Output Time Register 112 (PDIRTA112)	8/16/32
H'FFFF 3870	PDAC Modulation A Rise Output Time Register 113 (PDIRTA113)	PDAC Modulation A Rise Output Time Register 114 (PDIRTA114)	PDAC Modulation A Rise Output Time Register 115 (PDIRTA115)	PDAC Modulation A Rise Output Time Register 116 (PDIRTA116)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 3874	PDAC Modulation A Rise Output Time Register 117 (PDIRTA117)	PDAC Modulation A Rise Output Time Register 118 (PDIRTA118)	PDAC Modulation A Rise Output Time Register 119 (PDIRTA119)	PDAC Modulation A Rise Output Time Register 120 (PDIRTA120)	8/16/32
:	(Reserved)				-
H'FFFF 3880	PDAC Modulation A Fall Output Time Register 1 (PDIFTA1)	PDAC Modulation A Fall Output Time Register 2 (PDIFTA2)	PDAC Modulation A Fall Output Time Register 3 (PDIFTA3)	PDAC Modulation A Fall Output Time Register 4 (PDIFTA4)	8/16/32
H'FFFF 3884	PDAC Modulation A Fall Output Time Register 5 (PDIFTA5)	PDAC Modulation A Fall Output Time Register 6 (PDIFTA6)	PDAC Modulation A Fall Output Time Register 7 (PDIFTA7)	PDAC Modulation A Fall Output Time Register 8 (PDIFTA8)	8/16/32
H'FFFF 3888	PDAC Modulation A Fall Output Time Register 9 (PDIFTA9)	PDAC Modulation A Fall Output Time Register 10 (PDIFTA10)	PDAC Modulation A Fall Output Time Register 11 (PDIFTA11)	PDAC Modulation A Fall Output Time Register 12 (PDIFTA12)	8/16/32
H'FFFF 388C	PDAC Modulation A Fall Output Time Register 13 (PDIFTA13)	PDAC Modulation A Fall Output Time Register 14 (PDIFTA14)	PDAC Modulation A Fall Output Time Register 15 (PDIFTA15)	PDAC Modulation A Fall Output Time Register 16 (PDIFTA16)	8/16/32
H'FFFF 3890	PDAC Modulation A Fall Output Time Register 17 (PDIFTA17)	PDAC Modulation A Fall Output Time Register 18 (PDIFTA18)	PDAC Modulation A Fall Output Time Register 19 (PDIFTA19)	PDAC Modulation A Fall Output Time Register 20 (PDIFTA20)	8/16/32
H'FFFF 3894	PDAC Modulation A Fall Output Time Register 21 (PDIFTA21)	PDAC Modulation A Fall Output Time Register 22 (PDIFTA22)	PDAC Modulation A Fall Output Time Register 23 (PDIFTA23)	PDAC Modulation A Fall Output Time Register 24 (PDIFTA24)	8/16/32
H'FFFF 3898	PDAC Modulation A Fall Output Time Register 25 (PDIFTA25)	PDAC Modulation A Fall Output Time Register 26 (PDIFTA26)	PDAC Modulation A Fall Output Time Register 27 (PDIFTA27)	PDAC Modulation A Fall Output Time Register 28 (PDIFTA28)	8/16/32
H'FFFF 389C	PDAC Modulation A Fall Output Time Register 29 (PDIFTA29)	PDAC Modulation A Fall Output Time Register 30 (PDIFTA30)	PDAC Modulation A Fall Output Time Register 31 (PDIFTA31)	PDAC Modulation A Fall Output Time Register 32 (PDIFTA32)	8/16/32
H'FFFF 38A0	PDAC Modulation A Fall Output Time Register 33 (PDIFTA33)	PDAC Modulation A Fall Output Time Register 34 (PDIFTA34)	PDAC Modulation A Fall Output Time Register 35 (PDIFTA35)	PDAC Modulation A Fall Output Time Register 36 (PDIFTA36)	8/16/32
H'FFFF 38A4	PDAC Modulation A Fall Output Time Register 37 (PDIFTA37)	PDAC Modulation A Fall Output Time Register 38 (PDIFTA38)	PDAC Modulation A Fall Output Time Register 39 (PDIFTA39)	PDAC Modulation A Fall Output Time Register 40 (PDIFTA40)	8/16/32
H'FFFF 38A8	PDAC Modulation A Fall Output Time Register 41 (PDIFTA41)	PDAC Modulation A Fall Output Time Register 42 (PDIFTA42)	PDAC Modulation A Fall Output Time Register 43 (PDIFTA43)	PDAC Modulation A Fall Output Time Register 44 (PDIFTA44)	8/16/32
H'FFFF 38AC	PDAC Modulation A Fall Output Time Register 45 (PDIFTA45)	PDAC Modulation A Fall Output Time Register 46 (PDIFTA46)	PDAC Modulation A Fall Output Time Register 47 (PDIFTA47)	PDAC Modulation A Fall Output Time Register 48 (PDIFTA48)	8/16/32
H'FFFF 38B0	PDAC Modulation A Fall Output Time Register 49 (PDIFTA49)	PDAC Modulation A Fall Output Time Register 50 (PDIFTA50)	PDAC Modulation A Fall Output Time Register 51 (PDIFTA51)	PDAC Modulation A Fall Output Time Register 52 (PDIFTA52)	8/16/32
H'FFFF 38B4	PDAC Modulation A Fall Output Time Register 53 (PDIFTA53)	PDAC Modulation A Fall Output Time Register 54 (PDIFTA54)	PDAC Modulation A Fall Output Time Register 55 (PDIFTA55)	PDAC Modulation A Fall Output Time Register 56 (PDIFTA56)	8/16/32
H'FFFF 38B8	PDAC Modulation A Fall Output Time Register 57 (PDIFTA57)	PDAC Modulation A Fall Output Time Register 58 (PDIFTA58)	PDAC Modulation A Fall Output Time Register 59 (PDIFTA59)	PDAC Modulation A Fall Output Time Register 60 (PDIFTA60)	8/16/32
H'FFFF 38BC	PDAC Modulation A Fall Output Time Register 61 (PDIFTA61)	PDAC Modulation A Fall Output Time Register 62 (PDIFTA62)	PDAC Modulation A Fall Output Time Register 63 (PDIFTA63)	PDAC Modulation A Fall Output Time Register 64 (PDIFTA64)	8/16/32
H'FFFF 38C0	PDAC Modulation A Fall Output Time Register 65 (PDIFTA65)	PDAC Modulation A Fall Output Time Register 66 (PDIFTA66)	PDAC Modulation A Fall Output Time Register 67 (PDIFTA67)	PDAC Modulation A Fall Output Time Register 68 (PDIFTA68)	8/16/32
H'FFFF 38C4	PDAC Modulation A Fall Output Time Register 69 (PDIFTA69)	PDAC Modulation A Fall Output Time Register 70 (PDIFTA70)	PDAC Modulation A Fall Output Time Register 71 (PDIFTA71)	PDAC Modulation A Fall Output Time Register 72 (PDIFTA72)	8/16/32
H'FFFF 38C8	PDAC Modulation A Fall Output Time Register 73 (PDIFTA73)	PDAC Modulation A Fall Output Time Register 74 (PDIFTA74)	PDAC Modulation A Fall Output Time Register 75 (PDIFTA75)	PDAC Modulation A Fall Output Time Register 76 (PDIFTA76)	8/16/32
H'FFFF 38CC	PDAC Modulation A Fall Output Time Register 77 (PDIFTA77)	PDAC Modulation A Fall Output Time Register 78 (PDIFTA78)	PDAC Modulation A Fall Output Time Register 79 (PDIFTA79)	PDAC Modulation A Fall Output Time Register 80 (PDIFTA80)	8/16/32
H'FFFF 38D0	PDAC Modulation A Fall Output Time Register 81 (PDIFTA81)	PDAC Modulation A Fall Output Time Register 82 (PDIFTA82)	PDAC Modulation A Fall Output Time Register 83 (PDIFTA83)	PDAC Modulation A Fall Output Time Register 84 (PDIFTA84)	8/16/32
H'FFFF 38D4	PDAC Modulation A Fall Output Time Register 85 (PDIFTA85)	PDAC Modulation A Fall Output Time Register 86 (PDIFTA86)	PDAC Modulation A Fall Output Time Register 87 (PDIFTA87)	PDAC Modulation A Fall Output Time Register 88 (PDIFTA88)	8/16/32
H'FFFF 38D8	PDAC Modulation A Fall Output Time Register 89 (PDIFTA89)	PDAC Modulation A Fall Output Time Register 90 (PDIFTA90)	PDAC Modulation A Fall Output Time Register 91 (PDIFTA91)	PDAC Modulation A Fall Output Time Register 92 (PDIFTA92)	8/16/32
H'FFFF 38DC	PDAC Modulation A Fall Output Time Register 93 (PDIFTA93)	PDAC Modulation A Fall Output Time Register 94 (PDIFTA94)	PDAC Modulation A Fall Output Time Register 95 (PDIFTA95)	PDAC Modulation A Fall Output Time Register 96 (PDIFTA96)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 38E0	PDAC Modulation A Fall Output Time Register 97 (PDIFTA97)	PDAC Modulation A Fall Output Time Register 98 (PDIFTA98)	PDAC Modulation A Fall Output Time Register 99 (PDIFTA99)	PDAC Modulation A Fall Output Time Register 100 (PDIFTA100)	8/16/32
H'FFFF 38E4	PDAC Modulation A Fall Output Time Register 101 (PDIFTA101)	PDAC Modulation A Fall Output Time Register 102 (PDIFTA102)	PDAC Modulation A Fall Output Time Register 103 (PDIFTA103)	PDAC Modulation A Fall Output Time Register 104 (PDIFTA104)	8/16/32
H'FFFF 38E8	PDAC Modulation A Fall Output Time Register 105 (PDIFTA105)	PDAC Modulation A Fall Output Time Register 106 (PDIFTA106)	PDAC Modulation A Fall Output Time Register 107 (PDIFTA107)	PDAC Modulation A Fall Output Time Register 108 (PDIFTA108)	8/16/32
H'FFFF 38EC	PDAC Modulation A Fall Output Time Register 109 (PDIFTA109)	PDAC Modulation A Fall Output Time Register 110 (PDIFTA110)	PDAC Modulation A Fall Output Time Register 111 (PDIFTA111)	PDAC Modulation A Fall Output Time Register 112 (PDIFTA112)	8/16/32
H'FFFF 38F0	PDAC Modulation A Fall Output Time Register 113 (PDIFTA113)	PDAC Modulation A Fall Output Time Register 114 (PDIFTA114)	PDAC Modulation A Fall Output Time Register 115 (PDIFTA115)	PDAC Modulation A Fall Output Time Register 116 (PDIFTA116)	8/16/32
H'FFFF 38F4	PDAC Modulation A Fall Output Time Register 117 (PDIFTA117)	PDAC Modulation A Fall Output Time Register 118 (PDIFTA118)	PDAC Modulation A Fall Output Time Register 119 (PDIFTA119)	PDAC Modulation A Fall Output Time Register 120 (PDIFTA120)	8/16/32
:	(Reserved)				-
H'FFFF 3900	PDAC Modulation B Rise Output Time Register 1 (PDIRTB1)	PDAC Modulation B Rise Output Time Register 2 (PDIRTB2)	PDAC Modulation B Rise Output Time Register 3 (PDIRTB3)	PDAC Modulation B Rise Output Time Register 4 (PDIRTB4)	8/16/32
H'FFFF 3904	PDAC Modulation B Rise Output Time Register 5 (PDIRTB5)	PDAC Modulation B Rise Output Time Register 6 (PDIRTB6)	PDAC Modulation B Rise Output Time Register 7 (PDIRTB7)	PDAC Modulation B Rise Output Time Register 8 (PDIRTB8)	8/16/32
H'FFFF 3908	PDAC Modulation B Rise Output Time Register 9 (PDIRTB9)	PDAC Modulation B Rise Output Time Register 10 (PDIRTB10)	PDAC Modulation B Rise Output Time Register 11 (PDIRTB11)	PDAC Modulation B Rise Output Time Register 12 (PDIRTB12)	8/16/32
H'FFFF 390C	PDAC Modulation B Rise Output Time Register 13 (PDIRTB13)	PDAC Modulation B Rise Output Time Register 14 (PDIRTB14)	PDAC Modulation B Rise Output Time Register 15 (PDIRTB15)	PDAC Modulation B Rise Output Time Register 16 (PDIRTB16)	8/16/32
H'FFFF 3910	PDAC Modulation B Rise Output Time Register 17 (PDIRTB17)	PDAC Modulation B Rise Output Time Register 18 (PDIRTB18)	PDAC Modulation B Rise Output Time Register 19 (PDIRTB19)	PDAC Modulation B Rise Output Time Register 20 (PDIRTB20)	8/16/32
H'FFFF 3914	PDAC Modulation B Rise Output Time Register 21 (PDIRTB21)	PDAC Modulation B Rise Output Time Register 22 (PDIRTB22)	PDAC Modulation B Rise Output Time Register 23 (PDIRTB23)	PDAC Modulation B Rise Output Time Register 24 (PDIRTB24)	8/16/32
H'FFFF 3918	PDAC Modulation B Rise Output Time Register 25 (PDIRTB25)	PDAC Modulation B Rise Output Time Register 26 (PDIRTB26)	PDAC Modulation B Rise Output Time Register 27 (PDIRTB27)	PDAC Modulation B Rise Output Time Register 28 (PDIRTB28)	8/16/32
H'FFFF 391C	PDAC Modulation B Rise Output Time Register 29 (PDIRTB29)	PDAC Modulation B Rise Output Time Register 30 (PDIRTB30)	PDAC Modulation B Rise Output Time Register 31 (PDIRTB31)	PDAC Modulation B Rise Output Time Register 32 (PDIRTB32)	8/16/32
H'FFFF 3920	PDAC Modulation B Rise Output Time Register 33 (PDIRTB33)	PDAC Modulation B Rise Output Time Register 34 (PDIRTB34)	PDAC Modulation B Rise Output Time Register 35 (PDIRTB35)	PDAC Modulation B Rise Output Time Register 36 (PDIRTB36)	8/16/32
H'FFFF 3924	PDAC Modulation B Rise Output Time Register 37 (PDIRTB37)	PDAC Modulation B Rise Output Time Register 38 (PDIRTB38)	PDAC Modulation B Rise Output Time Register 39 (PDIRTB39)	PDAC Modulation B Rise Output Time Register 40 (PDIRTB40)	8/16/32
H'FFFF 3928	PDAC Modulation B Rise Output Time Register 41 (PDIRTB41)	PDAC Modulation B Rise Output Time Register 42 (PDIRTB42)	PDAC Modulation B Rise Output Time Register 43 (PDIRTB43)	PDAC Modulation B Rise Output Time Register 44 (PDIRTB44)	8/16/32
H'FFFF 392C	PDAC Modulation B Rise Output Time Register 45 (PDIRTB45)	PDAC Modulation B Rise Output Time Register 46 (PDIRTB46)	PDAC Modulation B Rise Output Time Register 47 (PDIRTB47)	PDAC Modulation B Rise Output Time Register 48 (PDIRTB48)	8/16/32
H'FFFF 3930	PDAC Modulation B Rise Output Time Register 49 (PDIRTB49)	PDAC Modulation B Rise Output Time Register 50 (PDIRTB50)	PDAC Modulation B Rise Output Time Register 51 (PDIRTB51)	PDAC Modulation B Rise Output Time Register 52 (PDIRTB52)	8/16/32
H'FFFF 3934	PDAC Modulation B Rise Output Time Register 53 (PDIRTB53)	PDAC Modulation B Rise Output Time Register 54 (PDIRTB54)	PDAC Modulation B Rise Output Time Register 55 (PDIRTB55)	PDAC Modulation B Rise Output Time Register 56 (PDIRTB56)	8/16/32
H'FFFF 3938	PDAC Modulation B Rise Output Time Register 57 (PDIRTB57)	PDAC Modulation B Rise Output Time Register 58 (PDIRTB58)	PDAC Modulation B Rise Output Time Register 59 (PDIRTB59)	PDAC Modulation B Rise Output Time Register 60 (PDIRTB60)	8/16/32
H'FFFF 393C	PDAC Modulation B Rise Output Time Register 61 (PDIRTB61)	PDAC Modulation B Rise Output Time Register 62 (PDIRTB62)	PDAC Modulation B Rise Output Time Register 63 (PDIRTB63)	PDAC Modulation B Rise Output Time Register 64 (PDIRTB64)	8/16/32
H'FFFF 3940	PDAC Modulation B Rise Output Time Register 65 (PDIRTB65)	PDAC Modulation B Rise Output Time Register 66 (PDIRTB66)	PDAC Modulation B Rise Output Time Register 67 (PDIRTB67)	PDAC Modulation B Rise Output Time Register 68 (PDIRTB68)	8/16/32
H'FFFF 3944	PDAC Modulation B Rise Output Time Register 69 (PDIRTB69)	PDAC Modulation B Rise Output Time Register 70 (PDIRTB70)	PDAC Modulation B Rise Output Time Register 71 (PDIRTB71)	PDAC Modulation B Rise Output Time Register 72 (PDIRTB72)	8/16/32
H'FFFF 3948	PDAC Modulation B Rise Output Time Register 73 (PDIRTB73)	PDAC Modulation B Rise Output Time Register 74 (PDIRTB74)	PDAC Modulation B Rise Output Time Register 75 (PDIRTB75)	PDAC Modulation B Rise Output Time Register 76 (PDIRTB76)	8/16/32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF 394C	PDAC Modulation B Rise Output Time Register 77 (PDIRTB77)		PDAC Modulation B Rise Output Time Register 78 (PDIRTB78)		PDAC Modulation B Rise Output Time Register 79 (PDIRTB79)		PDAC Modulation B Rise Output Time Register 80 (PDIRTB80)		8/16/32
H'FFFF 3950	PDAC Modulation B Rise Output Time Register 81 (PDIRTB81)		PDAC Modulation B Rise Output Time Register 82 (PDIRTB82)		PDAC Modulation B Rise Output Time Register 83 (PDIRTB83)		PDAC Modulation B Rise Output Time Register 84 (PDIRTB84)		8/16/32
H'FFFF 3954	PDAC Modulation B Rise Output Time Register 85 (PDIRTB85)		PDAC Modulation B Rise Output Time Register 86 (PDIRTB86)		PDAC Modulation B Rise Output Time Register 87 (PDIRTB87)		PDAC Modulation B Rise Output Time Register 88 (PDIRTB88)		8/16/32
H'FFFF 3958	PDAC Modulation B Rise Output Time Register 89 (PDIRTB89)		PDAC Modulation B Rise Output Time Register 90 (PDIRTB90)		PDAC Modulation B Rise Output Time Register 91 (PDIRTB91)		PDAC Modulation B Rise Output Time Register 92 (PDIRTB92)		8/16/32
H'FFFF 395C	PDAC Modulation B Rise Output Time Register 93 (PDIRTB93)		PDAC Modulation B Rise Output Time Register 94 (PDIRTB94)		PDAC Modulation B Rise Output Time Register 95 (PDIRTB95)		PDAC Modulation B Rise Output Time Register 96 (PDIRTB96)		8/16/32
H'FFFF 3960	PDAC Modulation B Rise Output Time Register 97 (PDIRTB97)		PDAC Modulation B Rise Output Time Register 98 (PDIRTB98)		PDAC Modulation B Rise Output Time Register 99 (PDIRTB99)		PDAC Modulation B Rise Output Time Register 100 (PDIRTB100)		8/16/32
H'FFFF 3964	PDAC Modulation B Rise Output Time Register 101 (PDIRTB101)		PDAC Modulation B Rise Output Time Register 102 (PDIRTB102)		PDAC Modulation B Rise Output Time Register 103 (PDIRTB103)		PDAC Modulation B Rise Output Time Register 104 (PDIRTB104)		8/16/32
H'FFFF 3968	PDAC Modulation B Rise Output Time Register 105 (PDIRTB105)		PDAC Modulation B Rise Output Time Register 106 (PDIRTB106)		PDAC Modulation B Rise Output Time Register 107 (PDIRTB107)		PDAC Modulation B Rise Output Time Register 108 (PDIRTB108)		8/16/32
H'FFFF 396C	PDAC Modulation B Rise Output Time Register 109 (PDIRTB109)		PDAC Modulation B Rise Output Time Register 110 (PDIRTB110)		PDAC Modulation B Rise Output Time Register 111 (PDIRTB111)		PDAC Modulation B Rise Output Time Register 112 (PDIRTB112)		8/16/32
H'FFFF 3970	PDAC Modulation B Rise Output Time Register 113 (PDIRTB113)		PDAC Modulation B Rise Output Time Register 114 (PDIRTB114)		PDAC Modulation B Rise Output Time Register 115 (PDIRTB115)		PDAC Modulation B Rise Output Time Register 116 (PDIRTB116)		8/16/32
H'FFFF 3974	PDAC Modulation B Rise Output Time Register 117 (PDIRTB117)		PDAC Modulation B Rise Output Time Register 118 (PDIRTB118)		PDAC Modulation B Rise Output Time Register 119 (PDIRTB119)		PDAC Modulation B Rise Output Time Register 120 (PDIRTB120)		8/16/32
H'FFFF 3978	PDAC Modulation B Rise Output Time Register 121 (PDIRTB121)		PDAC Modulation B Rise Output Time Register 122 (PDIRTB122)		PDAC Modulation B Rise Output Time Register 123 (PDIRTB123)		PDAC Modulation B Rise Output Time Register 124 (PDIRTB124)		8/16/32
H'FFFF 397C	PDAC Modulation B Rise Output Time Register 125 (PDIRTB125)		PDAC Modulation B Rise Output Time Register 126 (PDIRTB126)		PDAC Modulation B Rise Output Time Register 127 (PDIRTB127)		PDAC Modulation B Rise Output Time Register 128 (PDIRTB128)		8/16/32
H'FFFF 3980	PDAC Modulation B Rise Output Time Register 129 (PDIRTB129)		PDAC Modulation B Rise Output Time Register 130 (PDIRTB130)		PDAC Modulation B Rise Output Time Register 131 (PDIRTB131)		PDAC Modulation B Rise Output Time Register 132 (PDIRTB132)		8/16/32
H'FFFF 3984	PDAC Modulation B Rise Output Time Register 133 (PDIRTB133)		PDAC Modulation B Rise Output Time Register 134 (PDIRTB134)		PDAC Modulation B Rise Output Time Register 135 (PDIRTB135)		PDAC Modulation B Rise Output Time Register 136 (PDIRTB136)		8/16/32
H'FFFF 3988	PDAC Modulation B Rise Output Time Register 137 (PDIRTB137)		PDAC Modulation B Rise Output Time Register 138 (PDIRTB138)		PDAC Modulation B Rise Output Time Register 139 (PDIRTB139)		PDAC Modulation B Rise Output Time Register 140 (PDIRTB140)		8/16/32
H'FFFF 398C	PDAC Modulation B Rise Output Time Register 141 (PDIRTB141)		PDAC Modulation B Rise Output Time Register 142 (PDIRTB142)		PDAC Modulation B Rise Output Time Register 143 (PDIRTB143)		PDAC Modulation B Rise Output Time Register 144 (PDIRTB144)		8/16/32
H'FFFF 3990	PDAC Modulation B Rise Output Time Register 145 (PDIRTB145)		PDAC Modulation B Rise Output Time Register 146 (PDIRTB146)		PDAC Modulation B Rise Output Time Register 147 (PDIRTB147)		PDAC Modulation B Rise Output Time Register 148 (PDIRTB148)		8/16/32
H'FFFF 3994	PDAC Modulation B Rise Output Time Register 149 (PDIRTB149)		PDAC Modulation B Rise Output Time Register 150 (PDIRTB150)		PDAC Modulation B Rise Output Time Register 151 (PDIRTB151)		PDAC Modulation B Rise Output Time Register 152 (PDIRTB152)		8/16/32
H'FFFF 3998	PDAC Modulation B Rise Output Time Register 153 (PDIRTB153)		PDAC Modulation B Rise Output Time Register 154 (PDIRTB154)		PDAC Modulation B Rise Output Time Register 155 (PDIRTB155)		PDAC Modulation B Rise Output Time Register 156 (PDIRTB156)		8/16/32
H'FFFF 399C	PDAC Modulation B Rise Output Time Register 157 (PDIRTB157)		PDAC Modulation B Rise Output Time Register 158 (PDIRTB158)		PDAC Modulation B Rise Output Time Register 159 (PDIRTB159)		PDAC Modulation B Rise Output Time Register 160 (PDIRTB160)		8/16/32
H'FFFF 39A0	PDAC Modulation B Rise Output Time Register 161 (PDIRTB161)		PDAC Modulation B Rise Output Time Register 162 (PDIRTB162)		PDAC Modulation B Rise Output Time Register 163 (PDIRTB163)		PDAC Modulation B Rise Output Time Register 164 (PDIRTB164)		8/16/32
H'FFFF 39A4	PDAC Modulation B Rise Output Time Register 165 (PDIRTB165)		PDAC Modulation B Rise Output Time Register 166 (PDIRTB166)		PDAC Modulation B Rise Output Time Register 167 (PDIRTB167)		PDAC Modulation B Rise Output Time Register 168 (PDIRTB168)		8/16/32
H'FFFF 39A8	PDAC Modulation B Rise Output Time Register 169 (PDIRTB169)		PDAC Modulation B Rise Output Time Register 170 (PDIRTB170)		PDAC Modulation B Rise Output Time Register 171 (PDIRTB171)		PDAC Modulation B Rise Output Time Register 172 (PDIRTB172)		8/16/32
H'FFFF 39AC	PDAC Modulation B Rise Output Time Register 173 (PDIRTB173)		PDAC Modulation B Rise Output Time Register 174 (PDIRTB174)		PDAC Modulation B Rise Output Time Register 175 (PDIRTB175)		PDAC Modulation B Rise Output Time Register 176 (PDIRTB176)		8/16/32
H'FFFF 39B0	PDAC Modulation B Rise Output Time Register 177 (PDIRTB177)		PDAC Modulation B Rise Output Time Register 178 (PDIRTB178)		PDAC Modulation B Rise Output Time Register 179 (PDIRTB179)		PDAC Modulation B Rise Output Time Register 180 (PDIRTB180)		8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 39B4	PDAC Modulation B Rise Output Time Register 181 (PDIRTB181)	PDAC Modulation B Rise Output Time Register 182 (PDIRTB182)	PDAC Modulation B Rise Output Time Register 183 (PDIRTB183)	PDAC Modulation B Rise Output Time Register 184 (PDIRTB184)	8/16/32
H'FFFF 39B8	PDAC Modulation B Rise Output Time Register 185 (PDIRTB185)	PDAC Modulation B Rise Output Time Register 186 (PDIRTB186)	PDAC Modulation B Rise Output Time Register 187 (PDIRTB187)	PDAC Modulation B Rise Output Time Register 188 (PDIRTB188)	8/16/32
H'FFFF 39BC	PDAC Modulation B Rise Output Time Register 189 (PDIRTB189)	PDAC Modulation B Rise Output Time Register 190 (PDIRTB190)	PDAC Modulation B Rise Output Time Register 191 (PDIRTB191)	PDAC Modulation B Rise Output Time Register 192 (PDIRTB192)	8/16/32
H'FFFF 39C0	PDAC Modulation B Rise Output Time Register 193 (PDIRTB193)	PDAC Modulation B Rise Output Time Register 194 (PDIRTB194)	PDAC Modulation B Rise Output Time Register 195 (PDIRTB195)	PDAC Modulation B Rise Output Time Register 196 (PDIRTB196)	8/16/32
H'FFFF 39C4	PDAC Modulation B Rise Output Time Register 197 (PDIRTB197)	PDAC Modulation B Rise Output Time Register 198 (PDIRTB198)	PDAC Modulation B Rise Output Time Register 199 (PDIRTB199)	PDAC Modulation B Rise Output Time Register 200 (PDIRTB200)	8/16/32
:	(Reserved)				-
H'FFFF 3A00	PDAC Modulation B Fall Output Time Register 1 (PDIFTB1)	PDAC Modulation B Fall Output Time Register 2 (PDIFTB2)	PDAC Modulation B Fall Output Time Register 3 (PDIFTB3)	PDAC Modulation B Fall Output Time Register 4 (PDIFTB4)	8/16/32
H'FFFF 3A04	PDAC Modulation B Fall Output Time Register 5 (PDIFTB5)	PDAC Modulation B Fall Output Time Register 6 (PDIFTB6)	PDAC Modulation B Fall Output Time Register 7 (PDIFTB7)	PDAC Modulation B Fall Output Time Register 8 (PDIFTB8)	8/16/32
H'FFFF 3A08	PDAC Modulation B Fall Output Time Register 9 (PDIFTB9)	PDAC Modulation B Fall Output Time Register 10 (PDIFTB10)	PDAC Modulation B Fall Output Time Register 11 (PDIFTB11)	PDAC Modulation B Fall Output Time Register 12 (PDIFTB12)	8/16/32
H'FFFF 3A0C	PDAC Modulation B Fall Output Time Register 13 (PDIFTB13)	PDAC Modulation B Fall Output Time Register 14 (PDIFTB14)	PDAC Modulation B Fall Output Time Register 15 (PDIFTB15)	PDAC Modulation B Fall Output Time Register 16 (PDIFTB16)	8/16/32
H'FFFF 3A10	PDAC Modulation B Fall Output Time Register 17 (PDIFTB17)	PDAC Modulation B Fall Output Time Register 18 (PDIFTB18)	PDAC Modulation B Fall Output Time Register 19 (PDIFTB19)	PDAC Modulation B Fall Output Time Register 20 (PDIFTB20)	8/16/32
H'FFFF 3A14	PDAC Modulation B Fall Output Time Register 21 (PDIFTB21)	PDAC Modulation B Fall Output Time Register 22 (PDIFTB22)	PDAC Modulation B Fall Output Time Register 23 (PDIFTB23)	PDAC Modulation B Fall Output Time Register 24 (PDIFTB24)	8/16/32
H'FFFF 3A18	PDAC Modulation B Fall Output Time Register 25 (PDIFTB25)	PDAC Modulation B Fall Output Time Register 26 (PDIFTB26)	PDAC Modulation B Fall Output Time Register 27 (PDIFTB27)	PDAC Modulation B Fall Output Time Register 28 (PDIFTB28)	8/16/32
H'FFFF 3A1C	PDAC Modulation B Fall Output Time Register 29 (PDIFTB29)	PDAC Modulation B Fall Output Time Register 30 (PDIFTB30)	PDAC Modulation B Fall Output Time Register 31 (PDIFTB31)	PDAC Modulation B Fall Output Time Register 32 (PDIFTB32)	8/16/32
H'FFFF 3A20	PDAC Modulation B Fall Output Time Register 33 (PDIFTB33)	PDAC Modulation B Fall Output Time Register 34 (PDIFTB34)	PDAC Modulation B Fall Output Time Register 35 (PDIFTB35)	PDAC Modulation B Fall Output Time Register 36 (PDIFTB36)	8/16/32
H'FFFF 3A24	PDAC Modulation B Fall Output Time Register 37 (PDIFTB37)	PDAC Modulation B Fall Output Time Register 38 (PDIFTB38)	PDAC Modulation B Fall Output Time Register 39 (PDIFTB39)	PDAC Modulation B Fall Output Time Register 40 (PDIFTB40)	8/16/32
H'FFFF 3A28	PDAC Modulation B Fall Output Time Register 41 (PDIFTB41)	PDAC Modulation B Fall Output Time Register 42 (PDIFTB42)	PDAC Modulation B Fall Output Time Register 43 (PDIFTB43)	PDAC Modulation B Fall Output Time Register 44 (PDIFTB44)	8/16/32
H'FFFF 3A2C	PDAC Modulation B Fall Output Time Register 45 (PDIFTB45)	PDAC Modulation B Fall Output Time Register 46 (PDIFTB46)	PDAC Modulation B Fall Output Time Register 47 (PDIFTB47)	PDAC Modulation B Fall Output Time Register 48 (PDIFTB48)	8/16/32
H'FFFF 3A30	PDAC Modulation B Fall Output Time Register 49 (PDIFTB49)	PDAC Modulation B Fall Output Time Register 50 (PDIFTB50)	PDAC Modulation B Fall Output Time Register 51 (PDIFTB51)	PDAC Modulation B Fall Output Time Register 52 (PDIFTB52)	8/16/32
H'FFFF 3A34	PDAC Modulation B Fall Output Time Register 53 (PDIFTB53)	PDAC Modulation B Fall Output Time Register 54 (PDIFTB54)	PDAC Modulation B Fall Output Time Register 55 (PDIFTB55)	PDAC Modulation B Fall Output Time Register 56 (PDIFTB56)	8/16/32
H'FFFF 3A38	PDAC Modulation B Fall Output Time Register 57 (PDIFTB57)	PDAC Modulation B Fall Output Time Register 58 (PDIFTB58)	PDAC Modulation B Fall Output Time Register 59 (PDIFTB59)	PDAC Modulation B Fall Output Time Register 60 (PDIFTB60)	8/16/32
H'FFFF 3A3C	PDAC Modulation B Fall Output Time Register 61 (PDIFTB61)	PDAC Modulation B Fall Output Time Register 62 (PDIFTB62)	PDAC Modulation B Fall Output Time Register 63 (PDIFTB63)	PDAC Modulation B Fall Output Time Register 64 (PDIFTB64)	8/16/32
H'FFFF 3A40	PDAC Modulation B Fall Output Time Register 65 (PDIFTB65)	PDAC Modulation B Fall Output Time Register 66 (PDIFTB66)	PDAC Modulation B Fall Output Time Register 67 (PDIFTB67)	PDAC Modulation B Fall Output Time Register 68 (PDIFTB68)	8/16/32
H'FFFF 3A44	PDAC Modulation B Fall Output Time Register 69 (PDIFTB69)	PDAC Modulation B Fall Output Time Register 70 (PDIFTB70)	PDAC Modulation B Fall Output Time Register 71 (PDIFTB71)	PDAC Modulation B Fall Output Time Register 72 (PDIFTB72)	8/16/32
H'FFFF 3A48	PDAC Modulation B Fall Output Time Register 73 (PDIFTB73)	PDAC Modulation B Fall Output Time Register 74 (PDIFTB74)	PDAC Modulation B Fall Output Time Register 75 (PDIFTB75)	PDAC Modulation B Fall Output Time Register 76 (PDIFTB76)	8/16/32
H'FFFF 3A4C	PDAC Modulation B Fall Output Time Register 77 (PDIFTB77)	PDAC Modulation B Fall Output Time Register 78 (PDIFTB78)	PDAC Modulation B Fall Output Time Register 79 (PDIFTB79)	PDAC Modulation B Fall Output Time Register 80 (PDIFTB80)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 3A50	PDAC Modulation B Fall Output Time Register 81 (PDIFTB81)	PDAC Modulation B Fall Output Time Register 82 (PDIFTB82)	PDAC Modulation B Fall Output Time Register 83 (PDIFTB83)	PDAC Modulation B Fall Output Time Register 84 (PDIFTB84)	8/16/32
H'FFFF 3A54	PDAC Modulation B Fall Output Time Register 85 (PDIFTB85)	PDAC Modulation B Fall Output Time Register 86 (PDIFTB86)	PDAC Modulation B Fall Output Time Register 87 (PDIFTB87)	PDAC Modulation B Fall Output Time Register 88 (PDIFTB88)	8/16/32
H'FFFF 3A58	PDAC Modulation B Fall Output Time Register 89 (PDIFTB89)	PDAC Modulation B Fall Output Time Register 90 (PDIFTB90)	PDAC Modulation B Fall Output Time Register 91 (PDIFTB91)	PDAC Modulation B Fall Output Time Register 92 (PDIFTB92)	8/16/32
H'FFFF 3A5C	PDAC Modulation B Fall Output Time Register 93 (PDIFTB93)	PDAC Modulation B Fall Output Time Register 94 (PDIFTB94)	PDAC Modulation B Fall Output Time Register 95 (PDIFTB95)	PDAC Modulation B Fall Output Time Register 96 (PDIFTB96)	8/16/32
H'FFFF 3A60	PDAC Modulation B Fall Output Time Register 97 (PDIFTB97)	PDAC Modulation B Fall Output Time Register 98 (PDIFTB98)	PDAC Modulation B Fall Output Time Register 99 (PDIFTB99)	PDAC Modulation B Fall Output Time Register 100 (PDIFTB100)	8/16/32
H'FFFF 3A64	PDAC Modulation B Fall Output Time Register 101 (PDIFTB101)	PDAC Modulation B Fall Output Time Register 102 (PDIFTB102)	PDAC Modulation B Fall Output Time Register 103 (PDIFTB103)	PDAC Modulation B Fall Output Time Register 104 (PDIFTB104)	8/16/32
H'FFFF 3A68	PDAC Modulation B Fall Output Time Register 105 (PDIFTB105)	PDAC Modulation B Fall Output Time Register 106 (PDIFTB106)	PDAC Modulation B Fall Output Time Register 107 (PDIFTB107)	PDAC Modulation B Fall Output Time Register 108 (PDIFTB108)	8/16/32
H'FFFF 3A6C	PDAC Modulation B Fall Output Time Register 109 (PDIFTB109)	PDAC Modulation B Fall Output Time Register 110 (PDIFTB110)	PDAC Modulation B Fall Output Time Register 111 (PDIFTB111)	PDAC Modulation B Fall Output Time Register 112 (PDIFTB112)	8/16/32
H'FFFF 3A70	PDAC Modulation B Fall Output Time Register 113 (PDIFTB113)	PDAC Modulation B Fall Output Time Register 114 (PDIFTB114)	PDAC Modulation B Fall Output Time Register 115 (PDIFTB115)	PDAC Modulation B Fall Output Time Register 116 (PDIFTB116)	8/16/32
H'FFFF 3A74	PDAC Modulation B Fall Output Time Register 117 (PDIFTB117)	PDAC Modulation B Fall Output Time Register 118 (PDIFTB118)	PDAC Modulation B Fall Output Time Register 119 (PDIFTB119)	PDAC Modulation B Fall Output Time Register 120 (PDIFTB120)	8/16/32
H'FFFF 3A78	PDAC Modulation B Fall Output Time Register 121 (PDIFTB121)	PDAC Modulation B Fall Output Time Register 122 (PDIFTB122)	PDAC Modulation B Fall Output Time Register 123 (PDIFTB123)	PDAC Modulation B Fall Output Time Register 124 (PDIFTB124)	8/16/32
H'FFFF 3A7C	PDAC Modulation B Fall Output Time Register 125 (PDIFTB125)	PDAC Modulation B Fall Output Time Register 126 (PDIFTB126)	PDAC Modulation B Fall Output Time Register 127 (PDIFTB127)	PDAC Modulation B Fall Output Time Register 128 (PDIFTB128)	8/16/32
H'FFFF 3A80	PDAC Modulation B Fall Output Time Register 129 (PDIFTB129)	PDAC Modulation B Fall Output Time Register 130 (PDIFTB130)	PDAC Modulation B Fall Output Time Register 131 (PDIFTB131)	PDAC Modulation B Fall Output Time Register 132 (PDIFTB132)	8/16/32
H'FFFF 3A84	PDAC Modulation B Fall Output Time Register 133 (PDIFTB133)	PDAC Modulation B Fall Output Time Register 134 (PDIFTB134)	PDAC Modulation B Fall Output Time Register 135 (PDIFTB135)	PDAC Modulation B Fall Output Time Register 136 (PDIFTB136)	8/16/32
H'FFFF 3A88	PDAC Modulation B Fall Output Time Register 137 (PDIFTB137)	PDAC Modulation B Fall Output Time Register 138 (PDIFTB138)	PDAC Modulation B Fall Output Time Register 139 (PDIFTB139)	PDAC Modulation B Fall Output Time Register 140 (PDIFTB140)	8/16/32
H'FFFF 3A8C	PDAC Modulation B Fall Output Time Register 141 (PDIFTB141)	PDAC Modulation B Fall Output Time Register 142 (PDIFTB142)	PDAC Modulation B Fall Output Time Register 143 (PDIFTB143)	PDAC Modulation B Fall Output Time Register 144 (PDIFTB144)	8/16/32
H'FFFF 3A90	PDAC Modulation B Fall Output Time Register 145 (PDIFTB145)	PDAC Modulation B Fall Output Time Register 146 (PDIFTB146)	PDAC Modulation B Fall Output Time Register 147 (PDIFTB147)	PDAC Modulation B Fall Output Time Register 148 (PDIFTB148)	8/16/32
H'FFFF 3A94	PDAC Modulation B Fall Output Time Register 149 (PDIFTB149)	PDAC Modulation B Fall Output Time Register 150 (PDIFTB150)	PDAC Modulation B Fall Output Time Register 151 (PDIFTB151)	PDAC Modulation B Fall Output Time Register 152 (PDIFTB152)	8/16/32
H'FFFF 3A98	PDAC Modulation B Fall Output Time Register 153 (PDIFTB153)	PDAC Modulation B Fall Output Time Register 154 (PDIFTB154)	PDAC Modulation B Fall Output Time Register 155 (PDIFTB155)	PDAC Modulation B Fall Output Time Register 156 (PDIFTB156)	8/16/32
H'FFFF 3A9C	PDAC Modulation B Fall Output Time Register 157 (PDIFTB157)	PDAC Modulation B Fall Output Time Register 158 (PDIFTB158)	PDAC Modulation B Fall Output Time Register 159 (PDIFTB159)	PDAC Modulation B Fall Output Time Register 160 (PDIFTB160)	8/16/32
H'FFFF 3AA0	PDAC Modulation B Fall Output Time Register 161 (PDIFTB161)	PDAC Modulation B Fall Output Time Register 162 (PDIFTB162)	PDAC Modulation B Fall Output Time Register 163 (PDIFTB163)	PDAC Modulation B Fall Output Time Register 164 (PDIFTB164)	8/16/32
H'FFFF 3AA4	PDAC Modulation B Fall Output Time Register 165 (PDIFTB165)	PDAC Modulation B Fall Output Time Register 166 (PDIFTB166)	PDAC Modulation B Fall Output Time Register 167 (PDIFTB167)	PDAC Modulation B Fall Output Time Register 168 (PDIFTB168)	8/16/32
H'FFFF 3AA8	PDAC Modulation B Fall Output Time Register 169 (PDIFTB169)	PDAC Modulation B Fall Output Time Register 170 (PDIFTB170)	PDAC Modulation B Fall Output Time Register 171 (PDIFTB171)	PDAC Modulation B Fall Output Time Register 172 (PDIFTB172)	8/16/32
H'FFFF 3AAC	PDAC Modulation B Fall Output Time Register 173 (PDIFTB173)	PDAC Modulation B Fall Output Time Register 174 (PDIFTB174)	PDAC Modulation B Fall Output Time Register 175 (PDIFTB175)	PDAC Modulation B Fall Output Time Register 176 (PDIFTB176)	8/16/32
H'FFFF 3AB0	PDAC Modulation B Fall Output Time Register 177 (PDIFTB177)	PDAC Modulation B Fall Output Time Register 178 (PDIFTB178)	PDAC Modulation B Fall Output Time Register 179 (PDIFTB179)	PDAC Modulation B Fall Output Time Register 180 (PDIFTB180)	8/16/32
H'FFFF 3AB4	PDAC Modulation B Fall Output Time Register 181 (PDIFTB181)	PDAC Modulation B Fall Output Time Register 182 (PDIFTB182)	PDAC Modulation B Fall Output Time Register 183 (PDIFTB183)	PDAC Modulation B Fall Output Time Register 184 (PDIFTB184)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 3AB8	PDAC Modulation B Fall Output Time Register 185 (PDIFTB185)	PDAC Modulation B Fall Output Time Register 186 (PDIFTB186)	PDAC Modulation B Fall Output Time Register 187 (PDIFTB187)	PDAC Modulation B Fall Output Time Register 188 (PDIFTB188)	8/16/32
H'FFFF 3ABC	PDAC Modulation B Fall Output Time Register 189 (PDIFTB189)	PDAC Modulation B Fall Output Time Register 190 (PDIFTB190)	PDAC Modulation B Fall Output Time Register 191 (PDIFTB191)	PDAC Modulation B Fall Output Time Register 192 (PDIFTB192)	8/16/32
H'FFFF 3AC0	PDAC Modulation B Fall Output Time Register 193 (PDIFTB193)	PDAC Modulation B Fall Output Time Register 194 (PDIFTB194)	PDAC Modulation B Fall Output Time Register 195 (PDIFTB195)	PDAC Modulation B Fall Output Time Register 196 (PDIFTB196)	8/16/32
H'FFFF 3AC4	PDAC Modulation B Fall Output Time Register 197 (PDIFTB197)	PDAC Modulation B Fall Output Time Register 198 (PDIFTB198)	PDAC Modulation B Fall Output Time Register 199 (PDIFTB199)	PDAC Modulation B Fall Output Time Register 200 (PDIFTB200)	8/16/32
:	(Reserved)				-
H'FFFF 3B00 to H'FFFF 3D57	PDAC Modulation C Rise Output Time Register 1 to 600 (PDIRTC1 to PDIRTC600)				8/16/32
:	(Reserved)				-
H'FFFF 3D80 to H'FFFF 3FD7	PDAC Modulation C Fall Output Time Register 1 to 600 (PDIRTC1 to PDIRTC600)				8/16/32
:	(Reserved)				-
H'FFFF 4000	A/D0 Control Register (AD0CSR)	(Reserved)	A/D0Conversion Status Register (AD0REF)	(Reserved)	8, -, 8, -
H'FFFF 4004	A/D0 Interrupt Trigger Enable Register (AD0TRE)		A/D0 Interrupt Trigger Conversion End Flag Register (AD0TRF)		8/16
H'FFFF 4008	A/D0 Interrupt Trigger Source Select Register (AD0TRS)		A/D0 Interrupt Software Trigger Register (AD0STRG)		8/16
H'FFFF 400C	A/D0 Interrupt Trigger Conversion End Interrupt Enable Register (AD0TRD)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 401C	A/D0-Converted Value Addition Mode Select Register (AD0ADS)	(Reserved)	A/D0-Converted Value Addition Count Select Register (AD0ADC)	(Reserved)	8, -, 8, -
H'FFFF 4020	A/D0 Channel Select Register (AD0ANS)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 4030	A/D0 Control Extended Register (AD0CER)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 403C	(Reserved)	(Reserved)	A/D0 Data Register DIAG0 (AD0DRD)		-, -, 16
H'FFFF 4040	A/D0 Data Register 0 (AD0DR0)		(Reserved)	(Reserved)	16, -, -
H'FFFF 4044	A/D0 Data Register 2 (AD0DR2)		(Reserved)	(Reserved)	16, -, -
H'FFFF 4048	A/D0 Data Register 4 (AD0DR4)		(Reserved)	(Reserved)	16, -, -
H'FFFF 404C	A/D0 Data Register 6 (AD0DR6)		(Reserved)	(Reserved)	16, -, -
H'FFFF 4050	A/D0 Data Register 8 (AD0DR8)		A/D0 Data Register 9 (AD0DR9)		16
H'FFFF 4054	A/D0 Data Register 10 (AD0DR10)		A/D0 Data Register 11 (AD0DR11)		16
H'FFFF 4058	A/D0 Data Register 12 (AD0DR12)		A/D0 Data Register 13 (AD0DR13)		16
H'FFFF 405C	A/D0 Data Register 14 (AD0DR14)		A/D0 Data Register 15 (AD0DR15)		16
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 4400	A/D1 Control Register (AD1CSR)	(Reserved)	A/D1Conversion Status Register (AD1REF)	(Reserved)	8, -, 8, -
:	(Reserved)				-
H'FFFF 4410	A/D1 Interrupt Trigger Enable Register (AD1TRE)	(Reserved)	A/D1 Interrupt Trigger Conversion End Flag Register (AD1TRF)	(Reserved)	8, -, 8, -
H'FFFF 4414	A/D1 Interrupt Trigger Source Select Register (AD1TRS)	(Reserved)	A/D1 Interrupt Software Trigger Register (AD1STRG)	(Reserved)	8, -, 8, -
H'FFFF 4418	A/D1 Interrupt Trigger Conversion End Interrupt Enable Register (AD1TRD)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF 441C	A/D1-Converted Value Addition Mode Select Register (AD1ADS)	(Reserved)	A/D1i-Converted Value Addition Count Select Register (AD1ADC)	(Reserved)	8, -, 8, -
H'FFFF 4420	A/D1 Channel Select Register (AD1ANS)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 4430	A/D1 Control Extended Register (AD1CER)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 443C	(Reserved)	(Reserved)	A/D1 Data Register DIAG1 (AD1DRD)		-, -, 16
H'FFFF 4440	A/D1 Data Register 0 (AD1DR0)		A/D1 Data Register 1 (AD1DR1)		16
H'FFFF 4444	(Reserved)				-
H'FFFF 4448	A/D1 Data Register 4 (AD1DR4)		A/D1 Data Register 5 (AD1DR5)		16
:	(Reserved)				-
H'FFFF 5000	(Reserved)	(Reserved)	Port A Data Register (PADR)		-, -, 8/16
H'FFFF 5004	(Reserved)	(Reserved)	Port A I/O Register (PAIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5010	Port A Control Register 4 (PACR4)		Port A Control Register 3 (PACR3)		8/16/32
H'FFFF 5014	Port A Control Register 2 (PACR2)		Port A Control Register 1 (PACR1)		8/16/32
:	(Reserved)				-
H'FFFF 501C	(Reserved)	(Reserved)	Port A Port Register (PAPR)		-, -, 8/16
H'FFFF 5020	Dummy Access Area (DUMMYHPB0)				8/16/32
:	(Reserved)				-
H'FFFF 5098	(Reserved)	(Reserved)	Port A Driving Ability Setting Register (PADSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5100	(Reserved)	(Reserved)	Port B Data Register (PBDR)		-, -, 8/16
H'FFFF 5104	(Reserved)	(Reserved)	Port B IO Register (PBIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5114	Port B Control Register 2 (PBCR2)		Port B Control Register 1 (PBCR1)		8/16/32
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 511C	(Reserved)	(Reserved)	Port B Port Register (PBPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5198	(Reserved)	(Reserved)	Port B Driving Ability Setting Register (PBDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5200	(Reserved)	(Reserved)	Port C Data Register (PCDR)		-, -, 8/16
H'FFFF 5204	(Reserved)	(Reserved)	Port C IO Register (PCIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5210	Port C Control Register 4 (PCCR4)		Port C Control Register 3 (PCCR3)		8/16/32
H'FFFF 5214	Port C Control Register 2 (PCCR2)		Port C Control Register 1 (PCCR1)		8/16/32
:	(Reserved)				-
H'FFFF 521C	(Reserved)	(Reserved)	Port C Port Register (PCPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5298	(Reserved)	(Reserved)	Port C Driving Ability Setting Register (PCDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5300	Port ABC Input Threshold Value Switching Register (PALVR)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 5340	Port DRI Input Channel Switching Register (PDRIR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF 5400	(Reserved)	(Reserved)	Port D Data Register (PDDR)		-, -, 8/16
H'FFFF 5404	(Reserved)	(Reserved)	Port D IO Register (PDIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5410	Port D Control Register 4 (PDCR4)		Port D Control Register 3 (PDCR3)		8/16/32
H'FFFF 5414	Port D Control Register 2 (PDCR2)		Port D Control Register 1 (PDCR1)		8/16/32
:	(Reserved)				-
H'FFFF 541C	(Reserved)	(Reserved)	Port D Port Register (PDPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5498	(Reserved)	(Reserved)	Port D Driving Ability Setting Register (PDDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5500	(Reserved)	(Reserved)	Port E Data Register (PEDR)		-, -, 8/16
H'FFFF 5504	(Reserved)	(Reserved)	Port E IO Register (PEIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5510	Port E Control Register 4 (PECR4)		Port E Control Register 3 (PECR3)		8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
:	(Reserved)				-
H'FFFF 551C	(Reserved)	(Reserved)	Port E Port Register (PEPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5598	(Reserved)	(Reserved)	Port E Driving Ability Setting Register (PEDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5600	(Reserved)	(Reserved)	Port F Data Register (PFDR)		-, -, 8/16
H'FFFF 5604	(Reserved)	(Reserved)	Port F IO Register (PFIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5614	Port F Control Register 2 (PFCR2)		Port F Control Register 1 (PFCR1)		8/16/32
:	(Reserved)				-
H'FFFF 561C	(Reserved)	(Reserved)	Port F Port Register (PFPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5698	(Reserved)	(Reserved)	Port F Driving Ability Setting Register (PFDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5700	Port DEF Input Threshold Value Switching Register (PDLVR)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 5800	(Reserved)	(Reserved)	Port G Data Register (PGDR)		-, -, 8/16
H'FFFF 5804	(Reserved)	(Reserved)	Port G IO Register (PGIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5814	Port G Control Register 2 (PGCR2)		Port G Control Register 1 (PGCR1)		8/16/32
:	(Reserved)				-
H'FFFF 581C	(Reserved)	(Reserved)	Port G Port Register (PGPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5898	(Reserved)	(Reserved)	Port G Driving Ability Setting Register (PGDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5900	(Reserved)	(Reserved)	Port H Data Register (PHDR)		-, -, 8/16
H'FFFF 5904	(Reserved)	(Reserved)	Port H IO Register (PHIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5910	Port H Control Register 4 (PHCR4)		Port H Control Register 3 (PHCR3)		8/16/32
H'FFFF 5914	Port H Control Register 2 (PHCR2)		Port H Control Register 1 (PHCR1)		8/16/32
:	(Reserved)				-
H'FFFF 591C	(Reserved)	(Reserved)	Port H Port Register (PHPR)		-, -, 8/16
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 5998	(Reserved)	(Reserved)	Port H Driving Ability Setting Register (PHDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5A00	(Reserved)	(Reserved)	Port J Data Register (PJDR)		-, -, 8/16
H'FFFF 5A04	(Reserved)	(Reserved)	Port J IO Register (PJIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5A10	Port J Control Register 4 (PJCR4)		Port J Control Register 3 (PJCR3)		8/16/32
H'FFFF 5A14	Port J Control Register 2 (PJCR2)		Port J Control Register 1 (PJCR1)		8/16/32
:	(Reserved)				-
H'FFFF 5A1C	(Reserved)	(Reserved)	Port J Port Register (PJPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5A98	(Reserved)	(Reserved)	Port J Driving Ability Setting Register (PJDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5B00	Port GHJ Input Threshold Value Switching Register (PGLVR)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 5C00	(Reserved)	(Reserved)	Port K Data Register (PKDR)		-, -, 8/16
H'FFFF 5C04	(Reserved)	(Reserved)	Port K IO Register (PKIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5C10	Port K Control Register 4 (PKCR4)		Port K Control Register 3 (PKCR3)		8/16/32
H'FFFF 5C14	Port K Control Register 2 (PKCR2)		Port K Control Register 1 (PKCR1)		8/16/32
:	(Reserved)				-
H'FFFF 5C1C	(Reserved)	(Reserved)	Port K Port Register (PKPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5C98	(Reserved)	(Reserved)	Port K Driving Ability Setting Register (PKDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5D00	(Reserved)	(Reserved)	Port L Data Register (PLDR)		-, -, 8/16
H'FFFF 5D04	(Reserved)	(Reserved)	Port L IO Register (PLIOR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5D10	(Reserved)	(Reserved)	Port L Control Register 3 (PLCR3)		-, -, 8/16
H'FFFF 5D14	Port L Control Register 2 (PLCR2)		Port L Control Register 1 (PLCR1)		8/16/32
:	(Reserved)				-
H'FFFF 5D1C	(Reserved)	(Reserved)	Port L Port Register (PLPR)		-, -, 8/16
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 5D98	(Reserved)	(Reserved)	Port L Driving Ability Setting Register (PLDSR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5E00	Port KL Input Threshold Value Switching Register (PKLVR)		(Reserved)	(Reserved)	8/16, -, -
:	(Reserved)				-
H'FFFF 5E10	Port M Control Register 4 (PMCR4)		Port M Control Register 3 (PMCR3)		8/16/32
H'FFFF 5E14	Port M Control Register 2 (PMCR2)		Port M Control Register 1 (PMCR1)		8/16/32
:	(Reserved)				-
H'FFFF 5E1C	(Reserved)	(Reserved)	Port M Port Register (PMPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 5F14	Port N Control Register 2 (PNCR2)		Port N Control Register 1 (PNCR1)		8/16/32
:	(Reserved)				-
H'FFFF 5F1C	(Reserved)	(Reserved)	Port N Port Register (PNPR)		-, -, 8/16
:	(Reserved)				-
H'FFFF 6000	CAN0 Mailbox Register 0 (C0MB0)				8/16/32
H'FFFF 6004	CAN0 Mailbox Register 0 (C0MB0)				8/16/32
H'FFFF 6008	CAN0 Mailbox Register 0 (C0MB0)				8/16/32
H'FFFF 600C	CAN0 Mailbox Register 0 (C0MB0)				8/16/32
H'FFFF 6010	CAN0 Mailbox Register 1 (C0MB1)				8/16/32
H'FFFF 6014	CAN0 Mailbox Register 1 (C0MB1)				8/16/32
H'FFFF 6018	CAN0 Mailbox Register 1 (C0MB1)				8/16/32
H'FFFF 601C	CAN0 Mailbox Register 1 (C0MB1)				8/16/32
H'FFFF 6020	CAN0 Mailbox Register 2 (C0MB2)				8/16/32
H'FFFF 6024	CAN0 Mailbox Register 2 (C0MB2)				8/16/32
H'FFFF 6028	CAN0 Mailbox Register 2 (C0MB2)				8/16/32
H'FFFF 602C	CAN0 Mailbox Register 2 (C0MB2)				8/16/32
H'FFFF 6030	CAN0 Mailbox Register 3 (C0MB3)				8/16/32
H'FFFF 6034	CAN0 Mailbox Register 3 (C0MB3)				8/16/32
H'FFFF 6038	CAN0 Mailbox Register 3 (C0MB3)				8/16/32
H'FFFF 603C	CAN0 Mailbox Register 3 (C0MB3)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 6040	CAN0 Mailbox Register 4 (C0MB4)				8/16/32
H'FFFF 6044	CAN0 Mailbox Register 4 (C0MB4)				8/16/32
H'FFFF 6048	CAN0 Mailbox Register 4 (C0MB4)				8/16/32
H'FFFF 604C	CAN0 Mailbox Register 4 (C0MB4)				8/16/32
H'FFFF 6050	CAN0 Mailbox Register 5 (C0MB5)				8/16/32
H'FFFF 6054	CAN0 Mailbox Register 5 (C0MB5)				8/16/32
H'FFFF 6058	CAN0 Mailbox Register 5 (C0MB5)				8/16/32
H'FFFF 605C	CAN0 Mailbox Register 5 (C0MB5)				8/16/32
H'FFFF 6060	CAN0 Mailbox Register 6 (C0MB6)				8/16/32
H'FFFF 6064	CAN0 Mailbox Register 6 (C0MB6)				8/16/32
H'FFFF 6068	CAN0 Mailbox Register 6 (C0MB6)				8/16/32
H'FFFF 606C	CAN0 Mailbox Register 6 (C0MB6)				8/16/32
H'FFFF 6070	CAN0 Mailbox Register 7 (C0MB7)				8/16/32
H'FFFF 6074	CAN0 Mailbox Register 7 (C0MB7)				8/16/32
H'FFFF 6078	CAN0 Mailbox Register 7 (C0MB7)				8/16/32
H'FFFF 607C	CAN0 Mailbox Register 7 (C0MB7)				8/16/32
H'FFFF 6080	CAN0 Mailbox Register 8 (C0MB8)				8/16/32
H'FFFF 6084	CAN0 Mailbox Register 8 (C0MB8)				8/16/32
H'FFFF 6088	CAN0 Mailbox Register 8 (C0MB8)				8/16/32
H'FFFF 608C	CAN0 Mailbox Register 8 (C0MB8)				8/16/32
H'FFFF 6090	CAN0 Mailbox Register 9 (C0MB9)				8/16/32
H'FFFF 6094	CAN0 Mailbox Register 9 (C0MB9)				8/16/32
H'FFFF 6098	CAN0 Mailbox Register 9 (C0MB9)				8/16/32
H'FFFF 609C	CAN0 Mailbox Register 9 (C0MB9)				8/16/32
H'FFFF 60A0	CAN0 Mailbox Register 10 (C0MB10)				8/16/32
H'FFFF 60A4	CAN0 Mailbox Register 10 (C0MB10)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 60A8	CAN0 Mailbox Register 10 (C0MB10)				8/16/32
H'FFFF 60AC	CAN0 Mailbox Register 10 (C0MB10)				8/16/32
H'FFFF 60B0	CAN0 Mailbox Register 11 (C0MB11)				8/16/32
H'FFFF 60B4	CAN0 Mailbox Register 11 (C0MB11)				8/16/32
H'FFFF 60B8	CAN0 Mailbox Register 11 (C0MB11)				8/16/32
H'FFFF 60BC	CAN0 Mailbox Register 11 (C0MB11)				8/16/32
H'FFFF 60C0	CAN0 Mailbox Register 12 (C0MB12)				8/16/32
H'FFFF 60C4	CAN0 Mailbox Register 12 (C0MB12)				8/16/32
H'FFFF 60C8	CAN0 Mailbox Register 12 (C0MB12)				8/16/32
H'FFFF 60CC	CAN0 Mailbox Register 12 (C0MB12)				8/16/32
H'FFFF 60D0	CAN0 Mailbox Register 13 (C0MB13)				8/16/32
H'FFFF 60D4	CAN0 Mailbox Register 13 (C0MB13)				8/16/32
H'FFFF 60D8	CAN0 Mailbox Register 13 (C0MB13)				8/16/32
H'FFFF 60DC	CAN0 Mailbox Register 13 (C0MB13)				8/16/32
H'FFFF 60E0	CAN0 Mailbox Register 14 (C0MB14)				8/16/32
H'FFFF 60E4	CAN0 Mailbox Register 14 (C0MB14)				8/16/32
H'FFFF 60E8	CAN0 Mailbox Register 14 (C0MB14)				8/16/32
H'FFFF 60EC	CAN0 Mailbox Register 14 (C0MB14)				8/16/32
H'FFFF 60F0	CAN0 Mailbox Register 15 (C0MB15)				8/16/32
H'FFFF 60F4	CAN0 Mailbox Register 15 (C0MB15)				8/16/32
H'FFFF 60F8	CAN0 Mailbox Register 15 (C0MB15)				8/16/32
H'FFFF 60FC	CAN0 Mailbox Register 15 (C0MB15)				8/16/32
H'FFFF 6100	CAN0 Mailbox Register 16 (C0MB16)				8/16/32
H'FFFF 6104	CAN0 Mailbox Register 16 (C0MB16)				8/16/32
H'FFFF 6108	CAN0 Mailbox Register 16 (C0MB16)				8/16/32
H'FFFF 610C	CAN0 Mailbox Register 16 (C0MB16)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 6110	CAN0 Mailbox Register 17 (C0MB17)				8/16/32
H'FFFF 6114	CAN0 Mailbox Register 17 (C0MB17)				8/16/32
H'FFFF 6118	CAN0 Mailbox Register 17 (C0MB17)				8/16/32
H'FFFF 611C	CAN0 Mailbox Register 17 (C0MB17)				8/16/32
H'FFFF 6120	CAN0 Mailbox Register 18 (C0MB18)				8/16/32
H'FFFF 6124	CAN0 Mailbox Register 18 (C0MB18)				8/16/32
H'FFFF 6128	CAN0 Mailbox Register 18 (C0MB18)				8/16/32
H'FFFF 612C	CAN0 Mailbox Register 18 (C0MB18)				8/16/32
H'FFFF 6130	CAN0 Mailbox Register 19 (C0MB19)				8/16/32
H'FFFF 6134	CAN0 Mailbox Register 19 (C0MB19)				8/16/32
H'FFFF 6138	CAN0 Mailbox Register 19 (C0MB19)				8/16/32
H'FFFF 613C	CAN0 Mailbox Register 19 (C0MB19)				8/16/32
H'FFFF 6140	CAN0 Mailbox Register 20 (C0MB20)				8/16/32
H'FFFF 6144	CAN0 Mailbox Register 20 (C0MB20)				8/16/32
H'FFFF 6148	CAN0 Mailbox Register 20 (C0MB20)				8/16/32
H'FFFF 614C	CAN0 Mailbox Register 20 (C0MB20)				8/16/32
H'FFFF 6150	CAN0 Mailbox Register 21 (C0MB21)				8/16/32
H'FFFF 6154	CAN0 Mailbox Register 21 (C0MB21)				8/16/32
H'FFFF 6158	CAN0 Mailbox Register 21 (C0MB21)				8/16/32
H'FFFF 615C	CAN0 Mailbox Register 21 (C0MB21)				8/16/32
H'FFFF 6160	CAN0 Mailbox Register 22 (C0MB22)				8/16/32
H'FFFF 6164	CAN0 Mailbox Register 22 (C0MB22)				8/16/32
H'FFFF 6168	CAN0 Mailbox Register 22 (C0MB22)				8/16/32
H'FFFF 616C	CAN0 Mailbox Register 22 (C0MB22)				8/16/32
H'FFFF 6170	CAN0 Mailbox Register 23 (C0MB23)				8/16/32
H'FFFF 6174	CAN0 Mailbox Register 23 (C0MB23)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 6178	CAN0 Mailbox Register 23 (C0MB23)				8/16/32
H'FFFF 617C	CAN0 Mailbox Register 23 (C0MB23)				8/16/32
H'FFFF 6180	CAN0 Mailbox Register 24 (C0MB24)				8/16/32
H'FFFF 6184	CAN0 Mailbox Register 24 (C0MB24)				8/16/32
H'FFFF 6188	CAN0 Mailbox Register 24 (C0MB24)				8/16/32
H'FFFF 618C	CAN0 Mailbox Register 24 (C0MB24)				8/16/32
H'FFFF 6190	CAN0 Mailbox Register 25 (C0MB25)				8/16/32
H'FFFF 6194	CAN0 Mailbox Register 25 (C0MB25)				8/16/32
H'FFFF 6198	CAN0 Mailbox Register 25 (C0MB25)				8/16/32
H'FFFF 619C	CAN0 Mailbox Register 25 (C0MB25)				8/16/32
H'FFFF 61A0	CAN0 Mailbox Register 26 (C0MB26)				8/16/32
H'FFFF 61A4	CAN0 Mailbox Register 26 (C0MB26)				8/16/32
H'FFFF 61A8	CAN0 Mailbox Register 26 (C0MB26)				8/16/32
H'FFFF 61AC	CAN0 Mailbox Register 26 (C0MB26)				8/16/32
H'FFFF 61B0	CAN0 Mailbox Register 27 (C0MB27)				8/16/32
H'FFFF 61B4	CAN0 Mailbox Register 27 (C0MB27)				8/16/32
H'FFFF 61B8	CAN0 Mailbox Register 27 (C0MB27)				8/16/32
H'FFFF 61BC	CAN0 Mailbox Register 27 (C0MB27)				8/16/32
H'FFFF 61C0	CAN0 Mailbox Register 28 (C0MB28)				8/16/32
H'FFFF 61C4	CAN0 Mailbox Register 28 (C0MB28)				8/16/32
H'FFFF 61C8	CAN0 Mailbox Register 28 (C0MB28)				8/16/32
H'FFFF 61CC	CAN0 Mailbox Register 28 (C0MB28)				8/16/32
H'FFFF 61D0	CAN0 Mailbox Register 29 (C0MB29)				8/16/32
H'FFFF 61D4	CAN0 Mailbox Register 29 (C0MB29)				8/16/32
H'FFFF 61D8	CAN0 Mailbox Register 29 (C0MB29)				8/16/32
H'FFFF 61DC	CAN0 Mailbox Register 29 (C0MB29)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 61E0	CAN0 Mailbox Register 30 (C0MB30)				8/16/32
H'FFFF 61E4	CAN0 Mailbox Register 30 (C0MB30)				8/16/32
H'FFFF 61E8	CAN0 Mailbox Register 30 (C0MB30)				8/16/32
H'FFFF 61EC	CAN0 Mailbox Register 30 (C0MB30)				8/16/32
H'FFFF 61F0	CAN0 Mailbox Register 31 (C0MB31)				8/16/32
H'FFFF 61F4	CAN0 Mailbox Register 31 (C0MB31)				8/16/32
H'FFFF 61F8	CAN0 Mailbox Register 31 (C0MB31)				8/16/32
H'FFFF 61FC	CAN0 Mailbox Register 31 (C0MB31)				8/16/32
H'FFFF 6200	CAN0 Mailbox Register 32 (C0MB32)				8/16/32
H'FFFF 6204	CAN0 Mailbox Register 32 (C0MB32)				8/16/32
H'FFFF 6208	CAN0 Mailbox Register 32 (C0MB32)				8/16/32
H'FFFF 620C	CAN0 Mailbox Register 32 (C0MB32)				8/16/32
H'FFFF 6210	CAN0 Mailbox Register 33 (C0MB33)				8/16/32
H'FFFF 6214	CAN0 Mailbox Register 33 (C0MB33)				8/16/32
H'FFFF 6218	CAN0 Mailbox Register 33 (C0MB33)				8/16/32
H'FFFF 621C	CAN0 Mailbox Register 33 (C0MB33)				8/16/32
H'FFFF 6220	CAN0 Mailbox Register 34 (C0MB34)				8/16/32
H'FFFF 6224	CAN0 Mailbox Register 34 (C0MB34)				8/16/32
H'FFFF 6228	CAN0 Mailbox Register 34 (C0MB34)				8/16/32
H'FFFF 622C	CAN0 Mailbox Register 34 (C0MB34)				8/16/32
H'FFFF 6230	CAN0 Mailbox Register 35 (C0MB35)				8/16/32
H'FFFF 6234	CAN0 Mailbox Register 35 (C0MB35)				8/16/32
H'FFFF 6238	CAN0 Mailbox Register 35 (C0MB35)				8/16/32
H'FFFF 623C	CAN0 Mailbox Register 35 (C0MB35)				8/16/32
H'FFFF 6240	CAN0 Mailbox Register 36 (C0MB36)				8/16/32
H'FFFF 6244	CAN0 Mailbox Register 36 (C0MB36)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 6248	CAN0 Mailbox Register 36 (C0MB36)				8/16/32
H'FFFF 624C	CAN0 Mailbox Register 36 (C0MB36)				8/16/32
H'FFFF 6250	CAN0 Mailbox Register 37 (C0MB37)				8/16/32
H'FFFF 6254	CAN0 Mailbox Register 37 (C0MB37)				8/16/32
H'FFFF 6258	CAN0 Mailbox Register 37 (C0MB37)				8/16/32
H'FFFF 625C	CAN0 Mailbox Register 37 (C0MB37)				8/16/32
H'FFFF 6260	CAN0 Mailbox Register 38 (C0MB38)				8/16/32
H'FFFF 6264	CAN0 Mailbox Register 38 (C0MB38)				8/16/32
H'FFFF 6268	CAN0 Mailbox Register 38 (C0MB38)				8/16/32
H'FFFF 626C	CAN0 Mailbox Register 38 (C0MB38)				8/16/32
H'FFFF 6270	CAN0 Mailbox Register 39 (C0MB39)				8/16/32
H'FFFF 6274	CAN0 Mailbox Register 39 (C0MB39)				8/16/32
H'FFFF 6278	CAN0 Mailbox Register 39 (C0MB39)				8/16/32
H'FFFF 627C	CAN0 Mailbox Register 39 (C0MB39)				8/16/32
H'FFFF 6280	CAN0 Mailbox Register 40 (C0MB40)				8/16/32
H'FFFF 6284	CAN0 Mailbox Register 40 (C0MB40)				8/16/32
H'FFFF 6288	CAN0 Mailbox Register 40 (C0MB40)				8/16/32
H'FFFF 628C	CAN0 Mailbox Register 40 (C0MB40)				8/16/32
H'FFFF 6290	CAN0 Mailbox Register 41 (C0MB41)				8/16/32
H'FFFF 6294	CAN0 Mailbox Register 41 (C0MB41)				8/16/32
H'FFFF 6298	CAN0 Mailbox Register 41 (C0MB41)				8/16/32
H'FFFF 629C	CAN0 Mailbox Register 41 (C0MB41)				8/16/32
H'FFFF 62A0	CAN0 Mailbox Register 42 (C0MB42)				8/16/32
H'FFFF 62A4	CAN0 Mailbox Register 42 (C0MB42)				8/16/32
H'FFFF 62A8	CAN0 Mailbox Register 42 (C0MB42)				8/16/32
H'FFFF 62AC	CAN0 Mailbox Register 42 (C0MB42)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 62B0	CAN0 Mailbox Register 43 (C0MB43)				8/16/32
H'FFFF 62B4	CAN0 Mailbox Register 43 (C0MB43)				8/16/32
H'FFFF 62B8	CAN0 Mailbox Register 43 (C0MB43)				8/16/32
H'FFFF 62BC	CAN0 Mailbox Register 43 (C0MB43)				8/16/32
H'FFFF 62C0	CAN0 Mailbox Register 44 (C0MB44)				8/16/32
H'FFFF 62C4	CAN0 Mailbox Register 44 (C0MB44)				8/16/32
H'FFFF 62C8	CAN0 Mailbox Register 44 (C0MB44)				8/16/32
H'FFFF 62CC	CAN0 Mailbox Register 44 (C0MB44)				8/16/32
H'FFFF 62D0	CAN0 Mailbox Register 45 (C0MB45)				8/16/32
H'FFFF 62D4	CAN0 Mailbox Register 45 (C0MB45)				8/16/32
H'FFFF 62D8	CAN0 Mailbox Register 45 (C0MB45)				8/16/32
H'FFFF 62DC	CAN0 Mailbox Register 45 (C0MB45)				8/16/32
H'FFFF 62E0	CAN0 Mailbox Register 46 (C0MB46)				8/16/32
H'FFFF 62E4	CAN0 Mailbox Register 46 (C0MB46)				8/16/32
H'FFFF 62E8	CAN0 Mailbox Register 46 (C0MB46)				8/16/32
H'FFFF 62EC	CAN0 Mailbox Register 46 (C0MB46)				8/16/32
H'FFFF 62F0	CAN0 Mailbox Register 47 (C0MB47)				8/16/32
H'FFFF 62F4	CAN0 Mailbox Register 47 (C0MB47)				8/16/32
H'FFFF 62F8	CAN0 Mailbox Register 47 (C0MB47)				8/16/32
H'FFFF 62FC	CAN0 Mailbox Register 47 (C0MB47)				8/16/32
H'FFFF 6300	CAN0 Mailbox Register 48 (C0MB48)				8/16/32
H'FFFF 6304	CAN0 Mailbox Register 48 (C0MB48)				8/16/32
H'FFFF 6308	CAN0 Mailbox Register 48 (C0MB48)				8/16/32
H'FFFF 630C	CAN0 Mailbox Register 48 (C0MB48)				8/16/32
H'FFFF 6310	CAN0 Mailbox Register 49 (C0MB49)				8/16/32
H'FFFF 6314	CAN0 Mailbox Register 49 (C0MB49)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 6318	CAN0 Mailbox Register 49 (C0MB49)				8/16/32
H'FFFF 631C	CAN0 Mailbox Register 49 (C0MB49)				8/16/32
H'FFFF 6320	CAN0 Mailbox Register 50 (C0MB50)				8/16/32
H'FFFF 6324	CAN0 Mailbox Register 50 (C0MB50)				8/16/32
H'FFFF 6328	CAN0 Mailbox Register 50 (C0MB50)				8/16/32
H'FFFF 632C	CAN0 Mailbox Register 50 (C0MB50)				8/16/32
H'FFFF 6330	CAN0 Mailbox Register 51 (C0MB51)				8/16/32
H'FFFF 6334	CAN0 Mailbox Register 51 (C0MB51)				8/16/32
H'FFFF 6338	CAN0 Mailbox Register 51 (C0MB51)				8/16/32
H'FFFF 633C	CAN0 Mailbox Register 51 (C0MB51)				8/16/32
H'FFFF 6340	CAN0 Mailbox Register 52 (C0MB52)				8/16/32
H'FFFF 6344	CAN0 Mailbox Register 52 (C0MB52)				8/16/32
H'FFFF 6348	CAN0 Mailbox Register 52 (C0MB52)				8/16/32
H'FFFF 634C	CAN0 Mailbox Register 52 (C0MB52)				8/16/32
H'FFFF 6350	CAN0 Mailbox Register 53 (C0MB53)				8/16/32
H'FFFF 6354	CAN0 Mailbox Register 53 (C0MB53)				8/16/32
H'FFFF 6358	CAN0 Mailbox Register 53 (C0MB53)				8/16/32
H'FFFF 635C	CAN0 Mailbox Register 53 (C0MB53)				8/16/32
H'FFFF 6360	CAN0 Mailbox Register 54 (C0MB54)				8/16/32
H'FFFF 6364	CAN0 Mailbox Register 54 (C0MB54)				8/16/32
H'FFFF 6368	CAN0 Mailbox Register 54 (C0MB54)				8/16/32
H'FFFF 636C	CAN0 Mailbox Register 54 (C0MB54)				8/16/32
H'FFFF 6370	CAN0 Mailbox Register 55 (C0MB55)				8/16/32
H'FFFF 6374	CAN0 Mailbox Register 55 (C0MB55)				8/16/32
H'FFFF 6378	CAN0 Mailbox Register 55 (C0MB55)				8/16/32
H'FFFF 637C	CAN0 Mailbox Register 55 (C0MB55)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 6380	CAN0 Mailbox Register 56 (C0MB56)				8/16/32
H'FFFF 6384	CAN0 Mailbox Register 56 (C0MB56)				8/16/32
H'FFFF 6388	CAN0 Mailbox Register 56 (C0MB56)				8/16/32
H'FFFF 638C	CAN0 Mailbox Register 56 (C0MB56)				8/16/32
H'FFFF 6390	CAN0 Mailbox Register 57 (C0MB57)				8/16/32
H'FFFF 6394	CAN0 Mailbox Register 57 (C0MB57)				8/16/32
H'FFFF 6398	CAN0 Mailbox Register 57 (C0MB57)				8/16/32
H'FFFF 639C	CAN0 Mailbox Register 57 (C0MB57)				8/16/32
H'FFFF 63A0	CAN0 Mailbox Register 58 (C0MB58)				8/16/32
H'FFFF 63A4	CAN0 Mailbox Register 58 (C0MB58)				8/16/32
H'FFFF 63A8	CAN0 Mailbox Register 58 (C0MB58)				8/16/32
H'FFFF 63AC	CAN0 Mailbox Register 58 (C0MB58)				8/16/32
H'FFFF 63B0	CAN0 Mailbox Register 59 (C0MB59)				8/16/32
H'FFFF 63B5	CAN0 Mailbox Register 59 (C0MB59)				8/16/32
H'FFFF 63B8	CAN0 Mailbox Register 59 (C0MB59)				8/16/32
H'FFFF 63BC	CAN0 Mailbox Register 59 (C0MB59)				8/16/32
H'FFFF 63C0	CAN0 Mailbox Register 60 (C0MB60)				8/16/32
H'FFFF 63C4	CAN0 Mailbox Register 60 (C0MB60)				8/16/32
H'FFFF 63C8	CAN0 Mailbox Register 60 (C0MB60)				8/16/32
H'FFFF 63CC	CAN0 Mailbox Register 60 (C0MB60)				8/16/32
H'FFFF 63D0	CAN0 Mailbox Register 61 (C0MB61)				8/16/32
H'FFFF 63D4	CAN0 Mailbox Register 61 (C0MB61)				8/16/32
H'FFFF 63D8	CAN0 Mailbox Register 61 (C0MB61)				8/16/32
H'FFFF 63DC	CAN0 Mailbox Register 61 (C0MB61)				8/16/32
H'FFFF 63E0	CAN0 Mailbox Register 62 (C0MB62)				8/16/32
H'FFFF 63E4	CAN0 Mailbox Register 62 (C0MB62)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 63E8	CAN0 Mailbox Register 62 (C0MB62)				8/16/32
H'FFFF 63EC	CAN0 Mailbox Register 62 (C0MB62)				8/16/32
H'FFFF 63F0	CAN0 Mailbox Register 63 (C0MB63)				8/16/32
H'FFFF 63F4	CAN0 Mailbox Register 63 (C0MB63)				8/16/32
H'FFFF 63F8	CAN0 Mailbox Register 63 (C0MB63)				8/16/32
H'FFFF 63FC	CAN0 Mailbox Register 63 (C0MB63)				8/16/32
H'FFFF 6400	CAN0 Mask Register 2 (C0MKR2)				8/16/32
H'FFFF 6404	CAN0 Mask Register 3 (C0MKR3)				8/16/32
H'FFFF 6408	CAN0 Mask Register 4 (C0MKR4)				8/16/32
H'FFFF 640C	CAN0 Mask Register 5 (C0MKR5)				8/16/32
H'FFFF 6410	CAN0 Mask Register 6 (C0MKR6)				8/16/32
H'FFFF 6414	CAN0 Mask Register 7 (C0MKR7)				8/16/32
H'FFFF 6418	CAN0 Mask Register 8 (C0MKR8)				8/16/32
H'FFFF 641C	CAN0 Mask Register 9 (C0MKR9)				8/16/32
H'FFFF 6420	CAN0 FIFO Received ID Compare Register 0 (C0FIDCR0)				8/16/32
H'FFFF 6424	CAN0 FIFO Received ID Compare Register 1 (C0FIDCR1)				8/16/32
H'FFFF 6428	CAN0 Mask Invalid Register 1 (C0MKIVLR1)				8/16/32
H'FFFF 642C	CAN0 Mailbox Interrupt Enable Register 1 (C0MIER1)				8/16/32
H'FFFF 6430	CAN0 Mask Register 0 (C0MKR0)				8/16/32
H'FFFF 6434	CAN0 Mask Register 1 (C0MKR1)				8/16/32
H'FFFF 6438	CAN0 Mask Invalid Register 0 (C0MKIVLR0)				8/16/32
H'FFFF 643C	CAN0 Mailbox Interrupt Enable Register 0 (C0MIER0)				8/16/32
:	(Reserved)				-
H'FFFF 6800	CAN0 Message Control Register 0 (C0MCTL0)	CAN0 Message Control Register 1 (C0MCTL1)	CAN0 Message Control Register 2 (C0MCTL2)	CAN0 Message Control Register 3 (C0MCTL3)	8/16/32
H'FFFF 6804	CAN0 Message Control Register 4 (C0MCTL4)	CAN0 Message Control Register 5 (C0MCTL5)	CAN0 Message Control Register 6 (C0MCTL6)	CAN0 Message Control Register 7 (C0MCTL7)	8/16/32
H'FFFF 6808	CAN0 Message Control Register 8 (C0MCTL8)	CAN0 Message Control Register 9 (C0MCTL9)	CAN0 Message Control Register 10 (C0MCTL10)	CAN0 Message Control Register 11 (C0MCTL11)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 680C	CAN0 Message Control Register 12 (C0MCTL12)	CAN0 Message Control Register 13 (C0MCTL13)	CAN0 Message Control Register 14 (C0MCTL14)	CAN0 Message Control Register 15 (C0MCTL15)	8/16/32
H'FFFF 6810	CAN0 Message Control Register 16 (C0MCTL16)	CAN0 Message Control Register 17 (C0MCTL17)	CAN0 Message Control Register 18 (C0MCTL18)	CAN0 Message Control Register 19 (C0MCTL19)	8/16/32
H'FFFF 6814	CAN0 Message Control Register 20 (C0MCTL20)	CAN0 Message Control Register 21 (C0MCTL21)	CAN0 Message Control Register 22 (C0MCTL22)	CAN0 Message Control Register 23 (C0MCTL23)	8/16/32
H'FFFF 6818	CAN0 Message Control Register 24 (C0MCTL24)	CAN0 Message Control Register 25 (C0MCTL25)	CAN0 Message Control Register 26 (C0MCTL26)	CAN0 Message Control Register 27 (C0MCTL27)	8/16/32
H'FFFF 681C	CAN0 Message Control Register 28 (C0MCTL28)	CAN0 Message Control Register 29 (C0MCTL29)	CAN0 Message Control Register 30 (C0MCTL30)	CAN0 Message Control Register 31 (C0MCTL31)	8/16/32
H'FFFF 6820	CAN0 Message Control Register 32 (C0MCTL32)	CAN0 Message Control Register 33 (C0MCTL33)	CAN0 Message Control Register 34 (C0MCTL34)	CAN0 Message Control Register 35 (C0MCTL35)	8/16/32
H'FFFF 6824	CAN0 Message Control Register 36 (C0MCTL36)	CAN0 Message Control Register 37 (C0MCTL37)	CAN0 Message Control Register 38 (C0MCTL38)	CAN0 Message Control Register 39 (C0MCTL39)	8/16/32
H'FFFF 6828	CAN0 Message Control Register 40 (C0MCTL40)	CAN0 Message Control Register 41 (C0MCTL41)	CAN0 Message Control Register 42 (C0MCTL42)	CAN0 Message Control Register 43 (C0MCTL43)	8/16/32
H'FFFF 682C	CAN0 Message Control Register 44 (C0MCTL44)	CAN0 Message Control Register 45 (C0MCTL45)	CAN0 Message Control Register 46 (C0MCTL46)	CAN0 Message Control Register 47 (C0MCTL47)	8/16/32
H'FFFF 6830	CAN0 Message Control Register 48 (C0MCTL48)	CAN0 Message Control Register 49 (C0MCTL49)	CAN0 Message Control Register 50 (C0MCTL50)	CAN0 Message Control Register 51 (C0MCTL51)	8/16/32
H'FFFF 6834	CAN0 Message Control Register 52 (C0MCTL52)	CAN0 Message Control Register 53 (C0MCTL53)	CAN0 Message Control Register 54 (C0MCTL54)	CAN0 Message Control Register 55 (C0MCTL55)	8/16/32
H'FFFF 6838	CAN0 Message Control Register 56 (C0MCTL56)	CAN0 Message Control Register 57 (C0MCTL57)	CAN0 Message Control Register 58 (C0MCTL58)	CAN0 Message Control Register 59 (C0MCTL59)	8/16/32
H'FFFF 683C	CAN0 Message Control Register 60 (C0MCTL60)	CAN0 Message Control Register 61 (C0MCTL61)	CAN0 Message Control Register 62 (C0MCTL62)	CAN0 Message Control Register 63 (C0MCTL63)	8/16/32
H'FFFF 6840	CAN0 Control Register (C0CTLR)		CAN0 Status Register (C0STR)		8/16/32
H'FFFF 6844	CAN0 Bit Configuration Register (C0BCR)			CAN0 Clock Select Register (C0CLKR)	8/16/32
H'FFFF 6848	CAN0 Receive FIFO Control Register (C0RFCR)	CAN0 Receive FIFO Pointer Control Register (C0RFPCR)	CAN0 Transmit FIFO Control Register (C0TFCR)	CAN0 Transmit FIFO Pointer Control Register (C0TFPCR)	8/16/32
H'FFFF 684C	CAN0 Error Interrupt Enable Register (C0EIER)	CAN0 Error Interrupt Factor Judge Register (C0EIFR)	CAN0 Receive Error Count Register (C0RECR)	CAN0 Transmit Error Count Register (C0TECR)	8/16/32
H'FFFF 6850	CAN0 Error Code Store Register (C0ECSR)	CAN0 Channel Search Support Register (C0CSSR)	CAN0 Mailbox Search Status Register (C0MSSR)	CAN0 Mailbox Search Mode Register (C0MSMR)	8/16/32
H'FFFF 6854	CAN0 Time Stamp Register (C0TSR)		CAN0 Acceptance Filter Support Register (C0AFSR)		8/16/32
H'FFFF 6858	CAN0 Test Control Register (C0TCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF 6860	CAN0 Interrupt Enable Register (C0IER)	CAN0 Interrupt Status Register (C0ISR)	(Reserved)	CAN0 Mailbox Search Mask Register (C0MBSMR)	8/16, 8/16, -, 8
:	(Reserved)				-
H'FFFF 7000	CAN1 Mailbox Register 0 (C1MB0)				8/16/32
H'FFFF 7004	CAN1 Mailbox Register 0 (C1MB0)				8/16/32
H'FFFF 7008	CAN1 Mailbox Register 0 (C1MB0)				8/16/32
H'FFFF 700C	CAN1 Mailbox Register 0 (C1MB0)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7010	CAN1 Mailbox Register 1 (C1MB1)				8/16/32
H'FFFF 7014	CAN1 Mailbox Register 1 (C1MB1)				8/16/32
H'FFFF 7018	CAN1 Mailbox Register 1 (C1MB1)				8/16/32
H'FFFF 701C	CAN1 Mailbox Register 1 (C1MB1)				8/16/32
H'FFFF 7020	CAN1 Mailbox Register 2 (C1MB2)				8/16/32
H'FFFF 7024	CAN1 Mailbox Register 2 (C1MB2)				8/16/32
H'FFFF 7028	CAN1 Mailbox Register 2 (C1MB2)				8/16/32
H'FFFF 702C	CAN1 Mailbox Register 2 (C1MB2)				8/16/32
H'FFFF 7030	CAN1 Mailbox Register 3 (C1MB3)				8/16/32
H'FFFF 7034	CAN1 Mailbox Register 3 (C1MB3)				8/16/32
H'FFFF 7038	CAN1 Mailbox Register 3 (C1MB3)				8/16/32
H'FFFF 703C	CAN1 Mailbox Register 3 (C1MB3)				8/16/32
H'FFFF 7040	CAN1 Mailbox Register 4 (C1MB4)				8/16/32
H'FFFF 7044	CAN1 Mailbox Register 4 (C1MB4)				8/16/32
H'FFFF 7048	CAN1 Mailbox Register 4 (C1MB4)				8/16/32
H'FFFF 704C	CAN1 Mailbox Register 4 (C1MB4)				8/16/32
H'FFFF 7050	CAN1 Mailbox Register 5 (C1MB5)				8/16/32
H'FFFF 7054	CAN1 Mailbox Register 5 (C1MB5)				8/16/32
H'FFFF 7058	CAN1 Mailbox Register 5 (C1MB5)				8/16/32
H'FFFF 705C	CAN1 Mailbox Register 5 (C1MB5)				8/16/32
H'FFFF 7060	CAN1 Mailbox Register 6 (C1MB6)				8/16/32
H'FFFF 7064	CAN1 Mailbox Register 6 (C1MB6)				8/16/32
H'FFFF 7068	CAN1 Mailbox Register 6 (C1MB6)				8/16/32
H'FFFF 706C	CAN1 Mailbox Register 6 (C1MB6)				8/16/32
H'FFFF 7070	CAN1 Mailbox Register 7 (C1MB7)				8/16/32
H'FFFF 7074	CAN1 Mailbox Register 7 (C1MB7)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7078	CAN1 Mailbox Register 7 (C1MB7)				8/16/32
H'FFFF 707C	CAN1 Mailbox Register 7 (C1MB7)				8/16/32
H'FFFF 7080	CAN1 Mailbox Register 8 (C1MB8)				8/16/32
H'FFFF 7084	CAN1 Mailbox Register 8 (C1MB8)				8/16/32
H'FFFF 7088	CAN1 Mailbox Register 8 (C1MB8)				8/16/32
H'FFFF 708C	CAN1 Mailbox Register 8 (C1MB8)				8/16/32
H'FFFF 7090	CAN1 Mailbox Register 9 (C1MB9)				8/16/32
H'FFFF 7094	CAN1 Mailbox Register 9 (C1MB9)				8/16/32
H'FFFF 7098	CAN1 Mailbox Register 9 (C1MB9)				8/16/32
H'FFFF 709C	CAN1 Mailbox Register 9 (C1MB9)				8/16/32
H'FFFF 70A0	CAN1 Mailbox Register 10 (C1MB10)				8/16/32
H'FFFF 70A4	CAN1 Mailbox Register 10 (C1MB10)				8/16/32
H'FFFF 70A8	CAN1 Mailbox Register 10 (C1MB10)				8/16/32
H'FFFF 70AC	CAN1 Mailbox Register 10 (C1MB10)				8/16/32
H'FFFF 70B0	CAN1 Mailbox Register 11 (C1MB11)				8/16/32
H'FFFF 70B4	CAN1 Mailbox Register 11 (C1MB11)				8/16/32
H'FFFF 70B8	CAN1 Mailbox Register 11 (C1MB11)				8/16/32
H'FFFF 70BC	CAN1 Mailbox Register 11 (C1MB11)				8/16/32
H'FFFF 70C0	CAN1 Mailbox Register 12 (C1MB12)				8/16/32
H'FFFF 70C4	CAN1 Mailbox Register 12 (C1MB12)				8/16/32
H'FFFF 70C8	CAN1 Mailbox Register 12 (C1MB12)				8/16/32
H'FFFF 70CC	CAN1 Mailbox Register 12 (C1MB12)				8/16/32
H'FFFF 70D0	CAN1 Mailbox Register 13 (C1MB13)				8/16/32
H'FFFF 70D4	CAN1 Mailbox Register 13 (C1MB13)				8/16/32
H'FFFF 70D8	CAN1 Mailbox Register 13 (C1MB13)				8/16/32
H'FFFF 70DC	CAN1 Mailbox Register 13 (C1MB13)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 70E0	CAN1 Mailbox Register 14 (C1MB14)				8/16/32
H'FFFF 70E4	CAN1 Mailbox Register 14 (C1MB14)				8/16/32
H'FFFF 70E8	CAN1 Mailbox Register 14 (C1MB14)				8/16/32
H'FFFF 70EC	CAN1 Mailbox Register 14 (C1MB14)				8/16/32
H'FFFF 70F0	CAN1 Mailbox Register 15 (C1MB15)				8/16/32
H'FFFF 70F4	CAN1 Mailbox Register 15 (C1MB15)				8/16/32
H'FFFF 70F8	CAN1 Mailbox Register 15 (C1MB15)				8/16/32
H'FFFF 70FC	CAN1 Mailbox Register 15 (C1MB15)				8/16/32
H'FFFF 7100	CAN1 Mailbox Register 16 (C1MB16)				8/16/32
H'FFFF 7104	CAN1 Mailbox Register 16 (C1MB16)				8/16/32
H'FFFF 7108	CAN1 Mailbox Register 16 (C1MB16)				8/16/32
H'FFFF 710C	CAN1 Mailbox Register 16 (C1MB16)				8/16/32
H'FFFF 7110	CAN1 Mailbox Register 17 (C1MB17)				8/16/32
H'FFFF 7114	CAN1 Mailbox Register 17 (C1MB17)				8/16/32
H'FFFF 7118	CAN1 Mailbox Register 17 (C1MB17)				8/16/32
H'FFFF 711C	CAN1 Mailbox Register 17 (C1MB17)				8/16/32
H'FFFF 7120	CAN1 Mailbox Register 18 (C1MB18)				8/16/32
H'FFFF 7124	CAN1 Mailbox Register 18 (C1MB18)				8/16/32
H'FFFF 7128	CAN1 Mailbox Register 18 (C1MB18)				8/16/32
H'FFFF 712C	CAN1 Mailbox Register 18 (C1MB18)				8/16/32
H'FFFF 7130	CAN1 Mailbox Register 19 (C1MB19)				8/16/32
H'FFFF 7134	CAN1 Mailbox Register 19 (C1MB19)				8/16/32
H'FFFF 7138	CAN1 Mailbox Register 19 (C1MB19)				8/16/32
H'FFFF 713C	CAN1 Mailbox Register 19 (C1MB19)				8/16/32
H'FFFF 7140	CAN1 Mailbox Register 20 (C1MB20)				8/16/32
H'FFFF 7144	CAN1 Mailbox Register 20 (C1MB20)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7148	CAN1 Mailbox Register 20 (C1MB20)				8/16/32
H'FFFF 714C	CAN1 Mailbox Register 20 (C1MB20)				8/16/32
H'FFFF 7150	CAN1 Mailbox Register 21 (C1MB21)				8/16/32
H'FFFF 7154	CAN1 Mailbox Register 21 (C1MB21)				8/16/32
H'FFFF 7158	CAN1 Mailbox Register 21 (C1MB21)				8/16/32
H'FFFF 715C	CAN1 Mailbox Register 21 (C1MB21)				8/16/32
H'FFFF 7160	CAN1 Mailbox Register 22 (C1MB22)				8/16/32
H'FFFF 7164	CAN1 Mailbox Register 22 (C1MB22)				8/16/32
H'FFFF 7168	CAN1 Mailbox Register 22 (C1MB22)				8/16/32
H'FFFF 716C	CAN1 Mailbox Register 22 (C1MB22)				8/16/32
H'FFFF 7170	CAN1 Mailbox Register 23 (C1MB23)				8/16/32
H'FFFF 7174	CAN1 Mailbox Register 23 (C1MB23)				8/16/32
H'FFFF 7178	CAN1 Mailbox Register 23 (C1MB23)				8/16/32
H'FFFF 717C	CAN1 Mailbox Register 23 (C1MB23)				8/16/32
H'FFFF 7180	CAN1 Mailbox Register 24 (C1MB24)				8/16/32
H'FFFF 7184	CAN1 Mailbox Register 24 (C1MB24)				8/16/32
H'FFFF 7188	CAN1 Mailbox Register 24 (C1MB24)				8/16/32
H'FFFF 718C	CAN1 Mailbox Register 24 (C1MB24)				8/16/32
H'FFFF 7190	CAN1 Mailbox Register 25 (C1MB25)				8/16/32
H'FFFF 7194	CAN1 Mailbox Register 25 (C1MB25)				8/16/32
H'FFFF 7198	CAN1 Mailbox Register 25 (C1MB25)				8/16/32
H'FFFF 719C	CAN1 Mailbox Register 25 (C1MB25)				8/16/32
H'FFFF 71A0	CAN1 Mailbox Register 26 (C1MB26)				8/16/32
H'FFFF 71A4	CAN1 Mailbox Register 26 (C1MB26)				8/16/32
H'FFFF 71A8	CAN1 Mailbox Register 26 (C1MB26)				8/16/32
H'FFFF 71AC	CAN1 Mailbox Register 26 (C1MB26)				8/16/32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF 71B0	CAN1 Mailbox Register 27 (C1MB27)								8/16/32
H'FFFF 71B4	CAN1 Mailbox Register 27 (C1MB27)								8/16/32
H'FFFF 71B8	CAN1 Mailbox Register 27 (C1MB27)								8/16/32
H'FFFF 71BC	CAN1 Mailbox Register 27 (C1MB27)								8/16/32
H'FFFF 71C0	CAN1 Mailbox Register 28 (C1MB28)								8/16/32
H'FFFF 71C4	CAN1 Mailbox Register 28 (C1MB28)								8/16/32
H'FFFF 71C8	CAN1 Mailbox Register 28 (C1MB28)								8/16/32
H'FFFF 71CC	CAN1 Mailbox Register 28 (C1MB28)								8/16/32
H'FFFF 71D0	CAN1 Mailbox Register 29 (C1MB29)								8/16/32
H'FFFF 71D4	CAN1 Mailbox Register 29 (C1MB29)								8/16/32
H'FFFF 71D8	CAN1 Mailbox Register 29 (C1MB29)								8/16/32
H'FFFF 71DC	CAN1 Mailbox Register 29 (C1MB29)								8/16/32
H'FFFF 71E0	CAN1 Mailbox Register 30 (C1MB30)								8/16/32
H'FFFF 71E4	CAN1 Mailbox Register 30 (C1MB30)								8/16/32
H'FFFF 71E8	CAN1 Mailbox Register 30 (C1MB30)								8/16/32
H'FFFF 71EC	CAN1 Mailbox Register 30 (C1MB30)								8/16/32
H'FFFF 71F0	CAN1 Mailbox Register 31 (C1MB31)								8/16/32
H'FFFF 71F4	CAN1 Mailbox Register 31 (C1MB31)								8/16/32
H'FFFF 71F8	CAN1 Mailbox Register 31 (C1MB31)								8/16/32
H'FFFF 71FC	CAN1 Mailbox Register 31 (C1MB31)								8/16/32
H'FFFF 7200	CAN1 Mailbox Register 32 (C1MB32)								8/16/32
H'FFFF 7204	CAN1 Mailbox Register 32 (C1MB32)								8/16/32
H'FFFF 7208	CAN1 Mailbox Register 32 (C1MB32)								8/16/32
H'FFFF 720C	CAN1 Mailbox Register 32 (C1MB32)								8/16/32
H'FFFF 7210	CAN1 Mailbox Register 33 (C1MB33)								8/16/32
H'FFFF 7214	CAN1 Mailbox Register 33 (C1MB33)								8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7218	CAN1 Mailbox Register 33 (C1MB33)				8/16/32
H'FFFF 721C	CAN1 Mailbox Register 33 (C1MB33)				8/16/32
H'FFFF 7220	CAN1 Mailbox Register 34 (C1MB34)				8/16/32
H'FFFF 7224	CAN1 Mailbox Register 34 (C1MB34)				8/16/32
H'FFFF 7228	CAN1 Mailbox Register 34 (C1MB34)				8/16/32
H'FFFF 722C	CAN1 Mailbox Register 34 (C1MB34)				8/16/32
H'FFFF 7230	CAN1 Mailbox Register 35 (C1MB35)				8/16/32
H'FFFF 7234	CAN1 Mailbox Register 35 (C1MB35)				8/16/32
H'FFFF 7238	CAN1 Mailbox Register 35 (C1MB35)				8/16/32
H'FFFF 723C	CAN1 Mailbox Register 35 (C1MB35)				8/16/32
H'FFFF 7240	CAN1 Mailbox Register 36 (C1MB36)				8/16/32
H'FFFF 7244	CAN1 Mailbox Register 36 (C1MB36)				8/16/32
H'FFFF 7248	CAN1 Mailbox Register 36 (C1MB36)				8/16/32
H'FFFF 724C	CAN1 Mailbox Register 36 (C1MB36)				8/16/32
H'FFFF 7250	CAN1 Mailbox Register 37 (C1MB37)				8/16/32
H'FFFF 7254	CAN1 Mailbox Register 37 (C1MB37)				8/16/32
H'FFFF 7258	CAN1 Mailbox Register 37 (C1MB37)				8/16/32
H'FFFF 725C	CAN1 Mailbox Register 37 (C1MB37)				8/16/32
H'FFFF 7260	CAN1 Mailbox Register 38 (C1MB38)				8/16/32
H'FFFF 7264	CAN1 Mailbox Register 38 (C1MB38)				8/16/32
H'FFFF 7268	CAN1 Mailbox Register 38 (C1MB38)				8/16/32
H'FFFF 726C	CAN1 Mailbox Register 38 (C1MB38)				8/16/32
H'FFFF 7270	CAN1 Mailbox Register 39 (C1MB39)				8/16/32
H'FFFF 7274	CAN1 Mailbox Register 39 (C1MB39)				8/16/32
H'FFFF 7278	CAN1 Mailbox Register 39 (C1MB39)				8/16/32
H'FFFF 727C	CAN1 Mailbox Register 39 (C1MB39)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7280	CAN1 Mailbox Register 40 (C1MB40)				8/16/32
H'FFFF 7284	CAN1 Mailbox Register 40 (C1MB40)				8/16/32
H'FFFF 7288	CAN1 Mailbox Register 40 (C1MB40)				8/16/32
H'FFFF 728C	CAN1 Mailbox Register 40 (C1MB40)				8/16/32
H'FFFF 7290	CAN1 Mailbox Register 41 (C1MB41)				8/16/32
H'FFFF 7294	CAN1 Mailbox Register 41 (C1MB41)				8/16/32
H'FFFF 7298	CAN1 Mailbox Register 41 (C1MB41)				8/16/32
H'FFFF 729C	CAN1 Mailbox Register 41 (C1MB41)				8/16/32
H'FFFF 72A0	CAN1 Mailbox Register 42 (C1MB42)				8/16/32
H'FFFF 72A4	CAN1 Mailbox Register 42 (C1MB42)				8/16/32
H'FFFF 72A8	CAN1 Mailbox Register 42 (C1MB42)				8/16/32
H'FFFF 72AC	CAN1 Mailbox Register 42 (C1MB42)				8/16/32
H'FFFF 72B0	CAN1 Mailbox Register 43 (C1MB43)				8/16/32
H'FFFF 72B4	CAN1 Mailbox Register 43 (C1MB43)				8/16/32
H'FFFF 72B8	CAN1 Mailbox Register 43 (C1MB43)				8/16/32
H'FFFF 72BC	CAN1 Mailbox Register 43 (C1MB43)				8/16/32
H'FFFF 72C0	CAN1 Mailbox Register 44 (C1MB44)				8/16/32
H'FFFF 72C4	CAN1 Mailbox Register 44 (C1MB44)				8/16/32
H'FFFF 72C8	CAN1 Mailbox Register 44 (C1MB44)				8/16/32
H'FFFF 72CC	CAN1 Mailbox Register 44 (C1MB44)				8/16/32
H'FFFF 72D0	CAN1 Mailbox Register 45 (C1MB45)				8/16/32
H'FFFF 72D4	CAN1 Mailbox Register 45 (C1MB45)				8/16/32
H'FFFF 72D8	CAN1 Mailbox Register 45 (C1MB45)				8/16/32
H'FFFF 72DC	CAN1 Mailbox Register 45 (C1MB45)				8/16/32
H'FFFF 72E0	CAN1 Mailbox Register 46 (C1MB46)				8/16/32
H'FFFF 72E4	CAN1 Mailbox Register 46 (C1MB46)				8/16/32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF 72E8	CAN1 Mailbox Register 46 (C1MB46)								8/16/32
H'FFFF 72EC	CAN1 Mailbox Register 46 (C1MB46)								8/16/32
H'FFFF 72F0	CAN1 Mailbox Register 47 (C1MB47)								8/16/32
H'FFFF 72F4	CAN1 Mailbox Register 47 (C1MB47)								8/16/32
H'FFFF 72F8	CAN1 Mailbox Register 47 (C1MB47)								8/16/32
H'FFFF 72FC	CAN1 Mailbox Register 47 (C1MB47)								8/16/32
H'FFFF 7300	CAN1 Mailbox Register 48 (C1MB48)								8/16/32
H'FFFF 7304	CAN1 Mailbox Register 48 (C1MB48)								8/16/32
H'FFFF 7308	CAN1 Mailbox Register 48 (C1MB48)								8/16/32
H'FFFF 730C	CAN1 Mailbox Register 48 (C1MB48)								8/16/32
H'FFFF 7310	CAN1 Mailbox Register 49 (C1MB49)								8/16/32
H'FFFF 7314	CAN1 Mailbox Register 49 (C1MB49)								8/16/32
H'FFFF 7318	CAN1 Mailbox Register 49 (C1MB49)								8/16/32
H'FFFF 731C	CAN1 Mailbox Register 49 (C1MB49)								8/16/32
H'FFFF 7320	CAN1 Mailbox Register 50 (C1MB50)								8/16/32
H'FFFF 7324	CAN1 Mailbox Register 50 (C1MB50)								8/16/32
H'FFFF 7328	CAN1 Mailbox Register 50 (C1MB50)								8/16/32
H'FFFF 732C	CAN1 Mailbox Register 50 (C1MB50)								8/16/32
H'FFFF 7330	CAN1 Mailbox Register 51 (C1MB51)								8/16/32
H'FFFF 7334	CAN1 Mailbox Register 51 (C1MB51)								8/16/32
H'FFFF 7338	CAN1 Mailbox Register 51 (C1MB51)								8/16/32
H'FFFF 733C	CAN1 Mailbox Register 51 (C1MB51)								8/16/32
H'FFFF 7340	CAN1 Mailbox Register 52 (C1MB52)								8/16/32
H'FFFF 7344	CAN1 Mailbox Register 52 (C1MB52)								8/16/32
H'FFFF 7348	CAN1 Mailbox Register 52 (C1MB52)								8/16/32
H'FFFF 734C	CAN1 Mailbox Register 52 (C1MB52)								8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7350	CAN1 Mailbox Register 53 (C1MB53)				8/16/32
H'FFFF 7354	CAN1 Mailbox Register 53 (C1MB53)				8/16/32
H'FFFF 7358	CAN1 Mailbox Register 53 (C1MB53)				8/16/32
H'FFFF 735C	CAN1 Mailbox Register 53 (C1MB53)				8/16/32
H'FFFF 7360	CAN1 Mailbox Register 54 (C1MB54)				8/16/32
H'FFFF 7364	CAN1 Mailbox Register 54 (C1MB54)				8/16/32
H'FFFF 7368	CAN1 Mailbox Register 54 (C1MB54)				8/16/32
H'FFFF 736C	CAN1 Mailbox Register 54 (C1MB54)				8/16/32
H'FFFF 7370	CAN1 Mailbox Register 55 (C1MB55)				8/16/32
H'FFFF 7374	CAN1 Mailbox Register 55 (C1MB55)				8/16/32
H'FFFF 7378	CAN1 Mailbox Register 55 (C1MB55)				8/16/32
H'FFFF 737C	CAN1 Mailbox Register 55 (C1MB55)				8/16/32
H'FFFF 7380	CAN1 Mailbox Register 56 (C1MB56)				8/16/32
H'FFFF 7384	CAN1 Mailbox Register 56 (C1MB56)				8/16/32
H'FFFF 7388	CAN1 Mailbox Register 56 (C1MB56)				8/16/32
H'FFFF 738C	CAN1 Mailbox Register 56 (C1MB56)				8/16/32
H'FFFF 7390	CAN1 Mailbox Register 57 (C1MB57)				8/16/32
H'FFFF 7394	CAN1 Mailbox Register 57 (C1MB57)				8/16/32
H'FFFF 7398	CAN1 Mailbox Register 57 (C1MB57)				8/16/32
H'FFFF 739C	CAN1 Mailbox Register 57 (C1MB57)				8/16/32
H'FFFF 73A0	CAN1 Mailbox Register 58 (C1MB58)				8/16/32
H'FFFF 73A4	CAN1 Mailbox Register 58 (C1MB58)				8/16/32
H'FFFF 73A8	CAN1 Mailbox Register 58 (C1MB58)				8/16/32
H'FFFF 73AC	CAN1 Mailbox Register 58 (C1MB58)				8/16/32
H'FFFF 73B0	CAN1 Mailbox Register 59 (C1MB59)				8/16/32
H'FFFF 73B5	CAN1 Mailbox Register 59 (C1MB59)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 73B8	CAN1 Mailbox Register 59 (C1MB59)				8/16/32
H'FFFF 73BC	CAN1 Mailbox Register 59 (C1MB59)				8/16/32
H'FFFF 73C0	CAN1 Mailbox Register 60 (C1MB60)				8/16/32
H'FFFF 73C4	CAN1 Mailbox Register 60 (C1MB60)				8/16/32
H'FFFF 73C8	CAN1 Mailbox Register 60 (C1MB60)				8/16/32
H'FFFF 73CC	CAN1 Mailbox Register 60 (C1MB60)				8/16/32
H'FFFF 73D0	CAN1 Mailbox Register 61 (C1MB61)				8/16/32
H'FFFF 73D4	CAN1 Mailbox Register 61 (C1MB61)				8/16/32
H'FFFF 73D8	CAN1 Mailbox Register 61 (C1MB61)				8/16/32
H'FFFF 73DC	CAN1 Mailbox Register 61 (C1MB61)				8/16/32
H'FFFF 73E0	CAN1 Mailbox Register 62 (C1MB62)				8/16/32
H'FFFF 73E4	CAN1 Mailbox Register 62 (C1MB62)				8/16/32
H'FFFF 73E8	CAN1 Mailbox Register 62 (C1MB62)				8/16/32
H'FFFF 73EC	CAN1 Mailbox Register 62 (C1MB62)				8/16/32
H'FFFF 73F0	CAN1 Mailbox Register 63 (C1MB63)				8/16/32
H'FFFF 73F4	CAN1 Mailbox Register 63 (C1MB63)				8/16/32
H'FFFF 73F8	CAN1 Mailbox Register 63 (C1MB63)				8/16/32
H'FFFF 73FC	CAN1 Mailbox Register 63 (C1MB63)				8/16/32
H'FFFF 7400	CAN1 Mask Register 2 (C1MKR2)				8/16/32
H'FFFF 7404	CAN1 Mask Register 3 (C1MKR3)				8/16/32
H'FFFF 7408	CAN1 Mask Register 4 (C1MKR4)				8/16/32
H'FFFF 740C	CAN1 Mask Register 5 (C1MKR5)				8/16/32
H'FFFF 7410	CAN1 Mask Register 6 (C1MKR6)				8/16/32
H'FFFF 7414	CAN1 Mask Register 7 (C1MKR7)				8/16/32
H'FFFF 7418	CAN1 Mask Register 8 (C1MKR8)				8/16/32
H'FFFF 741C	CAN1 Mask Register 9 (C1MKR9)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7420	CAN1 FIFO Received ID Compare Register 0 (C1FIDCR0)				8/16/32
H'FFFF 7424	CAN1 FIFO Received ID Compare Register 1 (C1FIDCR1)				8/16/32
H'FFFF 7428	CAN1 Mask Invalid Register 1 (C1MKIVLR1)				8/16/32
H'FFFF 742C	CAN1 Mailbox Interrupt Enable Register 1 (C1MIER1)				8/16/32
H'FFFF 7430	CAN1 Mask Register 0 (C1MKR0)				8/16/32
H'FFFF 7434	CAN1 Mask Register 1 (C1MKR1)				8/16/32
H'FFFF 7438	CAN1 Mask Invalid Register 0 (C1MKIVLR0)				8/16/32
H'FFFF 743C	CAN1 Mailbox Interrupt Enable Register 0 (C1MIER0)				8/16/32
:	(Reserved)				-
H'FFFF 7800	CAN1 Message Control Register 0 (C1MCTL0)	CAN1 Message Control Register 1 (C1MCTL1)	CAN1 Message Control Register 2 (C1MCTL2)	CAN1 Message Control Register 3 (C1MCTL3)	8/16/32
H'FFFF 7804	CAN1 Message Control Register 4 (C1MCTL4)	CAN1 Message Control Register 5 (C1MCTL5)	CAN1 Message Control Register 6 (C1MCTL6)	CAN1 Message Control Register 7 (C1MCTL7)	8/16/32
H'FFFF 7808	CAN1 Message Control Register 8 (C1MCTL8)	CAN1 Message Control Register 9 (C1MCTL9)	CAN1 Message Control Register 10 (C1MCTL10)	CAN1 Message Control Register 11 (C1MCTL11)	8/16/32
H'FFFF 780C	CAN1 Message Control Register 12 (C1MCTL12)	CAN1 Message Control Register 13 (C1MCTL13)	CAN1 Message Control Register 14 (C1MCTL14)	CAN1 Message Control Register 15 (C1MCTL15)	8/16/32
H'FFFF 7810	CAN1 Message Control Register 16 (C1MCTL16)	CAN1 Message Control Register 17 (C1MCTL17)	CAN1 Message Control Register 18 (C1MCTL18)	CAN1 Message Control Register 19 (C1MCTL19)	8/16/32
H'FFFF 7814	CAN1 Message Control Register 20 (C1MCTL20)	CAN1 Message Control Register 21 (C1MCTL21)	CAN1 Message Control Register 22 (C1MCTL22)	CAN1 Message Control Register 23 (C1MCTL23)	8/16/32
H'FFFF 7818	CAN1 Message Control Register 24 (C1MCTL24)	CAN1 Message Control Register 25 (C1MCTL25)	CAN1 Message Control Register 26 (C1MCTL26)	CAN1 Message Control Register 27 (C1MCTL27)	8/16/32
H'FFFF 781C	CAN1 Message Control Register 28 (C1MCTL28)	CAN1 Message Control Register 29 (C1MCTL29)	CAN1 Message Control Register 30 (C1MCTL30)	CAN1 Message Control Register 31 (C1MCTL31)	8/16/32
H'FFFF 7820	CAN1 Message Control Register 32 (C1MCTL32)	CAN1 Message Control Register 33 (C1MCTL33)	CAN1 Message Control Register 34 (C1MCTL34)	CAN1 Message Control Register 35 (C1MCTL35)	8/16/32
H'FFFF 7824	CAN1 Message Control Register 36 (C1MCTL36)	CAN1 Message Control Register 37 (C1MCTL37)	CAN1 Message Control Register 38 (C1MCTL38)	CAN1 Message Control Register 39 (C1MCTL39)	8/16/32
H'FFFF 7828	CAN1 Message Control Register 40 (C1MCTL40)	CAN1 Message Control Register 41 (C1MCTL41)	CAN1 Message Control Register 42 (C1MCTL42)	CAN1 Message Control Register 43 (C1MCTL43)	8/16/32
H'FFFF 782C	CAN1 Message Control Register 44 (C1MCTL44)	CAN1 Message Control Register 45 (C1MCTL45)	CAN1 Message Control Register 46 (C1MCTL46)	CAN1 Message Control Register 47 (C1MCTL47)	8/16/32
H'FFFF 7830	CAN1 Message Control Register 48 (C1MCTL48)	CAN1 Message Control Register 49 (C1MCTL49)	CAN1 Message Control Register 50 (C1MCTL50)	CAN1 Message Control Register 51 (C1MCTL51)	8/16/32
H'FFFF 7834	CAN1 Message Control Register 52 (C1MCTL52)	CAN1 Message Control Register 53 (C1MCTL53)	CAN1 Message Control Register 54 (C1MCTL54)	CAN1 Message Control Register 55 (C1MCTL55)	8/16/32
H'FFFF 7838	CAN1 Message Control Register 56 (C1MCTL56)	CAN1 Message Control Register 57 (C1MCTL57)	CAN1 Message Control Register 58 (C1MCTL58)	CAN1 Message Control Register 59 (C1MCTL59)	8/16/32
H'FFFF 783C	CAN1 Message Control Register 60 (C1MCTL60)	CAN1 Message Control Register 61 (C1MCTL61)	CAN1 Message Control Register 62 (C1MCTL62)	CAN1 Message Control Register 63 (C1MCTL63)	8/16/32
H'FFFF 7840	CAN1 Control Register (C1CTLR)		CAN1 Status Register (C1STR)		8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 7844	CAN1 Bit Configuration Register (C1BCR)			CAN1 Clock Select Register (C1CLKR)	8/16/32
H'FFFF 7848	CAN1 Receive FIFO Control Register (C1RFCR)	CAN1 Receive FIFO Pointer Control Register (C1RFPCR)	CAN1 Transmit FIFO Control Register (C1TFCR)	CAN1 Transmit FIFO Pointer Control Register (C1TFPCR)	8/16/32
H'FFFF 784C	CAN1 Error Interrupt Enable Register (C1EIER)	CAN1Error Interrupt Factor Judge Register (C1EIFR)	CAN1 Receive Error Count Register (C1RECR)	CAN1 Transmit Error Count Register (C1TECR)	8/16/32
H'FFFF 7850	CAN1 Error Code Store Register (C1ECSR)	CAN1 Channel Search Support Register (C1CSSR)	CAN1 Mailbox Search Status Register (C1MSSR)	CAN1 Mailbox Search Mode Register (C1MSMR)	8/16/32
H'FFFF 7854	CAN1 Time Stamp Register (C1TSR)		CAN1 Acceptance Filter Support Register (C1AFSR)		8/16/32
H'FFFF 7858	CAN1 Test Control Register (C1TCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF 7860	CAN1 Interrupt Enable Register (C1IER)	CAN1 Interrupt Status Register (C1ISR)	(Reserved)	CAN1 Mailbox Search Mask Register (C1MBSMR)	8/16, 8/16, -, 8
:	(Reserved)				-
H'FFFF 8000	CAN2 Mailbox Register 0 (C2MB0)				8/16/32
H'FFFF 8004	CAN2 Mailbox Register 0 (C2MB0)				8/16/32
H'FFFF 8008	CAN2 Mailbox Register 0 (C2MB0)				8/16/32
H'FFFF 800C	CAN2 Mailbox Register 0 (C2MB0)				8/16/32
H'FFFF 8010	CAN2 Mailbox Register 1 (C2MB1)				8/16/32
H'FFFF 8014	CAN2 Mailbox Register 1 (C2MB1)				8/16/32
H'FFFF 8018	CAN2 Mailbox Register 1 (C2MB1)				8/16/32
H'FFFF 801C	CAN2 Mailbox Register 1 (C2MB1)				8/16/32
H'FFFF 8020	CAN2 Mailbox Register 2 (C2MB2)				8/16/32
H'FFFF 8024	CAN2 Mailbox Register 2 (C2MB2)				8/16/32
H'FFFF 8028	CAN2 Mailbox Register 2 (C2MB2)				8/16/32
H'FFFF 802C	CAN2 Mailbox Register 2 (C2MB2)				8/16/32
H'FFFF 8030	CAN2 Mailbox Register 3 (C2MB3)				8/16/32
H'FFFF 8034	CAN2 Mailbox Register 3 (C2MB3)				8/16/32
H'FFFF 8038	CAN2 Mailbox Register 3 (C2MB3)				8/16/32
H'FFFF 803C	CAN2 Mailbox Register 3 (C2MB3)				8/16/32
H'FFFF 8040	CAN2 Mailbox Register 4 (C2MB4)				8/16/32
H'FFFF 8044	CAN2 Mailbox Register 4 (C2MB4)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 8048	CAN2 Mailbox Register 4 (C2MB4)				8/16/32
H'FFFF 804C	CAN2 Mailbox Register 4 (C2MB4)				8/16/32
H'FFFF 8050	CAN2 Mailbox Register 5 (C2MB5)				8/16/32
H'FFFF 8054	CAN2 Mailbox Register 5 (C2MB5)				8/16/32
H'FFFF 8058	CAN2 Mailbox Register 5 (C2MB5)				8/16/32
H'FFFF 805C	CAN2 Mailbox Register 5 (C2MB5)				8/16/32
H'FFFF 8060	CAN2 Mailbox Register 6 (C2MB6)				8/16/32
H'FFFF 8064	CAN2 Mailbox Register 6 (C2MB6)				8/16/32
H'FFFF 8068	CAN2 Mailbox Register 6 (C2MB6)				8/16/32
H'FFFF 806C	CAN2 Mailbox Register 6 (C2MB6)				8/16/32
H'FFFF 8070	CAN2 Mailbox Register 7 (C2MB7)				8/16/32
H'FFFF 8074	CAN2 Mailbox Register 7 (C2MB7)				8/16/32
H'FFFF 8078	CAN2 Mailbox Register 7 (C2MB7)				8/16/32
H'FFFF 807C	CAN2 Mailbox Register 7 (C2MB7)				8/16/32
H'FFFF 8080	CAN2 Mailbox Register 8 (C2MB8)				8/16/32
H'FFFF 8084	CAN2 Mailbox Register 8 (C2MB8)				8/16/32
H'FFFF 8088	CAN2 Mailbox Register 8 (C2MB8)				8/16/32
H'FFFF 808C	CAN2 Mailbox Register 8 (C2MB8)				8/16/32
H'FFFF 8090	CAN2 Mailbox Register 9 (C2MB9)				8/16/32
H'FFFF 8094	CAN2 Mailbox Register 9 (C2MB9)				8/16/32
H'FFFF 8098	CAN2 Mailbox Register 9 (C2MB9)				8/16/32
H'FFFF 809C	CAN2 Mailbox Register 9 (C2MB9)				8/16/32
H'FFFF 80A0	CAN2 Mailbox Register 10 (C2MB10)				8/16/32
H'FFFF 80A4	CAN2 Mailbox Register 10 (C2MB10)				8/16/32
H'FFFF 80A8	CAN2 Mailbox Register 10 (C2MB10)				8/16/32
H'FFFF 80AC	CAN2 Mailbox Register 10 (C2MB10)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 80B0	CAN2 Mailbox Register 11 (C2MB11)				8/16/32
H'FFFF 80B4	CAN2 Mailbox Register 11 (C2MB11)				8/16/32
H'FFFF 80B8	CAN2 Mailbox Register 11 (C2MB11)				8/16/32
H'FFFF 80BC	CAN2 Mailbox Register 11 (C2MB11)				8/16/32
H'FFFF 80C0	CAN2 Mailbox Register 12 (C2MB12)				8/16/32
H'FFFF 80C4	CAN2 Mailbox Register 12 (C2MB12)				8/16/32
H'FFFF 80C8	CAN2 Mailbox Register 12 (C2MB12)				8/16/32
H'FFFF 80CC	CAN2 Mailbox Register 12 (C2MB12)				8/16/32
H'FFFF 80D0	CAN2 Mailbox Register 13 (C2MB13)				8/16/32
H'FFFF 80D4	CAN2 Mailbox Register 13 (C2MB13)				8/16/32
H'FFFF 80D8	CAN2 Mailbox Register 13 (C2MB13)				8/16/32
H'FFFF 80DC	CAN2 Mailbox Register 13 (C2MB13)				8/16/32
H'FFFF 80E0	CAN2 Mailbox Register 14 (C2MB14)				8/16/32
H'FFFF 80E4	CAN2 Mailbox Register 14 (C2MB14)				8/16/32
H'FFFF 80E8	CAN2 Mailbox Register 14 (C2MB14)				8/16/32
H'FFFF 80EC	CAN2 Mailbox Register 14 (C2MB14)				8/16/32
H'FFFF 80F0	CAN2 Mailbox Register 15 (C2MB15)				8/16/32
H'FFFF 80F4	CAN2 Mailbox Register 15 (C2MB15)				8/16/32
H'FFFF 80F8	CAN2 Mailbox Register 15 (C2MB15)				8/16/32
H'FFFF 80FC	CAN2 Mailbox Register 15 (C2MB15)				8/16/32
H'FFFF 8100	CAN2 Mailbox Register 16 (C2MB16)				8/16/32
H'FFFF 8104	CAN2 Mailbox Register 16 (C2MB16)				8/16/32
H'FFFF 8108	CAN2 Mailbox Register 16 (C2MB16)				8/16/32
H'FFFF 810C	CAN2 Mailbox Register 16 (C2MB16)				8/16/32
H'FFFF 8110	CAN2 Mailbox Register 17 (C2MB17)				8/16/32
H'FFFF 8114	CAN2 Mailbox Register 17 (C2MB17)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 8118	CAN2 Mailbox Register 17 (C2MB17)				8/16/32
H'FFFF 811C	CAN2 Mailbox Register 17 (C2MB17)				8/16/32
H'FFFF 8120	CAN2 Mailbox Register 18 (C2MB18)				8/16/32
H'FFFF 8124	CAN2 Mailbox Register 18 (C2MB18)				8/16/32
H'FFFF 8128	CAN2 Mailbox Register 18 (C2MB18)				8/16/32
H'FFFF 812C	CAN2 Mailbox Register 18 (C2MB18)				8/16/32
H'FFFF 8130	CAN2 Mailbox Register 19 (C2MB19)				8/16/32
H'FFFF 8134	CAN2 Mailbox Register 19 (C2MB19)				8/16/32
H'FFFF 8138	CAN2 Mailbox Register 19 (C2MB19)				8/16/32
H'FFFF 813C	CAN2 Mailbox Register 19 (C2MB19)				8/16/32
H'FFFF 8140	CAN2 Mailbox Register 20 (C2MB20)				8/16/32
H'FFFF 8144	CAN2 Mailbox Register 20 (C2MB20)				8/16/32
H'FFFF 8148	CAN2 Mailbox Register 20 (C2MB20)				8/16/32
H'FFFF 814C	CAN2 Mailbox Register 20 (C2MB20)				8/16/32
H'FFFF 8150	CAN2 Mailbox Register 21 (C2MB21)				8/16/32
H'FFFF 8154	CAN2 Mailbox Register 21 (C2MB21)				8/16/32
H'FFFF 8158	CAN2 Mailbox Register 21 (C2MB21)				8/16/32
H'FFFF 815C	CAN2 Mailbox Register 21 (C2MB21)				8/16/32
H'FFFF 8160	CAN2 Mailbox Register 22 (C2MB22)				8/16/32
H'FFFF 8164	CAN2 Mailbox Register 22 (C2MB22)				8/16/32
H'FFFF 8168	CAN2 Mailbox Register 22 (C2MB22)				8/16/32
H'FFFF 816C	CAN2 Mailbox Register 22 (C2MB22)				8/16/32
H'FFFF 8170	CAN2 Mailbox Register 23 (C2MB23)				8/16/32
H'FFFF 8174	CAN2 Mailbox Register 23 (C2MB23)				8/16/32
H'FFFF 8178	CAN2 Mailbox Register 23 (C2MB23)				8/16/32
H'FFFF 817C	CAN2 Mailbox Register 23 (C2MB23)				8/16/32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF 8180	CAN2 Mailbox Register 24 (C2MB24)								8/16/32
H'FFFF 8184	CAN2 Mailbox Register 24 (C2MB24)								8/16/32
H'FFFF 8188	CAN2 Mailbox Register 24 (C2MB24)								8/16/32
H'FFFF 818C	CAN2 Mailbox Register 24 (C2MB24)								8/16/32
H'FFFF 8190	CAN2 Mailbox Register 25 (C2MB25)								8/16/32
H'FFFF 8194	CAN2 Mailbox Register 25 (C2MB25)								8/16/32
H'FFFF 8198	CAN2 Mailbox Register 25 (C2MB25)								8/16/32
H'FFFF 819C	CAN2 Mailbox Register 25 (C2MB25)								8/16/32
H'FFFF 81A0	CAN2 Mailbox Register 26 (C2MB26)								8/16/32
H'FFFF 81A4	CAN2 Mailbox Register 26 (C2MB26)								8/16/32
H'FFFF 81A8	CAN2 Mailbox Register 26 (C2MB26)								8/16/32
H'FFFF 81AC	CAN2 Mailbox Register 26 (C2MB26)								8/16/32
H'FFFF 81B0	CAN2 Mailbox Register 27 (C2MB27)								8/16/32
H'FFFF 81B4	CAN2 Mailbox Register 27 (C2MB27)								8/16/32
H'FFFF 81B8	CAN2 Mailbox Register 27 (C2MB27)								8/16/32
H'FFFF 81BC	CAN2 Mailbox Register 27 (C2MB27)								8/16/32
H'FFFF 81C0	CAN2 Mailbox Register 28 (C2MB28)								8/16/32
H'FFFF 81C4	CAN2 Mailbox Register 28 (C2MB28)								8/16/32
H'FFFF 81C8	CAN2 Mailbox Register 28 (C2MB28)								8/16/32
H'FFFF 81CC	CAN2 Mailbox Register 28 (C2MB28)								8/16/32
H'FFFF 81D0	CAN2 Mailbox Register 29 (C2MB29)								8/16/32
H'FFFF 81D4	CAN2 Mailbox Register 29 (C2MB29)								8/16/32
H'FFFF 81D8	CAN2 Mailbox Register 29 (C2MB29)								8/16/32
H'FFFF 81DC	CAN2 Mailbox Register 29 (C2MB29)								8/16/32
H'FFFF 81E0	CAN2 Mailbox Register 30 (C2MB30)								8/16/32
H'FFFF 81E4	CAN2 Mailbox Register 30 (C2MB30)								8/16/32

