User's Manual

32

R32C/142 Group and R32C/145 Group

User's Manual: Hardware

RENESAS MCU

M16C Family / R32C/100 Series

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 occur due to the false recognition of the pin state as an input signal become possible. 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 part number, confirm that the change will not lead to problems.

   — The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

About This Manual

1. Purpose and Target User

This manual is designed to be read primarily by application developers who have an understanding of this microcomputer (MCU) including its hardware functions and electrical characteristics. The user should have a basic understanding of electric circuits, logic circuits and, MCUs.
This manual consists of 30 chapters covering six main categories: Overview, CPU, System Control, Peripherals, Electrical Characteristics, and Usage Notes.

Carefully read all notes in this document prior to use. Notes are found throughout each chapter, at the end of each chapter, and in the dedicated Usage Notes chapter.

The Revision History at the end of this manual summarizes primary modifications and additions to the previous versions. For details, please refer to the relative chapters or sections of this manual.

The R32C/142 Group and R32C/145 Group include the documents listed below. Verify this manual is the latest version by visiting the Renesas Electronics website.

Type of Document	Contents	Document Name	Document Number
Datasheet	Overview of Hardware and Electrical Characteristics	R32C/142 Group and R32C/145 Group Datasheet	R01DS0071EJ0110
User’s Manual: Hardware	Specifications and detailed descriptions of: -pin layout -memory map -peripherals -electrical characteristics -timing characteristics Refer to the Application Manual for peripheral usage.	R32C/142 Group and R32C/145 Group User’s Manual: Hardware	This publication
User’s Manual: Software/Software Manual	Descriptions of instruction set	R32C/100 Series Software Manual	REJ09B0267-0100
Application Note	-Usages -Applications -Sample programs -Programing technics using Assembly language or C programming language	Available on the Renesas Electronics website.	
Renesas Technical Update	Bulletins on product specifications, documents, etc.		

2. Numbers and Symbols

The following explains the denotations used in this manual for registers, bits, pins and various numbers.

(1) Registers, bits, and pins

Registers, bits, and pins are indicated by symbols. Each symbol has a register/bit/pin identifier after the symbol.

Example: PM03 bit in the PM0 register
P3_5 pin, VCC pin

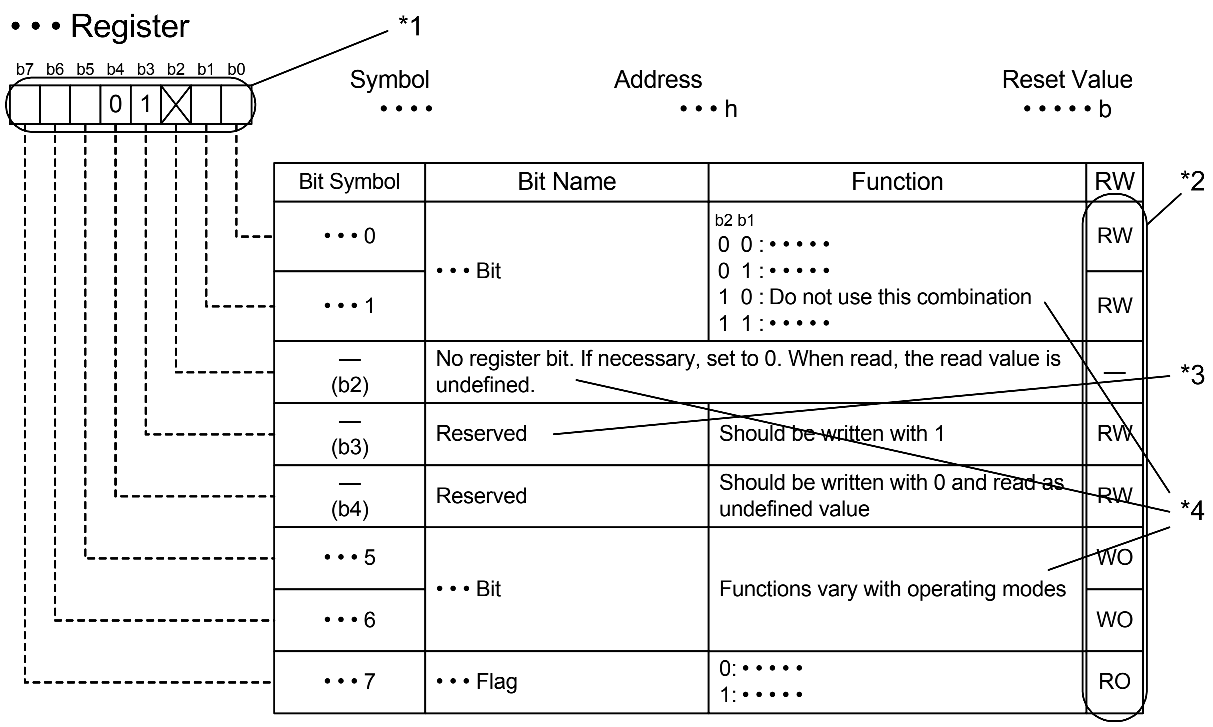
(2) Numbers

A binary number has the suffix “b” except for a 1-bit value.
A hexadecimal number has the suffix “h”.
A decimal number has no suffix.

Example: Binary notation: 11b
Hexadecimal notation: EFA0h
Decimal notation: 1234

3. Registers

The following illustration describes registers used throughout this manual.



Bit Symbol	Bit Name	Function	RW
• • • 0	• • • Bit	b2 b1 0 0 : • • • • • 0 1 : • • • • • 1 0 : Do not use this combination 1 1 : • • • • •	RW
• • • 1			RW
— (b2)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
— (b3)	Reserved	Should be written with 1	RW
— (b4)	Reserved	Should be written with 0 and read as undefined value	RW
• • • 5	• • • Bit	Functions vary with operating modes	WO
• • • 6			WO
• • • 7	• • • Flag	0: • • • • • 1: • • • • •	RO

*1

Blank box: Set this bit to 0 or 1 according to the function.
0: Set this bit to 0.
1: Set this bit to 1.
X: Nothing is assigned to this bit.

*2

RW: Read and write
RO: Read only
WO: Write only (the read value is undefined)
—: Not applicable

*3

- Reserved bit: This bit field is reserved. Set this bit to a specified value. For RW bits, the written value is read unless otherwise noted.

*4

- No register bit(s): No register bit(s) is/are assigned to this field. If necessary, set to 0 for possible future implementation.
- Do not use this combination: Proper operation is not guaranteed when this value is set.
- Functions vary with operating modes: Functions vary with peripheral operating modes. Refer to register illustrations of the respective mode.

4. Abbreviations and Acronyms

The following acronyms and terms are used throughout this manual.

Abbreviation/Acronym	Meaning
ACIA	Asynchronous Communication Interface Adapter
bps	bits per second
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
GSM	Global System for Mobile Communications
Hi-Z	High Impedance
IEBus	Inter Equipment Bus
I/O	Input/Output
IrDA	Infrared Data Association
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connection
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
SIM	Subscriber Identity Module
UART	Universal Asynchronous Receiver/Transmitter
VCO	Voltage Controlled Oscillator

TABLE OF CONTENTS

26. CAN Gateway Module 509

R01UH0218EJ0110
Rev.1.10
Sep 12, 2011

1. Overview

1.1 Features

The M16C Family offers a robust platform of 32-/16-bit CISC microcomputers (MCUs) featuring high ROM code efficiency, extensive EMI/EMS noise immunity, ultra-low power consumption, high-speed processing in actual applications, and numerous and varied integrated peripherals. Extensive device scalability from low- to high-end, featuring a single architecture as well as compatible pin assignments and peripheral functions, provides support for a vast range of application fields.

The R32C/100 Series is a high-end microcontroller series in the M16C Family. With a 4-Gbyte memory space, it achieves maximum code efficiency and high-speed processing with 32-bit CISC architecture, multiplier, multiply-accumulate unit, and floating point unit. The selection from the broadest choice of on-chip peripheral devices — UART, CRC, DMAC, A/D and D/A converters, timers, I^2C, and watchdog timer enables to minimize external components.

The R32C/100 Series, in particular, provides the R32C/142 Group and R32C/145 Group, products specific to gateway for in-vehicle network. These products, provided as 100-pin plastic molded LQFP package, have two channels of LIN module, three channels (R32C/142 Group) or six channels (R32C/145 Group) of CAN module, a CAN gateway module, and standard peripherals.

1.1.1 Applications

Automotive, etc.

1.1.2 Performance Overview

Table 1.1 and Table 1.2 show the performance overview of the R32C/142 Group and R32C/145 Group.

Table 1.1 Performance Overview (1/2)

Unit	Function	Explanation
CPU	Central processing unit	R32C/100 Series CPU Core • Basic instructions: 108 • Minimum instruction execution time: 15.625 ns (f(CPU) = 64 MHz) • Multiplier: 32-bit × 32-bit → 64-bit • Multiply-accumulate unit: 32-bit × 32-bit + 64-bit → 64-bit • IEEE-754 compatible FPU: Single precision • 32-bit barrel shifter • Operating mode: Single-chip mode
Memory		Flash memory: 256/512 Kbytes RAM: 32 Kbytes Data flash: 4 Kbytes × 2 blocks Refer to Tables 1.3 and 1.4 for details
Clock	Clock generator	• 4 circuits (main clock, sub clock, PLL, on-chip oscillator) • Oscillation stop detector: Main clock oscillator stop/restart detection • Frequency divide circuit: Divide-by-2 to divide-by-24 selectable • Low power modes: Wait mode, stop mode
Interrupts		Interrupt vectors: 261 External interrupt inputs: $\overline{\text{NMI}}$, $\overline{\text{INT}}$ × 6, key input × 4 Interrupt priority levels: 7
Watchdog Timer		15 bits × 1 (selectable input frequency from prescaler output) Automatic timer start function is available
DMA	DMAC	4 channels • Cycle-steal transfer mode • Request sources: 45 • 2 transfer modes: Single transfer, repeat transfer
	DMAC II	• Can be activated by any peripheral interrupt source • 3 transfer functions: Immediate data transfer, calculation transfer, chained transfer
I/O Ports	Programmable I/O ports	• 2 input-only ports • 84 CMOS I/O ports • A pull-up resistor is selectable for every 4 input ports
Timer	Timer A	16-bit timer × 5 Timer mode, event counter mode, one-shot timer mode, pulse-width modulation (PWM) mode Two-phase pulse signal processing in event counter mode (two-phase encoder input) × 3
	Timer B	16-bit timer × 6 Timer mode, event counter mode, pulse frequency measurement mode, pulse-width measurement mode
	Three-phase motor control timer	Three-phase motor control timer × 1 (timers A1, A2, A4, and B2 used) 8-bit programmable dead time timer

Table 1.2 Performance Overview (2/2)

Unit	Function	Explanation
Serial Interface	UART0 to UART4	Asynchronous/synchronous serial interface × 5 channels • I^2C-bus (UART0 to UART2) • Special mode 2 (UART0 to UART2)
A/D Converter		10-bit resolution × 26 channels Sample and hold functionality integrated Self test/Open-circuit detection assist
D/A Converter		8-bit resolution × 2
CRC Calculator		CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$)
X-Y Converter		16 bits × 16 bits
Intelligent I/O		Time measurement (input capture): 16 bits × 16 Digital debounce circuit contained Waveform generation (output compare): 16 bits × 16 Phase shift waveform output mode contained
Serial Bus Interface		2 channels • Synchronous serial communication mode • 4-wire serial bus mode Programmable character length: 8 to 16 bits
LIN Module		2 channels
CAN Module		3 channels for the R32C/142 Group 6 channels for the R32C/145 Group CAN functionality compliant with ISO 11898-1 16 mailboxes
Gateway Module		Up to 3 CAN channel routing control available for the R32C/142 Group Up to 6 CAN channel routing control available for the R32C/145 Group Routing table: up to 384 entries
Flash Memory		Programming and erasure supply voltage: VCC = 4.2 to 5.5 V, VCC0 = 3.0 V to VCC Minimum endurance: 100 program/erase cycles Security protection: ROM code protect, ID code protect Debugging: On-chip debug, on-board flash programming
Operating Frequency/Supply Voltage		64 MHz/VCC = 4.2 to 5.5 V, VCC0 = 3.0 V to VCC
Operating Temperature		-40°C to 85°C (J version) -40°C to 105°C (L version) (1) -40°C to 125°C (K version) (1)
Current Consumption		46 mA (VCC = 5.0 V, VCC0 = 3.3 V, f(CPU) = 64 MHz) 8 µA (VCC = 5.0 V, VCC0 = 3.3 V, f(XCIN) = 32.768 kHz, wait mode)
Package		100-pin plastic molded LQFP (PLQP0100KB-A)

Note:

1. Contact a Renesas Electronics sales office to use the L or K version products.

1.2 Product Information

Table 1.3 and Table 1.4 list the product information of the R32C/142 Group and R32C/145 Group, and Figure 1.1 shows the details of the part number.

Table 1.3 R32C/142 Group Product List **As of September, 2011**

Part Number	Package Code (1)	ROM Capacity (2)	RAM Capacity	Remarks
R5F6442FJFB	PLQP0100KB-A	256 Kbytes + 8 Kbytes	32 Kbytes	J Version
R5F6442FLFB				L Version (3)
R5F6442FKFB				K Version (3)
R5F6442HJFB		512 Kbytes + 8 Kbytes		J Version
R5F6442HLFB				L Version (3)
R5F6442HKFB				K Version (3)

Notes:

1. The old package code is as follows:PLQP0100KB-A: 100P6Q-A
2. Data flash memory provides an additional 8 Kbytes of ROM.
3. Contact a Renesas Electronics sales office to use the L or K version products.

Table 1.4 R32C/145 Group Product List **As of September, 2011**

Part Number	Package Code (1)	ROM Capacity (2)	RAM Capacity	Remarks
R5F6445FJFB	PLQP0100KB-A	256 Kbytes + 8 Kbytes	32 Kbytes	J Version
R5F6445FLFB				L Version (3)
R5F6445FKFB				K Version (3)
R5F6445HJFB		512 Kbytes + 8 Kbytes		J Version
R5F6445HLFB				L Version (3)
R5F6445HKFB				K Version (3)

Notes:

1. The old package code is as follows:PLQP0100KB-A: 100P6Q-A
2. Data flash memory provides an additional 8 Kbytes of ROM.
3. Contact a Renesas Electronics sales office to use the L or K version products.

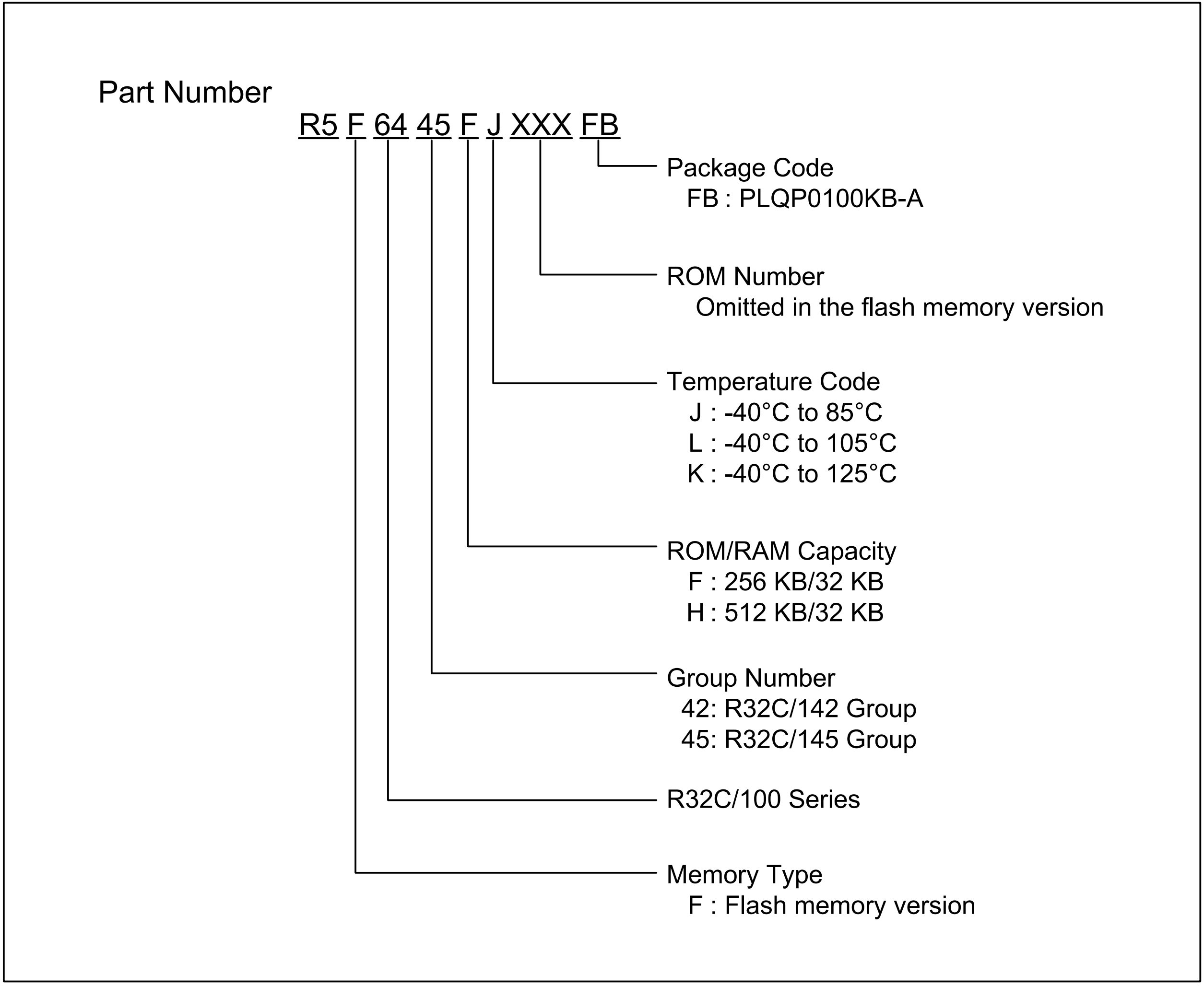


Figure 1.1 Part Numbering

1.3 Block Diagram

Figure 1.2 and Figure 1.3 show block diagrams of the R32C/142 Group and R32C/145 Group.

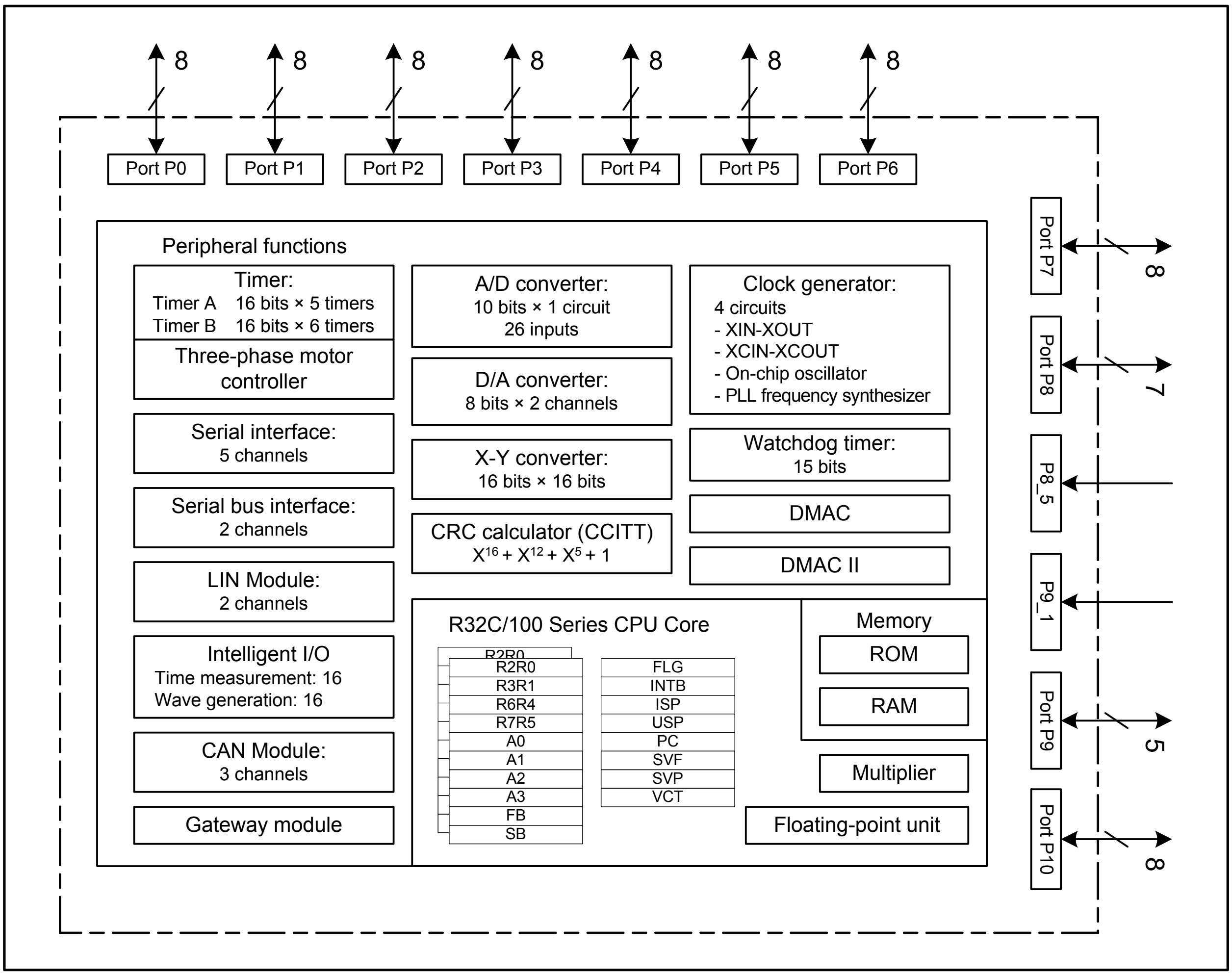


Figure 1.2 R32C/142 Group Block Diagram

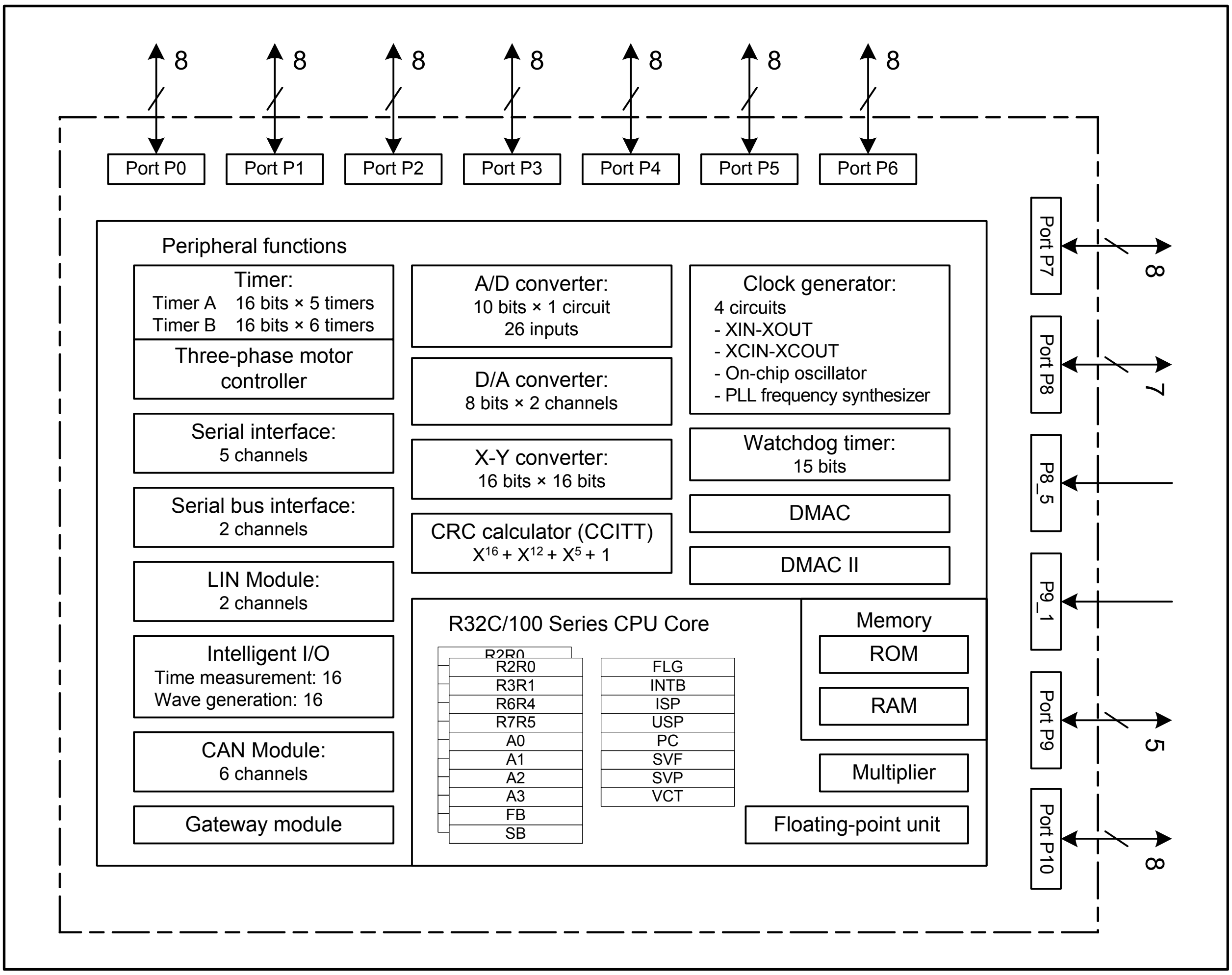
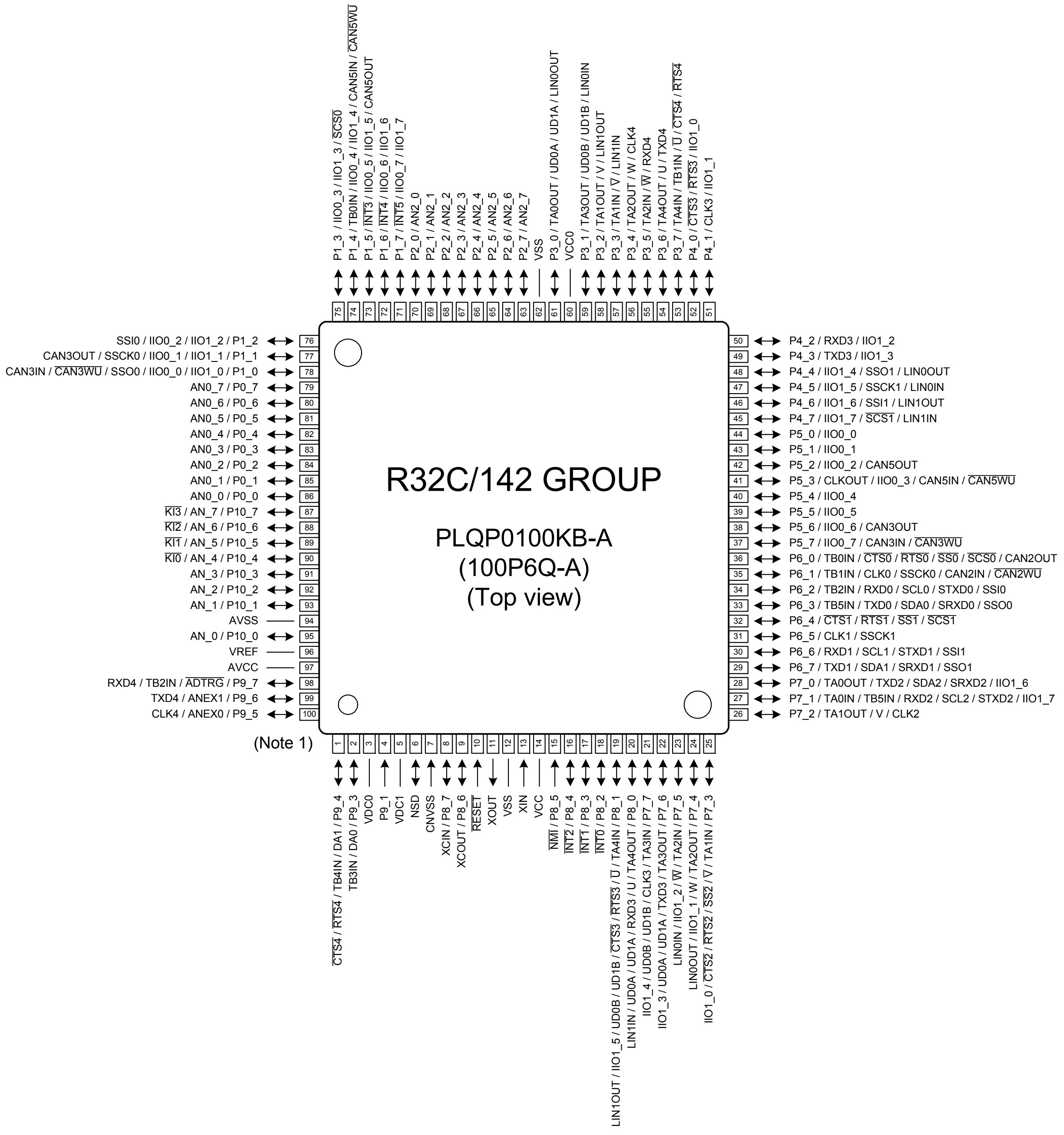


Figure 1.3 **R32C/145 Group Block Diagram**

1.4 Pin Assignment

Figure 1.4 and Figure 1.5 show the pin assignments (top view) and Table 1.5 to Table 1.10 show the pin characteristics of the R32C/142 Group and R32C/145 Group.



Note:

1. The position of pin number 1 varies by product. Refer to the index mark in attached “Package Dimensions”.

Figure 1.4 R32C/142 Group Pin Assignment (top view)

Table 1.5 Pin Characteristics for the R32C/142 Group (1/3)

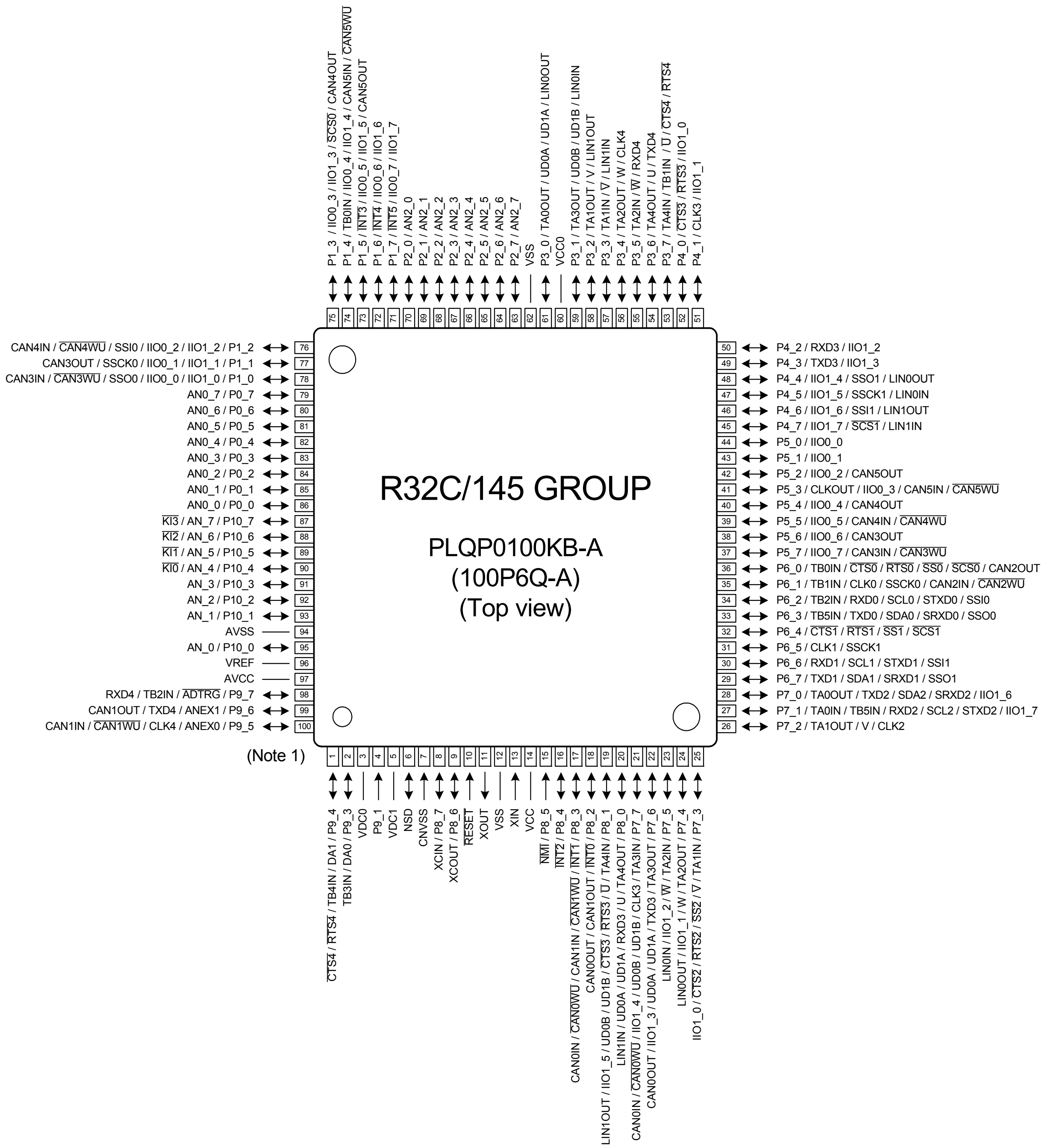
Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Intelligent I/O Pin	LIN / CAN Module Pin	Analog Pin
1		P9_4		TB4IN	$\overline{\text{CTS4}}$/$\overline{\text{RTS4}}$			DA1
2		P9_3		TB3IN				DA0
3	VDC0							
4		P9_1						
5	VDC1							
6	NSD							
7	CNVSS							
8	XCIN	P8_7						
9	XCOUT	P8_6						
10	$\overline{\text{RESET}}$							
11	XOUT							
12	VSS							
13	XIN							
14	VCC							
15		P8_5	$\overline{\text{NMI}}$					
16		P8_4	$\overline{\text{INT2}}$					
17		P8_3	$\overline{\text{INT1}}$					
18		P8_2	$\overline{\text{INT0}}$					
19		P8_1		TA4IN/$\overline{\text{U}}$	$\overline{\text{CTS3}}$/$\overline{\text{RTS3}}$	IIO1_5/UD0B/UD1B	LIN1OUT	
20		P8_0		TA4OUT/U	RXD3	UD0A/UD1A	LIN1IN	
21		P7_7		TA3IN	CLK3	IIO1_4/UD0B/UD1B		
22		P7_6		TA3OUT	TXD3	IIO1_3/UD0A/UD1A		
23		P7_5		TA2IN/$\overline{\text{W}}$		IIO1_2	LIN0IN	
24		P7_4		TA2OUT/W		IIO1_1	LIN0OUT	
25		P7_3		TA1IN/$\overline{\text{V}}$	$\overline{\text{CTS2}}$/$\overline{\text{RTS2}}$/$\overline{\text{SS2}}$	IIO1_0		
26		P7_2		TA1OUT/V	CLK2			
27		P7_1		TA0IN/TB5IN	RXD2/SCL2/STXD2	IIO1_7		
28		P7_0		TA0OUT	TXD2/SDA2/SRXD2	IIO1_6		
29		P6_7			TXD1/SDA1/SRXD1/SSO1			
30		P6_6			RXD1/SCL1/STXD1/SSI1			
31		P6_5			CLK1/SSCK1			
32		P6_4			$\overline{\text{CTS1}}$/$\overline{\text{RTS1}}$/$\overline{\text{SS1}}$/$\overline{\text{SCS1}}$			
33		P6_3		TB5IN	TXD0/SDA0/SRXD0/SSO0			

Table 1.6 Pin Characteristics for the R32C/142 Group (2/3)

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Intelligent I/O Pin	LIN / CAN Module Pin	Analog Pin
34		P6_2		TB2IN	RXD0/SCL0/ STXD0/SSI0			
35		P6_1		TB1IN	CLK0/SSCK0		CAN2IN/$\overline{\mathrm{CAN2WU}}$	
36		P6_0		TB0IN	$\overline{\mathrm{CTS0}}$/$\overline{\mathrm{RTS0}}$/$\overline{\mathrm{SS0}}$/ $\overline{\mathrm{SCS0}}$		CAN2OUT	
37		P5_7				IIO0_7	CAN3IN/$\overline{\mathrm{CAN3WU}}$	
38		P5_6				IIO0_6	CAN3OUT	
39		P5_5				IIO0_5		
40		P5_4				IIO0_4		
41	CLK-OUT	P5_3				IIO0_3	CAN5IN/$\overline{\mathrm{CAN5WU}}$	
42		P5_2				IIO0_2	CAN5OUT	
43		P5_1				IIO0_1		
44		P5_0				IIO0_0		
45		P4_7			$\overline{\mathrm{SCS1}}$	IIO1_7	LIN1IN	
46		P4_6			SSI1	IIO1_6	LIN1OUT	
47		P4_5			SSCK1	IIO1_5	LIN0IN	
48		P4_4			SSO1	IIO1_4	LIN0OUT	
49		P4_3			TXD3	IIO1_3		
50		P4_2			RXD3	IIO1_2		
51		P4_1			CLK3	IIO1_1		
52		P4_0			$\overline{\mathrm{CTS3}}$/$\overline{\mathrm{RTS3}}$	IIO1_0		
53		P3_7		TA4IN/ TB1IN/$\overline{\mathrm{U}}$	$\overline{\mathrm{CTS4}}$/$\overline{\mathrm{RTS4}}$			
54		P3_6		TA4OUT/U	TXD4			
55		P3_5		TA2IN/$\overline{\mathrm{W}}$	RXD4			
56		P3_4		TA2OUT/W	CLK4			
57		P3_3		TA1IN/$\overline{\mathrm{V}}$			LIN1IN	
58		P3_2		TA1OUT/V			LIN1OUT	
59		P3_1		TA3OUT		UD0B/UD1B	LIN0IN	
60	VCC0							
61		P3_0		TA0OUT		UD0A/UD1A	LIN0OUT	
62	VSS							
63		P2_7						AN2_7
64		P2_6						AN2_6
65		P2_5						AN2_5
66		P2_4						AN2_4
67		P2_3						AN2_3
68		P2_2						AN2_2
69		P2_1						AN2_1
70		P2_0						AN2_0
71		P1_7	$\overline{\mathrm{INT5}}$			IIO0_7/IIO1_7		
72		P1_6	$\overline{\mathrm{INT4}}$			IIO0_6/IIO1_6		

Table 1.7 Pin Characteristics for the R32C/142 Group (3/3)

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Intelligent I/O Pin	LIN / CAN Module Pin	Analog Pin
73		P1_5	$\overline{\text{INT3}}$			IIO0_5/IIO1_5	CAN5OUT	
74		P1_4		TB0IN		IIO0_4/IIO1_4	CAN5IN/$\overline{\text{CAN5WU}}$	
75		P1_3			$\overline{\text{SCS0}}$	IIO0_3/IIO1_3		
76		P1_2			SSI0	IIO0_2/IIO1_2		
77		P1_1			SSCK0	IIO0_1/IIO1_1	CAN3OUT	
78		P1_0			SSO0	IIO0_0/IIO1_0	CAN3IN/$\overline{\text{CAN3WU}}$	
79		P0_7						AN0_7
80		P0_6						AN0_6
81		P0_5						AN0_5
82		P0_4						AN0_4
83		P0_3						AN0_3
84		P0_2						AN0_2
85		P0_1						AN0_1
86		P0_0						AN0_0
87		P10_7	$\overline{\text{KI3}}$					AN_7
88		P10_6	$\overline{\text{KI2}}$					AN_6
89		P10_5	$\overline{\text{KI1}}$					AN_5
90		P10_4	$\overline{\text{KI0}}$					AN_4
91		P10_3						AN_3
92		P10_2						AN_2
93		P10_1						AN_1
94	AVSS							
95		P10_0						AN_0
96	VREF							
97	AVCC							
98		P9_7		TB2IN	RXD4			$\overline{\text{ADTRG}}$
99		P9_6			TXD4			ANEX1
100		P9_5			CLK4			ANEX0



Note:
1. The position of pin number 1 varies by product. Refer to the index mark in attached “Package Dimensions”.

Figure 1.5 R32C/145 Group Pin Assignment (top view)

Table 1.8 Pin Characteristics for the R32C/145 Group (1/3)

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Intelligent I/O Pin	LIN / CAN Module Pin	Analog Pin
1		P9_4		TB4IN	CTS4/RTS4			DA1
2		P9_3		TB3IN				DA0
3	VDC0							
4		P9_1						
5	VDC1							
6	NSD							
7	CNVSS							
8	XCIN	P8_7						
9	XCOUT	P8_6						
10	RESET							
11	XOUT							
12	VSS							
13	XIN							
14	VCC							
15		P8_5	NMI					
16		P8_4	INT2					
17		P8_3	INT1				CAN0IN/CAN0WU/ CAN1IN/CAN1WU	
18		P8_2	INT0				CAN0OUT/ CAN1OUT	
19		P8_1		TA4IN/U	CTS3/RTS3	IIO1_5/UD0B/UD1B	LIN1OUT	
20		P8_0		TA4OUT/U	RXD3	UD0A/UD1A	LIN1IN	
21		P7_7		TA3IN	CLK3	IIO1_4/UD0B/UD1B	CAN0IN/CAN0WU	
22		P7_6		TA3OUT	TXD3	IIO1_3/UD0A/UD1A	CAN0OUT	
23		P7_5		TA2IN/W		IIO1_2	LIN0IN	
24		P7_4		TA2OUT/W		IIO1_1	LIN0OUT	
25		P7_3		TA1IN/V	CTS2/RTS2/SS2	IIO1_0		
26		P7_2		TA1OUT/V	CLK2			
27		P7_1		TA0IN/ TB5IN	RXD2/SCL2/ STXD2	IIO1_7		
28		P7_0		TA0OUT	TXD2/SDA2/ SRXD2	IIO1_6		
29		P6_7			TXD1/SDA1/ SRXD1/SSO1			
30		P6_6			RXD1/SCL1/ STXD1/SSI1			
31		P6_5			CLK1/SSCK1			
32		P6_4			CTS1/RTS1/SS1/ SCS1			
33		P6_3		TB5IN	TXD0/SDA0/ SRXD0/SSO0			

Table 1.9 Pin Characteristics for the R32C/145 Group (2/3)

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Intelligent I/O Pin	LIN / CAN Module Pin	Analog Pin
34		P6_2		TB2IN	RXD0/SCL0/ STXD0/SSI0			
35		P6_1		TB1IN	CLK0/SSCK0		CAN2IN/CAN2WU	
36		P6_0		TB0IN	CTS0/RTS0/SS0/ SCS0		CAN2OUT	
37		P5_7				IIO0_7	CAN3IN/CAN3WU	
38		P5_6				IIO0_6	CAN3OUT	
39		P5_5				IIO0_5	CAN4IN/CAN4WU	
40		P5_4				IIO0_4	CAN4OUT	
41	CLK-OUT	P5_3				IIO0_3	CAN5IN/CAN5WU	
42		P5_2				IIO0_2	CAN5OUT	
43		P5_1				IIO0_1		
44		P5_0				IIO0_0		
45		P4_7			SCS1	IIO1_7	LIN1IN	
46		P4_6			SSI1	IIO1_6	LIN1OUT	
47		P4_5			SSCK1	IIO1_5	LIN0IN	
48		P4_4			SSO1	IIO1_4	LIN0OUT	
49		P4_3			TXD3	IIO1_3		
50		P4_2			RXD3	IIO1_2		
51		P4_1			CLK3	IIO1_1		
52		P4_0			CTS3/RTS3	IIO1_0		
53		P3_7		TA4IN/ TB1IN/U	CTS4/RTS4			
54		P3_6		TA4OUT/U	TXD4			
55		P3_5		TA2IN/W	RXD4			
56		P3_4		TA2OUT/W	CLK4			
57		P3_3		TA1IN/V			LIN1IN	
58		P3_2		TA1OUT/V			LIN1OUT	
59		P3_1		TA3OUT		UD0B/UD1B	LIN0IN	
60	VCC0							
61		P3_0		TA0OUT		UD0A/UD1A	LIN0OUT	
62	VSS							
63		P2_7						AN2_7
64		P2_6						AN2_6
65		P2_5						AN2_5
66		P2_4						AN2_4
67		P2_3						AN2_3
68		P2_2						AN2_2
69		P2_1						AN2_1
70		P2_0						AN2_0
71		P1_7	INT5			IIO0_7/IIO1_7		
72		P1_6	INT4			IIO0_6/IIO1_6		

Table 1.10 Pin Characteristics for the R32C/145 Group (3/3)

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Intelligent I/O Pin	LIN / CAN Module Pin	Analog Pin
73		P1_5	$\overline{\mathrm{INT3}}$			IIO0_5/IIO1_5	CAN5OUT	
74		P1_4		TB0IN		IIO0_4/IIO1_4	CAN5IN/$\overline{\mathrm{CAN5WU}}$	
75		P1_3			$\overline{\mathrm{SCS0}}$	IIO0_3/IIO1_3	CAN4OUT	
76		P1_2			SSI0	IIO0_2/IIO1_2	CAN4IN/$\overline{\mathrm{CAN4WU}}$	
77		P1_1			SSCK0	IIO0_1/IIO1_1	CAN3OUT	
78		P1_0			SSO0	IIO0_0/IIO1_0	CAN3IN/$\overline{\mathrm{CAN3WU}}$	
79		P0_7						AN0_7
80		P0_6						AN0_6
81		P0_5						AN0_5
82		P0_4						AN0_4
83		P0_3						AN0_3
84		P0_2						AN0_2
85		P0_1						AN0_1
86		P0_0						AN0_0
87		P10_7	$\overline{\mathrm{KI3}}$					AN_7
88		P10_6	$\overline{\mathrm{KI2}}$					AN_6
89		P10_5	$\overline{\mathrm{KI1}}$					AN_5
90		P10_4	$\overline{\mathrm{KI0}}$					AN_4
91		P10_3						AN_3
92		P10_2						AN_2
93		P10_1						AN_1
94	AVSS							
95		P10_0						AN_0
96	VREF							
97	AVCC							
98		P9_7		TB2IN	RXD4			$\overline{\mathrm{ADTRG}}$
99		P9_6			TXD4		CAN1OUT	ANEX1
100		P9_5			CLK4		CAN1IN/$\overline{\mathrm{CAN1WU}}$	ANEX0

1.5 Pin Definitions and Functions

Table 1.11 to Table 1.13 show the pin definitions and functions.

Table 1.11 Pin Definitions and Functions (1/3)

Function	Symbol	I/O	Description
Power supply	VCC, VCC0, VSS	I	Applicable as follows: VCC = 4.2 to 5.5 V, VCC0 = 3.0 to 5.5 V, VSS = 0 V. It must be VCC0 ≤ VCC
Analog power supply	AVCC, AVSS	I	Power supply for the A/D converter. AVCC and AVSS should be connected to VCC and VSS, respectively
Connecting pins for decoupling capacitor	VDC0, VDC1	—	A decoupling capacitor for internal voltage should be connected between VDC0 and VDC1
Reset input	$\overline{\text{RESET}}$	I	The MCU is reset when this pin is driven low
CNVSS	CNVSS	I	This pin should be connected to VSS via a resistor
Debug port	NSD	I/O	This pin is to communicate with a debugger. It should be connected to VCC via a resistor of 1 to 4.7 kΩ
Main clock input	XIN	I	Input/output for the main clock oscillator. A crystal, or a ceramic resonator should be connected between pins XIN and XOUT. An external clock should be input at the XIN while leaving the XOUT open
Main clock output	XOUT	O	
Sub clock input	XCIN	I	Input/output for the sub clock oscillator. A crystal oscillator should be connected between pins XCIN and XCOUT. An external clock should be input at the XCIN while leaving the XCOUT open
Sub clock output	XCOUT	O	
Clock output	CLKOUT	O	Output of the clock with the same frequency as low speed clocks, f8, or f32
External interrupt input	$\overline{\text{INT0}}$ to $\overline{\text{INT5}}$	I	Input for external interrupts
NMI input	P8_5/$\overline{\text{NMI}}$	I	Input for NMI
Key input interrupt	$\overline{\text{KI0}}$ to $\overline{\text{KI3}}$	I	Input for the key input interrupt
I/O ports	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7	I/O	I/O ports in CMOS. Each port can be programmed to input or output under the control of the direction register. Pull-up resistors are selected for following 4-pin units, but are enabled only for the input pins: Pi_0 to Pi_3 and Pi_4 to Pi_7 (i = 0 to 10)
Input port	P9_1	I	Input port in CMOS. Pull-up resistors are selectable for P9_1 and P9_3
Timer A	TA0OUT to TA4OUT	I/O	Timers A0 to A4 input/output
	TA0IN to TA4IN	I	Timers A0 to A4 input
Timer B	TB0IN to TB5IN	I	Timers B0 to B5 input

Table 1.12 Pin Definitions and Functions (2/3)

Function	Symbol	I/O	Description
Three-phase motor control timer output	U, $\overline{\text{U}}$, V, $\overline{\text{V}}$, W, $\overline{\text{W}}$	O	Three-phase motor control timer output
Serial interface	$\overline{\text{CTS0}}$ to $\overline{\text{CTS4}}$	I	Handshake input
	$\overline{\text{RTS0}}$ to $\overline{\text{RTS4}}$	O	Handshake output
	CLK0 to CLK4	I/O	Transmit/receive clock input/output
	RXD0 to RXD4	I	Serial data input
	TXD0 to TXD4	O	Serial data output
I^{2}C-bus (simplified)	SDA0 to SDA2	I/O	Serial data input/output
	SCL0 to SCL2	I/O	Transmit/receive clock input/output
Serial interface special functions	STXD0 to STXD2	O	Serial data output in slave mode
	SRXD0 to SRXD2	I	Serial data input in slave mode
	$\overline{\text{SS0}}$ to $\overline{\text{SS2}}$	I	Input to control serial interface special functions
A/D converter	AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7	I	Analog input for the A/D converter
	$\overline{\text{ADTRG}}$	I	External trigger input for the A/D converter
	ANEX0	I/O	Expanded analog input for the A/D converter and output in external op-amp connection mode
	ANEX1	I	Expanded analog input for the A/D converter
D/A converter	DA0, DA1	O	Output for the D/A converter
Reference voltage input	VREF	I	Reference voltage input for the A/D converter and D/A converter
Intelligent I/O	IIO0_0 to IIO0_7	I/O	Input/output for the Intelligent I/O group 0. Either input capture or output compare is selectable
	IIO1_0 to IIO1_7	I/O	Input/output for the Intelligent I/O group 1. Either input capture or output compare is selectable
	UD0A, UD0B, UD1A, UD1B	I	Input for the two-phase encoder
Serial bus interface	SSO0, SSO1	I/O	Serial data output. Functions as serial data input/output in 4-wire serial bus mode
	SSI0, SSI1	I/O	Serial data input. Functions as serial data input/output in 4-wire serial bus mode
	SSCK0, SSCK1	I/O	Transmit/receive clock input/output
	$\overline{\text{SCS0}}$, $\overline{\text{SCS1}}$	I/O	Input/output to control the synchronous serial interface
LIN module	LIN0OUT, LIN1OUT	O	Transmit data output for the LIN communications
	LIN0IN, LIN1IN	I	Receive data input for the LIN communications

Table 1.13 Pin Definitions and Functions (3/3)

Function	Symbol	I/O	Description
CAN module for the R32C/142 Group	CAN2IN, CAN3IN, CAN5IN	I	Receive data input for the CAN communications
	CAN2OUT, CAN3OUT, CAN5OUT	O	Transmit data output for the CAN communications
	$\overline{\text{CAN2WU}}$, $\overline{\text{CAN3WU}}$, $\overline{\text{CAN5WU}}$	I	Input for the CAN wake-up interrupt
CAN module for the R32C/145 Group	CAN0IN to CAN5IN	I	Receive data input for the CAN communications
	CAN0OUT to CAN5OUT	O	Transmit data output for the CAN communications
	$\overline{\text{CAN0WU}}$ to $\overline{\text{CAN5WU}}$	I	Input for the CAN wake-up interrupt

2. Central Processing Unit (CPU)

The CPU contains the registers shown below. There are two register banks each consisting of registers R2R0, R3R1, R6R4, R7R5, A0 to A3, SB, and FB.

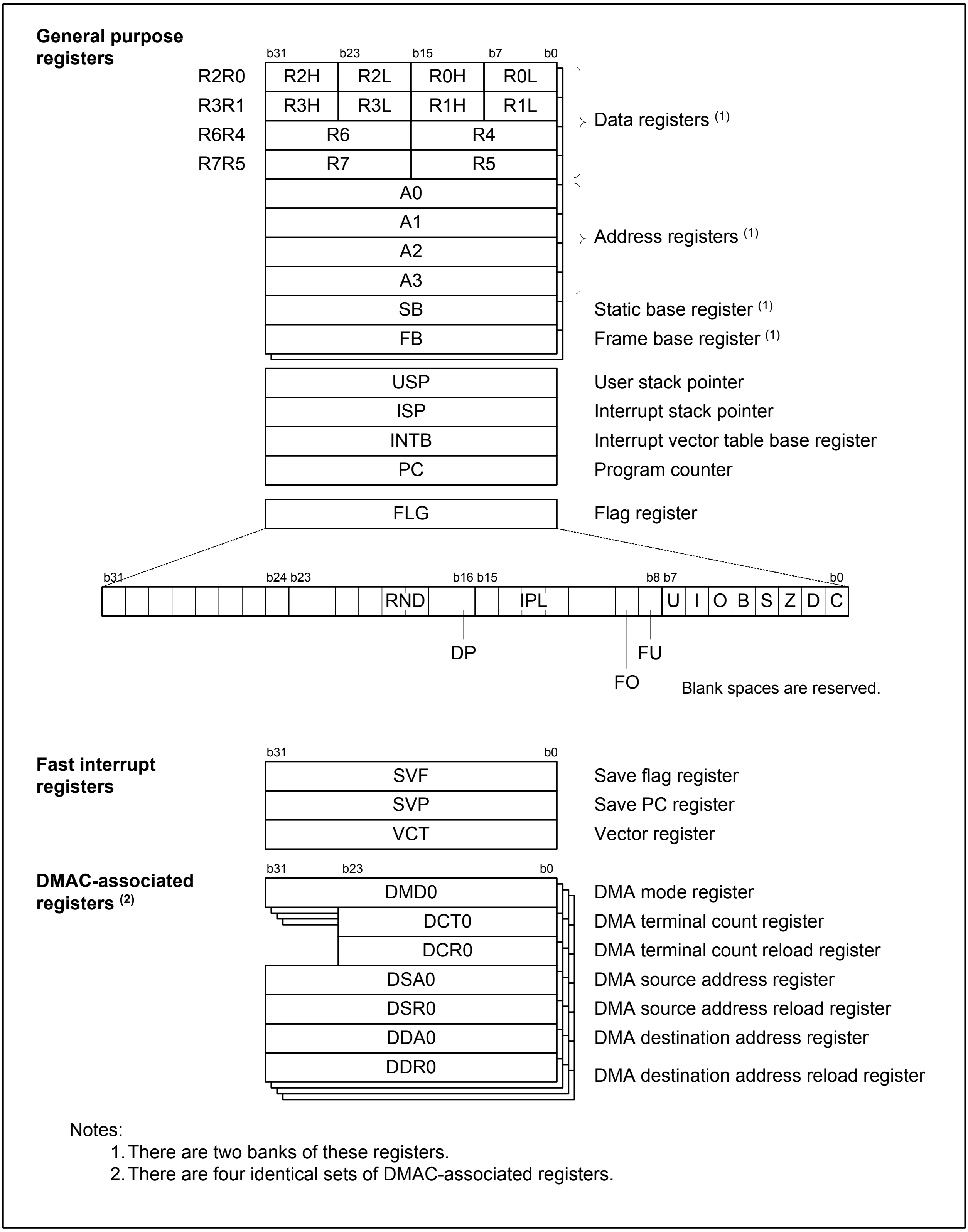


Figure 2.1 CPU Registers

2.1 General Purpose Registers

2.1.1 Data Registers (R2R0, R3R1, R6R4, and R7R5)

These 32-bit registers are primarily used for transfers and arithmetic/logic operations.
Each of the registers can be divided into upper and lower 16-bit registers, e.g. R2R0 can be divided into R2 and R0, R3R1 can be divided into R3 and R1, etc.
Moreover, data registers R2R0 and R3R1 can be divided into four 8-bit data registers: upper (R2H and R3H), mid-upper (R2L and R3L), mid-lower (R0H and R1H), and lower (R0L and R1L).

2.1.2 Address Registers (A0, A1, A2, and A3)

These 32-bit registers have functions similar to data registers. They are also used for address register indirect addressing and address register relative addressing.

2.1.3 Static Base Register (SB)

This 32-bit register is used for SB relative addressing.

2.1.4 Frame Base Register (FB)

This 32-bit register is used for FB relative addressing.

2.1.5 Program Counter (PC)

This 32-bit counter indicates the address of the instruction to be executed next.

2.1.6 Interrupt Vector Table Base Register (INTB)

This 32-bit register indicates the start address of a relocatable vector table.

2.1.7 User Stack Pointer (USP) and Interrupt Stack Pointer (ISP)

Two types of 32-bit stack pointers (SPs) are provided: user stack pointer (USP) and interrupt stack pointer (ISP).
Use the stack pointer select flag (U flag) to select either the user stack pointer (USP) or the interrupt stack pointer (ISP). The U flag is bit 7 in the flag register (FLG). Refer to 2.1.8 “Flag Register (FLG)” for details.
To minimize the overhead of interrupt sequence due to less memory access, set the user stack pointer (USP) or the interrupt stack pointer (ISP) to a multiple of 4.

2.1.8 Flag Register (FLG)

This 32-bit register indicates the CPU status.

2.1.8.1 Carry Flag (C flag)

This flag retains a carry, borrow, or shifted-out bit generated by the arithmetic logic unit (ALU).

2.1.8.2 Debug Flag (D flag)

This flag is only for debugging. Only set this bit to 0.

2.1.8.3 Zero Flag (Z flag)

This flag becomes 1 when the result of an operation is 0; otherwise it is 0.

2.1.8.4 Sign Flag (S flag)

This flag becomes 1 when the result of an operation is a negative value; otherwise it is 0.

2.1.8.5 Register Bank Select Flag (B flag)

This flag selects a register bank. It indicates 0 when register bank 0 is selected, and 1 when register bank 1 is selected.

2.1.8.6 Overflow Flag (O flag)

This flag becomes 1 when the result of an operation overflows; otherwise it is 0.

2.1.8.7 Interrupt Enable Flag (I flag)

This flag enables maskable interrupts. To disable maskable interrupts, set this flag to 0. To enable them, set this flag to 1. When an interrupt is accepted, the flag becomes 0.

2.1.8.8 Stack Pointer Select Flag (U flag)

To select the interrupt stack pointer (ISP), set this flag to 0. To select the user stack pointer (USP), set this flag to 1.

It becomes 0 when a hardware interrupt is accepted or when an INT instruction designated by a software interrupt number from 0 to 127 is executed.

2.1.8.9 Floating-point Underflow Flag (FU flag)

This flag becomes 1 when an underflow occurs in a floating-point operation; otherwise it is 0. It also becomes 1 when the operand contains invalid numbers (subnormal numbers).

2.1.8.10 Floating-point Overflow Flag (FO flag)

This flag becomes 1 when an overflow occurs in a floating-point operation; otherwise it is 0. It also becomes 1 when the operand contains invalid numbers (subnormal numbers).

2.1.8.11 Processor Interrupt Priority Level (IPL)

The processor interrupt priority level (IPL), consisting of 3 bits, selects a processor interrupt priority level from level 0 to 7. An interrupt is enabled when the interrupt request level is higher than the selected IPL.

When the processor interrupt priority level (IPL) is set to 111b (level 7), all interrupts are disabled.

2.1.8.12 Fixed-point Radix Point Designation Bit (DP bit)

This bit designates the radix point. It also specifies which portion of the fixed-point multiplication result to extract. It is used for the MULX instruction.

2.1.8.13 Floating-point Rounding Mode (RND)

The 2-bit floating-point rounding mode selects a rounding mode for floating-point calculation results.

2.1.8.14 Reserved

Only set this bit to 0. The read value is undefined.

2.2 Fast Interrupt Registers

The following three registers are provided to minimize the overhead of the interrupt sequence. Refer to 10.4 “Fast Interrupt” for details.

2.2.1 Save Flag Register (SVF)

This 32-bit register is used to save the flag register when a fast interrupt is generated.

2.2.2 Save PC Register (SVP)

This 32-bit register is used to save the program counter when a fast interrupt is generated.

2.2.3 Vector Register (VCT)

This 32-bit register is used to indicate a jump address when a fast interrupt is generated.

2.3 DMAC-associated Registers

There are seven types of DMAC-associated registers. Refer to 12. “DMAC” for details.

2.3.1 DMA Mode Registers (DMD0, DMD1, DMD2, and DMD3)

These 32-bit registers are used to set DMA transfer mode, bit rate, etc.

2.3.2 DMA Terminal Count Registers (DCT0, DCT1, DCT2, and DCT3)

These 24-bit registers are used to set the number of DMA transfers.

2.3.3 DMA Terminal Count Reload Registers (DCR0, DCR1, DCR2, and DCR3)

These 24-bit registers are used to set the reloaded values for DMA terminal count registers.

2.3.4 DMA Source Address Registers (DSA0, DSA1, DSA2, and DSA3)

These 32-bit registers are used to set DMA source addresses.

2.3.5 DMA Source Address Reload Registers (DSR0, DSR1, DSR2, and DSR3)

These 32-bit registers are used to set the reloaded values for DMA source address registers.

2.3.6 DMA Destination Address Registers (DDA0, DDA1, DDA2, and DDA3)

These 32-bit registers are used to set DMA destination addresses.

2.3.7 DMA Destination Address Reload Registers (DDR0, DDR1, DDR2, and DDR3)

These 32-bit registers are used to set reloaded values for DMA destination address registers.

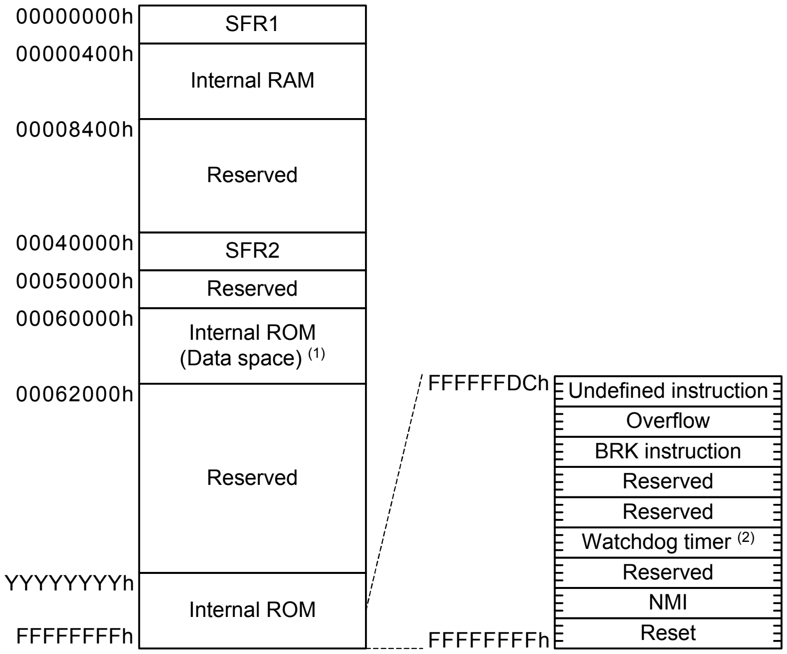
3. Memory

Figure 3.1 shows the memory map of the R32C/142 Group and R32C/145 Group.
The R32C/142 Group and R32C/145 Group provide a 4-Gbyte address space from 00000000h to FFFFFFFFh.

The internal ROM is mapped from address FFFFFFFFh in the inferior direction. For example, the 512-Kbyte internal ROM is mapped from FFF80000h to FFFFFFFFh.
The fixed interrupt vector table contains the start address of interrupt handlers and is mapped from FFFFFFDCh to FFFFFFFFh.
The internal RAM is mapped from address 00000400h in the superior direction. For example, the 32-Kbyte internal RAM is mapped from 00000400h to 000083FFh. Besides being used for data storage, the internal RAM functions as a stack(s) for subroutine calls and/or interrupt handlers.

Special function registers (SFRs), which are control registers for peripheral functions, are mapped from 00000000h to 000003FFh, and from 00040000h to 0004FFFFh. Unoccupied SFR locations are reserved, and no access is allowed.

Internal ROM	
Capacity	YYYYYYYYh
256 Kbytes	FFFC0000h
512 Kbytes	FFF80000h



Notes:
1. Additional two 4-Kbyte spaces (blocks A and B) for storing data are provided in the flash memory version.
2. The watchdog timer interrupt shares the vector table with the oscillator stop detection interrupt.

Figure 3.1 Memory Map

4. Special Function Registers (SFRs)

SFRs are memory-mapped peripheral registers that control the operation of peripherals. There are no SFRs associated with channels CAN0, CAN1, and CAN4 in the R32C/142 Group.
Table 4.1 SFR List (1) to Table 4.75 SFR List (75) list the SFR details.

Table 4.1 SFR List (1)

Address	Register	Symbol	Reset Value
000000h			
000001h			
000002h			
000003h			
000004h	Clock Control Register	CCR	0001 1000b
000005h			
000006h	Flash Memory Control Register	FMCR	0000 0001b
000007h	Protect Release Register	PRR	00h
000008h			
000009h			
00000Ah			
00000Bh			
00000Ch			
00000Dh			
00000Eh			
00000Fh			
000010h			
000011h			
000012h			
000013h			
000014h			
000015h			
000016h			
000017h			
000018h			
000019h			
00001Ah			
00001Bh			
00001Ch	Flash Memory Rewrite Bus Control Register	FEBC	0000h
00001Dh			
00001Eh	Peripheral Bus Control Register	PBC	0504h
00001Fh			
000020h to 00005Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.2 SFR List (2)

Address	Register	Symbol	Reset Value
000060h			
000061h	Timer B5 Interrupt Control Register	TB5IC	XXXX X000b
000062h			
000063h	UART2 Receive/ACK Interrupt Control Register	S2RIC	XXXX X000b
000064h			
000065h			
000066h			
000067h			
000068h	DMA0 Transfer Complete Interrupt Control Register	DM0IC	XXXX X000b
000069h	UART0 Start Condition/Stop Condition Detection Interrupt Control Register	BCN0IC	XXXX X000b
00006Ah	DMA2 Transfer Complete Interrupt Control Register	DM2IC	XXXX X000b
00006Bh	A/D Converter 0 Convert Completion Interrupt Control Register	AD0IC	XXXX X000b
00006Ch	Timer A0 Interrupt Control Register	TA0IC	XXXX X000b
00006Dh	Intelligent I/O Interrupt Control Register 0	IIO0IC	XXXX X000b
00006Eh	Timer A2 Interrupt Control Register	TA2IC	XXXX X000b
00006Fh	Intelligent I/O Interrupt Control Register 2	IIO2IC	XXXX X000b
000070h	Timer A4 Interrupt Control Register	TA4IC	XXXX X000b
000071h	Intelligent I/O Interrupt Control Register 4	IIO4IC	XXXX X000b
000072h	UART0 Receive/ACK Interrupt Control Register	S0RIC	XXXX X000b
000073h	Intelligent I/O Interrupt Control Register 6	IIO6IC	XXXX X000b
000074h	UART1 Receive/ACK Interrupt Control Register	S1RIC	XXXX X000b
000075h	Intelligent I/O Interrupt Control Register 8	IIO8IC	XXXX X000b
000076h	Timer B1 Interrupt Control Register	TB1IC	XXXX X000b
000077h	Intelligent I/O Interrupt Control Register 10	IIO10IC	XXXX X000b
000078h	Timer B3 Interrupt Control Register	TB3IC	XXXX X000b
000079h	CAN4 Wake-up Interrupt Control Register (1)	C4WIC	XXXX X000b
00007Ah	INT5 Interrupt Control Register	INT5IC	XX00 X000b
00007Bh	CAN0 Wake-up Interrupt Control Register (1)	C0WIC	XXXX X000b
00007Ch	INT3 Interrupt Control Register	INT3IC	XX00 X000b
00007Dh	CAN2 Wake-up Interrupt Control Register	C2WIC	XXXX X000b
00007Eh	INT1 Interrupt Control Register	INT1IC	XX00 X000b
00007Fh	LIN Low Detection Interrupt Control Register	LLDIC	XXXX X000b
000080h			
000081h	UART2 Transmit/NACK Interrupt Control Register	S2TIC	XXXX X000b
000082h			
000083h			
000084h			
000085h			
000086h			
000087h	UART2 Start Condition/Stop Condition Detection Interrupt Control Register	BCN2IC	XXXX X000b

X: Undefined
Blanks are reserved. No access is allowed.

Note:

1. Channels CAN0 and CAN4 are not available in the R32C/142 Group.

Table 4.3 SFR List (3)

Address	Register	Symbol	Reset Value
000088h	DMA1 Transfer Complete Interrupt Control Register	DM1IC	XXXX X000b
000089h	UART1 Start Condition/Stop Condition Detection Interrupt Control Register	BCN1IC	XXXX X000b
00008Ah	DMA3 Transfer Complete Interrupt Control Register	DM3IC	XXXX X000b
00008Bh	Key Input Interrupt Control Register	KUPIC	XXXX X000b
00008Ch	Timer A1 Interrupt Control Register	TA1IC	XXXX X000b
00008Dh	Intelligent I/O Interrupt Control Register 1	IIO1IC	XXXX X000b
00008Eh	Timer A3 Interrupt Control Register	TA3IC	XXXX X000b
00008Fh	Intelligent I/O Interrupt Control Register 3	IIO3IC	XXXX X000b
000090h	UART0 Transmit/NACK Interrupt Control Register	S0TIC	XXXX X000b
000091h	Intelligent I/O Interrupt Control Register 5	IIO5IC	XXXX X000b
000092h	UART1 Transmit/NACK Interrupt Control Register	S1TIC	XXXX X000b
000093h	Intelligent I/O Interrupt Control Register 7	IIO7IC	XXXX X000b
000094h	Timer B0 Interrupt Control Register	TB0IC	XXXX X000b
000095h	Intelligent I/O Interrupt Control Register 9	IIO9IC	XXXX X000b
000096h	Timer B2 Interrupt Control Register	TB2IC	XXXX X000b
000097h	Intelligent I/O Interrupt Control Register 11	IIO11IC	XXXX X000b
000098h	Timer B4 Interrupt Control Register	TB4IC	XXXX X000b
000099h	CAN5 Wake-up Interrupt Control Register	C5WIC	XXXX X000b
00009Ah	INT4 Interrupt Control Register	INT4IC	XX00 X000b
00009Bh	CAN1 Wake-up Interrupt Control Register (1)	C1WIC	XXXX X000b
00009Ch	INT2 Interrupt Control Register	INT2IC	XX00 X000b
00009Dh	CAN3 Wake-up Interrupt Control Register	C3WIC	XXXX X000b
00009Eh	INT0 Interrupt Control Register	INT0IC	XX00 X000b
00009Fh			
0000A0h	Intelligent I/O Interrupt Request Register 0	IIO0IR	0000 0XX1b
0000A1h	Intelligent I/O Interrupt Request Register 1	IIO1IR	0000 0XX1b
0000A2h	Intelligent I/O Interrupt Request Register 2	IIO2IR	0000 0X01b
0000A3h	Intelligent I/O Interrupt Request Register 3	IIO3IR	0000 0XX1b
0000A4h	Intelligent I/O Interrupt Request Register 4	IIO4IR	000X 0XX1b
0000A5h	Intelligent I/O Interrupt Request Register 5	IIO5IR	0000 00X1b
0000A6h	Intelligent I/O Interrupt Request Register 6	IIO6IR	0000 00X1b
0000A7h	Intelligent I/O Interrupt Request Register 7	IIO7IR	000X 00X1b
0000A8h	Intelligent I/O Interrupt Request Register 8	IIO8IR	0000 00X1b
0000A9h	Intelligent I/O Interrupt Request Register 9	IIO9IR	0000 00X1b
0000AAh	Intelligent I/O Interrupt Request Register 10	IIO10IR	0000 00X1b
0000ABh	Intelligent I/O Interrupt Request Register 11	IIO11IR	0000 00X1b
0000ACh			
0000ADh			
0000AEh			
0000AFh			

X: Undefined
Blanks are reserved. No access is allowed.

Note:

1. Channel CAN1 is not available in the R32C/142 Group.

Table 4.4 SFR List (4)

Address	Register	Symbol	Reset Value
0000B0h	Intelligent I/O Interrupt Enable Register 0	IIO0IE	00h
0000B1h	Intelligent I/O Interrupt Enable Register 1	IIO1IE	00h
0000B2h	Intelligent I/O Interrupt Enable Register 2	IIO2IE	00h
0000B3h	Intelligent I/O Interrupt Enable Register 3	IIO3IE	00h
0000B4h	Intelligent I/O Interrupt Enable Register 4	IIO4IE	00h
0000B5h	Intelligent I/O Interrupt Enable Register 5	IIO5IE	00h
0000B6h	Intelligent I/O Interrupt Enable Register 6	IIO6IE	00h
0000B7h	Intelligent I/O Interrupt Enable Register 7	IIO7IE	00h
0000B8h	Intelligent I/O Interrupt Enable Register 8	IIO8IE	00h
0000B9h	Intelligent I/O Interrupt Enable Register 9	IIO9IE	00h
0000BAh	Intelligent I/O Interrupt Enable Register 10	IIO10IE	00h
0000BBh	Intelligent I/O Interrupt Enable Register 11	IIO11IE	00h
0000BCh			
0000BDh			
0000BEh			
0000BFh			
0000C0h	Serial Bus Interface 0 Interrupt Control Register	SS0IC	XXXX X000b
0000C1h	CAN0 Transmit Interrupt Control Register (1)	C0TIC	XXXX X000b
0000C2h			
0000C3h	CAN0 Error Interrupt Control Register (1)	C0EIC	XXXX X000b
0000C4h			
0000C5h	CAN1 Receive Interrupt Control Register (1)	C1RIC	XXXX X000b
0000C6h			
0000C7h	CAN2 Transmit Interrupt Control Register	C2TIC	
0000C8h	CAN4 Transmit FIFO Interrupt Control Register (1)	C4FTIC	XXXX X000b
0000C9h	CAN2 Error Interrupt Control Register	C2EIC	XXXX X000b
0000CAh	CAN5 Transmit FIFO Interrupt Control Register	C5FTIC	XXXX X000b
0000CBh	CAN3 Receive Interrupt Control Register	C3RIC	XXXX X000b
0000CCh			
0000CDh	CAN4 Transmit Interrupt Control Register (1)	C4TIC	XXXX X000b
0000CEh			
0000CFh	CAN4 Error Interrupt Control Register (1)	C4EIC	XXXX X000b
0000D0h	CAN0 Transmit FIFO Interrupt Control Register (1)	C0FTIC	XXXX X000b
0000D1h	CAN5 Receive Interrupt Control Register	C5RIC	XXXX X000b
0000D2h	CAN1 Transmit FIFO Interrupt Control Register (1)	C1FTIC	XXXX X000b
0000D3h			
0000D4h	CAN2 Transmit FIFO Interrupt Control Register	C2FTIC	XXXX X000b
0000D5h	LIN0 Interrupt Control Register	L0IC	XXXX X000b
0000D6h	CAN3 Transmit FIFO Interrupt Control Register	C3FTIC	XXXX X000b
0000D7h			
0000D8h			
0000D9h			
0000DAh			
0000DBh			
0000DCh			
0000DDh	UART3 Transmit Interrupt Control Register	S3TIC	XXXX X000b
0000DEh			
0000DFh	UART4 Transmit Interrupt Control Register	S4TIC	XXXX X000b

X: Undefined

Blanks are reserved. No access is allowed.

Note:

1. Channels CAN0, CAN1, and CAN4 are not available in the R32C/142 Group.

Table 4.5 SFR List (5)

Address	Register	Symbol	Reset Value
0000E0h	Serial Bus Interface 1 Interrupt Control Register	SS1IC	XXXX X000b
0000E1h	CAN0 Receive Interrupt Control Register (1)	C0RIC	XXXX X000b
0000E2h			
0000E3h	CAN1 Transmit Interrupt Control Register (1)	C1TIC	XXXX X000b
0000E4h			
0000E5h	CAN1 Error Interrupt Control Register (1)	C1EIC	XXXX X000b
0000E6h			
0000E7h	CAN2 Receive Interrupt Control Register	C2RIC	XXXX X000b
0000E8h	CAN4 Receive FIFO/Gateway Channel 4 Interrupt Control Register (1)	C4FRIC/GW4IC	XXXX X000b
0000E9h	CAN3 Transmit Interrupt Control Register	C3TIC	XXXX X000b
0000EAh	CAN5 Receive FIFO/Gateway Channel 5 Interrupt Control Register	C5FRIC/GW5IC	XXXX X000b
0000EBh	CAN3 Error Interrupt Control Register	C3EIC	XXXX X000b
0000ECh			
0000EDh	CAN4 Receive Interrupt Control Register (1)	C4RIC	XXXX X000b
0000EEh			
0000EFh	CAN5 Transmit Interrupt Control Register	C5TIC	XXXX X000b
0000F0h	CAN0 Receive FIFO/Gateway Channel 0 Interrupt Control Register (1)	C0FRIC/GW0IC	XXXX X000b
0000F1h	CAN5 Error Interrupt Control Register	C5EIC	XXXX X000b
0000F2h	CAN1 Receive FIFO/Gateway Channel 1 Interrupt Control Register (1)	C1FRIC/GW1IC	XXXX X000b
0000F3h			
0000F4h	CAN2 Receive FIFO/Gateway Channel 2 Interrupt Control Register	C2FRIC/GW2IC	XXXX X000b
0000F5h	LIN1 Interrupt Control Register	L1IC	XXXX X000b
0000F6h	CAN3 Receive FIFO/Gateway Channel 3 Interrupt Control Register	C3FRIC/GW3IC	XXXX X000b
0000F7h			
0000F8h	Gateway Error Interrupt Control Register	GWEIC	XXXX X000b
0000F9h			
0000FAh			
0000FBh			
0000FCh			
0000FDh	UART3 Receive Interrupt Control Register	S3RIC	XXXX X000b
0000FEh			
0000FFh	UART4 Receive Interrupt Control Register	S4RIC	XXXX X000b
000100h	Group 1 Time Measurement/Waveform Generation Register 0	G1TM0/G1PO0	XXXXh
000101h			
000102h	Group 1 Time Measurement/Waveform Generation Register 1	G1TM1/G1PO1	XXXXh
000103h			
000104h	Group 1 Time Measurement/Waveform Generation Register 2	G1TM2/G1PO2	XXXXh
000105h			
000106h	Group 1 Time Measurement/Waveform Generation Register 3	G1TM3/G1PO3	XXXXh
000107h			

X: Undefined
Blanks are reserved. No access is allowed.

Note:
1. Channels CAN0, CAN1, and CAN4 are not available in the R32C/142 Group.

Table 4.6 SFR List (6)

Address	Register	Symbol	Reset Value
000108h	Group 1 Time Measurement/Waveform Generation Register 4	G1TM4/G1PO4	XXXXh
000109h			
00010Ah	Group 1 Time Measurement/Waveform Generation Register 5	G1TM5/G1PO5	XXXXh
00010Bh			
00010Ch	Group 1 Time Measurement/Waveform Generation Register 6	G1TM6/G1PO6	XXXXh
00010Dh			
00010Eh	Group 1 Time Measurement/Waveform Generation Register 7	G1TM7/G1PO7	XXXXh
00010Fh			
000110h	Group 1 Waveform Generation Control Register 0	G1POCR0	0000 X000b
000111h	Group 1 Waveform Generation Control Register 1	G1POCR1	0X00 X000b
000112h	Group 1 Waveform Generation Control Register 2	G1POCR2	0X00 X000b
000113h	Group 1 Waveform Generation Control Register 3	G1POCR3	0X00 X000b
000114h	Group 1 Waveform Generation Control Register 4	G1POCR4	0X00 X000b
000115h	Group 1 Waveform Generation Control Register 5	G1POCR5	0X00 X000b
000116h	Group 1 Waveform Generation Control Register 6	G1POCR6	0X00 X000b
000117h	Group 1 Waveform Generation Control Register 7	G1POCR7	0X00 X000b
000118h	Group 1 Time Measurement Control Register 0	G1TMCR0	00h
000119h	Group 1 Time Measurement Control Register 1	G1TMCR1	00h
00011Ah	Group 1 Time Measurement Control Register 2	G1TMCR2	00h
00011Bh	Group 1 Time Measurement Control Register 3	G1TMCR3	00h
00011Ch	Group 1 Time Measurement Control Register 4	G1TMCR4	00h
00011Dh	Group 1 Time Measurement Control Register 5	G1TMCR5	00h
00011Eh	Group 1 Time Measurement Control Register 6	G1TMCR6	00h
00011Fh	Group 1 Time Measurement Control Register 7	G1TMCR7	00h
000120h	Group 1 Base Timer Register	G1BT	XXXXh
000121h			
000122h	Group 1 Base Timer Control Register 0	G1BCR0	0000 0000b
000123h	Group 1 Base Timer Control Register 1	G1BCR1	0000 0000b
000124h	Group 1 Time Measurement Prescaler Register 6	G1TPR6	00h
000125h	Group 1 Time Measurement Prescaler Register 7	G1TPR7	00h
000126h	Group 1 Function Enable Register	G1FE	00h
000127h	Group 1 Function Select Register	G1FS	00h
000128h			
000129h			
00012Ah			
00012Bh			
00012Ch			
00012Dh			
00012Eh			
00012Fh			

X: Undefined

Blanks are reserved. No access is allowed.

Table 4.7 SFR List (7)

Address	Register	Symbol	Reset Value
000130h to 00016Fh			
000170h			
000171h			
000172h			
000173h			
000174h			
000175h			
000176h			
000177h			
000178h			
000179h			
00017Ah			
00017Bh			
00017Ch			
00017Dh			
00017Eh			
00017Fh			
000180h	Group 0 Time Measurement/Waveform Generation Register 0	G0TM0/G0PO0	XXXXh
000181h			
000182h	Group 0 Time Measurement/Waveform Generation Register 1	G0TM1/G0PO1	XXXXh
000183h			
000184h	Group 0 Time Measurement/Waveform Generation Register 2	G0TM2/G0PO2	XXXXh
000185h			
000186h	Group 0 Time Measurement/Waveform Generation Register 3	G0TM3/G0PO3	XXXXh
000187h			
000188h	Group 0 Time Measurement/Waveform Generation Register 4	G0TM4/G0PO4	XXXXh
000189h			
00018Ah	Group 0 Time Measurement/Waveform Generation Register 5	G0TM5/G0PO5	XXXXh
00018Bh			
00018Ch	Group 0 Time Measurement/Waveform Generation Register 6	G0TM6/G0PO6	XXXXh
00018Dh			
00018Eh	Group 0 Time Measurement/Waveform Generation Register 7	G0TM7/G0PO7	XXXXh
00018Fh			
000190h	Group 0 Waveform Generation Control Register 0	G0POCR0	0000 X000b
000191h	Group 0 Waveform Generation Control Register 1	G0POCR1	0X00 X000b
000192h	Group 0 Waveform Generation Control Register 2	G0POCR2	0X00 X000b
000193h	Group 0 Waveform Generation Control Register 3	G0POCR3	0X00 X000b
000194h	Group 0 Waveform Generation Control Register 4	G0POCR4	0X00 X000b
000195h	Group 0 Waveform Generation Control Register 5	G0POCR5	0X00 X000b
000196h	Group 0 Waveform Generation Control Register 6	G0POCR6	0X00 X000b
000197h	Group 0 Waveform Generation Control Register 7	G0POCR7	0X00 X000b
000198h	Group 0 Time Measurement Control Register 0	G0TMCR0	00h
000199h	Group 0 Time Measurement Control Register 1	G0TMCR1	00h
00019Ah	Group 0 Time Measurement Control Register 2	G0TMCR2	00h
00019Bh	Group 0 Time Measurement Control Register 3	G0TMCR3	00h
00019Ch	Group 0 Time Measurement Control Register 4	G0TMCR4	00h
00019Dh	Group 0 Time Measurement Control Register 5	G0TMCR5	00h
00019Eh	Group 0 Time Measurement Control Register 6	G0TMCR6	00h
00019Fh	Group 0 Time Measurement Control Register 7	G0TMCR7	00h

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.8 SFR List (8)

Address	Register	Symbol	Reset Value
0001A0h	Group 0 Base Timer Register	G0BT	XXXXh
0001A1h			
0001A2h	Group 0 Base Timer Control Register 0	G0BCR0	0000 0000b
0001A3h	Group 0 Base Timer Control Register 1	G0BCR1	0000 0000b
0001A4h	Group 0 Time Measurement Prescaler Register 6	G0TPR6	00h
0001A5h	Group 0 Time Measurement Prescaler Register 7	G0TPR7	00h
0001A6h	Group 0 Function Enable Register	G0FE	00h
0001A7h	Group 0 Function Select Register	G0FS	00h
0001A8h			
0001A9h			
0001AAh			
0001ABh			
0001ACh			
0001ADh			
0001AEh			
0001AFh			
0001B0h			
0001B1h			
0001B2h			
0001B3h			
0001B4h			
0001B5h			
0001B6h			
0001B7h			
0001B8h			
0001B9h			
0001BAh			
0001BBh			
0001BCh			
0001BDh			
0001BEh			
0001BFh			
0001C0h			
0001C1h			
0001C2h			
0001C3h			
0001C4h			
0001C5h			
0001C6h			
0001C7h			
0001C8h			
0001C9h			
0001CAh			
0001CBh			
0001CCh			
0001CDh			
0001CEh			
0001CFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.9 SFR List (9)

Address	Register	Symbol	Reset Value
0001D0h			
0001D1h			
0001D2h			
0001D3h			
0001D4h			
0001D5h			
0001D6h			
0001D7h			
0001D8h			
0001D9h			
0001DAh			
0001DBh			
0001DCh			
0001DDh			
0001DEh			
0001DFh			
0001E0h	UART3 Transmit/Receive Mode Register	U3MR	00h
0001E1h	UART3 Bit Rate Register	U3BRG	XXh
0001E2h	UART3 Transmit Buffer Register	U3TB	XXXXh
0001E3h			
0001E4h	UART3 Transmit/Receive Control Register 0	U3C0	00X0 1000b
0001E5h	UART3 Transmit/Receive Control Register 1	U3C1	XXXX 0010b
0001E6h	UART3 Receive Buffer Register	U3RB	XXXXh
0001E7h			
0001E8h	UART4 Transmit/Receive Mode Register	U4MR	00h
0001E9h	UART4 Bit Rate Register	U4BRG	XXh
0001EAh	UART4 Transmit Buffer Register	U4TB	XXXXh
0001EBh			
0001ECh	UART4 Transmit/Receive Control Register 0	U4C0	00X0 1000b
0001EDh	UART4 Transmit/Receive Control Register 1	U4C1	XXXX 0010b
0001EEh	UART4 Receive Buffer Register	U4RB	XXXXh
0001EFh			
0001F0h	UART3, UART4 Transmit/Receive Control Register 2	U34CON	X000 0000b
0001F1h			
0001F2h			
0001F3h			
0001F4h			
0001F5h			
0001F6h			
0001F7h			
0001F8h			
0001F9h			
0001FAh			
0001FBh			
0001FCh			
0001FDh			
0001FEh			
0001FFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.10 SFR List (10)

Address	Register	Symbol	Reset Value
000200h	Group0 Phase Shift Waveform Output Mode Clock Division Setting Register	G0SDR	00h
000201h	Group0 Phase Shift Waveform Output Mode Control Register	G0PSCR	00h
000202h	Group1 Phase Shift Waveform Output Mode Clock Division Setting Register	G1SDR	00h
000203h	Group1 Phase Shift Waveform Output Mode Control Register	G1PSCR	00h
000204h			
000205h			
000206h			
000207h			
000208h	Timer B Event Clock Select Register	TBECKS	0000 0000b
000209h			
00020Ah			
00020Bh			
00020Ch			
00020Dh			
00020Eh			
00020Fh			
000210h	IIO0_7 Digital Debounce Register	IC07DDR	FFh
000211h	IIO1_7 Digital Debounce Register	IC17DDR	FFh
000212h			
000213h			
000214h			
000215h			
000216h			
000217h			
000218h			
000219h			
00021Ah			
00021Bh			
00021Ch			
00021Dh			
00021Eh			
00021Fh			
000220h	Timer A1 Mirror Register	TA1M	XXXXh
000221h			
000222h	Timer A1-1 Mirror Register	TA11M	XXXXh
000223h			
000224h	Timer A2 Mirror Register	TA2M	XXXXh
000225h			
000226h	Timer A2-1 Mirror Register	TA21M	XXXXh
000227h			
000228h	Timer A4 Mirror Register	TA4M	XXXXh
000229h			
00022Ah	Timer A4-1 Mirror Register	TA41M	XXXXh
00022Bh			
00022Ch			
00022Dh			
00022Eh			
00022Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.11 SFR List (11)

Address	Register	Symbol	Reset Value
000230h to 0002BFh			
0002C0h	X0 Register/Y0 Register	X0R/Y0R	XXXXh
0002C1h			
0002C2h	X1 Register/Y1 Register	X1R/Y1R	XXXXh
0002C3h			
0002C4h	X2 Register/Y2 Register	X2R/Y2R	XXXXh
0002C5h			
0002C6h	X3 Register/Y3 Register	X3R/Y3R	XXXXh
0002C7h			
0002C8h	X4 Register/Y4 Register	X4R/Y4R	XXXXh
0002C9h			
0002CAh	X5 Register/Y5 Register	X5R/Y5R	XXXXh
0002CBh			
0002CCh	X6 Register/Y6 Register	X6R/Y6R	XXXXh
0002CDh			
0002CEh	X7 Register/Y7 Register	X7R/Y7R	XXXXh
0002CFh			
0002D0h	X8 Register/Y8 Register	X8R/Y8R	XXXXh
0002D1h			
0002D2h	X9 Register/Y9 Register	X9R/Y9R	XXXXh
0002D3h			
0002D4h	X10 Register/Y10 Register	X10R/Y10R	XXXXh
0002D5h			
0002D6h	X11 Register/Y11 Register	X11R/Y11R	XXXXh
0002D7h			
0002D8h	X12 Register/Y12 Register	X12R/Y12R	XXXXh
0002D9h			
0002DAh	X13 Register/Y13 Register	X13R/Y13R	XXXXh
0002DBh			
0002DCh	X14 Register/Y14 Register	X14R/Y14R	XXXXh
0002DDh			
0002DEh	X15 Register/Y15 Register	X15R/Y15R	XXXXh
0002DFh			
0002E0h	X-Y Control Register	XYC	XXXX XX00b
0002E1h			
0002E2h			
0002E3h			
0002E4h	UART1 Special Mode Register 4	U1SMR4	00h
0002E5h	UART1 Special Mode Register 3	U1SMR3	00h
0002E6h	UART1 Special Mode Register 2	U1SMR2	00h
0002E7h	UART1 Special Mode Register	U1SMR	00h
0002E8h	UART1 Transmit/Receive Mode Register	U1MR	00h
0002E9h	UART1 Bit Rate Register	U1BRG	XXh
0002EAh	UART1 Transmit Buffer Register	U1TB	XXXXh
0002EBh			
0002ECh	UART1 Transmit/Receive Control Register 0	U1C0	0000 1000b
0002EDh	UART1 Transmit/Receive Control Register 1	U1C1	0000 0010b
0002EEh	UART1 Receive Buffer Register	U1RB	XXXXh
0002EFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.12 SFR List (12)

Address	Register	Symbol	Reset Value
0002F0h			
0002F1h			
0002F2h			
0002F3h			
0002F4h			
0002F5h			
0002F6h			
0002F7h			
0002F8h			
0002F9h			
0002FAh			
0002FBh			
0002FCh			
0002FDh			
0002FEh			
0002FFh			
000300h	Count Start Register for Timers B3, B4, and B5	TBSR	000X XXXXb
000301h			
000302h	Timer A1-1 Register	TA11	XXXXh
000303h			
000304h	Timer A2-1 Register	TA21	XXXXh
000305h			
000306h	Timer A4-1 Register	TA41	XXXXh
000307h			
000308h	Three-phase PWM Control Register 0	INVC0	00h
000309h	Three-phase PWM Control Register 1	INVC1	00h
00030Ah	Three-phase Output Buffer Register 0	IDB0	XX11 1111b
00030Bh	Three-phase Output Buffer Register 1	IDB1	XX11 1111b
00030Ch	Dead Time Timer	DTT	XXh
00030Dh	Timer B2 Interrupt Generating Frequency Set Counter	ICTB2	XXh
00030Eh			
00030Fh			
000310h	Timer B3 Register	TB3	XXXXh
000311h			
000312h	Timer B4 Register	TB4	XXXXh
000313h			
000314h	Timer B5 Register	TB5	XXXXh
000315h			
000316h			
000317h			
000318h			
000319h			
00031Ah			
00031Bh	Timer B3 Mode Register	TB3MR	00XX 0000b
00031Ch	Timer B4 Mode Register	TB4MR	00XX 0000b
00031Dh	Timer B5 Mode Register	TB5MR	00XX 0000b
00031Eh			
00031Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.13 SFR List (13)

Address	Register	Symbol	Reset Value
000320h			
000321h			
000322h			
000323h			
000324h			
000325h			
000326h			
000327h			
000328h			
000329h			
00032Ah			
00032Bh			
00032Ch			
00032Dh			
00032Eh			
00032Fh			
000330h			
000331h			
000332h			
000333h			
000334h	UART2 Special Mode Register 4	U2SMR4	00h
000335h	UART2 Special Mode Register 3	U2SMR3	00h
000336h	UART2 Special Mode Register 2	U2SMR2	00h
000337h	UART2 Special Mode Register	U2SMR	00h
000338h	UART2 Transmit/Receive Mode Register	U2MR	00h
000339h	UART2 Bit Rate Register	U2BRG	XXh
00033Ah	UART2 Transmit Buffer Register	U2TB	XXXXh
00033Bh			
00033Ch	UART2 Transmit/Receive Control Register 0	U2C0	0000 1000b
00033Dh	UART2 Transmit/Receive Control Register 1	U2C1	0000 0010b
00033Eh	UART2 Receive Buffer Register	U2RB	XXXXh
00033Fh			
000340h	Count Start Register	TABSR	0000 0000b
000341h	Clock Prescaler Reset Register	CPSRF	0XXX XXXXb
000342h	One-shot Start Register	ONSF	0000 0000b
000343h	Trigger Select Register	TRGSR	0000 0000b
000344h	Increment/Decrement Select Register	UDF	0000 0000b
000345h			
000346h	Timer A0 Register	TA0	XXXXh
000347h			
000348h	Timer A1 Register	TA1	XXXXh
000349h			
00034Ah	Timer A2 Register	TA2	XXXXh
00034Bh			
00034Ch	Timer A3 Register	TA3	XXXXh
00034Dh			
00034Eh	Timer A4 Register	TA4	XXXXh
00034Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.14 SFR List (14)

Address	Register	Symbol	Reset Value
000350h	Timer B0 Register	TB0	XXXXh
000351h			
000352h	Timer B1 Register	TB1	XXXXh
000353h			
000354h	Timer B2 Register	TB2	XXXXh
000355h			
000356h	Timer A0 Mode Register	TA0MR	0000 0000b
000357h	Timer A1 Mode Register	TA1MR	0000 0000b
000358h	Timer A2 Mode Register	TA2MR	0000 0000b
000359h	Timer A3 Mode Register	TA3MR	0000 0000b
00035Ah	Timer A4 Mode Register	TA4MR	0000 0000b
00035Bh	Timer B0 Mode Register	TB0MR	00XX 0000b
00035Ch	Timer B1 Mode Register	TB1MR	00XX 0000b
00035Dh	Timer B2 Mode Register	TB2MR	00XX 0000b
00035Eh	Timer B2 Special Mode Register	TB2SC	XXXX XXX0b
00035Fh	Count Source Prescaler Register	TCSPR	0000 0000b
000360h			
000361h			
000362h			
000363h			
000364h	UART0 Special Mode Register 4	U0SMR4	00h
000365h	UART0 Special Mode Register 3	U0SMR3	00h
000366h	UART0 Special Mode Register 2	U0SMR2	00h
000367h	UART0 Special Mode Register	U0SMR	00h
000368h	UART0 Transmit/Receive Mode Register	U0MR	00h
000369h	UART0 Bit Rate Register	U0BRG	XXh
00036Ah	UART0 Transmit Buffer Register	U0TB	XXXXh
00036Bh			
00036Ch	UART0 Transmit/Receive Control Register 0	U0C0	0000 1000b
00036Dh	UART0 Transmit/Receive Control Register 1	U0C1	0000 0010b
00036Eh	UART0 Receive Buffer Register	U0RB	XXXXh
00036Fh			
000370h			
000371h			
000372h			
000373h			
000374h			
000375h			
000376h			
000377h			
000378h			
000379h			
00037Ah			
00037Bh			
00037Ch	CRC Data Register	CRCD	XXXXh
00037Dh			
00037Eh	CRC Input Register	CRCIN	XXh
00037Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.15 SFR List (15)

Address	Register	Symbol	Reset Value
000380h	A/D0 Register 0	AD00	00XXh
000381h			
000382h	A/D0 Register 1	AD01	00XXh
000383h			
000384h	A/D0 Register 2	AD02	00XXh
000385h			
000386h	A/D0 Register 3	AD03	00XXh
000387h			
000388h	A/D0 Register 4	AD04	00XXh
000389h			
00038Ah	A/D0 Register 5	AD05	00XXh
00038Bh			
00038Ch	A/D0 Register 6	AD06	00XXh
00038Dh			
00038Eh	A/D0 Register 7	AD07	00XXh
00038Fh			
000390h			
000391h			
000392h	A/D0 Control Register 4	AD0CON4	XXXX 00XXb
000393h	A/D0 Control Register 5	AD0CON5	00h
000394h	A/D0 Control Register 2	AD0CON2	X00X X000b
000395h	A/D0 Control Register 3	AD0CON3	XXXX X000b
000396h	A/D0 Control Register 0	AD0CON0	00h
000397h	A/D0 Control Register 1	AD0CON1	00h
000398h	D/A Register 0	DA0	XXh
000399h			
00039Ah	D/A Register 1	DA1	XXh
00039Bh			
00039Ch	D/A Control Register	DACON	XXXX XX00b
00039Dh			
00039Eh			
00039Fh			
0003A0h			
0003A1h			
0003A2h			
0003A3h			
0003A4h			
0003A5h			
0003A6h			
0003A7h			
0003A8h			
0003A9h			
0003AAh			
0003ABh			
0003ACh			
0003ADh			
0003AEh			
0003AFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.16 SFR List (16)

Address	Register	Symbol	Reset Value
0003B0h			
0003B1h			
0003B2h			
0003B3h			
0003B4h			
0003B5h			
0003B6h			
0003B7h			
0003B8h			
0003B9h			
0003BAh			
0003BBh			
0003BCh			
0003BDh			
0003BEh			
0003BFh			
0003C0h	Port P0 Register	P0	XXh
0003C1h	Port P1 Register	P1	XXh
0003C2h	Port P0 Direction Register	PD0	0000 0000b
0003C3h	Port P1 Direction Register	PD1	0000 0000b
0003C4h	Port P2 Register	P2	XXh
0003C5h	Port P3 Register	P3	XXh
0003C6h	Port P2 Direction Register	PD2	0000 0000b
0003C7h	Port P3 Direction Register	PD3	0000 0000b
0003C8h	Port P4 Register	P4	XXh
0003C9h	Port P5 Register	P5	XXh
0003CAh	Port P4 Direction Register	PD4	0000 0000b
0003CBh	Port P5 Direction Register	PD5	0000 0000b
0003CCh	Port P6 Register	P6	XXh
0003CDh	Port P7 Register	P7	XXh
0003CEh	Port P6 Direction Register	PD6	0000 0000b
0003CFh	Port P7 Direction Register	PD7	0000 0000b
0003D0h	Port P8 Register	P8	XXh
0003D1h	Port P9 Register	P9	XXh
0003D2h	Port P8 Direction Register	PD8	00X0 0000b
0003D3h	Port P9 Direction Register	PD9	0000 0000b
0003D4h	Port P10 Register	P10	XXh
0003D5h			
0003D6h	Port P10 Direction Register	PD10	0000 0000b
0003D7h			
0003D8h			
0003D9h			
0003DAh			
0003DBh			
0003DCh			
0003DDh			
0003DEh			
0003DFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.17 SFR List (17)

Address	Register	Symbol	Reset Value
0003E0h			
0003E1h			
0003E2h			
0003E3h			
0003E4h			
0003E5h			
0003E6h			
0003E7h			
0003E8h			
0003E9h			
0003EAh			
0003EBh			
0003ECh			
0003EDh			
0003EEh			
0003EFh			
0003F0h	Pull-up Control Register 0	PUR0	0000 0000b
0003F1h	Pull-up Control Register 1	PUR1	XXXX 0000b
0003F2h	Pull-up Control Register 2	PUR2	0000 0000b
0003F3h	Pull-up Control Register 3	PUR3	XXXX XX00b
0003F4h			
0003F5h			
0003F6h			
0003F7h			
0003F8h			
0003F9h			
0003FAh			
0003FBh			
0003FCh			
0003FDh			
0003FEh			
0003FFh	Port Control Register	PCR	XXXX XXX0b

X: Undefined

Blanks are reserved. No access is allowed.

Table 4.18 SFR List (18)

Address	Register	Symbol	Reset Value
040000h	Flash Memory Control Register 0	FMR0	0X01 XX00b
040001h	Flash Memory Status Register 0	FMSR0	1000 0000b
040002h			
040003h			
040004h			
040005h			
040006h			
040007h			
040008h	Flash Register Protection Unlock Register 0	FPR0	00h
040009h	Flash Memory Control Register 1	FMR1	0000 0010b
04000Ah	Block Protect Bit Monitor Register 0	FBPM0	??X? ????b (1)
04000Bh	Block Protect Bit Monitor Register 1	FBPM1	XXX? ????b (1)
04000Ch			
04000Dh			
04000Eh			
04000Fh			
040010h			
040011h			
040012h			
040013h			
040014h			
040015h			
040016h			
040017h			
040018h			
040019h			
04001Ah			
04001Bh			
04001Ch			
04001Dh			
04001Eh			
04001Fh			
040020h	PLL Control Register 0	PLC0	0000 0001b
040021h	PLL Control Register 1	PLC1	0001 1111b
040022h			
040023h			
040024h			
040025h			
040026h			
040027h			
040028h			
040029h			
04002Ah			
04002Bh			
04002Ch			
04002Dh			
04002Eh			
04002Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Note:
1. The reset value reflects the value of the protect bit for each block in the flash memory.

Table 4.19 SFR List (19)

Address	Register	Symbol	Reset Value
040030h to 04003Fh			
040040h			
040041h			
040042h			
040043h			
040044h	Processor Mode Register 0	PM0	1000 0000b
040045h			
040046h	System Clock Control Register 0	CM0	0000 1000b
040047h	System Clock Control Register 1	CM1	0010 0000b
040048h	Processor Mode Register 3	PM3	00h
040049h			
04004Ah	Protect Register	PRCR	XXXX X000b
04004Bh			
04004Ch	Protect Register 3	PRCR3	0000 0000b
04004Dh	Oscillator Stop Detection Register	CM2	00h
04004Eh			
04004Fh			
040050h			
040051h			
040052h			
040053h	Processor Mode Register 2	PM2	00h
040054h			
040055h			
040056h			
040057h			
040058h			
040059h			
04005Ah	Low Speed Mode Clock Control Register	CM3	XXXX XX00b
04005Bh			
04005Ch			
04005Dh			
04005Eh			
04005Fh			
040060h	Voltage Regulator Control Register	VRCR	0000 0000b
040061h			
040062h			
040063h			
040064h			
040065h			
040066h			
040067h			
040068h to 040093h			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.20 SFR List (20)

Address	Register	Symbol	Reset Value
040094h			
040095h			
040096h			
040097h	Three-phase Output Buffer Control Register	IOBC	0XXX XX0Xb
040098h	Input Function Select Register 0	IFS0	X0X0 X0X0b
040099h	Input Function Select Register 1	IFS1	00XX X0X0b
04009Ah	Input Function Select Register 2	IFS2	0000 0000b
04009Bh	Input Function Select Register 3	IFS3	0000 XXXXb
04009Ch			
04009Dh	Input Function Select Register 5	IFS5	XXXX X0X0b
04009Eh	Input Function Select Register 6	IFS6	XXXX 0000b
04009Fh			
0400A0h	Port P0_0 Function Select Register	P0_0S	0XXX X000b
0400A1h	Port P1_0 Function Select Register	P1_0S	XXXX X000b
0400A2h	Port P0_1 Function Select Register	P0_1S	0XXX X000b
0400A3h	Port P1_1 Function Select Register	P1_1S	XXXX X000b
0400A4h	Port P0_2 Function Select Register	P0_2S	0XXX X000b
0400A5h	Port P1_2 Function Select Register	P1_2S	XXXX X000b
0400A6h	Port P0_3 Function Select Register	P0_3S	0XXX X000b
0400A7h	Port P1_3 Function Select Register	P1_3S	XXXX X000b
0400A8h	Port P0_4 Function Select Register	P0_4S	0XXX X000b
0400A9h	Port P1_4 Function Select Register	P1_4S	XXXX X000b
0400AAh	Port P0_5 Function Select Register	P0_5S	0XXX X000b
0400ABh	Port P1_5 Function Select Register	P1_5S	XXXX X000b
0400ACh	Port P0_6 Function Select Register	P0_6S	0XXX X000b
0400ADh	Port P1_6 Function Select Register	P1_6S	XXXX X000b
0400AEh	Port P0_7 Function Select Register	P0_7S	0XXX X000b
0400AFh	Port P1_7 Function Select Register	P1_7S	XXXX X000b
0400B0h	Port P2_0 Function Select Register	P2_0S	0XXX X000b
0400B1h	Port P3_0 Function Select Register	P3_0S	XXXX X000b
0400B2h	Port P2_1 Function Select Register	P2_1S	0XXX X000b
0400B3h	Port P3_1 Function Select Register	P3_1S	XXXX X000b
0400B4h	Port P2_2 Function Select Register	P2_2S	0XXX X000b
0400B5h	Port P3_2 Function Select Register	P3_2S	XXXX X000b
0400B6h	Port P2_3 Function Select Register	P2_3S	0XXX X000b
0400B7h	Port P3_3 Function Select Register	P3_3S	XXXX X000b
0400B8h	Port P2_4 Function Select Register	P2_4S	0XXX X000b
0400B9h	Port P3_4 Function Select Register	P3_4S	XXXX X000b
0400BAh	Port P2_5 Function Select Register	P2_5S	0XXX X000b
0400BBh	Port P3_5 Function Select Register	P3_5S	XXXX X000b
0400BCh	Port P2_6 Function Select Register	P2_6S	0XXX X000b
0400BDh	Port P3_6 Function Select Register	P3_6S	XXXX X000b
0400BEh	Port P2_7 Function Select Register	P2_7S	0XXX X000b
0400BFh	Port P3_7 Function Select Register	P3_7S	XXXX X000b

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.21 SFR List (21)

Address	Register	Symbol	Reset Value
0400C0h	Port P4_0 Function Select Register	P4_0S	XXXX X000b
0400C1h	Port P5_0 Function Select Register	P5_0S	XXXX X000b
0400C2h	Port P4_1 Function Select Register	P4_1S	XXXX X000b
0400C3h	Port P5_1 Function Select Register	P5_1S	XXXX X000b
0400C4h	Port P4_2 Function Select Register	P4_2S	XXXX X000b
0400C5h	Port P5_2 Function Select Register	P5_2S	XXXX X000b
0400C6h	Port P4_3 Function Select Register	P4_3S	XXXX X000b
0400C7h	Port P5_3 Function Select Register	P5_3S	XXXX X000b
0400C8h	Port P4_4 Function Select Register	P4_4S	XXXX X000b
0400C9h	Port P5_4 Function Select Register	P5_4S	XXXX X000b
0400CAh	Port P4_5 Function Select Register	P4_5S	XXXX X000b
0400CBh	Port P5_5 Function Select Register	P5_5S	XXXX X000b
0400CCh	Port P4_6 Function Select Register	P4_6S	XXXX X000b
0400CDh	Port P5_6 Function Select Register	P5_6S	XXXX X000b
0400CEh	Port P4_7 Function Select Register	P4_7S	XXXX X000b
0400CFh	Port P5_7 Function Select Register	P5_7S	XXXX X000b
0400D0h	Port P6_0 Function Select Register	P6_0S	XXXX X000b
0400D1h	Port P7_0 Function Select Register	P7_0S	XXXX X000b
0400D2h	Port P6_1 Function Select Register	P6_1S	XXXX X000b
0400D3h	Port P7_1 Function Select Register	P7_1S	XXXX X000b
0400D4h	Port P6_2 Function Select Register	P6_2S	XXXX X000b
0400D5h	Port P7_2 Function Select Register	P7_2S	XXXX X000b
0400D6h	Port P6_3 Function Select Register	P6_3S	XXXX X000b
0400D7h	Port P7_3 Function Select Register	P7_3S	XXXX X000b
0400D8h	Port P6_4 Function Select Register	P6_4S	XXXX X000b
0400D9h	Port P7_4 Function Select Register	P7_4S	XXXX X000b
0400DAh	Port P6_5 Function Select Register	P6_5S	XXXX X000b
0400DBh	Port P7_5 Function Select Register	P7_5S	XXXX X000b
0400DCh	Port P6_6 Function Select Register	P6_6S	XXXX X000b
0400DDh	Port P7_6 Function Select Register	P7_6S	XXXX X000b
0400DEh	Port P6_7 Function Select Register	P6_7S	XXXX X000b
0400DFh	Port P7_7 Function Select Register	P7_7S	XXXX X000b
0400E0h	Port P8_0 Function Select Register	P8_0S	XXXX X000b
0400E1h			
0400E2h	Port P8_1 Function Select Register	P8_1S	XXXX X000b
0400E3h			
0400E4h	Port P8_2 Function Select Register	P8_2S	XXXX X000b
0400E5h			
0400E6h	Port P8_3 Function Select Register	P8_3S	XXXX X000b
0400E7h	Port P9_3 Function Select Register	P9_3S	0XXX X000b
0400E8h	Port P8_4 Function Select Register	P8_4S	XXXX X000b
0400E9h	Port P9_4 Function Select Register	P9_4S	0XXX X000b
0400EAh			
0400EBh	Port P9_5 Function Select Register	P9_5S	0XXX X000b
0400ECh	Port P8_6 Function Select Register	P8_6S	XXXX X000b
0400EDh	Port P9_6 Function Select Register	P9_6S	0XXX X000b
0400EEh	Port P8_7 Function Select Register	P8_7S	XXXX X000b
0400EFh	Port P9_7 Function Select Register	P9_7S	XXXX X000b

X: Undefined

Blanks are reserved. No access is allowed.

Table 4.22 SFR List (22)

Address	Register	Symbol	Reset Value
0400F0h	Port P10_0 Function Select Register	P10_0S	0XXX X000b
0400F1h			
0400F2h	Port P10_1 Function Select Register	P10_1S	0XXX X000b
0400F3h			
0400F4h	Port P10_2 Function Select Register	P10_2S	0XXX X000b
0400F5h			
0400F6h	Port P10_3 Function Select Register	P10_3S	0XXX X000b
0400F7h			
0400F8h	Port P10_4 Function Select Register	P10_4S	0XXX X000b
0400F9h			
0400FAh	Port P10_5 Function Select Register	P10_5S	0XXX X000b
0400FBh			
0400FCh	Port P10_6 Function Select Register	P10_6S	0XXX X000b
0400FDh			
0400FEh	Port P10_7 Function Select Register	P10_7S	0XXX X000b
0400FFh			
040100h			
040101h			
040102h			
040103h			
040104h			
040105h			
040106h			
040107h			
040108h			
040109h			
04010Ah			
04010Bh			
04010Ch			
04010Dh			
04010Eh			
04010Fh			
040110h			
040111h			
040112h			
040113h			
040114h			
040115h			
040116h			
040117h			
040118h			
040119h			
04011Ah			
04011Bh			
04011Ch			
04011Dh			
04011Eh			
04011Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.23 SFR List (23)

Address	Register	Symbol	Reset Value
040120h to 04403Fh			
044040h			
044041h			
044042h			
044043h			
044044h			
044045h			
044046h			
044047h			
044048h			
044049h			
04404Ah			
04404Bh			
04404Ch	Protect Register 4	PRCR4	0000 0000b
04404Dh	Watchdog Timer Clock Control Register	WDK	0000 0000b
04404Eh	Watchdog Timer Start Register	WDTS	XXXX XXXXb
04404Fh	Watchdog Timer Control Register	WDC	000X XXXXb
044050h			
044051h			
044052h			
044053h			
044054h			
044055h			
044056h			
044057h			
044058h			
044059h			
04405Ah			
04405Bh			
04405Ch			
04405Dh			
04405Eh			
04405Fh	Protect Register 2	PRCR2	0XXX XXXXb

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.24 SFR List (24)

Address	Register	Symbol	Reset Value
044060h			
044061h			
044062h			
044063h			
044064h			
044065h			
044066h			
044067h			
044068h			
044069h			
04406Ah			
04406Bh			
04406Ch			
04406Dh			
04406Eh			
04406Fh	External Interrupt Request Source Select Register 0	IFSR0	0000 0000b
044070h	DMA0 Request Source Select Register 2	DM0SL2	XX00 0000b
044071h	DMA1 Request Source Select Register 2	DM1SL2	XX00 0000b
044072h	DMA2 Request Source Select Register 2	DM2SL2	XX00 0000b
044073h	DMA3 Request Source Select Register 2	DM3SL2	XX00 0000b
044074h			
044075h			
044076h			
044077h			
044078h	DMA0 Request Source Select Register	DM0SL	XXX0 0000b
044079h	DMA1 Request Source Select Register	DM1SL	XXX0 0000b
04407Ah	DMA2 Request Source Select Register	DM2SL	XXX0 0000b
04407Bh	DMA3 Request Source Select Register	DM3SL	XXX0 0000b
04407Ch			
04407Dh	Wake-up IPL Setting Register 2	RIPL2	XX0X 0000b
04407Eh			
04407Fh	Wake-up IPL Setting Register 1	RIPL1	XX0X 0000b
044080h	External Interrupt Input Filter Select Register 0	INTF0	0000 0000b
044081h			
044082h	External Interrupt Input Filter Select Register 1	INTF1	0000 0000b
044083h			
044084h			
044085h			
044086h			
044087h			
044088h			
044089h			
04408Ah			
04408Bh			
04408Ch			
04408Dh			
04408Eh			
04408Fh			

X: Undefined

Blanks are reserved. No access is allowed.

Table 4.25 SFR List (25)

Address	Register	Symbol	Reset Value
044090h to 044DFFh			
044E00h	LIN Channel Window Select/Input Signal Low Detection Status Register	LCW	0000 0000b
044E01h	LIN Baud Rate Generator Control Register	LBRG	0000 0000b
044E02h	LIN Baud Rate Prescaler 0	LBRP0	00h
044E03h	LIN Baud Rate Prescaler 1	LBRP1	00h
044E04h	LIN Mode Register 0	LMD0	0000 0000b
044E05h	LIN Mode Register 1	LMD1	00h
044E06h	LIN Wake-up Setting Register	LWUP	00h
044E07h			
044E08h	LIN Break Field Setting Register	LBRK	0000 0000b
044E09h	LIN Space Setting Register	LSPC	0000 0000b
044E0Ah	LIN Response Field Setting Register	LRFC	0000 0000b
044E0Bh	LIN ID Buffer Register	LIDB	00h
044E0Ch	LIN Status Control Register	LSC	0000 0000b
044E0Dh	LIN Transmission Control Register	LTC	0000 0000b
044E0Eh	LIN Status Register	LST	0000 0000b
044E0Fh	LIN Error Status Register	LEST	0000 0000b
044E10h	LIN Data 1 Buffer Register	LDB1	00h
044E11h	LIN Data 2 Buffer Register	LDB2	00h
044E12h	LIN Data 3 Buffer Register	LDB3	00h
044E13h	LIN Data 4 Buffer Register	LDB4	00h
044E14h	LIN Data 5 Buffer Register	LDB5	00h
044E15h	LIN Data 6 Buffer Register	LDB6	00h
044E16h	LIN Data 7 Buffer Register	LDB7	00h
044E17h	LIN Data 8 Buffer Register	LDB8	00h
044E18h			
044E19h			
044E1Ah			
044E1Bh			
044E1Ch			
044E1Dh			
044E1Eh			
044E1Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.26 SFR List (26)

Address	Register	Symbol	Reset Value
044E20h to 044EFFh			
044F00h			
044F01h			
044F02h			
044F03h			
044F04h			
044F05h			
044F06h	SS0 Receive Data Register	SS0RDR	FFh
044F07h	SS0 Receive Data Register (H)	SS0RDR (H)	FFh
044F08h	SS0 Control Register H	SS0CRH	00h
044F09h	SS0 Control Register L	SS0CRL	0111 1101b
044F0Ah	SS0 Mode Register	SS0MR	0001 0000b
044F0Bh	SS0 Enable Register	SS0ER	00h
044F0Ch	SS0 Status Register	SS0SR	00h
044F0Dh	SS0 Mode Register 2	SS0MR2	00h
044F0Eh	SS0 Transmit Data Register	SS0TDR	FFh
044F0Fh	SS0 Transmit Data Register (H)	SS0TDR (H)	FFh
044F10h			
044F11h			
044F12h			
044F13h			
044F14h			
044F15h			
044F16h	SS1 Receive Data Register	SS1RDR	FFh
044F17h	SS1 Receive Data Register (H)	SS1RDR (H)	FFh
044F18h	SS1 Control Register H	SS1CRH	00h
044F19h	SS1 Control Register L	SS1CRL	0111 1101b
044F1Ah	SS1 Mode Register	SS1MR	0001 0000b
044F1Bh	SS1 Enable Register	SS1ER	00h
044F1Ch	SS1 Status Register	SS1SR	00h
044F1Dh	SS1 Mode Register 2	SS1MR2	00h
044F1Eh	SS1 Transmit Data Register	SS1TDR	FFh
044F1Fh	SS1 Transmit Data Register (H)	SS1TDR (H)	FFh
044F20h			
044F21h			
044F22h			
044F23h			
044F24h			
044F25h			
044F26h			
044F27h			
044F28h to 0471FFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.27 SFR List (27)

Address	Register	Symbol	Reset Value
047200h	Gateway Mode Register	GMR	0000 0000b
047201h			
047202h			
047203h			
047204h	Gateway Routing Table Checksum Control Register	GRMCC	0000 0000b
047205h	Gateway Transmit FIFO Check Control Register	GTFCC	0000 0000b
047206h			
047207h			
047208h	Gateway Transmit FIFO Clear Register	GTFCR	0000 0000b
047209h			
04720Ah			
04720Bh			
04720Ch	Gateway Channel Control Register	GCCR	0000 0000b
04720Dh			
04720Eh			
04720Fh			
047210h			
047211h			
047212h			
047213h			
047214h	Gateway Parity Check Control Register	GPCCR	0000 0000b
047215h			
047216h			
047217h			
047218h	Gateway Time Stamp Timer Control Register	GTSCR	0000 0000b
047219h			
04721Ah			
04721Bh			
04721Ch	Gateway Routing Table Base Pointer Register	GRMBP	00h
04721Dh			
04721Eh			
04721Fh			
047220h	Gateway Transmit FIFO Read Control Register	GTFRC	0000 0000b
047221h	Gateway Transmit FIFO Read Status Register	GTFRS	0000 0000b
047222h			
047223h			
047224h			
047225h			
047226h			
047227h			
047228h			
047229h			
04722Ah			
04722Bh			
04722Ch	Gateway Routing Table Entries Configuration Register	GMREC	0000h
04722Dh			
04722Eh	Gateway Echo-back Control Register	GEBCR	0000 0000b
04722Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.28 SFR List (28)

Address	Register	Symbol	Reset Value
047230h	Gateway Channel 0 FIFO0 Critical Level Configuration Register (1)	GF00CL	00h
047231h	Gateway Channel 1 FIFO0 Critical Level Configuration Register (1)	GF10CL	00h
047232h	Gateway Channel 2 FIFO0 Critical Level Configuration Register	GF20CL	00h
047233h	Gateway Channel 3 FIFO0 Critical Level Configuration Register	GF30CL	00h
047234h	Gateway Channel 4 FIFO0 Critical Level Configuration Register (1)	GF40CL	00h
047235h	Gateway Channel 5 FIFO0 Critical Level Configuration Register	GF50CL	00h
047236h			
047237h			
047238h	Gateway Channel 0 FIFO1 Critical Level Configuration Register (1)	GF01CL	00h
047239h	Gateway Channel 1 FIFO1 Critical Level Configuration Register (1)	GF11CL	00h
04723Ah	Gateway Channel 2 FIFO1 Critical Level Configuration Register	GFF21CL	00h
04723Bh	Gateway Channel 3 FIFO1 Critical Level Configuration Register	GFF31CL	00h
04723Ch	Gateway Channel 4 FIFO1 Critical Level Configuration Register (1)	GFF41CL	00h
04723Dh	Gateway Channel 5 FIFO1 Critical Level Configuration Register	GFF51CL	00h
04723Eh			
04723Fh			
047240h	Gateway Channel Status Register	GCSR	0000 0000b
047241h	Gateway Checksum Calculation/FIFO Check Status Register	GSCFC	0000 0000b
047242h			
047243h			
047244h	Gateway Routing Table Checksum Register	GRMSR	0000 0000h
047245h			
047246h			
047247h			
047248h	Gateway Channel 0 FIFO0 Fill Level (1)	GF00FL	00h
047249h			
04724Ah	Gateway Channel 0 FIFO1 Fill Level (1)	GF01FL	00h
04724Bh			
04724Ch	Gateway Channel 1 FIFO0 Fill Level (1)	GF10FL	00h
04724Dh			
04724Eh	Gateway Channel 1 FIFO1 Fill Level (1)	GF11FL	00h
04724Fh			

X: Undefined

Blanks are reserved. No access is allowed.

Note:

1. Channels CAN0, CAN1, and CAN 4 are not available in the R32C/142 Group.

Table 4.29 SFR List (29)

Address	Register	Symbol	Reset Value
047250h	Gateway Channel 2 FIFO0 Fill Level	GF20FL	00h
047251h			
047252h	Gateway Channel 2 FIFO1 Fill Level	GF21FL	00h
047253h			
047254h	Gateway Channel 3 FIFO0 Fill Level	GF30FL	00h
047255h			
047256h	Gateway Channel 3 FIFO1 Fill Level	GF31FL	00h
047257h			
047258h	Gateway Channel 4 FIFO0 Fill Level (1)	GF40FL	00h
047259h			
04725Ah	Gateway Channel 4 FIFO1 Fill Level (1)	GF41FL	00h
04725Bh			
04725Ch	Gateway Channel 5 FIFO0 Fill Level	GF50FL	00h
04725Dh			
04725Eh	Gateway Channel 5 FIFO1 Fill Level	GF51FL	00h
04725Fh			
047260h	Gateway Routing Error Status Register	GRESR	0000 0000b
047261h			
047262h	Gateway Error Entry Indication Register	GEEIR	0000h
047263h			
047264h			
047265h			
047266h			
047267h			
047268h	Gateway Time Stamp Timer Register	GTSTR	0000h
047269h			
04726Ah			
04726Bh			
04726Ch			
04726Dh			
04726Eh			
04726Fh			
047270h	Gateway Channel 0 Transmit FIFO Interrupt Enable Register (1)	GC0IE	0000 0000b
047271h	Gateway Channel 1 Transmit FIFO Interrupt Enable Register (1)	GC1IE	0000 0000b
047272h	Gateway Channel 2 Transmit FIFO Interrupt Enable Register	GC2IE	0000 0000b
047273h	Gateway Channel 3 Transmit FIFO Interrupt Enable Register	GC3IE	0000 0000b
047274h	Gateway Channel 4 Transmit FIFO Interrupt Enable Register (1)	GC4IE	0000 0000b
047275h	Gateway Channel 5 Transmit FIFO Interrupt Enable Register	GC5IE	0000 0000b
047276h			
047277h			
047278h	Gateway Channel 0 Transmit FIFO Status Register (1)	GC0SR	0000 0000b
047279h	Gateway Channel 1 Transmit FIFO Status Register (1)	GC1SR	0000 0000b
04727Ah	Gateway Channel 2 Transmit FIFO Status Register	GC2SR	0000 0000b
04727Bh	Gateway Channel 3 Transmit FIFO Status Register	GC3SR	0000 0000b
04727Ch	Gateway Channel 4 Transmit FIFO Status Register (1)	GC4SR	0000 0000b
04727Dh	Gateway Channel 5 Transmit FIFO Status Register	GC5SR	0000 0000b
04727Eh			
04727Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Note:

1. Channels CAN0, CAN1, and CAN 4 are not available in the R32C/142 Group.

Table 4.30 SFR List (30)

Address	Register	Symbol	Reset Value
047280h	Gateway Error Interrupt Enable Register	GIER	0000 0000b
047281h			
047282h			
047283h			
047284h	Gateway Error Status Register	GSR	0000 0000b
047285h			
047286h			
047287h			
047288h			
047289h			
04728Ah			
04728Bh			
04728Ch			
04728Dh			
04728Eh			
04728Fh			
047290h	Gateway Transmit FIFO Read Register 0	GFRR0	0000 0000h
047291h			
047292h			
047293h			
047294h	Gateway Transmit FIFO Read Register 1	GFRR1	0000 0000h
047295h			
047296h			
047297h			
047298h	Gateway Transmit FIFO Read Register 2	GFRR2	0000 0000h
047299h			
04729Ah			
04729Bh			
04729Ch	Gateway Transmit FIFO Read Register 3	GFRR3	0000 0000h
04729Dh			
04729Eh			
04729Fh			
0472A0h			
0472A1h			
0472A2h			
0472A3h			
0472A4h			
0472A5h			
0472A6h			
0472A7h			
0472A8h			
0472A9h			
0472AAh			
0472ABh			
0472ACh			
0472ADh			
0472AEh			
0472AFh			
0472B0h to 0472FFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.31 SFR List (31)

Address	Register	Symbol	Reset Value
047300h	Gateway Routing Table Register 0L	GRM0L	XXXX XXXXh
047301h			
047302h			
047303h			
047304h	Gateway Routing Table Register 0H	GRM0H	XXXX XXXXh
047305h			
047306h			
047307h			
047308h	Gateway Routing Table Register 1L	GRM1L	XXXX XXXXh
047309h			
04730Ah			
04730Bh			
04730Ch	Gateway Routing Table Register 1H	GRM1H	XXXX XXXXh
04730Dh			
04730Eh			
04730Fh			
047310h	Gateway Routing Table Register 2L	GRM2L	XXXX XXXXh
047311h			
047312h			
047313h			
047314h	Gateway Routing Table Register 2H	GRM2H	XXXX XXXXh
047315h			
047316h			
047317h			
047318h	Gateway Routing Table Register 3L	GRM3L	XXXX XXXXh
047319h			
04731Ah			
04731Bh			
04731Ch	Gateway Routing Table Register 3H	GRM3H	XXXX XXXXh
04731Dh			
04731Eh			
04731Fh			
047320h	Gateway Routing Table Register 4L	GRM4L	XXXX XXXXh
047321h			
047322h			
047323h			
047324h	Gateway Routing Table Register 4H	GRM4H	XXXX XXXXh
047325h			
047326h			
047327h			
047328h	Gateway Routing Table Register 5L	GRM5L	XXXX XXXXh
047329h			
04732Ah			
04732Bh			
04732Ch	Gateway Routing Table Register 5H	GRM5H	XXXX XXXXh
04732Dh			
04732Eh			
04732Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.32 SFR List (32)

Address	Register	Symbol	Reset Value
047330h	Gateway Routing Table Register 6L	GRM6L	XXXX XXXXh
047331h			
047332h			
047333h			
047334h	Gateway Routing Table Register 6H	GRM6H	XXXX XXXXh
047335h			
047336h			
047337h			
047338h	Gateway Routing Table Register 7L	GRM7L	XXXX XXXXh
047339h			
04733Ah			
04733Bh			
04733Ch	Gateway Routing Table Register 7H	GRM7H	XXXX XXXXh
04733Dh			
04733Eh			
04733Fh			
047340h	Gateway Routing Table Register 8L	GRM8L	XXXX XXXXh
047341h			
047342h			
047343h			
047344h	Gateway Routing Table Register 8H	GRM8H	XXXX XXXXh
047345h			
047346h			
047347h			
047348h	Gateway Routing Table Register 9L	GRM9L	XXXX XXXXh
047349h			
04734Ah			
04734Bh			
04734Ch	Gateway Routing Table Register 9H	GRM9H	XXXX XXXXh
04734Dh			
04734Eh			
04734Fh			
047350h	Gateway Routing Table Register 10L	GRM10L	XXXX XXXXh
047351h			
047352h			
047353h			
047354h	Gateway Routing Table Register 10H	GRM10H	XXXX XXXXh
047355h			
047356h			
047357h			
047358h	Gateway Routing Table Register 11L	GRM11L	XXXX XXXXh
047359h			
04735Ah			
04735Bh			
04735Ch	Gateway Routing Table Register 11H	GRM11H	XXXX XXXXh
04735Dh			
04735Eh			
04735Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.33 SFR List (33)

Address	Register	Symbol	Reset Value
047360h	Gateway Routing Table Register 12L	GRM12L	XXXX XXXXh
047361h			
047362h			
047363h			
047364h	Gateway Routing Table Register 12H	GRM12H	XXXX XXXXh
047365h			
047366h			
047367h			
047368h	Gateway Routing Table Register 13L	GRM13L	XXXX XXXXh
047369h			
04736Ah			
04736Bh			
04736Ch	Gateway Routing Table Register 13H	GRM13H	XXXX XXXXh
04736Dh			
04736Eh			
04736Fh			
047370h	Gateway Routing Table Register 14L	GRM14L	XXXX XXXXh
047371h			
047372h			
047373h			
047374h	Gateway Routing Table Register 14H	GRM14H	XXXX XXXXh
047375h			
047376h			
047377h			
047378h	Gateway Routing Table Register 15L	GRM15L	XXXX XXXXh
047379h			
04737Ah			
04737Bh			
04737Ch	Gateway Routing Table Register 15H	GRM15H	XXXX XXXXh
04737Dh			
04737Eh			
04737Fh			
047380h	Gateway Bit Search Support Register 0	GBSR0	0000h
047381h			
047382h	Gateway Bit Search Status Register 0	GBSS0	1000 0000b
047383h	Gateway Bit Search Control Register 0	GBSC0	0000 0000b
047384h	Gateway Bit Search Support Register 1	GBSR1	0000h
047385h			
047386h	Gateway Bit Search Status Register 1	GBSS1	1000 0000b
047387h	Gateway Bit Search Control Register 1	GBSC1	0000 0000b
047388h			
047389h			
04738Ah			
04738Bh			
04738Ch			
04738Dh			
04738Eh			
04738Fh			
047390h to 0473FFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.34 SFR List (34)

Address	Register	Symbol	Reset Value
047400h	CAN5 Mailbox 0: Message Identifier	C5MB0	XXXX XXXXh
047401h			
047402h			
047403h			
047404h			
047405h	CAN5 Mailbox 0: Data Length		XXh
047406h	CAN5 Mailbox 0: Data Field		XXXX XXXX XXXX XXXXh
047407h			
047408h			
047409h			
04740Ah			
04740Bh			
04740Ch			
04740Dh			
04740Eh	CAN5 Mailbox 0: Time Stamp		XXXXh
04740Fh			
047410h	CAN5 Mailbox 1: Message Identifier	C5MB1	XXXX XXXXh
047411h			
047412h			
047413h			
047414h			
047415h	CAN5 Mailbox 1: Data Length		XXh
047416h	CAN5 Mailbox 1: Data Field		XXXX XXXX XXXX XXXXh
047417h			
047418h			
047419h			
04741Ah			
04741Bh			
04741Ch			
04741Dh			
04741Eh	CAN5 Mailbox 1: Time Stamp		XXXXh
04741Fh			
047420h	CAN5 Mailbox 2: Message Identifier	C5MB2	XXXX XXXXh
047421h			
047422h			
047423h			
047424h			
047425h	CAN5 Mailbox 2: Data Length		XXh
047426h	CAN5 Mailbox 2: Data Field		XXXX XXXX XXXX XXXXh
047427h			
047428h			
047429h			
04742Ah			
04742Bh			
04742Ch			
04742Dh			
04742Eh	CAN5 Mailbox 2: Time Stamp		XXXXh
04742Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.35 SFR List (35)

Address	Register	Symbol	Reset Value
047430h	CAN5 Mailbox 3: Message Identifier	C5MB3	XXXX XXXXh
047431h			
047432h			
047433h			
047434h			
047435h	CAN5 Mailbox 3: Data Length		XXh
047436h	CAN5 Mailbox 3: Data Field		XXXX XXXX XXXX XXXXh
047437h			
047438h			
047439h			
04743Ah			
04743Bh			
04743Ch			
04743Dh			
04743Eh	CAN5 Mailbox 3: Time Stamp		XXXXh
04743Fh			
047440h	CAN5 Mailbox 4: Message Identifier	C5MB4	XXXX XXXXh
047441h			
047442h			
047443h			
047444h			
047445h	CAN5 Mailbox 4: Data Length		XXh
047446h	CAN5 Mailbox 4: Data Field		XXXX XXXX XXXX XXXXh
047447h			
047448h			
047449h			
04744Ah			
04744Bh			
04744Ch			
04744Dh			
04744Eh	CAN5 Mailbox 4: Time Stamp		XXXXh
04744Fh			
047450h	CAN5 Mailbox 5: Message Identifier	C5MB5	XXXX XXXXh
047451h			
047452h			
047453h			
047454h			
047455h	CAN5 Mailbox 5: Data Length		XXh
047456h	CAN5 Mailbox 5: Data Field		XXXX XXXX XXXX XXXXh
047457h			
047458h			
047459h			
04745Ah			
04745Bh			
04745Ch			
04745Dh			
04745Eh	CAN5 Mailbox 5: Time Stamp		XXXXh
04745Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.36 SFR List (36)

Address	Register	Symbol	Reset Value
047460h	CAN5 Mailbox 6: Message Identifier	C5MB6	XXXX XXXXh
047461h			
047462h			
047463h			
047464h			
047465h	CAN5 Mailbox 6: Data Length		XXh
047466h	CAN5 Mailbox 6: Data Field		XXXX XXXX XXXX XXXXh
047467h			
047468h			
047469h			
04746Ah			
04746Bh			
04746Ch			
04746Dh			
04746Eh	CAN5 Mailbox 6: Time Stamp		XXXXh
04746Fh			
047470h	CAN5 Mailbox 7: Message Identifier	C5MB7	XXXX XXXXh
047471h			
047472h			
047473h			
047474h			
047475h	CAN5 Mailbox 7: Data Length		XXh
047476h	CAN5 Mailbox 7: Data Field		XXXX XXXX XXXX XXXXh
047477h			
047478h			
047479h			
04747Ah			
04747Bh			
04747Ch			
04747Dh			
04747Eh	CAN5 Mailbox 7: Time Stamp		XXXXh
04747Fh			
047480h	CAN5 Mailbox 8: Message Identifier	C5MB8	XXXX XXXXh
047481h			
047482h			
047483h			
047484h			
047485h	CAN5 Mailbox 8: Data Length		XXh
047486h	CAN5 Mailbox 8: Data Field		XXXX XXXX XXXX XXXXh
047487h			
047488h			
047489h			
04748Ah			
04748Bh			
04748Ch			
04748Dh			
04748Eh	CAN5 Mailbox 8: Time Stamp		XXXXh
04748Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.37 SFR List (37)

Address	Register	Symbol	Reset Value
047490h	CAN5 Mailbox 9: Message Identifier	C5MB9	XXXX XXXXh
047491h			
047492h			
047493h			
047494h			
047495h	CAN5 Mailbox 9: Data Length		XXh
047496h	CAN5 Mailbox 9 Data Field		XXXX XXXX XXXX XXXXh
047497h			
047498h			
047499h			
04749Ah			
04749Bh			
04749Ch			
04749Dh			
04749Eh	CAN5 Mailbox 9: Time Stamp		XXXXh
04749Fh			
0474A0h	CAN5 Mailbox 10: Message Identifier	C5MB10	XXXX XXXXh
0474A1h			
0474A2h			
0474A3h			
0474A4h			
0474A5h	CAN5 Mailbox 10: Data Length		XXh
0474A6h	CAN5 Mailbox 10: Data Field		XXXX XXXX XXXX XXXXh
0474A7h			
0474A8h			
0474A9h			
0474AAh			
0474ABh			
0474ACh			
0474ADh			
0474AEh	CAN5 Mailbox 10: Time Stamp		XXXXh
0474AFh			
0474B0h	CAN5 Mailbox 11: Message Identifier	C5MB11	XXXX XXXXh
0474B1h			
0474B2h			
0474B3h			
0474B4h			
0474B5h	CAN5 Mailbox 11: Data Length		XXh
0474B6h	CAN5 Mailbox 11: Data Field		XXXX XXXX XXXX XXXXh
0474B7h			
0474B8h			
0474B9h			
0474BAh			
0474BBh			
0474BCh			
0474BDh			
0474BEh	CAN5 Mailbox 11: Time Stamp		XXXXh
0474BFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.38 SFR List (38)

Address	Register	Symbol	Reset Value
0474C0h	CAN5 Mailbox 12: Message Identifier	C5MB12	XXXX XXXXh
0474C1h			
0474C2h			
0474C3h			
0474C4h			
0474C5h	CAN5 Mailbox 12 Data Length		XXh
0474C6h	CAN5 Mailbox 12: Data Field		XXXX XXXX XXXX XXXXh
0474C7h			
0474C8h			
0474C9h			
0474CAh			
0474CBh			
0474CCh			
0474CDh			
0474CEh	CAN5 Mailbox 12: Time Stamp		XXXXh
0474CFh			
0474D0h	CAN5 Mailbox 13: Message Identifier	C5MB13	XXXX XXXXh
0474D1h			
0474D2h			
0474D3h			
0474D4h			
0474D5h	CAN5 Mailbox 13: Data Length		XXh
0474D6h	CAN5 Mailbox 13: Data Field		XXXX XXXX XXXX XXXXh
0474D7h			
0474D8h			
0474D9h			
0474DAh			
0474DBh			
0474DCh			
0474DDh			
0474DEh	CAN5 Mailbox 13: Time Stamp		XXXXh
0474DFh			
0474E0h	CAN5 Mailbox 14: Message Identifier	C5MB14	XXXX XXXXh
0474E1h			
0474E2h			
0474E3h			
0474E4h			
0474E5h	CAN5 Mailbox 14: Data Length		XXh
0474E6h	CAN5 Mailbox 14: Data Field		XXXX XXXX XXXX XXXXh
0474E7h			
0474E8h			
0474E9h			
0474EAh			
0474EBh			
0474ECh			
0474EDh			
0474EEh	CAN5 Mailbox 14: Time Stamp		XXXXh
0474EFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.39 SFR List (39)

Address	Register	Symbol	Reset Value
0474F0h	CAN5 Mailbox 15: Message Identifier	C5MB15	XXXX XXXXh
0474F1h			
0474F2h			
0474F3h			
0474F4h			
0474F5h	CAN5 Mailbox 15: Data Length		XXh
0474F6h	CAN5 Mailbox 15: Data Field		XXXX XXXX XXXX XXXXh
0474F7h			
0474F8h			
0474F9h			
0474FAh			
0474FBh			
0474FCh			
0474FDh			
0474FEh	CAN5 Mailbox 15: Time Stamp		XXXXh
0474FFh			
047500h to 04750Fh			
047510h	CAN5 Mask Register 0	C5MKR0	XXXX XXXXh
047511h			
047512h			
047513h			
047514h	CAN5 Mask Register 1	C5MKR1	XXXX XXXXh
047515h			
047516h			
047517h			
047518h	CAN5 Mask Register 2	C5MKR2	XXXX XXXXh
047519h			
04751Ah			
04751Bh			
04751Ch	CAN5 Mask Register 3	C5MKR3	XXXX XXXXh
04751Dh			
04751Eh			
04751Fh			
047520h	CAN5 FIFO Receive ID Compare Register 0	C5FIDCR0	XXXX XXXXh
047521h			
047522h			
047523h			
047524h	CAN5 FIFO Receive ID Compare Register 1	C5FIDCR1	XXXX XXXXh
047525h			
047526h			
047527h			
047528h			
047529h			
04752Ah	CAN5 Mask Invalid Register	C5MKIVLR	XXXXh
04752Bh			
04752Ch			
04752Dh			
04752Eh	CAN5 Mailbox Interrupt Enable Register	C5MIER	XXXXh
04752Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.40 SFR List (40)

Address	Register	Symbol	Reset Value
047530h	CAN5 Message Control Register 0	C5MCTL0	00h
047531h	CAN5 Message Control Register 1	C5MCTL1	00h
047532h	CAN5 Message Control Register 2	C5MCTL2	00h
047533h	CAN5 Message Control Register 3	C5MCTL3	00h
047534h	CAN5 Message Control Register 4	C5MCTL4	00h
047535h	CAN5 Message Control Register 5	C5MCTL5	00h
047536h	CAN5 Message Control Register 6	C5MCTL6	00h
047537h	CAN5 Message Control Register 7	C5MCTL7	00h
047538h	CAN5 Message Control Register 8	C5MCTL8	00h
047539h	CAN5 Message Control Register 9	C5MCTL9	00h
04753Ah	CAN5 Message Control Register 10	C5MCTL10	00h
04753Bh	CAN5 Message Control Register 11	C5MCTL11	00h
04753Ch	CAN5 Message Control Register 12	C5MCTL12	00h
04753Dh	CAN5 Message Control Register 13	C5MCTL13	00h
04753Eh	CAN5 Message Control Register 14	C5MCTL14	00h
04753Fh	CAN5 Message Control Register 15	C5MCTL15	00h
047540h	CAN5 Control Register	C5CTLR	0000 0101b
047541h			0000 0000b
047542h	CAN5 Status Register	C5STR	0000 0101b
047543h			0000 0000b
047544h	CAN5 Bit Configuration Register	C5BCR	00 0000h
047545h			
047546h			
047547h	CAN5 Clock Select Register	C5CLKR	000X 0000b
047548h	CAN5 Receive FIFO Control Register	C5RFCR	1000 0000b
047549h	CAN5 Receive FIFO Pointer Control Register	C5RFPCR	XXh
04754Ah	CAN5 Transmit FIFO Control Register	C5TFCR	1000 0000b
04754Bh	CAN5 Transmit FIFO Pointer Control Register	C5TFPCR	XXh
04754Ch	CAN5 Error Interrupt Enable Register	C5EIER	00h
04754Dh	CAN5 Error Interrupt Factor Judge Register	C5EIFR	00h
04754Eh	CAN5 Receive Error Count Register	C5RECR	00h
04754Fh	CAN5 Transmit Error Count Register	C5TECR	00h
047550h	CAN5 Error Code Store Register	C5ECSR	00h
047551h	CAN5 Channel Search Support Register	C5CSSR	XXh
047552h	CAN5 Mailbox Search Status Register	C5MSSR	1000 0000b
047553h	CAN5 Mailbox Search Mode Register	C5MSMR	0000 0000b
047554h	CAN5 Time Stamp Register	C5TSR	0000h
047555h			
047556h	CAN5 Acceptance Filter Support Register	C5AFSR	XXXXh
047557h			
047558h	CAN5 Test Control Register	C5TCR	00h
047559h			
04755Ah			
04755Bh			
04755Ch to 0475FFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.41 SFR List (41)

Address	Register	Symbol	Reset Value
047600h	CAN4 Mailbox 0: Message Identifier	C4MB0	XXXX XXXXh
047601h			
047602h			
047603h			
047604h			
047605h	CAN4 Mailbox 0: Data Length		XXh
047606h	CAN4 Mailbox 0: Data Field		XXXX XXXX XXXX XXXXh
047607h			
047608h			
047609h			
04760Ah			
04760Bh			
04760Ch			
04760Dh			
04760Eh	CAN4 Mailbox 0: Time Stamp		XXXXh
04760Fh			
047610h	CAN4 Mailbox 1: Message Identifier	C4MB1	XXXX XXXXh
047611h			
047612h			
047613h			
047614h			
047615h	CAN4 Mailbox 1: Data Length		XXh
047616h	CAN4 Mailbox 1: Data Field		XXXX XXXX XXXX XXXXh
047617h			
047618h			
047619h			
04761Ah			
04761Bh			
04761Ch			
04761Dh			
04761Eh	CAN4 Mailbox 1: Time Stamp		XXXXh
04761Fh			
047620h	CAN4 Mailbox 2: Message Identifier	C4MB2	XXXX XXXXh
047621h			
047622h			
047623h			
047624h			
047625h	CAN4 Mailbox 2: Data Length		XXh
047626h	CAN4 Mailbox 2: Data Field		XXXX XXXX XXXX XXXXh
047627h			
047628h			
047629h			
04762Ah			
04762Bh			
04762Ch			
04762Dh			
04762Eh	CAN4 Mailbox 2: Time Stamp		XXXXh
04762Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.42 SFR List (42)

Address	Register	Symbol	Reset Value
047630h	CAN4 Mailbox 3: Message Identifier	C4MB3	XXXX XXXXh
047631h			
047632h			
047633h			
047634h			
047635h	CAN4 Mailbox 3: Data Length		XXh
047636h	CAN4 Mailbox 3: Data Field		XXXX XXXX XXXX XXXXh
047637h			
047638h			
047639h			
04763Ah			
04763Bh			
04763Ch			
04763Dh			
04763Eh	CAN4 Mailbox 3: Time Stamp		XXXXh
04763Fh			
047640h	CAN4 Mailbox 4: Message Identifier	C4MB4	XXXX XXXXh
047641h			
047642h			
047643h			
047644h			
047645h	CAN4 Mailbox 4: Data Length		XXh
047646h	CAN4 Mailbox 4: Data Field		XXXX XXXX XXXX XXXXh
047647h			
047648h			
047649h			
04764Ah			
04764Bh			
04764Ch			
04764Dh			
04764Eh	CAN4 Mailbox 4: Time Stamp		XXXXh
04764Fh			
047650h	CAN4 Mailbox 5: Message Identifier	C4MB5	XXXX XXXXh
047651h			
047652h			
047653h			
047654h			
047655h	CAN4 Mailbox 5: Data Length		XXh
047656h	CAN4 Mailbox 5: Data Field		XXXX XXXX XXXX XXXXh
047657h			
047658h			
047659h			
04765Ah			
04765Bh			
04765Ch			
04765Dh			
04765Eh	CAN4 Mailbox 5: Time Stamp		XXXXh
04765Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.43 SFR List (43)

Address	Register	Symbol	Reset Value
047660h	CAN4 Mailbox 6: Message Identifier	C4MB6	XXXX XXXXh
047661h			
047662h			
047663h			
047664h			
047665h	CAN4 Mailbox 6: Data Length		XXh
047666h	CAN4 Mailbox 6: Data Field		XXXX XXXX XXXX XXXXh
047667h			
047668h			
047669h			
04766Ah			
04766Bh			
04766Ch			
04766Dh			
04766Eh	CAN4 Mailbox 6: Time Stamp		XXXXh
04766Fh			
047670h	CAN4 Mailbox 7: Message Identifier	C4MB7	XXXX XXXXh
047671h			
047672h			
047673h			
047674h			
047675h	CAN4 Mailbox 7: Data Length		XXh
047676h	CAN4 Mailbox 7: Data Field		XXXX XXXX XXXX XXXXh
047677h			
047678h			
047679h			
04767Ah			
04767Bh			
04767Ch			
04767Dh			
04767Eh	CAN4 Mailbox 7: Time Stamp		XXXXh
04767Fh			
047680h	CAN4 Mailbox 8: Message Identifier	C4MB8	XXXX XXXXh
047681h			
047682h			
047683h			
047684h			
047685h	CAN4 Mailbox 8: Data Length		XXh
047686h	CAN4 Mailbox 8: Data Field		XXXX XXXX XXXX XXXXh
047687h			
047688h			
047689h			
04768Ah			
04768Bh			
04768Ch			
04768Dh			
04768Eh	CAN4 Mailbox 8:Time Stamp		XXXXh
04768Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.44 SFR List (44)

Address	Register	Symbol	Reset Value
047690h	CAN4 Mailbox 9: Message Identifier	C4MB9	XXXX XXXXh
047691h			
047692h			
047693h			
047694h			
047695h	CAN4 Mailbox 9: Data Length		XXh
047696h	CAN4 Mailbox 9: Data Field		XXXX XXXX XXXX XXXXh
047697h			
047698h			
047699h			
04769Ah			
04769Bh			
04769Ch			
04769Dh			
04769Eh	CAN4 Mailbox 9: Time Stamp		XXXXh
04769Fh			
0476A0h	CAN4 Mailbox 10: Message Identifier	C4MB10	XXXX XXXXh
0476A1h			
0476A2h			
0476A3h			
0476A4h			
0476A5h	CAN4 Mailbox 10: Data Length		XXh
0476A6h	CAN4 Mailbox 10: Data Field		XXXX XXXX XXXX XXXXh
0476A7h			
0476A8h			
0476A9h			
0476AAh			
0476ABh			
0476ACh			
0476ADh			
0476AEh	CAN4 Mailbox 10: Time Stamp		XXXXh
0476AFh			
0476B0h	CAN4 Mailbox 11: Message Identifier	C4MB11	XXXX XXXXh
0476B1h			
0476B2h			
0476B3h			
0476B4h			
0476B5h	CAN4 Mailbox 11: Data Length		XXh
0476B6h	CAN4 Mailbox 11: Data Field		XXXX XXXX XXXX XXXXh
0476B7h			
0476B8h			
0476B9h			
0476BAh			
0476BBh			
0476BCh			
0476BDh			
0476BEh	CAN4 Mailbox 11: Time Stamp		XXXXh
0476BFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.45 SFR List (45)

Address	Register	Symbol	Reset Value
0476C0h	CAN4 Mailbox 12: Message Identifier	C4MB12	XXXX XXXXh
0476C1h			
0476C2h			
0476C3h			
0476C4h			
0476C5h	CAN4 Mailbox 12: Data Length		XXh
0476C6h	CAN4 Mailbox 12: Data Field		XXXX XXXX XXXX XXXXh
0476C7h			
0476C8h			
0476C9h			
0476CAh			
0476CBh			
0476CCh			
0476CDh			
0476CEh	CAN4 Mailbox 12: Time Stamp		XXXXh
0476CFh			
0476D0h	CAN4 Mailbox 13: Message Identifier	C4MB13	XXXX XXXXh
0476D1h			
0476D2h			
0476D3h			
0476D4h			
0476D5h	CAN4 Mailbox 13: Data Length		XXh
0476D6h	CAN4 Mailbox 13: Data Field		XXXX XXXX XXXX XXXXh
0476D7h			
0476D8h			
0476D9h			
0476DAh			
0476DBh			
0476DCh			
0476DDh			
0476DEh	CAN4 Mailbox 13: Time Stamp		XXXXh
0476DFh			
0476E0h	CAN4 Mailbox 14: Message Identifier	C4MB14	XXXX XXXXh
0476E1h			
0476E2h			
0476E3h			
0476E4h			
0476E5h	CAN4 Mailbox 14: Data Length		XXh
0476E6h	CAN4 Mailbox 14: Data Field		XXXX XXXX XXXX XXXXh
0476E7h			
0476E8h			
0476E9h			
0476EAh			
0476EBh			
0476ECh			
0476EDh			
0476EEh	CAN4 Mailbox 14: Time Stamp		XXXXh
0476EFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.46 SFR List (46)

Address	Register	Symbol	Reset Value
0476F0h	CAN4 Mailbox 15: Message Identifier	C4MB15	XXXX XXXXh
0476F1h			
0476F2h			
0476F3h			
0476F4h			
0476F5h	CAN4 Mailbox 15: Data Length		XXh
0476F6h	CAN4 Mailbox 15: Data Field		XXXX XXXX XXXX XXXXh
0476F7h			
0476F8h			
0476F9h			
0476FAh			
0476FBh			
0476FCh			
0476FDh			
0476FEh	CAN4 Mailbox 15: Time Stamp		XXXXh
0476FFh			
047700h to 04770Fh			
047710h	CAN4 Mask Register 0	C4MKR0	XXXX XXXXh
047711h			
047712h			
047713h			
047714h	CAN4 Mask Register 1	C4MKR1	XXXX XXXXh
047715h			
047716h			
047717h			
047718h	CAN4 Mask Register 2	C4MKR2	XXXX XXXXh
047719h			
04771Ah			
04771Bh			
04771Ch	CAN4 Mask Register 3	C4MKR3	XXXX XXXXh
04771Dh			
04771Eh			
04771Fh			
047720h	CAN4 FIFO Receive ID Compare Register 0	C4FIDCR0	XXXX XXXXh
047721h			
047722h			
047723h			
047724h	CAN4 FIFO Receive ID Compare Register 1	C4FIDCR1	XXXX XXXXh
047725h			
047726h			
047727h			
047728h			
047729h			
04772Ah	CAN4 Mask Invalid Register	C4MKIVLR	XXXXh
04772Bh			
04772Ch			
04772Dh			
04772Eh	CAN4 Mailbox Interrupt Enable Register	C4MIER	XXXXh
04772Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.47 SFR List (47)

Address	Register	Symbol	Reset Value
047730h	CAN4 Message Control Register 0	C4MCTL0	00h
047731h	CAN4 Message Control Register 1	C4MCTL1	00h
047732h	CAN4 Message Control Register 2	C4MCTL2	00h
047733h	CAN4 Message Control Register 3	C4MCTL3	00h
047734h	CAN4 Message Control Register 4	C4MCTL4	00h
047735h	CAN4 Message Control Register 5	C4MCTL5	00h
047736h	CAN4 Message Control Register 6	C4MCTL6	00h
047737h	CAN4 Message Control Register 7	C4MCTL7	00h
047738h	CAN4 Message Control Register 8	C4MCTL8	00h
047739h	CAN4 Message Control Register 9	C4MCTL9	00h
04773Ah	CAN4 Message Control Register 10	C4MCTL10	00h
04773Bh	CAN4 Message Control Register 11	C4MCTL11	00h
04773Ch	CAN4 Message Control Register 12	C4MCTL12	00h
04773Dh	CAN4 Message Control Register 13	C4MCTL13	00h
04773Eh	CAN4 Message Control Register 14	C4MCTL14	00h
04773Fh	CAN4 Message Control Register 15	C4MCTL15	00h
047740h	CAN4 Control Register	C4CTLR	0000 0101b
047741h			0000 0000b
047742h	CAN4 Status Register	C4STR	0000 0101b
047743h			0000 0000b
047744h	CAN4 Bit Configuration Register	C4BCR	00 0000h
047745h			
047746h			
047747h	CAN4 Clock Select Register	C4CLKR	000X 0000b
047748h	CAN4 Receive FIFO Control Register	C4RFCR	1000 0000b
047749h	CAN4 Receive FIFO Pointer Control Register	C4RFPCR	XXh
04774Ah	CAN4 Transmit FIFO Control Register	C4TFCR	1000 0000b
04774Bh	CAN4 Transmit FIFO Pointer Control Register	C4TFPCR	XXh
04774Ch	CAN4 Error Interrupt Enable Register	C4EIER	00h
04774Dh	CAN4 Error Interrupt Factor Judge Register	C4EIFR	00h
04774Eh	CAN4 Receive Error Count Register	C4RECR	00h
04774Fh	CAN4 Transmit Error Count Register	C4TECR	00h
047750h	CAN4 Error Code Store Register	C4ECSR	00h
047751h	CAN4 Channel Search Support Register	C4CSSR	XXh
047752h	CAN4 Mailbox Search Status Register	C4MSSR	1000 0000b
047753h	CAN4 Mailbox Search Mode Register	C4MSMR	0000 0000b
047754h	CAN4 Time Stamp Register	C4TSR	0000h
047755h			
047756h	CAN4 Acceptance Filter Support Register	C4AFSR	XXXXh
047757h			
047758h	CAN4 Test Control Register	C4TCR	00h
047759h			
04775Ah			
04775Bh			
04775Ch to 0477FFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.48 SFR List (48)

Address	Register	Symbol	Reset Value
047800h	CAN3 Mailbox 0: Message Identifier	C3MB0	XXXX XXXXh
047801h			
047802h			
047803h			
047804h			
047805h	CAN3 Mailbox 0: Data Length		XXh
047806h	CAN3 Mailbox 0: Data Field		XXXX XXXX XXXX XXXXh
047807h			
047808h			
047809h			
04780Ah			
04780Bh			
04780Ch			
04780Dh			
04780Eh	CAN3 Mailbox 0: Time Stamp		XXXXh
04780Fh			
047810h	CAN3 Mailbox 1: Message Identifier	C3MB1	XXXX XXXXh
047811h			
047812h			
047813h			
047814h			
047815h	CAN3 Mailbox 1: Data Length		XXh
047816h	CAN3 Mailbox 1: Data Field		XXXX XXXX XXXX XXXXh
047817h			
047818h			
047819h			
04781Ah			
04781Bh			
04781Ch			
04781Dh			
04781Eh	CAN3 Mailbox 1: Time Stamp		XXXXh
04781Fh			
047820h	CAN3 Mailbox 2: Message Identifier	C3MB2	XXXX XXXXh
047821h			
047822h			
047823h			
047824h			
047825h	CAN3 Mailbox 2: Data Length		XXh
047826h	CAN3 Mailbox 2: Data Field		XXXX XXXX XXXX XXXXh
047827h			
047828h			
047829h			
04782Ah			
04782Bh			
04782Ch			
04782Dh			
04782Eh	CAN3 Mailbox 2: Time Stamp		XXXXh
04782Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.49 SFR List (49)

Address	Register	Symbol	Reset Value
047830h	CAN3 Mailbox 3: Message Identifier	C3MB3	XXXX XXXXh
047831h			
047832h			
047833h			
047834h			
047835h	CAN3 Mailbox 3: Data Length		XXh
047836h	CAN3 Mailbox 3: Data Field		XXXX XXXX XXXX XXXXh
047837h			
047838h			
047839h			
04783Ah			
04783Bh			
04783Ch			
04783Dh			
04783Eh	CAN3 Mailbox 3: Time Stamp		XXXXh
04783Fh			
047840h	CAN3 Mailbox 4: Message Identifier	C3MB4	XXXX XXXXh
047841h			
047842h			
047843h			
047844h			
047845h	CAN3 Mailbox 4: Data Length		XXh
047846h	CAN3 Mailbox 4: Data Field		XXXX XXXX XXXX XXXXh
047847h			
047848h			
047849h			
04784Ah			
04784Bh			
04784Ch			
04784Dh			
04784Eh	CAN3 Mailbox 4: Time Stamp		XXXXh
04784Fh			
047850h	CAN3 Mailbox 5: Message Identifier	C3MB5	XXXX XXXXh
047851h			
047852h			
047853h			
047854h			
047855h	CAN3 Mailbox 5: Data Length		XXh
047856h	CAN3 Mailbox 5: Data Field		XXXX XXXX XXXX XXXXh
047857h			
047858h			
047859h			
04785Ah			
04785Bh			
04785Ch			
04785Dh			
04785Eh	CAN3 Mailbox 5: Time Stamp		XXXXh
04785Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.50 SFR List (50)

Address	Register	Symbol	Reset Value
047860h	CAN3 Mailbox 6: Message Identifier	C3MB6	XXXX XXXXh
047861h			
047862h			
047863h			
047864h			
047865h	CAN3 Mailbox 6: Data Length		XXh
047866h	CAN3 Mailbox 6: Data Field		XXXX XXXX XXXX XXXXh
047867h			
047868h			
047869h			
04786Ah			
04786Bh			
04786Ch			
04786Dh			
04786Eh	CAN3 Mailbox 6: Time Stamp		XXXXh
04786Fh			
047870h	CAN3 Mailbox 7: Message Identifier	C3MB7	XXXX XXXXh
047871h			
047872h			
047873h			
047874h			
047875h	CAN3 Mailbox 7: Data Length		XXh
047876h	CAN3 Mailbox 7: Data Field		XXXX XXXX XXXX XXXXh
047877h			
047878h			
047879h			
04787Ah			
04787Bh			
04787Ch			
04787Dh			
04787Eh	CAN3 Mailbox 7: Time Stamp		XXXXh
04787Fh			
047880h	CAN3 Mailbox 8: Message Identifier	C3MB8	XXXX XXXXh
047881h			
047882h			
047883h			
047884h			
047885h	CAN3 Mailbox 8: Data Length		XXh
047886h	CAN3 Mailbox 8: Data Field		XXXX XXXX XXXX XXXXh
047887h			
047888h			
047889h			
04788Ah			
04788Bh			
04788Ch			
04788Dh			
04788Eh	CAN3 Mailbox 8:Time Stamp		XXXXh
04788Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.51 SFR List (51)

Address	Register	Symbol	Reset Value
047890h	CAN3 Mailbox 9: Message Identifier	C3MB9	XXXX XXXXh
047891h			
047892h			
047893h			
047894h			
047895h	CAN3 Mailbox 9: Data Length		XXh
047896h	CAN3 Mailbox 9: Data Field		XXXX XXXX XXXX XXXXh
047897h			
047898h			
047899h			
04789Ah			
04789Bh			
04789Ch			
04789Dh			
04789Eh	CAN3 Mailbox 9: Time Stamp		XXXXh
04789Fh			
0478A0h	CAN3 Mailbox 10: Message Identifier	C3MB10	XXXX XXXXh
0478A1h			
0478A2h			
0478A3h			
0478A4h			
0478A5h	CAN3 Mailbox 10: Data Length		XXh
0478A6h	CAN3 Mailbox 10: Data Field		XXXX XXXX XXXX XXXXh
0478A7h			
0478A8h			
0478A9h			
0478AAh			
0478ABh			
0478ACh			
0478ADh			
0478AEh	CAN3 Mailbox 10: Time Stamp		XXXXh
0478AFh			
0478B0h	CAN3 Mailbox 11: Message Identifier	C3MB11	XXXX XXXXh
0478B1h			
0478B2h			
0478B3h			
0478B4h			
0478B5h	CAN3 Mailbox 11: Data Length		XXh
0478B6h	CAN3 Mailbox 11: Data Field		XXXX XXXX XXXX XXXXh
0478B7h			
0478B8h			
0478B9h			
0478BAh			
0478BBh			
0478BCh			
0478BDh			
0478BEh	CAN3 Mailbox 11: Time Stamp		XXXXh
0478BFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.52 SFR List (52)

Address	Register	Symbol	Reset Value
0478C0h	CAN3 Mailbox 12: Message Identifier	C3MB12	XXXX XXXXh
0478C1h			
0478C2h			
0478C3h			
0478C4h			
0478C5h	CAN3 Mailbox 12: Data Length		XXh
0478C6h	CAN3 Mailbox 12: Data Field		XXXX XXXX XXXX XXXXh
0478C7h			
0478C8h			
0478C9h			
0478CAh			
0478CBh			
0478CCh			
0478CDh			
0478CEh	CAN3 Mailbox 12: Time Stamp		XXXXh
0478CFh			
0478D0h	CAN3 Mailbox 13: Message Identifier	C3MB13	XXXX XXXXh
0478D1h			
0478D2h			
0478D3h			
0478D4h			
0478D5h	CAN3 Mailbox 13: Data Length		XXh
0478D6h	CAN3 Mailbox 13: Data Field		XXXX XXXX XXXX XXXXh
0478D7h			
0478D8h			
0478D9h			
0478DAh			
0478DBh			
0478DCh			
0478DDh			
0478DEh	CAN3 Mailbox 13: Time Stamp		XXXXh
0478DFh			
0478E0h	CAN3 Mailbox 14: Message Identifier	C3MB14	XXXX XXXXh
0478E1h			
0478E2h			
0478E3h			
0478E4h			
0478E5h	CAN3 Mailbox 14: Data Length		XXh
0478E6h	CAN3 Mailbox 14: Data Field		XXXX XXXX XXXX XXXXh
0478E7h			
0478E8h			
0478E9h			
0478EAh			
0478EBh			
0478ECh			
0478EDh			
0478EEh	CAN3 Mailbox 14: Time Stamp		XXXXh
0478EFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.53 SFR List (53)

Address	Register	Symbol	Reset Value
0478F0h	CAN3 Mailbox 15: Message Identifier	C3MB15	XXXX XXXXh
0478F1h			
0478F2h			
0478F3h			
0478F4h			
0478F5h	CAN3 Mailbox 15: Data Length		XXh
0478F6h	CAN3 Mailbox 15: Data Field		XXXX XXXX XXXX XXXXh
0478F7h			
0478F8h			
0478F9h			
0478FAh			
0478FBh			
0478FCh			
0478FDh			
0478FEh	CAN3 Mailbox 15: Time Stamp		XXXXh
0478FFh			
047900h to 04790Fh			
047910h	CAN3 Mask Register 0	C3MKR0	XXXX XXXXh
047911h			
047912h			
047913h			
047914h	CAN3 Mask Register 1	C3MKR1	XXXX XXXXh
047915h			
047916h			
047917h			
047918h	CAN3 Mask Register 2	C3MKR2	XXXX XXXXh
047919h			
04791Ah			
04791Bh			
04791Ch	CAN3 Mask Register 3	C3MKR3	XXXX XXXXh
04791Dh			
04791Eh			
04791Fh			
047920h	CAN3 FIFO Receive ID Compare Register 0	C3FIDCR0	XXXX XXXXh
047921h			
047922h			
047923h			
047924h	CAN3 FIFO Receive ID Compare Register 1	C3FIDCR1	XXXX XXXXh
047925h			
047926h			
047927h			
047928h			
047929h			
04792Ah	CAN3 Mask Invalid Register	C3MKIVLR	XXXXh
04792Bh			
04792Ch			
04792Dh			
04792Eh	CAN3 Mailbox Interrupt Enable Register	C3MIER	XXXXh
04792Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.54 SFR List (54)

Address	Register	Symbol	Reset Value
047930h	CAN3 Message Control Register 0	C3MCTL0	00h
047931h	CAN3 Message Control Register 1	C3MCTL1	00h
047932h	CAN3 Message Control Register 2	C3MCTL2	00h
047933h	CAN3 Message Control Register 3	C3MCTL3	00h
047934h	CAN3 Message Control Register 4	C3MCTL4	00h
047935h	CAN3 Message Control Register 5	C3MCTL5	00h
047936h	CAN3 Message Control Register 6	C3MCTL6	00h
047937h	CAN3 Message Control Register 7	C3MCTL7	00h
047938h	CAN3 Message Control Register 8	C3MCTL8	00h
047939h	CAN3 Message Control Register 9	C3MCTL9	00h
04793Ah	CAN3 Message Control Register 10	C3MCTL10	00h
04793Bh	CAN3 Message Control Register 11	C3MCTL11	00h
04793Ch	CAN3 Message Control Register 12	C3MCTL12	00h
04793Dh	CAN3 Message Control Register 13	C3MCTL13	00h
04793Eh	CAN3 Message Control Register 14	C3MCTL14	00h
04793Fh	CAN3 Message Control Register 15	C3MCTL15	00h
047940h	CAN3 Control Register	C3CTLR	0000 0101b
047941h			0000 0000b
047942h	CAN3 Status Register	C3STR	0000 0101b
047943h			0000 0000b
047944h	CAN3 Bit Configuration Register	C3BCR	00 0000h
047945h			
047946h			
047947h	CAN3 Clock Select Register	C3CLKR	000X 0000b
047948h	CAN3 Receive FIFO Control Register	C3RFCR	1000 0000b
047949h	CAN3 Receive FIFO Pointer Control Register	C3RFPCR	XXh
04794Ah	CAN3 Transmit FIFO Control Register	C3TFCR	1000 0000b
04794Bh	CAN3 Transmit FIFO Pointer Control Register	C3TFPCR	XXh
04794Ch	CAN3 Error Interrupt Enable Register	C3EIER	00h
04794Dh	CAN3 Error Interrupt Factor Judge Register	C3EIFR	00h
04794Eh	CAN3 Receive Error Count Register	C3RECR	00h
04794Fh	CAN3 Transmit Error Count Register	C3TECR	00h
047950h	CAN3 Error Code Store Register	C3ECSR	00h
047951h	CAN3 Channel Search Support Register	C3CSSR	XXh
047952h	CAN3 Mailbox Search Status Register	C3MSSR	1000 0000b
047953h	CAN3 Mailbox Search Mode Register	C3MSMR	0000 0000b
047954h	CAN3 Time Stamp Register	C3TSR	0000h
047955h			
047956h	CAN3 Acceptance Filter Support Register	C3AFSR	XXXXh
047957h			
047958h	CAN3 Test Control Register	C3TCR	00h
047959h			
04795Ah			
04795Bh			
04795Ch to 0479FFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.55 SFR List (55)

Address	Register	Symbol	Reset Value
047A00h	CAN2 Mailbox 0: Message Identifier	C2MB0	XXXX XXXXh
047A01h			
047A02h			
047A03h			
047A04h			
047A05h	CAN2 Mailbox 0: Data Length		XXh
047A06h	CAN2 Mailbox 0: Data Field		XXXX XXXX XXXX XXXXh
047A07h			
047A08h			
047A09h			
047A0Ah			
047A0Bh			
047A0Ch			
047A0Dh			
047A0Eh	CAN2 Mailbox 0: Time Stamp		XXXXh
047A0Fh			
047A10h	CAN2 Mailbox 1: Message Identifier	C2MB1	XXXX XXXXh
047A11h			
047A12h			
047A13h			
047A14h			
047A15h	CAN2 Mailbox 1: Data Length		XXh
047A16h	CAN2 Mailbox 1: Data Field		XXXX XXXX XXXX XXXXh
047A17h			
047A18h			
047A19h			
047A1Ah			
047A1Bh			
047A1Ch			
047A1Dh			
047A1Eh	CAN2 Mailbox 1: Time Stamp		XXXXh
047A1Fh			
047A20h	CAN2 Mailbox 2: Message Identifier	C2MB2	XXXX XXXXh
047A21h			
047A22h			
047A23h			
047A24h			
047A25h	CAN2 Mailbox 2: Data Length		XXh
047A26h	CAN2 Mailbox 2: Data Field		XXXX XXXX XXXX XXXXh
047A27h			
047A28h			
047A29h			
047A2Ah			
047A2Bh			
047A2Ch			
047A2Dh			
047A2Eh	CAN2 Mailbox 2: Time Stamp		XXXXh
047A2Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.56 SFR List (56)

Address	Register	Symbol	Reset Value
047A30h	CAN2 Mailbox 3: Message Identifier	C2MB3	XXXX XXXXh
047A31h			
047A32h			
047A33h			
047A34h			
047A35h	CAN2 Mailbox 3: Data Length		XXh
047A36h	CAN2 Mailbox 3: Data Field		XXXX XXXX XXXX XXXXh
047A37h			
047A38h			
047A39h			
047A3Ah			
047A3Bh			
047A3Ch			
047A3Dh			
047A3Eh	CAN2 Mailbox 3: Time Stamp		XXXXh
047A3Fh			
047A40h	CAN2 Mailbox 4: Message Identifier	C2MB4	XXXX XXXXh
047A41h			
047A42h			
047A43h			
047A44h			
047A45h	CAN2 Mailbox 4: Data Length		XXh
047A46h	CAN2 Mailbox 4: Data Field		XXXX XXXX XXXX XXXXh
047A47h			
047A48h			
047A49h			
047A4Ah			
047A4Bh			
047A4Ch			
047A4Dh			
047A4Eh	CAN2 Mailbox 4: Time Stamp		XXXXh
047A4Fh			
047A50h	CAN2 Mailbox 5: Message Identifier	C2MB5	XXXX XXXXh
047A51h			
047A52h			
047A53h			
047A54h			
047A55h	CAN2 Mailbox 5: Data Length		XXh
047A56h	CAN2 Mailbox 5: Data Field		XXXX XXXX XXXX XXXXh
047A57h			
047A58h			
047A59h			
047A5Ah			
047A5Bh			
047A5Ch			
047A5Dh			
047A5Eh	CAN2 Mailbox 5: Time Stamp		XXXXh
047A5Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.57 SFR List (57)

Address	Register	Symbol	Reset Value
047A60h	CAN2 Mailbox 6: Message Identifier	C2MB6	XXXX XXXXh
047A61h			
047A62h			
047A63h			
047A64h			
047A65h	CAN2 Mailbox 6: Data Length		XXh
047A66h	CAN2 Mailbox 6: Data Field		XXXX XXXX XXXX XXXXh
047A67h			
047A68h			
047A69h			
047A6Ah			
047A6Bh			
047A6Ch			
047A6Dh			
047A6Eh	CAN2 Mailbox 6: Time Stamp		XXXXh
047A6Fh			
047A70h	CAN2 Mailbox 7: Message Identifier	C2MB7	XXXX XXXXh
047A71h			
047A72h			
047A73h			
047A74h			
047A75h	CAN2 Mailbox 7: Data Length		XXh
047A76h	CAN2 Mailbox 7: Data Field		XXXX XXXX XXXX XXXXh
047A77h			
047A78h			
047A79h			
047A7Ah			
047A7Bh			
047A7Ch			
047A7Dh			
047A7Eh	CAN2Mailbox 7: Time Stamp		XXXXh
047A7Fh			
047A80h	CAN2 Mailbox 8: Message Identifier	C2MB8	XXXX XXXXh
047A81h			
047A82h			
047A83h			
047A84h			
047A85h	CAN2 Mailbox 8: Data Length		XXh
047A86h	CAN2 Mailbox 8: Data Field		XXXX XXXX XXXX XXXXh
047A87h			
047A88h			
047A89h			
047A8Ah			
047A8Bh			
047A8Ch			
047A8Dh			
047A8Eh	CAN2 Mailbox 8: Time Stamp		XXXXh
047A8Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.58 SFR List (58)

Address	Register	Symbol	Reset Value
047A90h	CAN2 Mailbox 9: Message Identifier	C2MB9	XXXX XXXXh
047A91h			
047A92h			
047A93h			
047A94h			
047A95h	CAN2 Mailbox 9: Data Length		XXh
047A96h	CAN2 Mailbox 9 Data Field		XXXX XXXX XXXX XXXXh
047A97h			
047A98h			
047A99h			
047A9Ah			
047A9Bh			
047A9Ch			
047A9Dh			
047A9Eh	CAN2 Mailbox 9: Time Stamp		XXXXh
047A9Fh			
047AA0h	CAN2 Mailbox 10: Message Identifier	C2MB10	XXXX XXXXh
047AA1h			
047AA2h			
047AA3h			
047AA4h			
047AA5h	CAN2 Mailbox 10: Data Length		XXh
047AA6h	CAN2 Mailbox 10: Data Field		XXXX XXXX XXXX XXXXh
047AA7h			
047AA8h			
047AA9h			
047AAAh			
047AABh			
047AACh			
047AADh			
047AAEh	CAN5 Mailbox 10: Time Stamp		XXXXh
047AAFh			
047AB0h	CAN5 Mailbox 11: Message Identifier	C2MB11	XXXX XXXXh
047AB1h			
047AB2h			
047AB3h			
047AB4h			
047AB5h	CAN2 Mailbox 11: Data Length		XXh
047AB6h	CAN2 Mailbox 11: Data Field		XXXX XXXX XXXX XXXXh
047AB7h			
047AB8h			
047AB9h			
047ABAh			
047ABBh			
047ABCh			
047ABDh			
047ABEh	CAN2 Mailbox 11: Time Stamp		XXXXh
047ABFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.59 SFR List (59)

Address	Register	Symbol	Reset Value
047AC0h	CAN2 Mailbox 12: Message Identifier	C2MB12	XXXX XXXXh
047AC1h			
047AC2h			
047AC3h			
047AC4h			
047AC5h	CAN2 Mailbox 12 Data Length		XXh
047AC6h	CAN2 Mailbox 12: Data Field		XXXX XXXX XXXX XXXXh
047AC7h			
047AC8h			
047AC9h			
047ACAh			
047ACBh			
047ACCh			
047ACDh			
047ACEh	CAN2 Mailbox 12: Time Stamp		XXXXh
047ACFh			
047AD0h	CAN2 Mailbox 13: Message Identifier	C2MB13	XXXX XXXXh
047AD1h			
047AD2h			
047AD3h			
047AD4h			
047AD5h	CAN2 Mailbox 13: Data Length		XXh
047AD6h	CAN2 Mailbox 13: Data Field		XXXX XXXX XXXX XXXXh
047AD7h			
047AD8h			
047AD9h			
047ADAh			
047ADBh			
047ADCh			
047ADDh			
047ADEh	CAN2 Mailbox 13: Time Stamp		XXXXh
047ADFh			
047AE0h	CAN2 Mailbox 14: Message Identifier	C2MB14	XXXX XXXXh
047AE1h			
047AE2h			
047AE3h			
047AE4h			
047AE5h	CAN2 Mailbox 14: Data Length		XXh
047AE6h	CAN2 Mailbox 14: Data Field		XXXX XXXX XXXX XXXXh
047AE7h			
047AE8h			
047AE9h			
047AEAh			
047AEBh			
047AECh			
047AEDh			
047AEEh	CAN5 Mailbox 14: Time Stamp		XXXXh
047AEFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.60 SFR List (60)

Address	Register	Symbol	Reset Value
047AF0h	CAN2 Mailbox 15: Message Identifier	C2MB15	XXXX XXXXh
047AF1h			
047AF2h			
047AF3h			
047AF4h			
047AF5h	CAN2 Mailbox 15: Data Length		XXh
047AF6h	CAN2 Mailbox 15: Data Field		XXXX XXXX XXXX XXXXh
047AF7h			
047AF8h			
047AF9h			
047AFAh			
047AFBh			
047AFCh			
047AFDh			
047AFEh	CAN2 Mailbox 15: Time Stamp		XXXXh
047AFFh			
047B00h to 047B0Fh			
047B10h	CAN2 Mask Register 0	C2MKR0	XXXX XXXXh
047B11h			
047B12h			
047B13h			
047B14h	CAN2 Mask Register 1	C2MKR1	XXXX XXXXh
047B15h			
047B16h			
047B17h			
047B18h	CAN2 Mask Register 2	C2MKR2	XXXX XXXXh
047B19h			
047B1Ah			
047B1Bh			
047B1Ch	CAN2 Mask Register 3	C2MKR3	XXXX XXXXh
047B1Dh			
047B1Eh			
047B1Fh			
047B20h	CAN2 FIFO Receive ID Compare Register 0	C2FIDCR0	XXXX XXXXh
047B21h			
047B22h			
047B23h			
047B24h	CAN2 FIFO Receive ID Compare Register 1	C2FIDCR1	XXXX XXXXh
047B25h			
047B26h			
047B27h			
047B28h			
047B29h			
047B2Ah	CAN2 Mask Invalid Register	C2MKIVLR	XXXXh
047B2Bh			
047B2Ch			
047B2Dh			
047B2Eh	CAN2 Mailbox Interrupt Enable Register	C2MIER	XXXXh
047B2Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.61 SFR List (61)

Address	Register	Symbol	Reset Value
047B30h	CAN2 Message Control Register 0	C2MCTL0	00h
047B31h	CAN2 Message Control Register 1	C2MCTL1	00h
047B32h	CAN2 Message Control Register 2	C2MCTL2	00h
047B33h	CAN2 Message Control Register 3	C2MCTL3	00h
047B34h	CAN2 Message Control Register 4	C2MCTL4	00h
047B35h	CAN2 Message Control Register 5	C2MCTL5	00h
047B36h	CAN2 Message Control Register 6	C2MCTL6	00h
047B37h	CAN2 Message Control Register 7	C2MCTL7	00h
047B38h	CAN2 Message Control Register 8	C2MCTL8	00h
047B39h	CAN2 Message Control Register 9	C2MCTL9	00h
047B3Ah	CAN2 Message Control Register 10	C2MCTL10	00h
047B3Bh	CAN2 Message Control Register 11	C2MCTL11	00h
047B3Ch	CAN2 Message Control Register 12	C2MCTL12	00h
047B3Dh	CAN2 Message Control Register 13	C2MCTL13	00h
047B3Eh	CAN2 Message Control Register 14	C2MCTL14	00h
047B3Fh	CAN2 Message Control Register 15	C2MCTL15	00h
047B40h	CAN2 Control Register	C2CTLR	0000 0101b
047B41h			0000 0000b
047B42h	CAN2 Status Register	C2STR	0000 0101b
047B43h			0000 0000b
047B44h	CAN2 Bit Configuration Register	C2BCR	00 0000h
047B45h			
047B46h			
047B47h	CAN2 Clock Select Register	C2CLKR	000X 0000b
047B48h	CAN2 Receive FIFO Control Register	C2RFCR	1000 0000b
047B49h	CAN2 Receive FIFO Pointer Control Register	C2RFPCR	XXh
047B4Ah	CAN2 Transmit FIFO Control Register	C2TFCR	1000 0000b
047B4Bh	CAN2 Transmit FIFO Pointer Control Register	C2TFPCR	XXh
047B4Ch	CAN2 Error Interrupt Enable Register	C2EIER	00h
047B4Dh	CAN2 Error Interrupt Factor Judge Register	C2EIFR	00h
047B4Eh	CAN2 Receive Error Count Register	C2RECR	00h
047B4Fh	CAN2 Transmit Error Count Register	C2TECR	00h
047B50h	CAN2 Error Code Store Register	C2ECSR	00h
047B51h	CAN2 Channel Search Support Register	C2CSSR	XXh
047B52h	CAN2 Mailbox Search Status Register	C2MSSR	1000 0000b
047B53h	CAN2 Mailbox Search Mode Register	C2MSMR	0000 0000b
047B54h	CAN2 Time Stamp Register	C2TSR	0000h
047B55h			
047B56h	CAN2 Acceptance Filter Support Register	C2AFSR	XXXXh
047B57h			
047B58h	CAN2 Test Control Register	C2TCR	00h
047B59h			
047B5Ah			
047B5Bh			
047B5Ch to 047BFFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.62 SFR List (62)

Address	Register	Symbol	Reset Value
047C00h	CAN1 Mailbox 0: Message Identifier	C1MB0	XXXX XXXXh
047C01h			
047C02h			
047C03h			
047C04h			
047C05h	CAN1 Mailbox 0: Data Length		XXh
047C06h	CAN1 Mailbox 0: Data Field		XXXX XXXX XXXX XXXXh
047C07h			
047C08h			
047C09h			
047C0Ah			
047C0Bh			
047C0Ch			
047C0Dh			
047C0Eh	CAN1 Mailbox 0: Time Stamp		XXXXh
047C0Fh			
047C10h	CAN1 Mailbox 1: Message Identifier	C1MB1	XXXX XXXXh
047C11h			
047C12h			
047C13h			
047C14h			
047C15h	CAN1 Mailbox 1: Data Length		XXh
047C16h	CAN1 Mailbox 1: Data Field		XXXX XXXX XXXX XXXXh
047C17h			
047C18h			
047C19h			
047C1Ah			
047C1Bh			
047C1Ch			
047C1Dh			
047C1Eh	CAN1 Mailbox 1: Time Stamp		XXXXh
047C1Fh			
047C20h	CAN1 Mailbox 2: Message Identifier	C1MB2	XXXX XXXXh
047C21h			
047C22h			
047C23h			
047C24h			
047C25h	CAN1 Mailbox 2: Data Length		XXh
047C26h	CAN1 Mailbox 2: Data Field		XXXX XXXX XXXX XXXXh
047C27h			
047C28h			
047C29h			
047C2Ah			
047C2Bh			
047C2Ch			
047C2Dh			
047C2Eh	CAN1 Mailbox 2: Time Stamp		XXXXh
047C2Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.63 SFR List (63)

Address	Register	Symbol	Reset Value
047C30h	CAN1 Mailbox 3: Message Identifier	C1MB3	XXXX XXXXh
047C31h			
047C32h			
047C33h			
047C34h			
047C35h	CAN1 Mailbox 3: Data Length		XXh
047C36h	CAN1 Mailbox 3: Data Field		XXXX XXXX XXXX XXXXh
047C37h			
047C38h			
047C39h			
047C3Ah			
047C3Bh			
047C3Ch			
047C3Dh			
047C3Eh	CAN1 Mailbox 3: Time Stamp		XXXXh
047C3Fh			
047C40h	CAN1 Mailbox 4: Message Identifier	C1MB4	XXXX XXXXh
047C41h			
047C42h			
047C43h			
047C44h			
047C45h	CAN1 Mailbox 4: Data Length		XXh
047C46h	CAN1 Mailbox 4: Data Field		XXXX XXXX XXXX XXXXh
047C47h			
047C48h			
047C49h			
047C4Ah			
047C4Bh			
047C4Ch			
047C4Dh			
047C4Eh	CAN1 Mailbox 4: Time Stamp		XXXXh
047C4Fh			
047C50h	CAN1 Mailbox 5: Message Identifier	C1MB5	XXXX XXXXh
047C51h			
047C52h			
047C53h			
047C54h			
047C55h	CAN1 Mailbox 5: Data Length		XXh
047C56h	CAN1 Mailbox 5: Data Field		XXXX XXXX XXXX XXXXh
047C57h			
047C58h			
047C59h			
047C5Ah			
047C5Bh			
047C5Ch			
047C5Dh			
047C5Eh	CAN1 Mailbox 5: Time Stamp		XXXXh
047C5Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.64 SFR List (64)

Address	Register	Symbol	Reset Value
047C60h	CAN1 Mailbox 6: Message Identifier	C1MB6	XXXX XXXXh
047C61h			
047C62h			
047C63h			
047C64h			
047C65h	CAN1 Mailbox 6: Data Length		XXh
047C66h	CAN1 Mailbox 6: Data Field		XXXX XXXX XXXX XXXXh
047C67h			
047C68h			
047C69h			
047C6Ah			
047C6Bh			
047C6Ch			
047C6Dh			
047C6Eh	CAN1 Mailbox 6: Time Stamp		XXXXh
047C6Fh			
047C70h	CAN1 Mailbox 7: Message Identifier	C1MB7	XXXX XXXXh
047C71h			
047C72h			
047C73h			
047C74h			
047C75h	CAN1 Mailbox 7: Data Length		XXh
047C76h	CAN1 Mailbox 7: Data Field		XXXX XXXX XXXX XXXXh
047C77h			
047C78h			
047C79h			
047C7Ah			
047C7Bh			
047C7Ch			
047C7Dh			
047C7Eh	CAN1 Mailbox 7: Time Stamp		XXXXh
047C7Fh			
047C80h	CAN1 Mailbox 8: Message Identifier	C1MB8	XXXX XXXXh
047C81h			
047C82h			
047C83h			
047C84h			
047C85h	CAN1 Mailbox 8: Data Length		XXh
047C86h	CAN1 Mailbox 8: Data Field		XXXX XXXX XXXX XXXXh
047C87h			
047C88h			
047C89h			
047C8Ah			
047C8Bh			
047C8Ch			
047C8Dh			
047C8Eh	CAN1 Mailbox 8: Time Stamp		XXXXh
047C8Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.65 SFR List (65)

Address	Register	Symbol	Reset Value
047C90h	CAN1 Mailbox 9: Message Identifier	C1MB9	XXXX XXXXh
047C91h			
047C92h			
047C93h			
047C94h			
047C95h	CAN1 Mailbox 9: Data Length		XXh
047C96h	CAN1 Mailbox 9: Data Field		XXXX XXXX XXXX XXXXh
047C97h			
047C98h			
047C99h			
047C9Ah			
047C9Bh			
047C9Ch			
047C9Dh			
047C9Eh	CAN1 Mailbox 9: Time Stamp		XXXXh
047C9Fh			
047CA0h	CAN1 Mailbox 10: Message Identifier	C1MB10	XXXX XXXXh
047CA1h			
047CA2h			
047CA3h			
047CA4h			
047CA5h	CAN1 Mailbox 10: Data Length		XXh
047CA6h	CAN1 Mailbox 10: Data Field		XXXX XXXX XXXX XXXXh
047CA7h			
047CA8h			
047CA9h			
047CAAh			
047CABh			
047CACh			
047CADh			
047CAEh	CAN1 Mailbox 10: Time Stamp		XXXXh
047CAFh			
047CB0h	CAN1 Mailbox 11: Message Identifier	C1MB11	XXXX XXXXh
047CB1h			
047CB2h			
047CB3h			
047CB4h			
047CB5h	CAN1 Mailbox 11: Data Length		XXh
047CB6h	CAN1 Mailbox 11: Data Field		XXXX XXXX XXXX XXXXh
047CB7h			
047CB8h			
047CB9h			
047CBAh			
047CBBh			
047CBCh			
047CBDh			
047CBEh	CAN1 Mailbox 11: Time Stamp		XXXXh
047CBFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.66 SFR List (66)

Address	Register	Symbol	Reset Value
047CC0h	CAN1 Mailbox 12: Message Identifier	C1MB12	XXXX XXXXh
047CC1h			
047CC2h			
047CC3h			
047CC4h			
047CC5h	CAN1 Mailbox 12: Data Length		XXh
047CC6h	CAN1 Mailbox 12: Data Field		XXXX XXXX XXXX XXXXh
047CC7h			
047CC8h			
047CC9h			
047CCAh			
047CCBh			
047CCCh			
047CCDh			
047CCEh	CAN1 Mailbox 12: Time Stamp		XXXXh
047CCFh			
047CD0h	CAN1 Mailbox 13: Message Identifier	C1MB13	XXXX XXXXh
047CD1h			
047CD2h			
047CD3h			
047CD4h			
047CD5h	CAN1 Mailbox 13: Data Length		XXh
047CD6h	CAN1 Mailbox 13: Data Field		XXXX XXXX XXXX XXXXh
047CD7h			
047CD8h			
047CD9h			
047CDAh			
047CDBh			
047CDCh			
047CDDh			
047CDEh	CAN1 Mailbox 13: Time Stamp		XXXXh
047CDFh			
047CE0h	CAN1 Mailbox 14: Message Identifier	C1MB14	XXXX XXXXh
047CE1h			
047CE2h			
047CE3h			
047CE4h			
047CE5h	CAN1 Mailbox 14: Data Length		XXh
047CE6h	CAN1 Mailbox 14: Data Field		XXXX XXXX XXXX XXXXh
047CE7h			
047CE8h			
047CE9h			
047CEAh			
047CEBh			
047CECh			
047CEDh			
047CEEh	CAN1 Mailbox 14: Time Stamp		XXXXh
047CEFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.67 SFR List (67)

Address	Register	Symbol	Reset Value
047CF0h	CAN1 Mailbox 15: Message Identifier	C1MB15	XXXX XXXXh
047CF1h			
047CF2h			
047CF3h			
047CF4h			
047CF5h	CAN1 Mailbox 15: Data Length		XXh
047CF6h	CAN1 Mailbox 15: Data Field		XXXX XXXX XXXX XXXXh
047CF7h			
047CF8h			
047CF9h			
047CFAh			
047CFBh			
047CFCh			
047CFDh			
047CFEh	CAN1 Mailbox 15: Time Stamp		XXXXh
047CFFh			
047D00h to 047D0Fh			
047D10h	CAN1 Mask Register 0	C1MKR0	XXXX XXXXh
047D11h			
047D12h			
047D13h			
047D14h	CAN1 Mask Register 1	C1MKR1	XXXX XXXXh
047D15h			
047D16h			
047D17h			
047D18h	CAN1 Mask Register 2	C1MKR2	XXXX XXXXh
047D19h			
047D1Ah			
047D1Bh			
047D1Ch	CAN1 Mask Register 3	C1MKR3	XXXX XXXXh
047D1Dh			
047D1Eh			
047D1Fh			
047D20h	CAN1 FIFO Receive ID Compare Register 0	C1FIDCR0	XXXX XXXXh
047D21h			
047D22h			
047D23h			
047D24h	CAN1 FIFO Receive ID Compare Register 1	C1FIDCR1	XXXX XXXXh
047D25h			
047D26h			
047D27h			
047D28h			
047D29h			
047D2Ah	CAN1 Mask Invalid Register	C1MKIVLR	XXXXh
047D2Bh			
047D2Ch			
047D2Dh			
047D2Eh	CAN1 Mailbox Interrupt Enable Register	C1MIER	XXXXh
047D2Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.68 SFR List (68)

Address	Register	Symbol	Reset Value
047D30h	CAN1 Message Control Register 0	C1MCTL0	00h
047D31h	CAN1 Message Control Register 1	C1MCTL1	00h
047D32h	CAN1 Message Control Register 2	C1MCTL2	00h
047D33h	CAN1 Message Control Register 3	C1MCTL3	00h
047D34h	CAN1 Message Control Register 4	C1MCTL4	00h
047D35h	CAN1 Message Control Register 5	C1MCTL5	00h
047D36h	CAN1 Message Control Register 6	C1MCTL6	00h
047D37h	CAN1 Message Control Register 7	C1MCTL7	00h
047D38h	CAN1 Message Control Register 8	C1MCTL8	00h
047D39h	CAN1 Message Control Register 9	C1MCTL9	00h
047D3Ah	CAN1 Message Control Register 10	C1MCTL10	00h
047D3Bh	CAN1 Message Control Register 11	C1MCTL11	00h
047D3Ch	CAN1 Message Control Register 12	C1MCTL12	00h
047D3Dh	CAN1 Message Control Register 13	C1MCTL13	00h
047D3Eh	CAN1 Message Control Register 14	C1MCTL14	00h
047D3Fh	CAN1 Message Control Register 15	C1MCTL15	00h
047D40h	CAN1 Control Register	C1CTLR	0000 0101b
047D41h			0000 0000b
047D42h	CAN1 Status Register	C1STR	0000 0101b
047D43h			0000 0000b
047D44h	CAN1 Bit Configuration Register	C1BCR	00 0000h
047D45h			
047D46h			
047D47h	CAN1 Clock Select Register	C1CLKR	000X 0000b
047D48h	CAN1 Receive FIFO Control Register	C1RFCR	1000 0000b
047D49h	CAN1 Receive FIFO Pointer Control Register	C1RFPCR	XXh
047D4Ah	CAN1 Transmit FIFO Control Register	C1TFCR	1000 0000b
047D4Bh	CAN1 Transmit FIFO Pointer Control Register	C1TFPCR	XXh
047D4Ch	CAN1 Error Interrupt Enable Register	C1EIER	00h
047D4Dh	CAN1 Error Interrupt Factor Judge Register	C1EIFR	00h
047D4Eh	CAN1 Receive Error Count Register	C1RECR	00h
047D4Fh	CAN1 Transmit Error Count Register	C1TECR	00h
047D50h	CAN1 Error Code Store Register	C1ECSR	00h
047D51h	CAN1 Channel Search Support Register	C1CSSR	XXh
047D52h	CAN1 Mailbox Search Status Register	C1MSSR	1000 0000b
047D53h	CAN1 Mailbox Search Mode Register	C1MSMR	0000 0000b
047D54h	CAN1 Time Stamp Register	C1TSR	0000h
047D55h			
047D56h	CAN1 Acceptance Filter Support Register	C1AFSR	XXXXh
047D57h			
047D58h	CAN1 Test Control Register	C1TCR	00h
047D59h			
047D5Ah			
047D5Bh			
047D5Ch to 047DFFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.69 SFR List (69)

Address	Register	Symbol	Reset Value
047E00h	CAN0 Mailbox 0: Message Identifier	C0MB0	XXXX XXXXh
047E01h			
047E02h			
047E03h			
047E04h			
047E05h	CAN0 Mailbox 0: Data Length		XXh
047E06h	CAN0 Mailbox 0: Data Field		XXXX XXXX XXXX XXXXh
047E07h			
047E08h			
047E09h			
047E0Ah			
047E0Bh			
047E0Ch			
047E0Dh			
047E0Eh	CAN0 Mailbox 0: Time Stamp		XXXXh
047E0Fh			
047E10h	CAN0 Mailbox 1: Message Identifier	C0MB1	XXXX XXXXh
047E11h			
047E12h			
047E13h			
047E14h			
047E15h	CAN0 Mailbox 1: Data Length		XXh
047E16h	CAN0 Mailbox 1: Data Field		XXXX XXXX XXXX XXXXh
047E17h			
047E18h			
047E19h			
047E1Ah			
047E1Bh			
047E1Ch			
047E1Dh			
047E1Eh	CAN0 Mailbox 1: Time Stamp		XXXXh
047E1Fh			
047E20h	CAN0 Mailbox 2: Message Identifier	C0MB2	XXXX XXXXh
047E21h			
047E22h			
047E23h			
047E24h			
047E25h	CAN0 Mailbox 2: Data Length		XXh
047E26h	CAN0 Mailbox 2: Data Field		XXXX XXXX XXXX XXXXh
047E27h			
047E28h			
047E29h			
047E2Ah			
047E2Bh			
047E2Ch			
047E2Dh			
047E2Eh	CAN0 Mailbox 2: Time Stamp		XXXXh
047E2Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.70 SFR List (70)

Address	Register	Symbol	Reset Value
047E30h	CAN0 Mailbox 3: Message Identifier	C0MB3	XXXX XXXXh
047E31h			
047E32h			
047E33h			
047E34h			
047E35h	CAN0 Mailbox 3: Data Length		XXh
047E36h	CAN0 Mailbox 3: Data Field		XXXX XXXX XXXX XXXXh
047E37h			
047E38h			
047E39h			
047E3Ah			
047E3Bh			
047E3Ch			
047E3Dh			
047E3Eh	CAN0 Mailbox 3: Time Stamp		XXXXh
047E3Fh			
047E40h	CAN0 Mailbox 4: Message Identifier	C0MB4	XXXX XXXXh
047E41h			
047E42h			
047E43h			
047E44h			
047E45h	CAN0 Mailbox 4: Data Length		XXh
047E46h	CAN0 Mailbox 4: Data Field		XXXX XXXX XXXX XXXXh
047E47h			
047E48h			
047E49h			
047E4Ah			
047E4Bh			
047E4Ch			
047E4Dh			
047E4Eh	CAN0 Mailbox 4: Time Stamp		XXXXh
047E4Fh			
047E50h	CAN0 Mailbox 5: Message Identifier	C0MB5	XXXX XXXXh
047E51h			
047E52h			
047E53h			
047E54h			
047E55h	CAN0 Mailbox 5: Data Length		XXh
047E56h	CAN0 Mailbox 5: Data Field		XXXX XXXX XXXX XXXXh
047E57h			
047E58h			
047E59h			
047E5Ah			
047E5Bh			
047E5Ch			
047E5Dh			
047E5Eh	CAN0 Mailbox 5: Time Stamp		XXXXh
047E5Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.71 SFR List (71)

Address	Register	Symbol	Reset Value
047E60h	CAN0 Mailbox 6: Message Identifier	C0MB6	XXXX XXXXh
047E61h			
047E62h			
047E63h			
047E64h			
047E65h	CAN0 Mailbox 6: Data Length		XXh
047E66h	CAN0 Mailbox 6: Data Field		XXXX XXXX XXXX XXXXh
047E67h			
047E68h			
047E69h			
047E6Ah			
047E6Bh			
047E6Ch			
047E6Dh			
047E6Eh	CAN0 Mailbox 6: Time Stamp		XXXXh
047E6Fh			
047E70h	CAN0 Mailbox 7: Message Identifier	C0MB7	XXXX XXXXh
047E71h			
047E72h			
047E73h			
047E74h			
047E75h	CAN0 Mailbox 7: Data Length		XXh
047E76h	CAN0 Mailbox 7: Data Field		XXXX XXXX XXXX XXXXh
047E77h			
047E78h			
047E79h			
047E7Ah			
047E7Bh			
047E7Ch			
047E7Dh			
047E7Eh	CAN0 Mailbox 7: Time Stamp		XXXXh
047E7Fh			
047E80h	CAN0 Mailbox 8: Message Identifier	C0MB8	XXXX XXXXh
047E81h			
047E82h			
047E83h			
047E84h			
047E85h	CAN0 Mailbox 8: Data Length		XXh
047E86h	CAN0 Mailbox 8: Data Field		XXXX XXXX XXXX XXXXh
047E87h			
047E88h			
047E89h			
047E8Ah			
047E8Bh			
047E8Ch			
047E8Dh			
047E8Eh	CAN0 Mailbox 8:Time Stamp		XXXXh
047E8Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.72 SFR List (72)

Address	Register	Symbol	Reset Value
047E90h	CAN0 Mailbox 9: Message Identifier	C0MB9	XXXX XXXXh
047E91h			
047E92h			
047E93h			
047E94h			
047E95h	CAN0 Mailbox 9: Data Length		XXh
047E96h	CAN0 Mailbox 9: Data Field		XXXX XXXX XXXX XXXXh
047E97h			
047E98h			
047E99h			
047E9Ah			
047E9Bh			
047E9Ch			
047E9Dh			
047E9Eh	CAN0 Mailbox 9: Time Stamp		XXXXh
047E9Fh			
047EA0h	CAN0 Mailbox 10: Message Identifier	C0MB10	XXXX XXXXh
047EA1h			
047EA2h			
047EA3h			
047EA4h			
047EA5h	CAN0 Mailbox 10: Data Length		XXh
047EA6h	CAN0 Mailbox 10: Data Field		XXXX XXXX XXXX XXXXh
047EA7h			
047EA8h			
047EA9h			
047EAAh			
047EABh			
047EACh			
047EADh			
047EAEh	CAN0 Mailbox 10: Time Stamp		XXXXh
047EAFh			
047EB0h	CAN0 Mailbox 11: Message Identifier	C0MB11	XXXX XXXXh
047EB1h			
047EB2h			
047EB3h			
047EB4h			
047EB5h	CAN0 Mailbox 11: Data Length		XXh
047EB6h	CAN0 Mailbox 11: Data Field		XXXX XXXX XXXX XXXXh
047EB7h			
047EB8h			
047EB9h			
047EBAh			
047EBBh			
047EBCh			
047EBDh			
047EBEh	CAN0 Mailbox 11: Time Stamp		XXXXh
047EBFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.73 SFR List (73)

Address	Register	Symbol	Reset Value
047EC0h	CAN0 Mailbox 12: Message Identifier	C0MB12	XXXX XXXXh
047EC1h			
047EC2h			
047EC3h			
047EC4h			
047EC5h	CAN0 Mailbox 12: Data Length		XXh
047EC6h	CAN0 Mailbox 12: Data Field		XXXX XXXX XXXX XXXXh
047EC7h			
047EC8h			
047EC9h			
047ECAh			
047ECBh			
047ECCh			
047ECDh			
047ECEh	CAN0 Mailbox 12: Time Stamp		XXXXh
047ECFh			
047ED0h	CAN0 Mailbox 13: Message Identifier	C0MB13	XXXX XXXXh
047ED1h			
047ED2h			
047ED3h			
047ED4h			
047ED5h	CAN0 Mailbox 13: Data Length		XXh
047ED6h	CAN0 Mailbox 13: Data Field		XXXX XXXX XXXX XXXXh
047ED7h			
047ED8h			
047ED9h			
047EDAh			
047EDBh			
047EDCh			
047EDDh			
047EDEh	CAN0 Mailbox 13: Time Stamp		XXXXh
047EDFh			
047EE0h	CAN0 Mailbox 14: Message Identifier	C0MB14	XXXX XXXXh
047EE1h			
047EE2h			
047EE3h			
047EE4h			
047EE5h	CAN0 Mailbox 14: Data Length		XXh
047EE6h	CAN0 Mailbox 14: Data Field		XXXX XXXX XXXX XXXXh
047EE7h			
047EE8h			
047EE9h			
047EEAh			
047EEBh			
047EECh			
047EEDh			
047EEEh	CAN0 Mailbox 14: Time Stamp		XXXXh
047EEFh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.74 SFR List (74)

Address	Register	Symbol	Reset Value
047EF0h	CAN0 Mailbox 15: Message Identifier	C0MB15	XXXX XXXXh
047EF1h			
047EF2h			
047EF3h			
047EF4h			
047EF5h	CAN0 Mailbox 15: Data Length		XXh
047EF6h	CAN0 Mailbox 15: Data Field		XXXX XXXX XXXX XXXXh
047EF7h			
047EF8h			
047EF9h			
047EFAh			
047EFBh			
047EFCh			
047EFDh			
047EFEh	CAN0 Mailbox 15: Time Stamp		XXXXh
047EFFh			
047F00h to 047F0Fh			
047F10h	CAN0 Mask Register 0	C0MKR0	XXXX XXXXh
047F11h			
047F12h			
047F13h			
047F14h	CAN0 Mask Register 1	C0MKR1	XXXX XXXXh
047F15h			
047F16h			
047F17h			
047F18h	CAN0 Mask Register 2	C0MKR2	XXXX XXXXh
047F19h			
047F1Ah			
047F1Bh			
047F1Ch	CAN0 Mask Register 3	C0MKR3	XXXX XXXXh
047F1Dh			
047F1Eh			
047F1Fh			
047F20h	CAN0 FIFO Receive ID Compare Register 0	C0FIDCR0	XXXX XXXXh
047F21h			
047F22h			
047F23h			
047F24h	CAN0 FIFO Receive ID Compare Register 1	C0FIDCR1	XXXX XXXXh
047F25h			
047F26h			
047F27h			
047F28h			
047F29h			
047F2Ah	CAN0 Mask Invalid Register	C0MKIVLR	XXXXh
047F2Bh			
047F2Ch			
047F2Dh			
047F2Eh	CAN0 Mailbox Interrupt Enable Register	C0MIER	XXXXh
047F2Fh			

X: Undefined
Blanks are reserved. No access is allowed.

Table 4.75 SFR List (75)

Address	Register	Symbol	Reset Value
047F30h	CAN0 Message Control Register 0	C0MCTL0	00h
047F31h	CAN0 Message Control Register 1	C0MCTL1	00h
047F32h	CAN0 Message Control Register 2	C0MCTL2	00h
047F33h	CAN0 Message Control Register 3	C0MCTL3	00h
047F34h	CAN0 Message Control Register 4	C0MCTL4	00h
047F35h	CAN0 Message Control Register 5	C0MCTL5	00h
047F36h	CAN0 Message Control Register 6	C0MCTL6	00h
047F37h	CAN0 Message Control Register 7	C0MCTL7	00h
047F38h	CAN0 Message Control Register 8	C0MCTL8	00h
047F39h	CAN0 Message Control Register 9	C0MCTL9	00h
047F3Ah	CAN0 Message Control Register 10	C0MCTL10	00h
047F3Bh	CAN0 Message Control Register 11	C0MCTL11	00h
047F3Ch	CAN0 Message Control Register 12	C0MCTL12	00h
047F3Dh	CAN0 Message Control Register 13	C0MCTL13	00h
047F3Eh	CAN0 Message Control Register 14	C0MCTL14	00h
047F3Fh	CAN0 Message Control Register 15	C0MCTL15	00h
047F40h	CAN0 Control Register	C0CTLR	0000 0101b
047F41h			0000 0000b
047F42h	CAN0 Status Register	C0STR	0000 0101b
047F43h			0000 0000b
047F44h	CAN0 Bit Configuration Register	C0BCR	00 0000h
047F45h			
047F46h			
047F47h	CAN0 Clock Select Register	C0CLKR	000X 0000b
047F48h	CAN0 Receive FIFO Control Register	C0RFCR	1000 0000b
047F49h	CAN0 Receive FIFO Pointer Control Register	C0RFPCR	XXh
047F4Ah	CAN0 Transmit FIFO Control Register	C0TFCR	1000 0000b
047F4Bh	CAN0 Transmit FIFO Pointer Control Register	C0TFPCR	XXh
047F4Ch	CAN0 Error Interrupt Enable Register	C0EIER	00h
047F4Dh	CAN0 Error Interrupt Factor Judge Register	C0EIFR	00h
047F4Eh	CAN0 Receive Error Count Register	C0RECR	00h
047F4Fh	CAN0 Transmit Error Count Register	C0TECR	00h
047F50h	CAN0 Error Code Store Register	C0ECSR	00h
047F51h	CAN0 Channel Search Support Register	C0CSSR	XXh
047F52h	CAN0 Mailbox Search Status Register	C0MSSR	1000 0000b
047F53h	CAN0 Mailbox Search Mode Register	C0MSMR	0000 0000b
047F54h	CAN0 Time Stamp Register	C0TSR	0000h
047F55h			
047F56h	CAN0 Acceptance Filter Support Register	C0AFSR	XXXXh
047F57h			
047F58h	CAN0 Test Control Register	C0TCR	00h
047F59h			
047F5Ah			
047F5Bh			
047F5Ch			
047F5Eh			
047F5Eh			
047F5Fh			
047F60h to 04FFFFh			

X: Undefined
Blanks are reserved. No access is allowed.

5. Resets

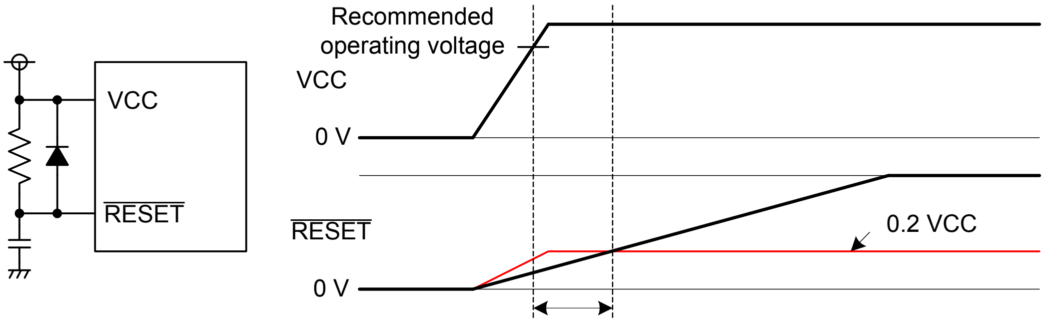
There are three types of operations for resetting the MCU: hardware reset, software reset, and watchdog timer reset.

5.1 Hardware Reset

A hardware reset is generated when a low signal is applied to the $\overline{\text{RESET}}$ pin under the recommended operating conditions of the supply voltage. When the $\overline{\text{RESET}}$ pin is driven low, all pins, and oscillators are initialized (refer to Table 5.1 for details), and the main clock starts oscillating. The CPU and SFRs are initialized by a low-to-high transition on the $\overline{\text{RESET}}$ pin. Then, the CPU starts executing the program from the address indicated by the reset vector. Internal RAM is not affected by a hardware reset. However, if a hardware reset occurs during a write operation to the internal RAM, the value is undefined.
Figure 5.1 shows an example of the reset circuit. Figure 5.2 shows the reset sequence. Table 5.1 lists pin states while the $\overline{\text{RESET}}$ pin is held low. Figure 5.3 shows CPU register states after reset. Refer to 4. “Special Function Registers (SFRs)” for details on the states of SFRs after a reset.

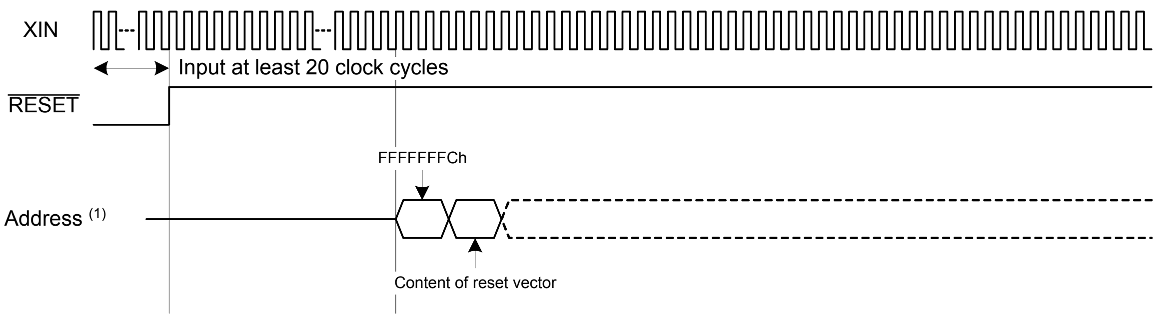
A. Reset when the supply voltage is stable
  (1) Drive the $\overline{\text{RESET}}$ pin low.
  (2) Input at least 20 clock cycles to the XIN pin.
  (3) Drive the $\overline{\text{RESET}}$ pin high.

B. Reset when turning on the power
  (1) Drive the $\overline{\text{RESET}}$ pin low.
  (2) Raise the supply voltage to the recommended operating voltage.
  (3) Wait td(P-R) ms until the internal voltage is stabilized.
  (4) Input at least 20 clock cycles to the XIN pin.
  (5) Drive the $\overline{\text{RESET}}$ pin high.



This width indicates internal power supply stabilization time (td(P-R)) + at least 20 cycles of an clock input to the XIN pin

Figure 5.1 Reset Circuitry



Note:

1. Address data is not output from pins.

Figure 5.2 Reset Sequence

Table 5.1 Pin States while RESET Pin is Held Low (1)

Pin Name	Pin States
P0 to P10	Input port (high-impedance)

Note:

1. Whether a pull-up resistor is enabled or not is undefined until the internal voltage has stabilized.

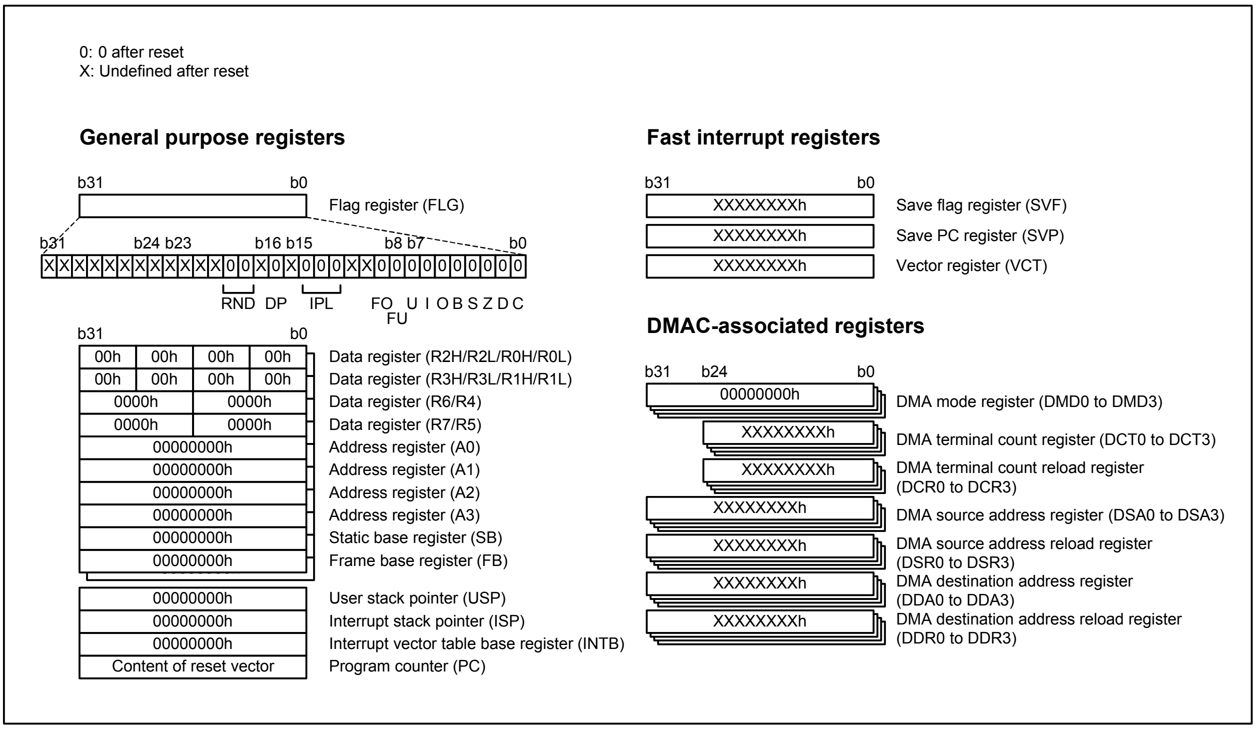


Figure 5.3 CPU Registers after Reset

5.2 Software Reset

The CPU, SFRs, and pins are initialized when the PM03 bit in the PM0 register is set to 1 (the MCU is reset). Then, the CPU executes the program from the address indicated by the reset vector. Figure 5.4 shows the PM0 register.

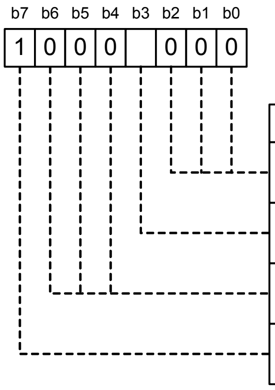
The PM03 bit should be set to 1 while the PLL clock is selected as the CPU clock source and the main clock oscillation is completely stable.

Processor Mode Register 0 (1)

Symbol PM0

Address 40044h

Reset Value 1000 0000b



Bit Symbol	Bit Name	Function	RW
— (b2-b0)	Reserved	Should be written with 0	RW
PM03	Software Reset Bit	The MCU is reset when this bit is set to 1. The bit is read as 0	RW
— (b6-b4)	Reserved	Should be written with 0	RW
— (b7)	Reserved	Should be written with 1	RW

Note:
1. Set the PRC1 bit in the PRCR register to 1 (write enabled) before rewriting this register.

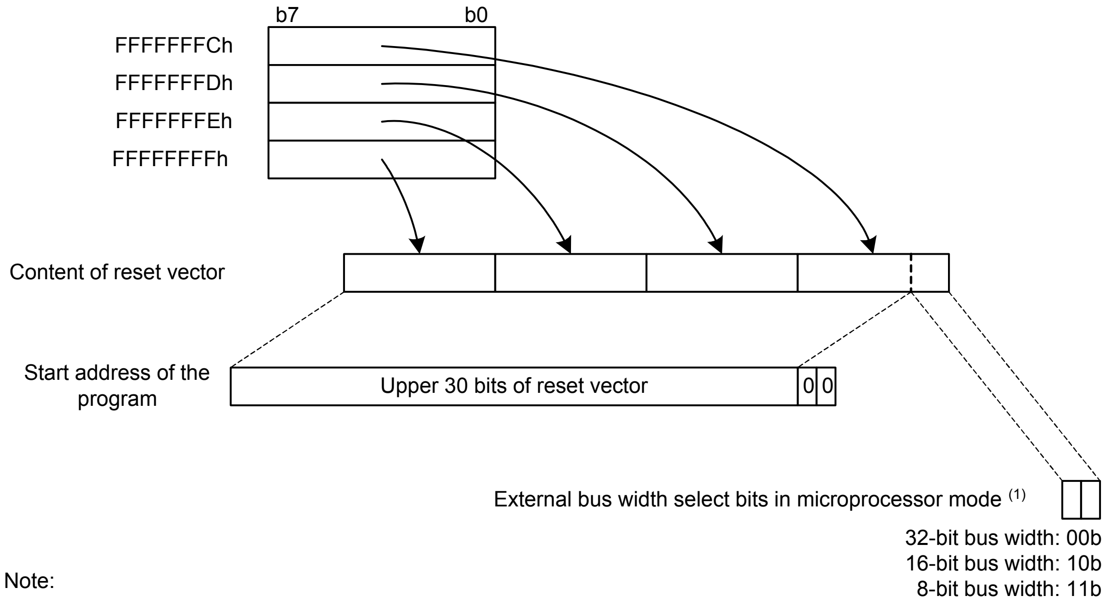
Figure 5.4 PM0 Register

5.3 Watchdog Timer Reset

The CPU, SFRs, and pins are initialized when the watchdog timer underflows while the CM06 bit in the CM0 register is 1 (reset when watchdog timer underflows). Then, the CPU executes the program from the address indicated by the reset vector.

5.4 Reset Vector

The reset vector in the R32C/100 Series is configured as shown in Figure 5.5.
The start address of a program consists of the upper 30 bits of the reset vector and 00b as lower 2 bits.
The lower 2 bits of the reset vector are bits to select the external bus width in microprocessor mode.
Therefore, the start address of a program requires 4-byte alignment so that the lower 2 bits are 00b.
In single-chip mode, set the external bus width select bits to 00b.



Note:
1. The bits should be set to 00b in single-chip mode.

Figure 5.5 Reset Vector Configuration

6. Power Management

6.1 Voltage Regulators for Internal Logic

The supply voltage for internal logic is generated by reducing the input voltage from the VCC0 pin with the voltage regulators. Figure 6.1 shows a block diagram of the voltage regulators for internal logic, and Figure 6.2 shows the VRCR register.

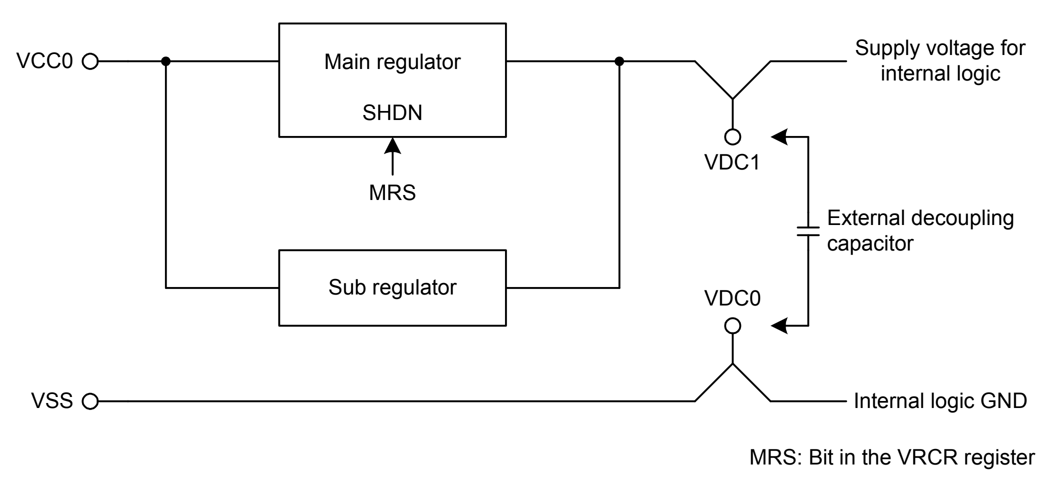
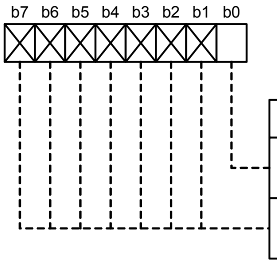


Figure 6.1 Block Diagram of Voltage Regulators for Internal Logic

Voltage Regulator Control Register [(1)]



Symbol	Address	Reset Value
VRCR	040060h	0000 0000b

Bit Symbol	Bit Name	Function	RW
MRS	Main Regulator Shut-down Bit (2)	0: Main regulator active 1: Main regulator stopped (3)	RW
— (b7-b1)	No register bits; should be written with 0 and read as 0		—

Notes:

1. Set the PRC31 bit in the PRCR3 register to 1 (write enabled) before rewriting this register.
2. This bit is fixed to 0 if the CM05 bit in the CM0 register is 0 (main clock oscillator enabled) or the CM10 bit in the CM1 register is 0 (PLL oscillator enabled) .
3. While the main regulator is stopped, do not rewrite the flash memory.

Figure 6.2 VRCR Register

6.1.1 Decoupling Capacitor

An external decoupling capacitor is required to stabilize internal voltage. The capacitor should be beneficially effective at higher frequencies and maintain a more stable capacitance irrespective of temperature change. In general, ceramic capacitors are recommended. The capacitance varies by conditions such as operating temperature, DC bias, and aging. To select an appropriate capacitor, these conditions should be considered. Also, refer to the recommended capacitor specifications listed in Table 6.1.

The traces between the capacitor and the VDC1/VDC0 pins should be as short and wide as physically possible.

Table 6.1 Recommended Capacitor Specifications

Temperature Characteristics				Rated Voltage (V)	Nominal Capacitance (μF)	Capacitance Tolerance (%)
Applicable standard		Operating temperature range (°C)	Capacitance change (%)			
B	JIS	-25 to 85	±10	6.3 or higher	4.7	±20 or better
R	JIS	-55 to 125	±15	6.3 or higher	4.7	±20 or better
X5R	EIA	-55 to 85	±15	6.3 or higher	4.7	±20 or better
X7R	EIA	-55 to 125	±15	6.3 or higher	4.7	±20 or better
X8R	EIA	-55 to 150	±15	6.3 or higher	4.7	±20 or better
X6S	EIA	-55 to 105	±22	6.3 or higher	4.7	±20 or better
X7S	EIA	-55 to 125	±22	6.3 or higher	4.7	±20 or better

7. Clock Generator

7.1 Clock Generator Types

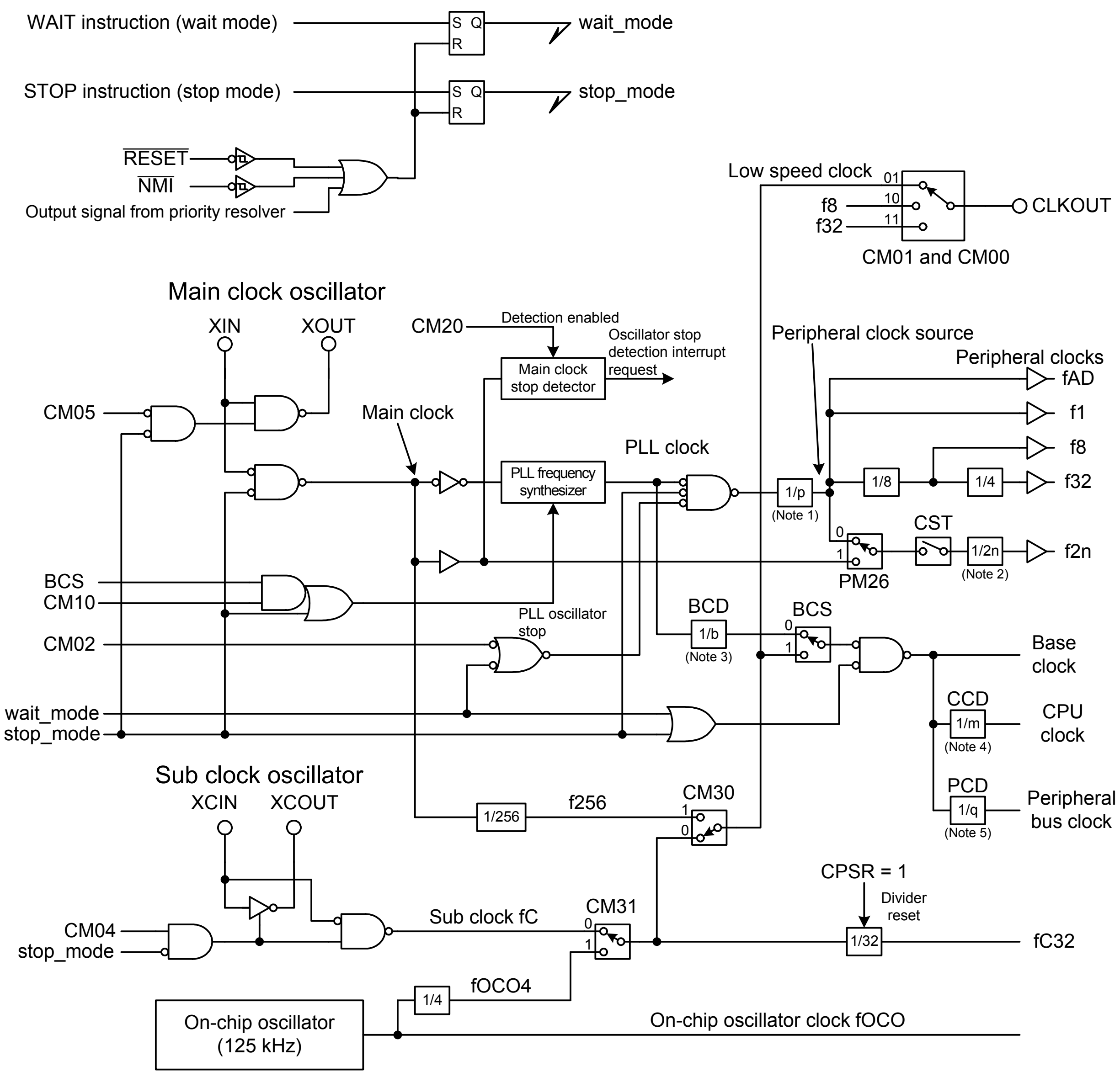
The clock generator consists of four circuits:

- Main clock oscillator
- Sub clock oscillator
- PLL frequency synthesizer
- On-chip oscillator (OCO)

Table 7.1 lists the specifications of the clock generator. Figure 7.1 shows a block diagram of the clock generator, and Figure 7.2 to Figure 7.10 show registers associated with clock control.

Table 7.1 Clock Generator Specifications

Item	Main Clock Oscillator	Sub Clock Oscillator	PLL Frequency Synthesizer	On-chip Oscillator
Used as	• PLL reference clock source • Peripheral clock source	• CPU clock source • Clock source for timers A and B	• CPU clock source • Peripheral clock source	• CPU clock source • Clock source for timers A and B
Clock frequency	4 to 8 MHz	32.768 kHz	$f_{SO(PLL)}$ or $f_{(PLL)}$	Approx. 125 kHz
Connectable oscillators or additional circuits	Ceramic resonator Crystal oscillator	Crystal oscillator	—	—
Pins for oscillators or additional circuits	XIN, XOUT	XCIN, XCOUT	—	—
Oscillator stop/ restart function	Available	Available	Available	Available
Oscillator state after a reset	Running	Stopped	Running	Stopped
Note	Externally generated clock can be input	Externally generated clock can be input	When the main clock oscillator stops running, the PLL frequency synthesizer oscillates at its own frequency of $f_{SO(PLL)}$	The on-chip oscillator starts running by setting the CSPM bit in the OFS area to 0 after a reset



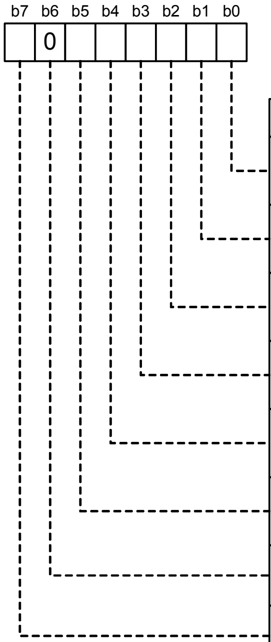
CM00 to CM02, CM04, and CM05: Bits in the CM0 register
CM10: Bit in the CM1 register
CM20: Bit in the CM2 register
CM30 and CM31: Bits in the CM3 register
PM26: Bit in the PM2 register
CST: Bit in the TCSPR register
CPSR: Bit in the CPSRF register
BCS: Bit in the CCR register

Notes:

1. The value of p (p = 2, 4, 6, 8) can be selected by setting bits PM36 and PM35 in the PM3 register.
2. The value of n (n = 0 to 15) can be selected by setting bits CNT3 to CNT0 in the TCSPR register. When n is 0, the clock is not divided.
3. The value of b (b = 2, 3, 4, 6) can be selected by setting bits BCD1 and BCD0 in the CCR register.
4. The value of m (m = 1 to 4) can be selected by setting bits CCD1 and CCD0 in the CCR register.
5. The value of q (q = 2 to 4) can be selected by setting bits PCD1 and PCD0 in the CCR register.

Figure 7.1 Clock Generation Circuitry

Clock Control Register [1]



Symbol	Address	Reset Value
CCR	0004h	0001 1000b

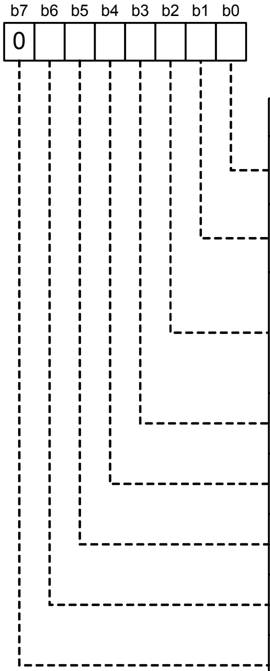
Bit Symbol	Bit Name	Function	RW
BCD0	Base Clock Divide Ratio Select Bit [2]	b1 b0 0 0 : Divide-by-6 0 1 : Divide-by-4 1 0 : Divide-by-3 1 1 : Divide-by-2	RW
BCD1			RW
CCD0	CPU Clock Divide Ratio Select Bit [3]	b3 b2 0 0 : Divide-by-4 0 1 : Divide-by-3 1 0 : Divide-by-2 1 1 : No division	RW
CCD1			RW
PCD0	Peripheral Bus Clock Divide Ratio Select Bit [2, 3, 4]	b5 b4 0 0 : Do not use this combination 0 1 : Divide-by-2 1 0 : Divide-by-3 1 1 : Divide-by-4	RW
PCD1			RW
— (b6)	Reserved	Should be written with 0	RW
BCS	Base Clock Source Select Bit	0: PLL clock 1: fC, fOCO4, or f256 [5, 6]	RW

Notes:

1. Set the PRR register to AAh (write enabled) before rewriting this register.
2. The divide ratios of the base clock and peripheral bus clock should not be changed simultaneously. Doing so may cause the peripheral bus clock frequency to go over the maximum operating frequency.
3. The divide ratio of the CPU clock should be equal to or lower than that of peripheral bus clock.
4. This bit should be set only once after a reset and the setting should not be changed. To rewrite this bit, the PBC register should be rewritten first.
5. To set this bit to 1, a 32-bit write access to addresses 0004h to 0007h should be performed.
6. To use these low speed clocks, select one of them by setting bits CM31 and CM30 in the CM3 register and then set the BCS bit to 1.

Figure 7.2 CCR Register

System Clock Control Register 0 (1)



Symbol	Address	Reset Value
CM0	40046h	0000 1000b

Bit Symbol	Bit Name	Function	RW
CM00	Clock Output Function Select Bit	b1 b0 0 0 : I/O port P5_3 0 1 : Output a low speed clock 1 0 : Output f8 1 1 : Output f32	RW
CM01			RW
CM02	Peripheral Clock Source Stop Bit (2)	0: Peripheral clock source not stopped in wait mode 1: Peripheral clock source stopped in wait mode (3)	RW
CM03	XCIN-XCOUT Drive Power Select Bit (4)	0: Low 1: High	RW
CM04	Port XC Switch Bit	0: I/O port 1: XCIN-XCOUT oscillator (5)	RW
CM05	Main Clock Oscillator (XIN-XOUT) Stop Bit (2, 6)	0: Main clock oscillator enabled 1: Main clock oscillator disabled	RW
CM06	Watchdog Timer Function Select Bit (7)	0: Watchdog timer interrupt 1: Reset (8)	RW
— (b7)	Reserved	Should be written with 0	RW

Notes:

1. Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register.
2. When the PM21 bit in the PM2 register is 1 (clock change disabled), bits CM02 and CM05 are not changed by a write access.
3. fC32 and f2n (whose clock source is the main clock) do not stop.
4. When entering stop mode, the CM03 bit becomes 1.
5. To set the CM04 bit to 1, bits PD8_7 and PD8_6 in the PD8 register should be set to 0 (input), and the PU25 bit in the PUR2 register should be set to 0 (pull-up resistor disabled).
6. This bit stops the main clock when entering low power mode. It cannot detect whether or not the main clock oscillator stops. When this bit is set to 1, the clock applied to the XOUT pin becomes high. Since the on-chip feedback resistor remains connected, the XIN pin is connected to the XOUT pin via the feedback resistor.
7. Set this bit before activating the watchdog timer. When rewriting this bit while the watchdog timer is running, set it immediately after writing to the WDTS register.
8. Once this bit is set to 1, it cannot be set to 0 by a program.

Figure 7.3 CM0 Register

System Clock Control Register 1 (1)

b7	b6	b5	b4	b3	b2	b1	b0
0	0	1	0	0	0	0	

Symbol
CM1

Address
40047h

Reset Value
0010 0000b

Bit Symbol	Bit Name	Function	RW
CM10	PLL Oscillator Stop Bit (2, 3)	0: PLL oscillator enabled 1: PLL oscillator disabled	RW
— (b4-b1)	Reserved	Should be written with 0	RW
— (b5)	Reserved	Should be written with 1	RW
— (b7-b6)	Reserved	Should be written with 0	RW

Notes:

1. Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register.
2. When the BCS bit in the CCR register is 0 (PLL clock selected as base clock source), the PLL frequency synthesizer does not stop oscillating even if the CM10 bit is set to 1.
3. When the PM21 bit in the PM2 register is 1 (clock change disabled), the CM10 bit cannot be changed by a write access.

Figure 7.4 CM1 Register

Oscillator Stop Detection Register (1)

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0			0	

Symbol
CM2

Address
4004Dh

Reset Value
0000 0000b

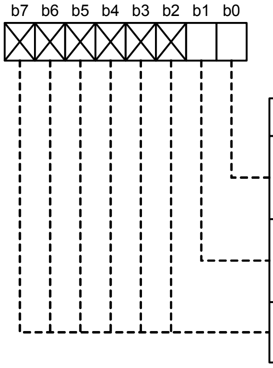
Bit Symbol	Bit Name	Function	RW
CM20	Oscillator Stop Detection Enable Bit (2, 3)	0: Disable oscillator stop detection 1: Enable oscillator stop detection	RW
— (b1)	Reserved	Should be written with 0	RW
CM22	Oscillator Stop Detection Flag (4)	0: Main clock oscillator has not been stopped 1: Main clock oscillator stop detected	RW
CM23	Main Clock Monitor Flag (5)	0: Main clock oscillator active 1: Main clock oscillator stopped	RO
— (b7-b4)	Reserved	Should be written with 0	RW

Notes:

1. Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register.
2. This bit should be set to 0 when f256 is selected as the base clock source in low speed mode.
3. When the PM21 bit in the PM2 register is 1 (clock change disabled), the CM20 bit cannot be changed by a write access.
4. When a main clock oscillator stop is detected, this bit becomes 1. It can be set to 0 by a program, however not to 1. When it is set to 0 while the main clock oscillator is stopped, it does not become 1 until the next main clock oscillator stop is detected.
5. The main clock state should be determined by several read accesses of this bit after an oscillator stop detection interrupt is generated.

Figure 7.5 CM2 Register

Low Speed Mode Clock Control Register (1)



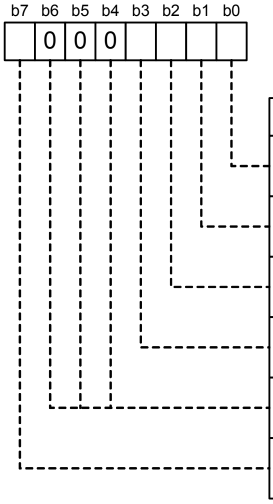
Symbol	Address	Reset Value
CM3	4005Ah	XXXX XX00b

Bit Symbol	Bit Name	Function	RW
CM30	Low Speed Mode Base Clock Select Bit	b1 b0 0 0 : fC 0 1 : f256 (main clock divided by 256) 1 0 : fOCO4 (on-chip oscillator divided by 4) (2) 1 1 : Do not use this combination	RW
CM31			RW
— (b7-b2)	No register bits; should be written with 0 and read as undefined value		—

Notes:
1. Rewrite this register after setting the PRC27 bit in the PRCR2 register to 1 (write enabled) and while the BCS bit in the CCR register is 0 (PLL clock).
2. The on-chip oscillator clock starts when the CM31 bit is set to 1.

Figure 7.6 CM3 Register

Count Source Prescaler Register



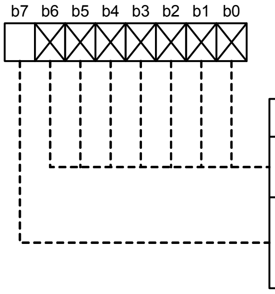
Symbol	Address	Reset Value
TCSPR	035Fh	0000 0000b

Bit Symbol	Bit Name	Function	RW
CNT0	Divide Ratio Select Bit (1)	f2n is either the main clock or peripheral clock source divided by 2n. If n = 0, the clock is not divided (n = setting value)	RW
CNT1			RW
CNT2			RW
CNT3			RW
— (b6-b4)	Reserved	Should be written with 0	RW
CST	Divider Operation Enable Bit	0: Stop divider operation 1: Start divider operation	RW

Note:
1. The CST bit should be set to 0 before rewriting bits CNT3 to CNT0.

Figure 7.7 TCSPR Register

Clock Prescaler Reset Register

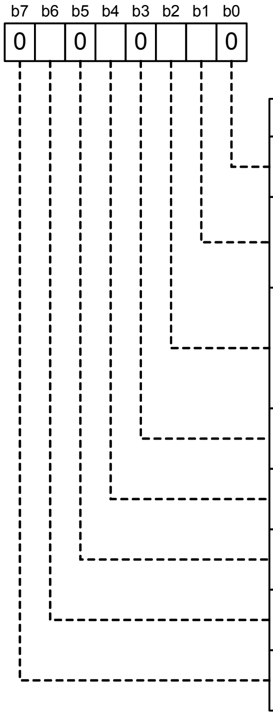


Symbol	Address	Reset Value
CPSRF	0341h	0XXX XXXXb

Bit Symbol	Bit Name	Function	RW
— (b6-b0)	No register bits; should be written with 0 and read as undefined value		—
CPSR	Clock Prescaler Reset Bit	When this bit is set to 1, the fC divided-by-32 divider is initialized. The bit is read as 0	RW

Figure 7.8 CPSRF Register

Processor Mode Register 2 (1)



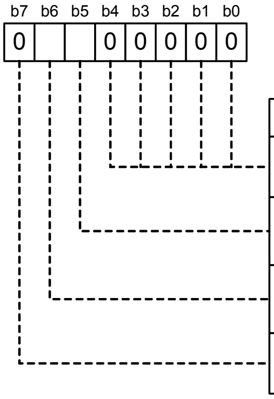
Symbol	Address	Reset Value
PM2	40053h	0000 0000b

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved	Should be written with 0	RW
PM21	System Clock Protect Bit (2, 3)	0: Protect the clock by the PRCR register 1: Clock change disabled	RW
PM22	Watchdog Timer Count Source Protect Bit (2, 4)	0: Peripheral bus clock functions as watchdog timer count source 1: On-chip oscillator clock functions as watchdog timer count source	RW
— (b3)	Reserved	Should be written with 0	RW
PM24	NMI Enable Bit (2)	0: NMI disabled (5) 1: NMI enabled	RW
— (b5)	Reserved	Should be written with 0	RW
PM26	f2n Clock Source Select Bit (6)	0: Peripheral clock source 1: Main clock	RW
— (b7)	Reserved	Should be written with 0	RW

Notes:
1. Set the PRC1 bit in the PRCR register to 1 (write enabled) before rewriting this register.
2. Once this bit is set to 1, it cannot be set to 0 by a program.
3. When the PM21 bit is set to 1, the following bits are not changed by a write access:
   CM02 bit in the CM0 register (the peripheral clock source state in wait mode)
   CM05 bit in the CM0 register (main clock oscillator enabled/disabled)
   CM10 bit in the CM1 register (PLL oscillator enabled/disabled)
   CM20 bit in the CM2 register (oscillator stop detection enabled/disabled)
4. This bit should be set before the watchdog timer activates.
   When the PM22 bit is set to 1, the state changes as follows:
   -On-chip oscillator clock starts running
   -On-chip oscillator clock functions as the count source of the watchdog timer
   -The watchdog timer is not stopped
5. When the PM24 bit is 0, the forced cutoff of the three-phase motor control timers is also disabled.
6. Disable all the peripheral functions that use f2n before rewriting this bit.

Figure 7.9 PM2 Register

Processor Mode Register 3 (1)



Symbol	Address	Reset Value
PM3	40048h	0000 0000b

Bit Symbol	Bit Name	Function	RW
— (b4-b0)	Reserved	Should be written with 0	RW
PM35	Peripheral Clock Source Divide Ratio Select Bit (2)	b6 b5 0 0 : Divide-by-8 0 1 : Divide-by-6 1 0 : Divide-by-4 1 1 : Divide-by-2	RW
PM36			RW
— (b7)	Reserved	Should be written with 0	RW

Notes:

1. Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register. Disable all the peripheral functions that use fAD, f1, f8, f32, or f2n (when the clock source is the peripheral clock source) to rewrite this register.
2. Select a divide ratio so that the peripheral clock source frequency does not exceed the maximum value specified in the electrical characteristics

Figure 7.10 PM3 Register

The following sections illustrate clocks generated in clock generators.

7.1.1 Main Clock

The main clock is generated by the main clock oscillator. This clock can be a clock source for the PLL reference clock or peripheral clocks. It also functions as an operating clock for the CAN module.

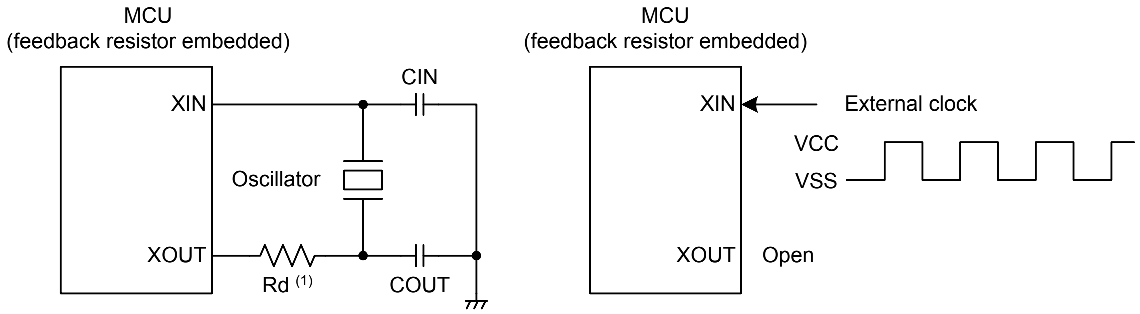
The main clock oscillator is configured with two pins, XIN and XOUT, connected by an oscillator or resonator. The circuit has an on-chip feedback resistor which is separated from the oscillator in stop mode to save power consumption. An external clock can be applied to the XIN pin in this circuit. Figure 7.11 shows an example of a main clock circuit connection.

Circuit constants vary depending on the oscillator. Circuit constants should be set as per the oscillator manufacturer’s recommendations.

After a reset, the main clock oscillator is still independently active and disconnected from the PLL frequency synthesizer. A PLL frequency synthesizer self-oscillating clock divided by 12 is provided to the CPU.

Setting the CM05 bit in the CM0 register to 1 (main clock oscillator disabled) enables power-saving. In this case, the clock applied to the XOUT pin becomes high. The XIN pin connected to the XOUT pin by an embedded feedback resistor is also driven high. When an external clock is applied to the XIN pin, the CM05 bit should not be set to 1.

All clocks, including the main clock, stop in stop mode. Refer to 7.7 “Power Control” for details.



Note:

1. Insert a damping resistor if required. Resistance values may vary according to oscillator setting. Values recommended by the manufacturer should be set. A feedback resistor should be placed between XIN and XOUT if the manufacturer recommends placing a resistor externally.

Figure 7.11 Main Clock Circuit Connection

7.1.2 Sub Clock (fC)

The sub clock is generated by the sub clock oscillator. This clock can be a clock source for the CPU clock and a count source for timers A and B. It can be output from the CLKOUT pin.

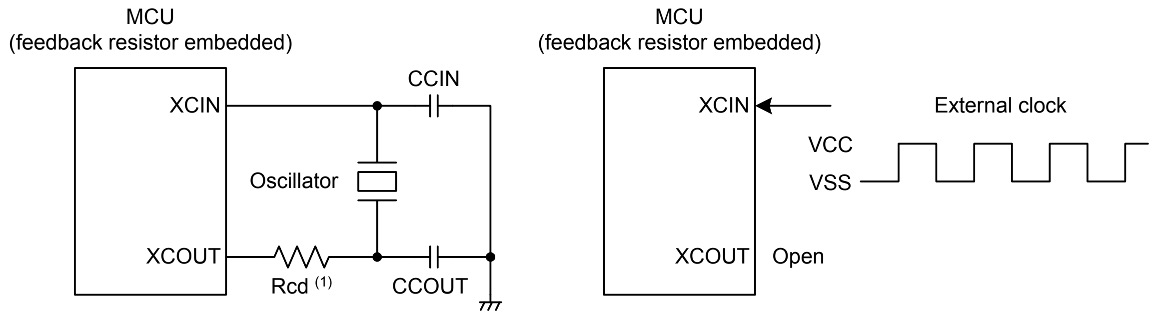
The sub clock oscillator is configured with pins XCIN and XCOUT connected by a crystal oscillator. The circuit has a on-chip feedback resistor which is separated from the oscillator in stop mode to save power consumption. An external clock can be applied to the XCIN pin. Figure 7.12 shows an example of a sub clock circuit connection. Circuit constants vary depending on the oscillator. Circuit constants should be set as per the oscillator manufacturer’s recommendations.

After a reset, the sub clock is stopped and the feedback resistor is separated from the oscillator. In order to start the sub clock oscillation, first set bits PD8_6 and PD8_7 in the PD8 register to 0 (input mode), and the PU25 bit in the PUR2 register to 0 (pull-up resistor disabled). Then, set the CM04 bit in the CM0 register to 1 (XCIN-XCOUT oscillator).

To input an external clock to the XCIN pin, bits PD8_7 and PU25 should be set to 0, then the CM04 bit should be set to 1. The clock applied to the XCIN pin becomes a clock source for the sub clock.

When the CM3 register is set to 00h (fC) and the BCS bit in the CCR register is set to 1 (fC, fOCO4, or f256) after the sub clock oscillation has stabilized, the sub clock becomes the base clock of the CPU clock and the peripheral bus clock.

All clocks, including the sub clock, stop in stop mode. Refer to 7.7 “Power Control” for details.



Note:

1. Insert a damping resistor if required. Resistance values vary according to oscillator setting. Values recommended by the manufacturer should be set. A feedback resistor should be placed between XCIN and XCOUT if the manufacturer recommends placing a resistor externally.

Figure 7.12 Sub Clock Circuit Connection

7.1.3 PLL Clock

The PLL clock is generated by the PLL frequency synthesizer based on the main clock. This clock can be a clock source for any clock including the CPU clock and the peripheral clock.

Figure 7.13 shows a block diagram of the PLL frequency synthesizer. Figure 7.14 and Figure 7.15 show registers PLC0 and PLC1, respectively.

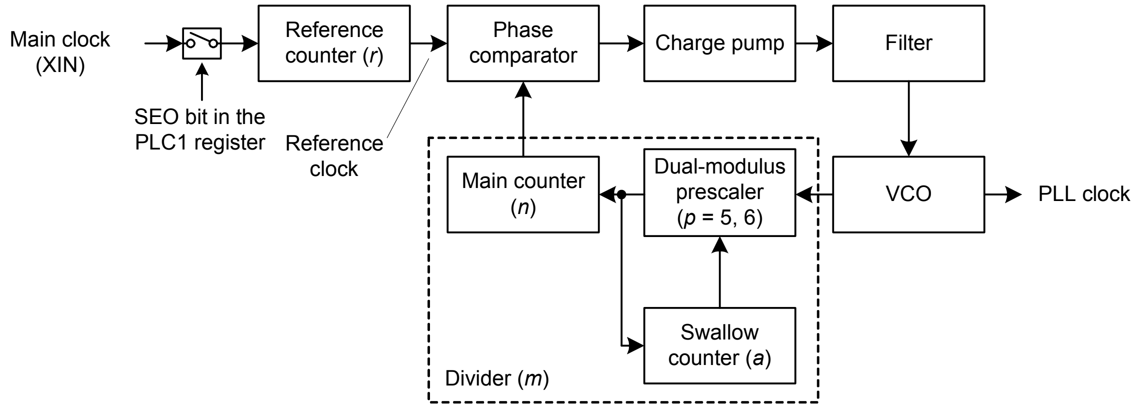


Figure 7.13 PLL Frequency Synthesizer Block Diagram

PLL Control Register 0 (1, 2)

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	Reset Value
PLC0	40020h	0000 0001b

Bit Symbol	Bit Name	Function	RW
MCV0	Main Counter Divide Ratio Setting Bit	Set the bits to *n* - 1 (*n* = divide ratio of the main counter)	RW
MCV1			RW
MCV2			RW
MCV3			RW
MCV4			RW
SCV0	Swallow Counter Divide Ratio Setting Bit	The divide ratio of the dual-modulus prescaler is 6 (in *a* out of *n* times) or 5 (in other cases) (*a* = setting value)	RW
SCV1			RW
SCV2			RW

Notes:
1. Set the PRC2 bit in the PRCR register to 1 (write enabled) just before rewriting this register. No interrupt handling or DMA transfers should be inserted between these two instructions.
2. This register can be rewritten only once after a reset.

Figure 7.14 PLC0 Register

PLL Control Register 1 (1)

b7 b6 b5 b4 b3 b2 b1 b0
0 0 0

Symbol	Address	Reset Value
PLC1	40021h	0001 1111b

Bit Symbol	Bit Name	Function	RW
RCV0	Reference Counter Divide Ratio Setting Bit	Set the bits to *r* - 1 (*r* = divide ratio of the main clock)	RW
RCV1			RW
RCV2			RW
RCV3			RW
SEO	Self-oscillating Setting Bit	0: PLL lock-in 1: Self-oscillating	RW
— (b7-b5)	Reserved	Should be written with 0	RW

Note:

1. Set the PRC2 bit in the PRCR register to 1 (write enabled) just before rewriting this register. No interrupt handling or DMA transfers should be inserted between these two instructions.

Figure 7.15 PLC1 Register

In the PLL frequency synthesizer, the pulse-swallow operation is implemented. The divide ratio *m* is simply expressed by *n*×*p*. However, with the swallow counter, the divide ratio *p* is 6 in *a* out of *n*, or 5 in other cases, the actual *m* is therefore given by the formula below:

$$\begin{aligned} m &= n \times p \\ &= n \times \left(\frac{a}{n} \cdot 6 + \frac{n-a}{n} \cdot 5\right) \\ &= 5n + a \end{aligned}$$

The setting range of *a* is $0 \le a < 5$, $0 \le a \le n$.

As *r* is the divide ratio of the reference counter, the PLL clock has a *m/r* times the main clock (XIN) frequency.

$$\begin{aligned} \text{PLL clock frequency } f(PLL) &= \frac{m}{r} \cdot \text{main clock frequency} \\ &= \frac{5n+a}{r} \cdot \text{main clock frequency} \end{aligned}$$

After a reset, the reference counter is divided by 16, and the PLL frequency synthesizer is multiplied by 10. Since the main clock as a reference clock is disconnected, the PLL frequency synthesizer may self-oscillate at its own frequency of $f_{SO(PLL)}$.

Each register should be set to meet the following conditions:

-The reference clock, which is the main clock divided by *r*, should be between 2 to 4 MHz.

-The divide ratio *m* is $25 \le m \le 100$.

For the setting of registers PLC1 and PLC0, Table 7.2 should be applied. While the main clock oscillation is stable, a wait time of $t_{LOCK(PLL)}$ is necessary between rewriting registers PLC1 and PLC0, and the PLL clock becoming stable.

Table 7.2 PLC1 and PLC0 Register Settings (1)

Main Clock	*r*	Reference Clock	*n*	*a*	*m*	PLC1 Register Setting	PLC0 Register Setting	*m/r*	PLL Clock
4 MHz	1	4 MHz	6	0	30	00h	05h	30	120 MHz
5 MHz	2	2.5 MHz	9	3	48	01h	68h	24	120 MHz
6 MHz	2	3 MHz	8	0	40	01h	07h	20	120 MHz
8 MHz	2	4 MHz	6	0	30	01h	05h	15	120 MHz
4 MHz	1	4 MHz	6	2	32	00h	45h	32	128 MHz
6 MHz	3	2 MHz	12	4	64	02h	8Bh	21.3333	128 MHz
8 MHz	2	4 MHz	6	2	32	01h	45h	16	128 MHz

Note:

1. Registers PLC1 and PLC0 should be set according to the list above.

7.1.4 On-chip Oscillator Clock

The on-chip oscillator clock is generated by the on-chip oscillator (OCO). This clock can be a clock source for the CPU clock and a count source for timers A and B. This clock has a frequency of approximately 125 kHz. The on-chip oscillator clock divided by 4 can be used as the base clock for the CPU clock and peripheral bus clock.

When the CSPM bit in the OFS area is set to 1, the on-chip oscillator clock is stopped after a reset. It starts running if the CM31 bit in the CM3 register or the PM22 bit in the PM2 register is set to 1. It is not necessary to wait for stabilization because the on-chip oscillator instantly starts oscillating.

7.2 Oscillator Stop Detection

This function detects the main clock is stopped when its oscillator stops running due to an external factor. When the CM20 bit in the CM2 register is set to 1 (enable oscillator stop detection), an oscillator stop detection interrupt request is generated as soon as the main clock stops. Simultaneously, the PLL frequency synthesizer starts to self-oscillate at its own frequency. If the PLL frequency synthesizer is the clock source for CPU clock and peripheral clock, these clocks continue running.

When an oscillator stop is detected, the following bits in the CM2 register become 1:

- The CM22 bit: main clock oscillator stop detected
- The CM23 bit: main clock oscillator stopped

7.2.1 How to Use Oscillator Stop Detection

The oscillator stop detection interrupt shares vectors with the watchdog timer interrupt. When using these interrupts simultaneously, read the CM22 bit with an interrupt handler to determine if an oscillator stop detection interrupt request has been generated.

When the main clock oscillator resumes running after an oscillator stop is detected, the PLL clock frequency may temporarily exceed the preset value before the PLL frequency synthesizer oscillation stabilizes. As soon as an oscillator stop is detected, the main clock oscillator should be stopped from resuming (set the CM05 bit in the CM0 register to 1) or the divide ratios of the base clock and peripheral clock source should be increased by a program. They can be set using bits BCD1 and BCD0 in the CCR register and bits PM36 and PM35 in the PM3 register.

In low speed mode, when the main clock oscillator stops running, an oscillator stop detection interrupt request is generated if the CM20 bit is set to 1 (enable oscillator stop detection). The CPU clock remains running with a low speed clock source. Note that if the base clock is f256, which is the main clock divided by 256, oscillator stop detection cannot be used.

The oscillator stop detection is provided to handle main clock stop caused by external factors. To stop the main clock oscillator by a program, i.e., to enter stop mode or to set the CM05 bit to 1 (main clock oscillator disabled), the CM20 bit in the CM2 register should be set to 0 (disable oscillator stop detection). To enter wait mode, this bit should be also set to 0.

The oscillator stop detection functions depending on the voltage of a capacitor which is being changed. In more concrete terms, this function detects that the oscillator is stopped when the main clock goes lower than approximately 500 kHz. Note that if the CM22 bit is set to 0 by a program in an interrupt handler while the frequency is around 500 kHz, a stack overflow may occur due to multiple interrupt requests.

7.3 Base Clock

The base clock is a reference clock for the CPU clock and peripheral bus clock. The base clock after a reset is the PLL clock divided by 6.

The base clock source is selected between the PLL clock and the low speed clocks which contain the sub clock (fC), on-chip oscillator clock divided by 4 (fOCO4), and main clock divided by 256 (f256).

If the PLL clock is selected, it is divided by 2, 3, 4, or 6 to become the base clock. If a low speed clock is selected, the clock itself can be the base clock.

The base clock source is set using the BCS bit in the CCR register and the divide ratio for the PLL clock is set using bits BCD1 and BCD0. Bits CM31 and CM30 in the CM3 register select a low speed clock.

7.4 CPU Clock and Peripheral Bus Clock

The CPU operating clock is referred to as the CPU clock. The CPU clock after a reset is the base clock divided by 2.

The CPU clock source is the base clock and the divide ratio is selected by setting bits CCD1 and CCD0 in the CCR register. The base clock divided by 2 to 4 becomes the peripheral bus clock. Its divide ratio is selected by setting bits PCD1 and PCD0 in the CCR register. The peripheral bus clock also functions as count source for the watchdog timer and operating clock for the serial bus interface and for the CAN module.

When the CPU becomes out of control, to prevent the CPU clock whose clock source is the PLL clock from stopping, the CM05 bit in the CM0 register should be set to 0 (main clock oscillator enabled) and the BCS bit in the CCR register should be set to 0 (PLL clock selected as base clock source). Then the following should be set.

(1) Set the PRC1 bit in the PRCR register to 1 (write enabled to the PM2 register).
(2) Set the PM21 bit in the PM2 register to 1 (clock change disabled).

7.5 Peripheral Clock

The peripheral clock is an operating clock or a count source for peripheral functions excluding the watchdog timer, the serial bus interface, and the CAN module. The source of this clock is generated by a clock, which has the same frequency as the PLL clock, divided by 2, 4, 6, or 8 according to the settings of bits PM36 and PM35 in the PM3 register. The peripheral clock is classified into three types of clock as follows:

(1) f1, f8, f32, f2n

f1, f8, and f32 are the peripheral clock sources divided by 1, 8, and 32, respectively. The clock source for f2n is selected between the peripheral clock source and the main clock by setting the PM26 bit in the PM2 register. The f2n divide ratio can be set using bits CNT3 to CNT0 in the TCSPR register (n = 1 to 15, not divided when n = 0).

f1, f8, f32, and f2n, whose clock source is the peripheral clock source, stop in low power mode or when the CM02 bit is set to 1 (peripheral clock source stopped in wait mode) to enter wait mode.

f1, f8, and f2n are used as a count source for timers A and B or an operating clock for the serial interface and the LIN module. f1 is used as an operating clock for the intelligent I/O as well. The f32 is used as an operating clock for the LIN module as well.

f8 and f32 can be output from the CLKOUT pin. Refer to 7.6 “Clock Output Function” for details.

(2) fAD

fAD, which has the same frequency as peripheral clock source, is an operating clock for the A/D converter.

This clock stops in low power mode or when the CM02 bit is set to 1 (peripheral clock source stopped in wait mode) to enter wait mode.

(3) fC32

fC32, which is a sub clock divided by 32, or on-chip oscillator clock divided by 128, is used as the count source for timers A and B. This clock is available when the sub clock or on-chip oscillator clock is active.

7.6 Clock Output Function

Low speed clocks, f8, and f32 can be output from the CLKOUT pin.
Table 7.3 lists the CLKOUT pin functions.

Table 7.3 CLKOUT Pin Functions

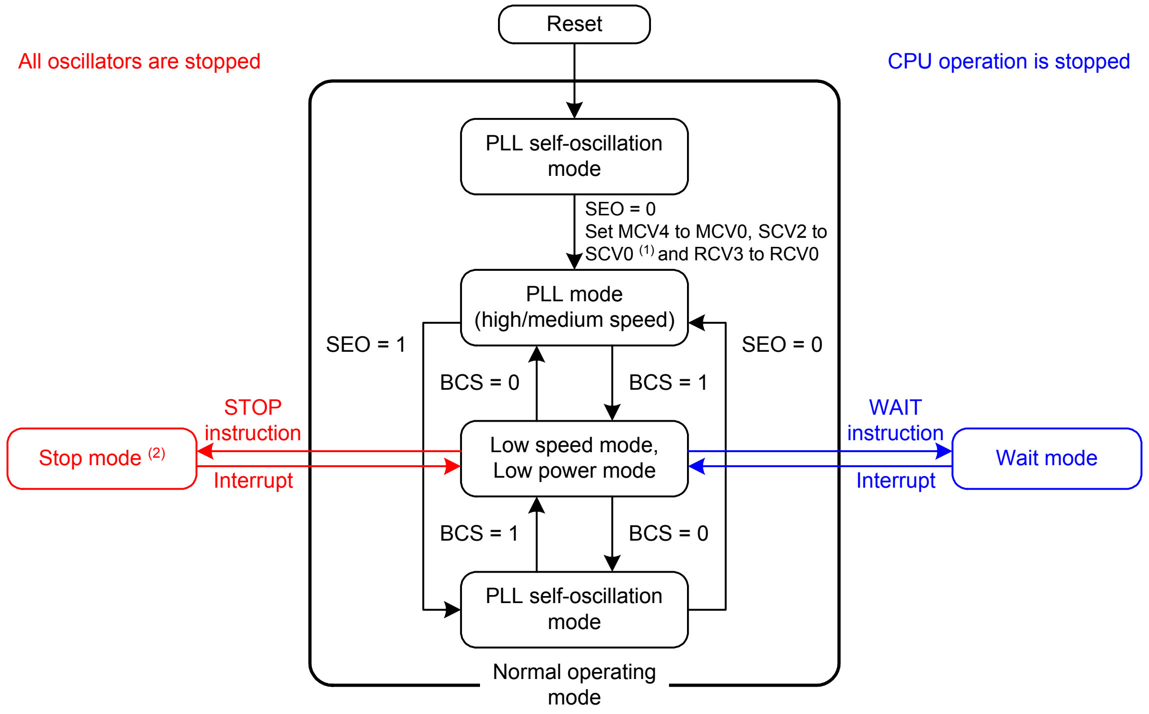
CM0 Register (1)		CLKOUT Pin Function
CM01	CM00	
0	0	I/O port P5_3
0	1	Output a low speed clock
1	0	Output f8
1	1	Output f32

Note:

1. This register should be rewritten after the PRC0 bit in the PRCR register is set to 1 (write enabled).

7.7 Power Control

Power control has three modes: wait mode, stop mode, and normal operating mode.
The name “normal operating mode” is used restrictively in this chapter, and it indicates all other modes except wait mode and stop mode. Figure 7.16 shows a block diagram of the state transition in normal operating mode, stop mode, and wait mode.



BCS: Bit in the CCR register
MCV4 to MCV0, SCV2 to SCV0: Bits in the PLC0 register
SEO, RCV3 to RCV0: Bits in the PLC1 register

Notes:

1. The PLC0 register can be set only once after a reset.
2. When the sub clock is selected as the base clock source, do not enter stop mode.

Figure 7.16 State Transition in Stop Mode and Wait Mode

7.7.1 Normal Operating Mode

Normal operating mode is classified into the five modes shown below.

In normal operating mode, the CPU clock and peripheral clock are provided to operate the CPU and peripheral functions. Power consumption is controlled by the CPU clock frequency. The higher the CPU clock frequency is, the more processing power increases. The lower the CPU clock frequency is, the less power consumption is required. Power consumption can be reduced by stopping oscillators that are not being used.

(1) PLL Mode (high speed mode)

In this mode, the PLL clock is selected as the base clock source, and the main clock is provided as the reference clock source for the PLL frequency synthesizer. High speed mode enables the CPU to operate at the maximum operating frequency. The PLL clock divided by 2 becomes the base clock. The base clock frequency should be identical to that of the CPU clock. fAD, f1, f8, f32, and f2n can be used as the peripheral clocks. When the sub clock or the on-chip oscillator clock is provided, fC32 can be used as the count source for timers A and B.

(2) PLL Mode (medium speed mode)

This mode indicates all modes in PLL mode except high speed mode. The PLL clock divided by 2, 3, 4, or 6 becomes the base clock and the base clock divided by 1 to 4 becomes the CPU clock. fAD, f1, f8, f32, and f2n can be used as the peripheral clocks. When the sub clock or the on-chip oscillator clock is provided, fC32 can be used as the count source for timers A and B.

(3) Low Speed Mode

In this mode, a low speed clock is used as the base clock source. The low speed clock becomes the base clock and the base clock divided by 1 to 4 becomes the CPU clock. fAD, f1, f8, f32, and f2n can be used as the peripheral clocks. When the sub clock or the on-chip oscillator clock is provided, fC32 can be used as the count source for timers A and B.

(4) Low Power Mode

This is a state where the main clock oscillator and the PLL frequency synthesizer are stopped after switching to low speed mode. The sub clock or the on-chip oscillator clock divided by 4 becomes the base clock and the base clock divided by 1 to 4 becomes the CPU clock. fC32, which is the only peripheral clock available, can be used as the count source for timers A and B. By setting the MRS bit in the VRCR register to 1 (main regulator stopped), this mode consumes even less power than the modes above.

(5) PLL Self-oscillation Mode

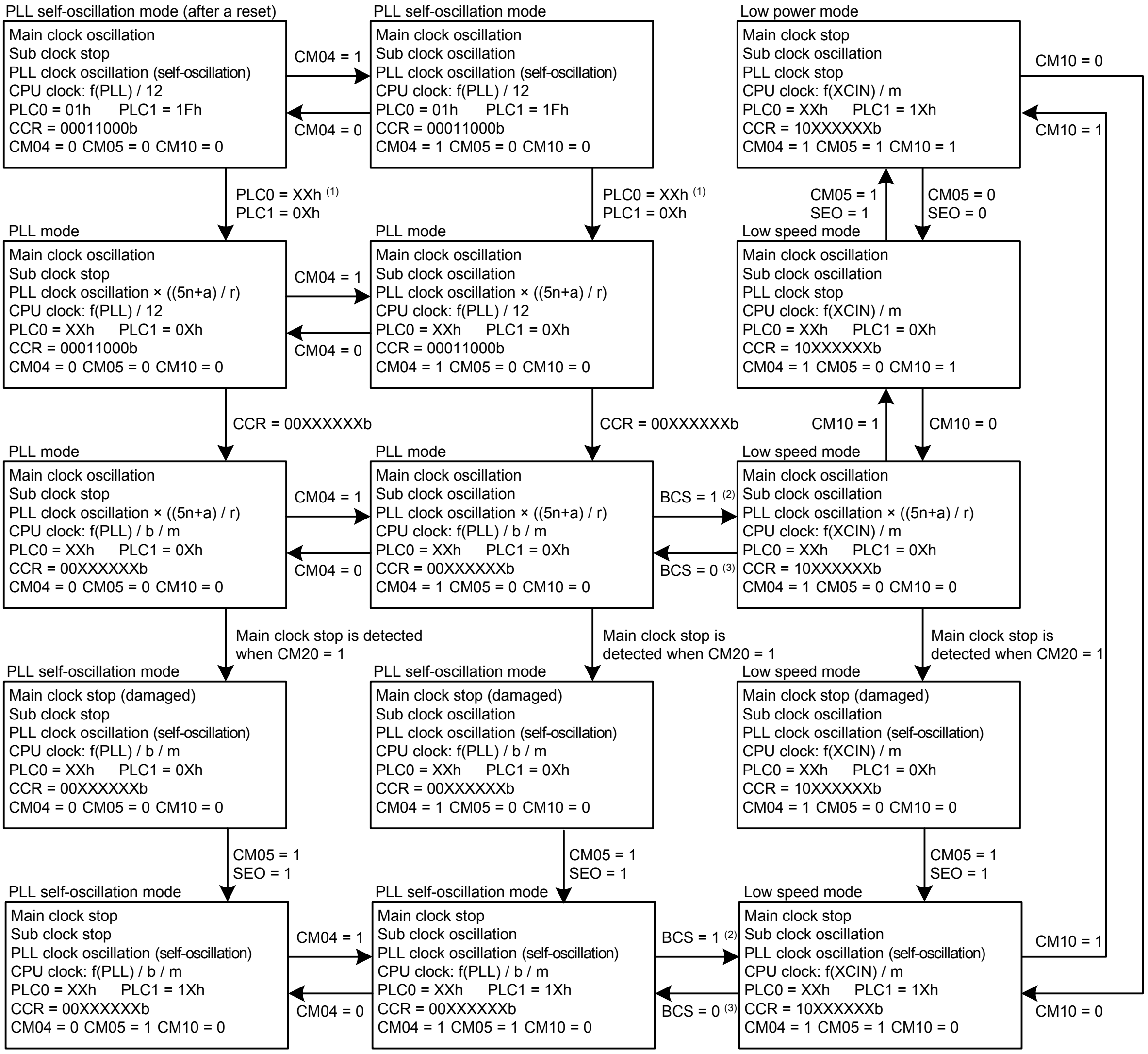
In this mode, the PLL clock is selected as the base clock source, and the main clock is not provided as the reference clock source for the PLL frequency synthesizer. The PLL frequency synthesizer self-oscillates at its own frequency. The PLL clock divided by 2, 3, 4, or 6 becomes the base clock and the base clock divided by 1 to 4 becomes the CPU clock. fAD, f1, f8, f32, and f2n can be used as the peripheral clocks. When the sub clock or the on-chip oscillator clock is provided, fC32 can be used as the count source for timers A and B.

The state transition within normal operating mode can be very complicated; therefore only the block diagrams of typical state transitions are shown. Figure 7.17 to Figure 7.19 show block diagrams of the respective state transitions: state when the sub clock is used, state when the main clock divided by 256 is used, and state when the on-chip oscillator clock is used. As for the state transitions other than the above, setting of each register and the usage notes below can be used as references.

- PLL can be switched from PLL oscillating to self-oscillating by setting the SEO bit in the PLC1 register to 1. Set the SEO bit to 1 (self-oscillating) before setting the CM05 bit in the CM0 register to 0 (main clock oscillator disabled) to stop the main clock.
- The divide ratio of the clock should be increased and the frequency should be decreased by using bits BCD1 to BCD0 in the CCR register or bits PM36 to PM35 in the PM3 register before setting the SEO bit to 0 (PLL oscillating) in order to switch back PLL self-oscillation mode to PLL mode. Set back the settings of bits BCD1 to BCD0 and bits PM36 to PM35 once PLL oscillation is stabilized after setting the SEO bit to 0.
- Before switching the CPU clock to another clock, that clock should be stabilized. In particular, the sub clock oscillator may require more time to stabilize (1). Therefore, certain waiting time to switch should be taken by a program immediately after turning the MCU on or exiting stop mode.

Note:

1. Contact the oscillator manufacturer for details on oscillator stabilization time.



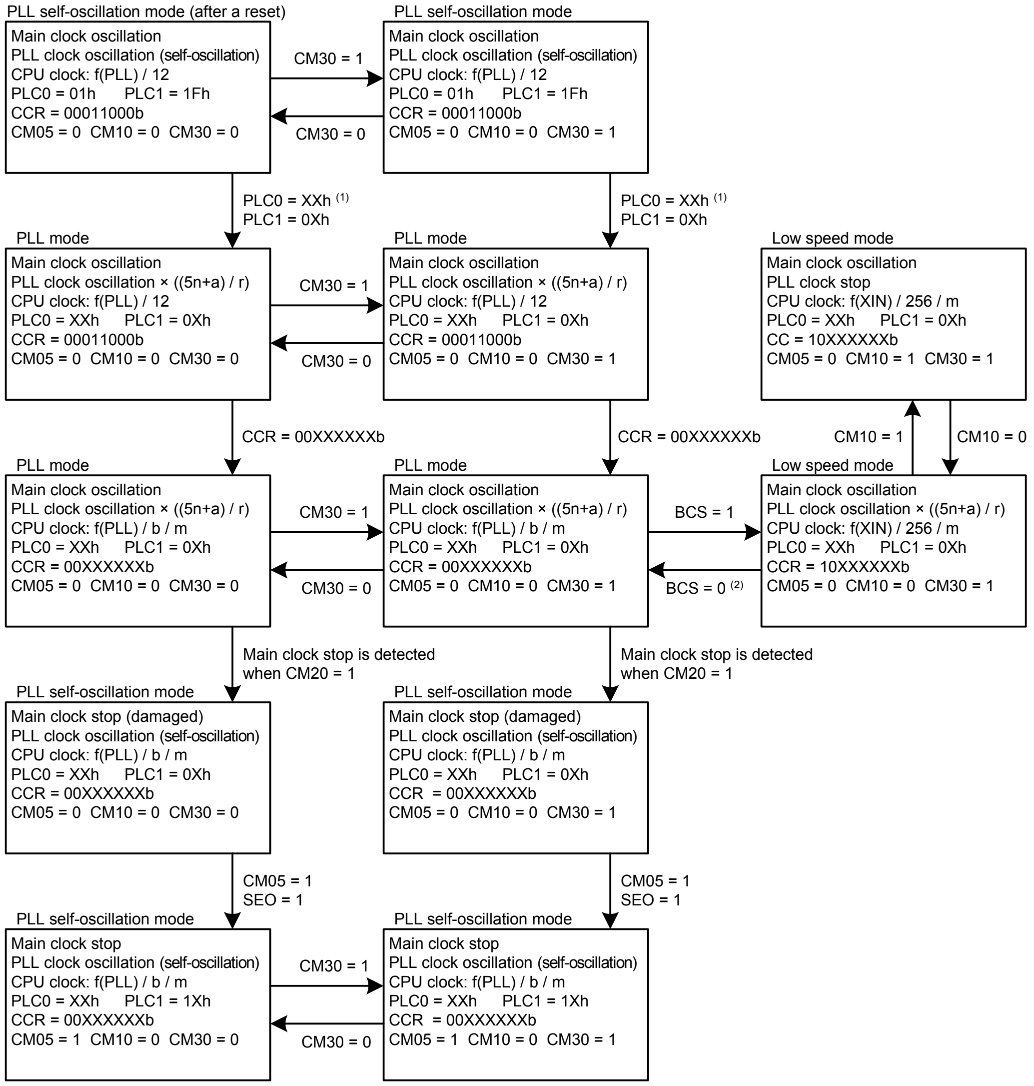
→ : Arrows indicate a one-way transition between modes. No transition should be made unless indicated.