	+0 Address	+1 Address	+2 Address	+3 Address	
Address	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	Access Size
H'FFFF 81E8	CAN2 Mailbox Register 30 (C2MB30)				8/16/32
H'FFFF 81EC	CAN2 Mailbox Register 30 (C2MB30)				8/16/32
H'FFFF 81F0	CAN2 Mailbox Register 31 (C2MB31)				8/16/32
H'FFFF 81F4	CAN2 Mailbox Register 31 (C2MB31)				8/16/32
H'FFFF 81F8	CAN2 Mailbox Register 31 (C2MB31)				8/16/32
H'FFFF 81FC	CAN2 Mailbox Register 31 (C2MB31)				8/16/32
H'FFFF 8200	CAN2 Mailbox Register 32 (C2MB32)				8/16/32
H'FFFF 8204	CAN2 Mailbox Register 32 (C2MB32)				8/16/32
H'FFFF 8208	CAN2 Mailbox Register 32 (C2MB32)				8/16/32
H'FFFF 820C	CAN2 Mailbox Register 32 (C2MB32)				8/16/32
H'FFFF 8210	CAN2 Mailbox Register 33 (C2MB33)				8/16/32
H'FFFF 8214	CAN2 Mailbox Register 33 (C2MB33)				8/16/32
H'FFFF 8218	CAN2 Mailbox Register 33 (C2MB33)				8/16/32
H'FFFF 821C	CAN2 Mailbox Register 33 (C2MB33)				8/16/32
H'FFFF 8220	CAN2 Mailbox Register 34 (C2MB34)				8/16/32
H'FFFF 8224	CAN2 Mailbox Register 34 (C2MB34)				8/16/32
H'FFFF 8228	CAN2 Mailbox Register 34 (C2MB34)				8/16/32
H'FFFF 822C	CAN2 Mailbox Register 34 (C2MB34)				8/16/32
H'FFFF 8230	CAN2 Mailbox Register 35 (C2MB35)				8/16/32
H'FFFF 8234	CAN2 Mailbox Register 35 (C2MB35)				8/16/32
H'FFFF 8238	CAN2 Mailbox Register 35 (C2MB35)				8/16/32
H'FFFF 823C	CAN2 Mailbox Register 35 (C2MB35)				8/16/32
H'FFFF 8240	CAN2 Mailbox Register 36 (C2MB36)				8/16/32
H'FFFF 8244	CAN2 Mailbox Register 36 (C2MB36)				8/16/32
H'FFFF 8248	CAN2 Mailbox Register 36 (C2MB36)				8/16/32
H'FFFF 824C	CAN2 Mailbox Register 36 (C2MB36)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 8250	CAN2 Mailbox Register 37 (C2MB37)				8/16/32
H'FFFF 8254	CAN2 Mailbox Register 37 (C2MB37)				8/16/32
H'FFFF 8258	CAN2 Mailbox Register 37 (C2MB37)				8/16/32
H'FFFF 825C	CAN2 Mailbox Register 37 (C2MB37)				8/16/32
H'FFFF 8260	CAN2 Mailbox Register 38 (C2MB38)				8/16/32
H'FFFF 8264	CAN2 Mailbox Register 38 (C2MB38)				8/16/32
H'FFFF 8268	CAN2 Mailbox Register 38 (C2MB38)				8/16/32
H'FFFF 826C	CAN2 Mailbox Register 38 (C2MB38)				8/16/32
H'FFFF 8270	CAN2 Mailbox Register 39 (C2MB39)				8/16/32
H'FFFF 8274	CAN2 Mailbox Register 39 (C2MB39)				8/16/32
H'FFFF 8278	CAN2 Mailbox Register 39 (C2MB39)				8/16/32
H'FFFF 827C	CAN2 Mailbox Register 39 (C2MB39)				8/16/32
H'FFFF 8280	CAN2 Mailbox Register 40 (C2MB40)				8/16/32
H'FFFF 8284	CAN2 Mailbox Register 40 (C2MB40)				8/16/32
H'FFFF 8288	CAN2 Mailbox Register 40 (C2MB40)				8/16/32
H'FFFF 828C	CAN2 Mailbox Register 40 (C2MB40)				8/16/32
H'FFFF 8290	CAN2 Mailbox Register 41 (C2MB41)				8/16/32
H'FFFF 8294	CAN2 Mailbox Register 41 (C2MB41)				8/16/32
H'FFFF 8298	CAN2 Mailbox Register 41 (C2MB41)				8/16/32
H'FFFF 829C	CAN2 Mailbox Register 41 (C2MB41)				8/16/32
H'FFFF 82A0	CAN2 Mailbox Register 42 (C2MB42)				8/16/32
H'FFFF 82A4	CAN2 Mailbox Register 42 (C2MB42)				8/16/32
H'FFFF 82A8	CAN2 Mailbox Register 42 (C2MB42)				8/16/32
H'FFFF 82AC	CAN2 Mailbox Register 42 (C2MB42)				8/16/32
H'FFFF 82B0	CAN2 Mailbox Register 43 (C2MB43)				8/16/32
H'FFFF 82B4	CAN2 Mailbox Register 43 (C2MB43)				8/16/32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF 82B8	CAN2 Mailbox Register 43 (C2MB43)								8/16/32
H'FFFF 82BC	CAN2 Mailbox Register 43 (C2MB43)								8/16/32
H'FFFF 82C0	CAN2 Mailbox Register 44 (C2MB44)								8/16/32
H'FFFF 82C4	CAN2 Mailbox Register 44 (C2MB44)								8/16/32
H'FFFF 82C8	CAN2 Mailbox Register 44 (C2MB44)								8/16/32
H'FFFF 82CC	CAN2 Mailbox Register 44 (C2MB44)								8/16/32
H'FFFF 82D0	CAN2 Mailbox Register 45 (C2MB45)								8/16/32
H'FFFF 82D4	CAN2 Mailbox Register 45 (C2MB45)								8/16/32
H'FFFF 82D8	CAN2 Mailbox Register 45 (C2MB45)								8/16/32
H'FFFF 82DC	CAN2 Mailbox Register 45 (C2MB45)								8/16/32
H'FFFF 82E0	CAN2 Mailbox Register 46 (C2MB46)								8/16/32
H'FFFF 82E4	CAN2 Mailbox Register 46 (C2MB46)								8/16/32
H'FFFF 82E8	CAN2 Mailbox Register 46 (C2MB46)								8/16/32
H'FFFF 82EC	CAN2 Mailbox Register 46 (C2MB46)								8/16/32
H'FFFF 82F0	CAN2 Mailbox Register 47 (C2MB47)								8/16/32
H'FFFF 82F4	CAN2 Mailbox Register 47 (C2MB47)								8/16/32
H'FFFF 82F8	CAN2 Mailbox Register 47 (C2MB47)								8/16/32
H'FFFF 82FC	CAN2 Mailbox Register 47 (C2MB47)								8/16/32
H'FFFF 8300	CAN2 Mailbox Register 48 (C2MB48)								8/16/32
H'FFFF 8304	CAN2 Mailbox Register 48 (C2MB48)								8/16/32
H'FFFF 8308	CAN2 Mailbox Register 48 (C2MB48)								8/16/32
H'FFFF 830C	CAN2 Mailbox Register 48 (C2MB48)								8/16/32
H'FFFF 8310	CAN2 Mailbox Register 49 (C2MB49)								8/16/32
H'FFFF 8314	CAN2 Mailbox Register 49 (C2MB49)								8/16/32
H'FFFF 8318	CAN2 Mailbox Register 49 (C2MB49)								8/16/32
H'FFFF 831C	CAN2 Mailbox Register 49 (C2MB49)								8/16/32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF 8320	CAN2 Mailbox Register 50 (C2MB50)								8/16/32
H'FFFF 8324	CAN2 Mailbox Register 50 (C2MB50)								8/16/32
H'FFFF 8328	CAN2 Mailbox Register 50 (C2MB50)								8/16/32
H'FFFF 832C	CAN2 Mailbox Register 50 (C2MB50)								8/16/32
H'FFFF 8330	CAN2 Mailbox Register 51 (C2MB51)								8/16/32
H'FFFF 8334	CAN2 Mailbox Register 51 (C2MB51)								8/16/32
H'FFFF 8338	CAN2 Mailbox Register 51 (C2MB51)								8/16/32
H'FFFF 833C	CAN2 Mailbox Register 51 (C2MB51)								8/16/32
H'FFFF 8340	CAN2 Mailbox Register 52 (C2MB52)								8/16/32
H'FFFF 8344	CAN2 Mailbox Register 52 (C2MB52)								8/16/32
H'FFFF 8348	CAN2 Mailbox Register 52 (C2MB52)								8/16/32
H'FFFF 834C	CAN2 Mailbox Register 52 (C2MB52)								8/16/32
H'FFFF 8350	CAN2 Mailbox Register 53 (C2MB53)								8/16/32
H'FFFF 8354	CAN2 Mailbox Register 53 (C2MB53)								8/16/32
H'FFFF 8358	CAN2 Mailbox Register 53 (C2MB53)								8/16/32
H'FFFF 835C	CAN2 Mailbox Register 53 (C2MB53)								8/16/32
H'FFFF 8360	CAN2 Mailbox Register 54 (C2MB54)								8/16/32
H'FFFF 8364	CAN2 Mailbox Register 54 (C2MB54)								8/16/32
H'FFFF 8368	CAN2 Mailbox Register 54 (C2MB54)								8/16/32
H'FFFF 836C	CAN2 Mailbox Register 54 (C2MB54)								8/16/32
H'FFFF 8370	CAN2 Mailbox Register 55 (C2MB55)								8/16/32
H'FFFF 8374	CAN2 Mailbox Register 55 (C2MB55)								8/16/32
H'FFFF 8378	CAN2 Mailbox Register 55 (C2MB55)								8/16/32
H'FFFF 837C	CAN2 Mailbox Register 55 (C2MB55)								8/16/32
H'FFFF 8380	CAN2 Mailbox Register 56 (C2MB56)								8/16/32
H'FFFF 8384	CAN2 Mailbox Register 56 (C2MB56)								8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 8388	CAN2 Mailbox Register 56 (C2MB56)				8/16/32
H'FFFF 838C	CAN2 Mailbox Register 56 (C2MB56)				8/16/32
H'FFFF 8390	CAN2 Mailbox Register 57 (C2MB57)				8/16/32
H'FFFF 8394	CAN2 Mailbox Register 57 (C2MB57)				8/16/32
H'FFFF 8398	CAN2 Mailbox Register 57 (C2MB57)				8/16/32
H'FFFF 839C	CAN2 Mailbox Register 57 (C2MB57)				8/16/32
H'FFFF 83A0	CAN2 Mailbox Register 58 (C2MB58)				8/16/32
H'FFFF 83A4	CAN2 Mailbox Register 58 (C2MB58)				8/16/32
H'FFFF 83A8	CAN2 Mailbox Register 58 (C2MB58)				8/16/32
H'FFFF 83AC	CAN2 Mailbox Register 58 (C2MB58)				8/16/32
H'FFFF 83B0	CAN2 Mailbox Register 59 (C2MB59)				8/16/32
H'FFFF 83B5	CAN2 Mailbox Register 59 (C2MB59)				8/16/32
H'FFFF 83B8	CAN2 Mailbox Register 59 (C2MB59)				8/16/32
H'FFFF 83BC	CAN2 Mailbox Register 59 (C2MB59)				8/16/32
H'FFFF 83C0	CAN2 Mailbox Register 60 (C2MB60)				8/16/32
H'FFFF 83C4	CAN2 Mailbox Register 60 (C2MB60)				8/16/32
H'FFFF 83C8	CAN2 Mailbox Register 60 (C2MB60)				8/16/32
H'FFFF 83CC	CAN2 Mailbox Register 60 (C2MB60)				8/16/32
H'FFFF 83D0	CAN2 Mailbox Register 61 (C2MB61)				8/16/32
H'FFFF 83D4	CAN2 Mailbox Register 61 (C2MB61)				8/16/32
H'FFFF 83D8	CAN2 Mailbox Register 61 (C2MB61)				8/16/32
H'FFFF 83DC	CAN2 Mailbox Register 61 (C2MB61)				8/16/32
H'FFFF 83E0	CAN2 Mailbox Register 62 (C2MB62)				8/16/32
H'FFFF 83E4	CAN2 Mailbox Register 62 (C2MB62)				8/16/32
H'FFFF 83E8	CAN2 Mailbox Register 62 (C2MB62)				8/16/32
H'FFFF 83EC	CAN2 Mailbox Register 62 (C2MB62)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 83F0	CAN2 Mailbox Register 63 (C2MB63)				8/16/32
H'FFFF 83F4	CAN2 Mailbox Register 63 (C2MB63)				8/16/32
H'FFFF 83F8	CAN2 Mailbox Register 63 (C2MB63)				8/16/32
H'FFFF 83FC	CAN2 Mailbox Register 63 (C2MB63)				8/16/32
H'FFFF 8400	CAN2 Mask Register 2 (C2MKR2)				8/16/32
H'FFFF 8404	CAN2 Mask Register 3 (C2MKR3)				8/16/32
H'FFFF 8408	CAN2 Mask Register 4 (C2MKR4)				8/16/32
H'FFFF 840C	CAN2 Mask Register 5 (C2MKR5)				8/16/32
H'FFFF 8410	CAN2 Mask Register 6 (C2MKR6)				8/16/32
H'FFFF 8414	CAN2 Mask Register 7 (C2MKR7)				8/16/32
H'FFFF 8418	CAN2 Mask Register 8 (C2MKR8)				8/16/32
H'FFFF 841C	CAN2 Mask Register 9 (C2MKR9)				8/16/32
H'FFFF 8420	CAN2 FIFO Received ID Compare Register 0 (C2FIDCR0)				8/16/32
H'FFFF 8424	CAN2 FIFO Received ID Compare Register 1 (C2FIDCR1)				8/16/32
H'FFFF 8428	CAN2 Mask Invalid Register 1 (C2MKIVLR1)				8/16/32
H'FFFF 842C	CAN2 Mailbox Interrupt Enable Register 1 (C2MIER1)				8/16/32
H'FFFF 8430	CAN2 Mask Register 0 (C2MKR0)				8/16/32
H'FFFF 8434	CAN2 Mask Register 1 (C2MKR1)				8/16/32
H'FFFF 8438	CAN2 Mask Invalid Register 0 (C2MKIVLR0)				8/16/32
H'FFFF 843C	CAN2 Mailbox Interrupt Enable Register 0 (C2MIER0)				8/16/32
:	(Reserved)				-
H'FFFF 8800	CAN2 Message Control Register 0 (C2MCTL0)	CAN2 Message Control Register 1 (C2MCTL1)	CAN2 Message Control Register 2 (C2MCTL2)	CAN2 Message Control Register 3 (C2MCTL3)	8/16/32
H'FFFF 8804	CAN2 Message Control Register 4 (C2MCTL4)	CAN2 Message Control Register 5 (C2MCTL5)	CAN2 Message Control Register 6 (C2MCTL6)	CAN2 Message Control Register 7 (C2MCTL7)	8/16/32
H'FFFF 8808	CAN2 Message Control Register 8 (C2MCTL8)	CAN2 Message Control Register 9 (C2MCTL9)	CAN2 Message Control Register 10 (C2MCTL10)	CAN2 Message Control Register 11 (C2MCTL11)	8/16/32
H'FFFF 880C	CAN2 Message Control Register 12 (C2MCTL12)	CAN2 Message Control Register 13 (C2MCTL13)	CAN2 Message Control Register 14 (C2MCTL14)	CAN2 Message Control Register 15 (C2MCTL15)	8/16/32
H'FFFF 8810	CAN2 Message Control Register 16 (C2MCTL16)	CAN2 Message Control Register 17 (C2MCTL17)	CAN2 Message Control Register 18 (C2MCTL18)	CAN2 Message Control Register 19 (C2MCTL19)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 8814	CAN2 Message Control Register 20 (C2MCTL20)	CAN2 Message Control Register 21 (C2MCTL21)	CAN2 Message Control Register 22 (C2MCTL22)	CAN2 Message Control Register 23 (C2MCTL23)	8/16/32
H'FFFF 8818	CAN2 Message Control Register 24 (C2MCTL24)	CAN2 Message Control Register 25 (C2MCTL25)	CAN2 Message Control Register 26 (C2MCTL26)	CAN2 Message Control Register 27 (C2MCTL27)	8/16/32
H'FFFF 881C	CAN2 Message Control Register 28 (C2MCTL28)	CAN2 Message Control Register 29 (C2MCTL29)	CAN2 Message Control Register 30 (C2MCTL30)	CAN2 Message Control Register 31 (C2MCTL31)	8/16/32
H'FFFF 8820	CAN2 Message Control Register 32 (C2MCTL32)	CAN2 Message Control Register 33 (C2MCTL33)	CAN2 Message Control Register 34 (C2MCTL34)	CAN2 Message Control Register 35 (C2MCTL35)	8/16/32
H'FFFF 8824	CAN2 Message Control Register 36 (C2MCTL36)	CAN2 Message Control Register 37 (C2MCTL37)	CAN2 Message Control Register 38 (C2MCTL38)	CAN2 Message Control Register 39 (C2MCTL39)	8/16/32
H'FFFF 8828	CAN2 Message Control Register 40 (C2MCTL40)	CAN2 Message Control Register 41 (C2MCTL41)	CAN2 Message Control Register 42 (C2MCTL42)	CAN2 Message Control Register 43 (C2MCTL43)	8/16/32
H'FFFF 882C	CAN2 Message Control Register 44 (C2MCTL44)	CAN2 Message Control Register 45 (C2MCTL45)	CAN2 Message Control Register 46 (C2MCTL46)	CAN2 Message Control Register 47 (C2MCTL47)	8/16/32
H'FFFF 8830	CAN2 Message Control Register 48 (C2MCTL48)	CAN2 Message Control Register 49 (C2MCTL49)	CAN2 Message Control Register 50 (C2MCTL50)	CAN2 Message Control Register 51 (C2MCTL51)	8/16/32
H'FFFF 8834	CAN2 Message Control Register 52 (C2MCTL52)	CAN2 Message Control Register 53 (C2MCTL53)	CAN2 Message Control Register 54 (C2MCTL54)	CAN2 Message Control Register 55 (C2MCTL55)	8/16/32
H'FFFF 8838	CAN2 Message Control Register 56 (C2MCTL56)	CAN2 Message Control Register 57 (C2MCTL57)	CAN2 Message Control Register 58 (C2MCTL58)	CAN2 Message Control Register 59 (C2MCTL59)	8/16/32
H'FFFF 883C	CAN2 Message Control Register 60 (C2MCTL60)	CAN2 Message Control Register 61 (C2MCTL61)	CAN2 Message Control Register 62 (C2MCTL62)	CAN2 Message Control Register 63 (C2MCTL63)	8/16/32
H'FFFF 8840	CAN2 Control Register (C2CTLR)		CAN2 Status Register (C2STR)		8/16/32
H'FFFF 8844	CAN2 Bit Configuration Register (C2BCR)			CAN2 Clock Select Register (C2CLKR)	8/16/32
H'FFFF 8848	CAN2 Receive FIFO Control Register (C2RFCR)	CAN2 Receive FIFO Pointer Control Register (C2RFPCR)	CAN2 Transmit FIFO Control Register (C2TFCR)	CAN2 Transmit FIFO Pointer Control Register (C2TFPCR)	8/16/32
H'FFFF 884C	CAN2 Error Interrupt Enable Register (C2EIER)	CAN2Error Interrupt Factor Judge Register (C2EIFR)	CAN2 Receive Error Count Register (C2RECR)	CAN2 Transmit Error Count Register (C2TECR)	8/16/32
H'FFFF 8850	CAN2 Error Code Store Register (C2ECSR)	CAN2 Channel Search Support Register (C2CSSR)	CAN2 Mailbox Search Status Register (C2MSSR)	CAN2 Mailbox Search Mode Register (C2MSMR)	8/16/32
H'FFFF 8854	CAN2 Time Stamp Register (C2TSR)		CAN2 Acceptance Filter Support Register (C2AFSR)		8/16/32
H'FFFF 8858	CAN2 Test Control Register (C2TCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF 8860	CAN2 Interrupt Enable Register (C2IER)	CAN2 Interrupt Status Register (C2ISR)	(Reserved)	CAN2 Mailbox Search Mask Register (C2MBSMR)	8/16, 8/16, -, 8
:	(Reserved)				-
H'FFFF 9000	CAN3 Mailbox Register 0 (C3MB0)				8/16/32
H'FFFF 9004	CAN3 Mailbox Register 0 (C3MB0)				8/16/32
H'FFFF 9008	CAN3 Mailbox Register 0 (C3MB0)				8/16/32
H'FFFF 900C	CAN3 Mailbox Register 0 (C3MB0)				8/16/32
H'FFFF 9010	CAN3 Mailbox Register 1 (C3MB1)				8/16/32
H'FFFF 9014	CAN3 Mailbox Register 1 (C3MB1)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 9018	CAN3 Mailbox Register 1 (C3MB1)				8/16/32
H'FFFF 901C	CAN3 Mailbox Register 1 (C3MB1)				8/16/32
H'FFFF 9020	CAN3 Mailbox Register 2 (C3MB2)				8/16/32
H'FFFF 9024	CAN3 Mailbox Register 2 (C3MB2)				8/16/32
H'FFFF 9028	CAN3 Mailbox Register 2 (C3MB2)				8/16/32
H'FFFF 902C	CAN3 Mailbox Register 2 (C3MB2)				8/16/32
H'FFFF 9030	CAN3 Mailbox Register 3 (C3MB3)				8/16/32
H'FFFF 9034	CAN3 Mailbox Register 3 (C3MB3)				8/16/32
H'FFFF 9038	CAN3 Mailbox Register 3 (C3MB3)				8/16/32
H'FFFF 903C	CAN3 Mailbox Register 3 (C3MB3)				8/16/32
H'FFFF 9040	CAN3 Mailbox Register 4 (C3MB4)				8/16/32
H'FFFF 9044	CAN3 Mailbox Register 4 (C3MB4)				8/16/32
H'FFFF 9048	CAN3 Mailbox Register 4 (C3MB4)				8/16/32
H'FFFF 904C	CAN3 Mailbox Register 4 (C3MB4)				8/16/32
H'FFFF 9050	CAN3 Mailbox Register 5 (C3MB5)				8/16/32
H'FFFF 9054	CAN3 Mailbox Register 5 (C3MB5)				8/16/32
H'FFFF 9058	CAN3 Mailbox Register 5 (C3MB5)				8/16/32
H'FFFF 905C	CAN3 Mailbox Register 5 (C3MB5)				8/16/32
H'FFFF 9060	CAN3 Mailbox Register 6 (C3MB6)				8/16/32
H'FFFF 9064	CAN3 Mailbox Register 6 (C3MB6)				8/16/32
H'FFFF 9068	CAN3 Mailbox Register 6 (C3MB6)				8/16/32
H'FFFF 906C	CAN3 Mailbox Register 6 (C3MB6)				8/16/32
H'FFFF 9070	CAN3 Mailbox Register 7 (C3MB7)				8/16/32
H'FFFF 9074	CAN3 Mailbox Register 7 (C3MB7)				8/16/32
H'FFFF 9078	CAN3 Mailbox Register 7 (C3MB7)				8/16/32
H'FFFF 907C	CAN3 Mailbox Register 7 (C3MB7)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 9080	CAN3 Mailbox Register 8 (C3MB8)				8/16/32
H'FFFF 9084	CAN3 Mailbox Register 8 (C3MB8)				8/16/32
H'FFFF 9088	CAN3 Mailbox Register 8 (C3MB8)				8/16/32
H'FFFF 908C	CAN3 Mailbox Register 8 (C3MB8)				8/16/32
H'FFFF 9090	CAN3 Mailbox Register 9 (C3MB9)				8/16/32
H'FFFF 9094	CAN3 Mailbox Register 9 (C3MB9)				8/16/32
H'FFFF 9098	CAN3 Mailbox Register 9 (C3MB9)				8/16/32
H'FFFF 909C	CAN3 Mailbox Register 9 (C3MB9)				8/16/32
H'FFFF 90A0	CAN3 Mailbox Register 10 (C3MB10)				8/16/32
H'FFFF 90A4	CAN3 Mailbox Register 10 (C3MB10)				8/16/32
H'FFFF 90A8	CAN3 Mailbox Register 10 (C3MB10)				8/16/32
H'FFFF 90AC	CAN3 Mailbox Register 10 (C3MB10)				8/16/32
H'FFFF 90B0	CAN3 Mailbox Register 11 (C3MB11)				8/16/32
H'FFFF 90B4	CAN3 Mailbox Register 11 (C3MB11)				8/16/32
H'FFFF 90B8	CAN3 Mailbox Register 11 (C3MB11)				8/16/32
H'FFFF 90BC	CAN3 Mailbox Register 11 (C3MB11)				8/16/32
H'FFFF 90C0	CAN3 Mailbox Register 12 (C3MB12)				8/16/32
H'FFFF 90C4	CAN3 Mailbox Register 12 (C3MB12)				8/16/32
H'FFFF 90C8	CAN3 Mailbox Register 12 (C3MB12)				8/16/32
H'FFFF 90CC	CAN3 Mailbox Register 12 (C3MB12)				8/16/32
H'FFFF 90D0	CAN3 Mailbox Register 13 (C3MB13)				8/16/32
H'FFFF 90D4	CAN3 Mailbox Register 13 (C3MB13)				8/16/32
H'FFFF 90D8	CAN3 Mailbox Register 13 (C3MB13)				8/16/32
H'FFFF 90DC	CAN3 Mailbox Register 13 (C3MB13)				8/16/32
H'FFFF 90E0	CAN3 Mailbox Register 14 (C3MB14)				8/16/32
H'FFFF 90E4	CAN3 Mailbox Register 14 (C3MB14)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 90E8	CAN3 Mailbox Register 14 (C3MB14)				8/16/32
H'FFFF 90EC	CAN3 Mailbox Register 14 (C3MB14)				8/16/32
H'FFFF 90F0	CAN3 Mailbox Register 15 (C3MB15)				8/16/32
H'FFFF 90F4	CAN3 Mailbox Register 15 (C3MB15)				8/16/32
H'FFFF 90F8	CAN3 Mailbox Register 15 (C3MB15)				8/16/32
H'FFFF 90FC	CAN3 Mailbox Register 15 (C3MB15)				8/16/32
H'FFFF 9100	CAN3 Mailbox Register 16 (C3MB16)				8/16/32
H'FFFF 9104	CAN3 Mailbox Register 16 (C3MB16)				8/16/32
H'FFFF 9108	CAN3 Mailbox Register 16 (C3MB16)				8/16/32
H'FFFF 910C	CAN3 Mailbox Register 16 (C3MB16)				8/16/32
H'FFFF 9110	CAN3 Mailbox Register 17 (C3MB17)				8/16/32
H'FFFF 9114	CAN3 Mailbox Register 17 (C3MB17)				8/16/32
H'FFFF 9118	CAN3 Mailbox Register 17 (C3MB17)				8/16/32
H'FFFF 911C	CAN3 Mailbox Register 17 (C3MB17)				8/16/32
H'FFFF 9120	CAN3 Mailbox Register 18 (C3MB18)				8/16/32
H'FFFF 9124	CAN3 Mailbox Register 18 (C3MB18)				8/16/32
H'FFFF 9128	CAN3 Mailbox Register 18 (C3MB18)				8/16/32
H'FFFF 912C	CAN3 Mailbox Register 18 (C3MB18)				8/16/32
H'FFFF 9130	CAN3 Mailbox Register 19 (C3MB19)				8/16/32
H'FFFF 9134	CAN3 Mailbox Register 19 (C3MB19)				8/16/32
H'FFFF 9138	CAN3 Mailbox Register 19 (C3MB19)				8/16/32
H'FFFF 913C	CAN3 Mailbox Register 19 (C3MB19)				8/16/32
H'FFFF 9140	CAN3 Mailbox Register 20 (C3MB20)				8/16/32
H'FFFF 9144	CAN3 Mailbox Register 20 (C3MB20)				8/16/32
H'FFFF 9148	CAN3 Mailbox Register 20 (C3MB20)				8/16/32
H'FFFF 914C	CAN3 Mailbox Register 20 (C3MB20)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 9150	CAN3 Mailbox Register 21 (C3MB21)				8/16/32
H'FFFF 9154	CAN3 Mailbox Register 21 (C3MB21)				8/16/32
H'FFFF 9158	CAN3 Mailbox Register 21 (C3MB21)				8/16/32
H'FFFF 915C	CAN3 Mailbox Register 21 (C3MB21)				8/16/32
H'FFFF 9160	CAN3 Mailbox Register 22 (C3MB22)				8/16/32
H'FFFF 9164	CAN3 Mailbox Register 22 (C3MB22)				8/16/32
H'FFFF 9168	CAN3 Mailbox Register 22 (C3MB22)				8/16/32
H'FFFF 916C	CAN3 Mailbox Register 22 (C3MB22)				8/16/32
H'FFFF 9170	CAN3 Mailbox Register 23 (C3MB23)				8/16/32
H'FFFF 9174	CAN3 Mailbox Register 23 (C3MB23)				8/16/32
H'FFFF 9178	CAN3 Mailbox Register 23 (C3MB23)				8/16/32
H'FFFF 917C	CAN3 Mailbox Register 23 (C3MB23)				8/16/32
H'FFFF 9180	CAN3 Mailbox Register 24 (C3MB24)				8/16/32
H'FFFF 9184	CAN3 Mailbox Register 24 (C3MB24)				8/16/32
H'FFFF 9188	CAN3 Mailbox Register 24 (C3MB24)				8/16/32
H'FFFF 918C	CAN3 Mailbox Register 24 (C3MB24)				8/16/32
H'FFFF 9190	CAN3 Mailbox Register 25 (C3MB25)				8/16/32
H'FFFF 9194	CAN3 Mailbox Register 25 (C3MB25)				8/16/32
H'FFFF 9198	CAN3 Mailbox Register 25 (C3MB25)				8/16/32
H'FFFF 919C	CAN3 Mailbox Register 25 (C3MB25)				8/16/32
H'FFFF 91A0	CAN3 Mailbox Register 26 (C3MB26)				8/16/32
H'FFFF 91A4	CAN3 Mailbox Register 26 (C3MB26)				8/16/32
H'FFFF 91A8	CAN3 Mailbox Register 26 (C3MB26)				8/16/32
H'FFFF 91AC	CAN3 Mailbox Register 26 (C3MB26)				8/16/32
H'FFFF 91B0	CAN3 Mailbox Register 27 (C3MB27)				8/16/32
H'FFFF 91B4	CAN3 Mailbox Register 27 (C3MB27)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 91B8	CAN3 Mailbox Register 27 (C3MB27)				8/16/32
H'FFFF 91BC	CAN3 Mailbox Register 27 (C3MB27)				8/16/32
H'FFFF 91C0	CAN3 Mailbox Register 28 (C3MB28)				8/16/32
H'FFFF 91C4	CAN3 Mailbox Register 28 (C3MB28)				8/16/32
H'FFFF 91C8	CAN3 Mailbox Register 28 (C3MB28)				8/16/32
H'FFFF 91CC	CAN3 Mailbox Register 28 (C3MB28)				8/16/32
H'FFFF 91D0	CAN3 Mailbox Register 29 (C3MB29)				8/16/32
H'FFFF 91D4	CAN3 Mailbox Register 29 (C3MB29)				8/16/32
H'FFFF 91D8	CAN3 Mailbox Register 29 (C3MB29)				8/16/32
H'FFFF 91DC	CAN3 Mailbox Register 29 (C3MB29)				8/16/32
H'FFFF 91E0	CAN3 Mailbox Register 30 (C3MB30)				8/16/32
H'FFFF 91E4	CAN3 Mailbox Register 30 (C3MB30)				8/16/32
H'FFFF 91E8	CAN3 Mailbox Register 30 (C3MB30)				8/16/32
H'FFFF 91EC	CAN3 Mailbox Register 30 (C3MB30)				8/16/32
H'FFFF 91F0	CAN3 Mailbox Register 31 (C3MB31)				8/16/32
H'FFFF 91F4	CAN3 Mailbox Register 31 (C3MB31)				8/16/32
H'FFFF 91F8	CAN3 Mailbox Register 31 (C3MB31)				8/16/32
H'FFFF 91FC	CAN3 Mailbox Register 31 (C3MB31)				8/16/32
H'FFFF 9200	CAN3 Mailbox Register 32 (C3MB32)				8/16/32
H'FFFF 9204	CAN3 Mailbox Register 32 (C3MB32)				8/16/32
H'FFFF 9208	CAN3 Mailbox Register 32 (C3MB32)				8/16/32
H'FFFF 920C	CAN3 Mailbox Register 32 (C3MB32)				8/16/32
H'FFFF 9210	CAN3 Mailbox Register 33 (C3MB33)				8/16/32
H'FFFF 9214	CAN3 Mailbox Register 33 (C3MB33)				8/16/32
H'FFFF 9218	CAN3 Mailbox Register 33 (C3MB33)				8/16/32
H'FFFF 921C	CAN3 Mailbox Register 33 (C3MB33)				8/16/32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF 9220	CAN3 Mailbox Register 34 (C3MB34)								8/16/32
H'FFFF 9224	CAN3 Mailbox Register 34 (C3MB34)								8/16/32
H'FFFF 9228	CAN3 Mailbox Register 34 (C3MB34)								8/16/32
H'FFFF 922C	CAN3 Mailbox Register 34 (C3MB34)								8/16/32
H'FFFF 9230	CAN3 Mailbox Register 35 (C3MB35)								8/16/32
H'FFFF 9234	CAN3 Mailbox Register 35 (C3MB35)								8/16/32
H'FFFF 9238	CAN3 Mailbox Register 35 (C3MB35)								8/16/32
H'FFFF 923C	CAN3 Mailbox Register 35 (C3MB35)								8/16/32
H'FFFF 9240	CAN3 Mailbox Register 36 (C3MB36)								8/16/32
H'FFFF 9244	CAN3 Mailbox Register 36 (C3MB36)								8/16/32
H'FFFF 9248	CAN3 Mailbox Register 36 (C3MB36)								8/16/32
H'FFFF 924C	CAN3 Mailbox Register 36 (C3MB36)								8/16/32
H'FFFF 9250	CAN3 Mailbox Register 37 (C3MB37)								8/16/32
H'FFFF 9254	CAN3 Mailbox Register 37 (C3MB37)								8/16/32
H'FFFF 9258	CAN3 Mailbox Register 37 (C3MB37)								8/16/32
H'FFFF 925C	CAN3 Mailbox Register 37 (C3MB37)								8/16/32
H'FFFF 9260	CAN3 Mailbox Register 38 (C3MB38)								8/16/32
H'FFFF 9264	CAN3 Mailbox Register 38 (C3MB38)								8/16/32
H'FFFF 9268	CAN3 Mailbox Register 38 (C3MB38)								8/16/32
H'FFFF 926C	CAN3 Mailbox Register 38 (C3MB38)								8/16/32
H'FFFF 9270	CAN3 Mailbox Register 39 (C3MB39)								8/16/32
H'FFFF 9274	CAN3 Mailbox Register 39 (C3MB39)								8/16/32
H'FFFF 9278	CAN3 Mailbox Register 39 (C3MB39)								8/16/32
H'FFFF 927C	CAN3 Mailbox Register 39 (C3MB39)								8/16/32
H'FFFF 9280	CAN3 Mailbox Register 40 (C3MB40)								8/16/32
H'FFFF 9284	CAN3 Mailbox Register 40 (C3MB40)								8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 9288	CAN3 Mailbox Register 40 (C3MB40)				8/16/32
H'FFFF 928C	CAN3 Mailbox Register 40 (C3MB40)				8/16/32
H'FFFF 9290	CAN3 Mailbox Register 41 (C3MB41)				8/16/32
H'FFFF 9294	CAN3 Mailbox Register 41 (C3MB41)				8/16/32
H'FFFF 9298	CAN3 Mailbox Register 41 (C3MB41)				8/16/32
H'FFFF 929C	CAN3 Mailbox Register 41 (C3MB41)				8/16/32
H'FFFF 92A0	CAN3 Mailbox Register 42 (C3MB42)				8/16/32
H'FFFF 92A4	CAN3 Mailbox Register 42 (C3MB42)				8/16/32
H'FFFF 92A8	CAN3 Mailbox Register 42 (C3MB42)				8/16/32
H'FFFF 92AC	CAN3 Mailbox Register 42 (C3MB42)				8/16/32
H'FFFF 92B0	CAN3 Mailbox Register 43 (C3MB43)				8/16/32
H'FFFF 92B4	CAN3 Mailbox Register 43 (C3MB43)				8/16/32
H'FFFF 92B8	CAN3 Mailbox Register 43 (C3MB43)				8/16/32
H'FFFF 92BC	CAN3 Mailbox Register 43 (C3MB43)				8/16/32
H'FFFF 92C0	CAN3 Mailbox Register 44 (C3MB44)				8/16/32
H'FFFF 92C4	CAN3 Mailbox Register 44 (C3MB44)				8/16/32
H'FFFF 92C8	CAN3 Mailbox Register 44 (C3MB44)				8/16/32
H'FFFF 92CC	CAN3 Mailbox Register 44 (C3MB44)				8/16/32
H'FFFF 92D0	CAN3 Mailbox Register 45 (C3MB45)				8/16/32
H'FFFF 92D4	CAN3 Mailbox Register 45 (C3MB45)				8/16/32
H'FFFF 92D8	CAN3 Mailbox Register 45 (C3MB45)				8/16/32
H'FFFF 92DC	CAN3 Mailbox Register 45 (C3MB45)				8/16/32
H'FFFF 92E0	CAN3 Mailbox Register 46 (C3MB46)				8/16/32
H'FFFF 92E4	CAN3 Mailbox Register 46 (C3MB46)				8/16/32
H'FFFF 92E8	CAN3 Mailbox Register 46 (C3MB46)				8/16/32
H'FFFF 92EC	CAN3 Mailbox Register 46 (C3MB46)				8/16/32