BCS: Bit in the CCR register
CM04 and CM05: Bits in the CM0 register
CM10: Bit in the CM1 register
CM20: Bit in the CM2 register
SEO: Bit in the PLC1 register
PLC0 = XXh: The multiplication ratio of the PLL clock should be set using bits MCV4 to MCV0 and bits SCV2 to SCV0 in the PLC0 register.
PLC1 = 0Xh: The divisor of the reference clock should be set using bits RCV3 to RCV0 in the PLC1 register. The PLL clock frequency should not exceed the maximum value specified in the electrical characteristics.
CCR = 00XXXXXXb: The divisor of each clock should be set using the CCR register. The CPU clock frequency and the peripheral bus clock frequency should not exceed the maximum values specified in the electrical characteristics.

Notes:

1. The PLC0 register can be set only once after a reset.
2. This clock should be switched after the sub clock oscillation is fully stabilized.
3. This clock should be switched after the PLL clock oscillation is fully stabilized.

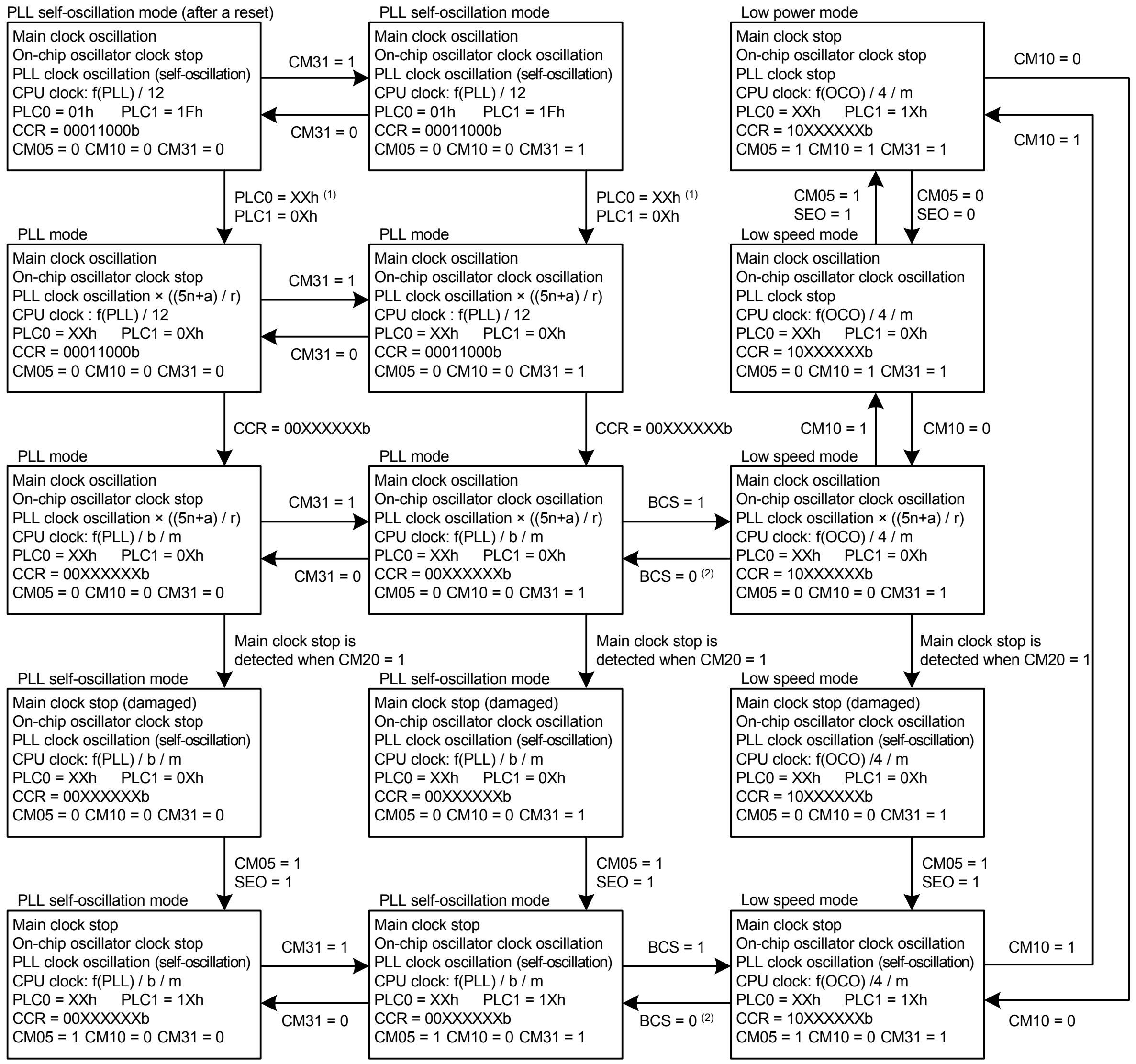
Figure 7.17 State Transition When Using the Sub Clock



→ : Arrows indicate a one-way transition between modes. No transition should be made unless indicated.
BCS: Bit in the CCR register
CM05: Bit in the CM0 register
CM10: Bit in the CM1 register
CM20: Bit in the CM2 register
CM30: Bit in the CM3 register
SEO: Bit in the PLC1 register
PLC0 = XXh: The multiplication ratio of the PLL clock should be set using bits MCV4 to MCV0 and bits SCV2 to SCV0 in the PLC0 register.
PLC1 = 0Xh: The divisor of the reference clock should be set using bits RCV3 to RCV0 in the PLC1 register. The PLL clock frequency should not exceed the maximum value specified in the electrical characteristics.
CCR = 00XXXXXXb: The divisor of each clock should be set using the CCR register. The CPU clock frequency and the peripheral bus clock frequency should not exceed the maximum values specified in the electrical characteristics.

Notes:
1. The PLC0 register can be set only once after a reset.
2. This clock should be switched after the PLL clock oscillation is fully stabilized.

Figure 7.18 State Transition When Using the Main Clock Divided by 256



⟶ : Arrows indicate a one-way transition between modes. No transition should be made unless indicated.
BCS: Bit in the CCR register
CM05: Bit in the CM0 register
CM10: Bit in the CM1 register
CM20: Bit in the CM2 register
CM31: Bit in the CM3 register
SEO: Bit in the PLC1 register
PLC0 = XXh: The multiplication ratio of the PLL clock should be set using bits MCV4 to MCV0 and bits SCV2 to SCV0 in the PLC0 register.
PLC1 = 0Xh: The divisor of the reference clock should be set using bits RCV3 to RCV0 in the PLC1 register. The PLL clock frequency should not exceed the maximum value specified in the electrical characteristics.
CCR = 00XXXXXXb: The divisor of each clock should be set using the CCR register. The CPU clock frequency and the peripheral bus clock frequency should not exceed the maximum values specified in the electrical characteristics.

Notes:

1. The PLC0 register can be set only once after a reset.
2. This clock should be switched after the PLL clock oscillation is fully stabilized.

Figure 7.19 State Transition When Using the On-chip Oscillator Clock

7.7.2 Wait Mode

The base clock stops in wait mode so that clocks generated by the base clock, the CPU clock and peripheral bus clock, stop running as well. Thus the CPU and watchdog timer, operated by these two clocks, also stop. However, the watchdog timer continues operating when the PM22 bit in the PM2 register is set to 1 (on-chip oscillator selected as count source for the watchdog timer). Since the main clock, sub clock, PLL clock, and on-chip oscillator clock continue running, peripheral functions using these clocks also continue operating.

7.7.2.1 Peripheral Clock Source Stop Function

When the CM02 bit in the CM0 register is set to 1 (peripheral clock source stopped in wait mode), power consumption is reduced since peripheral clocks f1, f8, f32, f2n (when the clock source is the peripheral clock source), and fAD stop running in wait mode. fC32 and f2n (when the clock source is the main clock) do not stop running.

7.7.2.2 Entering Wait Mode

To enter wait mode, the following procedures should be completed before the WAIT instruction is executed.

- Initial setting
  Set the wake-up interrupt priority level (bits RLVL2 to RLVL0 in registers RIPL1 and RIPL2) to 7. Then set each interrupt request level.
- Steps before entering wait mode
  (1) Set the I flag to 0.
  (2) Set the interrupt request level for each interrupt source (interrupt number from 1 to 127) to 0, if its interrupt request level is not 0.
  (3) Perform a dummy read of any of the interrupt control registers.
  (4) Set the processor interrupt priority level (IPL) in the flag register to 0.
  (5) Enable interrupts temporarily by executing the following instructions:
```
FSET I
NOP
NOP
FCLR I
```
  (6) Set the interrupt request level for the interrupt to exit wait mode.
  Do not rewrite the interrupt control register after this step.
  (7) Set the IPL in the flag register.
  (8) Set the interrupt priority level for resuming to the same level as the IPL.
  Interrupt request level for the interrupt to exit wait mode > IPL = Interrupt priority level for resuming
  (9) Set the CM20 bit in the CM2 register to 0 (disable oscillator stop detection) when the oscillator stop detection is used.
  (10)Enter either low speed mode or low power mode.
  (11)Set the I flag to 1.
  (12)Execute the WAIT instruction.
- After exiting wait mode
  Set the wake-up interrupt priority level to 7 immediately after exiting wait mode.

7.7.2.3 Pin State in Wait Mode

Table 7.4 lists the pin state in wait mode.

Table 7.4 Pin State in Wait Mode

Pin		State in Wait Mode
Ports		The state immediately before entering wait mode is held
DA0, DA1		The state immediately before entering wait mode is held
CLKOUT	When a low speed clock is selected	The clock is output
	When f8 or f32 is selected	The clock is output when the CM02 bit in the CM0 register is set to 0 (no peripheral clock source stopped in wait mode). The state immediately before entering wait mode is held when the CM02 bit is set to 1 (peripheral clock source stopped in wait mode)

7.7.2.4 Exiting Wait Mode

The MCU exits wait mode by a hardware reset, an NMI, or a peripheral interrupt assigned to software interrupt number from 0 to 63.

To exit wait mode using either a hardware reset or NMI, without using peripheral interrupts, bits ILVL2 to ILVL0 for the peripheral interrupts should be set to 000b (interrupt disabled) before executing the WAIT instruction.

The CM02 bit setting in the CM0 register affects the peripheral interrupts. When the CM02 bit is set to 0 (peripheral clock source not stopped in wait mode), peripheral interrupts for software interrupt numbers from 0 to 63 can be used to exit wait mode. When this bit is set to 1 (peripheral clock source stopped in wait mode), peripheral functions operated using clocks (f1, f8, f32, f2n whose clock source is the peripheral clock source, and fAD) generated by the peripheral clock source stop operating. Therefore, the peripheral interrupts cannot be used to exit wait mode. However, peripheral functions operated using clocks which are independent from the peripheral clock source (fC32, external clock, and f2n whose clock source is the main clock) do not stop operating. Thus, interrupts generated by peripheral functions and assigned to software interrupt numbers from 0 to 63 can be used to exit wait mode.

The CPU clock used when exiting wait mode by a peripheral interrupt or an NMI is the same clock used when the WAIT instruction is executed.

Table 7.5 lists interrupts used to exit wait mode and usage conditions.

Table 7.5 Interrupts for Exiting Wait Mode and Usage Conditions

Interrupt	When the CM02 Bit is 0	When the CM02 Bit is 1
NMI	Available	Available
External interrupt	Available	Available
Key input interrupt	Available	Available
Watchdog timer interrupt	Should not be used	Should not be used
Timer A interrupt Timer B interrupt	Available in any mode	Available in event counter mode, or when the count source is fC32 or f2n (when the main clock is selected as the clock source)
Serial interface interrupt (1)	Available when an internal or external clock is used	Available when the external clock or f2n (when the main clock is selected as the clock source) is used
A/D conversion interrupt	Available in single mode or single-sweep mode	Should not be used
Intelligent I/O interrupt	Available	Should not be used
LIN Low detection interrupt	Available	Available
CAN wake-up interrupt	Available	Available

Note:

1. UART3 and UART4 are excluded.

7.7.3 Stop Mode

In stop mode, all of the clocks, except for those that are protected, stop running. That is, the CPU and peripheral functions, operated by the CPU clock and peripheral clock, also stop. This mode saves the most power.

7.7.3.1 Entering Stop Mode

To enter stop mode, the following procedures should be done before the STOP instruction is executed.

- Initial setting
  Set the wake-up interrupt priority level (bits RLVL2 to RLVL0 in registers RIPL1 and RIPL2) to 7. Then set each interrupt request level.
- Steps before entering stop mode
  (1) Set the I flag to 0.
  (2) Set the interrupt request level for each interrupt source (interrupt number from 1 to 127) to 0, if the interrupt request level is not 0.
  (3) Perform a dummy read of any of the interrupt control registers.
  (4) Set the processor interrupt priority level (IPL) in the flag register to 0.
  (5) Enable interrupts temporarily by executing the following instructions:
  FSET I
  NOP
  NOP
  FCLR I
  (6) Set the interrupt request level for the interrupt to exit stop mode.
  Do not rewrite the interrupt control register after this step.
  (7) Set the IPL in the flag register.
  (8) Set the interrupt priority level for resuming to the same level as the IPL.
  Interrupt request level for the interrupt to exit stop mode > IPL = Interrupt priority level for resuming
  (9) Set the CM20 bit in the CM2 register to 0 (oscillator stop detection disabled) when the oscillator stop detection is used.
  (10)Change the base clock to either the main clock divided by 256 (f256) or the on-chip oscillator clock divided by 4 (fOCO4).
  (11)Set the I flag to 1.
  (12)Execute the STOP instruction.
- After exiting stop mode
  Set the wake-up interrupt priority level to 7 immediately after exiting stop mode.

7.7.3.2 Pin State in Stop Mode

Table 7.6 lists the pin state in stop mode.

Table 7.6 Pin State in Stop Mode

Pin		State in Stop Mode
Ports		The state immediately before entering stop mode is held
DA0, DA1		The state immediately before entering stop mode is held
CLKOUT	When a low speed clock is selected	High
	When f8 or f32 is selected	The state immediately before entering stop mode is held
XIN		High-impedance
XOUT		High
XCIN, XCOUT		High-impedance

7.7.3.3 Exiting Stop Mode

The MCU exits stop mode by a hardware reset, NMI, or a peripheral interrupt assigned to software interrupt number from 0 to 63.

To exit stop mode using either a hardware reset or NMI, without using peripheral interrupts, bits ILVL2 to ILVL0 for the peripheral interrupts should be set to 000b (interrupt disabled) before executing the STOP instruction.

The CPU clock used when exiting stop mode by a peripheral interrupt or NMI is the same clock used when the STOP instruction is executed.

Table 7.7 lists interrupts used to exit stop mode and usage conditions.

Table 7.7 Interrupts for Exiting Stop Mode and Usage Conditions

Interrupt	Usage Condition
NMI	
External interrupt	
Key input interrupt	
Timer A interrupt Timer B interrupt	Available when a timer counts an external pulse with a frequency of 100 Hz or less in event counter mode
Serial interface interrupt (1)	Available when an external clock is used
LIN Low detection interrupt	
CAN wake-up interrupt	

Note:

1. UART3 and UART4 are excluded.

7.8 System Clock Protection

The system clock protection disables clock change when the PLL clock is selected as the base clock source. This prevents the CPU clock from stopping due to a runaway program.
When the PM21 bit in the PM2 register is set to 1 (clock change disabled), the following bits cannot be written to:

- Bits CM02 and CM05 in the CM0 register
- The CM10 bit in the CM1 register
- The CM20 bit in the CM2 register
- The PM27 bit in the PM2 register

To use the system clock protection, the CM05 bit in the CM0 register should be set to 0 (main clock oscillator enabled) and the BCS bit in the CCR register should be set to 0 (PLL clock selected as base clock source) before the following procedure is done:

(1) The PRC1 bit in the PRCR register should be set to 1 (write to the PM2 register enabled).
(2) The PM21 bit in the PM2 register should be set to 1 (clock change disabled).
(3) The PRC1 bit in the PRCR register should be set to 0 (write to the PM2 register disabled).

7.9 Notes on Clock Generator

7.9.1 Sub Clock

7.9.1.1 Oscillator Constant Matching

The constant matching of the sub clock oscillator should be evaluated in both cases when the drive power is high and low.

Contact the oscillator manufacturer for details on the oscillation circuit constant matching.

7.9.2 Power Control

Do not switch the base clock source until the oscillation of the clock to be used has stabilized. However, this does not apply to the on-chip oscillator since it starts running immediately after the CM31 bit in the CM3 register is set to 1.

To switch the base clock source from the PLL clock to a low speed clock, use the MOV.L or OR.L instruction to set the BCS bit in the CCR register to 1.

- Program example in assembly language

```
        OR.L      #80h, 0004h
```

- Program example in C language

```
        asm("OR.L  #80h, 0004h");
```

7.9.2.1 Stop Mode

- To exit stop mode using a reset, apply a low signal to the $\overline{\text{RESET}}$ pin until the main clock oscillation stabilizes.

7.9.2.2 Suggestions for Power Saving

The followings are suggestions to reduce power consumption when programming or designing systems.

- I/O pins:
  If inputs are floating, both transistors may be conducting. Set unassigned pins to input mode and connect each of them to VSS via a resistor, or set them to output mode and leave them open.
- A/D converter:
  When not performing the A/D conversion, set the VCUT bit in the AD0CON1 register to 0 (VREF disconnected). To perform the A/D conversion, set the VCUT bit to 1 (VREF connected) and wait at least 1 μs before starting conversion.
- D/A converter:
  When not performing the D/A conversion, set the DAiE bit in the DACON register (i = 0, 1) to 0 (output disabled) and the DAi register to 00h.
- Peripheral clock stop:
  When entering wait mode, power consumption can be reduced by setting the CM02 bit in the CM0 register to 1 to stop the peripheral clock source. However, this setting does not stop the fC32.

8. Bus

This MCU has a fast bus (CPU bus) and a slow bus (peripheral bus). Figure 8.1 shows a block diagram of the bus.

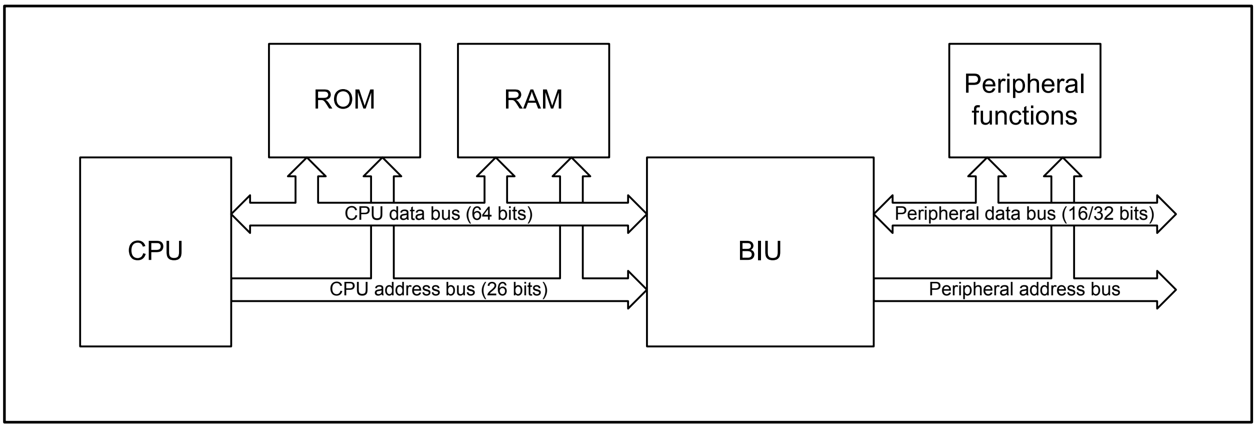


Figure 8.1 Bus Block Diagram

8.1 Bus Setting

The bus setting is controlled by the PBC register.
Table 8.1 lists the bus setting and its source.

Table 8.1 Bus Setting and Source

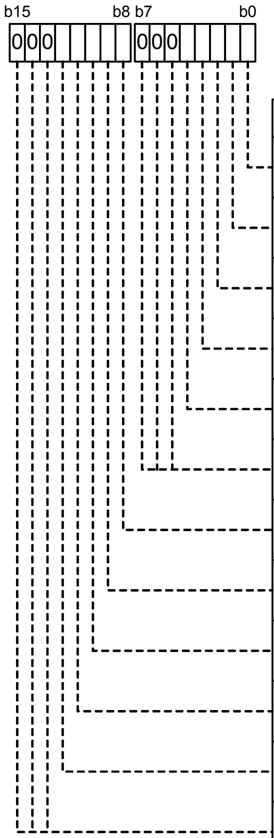
Bus Setting	Source
Internal SFR bus timing	PBC register

8.2 Peripheral Bus Timing Setting

The 16-/32-bit wide peripheral bus operates at a frequency up to 32 MHz (the theoretical value and the maximum frequency of each product group are as defined by f(BCLK) in 29. “Electrical Characteristics”). The timing adjustment and bus-width conversion with the faster, 64-bit wide CPU bus are controlled in the bus interface unit (BIU).

Figure 8.2 shows the PBC register which determines the peripheral bus timing.

Peripheral Bus Control Register (1, 2)



Symbol	Address	Reset Value
PBC	001Fh-001Eh	0504h

Bit Symbol	Bit Name	Function	RW
PRD0	Read Timing Setting Bit	Select from the three options below according to the peripheral bus clock setting (bits PCD1 and PCD0 in the CCR register). When bits PCD1 and PCD0 are set to: 1. 01b : 00100b 2. 10b : 01101b 3. 11b : 01111b	RW
PRD1			
PRD2			
PRD3			
PRD4			
— (b7-b5)	Reserved	Should be written with 0	RW
PWR0	Write Timing Setting Bit	Select from the three options below according to the peripheral bus clock setting (bits PCD1 and PCD0 in the CCR register). When bits PCD1 and PCD0 are set to: 1. 01b : 00101b 2. 10b : 01010b 3. 11b : 01111b	RW
PWR1			
PWR2			
PWR3			
PWR4			
— (b15-b13)	Reserved	Should be written with 0	RW

Notes:
1. Set the PRR register to AAh (write enabled) before rewriting this register.
2. This register should be set only once after a reset. It should not be rewritten after the CCR register is set.

Figure 8.2 PBC Register

9. Protection

This function protects important registers from being easily overwritten when a program goes out of control. Registers used to protect other registers from being rewritten are as follows: PRCR, PRCR2 to PRCR4, and PRR.

9.1 Protect Register (PRCR Register)

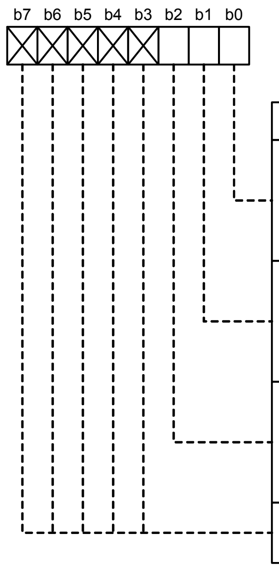
Figure 9.1 shows the PRCR register. Registers protected by bits in the PRCR register are listed in Table 9.1.

Table 9.1 Registers Protected by the PRCR Register

Bit	Protected Registers
PRC0	CM0, CM1, CM2, and PM3
PRC1	PM0, PM2, INVC0, INVC1, and IOBC
PRC2	PLC0, PLC1, PD9, and P9_iS (i = 3 to 7)

The PRC2 bit becomes 0 (write disabled) when a write operation is performed in a given address after this bit is set to 1 (write enabled). Set the PRC2 bit to 1 just before rewriting registers PD9, P9_iS (i = 3 to 7), PLC0, and PLC1. No interrupt handling or DMA transfers should be inserted between these two instructions. Bits PRC0 and PRC1 do not become 0 even if data is written to a given address. These bits should be set to 0 by a program.

Protect Register



Symbol	Address	Reset Value
PRCR	4004Ah	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PRC0	Protect Bit 0	Enable writing to registers CM0, CM1, CM2, and PM3 0: Write disabled 1: Write enabled	RW
PRC1	Protect Bit 1	Enable writing to registers PM0, PM2, INVC0, INVC1, and IOBC 0: Write disabled 1: Write enabled	RW
PRC2	Protect Bit 2 (1)	Enable writing to registers PLC0, PLC1, PD9, and P9_iS (i = 3 to 7) 0: Write disabled 1: Write enabled	RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

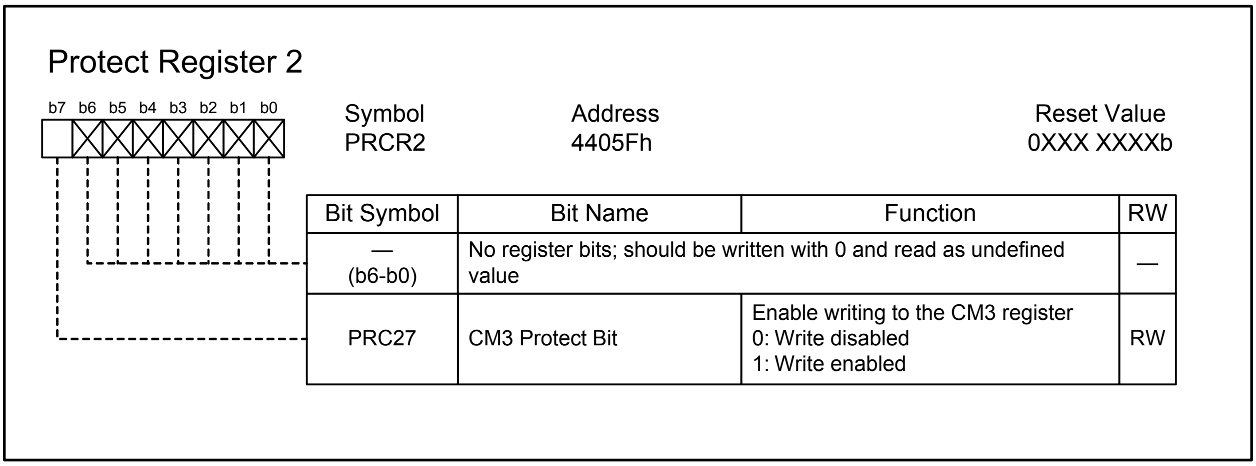
Note:

1. The PRC2 bit becomes 0 if a write operation is performed in a given address after this bit is set to 1. Bits PRC0 and PRC1 do not automatically become 0. They should be set to 0 by a program.

Figure 9.1 PRCR Register

9.2 Protect Register 2 (PRCR2 Register)

Figure 9.2 shows the PRCR2 register which protects the CM3 register only.



Protect Register 2

Symbol	Address	Reset Value
PRCR2	4405Fh	0XXX XXXXb

Bit Symbol	Bit Name	Function	RW
— (b6-b0)	No register bits; should be written with 0 and read as undefined value		—
PRC27	CM3 Protect Bit	Enable writing to the CM3 register 0: Write disabled 1: Write enabled	RW

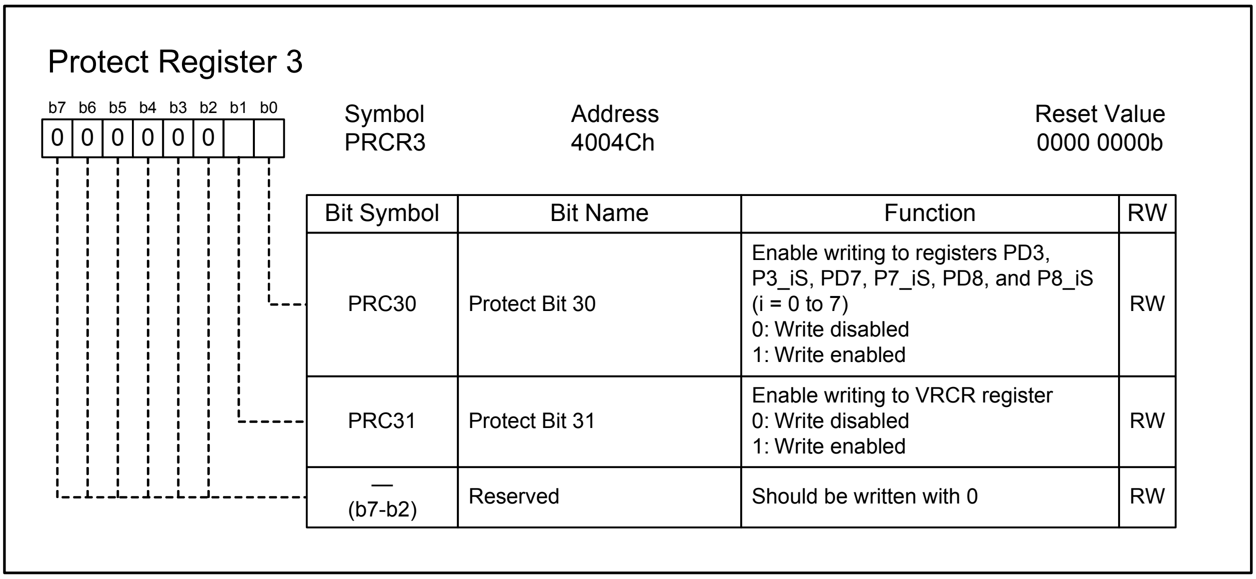
Figure 9.2 PRCR2 Register

9.3 Protect Register 3 (PRCR3 Register)

Figure 9.3 shows the PRCR3 register. Registers protected by the bits in the PRCR3 register are listed in Table 9.2.

Table 9.2 Registers Protected by the PRCR3 Register (i = 0 to 7)

Bit	Protected Registers
PRC30	PD3, P3_iS, PD7, P7_iS, PD8, and P8_iS
PRC31	VRCR



Protect Register 3

Symbol	Address	Reset Value
PRCR3	4004Ch	0000 0000b

Bit Symbol	Bit Name	Function	RW
PRC30	Protect Bit 30	Enable writing to registers PD3, P3_iS, PD7, P7_iS, PD8, and P8_iS (i = 0 to 7) 0: Write disabled 1: Write enabled	RW
PRC31	Protect Bit 31	Enable writing to VRCR register 0: Write disabled 1: Write enabled	RW
— (b7-b2)	Reserved	Should be written with 0	RW

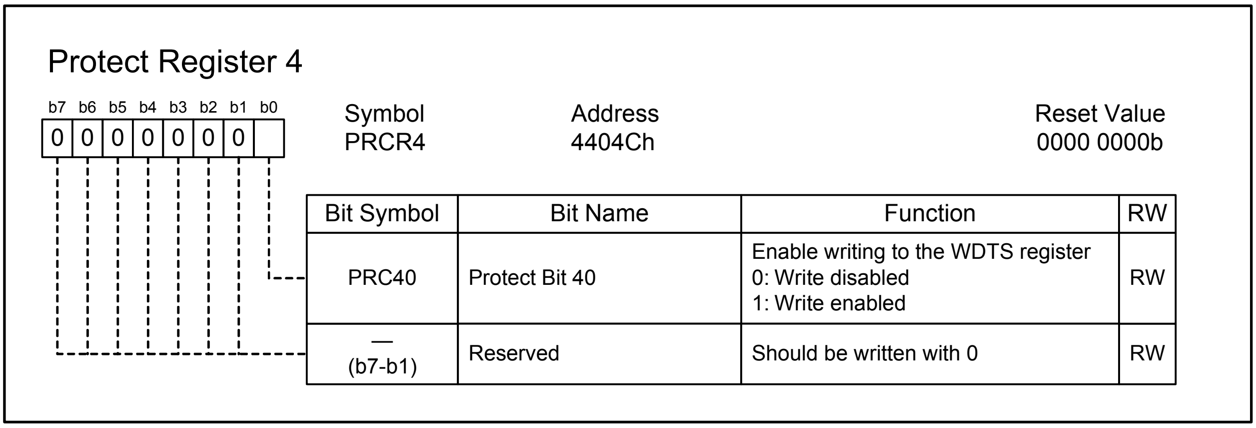
Figure 9.3 PRCR3 Register

9.4 Protect Register 4 (PRCR4 Register)

Figure 9.4 shows the PRCR4 register. Registers protected by the bits in the PRCR4 register are listed in Table 9.3.

Table 9.3 Registers Protected by the PRCR4 Register

Bit	Protected Registers
PRC40	WDTS



Protect Register 4

Symbol	Address	Reset Value
PRCR4	4404Ch	0000 0000b

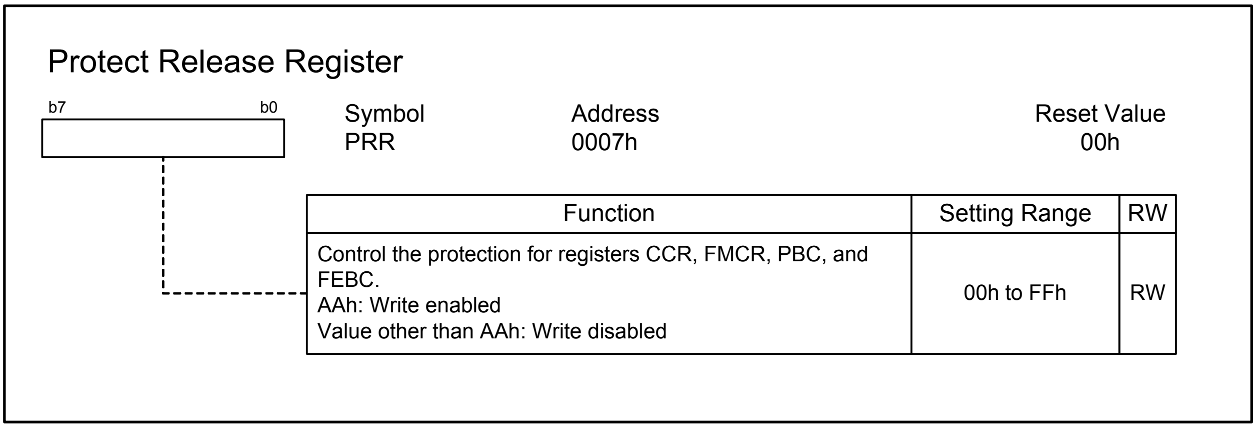
Bit Symbol	Bit Name	Function	RW
PRC40	Protect Bit 40	Enable writing to the WDTS register 0: Write disabled 1: Write enabled	RW
— (b7-b1)	Reserved	Should be written with 0	RW

Figure 9.4 PRCR4 Register

9.5 Protect Release Register (PRR Register)

Figure 9.5 shows the PRR register. Registers protected by the PRR register are as follows: CCR, FMCR, PBC, and FEBC.

To write to the registers above, the PRR register should be set to AAh (write enabled). Otherwise, the PRR register should be set to any value other than AAh to protect the above registers from unexpected write accesses.



Protect Release Register

Symbol	Address	Reset Value
PRR	0007h	00h

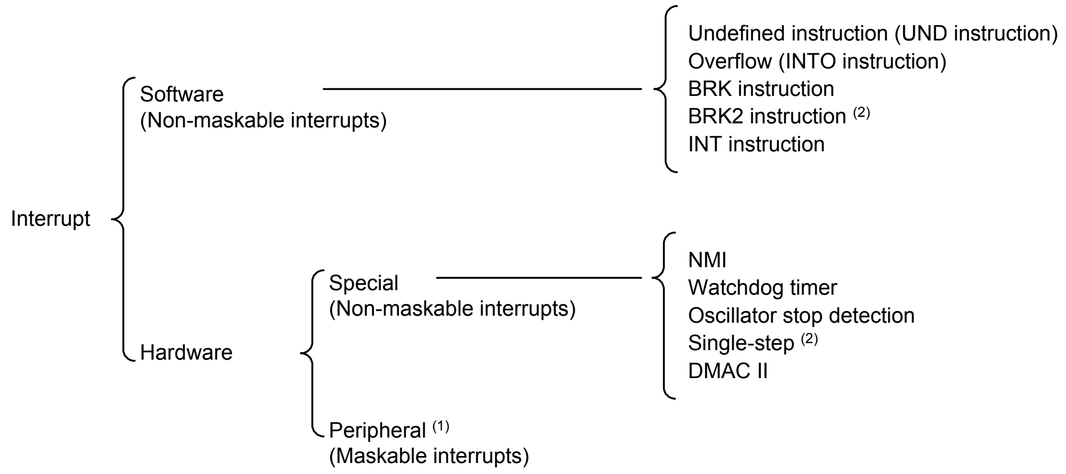
Function	Setting Range	RW
Control the protection for registers CCR, FMCR, PBC, and FEBC. AAh: Write enabled Value other than AAh: Write disabled	00h to FFh	RW

Figure 9.5 PRR Register

10. Interrupts

10.1 Interrupt Types

Figure 10.1 shows the types of interrupts.



Notes:
1. The peripheral interrupts are generated by the corresponding peripheral functions in the MCU.
2. This interrupt is used exclusively as a development support tool. Users are not allowed to use this interrupt.

Figure 10.1 Interrupts

Interrupts are also classified into maskable/non-maskable.

(1) Maskable Interrupts

Maskable interrupts can be disabled by the interrupt enable flag (I flag).
The priority can be configured by assigning an interrupt request level.

(2) Non-maskable Interrupts

Maskable interrupts cannot be disabled by the interrupt enable flag (I flag).
The interrupt priority cannot be configured.

10.2 Software Interrupts

Software interrupts are non-maskable. A software interrupt is generated by executing an instruction. There are five types of software interrupts shown below.

(1) Undefined Instruction Interrupt

This interrupt is generated when the UND instruction is executed.

(2) Overflow Interrupt

This interrupt is generated when the INTO instruction is executed while the O flag is 1. The following instructions may change the O flag to 1, depending on the operation result:
ABS, ADC, ADCF, ADD, ADDF, ADSF, CMP, CMPF, CNVIF, DIV, DIVF, DIVU, DIVX, EDIV, EDIVU, EDIVX, MUL, MULF, MULU, MULX, NEG, RMPA, ROUND, SBB, SCMPU, SHA, SUB, SUBF, SUNTIL, and SWHILE

(3) BRK Instruction Interrupt

This interrupt is generated when the BRK instruction is executed.

(4) BRK2 Instruction Interrupt

This interrupt is generated when the BRK2 instruction is executed.
This interrupt is only meant for use as a development support tool and users are not allowed to use it.

(5) INT Instruction Interrupt

This interrupt is generated when the INT instruction is executed with a selected software interrupt number from 0 to 255. Software interrupt numbers 0 to 127 are designated for peripheral interrupts. That is, the INT instruction with a software interrupt number from 0 to 127 has the same interrupt handler as that for peripheral interrupts.
The stack pointer (SP) used for this interrupt differs depending on the software interrupt numbers. For software interrupt numbers 0 to 127, when an interrupt request is accepted, the U flag is saved and set to 0 to select the interrupt stack pointer (ISP) during the interrupt sequence. The saved data of the U flag is restored upon returning from the interrupt handler. For software interrupt numbers 128 to 255, the stack pointer does not change during the interrupt sequence.

10.3 Hardware Interrupts

There are two kinds of hardware interrupts: special interrupts and peripheral interrupts.
In peripheral interrupts, only one interrupt with the highest priority can be specified as a fast interrupt.

10.3.1 Special Interrupts

Special interrupts are non-maskable. There are four special interrupts shown below.

(1) NMI (Non Maskable Interrupt)

This interrupt is generated if an input signal at the $\overline{\text{NMI}}$ pin switches from high to low. Refer to 10.11 “NMI” for details.

(2) Watchdog Timer Interrupt

The watchdog timer generates this interrupt. Refer to 11. “Watchdog Timer” for details.

(3) Oscillator Stop Detection Interrupt

This interrupt is generated if the MCU detects a main clock oscillator stop. Refer to 7.2 “Oscillator Stop Detection” for details.

(4) Single-step Interrupt

This interrupt is only meant for use as a development support tool and users are not allowed to use it.

10.3.2 Peripheral Interrupts

Peripheral interrupts occur when an interrupt request from a peripheral function in the MCU is accepted. They share the interrupt vector with software interrupt numbers 0 to 127 for the INT instruction.
Refer to Table 10.2 to Table 10.5 for details on the interrupt sources. Refer to the relevant descriptions for details on each function.

10.4 Fast Interrupt

A fast interrupt enables the CPU to accelerate interrupt response. In peripheral interrupts, only one interrupt with the highest priority can be specified as the fast interrupt.

Use the following procedure to enable a fast interrupt:

(1) Set the both FSIT bit in registers RIPL1 and RIPL2 to 1 (interrupt request level 7 available for fast interrupt).

(2) Set the both DMAII bit in registers RIPL1 and RIPL2 to 0 (interrupt request level 7 available for interrupts).

(3) Set the start address of the fast interrupt handler to the VCT register.

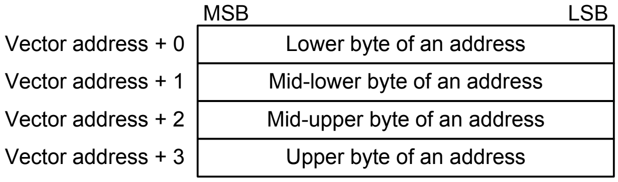
Under the conditions above, bits ILVL2 to ILVL0 in the interrupt control register should be set to 111b (level 7) to enable the fast interrupt. No other interrupts should be set to interrupt request level 7.

When the fast interrupt is accepted, the FLG register and the PC are saved to registers SVF and SVP, respectively. The program is executed from the address indicated by the VCT register.

To return from the fast interrupt handler, the FREIT instruction should be executed. The values saved into registers SVF and SVP are restored to the FLG register and the PC, respectively.

10.5 Interrupt Vectors

Each interrupt vector has a 4-byte memory space, in which the start address of the associated interrupt handler is stored. When an interrupt request is accepted, a jump to the address set in the interrupt vector takes place. Figure 10.2 shows an interrupt vector.



Vector address + 0	Lower byte of an address
Vector address + 1	Mid-lower byte of an address
Vector address + 2	Mid-upper byte of an address
Vector address + 3	Upper byte of an address

Figure 10.2 Interrupt Vector

10.5.1 Fixed Vector Table

The fixed vector table is allocated in addresses FFFFFFDCh to FFFFFFFFh. Table 10.1 lists the fixed vector table.

Table 10.1 Fixed Vector Table

Interrupt Source	Vector Addresses (Address (L) to Address (H))	Remarks	Reference
Undefined instruction	FFFFFFDCh to FFFFFFDFh	Interrupt by the UND instruction	R32C/100 Series Software Manual
Overflow	FFFFFFE0h to FFFFFFE3h	Interrupt by the INTO instruction	
BRK instruction	FFFFFFE4h to FFFFFFE7h	If address FFFFFFE7h is FFh, a jump to the interrupt vector of software interrupt number 0 in the relocatable vector table takes place	
—	FFFFFFE8h to FFFFFFEBh	Reserved	
—	FFFFFFECh to FFFFFFEFh	Reserved	
Watchdog timer Oscillator stop detection	FFFFFFF0h to FFFFFFF3h	These addresses are shared by the watchdog timer interrupt and oscillator stop detection interrupt	11. “Watchdog Timer” 7. “Clock Generator”
—	FFFFFFF4h to FFFFFFF7h	Reserved	
NMI	FFFFFFF8h to FFFFFFFBh	External interrupt by the $\overline{\text{NMI}}$ pin	
Reset	FFFFFFFCh to FFFFFFFFh		5. “Resets”

10.5.2 Relocatable Vector Table

The relocatable vector table occupies a 1024-byte memory space from the start address set in the INTB register. Table 10.2 to Table 10.5 list the relocatable vector table entries.

An address in a multiple of 4 should be set in the INTB register for a faster interrupt sequence.

Table 10.2 Relocatable Vector Table (1/4)

Interrupt Source	Vector Table Relative Addresses (Address (L) to Address (H)) (1)	Software Interrupt Number	Reference
BRK instruction (2)	+0 to +3 (0000h to 0003h)	0	R32C/100 Series Software Manual
Reserved	+4 to +7 (0004h to 0007h)	1	
Reserved	+8 to +11 (0008h to 000Bh)	2	17. “Serial Interface” (Reserved)
Reserved	+12 to +15 (000Ch to 000Fh)	3	
Reserved	+16 to +19 (0010h to 0013h)	4	
Reserved	+20 to +23 (0014h to 0017h)	5	
Reserved	+24 to +27 (0018h to 001Bh)	6	
Reserved	+28 to +31 (001Ch to 001Fh)	7	
DMA0 transfer complete	+32 to +35 (0020h to 0023h)	8	12. “DMAC”
DMA1 transfer complete	+36 to +39 (0024h to 0027h)	9	
DMA2 transfer complete	+40 to +43 (0028h to 002Bh)	10	
DMA3 transfer complete	+44 to +47 (002Ch to 002Fh)	11	
Timer A0	+48 to +51 (0030h to 0033h)	12	15.1 “Timer A”
Timer A1	+52 to +55 (0034h to 0037h)	13	
Timer A2	+56 to +59 (0038h to 003Bh)	14	
Timer A3	+60 to +63 (003Ch to 003Fh)	15	
Timer A4	+64 to +67 (0040h to 0043h)	16	
UART0 transmission, NACK (3)	+68 to +71 (0044h to 0047h)	17	17. “Serial Interface”
UART0 reception, ACK (3)	+72 to +75 (0048h to 004Bh)	18	
UART1 transmission, NACK (3)	+76 to +79 (004Ch to 004Fh)	19	
UART1 reception, ACK (3)	+80 to +83 (0050h to 0053h)	20	
Timer B0	+84 to +87 (0054h to 0057h)	21	15.2 “Timer B”
Timer B1	+88 to +91 (0058h to 005Bh)	22	
Timer B2	+92 to +95 (005Ch to 005Fh)	23	
Timer B3	+96 to +99 (0060h to 0063h)	24	
Timer B4	+100 to +103 (0064h to 0067h)	25	
INT5	+104 to +107 (0068h to 006Bh)	26	10.10 “External Interrupt”
INT4	+108 to +111 (006Ch to 006Fh)	27	
INT3	+112 to +115 (0070h to 0073h)	28	
INT2	+116 to +119 (0074h to 0077h)	29	
INT1	+120 to +123 (0078h to 007Bh)	30	
INT0	+124 to +127 (007Ch to 007Fh)	31	
Timer B5	+128 to +131 (0080h to 0083h)	32	15.2 “Timer B”

Notes:

1. Each entry is relative to the base address in the INTB register.
2. Interrupts from this source cannot be disabled by the I flag.
3. In I^{2}C mode, interrupts are generated by NACK, ACK, or detection of a start condition/stop condition.

Table 10.3 Relocatable Vector Table (2/4)

Interrupt Source	Vector Table Relative Addresses (Address (L) to Address (H)) (1)	Software Interrupt Number	Reference
UART2 transmission, NACK (2)	+132 to +135 (0084h to 0087h)	33	17. “Serial Interface”
UART2 reception, ACK (2)	+136 to +139 (0088h to 008Bh)	34	
Reserved	+140 to +143 (008Ch to 008Fh)	35	
Reserved	+144 to +147 (0090h to 0093h)	36	
Reserved	+148 to +151 (0094h to 0097h)	37	
Reserved	+152 to +155 (0098h to 009Bh)	38	
Start condition detection or stop condition detection (UART2) (2)	+156 to +159 (009Ch to 009Fh)	39	
Start condition detection or stop condition detection (UART0) (2)	+160 to +163 (00A0h to 00A3h)	40	
Start condition detection or stop condition detection (UART1) (2)	+164 to +167 (00A4h to 00A7h)	41	
A/D0	+168 to +171 (00A8h to 00ABh)	42	18. “A/D Converter”
Key input	+172 to +175 (00ACh to 00AFh)	43	10.12 “Key Input Interrupt”
Intelligent I/O interrupt 0	+176 to +179 (00B0h to 00B3h)	44	10.13 “Intelligent I/O Interrupt”, 22. “Intelligent I/O”
Intelligent I/O interrupt 1	+180 to +183 (00B4h to 00B7h)	45	
Intelligent I/O interrupt 2	+184 to +187 (00B8h to 00BBh)	46	
Intelligent I/O interrupt 3	+188 to +191 (00BCh to 00BFh)	47	
Intelligent I/O interrupt 4	+192 to +195 (00C0h to 00C3h)	48	
Intelligent I/O interrupt 5	+196 to +199 (00C4h to 00C7h)	49	
Intelligent I/O interrupt 6	+200 to +203 (00C8h to 00CBh)	50	
Intelligent I/O interrupt 7	+204 to +207 (00CCh to 00CFh)	51	
Intelligent I/O interrupt 8	+208 to +211 (00D0h to 00D3h)	52	
Intelligent I/O interrupt 9	+212 to +215 (00D4h to 00D7h)	53	
Intelligent I/O interrupt 10	+216 to +219 (00D8h to 00DBh)	54	
Intelligent I/O interrupt 11	+220 to +223 (00DCh to 00DFh)	55	
CAN4 wakeup (3)	+224 to +227 (00E0h to 00E3h)	56	25. “CAN Module”
CAN5 wakeup	+228 to +231 (00E4h to 00E7h)	57	
CAN0 wakeup (3)	+232 to +235 (00E8h to 00EBh)	58	25. “CAN Module”
CAN1 wakeup (3)	+236 to +239 (00ECh to 00EFh)	59	
CAN2 wakeup	+240 to +243 (00F0h to 00F3h)	60	
CAN3 wakeup	+244 to +247 (00F4h to 00F7h)	61	
LIN Low detection	+248 to +251 (00F8h to 00FBh)	62	24. “LIN Module”
Reserved	+252 to +255 (00FCh to 00FFh)	63	

Notes:

1. Each entry is relative to the base address in the INTB register.
2. In I^{2}C mode, interrupts are generated by NACK, ACK, or detection of a start condition/stop condition.
3. This is reserved in the R32C/142 Group.

Table 10.4 Relocatable Vector Table (3/4) (1)

Interrupt Source	Vector Table Relative Addresses (Address (L) to Address (H)) (2)	Software Interrupt Number	Reference
Serial bus interface 0	+256 to +259 (0100h to 0103h)	64	23. “Serial Bus Interface”
Serial bus interface 1	+260 to +263 (0104h to 0107h)	65	
Reserved	+264 to +267 (0108h to 010Bh)	66	
Reserved	+268 to +271 (010Ch to 010Fh)	67	
Reserved	+272 to +275 (0110h to 0113h)	68	
Reserved	+276 to +279 (0114h to 0117h)	69	
Reserved	+280 to +283 (0118h to 011Bh)	70	
Reserved	+284 to +287 (011Ch to 011Fh)	71	
CAN4 transmit FIFO (3)	+288 to +291 (0120h to 0123h)	72	25. “CAN Module”, 26. “CAN Gateway Module”
CAN4 receive FIFO/Gateway channel 4 (3)	+292 to +295 (0124h to 0127h)	73	
CAN5 transmit FIFO	+296 to +299 (0128h to 012Bh)	74	
CAN5 receive FIFO/Gateway channel 5	+300 to +303 (012Ch to 012Fh)	75	
Reserved	+304 to +307 (0130h to 0133h)	76	
Reserved	+308 to +311 (0134h to 0137h)	77	
Reserved	+312 to +315 (0138h to 013Bh)	78	
Reserved	+316 to +319 (013Ch to 013Fh)	79	
CAN0 transmit FIFO (3)	+320 to +323 (0140h to 0143h)	80	25. “CAN Module”, 26. “CAN Gateway Module”
CAN0 receive FIFO/Gateway channel 0 (3)	+324 to +327 (0144h to 0147h)	81	
CAN1 transmit FIFO (3)	+328 to +331 (0148h to 014Bh)	82	
CAN1 receive FIFO/Gateway channel 1 (3)	+332 to +335 (014Ch to 014Fh)	83	
CAN2 transmit FIFO	+336 to +339 (0150h to 0153h)	84	
CAN2 receive FIFO/Gateway channel 2	+340 to +343 (0154h to 0157h)	85	
CAN3 transmit FIFO	+344 to +347 (0158h to 015Bh)	86	
CAN3 receive FIFO/Gateway channel 3	+348 to +351 (015Ch to 015Fh)	87	
Reserved	+352 to +355 (0160h to 0163h)	88	
Gateway error	+356 to +359 (0164h to 0167h)	89	26. “CAN Gateway Module”
Reserved	+360 to +363 (0168h to 016Bh)	90	
Reserved	+364 to +367 (016Ch to 016Fh)	91	
Reserved	+368 to +371 (0170h to 0173h)	92	
Reserved	+372 to +375 (0174h to 0177h)	93	
Reserved	+376 to +379 (0178h to 017Bh)	94	
Reserved	+380 to +383 (017Ch to 017Fh)	95	

Notes:

1. Entries in this table cannot be used to exit wait mode or stop mode.
2. Each entry is relative to the base address in the INTB register.
3. This is reserved in the R32C/142 Group.

Table 10.5 Relocatable Vector Table (4/4) (1)

Interrupt Source	Vector Table Relative Addresses (Address (L) to Address (H)) (2)	Software Interrupt Number	Reference
CAN0 transmission (4)	+384 to +387 (0180h to 0183h)	96	25. “CAN Module”
CAN0 reception (4)	+388 to +391 (0184h to 0187h)	97	
CAN0 error (4)	+392 to +395 (0188h to 018Bh)	98	
CAN1 transmission (4)	+396 to +399 (018Ch to 018Fh)	99	
CAN1 reception (4)	+400 to +403 (0190h to 0193h)	100	
CAN1 error (4)	+404 to +407 (0194h to 0197h)	101	
CAN2 transmission	+408 to +411 (0198h to 019Bh)	102	
CAN2 reception	+412 to +415 (019Ch to 019Fh)	103	
CAN2 error	+416 to +419 (01A0h to 01A3h)	104	
CAN3 transmission	+420 to +423 (01A4h to 01A7h)	105	
CAN3 reception	+424 to +427 (01A8h to 01ABh)	106	
CAN3 error	+428 to +431 (01ACh to 01AFh)	107	
CAN4 transmission (4)	+432 to +435 (01B0h to 01B3h)	108	
CAN4 reception (4)	+436 to +439 (01B4h to 01B7h)	109	
CAN4 error (4)	+440 to +443 (01B8h to 01BBh)	110	
CAN5 transmission	+444 to +447 (01BCh to 01BFh)	111	
CAN5 reception	+448 to +451 (01C0h to 01C3h)	112	
CAN5 error	+452 to +455 (01C4h to 01C7h)	113	
Reserved	+456 to +459 (01C8h to 01CBh)	114	
Reserved	+460 to +463 (01CCh to 01CFh)	115	
LIN0	+464 to +467 (01D0h to 01D3h)	116	24. “LIN Module”
LIN1	+468 to +471 (01D4h to 01D7h)	117	
Reserved	+472 to +475 (01D8h to 01DBh)	118	
Reserved	+476 to +479 (01DCh to 01DFh)	119	
Reserved	+480 to +483 (01E0h to 01E3h)	120	
Reserved	+484 to +487 (01E4h to 01E7h)	121	
Reserved	+488 to +491 (01E8h to 01EBh)	122	
Reserved	+492 to +495 (01ECh to 01EFh)	123	
UART3 transmission	+496 to +499 (01F0h to 01F3h)	124	17. “Serial Interface”
UART3 reception	+500 to +503 (01F4h to 01F7h)	125	
UART4 transmission	+504 to +507 (01F8h to 01FBh)	126	
UART4 reception	+508 to +511 (01FCh to 01FFh)	127	
INT instruction (3)	+0 to +3 (0000h to 0003h) to +1020 to +1023 (03FCh to 03FFh)	0 to 255	10.2 “Software Interrupts”

Notes:

1. Entries in this table cannot be used to exit wait mode or stop mode.
2. Each entry is relative to the base address in the INTB register.
3. Interrupts from this source cannot be disabled by the I flag.
4. This is reserved in the R32C/142 Group.

10.6 Interrupt Request Acceptance

Software interrupts and special interrupts are accepted whenever their interrupt request is generated. Peripheral interrupts, however, are only accepted if the conditions below are met:

- I flag is 1
- IR bit is 1
- Bits ILVL2 to ILVL0 > IPL

The I flag, IPL, IR bit, and bits ILVL2 to ILVL0 do not affect each other. The I flag and IPL are in the flag register (FLG). The IR bit and bits ILVL2 to ILVL0 are in the interrupt control register.
The following section describes these flag and bits.

10.6.1 I Flag and IPL

The I flag (interrupt enable flag) enables or disables maskable interrupts. When the I flag is set to 1 (enabled), all maskable interrupts are enabled; when it is set to 0 (disabled), they are disabled. The I flag becomes 0 after a reset.
The IPL (processor interrupt priority level) consists of three bits and indicates eight interrupt priority levels from 0 to 7. An interrupt becomes acceptable when its interrupt request level is higher than the specified IPL (bits ILVL2 to ILVL0 > IPL).
Table 10.6 lists interrupt request levels classified by the IPL.

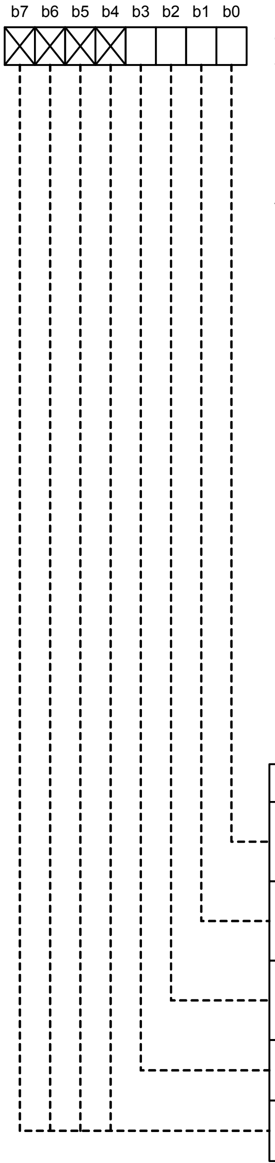
Table 10.6 Acceptable Interrupt Request Levels and IPL

IPL			Acceptable Interrupt Request Levels
IPL2	IPL1	IPL0	
1	1	1	All maskable interrupts are disabled
1	1	0	Level 7 only
1	0	1	Level 6 and above
1	0	0	Level 5 and above
0	1	1	Level 4 and above
0	1	0	Level 3 and above
0	0	1	Level 2 and above
0	0	0	Level 1 and above

10.6.2 Interrupt Control Registers

Each peripheral interrupt is controlled by an interrupt control register.
Figure 10.3 to Figure 10.5 show the interrupt control registers.

Interrupt Control Register



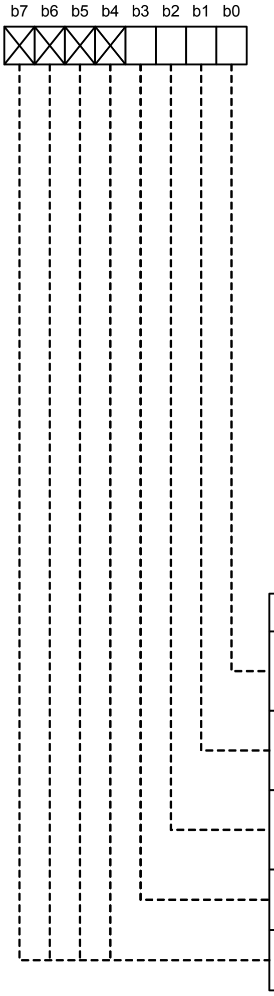
Symbol	Address	Reset Value
TA0IC to TA4IC	006Ch, 008Ch, 006Eh, 008Eh, 0070h	XXXX X000b
TB0IC to TB5IC	0094h, 0076h, 0096h, 0078h, 0098h, 0061h	XXXX X000b
S0TIC to S4TIC	0090h, 0092h, 0081h, 00DDh, 00DFh	XXXX X000b
S0RIC to S4RIC	0072h, 0074h, 0063h, 00FDh, 00FFh	XXXX X000b
BCN0IC to BCN2IC	0069h, 0089h, 0087h	XXXX X000b
DM0IC to DM3IC	0068h, 0088h, 006Ah, 008Ah	XXXX X000b
AD0IC	006Bh	XXXX X000b
KUPIC	008Bh	XXXX X000b
IIO0IC to IIO5IC	006Dh, 008Dh, 006Fh, 008Fh, 0071h, 0091h	XXXX X000b
IIO6IC to IIO11IC	0073h, 0093h, 0075h, 0095h, 0077h, 0097h	XXXX X000b
C2FTIC, C3FTIC	00D4h, 00D6h	XXXX X000b
C5FTIC	00CAh	XXXX X000b
C2FRIC, C3FRIC	00F4h, 00F6h (1)	XXXX X000b
C5FRIC	00EAh (1)	XXXX X000b
C2TIC, C3TIC	00C7h, 00E9h	XXXX X000b
C5TIC	00EFh	XXXX X000b
C2RIC, C3RIC	00E7h, 00CBh	XXXX X000b
C5RIC	00D1h	XXXX X000b
C2EIC, C3EIC	00C9h, 00EBh	XXXX X000b
C5EIC	00F1h	XXXX X000b
C2WIC, C3WIC	007Dh, 009Dh	XXXX X000b
C5WIC	0099h	XXXX X000b
GW2IC, GW3IC	00F4h, 00F6h (1)	XXXX X000b
GW5IC	00EAh (1)	XXXX X000b
GWEIC	00F8h	XXXX X000b
SS0IC, SS1IC	00C0h, 00E0h	XXXX X000b
L0IC, L1IC	00D5h, 00F5h	XXXX X000b
LLDIC	007Fh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
ILVL0	Interrupt Request Level Select Bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
ILVL1			RW
ILVL2			RW
IR	Interrupt Request Flag	0: No interrupt requested 1: Interrupt requested (2)	RW
— (b7-b4)	No register bits; should be written with 0 and read as undefined value		—

Notes:
1. CANi receive FIFO shares an address with the gateway channel i interrupt (i = 2, 3, 5).
2. This bit can only be set to 0 (do not set it to 1).

Figure 10.3 Interrupt Control Register (1/2) in the R32C/142 Group

Interrupt Control Register



Symbol	Address	Reset Value
TA0IC to TA4IC	006Ch, 008Ch, 006Eh, 008Eh, 0070h	XXXX X000b
TB0IC to TB5IC	0094h, 0076h, 0096h, 0078h, 0098h, 0061h	XXXX X000b
S0TIC to S4TIC	0090h, 0092h, 0081h, 00DDh, 00DFh	XXXX X000b
S0RIC to S4RIC	0072h, 0074h, 0063h, 00FDh, 00FFh	XXXX X000b
BCN0IC to BCN2IC	0069h, 0089h, 0087h	XXXX X000b
DM0IC to DM3IC	0068h, 0088h, 006Ah, 008Ah	XXXX X000b
AD0IC	006Bh	XXXX X000b
KUPIC	008Bh	XXXX X000b
IIO0IC to IIO5IC	006Dh, 008Dh, 006Fh, 008Fh, 0071h, 0091h	XXXX X000b
IIO6IC to IIO11IC	0073h, 0093h, 0075h, 0095h, 0077h, 0097h	XXXX X000b
C0FTIC to C5FTIC	00D0h, 00D2h, 00D4h, 00D6h, 00C8h, 00CFh	XXXX X000b
C0FRIC to C5FRIC	00F0h, 00F2h, 00F4h, 00F6h, 00E8h, 00EAh (1)	XXXX X000b
C0TIC to C5TIC	00C1h, 00E3h, 00C7h, 00E9h, 00CDh, 00EFh	XXXX X000b
C0RIC to C5RIC	00E1h, 00C5h, 00E7h, 00CBh, 00EDh, 00D1h	XXXX X000b
C0EIC to C5EIC	00C3h, 00E5h, 00C9h, 00EBh, 00CFh, 00F1h	XXXX X000b
C0WIC to C5WIC	007Bh, 009Bh, 007Dh, 009Dh, 0079h, 0099h	XXXX X000b
GW0IC to GW5IC	00F0h, 00F2h, 00F4h, 00F6h, 00E8h, 00EAh (1)	XXXX X000b
GWEIC	00F8h	XXXX X000b
SS0IC, SS1IC	00C0h, 00E0h	XXXX X000b
L0IC, L1IC	00D5h, 00F5h	XXXX X000b
LLDIC	007Fh	XXXX X000b

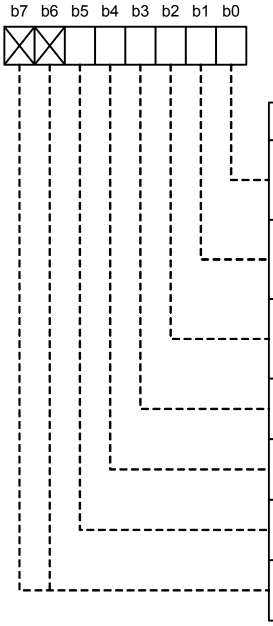
Bit Symbol	Bit Name	Function	RW
ILVL0	Interrupt Request Level Select Bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
ILVL1			RW
ILVL2			RW
IR	Interrupt Request Flag	0: No interrupt requested 1: Interrupt requested (2)	RW
— (b7-b4)	No registers bits; should be written with 0 and read as undefined value		—

Notes:

1. CANi receive FIFO shares an address with the gateway channel i interrupt (i = 0 to 5).
2. This bit can only be set to 0 (do not set it to 1).

Figure 10.4 Interrupt Control Register (1/2) in the R32C/145 Group

Interrupt Control Register



Symbol	Address	Reset Value
INT0IC to INT2IC	009Eh, 007Eh, 009Ch	XX00 X000b
INT3IC to INT5IC	007Ch, 009Ah, 007Ah	XX00 X000b

Bit Symbol	Bit Name	Function	RW
ILVL0	Interrupt Request Level Select Bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
ILVL1			RW
ILVL2			RW
IR	Interrupt Request Flag	0: No interrupt requested 1: Interrupt requested (1)	RW
POL	Polarity Select Bit	0: Select the falling edge or a low 1: Select the rising edge or a high (2)	RW
LVS	Level/Edge Sensitive Select Bit	0: Edge sensitive 1: Level sensitive (3)	RW
— (b7-b6)	No registers bits; should be written with 0 and read as undefined value		—

Notes:

1. This bit can only be set to 0 (do not set it to 1).
2. This bit should be set to 0 (the falling edge) to set the corresponding bit in the IFSR0 register to 1 (both edges).
3. To select the level sensitive, the corresponding bit in the IFSR0 register should be set to 0 (one edge).

Figure 10.5 Interrupt Control Register (2/2)

Bits ILVL2 to ILVL0

The interrupt request level is selected by setting bits ILVL2 to ILVL0. The higher the level is, the higher interrupt priority is.

When an interrupt request is generated, its request level is compared to the IPL. The interrupt is accepted only when the interrupt request level is higher than the IPL. When bits ILVL2 to ILVL0 are set to 000b, the interrupt is disabled.

IR bit

The IR bit becomes 1 (interrupt requested) when an interrupt request is generated; this bit setting is retained until the interrupt request is accepted. When the request is accepted and a jump to the corresponding interrupt vector takes place, the IR bit becomes 0 (no interrupt requested).

The IR bit can be set to 0 by a program. This bit should not be set to 1.

When rewriting the interrupt control register, no corresponding interrupt request should be generated. If there is a possibility that an interrupt request may be generated, disable all maskable interrupts before rewriting the register.
When enabling an interrupt immediately after changing the interrupt control register, insert NOPs between two instructions or perform a dummy read of the interrupt control register so that the interrupt enable flag (I flag) cannot become 1 (interrupt enabled) before writing to the interrupt control register is completed.
If an interrupt request is generated for the register being rewritten, the IR bit may not become 1 depending on the instruction being used. If it matters, use one of the following instructions to rewrite the register:

- AND
- OR
- BCLR
- BSET

If the AND or BCLR instruction is used to set the IR bit to 0, the IR bit may not become 0 as these instructions cause the interrupt request to be retained during the rewrite. To prevent this from happening, rewrite the register using the MOV instruction. To set just the IR bit to 0, first temporarily store the read value to memory or a CPU internal register, then execute either the AND or BCLR instruction in the stored area. After that, write the value back to the register using the MOV instruction.

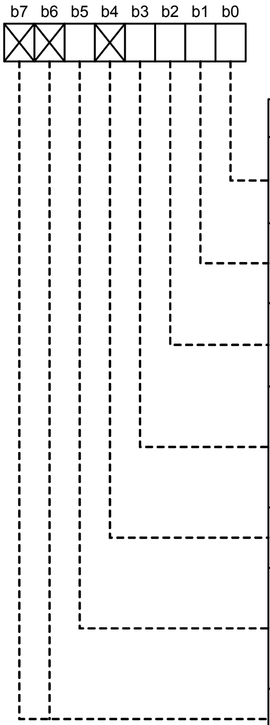
10.6.3 Wake-up IPL Setting Register

Set the wake-up IPL setting registers (registers RIPL1 and RIPL2) when using an interrupt to exit wait or stop mode, or using the fast interrupt.

Refer to 7.7.2 “Wait Mode”, 7.7.3 “Stop Mode”, or 10.4 “Fast Interrupt” for details.

Figure 10.6 shows registers RIPL1 and RIPL2.

Wake-up IPL Setting Register i (i = 1, 2) (1)



Symbol	Address	Reset Value
RIPL1, RIPL2	4407Fh, 4407Dh	XX0X 0000b

Bit Symbol	Bit Name	Function	RW
RLVL0	Interrupt Priority Level for Wake-up Select Bit (2)	b2 b1 b0 0 0 0 : Level 0 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
RLVL1			RW
RLVL2			RW
FSIT	Fast Interrupt Select Bit (3)	0: Use interrupt request level 7 for normal interrupt 1: Use interrupt request level 7 for fast interrupt (4)	RW
— (b4)	No register bit; should be written with 0 and read as undefined value		—
DMAII	DMA II Select Bit (5)	0: Use interrupt request level 7 for interrupt 1: Use interrupt request level 7 for DMA II transfer (4)	RW
— (b7-b6)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. Registers RIPL1 and RIPL2 should be set with the same values.
2. The MCU exits wait mode or stop mode if the request level of the requested interrupt is higher than the level selected using bits RLVL2 to RLVL0. These bits should be set to the same value as the IPL in the flag register (FLG).
3. When the FSIT bit is set to 1, an interrupt with interrupt request level 7 becomes the fast interrupt. In this case, set the interrupt request level to level 7 with only one interrupt.
4. Either the FSIT or DMAII bit should be set to 1. The fast interrupt and DMAC II cannot be used simultaneously.
5. Bits ILVL2 to ILVL0 in the interrupt control register should be set after the DMAII bit is set. DMA II transfer is not affected by the I flag or IPL.

Figure 10.6 Registers RIPL1 and RIPL2

10.6.4 Interrupt Sequence

An interrupt sequence is performed from when an interrupt request has been accepted until the interrupt handler starts.
When an interrupt request is generated while an instruction is being executed, the requested interrupt is evaluated in the priority resolver after the current instruction is completed, and the interrupt sequence starts from the next cycle.
However, for instructions RMPA, SCMPU, SIN, SMOVB, SMOVF, SMOVU, SOUT, SSTR, SUNTIL, and SWHILE, when an interrupt request is generated while an instruction is being executed, the current instruction is suspended, and the interrupt sequence starts.
The interrupt sequence is as follows:

(1) The CPU acknowledges the interrupt request to obtain the interrupt information (the interrupt number, and the interrupt request level) from the interrupt controller. Then the corresponding IR bit becomes 0 (no interrupt requested).
(2) The state of the flag register (FLG) before the interrupt sequence is stored to a temporary register in the CPU. The temporary register is inaccessible to users.
(3) The following bits in the flag register become 0:
  - The I flag (interrupt enable flag): interrupt disabled
  - The D flag (debug flag): single-step interrupt disabled
  - The U flag (stack pointer select flag): ISP selected
(4) The content of the temporary register in the CPU is saved to the stack, or to the save flag register (SVF) in case of the fast interrupt. Note that the temporary register is inaccessible to users.
(5) The content of the program counter (PC) is saved to the stack, or to the save PC register (SVP) in case of the fast interrupt.
(6) The interrupt request level for the accepted interrupt is set in the IPL (processor interrupt priority level).
(7) The corresponding interrupt vector is read from the interrupt vector table.
(8) This interrupt vector is stored into the program counter.

After the interrupt sequence is completed, an instruction is executed from the start address of the interrupt handler.

10.6.5 Interrupt Response Time

The interrupt response time, as shown in Figure 10.7, consists of two non-overlapping time segments: (a) the period from when an interrupt request is generated until the instruction being executed is completed; and (b) the period required for the interrupt sequence.

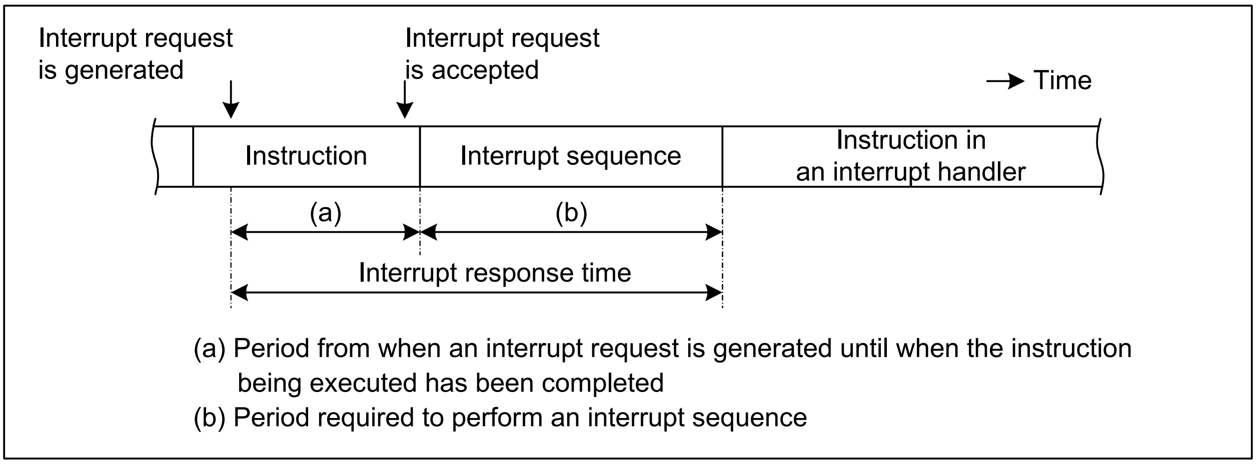


Figure 10.7 Interrupt Response Time

Period (a) varies depending on the instruction being executed. Instructions, such as LDCTX and STCTX in which registers are sequentially saved into or restored from the stack, require the longest time. For example, the STCTX instruction requires at least 30 cycles for 10 registers to be saved. It requires more time if the WAIT instruction is in the stack.

Period (b) is listed in Table 10.7.

Table 10.7 Interrupt Sequence Execution Time (1)

Interrupt	Execution Time in Terms of CPU Clock
Peripheral	13 + α cycles (2)
INT instruction	11 cycles
NMI	10 cycles
Watchdog timer Oscillator stop detection	11 cycles
Undefined instruction	12 cycles
Overflow	12 cycles
BRK instruction (relocatable vector table)	16 cycles
BRK instruction (fixed vector table)	19 cycles
BRK2 instruction	19 cycles
Fast interrupt	11 cycles

Notes:
1. These are the values when the interrupt vectors are aligned to the addresses in multiples of 4 in the internal ROM. However, the condition does not apply to the fast interrupt.
2. α is the number of waits to access SFRs minus 2.

10.6.6 IPL after Accepting an Interrupt Request

When a peripheral interrupt request is accepted, the interrupt request level is set in the IPL (processor interrupt priority level).

Software interrupts and special interrupts have no interrupt request level. When these interrupt requests are accepted, the value listed in Table 10.8 is set in the IPL as the interrupt request level.

Table 10.8 Interrupts without Interrupt Request Level and IPL

Interrupt Sources without Interrupt Request Level	IPL Value to be Set
NMI, watchdog timer, oscillator stop detection	7
Reset	0
Software	Unchanged

10.6.7 Register Saving

In the interrupt sequence, the flag register (FLG) and program counter (PC) values are saved to the stack, in that order. Figure 10.8 shows the stack status before and after an interrupt request is accepted.

In the fast interrupt sequence, the flag register and program counter values are saved to the save flag register (SVF) and save PC register (SVP), respectively.

If there are any other registers to be saved to the stack, save them at the beginning of the interrupt handler. A single PUSHM instruction saves all registers except the frame base register (FB) and stack pointer (SP).

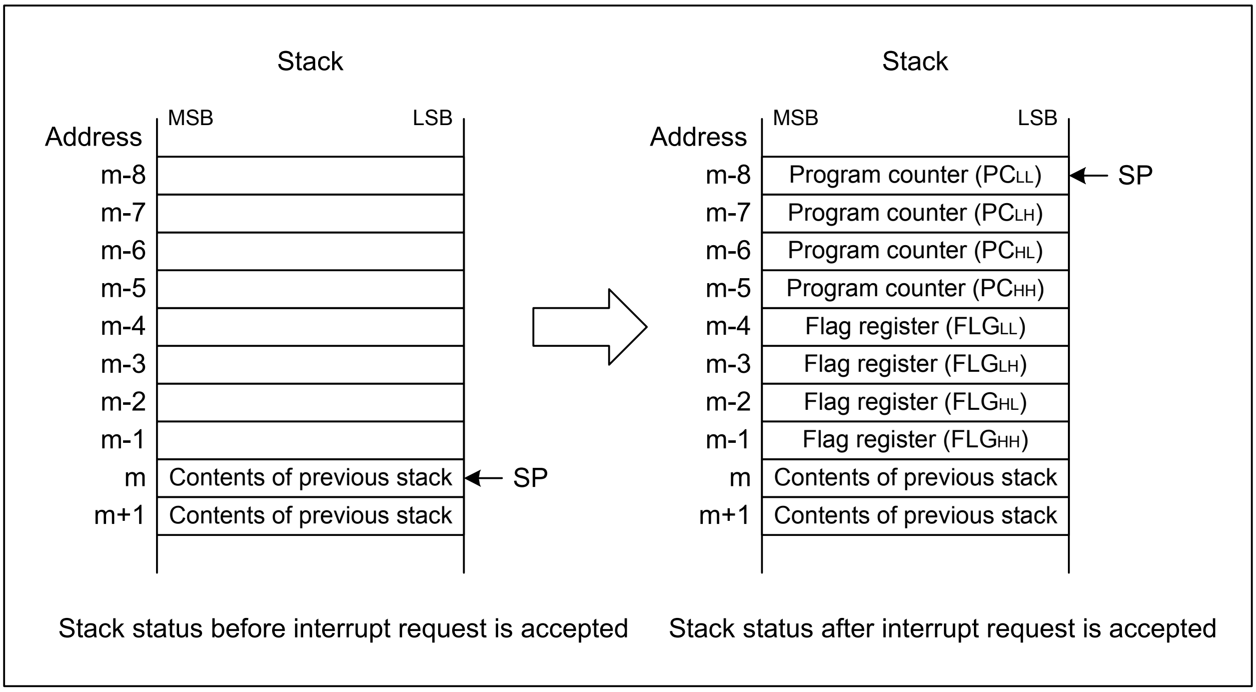


Figure 10.8 Stack Status Before and After an Interrupt Request is Accepted

10.7 Register Restoring from Interrupt Handler

When the REIT instruction is executed at the end of the interrupt handler, the values of the flag register (FLG) and the program counter (PC), which are saved in the stack, are restored, and the program resumes the operation that was interrupted. In the fast interrupt, execute the FREIT instruction to restore them from the save registers, instead.

To restore the register values which are saved by software in the interrupt handler, use an instruction such as POPM before the REIT or FREIT instruction.

If the register bank is switched in the interrupt handler, the bank is automatically switched back to the original register bank by the REIT or FREIT instruction.

10.8 Interrupt Priority

If two or more interrupt requests are detected at an interrupt request sampling point, the interrupt request with higher priority is accepted.

For maskable interrupts (peripheral interrupts), the interrupt request level select bits (bits ILVL2 to ILVL0) select a request level. If two or more interrupt requests have the same request level, the interrupt with higher priority, predetermined by hardware, is accepted.

The priorities of the reset and special interrupts, such as the watchdog timer interrupt, are determined by the hardware. Note that the reset has the highest priority. The following is the priority order of hardware interrupts:

$$\text{Reset} > \begin{matrix}\text{Watchdog timer}\\ \text{Oscillator stop detection}\end{matrix} > \text{NMI} > \text{Peripherals}$$

Software interrupts are not governed by priority. A jump to the interrupt handler takes place whenever the relevant instruction is executed.

10.9 Priority Resolver

The priority resolver selects an interrupt that has the highest priority among requested interrupts detected at the same sampling point.

Figure 10.9 and Figure 10.10 show the priority resolver of the R32C/142 Group and R32C/145 Group.

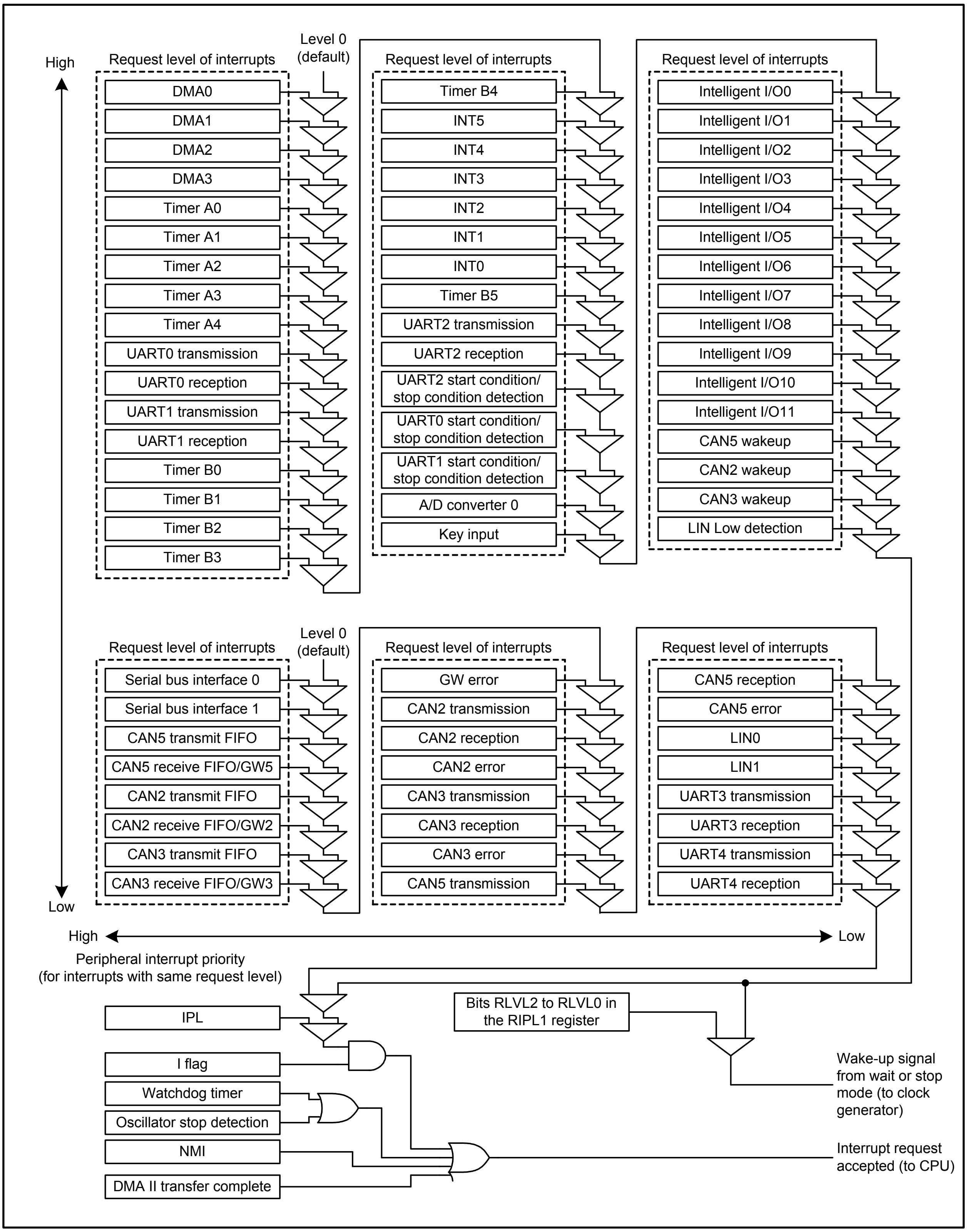


Figure 10.9 Priority Resolver of the R32C/142 Group

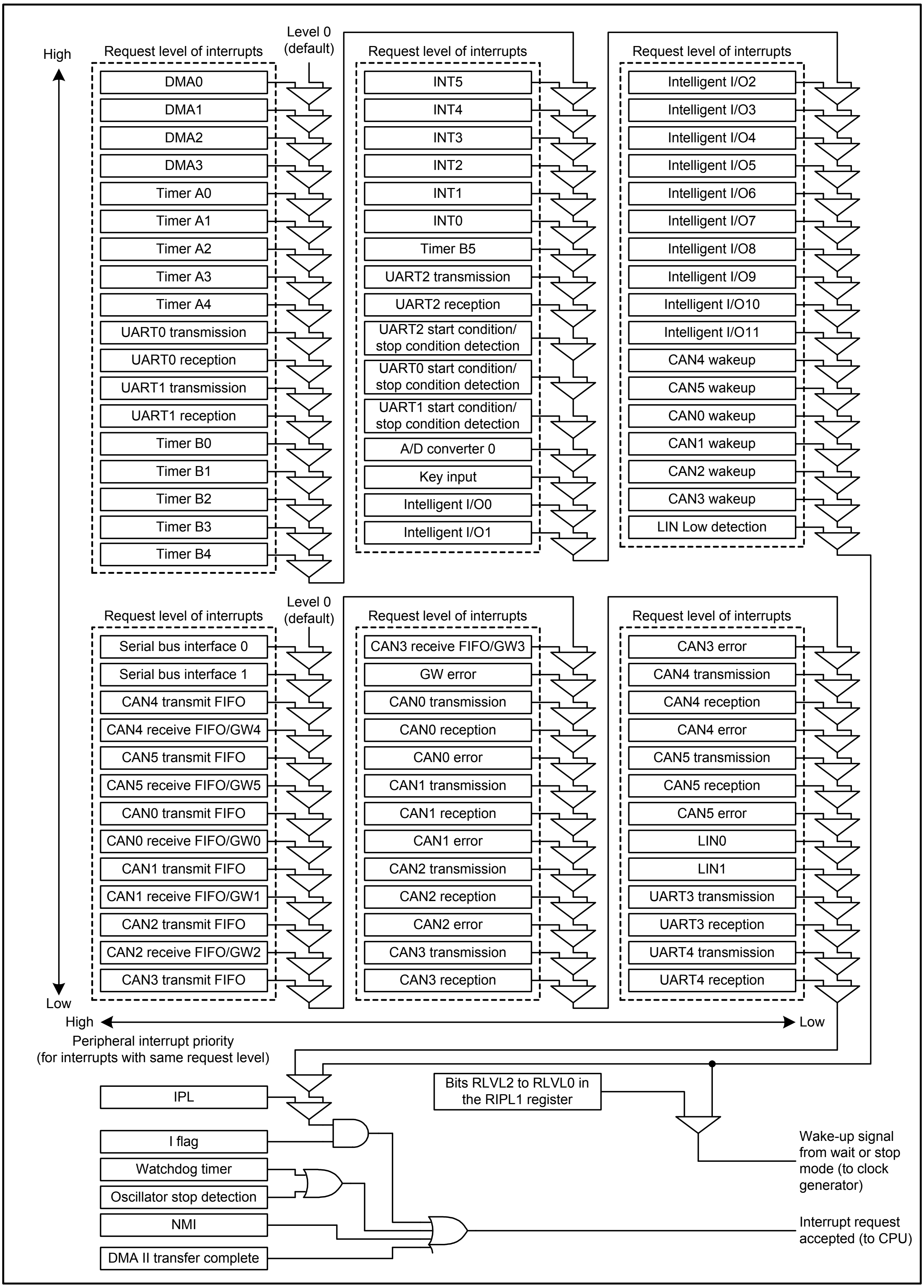


Figure 10.10 Priority Resolver of the R32C/145 Group

10.10 External Interrupt

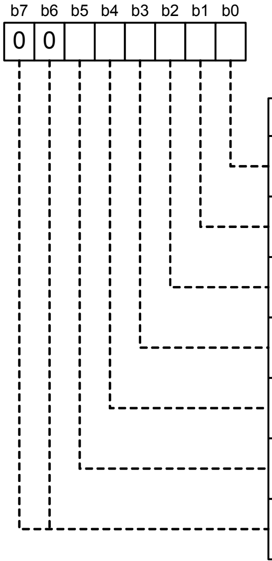
An external interrupt is generated by an external input applied to the $\overline{\mathrm{INTi}}$ pin (i = 0 to 5). Set the LVS bit in the INTiIC register to select whether an interrupt is triggered by the effective edge(s) (edge sensitive), or by the effective level (level sensitive) of the input signal. The polarity of the input signal is selected by setting the POL bit in the same register.

When using edge-triggered interrupts, setting the IFSR0i bit in the IFSR0 register to 1 (both edges) causes interrupt requests to be generated on both rising and falling edges of the external input. When the IFSR0i bit is set to 1, the POL bit in the corresponding register should be set to 0 (falling edge).

When using level-triggered interrupts, set the IFSR0i bit to 0 (one edge). When an effective level, which is selected by the POL bit, is detected on the $\overline{\mathrm{INTi}}$ pin, the IR bit in the INTiIC register becomes 1. The IR bit remains unchanged until the INTi interrupt is accepted, or set to 0 by a program, even if the signal level at the $\overline{\mathrm{INTi}}$ pin changes.

Figure 10.11 shows the IFSR0 register.

External Interrupt Request Source Select Register 0



Symbol	Address	Reset Value
IFSR0	4406Fh	0000 0000b

Bit Symbol	Bit Name	Function	RW
IFSR00	$\overline{\mathrm{INT0}}$ Pin Polarity Select Bit (1)	0: One edge 1: Both edges	RW
IFSR01	$\overline{\mathrm{INT1}}$ Pin Polarity Select Bit (1)	0: One edge 1: Both edges	RW
IFSR02	$\overline{\mathrm{INT2}}$ Pin Polarity Select Bit (1)	0: One edge 1: Both edges	RW
IFSR03	$\overline{\mathrm{INT3}}$ Pin Polarity Select Bit (1)	0: One edge 1: Both edges	RW
IFSR04	$\overline{\mathrm{INT4}}$ Pin Polarity Select Bit (1)	0: One edge 1: Both edges	RW
IFSR05	$\overline{\mathrm{INT5}}$ Pin Polarity Select Bit (1)	0: One edge 1: Both edges	RW
— (b7-b6)	Reserved	Should be written with 0	RW

Note:

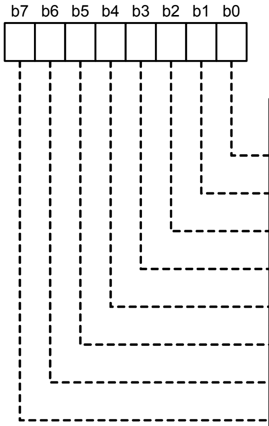
1. This bit should be set to 0 to select the level sensitive input as trigger. To set this bit to 1, the POL bit in the corresponding INTiIC register (i = 0 to 5) should be set to 0 (falling edge).

Figure 10.11 IFSR0 Register

The external interrupt signal has a digital filtering function for noise reduction. This function enables the interrupt controller to sample input signals with a selected clock and to pass pulses only which match the level three times sequentially.

Figure 10.12 and Figure 10.13 show registers INTF0 and INTF1.

External Interrupt Input Filter Select Register 0

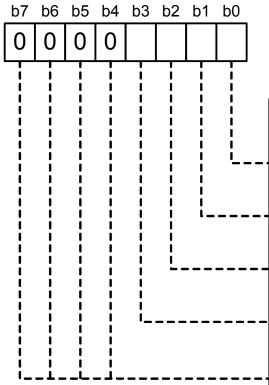


Symbol	Address	Reset Value
INTF0	44080h	0000 0000b

Bit Symbol	Bit Name	Function	RW
INT0F0	INT0 Input Filter Select Bit	Select enabled/disabled of filtering function and the sampling clock. b1 b0 b3 b2 b5 b4 b7 b6 0 0 : Filter is disabled 0 1 : Enabled, sample with f1 1 0 : Enabled, sample with f8 1 1 : Enabled, sample with f32	RW
INT0F1			RW
INT1F0	INT1 Input Filter Select Bit		RW
INT1F1			RW
INT2F0	INT2 Input Filter Select Bit		RW
INT2F1			RW
INT3F0	INT3 Input Filter Select Bit		RW
INT3F1			RW

Figure 10.12 INTF0 Register

External Interrupt Input Filter Select Register 1



Symbol	Address	Reset Value
INTF1	44082h	0000 0000b

Bit Symbol	Bit Name	Function	RW
INT4F0	INT4 Input Filter Select Bit	Select enabled/disabled of filtering function and the sampling clock. b1 b0 b3 b2 0 0 : Filter is disabled 0 1 : Enabled, sample with f1 1 0 : Enabled, sample with f8 1 1 : Enabled, sample with f32	RW
INT4F1			RW
INT5F0	INT5 Input Filter Select Bit		RW
INT5F1			RW
— (b7-b4)	Reserved	Should be written with 0	RW

Figure 10.13 INTF1 Register

10.11 NMI

The NMI (non maskable interrupt) is generated when an input signal at the $\overline{\text{NMI}}$ pin switches from high to low. This non maskable interrupt is disabled after a reset. To enable this interrupt, the PM24 bit in the PM2 register should be set to 1 after setting the interrupt stack pointer (ISP) at the beginning of the program. The $\overline{\text{NMI}}$ pin shares a pin with port P8_5, which enables the P8_5 bit in the P8 register to indicate the input level at the $\overline{\text{NMI}}$ pin.

Note:

1. When not using the NMI, do not change the reset value of the PM24 bit in the PM2 register.

10.12 Key Input Interrupt

The key input interrupt is enabled by setting ports P10_4 to P10_7 as input ports.

The interrupt request is generated if any of the signals applied to ports P10_4 to P10_7 switch from high to low. This interrupt also functions as key wake-up to exit wait or stop mode. Figure 10.14 shows a block diagram of the key input interrupt. If any of the ports are held low, signals applied to other ports are not detected as interrupt request signals.

To use the key input interrupt, every register from P10_4S to P10_7S should be set to 00h (I/O port) and bits PD10_4 to PD10_7 should be set to 0 (input). This is the only setting available for the key input interrupt.

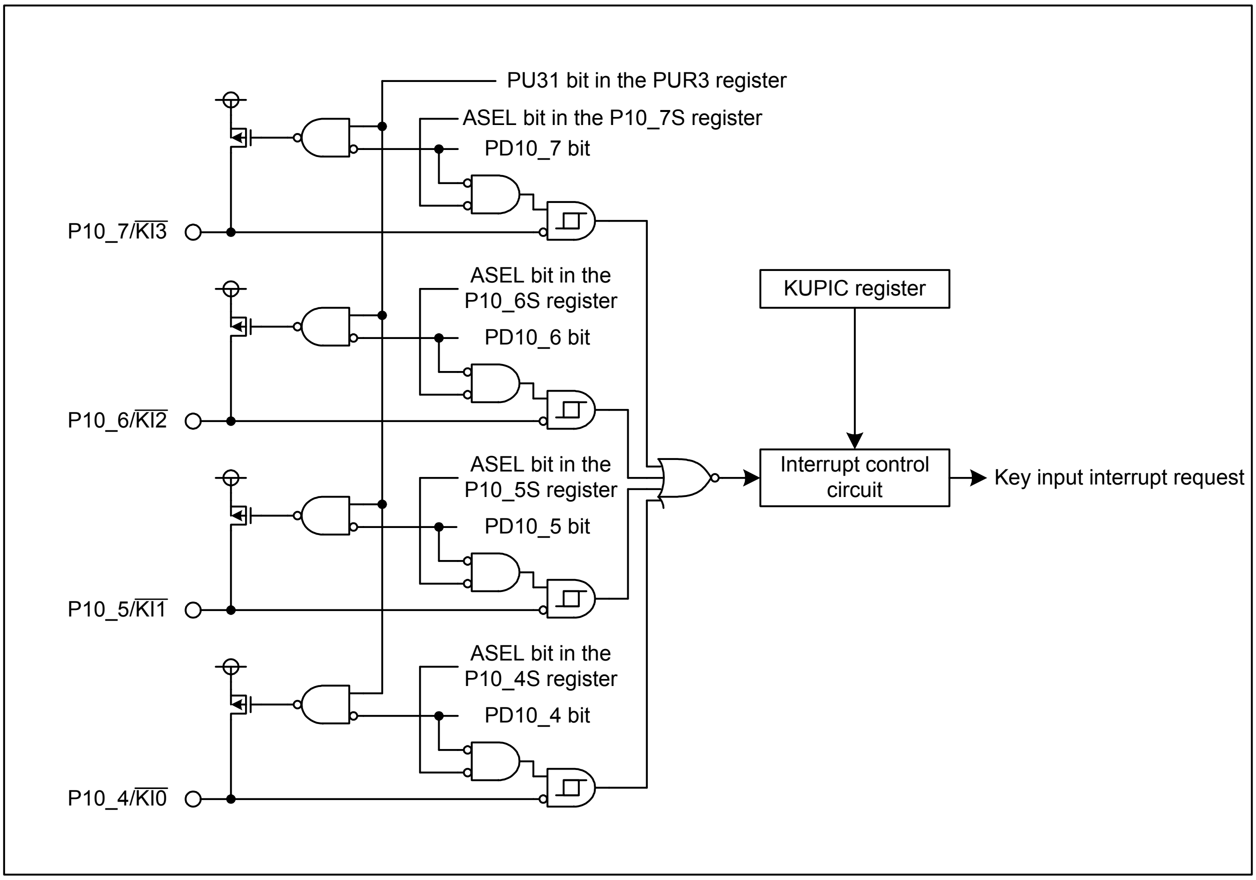


Figure 10.14 Key Input Interrupt Block Diagram

10.13 Intelligent I/O Interrupt

The intelligent I/O interrupt is assigned to software interrupt numbers 44 to 55.

Figure 10.15 shows a block diagram of the intelligent I/O interrupt. Figure 10.16 and Figure 10.17 show registers IIOiIR and IIOiIE (i = 0 to 11), respectively.

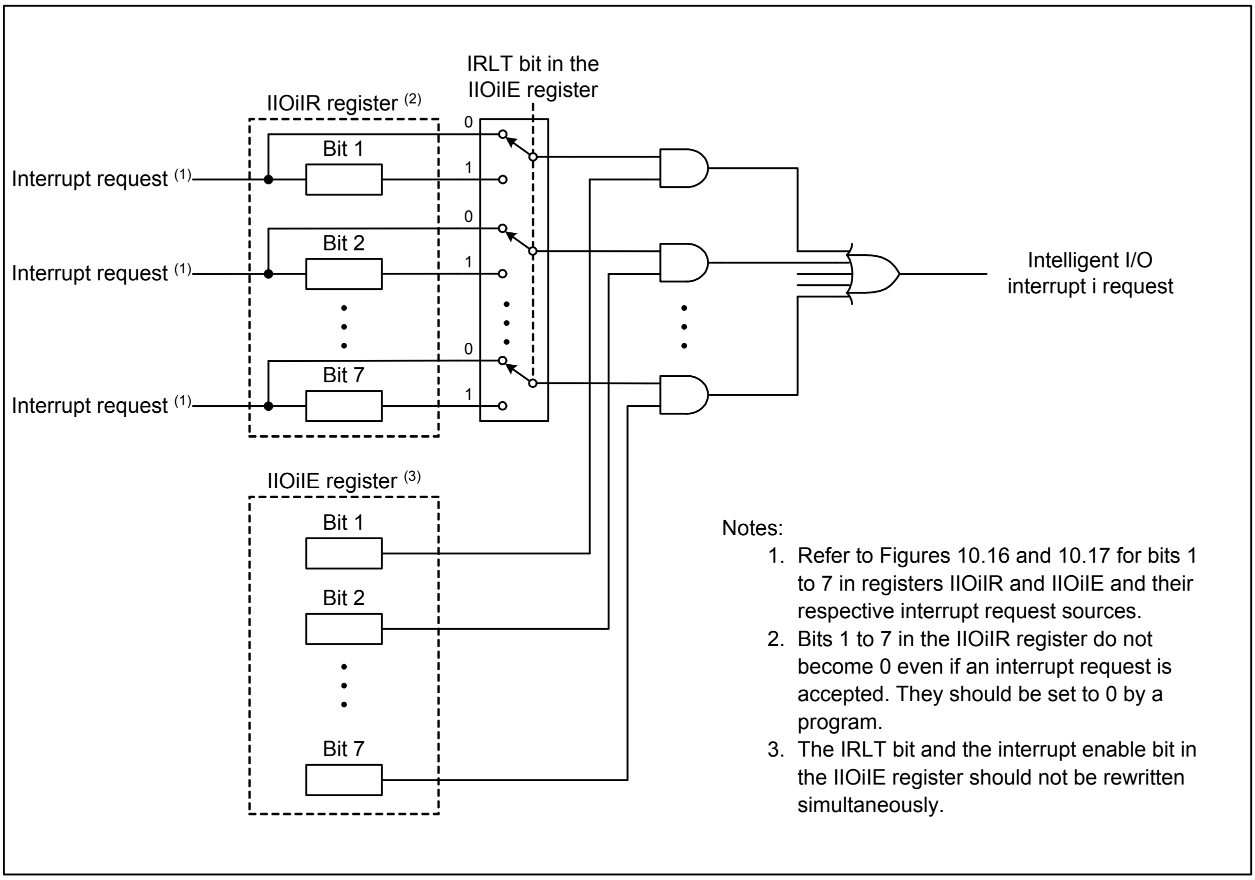
To use the intelligent I/O interrupt, the IRLT bit in the IIOiIE register should be set to 1 (interrupt requests used for interrupt).

The intelligent I/O interrupt has multiple request sources. When an interrupt request is generated with an intelligent I/O function, the corresponding bit in the IIOiIR register becomes 1 (interrupt requested). If the corresponding bit in the IIOiIE register is set to 1 (interrupt enabled), the IR bit in the corresponding IIOiIC register changes to 1 (interrupt requested).

After the IR bit setting changes from 0 to 1, it remains unchanged if a bit in the IIOiIR register becomes 1 by another interrupt request source and the corresponding bit in the IIOiIE register is 1.

Bits in the IIOiIR register do not become 0 even if an interrupt is accepted. They should be set to 0 by either the AND or BCLR instruction. Note that every generated interrupt request is ignored until these bits are set to 0.

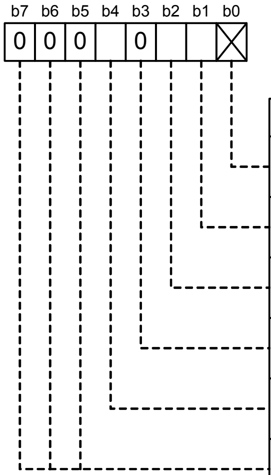
To use the intelligent I/O interrupt to activate DMAC II, the IRLT bit in the IIOiIE register should be set to 0 (interrupt requests used for DMA or DMA II) and the bit used for the interrupt source in the IIOiIE register should be set to 1 (interrupt enabled).



Notes:
1. Refer to Figures 10.16 and 10.17 for bits 1 to 7 in registers IIOiIR and IIOiIE and their respective interrupt request sources.
2. Bits 1 to 7 in the IIOiIR register do not become 0 even if an interrupt request is accepted. They should be set to 0 by a program.
3. The IRLT bit and the interrupt enable bit in the IIOiIE register should not be rewritten simultaneously.

Figure 10.15 Intelligent I/O Interrupt Block Diagram (i = 0 to 11)

Intelligent I/O Interrupt Request Register i (i = 0 to 11)



Symbol
IIO0IR to IIO11IR

Address
Refer to the table below

Reset Value
000? 0??1b (1)

Bit Symbol	Bit Name	Function	RW
— (b0)	No register bits; this bit is read as 1		—
(Note 2)	0: No interrupt requested 1: Interrupt requested (3)		RW
(Note 2)	0: No interrupt requested 1: Interrupt requested (3)		RW
— (b3)	Reserved	Should be written with 0	RW
(Note 2)	0: No interrupt requested 1: Interrupt requested (3)		RW
— (b7-b5)	Reserved	Should be written with 0	RW

Notes:

1. When the register has any function-assigned bit, the reset value is X (undefined); otherwise, the reset value is 0.
2. Refer to the table below for bit symbols.
3. When this bit is function-assigned, it can only be set to 0. It should not be set to 1. To set it to 0, either the AND or BCLR instruction should be used; when the bit is not function-assigned (reserved), it should be set to 0.

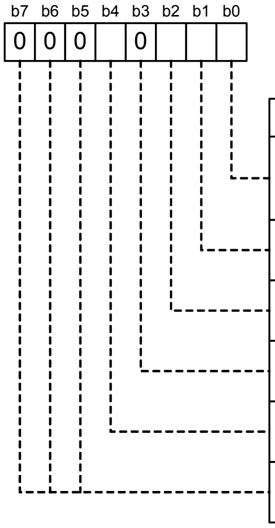
Bit Symbols for the Intelligent I/O Interrupt Request Register

Symbol	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IIO0IR	00A0h	—	—	—	—	—	TM13R/PO13R	TM02R/PO02R	—
IIO1IR	00A1h	—	—	—	—	—	TM14R/PO14R	TM00R/PO00R	—
IIO2IR	00A2h	—	—	—	—	—	TM12R/PO12R	—	—
IIO3IR	00A3h	—	—	—	—	—	TM10R/PO10R	TM03R/PO03R	—
IIO4IR	00A4h	—	—	—	BT1R	—	TM17R/PO17R	TM04R/PO04R	—
IIO5IR	00A5h	—	—	—	—	—	—	TM05R/PO05R	—
IIO6IR	00A6h	—	—	—	—	—	—	TM06R/PO06R	—
IIO7IR	00A7h	—	—	—	BT0R	—	—	TM07R/PO07R	—
IIO8IR	00A8h	—	—	—	—	—	—	TM11R/PO11R	—
IIO9IR	00A9h	—	—	—	—	—	—	TM15R/PO15R	—
IIO10IR	00AAh	—	—	—	—	—	—	TM16R/PO16R	—
IIO11IR	00ABh	—	—	—	—	—	—	TM01R/PO01R	—

BTxR: Intelligent I/O group x base timer interrupt request (x = 0, 1)
TMxyR: Intelligent I/O group x time measurement channel y interrupt request (x = 0, 1; y = 0 to 7)
POxyR: Intelligent I/O group x waveform generation channel y interrupt request (x = 0, 1; y = 0 to 7)

Figure 10.16 Registers IIO0IR to IIO11IR

Intelligent I/O Interrupt Enable Register i (i = 0 to 11)



Symbol	Address	Reset Value
IIO0IE to IIO11IE	Refer to the table below.	0000 0000b

Bit Symbol	Bit Name	Function	RW
IRLT	Interrupt Request Select Bit [(2)]	0: Use interrupt requests for DMA or DMA II 1: Use interrupt requests for interrupt	RW
(Note 1)	0: Disable the interrupt of bit 1 in the IIOiIR register 1: Enable the interrupt of bit 1 in the IIOiIR register		RW
(Note 1)	0: Disable the interrupt of bit 2 in the IIOiIR register 1: Enable the interrupt of bit 2 in the IIOiIR register		RW
— (b3)	Reserved	Should be written with 0	RW
(Note 1)	0: Disable the interrupt of bit 4 in the IIOiIR register 1: Enable the interrupt of bit 4 in the IIOiIR register		RW
— (b7-b5)	Reserved	Should be written with 0	RW

Notes:

1. Refer to the table below for bit symbols.
2. To use interrupt requests for interrupt, the IRLT bit should be set to 1, then bits 1, 2, and 4 should be set to 1.

Bit Symbols for the Intelligent I/O Interrupt Enable Register

Symbol	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IIO0IE	00B0h	—	—	—	—	—	TM13E/PO13E	TM02E/PO02E	IRLT
IIO1IE	00B1h	—	—	—	—	—	TM14E/PO14E	TM00E/PO00E	IRLT
IIO2IE	00B2h	—	—	—	—	—	TM12E/PO12E	—	IRLT
IIO3IE	00B3h	—	—	—	—	—	TM10E/PO10E	TM03E/PO03E	IRLT
IIO4IE	00B4h	—	—	—	BT1E	—	TM17E/PO17E	TM04E/PO04E	IRLT
IIO5IE	00B5h	—	—	—	—	—	—	TM05E/PO05E	IRLT
IIO6IE	00B6h	—	—	—	—	—	—	TM06E/PO06E	IRLT
IIO7IE	00B7h	—	—	—	BT0E	—	—	TM07E/PO07E	IRLT
IIO8IE	00B8h	—	—	—	—	—	—	TM11E/PO11E	IRLT
IIO9IE	00B9h	—	—	—	—	—	—	TM15E/PO15E	IRLT
IIO10IE	00BAh	—	—	—	—	—	—	TM16E/PO16E	IRLT
IIO11IE	00BBh	—	—	—	—	—	—	TM01E/PO01E	IRLT

BTxE: Intelligent I/O group x base timer interrupt enabled (x = 0, 1)
TMxyE: Intelligent I/O group x time measurement channel y interrupt enabled (x = 0, 1; y = 0 to 7)
POxyE: Intelligent I/O group x waveform generation channel y interrupt enabled (x = 0, 1; y = 0 to 7)

Figure 10.17 Registers IIO0IE to IIO11IE

10.14 Notes on Interrupts

10.14.1 ISP Setting

The interrupt stack pointer (ISP) is initialized to 00000000h after a reset. Set a value to the ISP before an interrupt is accepted, otherwise the program may go out of control. A multiple of 4 should be set to the ISP, which enables faster interrupt sequence due to less memory access.

When using NMI, in particular, since this interrupt cannot be disabled, the PM24 bit in the PM2 register should be set to 1 (NMI enabled) after the ISP is set at the beginning of the program.

10.14.2 NMI

- NMI cannot be disabled once the PM24 bit in the PM2 register is set to 1 (NMI enabled). This bit setting should be done only when using NMI.

- When the PM24 bit in the PM2 register is set to 1 (NMI enabled), the P8_5 bit in the P8 register is enabled just for monitoring the $\overline{\text{NMI}}$ pin state. It is not enabled as a general port.

10.14.3 External Interrupts

- The input signal to the $\overline{\text{INTi}}$ pin (i = 0 to 5) requires the pulse width specified in the electrical characteristics. If the pulse width is narrower than the specification, an external interrupt may not be accepted.

- When the effective level or edge of the $\overline{\text{INTi}}$ pin (i = 0 to 5) is changed by the following bits: bits POL, LVS in the INTiIC register, the IFSR0i bit (i = 0 to 5) in the IFSR0 register, the corresponding IR bit may become 1 (interrupt requested). When setting the above mentioned bits, preset bits ILVL2 to ILVL0 in the INTiIC register to 000b (interrupt disabled). After setting the above mentioned bits, set the corresponding IR bit to 0 (no interrupt requested), then rewrite bits ILVL2 to ILVL0.

11. Watchdog Timer

The watchdog timer is used to detect program runaway. The 15-bit watchdog counter decrements with the cycle which is the peripheral bus clock frequency or on-chip oscillator clock frequency divided by the prescaler.

Select either an interrupt request or a reset with the CM06 bit in the CM0 register for when the watchdog timer underflows. Once the CM06 bit is set to 1 (reset), it cannot be changed to 0 (watchdog timer interrupt) by a program. It can be set to 0 only by a reset.

The watchdog timer contains two types of prescaler: an on-chip oscillator clock prescaler and a general prescaler. The former is the on-chip oscillator clock divided by 1, 2, 4 or 8. The divide ratio is selected by setting bits WDK3 and WDK2 in the WKD register. The latter is the peripheral bus clock divided by 16 or 128. The divide ratio is selected by setting the WDC7 bit in the WDC register.

The count source for the watchdog timer is set by the PM22 bit in the PM2 register. When the peripheral bus clock is selected as the count source, the watchdog timer is stopped in wait mode, stop mode, or when the $\overline{\text{HOLD}}$ signal is driven low. It resumes counting from the value held when exiting the mode or state. When the on-chip oscillator clock is selected, the watchdog timer does not stop.

The general formula to calculate a watchdog timer period is:

$$\text{Watchdog timer period} = \frac{\text{Prescaler divisor (16 or 128)} \times 32768}{\text{Peripheral bus clock frequency}}$$

or

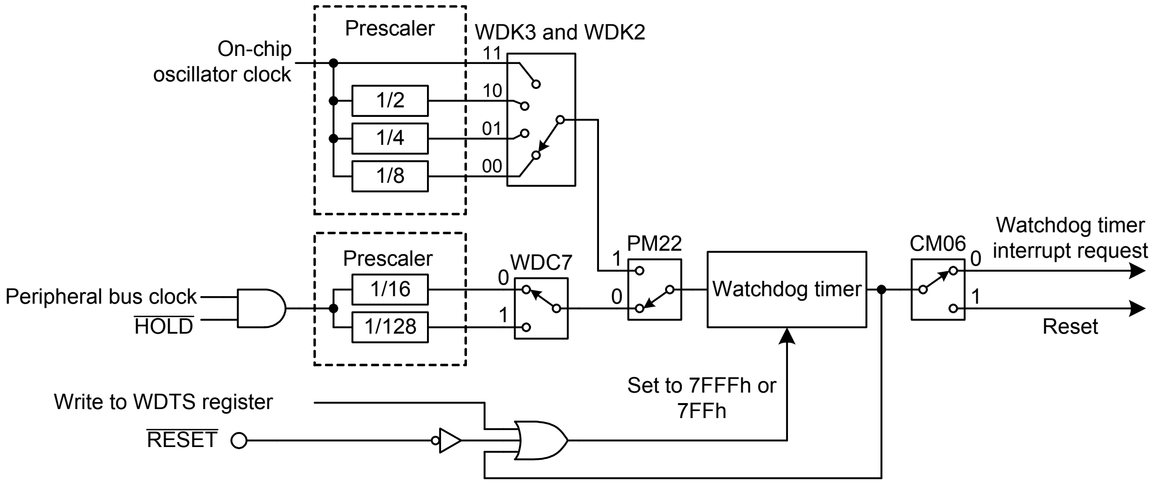
$$\text{Watchdog timer period} = \frac{\text{Prescaler divisor (1, 2, 4, or 8)} \times 2048}{\text{On-chip oscillator clock frequency}}$$

For example, when the peripheral bus clock is selected as the count source and it is 1/2 of 64 MHz CPU clock and the prescaler has a divide-by-16 operation, the watchdog timer period is approximately 16.4 ms. Note that marginal errors within one prescaler output cycle may occur in the watchdog timer period.

The watchdog timer is initialized when a write operation to the WDTS register is performed or when a watchdog timer interrupt request is generated. The prescaler is initialized only when the MCU is reset.

After a reset, the watchdog timer starts counting automatically if the OFS area of the flash memory are preset. When the WDTON bit in the OFS area is set to 1, both the watchdog timer and the prescaler are stopped. They start counting when a write operation to the WDTS register is performed. When the WDTON bit is set to 0, both the watchdog timer and the prescaler automatically start counting after a reset.

Figure 11.1 shows a block diagram of the watchdog timer. Figure 11.2 to Figure 11.5 show registers associated with the watchdog timer.

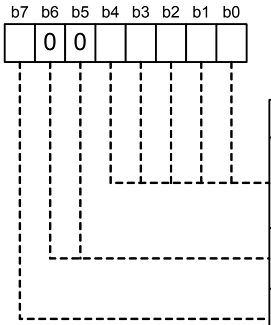


CM06: Bit in the CM0 register
WDC7: Bit in the WDC register

PM22: Bit in the PM2 register
WDK2 and WDK3: Bits in the WDK register

Figure 11.1 Watchdog Timer Block Diagram

Watchdog Timer Control Register



Symbol
WDC

Address
4404Fh

Reset Value
000X XXXXb

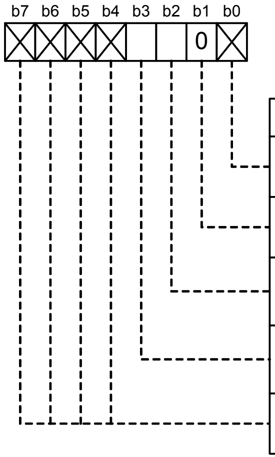
Bit Symbol	Bit Name	Function	RW
— (b4-b0)	Upper five bits of the watchdog timer, when the peripheral bus clock is used: b14 to b10 when the on-chip oscillator is used: b10 to b6 (1)		RO
— (b6-b5)	Reserved	Should be written with 0	RW
WDC7	Prescaler Select Bit (2)	0: Divide-by-16 1: Divide-by-128	RW

Notes:

1. When the on-chip oscillator clock is used as the count source, the read value may be undefined due to a change in the count value while being read.
2. Set this bit before activating the watchdog timer.

Figure 11.2 WDC Register

Watchdog Timer Clock Control Register (1)



Symbol	Address	Reset Value
WDK	4404Dh	0000 0000b

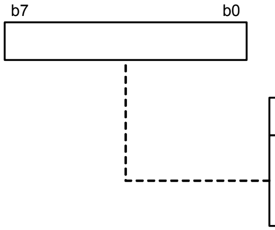
Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit; should be written with 0 and read as 0		—
— (b1)	Reserved	Should be written with 0	RW
WDK2	On-chip Oscillator Prescaler Select Bit (2)	b3 b2 0 0 : Divide-by-8 (approx. 15.6 kHz) 0 1 : Divide-by-4 (approx. 31.3 kHz) 1 0 : Divide-by-2 (approx. 62.5 kHz) 1 1 : No division (approx. 125 kHz)	RW
WDK3			RW
— (b7-b4)	No register bits; should be written with 0 and read as 0		—

Notes:

1. This register cannot be rewritten when the WDTON bit in the OFS area is set to 0 (the watchdog timer starts counting automatically after a reset). When the WDTON bit is set to 1 (the watchdog timer stops after a reset), this register can be rewritten only once after a reset. When the count source protect mode is enabled, this register should be set at the very beginning of the program.
2. Set these bits before activating the watchdog timer.

Figure 11.3 WDK Register

Watchdog Timer Start Register (1)



Symbol	Address	Reset Value
WDTS	4404Eh	Undefined

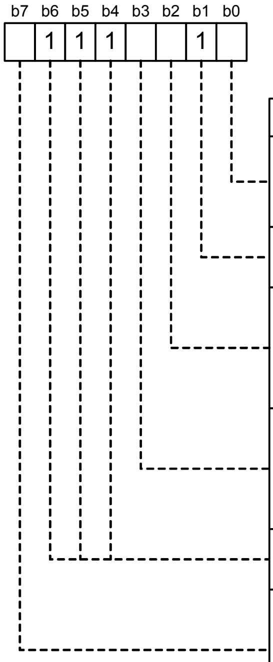
Function	RW
The watchdog timer is initialized by a write access. Then it starts counting downward. Either 7FFFh (when the peripheral bus clock is used) or 7FFh (when the on-chip oscillator clock is used) is set as the default value by writing a given value	WO

Note:

1. Set the PRC40 bit in the PRCR4 register to 1 (write enabled) before rewriting this register.

Figure 11.4 WDTS Register

Optional Function Select Area (1)



Symbol	Address	Factory Preset Value
OFS	FFFFFFEFh	FFh

Bit Symbol	Bit Name	Function	RW
WDTON	Watchdog Timer Start Timing Select Bit (2)	After a reset, the watchdog timer 0: Starts counting automatically 1: Stops	(1)
— (b1)	Reserved	Should be written with 1	(1)
WPSC0	Watchdog Timer On-chip Oscillator Prescaler Select Bit (3)	b3 b2 0 0 : Divide-by-8 (WDK3 and WDK2 = 00b) 0 1 : Divide-by-4 (WDK3 and WDK2 = 01b) 1 0 : Divide-by-2 (WDK3 and WDK2 = 10b) 1 1 : No division (WDK3 and WDK2 = 11b)	(1)
WPSC1			(1)
— (b6-b4)	Reserved	Should be written with 1	(1)
CSPM	Post-reset Count Source Protect Mode Select Bit (2, 3)	After a reset, count source protect mode is 0: Enabled (PM22 = 1) 1: Disabled (PM22 = 0)	(1)

Notes:

1. This area should be rewritten by the flash programmer. It cannot be rewritten by a program.
2. To enable the count source protect mode, set bits WDTON and CSPM to 0. Once the watchdog timer starts, the PM22 bit in the PM2 register cannot be rewritten.
3. These bit settings are disabled when the WDTON bit is 1. The values set to these bits are reflected to registers WDK and PM2 when the WDTON bit is 0.

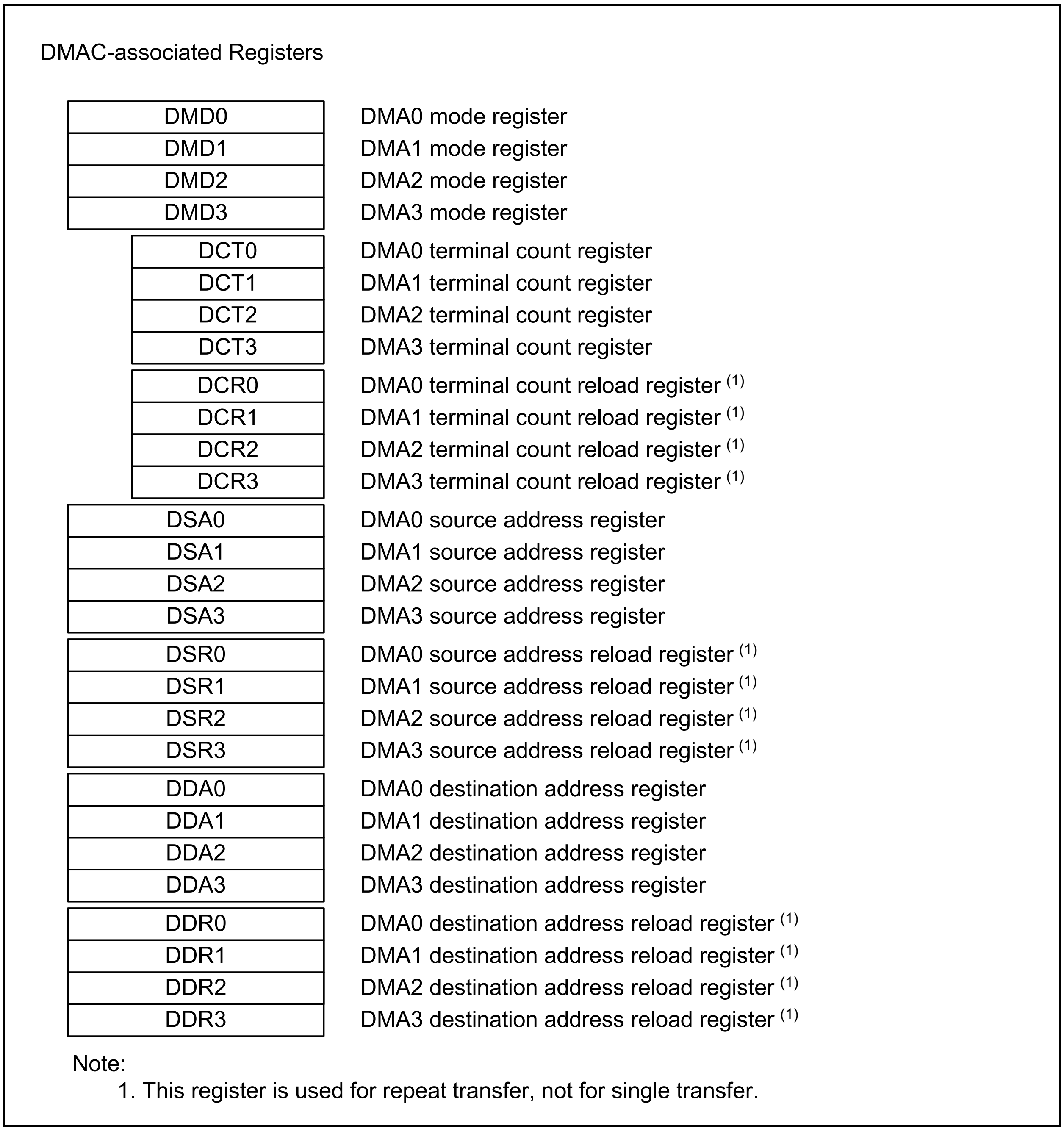
Figure 11.5 OFS Area

12. DMAC

Direct memory access (DMA) is a system that can control data transfer without using a CPU instruction. The R32C/100 Series’ four channel DMA controller (DMAC) transmits 8-bit (byte), 16-bit (word), or 32-bit (long word) data in cycle-steal mode from a source address to a destination address each time a transfer request is generated.

The DMAC, which shares a data bus with the CPU, has a higher bus access priority than the CPU. This allows the DMAC to perform fast data transfer when a transfer request is generated.

Figure 12.1 shows a map of the CPU-internal registers associated with DMAC. Table 12.1 lists DMAC specifications. Figure 12.2 to Figure 12.10 show registers associated with DMAC. Since the registers shown in Figure 12.1 are allocated in the CPU, the LDC or STC instruction should be used to write to the registers.



DMAC-associated Registers

DMD0	DMA0 mode register
DMD1	DMA1 mode register
DMD2	DMA2 mode register
DMD3	DMA3 mode register
DCT0	DMA0 terminal count register
DCT1	DMA1 terminal count register
DCT2	DMA2 terminal count register
DCT3	DMA3 terminal count register
DCR0	DMA0 terminal count reload register (1)
DCR1	DMA1 terminal count reload register (1)
DCR2	DMA2 terminal count reload register (1)
DCR3	DMA3 terminal count reload register (1)
DSA0	DMA0 source address register
DSA1	DMA1 source address register
DSA2	DMA2 source address register
DSA3	DMA3 source address register
DSR0	DMA0 source address reload register (1)
DSR1	DMA1 source address reload register (1)
DSR2	DMA2 source address reload register (1)
DSR3	DMA3 source address reload register (1)
DDA0	DMA0 destination address register
DDA1	DMA1 destination address register
DDA2	DMA2 destination address register
DDA3	DMA3 destination address register
DDR0	DMA0 destination address reload register (1)
DDR1	DMA1 destination address reload register (1)
DDR2	DMA2 destination address reload register (1)
DDR3	DMA3 destination address reload register (1)

Note:
1. This register is used for repeat transfer, not for single transfer.

Figure 12.1 CPU-internal Registers for DMAC

Table 12.1 DMAC Specifications

Item		Specification
Channels		4
Bus request mode		Cycle-steal mode
Transfer memory spaces		From a given address in a 64-Mbyte space (00000000h to 01FFFFFFh and FE000000h to FFFFFFFFh) to another given address in the same space
Maximum transfer bytes		64-Mbytes (when 32-bit data is transferred), 32-Mbytes (when 16-bit data is transferred), 16-Mbytes (when 8-bit data is transferred)
DMA request sources (1)		Falling edge or both edges of signals applied to pins INT0 to INT3 Interrupt requests from timers A0 to A4 Interrupt requests from timers B0 to B5 Transmit/receive interrupt requests from UART0 to UART4 A/D conversion interrupt requests Intelligent I/O interrupt requests Serial bus interface interrupt requests Software trigger
Channel priority		DMA0 > DMA1 > DMA2 > DMA3 (DMA0 has the highest priority)
Transfer sizes		8 bits, 16 bits, or 32 bits
Addressing modes		Incrementing addressing or non-incrementing addressing
Transfer modes	Single transfer	Transfer is completed when the DCTi register (i = 0 to 3) becomes 00000000h
	Repeat transfer	When the DCTi register becomes 00000000h, the value of the DCRi register is reloaded into the DCTi register to continue the DMA transfer
DMA transfer complete interrupt request generation timing		When the DCTi register changes from 00000001h to 00000000h
DMA transfer start-up	Single transfer	When a DMA transfer request is generated after the DCTi register is set to a value other than 00000000h and bits MDi1 and MDi0 in the DMDi register are set to 01b (single transfer)
	Repeat transfer	When a DMA transfer request is generated after the DCTi register is set to a value other than 00000000h and bits MDi1 and MDi0 are set to 11b (repeat transfer)
DMA transfer stop	Single transfer	When bits MDi1 and MDi0 are set to 00b (DMA transfer disabled)
	Repeat transfer	When bits MDi1 and MDi0 are set to 00b (DMA transfer disabled)
Reload timing to DCTi, DSAi, or DDAi register		When the DCTi register changes from 00000001h to 00000000h in repeat transfer mode
Minimum DMA transfer cycles		3

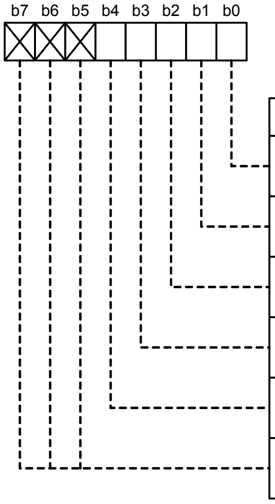
Note:

1. DMA transfer does not affect any interrupts.

The DMA transfer request is available by two different sources: software and hardware. More concretely, they are a write access to the DSR bit in the DMiSL2 register (i = 0 to 3) and an interrupt request output from a function specified in bits DSEL4 to DSEL0 in the DMiSL register, and in bits DSEL24 to DSEL20 in the DMiSL2 register. Unlike interrupt requests, the DMA transfer request is not affected by the I flag or the interrupt control register. Therefore this request can be accepted even when interrupts are disabled. Since the DMA transfer does not affect any interrupts, either, the IR bit in the interrupt control register is not changed by the DMA transfer.

DMAi Request Source Select Register (i = 0 to 3)

Symbol	Address	Reset Value
DM0SL to DM3SL	44078h, 44079h, 4407Ah, 4407Bh	XXX0 0000b



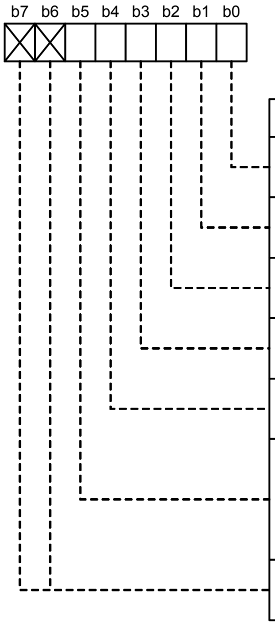
Bit Symbol	Bit Name	Function	RW
DSEL0	DMA Request Source Select Bit [1]	Refer to Table 12.2 “DMiSL Register (i = 0 to 3) Functions”	RW
DSEL1			RW
DSEL2			RW
DSEL3			RW
DSEL4			RW
— (b7-b5)	No register bits; should be written with 0 and read as undefined value		—

Note:

1. The bit settings of bits DSEL4 to DSEL0 should be changed while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are set to 00b (DMA transfer disabled).

Figure 12.2 Registers DM0SL to DM3SL

DMAi Request Source Select Register 2 (i = 0 to 3)



Symbol	Address	Reset Value
DM0SL2 to DM3SL2	44070h, 44071h, 44072h, 44073h	XX00 0000b

Bit Symbol	Bit Name	Function	RW
DSEL20	DMA Request Source Select Bit [1]	Refer to Table 12.3 “DMiSL2 Register (i = 0 to 3) Functions”	RW
DSEL21			RW
DSEL22			RW
DSEL23			RW
DSEL24			RW
DSR	Software DMA Transfer Request Bit	When a software trigger is selected, a DMA transfer request is generated by setting this bit to 1 (the bit is read as 0)	RW
— (b7-b6)	No register bits; should be written with 0 and read as undefined value		—

Note:

1. The bit settings of bits DSEL24 to DSEL20 should be changed while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are set to 00b (DMA transfer disabled).

Figure 12.3 Registers DM0SL2 to DM3SL2

Table 12.2 DMiSL Register (i = 0 to 3) Functions

Setting Value	DMA Request Source			
b4 b3 b2 b1 b0	DMA0	DMA1	DMA2	DMA3
0 0 0 0 0	Select from DMiSL2 register			
0 0 0 0 1	Falling edge of $\overline{INT0}$ (1)	Falling edge of $\overline{INT1}$ (1)	Falling edge of $\overline{INT2}$ (1)	Falling edge of $\overline{INT3}$ (1)
0 0 0 1 0	Both edges of $\overline{INT0}$ (1)	Both edges of $\overline{INT1}$ (1)	Both edges of $\overline{INT2}$ (1)	Both edges of $\overline{INT3}$ (1)
0 0 0 1 1	Timer A0 interrupt request			
0 0 1 0 0	Timer A1 interrupt request			
0 0 1 0 1	Timer A2 interrupt request			
0 0 1 1 0	Timer A3 interrupt request			
0 0 1 1 1	Timer A4 interrupt request			
0 1 0 0 0	Timer B0 interrupt request			
0 1 0 0 1	Timer B1 interrupt request			
0 1 0 1 0	Timer B2 interrupt request			
0 1 0 1 1	Timer B3 interrupt request			
0 1 1 0 0	Timer B4 interrupt request			
0 1 1 0 1	Timer B5 interrupt request			
0 1 1 1 0	UART0 transmit interrupt request			
0 1 1 1 1	UART0 receive interrupt request or ACK interrupt request (2)			
1 0 0 0 0	UART1 transmit interrupt request			
1 0 0 0 1	UART1 receive interrupt request or ACK interrupt request (2)			
1 0 0 1 0	UART2 transmit interrupt request			
1 0 0 1 1	UART2 receive interrupt request or ACK interrupt request (2)			
1 0 1 0 0	Reserved			
1 0 1 0 1	Reserved			
1 0 1 1 0	Reserved			
1 0 1 1 1	Reserved			
1 1 0 0 0	A/D0 interrupt request			
1 1 0 0 1	Intelligent I/O interrupt 0 request	Intelligent I/O interrupt 7 request	Intelligent I/O interrupt 2 request	Intelligent I/O interrupt 9 request
1 1 0 1 0	Intelligent I/O interrupt 1 request	Intelligent I/O interrupt 8 request	Intelligent I/O interrupt 3 request	Intelligent I/O interrupt 10 request
1 1 0 1 1	Intelligent I/O interrupt 2 request	Intelligent I/O interrupt 9 request	Intelligent I/O interrupt 4 request	Intelligent I/O interrupt 11 request
1 1 1 0 0	Intelligent I/O interrupt 3 request	Intelligent I/O interrupt 10 request	Intelligent I/O interrupt 5 request	Intelligent I/O interrupt 0 request
1 1 1 0 1	Intelligent I/O interrupt 4 request	Intelligent I/O interrupt 11 request	Intelligent I/O interrupt 6 request	Intelligent I/O interrupt 1 request
1 1 1 1 0	Intelligent I/O interrupt 5 request	Intelligent I/O interrupt 0 request	Intelligent I/O interrupt 7 request	Intelligent I/O interrupt 2 request
1 1 1 1 1	Intelligent I/O interrupt 6 request	Intelligent I/O interrupt 1 request	Intelligent I/O interrupt 8 request	Intelligent I/O interrupt 3 request

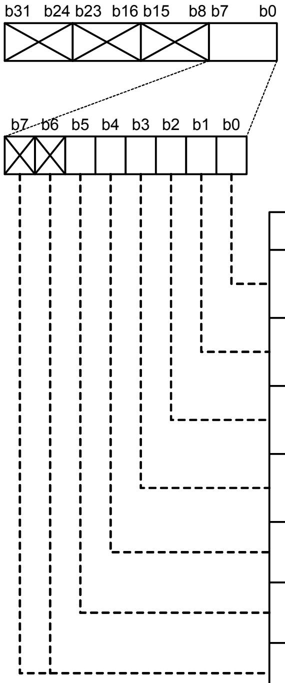
Notes:

1. The falling edge and both edges of signals applied to the $\overline{INTi}$ pin (i = 0 to 3) become the DMA request sources. These request sources are not affected by external interrupts (the IFSR0 register and bits POL and LVS in the INTiIC register), and vice versa.
2. Registers UiSMR and UiSMR2 (i = 0 to 2) are used to switch between the UARTi receive interrupt and ACK interrupt.

Table 12.3 DMiSL2 Register (i = 0 to 3) Functions

Setting Value	DMA Request Source			
b4 b3 b2 b1 b0	DMA0	DMA1	DMA2	DMA3
0 0 0 0 0	Software trigger			
0 0 0 0 1	Reserved			
0 0 0 1 0	Reserved			
0 0 0 1 1	Reserved			
0 0 1 0 0	Serial bus interface 0 interrupt request			
0 0 1 0 1	Serial bus interface 1 interrupt request			
0 0 1 1 0	Reserved			
0 0 1 1 1	Reserved			
0 1 0 0 0	Reserved			
0 1 0 0 1	Reserved			
0 1 0 1 0	Reserved			
0 1 0 1 1	Reserved			
0 1 1 0 0	Reserved			
0 1 1 0 1	Reserved			
0 1 1 1 0	Reserved			
0 1 1 1 1	Reserved			
1 0 0 0 0	Reserved			
1 0 0 0 1	Reserved			
1 0 0 1 0	Reserved			
1 0 0 1 1	Reserved			
1 0 1 0 0	Reserved			
1 0 1 0 1	Reserved			
1 0 1 1 0	Reserved			
1 0 1 1 1	Reserved			
1 1 0 0 0	UART3 transmit interrupt request			
1 1 0 0 1	UART3 receive interrupt request			
1 1 0 1 0	UART4 transmit interrupt request			
1 1 0 1 1	UART4 receive interrupt request			
1 1 1 0 0	Reserved			
1 1 1 0 1	Reserved			
1 1 1 1 0	Reserved			
1 1 1 1 1	Reserved			

DMAi Mode Register (i = 0 to 3) (1)



Symbol
DMD0 to DMD3

Address
(CPU internal register)

Reset Value
XXXX XXXX XXXX XXXX XXXX XXXX XX00 0000b

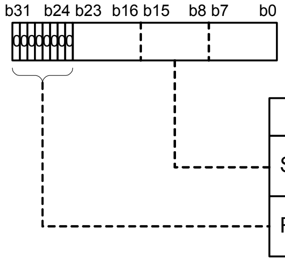
Bit Symbol	Bit Name	Function	RW
MDi0	Transfer Mode Select Bit (2)	b1 b0 0 0 : DMA transfer disabled 0 1 : Single transfer 1 0 : Do not use this combination 1 1 : Repeat transfer	RW
MDi1			RW
BWi0	Transfer Size Select Bit (3)	b3 b2 0 0 : 8 bits 0 1 : 16 bits 1 0 : 32 bits 1 1 : Do not use this combination	RW
BWi1			RW
USAi	Source Addressing Mode Select Bit (3)	0: Non-incrementing addressing 1: Incrementing addressing	RW
UDAi	Destination Addressing Mode Select Bit (3)	0: Non-incrementing addressing 1: Incrementing addressing	RW
— (b7-b6)	No register bits; should be written with 0 and read as undefined value		—
— (b31-b8)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. The LDC instruction should be used to write to this register.
2. This bit should be set after all other DMAC-associated registers are set.
3. Set bits MDi1 and MDi0 to 00b before rewriting these bits.

Figure 12.4 Registers DMD0 to DMD3

DMAi Terminal Count Register (i = 0 to 3) (1)



Symbol
DCT0 to DCT3

Address
(CPU internal register)

Reset Value
XXXX XXXXh

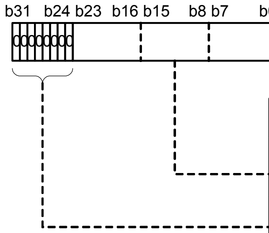
Function	Setting Range	RW
Set the number of transfers to be performed	000000h to FFFFFFh (2)	RW
Reserved	Should be set to 00h	RW

Notes:

1. The LDC instruction should be used to write to this register. The register should be set while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are 00b (DMA transfer disabled).
2. When these bits are set to 000000h, new DMA transfer requests cannot be accepted.

Figure 12.5 Registers DCT0 to DCT3

DMAi Terminal Count Reload Register (i = 0 to 3) (1)



Symbol	Address	Reset Value
DCR0 to DCR3	(CPU internal register)	XXXX XXXXh

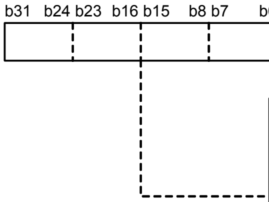
Function	Setting Range	RW
Set the number of transfers to be performed	000000h to FFFFFFh	RW
Reserved	Should be set to 00h	RW

Note:

1. The LDC instruction should be used to write to this register. The register should be set while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are 00b (DMA transfer disabled).

Figure 12.6 Registers DCR0 to DCR3

DMAi Source Address Register (i = 0 to 3) (1)



Symbol	Address	Reset Value
DSA0 to DSA3	(CPU internal register)	XXXX XXXXh

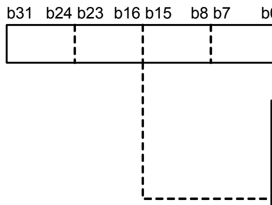
Function	Setting Range	RW
Set a source address	00000000h to 01FFFFFFh and FE000000h to FFFFFFFFh (64-Mbyte space)	RW

Note:

1. The LDC instruction should be used to write to this register. The register should be set while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are 00b (DMA transfer disabled).

Figure 12.7 Registers DSA0 to DSA3

DMAi Source Address Reload Register (i = 0 to 3) (1)



Symbol
DSR0 to DSR3

Address
(CPU internal register)

Reset Value
XXXX XXXXh

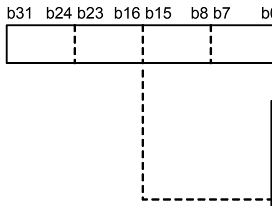
Function	Setting Range	RW
Set a source address	00000000h to 01FFFFFFh and FE000000h to FFFFFFFFh (64-Mbyte space)	RW

Note:

1. The LDC instruction should be used to write to this register. The register should be set while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are set to 00b (DMA transfer disabled).

Figure 12.8 Registers DSR0 to DSR3

DMAi Destination Address Register (i = 0 to 3) (1)



Symbol
DDA0 to DDA3

Address
(CPU internal register)

Reset Value
XXXX XXXXh

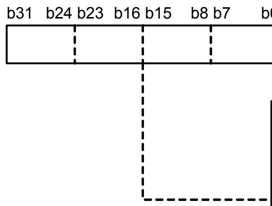
Function	Setting Range	RW
Set a destination address	00000000h to 01FFFFFFh and FE000000h to FFFFFFFFh (64-Mbyte space)	RW

Note:

1. The LDC instruction should be used to write to this register. The register should be set while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are set to 00b (DMA transfer disabled).

Figure 12.9 Registers DDA0 to DDA3

DMAi Destination Address Reload Register (i = 0 to 3) (1)



Symbol
DDR0 to DDR3

Address
(CPU internal register)

Reset Value
XXXX XXXXh

Function	Setting Range	RW
Set a destination address	00000000h to 01FFFFFFh and FE000000h to FFFFFFFFh (64-Mbyte space)	RW

Note:

1. The LDC instruction should be used to write to this register. The register should be set while bits MDi1 and MDi0 in the DMDi register of the corresponding channel are set to 00b (DMA transfer disabled).

Figure 12.10 Registers DDR0 to DDR3

12.1 Transfer Cycle

The transfer cycle is composed of bus cycles to read data from (source read) or to write data to (destination write) memory or an SFR.

The read and write bus cycles vary with the setting of registers DSAi (i = 0 to 3) and DDAi, the width and timing of the data bus connected to the relevant device.

12.1.1 Effect of Transfer Address and Data Bus Width

Table 12.4 lists the incremental bus cycles caused by transfer address alignment or data bus width.

Table 12.4 Incremental Bus Cycles Caused by Transfer Address and Data Bus Width

Transfer Data Unit	Data Bus Width	Transfer Address	Bus Cycles to be Incremented	Bus Cycles Generated
8-bit transfer	8 to 64 bits	n	0	[n]
16-bit transfer	8 bits	n	+1	[n] - [n + 1]
	16 bits	2n	0	[2n]
		2n + 1	+1	[2n + 1] - [2n + 2]
	32 bits	4n	0	[4n]
		4n + 1	0	[4n + 1]
		4n + 2	0	[4n + 2]
		4n + 3	+1	[4n + 3] - [4n + 4]
	64 bits	8n	0	[8n]
		8n + 1	0	[8n + 1]
		8n + 2	0	[8n + 2]
		8n + 3	0	[8n + 3]
		8n + 4	0	[8n + 4]
		8n + 5	0	[8n + 5]
		8n + 6	0	[8n + 6]
		8n + 7	+1	[8n + 7] - [8n + 8]
32-bit transfer	8 bits	n	+3	[n] - [n + 1] - [n + 2] - [n + 3]
	16 bits	4n	+1	[4n] - [4n + 2]
		4n + 1	+2	[4n + 1] - [4n + 2] - [4n + 4]
		4n + 2	+1	[4n + 2] - [4n + 4]
		4n + 3	+2	[4n + 3] - [4n + 4] - [4n + 6]
	32 bits	4n	0	[4n]
		4n + 1	+1	[4n + 1] - [4n + 4]
		4n + 2	+1	[4n + 2] - [4n + 4]
		4n + 3	+1	[4n + 3] - [4n + 4]
	64 bits	8n	0	[8n]
		8n + 1	0	[8n + 1]
		8n + 2	0	[8n + 2]
		8n + 3	0	[8n + 3]
		8n + 4	0	[8n + 4]
		8n + 5	+1	[8n + 5] - [8n + 8]
		8n + 6	+1	[8n + 6] - [8n + 8]
		8n + 7	+1	[8n + 7] - [8n + 8]

12.1.2 Effect of Bus Timing

In the R32C/100 Series, a separate bus is connected to each device. The bus width and bus timing vary with each device. Table 12.5 lists the bus width and access cycles for each device.

Table 12.5 Bus Width and Bus Cycles

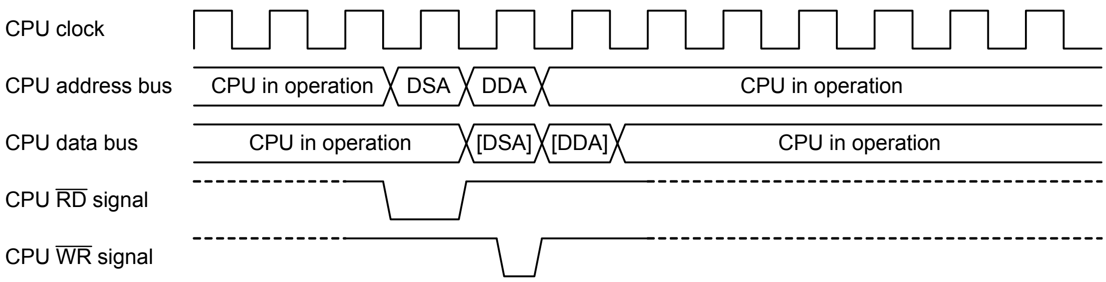
Device	Addresses (1)	Bus Width	Access Cycles (2)	Reference Clock
Flash memory	FFE00000h to FFFFFFFFh	64-bit	2 or 3 (3)	CPU clock
Data flash	00060000h to 00061FFFh	64-bit	5	CPU clock
RAM	00000400h to 0003FFFFh	64-bit	1 or 2 (4)	CPU clock
SFR space	00000000h to 0000001Fh	16-bit	3 (5)	Peripheral bus clock
	00000020h to 000003FFh	16-bit	2 (5)	Peripheral bus clock
SFR2 space	00040000h to 00041FFFh	16-bit	2 (5)	Peripheral bus clock
	00042000h to 00043FFFh	32-bit	2 (5)	Peripheral bus clock
	00044000h to 000440DFh	16-bit	2 (5, 6)	Peripheral bus clock
	000440E0h to 000443FFh	16-bit	3 (5, 6)	Peripheral bus clock
	00044400h to 00045FFFh	16-bit	2 (5, 6)	Peripheral bus clock
	00046000h to 000467FFh	32-bit	3 (5, 6)	Peripheral bus clock
	00046800h to 00047FFFh	32-bit	2 (5, 6)	Peripheral bus clock
	00048000h to 0004FFFFh	64-bit	2	CPU clock

Notes:

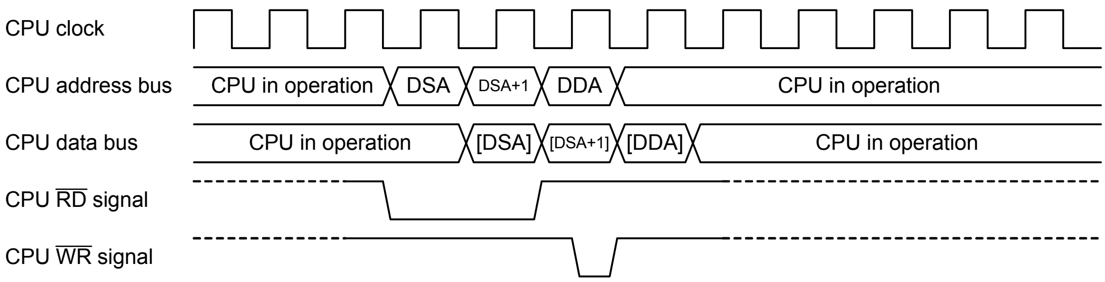
1. Reserved spaces are included.
2. Access cycles are based on each bus clock.
3. An access to the same page as the previous time requires two cycles. Otherwise, three cycles are required.
4. If write cycles are generated sequentially, each write cycle except the initial one has two access cycles. A read cycle just after a write cycle has also two access cycles.
5. If SFRs are sequentially accessed, each access except the initial one has one additional base clock cycle.
6. Up to one access cycle may be added depending on the phase of peripheral bus clock.

Figure 12.11 shows an example of source-read bus cycles in a transfer cycle. In this figure, the number of source-read bus cycles is shown under different conditions, provided that the destination address is in an internal RAM with one bus cycle of destination-write. In a real operation, the transfer cycles change according to conditions for destination-write bus cycles as well as for source-read bus cycles. To calculate a transfer cycle, respective conditions should be applied to both destination-write bus cycle and source-read bus cycle. In (2) of Figure 12.11, for example, if two bus cycles are generated, bus cycles required for the destination-write is two as well as for the source-read bus cycle.

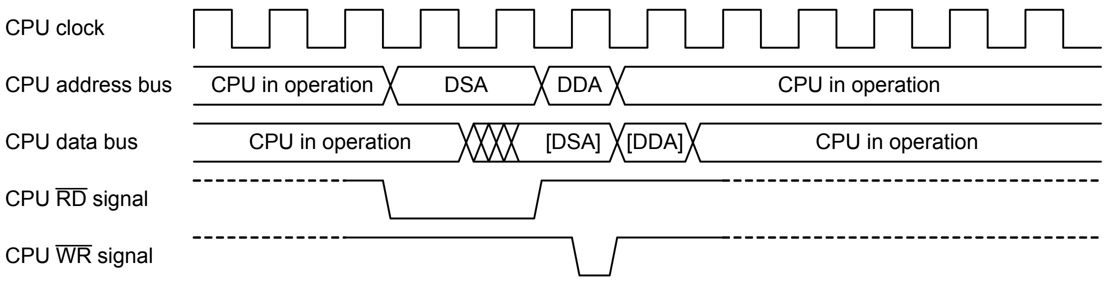
(1) One bus cycle of source-read is generated
Example: 16-bit data transfer from address 8n in the RAM



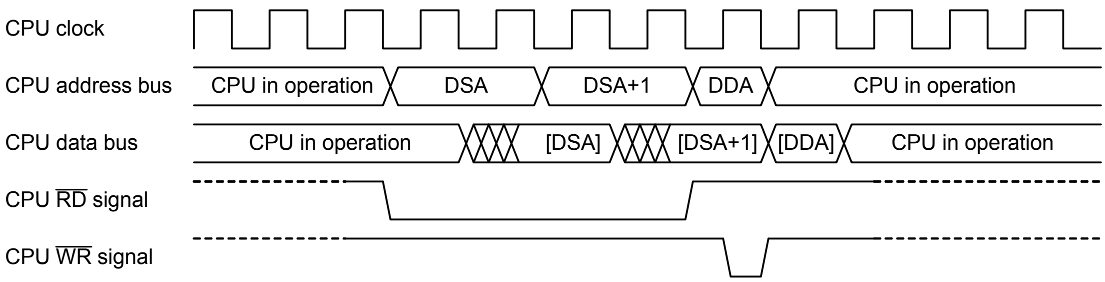
(2) Two bus cycles of source-read are generated
Example: 16-bit data transfer from address 8n+7 in the RAM



(3) One bus cycle of source-read is generated with one wait cycle
Example: 16-bit data transfer from address 16n in the ROM



(4) Two bus cycles of source-read is generated with one wait cycle
Example: 16-bit data transfer from address 16n+7 in the ROM



Note:
1. The above assumes that destination-write completes in one bus cycle. In actual use, the number of destination-write bus cycles should be considered according to the conditions such as above.

Figure 12.11 Source-read Bus Cycles in a Transfer Cycle

12.2 DMA Transfer Cycle

The DMA transfer cycles are calculated as follows:

$$\text{Number of transfer cycles} = \text{Source-read bus cycles} \times j + \text{Destination-write bus cycles} \times k + 1$$

where:

j = access cycles for read

k = access cycles for write (refer to Table 12.5)

Each bus cycle, source-read, and destination-write requires one or more cycles. In addition, more cycles may be required depending on the transfer address. Refer to Table 12.4 for details on the required bus cycles.

"+1" in the formula above means a cycle required to decrement the value of DCTi register (i = 0 to 3).

The following are calculation examples:

To transfer 32-bit data from address 400h in the RAM to address 800h in the RAM,

$$\begin{aligned}\text{Number of the transfer cycles} &= 1 \times 1 + 1 \times 1 + 1 \\ &= 3\end{aligned}$$

Thus, there are three cycles.

To transfer 16-bit data from the AD00 register at address 380h to registers P1 and P0 at addresses 3C1h and 3C0h, respectively, when the peripheral bus clock frequency is half the CPU clock,

$$\begin{aligned}\text{Number of the transfer cycles} &= 1 \times 2 \times 2 + 1 \times 2 \times 2 + 1 \\ &= 9\end{aligned}$$

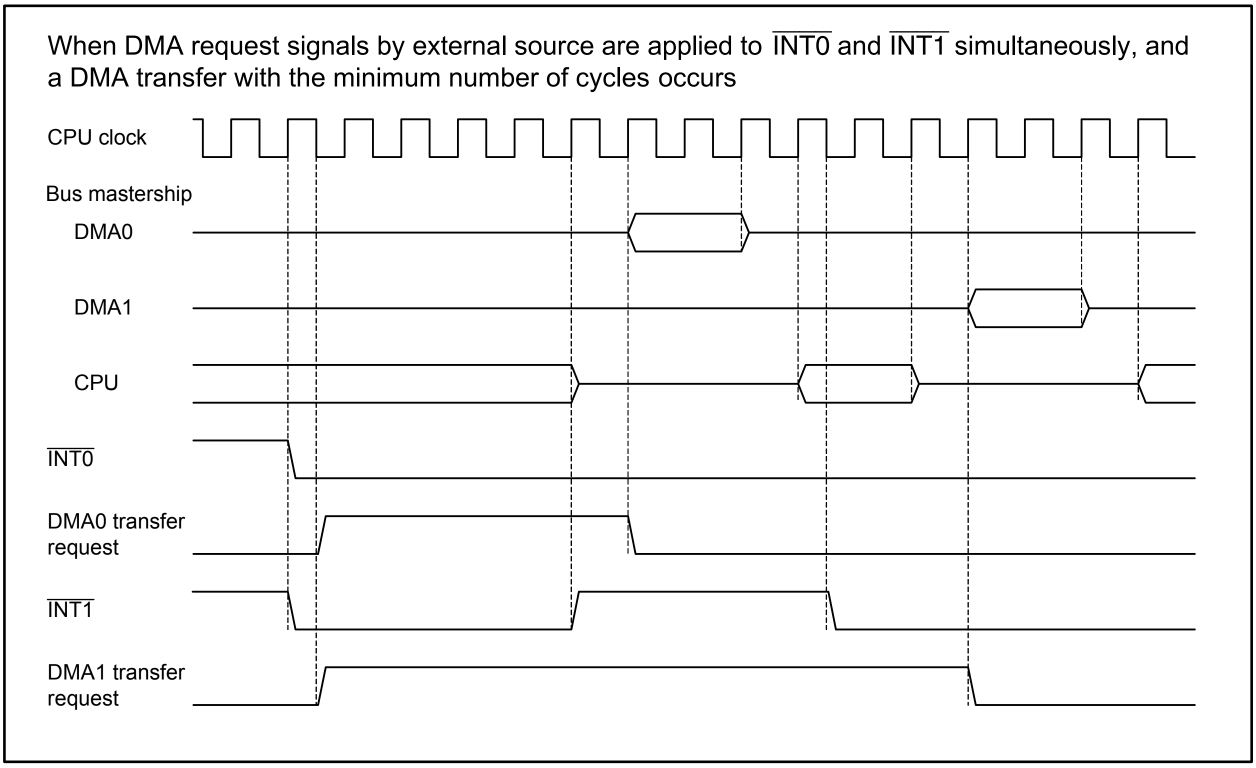
Thus, there are nine cycles.

12.3 Channel Priority and DMA Transfer Timing

When multiple DMA transfer requests are generated in the same sampling period, between the falling edge of the CPU clock and the next falling edge, these requests are simultaneously input into the DMAC. Channel priority in this case is: DMA0 > DMA1 > DMA2 > DMA3.

Figure 12.12 shows an example of the DMA transfer by external source, specifically when DMA0 and DMA1 requests are simultaneously generated. The DMA0, whose request priority is higher than that of DMA1, is received first to start the transfer and then hands over the bus to the CPU after completing one DMA0 transfer. Once the CPU completes one bus access, the DMA1 transfer starts. The CPU takes the bus back from the DMA1 after one DMA1 transfer is completed.

DMA transfer requests cannot be counted. The transfer occurs only once even when an $\overline{\text{INTi}}$ interrupt is generated more than once before the bus is granted, as shown by DMA1 in Figure 12.12.



When DMA request signals by external source are applied to $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$ simultaneously, and a DMA transfer with the minimum number of cycles occurs

Figure 12.12 DMA Transfer by External Source

12.4 Notes on DMAC

12.4.1 DMAC-associated Register Settings

- Set DMAC-associated registers while bits MDi1 and MDi0 (i = 0 to 3) in the DMDi register are 00b (DMA transfer disabled). Then, set bits MDi1 and MDi0 to 01b (single transfer) or 11b (repeat transfer) at the end of the setup procedure. This procedure also applies when rewriting bits UDAi, USAi, and BWi1 and BWi0 in the DMDi register.

- When rewriting the DMAC-associated registers while DMA transfer is enabled, disable the peripheral function as DMA request source so that no DMA transfer request is generated, then set bits MDi1 and MDi0 in the DMDi register of the corresponding channel to 00b (DMA transfer disabled).

- Once a DMA transfer request is accepted, DMA transfer cannot be disabled even if setting bits MDi1 and MDi0 in the DMDi register to 00b (DMA transfer disabled). Do not change the settings of any DMAC-associated registers other than bits MDi1 and MDi0 until the DMA transfer is completed.

- After setting registers DMiSL and DMiSL2, wait six or more peripheral bus clocks to set bits MDi1 and MDi0 in the DMDi register to 01b (single transfer) or 11b (repeat transfer).

12.4.2 Reading DMAC-associated Registers

- Use the following read order to sequentially read registers DMiSL and DMiSL2:
  DM0SL, DM1SL, DM2SL, and DM3SL
  DM0SL2, DM1SL2, DM2SL2, and DM3SL2

13. DMAC II

DMAC II is activated by an interrupt request from any peripheral function, and performs data transfer without a CPU instruction. Transfer sources are selectable from memory, immediate data, memory + memory, and immediate data + memory.

Table 13.1 lists specifications of DMAC II.

Table 13.1 DMAC II Specifications

Item	Specification
DMAC II triggers	Interrupt requests generated by any peripheral function when bits ILVL2 to ILVL0 in the corresponding interrupt control register are set to 111b (level 7)
Transfer types	• Data in memory is transferred to memory (memory-to-memory transfer) • Immediate data is transferred to memory (immediate data transfer) • Data in memory + data in memory are transferred to memory (calculation result transfer) • Immediate data + data in memory are transferred to memory (calculation result transfer)
Transfer sizes	8 bits or 16 bits
Transfer memory spaces	From a given address in a 64-Mbyte space (00000000h to 01FFFFFFh and FE000000h to FFFFFFFFh) to another given address in the same space (1)
Addressing modes	Individually selectable for each source address and destination address from the following two modes: • Non-incrementing addressing: Address is held constant throughout a data transfer/DMA II transaction • Incrementing addressing: Address increments by 1 (when 8-bit data is transferred) or 2 (when 16-bit data is transferred) after each data transfer
Transfer modes	• Single transfer: Only one data transfer is performed by one transfer request • Burst transfer: Data transfers are continuously performed for the number of times set in the transfer counter by one transfer request • Multiple transfer: Multiple memory-to-memory transfers are performed from different source addresses to different destination addresses by one transfer request
Chained transfer	Data transfer is sequentially performed according to a DMAC II index (transfer information) linked with the previous transfer
DMA II transfer complete interrupt request	An interrupt request is generated when the transfer counter reaches 0000h

Note:

1. When the transfer size is 16 bits and the destination address is FFFFFFFFh, data is transferred to FFFFFFFFh and 00000000h. This also applies when the source address is FFFFFFFFh.

13.1 DMAC II Settings

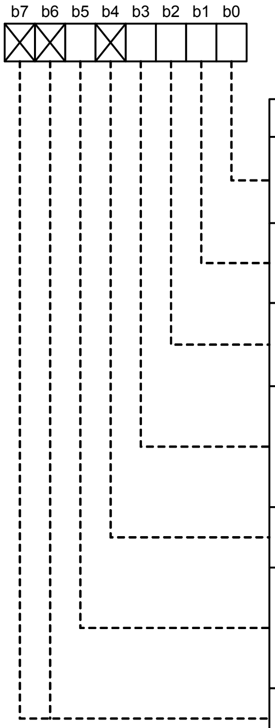
To activate DMAC II, set the following:

- Registers RIPL1 and RIPL2
- DMAC II index
- The interrupt control register of the peripheral function triggering DMAC II
- The relocatable vector of the peripheral function triggering DMAC II
- The IRLT bit in the IIOiIE register (i = 0 to 11) if the intelligent I/O interrupt is used. Refer to 10. “Interrupts” for details on the IIOiIE register.

13.1.1 Registers RIPL1 and RIPL2

When the DMAII bit in registers RIPL1 and RIPL2 is set to 1 (DMA II transfer selected) and the FSIT bit is set to 0 (normal interrupt selected), DMAC II is activated by an interrupt request from any peripheral function with bits ILVL2 to ILVL0 in the corresponding interrupt control register set to 111b (level 7). Figure 13.1 shows registers RIPL1 and RIPL2.

Wake-up IPL Setting Register i (i = 1, 2) (1)



Symbol	Address	Reset Value
RIPL1, RIPL2	4407Fh, 4407Dh	XX0X 0000b

Bit Symbol	Bit Name	Function	RW
RLVL0	Interrupt Priority Level for Wake-up Select Bit (2)	b2 b1 b0 0 0 0 : Level 0 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
RLVL1			RW
RLVL2			RW
FSIT	Fast Interrupt Select Bit (3)	0: Use interrupt request level 7 for normal interrupt 1: Use interrupt request level 7 for fast interrupt (4)	RW
— (b4)	No register bit; should be written with 0 and read as undefined value		—
DMAII	DMA II Select Bit (5)	0: Use interrupt request level 7 for interrupt 1: Use interrupt request level 7 for DMA II transfer (4)	RW
— (b7-b6)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. Registers RIPL1 and RIPL2 should be set with the same values.
2. The MCU exits wait mode or stop mode if the request level of the requested interrupt is higher than the level selected using bits RLVL2 to RLVL0. These bits should be set to the same value as the IPL in the flag register (FLG).
3. When the FSIT bit is set to 1, an interrupt with interrupt request level 7 becomes the fast interrupt. In this case, set the interrupt request level to level 7 with only one interrupt.
4. Either the FSIT or DMAII bit should be set to 1. The fast interrupt and DMAC II cannot be used simultaneously.
5. Bits ILVL2 to ILVL0 in the interrupt control register should be set after the DMAII bit is set. DMA II transfer is not affected by the I flag or IPL.

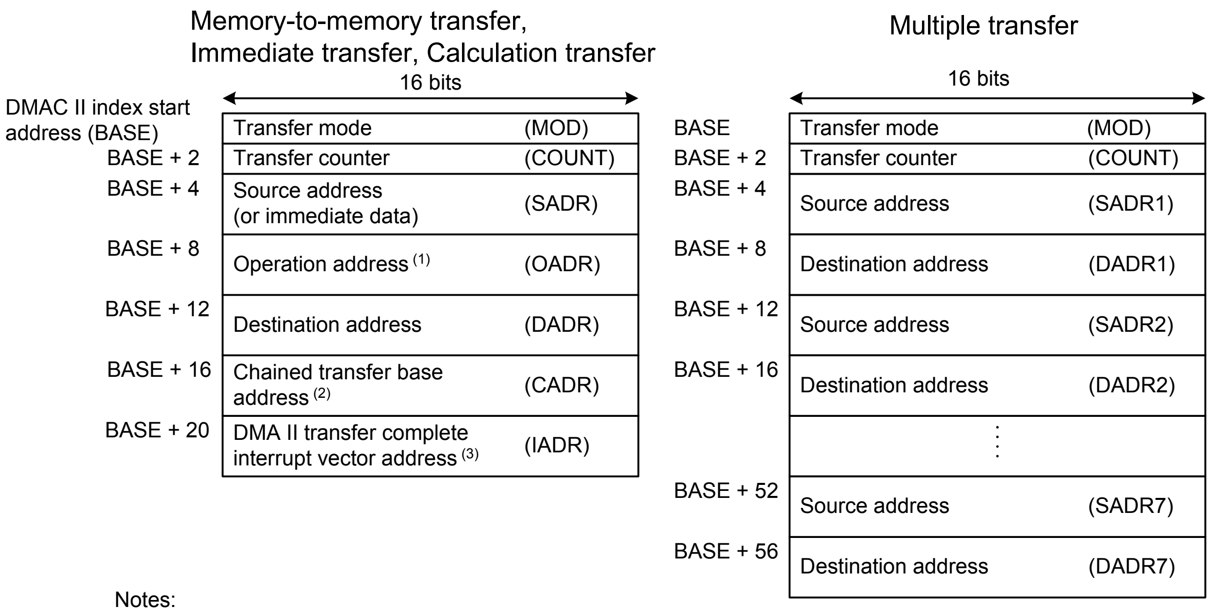
Figure 13.1 Registers RIPL1 and RIPL2

13.1.2 DMAC II Index

The DMAC II index is a data table of 12 to 60 bytes. It stores parameters for transfer mode, transfer counter, source address (or immediate data), operation address as an address to be calculated, destination address, chained transfer base address, and DMA II transfer complete interrupt vector address.

This DMAC II index should be allocated on the RAM.

Figure 13.2 shows a configuration of the DMAC II index and Table 13.2 lists a configuration example of the DMAC II index.



Memory-to-memory transfer, Immediate transfer, Calculation transfer

16 bits

DMAC II index start address (BASE)	Transfer mode	(MOD)
BASE + 2	Transfer counter	(COUNT)
BASE + 4	Source address (or immediate data)	(SADR)
BASE + 8	Operation address [1]	(OADR)
BASE + 12	Destination address	(DADR)
BASE + 16	Chained transfer base address [2]	(CADR)
BASE + 20	DMA II transfer complete interrupt vector address [3]	(IADR)

Multiple transfer

16 bits

BASE	Transfer mode	(MOD)
BASE + 2	Transfer counter	(COUNT)
BASE + 4	Source address	(SADR1)
BASE + 8	Destination address	(DADR1)
BASE + 12	Source address	(SADR2)
BASE + 16	Destination address	(DADR2)
	⋮	
BASE + 52	Source address	(SADR7)
BASE + 56	Destination address	(DADR7)

Notes:
1. This data is required only for the calculation transfer.
2. This data is required only for the chained transfer.
3. This data is required only for the DMA II transfer complete interrupt.

The DMAC II index should be allocated on the RAM. Required data should be set front-aligned. For example, when the calculation transfer is not used, the destination address should be set to BASE + 8 (refer to the “DMAC II Index Configuration” on the next page).
Start address of the DMAC II index should be set in the interrupt vector for the peripheral interrupt triggering DMAC II.

Figure 13.2 DMAC II Index

The following are the details on the DMAC II index. These parameters should be aligned in the order listed in Table 13.2 according to the transfer mode to be used.

- Transfer mode (MOD)
  Set a transfer mode in 2 bytes. Refer to Figure 13.3 for details on the setting of MOD.
- Transfer counter (COUNT)
  Set a number of transfers in 2 bytes.
- Source address (SADR)
  Set a source address or immediate data in 4 bytes. Note that the two upper bytes of immediate data are ignored.
- Operation address (OADR)
  Set an address in a to-be calculated memory in 4 bytes. This data setting is required only for the calculation transfer.
- Destination address (DADR)
  Set a destination address in 4 bytes.
- Chained transfer base address (CADR)
  Set the start address of the DMAC II index for the next transfer (BASE) in 4 bytes. This data setting is required only for the chained transfer.
- DMA II transfer complete interrupt vector address (IADR)
  Set a jump address for the DMA II transfer complete interrupt handler in 4 bytes. This data setting is required only for the DMA II transfer complete interrupt.

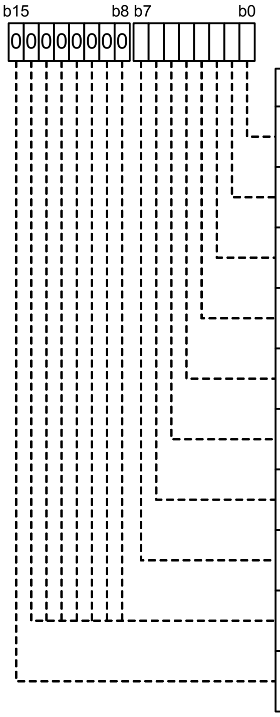
The symbols above are hereinafter used in place of their respective parameters.

Table 13.2 DMAC II Index Configuration

Transfer Data	Memory-to-memory Transfer/ Immediate Data Transfer				Calculation Transfer				Multiple Transfer
Chained transfer	Not used	Used	Not used	Used	Not used	Used	Not used	Used	Not available
DMA II transfer complete interrupt	Not used	Not used	Used	Used	Not used	Not used	Used	Used	Not available
DMAC II index	MOD COUNT SADR DADR 12 bytes	MOD COUNT SADR DADR CADR 16 bytes	MOD COUNT SADR DADR IADR 16 bytes	MOD COUNT SADR DADR CADR IADR 20 bytes	MOD COUNT SADR OADR DADR 16 bytes	MOD COUNT SADR OADR DADR CADR 20 bytes	MOD COUNT SADR OADR DADR IADR 20 bytes	MOD COUNT SADR OADR DADR CADR IADR 24 bytes	MOD COUNT SADR1 DADR1 SADRi DADRi i = 1 to 7 max. 60 bytes (when i = 7)

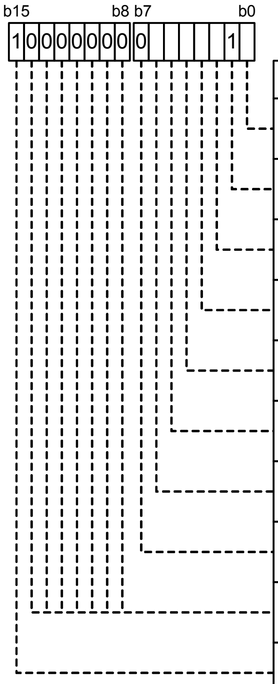
Transfer Mode (MOD) (1)

When multiple transfer is not selected (MULT = 0)



Bit Symbol	Bit Name	Function	RW
SIZE	Transfer Size Select Bit	0: 8 bits 1: 16 bits	RW
IMM	Transfer Source Select Bit	0: Immediate data 1: Memory	RW
UPDS	Source Addressing Select Bit	0: Non-incrementing addressing 1: Incrementing addressing	RW
UPDD	Destination Addressing Select Bit	0: Non-incrementing addressing 1: Incrementing addressing	RW
OPER	Calculation Result Transfer Select Bit	0: Not used 1: Used	RW
BRST	Burst Transfer Select Bit	0: Single transfer 1: Burst transfer	RW
INTE	DMA II Transfer Complete Interrupt Select Bit	0: Not used 1: Used	RW
CHAIN	Chained Transfer Select Bit	0: Not used 1: Used	RW
— (b14-b8)	Reserved	Should be written with 0	RW
MULT	Multiple Transfer Select Bit	0: Not used	RW

When multiple transfer is selected (MULT = 1)



Bit Symbol	Bit Name	Function	RW
SIZE	Transfer Size Select Bit	0: 8 bits 1: 16 bits	RW
IMM	Reserved	Should be written with 1	RW
UPDS	Source Addressing Select Bit	0: Non-incrementing addressing 1: Incrementing addressing	RW
UPDD	Destination Addressing Select Bit	0: Non-incrementing addressing 1: Incrementing addressing	RW
CNT0	Number of Transfers Setting Bit	b6 b5 b4 0 0 0 : Do not use this combination 0 0 1 : Once 0 1 0 : Twice : : 1 1 1 : Seven times	RW
CNT1			RW
CNT2			RW
CHAIN	Reserved	Should be written with 0	RW
— (b14-b8)	Reserved	Should be written with 0	RW
MULT	Multiple Transfer Select Bit	1: Used	RW

Note:

1. The MOD should be allocated on the RAM.

Figure 13.3 MOD

13.1.3 Interrupt Control Register of the Peripheral Function

Set bits ILVL2 to ILVL0 in the interrupt control register for the peripheral interrupt triggering DMAC II to 111b (level 7).

13.1.4 Relocatable Vector Table of the Peripheral Function

Set the start address of the DMAC II index to the interrupt vector for the peripheral interrupt triggering DMAC II.
To use the chained transfer, allocate the relocatable vector table on the RAM.

13.1.5 IRLT Bit in the IIOiIE Register (i = 0 to 11)

To use the intelligent I/O interrupt as a trigger for DMAC II, set the IRLT bit in the corresponding IIOiIE register to 0 (interrupt request for DMA or DMA II used).

13.2 DMAC II Operation

Set the DMAII bit in registers RIPL1 and RIPL2 to 1 (interrupt request level 7 used for DMA II transfer) to perform a DMA II transfer. DMAC II is activated by an interrupt request from any peripheral function with bits ILVL2 to ILVL0 in the corresponding interrupt control register set to 111b (level 7). These peripheral interrupt requests are available only for DMA II transfer and cannot be used for the CPU.
When an interrupt request is generated with interrupt request level 7, DMAC II is activated irrespective of the state of the I flag or IPL.
When a peripheral interrupt request triggering DMAC II and a higher-priority request such as the watchdog timer interrupt, oscillator stop detection interrupt, or NMI are simultaneously generated, the higher-priority interrupt is accepted prior to the DMA II transfer, and the DMA II transfer starts after the higher-priority interrupt sequence.

13.3 Transfer Types

DMAC II transfers three types of 8-bit or 16-bit data as follows:

• Memory-to-memory transfer:	Data is transferred from a given memory location in a 64-Mbyte space (addresses 00000000h to 01FFFFFFh and FE000000h to FFFFFFFFh) to another given memory location in the same space.
• Immediate data transfer:	Immediate data is transferred to a given memory location in a 64-Mbyte space.
• Calculation transfer:	Two data are added together and the result is transferred to a given memory location in a 64-Kbyte space.

When 16-bit data is transferred to DADR at FFFFFFFFh, it is transferred to 00000000h as well as FFFFFFFFh. The same transfer is performed when SADR is FFFFFFFFh.

13.3.1 Memory-to-memory Transfer

Data transfer between any two memory locations can be:

- A transfer from a fixed address to another fixed address
- A transfer from a fixed address to an address range in memory
- A transfer from an address range in memory to a fixed address
- A transfer from an address range in memory to another address range in memory

When increment addressing mode is selected, SADR and DADR increment by 1 in a 8-bit transfer and by 2 in a 16-bit transfer after a data transfer for the next transfer. When SADR or DADR exceeds FFFFFFFFh by the incrementation, it returns to 00000000h. Likewise, when SADR or DADR exceeds 01FFFFFFh, it becomes 02000000h, but an actual transfer is performed for FE000000h.

13.3.2 Immediate Data Transfer

DMAC II transfers immediate data to a given memory location. Either incrementing or non-incrementing addressing mode can be selected for the destination address. Store the immediate data to be transferred into SADR. To transfer 8-bit immediate data, set the data to the lower 1 byte of SADR. The upper 3 bytes are ignored. To transfer 16-bit immediate data, set the data to the lower 2 bytes. The upper 2 bytes are ignored.

13.3.3 Calculation Result Transfer

After two memory data or immediate data and memory data are added together, DMAC II transfers the calculated result to a given memory location. Set an address to be calculated or immediate data to SADR and set the other address to be calculated to OADR. Either incrementing or non-incrementing addressing mode can be selected for source and destination addresses when performing data in memory + data in memory calculation transfer. If the source addressing is in incrementing mode, the operation addressing is also in incrementing mode. When performing immediate data + data in memory calculation transfer, the addressing mode is selectable only for the destination address.

13.4 Transfer Modes

DMAC II provides three types of basic transfer mode: single transfer, burst transfer, and multiple transfer. COUNT determines the number of transfers to be performed. Transfers are not performed when COUNT is set to 0000h.

13.4.1 Single Transfer

Set the BRST bit in the MOD to 0.
A single data transfer is performed by one transfer request.
When incrementing addressing mode is selected for the source and/or destination address, the address or addresses increment after a data transfer for the next transfer.
COUNT is decremented each time a data transfer is performed. When COUNT reaches 0000h, the DMA II transfer complete interrupt request is generated if the INTE bit in the MOD is 1 (DMA II transfer complete interrupt used).

13.4.2 Burst Transfer

Set the BRST bit in the MOD to 1.
DMAC II continuously transfers data for the number of times determined by COUNT with one transfer request. COUNT decrements each time a data transfer is performed. When COUNT reaches 0000h, the burst transfer is completed. The DMA II transfer complete interrupt request is generated if the INTE bit is 1 (DMA II transfer complete interrupt used).
No interrupts are accepted during a burst transfer.

13.4.3 Multiple Transfer

Set the MULT bit in the MOD to 1.
Multiple memory-to-memory transfers are performed from different source addresses to different destination addresses using one transfer request.
Set bits CNT2 to CNT0 in the MOD to select the number of transfers to be performed from 001b (once) to 111b (seven times). These bits should not be set to 000b.
Allocate the required number of SDARs and DADRs alternately following MOD and COUNT.
When the multiple transfer is selected, the following transfer functions are not available: calculation result transfer, burst transfer, chained transfer, and DMA II transfer complete interrupt.

13.5 Chained Transfer

The chained transfer is available when the CHAIN bit in the MOD is set to 1.

The chained transfer is performed as follows:

(1) When a transfer request is generated, a data transfer is performed according to the DMAC II index specified by the corresponding interrupt vector. Either a single transfer (the BRST bit in the MOD is 0) or burst transfer (the BRST bit is 1) is performed according to the BRST bit setting.

(2) When COUNT reaches 0000h, the value in the interrupt vector in (1) above is overwritten with the value in CADR. Simultaneously, the DMA II transfer complete interrupt is generated when the INTE bit in the MOD is 1.

(3) When the next DMA II transfer request is generated, the data transfer is performed according to the DMAC II index specified by the peripheral interrupt vector in (2) above.

Figure 13.4 shows the relocatable vector and DMAC II index in a chained transfer.

To use the chained transfer, the relocatable vector table should be allocated on the RAM.

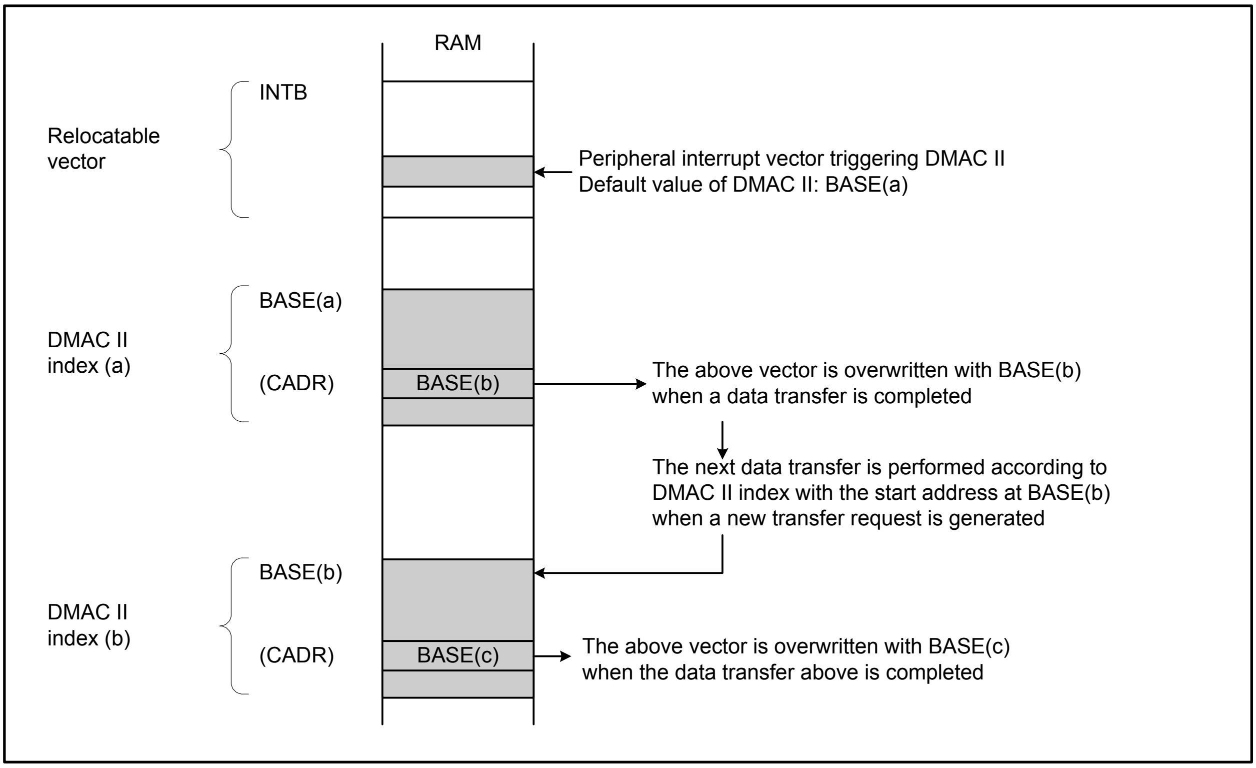


Figure 13.4 Relocatable Vector and DMAC II Index in a Chained Transfer

13.6 DMA II Transfer Complete Interrupt

The DMA II transfer complete interrupt is available when the INTE bit in the MOD is set to 1.

The start address of the DMA II transfer complete interrupt handler should be set to IADR. The interrupt is generated when COUNT reaches 0000h.

The initial instruction of the interrupt handler is executed in the eighth cycle after a DMA II transfer is completed.

13.7 Execution Time

The DMAC II execution cycle is calculated by the following equations:

Mode other than multiple transfer: $t = 6 + (26 + a + b + c + d) \times m + (4 + e) \times n$ cycles
When using multiple transfer: $t = 21 + (11 + b + c) \times k$ cycles

a: When IMM is 0 (transfer source is immediate data), a is 0;
When IMM is 1 (transfer source is memory), a is -1
b: When UPDS is 1 (source addressing is incrementing), b is 0;
When UPDS is 0 (source addressing is non-incrementing), b is 1
c: When UPDD is 1 (destination addressing is incrementing), c is 0;
When UPDD is 0 (destination addressing is non-incrementing), c is 1
d: When OPER is 0 (calculation transfer is not selected), d is 0;
When OPER is 1 (calculation transfer is selected) and UPDS is 0 (source addressing is immediate data or non-incrementing), d is 7;
When OPER is 1 (calculation transfer is selected) and UPDS is 1 (source addressing is incrementing), d is 8
e: When CHAIN is 0 (chained transfer is not selected), e is 0;
When CHAIN is 1 (chained transfer is selected), e is 4
m: When BRST is 0 (single transfer), m is 1;
When BRST is 1 (burst transfer), m is COUNT
n: When COUNT is 0001h, n is 0; if COUNT is 0002h or more, n is 1
k: The number of transfers set using bits CNT2 to CNT0

The equations above are estimations. The number of cycles may vary depending on CPU state, bus wait state, and DMAC II index allocation.

The following assumes a DMA II transfer complete interrupt (transfer counter = 2) is generated with no chained transfer after a memory-to-memory transfer is performed twice with a incremented source address and a non-incremented destination address in single transfer mode.

($a = -1, b = 0, c = 1, d = 0, e = 0, m = 1$)

First DMA II transfer $t = 6 + (26 - 1 + 0 + 1 + 0) \times 1 + (4 + 0) \times 1 = 36$ cycles
Second DMA II transfer $t = 6 + (26 - 1 + 0 + 1 + 0) \times 1 + (4 + 0) \times 0 = 32$ cycles

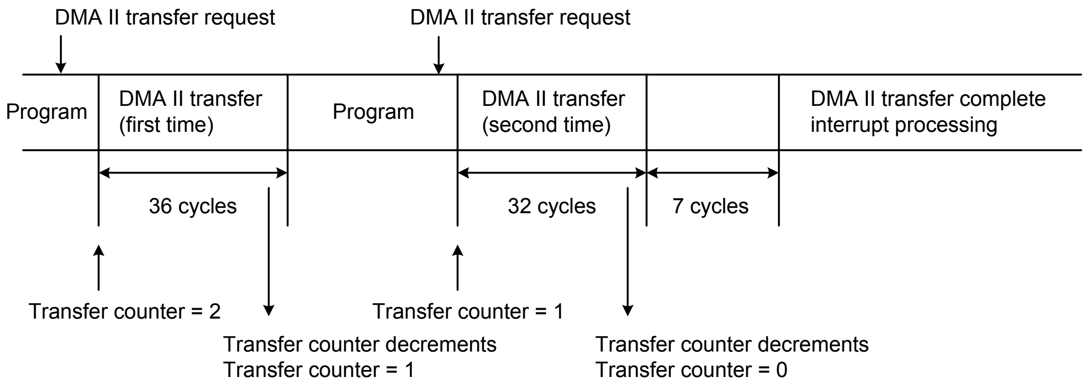
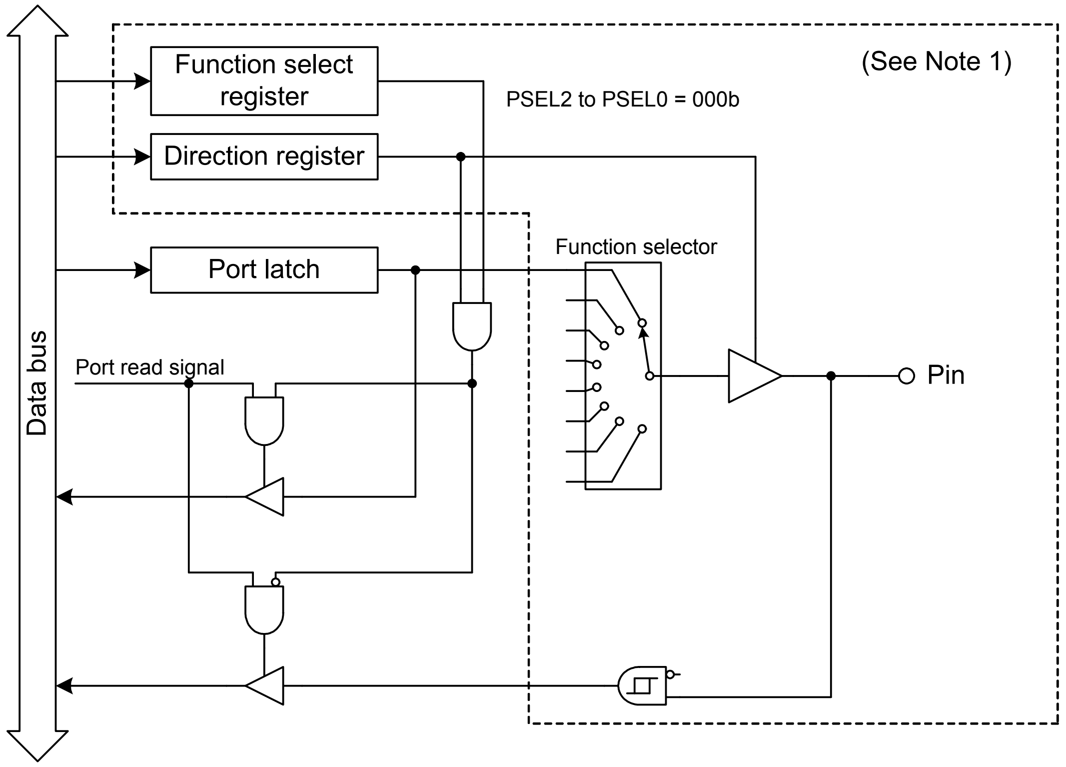


Figure 13.5 Transfer Cycles

14. Programmable I/O Ports

There are 84 programmable I/O ports designated from P0 to P10 (excluding P8_5, and P9_0 to P9_2). Each port status, input or output, can be selected using the direction register except P8_5 and P9_1 which are input only. The P8_5 bit in the P8 register indicates an $\overline{\text{NMI}}$ input level since the P8_5 shares a pin with the $\overline{\text{NMI}}$.

Figure 14.1 shows a configuration of programmable I/O ports, and Figure 14.2 and Figure 14.3 show a configuration of each input-only port.



Note:
1. Refer to 27. “I/O Pins” for details on the area enclosed with the dotted line above.

Figure 14.1 Programmable I/O Port Configuration

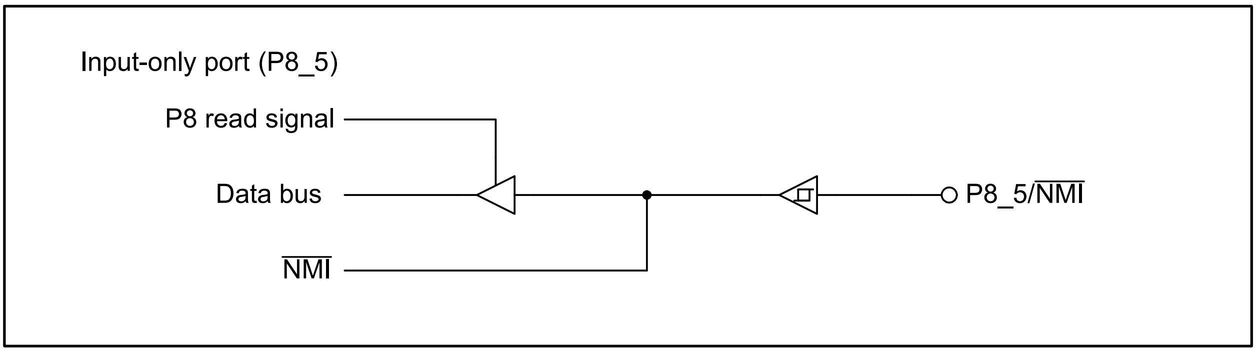


Figure 14.2 Input-only Port Configuration (1/2)

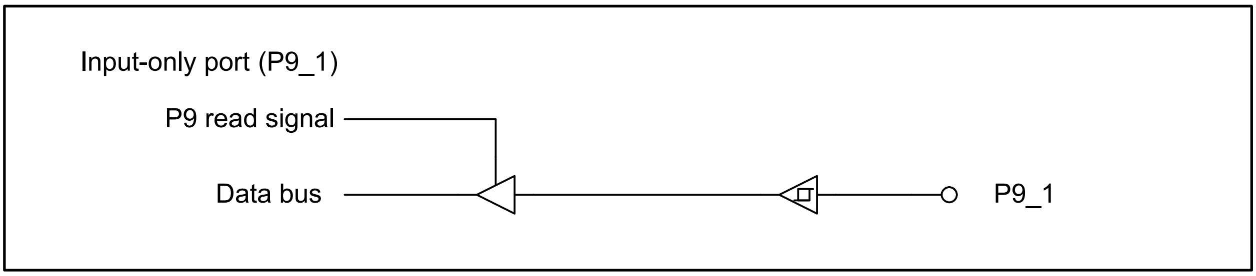


Figure 14.3 Input-only Port Configuration (2/2)

14.1 Port Pi Register (Pi register, i = 0 to 10)

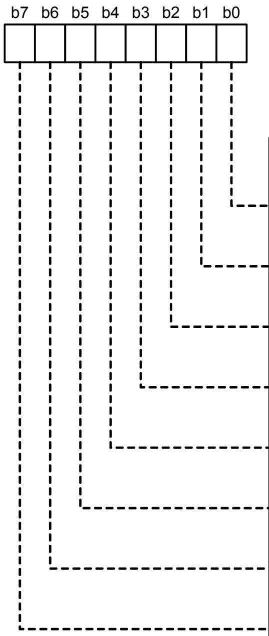
A write/read operation to the Pi register is required to communicate with external devices. This register consists of a port latch to hold output data and a circuit to read pin states. Bits in the Pi register correspond to respective ports.

When a programmable I/O port is selected in the output function select register, the value in the port latch is read for output and the pin state is read for input.

Figure 14.4 shows the Pi register.

Port Pi Register (i = 0 to 10)

Symbol	Address	Reset Value
P0 to P3	03C0h, 03C1h, 03C4h, 03C5h	Undefined
P4 to P7	03C8h, 03C9h, 03CCh, 03CDh	Undefined
P8 (1), P9 (1, 2), P10	03D0h, 03D1h, 03D4h	Undefined



Bit Symbol	Bit Name	Function	RW
Pi_0	Port Pi_0 Bit (2)	When the direction bit is 0 (input) A value is written to the corresponding bit. It is not output due to input mode selected. The read value is the corresponding pin state as follows: 0: Low 1: High When the direction bit is 1 (output) A value is written to the corresponding bit as follows: 0: Output low 1: Output high The read value has the same output level as that written to the corresponding bit	RW
Pi_1	Port Pi_1 Bit (1)		RW
Pi_2	Port Pi_2 Bit (2)		RW
Pi_3	Port Pi_3 Bit		RW
Pi_4	Port Pi_4 Bit		RW
Pi_5	Port Pi_5 Bit (1)		RW
Pi_6	Port Pi_6 Bit		RW
Pi_7	Port Pi_7 Bit		RW

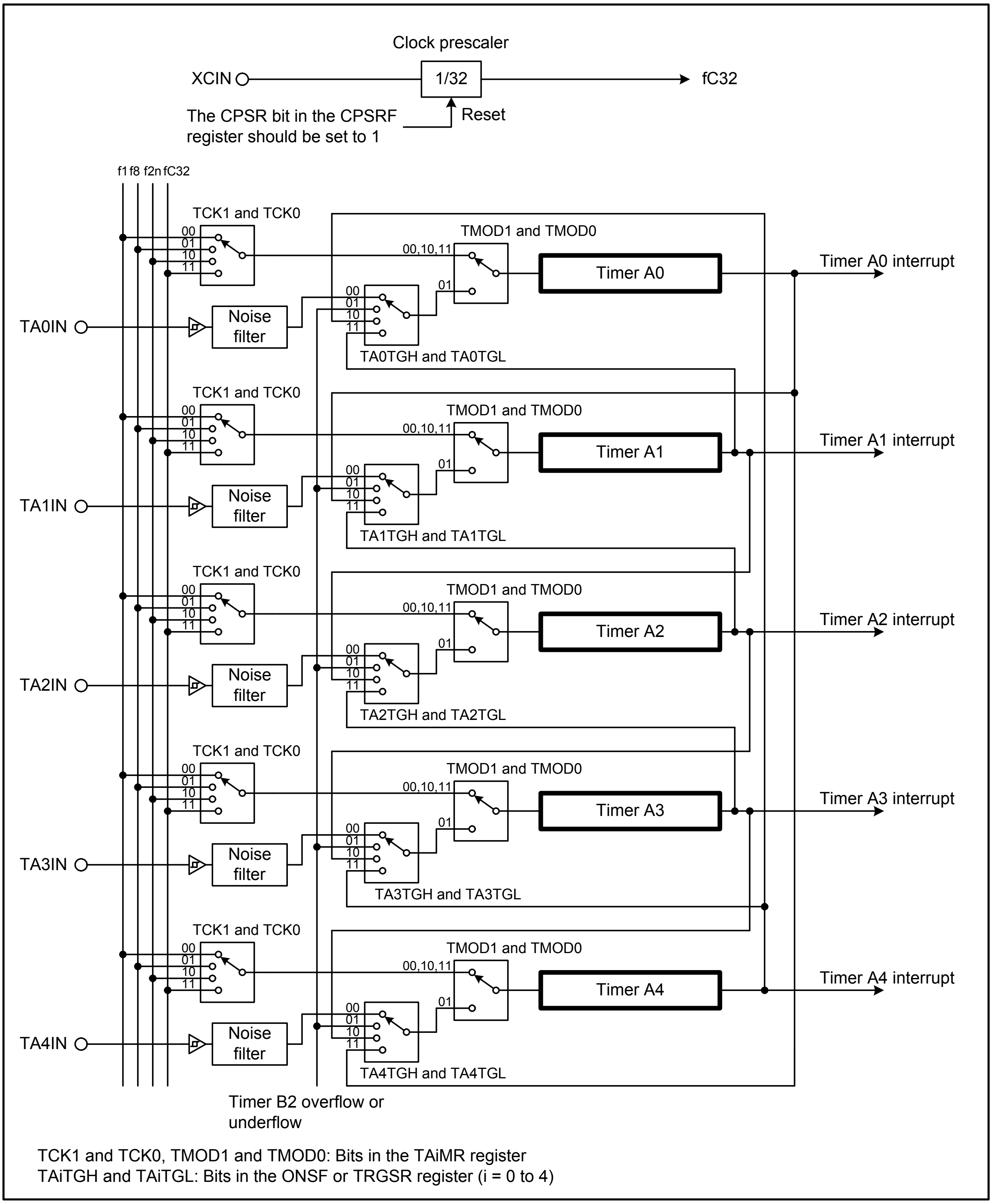
Notes:
1. The P8_5 bit in the P8 register and the P9_1 bit in the P9 register are read only.
2. Bits P9_0 and P9_2 in the P9 register are reserved. These bits should be written with 0 and read as undefined values.

Figure 14.4 Registers P0 to P10

15. Timers

This MCU has eleven 16-bit timers which are divided into two groups according to their functions: five timer As and six timer Bs. Each timer functions individually. The count source of each timer provides the clock for timer operations such as counting and reloading.

Figure 15.1 and Figure 15.2 show the configuration of timers A and B, respectively.



TCK1 and TCK0, TMOD1 and TMOD0: Bits in the TAiMR register
TAiTGH and TAiTGL: Bits in the ONSF or TRGSR register (i = 0 to 4)

Figure 15.1 Timer A Configuration

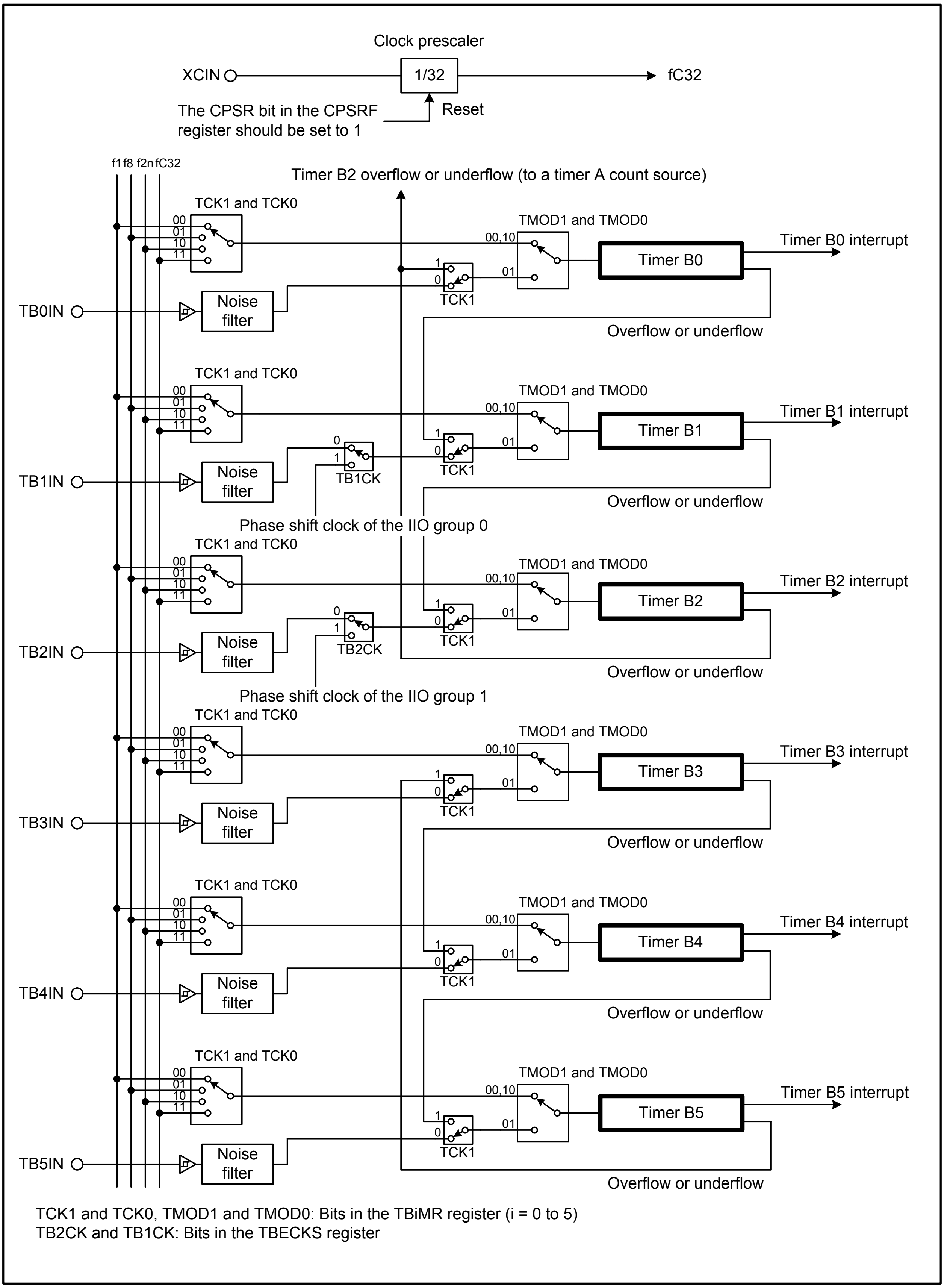


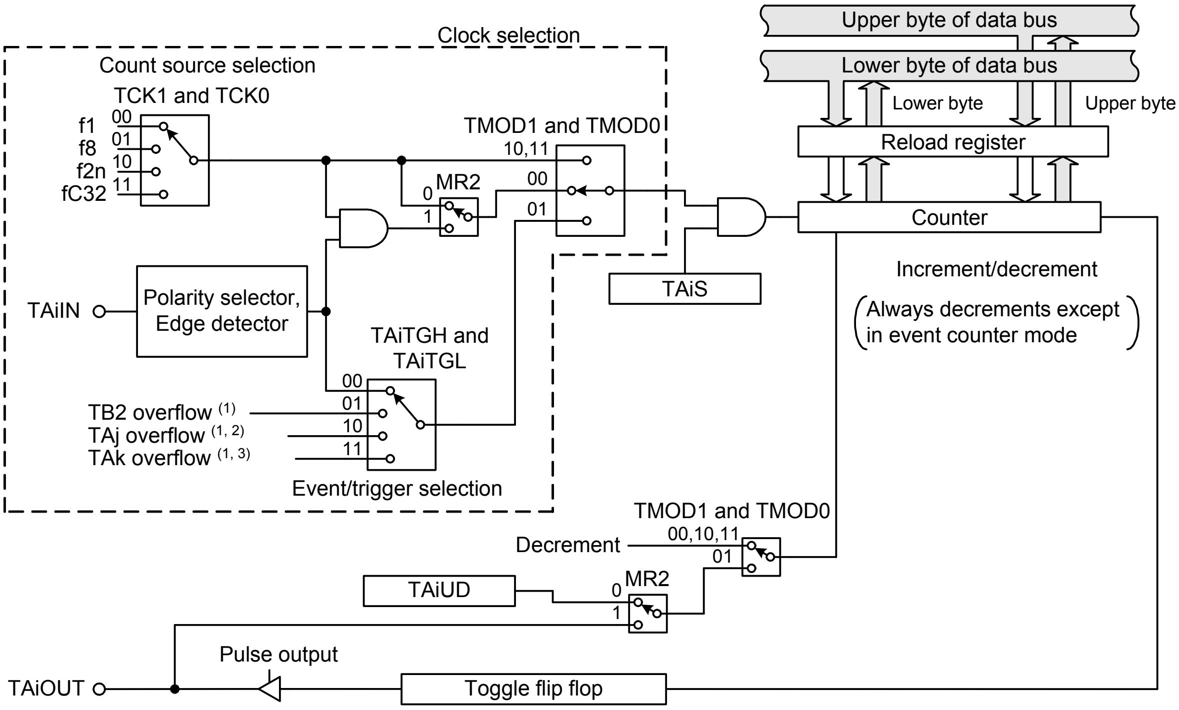
Figure 15.2 Timer B Configuration

15.1 Timer A

Figure 15.3 shows a block diagram of timer A and Figure 15.4 to Figure 15.10 show registers associated with timer A.

Timer A supports the four modes shown below. Timers A0 to A4 in any mode other than the event counter mode have the same function. Select a mode by setting bits TMOD1 and TMOD0 in the TAiMR register (i = 0 to 4).

- Timer mode: The timer counts an internal count source
- Event counter mode: The timer counts an external pulse or overflow and underflow of other timers
- One-shot timer mode: The timer outputs one valid pulse before the counter reaches 0000h
- Pulse-width modulation mode: The timer successively outputs pulses of a given width



i = 0 to 4

Notes:

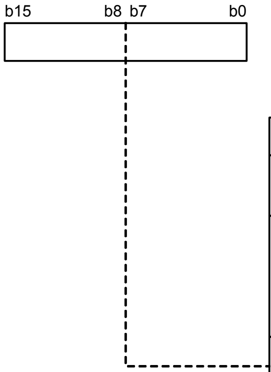
1. The timer overflows or underflows.
2. j = i - 1, or j = 4 if i = 0 (refer to the list on the right)
3. k = i + 1, or k = 0 if i = 4 (refer to the list on the right)

TAi	TAj	TAk
Timer A0	Timer A4	Timer A1
Timer A1	Timer A0	Timer A2
Timer A2	Timer A1	Timer A3
Timer A3	Timer A2	Timer A4
Timer A4	Timer A3	Timer A0

TCK1 and TCK0, TMOD1 and TMOD0, MR2: Bits in the TAiMR register
TAiTGH and TAiTGL: Bits in the ONSF register (i = 0) or in the TRGSR register (i = 1 to 4)
TAiS: Bits in the TABSR register
TAiUD: Bits in the UDF register

Figure 15.3 Timer A Block Diagram

Timer Ai Register (i = 0 to 4) (1)



Symbol	Address	Reset Value
TA0 to TA2	0347h-0346h, 0349h-0348h, 034Bh-034Ah	Undefined
TA3, TA4	034Dh-034Ch, 034Fh-034Eh	Undefined

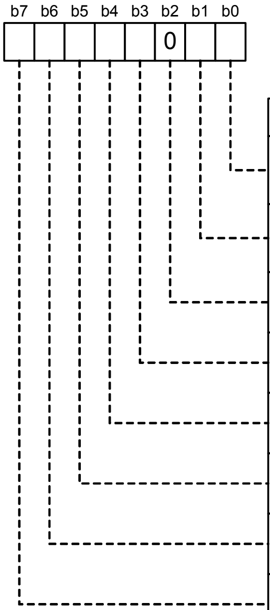
Mode	Function	Setting Range	RW
Timer Mode	Divides the count source by n+1 (n = setting value)	0000h to FFFFh	RW
Event Counter Mode	Divides the count source by FFFFh -n+1 (when incrementing) or by n+1 (when decrementing) (n = setting value) (2)	0000h to FFFFh	RW
One-shot Timer Mode	Divides the count source by n, then stops (n = setting value) (3)	0000h to FFFFh (4)	WO
Pulse-width Modulation Mode (16-bit PWM)	PWM period: $(2^{16}-1)$ / fj High level width of PWM pulse: n / fj (fj = count source frequency, n = setting value of the TAi register) (5)	0000h to FFFEh (4)	WO
Pulse-width Modulation Mode (8-bit PWM)	PWM period: $(2^{8}-1) \times (m+1)$ / fj High level width of PWM pulse: (m+1)n / fj (fj = count source frequency, n = setting value of the upper byte in the TAi register, m = setting value of the lower byte in the TAi register) (5)	00h to FEh (upper byte) 00h to FFh (lower byte) (4)	WO

fj: f1, f8, f2n, fC32

Notes:
1. A 16-bit read/write access to this register should be performed.
2. The timer counts an external input pulse or overflow and underflow of other timers.
3. When the TAi register is set to 0000h, the timer counter does not start, and the TAi interrupt request is not generated.
4. The MOV instruction should be used to set the TAi register.
5. When the TAi register is set to 0000h, the pulse-width modulator does not operate, the TAiOUT pin is held low, and the TAi interrupt request is not generated. The same restrictions apply in 8-bit pulse-width modulator mode if the upper byte in the TAi register is set to 00h.

Figure 15.4 Registers TA0 to TA4

Timer Ai Mode Register (i = 0 to 4)

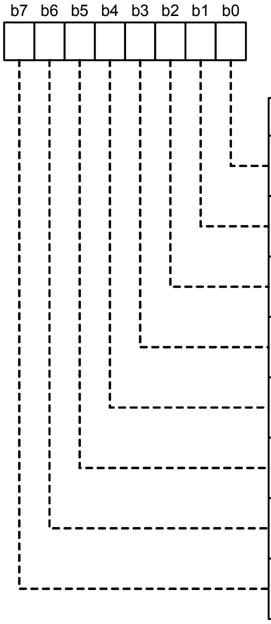


Symbol	Address	Reset Value
TA0MR to TA4MR	0356h, 0357h, 0358h, 0359h, 035Ah	0000 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : One-shot timer mode 1 1 : Pulse-width modulation mode	RW
TMOD1			RW
— (b2)	Reserved	Should be written with 0	RW
MR1	—	Function varies according to the operating mode	RW
MR2			RW
MR3			RW
TCK0	Count Source Select Bit	Function varies according to the operating mode	RW
TCK1			RW

Figure 15.5 Registers TA0MR to TA4MR

Count Start Register

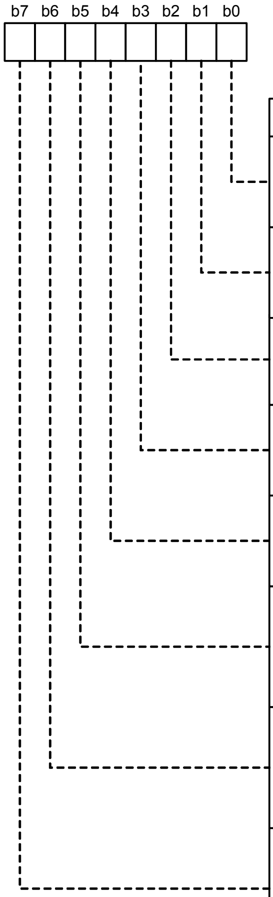


Symbol	Address	Reset Value
TABSR	0340h	0000 0000b

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 Count Start Bit	0: Stop counter 1: Start counter	RW
TA1S	Timer A1 Count Start Bit	0: Stop counter 1: Start counter	RW
TA2S	Timer A2 Count Start Bit	0: Stop counter 1: Start counter	RW
TA3S	Timer A3 Count Start Bit	0: Stop counter 1: Start counter	RW
TA4S	Timer A4 Count Start Bit	0: Stop counter 1: Start counter	RW
TB0S	Timer B0 Count Start Bit	0: Stop counter 1: Start counter	RW
TB1S	Timer B1 Count Start Bit	0: Stop counter 1: Start counter	RW
TB2S	Timer B2 Count Start Bit	0: Stop counter 1: Start counter	RW

Figure 15.6 TABSR Register

Increment/Decrement Select Register (1)



Symbol
UDF

Address
0344h

Reset Value
0000 0000b

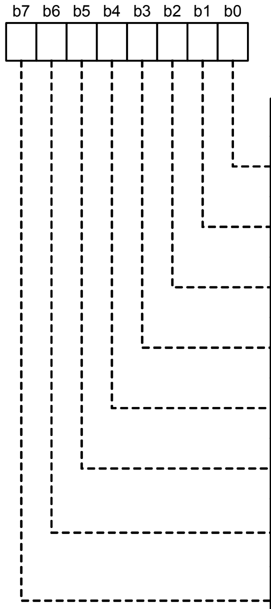
Bit Symbol	Bit Name	Function	RW
TA0UD	Timer A0 Increment/Decrement Select Bit	0: Count decremented 1: Count incremented (2)	RW
TA1UD	Timer A1 Increment/Decrement Select Bit	0: Count decremented 1: Count incremented (2)	RW
TA2UD	Timer A2 Increment/Decrement Select Bit	0: Count decremented 1: Count incremented (2)	RW
TA3UD	Timer A3 Increment/Decrement Select Bit	0: Count decremented 1: Count incremented (2)	RW
TA4UD	Timer A4 Increment/Decrement Select Bit	0: Count decremented 1: Count incremented (2)	RW
TA2P	Timer A2 Two-phase Pulse Signal Processing Select Bit	0: Two-phase pulse signal processing disabled 1: Two-phase pulse signal processing enabled (3)	WO
TA3P	Timer A3 Two-phase Pulse Signal Processing Select Bit	0: Two-phase pulse signal processing disabled 1: Two-phase pulse signal processing enabled (3)	WO
TA4P	Timer A4 Two-phase Pulse Signal Processing Select Bit	0: Two-phase pulse signal processing disabled 1: Two-phase pulse signal processing enabled (3)	WO

Notes:

1. The MOV instruction should be used to set this register.
2. This bit is enabled in event counter mode and when the MR2 bit in the TAiMR register (i = 0 to 4) is set to 0 (the UDF register setting is the source of increment/decrement switching).
3. This bit should be set to 0 when not using two-pulse signal processing.

Figure 15.7 UDF Register

One-shot Start Register



Symbol	Address	Reset Value
ONSF	0342h	0000 0000b

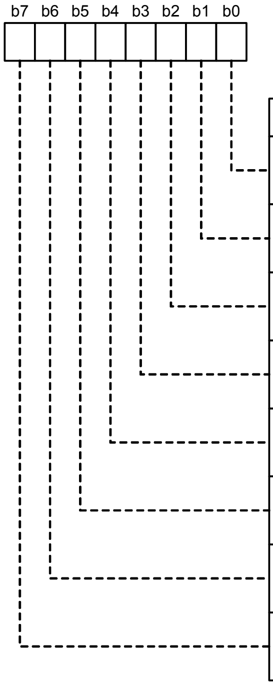
Bit Symbol	Bit Name	Function	RW
TA0OS	Timer A0 One-shot Start Bit	0: Timer in idle state 1: Start the timer (1)	RW
TA1OS	Timer A1 One-shot Start Bit	0: Timer in idle state 1: Start the timer (1)	RW
TA2OS	Timer A2 One-shot Start Bit	0: Timer in idle state 1: Start the timer (1)	RW
TA3OS	Timer A3 One-shot Start Bit	0: Timer in idle state 1: Start the timer (1)	RW
TA4OS	Timer A4 One-shot Start Bit	0: Timer in idle state 1: Start the timer (1)	RW
TAZIE	Z-phase Input Enable Bit	0: Z-phase input disabled 1: Z-phase input enabled	RW
TA0TGL	Timer A0 Event/Trigger Select Bit	b7 b6 0 0 : Select the input to the TA0IN pin 0 1 : Select the overflow of TB2 (2) 1 0 : Select the overflow of TA4 (2) 1 1 : Select the overflow of TA1 (2)	RW
TA0TGH			RW

Notes:

1. This bit is read as 0.
2. The timer overflows or underflows.

Figure 15.8 ONSF Register

Trigger Select Register



Symbol TRGSR
Address 0343h
Reset Value 0000 0000b

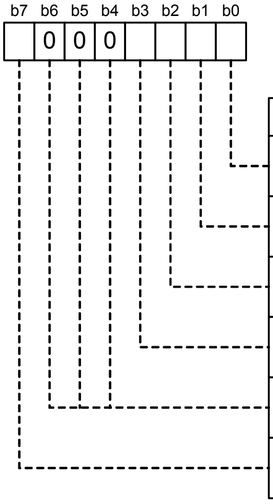
Bit Symbol	Bit Name	Function	RW
TA1TGL	Timer A1 Event/Trigger Select Bit	b1 b0 0 0 : Select the input to the TA1IN pin 0 1 : Select the overflow of TB2 (1) 1 0 : Select the overflow of TA0 (1) 1 1 : Select the overflow of TA2 (1)	RW
TA1TGH			RW
TA2TGL	Timer A2 Event/Trigger Select Bit	b2 b3 0 0 : Select the input to the TA2IN pin 0 1 : Select the overflow of TB2 (1) 1 0 : Select the overflow of TA1 (1) 1 1 : Select the overflow of TA3 (1)	RW
TA2TGH			RW
TA3TGL	Timer A3 Event/Trigger Select Bit	b4 b5 0 0 : Select the input to the TA3IN pin 0 1 : Select the overflow of TB2 (1) 1 0 : Select the overflow of TA2 (1) 1 1 : Select the overflow of TA4 (1)	RW
TA3TGH			RW
TA4TGL	Timer A4 Event/Trigger Select Bit	b6 b7 0 0 : Select the input to the TA4IN pin 0 1 : Select the overflow of TB2 (1) 1 0 : Select the overflow of TA3 (1) 1 1 : Select the overflow of TA0 (1)	RW
TA4TGH			RW

Note:

1. The timer overflows or underflows.

Figure 15.9 TRGSR Register

Count Source Prescaler Register



Symbol	Address	Reset Value
TCSPR	035Fh	0000 0000b

Bit Symbol	Bit Name	Function	RW
CNT0	Divide Ratio Select Bit (1)	f2n is either the main clock or peripheral clock source divided by 2n. If n = 0, the clock is not divided (n = setting value)	RW
CNT1			RW
CNT2			RW
CNT3			RW
— (b6-b4)	Reserved	Should be written with 0	RW
CST	Divider Operation Enable Bit	0: Stop divider operation 1: Start divider operation	RW

Note:

1. The CST bit should be set to 0 before rewriting bits CNT3 to CNT0.

Figure 15.10 TCSPR Register

15.1.1 Timer Mode

In timer mode, the timer counts an internally generated count source. Table 15.1 lists the specifications of timer mode. Figure 15.11 shows registers TA0MR to TA4MR in this mode.

Table 15.1 Timer Mode Specifications (i = 0 to 4)

Item	Specification
Count sources	f1, f8, f2n, or fC32
Count operations	• Decrement • When the timer counter underflows, the reload register value is reloaded into the counter to continue counting
Divide ratio	$\frac{1}{n+1}$ n: TAi register setting value, 0000h to FFFFh
Count start condition	Set the TAiS bit in the TABSR register to 1 (start counter)
Count stop condition	Set the TAiS bit in the TABSR register to 0 (stop counter)
Interrupt request generating timing	When the timer counter underflows
TAiIN pin function	Functions as a programmable I/O port or a gate input
TAiOUT pin function	Functions as a programmable I/O port or a pulse output
Read from timer	The TAi register indicates the counter value
Write to timer	• While the timer counter is stopped or before the initial count source is input after starting to count, the value written to the TAi register is written to both the reload register and the counter • While the timer counter is running, the value written to the TAi register is written to the reload register (it is transferred to the counter at the next reload timing)
Selectable functions	• Gate function Input signal to the TAiIN pin can control the count start/stop • Pulse output function The polarity of the TAiOUT pin is inverted each time the timer counter underflows. A low is output while the TAiS bit holds 0 (stop counter)

Timer Ai Mode Register (i = 0 to 4) (timer mode)

b7	b6	b5	b4	b3	b2	b1	b0
		0			0	0	0

Symbol	Address	Reset Value
TA0MR to TA4MR	0356h, 0357h, 0358h, 0359h, 035Ah	0000 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 0 0 : Timer mode	RW
TMOD1			RW
— (b2)	Reserved	Should be written with 0	RW
MR1	Gate Function Select Bit	b4 b3 0 X : No gate function [1] (TAiIN pin functions as programmable I/O port) 1 0 : Count only while the TAiIN pin is held low 1 1 : Count only while the TAiIN pin is held high	RW
MR2			RW
MR3	Should be written with 0 in timer mode		RW
TCK0	Count Source Select Bit	b7 b6 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : fC32	RW
TCK1			RW

Note:

1. X can be set to either 0 or 1.

Figure 15.11 Registers TA0MR to TA4MR in Timer Mode

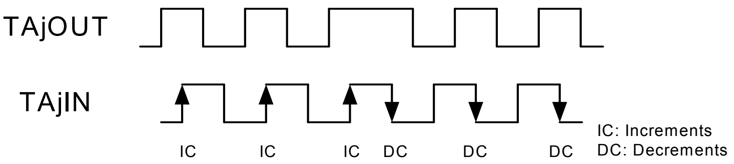
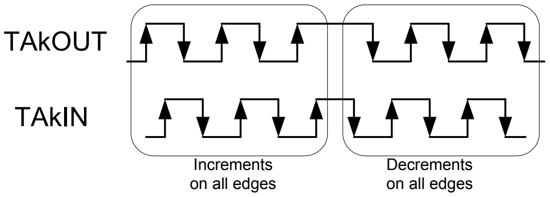
15.1.2 Event Counter Mode

In event counter mode, the timer counts an external signal or an overflow and underflow of other timers. Timers A2, A3, and A4 can count two-phase external signals. Table 15.2 lists the specifications in event count mode and Table 15.3 also lists the specifications when the timers use two-phase pulse signal processing. Figure 15.12 shows registers TA0MR to TA4MR in this mode.

Table 15.2 Event Counter Mode Specifications (without two-phase pulse signal processing) (i = 0 to 4)

Item	Specification
Count sources	• External signal applied to the TAiIN pin (valid edge is selectable by a program) • One of the following: the overflow and/or underflow signal of timer B2, the overflow and/or underflow signal of timer Aj (j = i - 1, or j = 4 if i = 0), or the overflow and/or underflow signal of timer Ak (k = i + 1, or k = 0 if i = 4)
Count operations	• Increment/decrement can be switched by an external signal or program • When the timer counter underflows or overflows, the reload register value is reloaded into the counter to continue counting. In a free-running count operation, the timer counter continues counting without reloading
Divide ratio	• $\frac{1}{FFFFh - n + 1}$ when incrementing • $\frac{1}{n + 1}$ when decrementing n: TAi register setting value, 0000h to FFFFh
Count start condition	Set the TAiS bit in the TABSR register is to 1 (start counter)
Count stop condition	Set the TAiS bit in the TABSR register is to 0 (stop counter)
Interrupt request generating timing	When the timer counter overflows or underflows
TAiIN pin function	Functions as a programmable I/O port or a count source input
TAiOUT pin function	Functions as a programmable I/O port, a pulse output, or an input for switching between increment/decrement
Read from timer	The TAi register indicates a counter value
Write to timer	• While the timer counter is stopped or before the initial count source is input after starting to count, the value written to the TAi register is written to both the reload register and the counter • While the timer counter is running, the value written to the TAi register is written to the reload register (it is transferred to the counter at the next reload timing)
Selectable functions	• Free-running count function The reload register value is not reloaded even if the timer counter overflows or underflows • Pulse output function The polarity of the TAiOUT pin is inverted whenever the timer counter overflows or underflows. A low is output while the TAiS bit holds 0 (stop counter)

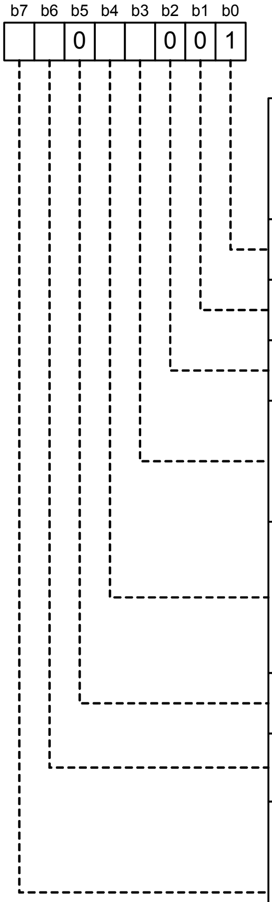
Table 15.3 Event Counter Mode Specifications (with two-phase pulse signal processing on timers A2 to A4) (i = 2 to 4)

Item	Specification
Count sources	Two-phase pulse signal applied to pins TAiIN and TAiOUT
Count operations	• Increment/decrement can be switched by a two-phase pulse signal • When the timer counter underflows or overflows, the reload register value is reloaded into the counter to continue counting. In a free-running count operation, the timer counter continues counting without reloading
Divide ratio	• $\frac{1}{FFFFh - n + 1}$ when incrementing • $\frac{1}{n + 1}$ when decrementing n: TAi register setting value, 0000h to FFFFh
Count start condition	Set the TAiS bit in the TABSR register to 1 (start counter)
Count stop condition	Set the TAiS bit in the TABSR register to 0 (stop counter)
Interrupt request generating timing	When the timer counter overflows or underflows
TAiIN pin function	A two-phase pulse input
TAiOUT pin function	A two-phase pulse input
Read from timer	The TAi register indicates a counter value
Write to timer	• While the timer counter is stopped or before the initial count source is input after starting to count, the value written to the TAi register is written to both the reload register and the counter • While the timer counter is running, the value written to the TAi register is written to the reload register (it is transferred to the counter at the next reload timing)
Selectable functions (1)	• Normal processing operation (timers A2 and A3) While the input signal applied to the TAjOUT pin (j = 2 or 3) is held high, the timer increments on the rising edge of the TAjIN pin and decrements on the falling edge  • Quadrupled processing operation (timers A3 and A4) When the input signal applied to the TAkOUT pin (k = 3 or 4) is held high on the rising edge of the TAkIN pin, the timer increments on both the rising and falling edges of pins TAkOUT and TAkIN. When the signal is held high on the falling edge of the TAkIN pin, the timer decrements on both the rising and falling edges of pins TAkOUT and TAkIN  • Counter reset by Z-phase input (timer A3) The counter value is set to 0 by Z-phase input

Note:

1. Only timer A3 is available for any of the selectable functions. Timer A2 is exclusively for normal processing operations and timer A4 is for the quadrupled processing operation.

Timer Ai Mode Register (i = 0 to 4) (event counter mode)



Symbol	Address	Reset Value
TA0MR to TA4MR	0356h, 0357h, 0358h, 0359h, 035Ah	0000 0000b

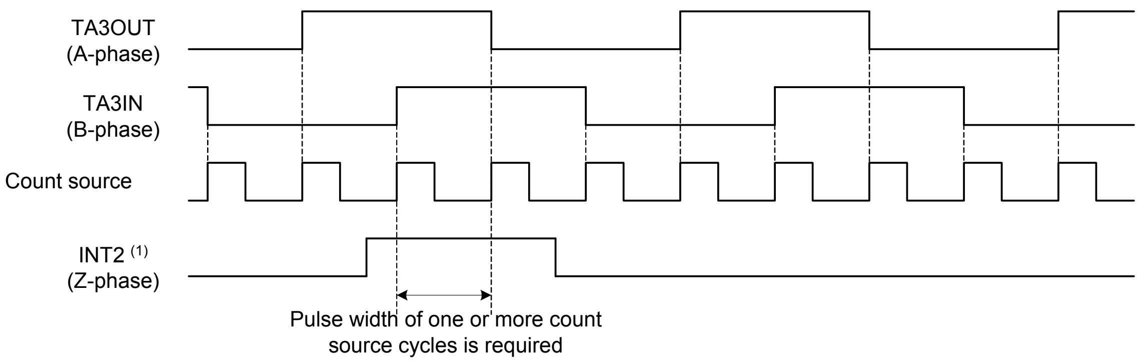
Bit Symbol	Bit Name	Function (without two-phase pulse signal processing)	Function (with two-phase pulse signal processing)	RW
TMOD0	Operating Mode Select Bit	b1 b0 0 1 : Event counter mode (1)		RW
TMOD1				RW
— (b2)	Reserved	Should be written with 0		RW
MR1	Count Polarity Select Bit (2)	0: Count falling edges 1: Count rising edges	Should be written with 0	RW
MR2	Increment/Decrement Switching Source Select Bit	0: UDF register setting 1: Input signal to the TAiOUT pin (3)	Should be written with 1	RW
MR3	Should be written with 0 in event counter mode			RW
TCK0	Count Operation Type Select Bit	0: Reloading 1: Free-running		RW
TCK1	Two-phase Pulse Processing Operation Select Bit (4, 5)	Should be written with 0	0: Normal processing operation 1: Quadrupled processing operation	RW

Notes:
1. Set bits TAiTGH and TAiTGL in the ONSF or TRGSR register to select the count source in event counter mode.
2. This bit setting is enabled only when an external signal is counted.
3. The timer decrements when the input signal to the TAiOUT pin is held low, and increments when the signal is held high.
4. The TCK1 bit is enabled only in the TA3MR register.
5. For two-phase pulse signal processing, the TAjP bit in the UDF register (j = 2 to 4) should be set to 1 (two-phase pulse signal processing enabled) and bits TAjTGH and TAjTGL should be set to 00b (input to the TAjIN pin).

Figure 15.12 Registers TA0MR to TA4MR in Event Counter Mode

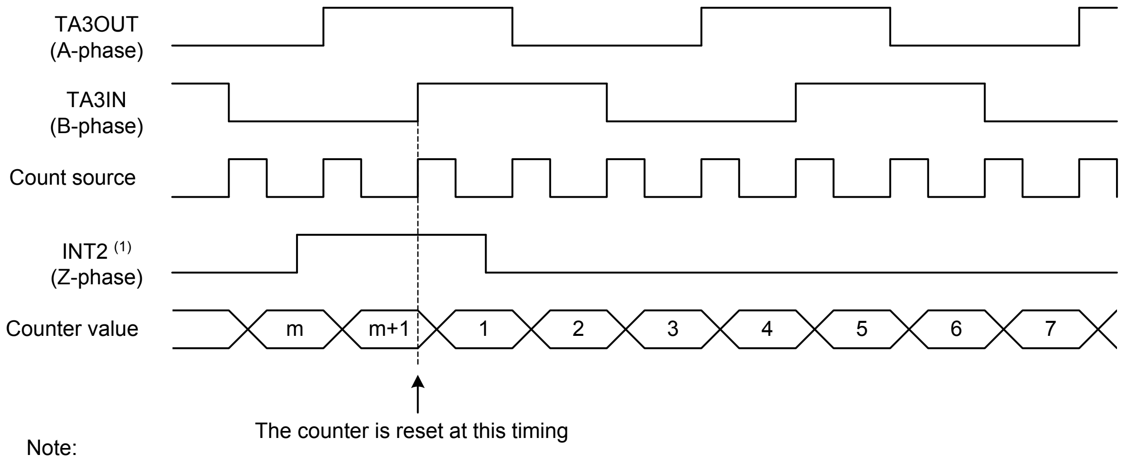
15.1.2.1 Counter Reset by Two-phase Pulse Signal Processing

A Z-phase input signal resets the timer counter when a two-phase pulse signal is being processed.
This function can be used under the following conditions: timer A3 event counter mode, two-phase pulse signal processing, free-running count operation, and quadrupled processing. The Z-phase signal is applied to the $\overline{\text{INT2}}$ pin.
When the TAZIE bit in the ONSF register is set to 1 (Z-phase input enabled), the timer counter can be reset by Z-phase input. To reset the counter, the TA3 register should be set to 0000h beforehand.
A Z-phase signal applied to the $\overline{\text{INT2}}$ pin is detected on an edge. The edge polarity is selected using the POL bit in the INT2IC register. The Z-phase signal should be input in order to have a pulse width of one or more count source cycles for timer A3. Figure 15.13 shows the two-phase pulse (phases A and B) and the Z-phase.
The timer counter is reset at the initial count source input after Z-phase input is detected. Figure 15.14 shows the counter reset timing.
When timer A3 overflows or underflows during a reset by the Z-phase input, two timer A3 interrupt requests are successively generated. To avoid this, the timer A3 interrupt request should not be used when using this function.



Note:
1. This example assumes the signal is detected on the rising edge.

Figure 15.13 Two-phase Pulse (phases A and B) and Z-phase



Note:
1. This example assumes the signal is detected on the rising edge.

Figure 15.14 Counter Reset Timing

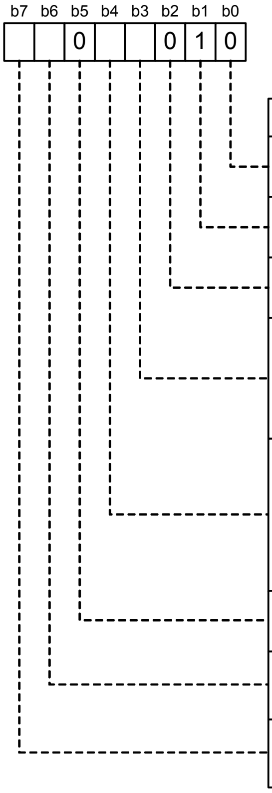
15.1.3 One-shot Timer Mode

In one-shot timer mode, the timer operates only once for each trigger. Table 15.4 lists specifications of one-shot timer mode. Once a trigger occurs, the timer starts and operates for a given period. Figure 15.15 shows registers TA0MR to TA4MR in this mode.

Table 15.4 One-shot Timer Mode Specifications (i = 0 to 4)

Item	Specification
Count sources	f1, f8, f2n, or fC32
Count operations	• Decrement • When the timer counter reaches 0000h, it stops running after the reload register value is reloaded • When a trigger occurs while counting, the reload register value is reloaded into the counter to continue counting
Divide ratio	$\frac{1}{n}$ n: TAi register setting value, 0000h to FFFFh (Note that the timer counter does not run if n = 0000h)
Count start conditions	The TAiS bit in the TABSR register is set to 1 (start counter) and any of following triggers occurs: • An external trigger applied to the TAiIN pin • One of the following: the overflow and/or underflow signal of timer B2, the overflow and/or underflow signal of timer Aj (j = i - 1, or j = 4 if i = 0), or the overflow and/or underflow signal of timer Ak (k = i + 1, or k = 0 if i = 4) • The TAiOS bit in the ONSF register is set to 1 (start the timer)
Count stop conditions	• The timer counter reaches 0000h and the reload register value is reloaded • The TAiS bit in the TABSR register is set to 0 (stop counter)
Interrupt request generating timing	When the timer counter reaches 0000h
TAiIN pin function	A programmable I/O port or a trigger input
TAiOUT pin function	A programmable I/O port or a pulse output
Read from timer	The TAi register indicates an undefined value
Write to timer	• While the timer counter is stopped or before the initial count source is input after starting to count, the value written to the TAi register is written to both the reload register and the counter • While the timer counter is running, the value written to the TAi register is written to the reload register (it is transferred to the counter at the next reload timing)
Selectable function	• Pulse output function A low is output while the timer counter is stopped and a high is output while the timer counter is running

Timer Ai Mode Register (i = 0 to 4) (one-shot timer mode)



Symbol	Address	Reset Value
TA0MR to TA4MR	0356h, 0357h, 0358h, 0359h, 035Ah	0000 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 1 0 : One-shot timer mode	RW
TMOD1			RW
— (b2)	Reserved	Should be written with 0	RW
MR1	External Trigger Select Bit (1)	0: Falling edge of input signal to the TAiIN pin 1: Rising edge of input signal to the TAiIN pin	RW
MR2	Trigger Select Bit	0: TAiOS bit in the ONSF register is enabled 1: Selected using bits TAiTGH and TAiTGL in the ONSF or TRGSR register	RW
MR3	Should be written with 0 in one-shot timer mode		RW
TCK0	Count Source Select Bit	b7 b6 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : fC32	RW
TCK1			RW

Note:

1. The MR1 bit setting is enabled only when bits TAiTGH and TAiTGL in the TRGSR register are set to 00b (input to the TAiIN pin). This bit can be set to either 0 or 1 when bits TAiTGH and TAiTGL are set to 01b (overflow or underflow of TB2), 10b (overflow or underflow of TAi), or 11b (overflow or underflow of TAi).

Figure 15.15 Registers TA0MR to TA4MR in One-shot Timer Mode

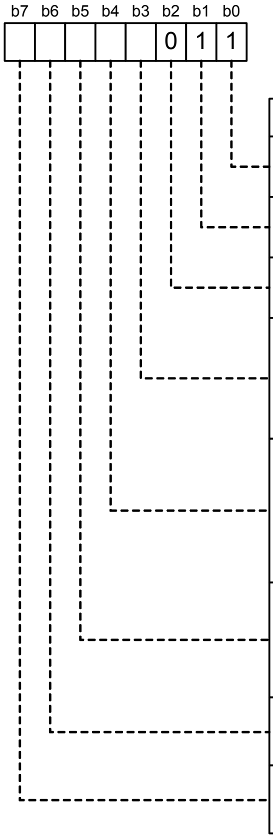
15.1.4 Pulse-width Modulation Mode

In pulse-width modulation mode, the timer outputs pulses of given width successively. Table 15.5 lists specifications of pulse-width modulation mode. The timer counter functions as either a 16-bit or 8-bit pulse-width modulator. Figure 15.16 shows registers TA0MR to TA4MR in this mode. Figure 15.17 and Figure 15.18 show operation examples of 16-bit and 8-bit pulse-width modulators.

Table 15.5 Pulse-width Modulation Mode Specifications (i = 0 to 4)

Item	Specification
Count sources	f1, f8, f2n, or fC32
Count operations	• Decrement (the timer counter functions as an 8-bit or a 16-bit pulse-width modulator) • The reload register value is reloaded on the rising edge of a PWM pulse to continue counting • The timer is not affected by a trigger that occurs while the counter is running
16-bit PWM	• High level width: $\frac{n}{fj}$ n: TAi register setting value, 0000h to FFFEh fj: Count source frequency • Cycles: fixed to $\frac{2^{16}-1}{fj}$
8-bit PWM	• High level width: $\frac{n \times (m+1)}{fj}$ • Cycles: $\frac{(2^{8}-1) \times (m+1)}{fj}$ n: Upper byte of the TAi register setting value, 00h to FEh m: Lower byte of the TAi register setting value, 00h to FFh
Count start conditions	• Set the TAiS bit in the TABSR register to 1 (start counter) • The TAiS bit is 1 and an external trigger is applied to the TAiIN pin • The TAiS bit is 1 and any of following triggers occurs: the overflow and/or underflow signal of timer B2, the overflow and/or underflow signal of timer Aj (j = i - 1, or j = 4 if i = 0), or the overflow and/or underflow signal of timer Ak (k = i + 1, or k = 0 if i = 4)
Count stop condition	The TAiS bit in the TABSR register is set to 0 (stop counter)
Interrupt request generating timing	On the falling edge of the PWM pulse
TAiIN pin function	A programmable I/O port or trigger input
TAiOUT pin function	A pulse output
Read from timer	The TAi register indicates an undefined value
Write to timer	• While the timer counter is stopped or before the initial count source is input after starting to count, the value written to the TAi register is written to both the reload register and the counter • While the timer counter is running, the value written to the TAi register is written to the reload register (it is transferred to the counter at the next reload timing)

Timer Ai Mode Register (i = 0 to 4) (pulse-width modulation mode)



Symbol	Address	Reset Value
TA0MR to TA4MR	0356h, 0357h, 0358h, 0359h, 035Ah	0000 0000b

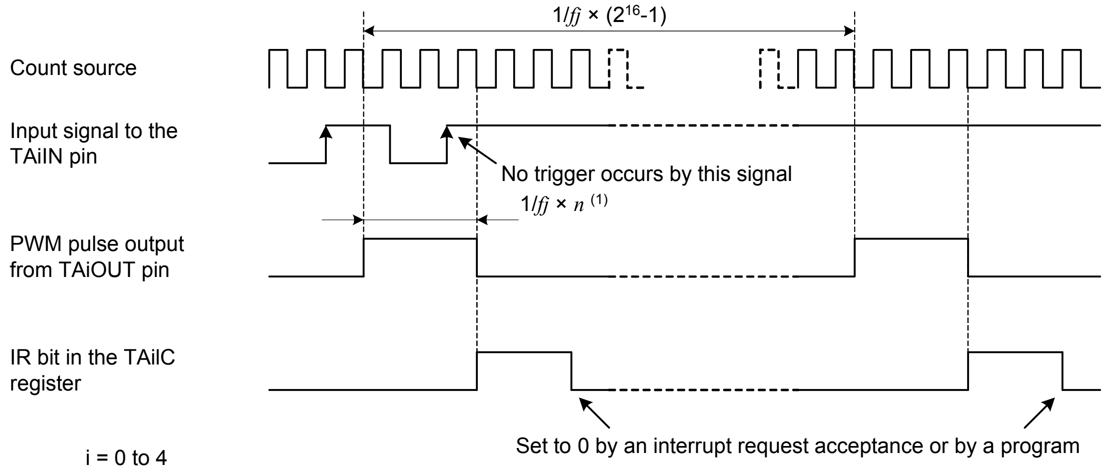
Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 1 1 : Pulse-width modulation (PWM) mode	RW
TMOD1			RW
— (b2)	Reserved	Should be written with 0	RW
MR1	External Trigger Select Bit (1)	0: Falling edge of the input signal to the TAiIN pin 1: Rising edge of the input signal to the TAiIN pin	RW
MR2	Trigger Select Bit	0: TAiS bit in the ONSF register is enabled 1: Selected using bits TAiTGH and TAiTGL in the ONSF or TRGSR register	RW
MR3	16-/8-bit PWM Mode Select Bit	0: Function as a 16-bit pulse-width modulator 1: Function as a 8-bit pulse-width modulator	RW
TCK0	Count Source Select Bit	b7 b6 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : fC32	RW
TCK1			RW

Note:

1. The MR1 bit setting is enabled only when bits TAiTGH and TAiTGL in the TRGSR register are set to 00b (input to the TAiIN pin). This bit can be set to either 0 or 1 when bits TAiTGH and TAiTGL are set to 01b (overflow or underflow of TB2), 10b (overflow or underflow of TAi), or 11b (overflow or underflow of TAi).

Figure 15.16 Registers TA0MR to TA4MR in Pulse-width Modulation Mode

When the reload register is set to 0003h and an external trigger (the rising edge of an input signal applied to the TAiIN pin) is selected.



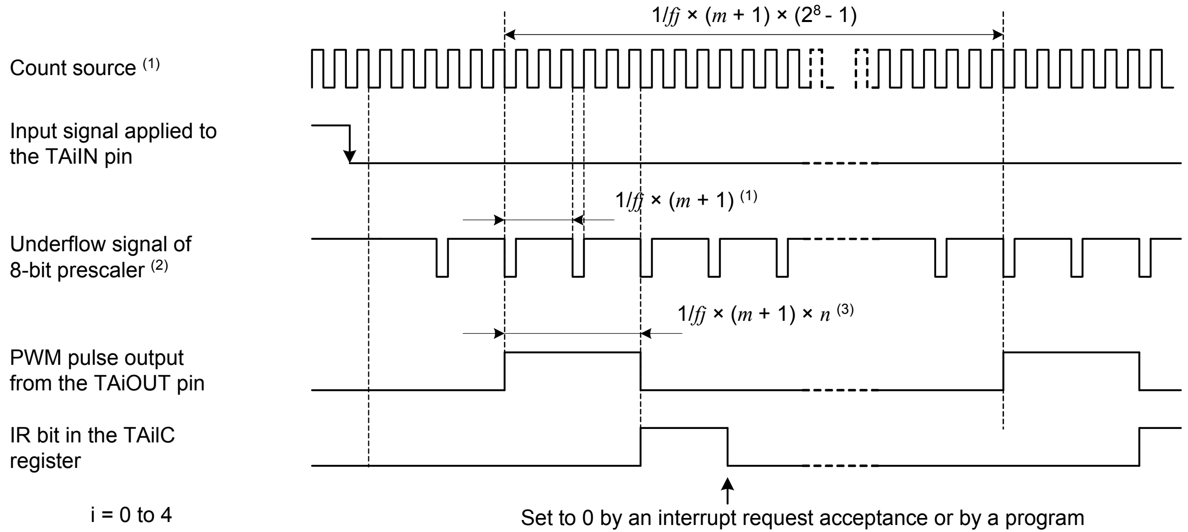
i = 0 to 4
fj: Count source frequency
(f1, f8, f2n, fC32)

Note:

1. n is a value from 0000h to FFFEh.

Figure 15.17 16-bit Pulse-width Modulator Operation

When the upper byte of the reload register is set to 02h, the lower byte is set to 02h and an external trigger (the falling edge of an input signal applied to the TAiIN pin) is selected.



i = 0 to 4
fj: Count source frequency
(f1, f8, f2n, fC32)

Notes:

1. The 8-bit prescaler counts a count source.
2. The 8-bit pulse-width modulator counts underflow signals of the 8-bit prescaler.
3. m is a value from 00h to FFh, and n is a value from 00h to FEh.

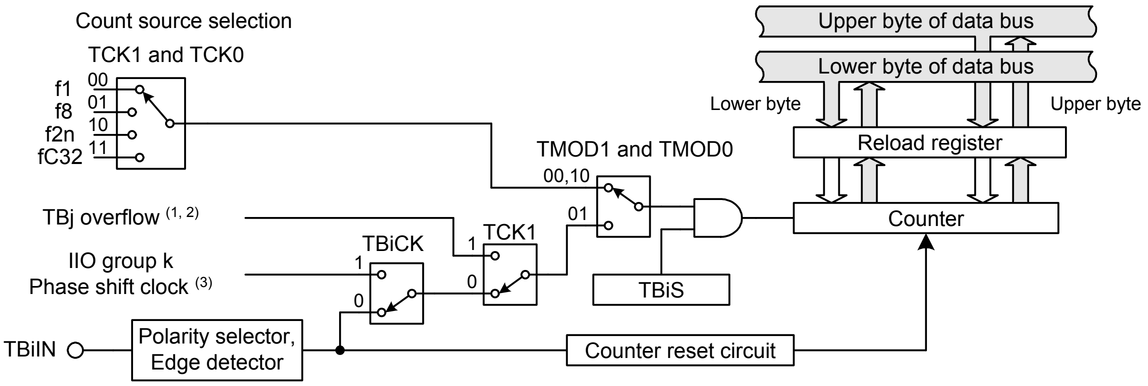
Figure 15.18 8-bit Pulse-width Modulator Operation

15.2 Timer B

Figure 15.19 shows a block diagram of timer B, and Figure 15.20 to Figure 15.23 show registers associated with timer B.

Timer B supports the three modes shown below. Select a mode by setting bits TMOD1 and TMOD0 in the TBiMR register (i = 0 to 5).

- Timer mode: The timer counts an internal count source.
- Event counter mode: The timer counts an external pulse or an overflow and underflow of other timers.
- Pulse period/pulse-width measure mode: The timer measures the pulse period or pulse width of an external signal.



i = 0 to 5

Notes:

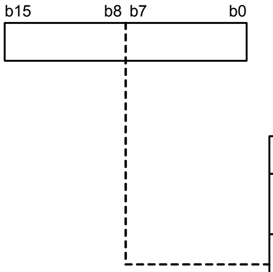
1. The timer overflows or underflows.
2. j = i - 1; j = 2 if i = 0; or j = 5 if i = 3 (refer to the list on the right)
3. k = i - 1 and i = 1 or 2

TCK1 and TCK0, TMOD1 and TMOD0: Bits in the TBiMR register
TBiS: Bits in the TABSR or TBSR register
TBiCK: Bits in the TBECKS register

TBi	TBj
Timer B0	Timer B2
Timer B1	Timer B0
Timer B2	Timer B1
Timer B3	Timer B5
Timer B4	Timer B3
Timer B5	Timer B4

Figure 15.19 Timer B Block Diagram

Timer Bi Register (i = 0 to 5) (1)



Symbol	Address	Reset Value
TB0 to TB2	0351h-0350h, 0353h-0352h, 0355h-0354h	Undefined
TB3 to TB5	0311h-0310h, 0313h-0312h, 0315h-0314h	Undefined

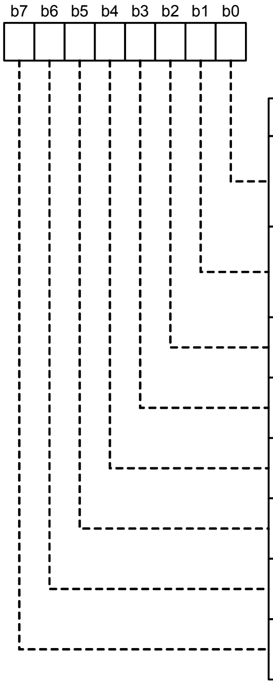
Mode	Function	Setting Range	RW
Timer Mode	Divides the count source by n+1 (n = setting value)	0000h to FFFFh	RW
Event Counter Mode	Divides the count source by n+1 (n = setting value) (2)	0000h to FFFFh	RW
Pulse Period/Pulse-width Measure Mode	Increments the counter between one valid edge and another of TBiIN input pulse	—	RO

Notes:

1. A 16-bit read/write access to this register should be performed.
2. The TBi register counts an external input pulse, or an overflow and underflow of other timers.

Figure 15.20 Registers TB0 to TB5

Timer Bi Mode Register (i = 0 to 5)



Symbol	Address	Reset Value
TB0MR to TB2MR	035Bh, 035Ch, 035Dh	00XX 0000b
TB3MR to TB5MR	031Bh, 031Ch, 031Dh	00XX 0000b

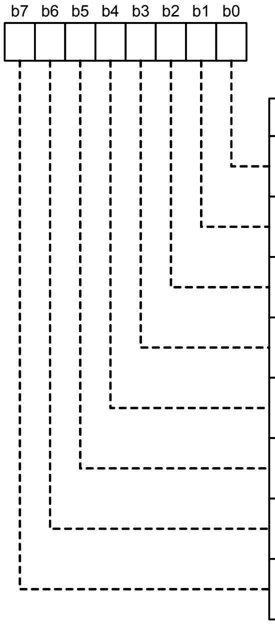
Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : Pulse period measure mode, pulse-width measure mode 1 1 : Do not use this combination	RW
TMOD1			RW
MR0	—	Function varies according to the operating mode (1, 2)	RW
MR1			RW
MR2			RW
MR3			RW
TCK0	Count Source Select Bit	Function varies according to the operating mode	RW
TCK1			RW

Notes:

1. The MR2 bit is available for registers TB0MR and TB3MR only.
2. The MR2 bit in registers TB1MR, TB2MR, TB4MR and TB5MR are unavailable on this MCU. This bit should be written with 0 and read as undefined value.

Figure 15.21 Registers TB0MR to TB5MR

Count Start Register

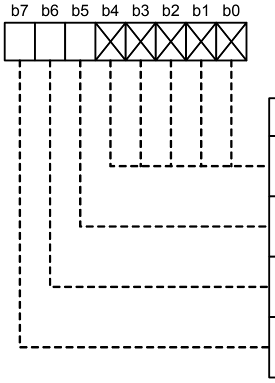


Symbol	Address	Reset Value
TABSR	0340h	0000 0000b

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 Count Start Bit	0: Stop counter 1: Start counter	RW
TA1S	Timer A1 Count Start Bit	0: Stop counter 1: Start counter	RW
TA2S	Timer A2 Count Start Bit	0: Stop counter 1: Start counter	RW
TA3S	Timer A3 Count Start Bit	0: Stop counter 1: Start counter	RW
TA4S	Timer A4 Count Start Bit	0: Stop counter 1: Start counter	RW
TB0S	Timer B0 Count Start Bit	0: Stop counter 1: Start counter	RW
TB1S	Timer B1 Count Start Bit	0: Stop counter 1: Start counter	RW
TB2S	Timer B2 Count Start Bit	0: Stop counter 1: Start counter	RW

Figure 15.22 TABSR Register

Count Start Register for Timers B3, B4, and B5



Symbol	Address	Reset Value
TBSR	0300h	000X XXXXb

Bit Symbol	Bit Name	Function	RW
— (b4-b0)	No register bits; should be written with 0 and read as undefined value		—
TB3S	Timer B3 Count Start Bit	0: Stop counter 1: Start counter	RW
TB4S	Timer B4 Count Start Bit	0: Stop counter 1: Start counter	RW
TB5S	Timer B5 Count Start Bit	0: Stop counter 1: Start counter	RW

Figure 15.23 TBSR Register

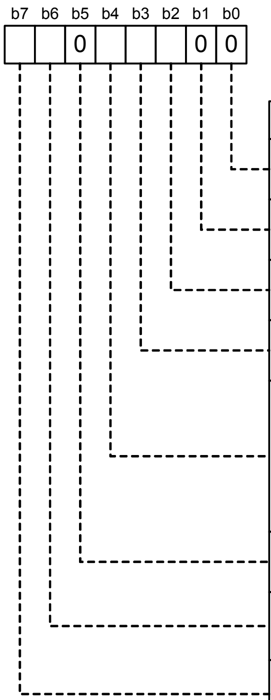
15.2.1 Timer Mode

In timer mode, the timer counts an internally generated count source. Table 15.6 lists specifications of timer mode. Figure 15.24 shows registers TB0MR to TB5MR in this mode.

Table 15.6 Timer Mode Specifications (i = 0 to 5)

Item	Specification
Count sources	f1, f8, f2n, or fC32
Count operations	• Decrement • When the timer counter underflows, the reload register value is reloaded into the counter to continue counting
Divide ratio	$\frac{1}{n+1}$ n: TBi register setting value, 0000h to FFFFh
Count start condition	Set the TBiS bit in the TABSR or TBSR register to 1 (start counter)
Count stop condition	Set the TBiS bit in the TABSR or TBSR register to 0 (stop counter)
Interrupt request generating timing	When the timer counter underflows
TBiIN pin function	Functions as a programmable I/O port
Read from timer	The TBi register indicates a counter value
Write to timer	• While the timer counter is stopped or before the initial count source is input after starting to count, the value written to the TBi register is written to both the reload register and the counter • While the timer counter is running, the value written to the TBi register is written to the reload register (it is transferred to the counter at the next reload timing)

Timer Bi Mode Register (i = 0 to 5) (timer mode)



Symbol	Address	Reset Value
TB0MR to TB2MR	035Bh, 035Ch, 035Dh	00XX 0000b
TB3MR to TB5MR	031Bh, 031Ch, 031Dh	00XX 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 0 0 : Timer mode	RW
TMOD1			RW
MR0	Disabled in timer mode. Can be set to 0 or 1		RW
MR1			RW
MR2	In registers TB0MR and TB3MR: Reserved; should be written with 0		RW
	In registers TB1MR, TB2MR, TB4MR, and TB5MR: No register bit; should be written with 0 and read as undefined value		—
MR3	Disabled in timer mode. Should be written with 0 and read as undefined value		—
TCK0	Count Source Select Bit	b7 b6 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : fC32	RW
TCK1			RW

Figure 15.24 Registers TB0MR to TB5MR in Timer Mode

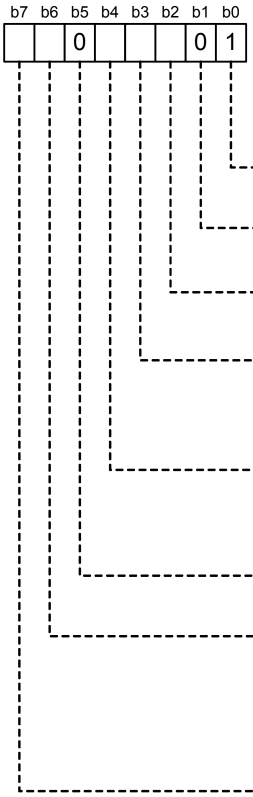
15.2.2 Event Counter Mode

In event counter mode, the timer counts an external signal or the overflow or underflow of other timers. Table 15.7 lists specifications of event counter mode. Figure 15.25 shows the TBiMR register (i = 0 to 5) in this mode.

Table 15.7 Event Counter Mode Specifications (i = 0 to 5)

Item	Specification
Count sources	• External signal applied to the TBiIN pin (valid edge is selectable among the falling edge, the rising edge, or both) • The overflow or underflow signal of TBj (j = i - 1; j = 2 if i = 0; or j = 5 if i = 3) • Phase shift clock of the intelligent I/O group k (k = i - 1 and i = 1 or 2)
Count operations	• Decrement • When the timer counter underflows, the reload register value is reloaded into the counter to continue counting
Divide ratio	$\frac{1}{n+1}$ n: TBi register setting value, 0000h to FFFFh
Count start condition	Set the TBiS bit in the TABSR or TBSR register to 1 (start counter)
Count stop condition	Set the TBiS bit in the TABSR or TBSR register to 0 (stop counter)
Interrupt request generation timing	When the timer counter underflows
TBiIN pin function	Functions as a programmable I/O port or count source input
Read from timer	The TBi register indicates a counter value
Write to timer	• While the timer counter is stopped or before the initial count source is input after starting to count, the value written to the TBi register is written to both the reload register and the counter • While the timer counter is running, the value written to the TBi register is written to the reload register (it is transferred to the counter at the next reload timing)

Timer Bi Mode Register (i = 0 to 5) (event counter mode)



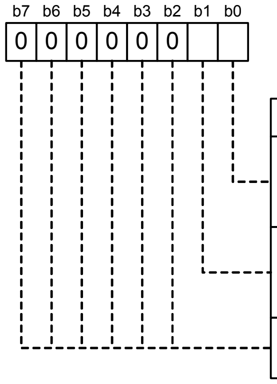
Symbol	Address	Reset Value
TB0MR to TB2MR	035Bh, 035Ch, 035Dh	00XX 0000b
TB3MR to TB5MR	031Bh, 031Ch, 031Dh	00XX 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 0 1 : Event counter mode	RW
TMOD1			RW
MR0	Count Polarity Select Bit [1]	b3 b2 0 0 : Count falling edges 0 1 : Count rising edges 1 0 : Count both edges [2] 1 1 : Do not use this combination	RW
MR1			RW
MR2	In registers TB0MR and TB3MR: Reserved; should be written with 0		RW
	In registers TB1MR, TB2MR, TB4MR, and TB5MR: No register bit; should be written with 0 and read as undefined value		—
MR3	Disabled in event counter mode. Should be written with 0 and read as undefined value		—
TCK0	Disabled in event counter mode. Can be set to 0 or 1		RW
TCK1	Event Clock Select Bit	In registers TB0MR, TB3MR, TB4MR, and TB5MR 0: Input signal to the TBiIN pin 1: Overflow or underflow of TBj [3] In registers TB1MR and TB2MR 0: Select a clock by the TBECKS register 1: Overflow or underflow of TBj [3]	RW

Notes:
1. These bit settings are enabled when the TCK1 bit is set to 0. When the TCK1 is set to 1, these bits can be set to either 0 or 1.
2. When the phase shift clock of the intelligent I/O is selected as event clock, 10b (both edges) should not be set.
3. j = i - 1; j = 2 if i = 0; or j = 5 if i = 3.

Figure 15.25 Registers TB0MR to TB5MR in Event Counter Mode

Timer B Event Clock Select Register (1)



Symbol	Address	Reset Value
TBECKS	0208h	0000 0000b

Bit Symbol	Bit Name	Function	RW
TB1CK	Timer B1 Event Clock Select Bit	0: Select TB1IN 1: Select phase shift clock of the intelligent I/O group 0	RW
TB2CK	Timer B2 Event Clock Select Bit	0: Select TB2IN 1: Select phase shift clock of the intelligent I/O group 1	RW
— (b7-b2)	Reserved	Should be written with 0	RW

Note:

1. If a clock selected by this register is also selected by the TCK1 bit in the TBiMR register (i = 1, 2), that clock functions as the event clock for timer Bi.

Figure 15.26 TBECKS Register

15.2.3 Pulse Period/Pulse-width Measure Mode

In pulse period/pulse-width measure mode, the timer measures the pulse period or pulse width of an external signal. Table 15.8 lists specifications of the pulse period/pulse-width measure mode. Figure 15.27 shows registers TB0MR to TB5MR in this mode. Figure 15.28 and Figure 15.29 show an operation example of pulse period measurement and pulse-width measurement, respectively.

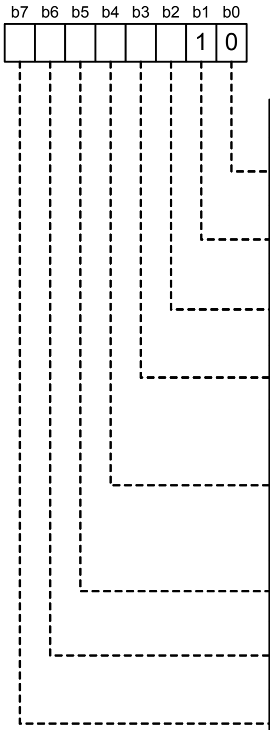
Table 15.8 Pulse Period/Pulse-width Measure Mode Specifications (i = 0 to 5)

Item	Specification
Count sources	f1, f8, f2n, or fC32
Count operations	• Increment • The counter value is transferred to the reload register on the valid edge of a pulse to be measured, then it is set to 0000h to resume counting
Count start condition	Set the TBiS bit in the TABSR or TBSR register to 1 (start counter)
Count stop condition	Set the TBiS bit in the TABSR or TBSR register to 0 (stop counter)
Interrupt request generating timing	• On the valid edge of a pulse to be measured (1) • When the timer counter overflows (when the MR3 bit in the TBiMR register becomes 1 (overflow)) (2)
TBiIN pin function	A pulse input to be measured
Read from timer	The TBi register indicates a reload register value (measurement results) (3)
Write to timer	The value written to the TBi register is written to neither the reload register nor the counter

Notes:

1. No interrupt request is generated when the pulse to be measured is applied on the initial valid edge after the timer counter starts.
2. While the TBiS bit is 1 (start counter), after the MR3 bit becomes 1 (overflow) and at least one count source cycle has elapsed, a write operation to the TBiMR register sets the MR3 bit to 0 (no overflow).
3. The TBi register indicates an undefined value until the pulse to be measured is applied on the second valid edge after the timer counter starts.

Timer Bi Mode Register (i = 0 to 5) (pulse period/pulse-width measure mode)



Symbol	Address	Reset Value
TB0MR to TB2MR	035Bh, 035Ch, 035Dh	00XX 0000b
TB3MR to TB5MR	031Bh, 031Ch, 031Dh	00XX 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1 b0 1 0 : Pulse period/pulse-width measure mode	RW
TMOD1			RW
MR0	Measure Mode Select Bit (1)	b3 b2 0 0 : Pulse period measurement 1 0 1 : Pulse period measurement 2 1 0 : Pulse-width measurement 1 1 : Do not use this combination	RW
MR1			RW
MR2	In registers TB0MR and TB3MR: Reserved; should be written with 0		RW
	In registers TB1MR, TB2MR, TB4MR, and TB5MR: No register bit; should be written with 0 and read as undefined value		—
MR3	Timer Bi Overflow Flag (2)	0: No overflow 1: Overflow	RO
TCK0	Count Source Select Bit	b7 b6 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : fC32	RW
TCK1			RW

Notes:

1. The measure modes selected by setting bits MR1 and MR0 are as follows:
   Pulse period measurement 1 (bits MR1 and MR0 = 00b):
   Measures between a falling edge and the next falling edge of a pulse
   Pulse period measurement 2 (bits MR1 and MR0 = 01b):
   Measures between a rising edge and the next rising edge of a pulse
   Pulse-width measurement (bits MR1 and MR0 = 10b):
   Measures between a falling edge and the next rising edge of a pulse and between the rising edge and the next falling edge of the pulse
2. The MR3 bit is undefined when the timer is reset.
   While the TBiS bit in the TABSR or TBSR register is 1 (start counter), after the MR3 bit becomes 1 and at least one count source cycle has elapsed, a write operation to the TBiMR register sets the MR3 bit to 0 (no overflow). The MR3 bit cannot be set to 1 by a program.

Figure 15.27 Registers TB0MR to TB5MR in Pulse Period/Pulse-width Measure Mode

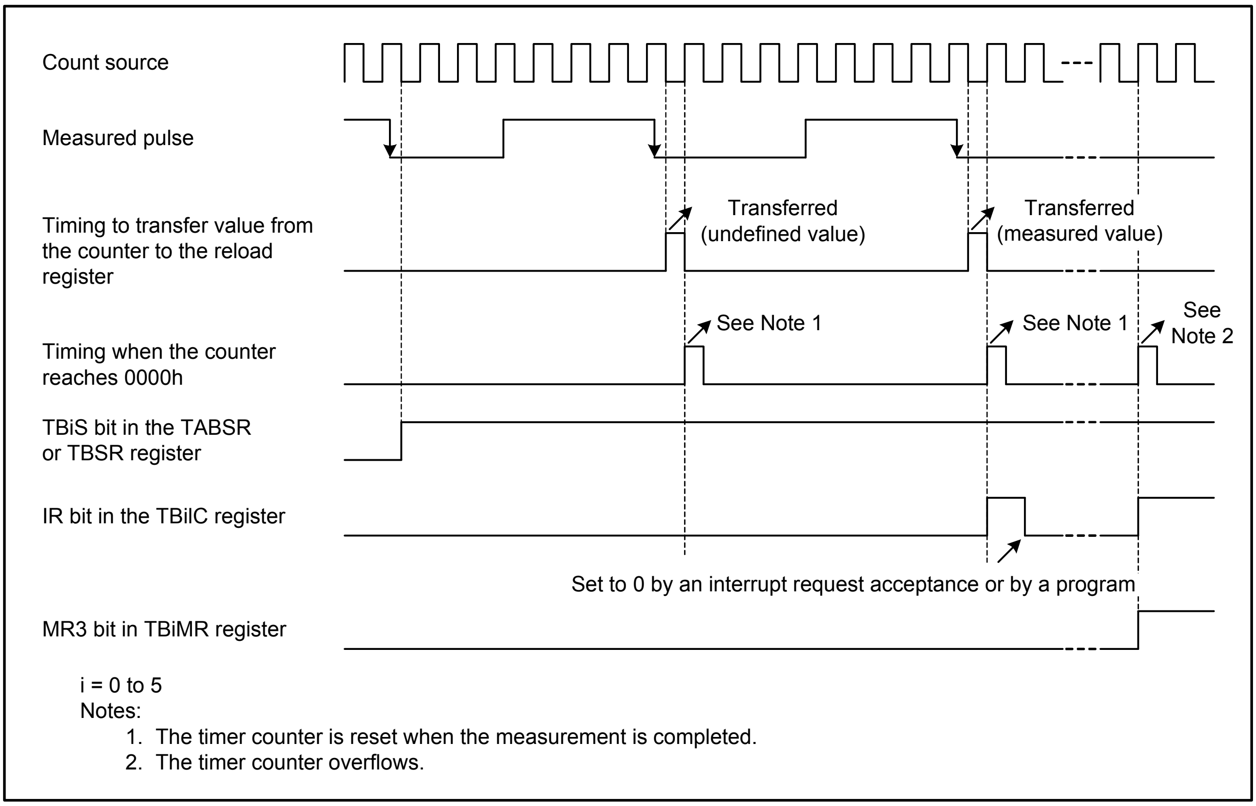


Figure 15.28 Operation Example in Pulse Period Measurement

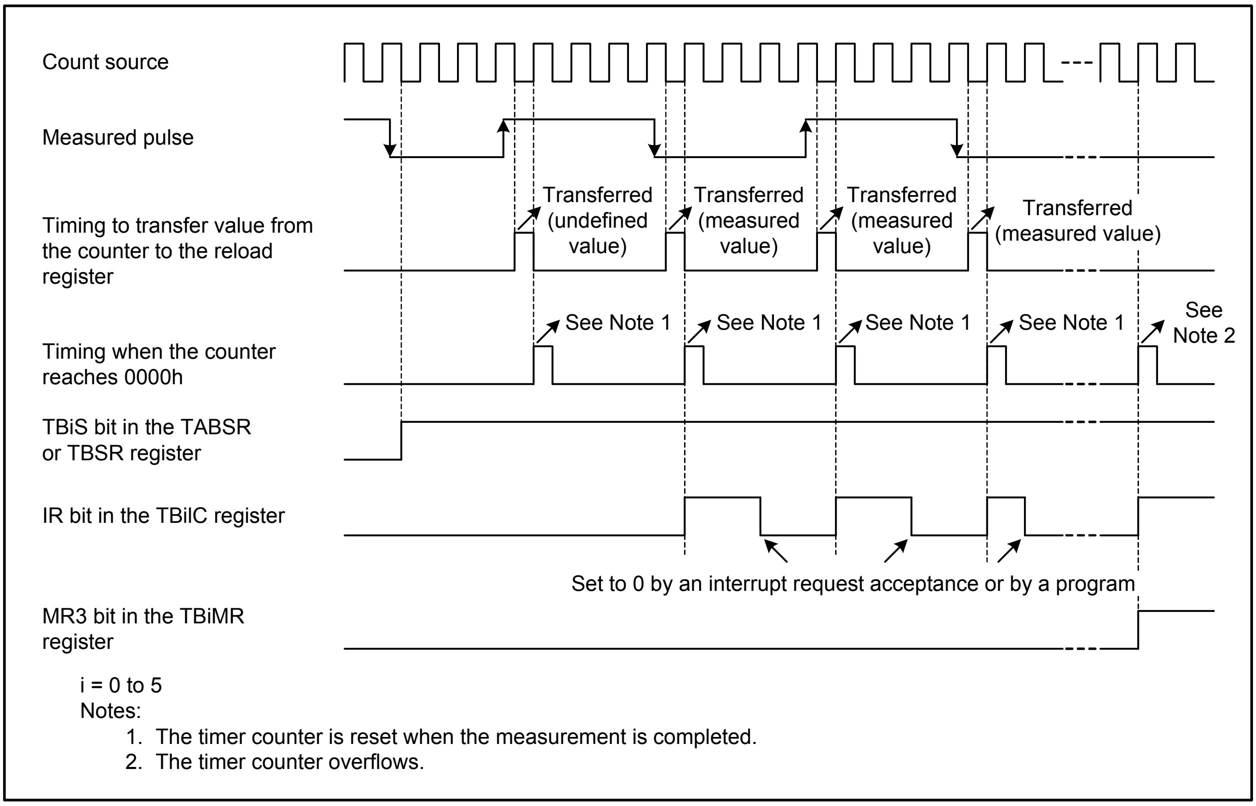


Figure 15.29 Operation Example in Pulse-width Measurement

15.3 Notes on Timers

15.3.1 Timer A and Timer B

All timers are stopped after a reset. To restart timers, configure parameters such as operating mode, count source, and counter value, then set the TAiS bit (i = 0 to 4) or TBjS bit (j = 0 to 5) in the TABSR or TBSR register to 1 (count starts).

The following registers and bits should be set while the TAiS bit or TBjS bit is 0 (count stops):

- Registers TAiMR and TBjMR
- UDF register
- Bits TAZIE, TA0TGL, and TA0TGH in the ONSF register
- TRGSR register

15.3.2 Timer A

15.3.2.1 Timer Mode

- While the timer counter is running, the TAi register indicates a counter value at any given time. However, FFFFh is read while reloading is in progress. A set value is read if the TAi register is set while the timer counter is stopped.

15.3.2.2 Event Counter Mode

- While the timer counter is running, the TAi register indicates a counter value at any given time. However, FFFFh is read if the timer counter underflows or 0000h if overflows while reloading is in progress. A set value is read if the TAi register is set while the timer counter is stopped.

15.3.2.3 One-shot Timer Mode

- If the TAiS bit in the TABSR register is set to 0 (count stops) while the timer counter is running, the following operations are performed:
  - The timer counter stops and the setting value of the TAi register is reloaded.
  - A low signal is output at the TAiOUT pin.
  - The IR bit in the TAiIC register becomes 1 (interrupts requested) after one CPU clock cycle.

- The one-shot timer is operated by an internal count source. When the trigger is an input to the TAiIN pin, the signal is output with a maximum one count source clock delay after a trigger input to the TAiIN pin.

- The IR bit becomes 1 by any of the settings below. To use the timer Ai interrupt, set the IR bit to 0 after one of the settings below is done:
  - Select one-shot timer mode after a reset.
  - Switch operating modes from timer mode to one-shot timer mode.
  - Switch operating modes from event counter mode to one-shot timer mode.

- If a retrigger occurs while counting, the timer counter decrements by one, reloads the setting value of the TAi register, and then continues counting. To generate a retrigger while counting, wait one or more count source cycles after the last trigger is generated.

- When an external trigger input is selected to start counting in timer A one-shot mode, do not provide an external retrigger for 300 ns before the timer counter reaches 0000h. Otherwise, it may stop counting.

15.3.2.4 Pulse-width Modulation Mode

- The IR bit becomes 1 by any of the settings below. To use the timer Ai interrupt (i = 0 to 4), set the IR bit to 0 after one of the settings below is done:
  - Select pulse-width modulation mode after a reset.
  - Switch operating modes from timer mode to pulse-width modulation mode.
  - Switch operating modes from event counter mode to pulse-width modulation mode.

- If the TAiS bit in the TABSR register is set to 0 (count stops) while PWM pulse is output, the following operations are performed:
  - The timer counter stops.
  - The output level at the TAiOUT pin changes from high to low. The IR bit becomes 1.
  - When a low signal is output at the TAiOUT pin, it does not change. The IR bit does not change, either.

15.3.3 Timer B

15.3.3.1 Timer Mode and Event Counter Mode

- While the timer counter is running, the TBj register (j = 0 to 5) indicates a counter value at any given time. However, FFFFh is read while reloading is in progress. When a value is set to the TBj register while the timer counter is stopped, if the TBj register is read before the count starts, the set value is read.

15.3.3.2 Pulse Period/Pulse-width Measure Mode

- While the TBjS bit in the TABSR or TBSR register is 1 (start counter), after the MR3 bit becomes 1 (overflow) and at least one count source cycle has elapsed, a write operation to the TBjMR register sets the MR3 bit to 0 (no overflow).

- Use the IR bit in the TBjIC register to detect overflow. The MR3 bit is used only to determine an interrupt request source within the interrupt handler.

- The counter value is undefined when the timer counter starts. Therefore, the timer counter may overflow before a measured pulse is applied on the initial valid edge and cause a timer Bj interrupt request to be generated.

- When the measured pulse is applied on the initial valid edge after the timer counter starts, an undefined value is transferred to the reload register. At this time, a timer Bj interrupt request is not generated.

- The IR bit may become 1 (interrupt requested) by changing bits MR1 and MR0 in the TBjMR register after the timer counter starts. However, if the same value is rewritten to bits MR1 and MR0, the IR bit does not change.

- Pulse width is continuously measured in pulse-width measure mode. Whether the measurement result is high-level width or not is determined by a program.

- When an overflow occurs at the same time a pulse is applied on the valid edge, this pulse is not recognized since an interrupt request is generated only once. Do not let an overflow occur in pulse period measure mode.