Address	+0 Address Bit 31 Bit 24	+1 Address Bit 23 Bit 16	+2 Address Bit 15 Bit 8	+3 Address Bit 7 Bit 0	Access Size
H'FFFF 92F0	CAN3 Mailbox Register 47 (C3MB47)				8/16/32
H'FFFF 92F4	CAN3 Mailbox Register 47 (C3MB47)				8/16/32
H'FFFF 92F8	CAN3 Mailbox Register 47 (C3MB47)				8/16/32
H'FFFF 92FC	CAN3 Mailbox Register 47 (C3MB47)				8/16/32
H'FFFF 9300	CAN3 Mailbox Register 48 (C3MB48)				8/16/32
H'FFFF 9304	CAN3 Mailbox Register 48 (C3MB48)				8/16/32
H'FFFF 9308	CAN3 Mailbox Register 48 (C3MB48)				8/16/32
H'FFFF 930C	CAN3 Mailbox Register 48 (C3MB48)				8/16/32
H'FFFF 9310	CAN3 Mailbox Register 49 (C3MB49)				8/16/32
H'FFFF 9314	CAN3 Mailbox Register 49 (C3MB49)				8/16/32
H'FFFF 9318	CAN3 Mailbox Register 49 (C3MB49)				8/16/32
H'FFFF 931C	CAN3 Mailbox Register 49 (C3MB49)				8/16/32
H'FFFF 9320	CAN3 Mailbox Register 50 (C3MB50)				8/16/32
H'FFFF 9324	CAN3 Mailbox Register 50 (C3MB50)				8/16/32
H'FFFF 9328	CAN3 Mailbox Register 50 (C3MB50)				8/16/32
H'FFFF 932C	CAN3 Mailbox Register 50 (C3MB50)				8/16/32
H'FFFF 9330	CAN3 Mailbox Register 51 (C3MB51)				8/16/32
H'FFFF 9334	CAN3 Mailbox Register 51 (C3MB51)				8/16/32
H'FFFF 9338	CAN3 Mailbox Register 51 (C3MB51)				8/16/32
H'FFFF 933C	CAN3 Mailbox Register 51 (C3MB51)				8/16/32
H'FFFF 9340	CAN3 Mailbox Register 52 (C3MB52)				8/16/32
H'FFFF 9344	CAN3 Mailbox Register 52 (C3MB52)				8/16/32
H'FFFF 9348	CAN3 Mailbox Register 52 (C3MB52)				8/16/32
H'FFFF 934C	CAN3 Mailbox Register 52 (C3MB52)				8/16/32
H'FFFF 9350	CAN3 Mailbox Register 53 (C3MB53)				8/16/32
H'FFFF 9354	CAN3 Mailbox Register 53 (C3MB53)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 9358	CAN3 Mailbox Register 53 (C3MB53)				8/16/32
H'FFFF 935C	CAN3 Mailbox Register 53 (C3MB53)				8/16/32
H'FFFF 9360	CAN3 Mailbox Register 54 (C3MB54)				8/16/32
H'FFFF 9364	CAN3 Mailbox Register 54 (C3MB54)				8/16/32
H'FFFF 9368	CAN3 Mailbox Register 54 (C3MB54)				8/16/32
H'FFFF 936C	CAN3 Mailbox Register 54 (C3MB54)				8/16/32
H'FFFF 9370	CAN3 Mailbox Register 55 (C3MB55)				8/16/32
H'FFFF 9374	CAN3 Mailbox Register 55 (C3MB55)				8/16/32
H'FFFF 9378	CAN3 Mailbox Register 55 (C3MB55)				8/16/32
H'FFFF 937C	CAN3 Mailbox Register 55 (C3MB55)				8/16/32
H'FFFF 9380	CAN3 Mailbox Register 56 (C3MB56)				8/16/32
H'FFFF 9384	CAN3 Mailbox Register 56 (C3MB56)				8/16/32
H'FFFF 9388	CAN3 Mailbox Register 56 (C3MB56)				8/16/32
H'FFFF 938C	CAN3 Mailbox Register 56 (C3MB56)				8/16/32
H'FFFF 9390	CAN3 Mailbox Register 57 (C3MB57)				8/16/32
H'FFFF 9394	CAN3 Mailbox Register 57 (C3MB57)				8/16/32
H'FFFF 9398	CAN3 Mailbox Register 57 (C3MB57)				8/16/32
H'FFFF 939C	CAN3 Mailbox Register 57 (C3MB57)				8/16/32
H'FFFF 93A0	CAN3 Mailbox Register 58 (C3MB58)				8/16/32
H'FFFF 93A4	CAN3 Mailbox Register 58 (C3MB58)				8/16/32
H'FFFF 93A8	CAN3 Mailbox Register 58 (C3MB58)				8/16/32
H'FFFF 93AC	CAN3 Mailbox Register 58 (C3MB58)				8/16/32
H'FFFF 93B0	CAN3 Mailbox Register 59 (C3MB59)				8/16/32
H'FFFF 93B5	CAN3 Mailbox Register 59 (C3MB59)				8/16/32
H'FFFF 93B8	CAN3 Mailbox Register 59 (C3MB59)				8/16/32
H'FFFF 93BC	CAN3 Mailbox Register 59 (C3MB59)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 93C0	CAN3 Mailbox Register 60 (C3MB60)				8/16/32
H'FFFF 93C4	CAN3 Mailbox Register 60 (C3MB60)				8/16/32
H'FFFF 93C8	CAN3 Mailbox Register 60 (C3MB60)				8/16/32
H'FFFF 93CC	CAN3 Mailbox Register 60 (C3MB60)				8/16/32
H'FFFF 93D0	CAN3 Mailbox Register 61 (C3MB61)				8/16/32
H'FFFF 93D4	CAN3 Mailbox Register 61 (C3MB61)				8/16/32
H'FFFF 93D8	CAN3 Mailbox Register 61 (C3MB61)				8/16/32
H'FFFF 93DC	CAN3 Mailbox Register 61 (C3MB61)				8/16/32
H'FFFF 93E0	CAN3 Mailbox Register 62 (C3MB62)				8/16/32
H'FFFF 93E4	CAN3 Mailbox Register 62 (C3MB62)				8/16/32
H'FFFF 93E8	CAN3 Mailbox Register 62 (C3MB62)				8/16/32
H'FFFF 93EC	CAN3 Mailbox Register 62 (C3MB62)				8/16/32
H'FFFF 93F0	CAN3 Mailbox Register 63 (C3MB63)				8/16/32
H'FFFF 93F4	CAN3 Mailbox Register 63 (C3MB63)				8/16/32
H'FFFF 93F8	CAN3 Mailbox Register 63 (C3MB63)				8/16/32
H'FFFF 93FC	CAN3 Mailbox Register 63 (C3MB63)				8/16/32
H'FFFF 9400	CAN3 Mask Register 2 (C3MKR2)				8/16/32
H'FFFF 9404	CAN3 Mask Register 3 (C3MKR3)				8/16/32
H'FFFF 9408	CAN3 Mask Register 4 (C3MKR4)				8/16/32
H'FFFF 940C	CAN3 Mask Register 5 (C3MKR5)				8/16/32
H'FFFF 9410	CAN3 Mask Register 6 (C3MKR6)				8/16/32
H'FFFF 9414	CAN3 Mask Register 7 (C3MKR7)				8/16/32
H'FFFF 9418	CAN3 Mask Register 8 (C3MKR8)				8/16/32
H'FFFF 941C	CAN3 Mask Register 9 (C3MKR9)				8/16/32
H'FFFF 9420	CAN3 FIFO Received ID Compare Register 0 (C3FIDCR0)				8/16/32
H'FFFF 9424	CAN3 FIFO Received ID Compare Register 1 (C3FIDCR1)				8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 9428	CAN3 Mask Invalid Register 1 (C3MKIVLR1)				8/16/32
H'FFFF 942C	CAN3 Mailbox Interrupt Enable Register 1 (C3MIER1)				8/16/32
H'FFFF 9430	CAN3 Mask Register 0 (C3MKR0)				8/16/32
H'FFFF 9434	CAN3 Mask Register 1 (C3MKR1)				8/16/32
H'FFFF 9438	CAN3 Mask Invalid Register 0 (C3MKIVLR0)				8/16/32
H'FFFF 943C	CAN3 Mailbox Interrupt Enable Register 0 (C3MIER0)				8/16/32
:	(Reserved)				-
H'FFFF 9800	CAN3 Message Control Register 0 (C3MCTL0)	CAN3 Message Control Register 1 (C3MCTL1)	CAN3 Message Control Register 2 (C3MCTL2)	CAN3 Message Control Register 3 (C3MCTL3)	8/16/32
H'FFFF 9804	CAN3 Message Control Register 4 (C3MCTL4)	CAN3 Message Control Register 5 (C3MCTL5)	CAN3 Message Control Register 6 (C3MCTL6)	CAN3 Message Control Register 7 (C3MCTL7)	8/16/32
H'FFFF 9808	CAN3 Message Control Register 8 (C3MCTL8)	CAN3 Message Control Register 9 (C3MCTL9)	CAN3 Message Control Register 10 (C3MCTL10)	CAN3 Message Control Register 11 (C3MCTL11)	8/16/32
H'FFFF 980C	CAN3 Message Control Register 12 (C3MCTL12)	CAN3 Message Control Register 13 (C3MCTL13)	CAN3 Message Control Register 14 (C3MCTL14)	CAN3 Message Control Register 15 (C3MCTL15)	8/16/32
H'FFFF 9810	CAN3 Message Control Register 16 (C3MCTL16)	CAN3 Message Control Register 17 (C3MCTL17)	CAN3 Message Control Register 18 (C3MCTL18)	CAN3 Message Control Register 19 (C3MCTL19)	8/16/32
H'FFFF 9814	CAN3 Message Control Register 20 (C3MCTL20)	CAN3 Message Control Register 21 (C3MCTL21)	CAN3 Message Control Register 22 (C3MCTL22)	CAN3 Message Control Register 23 (C3MCTL23)	8/16/32
H'FFFF 9818	CAN3 Message Control Register 24 (C3MCTL24)	CAN3 Message Control Register 25 (C3MCTL25)	CAN3 Message Control Register 26 (C3MCTL26)	CAN3 Message Control Register 27 (C3MCTL27)	8/16/32
H'FFFF 981C	CAN3 Message Control Register 28 (C3MCTL28)	CAN3 Message Control Register 29 (C3MCTL29)	CAN3 Message Control Register 30 (C3MCTL30)	CAN3 Message Control Register 31 (C3MCTL31)	8/16/32
H'FFFF 9820	CAN3 Message Control Register 32 (C3MCTL32)	CAN3 Message Control Register 33 (C3MCTL33)	CAN3 Message Control Register 34 (C3MCTL34)	CAN3 Message Control Register 35 (C3MCTL35)	8/16/32
H'FFFF 9824	CAN3 Message Control Register 36 (C3MCTL36)	CAN3 Message Control Register 37 (C3MCTL37)	CAN3 Message Control Register 38 (C3MCTL38)	CAN3 Message Control Register 39 (C3MCTL39)	8/16/32
H'FFFF 9828	CAN3 Message Control Register 40 (C3MCTL40)	CAN3 Message Control Register 41 (C3MCTL41)	CAN3 Message Control Register 42 (C3MCTL42)	CAN3 Message Control Register 43 (C3MCTL43)	8/16/32
H'FFFF 982C	CAN3 Message Control Register 44 (C3MCTL44)	CAN3 Message Control Register 45 (C3MCTL45)	CAN3 Message Control Register 46 (C3MCTL46)	CAN3 Message Control Register 47 (C3MCTL47)	8/16/32
H'FFFF 9830	CAN3 Message Control Register 48 (C3MCTL48)	CAN3 Message Control Register 49 (C3MCTL49)	CAN3 Message Control Register 50 (C3MCTL50)	CAN3 Message Control Register 51 (C3MCTL51)	8/16/32
H'FFFF 9834	CAN3 Message Control Register 52 (C3MCTL52)	CAN3 Message Control Register 53 (C3MCTL53)	CAN3 Message Control Register 54 (C3MCTL54)	CAN3 Message Control Register 55 (C3MCTL55)	8/16/32
H'FFFF 9838	CAN3 Message Control Register 56 (C3MCTL56)	CAN3 Message Control Register 57 (C3MCTL57)	CAN3 Message Control Register 58 (C3MCTL58)	CAN3 Message Control Register 59 (C3MCTL59)	8/16/32
H'FFFF 983C	CAN3 Message Control Register 60 (C3MCTL60)	CAN3 Message Control Register 61 (C3MCTL61)	CAN3 Message Control Register 62 (C3MCTL62)	CAN3 Message Control Register 63 (C3MCTL63)	8/16/32
H'FFFF 9840	CAN3 Control Register (C3CTLR)		CAN3 Status Register (C3STR)		8/16/32
H'FFFF 9844	CAN3 Bit Configuration Register (C3BCR)			CAN3 Clock Select Register (C3CLKR)	8/16/32
H'FFFF 9848	CAN3 Receive FIFO Control Register (C3RFCR)	CAN3 Receive FIFO Pointer Control Register (C3RFPCR)	CAN3 Transmit FIFO Control Register (C3TFCR)	CAN3 Transmit FIFO Pointer Control Register (C3TFPCR)	8/16/32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF 984C	CAN3 Error Interrupt Enable Register (C3EIER)	CAN3Error Interrupt Factor Judge Register (C3EIFR)	CAN3 Receive Error Count Register (C3RECR)	CAN3 Transmit Error Count Register (C3TECR)	8/16/32
H'FFFF 9850	CAN3 Error Code Store Register (C3ECSR)	CAN3 Channel Search Support Register (C3CSSR)	CAN3 Mailbox Search Status Register (C3MSSR)	CAN3 Mailbox Search Mode Register (C3MSMR)	8/16/32
H'FFFF 9854	CAN3 Time Stamp Register (C3TSR)		CAN3 Acceptance Filter Support Register (C3AFSR)		8/16/32
H'FFFF 9858	CAN3 Test Control Register (C3TCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF 9860	CAN3 Interrupt Enable Register (C3IER)	CAN3 Interrupt Status Register (C3ISR)	(Reserved)	CAN3 Mailbox Search Mask Register (C3MBSMR)	8/16, 8/16, -, 8
:	(Reserved)				-
H'FFFF B000	RSPI0 Control Register (SP0CR)	RSPI0 Slave Select Polarity Register (SP0SSLP)	RSPI0 Pin Control Register (SP0PCR)	RSPI0 Status Register (SP0SR)	8/16
H'FFFF B004	RSPI0 Data Register (SP0DR)				16/32
H'FFFF B008	RSPI0 Sequence Control Register (SP0SCR)	RSPI0 Sequence Status Register (SP0SSR)	RSPI0 Bit Rate Register (SP0BR)	RSPI0 Data Control Register (SP0DCR)	8/16
H'FFFF B00C	RSPI0 Clock Delay Register (SP0CKD)	RSPI0 Slave Select Negation Delay Register (SP0SSLND)	RSPI0 Next-Access Delay Register (SP0ND)	(Reserved)	8/16, 8/16, 8, -
H'FFFF B010	RSPI0 Command Register 0 (SP0CMD0)		RSPI0 Command Register 1 (SP0CMD1)		16
H'FFFF B014	RSPI0 Command Register 2 (SP0CMD2)		RSPI0 Command Register 3 (SP0CMD3)		16
:	(Reserved)				-
H'FFFF B100	RSPI1 Control Register (SP1CR)	RSPI1 Slave Select Polarity Register (SP1SSLP)	RSPI1 Pin Control Register (SP1PCR)	RSPI1 Status Register (SP1SR)	8/16
H'FFFF B104	RSPI1 Data Register (SP1DR)				16/32
H'FFFF B108	RSPI1 Sequence Control Register (SP1SCR)	RSPI1 Sequence Status Register (SP1SSR)	RSPI1 Bit Rate Register (SP1BR)	RSPI1 Data Control Register (SP1DCR)	8/16
H'FFFF B10C	RSPI1 Clock Delay Register (SP1CKD)	RSPI1 Slave Select Negation Delay Register (SP1SSLND)	RSPI1 Next-Access Delay Register (SP1ND)	(Reserved)	8/16, 8/16, 8, -
H'FFFF B110	RSPI1 Command Register 0 (SP1CMD0)		RSPI1 Command Register 1 (SP1CMD1)		16
H'FFFF B114	RSPI1 Command Register 2 (SP1CMD2)		RSPI1 Command Register 3 (SP1CMD3)		16
:	(Reserved)				-
H'FFFF B200	RSPI2 Control Register (SP2CR)	RSPI2 Slave Select Polarity Register (SP2SSLP)	RSPI2 Pin Control Register (SP2PCR)	RSPI2 Status Register (SP2SR)	8/16
H'FFFF B204	RSPI2 Data Register (SP2DR)				16/32
H'FFFF B208	RSPI2 Sequence Control Register (SP2SCR)	RSPI2 Sequence Status Register (SP2SSR)	RSPI2 Bit Rate Register (SP2BR)	RSPI2 Data Control Register (SP2DCR)	8/16
H'FFFF B20C	RSPI2 Clock Delay Register (SP2CKD)	RSPI2 Slave Select Negation Delay Register (SP2SSLND)	RSPI2 Next-Access Delay Register (SP2ND)	(Reserved)	8/16, 8/16, 8, -
H'FFFF B210	RSPI2 Command Register 0 (SP2CMD0)		RSPI2 Command Register 1 (SP2CMD1)		16
H'FFFF B214	RSPI2 Command Register 2 (SP2CMD2)		RSPI2 Command Register 3 (SP2CMD3)		16
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF C000	SC0 Serial Mode Register (SC0SMR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C004	SC0 Bit Rate Register (SC0BRR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C008	SC0 Serial Control Register (SC0SCR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C00C	SC0 Transmit FIFO Data Register (SC0FTDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C010	SC0 Serial Status Register (SC0FSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C014	SC0 Receive FIFO Data Register (SC0FRDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C018	SC0FIFO Control Register (SC0FCR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C01C	SC0FIFO Data Count Set Register (SC0FDR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C020	SC0 Serial Port Register (SC0SPTR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C024	SC0 Line Status Register (SC0LSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C028	SC0 Serial Extension Mode Register (SC0EMR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFFF C100	SC1 Serial Mode Register (SC1SMR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C104	SC1 Bit Rate Register (SC1BRR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C108	SC1 Serial Control Register (SC1SCR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C10C	SC1 Transmit FIFO Data Register (SC1FTDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C110	SC1 Serial Status Register (SC1FSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C114	SC1 Receive FIFO Data Register (SC1FRDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C118	SC1FIFO Control Register (SC1FCR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C11C	SC1FIFO Data Count Set Register (SC1FDR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C120	SC1 Serial Port Register (SC1SPTR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C124	SC1 Line Status Register (SC1LSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C128	SC1 Serial Extension Mode Register (SC1EMR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFFF C200	SC2 Serial Mode Register (SC2SMR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C204	SC2 Bit Rate Register (SC2BRR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C208	SC2 Serial Control Register (SC2SCR)		(Reserved)	(Reserved)	16, -, -