- In pulse-width measure mode, determine whether an interrupt source is a pulse applied on the valid edge or an overflow by reading the port level in the timer Bj interrupt handler.

16. Three-phase Motor Control Timers

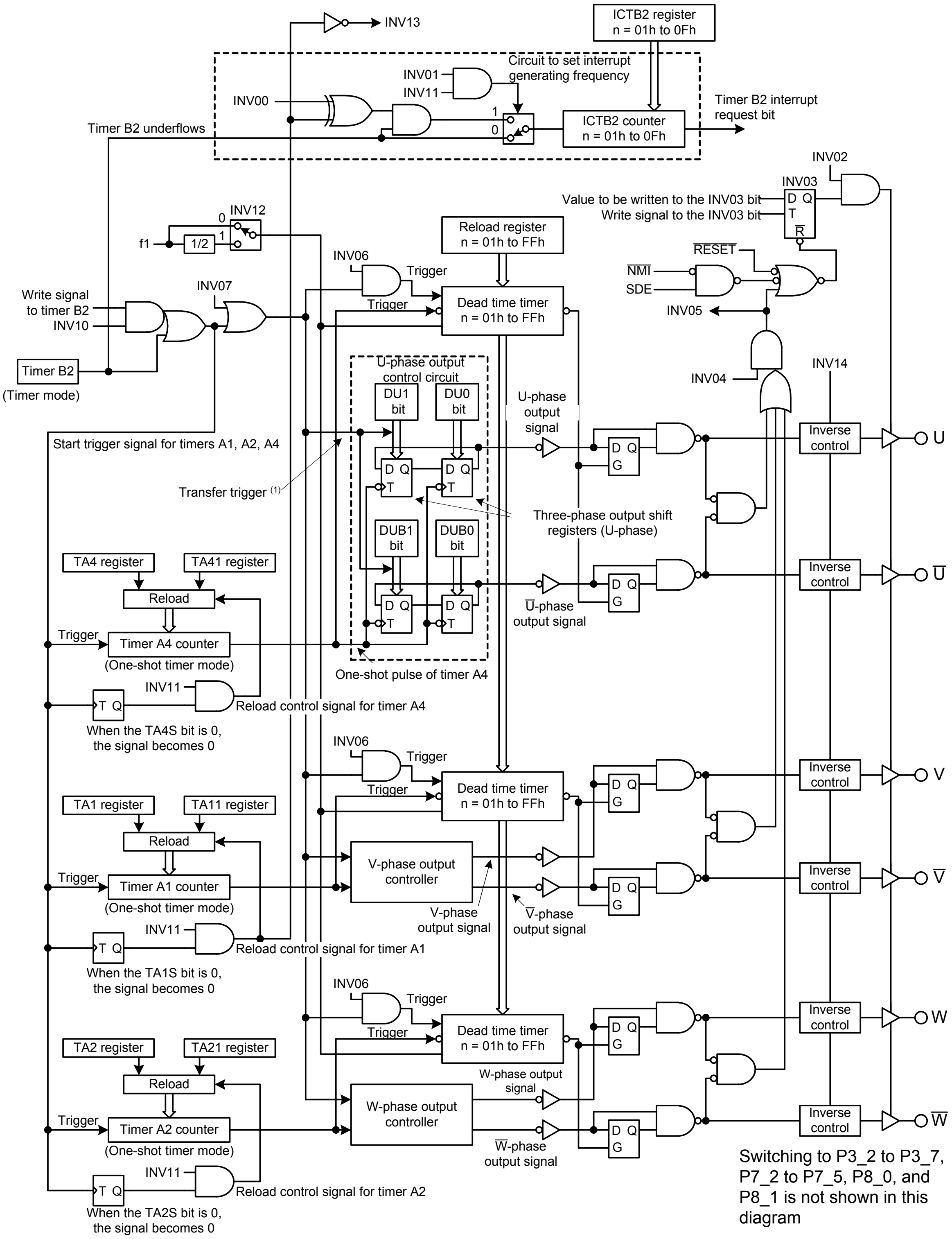
A three-phase motor driving waveform can be output using timers A1, A2, A4, and B2. The three-phase motor control timers are enabled by setting the INV02 bit in the INVC0 register to 1. Timer B2 is used for carrier wave control, and timers A1, A2, and A4 for three-phase PWM output (U, $\overline{\text{U}}$, V, $\overline{\text{V}}$, W, and $\overline{\text{W}}$) control. Table 16.1 lists the specifications of the three-phase motor control timers and Figure 16.1 shows its block diagram. Figure 16.2 to Figure 16.6 show registers associated with this function.

Table 16.1 Specifications for Three-phase Motor Control Timers

Item	Specification
Three-phase PWM waveform output pins	Six pins: U, $\overline{\text{U}}$, V, $\overline{\text{V}}$, W, and $\overline{\text{W}}$
Forced cutoff (1)	A low input to the $\overline{\text{NMI}}$ pin
Timers	Timers A4, A1, and A2 are used in one-shot timer mode: Timer A4 is used for U- and $\overline{\text{U}}$-phase waveform control Timer A1 is used for V- and $\overline{\text{V}}$-phase waveform control Timer A2 is used for W- and $\overline{\text{W}}$-phase waveform control Timer B2 is used in timer mode Carrier wave cycle control Dead time timer (three 8-bit timers share a reload register): Dead time control
Output waveforms	Triangular wave modulation and sawtooth wave modulation • Output of a high or a low waveform for one cycle • Separately settable levels of high side and low side
Carrier wave cycles	Triangular wave modulation: count source × (m+1) × 2 Sawtooth wave modulation: count source × (m+1) m: TB2 register setting value from 0000h to FFFFh Count source: f1, f8, f2n, or fC32
Three-phase PWM output width	Triangular wave modulation: count source × n × 2 Sawtooth wave modulation: count source × n n: Setting value of registers TA4, TA1, and TA2 (registers TA4, TA41, TA1, TA11, TA2, and TA21 when the INV11 bit in the INVC1 register is set to 1) from 0001h to FFFFh Count source: f1, f8, f2n, or fC32
Dead time (width)	Count source × p or no dead time p: DTT register setting value from 01h to FFh Count source: f1 or f1 divided by 2
Active level	Selectable either active high or active low
Simultaneous conduction prevention	Function to detect simultaneous turn-on signal outputs, function to disable signal output when simultaneous turn-on signal outputs are detected
Interrupt frequency	Selectable from one through 15 time-carrier wave cycle-to-cycle basis for the timer B2 interrupt

Note:

1. Forced cutoff by a signal input to the $\overline{\text{NMI}}$ pin can be performed when the PM24 bit in the PM2 register is set to 1 (NMI enabled), the SDE bit in the IOBC register is set to 1 (shutdown enabled), the INV02 bit in the INVC0 register is set to 1 (three-phase motor control timers used), and the INV03 bit is set to 1 (three-phase motor control timer output enabled).



Switching to P3_2 to P3_7, P7_2 to P7_5, P8_0, and P8_1 is not shown in this diagram

INV00 to INV07: Bits in the INVC0 register
INV10 to INV15: Bits in the INVC1 register
DUi and DUBi: Bits in the IDBi register (i = 0, 1)

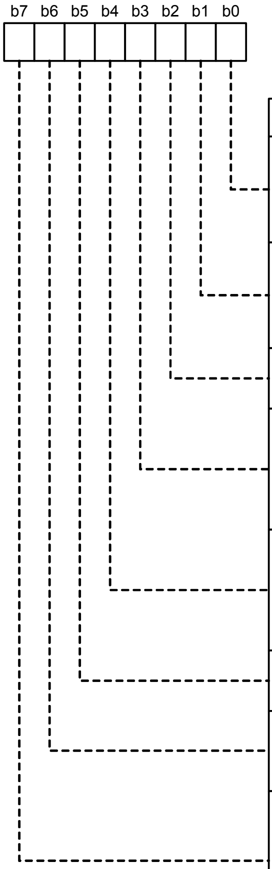
TA1S to TA4S: Bits in the TABSR register
SDE: Bit in the IOBC register

Note:

1. When the INV06 bit is 0 (triangular wave modulation mode), the transfer trigger is generated only when the initial underflow of timer B2 occurs after setting registers IDB0 and IDB1.

Figure 16.1 Block Diagram for Three-phase Motor Control Timers

Three-phase PWM Control Register 0 (1)



Symbol	Address	Reset Value
INVC0	0308h	0000 0000b

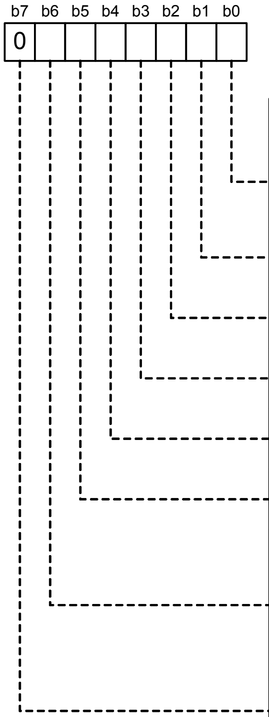
Bit Symbol	Bit Name	Function	RW
INV00	ICTB2 Count Condition Select Bit (2)	b1 b0 0 X : The underflow of timer B2 1 0 : The underflow of timer B2 when the reload control signal for timer A1 is set to 0 (3) 1 1 : The underflow of timer B2 when the reload control signal for timer A1 is set to 1 (3, 4)	RW
INV01			RW
INV02	Three-phase Motor Control Timers Select Bit	0: Do not use this function 1: Use this function (5, 6, 7)	RW
INV03	Three-phase Motor Control Timer Output Control Bit	0: Disables the three-phase motor control timer output (7) 1: Enables the three-phase motor control timer output (8)	RW
INV04	Simultaneous Conduction Prevention Bit	0: Ignores simultaneous turn-on signal output 1: Disables simultaneous turn-on signal output	RW
INV05	Simultaneous Conduction Detection Flag	0: Not detected 1: Detected (9)	RO
INV06	Modulation Mode Select Bit	0: Triangular wave modulation mode 1: Sawtooth wave modulation mode (10)	RW
INV07	Software Trigger Select Bit	A transfer trigger is generated when this bit is set to 1. When the INV06 bit is 1, another trigger to the dead time timer is also generated. This bit is read as 0	RW

Notes:

1. This register should be set after the PRC1 bit in the PRCR register is set to 1 (write enabled). Bits INV00 to INV02 and INV06 should be rewritten while timers A1, A2, A4, and B2 are stopped.
2. This bit is enabled when the INV11 bit in the INVC1 register is 1 (three-phase mode 1). When the INV11 bit is 0 (three-phase mode 0), the ICTB2 counter increments by one each time timer B2 underflows irrespective of the INV00 and INV01 bit settings.
3. Set the ICTB2 register before setting the INV01 bit to 1. The timer A1 count start flag should be set to 1 before the initial timer B2 underflow occurs.
4. When the INV00 bit is set to 1, the first interrupt is generated if timer B2 underflows n-1 times (n is the value set in the ICTB2 counter). Subsequent interrupts are generated every n times timer B2 underflows.
5. The INV02 bit should be set to 1 to operate the dead time timer, U-, V-, and W-phase output control circuits, and the ICTB2 counter.
6. After setting the INV02 bit to 1, pins should be configured first by the IOBC register then by the output function select register.
7. When the INV02 bit is set to 1 and the INV03 bit is set to 0, pins U, $\overline{\text{U}}$, V, $\overline{\text{V}}$, W, and $\overline{\text{W}}$, even when they are assigned to other peripheral functions, become high-impedance.
8. The INV03 bit becomes 0 when any of the following occurs:
   - Reset
   - Signals of both the high and low sides are simultaneously switched to active when the INV04 bit is set to 1.
   - The INV03 bit is set to 0 by a program.
   - The $\overline{\text{NMI}}$ pin goes from high to low when the PM24 bit in the PM2 register is set to 1 (NMI enabled) and the SDE bit in the IOBC register is set to 1 (shutdown enabled).
9. This bit cannot be set to 1 by a program. The INV04 bit should be set to 0 to set this bit to 0.
10. When the INV06 bit is set to 1, the INV11 bit in the INVC1 register should be set to 0 (three-phase mode 0). In this case, the PWCON bit in the TB2SC register should be set to 0 (timer B2 register reloaded when timer B2 underflows).

Figure 16.2 INVC0 Register

Three-phase PWM Control Register 1 (1)



Symbol	Address	Reset Value
INVC1	0309h	0000 0000b

Bit Symbol	Bit Name	Function	RW
INV10	Timers A1, A2, and A4 Start Trigger Select Bit	0: The underflow of timer B2 1: The underflow of timer B2 and a write operation to the TB2 register	RW
INV11	Timers A1-1, A2-1, and A4-1 Control Bit	0: Three-phase mode 0 (2, 3) 1: Three-phase mode 1	RW
INV12	Dead Time Timer Count Source Select Bit	0: f1 1: f1 divided-by-2	RW
INV13	Carrier Wave Detection Flag (4)	0: Timer A1 reload control signal is 0 1: Timer A1 reload control signal is 1	RO
INV14	Active Level Control Bit	0: Active low output 1: Active high output	RW
INV15	Dead Time Disable Bit	0: Enables dead time 1: Disables dead time	RW
INV16	Dead Time Timer Trigger Select Bit	0: Falling edge of a one-shot pulse of timer (A4, A1, and A2) (5) 1: Rising edge of the three-phase output shift register (phases U, V, and W)	RW
— (b7)	Reserved	Should be written with 0	RW

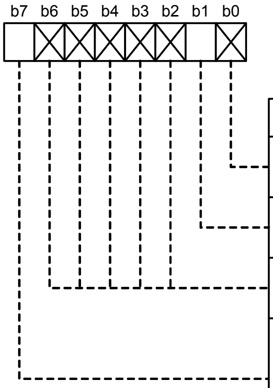
Notes:

1. This register should be set after the PRC1 bit in the PRCR register is set to 1 (write enabled). This register should be rewritten while timers A1, A2, A4, and B2 are stopped.
2. When the INV06 bit in the INVC0 register is 1 (sawtooth wave modulation mode), the INV11 bit should be set to 0 (three-phase mode 0).
3. When the INV11 bit is set to 0, the PWCON bit in the TB2SC register should be set to 0 (timer B2 register reloaded if timer B2 underflows).
4. This bit setting is enabled when the INV06 bit is 0 (triangular wave modulation mode) and the INV11 bit is 1 (three-phase mode 1).
5. If the following conditions are all met, the INV16 bit should be set to 1:
   - The INV15 bit is 0 (dead time enabled).
   - The Dij bit (i = U, V, or W; j = 0 to 1) has a different value from the DiBj bit whenever the INV03 bit is set to 1 (three-phase motor control timer output enabled); the high- and low-side output signals always have inverse levels on periods other than dead time.

   If any of the conditions above are not met, the INV16 bit should be set to 0.

Figure 16.3 INVC1 Register

Three-phase Output Buffer Control Register (1)



Symbol	Address	Reset Value
IOBC	40097h	0XXX XX0Xb

Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit; should be written with 0 and read as undefined value		—
SDE	Three-phase Output Pin Shutdown Enable Bit	0: Disable shutdown 1: Enable shutdown	RW
— (b6-b2)	No register bits; should be written with 0 and read as undefined value		—
TBSOUT	Three-phase Output Pin Select Bit (2)	0: Use pins U, $\overline{U}$, V, $\overline{V}$, W, and $\overline{W}$ of ports P7 and P8 1: Use pins U, $\overline{U}$, V, $\overline{V}$, W, and $\overline{W}$ of port P3	RW

Notes:

1. This register should be set after the PRC1 bit in the PRCR register is set to 1 (write enabled).
2. Set this bit after the INV02 bit in the INVC0 register is set to 1. Then, set the output function select register of corresponding port. When the INV03 bit in the INVC0 register is set to 0, output pins for the three-phase motor control timers become high-impedance by the output enable control of output buffers. However, the output enable cannot be controlled only by the output function select register when more than two ports are assigned for output. Thus, a three-state output buffer should be selected using the TBSOUT bit.

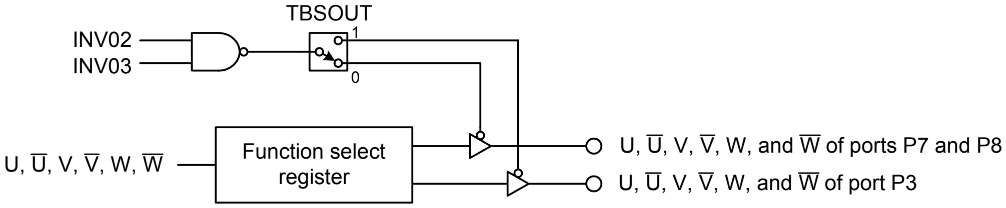
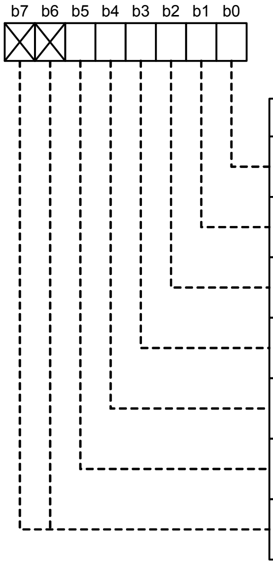


Figure 16.4 IOBC Register

Three-phase Output Buffer Register i (i = 0, 1) (1)



Symbol	Address	Reset Value
IDB0, IDB1	030Ah, 030Bh	XX11 1111b

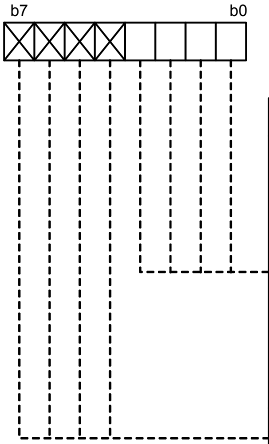
Bit Symbol	Bit Name	Function	RW
DUi	U-phase Output Buffer i	These bits should be written with an output level of the three-phase output shift register. The written value is reflected in each turn-on signal as follows: 0: Active (ON) 1: Inactive (OFF) The bits are read as the value of the three-phase output shift register	RW
DUBi	$\overline{\text{U}}$-phase Output Buffer i		RW
DVi	V-phase Output Buffer i		RW
DVBi	$\overline{\text{V}}$-phase Output Buffer i		RW
DWi	W-phase Output Buffer i		RW
DWBi	$\overline{\text{W}}$-phase Output Buffer i		RW
— (b7-b6)	No register bits; should be written with 0 and read as undefined value		—

Note:

1. Values of registers IDB0 and IDB1 are transferred to the three-phase output shift register by a transfer trigger. The initial output signal level of each phase is determined by the value written in the IDB0 register after the transfer trigger occurs. Then the output signal level is determined by the value written in the IDB1 register on the falling edge of a one-shot pulse of timers A1, A2, and A4.

Figure 16.5 Registers IDB0 and IDB1

Timer B2 Interrupt Generating Frequency Set Counter (1, 2, 3)



Symbol	Address	Reset Value
ICTB2	030Dh	Undefined

Function	Setting Range	RW
- When the INV01 bit is 0 (ICTB2 counter increments each time timer B2 underflows), a timer B2 interrupt request is generated every nth times timer B2 underflows - When the INV01 bit is 1 (ICTB2 counter increments when the timer A1 reload control signal is set to 0 or 1 and timer B2 underflows), a timer B2 interrupt request is generated every nth times timer B2 underflows when the timer A1 reload control signal is 0 or 1 (n = setting value)	01h to 0Fh	WO
No register bits; should be written with 0		—

Notes:
1. The MOV instruction should be used to set the ICTB2 register.
2. When the INV01 bit in the INVC0 register is 1, set this register while the TB2S bit in the TABSR register is 0 (timer B2 count stops). Although it can be set even when the TB2S bit is 1 (timer B2 count starts) when the INV01 bit is 0, do not set this register when timer B2 underflows.
3. When the INV00 bit in the INVC0 register is set to 1, the first interrupt is generated when timer B2 underflows n-1 times. Subsequent interrupts are generated every n times timer B2 underflows.
(n = setting value of the ICTB2 counter)

Figure 16.6 ICTB2 Register

16.1 Modulation Modes of Three-phase Motor Control Timers

The three-phase motor control timers support two modulation modes: triangular wave modulation mode and sawtooth wave modulation mode. The triangular wave modulation mode has two modes: three-phase mode 0 and three-phase mode 1. Table 16.2 lists bit settings and characteristics of each mode.

Table 16.2 Modulation Modes

Item	Triangular Wave Modulation Mode		Sawtooth Wave Modulation Mode
	Three-phase mode 0	Three-phase mode 1	(Three-phase mode 0)
Bit settings	INV06 is 0, INV11 is 0, PWCON is 0	INV06 is 0, INV11 is 1	INV06 is 1, INV11 is 0, PWCON is 0
Waveform	Triangular wave		Sawtooth wave
Registers TA11, TA21, and TA41	Not used	Used	Not used
Timing to transfer data from registers IDB0 and IDB1 to the three-phase output shift register	Only once when a transfer trigger (1) occurs after setting registers IDB0 and IDB1		Whenever a transfer trigger (1) occurs
Timing to trigger the dead time timer when the INV16 bit is 0	On the falling edge of a one-shot pulse of timers A1, A2, and A4		When a transfer trigger occurs, or on the falling edge of a one-shot pulse of timers A1, A2, and A4
Bits INV00 and INV01 in the INVC0 register	Disabled. The ICTB2 counter increments each time timer B2 underflows, irrespective of the INV00 and INV01 bit settings	Enabled	Disabled. The ICTB2 counter increments each time timer B2 underflows, irrespective of the INV00 and INV01 bit settings
INV13 bit	Disabled	Enabled	Disabled

Note:

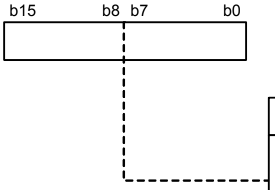
1. The transfer trigger is a timer B2 underflow, a write operation to the INV07 bit, or a write operation to the TB2 register when the INV10 bit is 1.

16.2 Timer B2

Timer B2, which operates in timer mode, is used for carrier wave control in the three-phase motor control timers.

Figure 16.7 and Figure 16.8 show registers TB2 and TB2MR in this function, respectively. Figure 16.9 shows the TB2SC register which switches timing to change the carrier wave frequency in three-phase mode 1.

Timer B2 Register (1)



Symbol	Address	Reset Value
TB2	0355h-0354h	Undefined

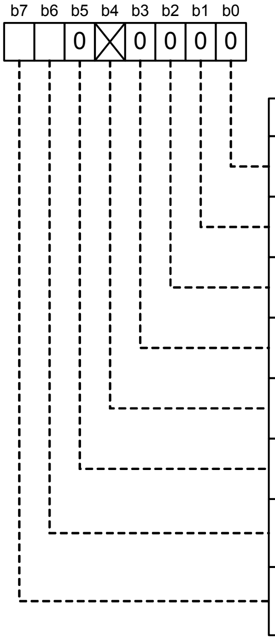
Function	Setting Range	RW
Divides the count source by n+1. Starts timers A1, A2, and A4 each time an underflow occurs (n = setting value)	0000h to FFFFh	RW

Note:

1. A 16-bit read/write access to this register should be performed.

Figure 16.7 TB2 Register When Using Three-phase Motor Control Timers

Timer B2 Mode Register

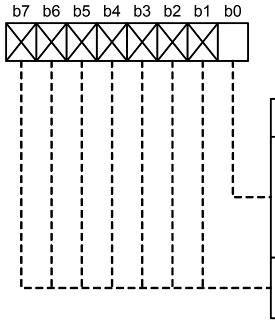


Symbol	Address	Reset Value
TB2MR	035Dh	00XX 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	Should be written with 00b (timer mode) when using the three-phase motor control timers	RW
TMOD1			RW
MR0	Disabled when using the three-phase motor control timers. Should be written with 0 and read as undefined value		RW
MR1			RW
MR2	No register bit; should be written with 0 and read as undefined value		—
MR3	Disabled when using the three-phase motor control timers. Should be written with 0 and read as undefined value		—
TCK0	Count Source Select Bit	b7 b6 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : fC32	RW
TCK1			RW

Figure 16.8 TB2MR Register When Using Three-phase Motor Control Timers

Timer B2 Special Mode Register



Symbol	Address	Reset Value
TB2SC	035Eh	XXXX XXX0b

Bit Symbol	Bit Name	Function	RW
PWCON	Timer B2 Reload Timing Switching Bit [1]	0: The underflow of timer B2 1: The underflow of timer B2 when the reload control signal for timer A1 is 0	RW
— (b7-b1)	No register bits; should be written with 0 and read as undefined value		—

Note:

1. When the INV11 bit is 0 (three-phase mode 0) or the INV06 bit is 1 (sawtooth wave modulation mode), this bit should be set to 0.

Figure 16.9 TB2SC Register

16.3 Timers A4, A1, and A2

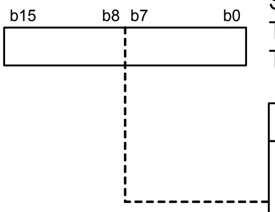
Timers A4, A1, and A2 are used for three-phase PWM output (U, $\overline{U}$, V, $\overline{V}$, W, and $\overline{W}$) control when using the three-phase motor control timers.

These timers should be operated in one-shot timer mode. Every time timer B2 underflows, a trigger is input to timers A4, A1, and A2 to generate a one-shot pulse. If the values of registers TA4, TA1, and TA2 are rewritten every time a timer B2 interrupt is generated, the duty ratio of the PWM waveform can be varied.

In three-phase mode 1, the value of registers TAi and TAi-1 (i = 4, 1, 2) is alternately reloaded to the counter at each timer B2 interrupt, which halves the timer B2 interrupt frequency.

Figure 16.10 shows registers TA1, TA2, TA4, TA11, TA21, and TA41 in the three-phase motor control timers. Figure 16.11 shows registers TA1M, TA2M, TA11M, TA21M, and TA41M in this function. Figure 16.12 shows registers TA1MR, TA2MR, and TA4MR in this function. Figure 16.13 and Figure 16.14 show registers TRGSR and TABSR, respectively, in this function.

Timer Ai/Timer Ai-1 Registers (i = 1, 2, 4) (1 to 6)



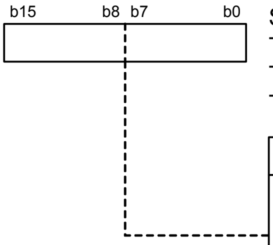
Symbol	Address	Reset Value
TA1, TA2, TA4	0349h-0348h, 034Bh-034Ah, 034Fh-034Eh	Undefined
TA11, TA21, TA41	0303h-0302h, 0305h-0304h, 0307h-0306h	Undefined

Function	Setting Range	RW
The timer stops when the nth count source is counted after a start trigger is generated. The output signal for each phase is switched when timers A1, A2, and A4 stop (n = setting value)	0000h to FFFFh	WO

Notes:
1. A 16-bit write access to these registers should be performed.
2. When these registers are set to 0000h, the counter does not start, and no timer Ai interrupt request is generated.
3. The MOV instruction should be used to set these registers.
4. When the INV15 bit in the INVC1 register is 0 (enables dead time), the turn-on output signal is switched to its active state with a delay. It switches when the dead time timer stops.
5. When the INV11 bit in the INVC1 register is 0 (three-phase mode 0), the value of the TAi register is transferred to the reload register by a timer Ai start trigger. When the INV11 bit is 1 (three-phase mode 1), first the value of the TAi1 register is transferred to the reload register by a timer Ai start trigger. Then the value of the TAi register is transferred by the next timer Ai start trigger. After that, the values of registers TAi1 and TAi are alternately transferred to the reload register.
6. These registers should not be written when timer B2 underflows.

Figure 16.10 Registers TA1, TA2, TA4, TA11, TA21, and TA41

Timer Ai/Timer Ai-1 Mirror Registers (i = 1, 2, 4) (1 to 7)

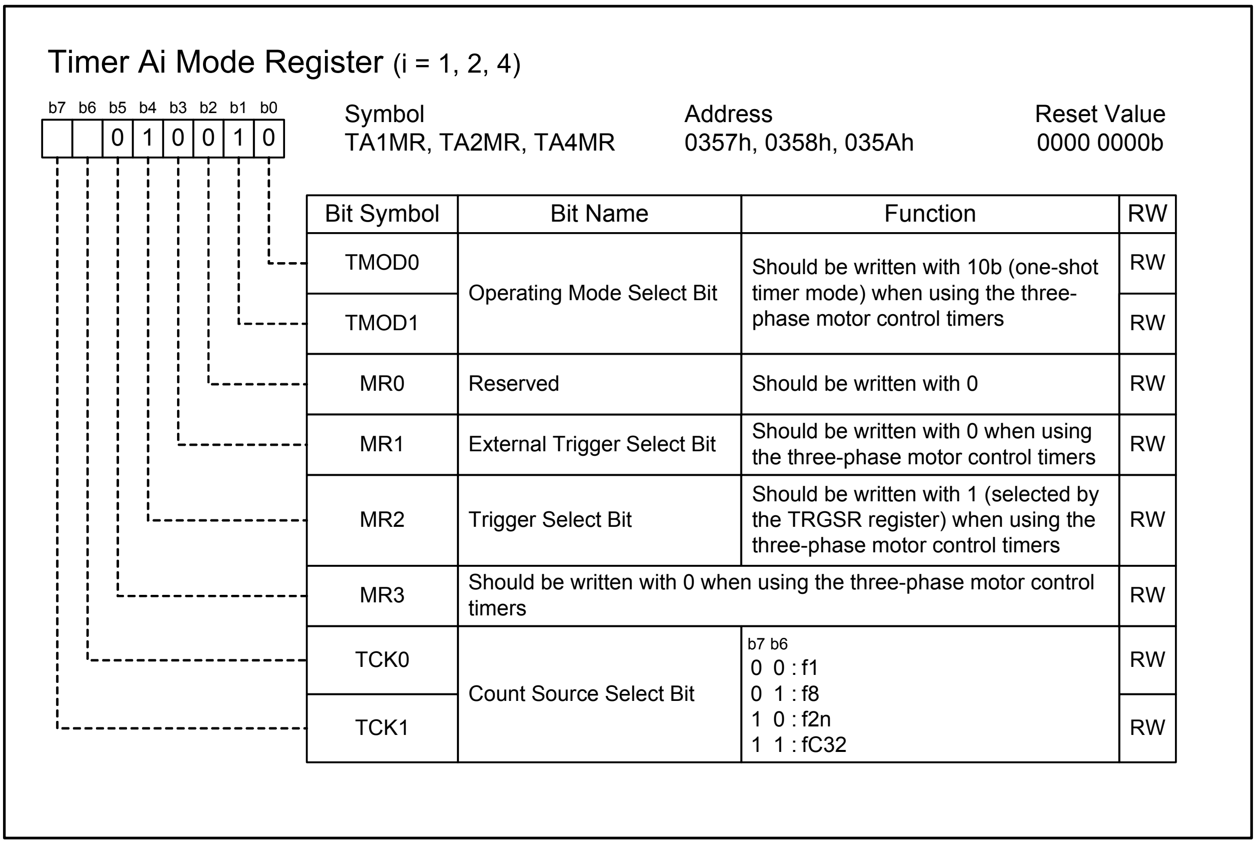


Symbol	Address	Reset Value
TA1M, TA11M	0221h-0220h, 0223h-0222h	Undefined
TA2M, TA21M	0225h-0224h, 0227h-0226h	Undefined
TA4M, TA41M	0229h-0228h, 022Bh-022Ah	Undefined

Function	Setting Range	RW
The timer stops when the nth count source is counted after a start trigger is generated. The output signal for each phase is switched when timers A1, A2, and A4 stop (n = setting value)	0000h to FFFFh	WO

Notes:
1. A 16-bit write access to these registers should be performed.
2. The values written in these registers are reflected to registers TAj and TAj1.
3. When these registers are set to 0000h, the counter does not start, nor the timer Ai interrupt request is generated.
4. The MOV instruction should be used to set these registers.
5. When the INV15 bit in the INVC1 register is 0 (enables dead time), the turn-on output signal is switched to its active state with a delay; it switches when the dead time timer stops.
6. When the INV11 bit in the INVC1 register is 0 (three-phase mode 0), the value of the TAi register is transferred to the reload register by a timer Ai start trigger. When the INV11 bit is 1 (three-phase mode 1), first the value of the TAi1 register is transferred to the reload register by a timer Ai start trigger. Then the value of the TAi register is transferred by the next timer Ai start trigger. After that, the values of these registers TAi1 and TAi are alternately transferred to the reload register.
7. These registers should not be written when timer B2 underflows.

Figure 16.11 Registers TA1M, TA2M, TA4M, TA11M, TA21M, and TA41M



Timer Ai Mode Register (i = 1, 2, 4)

Symbol	Address	Reset Value
TA1MR, TA2MR, TA4MR	0357h, 0358h, 035Ah	0000 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	Should be written with 10b (one-shot timer mode) when using the three-phase motor control timers	RW
TMOD1			RW
MR0	Reserved	Should be written with 0	RW
MR1	External Trigger Select Bit	Should be written with 0 when using the three-phase motor control timers	RW
MR2	Trigger Select Bit	Should be written with 1 (selected by the TRGSR register) when using the three-phase motor control timers	RW
MR3	Should be written with 0 when using the three-phase motor control timers		RW
TCK0	Count Source Select Bit	b7 b6 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : fC32	RW
TCK1			RW

Figure 16.12 Registers TA1MR, TA2MR, and TA4MR When Using Three-phase Motor Control Timers

Trigger Select Register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol TRGSR

Address 0343h

Reset Value 0000 0000b

Bit Symbol	Bit Name	Function	RW
TA1TGL	Timer A1 Event/Trigger Select Bit	Should be set to 01b (the underflow of TB2) to use the V-phase output control circuit	RW
TA1TGH			RW
TA2TGL	Timer A2 Event/Trigger Select Bit	Should be set to 01b (the underflow of TB2) to the use W-phase output control circuit	RW
TA2TGH			RW
TA3TGL	Timer A3 Event/Trigger Select Bit	b5 b4 0 0 : Select the input to the TA3IN pin 0 1 : Select the overflow of TB2 (1) 1 0 : Select the overflow of TA2 (1) 1 1 : Select the overflow of TA4 (1)	RW
TA3TGH			RW
TA4TGL	Timer A4 Event/Trigger Select Bit	Should be set to 01b (the underflow of TB2) to the use U-phase output control circuit	RW
TA4TGH			RW

Note:

1. The timer overflows or underflows.

Figure 16.13 TRGSR Register in Three-phase Motor Control Timers

Count Start Register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol TABSR

Address 0340h

Reset Value 0000 0000b

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 Count Start Bit	0: Stop counter 1: Start counter	RW
TA1S	Timer A1 Count Start Bit	0: Stop counter 1: Start counter	RW
TA2S	Timer A2 Count Start Bit	0: Stop counter 1: Start counter	RW
TA3S	Timer A3 Count Start Bit	0: Stop counter 1: Start counter	RW
TA4S	Timer A4 Count Start Bit	0: Stop counter 1: Start counter	RW
TB0S	Timer B0 Count Start Bit	0: Stop counter 1: Start counter	RW
TB1S	Timer B1 Count Start Bit	0: Stop counter 1: Start counter	RW
TB2S	Timer B2 Count Start Bit	0: Stop counter 1: Start counter	RW

Figure 16.14 TABSR Register

16.4 Simultaneous Conduction Prevention and Dead Time Timer

The three-phase motor control timers offer two ways to avoid shoot-through, which occurs when high-side and low-side transistors are simultaneously turned on.

One is “simultaneous turn-on signal output disable function”. This function prevents high-side and low-side transistors from being inadvertently switched to active due to events like program errors. The other is by the use of dead time timers. A dead time timer delays the turn-on of one transistor in order to ensure that an adequate time (the dead time) passes after the other is turned off.

To disable simultaneous turn-on output signals, the INV04 bit in the INVC0 register should be set to 1. If outputs for any pair of phases (U and $\overline{U}$, V and $\overline{V}$, or W and $\overline{W}$) are simultaneously switched to an active state, every three-phase motor control output pin becomes high-impedance. Figure 16.15 shows an example of output waveform when simultaneous turn-on signal output is disabled.

To enable the dead time timer, the INV15 bit in the INVC1 register should be set to 0. The DTT register determines the dead time. Figure 16.16 shows the DTT register and Figure 16.17 shows an example of output waveform on using dead time timer.

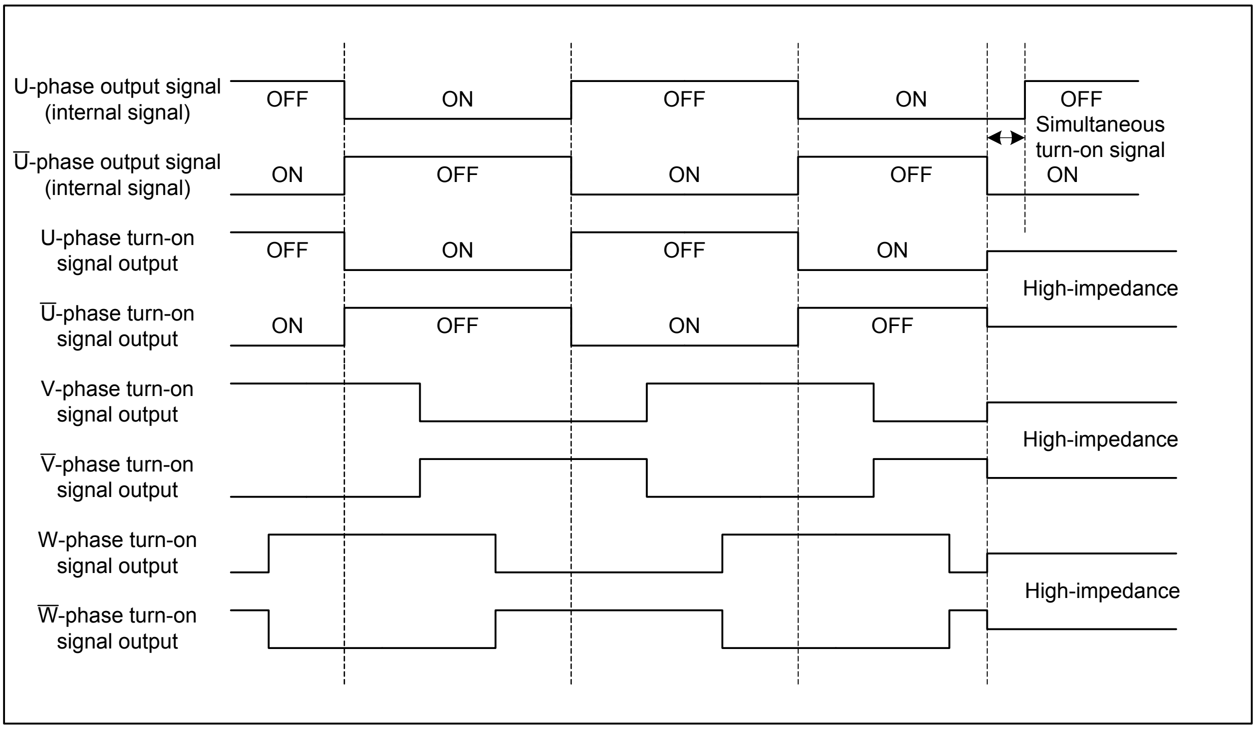


Figure 16.15 Output Waveform When Simultaneous Turn-on Signal Output is Disabled

Dead Time Timer (1, 2)

b7 b0

Symbol	Address	Reset Value
DTT	030Ch	Undefined

Function	Setting Range	RW
The dead time timer is a one-shot timer that delays the timing for a turn-on signal to be switched to its active state preventing a simultaneous conduction of high-side and low-side transistors. The timer stops when counting a count source n times after a start trigger occurs (n = setting value) (3)	01h to FFh	WO

Notes:

1. The MOV instruction should be used to set this register.
2. This register setting is enabled when the INV15 bit in the INVC1 register is set to 0 (enables dead time). No dead time can be set when the INV15 bit is set to 1 (disables dead time).
3. The trigger and count source should be selected using bits INV16 and INV12 in the INVC1 register, respectively.

Figure 16.16 DTT Register

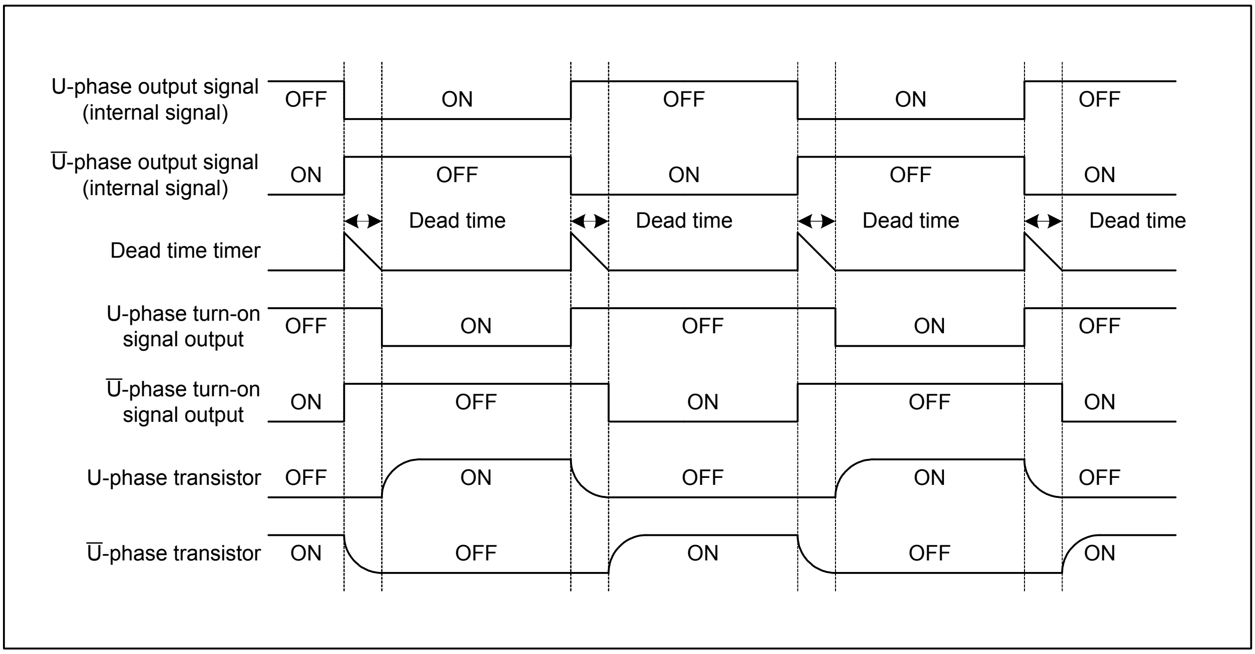
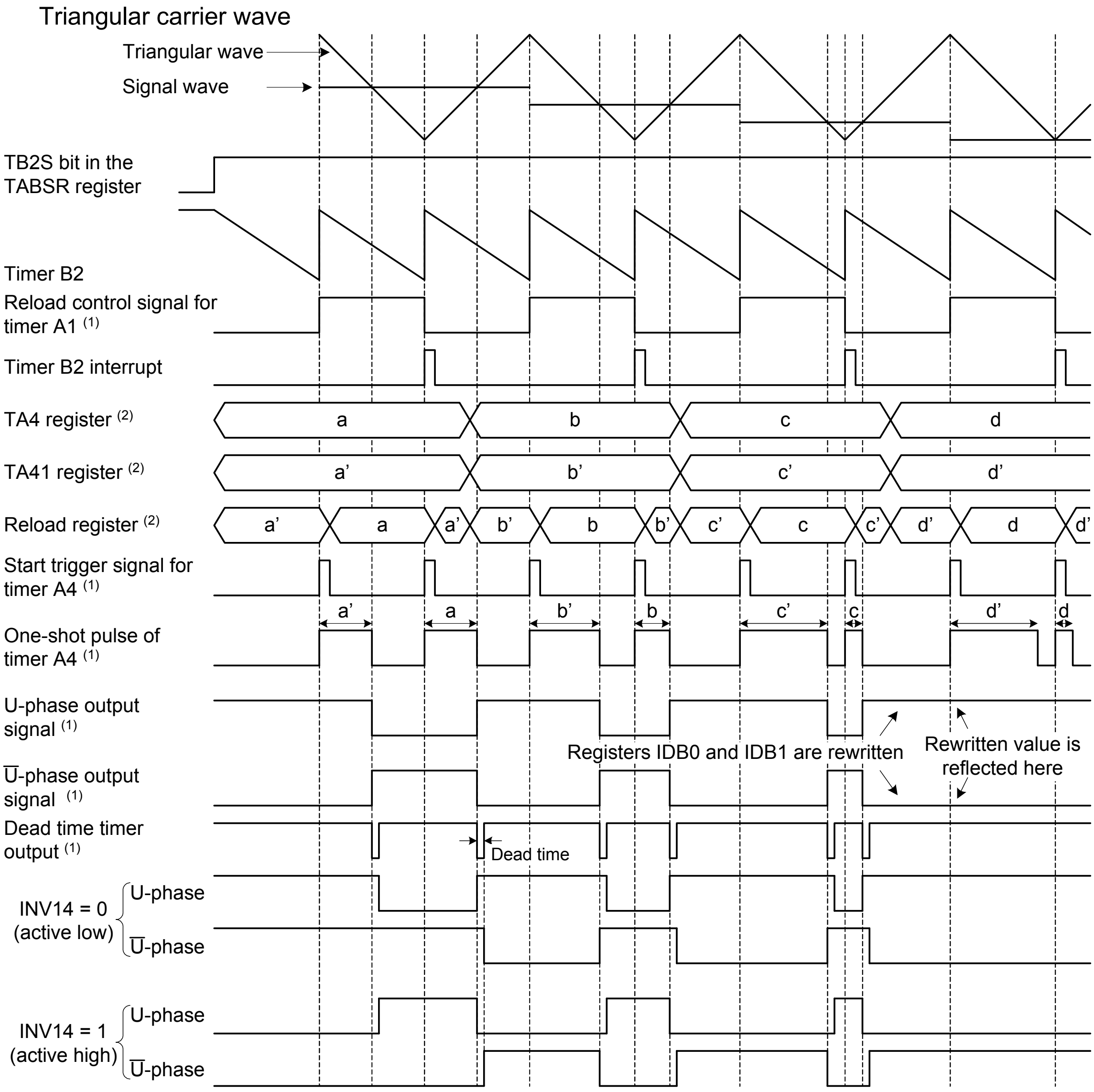


Figure 16.17 Output Waveform When Using Dead Time Timer

16.5 Three-phase Motor Control Timer Operation

Figure 16.18 and Figure 16.19 show an operation example of triangular wave modulation and sawtooth wave modulation, respectively.



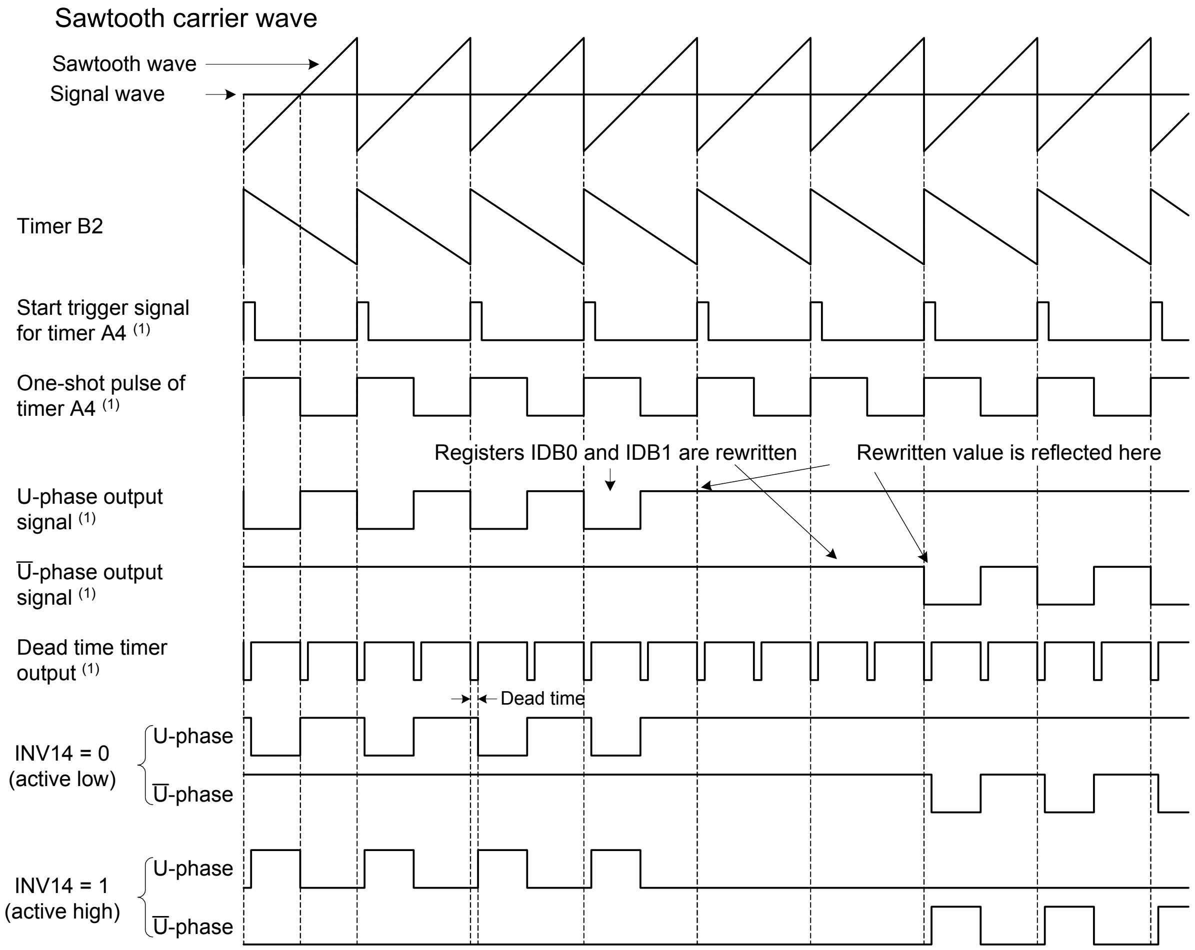
The above assumes when INVC0 = 00XX11XXb (X varies depending on each system) and INVC1 = 010XXXX0b.
PWM output may vary as follows:

Case 1: When INV11 = 1 (three-phase mode 1)
- INV01 = 0 and ICTB2 = 2h (timer B2 interrupt generated every second time timer B2 underflows), or INV01 = 1, INV00 = 1, and ICTB2 = 1h (timer B2 interrupt generated every time timer B2 underflows when the reload control signal for timer A1 is set to 1)
- The setting of registers TA4 and TA41 are varied every time a timer B2 interrupt is generated,
  Default value: TA41 = a', TA4 = a
  On the first timer B2 interrupt: TA41 = b', TA4 = b; the second time: TA41 = c', TA4 = c
- Default value of registers IDB0 and IDB1: DU0 = 1, DUB0 = 0, DU1 = 0, DUB1 = 1
  On the third time: DU0 = 1, DUB0 = 0, DU1 = 1, DUB1 = 0

Case 2: When INV11 = 0 (three-phase mode 0)
- INV01 = 0 and ICTB2 = 1h (timer B2 interrupt generated every time timer B2 underflows)
- TA4 register setting is varied every time a timer B2 interrupt is generated,
  Default value: TA4 = a'
  On the first timer B2 interrupt: TA4 = a; 2nd time: TA4 = b'; 3rd time: TA4 = b; 4th time: TA4 = c'; 5th time: TA4 = c
- Default value of registers IDB0 and IDB1: DU0 = 1, DUB0 = 0, DU1 = 0, DUB1 = 1
  On the sixth time: DU0 = 1, DUB0 = 0, DU1 = 1, DUB1 = 0

Notes:
1. Internal signal. Refer to the block diagram of three-phase motor control timers.
2. Applicable when the INV11 bit in the INV1 register is set to 1 (three-phase mode 1).

Figure 16.18 Triangular Wave Modulation Operation



The above assumes when INVC0 = 01XX110Xb (X varies depending on each system) and INVC1 = 000XXX00b.
This bit setting is applicable to turn-on control with a phase shift of 120 degrees.

PWM output may vary as follows:
Default value of registers IDB0 and IDB1: DU0 = 0, DUB0 = 1, DU1 = 1, DUB1 = 1
On the third timer B2 interrupt: DU0 = 1, DUB0 = 1, DU1 = 1, DUB1 = 1
On the fifth timer B2 interrupt: DU0 = 1, DUB0 = 0, DU1 = 1, DUB1 = 1

Note:

1. Internal signal. Refer to the block diagram of three-phase motor control timers.

Figure 16.19 Sawtooth Wave Modulation Operation

16.6 Notes on Three-phase Motor Control Timers

16.6.1 Shutdown

- When a low signal is applied to the $\overline{\text{NMI}}$ pin with the following bit settings, pins TA1OUT, TA2OUT, and TA4OUT become high-impedance: the PM24 bit in the PM2 register is 1 (NMI enabled), the SDE bit in the IOBC register is 1 (shutdown enabled), the INV02 bit in the INVC0 register is 1 (three-phase motor control timers used), and the INV03 bit is 1 (three-phase motor control timer output enabled).

16.6.2 Register Setting

- Do not write to the TAi1 register (i = 1, 2, 4) before and after timer B2 underflows. Before writing to the TAi1 register, read the TB2 register to verify that sufficient time remains until timer B2 underflows. Then, immediately write to the TAi1 register so no interrupt handling is performed during this write procedure. If the TB2 register indicates little time remains until the underflow, write to the TAi1 register after timer B2 underflows.

17. Serial Interface

The serial interface consists of five channels: UART0 to UART4.

Each channel has an exclusive timer to generate the transmit/receive clock and operates independently.

Figure 17.1 and Figure 17.2 show block diagrams of UART0 to UART2 and UART3 and UART4, respectively.

UARTi supports the following modes:

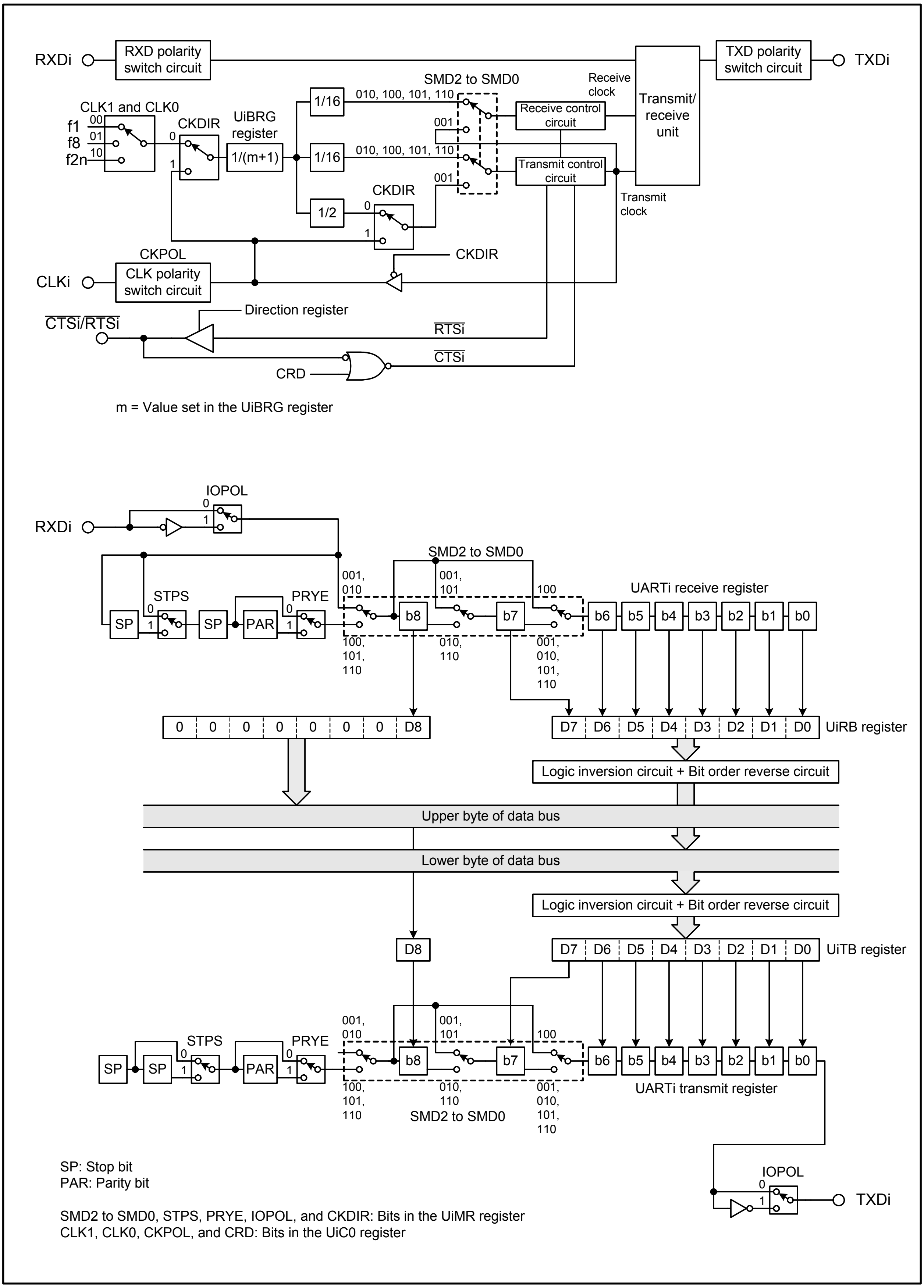
- Synchronous serial interface mode (for UART0 to UART4)
- Asynchronous serial interface mode (UART mode) (for UART0 to UART4)
- Special mode 1 (I^2C mode) (for UART0 to UART2)
- Special mode 2 (for UART0 to UART2)

Figure 17.3 to Figure 17.17 show registers associated with UARTi (i = 0 to 4).

Refer to the tables listing each mode for registers and pin settings.

Table 17.1 Comparison UART0 to UART4 Functions

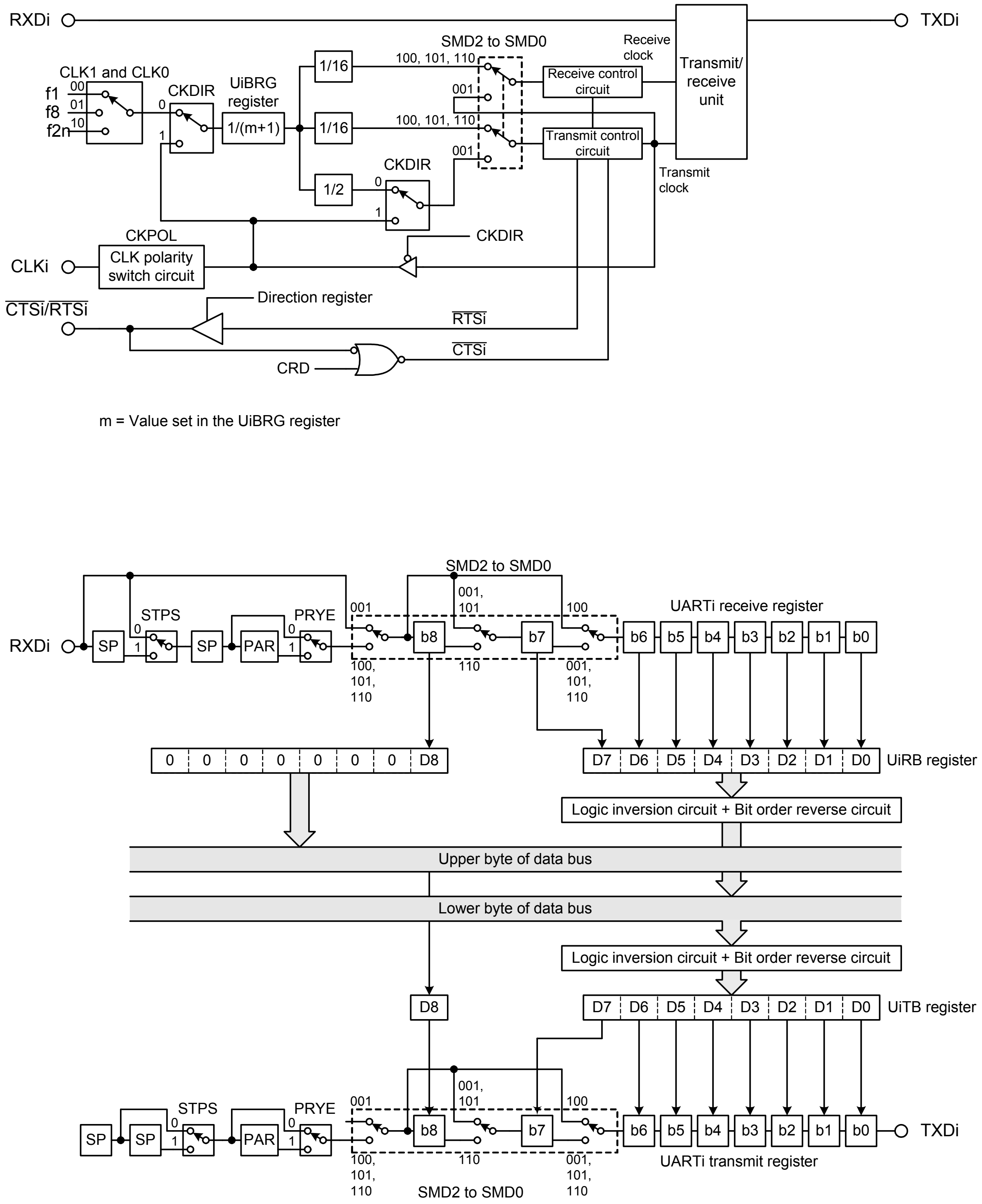
Mode/Function		UART0 to UART2	UART3, UART4
Synchronous serial interface mode		Available	Available
	Serial data logic inversion	Available	Not available
UART mode		Available	Available
	CTS/RTS function selection	Available	Available
	TXD and RXD I/O polarity selection	Available	Not available
Special mode 1 (I^2C mode)		Available	Not available
Special mode 2		Available	Not available



SP: Stop bit
PAR: Parity bit

SMD2 to SMD0, STPS, PRYE, IOPOL, and CKDIR: Bits in the UiMR register
CLK1, CLK0, CKPOL, and CRD: Bits in the UiC0 register

Figure 17.1 UARTi Block Diagram (i = 0 to 2)

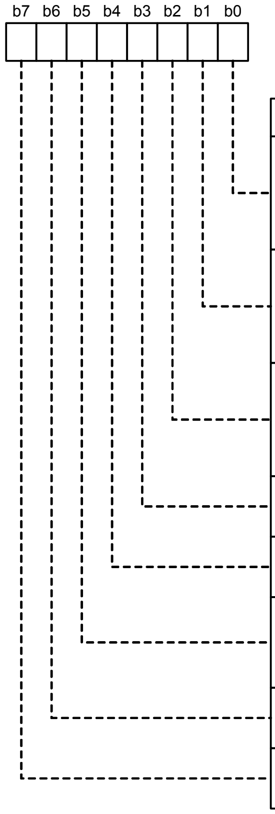


SP: Stop bit
PAR: Parity bit

SMD2 to SMD0, STPS, PRYE, IOPOL, and CKDIR: Bits in the UiMR register
CLK1, CLK0, CKPOL, and CRD: Bits in the UiC0 register

Figure 17.2 UARTi Block Diagram (i = 3, 4)

UARTi Transmit/Receive Mode Register (i = 0 to 2)



Symbol	Address	Reset Value
U0MR to U2MR	0368h, 02E8h, 0338h	0000 0000b

Bit Symbol	Bit Name	Function	RW
SMD0	Serial Interface Mode Select Bit	b2 b1 b0 0 0 0 : Serial interface disabled 0 0 1 : Synchronous serial interface mode 0 1 0 : I^2C mode 1 0 0 : UART mode, 7-bit character length 1 0 1 : UART mode, 8-bit character length 1 1 0 : UART mode, 9-bit character length Only use the combinations listed above	RW
SMD1			RW
SMD2			RW
CKDIR	Internal/External Clock Select Bit	0: Internal clock 1: External clock	RW
STPS	Stop Bit Length Select Bit	0: 1 stop bit 1: 2 stop bits	RW
PRY	Odd/Even Parity Select Bit	Enabled when the PRYE bit is 1 0: Odd parity 1: Even parity	RW
PRYE	Parity Enable Bit	0: Parity disabled 1: Parity enabled	RW
IOPOL	TXD, RXD Input/Output Polarity Switch Bit	0: Not inverted 1: Inverted	RW

Figure 17.3 Registers U0MR to U2MR

UARTi Transmit/Receive Mode Register (i = 3, 4)

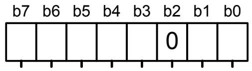
b7	b6	b5	b4	b3	b2	b1	b0
0							

Symbol	Address	Reset Value
U3MR, U4MR	01E0h, 01E8h	0000 0000b

Bit Symbol	Bit Name	Function	RW
SMD0	Serial Interface Mode Select Bit	b2 b1 b0 0 0 0 : Serial interface disabled 0 0 1 : Synchronous serial interface mode 1 0 0 : UART mode, 7-bit character length 1 0 1 : UART mode, 8-bit character length 1 1 0 : UART mode, 9-bit character length Only use the combinations listed above	RW
SMD1			RW
SMD2			RW
CKDIR	Internal/External Clock Select Bit	0: Internal clock 1: External clock	RW
STPS	Stop Bit Length Select Bit	0: 1 stop bit 1: 2 stop bits	RW
PRY	Odd/Even Parity Select Bit	Enabled when the PRYE bit is 1 0: Odd parity 1: Even parity	RW
PRYE	Parity Enable Bit	0: Parity disabled 1: Parity enabled	RW
— (b7)	Reserved	Should be written with 0	RW

Figure 17.4 Registers U3MR and U4MR

UARTi Transmit/Receive Control Register 0 (i = 0 to 2)



Symbol	Address	Reset Value
U0C0 to U2C0	036Ch, 02ECh, 033Ch	0000 1000b

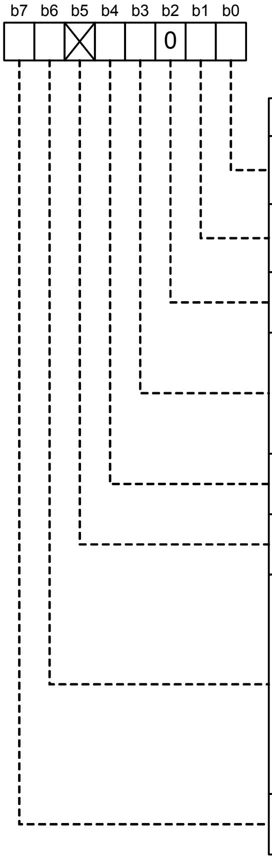
Bit Symbol	Bit Name	Function	RW
CLK0	UiBRG Count Source Select Bit	b1 b0 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : Do not use this combination	RW
CLK1			RW
— (b2)	Reserved	Should be written with 0	RW
TXEPT	Transmit Shift Register Empty Flag	0: Data held in the transmit shift register (transmission in progress) 1: No data held in the transmit shift register (transmission completed)	RO
CRD	CTS Function Disable Bit	0: CTS function enabled 1: CTS function disabled	RW
NCH	Data Output Select Bit	0: Pins TXDi/SDAi and SCLi are push-pull output 1: Pins TXDi/SDAi and SCLi are N-channel open drain output	RW
CKPOL	CLK Polarity Select Bit	0: Output transmit data on the falling edge of the transmit/receive clock and input receive data on the rising edge 1: Output transmit data on the rising edge of the transmit/receive clock and input receive data on the falling edge	RW
UFORM	Bit Order Select Bit (1)	0: LSB first 1: MSB first	RW

Note:

1. This bit is enabled when bits SMD2 to SMD0 in the UiMR register are set to 001b (synchronous serial interface mode) or 101b (UART mode, 8-bit character length). It should be set to 1 when bits SMD2 to SMD0 are set to 010b (I^2C mode) and should be set to 0 when they are set to 100b (UART mode, 7-bit character length) or 110b (UART mode, 9-bit character length).

Figure 17.5 Registers U0C0 to U2C0

UARTi Transmit/Receive Control Register 0 (i = 3, 4)



Symbol
U3C0, U4C0

Address
01E4h, 01ECh

Reset Value
00X0 1000b

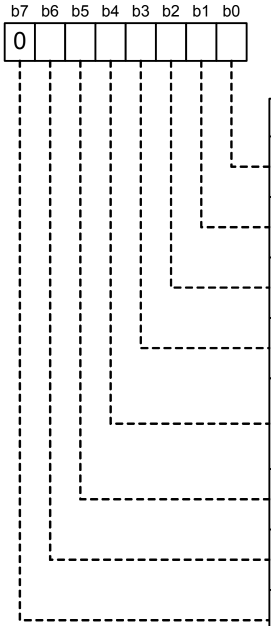
Bit Symbol	Bit Name	Function	RW
CLK0	UiBRG Count Source Select Bit	b1 b0 0 0 : f1 0 1 : f8 1 0 : f2n 1 1 : Do not use this combination	RW
CLK1			RW
— (b2)	Reserved	Should be written with 0	RW
TXEPT	Transmit Shift Register Empty Flag	0: Data held in the transmit shift register (transmission in progress) 1: No data held in the transmit shift register (transmission completed)	RO
CRD	CTS Function Disable Bit	0: CTS function enabled 1: CTS function disabled	RW
— (b5)	No register bit; should be written with 0 and read as undefined value		—
CKPOL	CLK Polarity Select Bit	0: Output transmit data on the falling edge of the transmit/receive clock and input receive data on the rising edge 1: Output transmit data on the rising edge of the transmit/receive clock and input receive data on the falling edge	RW
UFORM	Bit Order Select Bit (1)	0: LSB first 1: MSB first	RW

Note:
1. This bit is enabled when bits SMD2 to SMD0 in the UiMR register are set to 001b (synchronous serial interface mode) or 101b (UART mode, 8-bit character length). It should be set to 0 when they are set to 100b (UART mode, 7-bit character length) or 110b (UART mode, 9-bit character length).

Figure 17.6 Registers U3C0 and U4C0

UARTi Transmit/Receive Control Register 1 (i = 0 to 2)

Symbol	Address	Reset Value
U0C1 to U2C1	036Dh, 02EDh, 033Dh	0000 0010b



Bit Symbol	Bit Name	Function	RW
TE	Transmit Enable Bit	0: Transmission disabled 1: Transmission enabled	RW
TI	Transmit Buffer Empty Flag	0: Data held in the UiTB register 1: No data held in the UiTB register	RO
RE	Receive Enable Bit	0: Reception disabled 1: Reception enabled	RW
RI	Receive Complete Flag	0: No data held in the UiRB register 1: Data held in the UiRB register	RO
UiIRS	UARTi Transmit Interrupt Source Select Bit	0: UiTB register is empty (TI = 1) 1: Transmission is completed (TXEPT = 1)	RW
UiRRM	UARTi Continuous Receive Mode Enable Bit	0: Continuous receive mode disabled 1: Continuous receive mode enabled	RW
UiLCH	Logic Inversion Select Bit (1)	0: Data is not logic inverted 1: Data is logic inverted	RW
— (b7)	Reserved	Should be written with 0	RW

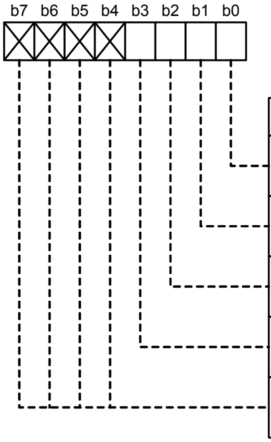
Note:

1. This bit is enabled when bits SMD2 to SMD0 in the UiMR register are set to 001b (synchronous serial interface mode), 100b (UART mode, 7-bit character length), or 101b (UART mode, 8-bit character length). It should be set to 0 when bits SMD2 to SMD0 are set to 010b (I^2C mode) or 110b (UART mode, 9-bit character length).

Figure 17.7 Registers U0C1 to U2C1

UARTi Transmit/Receive Control Register 1 (i = 3, 4)

Symbol	Address	Reset Value
U3C1, U4C1	01E5h, 01EDh	XXXX 0010b

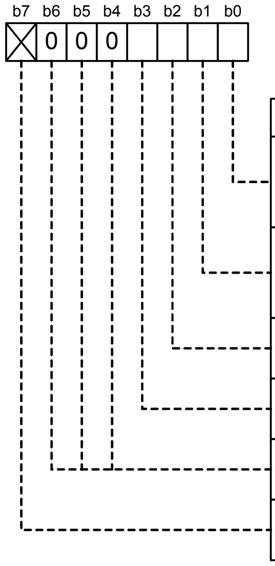


Bit Symbol	Bit Name	Function	RW
TE	Transmit Enable Bit	0: Transmission disabled 1: Transmission enabled	RW
TI	Transmit Buffer Empty Flag	0: Data held in the UiTB register 1: No data held in the UiTB register	RO
RE	Receive Enable Bit	0: Reception disabled 1: Reception enabled	RW
RI	Receive Complete Flag	0: No data held in the UiRB register 1: Data held in the UiRB register	RO
— (b7-b4)	No register bits; should be written with 0 and read as undefined value		—

Figure 17.8 Registers U3C1 and U4C1

UART3, UART4 Transmit/Receive Control Register 2

Symbol	Address	Reset Value
U34CON	01F0h	X000 0000b

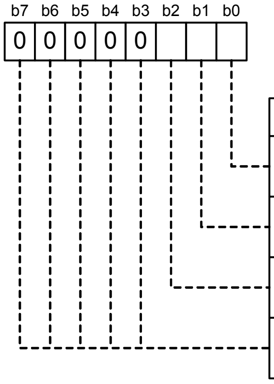


Bit Symbol	Bit Name	Function	RW
U3IRS	UART3 Transmit Interrupt Source Select Bit	0: Transmit buffer is empty (TI = 1) 1: Transmission is completed (TXEPT = 1)	RW
U4IRS	UART4 Transmit Interrupt Source Select Bit	0: Transmit buffer is empty (TI = 1) 1: Transmission is completed (TXEPT = 1)	RW
U3RRM	UART3 Continuous Receive Mode Enable Bit	0: Continuous receive mode disabled 1: Continuous receive mode enabled	RW
U4RRM	UART4 Continuous Receive Mode Enable Bit	0: Continuous receive mode disabled 1: Continuous receive mode enabled	RW
— (b6-b4)	Reserved	Should be written with 0	RW
— (b7)	No register bit; should be written with 0 and read as undefined value		—

Figure 17.9 U34CON Register

UARTi Special Mode Register (i = 0 to 2)

Symbol	Address	Reset Value
U0SMR to U2SMR	0367h, 02E7h, 0337h	0000 0000b

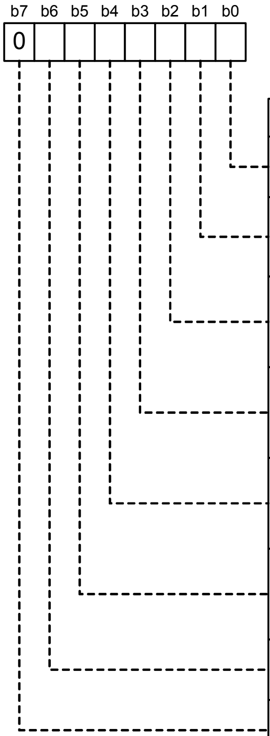


Bit Symbol	Bit Name	Function	RW
IICM	I^2C Mode Select Bit (1)	0: Mode other than I^2C mode 1: I^2C mode	RW
ABC	Arbitration Lost Detection Flag Control (1)	0: Update every bit 1: Update every byte	RW
BBS	Bus Busy Flag (1, 2)	0: Detect stop condition 1: Detect start condition (bus busy)	RW
— (b7-b3)	Reserved	Should be written with 0	RW

Notes:
1. This bit is used in I^2C mode.
2. The BBS bit can only be set to 0. This setting is unchanged even if it is set to 1.

Figure 17.10 Registers U0SMR to U2SMR

UARTi Special Mode Register 2 (i = 0 to 2)



Symbol	Address	Reset Value
U0SMR2 to U2SMR2	0366h, 02E6h, 0336h	0000 0000b

Bit Symbol	Bit Name	Function	RW
IICM2	I^2C Mode Select Bit 2	0: Use ACK/NACK interrupt 1: Use transmit/receive interrupt	RW
CSC	Clock Synchronization Bit (1)	0: Clock synchronization disabled 1: Clock synchronization enabled	RW
SWC	SCL Wait Auto Insert Bit (2)	0: No wait-state/wait-state cleared 1: Hold the SCLi pin low after the eighth bit is received	RW
ALS	SDA Output Auto Stop Bit (1)	When an arbitration lost is detected, 0: Do not stop the SDAi output 1: Stop the SDAi output	RW
STC	UARTi Auto Initialize Bit (2)	When a start condition is detected, 0: Do not initialize the circuit 1: Initialize the circuit	RW
SWC2	SCL Wait Output Bit 2 (1)	0: Output the transmit/receive clock at the SCLi pin 1: Hold the SCLi pin low	RW
SDHI	SDA Output Stop Bit (2)	0: Output data 1: Stop the output (high-impedance)	RW
— (b7)	Reserved	Should be written with 0	RW

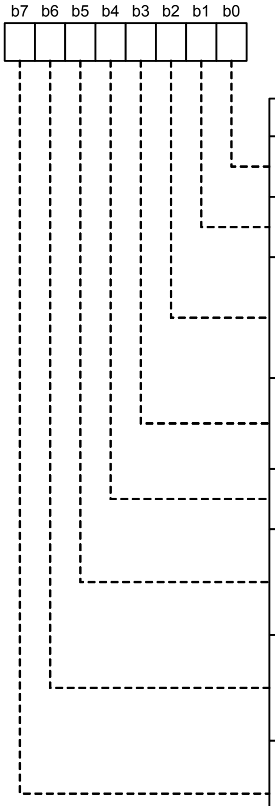
Notes:

1. This bit is used in master mode of I^2C mode.
2. This bit is used in slave mode of I^2C mode.

Figure 17.11 Registers U0SMR2 to U2SMR2

UARTi Special Mode Register 3 (i = 0 to 2)

Symbol	Address	Reset Value
U0SMR3 to U2SMR3	0365h, 02E5h, 0335h	0000 0000b



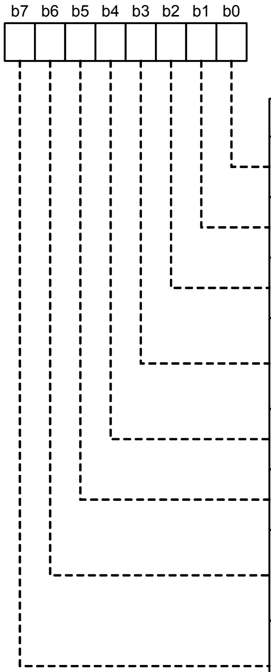
Bit Symbol	Bit Name	Function	RW
SSE	$\overline{\text{SS}}$ Pin Function Enable Bit [1, 2]	0: SS function disabled 1: SS function enabled	RW
CKPH	Clock-phase Set Bit	0: No clock delay 1: Clock delayed	RW
DINC	Serial Input Pin Set Bit [1]	0: Select the TXDi/RXDi pin (master mode) 1: Select the STXDi/SRXDi pin (slave mode)	RW
NODC	Clock Output Select Bit	0: The CLKi pin is push-pull output 1: The CLKi pin is N-channel open drain output	RW
ERR	Mode Fault Flag [1]	0: No mode fault detected 1: Mode fault detected [3]	RW
DL0	SDAi Digital Delay Time Set Bit [4, 5]	Based on the baud rate generator count source, the SDAi output is delayed as follows: b7 b6 b5 0 0 0 : No delay 0 0 1 : 1 to 2 cycles 0 1 0 : 2 to 3 cycles 0 1 1 : 3 to 4 cycles 1 0 0 : 4 to 5 cycles 1 0 1 : 5 to 6 cycles 1 1 0 : 6 to 7 cycles 1 1 1 : 7 to 8 cycles	RW
DL1			RW
DL2			RW

Notes:

1. This bit is used in special mode 2.
2. To use the SS function, the CRD bit in the UiC0 register should be set to 1 (CTS function disabled).
3. The ERR bit can only be set to 0. This setting is unchanged even if it is set to 1.
4. Bits DL2 to DL0 in I^{2}C mode generate a digital delay for the SDAi output. These bits should be set to 000b (no delay) in all modes other than I^{2}C mode.
5. When an external clock is selected, a delay of approximately 100 ns is added.

Figure 17.12 Registers U0SMR3 to U2SMR3

UARTi Special Mode Register 4 (i = 0 to 2)



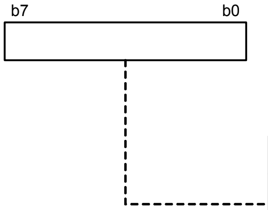
Symbol	Address	Reset Value
U0SMR4 to U2SMR4	0364h, 02E4h, 0334h	0000 0000b

Bit Symbol	Bit Name	Function	RW
STAREQ	Start Condition Generate Bit [1]	0: Clear 1: Start [2]	RW
RSTAREQ	Restart Condition Generate Bit [1]	0: Clear 1: Start [2]	RW
STPREQ	Stop Condition Generate Bit [1]	0: Clear 1: Start [2]	RW
STSPSEL	SCL, SDA Output Select Bit [1]	0: Select serial I/O circuit 1: Select start condition/stop condition generation circuit [3]	RW
ACKD	ACK Data Bit [4]	0: ACK 1: NACK	RW
ACKC	ACK Data Output Enable Bit [4]	0: Serial data output 1: ACK data output	RW
SCLHI	SCL Output Stop Bit [1]	When a stop condition is detected, 0: Do not stop SCLi output 1: Stop SCLi output	RW
SWC9	SCL Wait Auto Insert Bit 3 [4]	0: No wait-state/wait-state cleared 1: Hold the SCLi pin low after the ninth bit is received	RW

Notes:
1. This bit is used in master mode of I^{2}C mode. It can be set to 1 when the IICM bit in the UiSMR register is 1 (I^{2}C mode).
2. This bit becomes 0 when the condition is generated. The setting remains 1 when the condition is incomplete.
3. The STSPSEL bit should be set to 1 after setting the STAREQ, RSTAREQ, or STPREQ bit to 1.
4. This bit is used in slave mode of I^{2}C mode. It can be set to 1 when the IICM bit in the UiSMR register is 1 (I^{2}C mode).

Figure 17.13 Registers U0SMR4 to U2SMR4

UARTi Bit Rate Register (i = 0 to 4) (1, 2, 3)



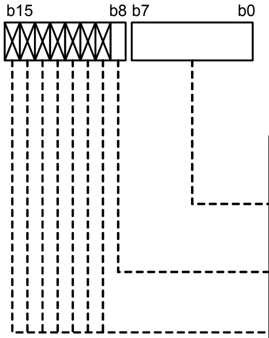
Symbol	Address	Reset Value
U0BRG to U3BRG	0369h, 02E9h, 0339h, 01E1h	Undefined
U4BRG	01E9h	Undefined

Function	Setting Range	RW
The UiBRG register divides the count source by n+1 (n = setting value)	00h to FFh	WO

Notes:
1. Set bits CLK1 and CLK0 in the UiC0 register before rewriting this register.
2. Use the MOV instruction to set this register.
3. The register should be set while no data is being transmitted/received.

Figure 17.14 Registers U0BRG to U4BRG

UARTi Transmit Buffer Register (i = 0 to 4) (1)



Symbol	Address	Reset Value
U0TB to U2TB	036Bh-036Ah, 02EBh-02EAh, 033Bh-033Ah	Undefined
U3TB, U4TB	01E3h-01E2h, 01EBh-01EAh	Undefined

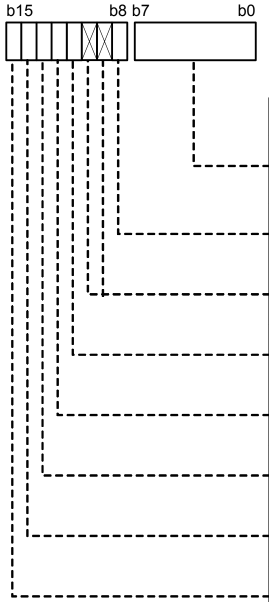
Bit Symbol	Function	RW
— (b7-b0)	Data (D7 to D0) transmitted	WO
— (b8)	Data (D8) transmitted	WO
— (b15-b9)	No register bits; should be written with 0	—

Note:

1. Use the MOV instruction to set this register.

Figure 17.15 Registers U0TB to U4TB

UARTi Receive Buffer Register (i = 0 to 2)



Symbol	Address	Reset Value
U0RB to U2RB	036Fh-036Eh, 033Fh-033Eh, 02EFh-02EEh	Undefined

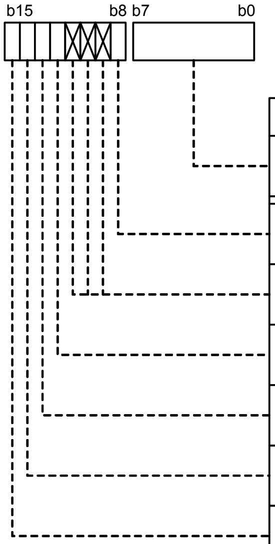
Bit Symbol	Bit Name	Function	RW
— (b7-b0)	—	Data (D7 to D0) received	RO
— (b8)	—	Data (D8) received	RO
— (b10-b9)	No register bits; should be written with 0 and read as 0		—
ABT	Arbitration Lost Detection Flag (1)	0: Not detected (win) 1: Detected (lose)	RW
OER	Overrun Error Flag (2)	0: No overrun error occurred 1: Overrun error occurred	RO
FER	Framing Error Flag (2, 3)	0: No framing error occurred 1: Framing error occurred	RO
PER	Parity Error Flag (2, 3)	0: No parity error occurred 1: Parity error occurred	RO
SUM	Error Sum Flag (2, 3)	0: No error occurred 1: Error occurred	RO

Notes:

1. The ABT bit can only be set to 0.
2. Bits OER, FER, PER, and SUM are set to 0 when bits SMD2 to SMD0 in the UiMR register are set to 000b (serial interface disabled) or the RE bit in the UiC1 register is set to 0 (reception disabled).
   When bits OER, FER, and PER are all set to 0, the SUM bit is also set to 0.
   Bits FER and PER are set to 0 when the lower byte in the UiRB register is read.
3. When bits SMD2 to SMD0 are 001b (synchronous serial interface mode) or 010b (I^2C mode), these error flags are disabled and read as an undefined value.

Figure 17.16 Registers U0RB to U2RB

UARTi Receive Buffer Register (i = 3, 4)



Symbol	Address	Reset Value
U3RB, U4RB	01E7h-01E6h, 01EFh-01EEh	Undefined

Bit Symbol	Bit Name	Function	RW
— (b7-b0)	—	Data (D7 to D0) received	RO
— (b8)	—	Data (D8) received	RO
— (b11-b9)	No register bits; should be written with 0 and read as 0		—
OER	Overrun Error Flag [1]	0: No overrun error occurred 1: Overrun error occurred	RO
FER	Framing Error Flag [1, 2]	0: No framing error occurred 1: Framing error occurred	RO
PER	Parity Error Flag [1, 2]	0: No parity error occurred 1: Parity error occurred	RO
SUM	Error Sum Flag [1, 2]	0: No error occurred 1: Error occurred	RO

Notes:

1. Bits OER, FER, PER, and SUM are set to 0 when bits SMD2 to SMD0 in the UiMR register are set to 000b (serial interface disabled) or the RE bit in the UiC1 register is set to 0 (reception disabled). When bits OER, FER, and PER are all set to 0, the SUM bit is also set to 0. Bits FER and PER are set to 0 when the lower byte in the UiRB register is read.
2. When bits SMD2 to SMD0 are 001b (synchronous serial interface mode) or 010b (I^2C mode), these error flags are disabled and read as an undefined value.

Figure 17.17 Registers U3RB and U4RB

17.1 Synchronous Serial Interface Mode

The synchronous serial interface mode allows data transmission/reception synchronized with the transmit/receive clock. Table 17.2 lists specifications of synchronous serial interface mode.

Table 17.2 Synchronous Serial Interface Mode Specifications

Item	Specification
Data format	8-bit character length
Transmit/receive clock	• The CKDIR bit in the UiMR register (i = 0 to 4) is set to 0 (internal clock): $\frac{fx}{2(m+1)}$ fx = f1, f8, f2n; m: UiBRG register setting value, 00h to FFh • The CKDIR bit is set to 1 (external clock): input to the CLKi pin
Transmit/receive control	CTS function enabled, RTS function enabled, or CTS/RTS function disabled
Transmit start conditions	The conditions for starting data transmission are as follows (1): • The TE bit in the UiC1 register is set to 1 (transmission enabled) • The TI bit in the UiC1 register is set to 0 (data held in the UiTB register) • Input level at the $\overline{\text{CTSi}}$ pin is low when the CTS function is selected
Receive start conditions	The conditions for starting data reception are as follows (1): • The RE bit in the UiC1 register is set to 1 (reception enabled) • The TE bit in the UiC1 register is set to 1 (transmission enabled) • The TI bit in the UiC1 register is set to 0 (data held in the UiTB register) • Input level at the $\overline{\text{CTSi}}$ pin is low when the CTS function is selected
Interrupt request generating timing	In transmit interrupt, one of the following conditions is selected to set the UiIRS bit in registers U0C1 to U2C1 and U34CON: • The UiIRS bit is set to 0 (transmit buffer in the UiTB register is empty): when data is transferred from the UiTB register to the UARTi transmit register (when the transmission has started) • The UiIRS bit is set to 1 (transmission is completed): when data transmission from the UARTi transmit register is completed In receive interrupt, • When data is transferred from the UARTi receive register to the UiRB register (when the reception is completed)
Error detection	Overrun error (2) This error occurs when the seventh bit of the next data is received before the UiRB register is read
Selectable functions	• CLK polarity Rising or falling edge of the transmit/receive clock for output and input of transmit/receive data • Bit order selection LSB first or MSB first • Continuous receive mode Data reception is enabled by a read access to the UiRB register • Serial data logic inversion (UART0 to UART2) This function logically inverses transmit/receive data

Notes:

1. When selecting an external clock, the following preconditions should be met:
   - The CLKi pin is held high when the CKPOL bit in the UiC0 register is set to 0 (transmit data output on the falling edge of the transmit/receive clock and receive data input on the rising edge).
   - The CLKi pin is held low when the CKPOL bit is set to 1 (transmit data output on the rising edge of the transmit/receive clock and receive data input on the falling edge).
2. If an overrun error occurs, the UiRB register is undefined. The IR bit in the SiRIC register does not change to 1 (interrupt requested).

Table 17.3 and Table 17.4 list register settings. When UARTi (i = 0 to 4) operating mode is selected, a high is output at the TXDi pin until transmission starts (the TXDi pin is high-impedance when the N-channel open drain output is selected).

Figure 17.18 and Figure 17.19 show examples of transmit and receive operations in synchronous serial interface mode, respectively.

Table 17.3 Register Settings in Synchronous Serial Interface Mode (for UART0 to UART2)

Register	Bits	Function
UiMR	7 to 4	Set the bits to 0000b
	CKDIR	Select either an internal clock or external clock
	SMD2 to SMD0	Set the bits to 001b
UiC0	UFORM	Select either LSB first or MSB first
	CKPOL	Select a transmit/receive clock polarity
	NCH	Select an output mode of the TXDi pin
	CRD	Select CTS function enabled or disabled
	TXEPT	Transmit register empty flag
	2	Set the bit to 0
	CLK1 and CLK0	Select a count source for the UiBRG register
UiC1	7	Set the bit to 0
	UiLCH	Set the bit to 1 to use logic inversion
	UiRRM	Set the bit to 1 to use continuous receive mode
	UiIRS	Select a source for the UARTi transmit interrupt
	RI	Receive complete flag
	RE	Set the bit to 1 to enable data reception
	TI	Transmit buffer empty flag
	TE	Set the bit to 1 to enable data transmission/reception
UiSMR	7 to 0	Set the bits to 00h
UiSMR2	7 to 0	Set the bits to 00h
UiSMR3	7 to 4	Set the bits to 0000b
	NODC	Select a clock output mode
	2 to 0	Set the bits to 000b
UiSMR4	7 to 0	Set the bits to 00h
UiBRG	7 to 0	Set the bit rate
UiTB	7 to 0	Set the data to be transmitted
UiRB	OER	Overrun error flag
	7 to 0	Received data is read

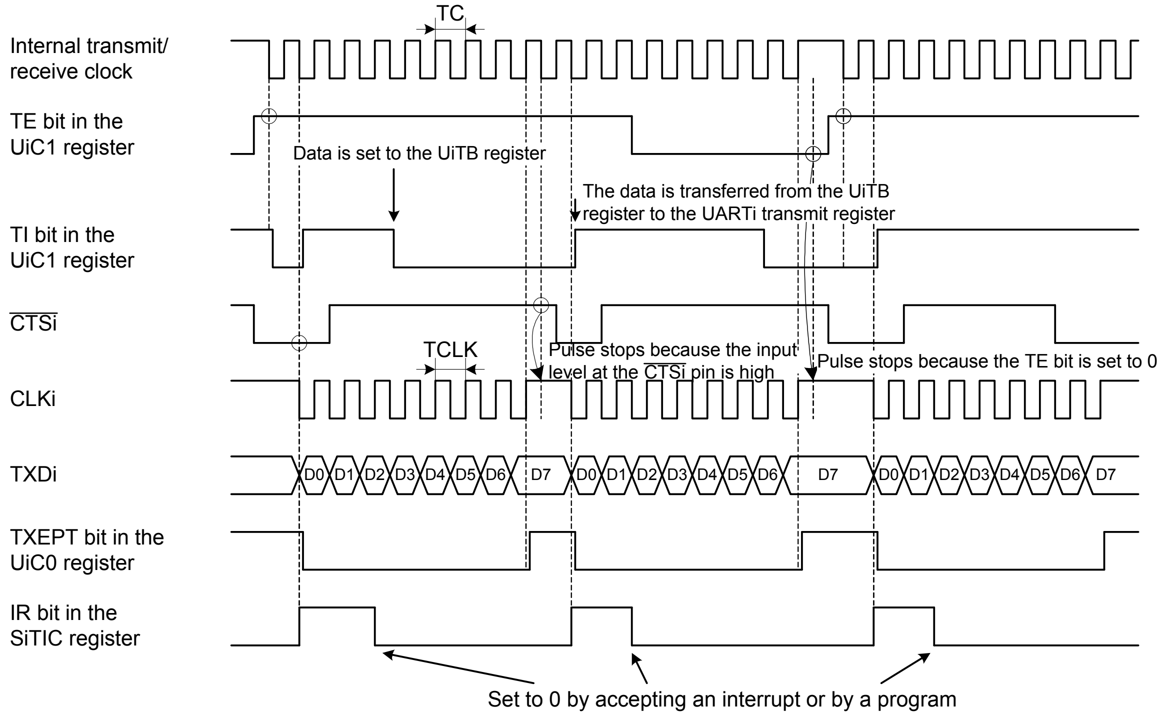
i = 0 to 2

Table 17.4 Register Settings in Synchronous Serial Interface Mode (for UART3 and UART4)

Register	Bits	Function
UiMR	7 to 4	Set the bits to 0000b
	CKDIR	Select an internal clock or external clock
	SMD2 to SMD0	Set the bits to 001b
UiC0	UFORM	Select either LSB first or MSB first
	CKPOL	Select a transmit/receive clock polarity
	5	Set the bit to 0
	CRD	Select CTS function enabled or disabled
	TXEPT	Transmit register empty flag
	2	Set the bit to 0
	CLK1 and CLK0	Select a count source for the UiBRG register
UiC1	RI	Receive complete flag
	RE	Set the bit to 1 to enable data reception
	TI	Transmit buffer empty flag
	TE	Set the bit to 1 to enable data transmission/reception
U34CON	UiRRM	Set the bit to 1 to use continuous receive mode
	UiIRS	Select an interrupt source for UARTi transmit
IFS0	IFS04	Select input pins for CLK4, RXD4, and $\overline{\text{CTS4}}$
	IFS02	Select input pins for CLK3, RXD3, and $\overline{\text{CTS3}}$
UiBRG	7 to 0	Set the bit rate
UiTB	7 to 0	Set the data to be transmitted
UiRB	OER	Overrun error flag
	7 to 0	Received data can be read

i = 3, 4

Transmit timing (when selecting an internal clock)



The figure above applies under the following conditions:

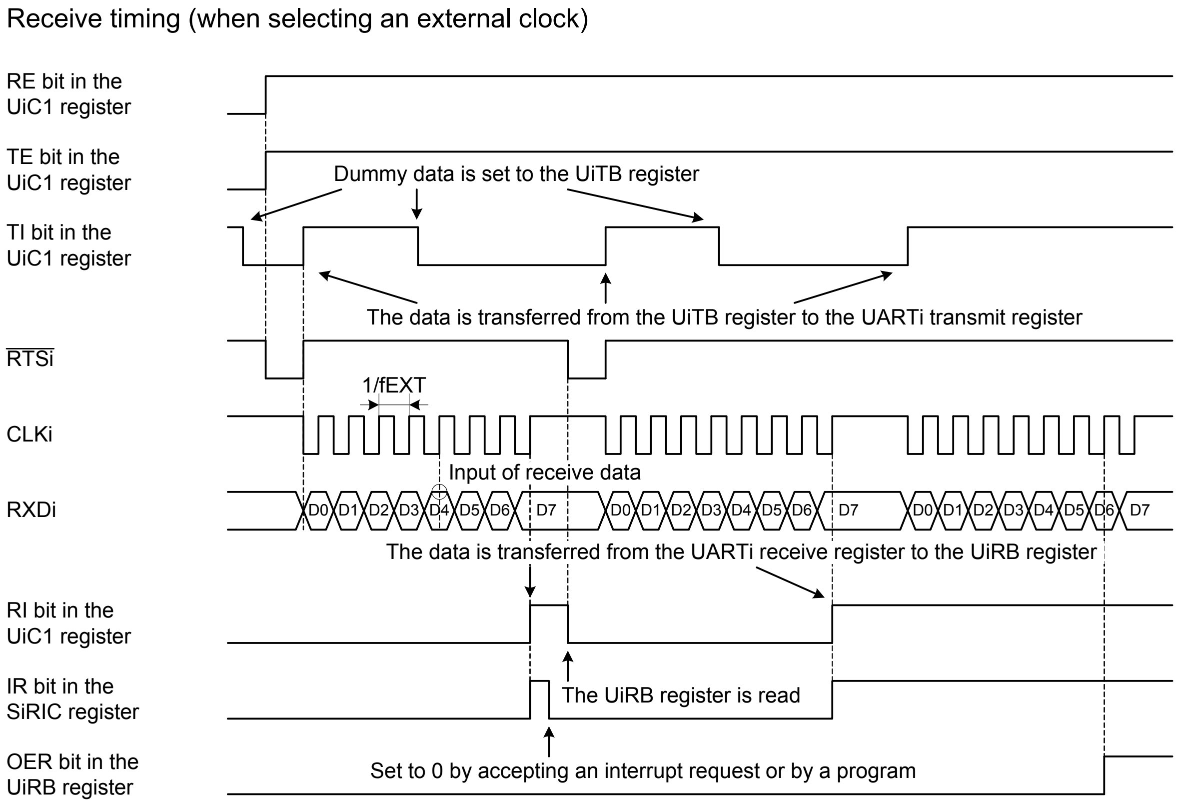
- The CKDIR bit in the UiMR register is 0 (internal clock).
- The CRD bit in the UiC0 register is 0 (CTS function enabled).
- The CKPOL bit in the UiC0 register is 0 (output transmit data on the falling edge of the transmit/receive clock).
- The UiIRS bit in registers UiC1 and U34CON is 0 (an interrupt request is generated when the transmit buffer is empty).

TC = TCLK = $2(m + 1)/fx$

fx: UiBRG count source frequency (f1, f8, or f2n)

m: Value set in the UiBRG register

Figure 17.18 Transmit Operation in Synchronous Serial Interface Mode



The figure above applies under the following conditions:
- The CKDIR bit in the UiMR register is 1 (external clock).
- The CKPOL bit in the UiC0 register is 0 (input receive data on the rising edge of the transmit/receive clock).

fEXT: External clock frequency

The following conditions should be met while an input level at the CLKi pin before receiving data is high:
- The TE bit in the UiC1 register is 1 (transmission enabled).
- The RE bit in the UiC1 register is 1 (reception enabled).
- Write of dummy data to the UiTB register.

Figure 17.19 Receive Operation in Synchronous Serial Interface Mode

17.1.1 Reset Procedure on Transmit/Receive Error

When a transmit/receive error occurs in synchronous serial interface mode, follow the procedures below to perform a reset:

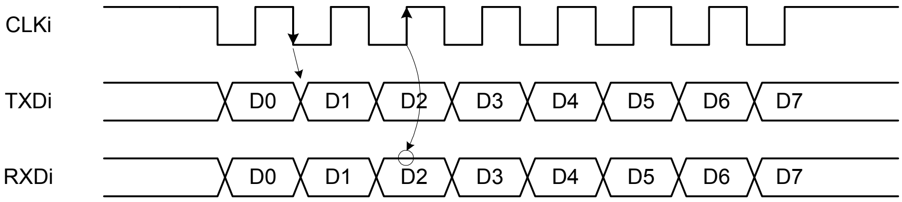
A. Reset procedure for the UiRB register (i = 0 to 4)
  (1) Set the RE bit in the UiC1 register to 0 (reception disabled).
  (2) Set bits SMD2 to SMD0 in the UiMR register to 000b (serial interface disabled).
  (3) Set bits SMD2 to SMD0 to 001b (synchronous serial interface mode).
  (4) Set the RE bit in the UiC1 register to 1 (reception enabled).

B. Reset procedure for the UiTB register
  (1) Set bits SMD2 to SMD0 in the UiMR register to 000b (serial interface disabled).
  (2) Set bits SMD2 to SMD0 to 001b (synchronous serial interface mode).
  (3) Irrespective of its status, set the TE bit in the UiC1 register to 1 (transmission enabled).

17.1.2 CLK Polarity

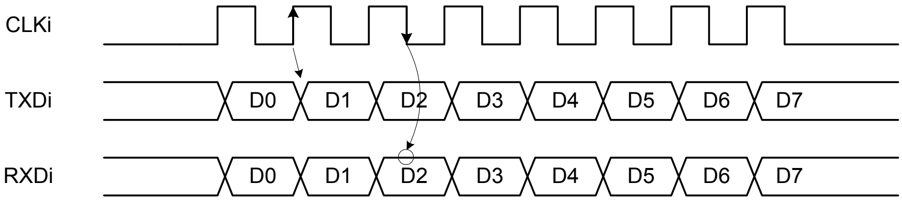
As shown in Figure 17.20, the polarity of the transmit/receive clock is selected using the CKPOL bit in the UiC0 register (i = 0 to 4).

Case 1: When the CKPOL bit in the UiC0 register is set to 0 (output transmit data on the falling edge of the transmit/receive clock and input receive data on the rising edge)



Notes:
1. The CLKi pin is held high when no data is transmitted/received.
2. The figure above applies under the following conditions:
   - The UFORM bit in the UiC0 register is 0 (LSB first).
   - The UiLCH bit in the UiC1 register is 0 (data is not logic inverted).

Case 2: When the CKPOL bit in the UiC0 register is set to 1 (output transmit data on the rising edge of the transmit/receive clock and input receive data on the falling edge)



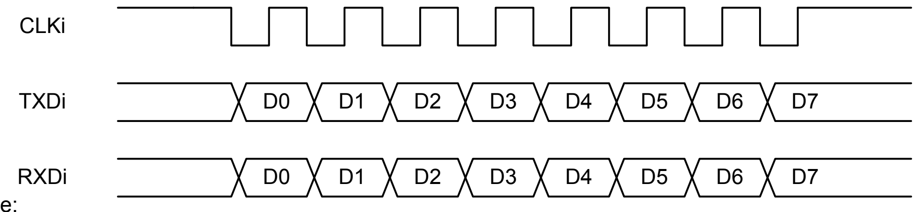
Notes:
3. The CLKi pin is held low when no data is transmitted/received.
4. The figure above applies under the following conditions:
   -The UFORM bit in the UiC0 register is 0 (LSB first).
   -The UiLCH bit in the UiC1 register is 0 (data is not logic inverted).

Figure 17.20 Transmit/Receive Clock Polarity (i = 0 to 4)

17.1.3 LSB First and MSB First Selection

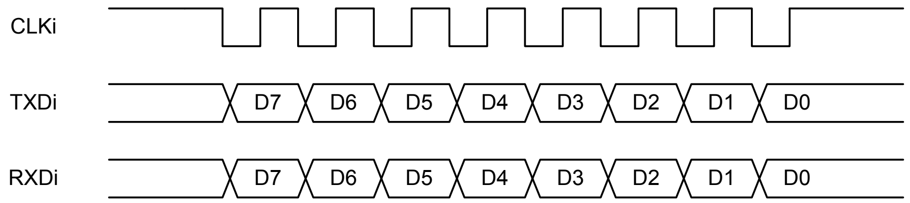
As shown in Figure 17.21, the bit order is selected by setting the UFORM bit in the UiC0 register (i = 0 to 4).

Case 1: When the UFORM bit in the UiC0 register is set to 0 (LSB first)



Note:
1. The figure above applies under the following conditions:
   - The CKPOL bit in the UiC0 register is 0 (output transmit data on the falling edge of the transmit/ receive clock and input receive data on the rising edge).
   - The UiLCH bit in the UiC1 register is 0 (data is not logic inverted).

Case 2: When the UFORM bit in the UiC0 register is set to 1 (MSB first)



Note:
2. The figure above applies under the following conditions:
   - The CKPOL bit in the UiC0 register is 0 (output transmit data on the falling edge of the transmit/ receive clock and input receive data on the rising edge).
   - The UiLCH bit in the UiC1 register is 0 (data is not logic inverted).

Figure 17.21 Bit Order (i = 0 to 4)

17.1.4 Continuous Receive Mode

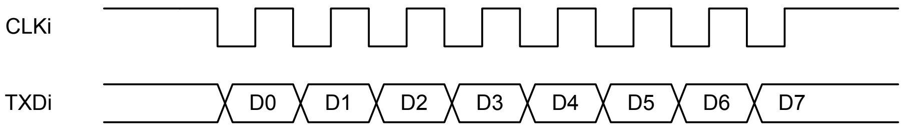
In continuous receive mode, data reception is automatically enabled by a read access to the receive buffer register without writing dummy data to the transmit buffer register. To start data reception, however, dummy data is required to read the receive buffer register.

When the UiRRM bit (i = 0 to 4) in registers U0C1 to U2C1 and the U34CON register is set to 1 (continuous receive mode enabled), the TI bit in the UiC1 register is set to 0 (data held in the UiTB register) by a read access to the UiRB register. In this UiRRM bit setting, no dummy data should be written to the UiTB register.

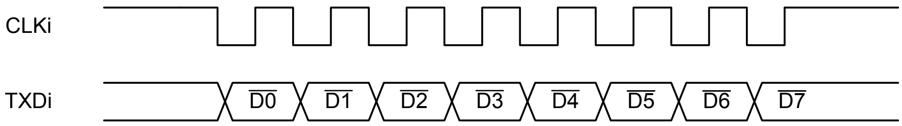
17.1.5 Serial Data Logic Inversion

When the UiLCH bit in the UiC1 register (i = 0 to 2) is set to 1 (data is logic inverted), the logical value written in the UiTB register is inverted before being transmitted. The UiRB register is read as logic-inverted receive data. Figure 17.22 shows the logic inversion of serial data.

Case 1: When the UiLCH bit in the UiC1 register is set to 0 (data is not logic inverted)



Case 2: When the UiLCH bit in the UiC1 register is set to 1 (data is logic inverted)



Note:

1. The figures above apply under the following conditions:
   - The CKPOL bit in the UiC0 register is 0 (output transmit data on the falling edge of the transmit/receive clock and input receive data on the rising edge).
   - The UFORM bit is 0 (LSB first).

Figure 17.22 Serial Data Logic Inversion (i = 0 to 2)

17.1.6 CTS/RTS Function

CTS function controls data transmission using the $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$ pin (i = 0 to 4). When an input level at the pin becomes low, data transmission starts. If the input level changes to high during transmission, the transmission of the next data is stopped.

In synchronous serial interface mode, the transmitter is required to operate even during the receive operation. If CTS function is enabled, the input level at the $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$ pin should be low to start data reception as well.

RTS function indicates receiver status using the $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$ pin. When data reception is ready, the output level at the pin becomes low. It becomes high on the first falling edge of the CLKi pin.

17.2 Asynchronous Serial Interface Mode (UART Mode)

The UART mode enables data transmission/reception synchronized with an internal clock generated by a trigger on the falling edge of the start bit. Table 17.5 lists specifications of UART mode.

Table 17.5 UART Mode Specifications

Item	Specification
Data format	• Start bit: 1-bit • Data bit (data character): 7-bit, 8-bit, or 9-bit • Parity bit: odd, even, or none • Stop bit: 1-bit or 2-bit
Transmit/receive clock	• The CKDIR bit in the UiMR register (i = 0 to 4) is set to 0 (internal clock): $\frac{fx}{16(m+1)}$ fx = f1, f8, f2n; m: UiBRG register setting value, 00h to FFh • The CKDIR bit is set to 1 (external clock) $\frac{fEXT}{16(m+1)}$ $fEXT$: Clock applied to the CLKi pin
Transmit/receive control	CTS function enabled, RTS function enabled, or CTS/RTS function disabled
Transmit start conditions	The conditions for starting data transmission are as follows: • The TE bit in the UiC1 register is set to 1 (transmission enabled) • The TI bit in the UiC1 register is set to 0 (data held in the UiTB register) • Input level at the $\overline{\text{CTSi}}$ pin is low when CTS function is selected
Receive start conditions	The conditions for starting data reception are as follows: • The RE bit in the UiC1 register is set to 1 (reception enabled) • The start bit is detected
Interrupt request generating timing	In transmit interrupt, one of the following conditions is selected to set the UiIRS bit in registers U0C1 to U2C1 and the U34CON register: • The UiIRS bit is set to 0 (transmit buffer in the UiTB register is empty): when data is transferred from the UiTB register to the UARTi transmit register (when the transmission has started) • The UiIRS bit is set to 1 (transmission is completed): when data transmission from the UARTi transmit register is completed In receive interrupt, • When data is transferred from the UARTi receive register to the UiRB register (when reception is completed)
Error detection	• Overrun error (1) This error occurs when 1 bit prior to the stop bit (when 1 stop bit length is selected) or the first stop bit (when 2 stop bit length is selected) of the next data is received before the UiRB register is read • Framing error This error occurs when the required number of stop bits is not detected • Parity error This error occurs when an even number of 1's in parity and character bits is detected while the odd number is set, or vice versa. The parity should be enabled • Error sum flag This flag becomes 1 when any of overrun error, framing error, or parity error occurs
Selectable functions	• Bit order selection LSB first or MSB first • Serial data logic inversion This function logically inverses transmit/receive data. The start bit and stop bit are not inverted • TXD/RXD I/O polarity switching The output level from the TXD pin and the input level to the RXD pin are inverted. All I/O levels are inverted

Note:

1. When an overrun error occurs, the UiRB register is undefined. The IR bit in the SiRIC register does not change to 1 (interrupt requested).

Table 17.6 and Table 17.7 list register settings. When UARTi (i = 0 to 4) operating mode is selected, a high is output at the TXDi pin until transmission starts (the TXDi pin is high-impedance when the N-channel open drain output is selected).
Figure 17.23 and Figure 17.24 show examples of transmit operations in UART mode. Figure 17.25 shows an example of receive operation.

Table 17.6 Register Settings in UART Mode (UART0 to UART2)

Register	Bits	Function
UiMR	IOPOL	Select I/O polarity of pins TXD and RXD
	PRY and PRYE	Select parity enabled or disabled, and odd or even
	STPS	Select a stop bit length
	CKDIR	Select an internal clock or external clock
	SMD2 to SMD0	Set the bits to 100b in 7-bit character length
		Set the bits to 101b in 8-bit character length
		Set the bits to 110b in 9-bit character length
UiC0	UFORM	Select LSB first or MSB first in 8-bit character length. Set the bit to 0 in 7-bit or 9-bit character length
	CKPOL	Set the bit to 0
	NCH	Select an output mode for the TXDi pin
	CRD	Select CTS function enabled or disabled
	TXEPT	Transmit register empty flag
	2	Set the bit to 0
	CLK1 and CLK0	Select a count source for the UiBRG register
UiC1	7	Set the bit to 0
	UiLCH	Set the bit to 1 to use logic inversion
	UiRRM	Set the bit to 0
	UiIRS	Select an interrupt source for UARTi transmission
	RI	Receive complete flag
	RE	Set the bit to 1 to enable data reception
	TI	Transmit buffer empty flag
	TE	Set the bit to 1 to enable data transmission
UiSMR	7 to 0	Set the bits to 00h
UiSMR2	7 to 0	Set the bits to 00h
UiSMR3	7 to 0	Set the bits to 00h
UiSMR4	7 to 0	Set the bits to 00h
UiBRG	7 to 0	Set the bit rate
UiTB	8 to 0	Set the data to be transmitted (1)
UiRB	OER, FER, PER, and SUM	Error flag
	8 to 0	Received data is read (1)

i = 0 to 2

Note:

1. The bits used are as follows: 7-bit character length: bits 6 to 0
8-bit character length: bits 7 to 0
9-bit character length: bits 8 to 0

Table 17.7 Register Settings in UART Mode (UART3, UART4)

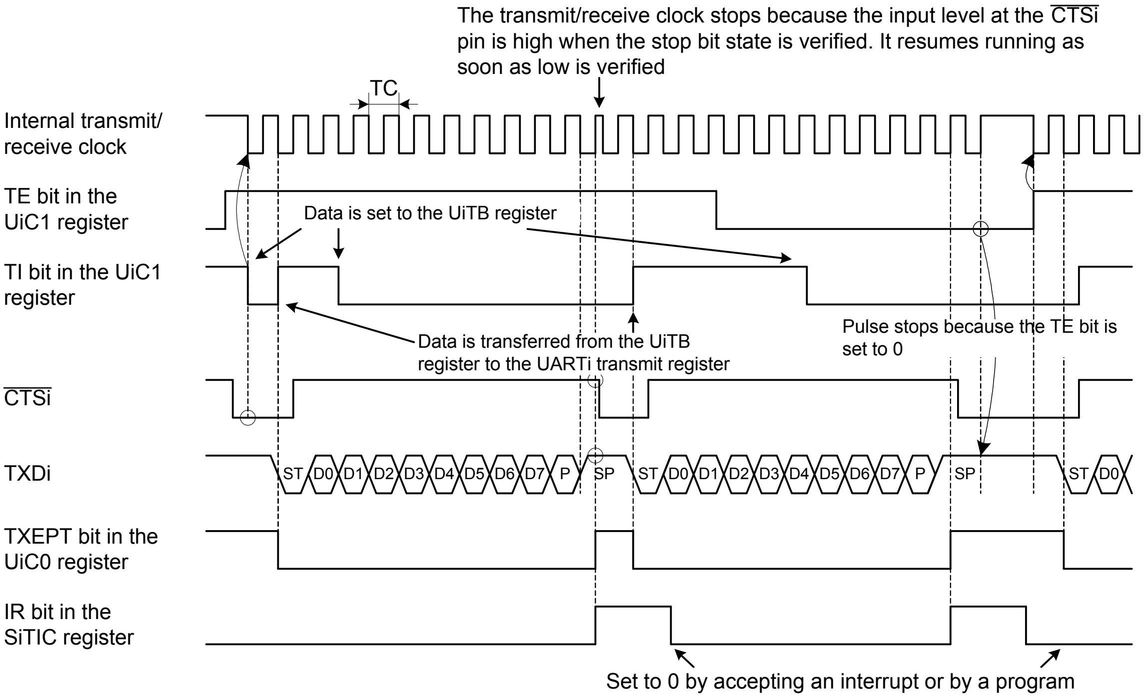
Register	Bits	Function
UiMR	PRY and PRYE	Select parity enabled or disabled, and odd or even
	STPS	Select a stop bit length
	CKDIR	Select an internal clock or external clock
	SMD2 to SMD0	Set the bits to 100b in 7-bit character length
		Set the bits to 101b in 8-bit character length
		Set the bits to 110b in 9-bit character length
UiC0	UFORM	Select LSB first or MSB first in 8-bit character length. Set the bit to 0 in 7-bit or 9-bit character length
	CKPOL	Set the bit to 0
	5	Set the bit to 0
	CRD	Select CTS function enabled or disabled
	TXEPT	Transmit register empty flag
	2	Set the bit to 0
	CLK1 and CLK0	Select a count source for the UiBRG register
UiC1	RI	Receive complete flag
	RE	Set the bit to 1 to enable data reception
	TI	Transmit buffer empty flag
	TE	Set the bit to 1 to enable data transmission
U34CON	UiRRM	Set the bit to 0
	UiIRS	Select an interrupt source for UARTi transmission
UiBRG	7 to 0	Set the bit rate
IFS0	IFS04	Select input pins for CLK4, RXD4, and $\overline{\text{CTS4}}$
	IFS02	Select input pins for CLK3, RXD3, and $\overline{\text{CTS3}}$
UiTB	8 to 0	Set the data to be transmitted (1)
UiRB	OER, FER, PER, and SUM	Error flag
	8 to 0	Received data is read (1)

i = 3, 4

Note:

1. The bits used are as follows: 7-bit character length: bits 6 to 0
   8-bit character length: bits 7 to 0
   9-bit character length: bits 8 to 0

Example of data transmit timing when the character length is 8-bit (parity enabled, 1 stop bit)



The transmit/receive clock stops because the input level at the $\overline{\text{CTSi}}$ pin is high when the stop bit state is verified. It resumes running as soon as low is verified

Data is set to the UiTB register

Data is transferred from the UiTB register to the UARTi transmit register

Pulse stops because the TE bit is set to 0

Set to 0 by accepting an interrupt or by a program

ST: Start bit
P: Parity bit
SP: Stop bit

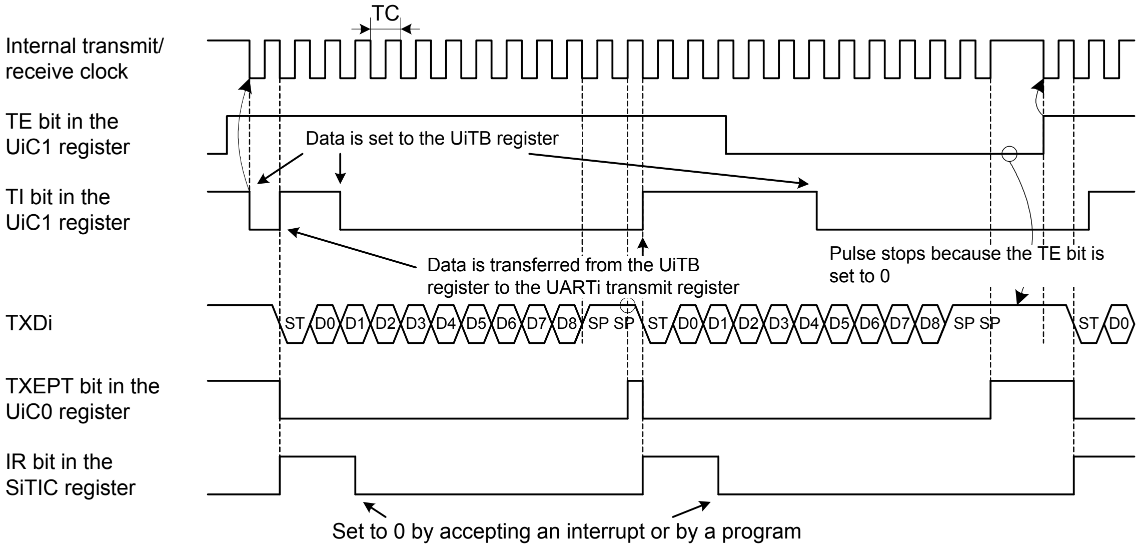
The figure above applies under the following conditions:

- The PRYE bit in the UiMR register is 1 (parity enabled).
- The STPS bit in the UiMR register is 0 (1 stop bit).
- The CRD bit in the UiC0 register is 0 (CTS function enabled).
- The UiIRS bit in the UiC1 register is 1 (an interrupt request is generated when transmission is comepleted).

TC = 16 (m + 1) / fx or 16 (m + 1) / $fEXT$
fx: UiBRG count source frequency (f1, f8, or f2n)
$fEXT$: UiBRG count source frequency (external clock)
m: Value set in the UiBRG register

Figure 17.23 Transmit Operation in UART Mode (1/2) (i = 0 to 4)

Example of data transmit timing when the character length is 9-bit (parity disabled, 2 stop bits)



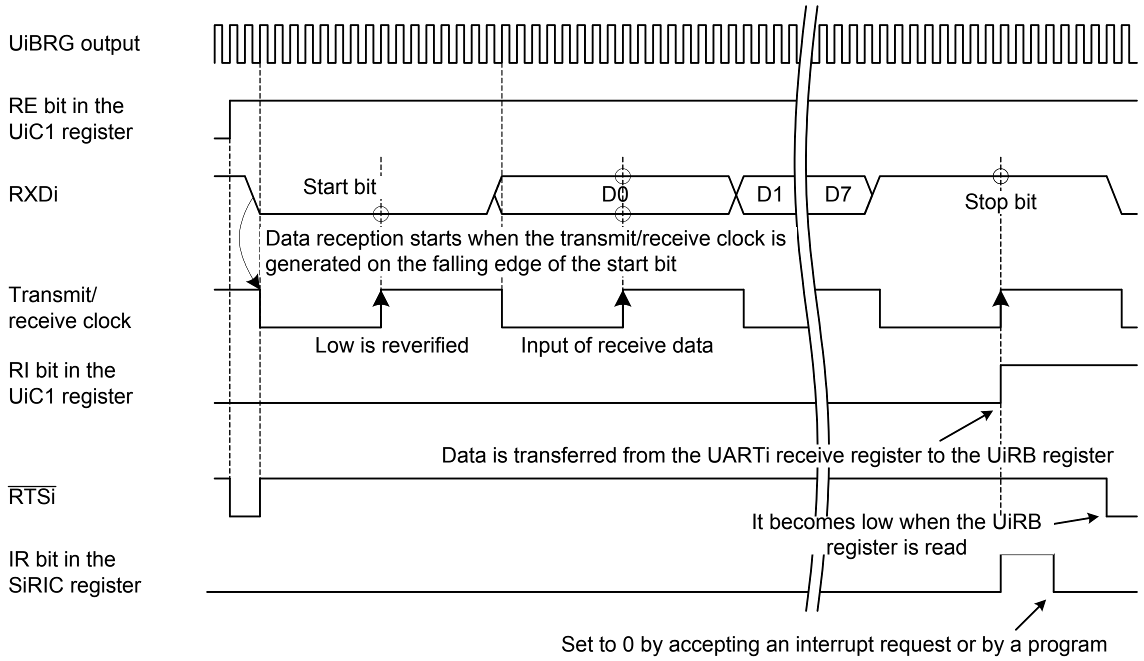
ST: Start bit
P: Parity bit
SP: Stop bit

The figure above applies under the following conditions:
- The PRYE bit in the UiMR register is 0 (parity disabled).
- The STPS bit in the UiMR register is 1 (2 stop bits).
- The CRD bit in the UiC0 register is 1 (CTS function disabled).
- The UiIRS bit in the UiC1 register is 0 (an interrupt request is generated when the transmit buffer is empty).

TC = 16 $(m + 1) / fx$ or 16 $(m + 1) / fEXT$
fx: UiBRG count source frequency (f1, f8, or f2n)
$fEXT$: UiBRG count source frequency (external clock)
m: Value set in the UiBRG register

Figure 17.24 Transmit Operation in UART Mode (2/2) (i = 0 to 4)

Example of data receive timing when the character length is 8-bit (parity disabled, 1 stop bit)



The figure above applies under the following conditions:
- The PRYE bit in the UiMR register is 0 (parity disabled).
- The STPS bit in the UiMR register is 0 (1 stop bit).

Figure 17.25 Receive Operation in UART Mode (i = 0 to 4)

17.2.1 Bit Rate

In UART mode, the bit rate is a clock frequency which is divided by a setting value of the UiBRG register (i = 0 to 4) and again divided by 16. Table 17.8 lists an example of bit rate setting.

Table 17.8 Bit Rate Setting

Bit Rate (bps)	Count Source of BRG	Peripheral Clock: 30 MHz		Peripheral Clock: 32 MHz	
		Setting value of BRG: n	Actual bit rate (bps)	Setting value of BRG: n	Actual bit rate (bps)
1200	f8	194 (C2h)	1202	207 (CHh)	1202
2400	f8	97 (61h)	2392	103 (67h)	2404
4800	f8	48 (30h)	4783	51 (33h)	4808
9600	f1	194 (C2h)	9615	207 (CFh)	9615
14400	f1	129 (81h)	14423	138 (8Ah)	14388
19200	f1	97 (61h)	19133	103 (67h)	19231
28800	f1	64 (40h)	28846	68 (44h)	28986
31250	f1	59 (3Bh)	31250	63 (3Fh)	31250
38400	f1	48 (30h)	38265	51 (33h)	38462
51200	f1	36 (24h)	50676	38 (26h)	51282

17.2.2 Reset Procedure on Transmit/Receive Error

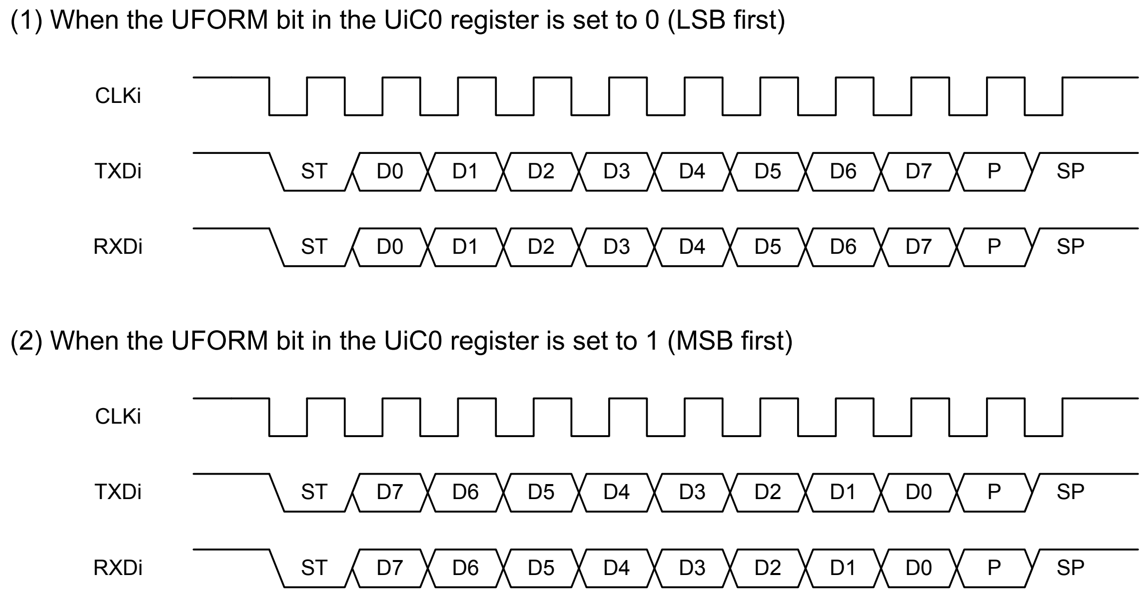
When a transmit/receive error occurs in UART mode, follow the procedure below to perform a reset:

A. Reset procedure for the UiRB register (i = 0 to 4)
  (1) Set the RE bit in the UiC1 register to 0 (reception disabled).
  (2) Set the RE bit in the UiC1 register to 1 (reception enabled).

B. Reset procedure for the UiTB register
  (1) Set bits SMD2 to SMD0 in the UiMR register to 000b (serial interface disabled).
  (2) Set again bits SMD2 to SMD0 to either of 001b, 101b, or 110b.
  (3) Irrespective of its status, set the TE bit in the UiC1 register to 1 (transmission enabled).

17.2.3 LSB First and MSB First Selection

As shown in Figure 17.26, the bit order is selected by setting the UFORM bit in the UiC0 register (i = 0 to 4). This function is available when the character length is 8-bit.


(1) When the UFORM bit in the UiC0 register is set to 0 (LSB first)

(2) When the UFORM bit in the UiC0 register is set to 1 (MSB first)

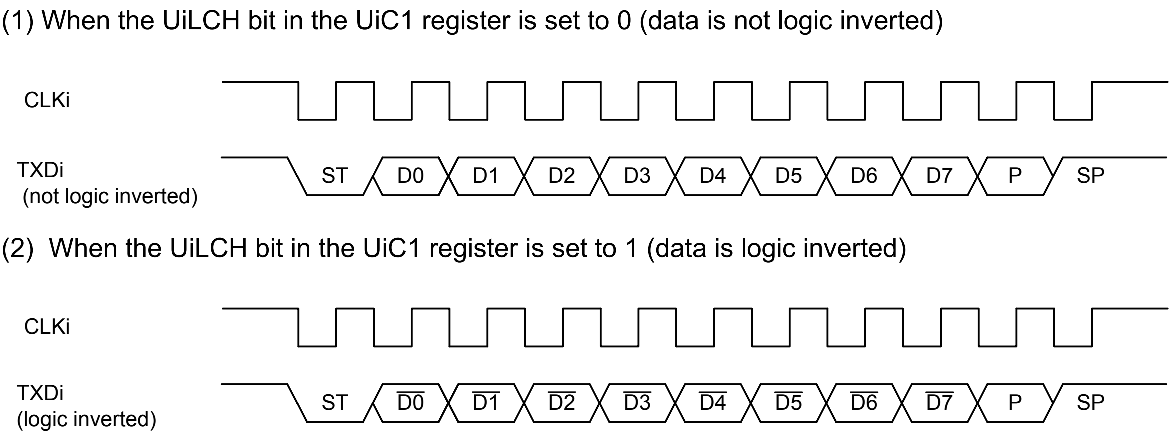
Note:
1. The figure above applies under the following conditions:
   - The UiLCH bit in the UiC1 register is 0 (data is not logic inverted).
   - The STPS bit in the UiMR register is 0 (1 stop bit).
   - The PRYE bit is 1 (parity enabled).

ST: Start bit
P: Parity bit
SP: Stop bit

Figure 17.26 Bit Order (i = 0 to 4)

17.2.4 Serial Data Logic Inversion

When the UiLCH bit in the UiC1 register (i = 0 to 2) is set to 1 (data is logic inverted), the logical value written in the UiTB register is inverted before being transmitted. The UiRB register is read as logic-inverted receive data. The parity bit is not inverted. Figure 17.27 shows the logic inversion of serial data.



Note:

1. The figure above applies under the following conditions:
   - The UFORM bit in the UiC0 register is 0 (LSB first).
   - The STPS bit in the UiMR register is 0 (1 stop bit).
   - The PRYE bit is 1 (parity enabled).

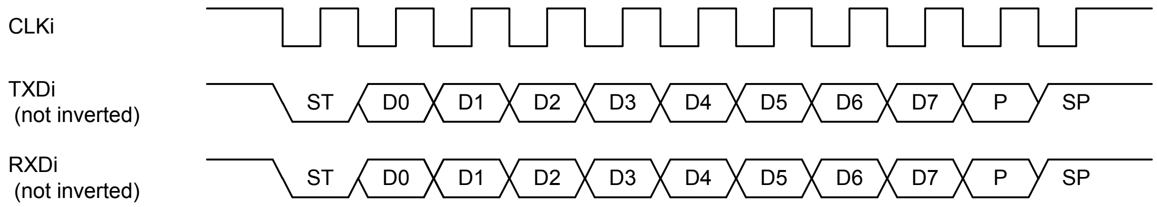
ST: Start bit
P: Parity bit
SP: Stop bit

Figure 17.27 Serial Data Logic Inversion (i = 0 to 2)

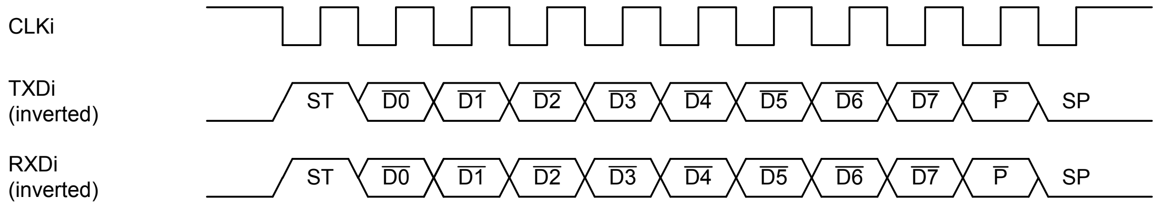
17.2.5 TXD and RXD I/O Polarity Inversion

The output level at the TXD pin and the input level at the RXD pin are inverted by this function. All I/O data levels, including the start bit, stop bit, and parity bit are inverted by setting the IOPOL bit in the UiMR register (i = 0 to 2) to 1 (inverted). Figure 17.28 shows TXD and RXD I/O polarity inversion.

(1) When the IOPOL bit in the UiMR register is set to 0 (not inverted)



(2) When the IOPOL bit in the UiMR register is set to 1 (inverted)



Note:
1. The figure above applies under the following conditions:
   - The UFORM bit in the UiC0 register is 0 (LSB first).
   - The STPS bit in the UiMR register is 0 (1 stop bit).
   - The PRYE bit is 1 (parity enabled).

ST: Start bit
P: Parity bit
SP: Stop bit

Figure 17.28 TXD and RXD I/O Polarity Inversion (i = 0 to 2)

17.2.6 CTS/RTS Function

CTS function controls data transmission using the $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$ pin (i = 0 to 4). When an input level at the pin becomes low, data transmission starts. If the input level changes to high during transmit operation, transmission of the next data is stopped.

RTS function indicates receiver status using the $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$ pin. When the MCU is ready to receive data, the output level at the pin becomes low. It becomes high on the first falling edge of the CLKi pin.

17.3 Special Mode 1 (I^2C Mode)

This mode uses an I^2C-typed interface for communication. Table 17.9 lists specifications of the I^2C mode.

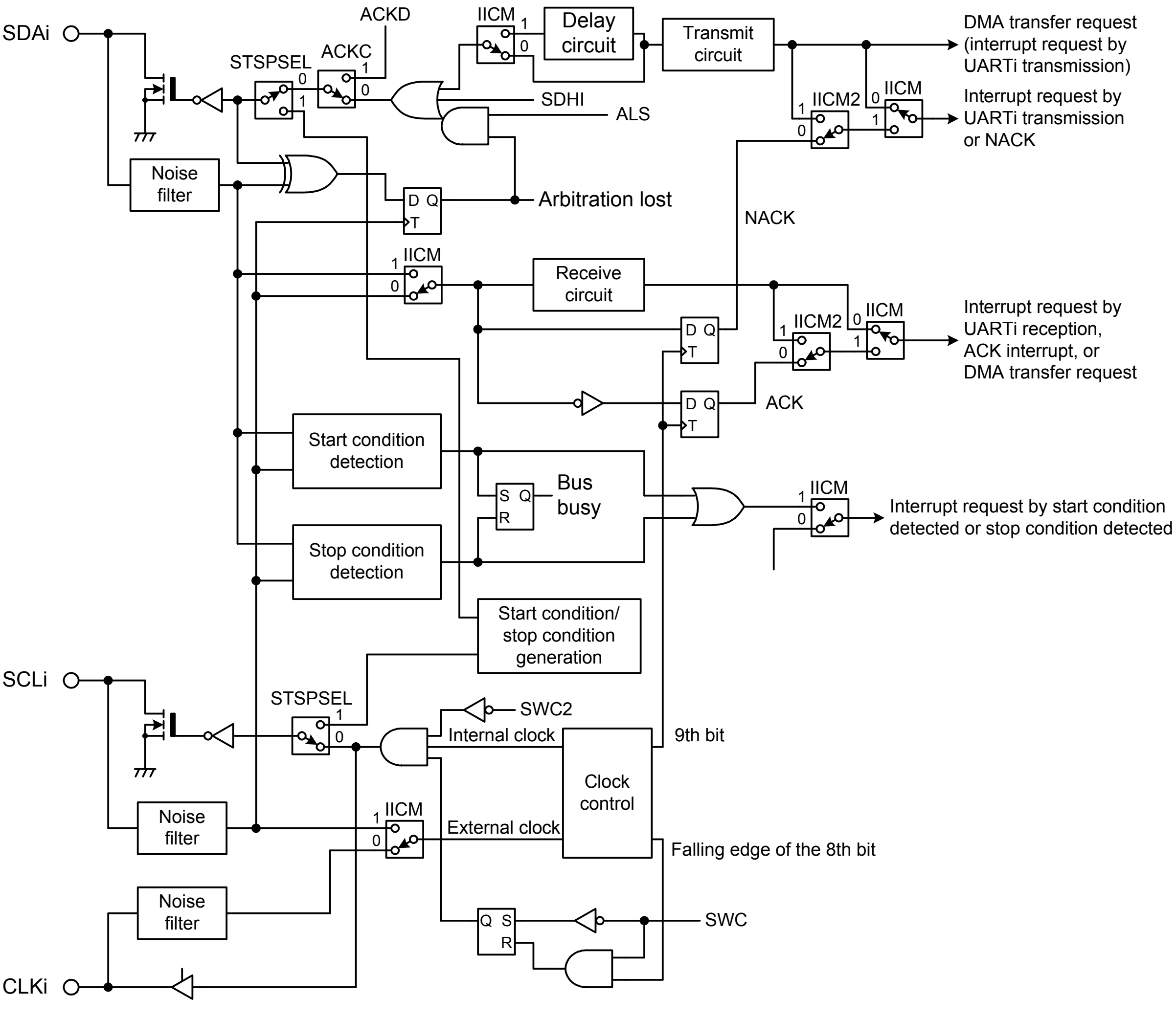
Table 17.9 I^2C Mode Specifications

Item	Specification
Data format	8-bit character length
Transmit/receive clock	In master mode • The CKDIR bit in the UiMR register (i = 0 to 2) is set to 0 (internal clock): $\frac{fx}{2(m+1)}$ fx = f1, f8, f2n m: UiBRG register setting value, 00h to FFh In slave mode • The CKDIR bit is set to 1 (external clock): input to the SCLi pin
Transmit start conditions	The conditions for starting data transmission are as follows (1): • The TE bit in the UiC1 register is set to 1 (transmission enabled) • The TI bit in the UiC1 register is set to 0 (data held in the UiTB register)
Receive start conditions	The conditions for starting data reception are as follows (1): • The RE bit in the UiC1 register is set to 1 (reception enabled) • The TE bit in the UiC1 register is set to 1 (transmission enabled) • The TI bit in the UiC1 register is set to 0 (data held in the UiTB register)
Interrupt request generating timing	When any of the following is detected: start condition, stop condition, NACK (not-acknowledge), or ACK (acknowledge)
Error detection	Overrun error (2) This error occurs when the eighth bit of the next data is received before the UiRB register is read
Selectable functions	• Arbitration lost Update timing of the ABT bit in the UiRB register can be selected • SDAi digital delay No digital delay or two to eight cycles of digital delay of UiBRG count source • Clock phase setting Clock delayed or no clock delay

Notes:
1. When an external clock is selected, the conditions should be met while the external clock signal is held high.
2. If an overrun error occurs, the UiRB register is undefined. The IR bit in the SiRIC register does not change to 1 (interrupt requested).