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF C20C	SC2 Transmit FIFO Data Register (SC2FTDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C210	SC2 Serial Status Register (SC2FSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C214	SC2 Receive FIFO Data Register (SC2FRDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C218	SC2FIFO Control Register (SC2FCR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C21C	SC2FIFO Data Count Set Register (SC2FDR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C220	SC2 Serial Port Register (SC2SPTR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C224	SC2 Line Status Register (SC2SLSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C228	SC2 Serial Extension Mode Register (SC2EMR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFFF C300	SC3 Serial Mode Register (SC3SMR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C304	SC3 Bit Rate Register (SC3BRR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C308	SC3 Serial Control Register (SC3SCR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C30C	SC3 Transmit FIFO Data Register (SC3FTDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C310	SC3 Serial Status Register (SC3FSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C314	SC3 Receive FIFO Data Register (SC3FRDR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF C318	SC3FIFO Control Register (SC3FCR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C31C	SC3FIFO Data Count Set Register (SC3FDR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C320	SC3 Serial Port Register (SC3SPTR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C324	SC3 Line Status Register (SC3SLSR)		(Reserved)	(Reserved)	16, -, -
H'FFFF C328	SC3 Serial Extension Mode Register (SC3EMR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFFF D004	TM Start Register (TMSTR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF D008	TM0 Constant Register (TM0COR)				32
H'FFFF D00C	TM0 Counter (TM0CNT)				32
H'FFFF D010	TM0 Control Register (TM0CR)		(Reserved)	(Reserved)	16, -, -
H'FFFF D014	TM1 Constant Register (TM1COR)				32
H'FFFF D018	TM1 Counter (TM1CNT)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF D01C	TM1 Control Register (TM1CR)		(Reserved)	(Reserved)	16, -, -
H'FFFF D020	TM2 Constant Register (TM2COR)				32
H'FFFF D024	TM2 Counter (TM2CNT)				32
H'FFFF D028	TM2 Control Register (TM2CR)		(Reserved)	(Reserved)	16, -, -
:	(Reserved)				-
H'FFFF E000	ATU-IIIS Master Enable Register (ATUENR)		ATU-IIIS Clock Bus Control Register (ATCBCNT)	ATU-IIIS Noise Cancellation Mode Register (ATNCMR)	16, 8, 8
:	(Reserved)				-
H'FFFF E010	ATU-IIIS Interrupt Select Register A0 (ATISRA0)	ATU-IIIS Interrupt Select Register A1 (ATISRA1)	(Reserved)	(Reserved)	8, 8, -, -
H'FFFF E014	ATU-IIIS Interrupt Select Register F (ATISRF)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E018	ATU-IIIS Interrupt Select Register G (ATISRG)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF E020	ATU-IIIS Interrupt Select Register TOU0 (ATISRT0)	ATU-IIIS Interrupt Select Register TOU1 (ATISRT1)	ATU-IIIS Interrupt Select Register TOU2 (ATISRT2)	ATU-IIIS Interrupt Select Register TOU3 (ATISRT3)	8
H'FFFF E024	ATU-IIIS Interrupt Select Register TOU4 (ATISRT4)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
:	(Reserved)				-
H'FFFF E100	ATU-IIIS ATU-IIIS Prescaler Register 0 (ATPSCR0)		ATU-IIIS ATU-IIIS Prescaler Register 1 (ATPSCR1)		16
H'FFFF E104	ATU-IIIS ATU-IIIS Prescaler Register 2 (ATPSCR2)		ATU-IIIS ATU-IIIS Prescaler Register 3 (ATPSCR3)		16
:	(Reserved)				-
H'FFFF E200	(Reserved)	(Reserved)	TA0 Control Register (TA0CR)	(Reserved)	-, -, 8, -
H'FFFF E204	TA0I/O Control Register 1 (TA0IO1)		TA0I/O Control Register 2 (TA0IO2)		16
H'FFFF E208	TA0 Status Register (TA0SR)	TA0 Interrupt Enable Register (TA0IER)	(Reserved)	(Reserved)	8, 8, -, -
:	(Reserved)				-
H'FFFF E210	TA00 Noise Canceler Counter (TA00NCNT)	TA00 Noise Canceler Register (TA00NCR)	TA01 Noise Canceler Counter (TA01NCNT)	TA01 Noise Canceler Register (TA01NCR)	8
H'FFFF E214	TA02 Noise Canceler Counter (TA02NCNT)	TA02 Noise Canceler Register (TA02NCR)	TA03 Noise Canceler Counter (TA03NCNT)	TA03 Noise Canceler Register (TA03NCR)	8
H'FFFF E218	TA04 Noise Canceler Counter (TA04NCNT)	TA04 Noise Canceler Register (TA04NCR)	(Reserved)	(Reserved)	8
:	(Reserved)				-
H'FFFF E220	TA0 Free-Running Counter (TA0TCNT)				32
:	(Reserved)				-
H'FFFF E228	TA00 Input Capture Register (TA00ICR)				32
H'FFFF E22C	TA01 Input Capture Register (TA01ICR)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF E230	TA02 Input Capture Register (TA02ICR)				32
H'FFFF E234	TA03 Input Capture Register (TA03ICR)				32
H'FFFF E238	TA04 Input Capture Register (TA04ICR)				32
:	(Reserved)				-
H'FFFF E300	(Reserved)	(Reserved)	TA1 Control Register (TA1CR)	(Reserved)	-, -, 8, -
H'FFFF E304	TA1I/O Control Register 1 (TA1IO1)		TA1I/O Control Register 2 (TA1IO2)		16
H'FFFF E308	TA1 Status Register (TA1SR)	TA1 Interrupt Enable Register (TA1IER)	(Reserved)	(Reserved)	8, 8, -, -
:	(Reserved)				-
H'FFFF E310	TA10 Noise Canceler Counter (TA10NCNT)	TA10 Noise Canceler Register (TA10NCR)	TA11 Noise Canceler Counter (TA11NCNT)	TA11 Noise Canceler Register (TA11NCR)	8
H'FFFF E314	TA12 Noise Canceler Counter (TA12NCNT)	TA12 Noise Canceler Register (TA12NCR)	(Reserved)	(Reserved)	8
H'FFFF E318	TA14 Noise Canceler Counter (TA14NCNT)	TA14 Noise Canceler Register (TA14NCR)	TA15 Noise Canceler Counter (TA15NCNT)	TA15 Noise Canceler Register (TA15NCR)	8
:	(Reserved)				-
H'FFFF E320	TA1 Free-Running Counter (TA1TCNT)				32
:	(Reserved)				-
H'FFFF E328	TA10 Input Capture Register (TA10ICR)				32
H'FFFF E32C	TA11 Input Capture Register (TA11ICR)				32
H'FFFF E330	TA12 Input Capture Register (TA12ICR)				32
:	(Reserved)				-
H'FFFF E338	TA14 Input Capture Register (TA14ICR)				32
H'FFFF E33C	TA15 Input Capture Register (TA15ICR)				32
H'FFFF E400	TF Start Register (TFSTR)				32
H'FFFF E404	TF Noise Canceller Control Register (TFNCCR)				32
:	(Reserved)				-
H'FFFF E410	TF0 Noise Canceler Counter A (TF0NCNTA)	TF0 Noise Canceler Register A (TF0NCRA)	TF1 Noise Canceler Counter A (TF1NCNTA)	TF1 Noise Canceler Register A (TF1NCRA)	8
H'FFFF E414	TF2 Noise Canceler Counter A (TF2NCNTA)	TF2 Noise Canceler Register A (TF2NCRA)	(Reserved)	(Reserved)	8
:	(Reserved)				-
H'FFFF E450	TF0 Noise Canceler Counter B (TF0NCNTB)	TF0 Noise Canceler Register B (TF0NCRB)	TF1 Noise Canceler Counter B (TF1NCNTB)	TF1 Noise Canceler Register B (TF1NCRB)	8
H'FFFF E454	TF2 Noise Canceler Counter B (TF2NCNTB)	TF2 Noise Canceler Register B (TF2NCRB)	(Reserved)	(Reserved)	8
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF E480	TF0 Control Register (TF0CR)	TF0 Interrupt Enable Register (TF0IER)	(Reserved)	TF0 Status Register (TF0SR)	8, 8, -, 8
H'FFFF E484	TF0 Timer Counter A (TF0ECNTA)				32
H'FFFF E488	TF0 Event Counter (TF0ECNTB)		TF0 General Register B (TF0GRB)		16
H'FFFF E48C	TF0 Timer Counter C (TF0ECNTC)				32
H'FFFF E490	TF0 General Register A (TF0GRA)				32
H'FFFF E494	TF0 Capture Output Register (TF0CDR)				32
H'FFFF E498	TF0 General Register C (TF0GRC)				32
H'FFFF E49C	TF0 General Register D (TF0GRD)				32
H'FFFF E4A0	TF1 Control Register (TF1CR)	TF1 Interrupt Enable Register (TF1IER)	(Reserved)	TF1 Status Register (TF1SR)	8, 8, -, 8
H'FFFF E4A4	TF1 Timer Counter A (TF1ECNTA)				32
H'FFFF E4A8	TF1 Event Counter (TF1ECNTB)		TF1 General Register B (TF1GRB)		16
H'FFFF E4AC	TF1 Timer Counter C (TF1ECNTC)				32
H'FFFF E4B0	TF1 General Register A (TF1GRA)				32
H'FFFF E4B4	TF1 Capture Output Register (TF1CDR)				32
H'FFFF E4B8	TF1 General Register C (TF1GRC)				32
H'FFFF E4BC	TF1 General Register D (TF1GRD)				32
H'FFFF E4C0	TF2 Control Register (TF2CR)	TF2 Interrupt Enable Register (TF2IER)	(Reserved)	TF2 Status Register (TF2SR)	8, 8, -, 8
H'FFFF E4C4	TF2 Timer Counter A (TF2ECNTA)				32
H'FFFF E4C8	TF2 Event Counter (TF2ECNTB)		TF2 General Register B (TF2GRB)		16
H'FFFF E4CC	TF2 Timer Counter C (TF2ECNTC)				32
H'FFFF E4D0	TF2 General Register A (TF2GRA)				32
H'FFFF E4D4	TF2 Capture Output Register (TF2CDR)				32
H'FFFF E4D8	TF2 General Register C (TF2GRC)				32
H'FFFF E4DC	TF2 General Register D (TF2GRD)				32
:	(Reserved)				-
H'FFFF E500	(Reserved)	TG Start Register (TGSTR)	(Reserved)	(Reserved)	-, 8, -, -
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF E580	TG0 Control Register (TG0CR)	TG0 Status Register (TG0SR)	(Reserved)	(Reserved)	8, 8, -, -
H'FFFF E584	TG0 Counter (TG0CNT)		TG0 Compare Match Register (TG0OCR)		16
:	(Reserved)				-
H'FFFF E590	TG1 Control Register (TG1CR)	TG1 Status Register (TG1SR)	(Reserved)	(Reserved)	8, 8, -, -
H'FFFF E594	TG1 Counter (TG1CNT)		TG1 Compare Match Register (TG1OCR)		16
:	(Reserved)				-
H'FFFF E5A0	TG2 Control Register (TG2CR)	TG2 Status Register (TG2SR)	(Reserved)	(Reserved)	8, 8, -, -
H'FFFF E5A4	TG2 Counter (TG2CNT)		TG2 Compare Match Register (TG2OCR)		16
:	(Reserved)				-
H'FFFF E5B0	TG3 Control Register (TG3CR)	TG3 Status Register (TG3SR)	(Reserved)	(Reserved)	8, 8, -, -
H'FFFF E5B4	TG3 Counter (TG3CNT)		TG3 Compare Match Register (TG3OCR)		16
:	(Reserved)				-
H'FFFF E5C0	TG4 Control Register (TG4CR)	TG4 Status Register (TG4SR)	(Reserved)	(Reserved)	8, 8, -, -
H'FFFF E5C4	TG4 Counter (TG4CNT)		TG4 Compare Match Register (TG4OCR)		16
:	(Reserved)				-
H'FFFF E5D0	TG5 Control Register (TG5CR)	TG5 Status Register (TG5SR)	(Reserved)	(Reserved)	8, 8, -, -
H'FFFF E5D4	TG5 Counter (TG5CNT)		TG5 Compare Match Register (TG5OCR)		16
:	(Reserved)				-
H'FFFF E600	TOU0 Control Register (TO0CR)	TOU0 Interrupt Enable Register (TO0IER)	TOU0 Output Control Register (TO0OUCR)	TOU0 Status Register (TO0SR)	8
H'FFFF E604	TOU0 Counter Enable Protect Register (TO0CEPR)	Flip-Flop Output Protect Register for TOU0 Short-Circuit Prevention Function (TO0SHFFPR)	TOU0 Flip-Flop Output Protect Register (TO0FFPR)	(Reserved)	8, 8, 8, -
H'FFFF E608	TOU0 Counter Enable Register (TO0CENR)	Flip-Flop Output Data Register for TOU0 Short-Circuit Prevention Function (TO0SHFFDR)	TOU0 Flip-Flop Output Data Register (TO0FFDR)	(Reserved)	8, 8, 8, -
H'FFFF E60C	TOU0 Noise Canceler Control Register (TO0NCCR)	(Reserved)	TOU0 Noise Canceler Counter (TO0NCNT)	TOU0 Noise Canceler Register (TO0NCR)	8, -, 8, 8
H'FFFF E610	TOU0PWMOFF Input Processing Register (TO0POCR)		(Reserved)	TOU0PWMOFF Function Enable Register (TO0POER)	16, -, 8
H'FFFF E614	TOU0PWM Output-Prohibit Control Register (TO0PODISCR)		(Reserved)	TOU0PWM Output-Prohibit Level Control Register (TO0POLVCR)	16, -, 8
:	(Reserved)				-
H'FFFF E620	TOU00 Mode Control Register (TO00MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E624	TOU00 Counter (TO00CNT)				32
H'FFFF E628	TOU00 Reload Register (TO00RLD)				32
:	(Reserved)				-

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF E630	TOU01 Mode Control Register (TO01MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E634	TOU01 Counter (TO01CNT)								32
H'FFFF E638	TOU01 Reload Register (TO01RLD)								32
:	(Reserved)								-
H'FFFF E640	TOU02 Mode Control Register (TO02MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E644	TOU02 Counter (TO02CNT)								32
H'FFFF E648	TOU02 Reload Register (TO02RLD)								32
:	(Reserved)								-
H'FFFF E650	TOU03 Mode Control Register (TO03MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E654	TOU03 Counter (TO03CNT)								32
H'FFFF E658	TOU03 Reload Register (TO03RLD)								32
:	(Reserved)								-
H'FFFF E660	TOU04 Mode Control Register (TO04MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E664	TOU04 Counter (TO04CNT)								32
H'FFFF E668	TOU04 Reload Register (TO04RLD)								32
:	(Reserved)								-
H'FFFF E670	TOU05 Mode Control Register (TO05MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E674	TOU05 Counter (TO05CNT)								32
H'FFFF E678	TOU05 Reload Register (TO05RLD)								32
:	(Reserved)								-
H'FFFF E680	TOU06 Mode Control Register (TO06MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E684	TOU06 Counter (TO06CNT)								32
H'FFFF E688	TOU06 Reload Register (TO06RLD)								32
:	(Reserved)								-
H'FFFF E690	TOU07 Mode Control Register (TO07MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E694	TOU07 Counter (TO07CNT)								32
H'FFFF E698	TOU07 Reload Register (TO07RLD)								32
:	(Reserved)								-
H'FFFF E700	TOU1 Control Register (TO1CR)		TOU1 Interrupt Enable Register (TO1IER)		TOU1 Output Control Register (TO1OUCR)		TOU1 Status Register (TO1SR)		8