Table 17.10 lists register settings in I^2C mode, and Table 17.11 and Table 17.12 list I^2C mode functions. Figure 17.29 shows a block diagram of I^2C mode, and Figure 17.30 shows timings for the transfer to the UiRB register (i = 0 to 2) and the interrupt.

As shown in Table 17.11 and Table 17.12, UARTi enters this mode when bits SMD2 to SMD0 in the UiMR register (i = 0 to 2) are set to 010b, and the IICM bit in the UiSMR register is set to 1. Since a transmit signal at the SDAi pin is output via the delay circuit, it changes after the SCLi pin is stably held low.



IICM: Bit in the UiSMR register
IICM2, SWC, ALS, SWC2, and SDHI: Bits in the UiSMR2 register
STSPSEL, ACKD, and ACKC: Bits in the UiSMR4 register

Figure 17.29 I^{2}C Mode Block Diagram (i = 0 to 2)

Table 17.10 Register Settings in I^{2}C Mode (i = 0 to 2)

Register	Bits	Function	
		Master	Slave
UiMR	IOPOL	Set the bit to 0	
	CKDIR	Set the bit to 0	Set the bit to 1
	SMD2 to SMD0	Set the bit to 010b	
UiC0	7 to 4	Set the bits to 1011b	
	TXEPT	Transmit register empty flag	
	2	Set the bit to 0	
	CLK1 and CLK0	Select a count source for the UiBRG register	Disabled
UiC1	7 to 5	Set the bits to 000b	
	UiIRS	Set the bit to 1	
	RI	Receive complete flag	
	RE	Set the bit to 1 to enable data reception	
	TI	Transmit buffer empty flag	
	TE	Set the bit to 1 to enable data transmission/reception	
UiSMR	7 to 3	Set the bits to 00000b	
	BBS	Bus busy flag	
	ABC	Select an arbitration lost detection timing	Disabled
	IICM	Set the bit to 1	
UiSMR2	7	Set the bit to 0	
	SDHI	Set the bit to 1 to disable the SDA output	
	SWC2	Set the bit to 1 to hold the SCL output at a forcible low	
	STC	Set the bit to 0	Set the bit to 1 to reset UARTi by detecting the start condition
	ALS	Set the bit to 1 to stop the output at the SDAi pin to detect an arbitration lost	Set the bit to 0
	SWC	Set the bit to 1 to hold a low output at the SCLi pin after receiving the eighth bit of the clock	
	CSC	Set the bit to 1 to enable clock synchronization	Set the bit to 0
	IICM2	Refer to Table 17.11 and Table 17.12	
UiSMR3	DL2 to DL0	Set the digital delay value of SDAi	
	4 to 2	Set the bit to 000b	
	CKPH	Refer to Table 17.11 and Table 17.12	
	SSE	Set the bit to 0	
UiSMR4	SWC9	Set the bit to 0	Set the bit to 1 to hold a low output at the SCLi pin after receiving the ninth bit of the clock
	SCLHI	Set the bit to 1 to stop the SCL output to detect stop condition	Set the bit to 0
	ACKC	Set the bit to 1 for ACK data output	
	ACKD	Select ACK or NACK	
	STSPSEL	Set the bit to 1 when any condition is output	Set the bit to 0
	STPREQ	Set the bit to 1 to generate a stop condition	Set the bit to 0
	RSTAREQ	Set the bit to 1 to generate a restart condition	Set the bit to 0
	STAREQ	Set the bit to 1 to generate a start condition	Set the bit to 0
UiBRG	7 to 0	Set the bit rate	Disabled
UiTB	8	Set the bit to 1 when transmitting. Set the bit to the value of the ACK bit when receiving	
	7 to 0	Set the data to be transmitted when transmitting. Set the register to FFh when receiving	
UiRB	OER	Overrun error flag	
	ABT	Arbitration lost detection flag	Disabled
	8	D0 is loaded immediately after a receive interrupt is generated. ACK or NACK is loaded after a transmit interrupt is generated	
	7 to 0	D7 to D1 are read immediately after a receive interrupt is generated. D7 to D0 are read after a transmit interrupt is generated	

Table 17.11 I^{2}C Mode Functions (i = 0 to 2) (1/2)

Function	Synchronous Serial Interface Mode (SMD2 to SMD0 are 001b, IICM is 0)	I^{2}C Mode (SMD2 to SMD0 are 010b, IICM is 1)			
		IICM2 is 0 (ACK/NACK interrupt)		IICM2 is 1 (Transmit/receive interrupt)	
		CKPH is 0 (No clock delay)	CKPH is 1 (Clock delayed)	CKPH is 0 (No clock delay)	CKPH is 1 (Clock delayed)
Source of software interrupt numbers 39 to 41 (1) (refer to Figure 17.30)	—	Start condition or stop condition detection (refer to Table 17.13)			
Source of software interrupt numbers 17, 19, and 33 (1) (refer to Figure 17.30)	UARTi transmission: Transmission started or completed (selected using the UiIRS register)	NACK detection: Rising edge of the ninth bit of SCLi		UARTi transmission: Rising edge of the ninth bit of SCLi	UARTi transmission: Falling edge of the ninth bit of SCLi
Source of software interrupt numbers 18, 20, and 34 (1) (refer to Figure 17.30)	UARTi reception: Receiving at eighth bit CKPOL is 0 (rising edge) CKPOL is 1 (falling edge)	ACK detection: Rising edge of the ninth bit of SCLi		UARTi reception: Falling edge of the eighth bit of SCLi	
Data transfer timing from the UART receive register to the UiRB register	CKPOL is 0 (rising edge) CKPOL is 1 (falling edge)	Rising edge of the ninth bit of SCLi		Falling edge of the eighth bit of SCLi	Falling edge of the eighth bit and rising edge of the ninth bit of SCLi
UARTi transmit output delay	No delay	Delayed			
Pins P6_3, P6_7, P7_0	TXDi output	SDAi I/O			
Pins P6_2, P6_6, P7_1	RXDi input	SCLi I/O			
Pins P6_1, P6_5, P7_2	Select CLKi input or output	— (Not used in I^{2}C mode)			

Note:

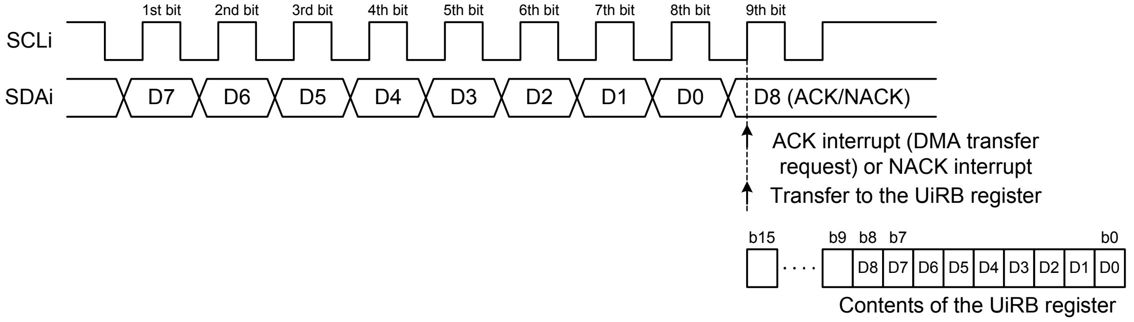
1. Steps to change an interrupt source are as follows:
   (1) Disable the interrupt of the corresponding software interrupt number.
   (2) Change the source of interrupt.
   (3) Set the IR bit of the corresponding software interrupt number to 0 (no interrupt requested).
   (4) Set bits ILVL2 to ILVL0 of the corresponding software interrupt number.

Table 17.12 I^{2}C Mode Functions (i = 0 to 2) (2/2)

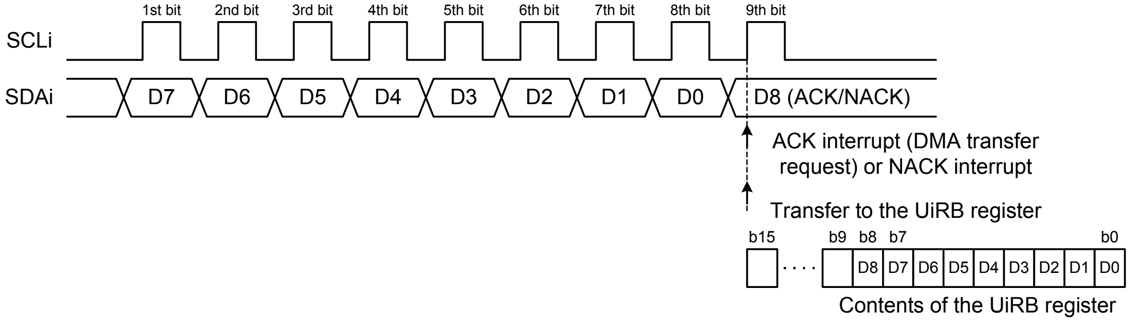
Function	Synchronous Serial Interface Mode (SMD2 to SMD0 are 001b, IICM is 0)	I^{2}C Mode (SMD2 to SMD0 are 010b, IICM is 1)			
		IICM2 is 0 (ACK/NACK interrupt)		IICM2 is 1 (Transmit/receive interrupt)	
		CKPH is 0 (No clock delay)	CKPH is 1 (Clock delayed)	CKPH is 0 (No clock delay)	CKPH is 1 (Clock delayed)
Read level at pins RXDi and SCLi	Readable irrespective of the port direction bit				
Default output value at the SDAi pin	—	High (Value set in the port Pi register (i = 0 to 7) if the I/O port is selected by output function select registers)			
SCLi default and end values	—	High	Low	High	Low
DMA source (refer to Figure 17.30)	UARTi reception	ACK detection		UARTi reception: Falling edge of the eighth bit of SCLi	
Store received data	The first to eighth bits of received data are stored into bits 0 to 7 in the UiRB register	The first to eighth bits of received data are stored into bits 7 to 0 in the UiRB register		The first to seventh bits of received data are stored into bits 6 to 0 in the UiRB register and the eighth bit is stored into bit 8	Same as on the left column on the first data storing. (1) The first to eighth bits of received data are stored into 7 to 0 bits in the UiRB register and the ninth bit is stored into bit 8 on the second data storing (2)
Read received data	The UiRB register status is read as it is			Bits 6 to 0 in the UiRB register are read as bits 7 to 1 and bit 8 is read as bit 0	Same as on the left column on the first read. (1) The UiRB register status is read as it is on the second read (2)

Notes:
1. The first data transfer to the UiRB register starts on the rising edge of the eighth bit of SCLi.
2. The second data transfer to the UiRB register starts on the rising edge of the ninth bit of SCLi.

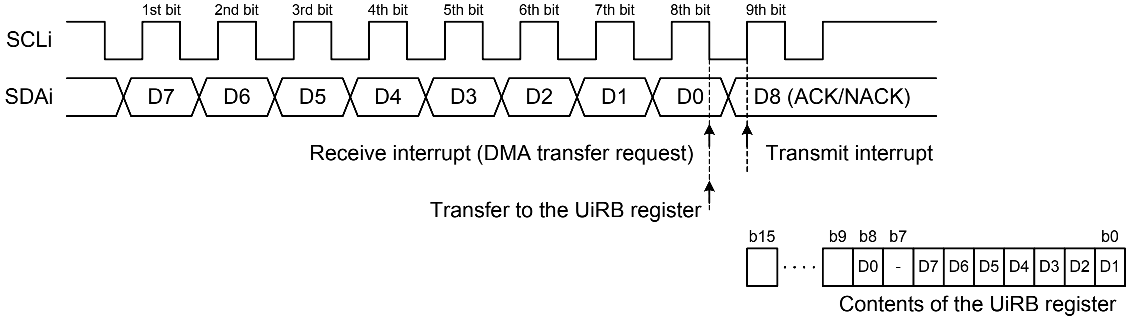
(1) When the IICM2 bit is set to 0 (Use ACK/NACK interrupt) and the CKPH bit is set to 0 (no clock delay)



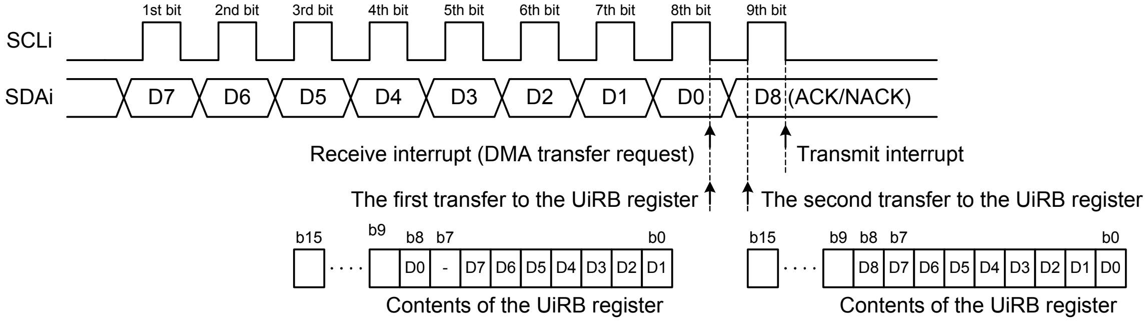
(2) When the IICM2 bit is set to 0 and the CKPH bit is set to 1 (clock delayed)



(3) When the IICM2 bit is set to 1 (Use transmit/receive interrupt) and the CKPH bit is set to 0



(4) When the IICM2 bit is set to 1 and the CKPH bit is set to 1



The figure above applies under the following condition:
- The CKDIR bit in the UiMR register is 0 (internal clock).

Figure 17.30 Timings for the Transfer and Interrupt to the UiRB Register (i = 0 to 2)

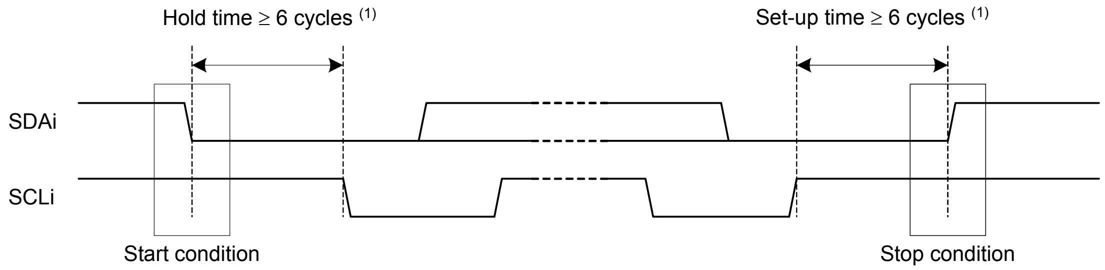
17.3.1 Start Condition and Stop Condition Detection

The start condition and stop condition are detected by their respective detectors.

The start condition detection interrupt request is generated by a high-to-low transition at the SDAi pin while the SCLi (i = 0 to 2) pin is held high. The stop condition detection interrupt request is generated by a low-to-high transition at the SDAi pin while the SCLi pin is held high.

The start condition detection interrupt shares interrupt control registers and vectors with the stop condition detection interrupt. The BBS bit in the UiSMR register determines which interrupt is requested.

To detect a start condition or stop condition, both set-up and hold times require at least six cycles of the peripheral clock (f1) as shown in Figure 17.31. To meet the condition for the Fast-mode specification, f1 must be at least 10 MHz.



Note:
1. These are cycles of the peripheral clock (f1).

Figure 17.31 Start Condition and Stop Condition Detection Timing (i = 0 to 2)

17.3.2 Start Condition and Stop Condition Generation

The start condition, restart condition, and stop condition are generated by bits STAREQ, RSTAREQ, and STPREQ in the UiSMR4 register (i = 0 to 2), respectively. To output a start condition, the STSPSEL bit in the UiSMR4 register should be set to 1 (start condition/stop condition generator selected) after setting the STAREQ bit to 1 (start). To output a restart condition or stop condition, the STSPSEL bit should be set to 1 after setting RSTAREQ bit or STPREQ bit to 1, respectively.

Table 17.13 and Figure 17.32 show the functions of the STSPSEL bit.

Table 17.13 STSPSEL Bit Functions

Function	STSPSEL is 0	STSPSEL is 1
Start condition and stop condition generation	Output is provided by the program with port (no auto generation by hardware)	Start condition or stop condition is output according to the STAREQ, RSTAREQ, or STPREQ bit
Start condition and stop condition interrupt request generating timing	When start condition or stop condition is detected	When start condition or stop condition generation is completed

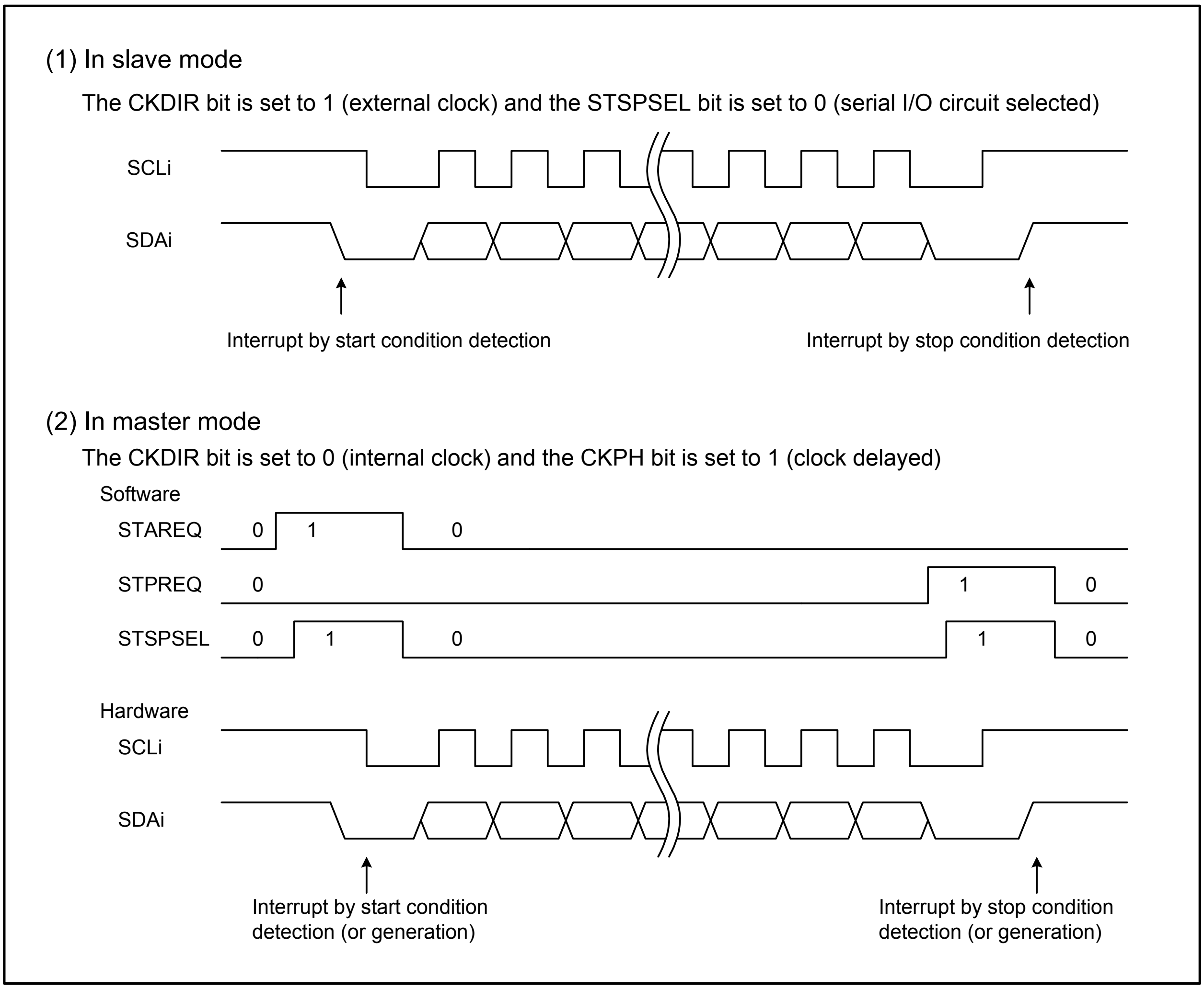


Figure 17.32 STSPSEL Bit Functions (i = 0 to 2)

17.3.3 Arbitration

The MCU determines whether the transmit data matches data input to the SDAi pin on the rising edge of the SCLi. If it does not match the input data, the arbitration takes place at the SDAi pin by switching off the data output stage.

The ABC bit in the UiSMR register (i = 0 to 2) determines the update timing for the ABT bit in the UiRB register.

When the ABC bit is set to 0 (update per bit), the ABT bit is set to 1 (arbitration is lost) as soon as a data discrepancy is detected. If not detected, the ABT bit is set to 0 (arbitration is won). When the ABC bit is set to 1 (update every byte), the ABT bit is set to 1 on the falling edge of the eighth bit of the SCLi if any discrepancy is detected. In this ABC bit setting, the ABT bit should be set to 0 to start the next 1-byte transfer.

When the ALS bit in the UiSMR2 register is set to 1 (SDA output stop enabled), an arbitration lost occurs. As soon as the ABT bit is set to 1, the SDAi pin becomes high-impedance.

17.3.4 SCL Control and Clock Synchronization

Data transmission/reception in I^2C mode uses the transmit/receive clock as shown in Figure 17.30. The clock speed increase makes it difficult to secure the required time for ACK generation and data transmit procedure. I^2C mode supports a function of wait-state insertion to secure this required time and a function of clock synchronization with a wait-state inserted by other devices.

The SWC bit in the UiSMR2 register (i = 0 to 2) is used to insert a wait-state for ACK generation. When the SWC bit is set to 1 (hold the SCLi pin low after the eighth bit is received), the SCLi pin is held low on the falling edge of the eighth bit of the SCLi. When the SWC bit is set to 0 (no wait-state/wait-state cleared), the SCLi line is released.

When the SWC2 bit in the UiSMR2 register is set to 1 (hold the SCLi pin low), the SCLi pin is forced low even during transmission or reception. When the SWC2 bit is set to 0 (output the transmit/receive clock at the SCLi pin), the SCLi line is released to output the transmit/receive clock.

The SWC9 bit in the UiSMR4 register is used to insert a wait-state for checking received acknowledge bits. While the CKPH bit in the UiSMR3 register is set to 1 (clock delayed), when the SWC9 bit is set to 1 (hold the SCLi pin low after the ninth bit is received), the SCLi pin is held low on the falling edge of the ninth bit of the SCLi. When the SWC9 bit is set to 0 (no wait-state/wait-state cleared), the SCLi line is released.

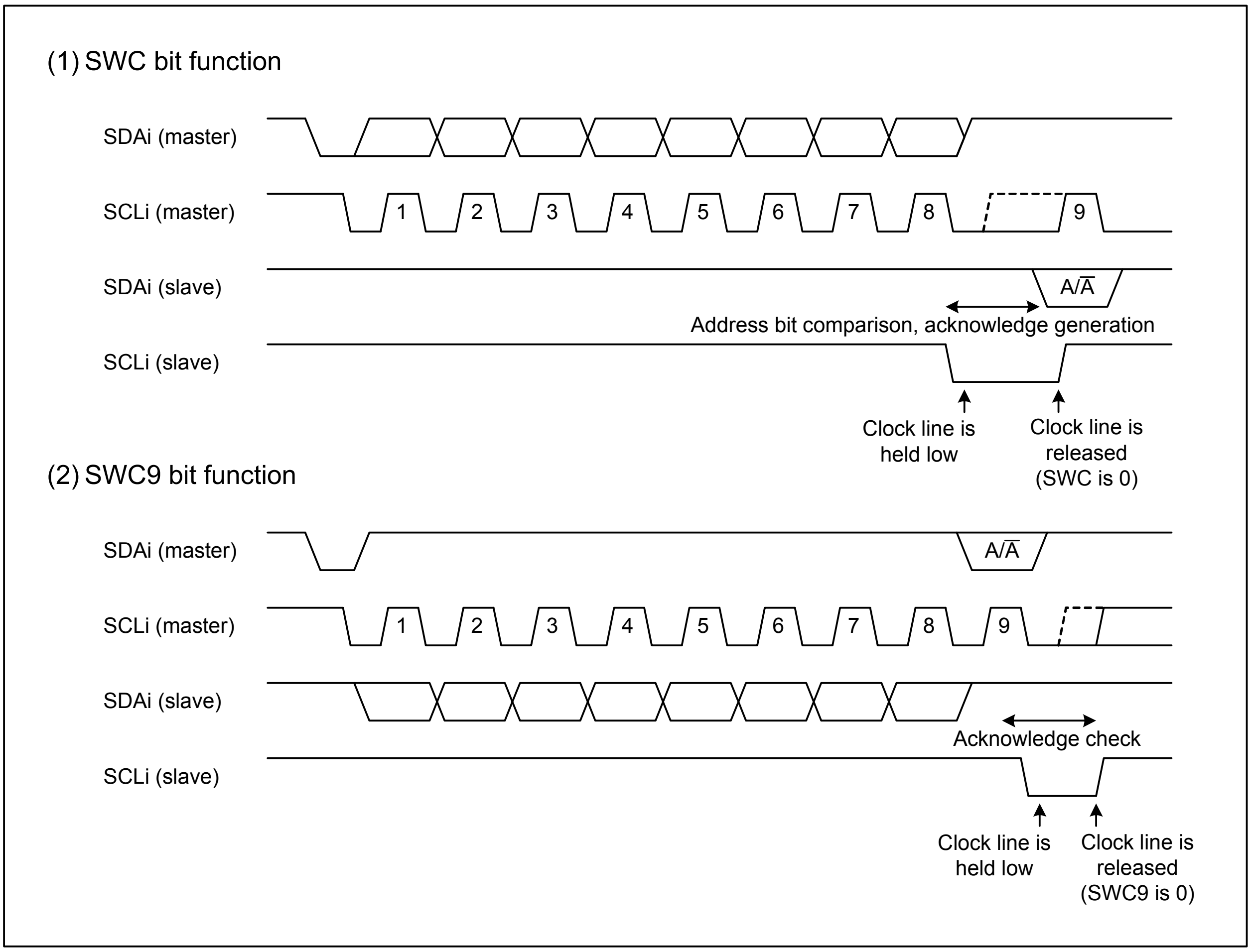


Figure 17.33 Wait-state Insertion Using the SWC or SWC9 Bit (i = 0 to 2)

The CSC bit in the UiSMR2 register is used to synchronize an internally generated clock with the clock applied to the SCLi pin. For example, if a wait-state is inserted from another device, the two clocks are not synchronized. While the CSC bit is set to 1 (clock synchronization enabled) and the internal clock is held high, when a high at the SCLi pin changes to low, the internal clock becomes low in order to reload the value of the UiBRG register and to resume counting. While the SCLi pin is held low, when the internal clock changes from low to high, the count is stopped until the SCLi pin becomes high. That is, the UARTi transmit/receive clock is the logical AND of the internal clock and the SCLi. The synchronized period starts from one clock prior to the first synchronized clock and ends when the ninth clock is completed. The CSC bit can be set to 1 only when the CKDIR bit in the UiMR register is set to 0 (internal clock selected).

The SCLHI bit in the UiSMR4 register is used to leave the SCLi pin open when another master generates a stop condition while the master is in transmit/receive operation. If the SCLHI bit is set to 1 (output stopped), the SCLi pin is open (the pin is high-impedance) when a stop condition is detected and the clock output is stopped.

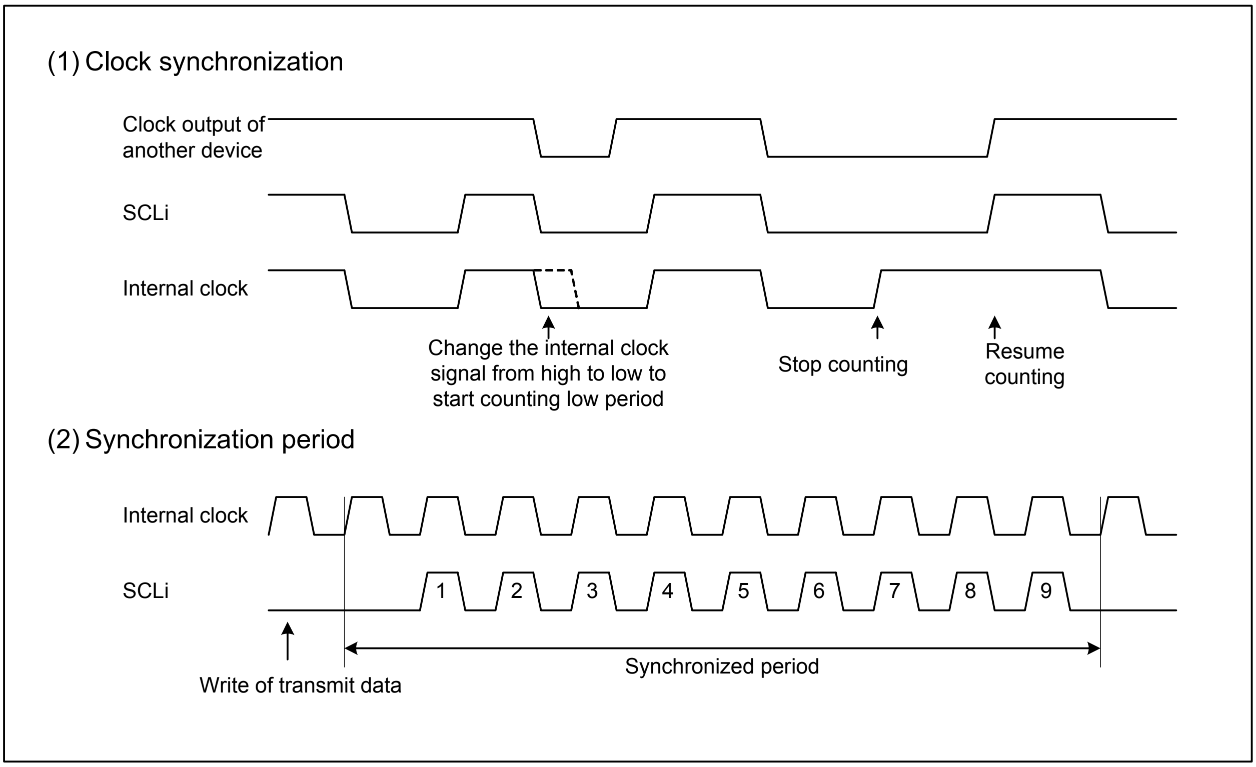


Figure 17.34 Clock Synchronization (i = 0 to 2)

17.3.5 SDA Output

Values set to bits 8 to 0 (D8 to D0) in the UiTB register (i = 0 to 2) are output starting from D7 to D0, and lastly D8, which is a bit for the acknowledge signal. When transmitting, D8 should be set to 1 to free the bus. When receiving, D8 should be set to ACK or NACK.

Bits DL2 to DL0 in the UiSMR3 register set a delay time of the SDAi on the falling edge of the SCLi. Based on the UiBRG count source, the delay time can be selected from zero cycles (no delay) or two to eight cycles.

The SDAi pin can be high-impedance at any given time once the SDHI bit in the UiSMR2 register is set to 1 (SDA output disabled). Output at the SDAi pin is low if an I/O port is selected for the SDAi and the pin is specified as the output port after selecting I^2C mode. In this case, if the SDHI bit is set to 1, the SDAi pin becomes high-impedance.

When the SDHI bit is rewritten while the SCLi pin is held high, a start condition or stop condition is generated. When it is rewritten immediately before the rising edge of SCLi, arbitration lost may be accidently detected. Therefore, the SDHI bit should be rewritten so the SDAi pin level changes while the SCLi pin is low.

17.3.6 SDA Input

When the IICM2 bit in the UiSMR2 register (i = 0 to 2) is set to 0, the first eight bits of received data (D7 to D0) are stored into bits 7 to 0 in the UiRB register and the ninth bit (ACK/NACK) is stored into bit 8.

When the IICM2 bit is set to 1, the first seven bits of received data (D7 to D1) are stored into bits 6 to 0 in the UiRB register and eighth bit (D0) is stored into bit 8.

If the IICM2 bit is set to 1 and the CKPH bit in the UiSMR3 register is set to 1 (clock delayed), the same data that is set when the IICM2 bit is 0 can be read. To read this data, read the UiRB register after data in the ninth bit is latched on the rising edge of the SCLi.

17.3.7 Acknowledge

When data is to be received in master mode, ACK is output after eight bits are received by setting the UiTB register to 00FFh as dummy data. When the STSPSEL bit in the UiSMR4 register (i = 0 to 2) is set to 0 (serial I/O circuit selected) and the ACKC bit is set to 1 (ACK data output), the value of the ACKD bit is output at the SDAi pin.

If the IICM2 bit is set to 0, NACK interrupt request is generated when the SDAi pin is held high on the rising edge of the ninth bit of the SCLi. ACK interrupt request is generated when the SDAi pin is held low.

If the DMA request source is “UARTi receive interrupt request or ACK interrupt request”, the DMA transfer is activated when an ACK is detected.

17.3.8 Initialization of Transmit/Receive Operation

When the CKDIR bit in the UiMR register (i = 0 to 2) is set to 1 (external clock), the STC bit in the UiSMR2 register is set to 1 (the circuit is initialized), and a start condition is detected, the following three operations are performed:

- The transmit register is reset and the content of the UiTB register is transferred to the transmit register. New data transmission starts on the falling edge of the first bit of the next SCLi as transmit clock. The content of the transmit register before the reset is output at the SDAi pin in the period from the falling edge of the SCLi until the first data output.
- The receive register is reset and the new data reception starts on the falling edge of the first bit of the next SCLi.
- The SWC bit in the UiSMR2 register is set to 1 (hold the SCLi pin low after the eighth bit is received).

When UARTi transmission/reception is started with this function, the TI bit in the UiC1 register does not change.

17.4 Special Mode 2

Special mode 2 enables serial communication between one or multiple masters and multiple slaves. The $\overline{\text{SSi}}$ input pin (i = 0 to 2) controls serial bus communication. Table 17.14 lists specifications of special mode 2.

Table 17.14 Special Mode 2 Specifications

Item	Specification
Data format	8-bit character length
Transmit/receive clock	• The CKDIR bit in the UiMR register (i = 0 to 2) is set to 0 (internal clock): $\frac{fx}{2(m+1)}$ fx = f1, f8, f2n m: UiBRG register setting value, 00h to FFh • The CKDIR bit is set to 1 (external clock): input to the CLKi pin
Transmit/receive control	SS function
Transmit start conditions	The conditions for starting data transmission are as follows (1): • The TE bit in the UiC1 register is set to 1 (transmission enabled) • The TI bit in the UiC1 register is set to 0 (data held in the UiTB register)
Receive start conditions	The conditions for starting data reception are as follows (1): • The RE bit in the UiC1 register is set to 1 (reception enabled) • The TE bit in the UiC1 register is set to 1 (transmission enabled) • The TI bit in the UiC1 register is set to 0 (data held in the UiTB register)
Interrupt request generating timing	In transmit interrupt, one of the following conditions is selected to set the UiIRS bit in registers U0C1 to U2C1: • The UiIRS bit is set to 0 (transmit buffer in the UiTB register is empty): when data is transferred from the UiTB register to the UARTi transmit register (when the transmission has started) • The UiIRS bit is set to 1 (transmission is completed): when data transmission from the UARTi transmit register is completed In receive interrupt, • When data is transferred from the UARTi receive register to the UiRB register (when the reception is completed)
Error detection	Overrun error (2) This error occurs when the seventh bit of the next data has been received before reading the UiRB register
Selectable functions	• CLK polarity Rising or falling edge of the transmit/receive clock for transfer data input and output • Bit order selection LSB first or MSB first • Continuous receive mode Data reception is enabled by a read access to the UiRB register • Serial data logic inversion This function logically inverses transmit/receive data • Clock phase selection One of four combinations of transmit/receive clock polarity and phases • $\overline{\text{SSi}}$ input pin function Output pin can be high-impedance when the $\overline{\text{SSi}}$ pin is high

Notes:

1. When selecting an external clock, the following preconditions should be met:
   - The CLKi pin is held high when the CKPOL bit in the UiC0 register is set to 0 (transmit data output on the falling edge of the transmit/receive clock and receive data input on the rising edge).
   - The CLKi pin is held low when the CKPOL bit is set to 1 (transmit data output on the rising edge of the transmit/receive clock and receive data input on the falling edge).
2. If an overrun error occurs, the UiRB register is undefined. The IR bit in the SiRIC register does not change to 1 (interrupt requested).

Table 17.15 lists register settings in special mode 2.

Table 17.15 Register Settings in Special Mode 2 (i = 0 to 2)

Register	Bits	Function
UiMR	7 to 4	Set the bits to 0000b
	CKDIR	Set the bit to 0 in master mode and set it to 1 in slave mode
	SMD2 to SMD0	Set the bits to 001b
UiC0	UFORM	Select either LSB first or MSB first
	CKPOL	Clock phase can be set by the combination of bits CKPOL and CKPH in the UiSMR3 register
	NCH	Select an output mode for the TXDi pin
	CRD	Set the bit to 1
	TXEPT	Transmit register empty flag
	2	Set the bit to 0
	CLK1 and CLK0	Select a count source for the UiBRG register
UiC1	7 and 6	Set the bits to 00b
	UiRRM	Set the bit to 1 to use continuous receive mode
	UiIRS	Select a source for UARTi transmit interrupt
	RI	Receive complete flag
	RE	Set the bit to 1 to enable data reception
	TI	Transmit buffer empty flag
	TE	Set the bit to 1 to enable data transmission/reception
UiSMR	7 to 0	Set the bits to 00h
UiSMR2	7 to 0	Set the bits to 00h
UiSMR3	7 to 5	Set the bits to 000b
	ERR	Mode fault flag
	NODC	Set the bit to 0
	DINC	Set to 0 in master mode and set to 1 in slave mode
	CKPH	Clock phase can be set by the combination of bits CKPH and CKPOL in the UiC0 register
	SSE	Set the bit to 1
UiSMR4	7 to 0	Set the bits to 00h
UiBRG	7 to 0	Set the bit rate
UiTB	7 to 0	Set the data to be transmitted
UiRB	OER	Overrun error flag
	7 to 0	Received data is read

17.4.1 $\overline{\text{SSi}}$ Input Pin Function (i = 0 to 2)

Special mode 2 is selected by setting the SSE bit in the UiSMR3 register to 1 (SS function enabled). The $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$/$\overline{\text{SSi}}$ pin functions as $\overline{\text{SSi}}$ input.
The DINC bit in the UiSMR3 register determines which MCU performs as a master or slave.
When multiple MCUs perform as masters (multi-master system), the $\overline{\text{SSi}}$ pin setting determines which master MCU is active and when.

17.4.1.1 SS Function in Slave Mode

When the DINC bit is set to 1 (slave mode selected) while input at the $\overline{\text{SSi}}$ pin is high, the STXDi pin becomes high-impedance and the clock input at the CLKi pin is ignored. When input at the $\overline{\text{SSi}}$ pin is low, the clock input is valid and serial data is output from the STXDi pin to enable serial communication.

17.4.1.2 SS Function in Master Mode

When the DINC bit is set to 0 (master mode selected) while input at the $\overline{\text{SSi}}$ pin is high, which means there is the only one master MCU or no other master MCU is active, the MCU as master starts communication. The master provides the transmit/receive clock output at the CLKi pin. When input at the $\overline{\text{SSi}}$ pin is low, which means that there are more masters, pins TXDi and CLKi become high-impedance. This error is called a mode fault. It can be verified using the ERR bit in the UiSMR3 register. The ongoing data transmission/reception does not stop even if a mode fault occurs. To stop transmission/reception, bits SMD2 to SMD0 in the UiMR register should be set to 000b (serial interface disabled).

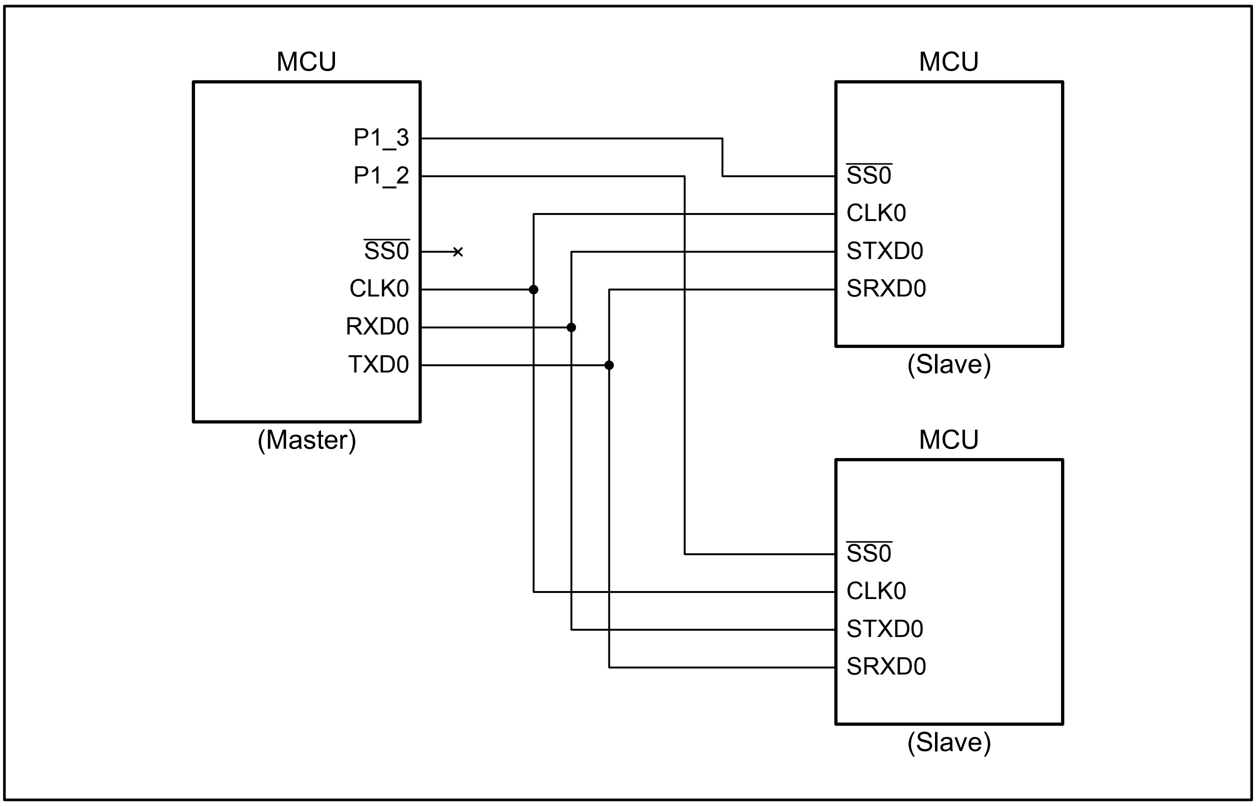


Figure 17.35 Serial Bus Communication Control with the $\overline{\text{SSi}}$ Pin

17.4.2 Clock Phase Setting

The CKPH bit in the UiSMR3 register (i = 0 to 2) and the CKPOL bit in the UiC0 register select one of four combinations of transmit/receive clock polarity and serial clock phase.
The transmit/receive clock phase and polarity should be identical for the master device and the communicating slave device.

17.4.2.1 Transmit/Receive Timing in Master Mode

When the DINC bit is set to 0 (master mode selected), the CKDIR bit in the UiMR register should be set to 0 (internal clock) to generate the clock. Figure 17.36 shows transmit/receive timing of each clock phase.

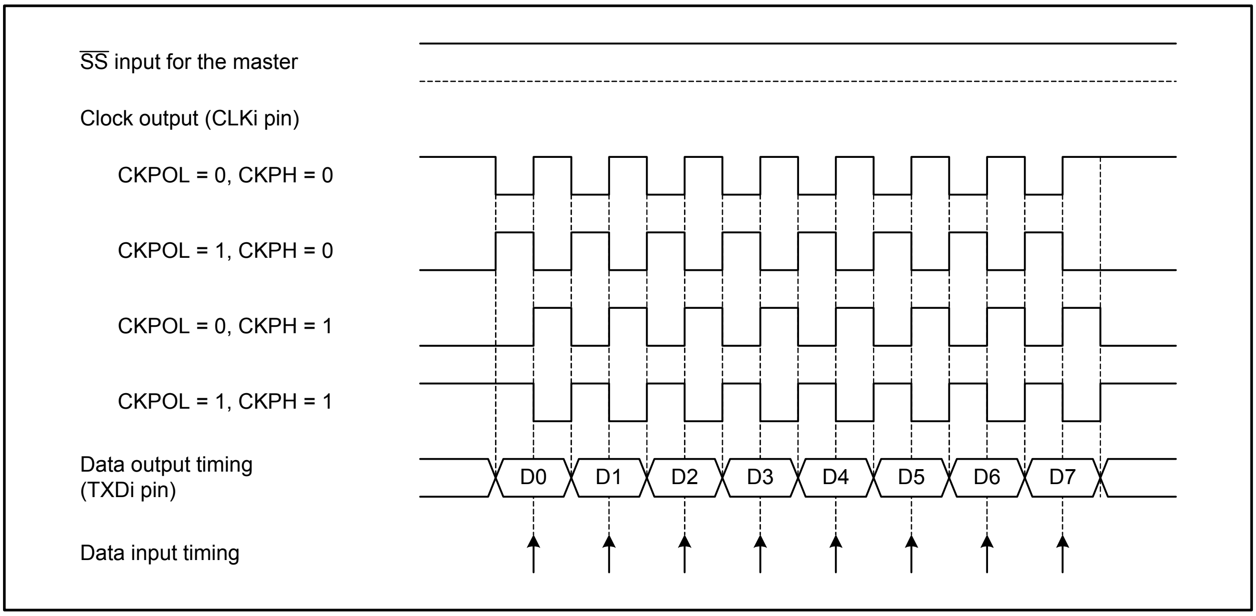


Figure 17.36 Transmit/Receive Timing in Master Mode

17.4.2.2 Transmit/Receive Timing in Slave Mode

When the DINC bit is set to 1 (slave mode selected), the CKDIR bit in the UiMR register should be set to 1 (external clock).

When the CKPH bit is set to 0 (no clock delay) while input at the $\overline{\text{SSi}}$ pin is high, the STXDi pin becomes high-impedance. When input at the $\overline{\text{SSi}}$ pin is low, the conditions for data transmission are all met, but output is undefined. Then the data transmission/reception starts synchronizing with the clock. Figure 17.37 shows the transmit/receive timing.

When the CKPH bit is set to 1 (clock delayed) while input at the $\overline{\text{SSi}}$ pin is high, the STXDi pin becomes high-impedance. When input at the $\overline{\text{SSi}}$ pin is low, the first data is output. Then the data transmission starts synchronizing with the clock. Figure 17.38 shows the transmit/receive timing.

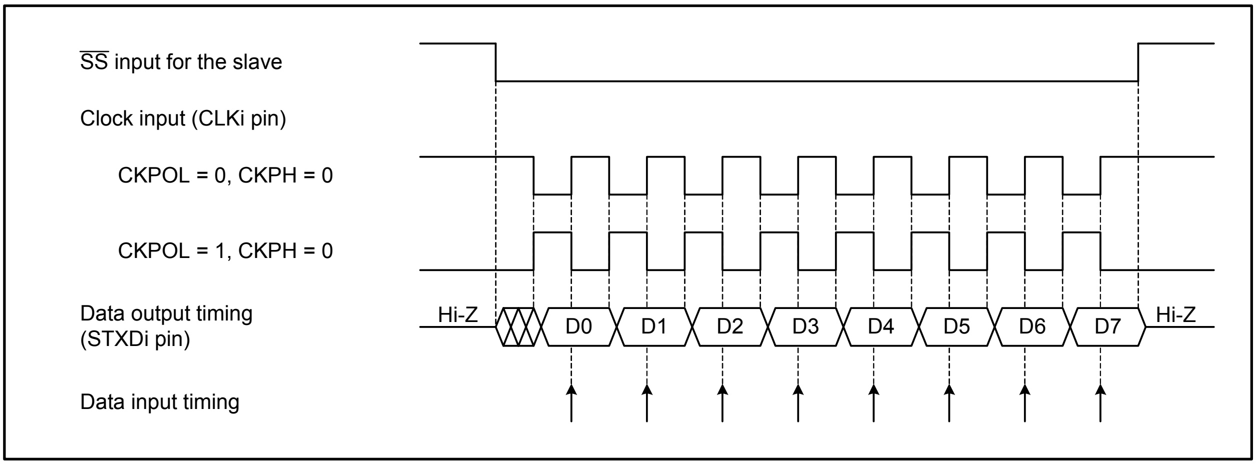


Figure 17.37 Transmit/Receive Timing in Slave Mode (CKPH = 0)

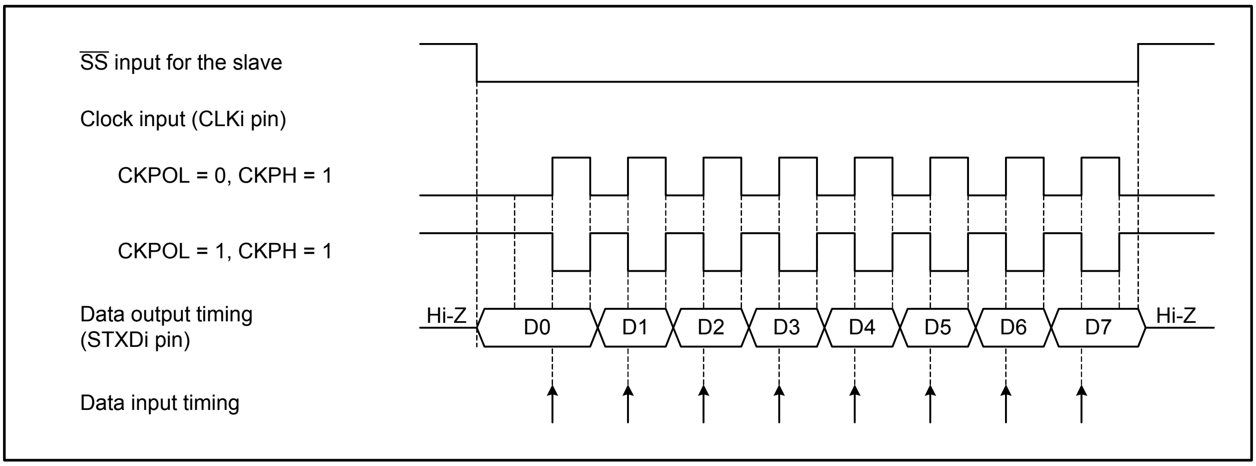


Figure 17.38 Transmit/Receive Timing in Slave Mode (CKPH = 1)

17.5 Notes on Serial Interface

17.5.1 Changing the UiBRG Register (i = 0 to 4)

- Set the UiBRG register after setting bits CLK1 and CLK0 in the UiC0 register. When these bits are changed, the UiBRG register must be set again.
- When a clock is input immediately after the UiBRG register is set to 00h, the counter may become FFh. In this case, it requires extra 256 clocks to reload 00h to the register. Once 00h is reloaded, the counter performs the operation without dividing the count source according to the setting.

17.5.2 Synchronous Serial Interface Mode

17.5.2.1 Selecting an External Clock

- If an external clock is selected, the following conditions must be met while the external clock is held high when the CKPOL bit in the UiC0 register (i = 0 to 4) is set to 0 (transmit data output on the falling edge of the transmit/receive clock and receive data input on the rising edge), or while the external clock is held low when the CKPOL bit is set to 1 (transmit data output on the rising edge of the transmit/receive clock and receive data input on the falling edge):
  - The TE bit in the UiC1 register is set to 1 (transmission enabled).
  - The RE bit in the UiC1 register is set to 1 (reception enabled). This bit setting is not required in transmit operation only.
  - The TI bit in the UiC1 register is set to 0 (data held in the UiTB register).

17.5.2.2 Receive Operation

- In synchronous serial interface mode, the transmit/receive clock is controlled by the transmit control circuit. Set UARTi-associated registers (i = 0 to 4) for a transmit operation, even if the MCU is used only for receive operation. Dummy data is output from the TXDi pin while receiving when the TXDi pin is set to output mode.

- When data is received continuously, an overrun error occurs when the RI bit in the UiC1 register is 1 (data held in the UiRB register) and the seventh bit of the next data is received in the UARTi receive shift register. Then, the OER bit in the UiRB register becomes 1 (overrun error occurred). In this case, the UiRB register becomes undefined. If an overrun error occurs, the IR bit in the SiRIC register does not change to 1.

17.5.3 Special Mode 1 (I^2C Mode)

- To generate a start condition, stop condition, or restart condition, set the STSPSEL bit in the UiSMR4 register (i = 0 to 2) to 0. Then, wait a half or more clock cycles of the transmit/receive clock to change the condition generate bits (STAREQ, RSTAREQ, or STPREQ bit) from 0 to 1.

17.5.4 Reset Procedure on Communication Error

Operations which result in communication errors such as rewriting function select registers during transmission/reception should not be performed. Follow the procedure below to reset the internal circuit once the communication error occurs in the following cases: when the operation above is performed by a receiver or transmitter or when a bit slip is caused by noise.

A. Synchronous Serial Interface Mode

(1) Set the TE bit in the UiC1 register (i = 0 to 4) to 0 (transmission disabled) and the RE bit to 0 (reception disabled).
(2) Set bits SMD2 to SMD0 in the UiMR register to 000b (serial interface disabled).
(3) Set bits SMD2 to SMD0 in the UiMR register to 001b (synchronous serial interface mode).
(4) Set the TE bit in the UiC1 register to 1 (transmission enabled) and the RE bit to 1 (reception enabled) if necessary.

B. UART Mode

(1) Set the TE bit in the UiC1 register to 0 (transmission disabled) and the RE bit to 0 (reception disabled).
(2) Set bits SMD2 to SMD0 in the UiMR register to 000b (serial interface disabled).
(3) Set bits SMD2 to SMD0 in the UiMR register to 100b (UART mode, 7-bit character length), 101b (UART mode, 8-bit character length), or 110b (UART mode, 9-bit character length).
(4) Set the TE bit in the UiC1 register to 1 (transmission enabled) and the RE bit to 1 (reception enabled) if necessary.

18. A/D Converter

The A/D converter consists of one 10-bit successive approximation A/D converter with a capacitive coupling amplifier.

A/D converted results are stored in the A/D registers corresponding to selected pins. Results are stored in the AD00 register only when the DMAC operating mode is enabled.

When the A/D converter is not in use, power consumption can be reduced by setting the VCUT bit in the AD0CON1 register to 0 (VREF disconnected). This bit setting enables the power supply from the VREF pin to the resistor ladder to stop.

Table 18.1 lists specifications of the A/D converter. Figure 18.1 shows a block diagram of the A/D converter. Figure 18.2 to Figure 18.8 show registers associated with the A/D converter.

Table 18.1 A/D Converter Specifications

Item	Specification
A/D conversion method	Capacitance-based successive approximation
Analog input voltage (1)	0 V to AVCC (VCC)
Operating clock, ϕAD (2)	fAD, fAD divided by 2, fAD divided by 3, fAD divided by 4, fAD divided by 6, or fAD divided by 8
Resolution	8 bits or 10 bits
Operating modes	One-shot mode, repeat mode, single sweep mode, repeat sweet mode 0, repeat sweep mode 1, multi-port single sweep mode, multi-port repeat sweep mode 0, and self test mode
Analog input pins (3)	26 8 pins each for AN, AN0, and AN2 2 function-extended input pins (ANEX0 and ANEX1)
A/D conversion start conditions	• Software trigger The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program • External trigger (retrigger is enabled) An input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program • Hardware trigger (retrigger is enabled) • Generation of a timer B2 interrupt request which has passed through the circuit to set an interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program • Generation of a timer A0 interrupt request after the ADST bit is set to 1
Conversion rates per pin	• Without sample and hold function 49 ϕAD cycles at 8-bit resolution 59 ϕAD cycles at 10-bit resolution including 2 ϕAD cycles for sampling time • With sample and hold function 28 ϕAD cycles at 8-bit resolution 33 ϕAD cycles at 10-bit resolution including 3 ϕAD cycles for sampling time

Notes:

1. The analog input voltage is not dependent on whether the sample and hold function is enabled or disabled.
2. The ϕAD frequency should be as follows:
   - 16 MHz or below
   - When not using the sample and hold function, 250 kHz or above
   - When using the sample and hold function, 1 MHz or above
3. When AVCC = VREF = VCC, A/D input voltage for pins AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0, and ANEX1 should be VCC or lower.

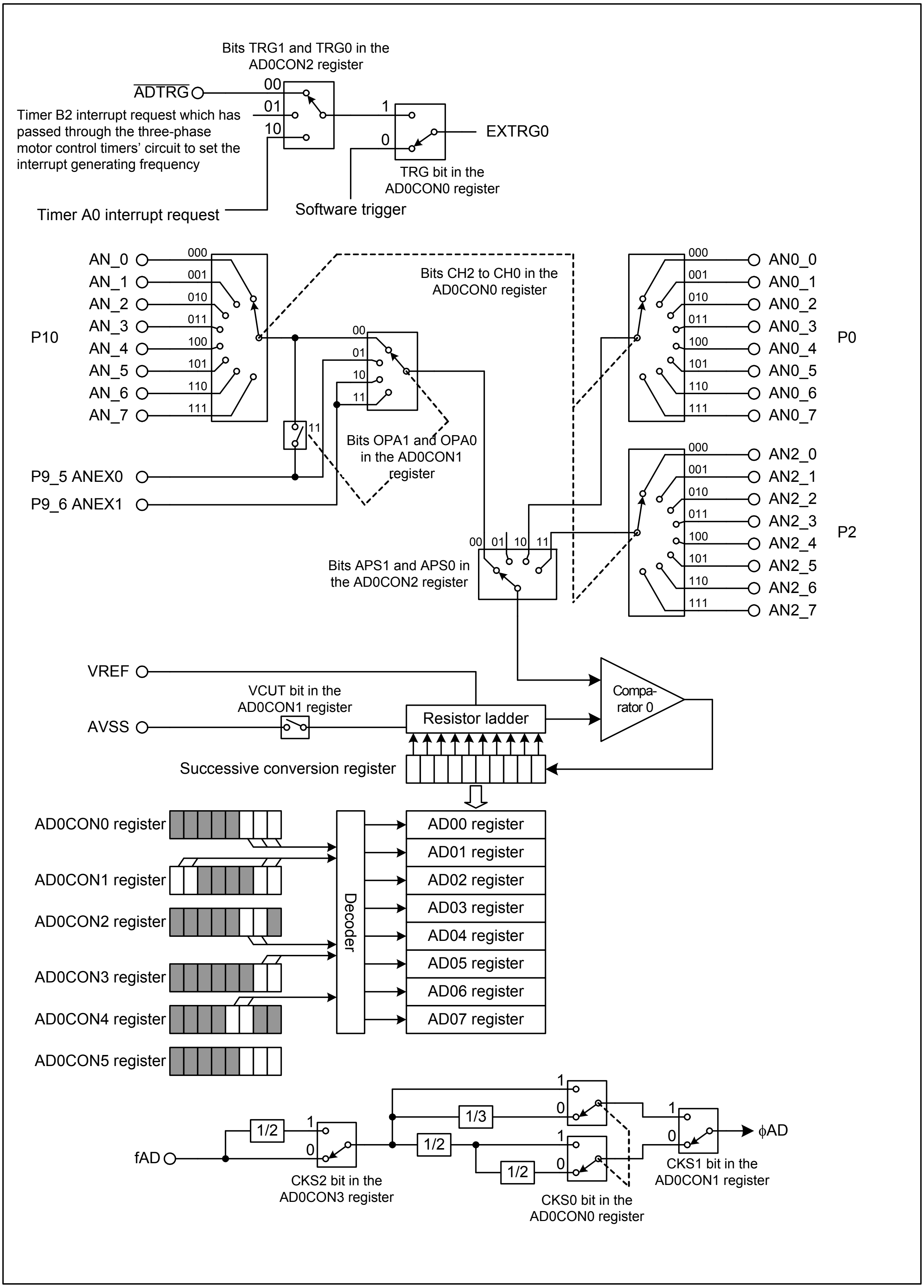
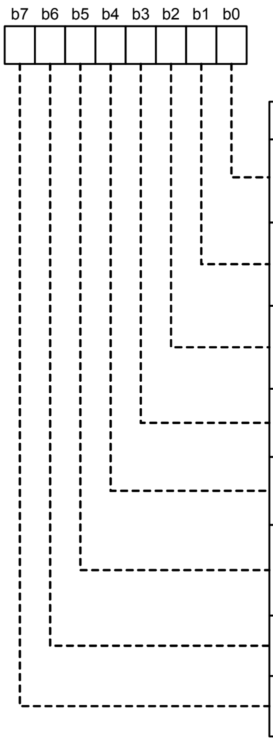


Figure 18.1 A/D Converter Block Diagram

A/D0 Control Register 0 (1)



Symbol	Address	Reset Value
AD0CON0	0396h	0000 0000b

Bit Symbol	Bit Name	Function	RW
CH0	Analog Input Pin Select Bit (2, 3, 4)	b2 b1 b0 0 0 0 : ANi_0 0 0 1 : ANi_1 0 1 0 : ANi_2 0 1 1 : ANi_3 1 0 0 : ANi_4 1 0 1 : ANi_5 1 1 0 : ANi_6 1 1 1 : ANi_7 (i = no value, 0, 2)	RW
CH1			RW
CH2			RW
MD0	A/D Operating Mode Select Bit 0 (2, 5, 6)	b4 b3 0 0 : One-shot mode 0 1 : Repeat mode 1 0 : Single sweep mode 1 1 : Repeat sweep mode 0 or 1	RW
MD1			RW
TRG	Trigger Select Bit	0: Software trigger 1: External trigger or hardware trigger (7)	RW
ADST	A/D Conversion Start Bit	0: A/D conversion stopped 1: A/D conversion started (7)	RW
CKS0	Frequency Select Bit	(Note 8)	RW

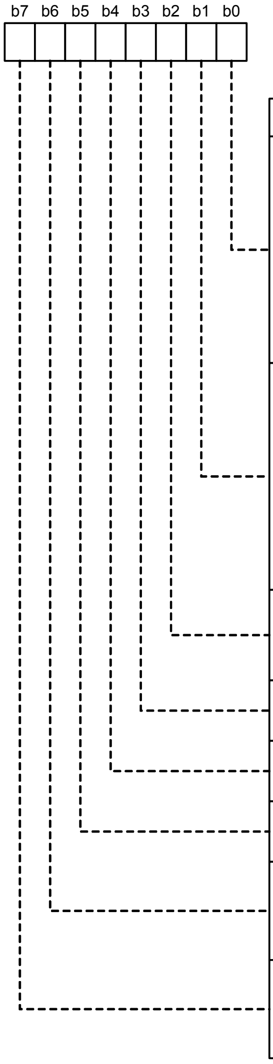
Notes:

1. When this register is rewritten during an A/D conversion, the converted result is undefined.
2. Analog input pins should be reset after the A/D operating mode is changed.
3. This bit setting is enabled in one-shot mode or repeat mode.
4. The AN, AN0, or AN2 port should be selected by using bits APS1 and APS0 in the AD0CON2 register.
5. When the MSS bit in the AD0CON3 register is set to 1 (multi-port sweep mode enabled), bits MD1 and MD0 should be set to 10b for multi-port single sweep mode and 11b for multi-port repeat sweep mode 0.
6. When the MSS bit in the AD0CON3 register is set to 1, bits MD1 and MD0 should be set to 10b or 11b.
7. To use the external trigger or the hardware trigger, the source of the trigger should be selected by setting bits TRG1 and TRG0 in the AD0CON2 register, the TRG bit should be set to 1, then the ADST bit should be set to 1.
8. The ϕAD frequency should be 16 MHz or below. It is selected from the combination of bits CKS0, CKS1, and CKS2 shown as below:

CKS2 Bit in the AD0CON3 Register	CKS0 Bit in the AD0CON0 Register	CKS1 Bit in the AD0CON1 Register	ϕAD
0	0	0	fAD divided by 4
		1	fAD divided by 3
	1	0	fAD divided by 2
		1	fAD
1	0	0	fAD divided by 8
		1	fAD divided by 6

Figure 18.2 AD0CON0 Register

A/D0 Control Register 1 (1)



Symbol	Address	Reset Value
AD0CON1	0397h	0000 0000b

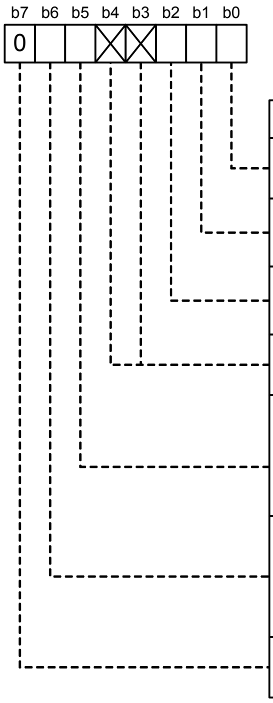
Bit Symbol	Bit Name	Function	RW
SCAN0	A/D Sweep Pin Select Bit (2, 3)	In single sweep mode or repeat sweep mode 0 b1 b0 0 0 : ANi_0, ANi_1 0 1 : ANi_0 to ANi_3 1 0 : ANi_0 to ANi_5 1 1 : ANi_0 to ANi_7 In repeat sweep mode 1 (4) b1 b0 0 0 : ANi_0 0 1 : ANi_0, ANi_1 1 0 : ANi_0 to ANi_2 1 1 : ANi_0 to ANi_3 In multi-port single sweep mode or multi-port repeat sweep mode 0 b1 b0 1 1 : ANi_0 to ANi_7 (5) (i = no value, 0, 2)	RW
SCAN1			RW
MD2	A/D Operating Mode Select Bit 1	0: Mode other than repeat sweep mode 1 1: Repeat sweep mode 1(6)	RW
BITS	8/10-bit Mode Select Bit	0: 8-bit mode 1: 10-bit mode	RW
CKS1	Frequency Select Bit	(Note 7)	RW
VCUT	VREF Connection Bit (8)	0: VREF disconnected (9) 1: VREF connected (10)	RW
OPA0	External Op-Amp Connect Mode Bit (11, 12)	b7 b6 0 0 : No use of ANEX0 or ANEX1 pin (convert input at pins ANi_0 to ANi_7) 0 1 : Convert input at the ANEX0 pin 1 0 : Convert input at the ANEX1 pin 1 1 : External op-amp connected	RW
OPA1			RW

Notes:

1. When this register is rewritten during A/D conversion, the converted result is undefined.
2. This bit setting is enabled in single sweep mode, repeat sweep mode 0, repeat sweep mode 1, multi-port single sweep mode, or multi-port repeat sweep mode 0.
3. The AN, AN0, or AN2 port should be selected by using bits APS1 and APS0 in the AD0CON2 register.
4. These pins are commonly used in A/D conversion when the MD2 bit is set to 1.
5. Set bits SCAN1 and SCAN0 to 11b in multi-port single sweep mode or multi-port repeat sweep mode 0.
6. When the MSS bit in the AD0CON3 register is set to 1 (multi-port sweep mode enabled), the MD2 bit should be set to 0.
7. Refer to the note on the CKS0 bit in the AD0CON0 register.
8. This bit controls the reference voltage to the A/D converter. It does not affect VREF performance of the D/A converter.
9. The VCUT bit should not be set to 0 during A/D conversion.
10. When the VCUT bit is changed from 0 to 1, A/D conversion should be started after 1 μs or more.
11. Bits OPA1 and OPA0 can be set to 01b or 10b only in one-shot mode or repeat mode. They should be set to 00b or 11b in other modes.
12. When the MSS bit in the AD0CON3 register is set to 1 (multi-port sweep mode enabled), bits OPA1 and OPA0 should be set to 00b.

Figure 18.3 AD0CON1 Register

A/D0 Control Register 2 (1)



Symbol	Address	Reset Value
AD0CON2	0394h	X00X X000b

Bit Symbol	Bit Name	Function	RW
SMP	A/D Conversion Method Select Bit	0: Without sample and hold function 1: With sample and hold function	RW
APS0	Analog Input Port Select Bit (2)	b2 b1 0 0 : AN_0 to AN_7, ANEX0, ANEX1 0 1 : Multi-port sweep mode 1 0 : AN0_0 to AN0_7 1 1 : AN2_0 to AN2_7	RW
APS1			RW
— (b4-b3)	No register bits; should be written with 0 and read as undefined value		—
TRG0	External Trigger Request Source Select Bit	b6 b5 0 0 : Select $\overline{\text{ADTRG}}$ pin 0 1 : Select a timer B2 interrupt request (after counting the ICTB2 register) in the three-phase motor control timers 1 0 : Select a timer A0 interrupt request 1 1 : Do not use this combination	RW
TRG1			RW
— (b7)	Reserved	Should be written with 0 and read as undefined value	RW

Notes:
1. When this register is rewritten during A/D conversion, the converted result is undefined.
2. Set bits APS1 and APS0 to 01b when the MSS bit in the AD0CON3 register is set to 1 (multi-port sweep mode enabled).

Figure 18.4 AD0CON2 Register

A/D0 Control Register 3 (1, 2)

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0					

Symbol	Address	Reset Value
AD0CON3	0395h	XXXX X000b

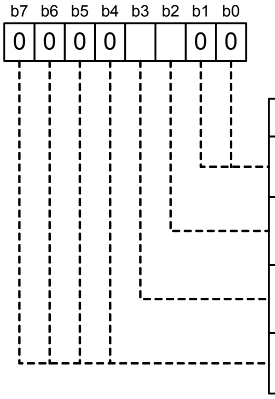
Bit Symbol	Bit Name	Function	RW
DUS	DMAC Operating Mode Select Bit (3)	0: DMAC operating mode disabled 1: DMAC operating mode enabled (4, 5)	RW
MSS	Multi-port Sweep Mode Select Bit	0: Multi-port sweep mode disabled 1: Multi-port sweep mode enabled (3, 6)	RW
CKS2	Frequency Select Bit	(Note 7)	RW
MSF0	Multi-port Sweep Status Flag (8)	b4 b3 0 0 : AN_0 to AN_7 0 1 : Reserved 1 0 : AN0_0 to AN0_7 1 1 : AN2_0 to AN2_7	RO
MSF1			RO
— (b7-b5)	Reserved	Should be written with 0 and read as undefined value	RW

Notes:

1. When this register is rewritten during A/D conversion, the converted result is undefined.
2. This register may be read incorrectly during A/D conversion. It should be read or written after the A/D converter stops operating.
3. To set the MSS bit to 1, the DUS bit should be also set to 1.
4. When the DUS bit is set to 1, all A/D converted results are stored into the AD00 register.
5. Configure DMAC when it is used to transfer converted results.
6. To set the MSS bit to 1:
   - Set the MD2 bit in the AD0CON1 register to 0 (mode other than repeat sweep mode 1).
   - Set bits APS1 and APS0 in the AD0CON2 register to 01b (multi-port sweep mode).
   - Set bits OPA1 and OPA0 in the AD0CON1 register to 00b (no use of ANEX0 or ANEX1).
7. Refer to the note for the CKS0 bit in the AD0CON0 register.
8. This bit setting is enabled when the MSS bit is set to 1. The read value is undefined when the MSS bit is set to 0.

Figure 18.5 AD0CON3 Register

A/D0 Control Register 4 (1, 2)



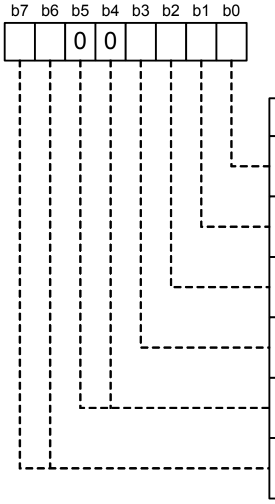
Symbol	Address	Reset Value
AD0CON4	0392h	XXXX 00XXb

Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved	Should be written with 0 and read as undefined value	RW
MPS10	Multi-port Sweep Port Select Bit	b3 b2 0 0 : (Note 3) 0 1 : Do not use this combination 1 0 : AN_0 to AN_7, AN0_0 to AN0_7 1 1 : AN_0 to AN_7, AN2_0 to AN2_7	RW
MPS11			RW
— (b7-b4)	Reserved	Should be written with 0 and read as undefined value	RW

Notes:
1. When this register is rewritten during A/D conversion, the converted result is undefined.
2. A 8-bit read access to this register should be performed.
3. When the MSS bit in the AD0CON3 register is set to 0 (multi-port sweep mode disabled), set bits MSP11 and MPS10 to 00b. When it is set to 1 (multi-port sweep mode enabled), set them to any value other than 00b.

Figure 18.6 AD0CON4 Register

A/D0 Control Register 5 (1, 2, 3)



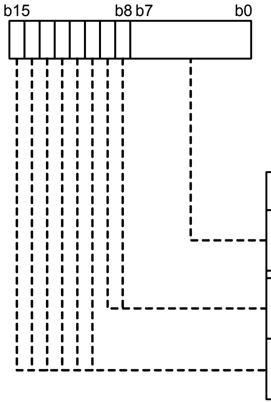
Symbol	Address	Reset Value
AD0CON5	0393h	0000 0000b

Bit Symbol	Bit Name	Function	RW
ASTEN	Self Test and Open-circuit Detection Assist Setting Bit	0: Assist disabled 1: Assist enabled (4)	RW
ASTSEL	Self Test/Open-circuit Detection Assist Select Bit	0: Open-circuit detection assist 1: Self test assist	RW
ASTMTD	Assist Mode Select Bit	0: Pull-up 1: Pull-down	RW
— (b3)	Reserved	If necessary, set to 0	RO
— (b5-b4)	Reserved	Should be written with 0	RW
— (b7-b6)	Reserved	If necessary, set to 0	RO

Notes:
1. When this register is rewritten during A/D conversion, the converted result is undefined.
2. A 8-bit read access to this register should be performed.
3. Appropriate evaluation for each system should be performed since A/D converted result for open-circuit detection varies with external circuits.
4. To use the self test assist or open-circuit detection assist, the SMP bit in the AD0CON2 register should be set to 1 (with sample and hold function).

Figure 18.7 AD0CON5 Register

A/D0 Register i (i = 0 to 7) (1 to 4)



Symbol	Address	Reset Value
AD00, AD01	0381h-0380h, 0383h-0382h	0000 0000 XXXX XXXXb
AD02, AD03	0385h-0384h, 0387h-0386h	0000 0000 XXXX XXXXb
AD04, AD05	0389h-0388h, 038Bh-038Ah	0000 0000 XXXX XXXXb
AD06, AD07	038Dh-038Ch, 038Fh-038Eh	0000 0000 XXXX XXXXb

Bit Symbol	Function	RW
— (b7-b0)	The lower byte in an A/D converted result	RO
— (b9-b8)	In 10-bit mode: 2 upper bits in an A/D converted result In 8-bit mode: These bits are read as 0	RO
— (b15-b10)	These bits are read as 0	RO

Notes:

1. If this register is read by a program while the DMAC is configured to transfer converted results, the value is undefined.
2. The register value written while the A/D converter stops operating is undefined.
3. Only the AD00 register is available when the DUS bit in the AD0CON3 register is 1 (DMAC operating mode enabled). Other registers are undefined.
4. When a converted result is transferred by DMAC at 10-bit mode, the DMAC should be set for a 16-bit transfer.

Figure 18.8 Registers AD00 to AD07

18.1 Mode Descriptions

18.1.1 One-shot Mode

In one-shot mode, the analog voltage applied to a selected pin is converted into a digital code only once. Table 18.2 lists specifications of one-shot mode.

Table 18.2 One-shot Mode Specifications

Item	Specification
Function	Converts the analog voltage applied to a pin into a digital code only once. The pin is selected by setting bits CH2 to CH0 in the AD0CON0 register, bits OPA1 and OPA0 in the AD0CON1 register, and bits APS1 and APS0 in the AD0CON2 register
Start conditions	When the TRG bit in the AD0CON0 register is 0 (software trigger) The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program. When the TRG bit is 1 (external trigger or hardware trigger) Set bits TRG1 and TRG0 in the AD0CON2 register to select external trigger request source. • When 00b is selected, an input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program. • When 01b is selected, generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program. • When 10b is selected, generation of a timer A0 interrupt request after the ADST bit is set to 1
Stop conditions	• A/D conversion is completed (the ADST bit is set to 0 when the software trigger is selected) • The ADST bit is set to 0 (A/D conversion stopped) by a program
Interrupt request generation timing	When A/D conversion is completed, an interrupt request is generated
Input pin to be selected	One pin is selected from among AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0, and ANEX1
Reading A/D converted result	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled) Read the AD0j register (j = 0 to 7) corresponding to the selected pin When the DUS bit is 1 (DMAC operating mode enabled) Configure the DMAC (refer to 12. “DMAC”). Then the A/D converted result is stored in the AD00 register after the conversion is completed. The DMAC transfers the converted result from the AD00 register to a given memory space. Do not read the AD00 register by a program

18.1.2 Repeat Mode

In repeat mode, the analog voltage applied to a selected pin is repeatedly converted into a digital code. Table 18.3 lists specifications of repeat mode.

Table 18.3 Repeat Mode Specifications

Item	Specification
Function	Converts the analog voltage input to a pin into a digital code repeatedly. The pin is selected by setting bits CH2 to CH0 in the AD0CON0 register, bits OPA1 and OPA0 in the AD0CON1 register, and bits APS1 and APS0 in the AD0CON2 register
Start conditions	When the TRG bit in the AD0CON0 register is 0 (software trigger) The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program. When the TRG bit is 1 (external trigger or hardware trigger) Set bits TRG1 and TRG0 in the AD0CON2 register to select external trigger request source. • When 00b is selected, an input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program. • When 01b is selected, generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program. • When 10b is selected, generation of a timer A0 interrupt request after the ADST bit is set to 1
Stop conditions	The ADST bit is set to 0 (A/D conversion stopped) by a program
Interrupt request generation timing	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled), no interrupt request is generated. When the DUS bit is 1 (DMAC operating mode enabled), each time A/D conversion is completed, an interrupt request is generated
Analog voltage input pins	One pin is selected from among AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0, and ANEX1
Reading A/D converted result	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled) Read the AD0j register (j = 0 to 7) corresponding to the selected pin When the DUS bit is 1 (DMAC operating mode enabled) • When the converted result is transferred by DMAC Configure the DMAC (refer to 12. “DMAC”). Then the A/D converted result is stored in the AD00 register after the conversion is completed. The DMAC transfers the converted result from the AD00 register to a given memory space. Do not read the AD00 register by a program • When the converted result is transferred by a program Read the AD00 register after the IR bit in the AD0IC register becomes 1. Set the IR bit back to 0

18.1.3 Single Sweep Mode

In single sweep mode, the analog voltage applied to selected pins is converted one-by-one into a digital code. Table 18.4 lists specifications of single sweep mode.

Table 18.4 Single Sweep Mode Specifications

Item	Specification
Function	Converts the analog voltage input to a set of pins into a digital code one-by-one. The pins are selected by setting bits SCAN1 and SCAN0 in the AD0CON1 register and bits APS1 and APS0 in the AD0CON2 register
Start conditions	When the TRG bit in the AD0CON0 register is 0 (software trigger) The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program. When the TRG bit is 1 (external trigger or hardware trigger) Set bits TRG1 and TRG0 in the AD0CON2 register to select external trigger request source. • When 00b is selected, an input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program. • When 01b is selected, generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program. • When 10b is selected, generation of a timer A0 interrupt request after the ADST bit is set to 1
Stop conditions	• A/D conversion is completed (the ADST bit is set to 0 when the software trigger is selected) • The ADST bit is set to 0 (A/D conversion stopped) by a program
Interrupt request generation timing	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled) when a sweep is completed, an interrupt request is generated. When the DUS bit is 1 (DMAC operating mode enabled), each time A/D conversion is completed, an interrupt request is generated
Analog voltage input pins	Selected from a group of 2 pins (ANi_0 and ANi_1) (i = no value, 0, 2), 4 pins (ANi_0 to ANi_3), 6 pins (ANi_0 to ANi_5), or 8 pins (ANi_0 to ANi_7)
Reading A/D converted result	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled) Read the AD0j register (j = 0 to 7) corresponding to the selected pin When the DUS bit is 1 (DMAC operating mode enabled) Configure the DMAC (refer to 12. “DMAC”). Then the A/D converted result is stored in the AD00 register after the conversion is completed. The DMAC transfers the converted result from the AD00 register to a given memory space. Do not read the AD00 register by a program

18.1.4 Repeat Sweep Mode 0

In repeat sweep mode 0, the analog voltage applied to selected pins is repeatedly converted into a digital code. Table 18.5 lists specifications of repeat sweep mode 0.

Table 18.5 Repeat Sweep Mode 0 Specifications

Item	Specification
Function	Converts the analog voltage input to a set of pins into a digital code repeatedly. The pins are selected by setting bits SCAN1 and SCAN0 in the AD0CON1 register and APS1 and APS0 in the AD0CON2 register
Start conditions	When the TRG bit in the AD0CON0 register is 0 (software trigger) The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program. When the TRG bit is 1 (external trigger or hardware trigger) Set bits TRG1 and TRG0 in the AD0CON2 register to select external trigger request source. • When 00b is selected, an input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program. • When 01b is selected, generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program. • When 10b is selected, generation of a timer A0 interrupt request after the ADST bit is set to 1
Stop conditions	The ADST bit is set to 0 (A/D conversion stopped) by a program
Interrupt request generation timing	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled), no interrupt request is generated. When the DUS bit is 1 (DMAC operating mode enabled), each time A/D conversion is completed, an interrupt request is generated
Analog voltage input pins	Selected from a group of 2 pins (ANi_0 and ANi_1) (i = no value, 0, 2), 4 pins (ANi_0 to ANi_3), 6 pins (ANi_0 to ANi_5), or 8 pins (ANi_0 to ANi_7)
Reading A/D converted result	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled) Read the AD0j register (j = 0 to 7) corresponding to the selected pin When the DUS bit is 1 (DMAC operating mode enabled) • When the converted result is transferred by DMAC Configure the DMAC (refer to 12. “DMAC”). Then the A/D converted result is stored in the AD00 register after the conversion is completed. The DMAC transfers the converted result from the AD00 register to a given memory space. Do not read the AD00 register by a program • When the converted result is transferred by a program Read the AD00 register after the IR bit in the AD0IC register becomes 1. Set the IR bit back to 0

18.1.5 Repeat Sweep Mode 1

In repeat sweep mode 1, the analog voltage applied to eight selected pins including one to four prioritized pins is repeatedly converted into a digital code. Table 18.6 lists specifications of repeat sweep mode 1.

Table 18.6 Repeat Sweep Mode 1 Specifications

Item	Specification
Function	Converts the analog voltage input to a set of eight pins into a digital code repeatedly. One to four pins are converted by priority For example, when AN_0 is prioritized, the analog voltage is converted into a digital code in the following order: AN_0→AN_1→AN_0→AN_2→AN_0→AN_3••• The eight pins are selected by setting bits SCAN1 and SCAN0 in the AD0CON1 register and bits APS1 and APS0 in the AD0CON2 register
Start conditions	When the TRG bit in the AD0CON0 register is 0 (software trigger) The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program. When the TRG bit is 1 (external trigger or hardware trigger) Set bits TRG1 and TRG0 in the AD0CON2 register to select external trigger request source. • When 00b is selected, an input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program. Retrigger is invalid. • When 01b is selected, generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program. • When 10b is selected, generation of a timer A0 interrupt request after the ADST bit is set to 1
Stop conditions	The ADST bit is set to 0 (A/D conversion stopped) by a program
Interrupt request generation timing	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled), no interrupt request is generated. When the DUS bit is 1 (DMAC operating mode enabled), each time A/D conversion is completed, an interrupt request is generated
Analog voltage input pins	8 (ANi_0 to ANi_7) (i = no value, 0, 2)
Prioritized pin(s)	Selected from a group of 1 pin (ANi_0), 2 pins (ANi_0 and ANi_1), 3 pins (ANi_0 to ANi_2), or 4 pins (ANi_0 to ANi_3)
Reading A/D converted result	When the DUS bit in the AD0CON3 register is 0 (DMAC operating mode disabled) Read the AD0j register (j = 0 to 7) corresponding to the selected pin When the DUS bit is 1 (DMAC operating mode enabled) • When the converted result is transferred by DMAC Configure the DMAC (refer to 12. “DMAC”). Then the A/D converted result is stored in the AD00 register after the conversion is completed. The DMAC transfers the converted result from the AD00 register to a given memory space. Do not read the AD00 register by a program • When the converted result is transferred by a program Read the AD00 register after the IR bit in the AD0IC register becomes 1. Set the IR bit back to 0

18.1.6 Multi-port Single Sweep Mode

In multi-port single sweep mode, the analog voltage applied to 16 selected pins is converted one-by-one into a digital code. The DUS bit in the AD0CON3 register should be set to 1 (DMAC operating mode enabled). Table 18.7 lists specifications of multi-port single sweep mode.

Table 18.7 Multi-port Single Sweep Mode Specifications

Item	Specification
Function	Converts the analog voltage input to a set of 16 selected pins into a digital code one-by-one in the following order: AN_0 to AN_7→ANi_0 to ANi_7 (i = 0, 2) The 16 pins are selected by setting bits MPS11 and MPS10 in the AD0CON4 register For example, when bits MPS11 and MPS10 are set to 10b (AN_0 to AN_7, AN0_0 to AN0_7), the analog voltage is converted into a digital code in the following order: AN_0→AN_1→AN_2→AN_3→AN_4→AN_5→AN_6→AN_7→AN0_0→•••→ AN0_6→AN0_7
Start conditions	When the TRG bit in the AD0CON0 register is 0 (software trigger) The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program. When the TRG bit is 1 (external trigger or hardware trigger) Set bits TRG1 and TRG0 in the AD0CON2 register to select external trigger request source. • When 00b is selected, an input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program. • When 01b is selected, generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program. • When 10b is selected, generation of a timer A0 interrupt request after the ADST bit is set to 1
Stop conditions	• A/D conversion is completed (the ADST bit is set to 0 when the software trigger is selected) • The ADST bit is set to 0 (A/D conversion stopped) by a program
Interrupt request generation timing	Every time A/D conversion is completed (set the DUS bit to 1)
Analog voltage input pins	A combination of pin group is selected from AN_0 to AN_7→AN0_0 to AN0_7, or AN_0 to AN_7→AN2_0 to AN2_7
Reading A/D converted result	Set the DUS bit to 1 and configure the DMAC (refer to 12. "DMAC"). Then the A/D converted result is stored in the AD00 register after the conversion is completed. The DMAC transfers the converted result from the AD00 register to a given memory space. Do not read the AD00 register by a program

18.1.7 Multi-port Repeat Sweep Mode 0

In multi-port repeat sweep mode 0, the analog voltage applied to 16 selected pins is repeatedly converted into a digital code. The DUS bit in the AD0CON3 register should be set to 1 (DMAC operating mode enabled). Table 18.8 lists specifications of multi-port repeat sweep mode 0.

Table 18.8 Multi-port Repeat Sweep Mode 0 Specifications

Item	Specification
Function	Converts the analog voltage input to a set of 16 selected pins into a digital code repeatedly in the following order: AN_0 to AN_7→ANi_0 to ANi_7 (i = 0, 2) The 16 pins are selected by setting bits MPS11 and MPS10 in the AD0CON4 register For example, when bits MPS11 and MPS10 are set to 10b (AN_0 to AN_7, AN0_0 to AN0_7),the analog voltage is converted into a digital code repeatedly in the following order: AN_0→AN_1→AN_2→AN_3→AN_4→AN_5→AN_6→AN_7→AN0_0→•••→ AN0_6→AN0_7
Start conditions	When the TRG bit in the AD0CON0 register is 0 (software trigger) The ADST bit in the AD0CON0 register is set to 1 (A/D conversion started) by a program. When the TRG bit is 1 (external trigger or hardware trigger) Set bits TRG1 and TRG0 in the AD0CON2 register to select external trigger request source. • When 00b is selected, an input signal at the $\overline{\text{ADTRG}}$ pin switches from high to low after the ADST bit is set to 1 by a program. • When 01b is selected, generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers after the ADST bit is set to 1 by a program. • When 10b is selected, generation of a timer A0 interrupt request after the ADST bit is set to 1
Stop conditions	The ADST bit is set to 0 (A/D conversion stopped) by a program
Interrupt request generation timing	Every time A/D conversion is completed (set the DUS bit to 1)
Analog voltage input pins	A combination of pin group is selected from AN_0 to AN_7→AN0_0 to AN0_7, or AN_0 to AN_7→AN2_0 to AN2_7
Reading A/D converted result	Set the DUS bit to 1 and configure the DMAC (refer to 12. “DMAC”). Then the A/D converted result is stored in the AD00 register after the conversion is completed. The DMAC transfers the converted result from the AD00 register to a given memory space. Do not read the AD00 register by a program

18.2 Functions

18.2.1 Resolution Selection

Resolution is selected by setting the BITS bit in the AD0CON1 register. When the BITS bit is set to 1 (10-bit precision), the A/D converted result is stored into bits 9 to 0 in the AD0i register (i = 0 to 7). When the BITS bit is set to 0 (8-bit precision), the result is stored into bits 7 to 0 in the AD0i register.

18.2.2 Sample and Hold Function

This function improves the conversion rate per pin to 28 ϕAD cycles at 8-bit resolution and 33 ϕAD cycles for 10-bit resolution. This function is available in all operating modes and is enabled by setting the SMP bit in the AD0CON2 register to 1 (with sample and hold function). Start A/D conversion after setting the SMP bit.

18.2.3 Trigger Selection

A trigger to start A/D conversion is specified by the combination of TRG bit in the AD0CON0 register and bits TRG1 and TRG0 in the AD0CON2 register. Table 18.9 lists the settings of the trigger selection.

Table 18.9 Trigger Selection Settings

Bit and Setting		Trigger
AD0CON0 register	AD0CON2 register	
TRG = 0	—	Software trigger The ADST bit in the AD0CON0 register is set to 1
TRG = 1 (1, 2)	TRG1 = 0, TRG0 = 0	External trigger Falling edge of a signal applied to the $\overline{\text{ADTRG}}$ pin
	TRG1 = 0, TRG0 = 1	Hardware trigger Generation of a timer B2 interrupt request which has passed through the circuit to set the interrupt generating frequency in the three-phase motor control timers
	TRG1 = 1, TRG0 = 0	Hardware trigger Generation of a timer A0 interrupt request

Notes:

1. A/D conversion is performed if a trigger is generated while the ADST bit is set to 1 (A/D conversion started).
2. If an external trigger or a hardware trigger is generated during A/D conversion, the A/D converter aborts the operation in progress. Then, it restarts the operation.

18.2.4 DMAC Operating Mode

DMAC operating mode can be used in all operating modes. DMAC operating mode is highly recommended when the A/D converter is in multi-port single sweep mode or multi-port repeat sweep mode 0. When the DUS bit in the AD0CON3 register is set to 1 (DMAC operating mode enabled), all A/D converted results are stored in the AD00 register. The DMAC transfers the data from the AD00 register to a given memory space every time A/D conversion is completed at a pin. 8-bit DMA transfer should be selected for 8-bit resolution. For 10-bit resolution, 16-bit DMA transfer should be selected. Refer to 12. “DMAC” for details.

18.2.5 Function-extended Analog Input Pins

In one-shot mode and repeat mode, pins ANEX0 and ANEX1 can be used as analog input pins by setting bits OPA1 and OPA0 in the AD0CON1 register (refer to Table 18.10). The A/D converted results of pins ANEX0 and ANEX1 are stored into registers AD00 and AD01, respectively. However, when the DUS bit in the AD0CON3 register is set to 1 (DMAC operating mode enabled), all results are stored into the AD00 register.

To use function-extended analog input pins, bits APS1 and APS0 in the AD0CON2 register should be set to 00b (AN0 to AN7, ANEX0, ANEX1 function as analog input ports) and the MSS bit in the AD0CON3 register to 0 (multi-port sweep mode disabled).

Table 18.10 Function-extended Analog Input Pin Settings

AD0CON1 Register		ANEX0	ANEX1
OPA1	OPA0		
0	0	Not used	Not used
0	1	Analog input	Not used
1	0	Not used	Analog input
1	1	Output to an external op-amp	Input from an external op-amp

18.2.6 External Operating Amplifier (Op-Amp) Connection Mode

In external op-amp connection mode, multiple analog inputs can be amplified by one external op-amp using function-extended analog input pins ANEX0 and ANEX1.

When bits OPA1 and OPA0 in the AD0CON1 register are set to 11b (external op-amp connected), the voltage applied to pins AN0 to AN7 is output from the ANEX0 pin. This output signal should be amplified by an external op-amp and applied to the ANEX1 pin.

The analog voltage applied to the ANEX1 pin is converted into a digital code. The converted result is stored into the corresponding AD0i register (i = 0 to 7). The conversion rate varies with the response of the external op-amp. The ANEX0 pin should not be connected to the ANEX1 pin directly.

To use external op-amp connection mode, bits APS1 and APS0 in the AD0CON2 register should be set to 00b.

Figure 18.9 shows an example of an external op-amp connection.

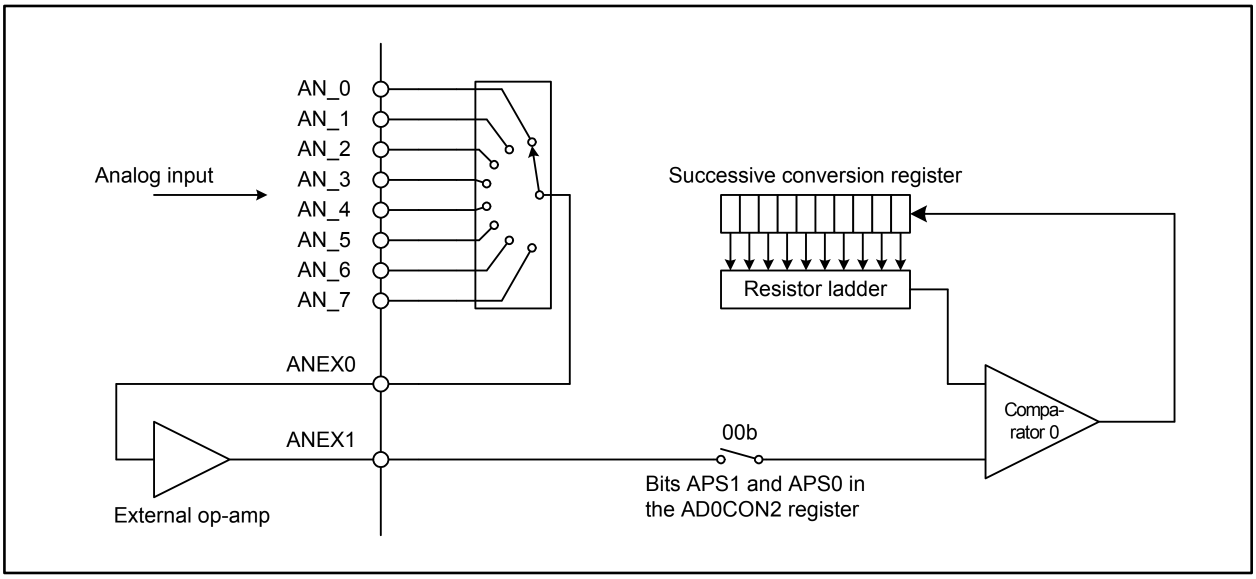


Figure 18.9 External Op-Amp Connection

18.2.7 Self Test/Open-circuit Detection Assist

This function enables the MCU to detect open-circuit in analog input pins. It also enables it to perform a self test.

Figure 18.10 shows a block diagram of open-circuit detection assist circuit.

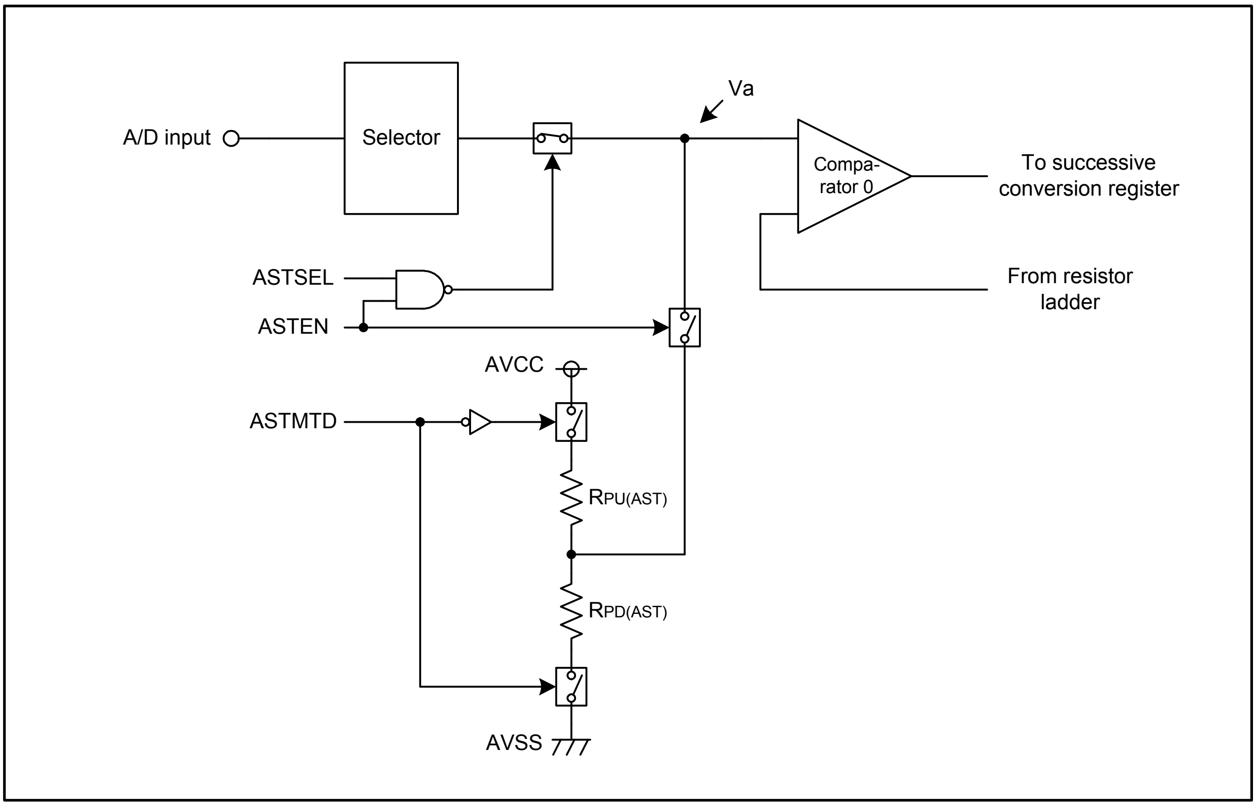


Figure 18.10 Open-circuit Detection Assist Circuit

To detect open-circuit, the ASTSEL bit in the AD0CON5 register should be set to 0 (open-circuit detection assist). Va in Figure 18.10 above is a voltage with a value between AVCC and the A/D applied voltage in the following bit settings: the ASTEN bit is 1 (assist enabled), the ASTSEL bit is 0, and the ASTMTD bit is 0 (pull-up). If the A/D input pin is open, Va is almost equal to AVCC. When the ASTEN bit is 1, the ASTSEL bit is 0, and the ASTMTD bit is 1 (pull-down), Va is between AVSS and the A/D applied voltage. If the A/D input pin is open, Va is almost AVSS. That is, the A/D input pin is considered open if the result value of A/D conversion is almost the maximum/minimum voltage.

To enable the self test function, the ASTSEL bit in the AD0CON5 register should be set to 1 (self test assist). Va is almost equal to AVCC in the bit settings as the ASTEN bit is 1, the ASTSEL bit is 1, and the ASTMTD bit is 0. When the ASTEN bit is 1, the ASTSEL bit is 1, and the ASTMTD bit is 1, Va is almost AVSS. That is, if the result value of A/D conversion is almost the maximum/minimum voltage in each bit setting, the A/D converter is considered to be functioning normally.

18.2.8 Power Saving

When the A/D converter is not in use, power consumption can be reduced by setting the VCUT bit in the AD0CON1 to 0 (VREF disconnected). With this bit setting, the reference voltage input pin (VREF) can be disconnected from the resistor ladder, which enables the power supply from the VREF to the resistor ladder to stop.

To use the A/D converter, set the VCUT bit to 1 (VREF connected) and wait at least 1 μs before setting the ADST bit in the AD0CON0 register to 1 (A/D conversion started). Bits ADST and VCUT should not be set to 1 simultaneously. The VCUT bit should not be set to 0 during A/D conversion.

The VCUT bit does not affect VREF performance of the D/A converter (refer to Figure 18.11).

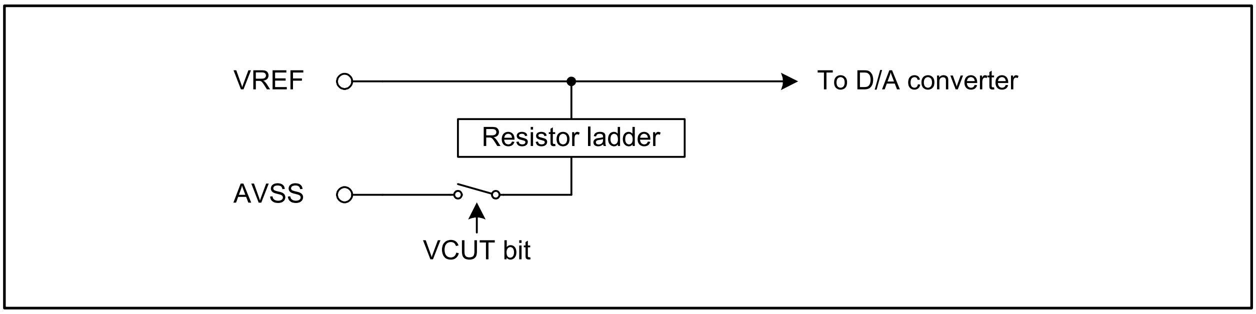


Figure 18.11 Power Supply by VCUT Bit

18.2.9 Output Impedance of Sensor Equivalent Circuit under A/D Conversion

Figure 18.12 shows an analog input pin and external sensor equivalent circuit.

To perform A/D conversion correctly, the internal capacitor (C) charging, shown in Figure 18.12, should be completed within the specified period. This period, called the sampling time, is 2 ϕAD cycles for conversion without the sample and hold function and 3 ϕAD cycles for conversion with this function.

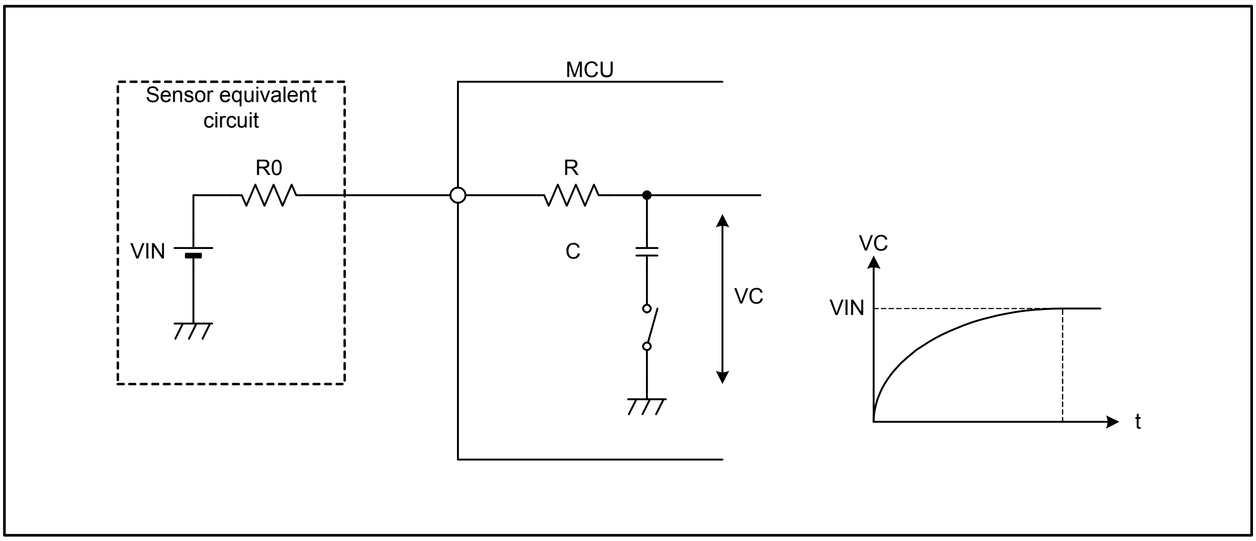


Figure 18.12 Analog Input Pin and External Sensor Equivalent Circuitry

The voltage between pins (VC) is expressed as follows:

$$VC = VIN\left\{1 - e^{-\frac{t}{C(R0+R)}}\right\}$$

When t = T and the precision (error) is x or less,

$$VC = VIN - \frac{x}{y}VIN = VIN\left(1 - \frac{x}{y}\right)$$

Thus, output impedance of the sensor equivalent circuit (R0) is determined by the following formulas:

$$e^{-\frac{T}{C(R0+R)}} = \frac{x}{y}$$

$$-\frac{T}{C(R0+R)} = \ln\frac{x}{y}$$

$$R0 = -\frac{T}{C\ln\frac{x}{y}} - R$$

where:
T[s] = Sampling time
R0[Ω] = Output impedance of the sensor equivalent circuit
VC = Potential difference between edges of capacitor C
R[Ω] = Internal resistance of the MCU
x[LSB] = Precision (error) of the A/D converter
y[step] = Resolution of the A/D converter (1024 steps at 10-bit mode, 256 steps at 8-bit mode)

When ϕAD = 10 MHz, the A/D conversion mode is 10-bit resolution with the sample and hold function, the output impedance (R0) with the precision (error) of 0.1 LSB or less is determined by the following formula:
Using T = 0.3 μs, R = 2.0 kΩ (reference value), C = 6.5 pF (reference value), x = 0.1, y = 1024,

$$R0 = -\frac{0.3 \times 10^{-6}}{6.5 \times 10^{-12} \times 1\mathrm{n}\frac{0.1}{1024}} - 2.0 \times 10^{3}$$

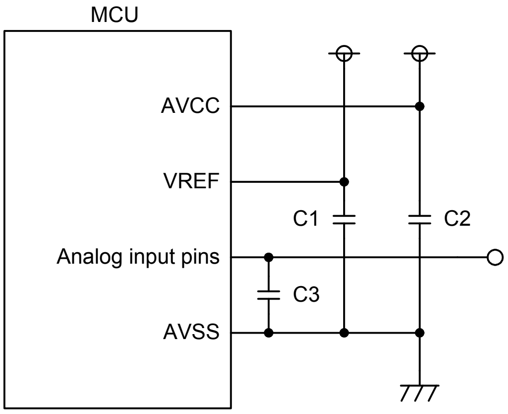
$$= 2998$$

Thus, the allowable output impedance of the sensor equivalent circuit (R0), making the precision (error) of 0.1 LSB or less, should be less than 3 kΩ.
The actual error, however, is the value of absolute precision added to the 0.1 LSB mentioned above.

18.3 Notes on A/D Converter

18.3.1 Notes on Designing Boards

- Three capacitors should be placed between the AVSS pin and pins such as AVCC, VREF, and analog inputs (AN_0 to AN_7, AN0_0 to AN0_7, and AN2_0 to AN2_7) to avoid erroneous operations caused by noise or latchup, and to reduce conversion errors. Figure 18.13 shows an example of pin configuration for A/D converter.



Notes:
1. C1 ≥ 0.47 μF, C2 ≥ 0.47 μF, and C3 ≥ 100 pF (reference values)
2. The traces for the capacitor and the MCU should be as short and wide as physically possible.

Figure 18.13 Pin Configuration for the A/D Converter

- Do not use AN_4 to AN_7 for analog input if the key input interrupt is to be used. Otherwise, a key input interrupt request occurs when the A/D input voltage becomes VIL or lower.

- When AVCC = VREF = VCC, A/D input voltage for pins AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0, and ANEX1 should be VCC or lower.

18.3.2 Notes on Programming

- The following registers should be written while A/D conversion is stopped. That is, before a trigger occurs: AD0CON0 (except the ADST bit), AD0CON1, AD0CON2, AD0CON3, AD0CON4, and AD0CON5.

- If the VCUT bit in the AD0CON1 register is changed from 0 (VREF connected) to 1 (VREF disconnected), wait for at least 1 μs before starting A/D conversion. When not performing A/D conversion, set the VCUT bit to 0 to reduce power consumption.

- Set the port direction bit for the pin to be used as an analog input pin to 0 (input). Set the ASEL bit of the corresponding port function select register to 1 (port is used as A/D input).

- If the TRG bit in the AD0CON0 register is set to 1 (external trigger or hardware trigger is selected), set the corresponding port direction bit (PD9_7 bit) for the $\overline{\text{ADTRG}}$ pin to 0 (input).

- The ϕAD frequency should be 16 MHz or lower. It should be 1 MHz or higher if the sample and hold function is enabled. If not, it should be 250 kHz or higher.

- If A/D operating mode (bits MD1 and MD0 in the AD0CON0 register or the MD2 bit in the AD0CON1 register) has been changed, reselect analog input pins by setting bits CH2 to CH0 in the AD0CON0 register or bits SCAN1 and SCAN0 in the AD0CON1 register.

- If the AD0i register (i = 0 to 7) is read when the A/D converted result is stored to the register, the stored value may have an error. Read the AD0i register after A/D conversion is completed.
  In one-shot mode or single sweep mode, read the AD0i register after the IR bit in the AD0IC register becomes 1 (interrupt requested).
  In repeat mode, repeat sweep mode 0, or repeat sweep mode 1, an interrupt request can be generated each time A/D conversion is completed if the DUS bit in the AD0CON3 register is set to 1 (DMAC operating mode enabled). Similar to the other modes above, read the AD00 register after the IR bit in the AD0IC register becomes 1 (interrupt requested).

- When an A/D conversion is halted by setting the ADST bit in the AD0CON0 register to 0, the converted result is undefined. In addition, the unconverted AD0i register may also become undefined. Consequently, the AD0i register should not be used just after A/D conversion is halted.

- External triggers cannot be used in DMAC operating mode. When the DMAC is configured to transfer converted results, do not read the AD00 register by a program.

- While in single sweep mode, if A/D conversion is halted by setting the ADST bit in the AD0CON0 register to 0 (A/D conversion is stopped), an interrupt request may be generated even though the sweep is not completed. To halt A/D conversion, disable interrupts before setting the ADST bit to 0.

19. D/A Converter

The MCU has two separate 8-bit R-2R resistor ladder D/A converters.

Digital code is converted to an analog voltage when a value is written to the corresponding DAi register (i = 0, 1). The DAiE bit in the DACON register determines whether the D/A conversion result is output or not. To output the converted value, the DAiE bit should be set to 1 (output enabled). This bit setting disables a pull-up resistor for the corresponding port.

Analog voltage to be output (V) is calculated based on the value (n) set in the DAi register (n is a decimal number).

$$V = \frac{VREF \times n}{256} \qquad \text{(n = 0 to 255)}$$

$VREF$: reference voltage

Table 19.1 lists specifications of the D/A converter. Figure 19.1 shows a block diagram of the D/A converter. Figure 19.2 and Figure 19.3 show registers associated with the D/A converter. Figure 19.4 shows a D/A converter equivalent circuit.

When the D/A converter is not used, the DAi register should be set to 00h and the DAiE bit should be set to 0 (output disabled).

Table 19.1 D/A Converter Specifications

Item	Specification
D/A conversion method	R-2R resistor ladder
Resolution	8 bits
Analog output pins	2 channels

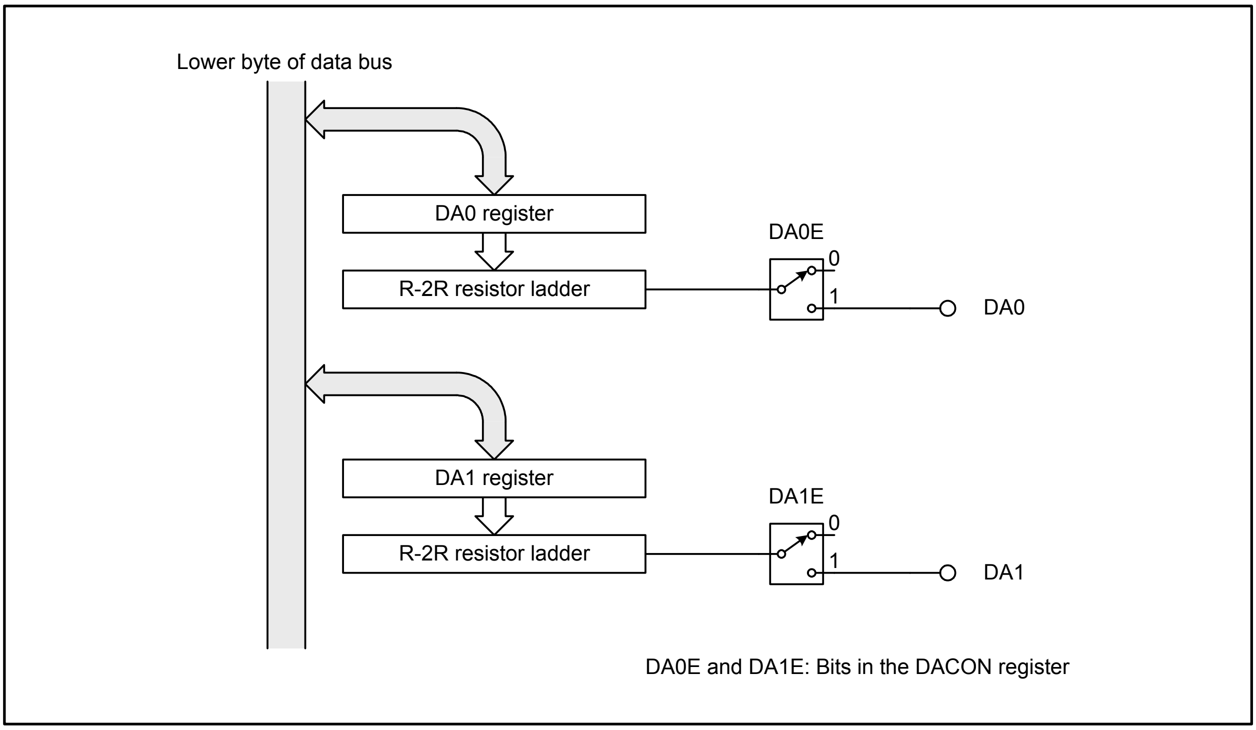
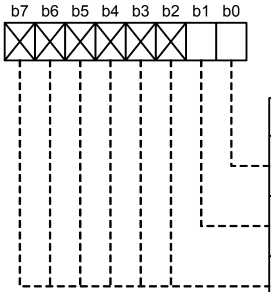


Figure 19.1 D/A Converter Block Diagram

D/A Control Register

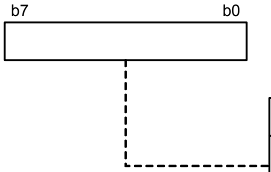


Symbol	Address	Reset Value
DACON	039Ch	XXXX XX00b

Bit Symbol	Bit Name	Function	RW
DA0E	D/A0 Output Enable Bit	0: Output disabled 1: Output enabled	RW
DA1E	D/A1 Output Enable Bit	0: Output disabled 1: Output enabled	RW
— (b7-b2)	No register bits; should be written with 0 and read as undefined value		—

Figure 19.2 DACON Register

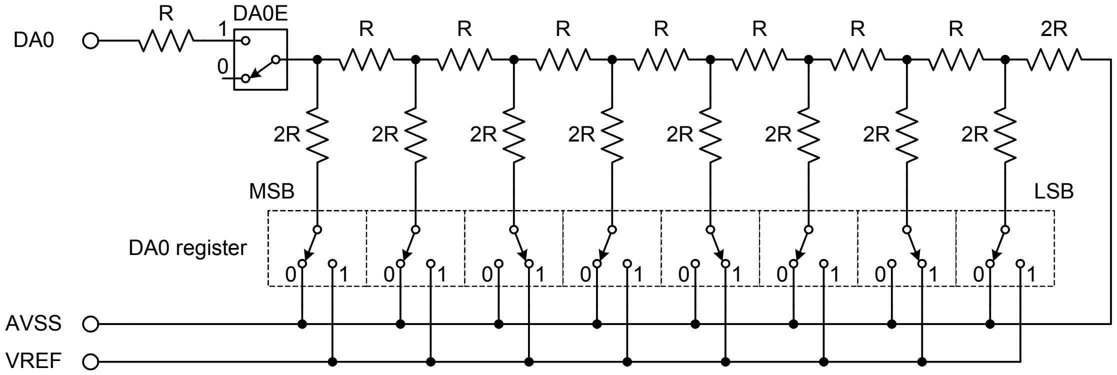
D/A Register i (i = 0, 1)



Symbol	Address	Reset Value
DA0, DA1	0398h, 039Ah	Undefined

Function	Setting Range	RW
Output value by the D/A conversion	00h to FFh	RW

Figure 19.3 Registers DA0 and DA1



Notes:

1. The figure above applies when the DA0 register is set to 2Ah.
2. This circuitry also applies to D/A converter 1.
3. To reduce power consumption when the D/A converter is not in use, the DAiE bit (i = 0, 1) should be set to 0 (output disabled) and the DAi register should be set to 00h to prevent the current from flowing into the R-2R resistor ladder.

Figure 19.4 D/A Converter Equivalent Circuitry

20. CRC Calculator

The Cyclic Redundancy Check (CRC) calculator is used for detecting errors in data blocks. A generator polynomial of CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) generates the CRC.
The CRC is a 16-bit code generated for a given set of blocks of 8-bit data. It is set in the CRCD register every time 1-byte data is written to the CRCIN register after a default value is set to the CRCD register.
Figure 20.1 shows a block diagram of the CRC calculator. Figure 20.2 and Figure 20.3 show registers associated with the CRC. Figure 20.4 shows an example of the CRC calculation.

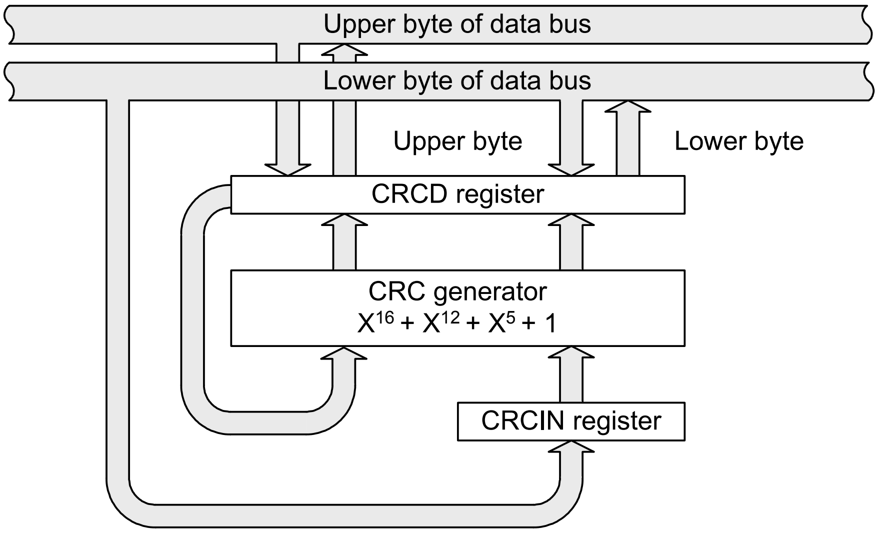
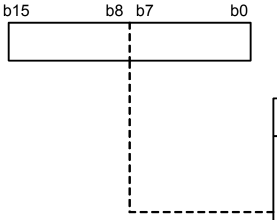


Figure 20.1 CRC Calculator Block Diagram

CRC Data Register



Symbol
CRCD

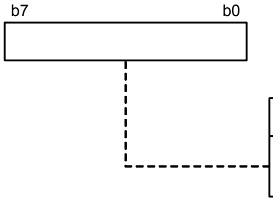
Address
037Dh-037Ch

Reset Value
Undefined

Function	Setting Range	RW
The CRC calculation result is stored in the CRCD register. When a default value in a reversed bit position is set in this register and then data in reversed bit position is written to the CRCIN register, the CRC in the reversed bit position is read from this register	0000h to FFFFh	RW

Figure 20.2 CRCD Register

CRC Input Register



Symbol	Address	Reset Value
CRCIN	037Eh	Undefined

Function	Setting Range	RW
This register is for input data. Input data should be in the reversed bit position	00h to FFh	RW

Figure 20.3 CRCIN Register

CRC Calculation and Setting Procedure to Generate CRC for 80C4h

- CRC Calculation for R32C

CRC: a remainder of the division as follows: $\frac{\text{reversed-bit-position value in the CRCIN register}}{\text{generator polynominal}}$

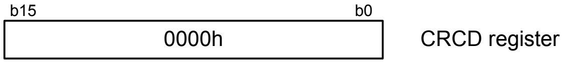
Generator Polynomial: $X^{16} + X^{12} + X^5 + 1$(1 0001 0000 0010 0001b)

- Setting Procedure

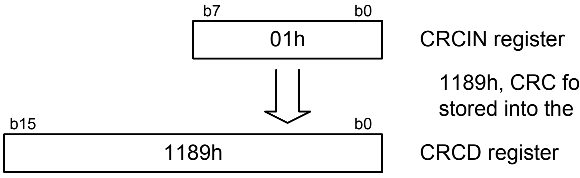
(1) Reverse the bit position of 80C4h in 1-byte units by a program

80h to 01h, C4h to 23h

(2) Set 0000h (default value in reversed bit position) in CRCD register

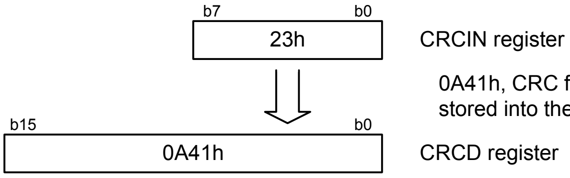


(3) Set 01h (80h in reversed bit position) in CRCIN register



1189h, CRC for 80h (9188h) in reversed bit position is stored into the CRCD register in the third cycle.

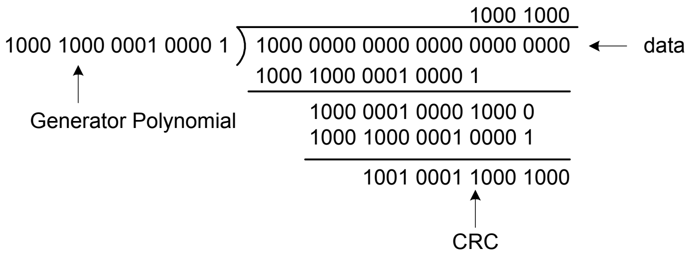
(4) Set 23h (C4h in reversed bit position) in CRCIN register



0A41h, CRC for 80C4h (8250h) in revered bit position is stored into the CRCD register in the third cycle.

- Details of the CRC Calculation

As shown in (3) above, add 1000 0000 0000 0000 0000 0000b as 80h (1000 0000b) plus 16 digits to 0000 0000 0000 0000 0000 0000b as the default value of the CRCD register, 0000h plus eight digits to perform the modulo-2 division.



The modulo-2 calculation is based on the following law

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 0
-1 = 1

0001 0001 1000 1001b (1189h), the reversed-bit-position value of remainder 1001 0001 1000 1000b (9188h) can be read from the CRCD register.

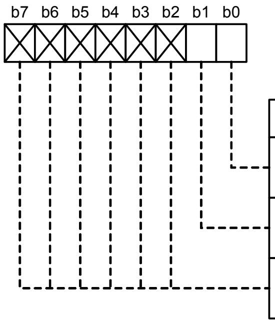
When continuing on to (4) above, add 1100 0100 0000 0000 0000 0000b as C4h (1100 0100b) plus 16 digits to 1001 0001 1000 1000 0000 0000b as the remainder of (3) left in the CRCD register plus eight digits to perform the modulo-2 division.
0000 1010 0100 0001b (0A41h), the reversed-bit-position value of remainder 1000 0010 0101 0000b (8250h) can be read from the CRCD register.

Figure 20.4 CRC Calculation

21. X-Y Conversion

X-Y conversion rotates a 16 × 16-bit matrix data 90 degrees or reverses the bit position of 16-bit data.
X-Y conversion is set using the XYC register shown in Figure 21.1.
Data is written to the write-only XiR registers (i = 0 to 15) and converted data is read from the read-only YjR register (j = 0 to 15). These registers are allocated to the same address. Figure 21.2 and Figure 21.3 show registers XiR and YjR, respectively. A write/read access to registers XiR and YjR should be performed in 16-bit units from an even address. 8-bit access operation results are undefined.

X-Y Control Register

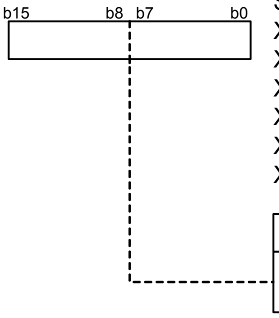


Symbol	Address	Reset Value
XYC	02E0h	XXXX XX00b

Bit Symbol	Bit Name	Function	RW
XYC0	Read Mode Set Bit	0: Data rotation 1: No data rotation	RW
XYC1	Write Mode Set Bit	0: No bit position reverse 1: Bit position reverse	RW
— (b7-b2)	No register bits; should be written with 0 and read as undefined value		—

Figure 21.1 XYC Register

Xi Register (i = 0 to 15) (1)



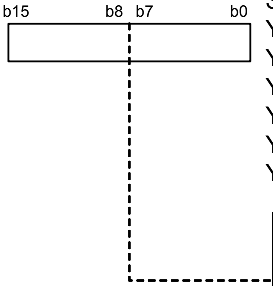
Symbol	Address	Reset Value
X0R to X2R	02C1h-02C0h, 02C3h-02C2h, 02C5h-02C4h	Undefined
X3R to X5R	02C7h-02C6h, 02C9h-02C8h, 02CBh-02CAh	Undefined
X6R to X8R	02CDh-02CCh, 02CFh-02CEh, 02D1h-02D0h	Undefined
X9R to X11R	02D3h-02D2h, 02D5h-02D4h, 02D7h-02D6h	Undefined
X12R to X14R	02D9h-02D8h, 02DBh-02DAh, 02DDh-02DCh	Undefined
X15R	02DFh-02DEh	Undefined

Function	Setting Range	RW
Input data for X-Y conversion	0000h to FFFFh	WO

Note:
1. A 16-bit write access to this register should be performed.

Figure 21.2 Registers X0R to X15R

Yj Register (j = 0 to 15) [(1)]



Symbol	Address	Reset Value
Y0R to Y2R	02C1h-02C0h, 02C3h-02C2h, 02C5h- 02C4h	Undefined
Y3R to Y5R	02C7h-02C6h, 02C9h-02C8h, 02CBh-02CAh	Undefined
Y6R to Y8R	02CDh-02CCh, 02CFh-02CEh, 02D1h-02D0h	Undefined
Y9R to Y11R	02D3h-02D2h, 02D5h-02D4h, 02D7h-02D6h	Undefined
Y12R to Y14R	02D9h-02D8h, 02DBh-02DAh, 02DDh-02DCh	Undefined
Y15R	02DFh-02DEh	Undefined

Function	RW
Result of X-Y conversion	RO

Note:

1. A 16-bit read access to this register should be performed.

Figure 21.3 Registers Y0R to Y15R

21.1 Data Conversion When Reading

Set the XYC0 bit in the XYC register to select a read mode for the YjR register. When the XYC0 bit is set to 0 (data rotation), bit j in the corresponding registers X0R to X15R is automatically read upon reading the YjR register (j = 0 to 15).

More concretely, upon reading bit i (i = 0 to 15) in the Y0R register, the data of bit 0 in the XiR register is read. That is, the read data of bit 0 in the Y15R register means the data of bit 15 in the X0R register and the data of bit 15 in the Y0R register is identical to that of bit 0 in the X15R register.

Figure 21.4 shows the conversion table when the XYC0 bit is set to 0 and Figure 21.5 shows an example of X-Y conversion.

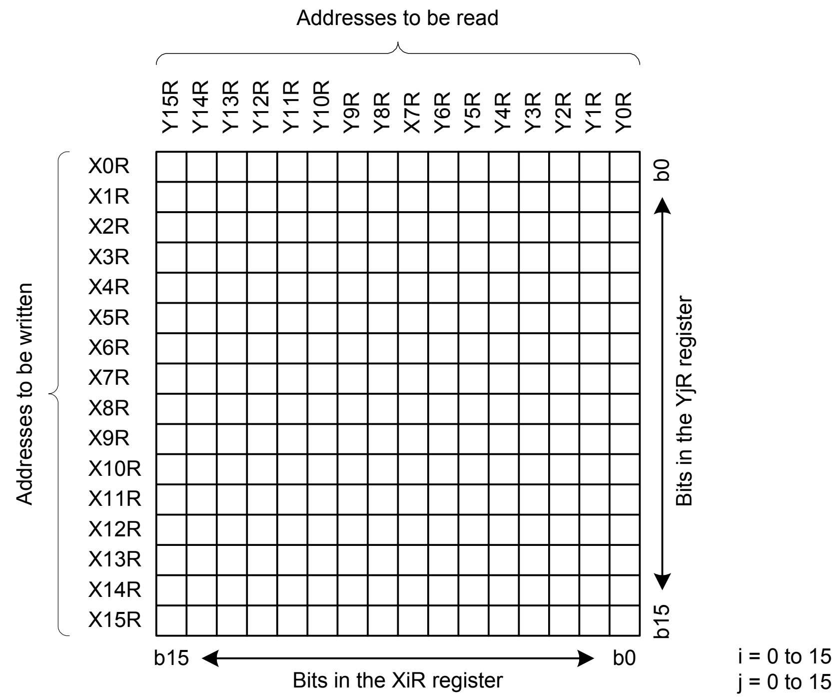


Figure 21.4 Conversion Table (XYC0 Bit is 0)

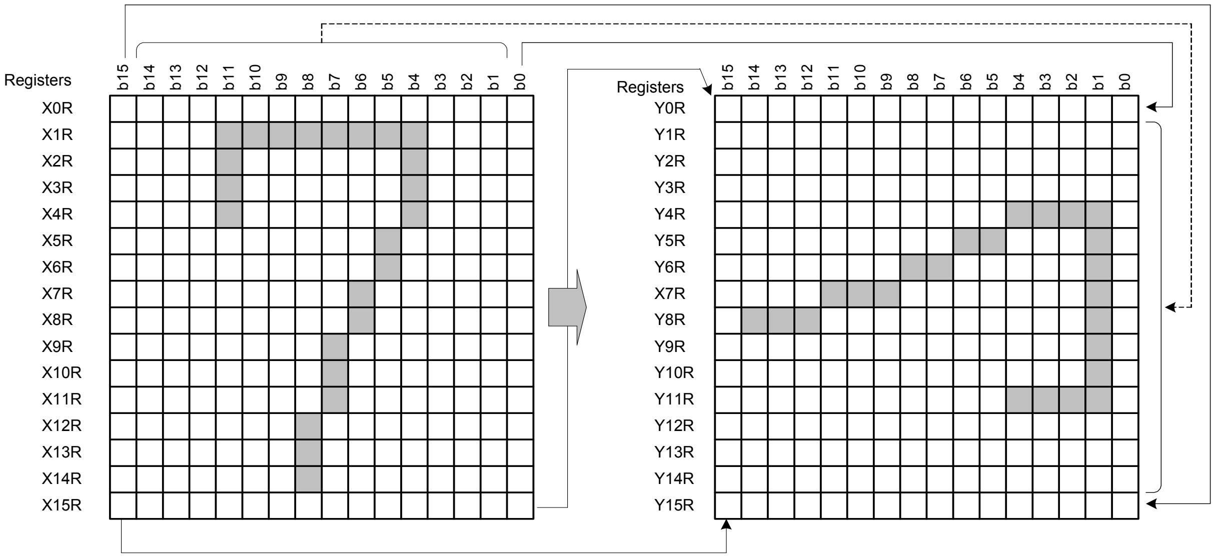


Figure 21.5 X-Y Conversion

When the XYC0 bit is set to 1 (no data rotation), the data of each bit in the YjR register is identical to that written in the XiR register. Figure 21.6 shows the conversion table when the XYC0 bit is set to 1.

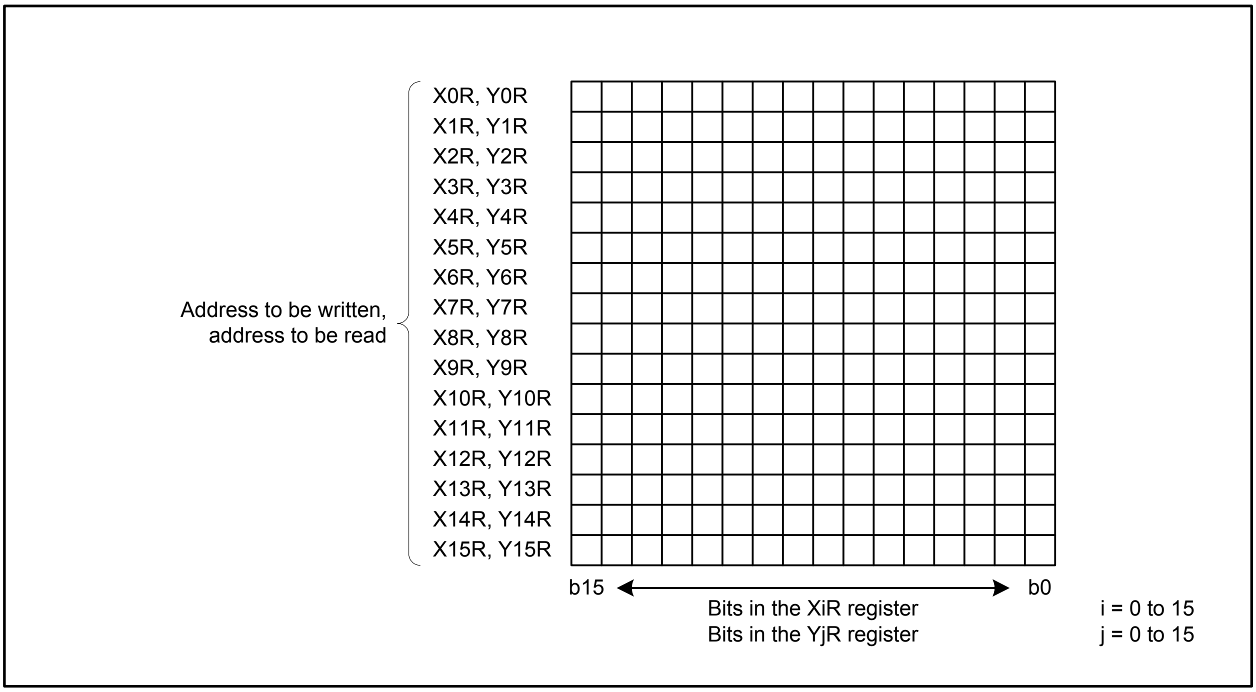


Figure 21.6 Conversion Table (XYC0 Bit is 1)

21.2 Data Conversion When Writing

Set the XYC1 bit in the XYC register to select a write mode for the XiR register.
When the XYC1 bit is set to 0 (no bit position reverse), the data is written in order. When it is set to 1 (bit position reverse), the data is written in reversed order. Figure 21.7 shows the conversion table when the XYC1 bit is set to 1.

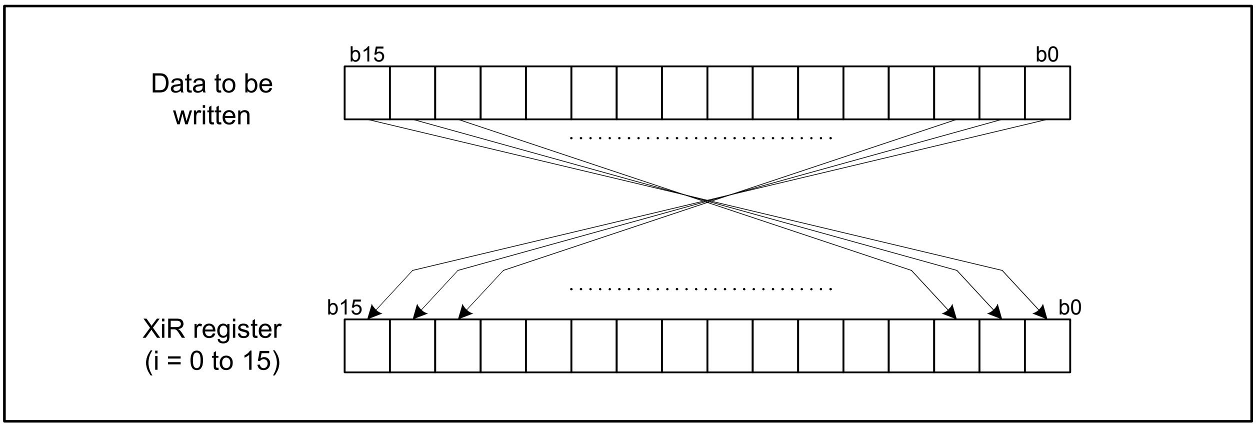


Figure 21.7 Conversion Table (XYC1 Bit is 1)

22. Intelligent I/O

The intelligent I/O is a multifunctional I/O port for time measurement and waveform generation.
It consists of two groups each of which has one free-running 16-bit base timer and eight 16-bit registers for time measurement or waveform generation.
Table 22.1 lists the functions and channels of the intelligent I/O.