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF E704	TOU1 Counter Enable Protect Register (TO1CEPR)		Flip-Flop Output Protect Register for TOU1 Short-Circuit Prevention Function (TO1SHFFPR)		TOU1 Flip-Flop Output Protect Register (TO1FFPR)		(Reserved)		8, 8, 8, -
H'FFFF E708	TOU1 Counter Enable Register (TO1CENR)		Flip-Flop Output Data Register for TOU1 Short-Circuit Prevention Function (TO1SHFFDR)		TOU1 Flip-Flop Output Data Register (TO1FFDR)		(Reserved)		8, 8, 8, -
H'FFFF E70C	TOU1 Noise Canceler Control Register (TO1NCCR)		(Reserved)		TOU1 Noise Canceler Counter (TO1NCNT)		TOU1 Noise Canceler Register (TO1NCR)		8, -, 8, 8
H'FFFF E710	TOU1PWMOFF Input Processing Register (TO1POCR)				(Reserved)		TOU1PWMOFF Function Enable Register (TO1POER)		16, -, 8
H'FFFF E714	TOU1PWM Output-Prohibit Control Register (TO1PODISCR)				(Reserved)		TOU1PWM Output-Prohibit Level Control Register (TO1POLVCR)		16, -, 8
:	(Reserved)								-
H'FFFF E720	TOU10 Mode Control Register (TO10MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E724	TOU10 Counter (TO10CNT)								32
H'FFFF E728	TOU10 Reload Register (TO10RLD)								32
:	(Reserved)								-
H'FFFF E730	TOU11 Mode Control Register (TO11MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E734	TOU11 Counter (TO11CNT)								32
H'FFFF E738	TOU11 Reload Register (TO11RLD)								32
:	(Reserved)								-
H'FFFF E740	TOU12 Mode Control Register (TO12MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E744	TOU12 Counter (TO12CNT)								32
H'FFFF E748	TOU12 Reload Register (TO12RLD)								32
:	(Reserved)								-
H'FFFF E750	TOU13 Mode Control Register (TO13MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E754	TOU13 Counter (TO13CNT)								32
H'FFFF E758	TOU13 Reload Register (TO13RLD)								32
:	(Reserved)								-
H'FFFF E760	TOU14 Mode Control Register (TO14MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E764	TOU14 Counter (TO14CNT)								32
H'FFFF E768	TOU14 Reload Register (TO14RLD)								32
:	(Reserved)								-
H'FFFF E770	TOU15 Mode Control Register (TO15MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E774	TOU15 Counter (TO15CNT)								32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF E778	TOU15 Reload Register (TO15RLD)								32
:	(Reserved)								-
H'FFFF E780	TOU16 Mode Control Register (TO16MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E784	TOU16 Counter (TO16CNT)								32
H'FFFF E788	TOU16 Reload Register (TO16RLD)								32
:	(Reserved)								-
H'FFFF E790	TOU17 Mode Control Register (TO17MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E794	TOU17 Counter (TO17CNT)								32
H'FFFF E798	TOU17 Reload Register (TO17RLD)								32
:	(Reserved)								-
H'FFFF E800	TOU2 Control Register (TO2CR)		TOU2 Interrupt Enable Register (TO2IER)		TOU2 Output Control Register (TO2OUCR)		TOU2 Status Register (TO2SR)		8
H'FFFF E804	TOU2 Counter Enable Protect Register (TO2CEPR)		Flip-Flop Output Protect Register for TOU2 Short-Circuit Prevention Function (TO2SHFFPR)		TOU2 Flip-Flop Output Protect Register (TO2FFPR)		(Reserved)		8, 8, 8, -
H'FFFF E808	TOU2 Counter Enable Register (TO2CENR)		Flip-Flop Output Data Register for TOU2 Short-Circuit Prevention Function (TO2SHFFDR)		TOU2 Flip-Flop Output Data Register (TO2FFDR)		(Reserved)		8, 8, 8, -
:	(Reserved)								-
H'FFFF E820	TOU20 Mode Control Register (TO20MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E824	TOU20 Counter (TO20CNT)								32
H'FFFF E828	TOU20 Reload Register (TO20RLD)								32
:	(Reserved)								-
H'FFFF E830	TOU21 Mode Control Register (TO21MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E834	TOU21 Counter (TO21CNT)								32
H'FFFF E838	TOU21 Reload Register (TO21RLD)								32
:	(Reserved)								-
H'FFFF E840	TOU22 Mode Control Register (TO22MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E844	TOU22 Counter (TO22CNT)								32
H'FFFF E848	TOU22 Reload Register (TO22RLD)								32
:	(Reserved)								-
H'FFFF E850	TOU23 Mode Control Register (TO23MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E854	TOU23 Counter (TO23CNT)								32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF E858	TOU23 Reload Register (TO23RLD)								32
:	(Reserved)								-
H'FFFF E860	TOU24 Mode Control Register (TO24MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E864	TOU24 Counter (TO24CNT)								32
H'FFFF E868	TOU24 Reload Register (TO24RLD)								32
:	(Reserved)								-
H'FFFF E870	TOU25 Mode Control Register (TO25MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E874	TOU25 Counter (TO25CNT)								32
H'FFFF E878	TOU25 Reload Register (TO25RLD)								32
:	(Reserved)								-
H'FFFF E880	TOU26 Mode Control Register (TO26MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E884	TOU26 Counter (TO26CNT)								32
H'FFFF E888	TOU26 Reload Register (TO26RLD)								32
:	(Reserved)								-
H'FFFF E890	TOU27 Mode Control Register (TO27MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E894	TOU27 Counter (TO27CNT)								32
H'FFFF E898	TOU27 Reload Register (TO27RLD)								32
:	(Reserved)								-
H'FFFF E900	TOU3 Control Register (TO3CR)		TOU3 Interrupt Enable Register (TO3IER)		TOU3 Output Control Register (TO3OUCR)		TOU3 Status Register (TO3SR)		8
H'FFFF E904	TOU3 Counter Enable Protect Register (TO3CEPR)		Flip-Flop Output Protect Register for TOU3 Short-Circuit Prevention Function (TO3SHFFPR)		TOU3 Flip-Flop Output Protect Register (TO3FFPR)		(Reserved)		8, 8, 8, -
H'FFFF E908	TOU3 Counter Enable Register (TO3CENR)		Flip-Flop Output Data Register for TOU3 Short-Circuit Prevention Function (TO3SHFFDR)		TOU3 Flip-Flop Output Data Register (TO3FFDR)		(Reserved)		8, 8, 8, -
H'FFFF E90C	TOU3 Noise Canceler Control Register (TO3NCCR)		(Reserved)		TOU3 Noise Canceler Counter (TO3NCNT)		TOU3 Noise Canceler Register (TO3NCR)		8, -, 8, 8
H'FFFF E910	TOU3PWMOFF Input Processing Register (TO3POCR)				(Reserved)		TOU3PWMOFF Function Enable Register (TO3POER)		16, -, 8
H'FFFF E914	TOU3PWM Output-Prohibit Control Register (TO3PODISCR)				(Reserved)		TOU3PWM Output-Prohibit Level Control Register (TO3POLVCR)		16, -, 8
:	(Reserved)								-
H'FFFF E920	TOU30 Mode Control Register (TO30MCR)		(Reserved)		(Reserved)		(Reserved)		8, -, -, -
H'FFFF E924	TOU30 Counter (TO30CNT)								32
H'FFFF E928	TOU30 Reload Register (TO30RLD)								32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
:	(Reserved)				-
H'FFFF E930	TOU31 Mode Control Register (TO31MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E934	TOU31 Counter (TO31CNT)				32
H'FFFF E938	TOU31 Reload Register (TO31RLD)				32
:	(Reserved)				-
H'FFFF E940	TOU32 Mode Control Register (TO32MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E944	TOU32 Counter (TO32CNT)				32
H'FFFF E948	TOU32 Reload Register (TO32RLD)				32
:	(Reserved)				-
H'FFFF E950	TOU33 Mode Control Register (TO33MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E954	TOU33 Counter (TO33CNT)				32
H'FFFF E958	TOU33 Reload Register (TO33RLD)				32
:	(Reserved)				-
H'FFFF E960	TOU34 Mode Control Register (TO34MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E964	TOU34 Counter (TO34CNT)				32
H'FFFF E968	TOU34 Reload Register (TO34RLD)				32
:	(Reserved)				-
H'FFFF E970	TOU35 Mode Control Register (TO35MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E974	TOU35 Counter (TO35CNT)				32
H'FFFF E978	TOU35 Reload Register (TO35RLD)				32
:	(Reserved)				-
H'FFFF E980	TOU36 Mode Control Register (TO36MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E984	TOU36 Counter (TO36CNT)				32
H'FFFF E988	TOU36 Reload Register (TO36RLD)				32
:	(Reserved)				-
H'FFFF E990	TOU37 Mode Control Register (TO37MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF E994	TOU37 Counter (TO37CNT)				32
H'FFFF E998	TOU37 Reload Register (TO37RLD)				32
:	(Reserved)				-