Table 22.1 Intelligent I/O Functions and Channels

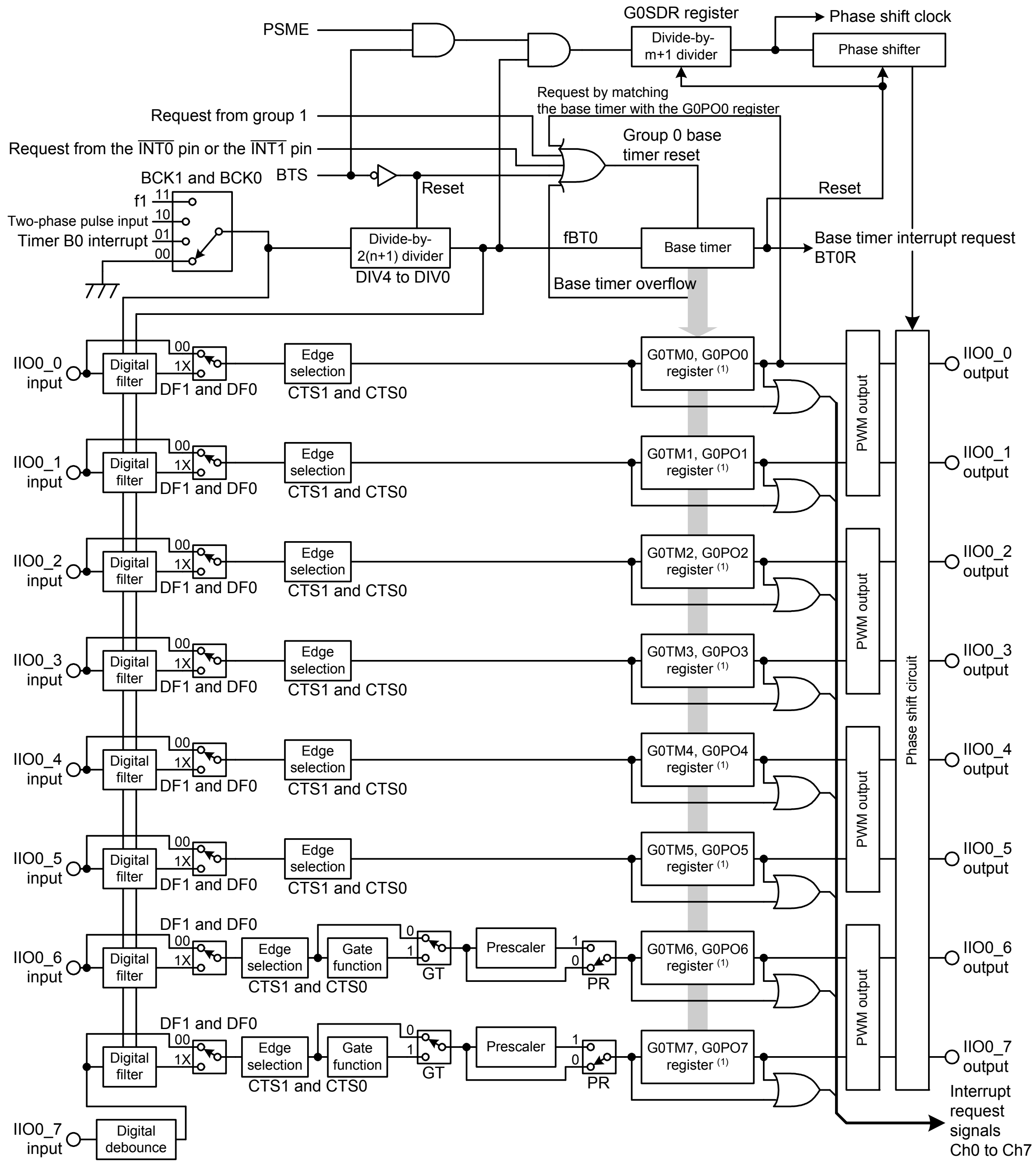
Functions		Group 0	Group 1
Time measurement (1)	Digital filter	8 channels	8 channels
	Prescaler	2 channels	2 channels
	Gating	2 channels	2 channels
	Digital debounce	1 channel	1 channel
Waveform generation (1)	Single-phase waveform output mode	8 channels	8 channels
	Inverted waveform output mode	8 channels	8 channels
	SR waveform output mode	8 channels	8 channels
	Phase shift waveform output mode	8 channels	8 channels

Note:

1. The time measurement and waveform generation functions share a pin.

Each channel can be individually assigned for time measurement or waveform generation function.

Figure 22.1 and Figure 22.2 show block diagrams of the intelligent I/O.

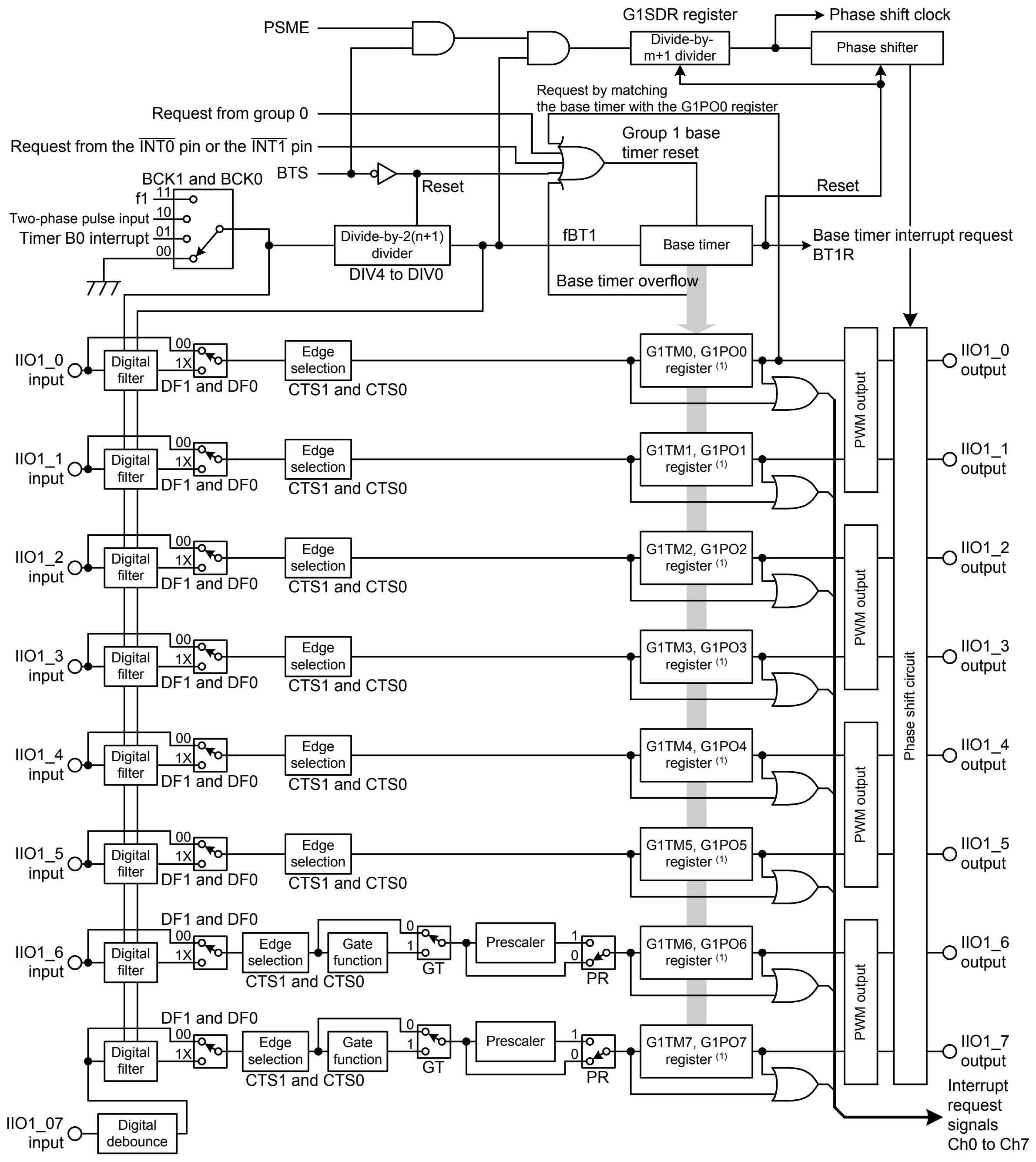


DIV4 to DIV0, BCK1, and BCK0: Bits in the G0BCR0 register
BTS: Bit in the G0BCR1 register
CTS1, CTS0, DF1, DF0, GT, and PR: Bits in the G0TMCRj register
PSME: Bit in the G0PSMR register
BT0R: Bit in the IIO7IR register

Note:

1. Each register is placed in a reset state after the clock is provided via the G0BCR0 register.

Figure 22.1 Intelligent I/O Group 0 Block Diagram (j = 0 to 7)



DIV4 to DIV0, BCK1, and BCK0: Bits in the G1BCR0 register
BTS: Bit in the G1BCR1 register
CTS1, CTS0, DF1, DF0, GT, and PR: Bits in the G1TMCRj register
PSME: Bit in the G1PSMR register
BT1R: Bit in the IIO4IR register

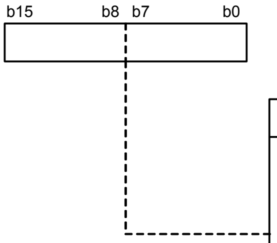
Note:

1. Each register is placed in a reset state after the clock is provided via the G1BCR0 register.

Figure 22.2 Intelligent I/O Group 1 Block Diagram (j = 0 to 7)

Figure 22.3 to Figure 22.14 show registers associated with the intelligent I/O base timer, time measurement, and waveform generation.

Group i Base Timer Register (i = 0, 1) (1)



Symbol	Address	Reset Value
G0BT, G1BT	01A1h-01A0h, 0121h-0120h	Undefined

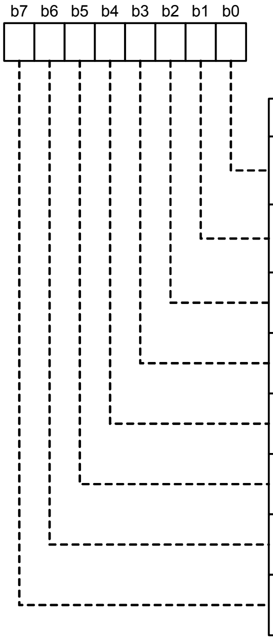
Function	Setting Range	RW
- While the base timer is running, this register indicates the value of base timer; when a value is written, the counter immediately starts counting from this value. The register is set to 0000h when the base timer is reset - While the base timer is being reset, this register is set to 0000h; the register is read as undefined value; no value can be set (2)	0000h to FFFFh	RW

Notes:

1. The GiBT register reflects the base timer value after a delay of a half fBTi cycle.
2. The base timer stops only when bits BCK1 and BCK0 in the GiBCR0 register are set to 00b (clock stopped). However, the base timer can be in a “no-counting” state, holding the value 0000h, by setting the BTS bit in the GiBCR1 register to 0 (reset the base timer). When this bit is set to 1 (start counting), this state is cleared and the base timer starts counting.

Figure 22.3 Registers G0BT and G1BT

Group i Base Timer Control Register 0 (i = 0, 1)



Symbol	Address	Reset Value
G0BCR0, G1BCR0	01A2h, 0122h	0000 0000b

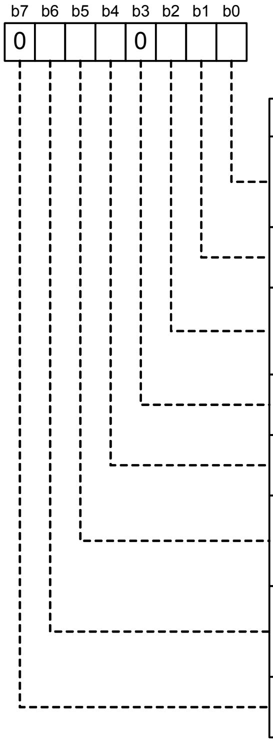
Bit Symbol	Bit Name	Function	RW
BCK0	Count Source Select Bit [1]	b1 b2 0 0 : Clock stopped 0 1 : Timer B0 interrupt [2] 1 0 : Two-phase pulse signal input [3] 1 1 : f1	RW
BCK1			RW
DIV0	Count Source Divide Ratio Select Bit	Divide the count source by 2(n+1). The count source is not divided when n = 31 (n = 0 to 31). b6 b5 b4 b3 b2 0 0 0 0 0 : divide-by-2 (n = 0) 0 0 0 0 1 : divide-by-4 (n = 1) 0 0 0 1 0 : divide-by-6 (n = 2) : 1 1 1 1 0 : divide-by-62 (n = 30) 1 1 1 1 1 : no division (n = 31) [2]	RW
DIV1			RW
DIV2			RW
DIV3			RW
DIV4			RW
IT	Base Timer Interrupt Source Select Bit	0: Overflow of bit 15 or bit 9 1: Overflow of bit 14	RW

Notes:

1. Each register of the intelligent I/O is initialized after the clock with a count source selected by this register is provided. For this initializing operation, f1 should be selected as count source.
2. Do not set bits DIV4 to DIV0 to 11111b when bits BCK1 and BCK0 are set to 01b.
3. This bit setting is enabled only when bits UD1 and UD0 in the GiBCR1 register are set to 10b (two-phase pulse signal processing mode). Bits BCK1 and BCK0 should not be set to 10b in other modes.

Figure 22.4 Registers G0BCR0 and G1BCR0

Group i Base Timer Control Register 1 (i = 0, 1)



Symbol	Address	Reset Value
G0BCR1, G1BCR1	01A3h, 0123h	0000 0000b

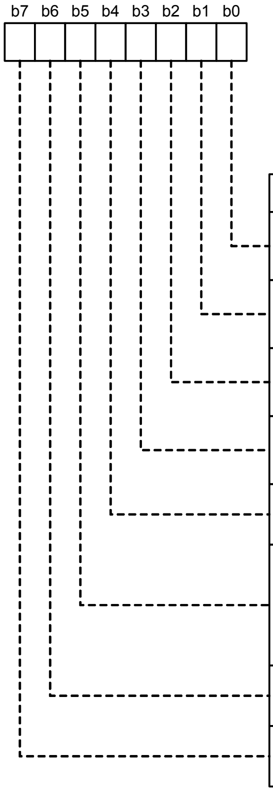
Bit Symbol	Bit Name	Function	RW
RST0	Base Timer Reset Source Select Bit 0	0: No reset 1: Synchronization with another base timer reset [1]	RW
RST1	Base Timer Reset Source Select Bit 1	0: No reset 1: Match with the GiPO0 register [2]	RW
RST2	Base Timer Reset Source Select Bit 2	0: No reset 1: Low signal input into the $\overline{\text{INT0}}$/ $\overline{\text{INT1}}$ pin [3]	RW
— (b3)	Reserved	Should be written with 0	RW
BTS	Base Timer Start Bit	0: Reset the base timer 1: Start counting	RW
UD0	Increment/Decrement Control Bit	b6 b5 0 0 : Increment mode 0 1 : Increment/decrement mode 1 0 : Two-phase pulse signal processing mode [4] 1 1 : Do not use this combination	RW
UD1			RW
— (b7)	Reserved	Should be written with 0	RW

Notes:

1. The group 0 base timer is reset by synchronizing with the reset of group 1 base timer, and vice versa.
2. The base timer is reset after two fBTi clock cycles if the base timer value matches the GiPO0 register setting. When the RST1 bit is set to 1, the value of GiPOj register (j = 1 to 7) used for waveform generation should be smaller than that of the GiPO0 register.
3. The base timer is reset by an input of low signal to the external interrupt input pin selected for the UDiZ signal by the IFS2 register.
4. In two-phase pulse signal processing mode, the base timer is not reset, even though the RST1 bit is set to 1, if the timer counter decrements after two clock cycles when the base timer value matches the GiPO0 register.

Figure 22.5 Registers G0BCR1 and G1BCR1

Group i Time Measurement Control Register j (i = 0, 1; j = 0 to 7)



Symbol	Address	Reset Value
G0TMCR0 to G0TMCR3	0198h, 0199h, 019Ah, 019Bh	0000 0000b
G0TMCR4 to G0TMCR7	019Ch, 019Dh, 019Eh, 019Fh	0000 0000b
G1TMCR0 to G1TMCR3	0118h, 0119h, 011Ah, 011Bh	0000 0000b
G1TMCR4 to G1TMCR7	011Ch, 011Dh, 011Eh, 011Fh	0000 0000b

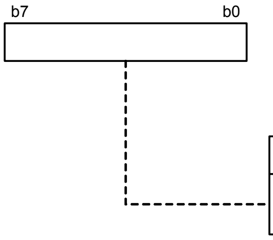
Bit Symbol	Bit Name	Function	RW
CTS0	Time Measurement Trigger Select Bit	b1 b0 0 0 : No time measurement 0 1 : Rising edge 1 0 : Falling edge 1 1 : Both edges	RW
CTS1			RW
DF0	Digital Filter Select Bit	b3 b2 0 0 : No digital filter used 0 1 : Do not use this combination 1 0 : fBTi 1 1 : f1	RW
DF1			RW
GT	Gating Select Bit (1)	0: Gating disabled 1: Gating enabled	RW
GOC	Gating Clear Select Bit (1, 2)	0: Gating not cleared 1: Gating cleared when the base timer matches the GiPOk register (k = j - 2)	RW
GSC	Gating Clear Bit (1, 2)	Gating is cleared by setting this bit to 1	RW
PR	Prescaler Select Bit (1)	0: Prescaler disabled 1: Prescaler enabled	RW

Notes:

1. These functions are available in registers GiTMCR6 and GiTMCR7. Bits 4 to 7 in registers GiTMCR0 to GiTMCR5 should be set to 0.
2. These bit settings are enabled when the GT bit is set to 1.

Figure 22.6 Registers G0TMCR0 to G0TMCR7 and G1TMCR0 to G1TMCR7

Group i Time Measurement Prescaler Register j (i = 0, 1; j = 6, 7)



Symbol	Address	Reset Value
G0TPR6, G0TPR7	01A4h, 01A5h	00h
G1TPR6, G1TPR7	0124h, 0125h	00h

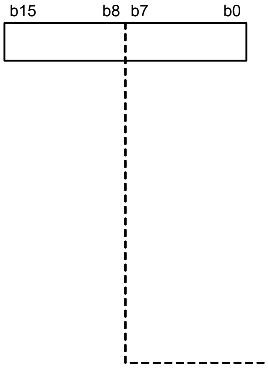
Function	Setting Range	RW
Time measurement is executed whenever a trigger input is counted by n+1 (n = setting value) (1)	00h to FFh	RW

Note:

1. The first prescaler, after the PR bit in the GiTMCRj register is changed from 0 (prescaler disabled) to 1 (prescaler enabled), may be divided by n rather than n+1. The subsequent prescaler is divided by n+1.

Figure 22.7 Registers G0TPR6, G0TPR7, G1TPR6, and G1TPR7

Group i Time Measurement Register j (i = 0, 1; j = 0 to 7)

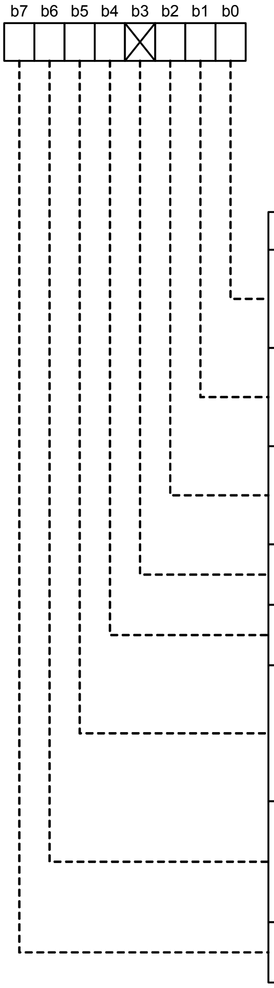


Symbol	Address	Reset Value
G0TM0, G0TM1	0181h-0180h, 0183h-0182h	Undefined
G0TM2, G0TM3	0185h-0184h, 0187h-0186h	Undefined
G0TM4, G0TM5	0189h-0188h, 018Bh-018Ah	Undefined
G0TM6, G0TM7	018Dh-018Ch, 018Fh-018Eh	Undefined
G1TM0, G1TM1	0101h-0100h, 0103h-0102h	Undefined
G1TM2, G1TM3	0105h-0104h, 0107h-0106h	Undefined
G1TM4, G1TM5	0109h-0108h, 010Bh-010Ah	Undefined
G1TM6, G1TM7	010Dh-010Ch, 010Fh-010Eh	Undefined

Function	Setting Range	RW
The base timer value is stored every measurement timing	—	RO

Figure 22.8 Registers G0TM0 to G0TM7 and G1TM0 to G1TM7

Group i Waveform Generation Control Register j (i = 0, 1; j = 0 to 7)



Symbol	Address	Reset Value
G0POCR0	0190h	0000 X000b
G0POCR1 to G0POCR3	0191h, 0192h, 0193h	0X00 X000b
G0POCR4 to G0POCR7	0194h, 0195h, 0196h, 0197h	0X00 X000b
G1POCR0	0110h	0000 X000b
G1POCR1 to G1POCR3	0111h, 0112h, 0113h	0X00 X000b
G1POCR4 to G1POCR7	0114h, 0115h, 0116h, 0117h	0X00 X000b

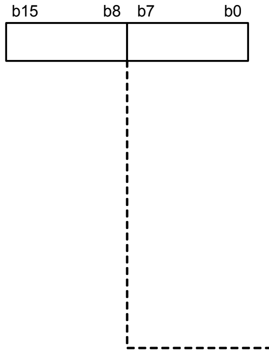
Bit Symbol	Bit Name	Function	RW
MOD0	Operating Mode Select Bit	b2 b1 b0 0 0 0 : Single-phase waveform output mode 0 0 1 : SR waveform output mode (1) 0 1 0 : Inverted waveform output mode 0 1 1 : Do not use this combination 1 0 0 : Do not use this combination 1 0 1 : Do not use this combination 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
MOD1			RW
MOD2			RW
— (b3)	No register bit; should be written with 0 and read as undefined value		—
IVL	Default Output Value Select Bit (2)	0: Output low as default value 1: Output high as default value	RW
RLD	GiPOj Register Value Reload Timing Select Bit	0: Reload the value into the GiPOj register on a write access 1: Reload the value into the GiPOj register when the base timer is reset	RW
BTRE	Base Timer Reset Enable Bit (3)	0: Reset the base timer when bit 15 overflows 1: Reset the base timer when bit 9 overflows (4)	RW
INV	Output Level Inversion Select Bit (5)	0: Do not invert the output level 1: Invert the output level	RW

Notes:

1. This bit setting is enabled only for even channels. In SR waveform output mode, the corresponding odd channel (the next channel after an even channel) setting is ignored. Waveforms are only output from even channels.
2. The setting value is output by a write operation to the IVL bit if the FSCj bit in the GiFS register is set to 0 (select the waveform generation) and the IFEj bit in the GiFE register is set to 1 (enable the channel j function).
3. This bit is available only in the GiPOCR0 register. Bit 6 in registers GiPOCR1 to GiPOCR7 should be set to 0.
4. To set the BTRE bit to 1, bits BCK1 and BCK0 in the GiBCR0 register should be set to 01b (timer B0 interrupt) or 11b (f1) and bits UD1 and UD0 in the GiBCR1 register should be set to 00b (increment mode).
5. The output level inversion is the final step in the waveform generation process. When the INV bit is set to 1 (invert the output level), high is output by setting the IVL bit to 0 (output low as default value), and vice versa.

Figure 22.9 Registers G0POCR0 to G0POCR7 and G1POCR0 to G1POCR7

Group i Waveform Generation Register j (i = 0, 1; j = 0 to 7)

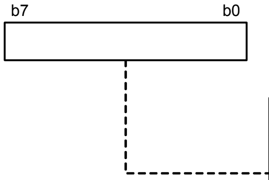


Symbol	Address	Reset Value
G0PO0 to G0PO2	0181h-0180h, 0183h-0182h, 0185h-0184h	Undefined
G0PO3 to G0PO5	0187h-0186h, 0189h-0188h, 018Bh-018Ah	Undefined
G0PO6, G0PO7	018Dh-018Ch, 018Fh-018Eh	Undefined
G1PO0 to G1PO2	0101h-0100h, 0103h-0102h, 0105h-0104h	Undefined
G1PO3 to G1PO5	0107h-0106h, 0109h-0108h, 010Bh-010Ah	Undefined
G1PO6, G1PO7	010Dh-010Ch, 010Fh-010Eh	Undefined

Function	Setting Range	RW
- When the RLD bit in the GiPOCRj register is set to 0, the value is reloaded into the GiPOj register immediately after being written and is reflected in the output waveform - When the RLD bit is set to 1, the value is reloaded when the base timer is reset. The register indicates the written value until the value is reloaded	0000h to FFFFh	RW

Figure 22.10 Registers G0PO0 to G0PO7 and G1PO0 to G1PO7

Group i Phase Shift Waveform Output Mode Clock Division Setting Register (i = 0, 1)(1)



Symbol	Address	Reset Value
G0SDR, G1SDR	0200h, 0202h	00h

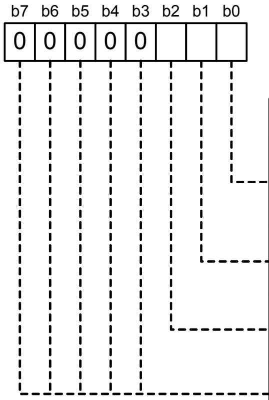
Function	Setting Range	RW
The GiSDR register divides a base timer count source (fBTi) by m+1 (m = setting value)	01h to FFh	RW

Note:

1. This register should be written while the base timer is reset.

Figure 22.11 Registers G0SDR and G1SDR

Group i Phase Shift Waveform Output Mode Control Register (i = 0, 1)



Symbol	Address	Reset Value
G0PSCR, G1PSCR	0201h, 0203h	0000 0000b

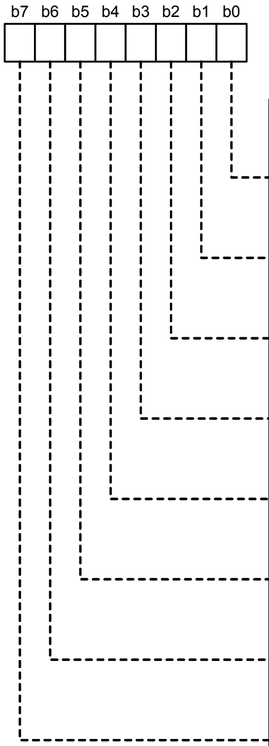
Bit Symbol	Bit Name	Function	RW
PSME	Phase Shift Waveform Output Mode Enable Bit (1)	0: Disabled 1: Enabled (2)	RW
PSS0	Phase Shift Waveform Output Channel Select Bit	b2 b1 0 0 : IIOi_0 to IIOi_4 0 1 : IIOi_0 to IIOi_5 1 0 : IIOi_0 to IIOi_6 1 1 : IIOi_0 to IIOi_7	RW
PSS1			
— (b7-b3)	No register bits; should be written with 0 and read as 0		—

Notes:

1. To set this bit to 1, select output channels by bits PSS1 and PSS0.
   Also set the operating mode of the selected channel to single waveform output mode (bits MOD2 to MOD0 in the GiPOCRj register are set to 000b).
2. In phase shift waveform output mode, start the base timer after setting this bit to 1.

Figure 22.12 Registers G0PSCR and G1PSCR

Group i Function Select Register (i = 0, 1)

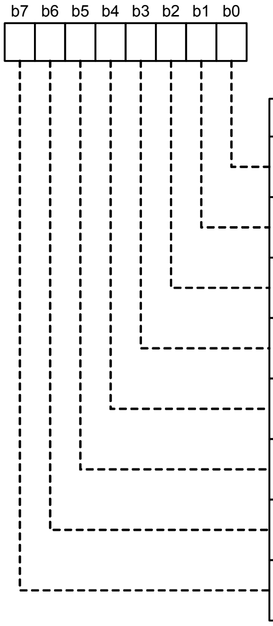


Symbol	Address	Reset Value
G0FS, G1FS	01A7h, 0127h	0000 0000b

Bit Symbol	Bit Name	Function	RW
FSC0	Channel 0 Time Measurement/Waveform Generation Select Bit	0: Select the waveform generation 1: Select the time measurement	RW
FSC1	Channel 1 Time Measurement/Waveform Generation Select Bit		RW
FSC2	Channel 2 Time Measurement/Waveform Generation Select Bit		RW
FSC3	Channel 3 Time Measurement/Waveform Generation Select Bit		RW
FSC4	Channel 4 Time Measurement/Waveform Generation Select Bit		RW
FSC5	Channel 5 Time Measurement/Waveform Generation Select Bit		RW
FSC6	Channel 6 Time Measurement/Waveform Generation Select Bit		RW
FSC7	Channel 7 Time Measurement/Waveform Generation Select Bit		RW

Figure 22.13 Registers G0FS and G1FS

Group i Function Enable Register (i = 0, 1)



Symbol	Address	Reset Value
G0FE, G1FE	01A6h, 0126h	0000 0000b

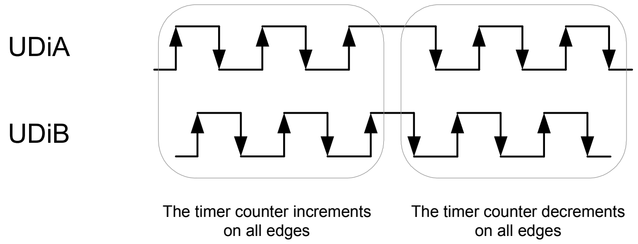
Bit Symbol	Bit Name	Function	RW
IFE0	Channel 0 Function Enable Bit	0: Disable the channel j function 1: Enable the channel j function (j = 0 to 7)	RW
IFE1	Channel 1 Function Enable Bit		RW
IFE2	Channel 2 Function Enable Bit		RW
IFE3	Channel 3 Function Enable Bit		RW
IFE4	Channel 4 Function Enable Bit		RW
IFE5	Channel 5 Function Enable Bit		RW
IFE6	Channel 6 Function Enable Bit		RW
IFE7	Channel 7 Function Enable Bit		RW

Figure 22.14 Registers G0FE and G1FE

22.1 Base Timer

The base timer is a free-running counter that counts an internally generated count source. Table 22.2 lists specifications of the base timer. Figure 22.3 to Figure 22.14 show registers associated with the base timer. Figure 22.15 shows a block diagram of the base timer. Figure 22.16, Figure 22.17, and Figure 22.18 show operation examples of the base timer in increment mode, increment/decrement mode, and two-phase pulse signal processing mode, respectively.

Table 22.2 Base Timer Specifications (i = 0, 1)

Item	Specification
Count source (fBTi)	f1 divided by 2(n+1), two-phase pulse input divided by 2(n+1) n: setting value using bits DIV4 to DIV0 in the GiBCR0 register n = 0 to 31; however no division when n = 31 Timer B0 interrupt divided by 2(m+1) m: setting value using bits DIV4 to DIV0 in the GiBCR0 register m = 0 to 30
Count operations	• Increment • Increment/decrement • Two-phase pulse signal processing
Count start conditions	Set the BTS bit in the GiBCR1 register to 1 (start counting)
Count stop condition	Set the BTS bit in the GiBCR1 register to 0 (reset the base timer)
Reset conditions	• The base timer value matches the GiPO0 register setting • An input of low signal into the external interrupt pin ($\overline{\text{INT0}}$ or $\overline{\text{INT1}}$) as follows: for group 0: selected using bits IFS23 and IFS22 in the IFS2 register for group 1: selected using bits IFS27 and IFS26 in the IFS2 register • The overflow of bit 15 or bit 9 in the base timer
Reset value	0000h
Interrupt request	When the BTiR bit in the interrupt request register is set to 1 (interrupt requested) by the overflow of bit 9, 14, or 15 in the base timer (refer to Figure 10.15)
Read from base timer	• The GiBT register indicates a counter value while the base timer is running • The GiBT register is undefined while the base timer is being reset
Write to base timer	When a value is written while the base timer is running, the timer counter immediately starts counting from this value. No value can be written while the base timer is being reset
Selectable functions	• Increment/decrement mode The base timer starts counting when the BTS bit is set to 1. When the base timer reaches FFFFh, it starts decrementing. When the RST1 bit in the GiBCR1 register is set to 1 (reset by match with the GiPO0 register), the timer counter starts decrementing two counts after the base timer value matches the GiPO0 register setting. When the timer counter reaches 0000h, it starts incrementing again (refer to Figure 22.17). • Two-phase pulse signal processing mode Two-phase pulse signals at pins UDiA and UDiB are counted (refer to Figure 22.18). 

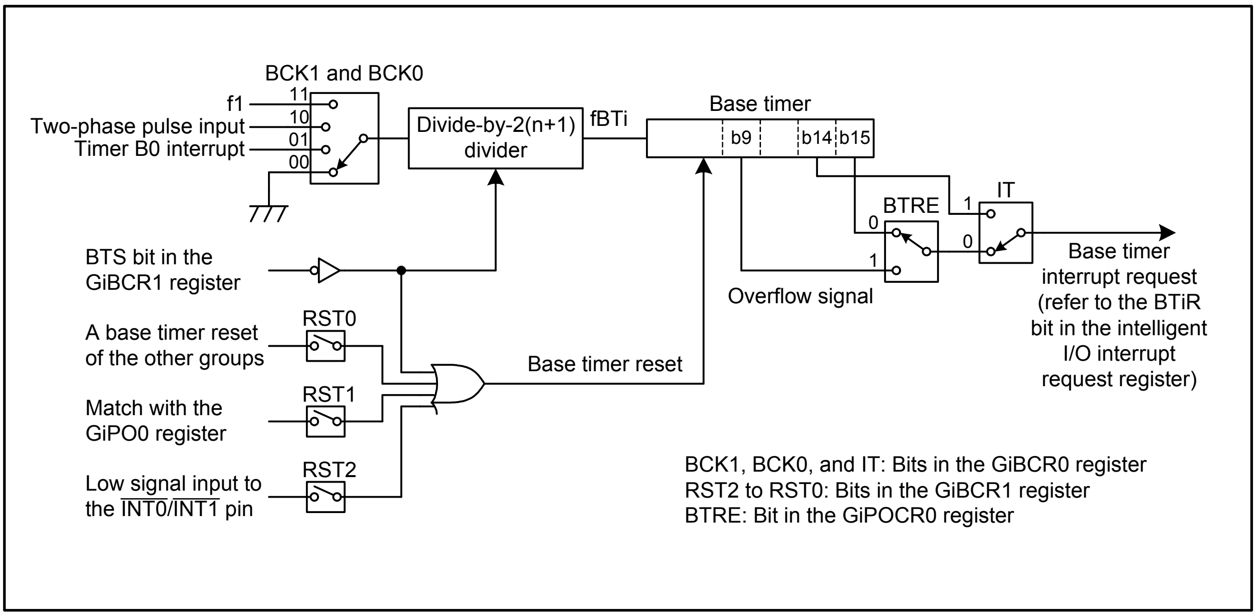


Figure 22.15 Base Timer Block Diagram (i = 0, 1)

Table 22.3 Base Timer Associated Register Settings (Common Settings for Time Measurement, Waveform Generation, and Serial Interface) (i = 0, 1)

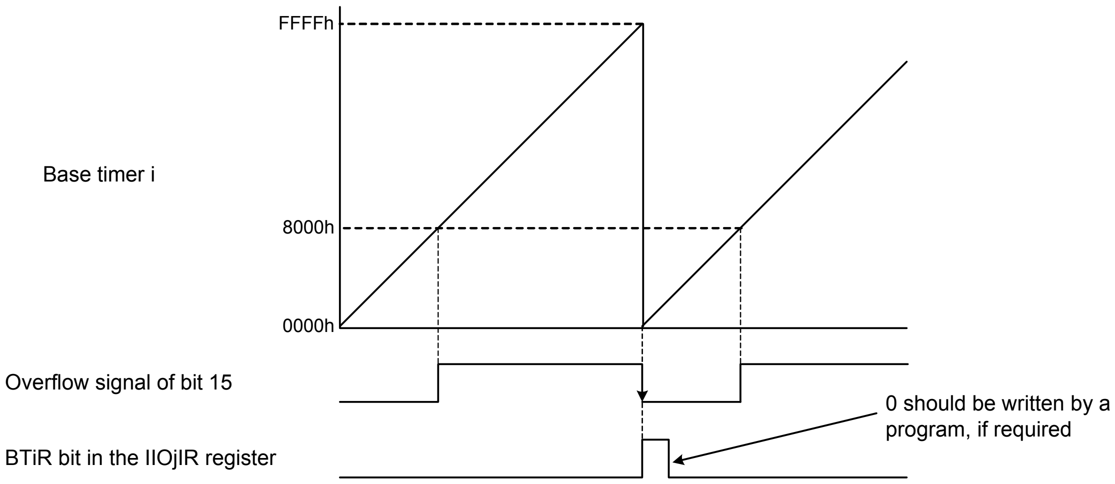
Register	Bits	Function
GiBCR0	BCK1 and BCK0	Select a count source
	DIV4 to DIV0	Select a count source divide ratio
	IT	Select a base timer interrupt source
GiBCR1	RST2 to RST0	Select a timing for base timer reset
	BTS	Use this bit when each base timer individually starts counting
	UD1 and UD0	Select a count operation
GiPOCR0	BTRE	Select a source for base timer reset
GiBT	—	Read or write the base timer value

The following register settings are required to set the RST1 bit to 1 (base timer is reset when it matches with the GiPO0 register).

GiPOCR0	MOD2 to MOD0	Set to 000b (single-phase waveform output mode)
GiPO0	—	Set the reset cycle
GiFS	FSC0	Set the bit to 0 (select the waveform generation)
GiFE	IFE0	Set the bit to 1 (channel operation starts)

Bit configurations and functions vary by group.

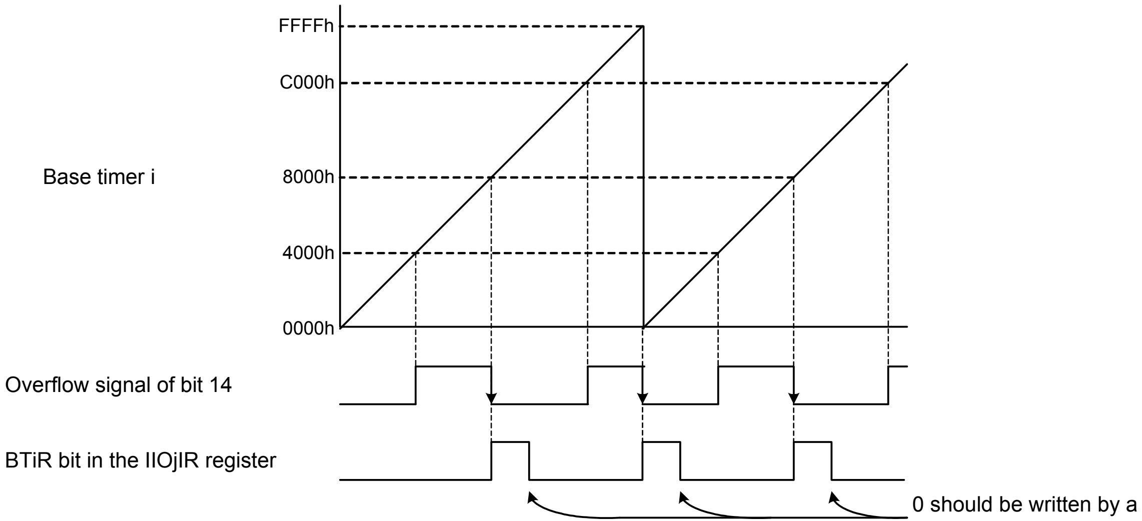
(1) When the IT bit in the GiBCR0 register is set to 0
(an interrupt is requested by the overflow of bit 15 in the base timer)



j = 7, 4

The figure above applies under the following conditions:
- The RST1 bit in the GiBCR1 register is set to 0 (the match with the GiPO0 register is not the reset source for the base timer)
- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)

(2) When the IT bit in the GiBCR0 register is set to 1
(an interrupt is requested by the overflow of bit 14 in the base timer)

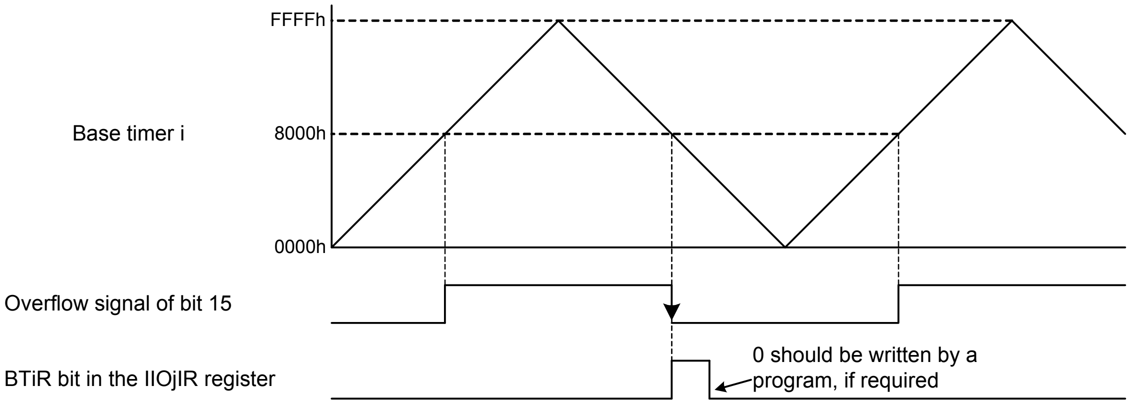


j = 7, 4

The figure above applies under the following conditions:
- The RST1 bit in the GiBCR1 register is set to 0 (the match with the GiPO0 register is not the reset source for the base timer)
- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)

Figure 22.16 Base Timer Increment Mode (i = 0, 1)

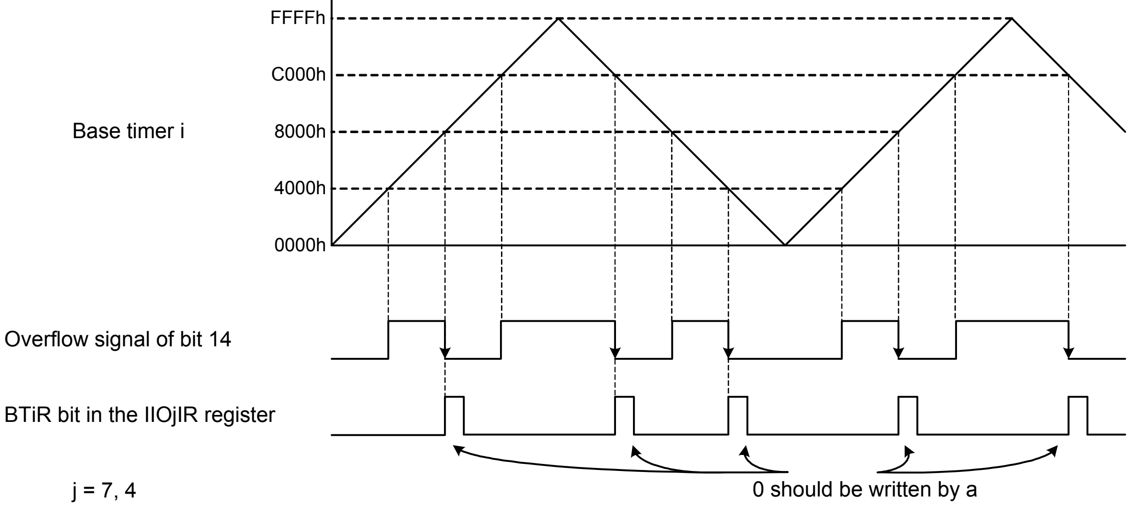
(1) When the IT bit in the GiBCR0 register is set to 0
(an interrupt is requested by the overflow of bit 15 in the base timer)



j = 7, 4

The figure above applies under the following conditions:
- The RST1 bit in the GiBCR1 register is set to 0 (the match with the GiPO0 register is not the reset source for the base timer)
- Bits UD1 and UD0 in the GiBCR1 register are set to 01b (increment/decrement mode)

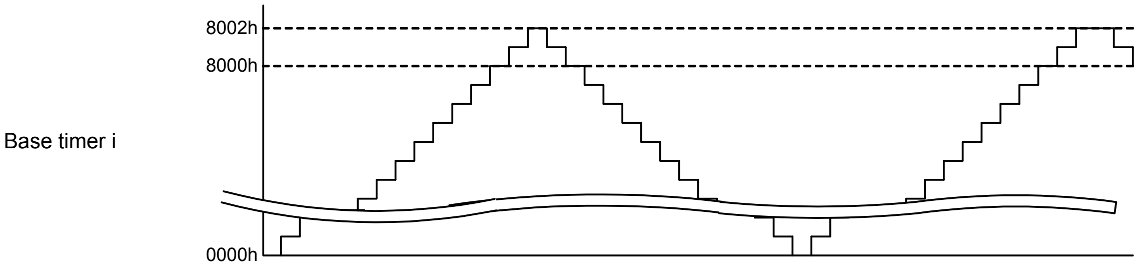
(2) When the IT bit in the GiBCR0 register is set to 1
(an interrupt request is requested by the overflow of bit 14 in the base timer)



j = 7, 4

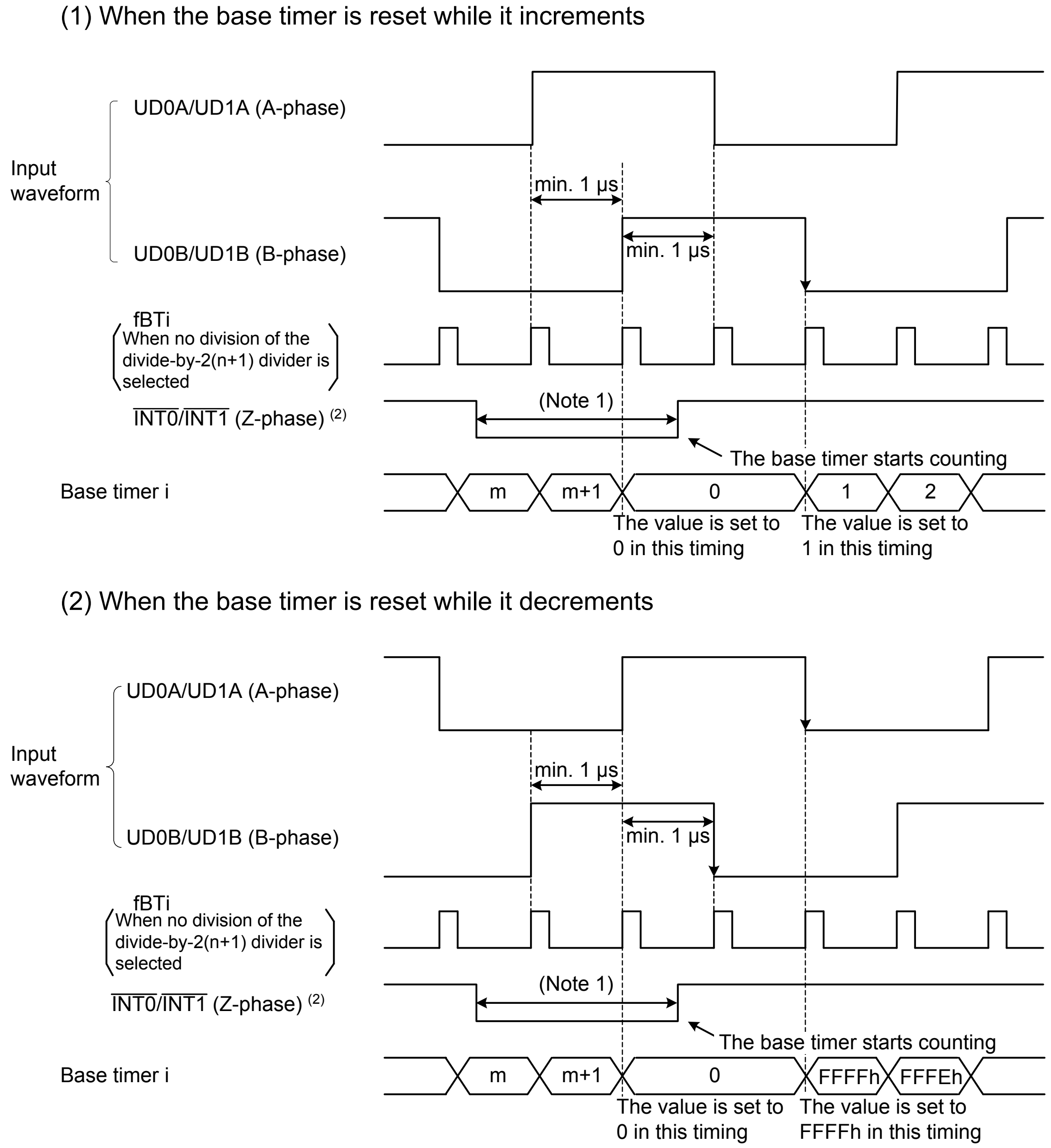
The figure above applies under the following conditions:
- The RST1 bit in the GiBCR1 register is set to 0 (the match with the GiPO0 register is not the reset source for the base timer)
- Bits UD1 and UD0 in the GiBCR1 register are set to 01b (increment/decrement mode)

(3) When the RST bit in the GiBCR1 register is set to 1 (the base timer is reset by matching with the GiPO0 register)



The figure above applies under the following conditions:
- The GiPO0 register value is set to 8000h
- Bits UD1 and UD0 in the GiBCR1 register are set to 01b (increment/decrement mode)

Figure 22.17 Base Timer Increment/Decrement (i = 0, 1)



Notes:
1. At least 1.5 fBTi clock cycles are required.
2. The RST2 bit in the GiBCR1 register should be set to 1 in two-phase pulse signal processing mode.

Figure 22.18 Base Timer Two-phase Pulse Signal Processing Mode (i = 0, 1)

22.2 Time Measurement

Every time an external trigger is input, the base timer value is stored into the GiTMj register (i = 0, 1; j = 0 to 7). Table 22.4 lists specifications of the time measurement and Table 22.5 lists its register settings. Figure 22.19 and Figure 22.20 show operation examples of the time measurement and Figure 22.21 shows operation examples with the prescaler or gate function.

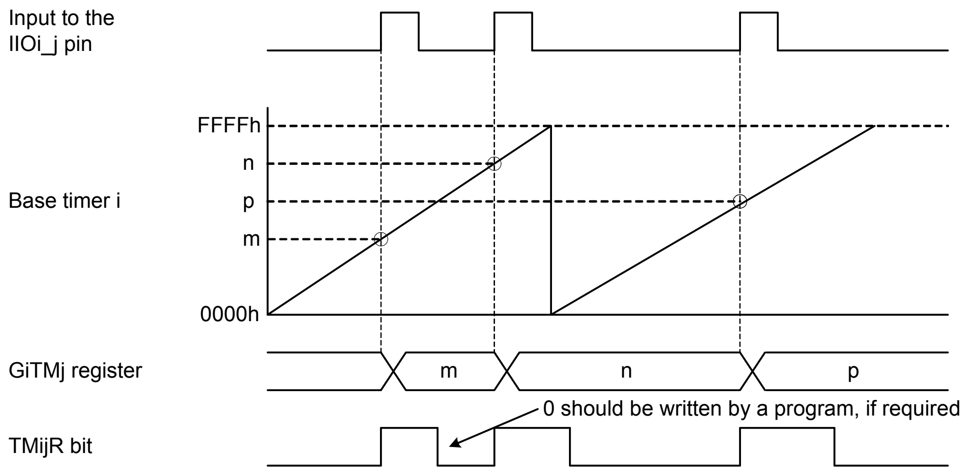
Table 22.4 Time Measurement Specifications (i = 0, 1; j = 0 to 7)

Item	Specification
Time measurement channels	Group 0: Channels 0 to 7 Group 1: Channels 0 to 7
Trigger input polarity	Rising edge, falling edge, or both edges of the IIOi_j pin
Time measurement start condition	The IFEj bit in the GiFE register is set to 1 (enable the channel j function) while the FSCj bit in the GiFS register is set to 1 (select the time measurement)
Time measurement stop condition	The IFEj bit is set to 0 (disable the channel j function)
Time measurement timing	• Without the prescaler: every time a trigger is input • With the prescaler (for channels 6 and 7): every (GiTPRk register (k = 6, 7) value + 1) times a trigger is input
Interrupt request	When the TMijR bit in the interrupt request register is set to 1 (interrupt requested) (refer to Figure 10.15)
IIOi_j input pin function	Trigger input
Selectable functions	• Digital filter The digital filter determines a trigger input level every f1 or fBTi cycle and passes the signals holding the same level during three sequential cycles • Prescaler (for channels 6 and 7) Time measurement is executed every (GiTPRk register value + 1) times a trigger is input • Gating (for channels 6 and 7) This function disables any trigger input to be accepted after the time measurement by the first trigger input. However, the trigger input can be accepted again if any of following conditions are met while the GOC bit in the GiTMCRk register is set to 1 (the gating is cleared when the base timer matches the GiPOp register) (p = 4, 5; p = 4 when k = 6; p = 5 when k = 7): • The base timer value matches the GiPOp register setting • The GSC bit in the GiTMCRk register is set to 1

Table 22.5 Time Measurement Associated Register Settings (i = 0, 1; j = 0 to 7; k = 6, 7)

Register	Bits	Function
GiTMCRj	CTS1 and CTS0	Select a time measurement trigger
	DF1 and DF0	Select a digital filter
	GT, GOC, GSC	Select if the gating is used
	PR	Select if the prescaler is used
GiTPRk	—	Set the prescaler value
GiFS	FSCj	Set the bit to 1 (select the time measurement)
GiFE	IFEj	Set the bit to 1 (enable the channel j function)

Bit configurations and functions vary with channels and groups.
Registers associated with the time measurement should be set after setting the base timer-associated registers.



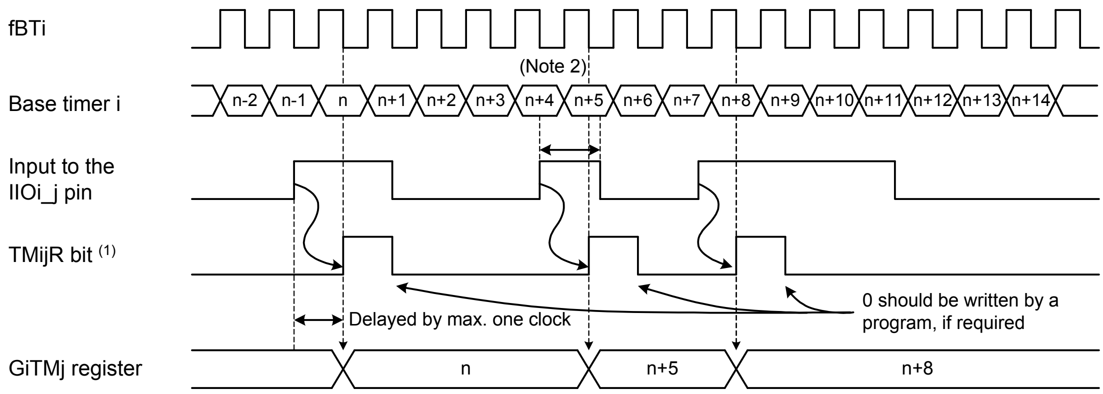
TMijR: Bits in registers IIO0IR to IIO11IR

This figure above applies under the following conditions:
- Bits CTS1 and CTS0 in the GiTMCRj register are set to 01b (rising edge as time measurement trigger), the PR bit is set to 0 (prescaler disabled) and the GT bit is set to 0 (gating disabled)
- Bits RST2 to RST0 in the GiBCR1 register are set to 000b (reset the base timer) and bits UD1 and UD0 are set to 00b (increment mode)

The base timer reset by matching the GiPO0 register setting (bits RST2 to RST0 in the GiBCR1 register are set to 010b) is done after the base timer reaches the GiPO0 register setting value + 2.

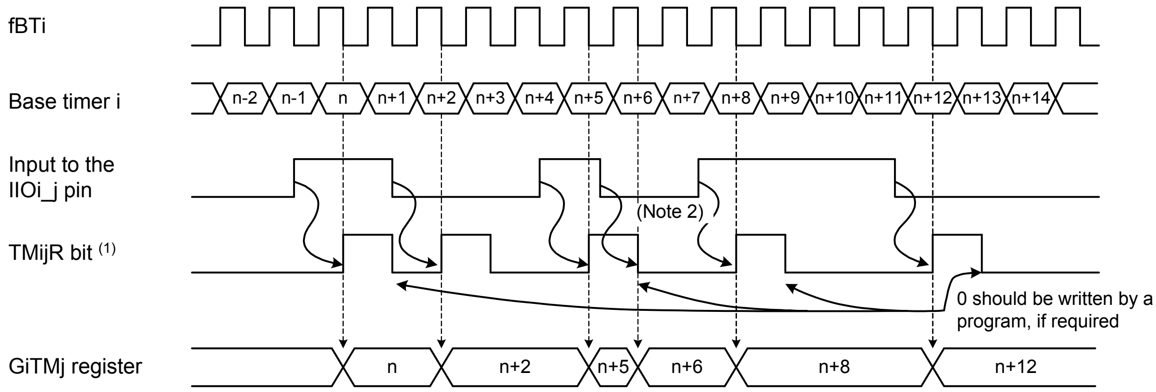
Figure 22.19 Time Measurement Operation (1/2) (i = 0, 1; j = 0 to 7)

Case 1: When selecting the rising edge as a time measurement trigger
(bits CTS1 and CTS0 in the GiTMCRj register are set to 01b)



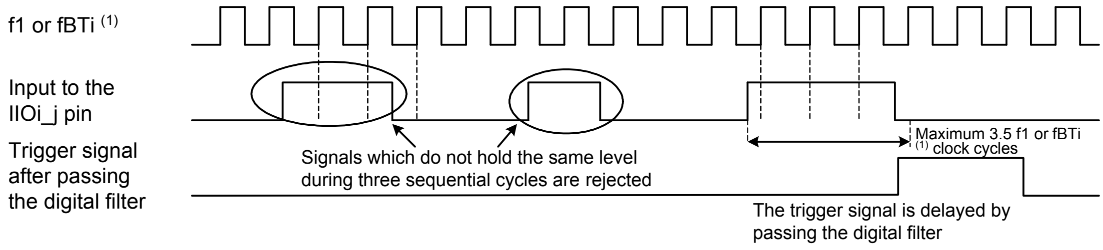
Notes:
1. Bits in registers IIO0IR to IIO11IR.
2. Input pulse applied to the IIOi_j pin requires at least 1.5 fBTi clock cycles.

Case 2: When selecting both edges as a time measurement trigger
(bits CTS1 and CTS0 in the GiTMCRj register are set to 11b)



Notes:
1. Bits in registers IIO0IR to IIO11IR.
2. No interrupt is generated if the MCU receives a trigger signal when the TMijR bit is set to 1. However, the value of GiTMj register changes.

Case 3: Trigger signal when using the digital filter
(bits DF1 and DF0 in the GiTMCRj register are set to 10b or 11b)

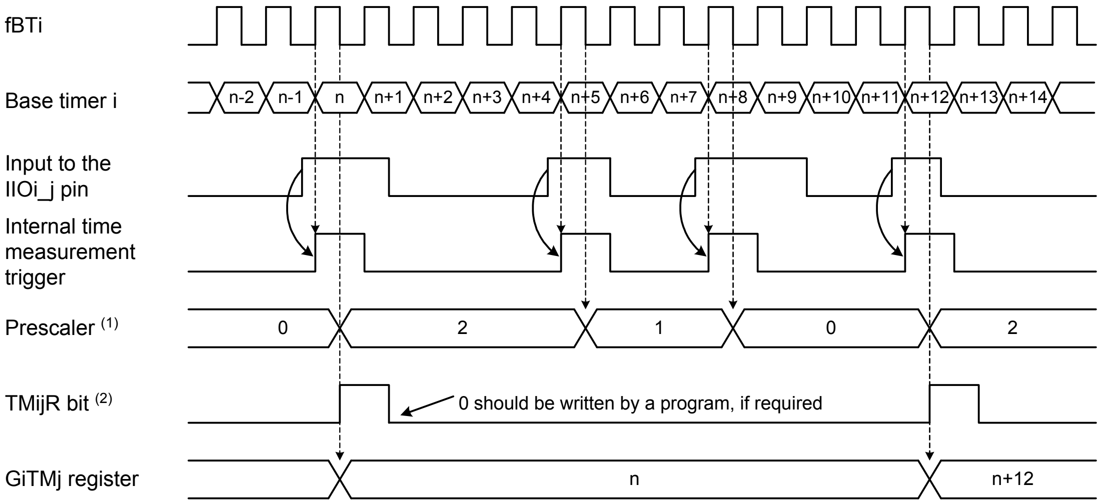


Note:
1. fBTi when bits DF1 and DF0 are set to 10b, f1 when the bits are set to 11b.

Figure 22.20 Time Measurement Operation (2/2) (i = 0, 1; j = 0 to 7)

Case 1: Operation with the prescaler

(the GiTPRj register is set to 02h and the PR bit in the GiTMCRj register is set to 1)

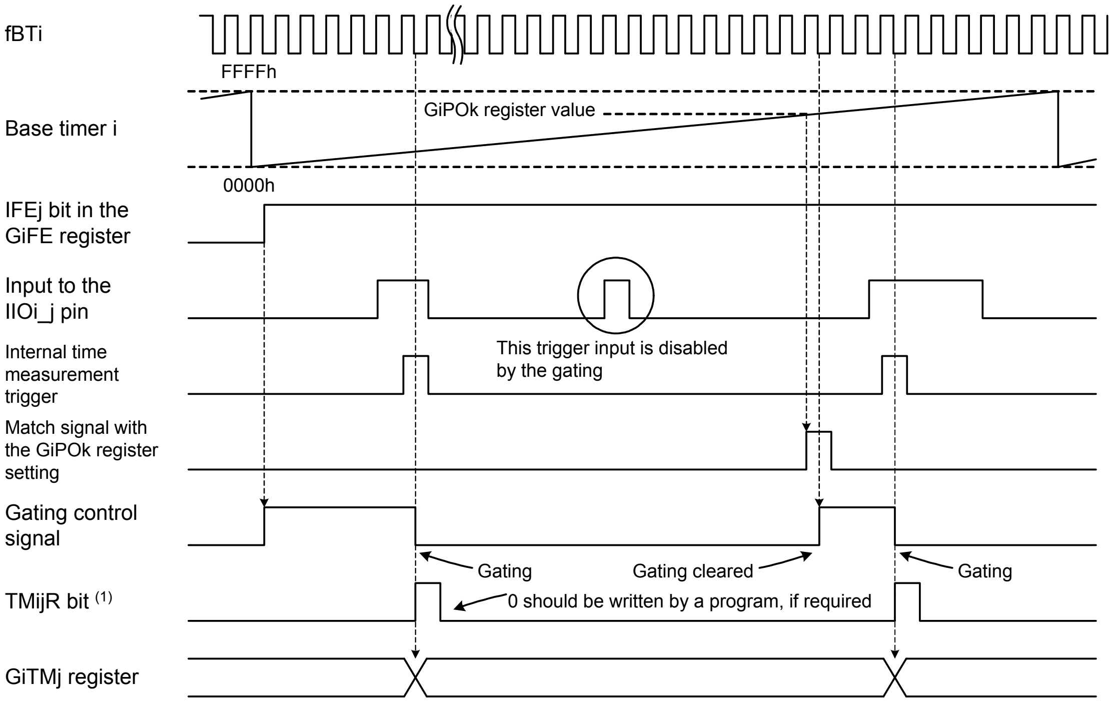


Notes:

1. This example applies to cycles following the first cycle after the PR bit in the GiTMCRj register is set to 1 (prescaler enabled).
2. Bits in registers IIO0IR to IIO11IR.

Case 2: Operation with the gating

(the gating is cleared by matching the base timer value with the GiPOk register setting, and bits GT and GOC in the GiTMCRj register are set to 1, respectively)



Note:

1. Bits in registers IIO0IR to IIO11IR.

Figure 22.21 Prescaler and Gate Operations (i = 0, 1; j = 6, 7; k = 4, 5)

22.3 Waveform Generation

Waveforms are generated when the base timer value matches the GiPOj register setting (i = 0, 1; j = 0 to 7).

Waveform generation has the following four modes:

- Single-phase waveform output mode
- Inverted waveform output mode
- Set/reset waveform output (SR waveform output) mode
- Phase shift waveform output mode

Table 22.6 lists registers associated with the waveform generation.

Table 22.6 Waveform Generation Associated Register Settings (i = 0, 1; j = 0 to 7)

Register	Bits	Function
GiPOCRj	MOD2 to MOD0	Select a waveform output mode
	IVL	Select a default value
	RLD	Select a timing to reload the value into the GiPOj register
	INV	Select if output level is inverted
GiPOj	—	Set the timing to invert output waveform level
GiFS	FSCj	Set the bit to 0 (select the waveform generation)
GiFE	IFEj	Set the bit to 1 (enable the channel j function)

Bit configurations and functions vary with channels and groups.

Registers associated with the waveform generation should be set after setting the base timer-associated registers.

22.3.1 Single-phase Waveform Output Mode

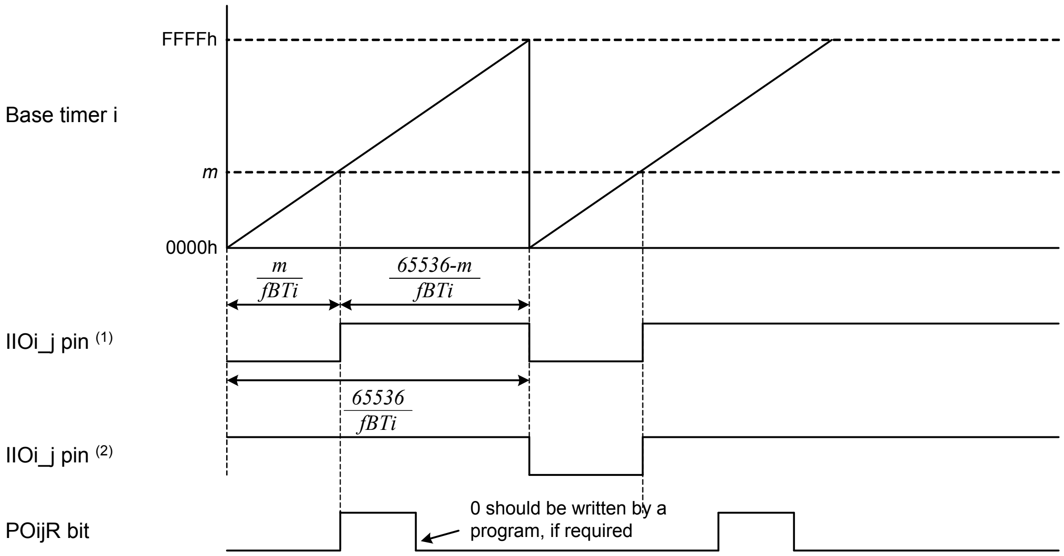
The output level at the IIOi_j pin becomes high when the base timer value matches the GiPOj register (i = 0, 1; j = 0 to 7). It switches to low when the base timer reaches 0000h. If the IVL bit in the GiPOCRj register is set to 1 (output high as default value), a high level output is provided when a waveform output starts. If the INV bit is set to 1 (invert the output level), a waveform with an inverted level is output. Refer to Figure 22.22 for details on single-phase waveform mode operation.
Table 22.7 lists specifications of single-phase waveform output mode.

Table 22.7 Single-phase Waveform Output Mode Specifications (i = 0, 1)

Item	Specification
Output waveform (1)	• Free-running operation (when bits RST2 to RST0 in the GiBCR1 register are set to 000b) Cycle: $\frac{65536}{fBTi}$ Low level width: $\frac{m}{fBTi}$ High level width: $\frac{65536-m}{fBTi}$ m: GiPOj register (j = 0 to 7) setting value, 0000h to FFFFh • The base timer is reset by matching the base timer value with the GiPO0 register setting (when bits RST2 to RST0 are set to 010b) Cycle: $\frac{n+2}{fBTi}$ Low level width: $\frac{m}{fBTi}$ High level width: $\frac{n+2-m}{fBTi}$ m: GiPOj register (j = 1 to 7) setting value, 0000h to FFFFh n: GiPO0 register setting value, 0001h to FFFDh If $m \geq n$ + 2, the output level is fixed to low
Waveform output start condition (2)	The IFEj (j = 0 to 7) bit in the GiFE register is set to 1 (enable the channel j function)
Waveform output stop condition	The IFEj bit is set to 0 (disable the channel j function)
Interrupt request	When the POijR bit in the intelligent I/O interrupt request register is set to 1 (interrupt requested) by matching the base timer value with the GiPOj register setting (refer to Figure 10.15)
IIOi_j output pin function	Pulse signal output
Selectable functions	• Default value setting This function determines the starting waveform output level • Output level inversion This function inverts the waveform output level and outputs the inverted signal from the IIOi_j pin

Notes:
1. When the INV bit in the GiPOCRj register is set to 1 (invert the output level), the high and low widths are inverted.
2. To use channels shared by time measurement and waveform generation, the FSCj bit in the GiFS register should be set to 0 (select the waveform generation).

Case 1: Free-running operation (bits RST2 to RST0 in the GiBCR register are set to 000b)



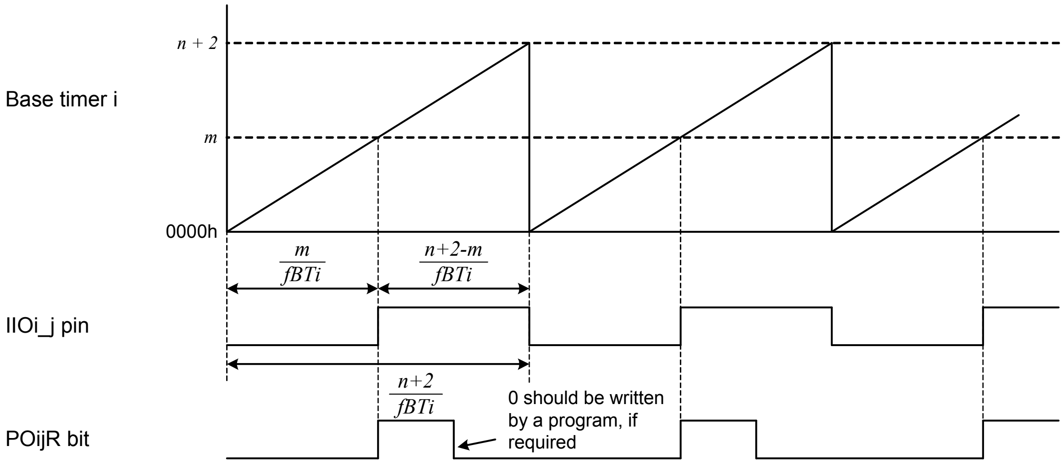
j = 0 to 7
m: GiPOj register setting value (0000h to FFFFh)
POijR: Bits in registers IIO0IR to IIO11IR

Notes:

1. Output waveform when the INV bit in the GiPOCRj register is set to 0 (do not invert the output level) and the IVL bit is set to 0 (output low as default value).
2. Output waveform when the INV bit is set to 0 (do not invert the output level) and the IVL bit is set to 1 (output high as default value).

This figure above applies under the following condition:
- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)

Case 2: The base timer is reset by matching the base timer value with the GiPO0 register setting (bits RST2 to RST0 in the GiBCR register are set to 010b)



j = 1 to 7
m: GiPOj register setting value (0000h to FFFFh)
n: GiPO0 register setting value (0001h to FFFDh)
POijR: Bits in registers IIO0IR to IIO11IR

This figure above applies under the following conditions:
- The IVL bit in the GiPOCRj register is set to 0 (output low as default value) and the INV bit is set to 0 (do not invert the output level)
- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)
- $m < n + 2$

Figure 22.22 Single-phase Waveform Output Mode Operation (i = 0, 1)

22.3.2 Inverted Waveform Output Mode

The output level at the IIOi_j pin is inverted every time the base timer value matches the GiPOj register setting (i = 0, 1; j = 0 to 7).

Table 22.8 lists specifications of the inverted waveform output mode. Figure 22.23 shows an example of the inverted waveform output mode operation.

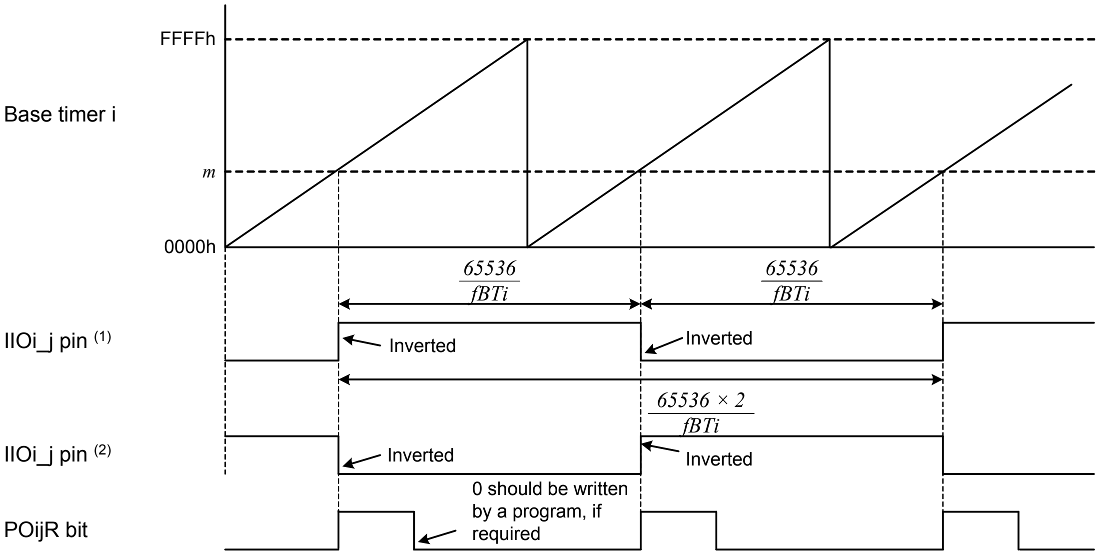
Table 22.8 Inverted Waveform Output Mode Specifications (i = 0, 1)

Item	Specification
Output waveform	• Free-running operation (when bits RST2 to RST0 in the GiBCR1 register are set to 000b) Cycle: $\frac{65536 \times 2}{fBTi}$ High or low level width: $\frac{65536}{fBTi}$ m: GiPOj register (j = 0 to 7) setting value, 0000h to FFFFh • The base timer is reset by matching the base timer value with the GiPO0 register setting (when bits RST2 to RST0 are set to 010b) Cycle: $\frac{2(n+2)}{fBTi}$ High or low level width: $\frac{n+2}{fBTi}$ n: GiPO0 register setting value, 0001h to FFFDh GiPOj register (j = 1 to 7) setting value, 0000h to FFFFh If the GiPOj register setting $\geq n$ + 2, the output level is not inverted
Waveform output start condition (1)	The IFEj bit in the GiFE register (j = 0 to 7) is set to 1 (enable the channel j function)
Waveform output stop condition	The IFEj bit is set to 0 (disable the channel j function)
Interrupt request	When the POijR bit in the intelligent I/O interrupt request register is set to 1 (interrupt requested) by matching the base timer value with the GiPOj register setting (refer to Figure 10.15)
IIOi_j output pin function	Pulse signal output
Selectable functions	• Default value setting This function determines the starting waveform output level • Output level inversion This function inverts the waveform output level and outputs the inverted signal from the IIOi_j pin

Note:

1. To use channels shared by time measurement and waveform generation, the FSCj bit in the GiFS register should be set to 0 (select the waveform generation).

Case 1: Free-running operation (bits RST2 to RST0 in the GiBCR1 register are set to 000b)



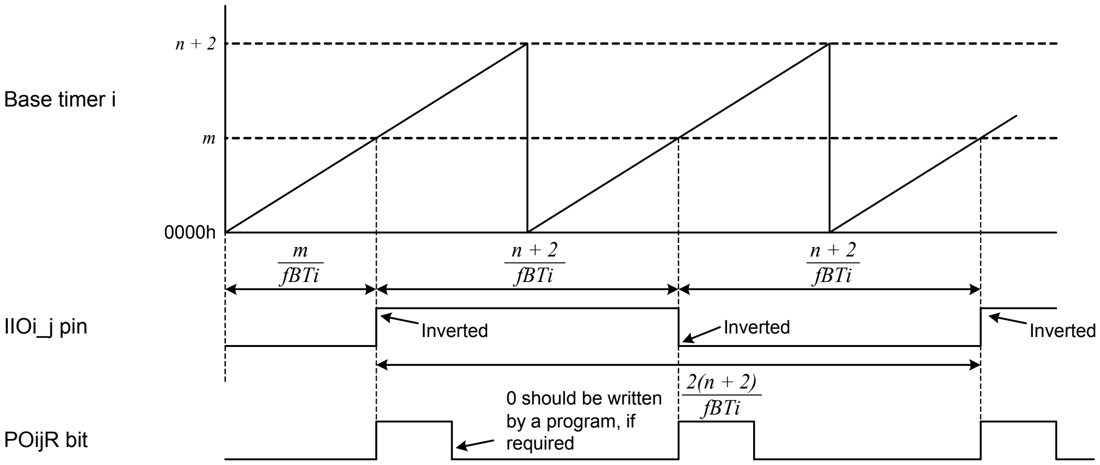
j = 0 to 7
m: GiPOj register setting value (0000h to FFFFh)
POijR: Bits in registers IIO0IR to IIO11IR

Notes:

1. Output waveform when the INV bit in the GiPOCRj register is set to 0 (do not invert the output level) and the IVL bit is set to 0 (output low as default value).
2. Output waveform when the INV bit is set to 0 (do not invert the output level) and the IVL bit is set to 1 (output high as default value).

This figure above applies under the following condition:
- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)

Case 2: The base timer is reset by matching the base timer value with the GiPO0 register setting (bits RST2 to RST0 are set to 010b)



j = 1 to 7
m: GiPOj register setting value (0000h to FFFFh)
n: GiPO0 register setting value (0001h to FFFDh)
POijR: Bits in registers IIO0IR to IIO11IR

This figure above applies under the following conditions:
- The IVL bit in the GiPOCRj register is set to 0 (output low as default value) and the INV bit is set to 0 (do not invert the output level)
- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)
- $m < n + 2$

Figure 22.23 Inverted Waveform Output Mode Operation (i = 0, 1)

22.3.3 Set/Reset Waveform Output Mode (SR Waveform Output Mode)

The output level at the IIOi_j pin becomes high when the base timer value matches the GiPOj register setting (i = 0, 1; j = 0, 2, 4, 6). It becomes low when the base timer value matches the GiPOk register setting (k = j + 1) or the base timer reaches 0000h. When the IVL bit in the GiPOCRj register (j = 0 to 7) is set to 1 (output high as default value), a high output level is provided when a waveform output starts. When the INV bit is set to 1 (invert the output level), a waveform with inverted level is output. Refer to Figure 22.24 for details on SR waveform mode operation. Table 22.9 and Table 22.10 list specifications of SR waveform output mode.

Table 22.9 SR Waveform Output Mode Specifications (i = 0, 1) (1/2)

Item	Specification
Output waveform (1)	• Free-running operation (when bits RST2 to RST0 in the GiBCR1 register are set to 000b) (1) $m < n$ High level width: $\frac{n-m}{fBTi}$ Low level width: $\frac{m}{fBTi}$ (See Note 2) + $\frac{65536-n}{fBTi}$ (See Note 3) (2) $m \geq n$ High level width: $\frac{65536-m}{fBTi}$ Low level width: $\frac{m}{fBTi}$ m: GiPOj register (j = 0, 2, 4, 6) setting value, 0000h to FFFFh n: GiPOk register (k = j + 1) setting value, 0000h to FFFFh • The base timer is reset by matching the base timer value with the GiPO0 register setting (when bits RST2 to RST0 are set to 010b) (4) (1) $m < n < p + 2$ High level width: $\frac{n+m}{fBTi}$ Low width: $\frac{m}{fBTi}$ (See Note 2) + $\frac{p+2-n}{fBTi}$ (See Note 3) (2) $m < p + 2 \leq n$ High level width: $\frac{p+2-m}{fBTi}$ Low level width: $\frac{m}{fBTi}$ (3) $m \geq p + 2$, output level is fixed to low p: GiPO0 register setting value, 0001h to FFFDh m: GiPOj register (j = 2, 4, 6) setting value, 0000h to FFFFh n: GiPOk register (k = j + 1) setting value, 0000h to FFFFh

Notes:

1. When the INV bit in the GiPOCRj register is set to 1 (invert the output level), the high and low widths are inverted.
2. Output period from a base timer reset until when the output level becomes high.
3. Output period from when the output level becomes low until the next base timer reset.
4. When the GiPO0 register resets the base timer, channel 0 and channel 1 SR waveform generation functions are not available.

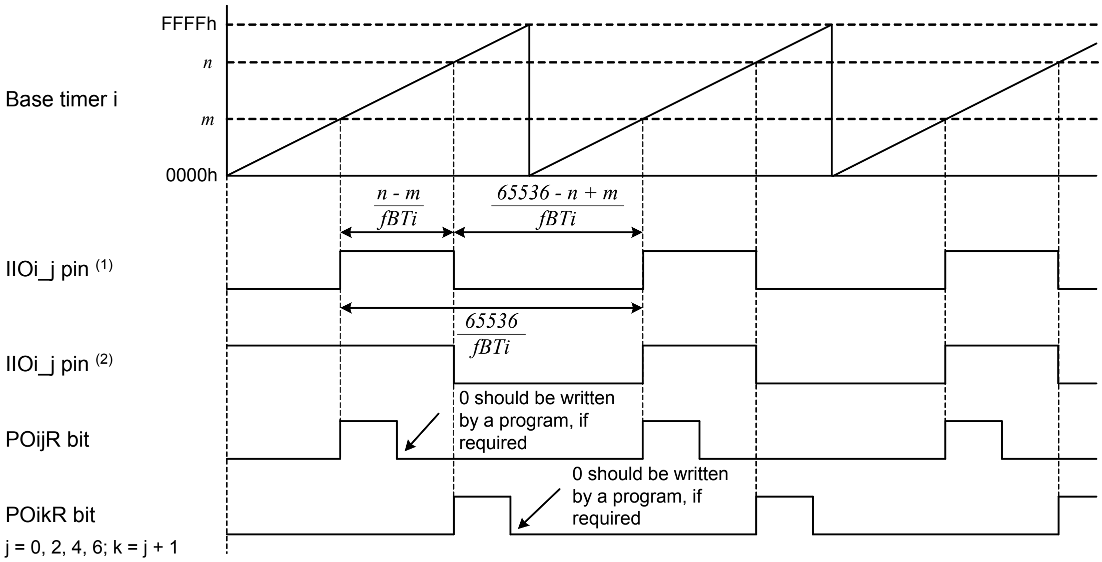
Table 22.10 SR Waveform Output Mode Specifications (i = 0, 1) (2/2)

Item	Specification
Waveform output start condition (1)	The IFEq bit (q = 0 to 7) in the GiFE register is set to 1 (enable the channel q function)
Waveform output stop condition	The IFEq bit is set to 0 (disable the channel q function)
Interrupt request	When the POijR bit in the intelligent I/O interrupt request register is set to 1 (interrupt requested) by matching the base timer value with the GiPOj register setting. When the POikR bit is set to 1 (interrupt requested) by matching the base timer value with the GiPOk register setting (refer to Figure 10.15)
IIOi_j output pin function	Pulse signal output
Selectable functions	• Default value setting This function determines the starting waveform output level • Output level inversion This function inverts the waveform output level and outputs the inverted signal from the IIOi_j pin

Note:

1. To use channels shared by time measurement and waveform generation, the FSCj bit in the GiFS register should be set to 0 (select the waveform generation).

Case 1: Free-running operation (bits RST2 to RST0 in the GiBCR1 register are set to 000b)



j = 0, 2, 4, 6; k = j + 1

m: GiPOj register setting value, 0000h to FFFFh

n: GiPOk register setting value, 0000h to FFFFh

POijR and POikR: Bits in registers IIO0IR to IIO11IR

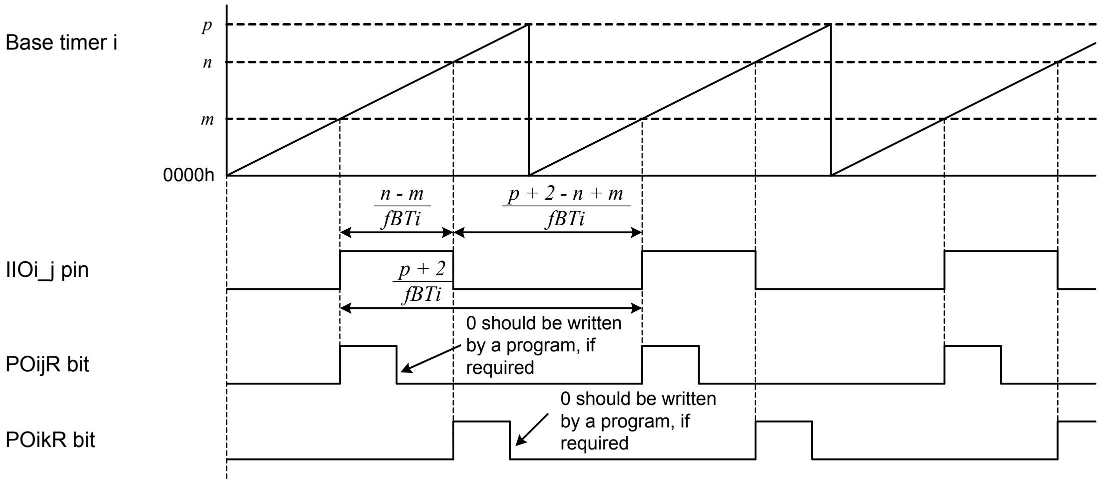
Notes:

1. Output waveform when the INV bit in the GiPOCRj register is set to 0 (do not invert the output level) and the IVL bit is set to 0 (output low as default value).
2. Output waveform when the INV bit is set to 0 (do not invert the output level) and the IVL bit is set to 1 (output high as default value).

This figure above applies under the following conditions:

- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)
- $m < n$

Case 2: The base timer is reset by matching the base timer value with the GiPO0 register setting (bits RST2 to RST0 in the GiBCR1 register are set to 010b)



j = 2, 4, 6; k = j + 1

m: GiPOj register setting value, 0000h to FFFFh

n: GiPOk register setting value, 0000h to FFFDh

p: GiPO0 register setting value, 0001h to FFFDh

POijR and POikR: Bits in registers IIO0IR to IIO11IR

This figure above applies under the following conditions:

- The IVL bit is set to 0 (output low as default value) and the INV bit is set to 0 (do not invert the output level).
- Bits UD1 and UD0 in the GiBCR1 register are set to 00b (increment mode)
- $m < n < p + 2$

Figure 22.24 SR Waveform Output Mode Operation (i = 0, 1)

22.3.4 Phase Shift Waveform Output Mode

PWM output is phase-shifted at every channel in this mode. This mode enables functions that help to reduce switching noise and instantaneous power consumption.

The following bit settings are necessary to enable this mode; set bits MOD2 to MOD0 in the GiPOCRj register (i = 0, 1; j = 0 to 7) to 000b (single-phase waveform output mode), set the IVL bit to 0 (output low as default value), set the BTRE bit to 1 (reset the base timer when bit 9 overflows), set the INV bit to 1 (invert the output level), and set the IT bit in the GiBCR0 register to 0 (interrupt is generated when bit 15 or bit 9 overflows).

When the phase shift waveform output mode is disabled, the output level at the IIOi_j pin becomes low if the base timer matches the GiPOj register, and becomes high when the base timer reaches 0000h (refer to Case 1 in Figure 22.25).

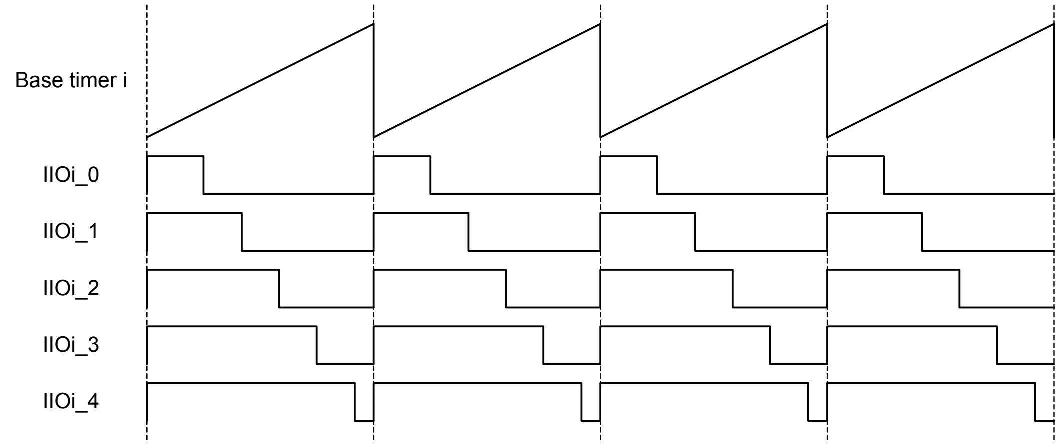
When this mode is enabled, the phase shifter controls when the output level becomes high. The phase shift clock to be input to the phase shifter is the count source for the base timer fBTi divided by m + 1. Every time the phase shift clock is input, each output level of the IIOi_j pin starting from the IIOi_0 sequentially becomes high (refer to Case 2 in Figure 22.25). The base timer overflow interrupt signal BTiR controls reloading of the divided-by-m + 1 divider and reset of the phase shifter.

Figure 22.26 shows some more examples of output waveforms using different settings.

Note that if all channels are set to phase shift waveform output mode and the phase shift clock frequency is set to the lowest, the phase shift clock for channel 7 is not generated. Consequently, the output level at the IIOi_7 pin is held low.

When phase shift output mode is enabled, the output level of a pin cannot be fixed to low. To hold the level low, the corresponding pin should be switched to the port by the output function select register.

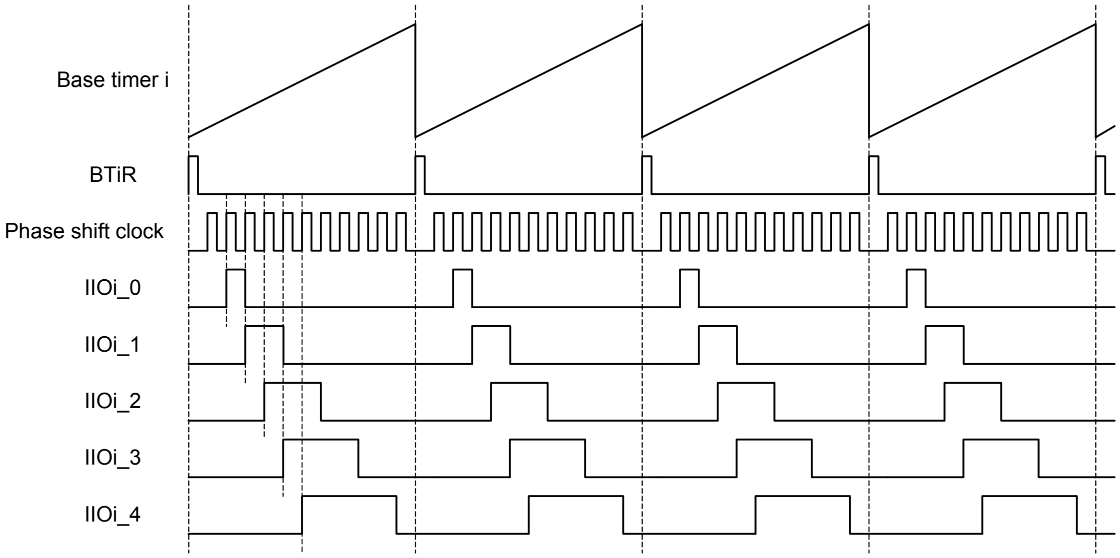
Case 1: The phase shift waveform output mode is disabled



This figure above applies under the following conditions:

- The PSME bit in the GiPSCR register is set to 0 (phase shift waveform output mode is disabled)
- Bits MOD2 to MOD0 in the GiPOCRj register (j = 0 to 4) are set to 000b (single-phase waveform output mode), the IVL bit is set to 0 (output low as default value), and the INV bit is set to 1 (invert the output level)
- Bits RST2 to RST0 in the GiBCR1 register are set to 000b (base timer is not reset) and bits UD1 and UD0 are set to 00b (increment mode)

Case 2: The phase shift waveform output mode is enabled



This figure above applies under the following conditions:

- The PSME bit in the GiPSCR register is set to 1 (phase shift waveform output mode is enabled) and bits PSS1 to PSS0 are set to 00b (pins IIOi_0 to IIOi_4 are available)
- Bits MOD2 to MOD0 in the GiPOCRj register (j = 0 to 4) are set to 000b (single-phase waveform output mode), the IVL bit is set to 0 (output low as default value), the BTRE bit is set to 1 (reset the base timer when bit 9 overflows), and the INV bit is set to 1 (invert the output level)
- The IT bit in the GiBCR0 register is set to 0 (interrupt is generated when the bit 15 or the bit 9 overflows)
- Bits RST2 to RST0 in the GiBCR1 register are set to 000b (base timer is not reset) and bits UD1 and UD0 are set to 00b (increment mode)

Figure 22.25 Phase Shift Waveform Output Mode (1/2) (i = 0, 1)

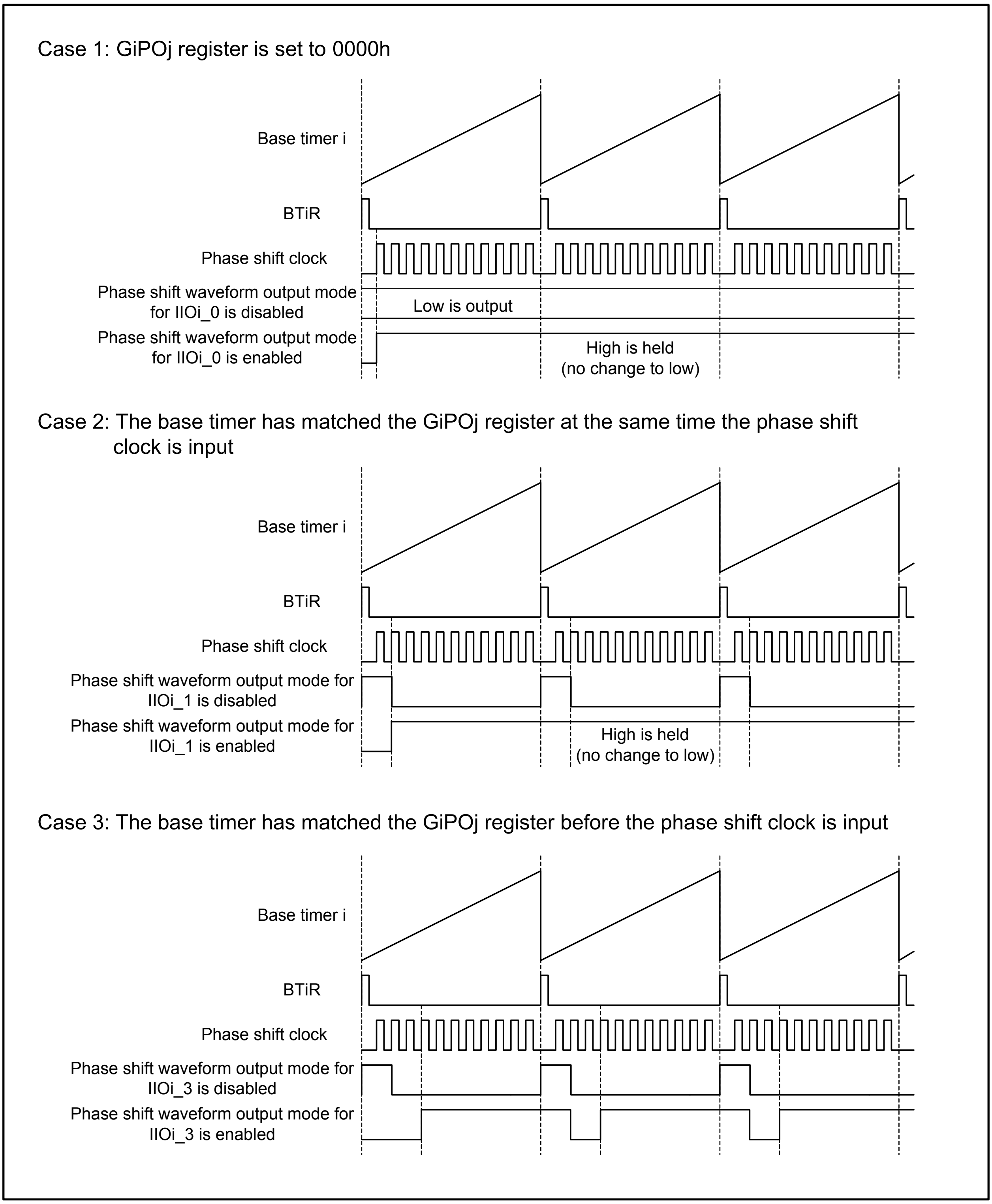


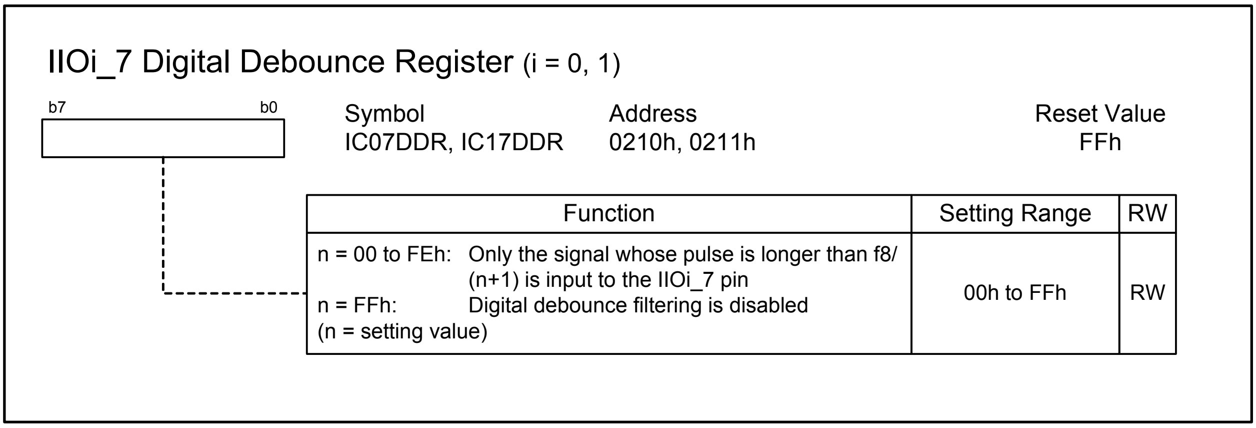
Figure 22.26 Phase Shift Waveform Output Mode (2/2) (i = 0, 1)

22.3.5 Digital Debounce Circuit

Pins IIO0_7 and IIO1_7 have a digital debounce function for noise reduction. This function helps to determine the signal level when the pulse becomes longer than the filter width set by a program after the signal is input on a rising or falling edge.

This function does not affect any signals that share the IIO0_7 or IIO1_7 pin.

Figure 22.27 shows the registers IC07DDR and IC17DDR.



IIOi_7 Digital Debounce Register (i = 0, 1)

Symbol	Address	Reset Value
IC07DDR, IC17DDR	0210h, 0211h	FFh

Function	Setting Range	RW
n = 00 to FEh: Only the signal whose pulse is longer than f8/(n+1) is input to the IIOi_7 pin n = FFh: Digital debounce filtering is disabled (n = setting value)	00h to FFh	RW

Figure 22.27 Registers IC07DDR and IC17DDR

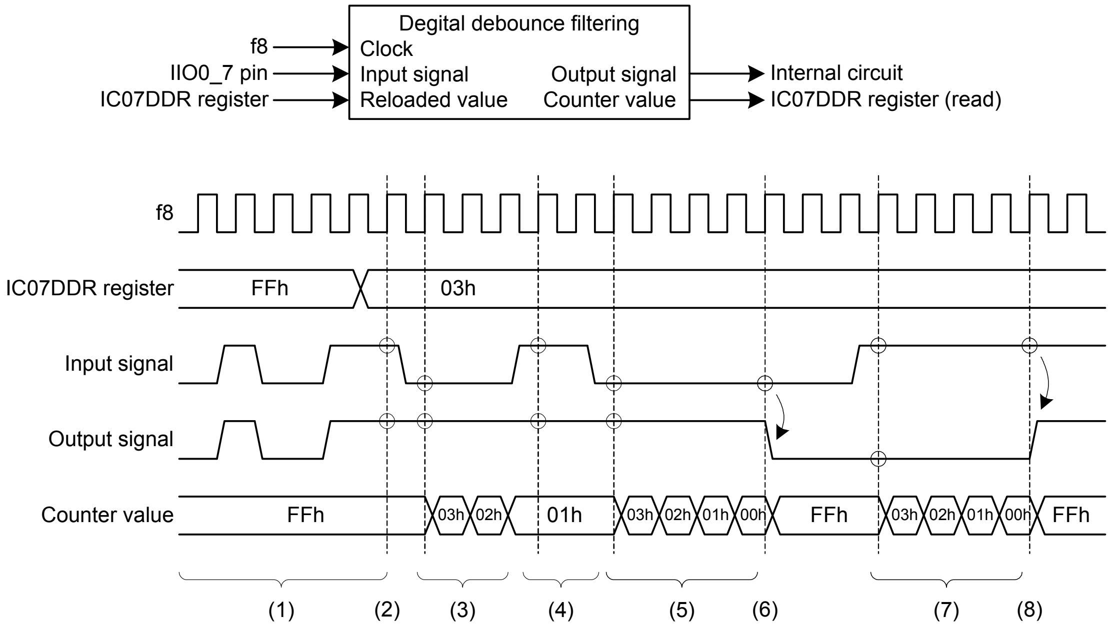
The counter for the digital debounce at the IIOi_7 pin (i = 0, 1) decrements with a count source of f8. Every time the signal level at the pin changes, the setting value of the ICi7DDR register is reloaded to resume counting. The counted value can be read from the ICi7DDR register.

The input signal to the IIOi_7 pin is output from the digital debounce circuit on the first rising edge of f8 after the counter value becomes 00h.

The valid setting values of the ICi7DDR register for digital debounce function are within the 00h to FEh. If the value is set to FFh, the input signal is transmitted without filtering.

Figure 22.28 shows an example of digital debounce filtering.

An example of digital debounce function of IIO0_7 signal (IC07DDR = 03h)



(1) After a reset the IC07DDR register is set to FFh. That is, the filtering is disabled and the input signal is automatically output.

(2) After the IC07DDR register is set to 03h, if the input signal level is identical to that of output, the counter is stopped.

(3) When the input signal level changes, the counter reloads the setting value of the IC07DDR register to start counting.

(4) When the signal level becomes identical again during counting, the counter stops. The output signal holds the same level until the input signal has enough low pulse width so that the counter reaches 00h.

(5) When the input signal level changes again, the counter reloads the setting value of the IC07DDR register to start counting.

(6) The low signal is applied on the first rising edge of f8 after the counter reaches 00h. The output signal becomes low and the counter stops.

(7) When the input signal level changes again, the counter reloads the setting value of the IC07DDR register to start counting.

(8) The high signal is applied on the first rising edge of f8 after the counter reaches 00h. The output signal becomes high and the counter stops.

If the IC07DDR register is set to FFh, the counter value becomes FFh after f1 of 1.5 to 2.5 cycles is input and the counter stops. Simultaneously, the input signal is automatically output without filtering.

Figure 22.28 Digital Debounce Filtering

23. Serial Bus Interface

The serial bus interface has two independent channels: SBI0 and SBI1.
SBIi (i = 0, 1) has its own transmit/receive clock generator.
SBIi has the following modes:

- Synchronous serial communication mode
- 4-wire serial bus mode

To select a mode, set the SSUMS bit in the SSiMR2 register.
Note that each mode has its own bit names, bit symbols, and bit functions in some registers.

23.1 Synchronous Serial Communication Mode and 4-wire Serial Bus Mode

To switch between synchronous serial communication mode and 4-wire serial bus mode, set the SSUMS bit in the SSiMR2 register (i = 0, 1).

Table 23.1 and Figure 23.1 show the specifications of synchronous serial communication mode and its block diagram, respectively. Table 23.2 and Figure 23.2 show the specifications and block diagram of 4-wire serial bus mode, respectively.

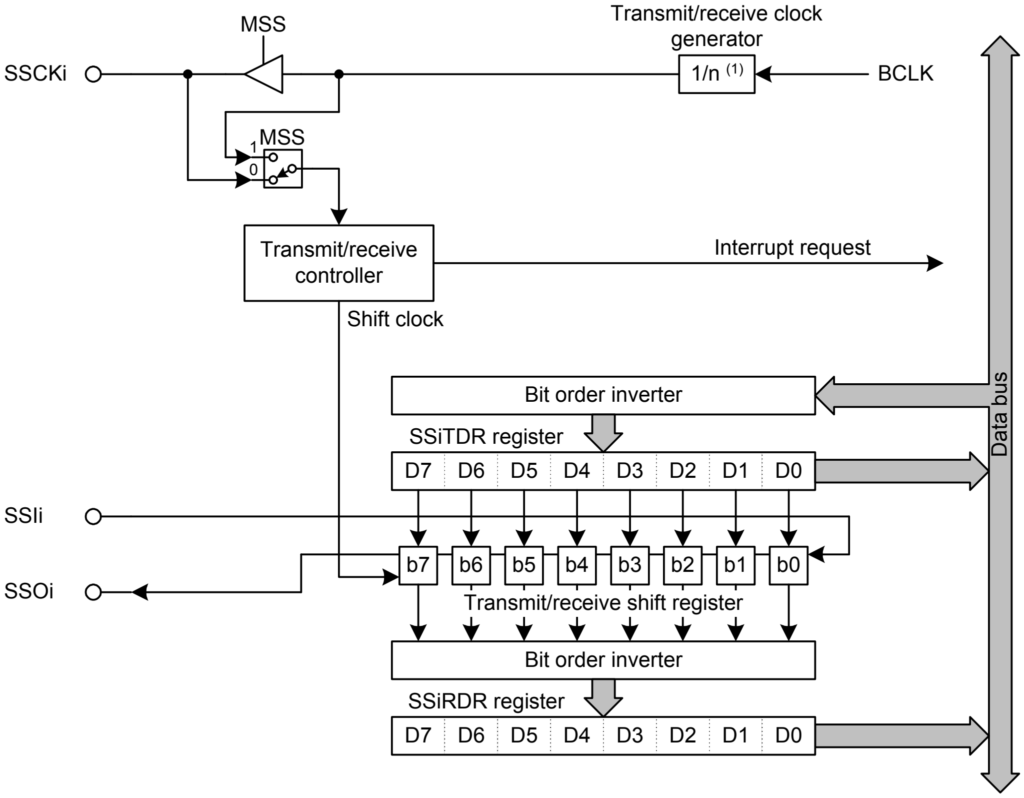
Figure 23.3 to Figure 23.11 show the associated registers.

Table 23.1 Synchronous Serial Communication Mode Specifications (i = 0, 1)

Item	Specification
Data format	Character length: 8 bits (fixed length)
Master/slave mode	Selectable
I/O pins	SSCKi (I/O): Clock I/O pin SSIi (input): Data input pin SSOi (output): Data output pin
Transmit/receive clocks	• When the MSS bit in the SSiCRH register is 0 (slave mode): External clock (input at the SSCKi pin) • When the MSS bit is 1 (master mode): Internal clock (output from the SSCKi pin) $\frac{f_{(BCLK)}}{n}$ $f_{(BCLK)}$: peripheral bus clock frequency n: 4, 8, 16, 32, 64, 128, or 256
Transmit start conditions	• TE bit in the SSiER register is 1 (transmission enabled) • TDRE bit in the SSiSR register is 0 (unsent data left in the SSiTDR register) • ORER bit in the SSiSR register is 0 (no overrun error) • Clock input at the SSCKi pin (applicable only when the MSS bit in the SSiCRH register is 0 (slave mode))
Receive start conditions	• RE bit in the SSiER register is 1 (reception enabled) • ORER bit in the SSiSR register is 0 (no overrun error) • Read operation to the SSiRDR register (applicable only when the MSS bit in the SSiCRH register is 1 (master mode)) • Clock input at the SSCKi pin (applicable only when the MSS bit in the SSiCRH register is 0 (slave mode))
Error detection	• Overrun error An overrun error occurs if, while the RDRF bit in the SSiSR register is 1 (received data left in the SSiRDR register), the next serial data reception is completed
Interrupt request sources	Four interrupt request sources: transmit end, transmit data register empty, receive data register full, and overrun error (1)
Selectable functions	• Bit order select Selectable either LSB first or MSB first • SSCKi clock polarity Selectable either low or high for the SSCKi pin when clock is stopped • SSCKi clock phase Combination of odd and even edges for changing and loading data

Note:

1. Each channel has its own interrupt vector table.



MSS: Bit in the SSiCRH register

Note:

1. Set bits CKS2 to CKS0 in the SSiCRH register to select divide-by-n (n = 4, 8, 16, 32, 64, 128, and 256).

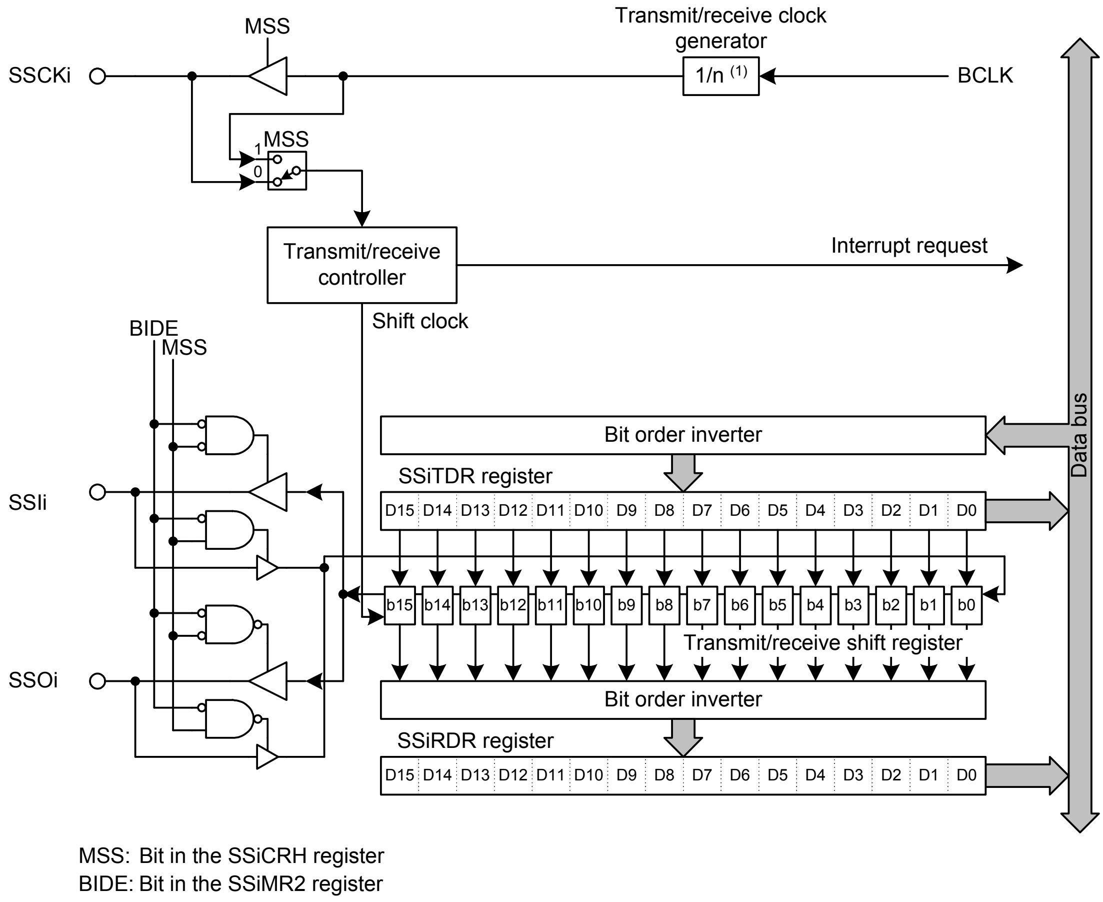
Figure 23.1 Block Diagram of Synchronous Serial Communication Mode (i = 0, 1)

Table 23.2 4-wire Serial Bus Mode Specifications (i = 0, 1)

Item	Specification
Data format	Character length: 8 to 16 bits (variable length)
Master/slave mode	Selectable
I/O pins	SSCKi (I/O): Clock I/O pin SSIi (I/O): Data I/O pin SSOi (I/O): Data I/O pin $\overline{\text{SCSi}}$ (I/O): Chip select I/O pin
Transmit/receive clocks	• When the MSS bit in the SSiCRH register is 0 (slave mode): External clock (input at the SSCKi pin) • When the MSS bit 1 (master mode): Internal clock (output from the SSCKi pin) $\frac{f_{(BCLK)}}{n}$ $f_{(BCLK)}$: peripheral bus clock frequency n: 4, 8, 16, 32, 64, 128, or 256
Transmit start conditions	• TE bit in the SSiER register is 1 (transmission enabled) • TDRE bit in the SSiSR register is 0 (unsent data left in the SSiTDR register) • ORER bit in the SSiSR register is 0 (no overrun error) • Clock input at the SSCKi pin while the $\overline{\text{SCSi}}$ pin is low (applicable only when the MSS bit in the SSiCRH register is 0 (slave mode))
Receive start conditions	• RE bit in the SSiER register is 1 (reception enabled) • ORER bit in the SSiSR register is 0 (no overrun error) • Read operation to the SSiRDR register while the TE bit in the SSiER register is 0 (transmission disabled) (applicable only when the MSS bit in the SSiCRH register is 1 (master mode)) • Clock input at the SSCKi pin (applicable only when the MSS bit in the SSiCRH register is 0 (slave mode))
Error detection	• Overrun error An overrun error occurs if, while the RDRF bit in the SSiSR register is 1 (received data left in the SSiRDR register), the next serial data is received • Conflict error A conflict error occurs in either of the following cases: (1) An attempt to start serial communication while the $\overline{\text{SCSi}}$ pin is low (applicable when the MSS bit in the SSiCRH register is 1 (master mode)) (2) The $\overline{\text{SCSi}}$ pin becomes high during data transfer (applicable when the MSS bit in the SSiCRH register 0 (slave mode))
Interrupt request sources	Five interrupt request sources: transmit end, transmit data register empty, receive data register full, overrun error, and conflict error [1]
Selectable functions	• Bit order select Selectable either LSB first or MSB first • SSCKi clock polarity Selectable either low or high for the SSCKi pin when clock is stopped • SSCKi clock phase Combination of odd and even edges for changing and loading data

Note:

1. Each channel has its own interrupt vector table.



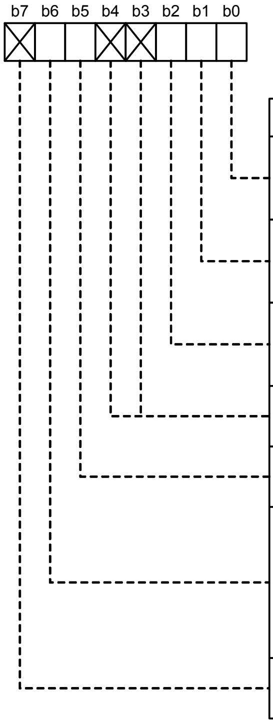
MSS: Bit in the SSiCRH register
BIDE: Bit in the SSiMR2 register

Note:

1. Set bits CKS2 to CKS0 in the SSiCRH register to select divide-by-n (n = 4, 8, 16, 32, 64, 128, or 256).

Figure 23.2 Block Diagram of 4-wire Serial Bus Mode (i = 0, 1)

SSi Control Register H (i = 0, 1)



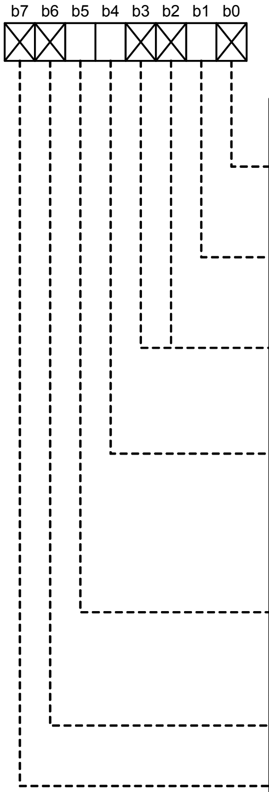
Symbol	Address	Reset Value
SS0CRH, SS1CRH	044F08h, 044F18h	0000 0000b

Bit Symbol	Bit Name	Function	RW
CKS0	Transmit/Receive Clock Rate Select Bit (1)	b2 b1 b0 0 0 0 : f(BCLK) divided-by-256 0 0 1 : f(BCLK) divided-by-128 0 1 0 : f(BCLK) divided-by-64 0 1 1 : f(BCLK) divided-by-32 1 0 0 : f(BCLK) divided-by-16 1 0 1 : f(BCLK) divided-by-8 1 1 0 : f(BCLK) divided-by-4 1 1 1 : Do not use this combination	RW
CKS1			RW
CKS2			RW
— (b4-b3)	No register bits; should be written with 0 and read as 0		—
MSS	Master/Slave Mode Select Bit (2, 3)	0: Slave mode 1: Master mode	RW
RSSTP	Receive Single Stop Bit (4)	After receiving the current frame, 0: Continue receive operation (output the receive clock) 1: End receive operation (stop the receive clock)	RW
— (b7)	No register bit; should be written with 0 and read as 0		—

Notes:
1. The clock rate selection is applicable only when the MSS bit is set to 1 (master mode).
2. To output the transmit/receive clock from the SSCKi pin, set this bit to 1.
3. This bit becomes 0 when the CE bit in the SSiSR register becomes 1 (conflict error).
4. This bit is disabled when the MSS bit is 0 (slave mode).

Figure 23.3 Registers SS0CRH and SS1CRH

SSi Control Register L (i = 0, 1)



Symbol	Address	Reset Value
SS0CRL, SS1CRL	044F09h, 044F19h	0111 1101b

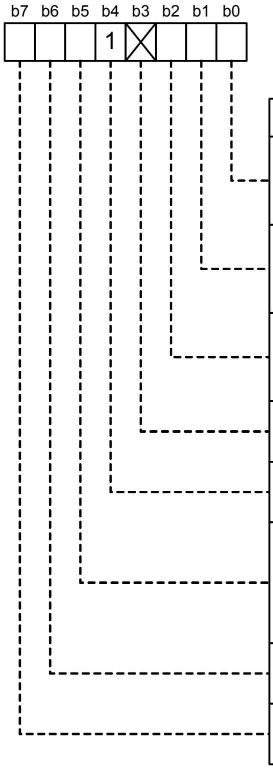
Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit; should be written with 0 and read as 1		—
SRES	Synchronous Serial Interface Controller Reset Bit	Setting to 1 initializes the transmit/ receive controller and the transmit/ receive shift register. The internal register (1) value remains unchanged	RW
— (b3-b2)	No register bits; should be written with 0 and read as 1		—
SOLP	SOL Write Protect Bit (2)	0: Unlock the protection of the SOL bit (SOL bit setting is reflected to output pin.) 1: Output pin level remains unchanged. This bit is read as 1	RW
SOL	Serial Data Output Setting Bit	This bit can be read as follows: 0: Serial data output pin is low 1: Serial data output pin is high This bit can be written as follows (2, 3): 0: Drive serial data output pin low 1: Drive serial data output pin high	RW
— (b6)	No register bit; should be written with 0 and read as 1		—
— (b7)	No register bit; should be written with 0 and read as 0		—

Notes:

1. Registers SSiCRH, SSiCRL, SSiMR (excluding bits BC3 to BC0 in 4-wire serial bus mode), SSiER, SSiSR, SSiMR2, SSiTDR, and SSiRDR.
2. To change the output pin level, the SOL bit should be set before or after transmitting data. Use the MOV instruction to simultaneously set the SOL bit to 0 or 1, and the SOLP bit to 0.
3. This bit should not be set during data transmission.

Figure 23.4 Registers SS0CRL and SS1CRL

SSi Mode Register (i = 0, 1)



Symbol	Address	Reset Value
SS0MR, SS1MR	044F0Ah, 044F1Ah	0001 0000b

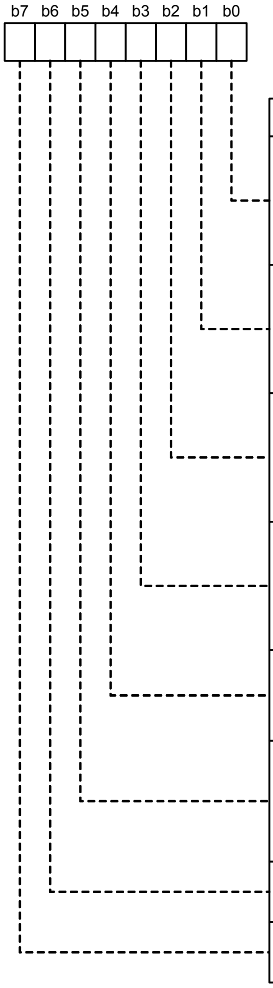
Bit Symbol	Bit Name	Function	RW
BC0	Bit Counter	Bits left during transmission/reception b2 b1 b0 0 0 0 : 8 bits left 0 0 1 : 1 bit left 0 1 0 : 2 bits left 0 1 1 : 3 bits left 1 0 0 : 4 bits left 1 0 1 : 5 bits left 1 1 0 : 6 bits left 1 1 1 : 7 bits left	RO
BC1			RO
BC2			RO
— (b3)	No register bit; should be written with 0 and read as 0		—
— (b4)	Reserved	Should be written with 1	RW
CPHS	SSCKi Clock Phase Select Bit (1)	0: Change data at odd edges (load data at even edges) 1: Change data at even edges (load data at odd edges)	RW
CPOS	SSCKi Clock Polarity Select Bit	0: High when clock is stopped 1: Low when clock is stopped	RW
MLS	Bit Order Select Bit	0: MSB first 1: LSB first	RW

Note:

1. This bit should be set to 0 in synchronous serial communication mode.

Figure 23.5 Registers SS0MR and SS1MR in Synchronous Serial Communication Mode

SSi Mode Register (i = 0, 1)

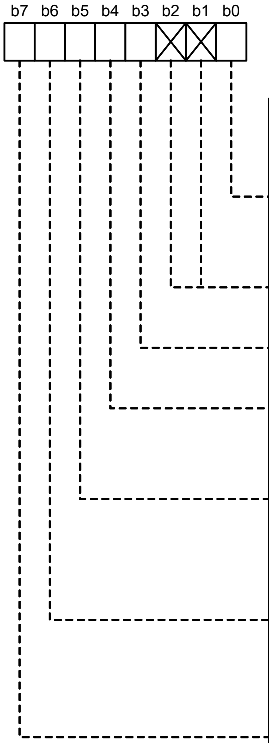


Symbol	Address	Reset Value
SS0MR, SS1MR	044F0Ah, 044F1Ah	0001 0000b

Bit Symbol	Bit Name	Function	RW
BC0	Character Length Setting Bit / Bit Counter	Character length is set./Number of bits left is read. b3 b2 b1 b0 0 0 0 0 : 16-bit/16 bits left 0 0 0 1 : Setting disabled/1 bit left 0 0 1 0 : Setting disabled/2 bits left 0 0 1 1 : Setting disabled/3 bits left 0 1 0 0 : Setting disabled/4 bits left 0 1 0 1 : Setting disabled/5 bits left 0 1 1 0 : Setting disabled/6 bits left 0 1 1 1 : Setting disabled/7 bits left 1 0 0 0 : 8-bit/8 bits left 1 0 0 1 : 9-bit/9 bits left 1 0 1 0 : 10-bit/10 bits left 1 0 1 1 : 11-bit/11 bits left 1 1 0 0 : 12-bit/12 bits left 1 1 0 1 : 13-bit/13 bits left 1 1 1 0 : 14-bit/14 bits left 1 1 1 1 : 15-bit/15 bits left	RW
BC1			RW
BC2			RW
BC3			RW
BCWP	BC Write Protect Bit	Set this bit to 0 at the same time bits BC3 to BC0 are rewritten. This bit is read as 1	RW
CPHS	SSCKi Clock Phase Select Bit	0: Change data at odd edges (load data at even edges) 1: Change data at even edges (load data at odd edges)	RW
CPOS	SSCKi Clock Polarity Select Bit	0: High when clock is stopped 1: Low when clock is stopped	RW
MLS	Bit Order Select Bit	0: MSB first 1: LSB first	RW

Figure 23.6 Registers SS0MR and SS1MR in 4-wire Serial Bus Mode

SSi Enable Register (i = 0, 1)

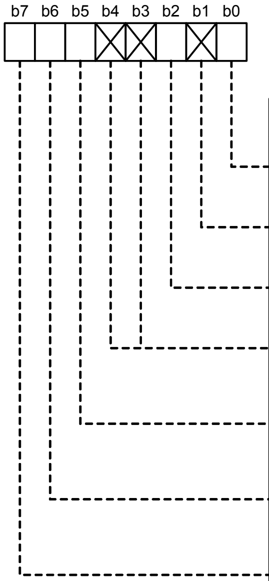


Symbol	Address	Reset Value
SS0ER, SS1ER	044F0Bh, 044F1Bh	0000 0000b

Bit Symbol	Bit Name	Function	RW
CEIE	Conflict Error Interrupt Enable Bit	0: Disable conflict error interrupt request 1: Enable conflict error interrupt request	RW
— (b2-b1)	No register bits; should be written with 0 and read as 0		—
RE	Receive Enable Bit	0: Disable data reception 1: Enable data reception	RW
TE	Transmit Enable Bit	0: Disable data transmission 1: Enable data transmission	RW
RIE	Receive Interrupt Enable Bit	0: Disable receive data register full or overrun error interrupt request 1: Enable receive data register full or overrun error interrupt request	RW
TEIE	Transmit End Interrupt Enable Bit	0: Disable transmit end interrupt request 1: Enable transmit end interrupt request	RW
TIE	Transmit Interrupt Enable Bit	0: Disable transmit data register empty interrupt request 1: Enable transmit data register empty interrupt request	RW

Figure 23.7 Registers SS0ER and SS1ER

SSi Status Register (i = 0, 1)



Symbol	Address	Reset Value
SS0SR, SS1SR	044F0Ch, 044F1Ch	0000 0000b

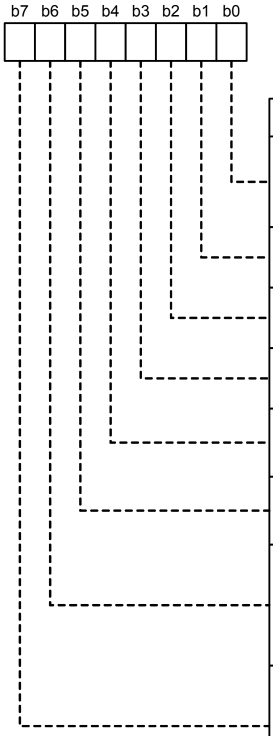
Bit Symbol	Bit Name	Function	RW
CE	Conflict Error Flag (1)	0: No conflict error 1: Conflict error occurred (2)	RW
— (b1)	No register bit; should be written with 0 and read as 0		—
ORER	Overrun Error Flag (1, 3)	0: No overrun error 1: Overrun error occurred	RW
— (b4-b3)	No register bits; should be written with 0 and read as 0		—
RDRF	Receive Data Register Full Flag (1, 4)	0: No data left in the SSiRDR register 1: Received data left in the SSiRDR register	RW
TEND	Transmit End Flag (1, 5)	0: Transmission continues 1: Transmission is completed	RW
TDRE	Transmit Data Register Empty Flag (1, 6)	0: Unsent data left in the SSiTDR register 1: No data left in the SSiTDR register	RW

Notes:

1. This bit is unaffected if a 1 is written. To set this bit to 0, first read the register then write 0 to the bit and write 1 to the rest of the bits.
2. This bit becomes 1 under the following conditions:
   -As an input pin, the $\overline{\text{SCSi}}$ pin is low when the SSUMS bit in the SSiMR2 register is 1 (4-wire serial bus mode), and the MSS bit in the SSiCRH register is 1 (master mode).
   or
   -As an input pin, the $\overline{\text{SCSi}}$ pin becomes high during frame transmission/reception when the SSUMS bit is 1 (4-wire serial bus mode), and the MSS bit is 0 (slave mode).
3. This flag indicates data reception ended unexpectedly because of an overrun error. This bit becomes 1 when the next serial data reception is completed while the RDRF bit is 1. Note that the data reception cannot start while this bit is 1. Data transmission cannot start either, while the MSS bit is 1 (master mode).
4. This bit becomes 0 when data is read from the SSiRDR register.
5. This bit becomes 1 when no data is left to be transmitted in the SSiTDR register before the end of the transmission of the last bit of current data. This bit becomes 0 when data is written to the SSiTDR register.
6. This bit becomes 1 when the TE bit in the SSiER register is switched from 0 (data transmission disabled) to 1 (data transmission enabled). It also becomes 1 when data is transferred from the SSiTDR register to the transmit/receive shift register. This bit becomes 0 when data is written to the SSiTDR register.

Figure 23.8 Registers SS0SR and SS1SR

SSi Mode Register 2 (i = 0, 1)



Symbol	Address	Reset Value
SS0MR2, SS1MR2	044F0Dh, 044F1Dh	0000 0000b

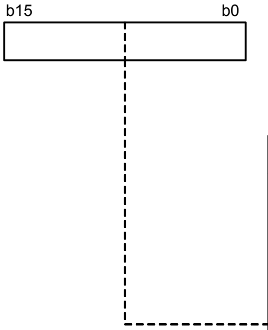
Bit Symbol	Bit Name	Function	RW
SSUMS	Mode Select Bit	0: Synchronous serial communication mode 1: 4-wire serial bus mode	RW
CSOS	$\overline{\text{SCSi}}$ Pin Open Drain Output Select Bit	0: Push-pull output 1: N-channel open drain output	RW
SOOS	Serial Data Open Drain Output Select Bit	0: Push-pull output 1: N-channel open drain output	RW
SCKOS	SSCKi Pin Open Drain Output Select Bit	0: Push-pull output 1: N-channel open drain output	RW
CSS0	$\overline{\text{SCSi}}$ Pin Function Select Bit (1, 2, 3)	b5 b4 0 0 : No function as chip select 0 1 : $\overline{\text{SCSi}}$ pin as input 1 0 : $\overline{\text{SCSi}}$ pin as output 1 1 : $\overline{\text{SCSi}}$ pin as input/output (4)	RW
CSS1			RW
SCKS	SSCKi Pin Function Select Bit (2)	0: No function as I/O pin for transmit/receive clock 1: I/O pin for transmit/receive clock	RW
BIDE	Bidirectional Mode Enable Bit (5)	0: Standard mode (two individual pins for data I/O) 1: Bidirectional mode (one common pin for data I/O)	RW

Notes:

1. Irrespective of the CSS1 and CSS0 bit setting, the $\overline{\text{SCSi}}$ pin function is not applicable when the SSUMS bit is 0 (synchronous serial communication mode).
2. Settings of the port direction register and output function select register are prioritized over a setting of this bit. Set these two registers according to the function selected by the bit(s).
3. To set bits CSS1 and CSS0 to 11b, first, set them to 10b, and then to 11b after the corresponding port function select register is read as 1.
4. This pin functions as output only during data transmission/reception. Otherwise, it functions as input pin.
5. When the SSUMS bit is 0 (synchronous serial communication mode), the BIDE bit setting is ignored and standard mode is always selected.

Figure 23.9 Registers SS0MR2 and SS1MR2

SSi Transmit Data Register (i = 0, 1)

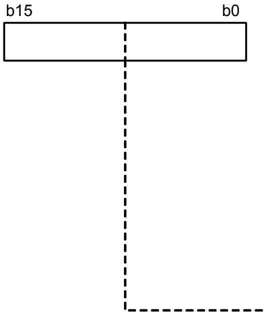


Symbol	Address	Reset Value
SS0TDR	044F0Fh-044F0Eh	FFFFh
SS1TDR	044F1Fh-044F1Eh	FFFFh

Function	RW
This register stores transmit data. Data should be written with the LSB in bit 0. When the character length is 9 bits or longer, write to this register in 16-bit units. When the transmit/receive shift register is detected as being empty, the transmit data stored in this register is transferred to the transmit/receive shift register to start data transmission. If the next transmit data is set to this register while the current data is being transmitted from the transmit/receive shift register, data transmission continues. When the SSUMS bit in the SSiMR2 register is 1 (4-wire serial bus mode) and the MLS bit in the SSiMR register is 1 (LSB first), the MSB-LSB reversed data is read from this register	RW

Figure 23.10 Registers SS0TDR and SS1TDR

SSi Receive Data Register (i = 0, 1)



Symbol	Address	Reset Value
SS0RDR	044F07h-044F06h	FFFFh
SS1RDR	044F17h-044F16h	FFFFh

Function	RW
This register stores received data. (1) When the character length is 9 bits or longer, read this register in 16-bit units. The receive data operation is completed when one frame of data received by the transmit/receive shift register is transferred to this register. The next data reception is enabled simultaneously when one sequence of the receive operation is completed. Continuous data reception is supported by this double buffer structure of the transmit/receive shift register and the SSiRDR register. The read value of bits other than that specified by bits BC3 to BC0 in the SSiMR register is undefined	RO

Note:

1. When the ORER bit in the SSiSR register becomes 1 (overrun error occurred), this register discards the current stored data, but keeps the last stored data.

Figure 23.11 Registers SS0RDR and SS1RDR

23.1.1 Transmit/Receive Clock

To use the serial bus interface in synchronous serial communication mode or 4-wire serial bus mode, set the SCKS bit in the SSiMR2 register to 1 to configure the SSCKi pin to be a transmit/receive clock pin.

(1) To use the serial bus interface in master mode

When the MSS bit in the SSiCRH register is 1 (master mode), select a transmit/receive clock from among seven internal clocks (f(BCLK) divided-by-4, -8, -16, -32, -64, -128, and -256).

Set the corresponding pin to SSCKi output by the output function select register and set its direction to output by the port direction register.

Since the SSCKi pin functions as clock output, the selected transmit/receive clock is output from the SSCKi pin as soon as data transmission/reception starts.

(2) To use the serial bus interface in slave mode

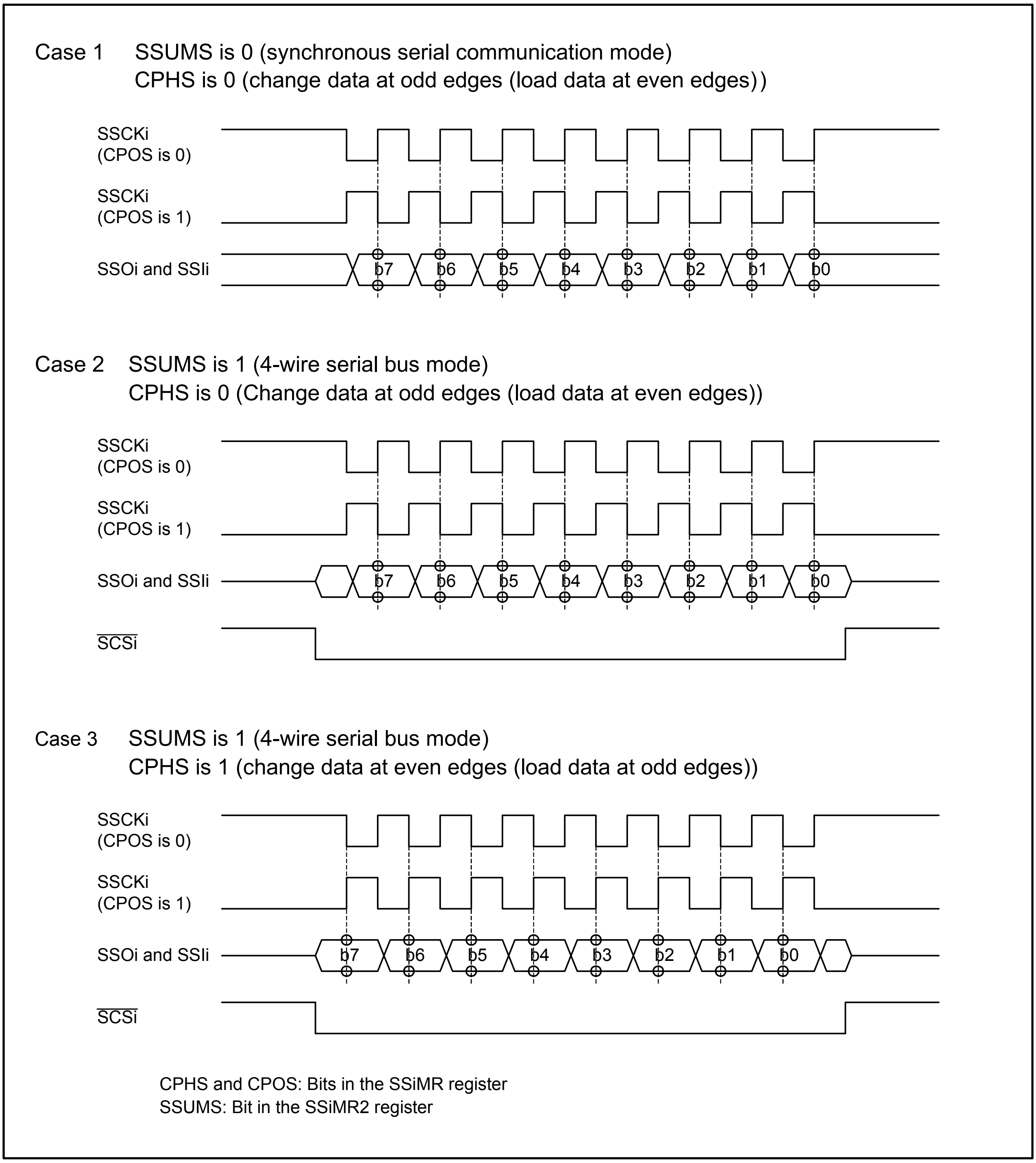
When the MSS bit in the SSiCRH register is 0 (slave mode), the external clock is used.