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF EA00	TOU4 Control Register (TO4CR)	TOU4 Interrupt Enable Register (TO4IER)	TOU4 Output Control Register (TO4OUCR)	TOU4 Status Register (TO4SR)	8
H'FFFF EA04	TOU4 Counter Enable Protect Register (TO4CEPR)	Flip-Flop Output Protect Register for TOU4 Short-Circuit Prevention Function (TO4SHFFPR)	TOU4 Flip-Flop Output Protect Register (TO4FFPR)	(Reserved)	8, 8, 8, -
H'FFFF EA08	TOU4 Counter Enable Register (TO4CENR)	Flip-Flop Output Data Register for TOU4 Short-Circuit Prevention Function (TO4SHFFDR)	TOU4 Flip-Flop Output Data Register (TO4FFDR)	(Reserved)	8, 8, 8, -
H'FFFF EA0C	TOU4 Noise Canceler Control Register (TO4NCCR)	(Reserved)	TOU4 Noise Canceler Counter (TO4NCNT)	TOU4 Noise Canceler Register (TO4NCR)	8, -, 8, 8
H'FFFF EA10	TOU4PWMOFF Input Processing Register (TO4POCR)		(Reserved)	TOU4PWMOFF Function Enable Register (TO4POER)	16, -, 8
H'FFFF EA14	TOU4PWM Output-Prohibit Control Register (TO4PODISCR)		(Reserved)	TOU4PWM Output-Prohibit Level Control Register (TO4POLVCR)	16, -, 8
:	(Reserved)				-
H'FFFF EA20	TOU40 Mode Control Register (TO40MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF EA24	TOU40 Counter (TO40CNT)				32
H'FFFF EA28	TOU40 Reload Register (TO40RLD)				32
:	(Reserved)				-
H'FFFF EA30	TOU41 Mode Control Register (TO41MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF EA34	TOU41 Counter (TO41CNT)				32
H'FFFF EA38	TOU41 Reload Register (TO41RLD)				32
:	(Reserved)				-
H'FFFF EA40	TOU42 Mode Control Register (TO42MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF EA44	TOU42 Counter (TO42CNT)				32
H'FFFF EA48	TOU42 Reload Register (TO42RLD)				32
:	(Reserved)				-
H'FFFF EA50	TOU43 Mode Control Register (TO43MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF EA54	TOU43 Counter (TO43CNT)				32
H'FFFF EA58	TOU43 Reload Register (TO43RLD)				32
:	(Reserved)				-
H'FFFF EA60	TOU44 Mode Control Register (TO44MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF EA64	TOU44 Counter (TO44CNT)				32
H'FFFF EA68	TOU44 Reload Register (TO44RLD)				32
:	(Reserved)				-
H'FFFF EA70	TOU45 Mode Control Register (TO45MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF EA74	TOU45 Counter (TO45CNT)				32
H'FFFF EA78	TOU45 Reload Register (TO45RLD)				32
:	(Reserved)				-
H'FFFF EA80	TOU46 Mode Control Register (TO46MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF EA84	TOU46 Counter (TO46CNT)				32
H'FFFF EA88	TOU46 Reload Register (TO46RLD)				32
:	(Reserved)				-
H'FFFF EA90	TOU47 Mode Control Register (TO47MCR)	(Reserved)	(Reserved)	(Reserved)	8, -, -, -
H'FFFF EA94	TOU47 Counter (TO47CNT)				32
H'FFFF EA98	TOU47 Reload Register (TO47RLD)				32
:	(Reserved)				-
H'FFFF F000	Interrupt Control Register 0 (ICR0)				32
:	(Reserved)				-
H'FFFF F010	Interrupt Priority Register (INTPRI)				32
:	(Reserved)				-
H'FFFF F01C	Interrupt Control Register 1 (ICR1)				32
:	(Reserved)				-
H'FFFF F024	Interrupt Source Register (INTREQ)				32
:	(Reserved)				-
H'FFFF F044	Interrupt Mask Register (INTMSK)				32
:	(Reserved)				-
H'FFFF F064	Interrupt Mask Clear Register (INTMSKCLR)				32
:	(Reserved)				-
H'FFFF F0C0	NMI Flag Control Register (NMIFCR)				32
:	(Reserved)				-
H'FFFF F300	User Interrupt Mask Level Register (USERIMASK)				32
:	(Reserved)				-
H'FFFF F400	Interrupt Priority Setting Register 0 (INT2PRI0)				32
H'FFFF F404	Interrupt Priority Setting Register 1 (INT2PRI1)				32
H'FFFF F408	Interrupt Priority Setting Register 2 (INT2PRI2)				32
H'FFFF F40C	Interrupt Priority Setting Register 3 (INT2PRI3)				32

Address	+0 Address	+1 Address	+2 Address	+3 Address	Access Size
	Bit 31 Bit 24	Bit 23 Bit 16	Bit 15 Bit 8	Bit 7 Bit 0	
H'FFFF F410	Interrupt Priority Setting Register 4 (INT2PRI4)				32
H'FFFF F414	Interrupt Priority Setting Register 5 (INT2PRI5)				32
H'FFFF F418	Interrupt Priority Setting Register 6 (INT2PRI6)				32
H'FFFF F41C	Interrupt Priority Setting Register 7 (INT2PRI7)				32
:	(Reserved)				-
H'FFFF F430	Interrupt Source Register 00 (Mask State is not affected) (INT2A00)				32
H'FFFF F434	Interrupt Source Register 10 (Mask State is affected) (INT2A10)				32
H'FFFF F438	Interrupt Mask Register 0 (INT2MSKR)				32
H'FFFF F43C	Interrupt Mask Clear Register 0 (INT2MSKCR)				32
H'FFFF F440	Per-Module Interrupt Source Register 0 (INT2B0)				32
H'FFFF F444	Per-Module Interrupt Source Register 1 (INT2B1)				32
H'FFFF F448	Per-Module Interrupt Source Register 2 (INT2B2)				32
H'FFFF F44C	Per-Module Interrupt Source Register 3 (INT2B3)				32
H'FFFF F450	Per-Module Interrupt Source Register 4 (INT2B4)				32
H'FFFF F454	Per-Module Interrupt Source Register 5 (INT2B5)				32
H'FFFF F458	Per-Module Interrupt Source Register 6 (INT2B6)				32
H'FFFF F45C	Per-Module Interrupt Source Register 7 (INT2B7)				32
H'FFFF F460	Per-Module Interrupt Source Register 8 (INT2B8)				32
:	(Reserved)				-
H'FFFF F468	Per-Module Interrupt Source Register 10 (INT2B10)				32
H'FFFF F46C	Per-Module Interrupt Source Register 11 (INT2B11)				32
:	(Reserved)				-
H'FFFF F494	Per-Module Interrupt Source Register 12 (INT2B12)				32
:	(Reserved)				-
H'FFFF F4A0	Interrupt Priority Setting Register 8 (INT2PRI8)				32
H'FFFF F4A4	Interrupt Priority Setting Register 9 (INT2PRI9)				32
H'FFFF F4A8	Interrupt Priority Setting Register 10 (INT2PRI10)				32
H'FFFF F4AC	Interrupt Priority Setting Register 11 (INT2PRI11)				32

Address	+0 Address		+1 Address		+2 Address		+3 Address		Access Size
	Bit 31	Bit 24	Bit 23	Bit 16	Bit 15	Bit 8	Bit 7	Bit 0	
H'FFFF F4B0	Interrupt Priority Setting Register 12 (INT2PRI12)								32
:	(Reserved)								-
H'FFFF F4C0	Interrupt Source Register 01(Mask State is not affected) (INT2A01)								32
H'FFFF F4C4	Interrupt Source Register 11(Mask State is affected) (INT2A11)								32
:	(Reserved)								-
H'FFFF F4D0	Interrupt Mask Register 1 (INT2MSKR1)								32
H'FFFF F4D4	Interrupt Mask Clear Register 1 (INT2MSKCR1)								32
: H'FFFF FFFF	(Reserved)								-

Appendix H Processing of Unused Pins

Table H.1 lists examples of the processing of unused pins.

Table H.1 Examples of Processing of Unused Pins

Pin Name		Processing
Pins with port input disable function (pins other than PG0 to PG3, PJ1, PJ3 to PJ5, PM0, PM2, PM4, PM6, PM8 to PM15, PN0, PN1, PN4, and PN5)		• Set the port input disabled state (port n input level setting bits). • Set to input mode and pull each pin low to V_{SS} or pull high to V_{CC} via a 1 to 10 kΩ resistor. • Set to output mode and leave the pin open.
Pins with no port input disable function*[1] PG0 to PG3, PJ1, PJ3 to PJ5		• Set to input mode and pull each pin low to V_{SS} or pull high to V_{CC} via a 1 to 10 kΩ resistor. • Set to output mode and leave the pin open.
NMI		Pull low to V_{SS} via a 0 to 10 kΩ resistor.
XTAL, WDTOVF#		Leave open.
PM0/AD0IN0, PM2/AD0IN2, PM4/AD0IN4, PM6/AD0IN6, PM8/AD0IN8 to PM15/AD0IN15	When selected as analog input pin	Leave open or pull each pin low to V_{SS} or pull high to V_{CC} via a 0 to 10 kΩ resistor.
	When selected as general port input	Pull each pin low to V_{SS} or pull high to V_{CC} via a 0 to 10 kΩ resistor. Set PM0, PM2, PM4, PM6, PM8 to PM15 as either all analog input pins or all general ports. Setting and using them as a mixture of the two is not supported.
PN0/AD1IN0, PN1/AD1IN1, PN4/AD1IN4, PN5/AD1IN5	When selected as analog input pin	Leave open or pull each pin low to V_{SS} or pull high to V_{CC} via a 0 to 10 kΩ resistor.
	When selected as general port input	Pull each pin low to V_{SS} or pull high to V_{CC} via a 0 to 10 kΩ resistor. Set PN0, PN1, PN4, and PN5 as either all analog input pins or all general ports. Setting and using them as a mixture of the two is not supported.
AV_{CC} , AVREFH		$AV_{CC} = V_{CC}$, AVREFH $\leq AV_{CC}$
AV_{SS} , AVREFL		Connect to V_{SS}.
H-UDI	TCK, TMS, TDI, MPMD	Pull each pin high to V_{CC} via a 0 to 100 kΩ resistor. (MPMD is pull high or leave open)
	ASEBRK#/BRKACK, TDO	Pull each pin high to V_{CC} via a 1 to 100 kΩ resistor. (TDO is pull high or leave open)
	TRST#	Pull each pin low to V_{SS} via a 0 to 100 kΩ resistor or connect to RESET#.

Note: *1 When unused pins with no port input disable function are processed by leaving them in the output-open state, there is a possibility of current flow in the period between power supply rise and when the pins are set to output because the pins are in the input-open state during that period.

- For processing unused pins, use wiring that extends as short a length as possible (within 20 mm) from the pins of the MCU.

This page is blank for reasons of layout.

SH7455 Group, SH7456 Group User's Manual: Hardware

Publication Date: Rev.1.10 Sep 22, 2011

Published by: Renesas Electronics Corporation

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

© 2011 Renesas Electronics Corporation. All rights reserved.

Colophon 1.1

SH7455 Group, SH7456 Group

Renesas Electronics Corporation

R01UH0030EJ0110