Set the corresponding pin to input by the port direction register. The SSCKi pin functions as clock input.

23.1.1.1 Transmit/Receive Clock Polarity and Phase

The relation between the transmit/receive clock polarity and phase for transmit/receive data varies with the setting combination of the SSUMS bit in the SSiMR2 register, and bits CPHS and CPOS in the SSiMR register.

Figure 23.12 shows the relation between those parameters.



Case 1 SSUMS is 0 (synchronous serial communication mode)
CPHS is 0 (change data at odd edges (load data at even edges))

Case 2 SSUMS is 1 (4-wire serial bus mode)
CPHS is 0 (Change data at odd edges (load data at even edges))

Case 3 SSUMS is 1 (4-wire serial bus mode)
CPHS is 1 (change data at even edges (load data at odd edges))

CPHS and CPOS: Bits in the SSiMR register
SSUMS: Bit in the SSiMR2 register

Figure 23.12 Transmit/Receive Clock Polarity and Phase of Transmit/Receive Data (i = 0, 1)

23.1.2 Transmit/Receive Shift Register

The transmit/receive shift register transmits and receives serial data. Data transferred to the transmit/receive shift register is shifted-out with the MSB first. Received data is shifted-in with the LSB first.
Figure 23.13 shows transmit/receive shift register operation.

23.1.2.1 Bit Order

The bit order to be transmitted/received can be selected using the MLS bit in the SSiMR register. When the MLS bit is 1 (LSB first), transmission starts from the LSB and completes with the MSB. In a receive operation, the first received bit is considered to be the LSB. When the MLS bit is 0 (MSB first), the bit order transmitted is reversed and the first received bit is considered to be the MSB.

(1) Transmit operation

When the MLS bit is 0 (MSB first), the written value is reflected in the SSiTDR register in the order written.
When the MLS bit is 1 (LSB first), the written value is reflected in the SSiTDR register in the reversed order, that is, the reversed written value is read from the SSiTDR register.
In both cases, however, data is transferred to the transmit/receive shift register in the transmitted order.

(2) Receive operation

When the MLS bit is 0 (MSB first), the received data is transferred from the transmit/receive shift register to the SSiRDR register in the order received.
When the MLS bit is 1 (LSB first), the received data is transferred from the transmit/receive shift register to the SSiRDR register in the reversed order.

23.1.2.2 Variable-Length Data Transmission/Reception

When the data length is less than 16 bits, the shift-in/shift-out position is changed to adjust the bit position.
This function enables transmit/receive data with the LSB in bit 0.

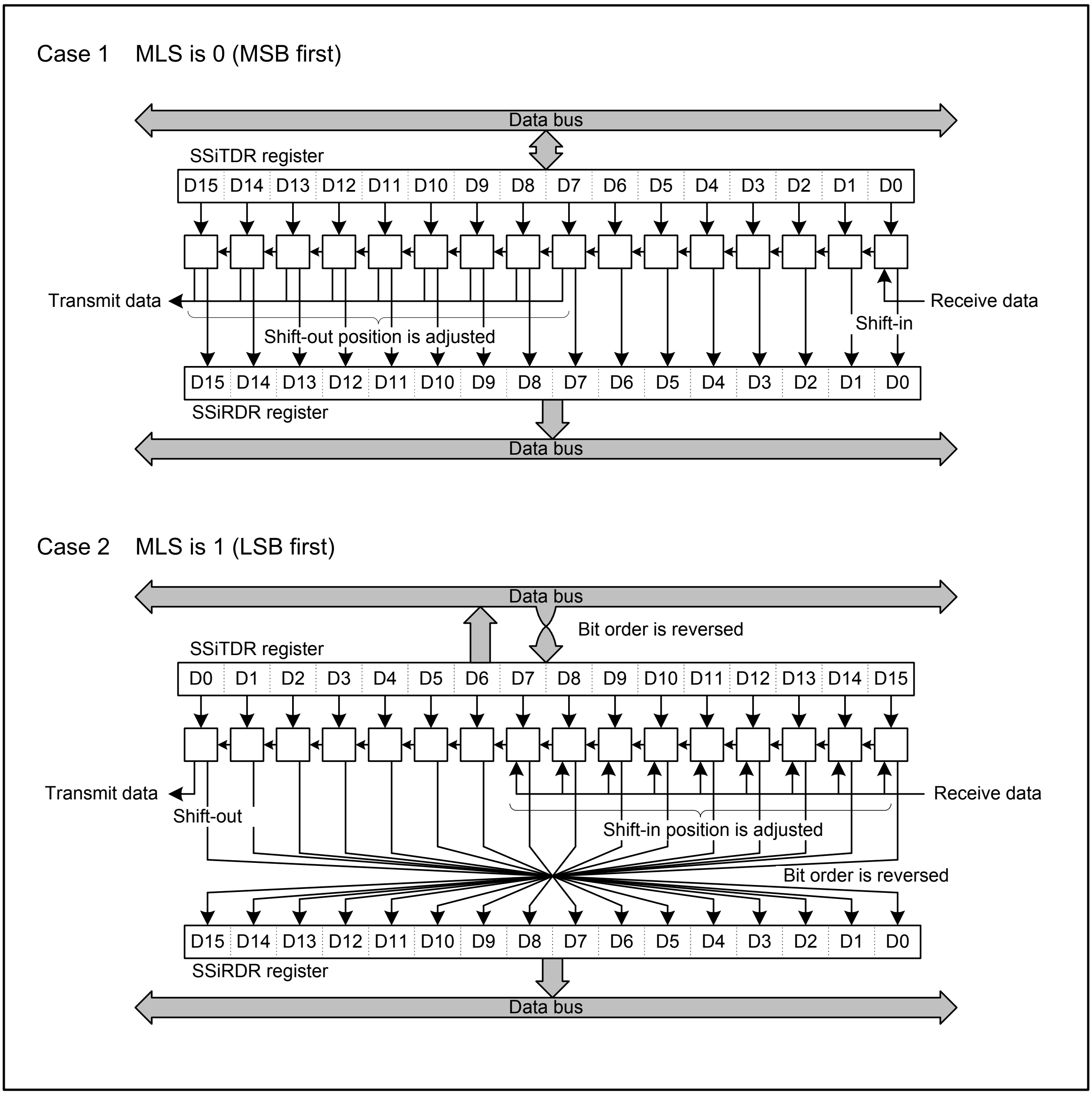


Figure 23.13 Transmit/Receive Shift Register Operation

23.1.3 Data I/O Pin and SSi Shift Register

The connection pattern of the data I/O pins and the transmit/receive shift register varies with the setting combination of the MSS bit in the SSiCRH register, and bits SSUMS and BIDE in the SSiMR2 register. Figure 23.14 shows the relation between those parameters.

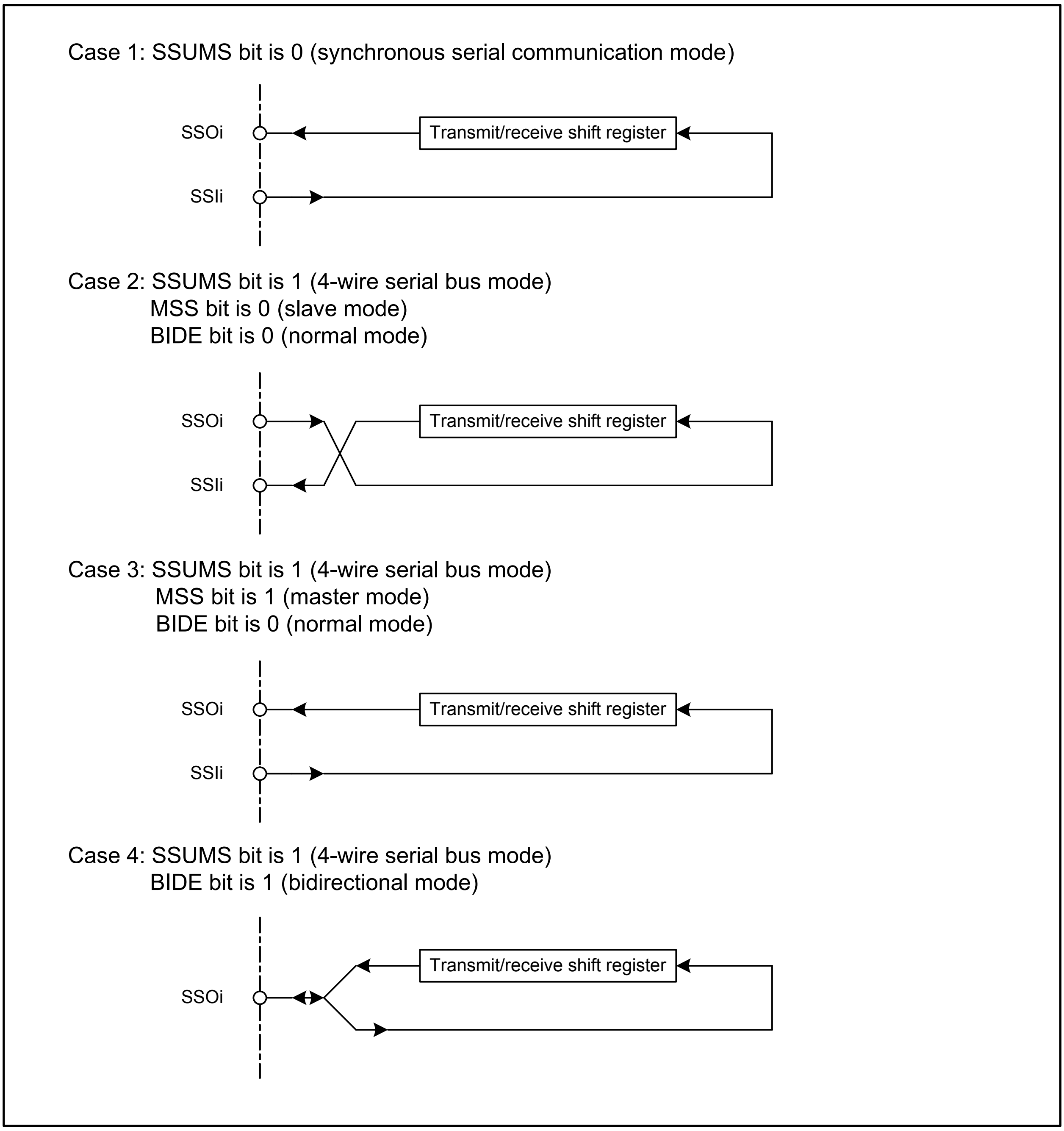


Figure 23.14 Data I/O Pins and Transmit/Receive Shift Register (i = 0, 1)

23.1.4 Interrupt Requests

In synchronous serial communication mode and 4-wire serial bus mode, there are five interrupt request sources: transmit data register empty, transmit end, receive data register full, overrun error, and conflict error. Since all of these interrupt request sources are assigned to an interrupt vector table, they should be identified by flags in the interrupt handler.

Table 23.3 lists the serial bus interface interrupt request sources.

Table 23.3 Interrupt Request Sources in Synchronous Serial Communication Mode and 4-wire Serial Bus Mode

Interrupt Request Source	Generating Condition for Interrupt Requests (1, 2)
Transmit data register empty (TXI)	TIE = 1 and TDRE = 1
Transmit end (TEI)	TEIE = 1 and TEND = 1
Receive data register full (RXI)	RIE = 1 and RDRF = 1
Overrun error (OEI)	RIE = 1 and ORER = 1
Conflict error (CEI)	CEIE = 1 and CE = 1

Notes:

1. CEIE, RIE, TEIE, and TIE: Bits in the SSiER register
2. CE, ORER, RDRF, TEND, and TDRE: Bits in the SSiSR register

When one or more conditions to generate an interrupt request listed in Table 23.3 are met, a serial bus interface interrupt request is generated. The corresponding interrupt sources should be set to 0 in the interrupt handler.

However, bits TDRE (transmit data register empty) and TEND (transmit end) automatically become 0 when the next transmit data is written to the SSiTDR register. The RDRF bit (receive data register full) automatically becomes 0 when the SSiRDR register is read.

Note that if no data is being transmitted, the TDRE bit, which became 0 as mentioned above, is set back to 1 (no data left in the SSiTDR register) since the written data is immediately transferred to the transmit/receive shift register. And, if the TDRE bit is set to 0 (unsent data left in SSiTDR register) by a program, it may cause the transmission of an additional data frame.

23.1.5 Communication Modes and Pin Functions

In synchronous serial communication mode and 4-wire serial bus mode, the function of I/O pins varies with the setting combinations of the MSS bit in the SSiCRH register, and bits RE and TE in the SSiER register. The SSCKi pin is used for output in master mode, and input in slave mode. The port direction register and the port function select register need to be configured according to the direction of pins SSIi, SSOi, and SSCKi.

Table 23.4 shows the relation between communication modes and I/O pins.

Table 23.4 Communication Modes and I/O Pins (i = 0, 1)

Communication Mode	Bit Setting					Pin State		
	SSUMS	BIDE	MSS	TE	RE	SSIi	SSOi	SSCKi
Synchronous serial communication mode	0	Disabled	0 (slave)	0	1	Input	– (1)	Input
				1	0	– (1)	Output	
					1	Input	Output	
			1 (master)	0	1	Input	– (1)	Output
				1	0	– (1)	Output	
					1	Input	Output	
4-wire serial bus mode (in standard mode)	1	0	0 (slave)	0	1	– (1)	Input	Input
				1	0	Output	– (1)	
					1	Output	Input	
			1 (master)	0	1	Input	– (1)	Output
				1	0	– (1)	Output	
					1	Input	Output	
4-wire bus communication mode (in bidirectional mode) (2)	1	1	0 (slave)	0	1	– (1)	Input	Input
				1	0	– (1)	Output	
			1 (master)	0	1	– (1)	Input	Output
				1	0	– (1)	Output	

Notes:

1. This pin can be used as a programmable I/O port.
2. In 4-wire bus communication mode (in bidirectional mode), do not use the following combination: TE bit is 1 and RE bit is 1.

SSUMS and BIDE: Bits in the SSiMR2 register
MSS: Bit in the SSiCRH register
TE and RE: Bits in the SSiER register

23.1.6 Synchronous Serial Communication Mode

23.1.6.1 Initialization of Synchronous Communication Mode

Figure 23.15 shows the initialization procedure of synchronous serial communication mode. Set the TE bit in the SSiER register to 0 (transmission disabled) and the RE bit in the SSiER register to 0 (reception disabled) before the data transmit/receive operation.

To change the communication mode or the communication data format, set bits TE and RE to 0 in advance. Note that flags RDRF and ORER and the SSiRDR register are unaffected even if the RE bit is set to 0.

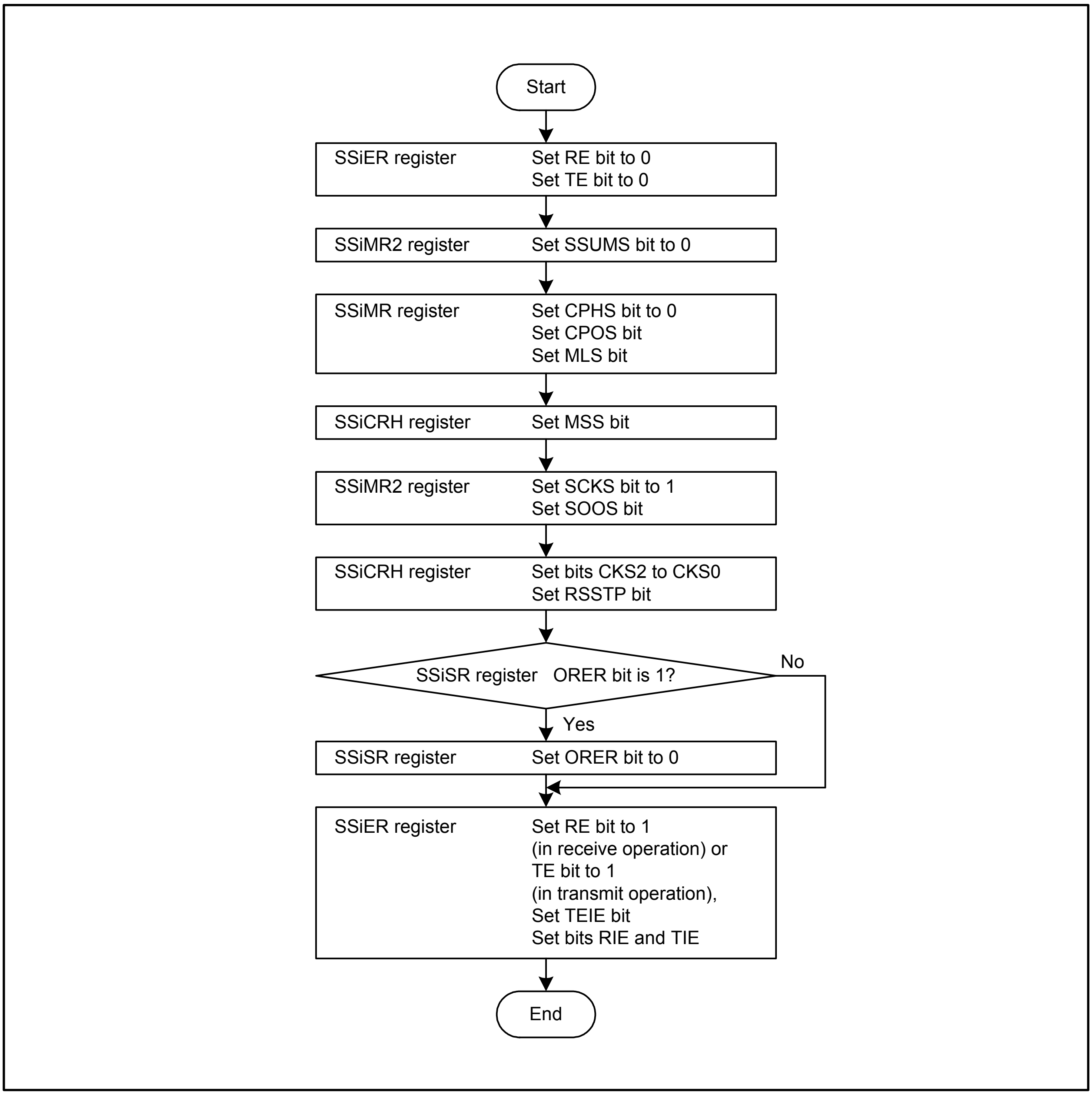


Figure 23.15 Initialization Procedure in Synchronous Communication Mode (i = 0, 1)

23.1.6.2 Data Transmission

Figure 23.16 shows an example of the transmit operation in synchronous serial communication mode. During data transmission, the serial bus interface operates as described below.

When operating in master mode, an internally generated clock as a transmit/receive clock is output from the SSCKi pin. When operating in slave mode, the transmit/receive clock is a clock input from the SSCKi pin. In both cases, transmit data is output synchronized with the transmit/receive clock.

After setting the TE bit in the SSiER register to 1 (transmission enabled), if transmit data is written to the SSiTDR register, the TDRE bit in the SSiSR register automatically becomes 0 (unsent data left in the SSiTDR register), and data in the SSiTDR register is transferred to the transmit/receive shift register. Then the TDRE bit becomes 1 (no data left in the SSiTDR register) and data transmission starts. And, if the TIE bit in the SSiER register is 1 (transmit data register empty interrupt enabled), a transmit data empty interrupt request (TXI) is generated.

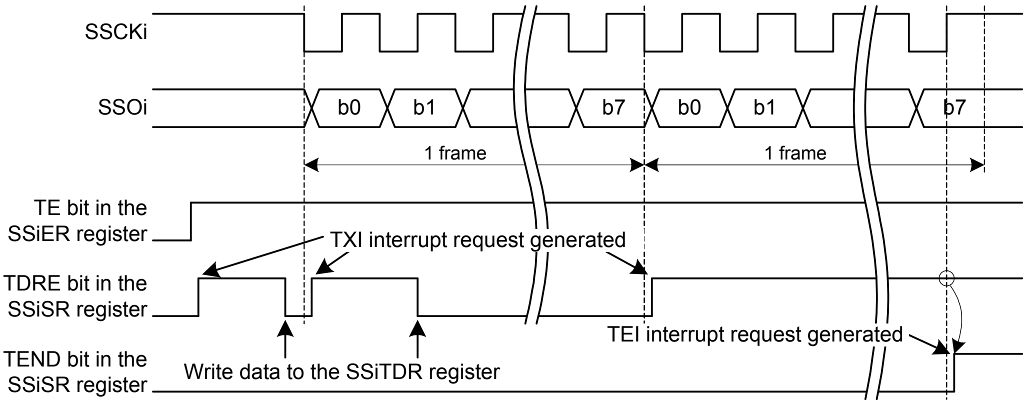
When the TDRE bit is 0, after transmitting one frame of data, the data is transferred from the SSiTDR register to the transmit/receive shift register to start the next data transmission. When the TDRE bit is 1, after transmitting the eighth bit of data, the TEND bit in the SSiSR register becomes 1 (transmission is completed). And, if the TEIE bit in the SSiER register is 1 (transmit end interrupt enabled), a transmit end interrupt request (TEI) is generated.

After data transmission is completed, the SSCKi pin is fixed to the level specified by the CPOS bit in the SSiMR register.

Note that data cannot be transmitted while the ORER bit in the SSiSR register is 1 (overrun error occurred). Set the ORER bit to 0 before data transmission.

Figure 23.17 shows an example of data transmission procedure in synchronous serial communication mode.

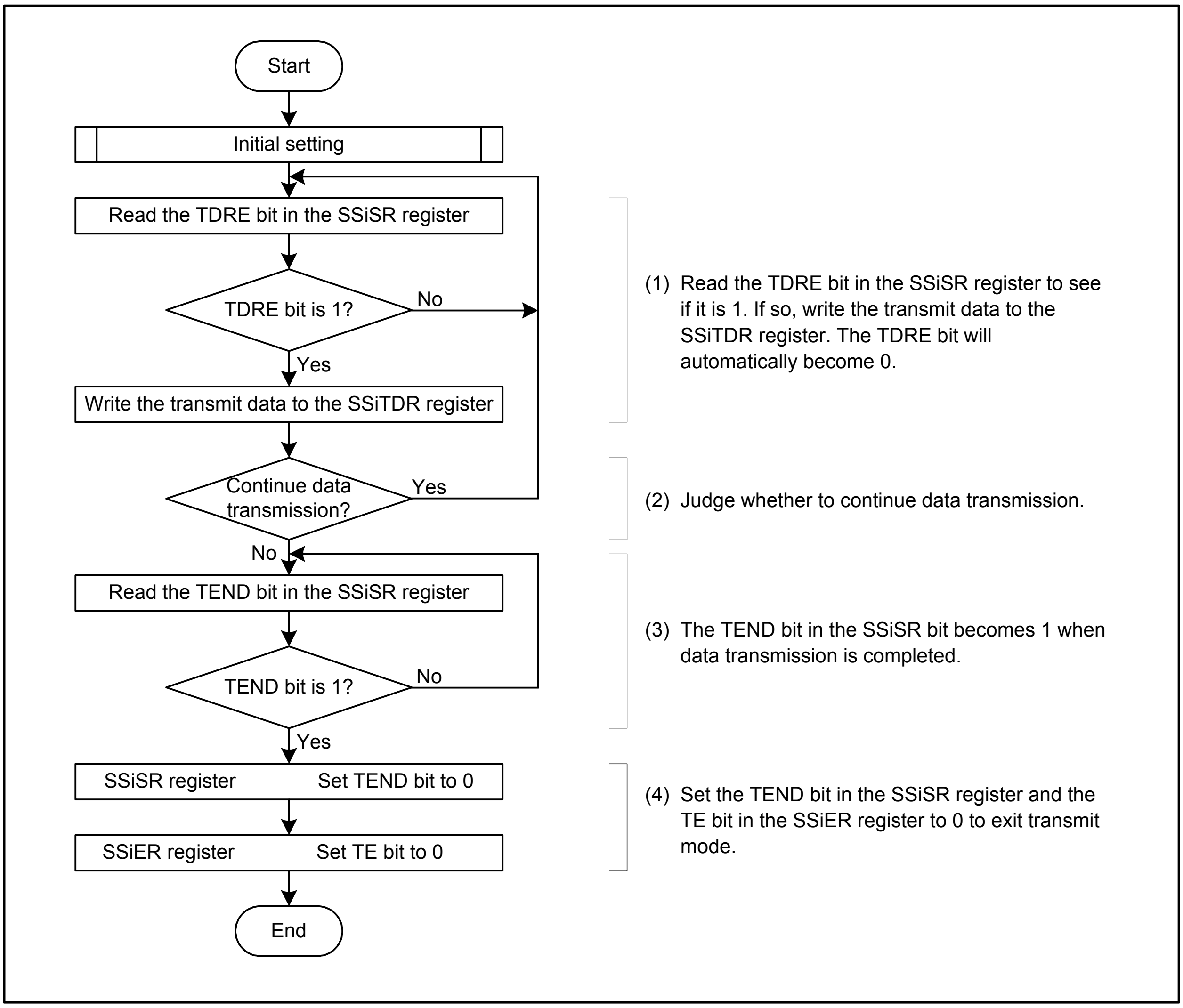
Transmit operation example in synchronous serial communication mode



Note:

1. This figure is subject to the following conditions:
   The MSS bit in the SSiCRH register is 1 (master mode).
   The CPHS bit in the SSiMR register is 0 (change data at odd edges (load data at even edges)).
   The CPOS bit in the SSiMR register is 0 (high when clock is stopped).
   The MLS bit in the SSiMR register is 1 (LSB first).

Figure 23.16 Transmit Operation Example in Synchronous Serial Communication Mode (i = 0, 1)



(1) Read the TDRE bit in the SSiSR register to see if it is 1. If so, write the transmit data to the SSiTDR register. The TDRE bit will automatically become 0.

(2) Judge whether to continue data transmission.

(3) The TEND bit in the SSiSR bit becomes 1 when data transmission is completed.

(4) Set the TEND bit in the SSiSR register and the TE bit in the SSiER register to 0 to exit transmit mode.

Figure 23.17 Data Transmission Example in Synchronous Serial Communication Mode (i = 0, 1)

23.1.6.3 Data Reception

Figure 23.18 shows an example of the receive operation in synchronous serial communication mode. During data reception, the serial bus interface operates as described below.

When operating in master mode, an internally generated clock as transmit/receive clock is output from the SSCKi pin. When operating in slave mode, the transmit/receive clock is a clock input from the SSCKi pin. In both cases, receive data is input synchronized with the transmit/receive clock.

In master mode, a dummy read of the SSiRDR register evokes the receive clock output to start data reception.

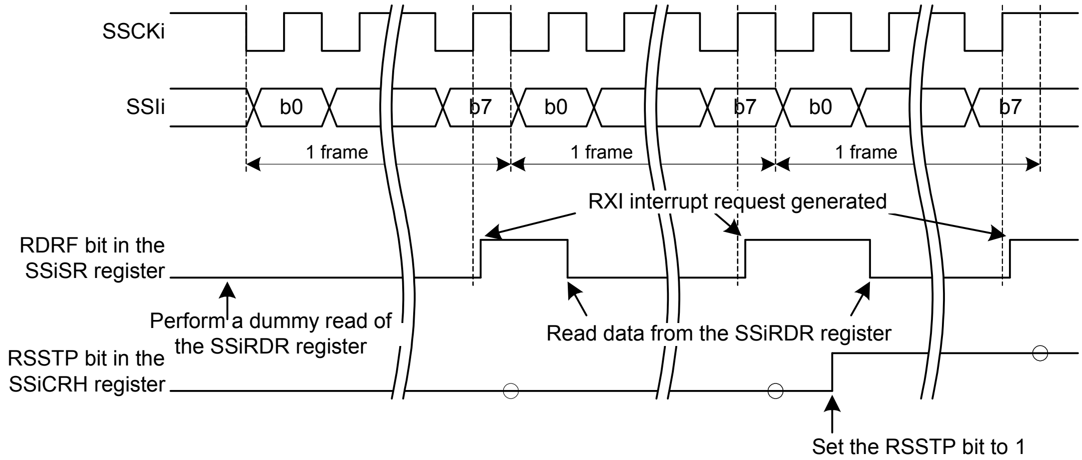
When the eighth bit of data is received, the RDRF bit in the SSiSR register becomes 1 (received data left in the SSiRDR register) and the received data is stored into the SSiRDR register. And, if the RIE bit in the SSiER register is 1 (receive data register full interrupt/overrun error interrupt enabled), a receive data register full interrupt request (RXI) is generated. When the SSiRDR register is read, the RDRF bit automatically becomes 0 (no data left in the SSiRDR register).

To end the receive data operation in master mode, set the RSSTP bit in the SSiCRH register to 1 (receive operation ended after receiving the current frame) before reading the data of the second to last frame from the SSiRDR register. Then the transmit/receive clock stops when the last frame reception is completed. When the clock stops, set the RE bit in the SSiER register to 0 (reception disabled) and the RSSTP bit to 0 (receive operation continued after receiving the current frame) and read the data of the last frame. Note that if the SSiRDR register is read while the RE bit is 1, then the receive clock is output again.

If the eighth bit of the data is received when the RDRF bit is set to 1, the ORER bit in the SSiSR register becomes 1 (overrun error occurred), and the receive operation stops. Note that data cannot be received while the ORER bit is 1. Set the ORER bit to 0 before resuming data reception.

Figure 23.19 shows an example of the data reception procedure in master mode of synchronous communication mode.

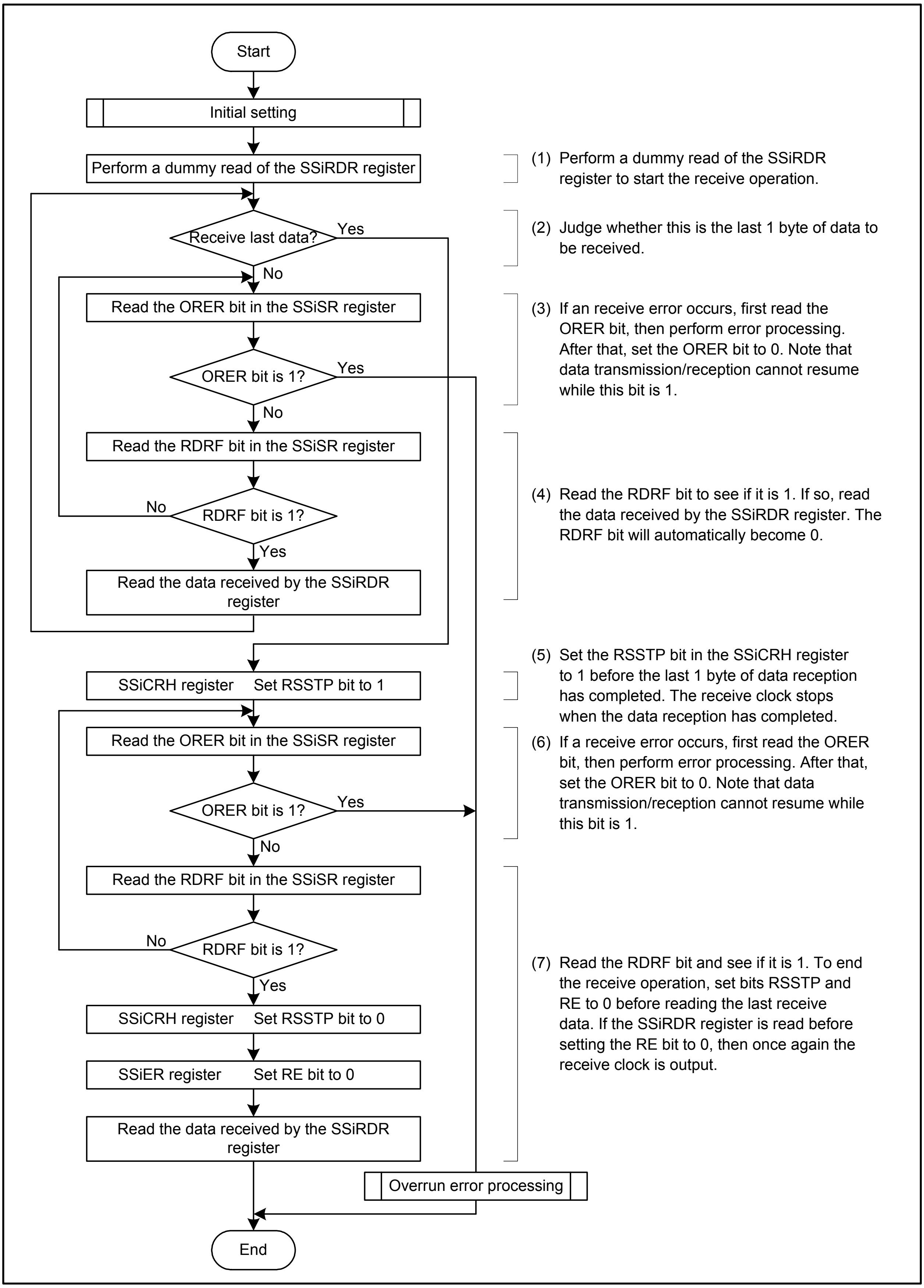
Receive operation example in synchronous serial communication mode



Note:

1. This figure is subject to the following conditions:
   The MSS bit in the SSiCRH register is 1 (master mode).
   The CPHS bit in the SSiMR register is 0 (change data at odd edges (load data at even edged)).
   The CPOS bit in the SSiMR register is 0 (high when clock is stopped).
   The MLS bit in the SSiMR register is 1 (LSB first).

Figure 23.18 Receive Operation Example in Master Mode of Synchronous Serial Communication Mode (i = 0, 1)



(1) Perform a dummy read of the SSiRDR register to start the receive operation.

(2) Judge whether this is the last 1 byte of data to be received.

(3) If an receive error occurs, first read the ORER bit, then perform error processing. After that, set the ORER bit to 0. Note that data transmission/reception cannot resume while this bit is 1.

(4) Read the RDRF bit to see if it is 1. If so, read the data received by the SSiRDR register. The RDRF bit will automatically become 0.

(5) Set the RSSTP bit in the SSiCRH register to 1 before the last 1 byte of data reception has completed. The receive clock stops when the data reception has completed.

(6) If a receive error occurs, first read the ORER bit, then perform error processing. After that, set the ORER bit to 0. Note that data transmission/reception cannot resume while this bit is 1.

(7) Read the RDRF bit and see if it is 1. To end the receive operation, set bits RSSTP and RE to 0 before reading the last receive data. If the SSiRDR register is read before setting the RE bit to 0, then once again the receive clock is output.

Figure 23.19 Data Reception Example in Master Mode of Synchronous Serial Communication Mode (i = 0, 1)

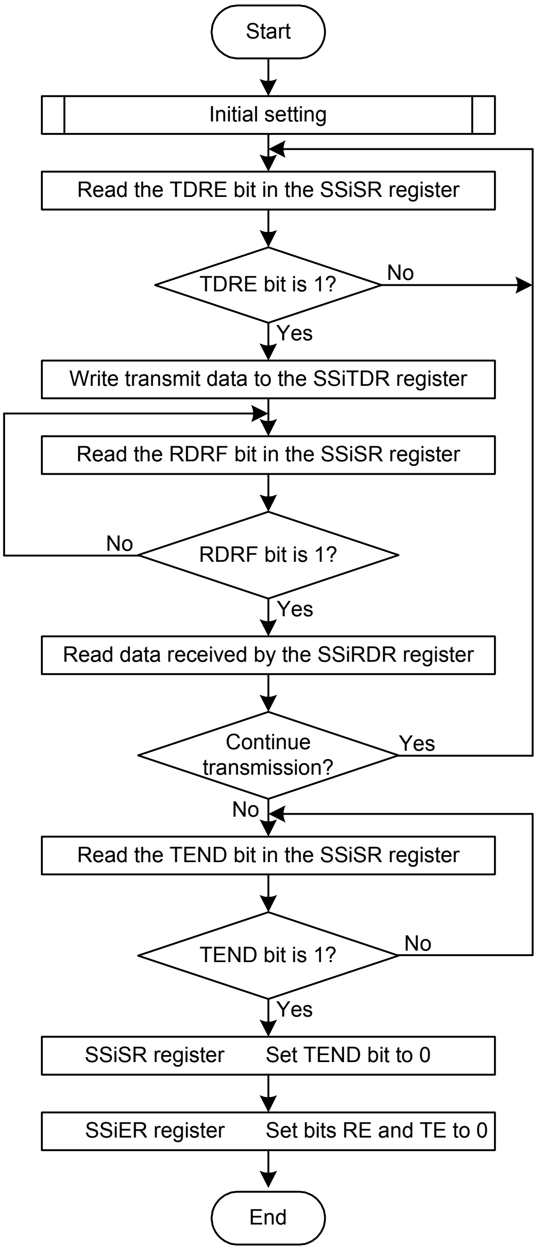
23.1.6.4 Data Transmission/Reception

Data transmission/reception is a combined operation of data transmission and data reception mentioned in sections 23.1.6.2 and 23.1.6.3.

Data transmission/reception starts when data is written to the SSiTDR register, and ends when the eighth bit of data is transmitted while the TDRE bit is 1 (no data left in the SSiTDR register). The data transmit/receive operation stops when the ORER bit becomes 1 (overrun error occurred) due to an error.

To switch modes from transmit mode (TE bit is 1 and RE bit is 0) or receive mode (TE bit is 0 and RE bit is 1) to transmit/receive mode (TE bit is 1 and RE bit is 1), set bits TE and RE to 0. Then, check that the TEND bit is 0 (transmission continued), the RDRF bit is 0 (no data left in the SSiRDR register), the ORER bit is 0 (no overrun error), and subsequently set bits TE and RE to 1 at the same time.

Figure 23.20 shows an example of data transmission/reception procedure in synchronous serial communication mode.



(1) Read the TDRE bit in the SSiSR register to see if it is 1. If so, write transmit data to the SSiTDR register. The TDRE bit in the SSiTDR register will automatically become 0.

(2) Read the RDRF bit to see if it is 1. If so, read the SSiRDR register. The RDRF bit will automatically become 0.

(3) Judge whether to continue data transmission.

(4) The TEND bit in the SSiSR register becomes 1 when data transmission is completed.

(5) Set the TEND bit in the SSiSR register and bits RE and TE in the SSiER register to 0 to exit transmit/receive mode.

Figure 23.20 Data Transmission/Reception Example in Synchronous Communication Mode (i = 0, 1)

23.1.7 4-wire Serial Bus Mode

In 4-wire serial bus mode, the serial bus interface operates serial communication via four logic signal lines: clock line, data input line, data output line, and chip select line. This mode also contains a bidirectional mode, in which a single pin is used for the data input line and data output line.

The data input line and data output line share the SSIi pin and SSOi pin. The combined setting of the MSS bit in the SSiCRH register and the BIDE bit in the SSiMR2 register determines which line is assigned to which pin. Refer to 23.1.3 “Data I/O Pin and SSi Shift Register” for more details. In this mode, the clock polarity and phase for transmit/receive data can be configured by setting bits CPOS and CPHS in the SSiMR register, respectively. Refer to 23.1.1.1 “Transmit/Receive Clock Polarity and Phase” for more details.

The chip select line functions as an output pin in master mode and as an input pin in slave mode. In master mode, the $\overline{\text{SCSi}}$ pin can be assigned as an output pin by setting bits CSS1 and CSS0 in the SSiMR2 register to 10b. Also, a programmable I/O port can be assigned as the chip select pin in this mode. In slave mode, the $\overline{\text{SCSi}}$ pin is assigned as an input pin by setting bits CSS1 and CSS0 to 01b.

In 4-wire serial bus mode, serial communication is normally performed with the MSB first (the MSL bit in the SSiMR register is set to 0).

23.1.7.1 Initialization of 4-wire Serial Bus Mode

Figure 23.21 shows the initialization of 4-wire serial bus mode. Initialize the TE bit in the SSiER register to 0 (transmission disabled) and RE bit in the SSiER register to 0 (reception disabled) before data transmission/reception.

To change the communication mode or the communication data format, set bits TE and RE to 0 in advance. Note that bits RDRF and ORER and the SSiRDR register are unaffected even if the RE bit is set to 0.

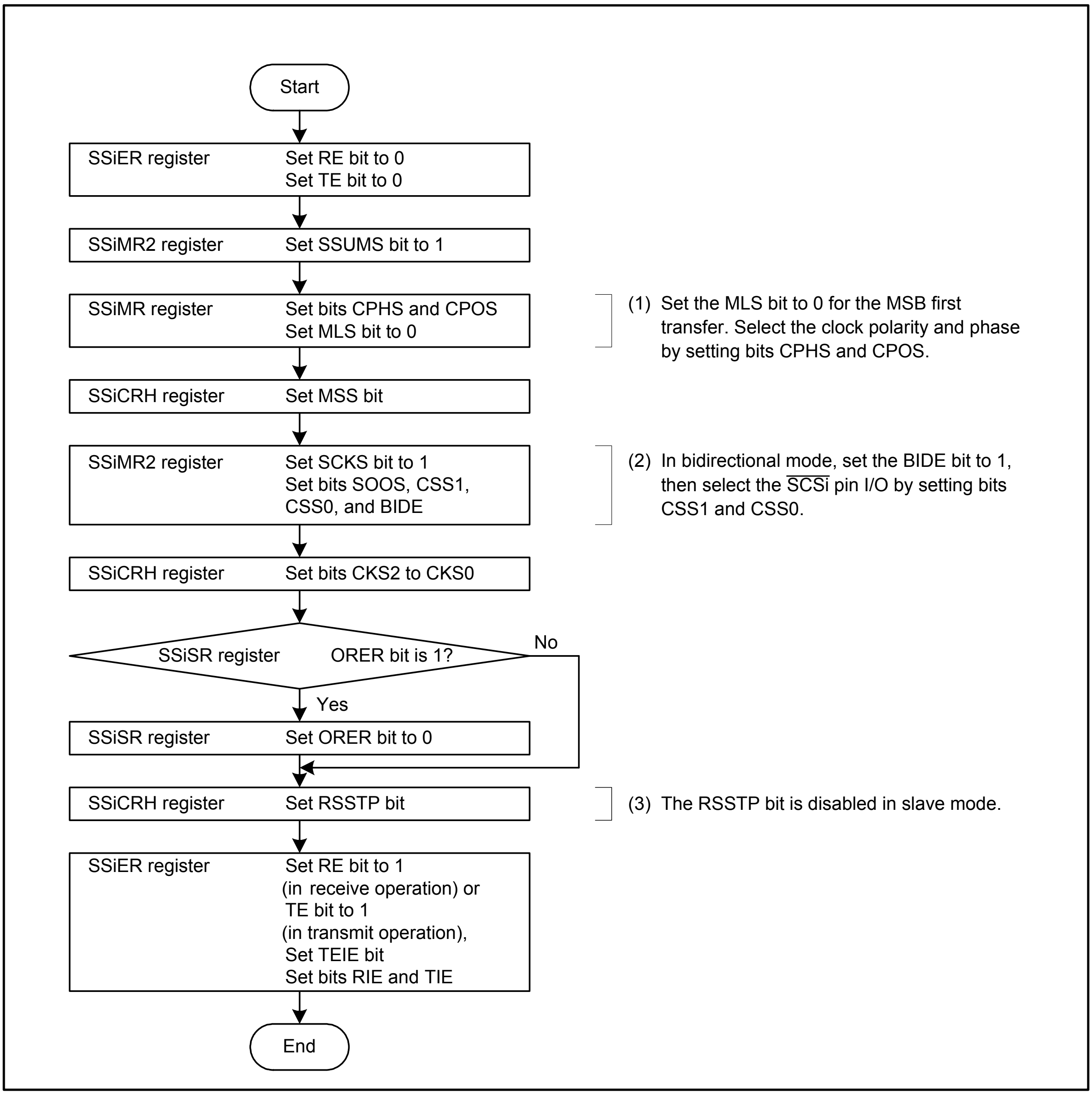


Figure 23.21 Initialization in 4-wire Serial Bus Mode (i = 0, 1)

23.1.7.2 Data Transmission

Figure 23.22 shows an example of the transmit operation in 4-wire serial bus mode. During data transmission, the serial bus interface operates as described below.

When operating in master mode, an internally generated clock as a transmit/receive clock is output from the SSCKi pin. When operating in slave mode, the transmit/receive clock is a clock input from the SSCKi pin while the $\overline{\text{SCSi}}$ pin is low. In both cases, transmit data is output synchronized with the transmit/receive clock.

After setting the TE bit in the SSiER register to 1 (transmission enabled), if transmit data is written to the SSiTDR register, the TDRE bit in the SSiSR register automatically becomes 0 (unsent data left in the SSiTDR register), and data in the SSiTDR register is transferred to the transmit/receive shift register. Then the TDRE bit becomes 1 (no data left in the SSiTDR register) and data transmission starts. And, if the TIE bit in the SSiER register is 1 (transmit data register empty interrupt enabled), a transmit data empty interrupt request (TXI) is generated.

When the TDRE bit is 0, after transmitting one frame of data, the data is transferred from the SSiTDR register to the transmit/receive shift register to start the next data transmission. When the TDRE bit is 1, after transmitting the last bit of data, the TEND bit in the SSiSR register becomes 1 (transmission completed). And, if the TEIE bit in the SSiER register is 1 (transmit end interrupt enabled), a transmit end interrupt request (TEI) is generated.

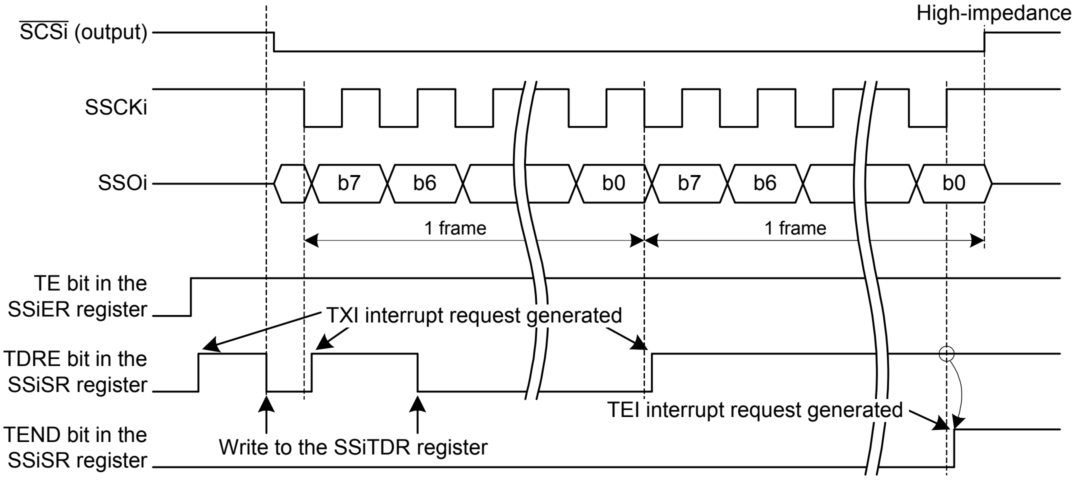
After the data transmission is completed, the SSCKi pin and the $\overline{\text{SCSi}}$ pin become high. To continue data transmission with the $\overline{\text{SCSi}}$ pin low, write the data to be transmitted next to the SSiTDR register before the last bit is transmitted.

Note that data cannot be transmitted while the ORER bit in the SSiSR register is 1 (overrun error occurred). Set the ORER bit to 0 before data transmission.

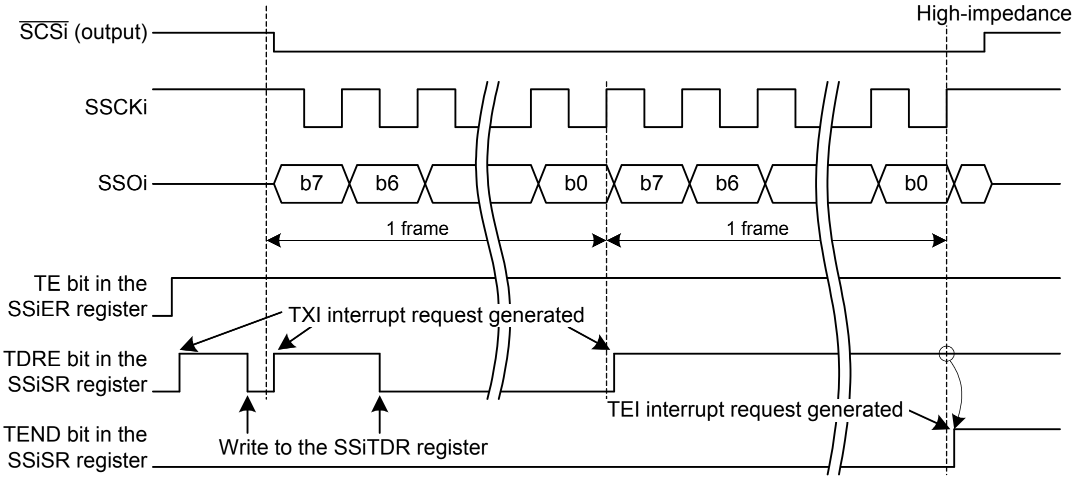
In contrast to synchronous serial communication mode, the $\overline{\text{SCSi}}$ pin is used in 4-wire serial bus mode. When the serial bus interface is in master mode, the SSOi pin is high-impedance while the output level at the $\overline{\text{SCSi}}$ pin is high. In slave mode, the SSIi pin is high-impedance while the input level at the $\overline{\text{SCSi}}$ pin is high.

Refer to Figure 23.17 for an example of data transmission procedure.

Case 1 CPHS is 0 (change data at odd edges and load data at even edges)



Case 2 CPHS is 1 (change data at even edges and load data at odd edges)



CPOS: Bit in the SSiMR register

Note:

1. This figure is subject to the following conditions:
   The MSS bit in the SSiCRH register is 1 (master mode).
   The CPOS bit in the SSiMR register is 0 (high when clock is stopped).
   The MLS bit in the SSiMR register is 0 (MSB first).
   Bits BC3 to BC0 are 1000b (8-bit length).
   The CSOS bit in the SSiMR2 is 1 (SCSi pin is N-channel open drain output).

Figure 23.22 Transmit Operation Example in 4-wire Serial Bus Mode (i = 0, 1)

23.1.7.3 Data Reception

Figure 23.23 shows an example of receive operation in 4-wire serial bus mode. During data reception, the serial bus interface operates as described below.

When operating in master mode, an internally generated clock as a transmit/receive clock is output from the SSCKi pin. When operating in slave mode, the transmit/receive clock is a clock input from the SSCKi pin while the $\overline{\text{SCSi}}$ pin is low. In both cases, receive data is input synchronized with the transmit/receive clock.

In master mode, a dummy read of the SSiRDR register evokes the receive clock output to start data reception. Note that the TE bit in the SSiER register should be set to 0 before performing the dummy read to exit transmit mode.

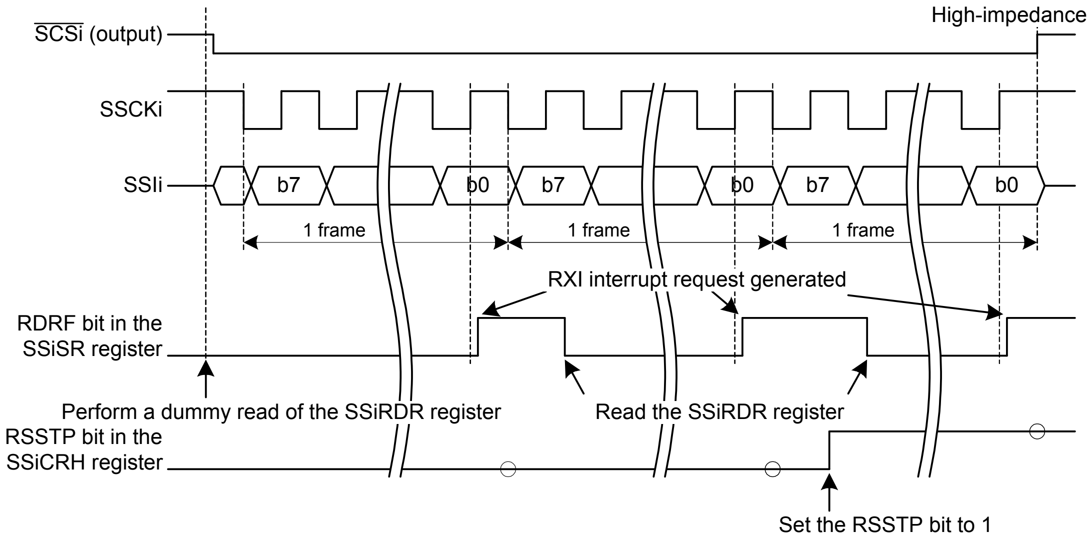
When the specified data length is received, the RDRF bit in the SSiSR register becomes 1 (received data left in the SSiRDR register) and the received data is stored into the SSiRDR register. And, if the RIE bit in the SSiER register is 1 (receive data register full interrupt/overrun error interrupt enabled), a receive data register full interrupt request (RXI) is generated. When the SSiRDR register is read, the RDRF bit automatically becomes 0 (no data left in the SSiRDR register).

To end the receive data operation in master mode, set the RSSTP bit in the SSiCRH register to 1 (receive operation ended after receiving the current frame) before reading the data of the second to last frame from the SSiRDR register. Then the transmit/receive clock stops when the last frame is received. When the clock stops, set the RE bit in the SSiER register to 0 (reception disabled) and the RSSTP bit to 0 (receive operation continued after receiving the current frame) and read the data of the last frame. Note that if the SSiRDR register is read while the RE bit is 1, then the receive clock is output again.

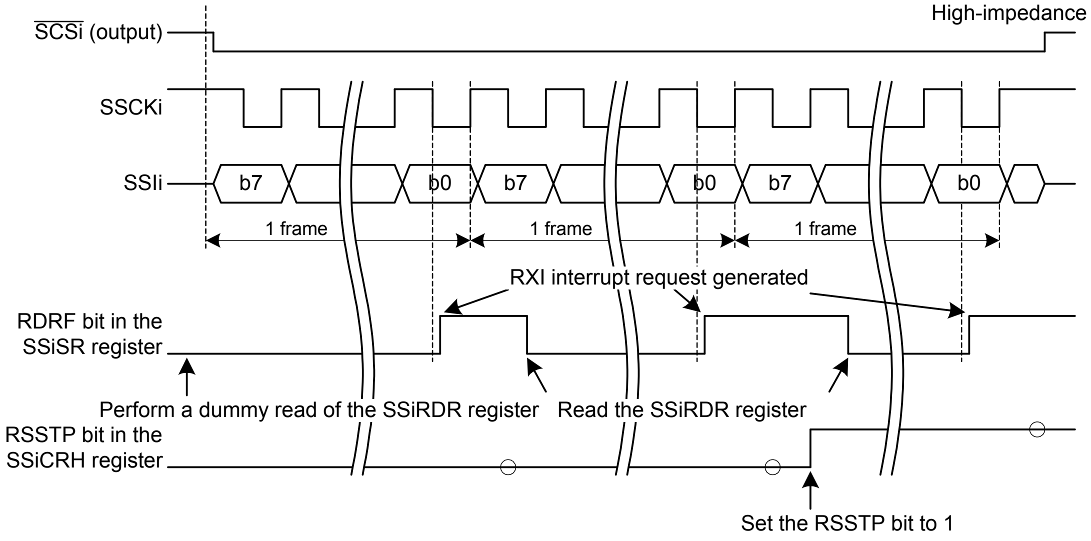
If the last bit of the data is received when the RDRF bit is set to 1, the ORER bit in the SSiSR register becomes 1 (overrun error occurred), and the receive operation stops. Note that data cannot be received while the ORER bit is 1. Set the ORER bit to 0 before resuming data reception.

Refer to Figure 23.19 for an example of data reception procedure.

Case 1 CPHS is 0 (change data at odd edges and load data at even edges)



Case 2 CPHS is 1 (change data at even edges and load data at odd edges)



CPOS: Bit in the SSiMR register

Note:

1. This figure is subject to the following conditions:
   The MSS bit in the SSiCRH register is 1 (master mode).
   The CPOS bit in the SSiMR register is 0 (high when clock is stopped).
   The MLS bit in the SSiMR register is 0 (MSB first).
   Bits BC3 to BC0 in the SSiMR register is 1000b (8-bit length).
   The CSOS bit in the SSiMR2 register is 1 ($\overline{\text{SCSi}}$ is N-channel open drain output).

Figure 23.23 Receive Operation Example in 4-wire Serial Bus Mode (i = 0, 1)

23.1.7.4 $\overline{\text{SCSi}}$ Pin Control and Arbitration

In 4-wire serial bus mode, when bits CSS1 and CSS0 are set to 10b (functions as $\overline{\text{SCSi}}$ output pin), and the MSS bit in the SSiCRH register is 1 (master mode), then arbitration is performed by monitoring the $\overline{\text{SCSi}}$ pin before starting serial data transmission. When the serial bus interface detects that the synchronized internal $\overline{\text{SCSi}}$ signal level drops to low before data transmission, the CE bit in the SSiSR register becomes 1 (conflict error occurred) and the MSS bit automatically becomes 0 (slave mode). Figure 23.24 shows the arbitration timing.

Note that the serial bus interface can no longer continue transmitting while the CE bit is 1. To start data transmission, confirm the CE bit is 1 and then set it to 0 (no conflict error).

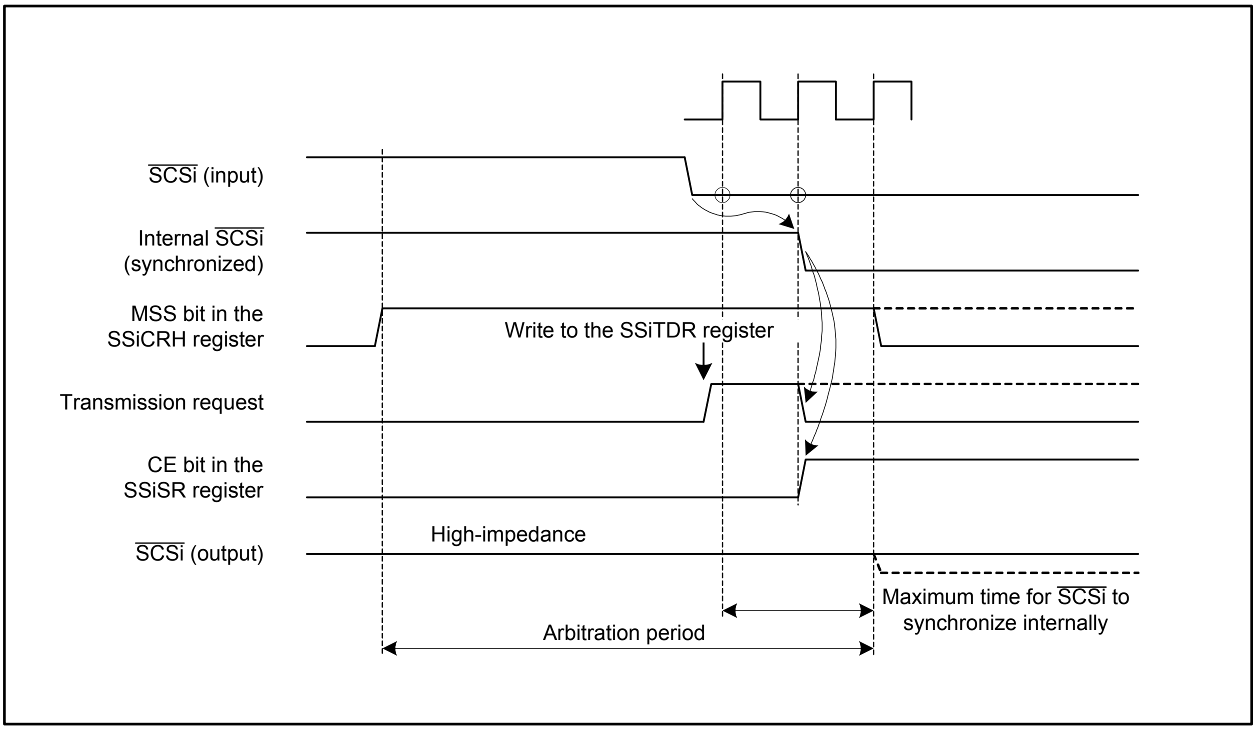


Figure 23.24 Arbitration Timing (i = 0, 1)

23.2 Note on Serial Bus Interface

23.2.1 Note on Synchronous Serial Communication Mode and 4-wire Serial Bus Mode

23.2.1.1 Accessing SBI Associated Registers

To read the SBI associated registers (44F06h to 44F1Fh), insert at least three instructions after writing to the registers.

- An example of inserting at least three instructions

```
Program example        MOV.B   #00h, 44F0Bh ; Set the SS0ER register to 00h.
                       NOP
                       NOP
                       NOP
                       MOV.B   44F0Bh, R0L
```

24. LIN Module

The R32C/142 Group and R32C/145 Group implement a local interconnect network (LIN) module, compliant with LIN Protocol Specification Revisions 1.3, 2.0, and 2.1, which transmits and receives frames, and judges errors automatically. The module incorporates two channels of master controllers.
Table 24.1 lists the LIN module specifications and Figure 24.1 shows the LIN module block diagram.

Table 24.1 LIN Module Specifications

Item	Specification
Protocols	LIN Protocol Specification Revisions 1.3, 2.0, and 2.1
Channels	2 (LIN master)
Variable frame composition	Transmit break length: 13 to 28 Tbit Transmit break delimiter length: 1 to 4 Tbit Interbyte space (Header): 0 to 7 Tbit (space between Sync field and ID field) (1) Response space: 0 to 7 Tbit (1) Interbyte space: 0 to 3 Tbit (space between bytes in response space)
Checksum	Classic checksum or enhanced checksum selectable (changeable at each channel and each frame)
Data byte(s) in response field	Variable from 0 to 8
Frame transmission modes	Non-frame separate mode or frame separate mode selectable; the former mode transmits a header and response by a single start command and the latter separately transmits them by respective start command
Wake-up transmission/reception	Available in LIN wake-up mode • Wake-up transmission (1 to 16 Tbit) • Wake-up reception • Input signal low time count (0.5 to 15.5 Tbit) • Input signal low detection
Operational status	• Frame/wake-up transmit completion • Frame/wake-up receive completion • Data 1 receive completion • Input signal low detection • Error detection • Operation mode
Error status	• Bit error • Checksum error • Frame timeout error • Physical bus error • Framing error • Overrun error Excluding the checksum error, these errors are detection enable or detection disable selectable
Baud rates	The baud rate generator can generate the baud rates in the LIN specifications (each channel can be selected separately)

Note:

1. The interbyte space (Header) has the same value as the response space since both spaces are set in the same register.

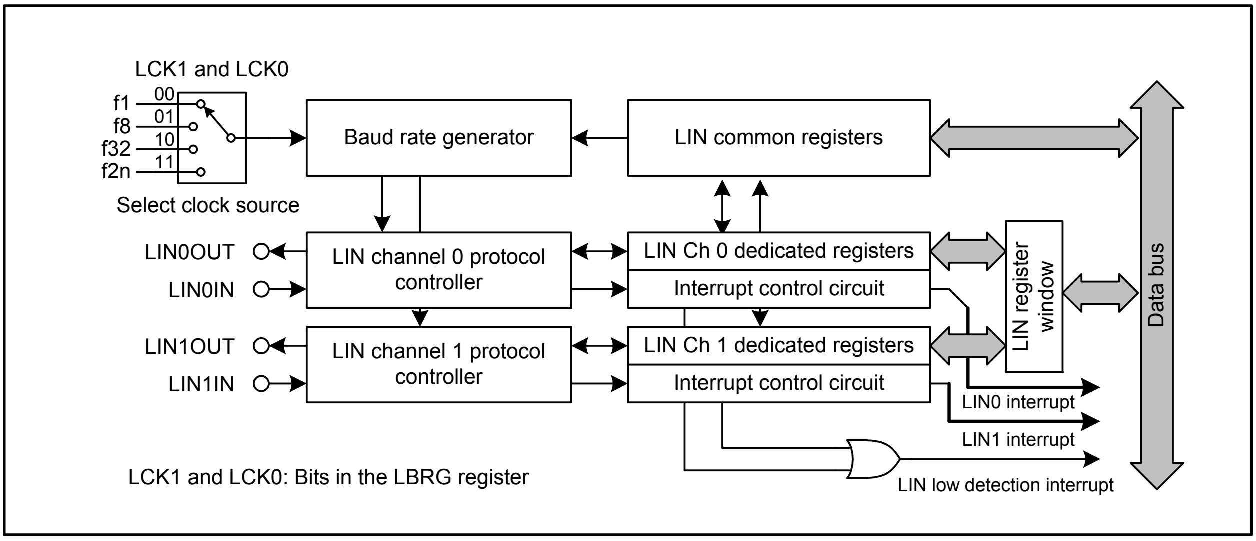


Figure 24.1 LIN Module Block Diagram

- LINjOUT, LINjIN (j = 0, 1): LIN I/O pins
- Baud rate generator: Generates a communication clock for LIN
- LIN common registers: Common registers for the LIN module
- LIN channel j dedicated registers:
  A set of dedicated registers for each channel. Each dedicated register can be switched to that of another channel by the LCW register
- Interrupt control circuit:
  Controls the interrupt requests (LINj interrupt and LIN low detection interrupt) generated by the LIN modules

24.1 LIN Module Associated Registers

The LIN module has two types of registers: LIN module common registers and LIN channel dedicated registers.

The latter compose up to four register banks for channels 0 to 3, of which an appropriate register bank is selected by switching to a channel indicated on the window. The number of channels per module depends on total channels incorporated.

24.1.1 LIN Common Registers

- LIN channel window select/input signal low detection status register (LCW register)
  This register is to select a channel and monitor the status of input signal low detection of each channel.
- LIN baud rate generator control register (LBRG register)
  This register is to select a count source for the baud rate generator and fLn clock source. Refer to 24.4 “Baud Rate Generator” for details.
- LIN baud rate prescaler k (LBRPk register) (k = 0, 1)
  This register is to set the division value for the baud rate prescaler to generate a transmit/receive clock with the count source selected by the LBRG register.

24.1.2 LIN Channel Dedicated Registers

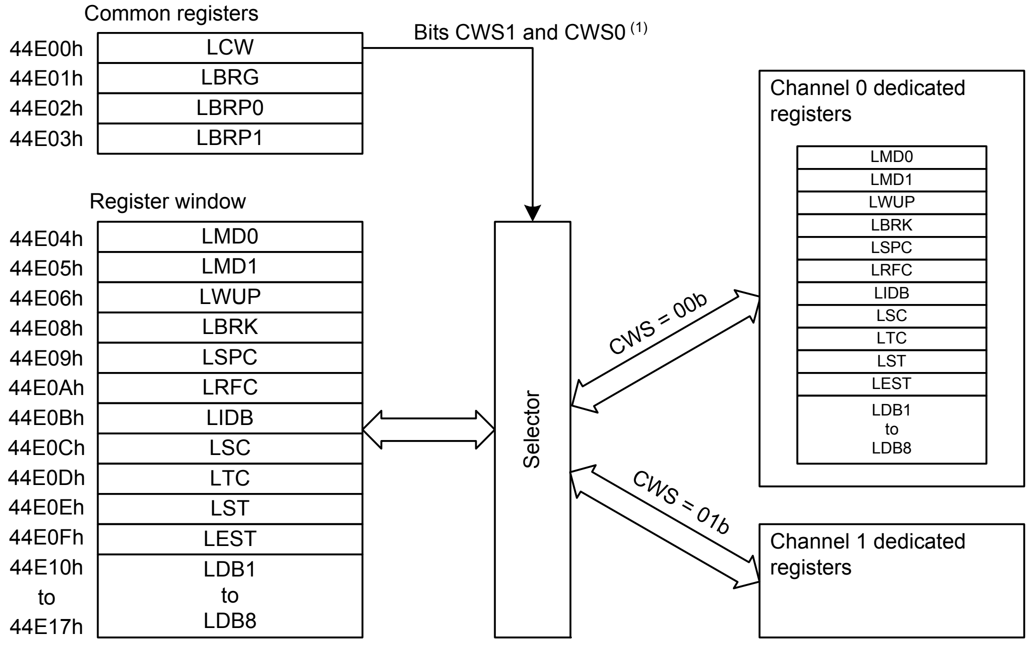
Select a channel by setting bits CWS1 and CWS0 in the LCW register. Each LIN channel dedicated register of the selected channel is set via the following registers:

- LIN mode register 0 (LMD0 register)
  This register is used for the initial setting of LIN module including mode selections and interrupt enable settings.
- LIN mode register 1 (LMD1 register)
  This register is used for the initial setting of LIN module including system clock selection and error detection enable settings.
- LIN wake-up setting register (LWUP register)
  This register sets the low time pulse width of wake-up transmission and reception.
- LIN break field setting register (LBRK register)
  This register sets the time pulse width for break and break delimiter in break field.
- LIN space setting register (LSPC register)
  This register sets each space width required for transmission.
- LIN response field setting register (LRFC register)
  This register sets the number of communication data bytes, the transmit/receive direction, and a checksum model.
- LIN ID buffer register (LIDB register)
  This register sets the transmission ID and ID parity bits.
- LIN status control register (LSC register)
  This register switches the mode of LIN module.
- LIN transmission control register (LTC register)
  This register sets if frame transmission starts and if response field transmission starts.
- LIN status register (LST register)
  This register monitors the status including frame/wake-up transmit completion, frame/wake-up receive completion, Data 1 receive completion, input signal low detection, error detection, and operation mode.
- LIN error status register (LEST register)
  This register monitors error status caused by bit, checksum, physical bus, frame timeout, framing, and overrun errors).
- LIN data n buffer register (n = 1 to 8) (LDBn register)
  This register, commonly used as transmission and reception data buffer, sets data to be transmitted and to read received data.

24.1.3 Register Window and Channel Switching

The LIN channel dedicated registers, which are not directly mapped on the memory map of the CPU, are accessed via a register window. The register window is mapped at addresses 44E04h to 44E17h. According to the setting of bits CWS1 and CWS0 in the LCW register, the LIN channel dedicated registers of the corresponding channel are collectively mapped on the register window.

Figure 24.2 shows the block diagram of the register window.



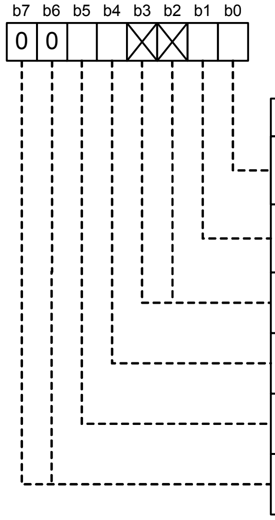
Note:

1. Bits CSW1 and CSW0 should not be set to 10b or 11b.

Figure 24.2 Register Window

Figure 24.3 to Figure 24.18 show LIN module associated registers.

LIN Channel Window Select/Input Low Detection Status Register



Symbol	Address	Reset Value
LCW	44E00h	0000 0000b

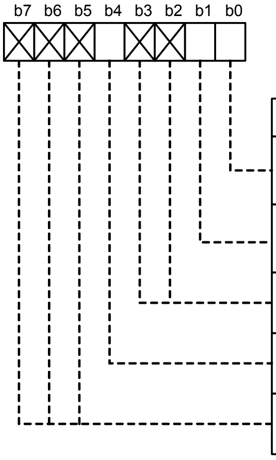
Bit Symbol	Bit Name	Function	RW
CWS0	Channel Window Select Bit	b1 b0 0 0 : Select registers of channel 0 0 1 : Select registers of channel 1 1 0 : Do not use this combination 1 1 : Do not use this combination	RW
CWS1			RW
— (b3-b2)	No register bits; should be written with 0 and read as 0		—
L0LD	Channel 0 Input Signal Low Detection Interrupt Flag (1)	The bit is read as the value of the LD bit in the LST register of channel 0	RO
L1LD	Channel 1 Input Signal Low Detection Interrupt Flag (1)	The bit is read as the value of the LD bit in the LST register of channel 1	RO
— (b7-b6)	Reserved	Should be written with 0	RO

Note:

1. No new interrupt is generated by the input signal low detection when any one of these bits is 1.

Figure 24.3 LCW Register

LIN Baud Rate Generator Control Register (1)



Symbol	Address	Reset Value
LBRG	44E01h	0000 0000b

Bit Symbol	Bit Name	Funciton	RW
LCK0	BRG Count Source Select Bit	b1 b0 0 0 : f1 0 1 : f8 1 0 : f32 1 1 : f2n	RW
LCK1			RW
— (b3-b2)	No register bits; should be written with 0 and read as 0		—
FLNS	fLn Clock Source Select Bit	0: BRP0 output 1: BRP0 output divided-by-2	RW
— (b7-b5)	No register bits; should be written with 0 and read as 0		—

Note:

1. This register should be set when all channels are in LIN reset mode.

Figure 24.4 LBRG Register

LIN Baud Rate Prescaler 0 (1)

b7 b0

Symbol	Address	Reset Value
LBRP0	44E02h	00h

Function	Setting Range	RW
The peripheral clock is divided by (n + 1) (2) (n = setting value)	00h to FFh	RW

Notes:

1. This register should be set when all channels are in LIN reset mode.
2. When 00h is set, baud rate prescaler 0 does not provide any clock.

Figure 24.5 LBRP0 Register

LIN Baud Rate Prescaler 1 (1)

b7 b0

Symbol	Address	Reset Value
LBRP1	44E03h	00h

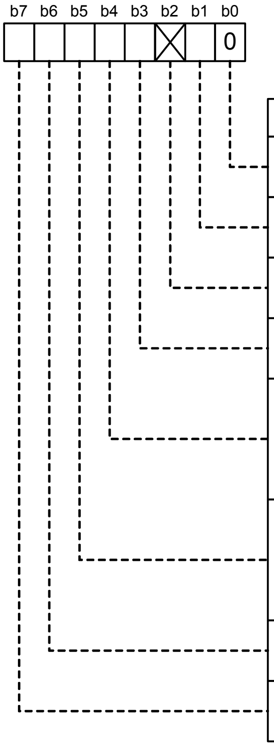
Function	Setting Range	RW
The peripheral clock is divided by (m + 1) (2) (m = setting value)	00h to FFh	RW

Notes:

1. This register should be set when all channels are in LIN reset mode.
2. When 00h is set, baud rate prescaler 1 does not provide any clock.

Figure 24.6 LBRP1 Register

LIN Mode Register 0



Symbol	Address	Reset Value
LMD0	44E04h	0000 0000b

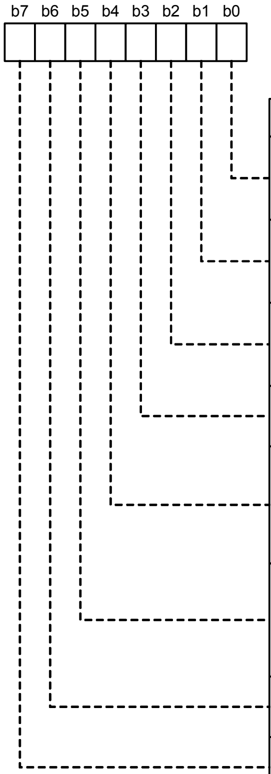
Bit Symbol	Bit Name	Function	RW
LMD	LIN Mode Select Bit [1]	0: LIN master mode 1: Do not set this value	RW
BTD	LIN T Bit Configuration Setting Bit [1]	0: 1 Tbit is generated by 8 fLIN [2] 1: 1 Tbit is generated by 16 fLIN	RW
— (b2)	No register bit; should be written with 0 and read as 0		—
FSM	Frame Separate Mode Select Bit [1, 3]	0: Non-frame separate mode 1: Frame separate mode	RW
FTCIE	Frame/Wake-up Transmit Completion Interrupt Enable Bit [1]	0: Frame/wake-up transmit completion interrupt disabled 1: Frame/wake-up transmit completion interrupt enabled	RW
FRCIE	Frame/Wake-up Receive Completion Interrupt Enable Bit [1]	0: Frame/wake-up receive completion interrupt disabled 1: Frame/wake-up receive completion interrupt enabled	RW
LDE	Input Signal Low Detection Enable Bit	0: Input signal low detection disabled 1: Input signal low detection enabled [4]	RW
ERRIE	Error Detection Interrupt Enable Bit [1]	0: Error detection interrupt disabled 1: Error detection interrupt enabled	RW

Notes:

1. These bits should be set when the corresponding channel is in LIN reset mode.
2. fLIN is the LIN module system clock.
3. This bit setting is enabled only when the RFT bit in the LRFC register is set to 1 (response transmission selected).
4. The LD bit in the LST register becomes 1 and an interrupt request is generated in the following cases:
   - When the falling edge of the input signal is detected when this bit is set to 1
   - When this bit is set to 1 while the input signal is low

Figure 24.7 LMD0 Register

LIN Mode Register 1



Symbol	Address	Reset Value
LMD1	44E05h	00h

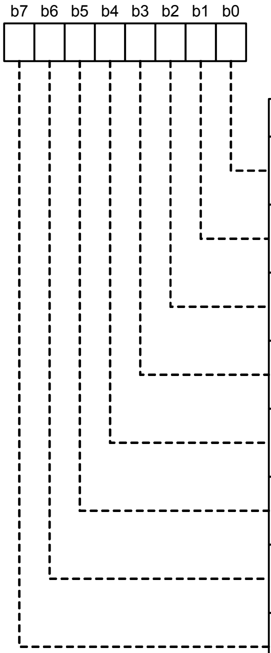
Bit Symbol	Bit Name	Function	RW
CKS0	LIN Module System Clock Select Bit (1)	b2 b1 b0 0 0 0 : Select fLn 0 0 1 : Select fLn2 0 1 0 : Select fLn8 0 1 1 : Select fLm 1 0 0 : Select fLm2 1 0 1 : Select fLm8 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
CKS1			RW
CKS2			RW
BERE	Bit Error Detection Enable Bit (2)	0: Bit error detection disabled 1: Bit error detection enabled	RW
PBERE	Physical Bus Error Detection Enable Bit (2)	0: Physical bus error detection disabled 1: Physical bus error detection enabled	RW
FTERE	Frame Time Out Error Detection Enable Bit (2)	0: Frame time out error detection disabled 1: Frame time out error detection enabled	RW
FERE	Framing Error Detection Enable Bit (2)	0: Framing error detection disabled 1: Framing error detection enabled	RW
OERE	Overrun Error Detection Enable Bit (2)	0: Overrun error detection disabled 1: Overrun error detection enabled	RW

Notes:

1. These bits should be set when the corresponding channel is in LIN reset mode.
2. These bits should be set when the FTS bit in the LTC register of corresponding channel is 0 (frame/wake-up transmission stopped).

Figure 24.8 LMD1 Register

LIN Wake-up Setting Register (1)



Symbol	Address	Reset Value
LWUP	44E06h	00h

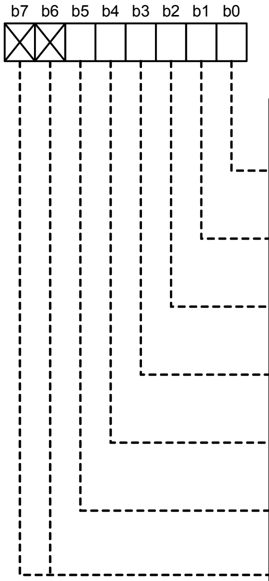
Bit Symbol	Bit Name	Function	RW
WURL0	Wake-up Reception Low Time Pulse Width Setting Bit (2)	b3 b2 b1 b0 0 0 0 0 : 0.5 Tbit 0 0 0 1 : 1.5 Tbit 0 0 1 0 : 2.5 Tbit 0 0 1 1 : 3.5 Tbit : : 1 1 0 0 : 12.5 Tbit 1 1 0 1 : 13.5 Tbit 1 1 1 0 : 14.5 Tbit 1 1 1 1 : 15.5 Tbit	RW
WURL1			RW
WURL2			RW
WURL3			RW
WUTL0	Wake-up Transmission Low Time Pulse Width Setting Bit	b7 b6 b5 b4 0 0 0 0 : 1 Tbit 0 0 0 1 : 2 Tbit 0 0 1 0 : 3 Tbit 0 0 1 1 : 4 Tbit : : 1 1 0 0 : 13 Tbit 1 1 0 1 : 14 Tbit 1 1 1 0 : 15 Tbit 1 1 1 1 : 16 Tbit	RW
WUTL1			RW
WUTL2			RW
WUTL3			RW

Notes:

1. This register should be set when the corresponding channel is in LIN reset mode or LIN operation mode.
2. The low-level pulse set by these bits or longer is recognized as wake-up signal.

Figure 24.9 LWUP Register

LIN Break Field Setting Register [1]



Symbol	Address	Reset Value
LBRK	44E08h	0000 0000b

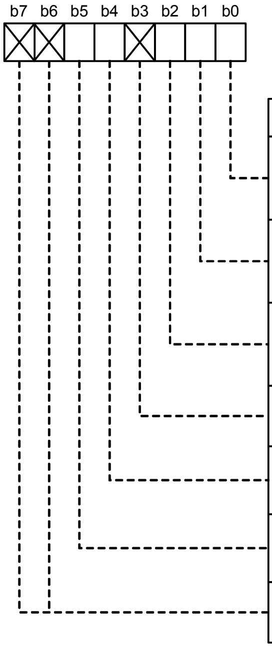
Bit Symbol	Bit Name	Function	RW
BLT0	Transmit Break (Low) Length Setting Bit	b3 b2 b1 b0 0 0 0 0 : 13 Tbit 0 0 0 1 : 14 Tbit 0 0 1 0 : 15 Tbit 0 0 1 1 : 16 Tbit : : 1 1 0 0 : 25 Tbit 1 1 0 1 : 26 Tbit 1 1 1 0 : 27 Tbit 1 1 1 1 : 28 Tbit	RW
BLT1			RW
BLT2			RW
BLT3			RW
BDT0	Transmit Break Delimiter (High) Length Setting Bit	b5 b4 0 0 : 1 Tbit 0 1 : 2 Tbit 1 0 : 3 Tbit 1 1 : 4 Tbit	RW
BDT1			RW
— (b7-b6)	No register bits; should be written with 0 and read as 0		—

Note:

1. This register should be set when the FTS bit in the LTC register of the corresponding channel is 0 (frame/ wake-up transmission stopped).

Figure 24.10 LBRK Register

LIN Space Setting Register (1)



Symbol
LSPC

Address
44E09h

Reset Value
0000 0000b

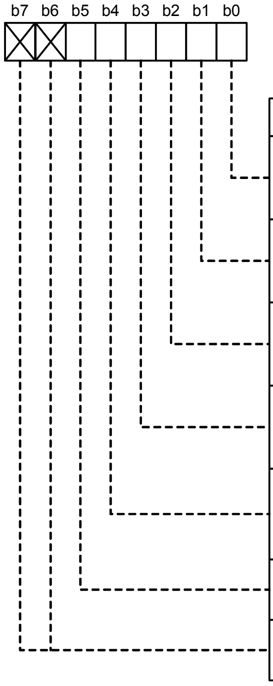
Bit Symbol	Bit Name	Function	RW
IBSH0	Interbyte Space (Header)/ Response Space Setting Bit	b2 b1 b0 0 0 0 : 0 Tbit 0 0 1 : 1 Tbit 0 1 0 : 2 Tbit 0 1 1 : 3 Tbit 1 0 0 : 4 Tbit 1 0 1 : 5 Tbit 1 1 0 : 6 Tbit 1 1 1 : 7 Tbit	RW
IBSH1			RW
IBSH2			RW
— (b3)	No register bit; should be written with 0 and read as 0		—
IBS0	Interbyte Space Setting Bit	b5 b4 0 0 : 0 Tbit 0 1 : 1 Tbit 1 0 : 2 Tbit 1 1 : 3 Tbit	RW
IBS1			RW
— (b7-b6)	No register bits; should be written with 0 and read as 0		—

Note:

1. This register should be set when the FTS bit in the LTC register of the corresponding channel is 0 (frame/ wake-up transmission stopped).

Figure 24.11 LSPC Register

LIN Response Field Setting Register [1]



Symbol
LRFC

Address
44E0Ah

Reset Value
0000 0000b

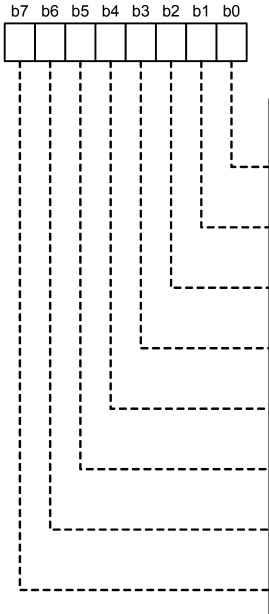
Bit Symbol	Bit Name	Function	RW
RFDL0	Response Field Data Length Setting Bit	b3 b2 b1 b0 0 0 0 0 : 0 bytes + checksum 0 0 0 1 : 1 byte + checksum 0 0 1 0 : 2 bytes + checksum 0 0 1 1 : 3 bytes + checksum 0 1 0 0 : 4 bytes + checksum 0 1 0 1 : 5 bytes + checksum 0 1 1 0 : 6 bytes + checksum 0 1 1 1 : 7 bytes + checksum 1 0 0 0 : 8 bytes + checksum Only use the combinations listed above	RW
RFDL1			RW
RFDL2			RW
RFDL3			RW
RFT	Response Field Transmit/ Receive Direction Setting Bit	0: Reception (data buffer is read only) 1: Transmission (data buffer is write only)	RW
CSM	Checksum Model Select Bit	0: Classic checksum 1: Enhanced checksum	RW
— (b7-b6)	No register bits; should be written with 0 and read as 0		—

Note:

1. This register should be set when the FTS bit in the LTC register of the corresponding channel is 0 (frame/ wake-up transmission stopped).

Figure 24.12 LRFC Register

LIN ID Buffer Register (1)



Symbol
LIDB

Address
44E0Bh

Reset Value
00h

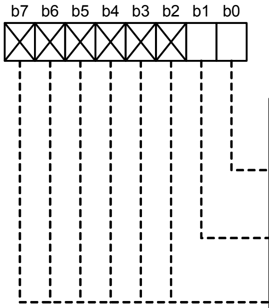
Bit Symbol	Bit Name	Function	RW
ID0	ID Setting Bit	Set a 6-bit ID value to be transmitted by ID field	RW
ID1			RW
ID2			RW
ID3			RW
ID4			RW
ID5			RW
IDP0	Parity 0 Setting Bit	Set the parity bit (P0) to be transmitted by ID field	RW
IDP1	Parity 1 Setting Bit	Set the parity bit (P1) to be transmitted by ID field	RW

Note:

1. This register should be set when the FTS bit in the LTC register of the corresponding channel is 0 (frame/wake-up transmission stopped).

Figure 24.13 LIDB Register

LIN Status Control Register



Symbol
LSC

Address
44E0Ch

Reset Value
0000 0000b

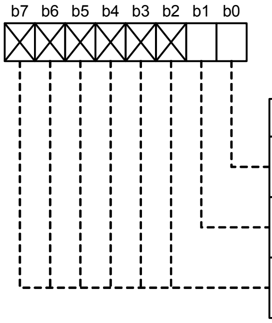
Bit Symbol	Bit Name	Function	RW
OM0	Operation Mode Select Bit (1)	b1 b0 0 0 : LIN reset mode 0 1 : LIN wake-up mode 1 0 : LIN reset mode 1 1 : LIN operation mode	RW
OM1			RW
— (b7-b2)	No register bits; should be written with 0 and read as 0		—

Note:

1. The channel status can be checked by reading the value of bits OMM1 and OMM0 in the LST register of the corresponding channel.

Figure 24.14 LSC Register

LIN Transmission Control Register



Symbol	Address	Reset Value
LTC	44E0Dh	0000 0000b

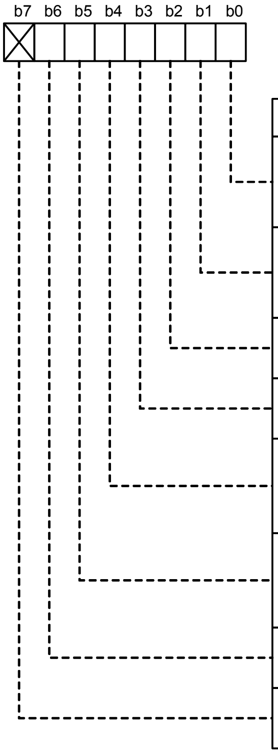
Bit Symbol	Bit Name	Function	RW
FTS	Frame/Wake-up Transmission Start Bit (1, 2)	0: Stop frame/wake-up transmission 1: Start frame/wake-up transmission	RW
RTS	Response Transmission Start Bit (1, 2, 3)	0: Stop response transmission 1: Start response transmission	RW
— (b7-b2)	No register bits; should be written with 0 and read as 0		—

Notes:

1. These bits can only be set to 1. They automatically become 0 when the transmission is completed.
2. These bits are set to 0 when transiting to LIN reset mode or LIN wake-up mode.
3. This bit should be set when bits FSM and FTS in the LMD0 register of the corresponding channel are both 1 (frame separate mode selected, and frame/wake-up transmission started).

Figure 24.15 LTC Register

LIN Status Register



Symbol	Address	Reset Value
LST	44E0Eh	0000 0000b

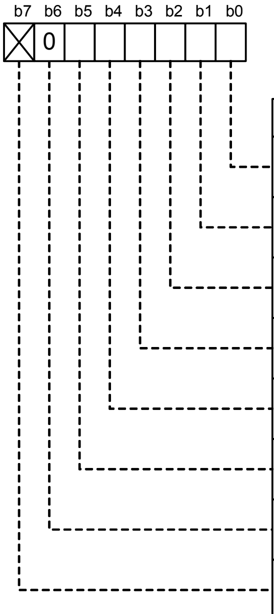
Bit Symbol	Bit Name	Function	RW
FTC	Frame/Wake-up Transmit Completion Flag (1)	0: Transmission incomplete 1: Transmission of frame or wake-up has been completed (2)	RW
FRC	Frame/Wake-up Receive Completion Flag (1)	0: Reception incomplete 1: Reception of frame or wake-up has been completed	RW
LD	Input Signal Low Detection Interrupt Status Flag (1)	0: No interrupts requested 1: Interrupts requested (3)	RW
ERR	Error Detection Flag	0: Error not detected 1: Error detected	RO
OMM0	Operation Mode Monitor Flag	b5 b4 0 0 : LIN reset mode (operation stopped) 0 1 : LIN reset mode (operation stopped) 1 0 : LIN wake-up mode 1 1 : LIN operation mode	RO
OMM1			RO
D1RC	Data 1 Receive Completion Flag (1)	0: Reception incomplete 1: Data 1 reception completed	RW
— (b7)	No register bit; should be written with 0 and read as 0		—

Notes:

1. These bits do not become 0 automatically. Set them to 0 by a program. Writing 1 to these bits has no effect.
2. This flag does not become 1 when a header transmission is completed in frame separate mode.
3. When this bit is 1, no new interrupt request is generated in the following cases:
   - When the LD bits of other channels become 1.
   - When the conditions for the LD bit to become 1 are met in its channel.

Figure 24.16 LST Register

LIN Error Status Register



Symbol	Address	Reset Value
LEST	44E0Fh	0000 0000b

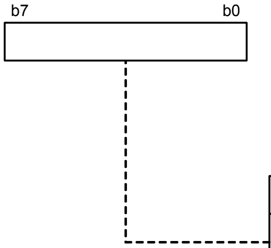
Bit Symbol	Bit Name	Function	RW
BER	Bit Error Flag [1]	0: Bit error not detected 1: Bit error detected	RW
PBER	Physical Bus Error Flag [1]	0: Physical bus error not detected 1: Physical bus error detected	RW
CSER	Checksum Error Flag [1]	0: Checksum error not detected 1: Checksum error detected	RW
FTER	Frame Timeout Error Flag [1]	0: Frame timeout error not detected 1: Frame timeout error detected	RW
FER	Framing Error Flag [1]	0: Framing error not detected 1: Framing error detected	RW
OER	Overrun Error Flag [1]	0: Overrun error not detected 1: Overrun error detected	RW
— (b6)	Reserved	Should be written with 0	RW
— (b7)	No register bit; should be written with 0 and read as 0		—

Note:

1. These bits do not become 0 automatically. Set them to 0 by a program. Writing 1 to these bits has no effect.

Figure 24.17 LEST Register

LIN Data n Buffer Register (n = 1 to 8) [1]



Symbol	Address	Reset Value
LDB1, LDB2	44E10h, 44E11h	00h
LDB3, LDB4	44E12h, 44E13h	00h
LDB5, LDB6	44E14h, 44E15h	00h
LDB7, LDB8	44E16h, 44E17h	00h

Function	Setting Range	RW
Set the data to be transmitted or read the received data	00h to FFh	RW

Note:

1. This register should be set in the following cases:
   - The RFT bit in the LRFC register of the corresponding channel is 1 (transmission).
   - The FSM bit in the LMD0 register of the corresponding channel is 0 (non-frame separate mode).
   - The FTS bit in the LTC register of the corresponding channel is 0 (frame/wake-up transmission stopped).

   or
   - The RFT bit of the corresponding channel is 1 (transmission).
   - The FSM bit of the corresponding channel is 1 (frame separate mode).
   - The RTS bit in the LTC register of the corresponding channel is 0 (response transmission stopped).

Figure 24.18 Registers LDB1 to LDB8

24.2 Operational Modes

The LIN modules have the following three operational modes:

- LIN reset mode
- LIN operation mode
- LIN wake-up mode

Entering operation modes is controlled for each channel. When all channels are in LIN reset mode, power consumption is reduced since the clock is not provided to the LIN module. Figure 24.19 shows the transition processes between LIN operational modes and Table 24.2 lists functions available in each mode.

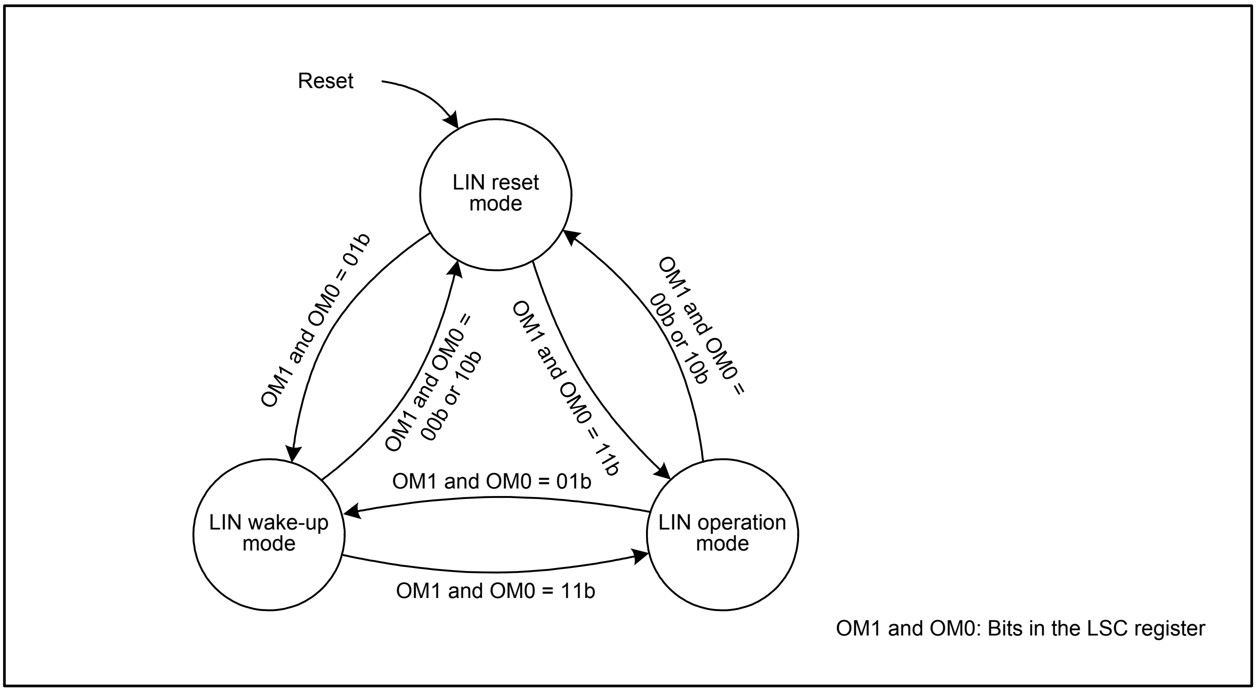


Figure 24.19 LIN Operational Mode Transition

Table 24.2 Functions Available in Each Operational Mode

LIN Reset Mode	LIN Operation Mode	LIN Wake-up Mode
Input signal low detection	Header transmission Response transmission Response reception Error detection Input signal low detection	Wake-up transmission Wake-up reception Error detection Input signal low detection

Read bits OMM1 and OMM0 in the LST register of the corresponding channel to verify that the LIN module has entered each operation mode.

Note that bits OM1 and OM0 and bits OMM1 and OMM0 are allocated differently in each operational mode.

24.2.1 LIN Reset Mode

The LIN reset mode for the selected channel is activated by setting bits OM1 and OM0 in the LSC register of the corresponding channel to 00b or 10b, and bits OMM1 and OMM0 in the LST register become 00b or 01b. In this mode, all LIN functions of the corresponding channel are stopped as well as the clock supply to the LIN module (fLIN) of the channel.

24.2.2 LIN Operation Mode

The LIN operation mode for the selected channel is activated by setting bits OM1 and OM0 in the LSC register of the corresponding channel to 11b, and bits OMM1 and OMM0 in the LST register become 11b.

24.2.3 LIN Wake-up Mode

The LIN wake-up mode for the selected channel is activated by setting bits OM1 and OM0 in the LSC register of the corresponding channel to 01b, and bits OMM1 and OMM0 in the LST register become 10b.

24.3 Operational Overview

24.3.1 Header Transmission

Figure 24.20 shows the operation in header transmission and Table 24.3 lists the processing in header transmission.

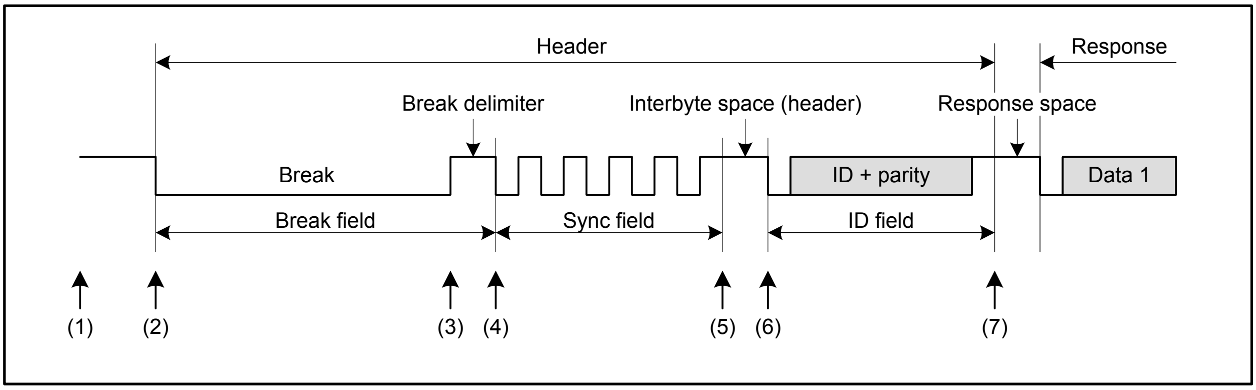


Figure 24.20 Operation in Header Transmission

Table 24.3 Processing in Header Transmission

	Software Processing	LIN Module Processing
(1)	• Set a baud rate • Set the following bits in the LMD0 register: FTCIE bit to 1 (frame/wake-up transmit completion interrupt enabled); FRCIE bit to 1 (frame/wake-up receive completion interrupt enabled); ERRIE bit to 1 (error detection interrupt enabled) • Set bits OM1 and OM0 in the LSC register for LIN mode operation • Set the following bits in the LBRK register: Bits BLT3 to BLT0 for break low (13 to 28 Tbit); Bits BDT1 and BDT0 for break delimiter (1 to 4 Tbit) • Set the following bits in the LSPC register: Bits IBSH2 to IBSH0 for interbyte space (Header)/response space (0 to 7 Tbit); Bits IBS1 and IBS0 for interbyte space (0 to 3 Tbit) • Set the following bits in the LIDB register: Bits ID5 to ID0 for ID value; Bits IDP1 and IDP0 for parity value • Set the following bits in the LRFC register: Bits RFDL3 to FRDL0 for data length; RFT bit for response transmit/receive direction CSM bit for checksum model • Set the transmit data	Wait for frame/wake-up transmission start by software (idle)
(2)	• Set the FTS bit in the LTC register to 1 (frame/wake-up transmission started)	Transmit break
(3)	• Wait for an interrupt request to be generated	Transmit break delimiter
(4)		Transmit Sync field (55h)
(5)		Transmit interbyte space (Header)
(6)		Transmit ID field
(7)		Transmit response space

24.3.2 Response Transmission

Figure 24.21 shows the operation in response transmission and Table 24.4 lists the processing in response transmission.

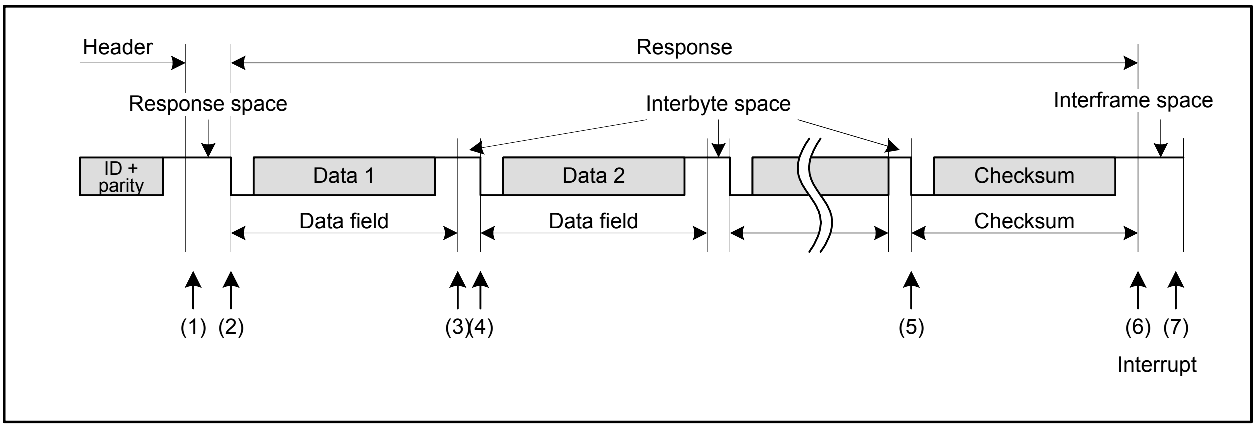


Figure 24.21 Operation in Response Transmission

Table 24.4 Processing in Response Transmission

	Software Processing	LIN Module Processing
(1)	• In frame separate mode Set the RTS bit in the LTC register to 1 (response transmission started) • In non-frame separate mode Wait for an interrupt request to be generated	• In frame separate mode Transmit response space while waiting for response transmission enabled • In non-frame separate mode Go to (2) if the response space transmission is completed
(2)	Wait for an interrupt request to be generated	Transmit Data 1
(3)		Transmit interbyte space
(4)		Transmit Data 2, then the next interbyte space Transmit Data 3, then the next interbyte space (Repeat this process for the data length specified in bits RFDL3 to RFDL0 in the LRFC register. Go to (6) if an error occurs.)
(5)		Transmit checksum
(6)		• Set frame/wake-up transmit completion flag or error flag • Set the following bits in the LTC register: FTS bit = 0 (frame/wake-up transmission stopped); RTS bit = 0 (response transmission stopped)
(7)	• Processing after communication Check the LST register and set flags to 0	Idle

24.3.3 Response Reception

Figure 24.22 shows the operation in response reception and Table 24.5 lists the processing in response reception.

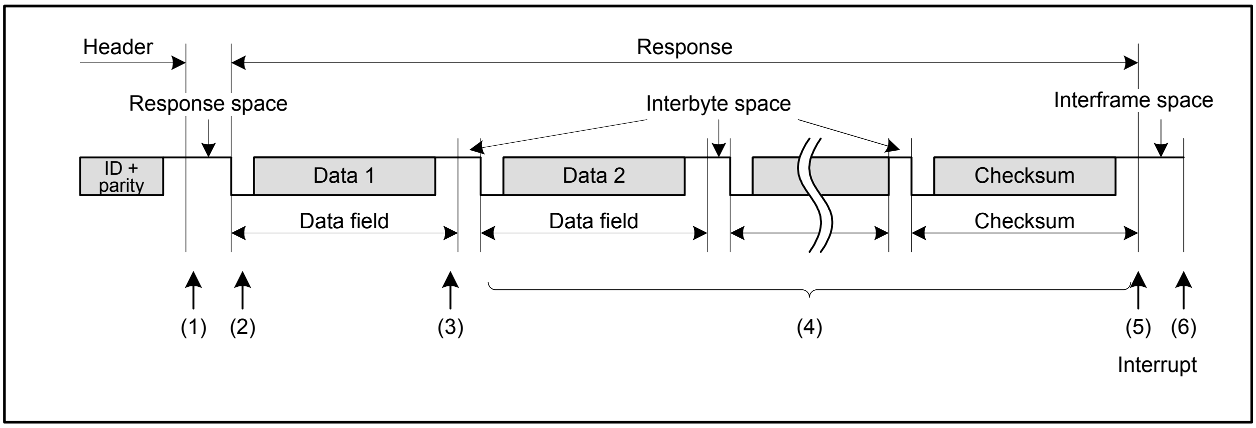


Figure 24.22 Operation in Response Reception

Table 24.5 Processing in Response Reception

	Software Processing	LIN Module Processing
(1)	Wait for an interrupt request to be generated (without processing)	Wait for a start bit to be detected
(2)	Wait for an interrupt request to be generated	Receive Data 1 due to start bit detection
(3)		• Set Data 1 receive completion flag
(4)		Receive Data 2 due to start bit detection Receive Data 3 due to start bit detection (Repeat this process for the data length specified in bits RFDL3 to RFDL0 in the LRFC register. Abort the reception and go to (5) if an error occurs. Checksum judgement is not performed in this case.) : : Receive checksum due to start bit detection
(5)		• Judge the checksum • Set frame receive completion flag or error flag • Set the FTS bit in the LTC register to 0 (frame/wake-up transmission stopped)
(6)	Processing after communication Check the LST register and set flags to 0	Idle

24.4 Baud Rate Generator

The LIN module system clock (fLIN) is generated by peripheral clock divided by the baud rate generator. The fLIN divided by 8 or 16 is used as the bit rate. The reciprocal of the bit rate is called bit time and expressed as “Tbit”.

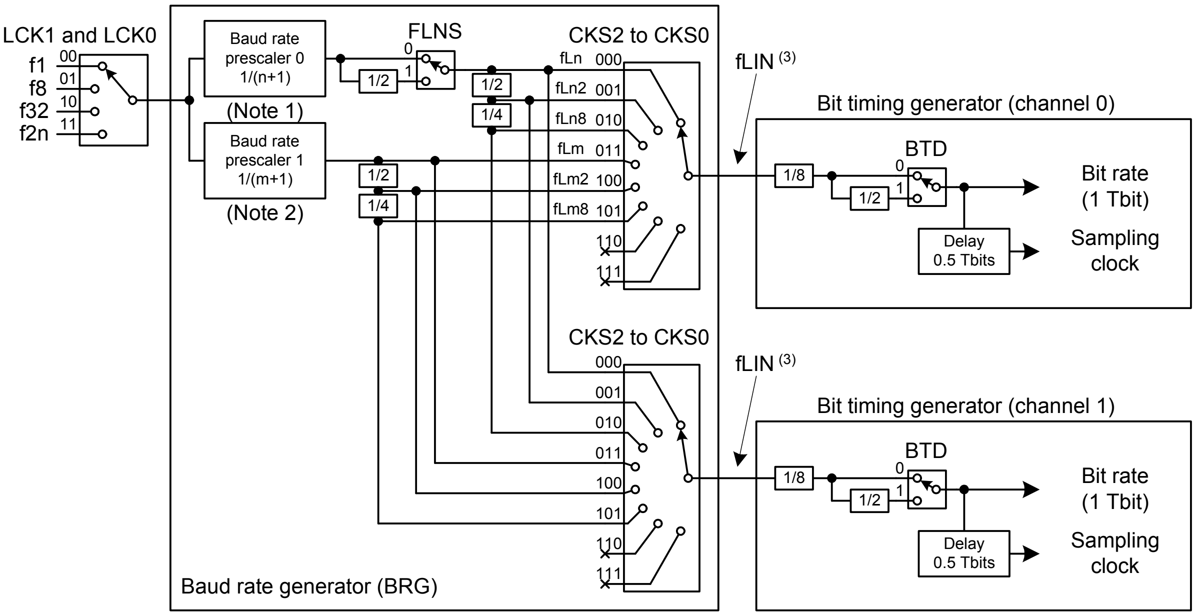
When the BTD bit in the LMD0 register is set to 0 (1 Tbit is generated by 8 fLIN), the FLNS bit in registers LBRG and LBRP0 should be set so that fLn is 153600 Hz (19200 bps × 8). Therefore,

fLn = 19200 bps × 8 [Hz]
fLn2 = 9600 bps × 8 [Hz]
fLn8 = 2400 bps × 8 [Hz]

Then they are divided by 8 in the bit timing generator, which results in 19200 bps, 9600 bps, and 2400 bps, respectively. The baud rate of 10417 bps is generated by the bit setting in the LBRP1 register.

When the BTD bit is set to 1 (1 Tbit is generated by 16 fLIN), the FLNS bit in registers LBRG and LBRP0 should be set so fLn is 307200 Hz (= 19200 bps × 16).

Figure 24.23 shows the block diagram of baud rate generation.



Notes:
1. Divide-by-(n+1) (n = 1 to 255) is set by the LBRP0 register.
2. Divide-by-(m+1) (m = 1 to 255) is set by the LBRP1 register.
3. Set fLIN to a frequency lower than the peripheral bus clock divided by 2.

fLIN: LIN module system clock
LCK1, LCK0, and FLNS: Bits in the LBRG register
BTD: Bit in the LMD0 register
CKS2 to CKS0: Bits in the LMD1 register

Figure 24.23 Baud Rate Generation Block Diagram

Table 24.6 and Table 24.7 list the baud rates (19200, 9600, 2400, and 10417 bps) generated by the peripheral clock frequency, and their errors.

Table 24.6 Baud Rate Generation Example (for 19200, 9600, and 2400 bps)

Peripheral Clock	Divide-by-(n + 1)	BRP0 Output	Bit Times	fLn [19200 bps] (unit: bps)	fL2n [9600 bps] (unit: bps)	fL8n [2400 bps] (unit: bps)	Error (unit: %)
32 MHz	104	No division/ divide-by-2	16 fLIN/ 8 fLIN	19230.77	9615.38	2403.85	+0.16
30 MHz	98	No division/ divide-by-2	16 fLIN/ 8 fLIN	19132.65	9566.33	2391.58	-0.35
24 MHz	78	No division/ divide-by-2	16 fLIN/ 8 fLIN	19230.77	9615.38	2403.85	+0.16
20 MHz	65	No division/ divide-by-2	16 fLIN/ 8 fLIN	19230.77	9615.38	2403.85	+0.16
16 MHz	52	No division/ divide-by-2	16 fLIN/ 8 fLIN	19230.77	9615.38	2403.85	+0.16
12 MHz	39	No division/ divide-by-2	16 fLIN/ 8 fLIN	19230.77	9615.38	2403.85	+0.16
10 MHz	65	No division (1)	8 fLIN	19230.77	9615.38	2403.85	+0.16
8 MHz	26	No division/ divide-by-2	16 fLIN/ 8 fLIN	19230.77	9615.38	2403.85	+0.16
4 MHz	13	No division/ divide-by-2	16 fLIN/ 8 fLIN	19230.77	9615.38	2403.85	+0.16
2 MHz	13	No division (1)	8 fLIN	19230.77	9615.38	2403.85	+0.16

Note:

1. An error does not fulfill the LIN protocol specification (±0.5%) if the BRP0 clock is divided by 2.

Table 24.7 Baud Rate Generation Example for 10417 bps

Peripheral Clock	Divide-by-(m + 1)	Bit Times	fLIN	Baud Rate [10417 bps] (unit: bps)	Error (unit:%)
32 MHz	48 / 96	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
30 MHz	45 / 90	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
24 MHz	36 / 72	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
20 MHz	30 / 60	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
16 MHz	24 / 48	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
12 MHz	18 / 36	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
10 MHz	15 / 30	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
8 MHz	12 / 24	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
4 MHz	6 / 12	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003
2 MHz	3 / 6	8 fLIN / 16 fLIN	fLm8 / fLm2	10416.67	-0.003

24.5 Data Transmission and Reception

24.5.1 Data Transmission

The LIN module transmits 1-bit data per Tbit.

The transmitted data returns to the input pin for data reception via the LIN transceiver. Then the received data is compared to the transmitted data and the result is stored in the BER bit in the LEST register (refer to 24.10 “Error Status” for details). The timing to compare the received data is at the fifth clock when the BTD bit in the LMD0 register is 0 (1 Tbit is generated by 8 fLIN), and at the 13th clock when the BTD bit is 1 (1 Tbit is generated by 16 fLIN).

Figure 24.24 shows data transmission timing.

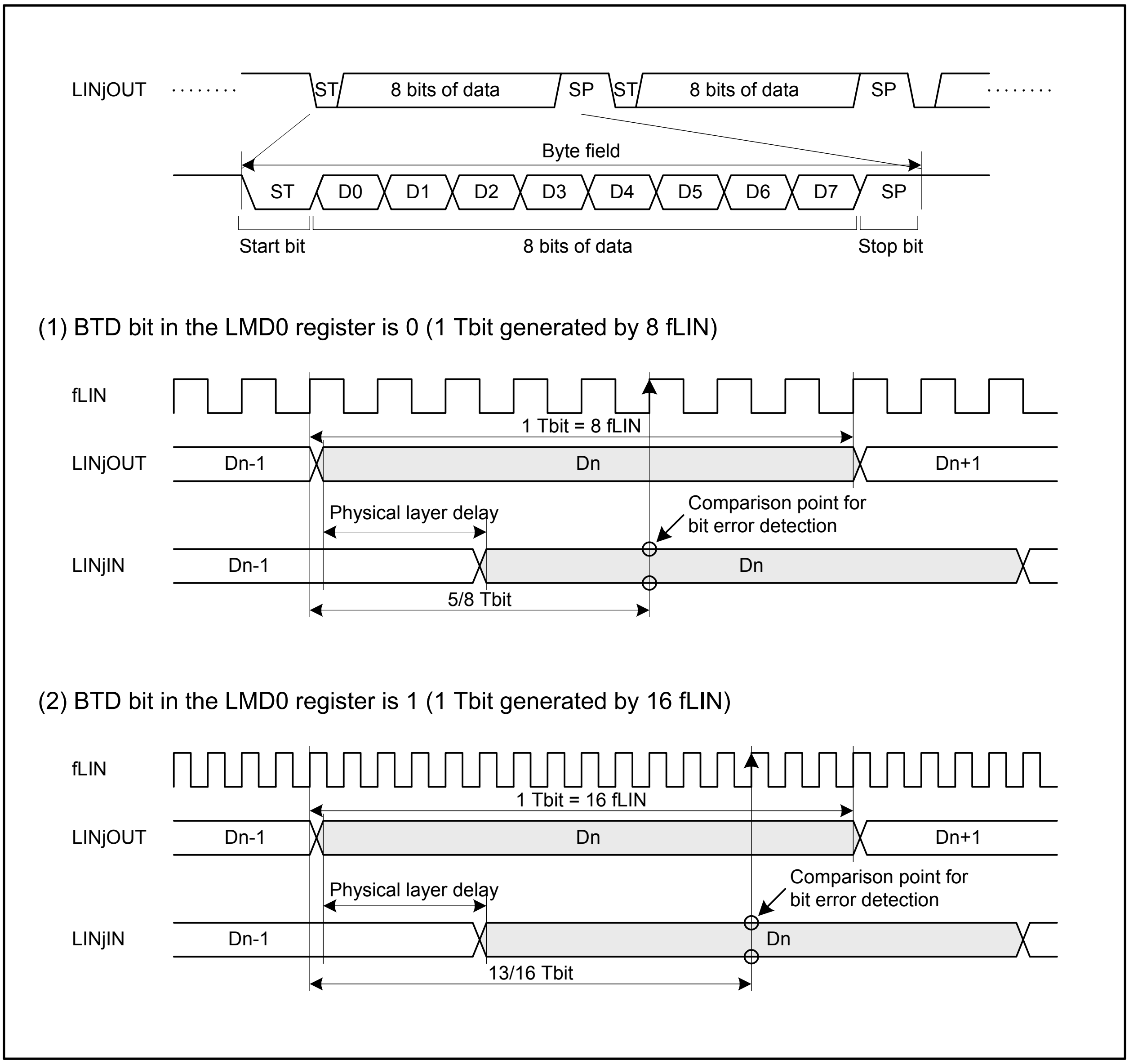


Figure 24.24 Data Transmission Timing (j = 0, 1)

24.5.2 Data Reception

Data reception in the LIN module requires an internal signal generated from the input signal at the LINjIN (j = 0, 1) pin by a double-latch clocking scheme.

The byte field of this internal signal, called the synchronized LINjIN, is synchronized with fLIN at the falling edge of the start bit. The start bit is verified if the synchronized LINjIN signal is held low for 0.5 Tbit after the falling edge is detected. If the signal remains low after exiting reset mode or if it is held high on sampling, no start bit is detected.

Once the start bit is detected, data bits are sampled every Tbit.

Figure 24.25 shows the data reception timing.

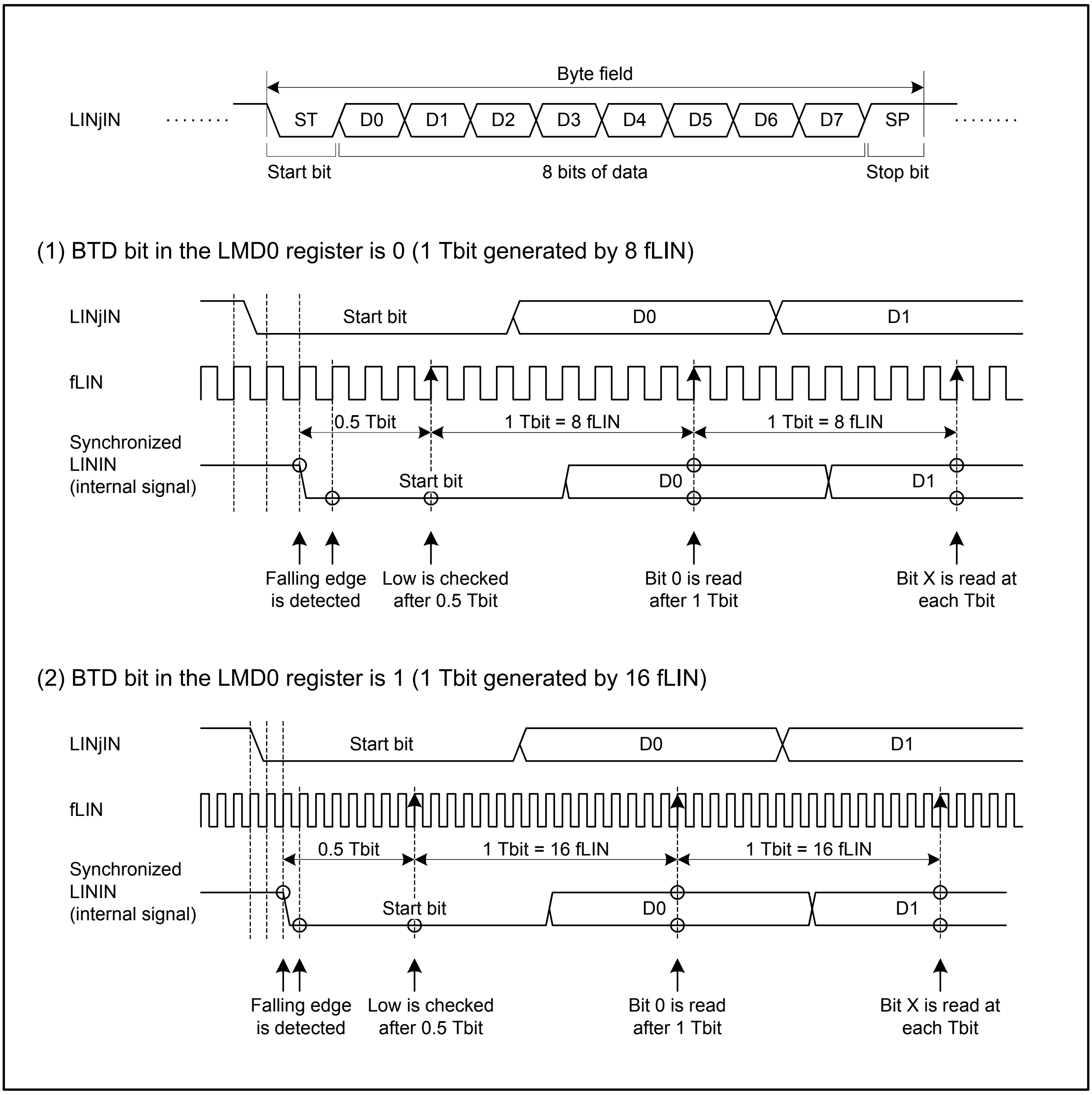


Figure 24.25 Data Reception Timing (j = 0, 1)

24.6 Buffer Processing of Transmit/Receive Data

This section describes buffer processing when the LIN module is continuously transmitting and receiving data.

24.6.1 Transmission of LIN Frame

In 8-byte transmission, the values stored in registers LDB1 to LDB8 of each channel are transmitted in order to Data 1 to Data 8 of the LIN frame.
In 4-byte transmission, the values stored in registers LDB1 to LDB4 are transmitted to Data 1 to Data 4 of the LIN frame, and the values stored in registers LDB5 to LDB8 are not transmitted.
Figure 24.26 shows the processing of LIN transmission and buffer.

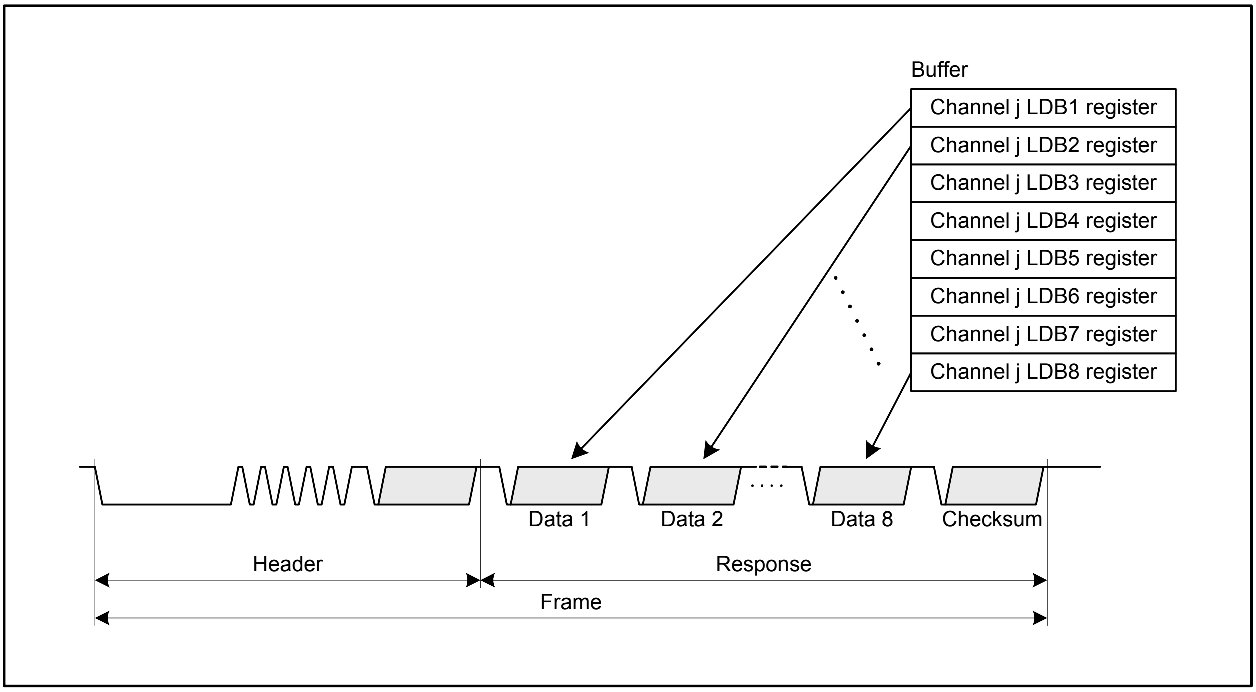


Figure 24.26 Processing of LIN Transmission and Buffer (j = 0, 1)

24.6.2 Reception of LIN Frame

In 8-byte reception, the values of Data 1 to Data 8 of the LIN frame are stored in registers LDB1 to LDB8 of each channel in order at every completion of stop bit reception. In 4-byte reception, the values of Data 1 to Data 4 of the LIN frame are stored in registers LDB1 to LDB4 respectively, and no bits are stored in registers LDB5 to LDB8.

Figure 24.27 shows the processing of LIN reception and buffer.

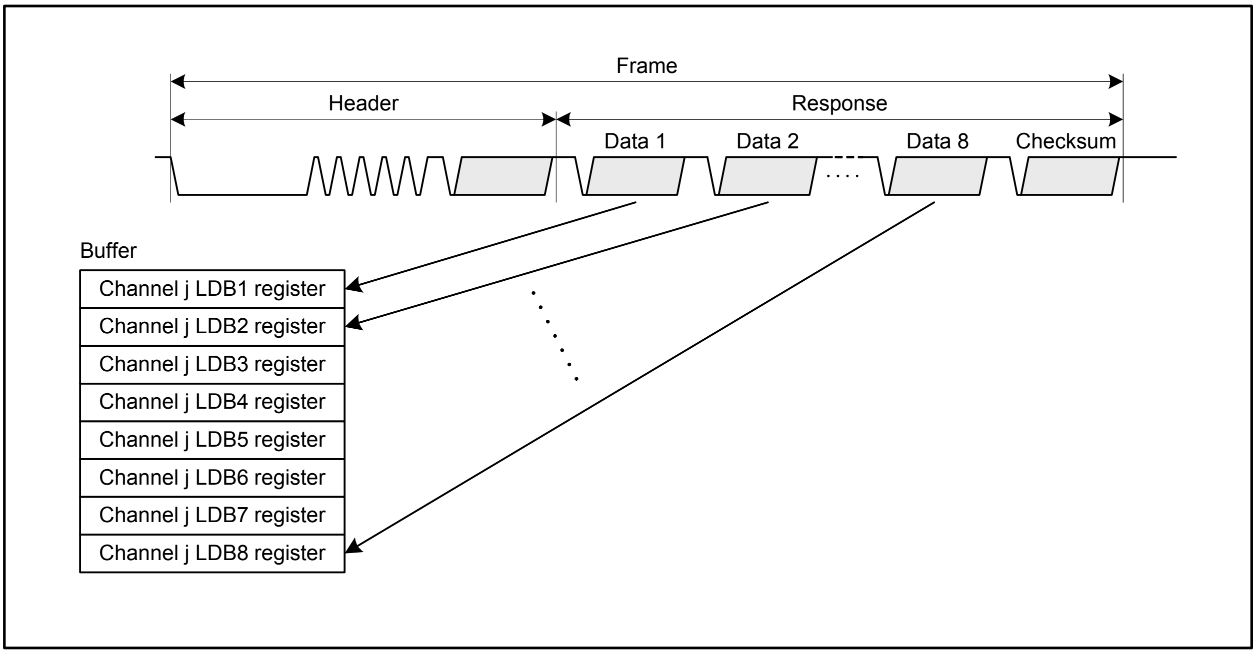


Figure 24.27 Processing of LIN Reception and Buffer (j = 0, 1)

24.7 Wake-up Transmission and Reception

Wake-up transmission and reception can be used in LIN wake-up mode.

24.7.1 Wake-up Transmission

24.7.1.1 Operation of Wake-up Transmission

In LIN wake-up mode, if the FTS bit in the LTC register is set to 1 (frame/wake-up transmission started), the wake-up signal is output at the output pin of the selected channel. Low time pulse width of the wake-up signal is set by bits WUTL3 to WUTL0 in the LWUP register.

When low of wake-up is output without any bit error detected, the FTC bit in the LST register becomes 1 (frame/wake-up transmission completed). If the FTCIE bit in the LMD0 register is set to 1 (frame/wake-up transmit completion interrupt enabled), an interrupt request is generated.

When a bit error is detected, the operation is aborted and the BER bit in the LEST register becomes 1 (bit error is detected).

Figure 24.28 shows the wake-up transmission timing.

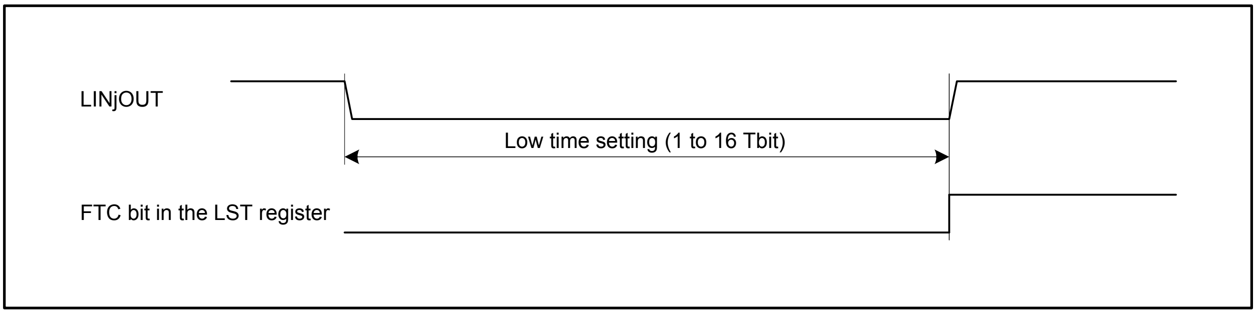


Figure 24.28 Wake-up Transmission Timing (j = 0, 1)

24.7.1.2 Wake-up Collision

When the master and slave transmit wake-up signals simultaneously, a signal collision occurs on the LIN bus. However, no collision of the wake-up signals is detected in the LIN module.

Even if the LIN bus is fixed low, it is not recognized as a wake-up signal: the input signal low time count does not function unless a recessive (high) is detected once after the wake-up signals are transmitted.

24.7.2 Wake-up Reception

24.7.2.1 Wake-up Reception Operation

A wake-up is detected by either of following functions: Input signal low time count or input signal low detection. The input signal low time count is available in LIN wake-up mode and the input signal low detection is available in all LIN operational modes.

The input signal low time count measures the low time of an input signal at the LINjIN pin (j = 0, 1) by counting at the same sampling timing as that of data reception. When the counted low time reaches the preset value by bits WURL3 to WURL0 in the LWUP register, the FRC bit in the LST register becomes 1 (frame/wake-up reception completed). When the FRCIE bit in the LMD0 register is set to 1 (frame/wake-up receive completion interrupt enabled), an interrupt request is generated.

On the other hand, the input signal low detection is to asynchronously detect the falling edge of the input signal at the LINjIN pin. While the LDE bit in the LMD0 register is set to 1 (input signal low detection enabled), the LD bit in the LST register becomes 1 (input signal low detected) when the falling edge is detected, and an interrupt request is generated.

Figure 24.29 shows an example of wake-up detection.

When the bit rate is 19200 bps (1 Tbit = 52 µs) and the pulse width is 3.5 Tbit (bits WURL3 to WURL0 is 0011b), the low time pulse over 182 µs (= 52 µs × 3.5) is recognized as the wake-up signal.

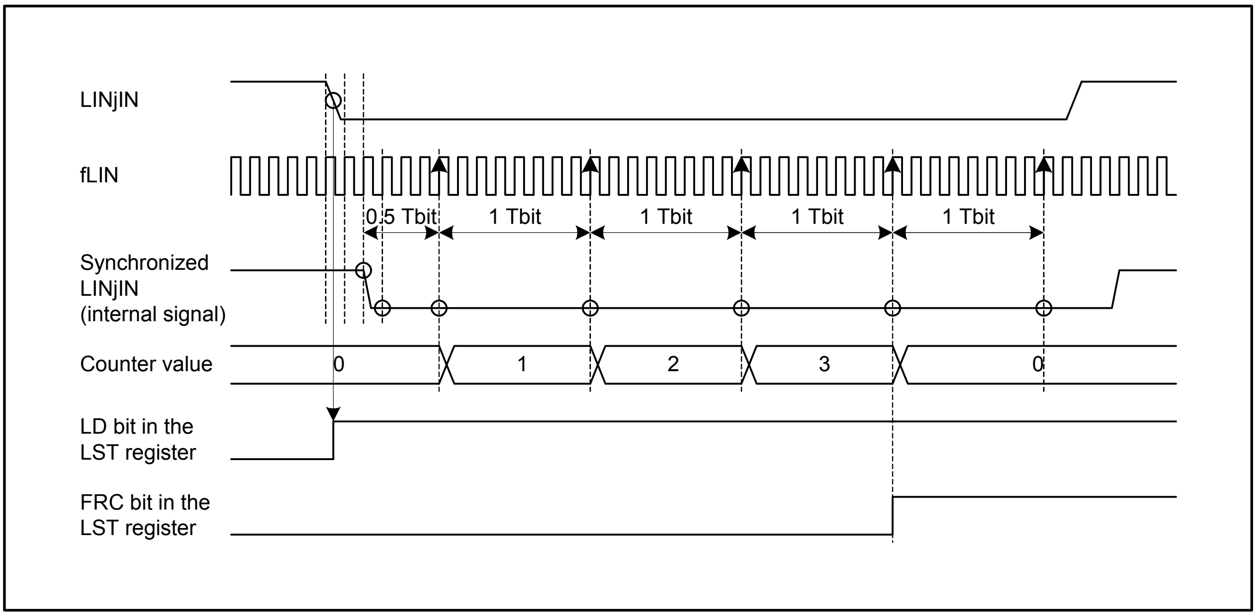


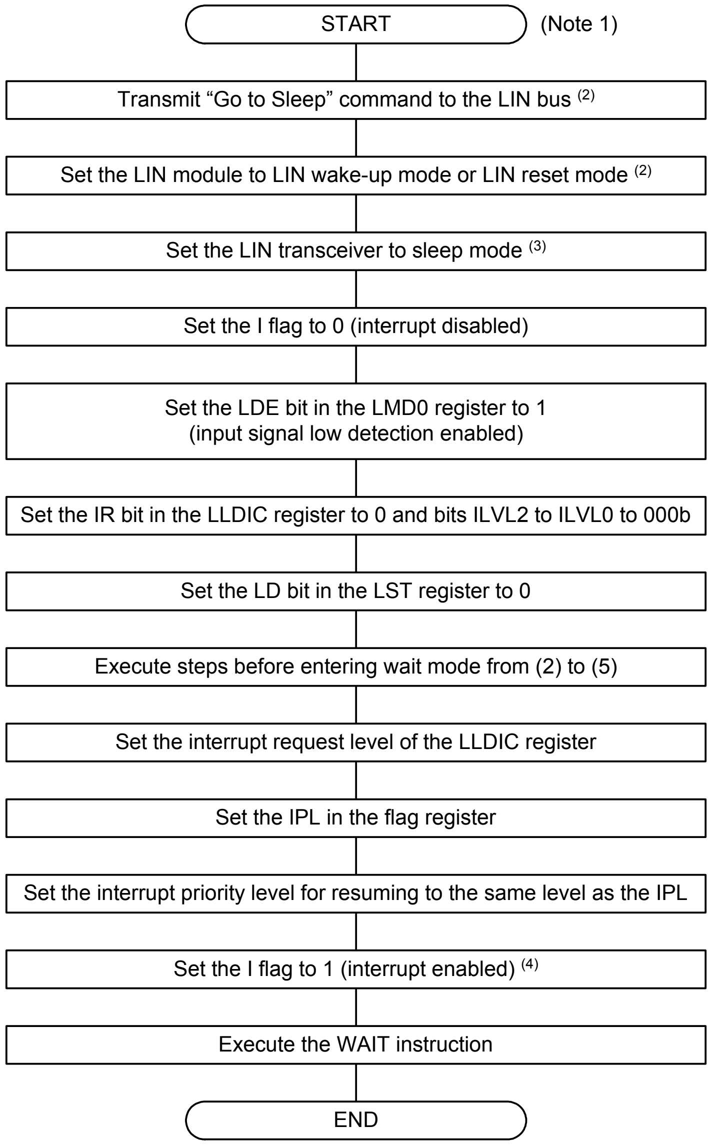
Figure 24.29 Wake-up Detection (j = 0, 1)

The input signal low time count is unavailable during wake-up transmission processing. The input signal low detection functions. If the FTS bit in the LTC register is set to 1 (start frame/wake-up transmission) while a wake-up reception is being counted, the reception processing is aborted to perform the wake-up transmission.

24.8 Low Power Mode Control Using Input Signal Low Detection

The input signal low detection can be used as the interrupt to exit wait mode or stop mode.

Figure 24.30 shows an example of setting before entering wait mode. Refer to 7.7.2 “Wait Mode” and 7.7.3 “Stop Mode” for details on entering wait or stop mode.



Notes:

1. This flowchart shows an example of the setting procedure to exit wait mode using a wake-up signal from the LIN bus after entering wait mode.
2. Set the LIN module in PLL mode (high speed mode or medium speed mode).
3. The setting depends on the LIN transceiver specification.
4. Enter low speed mode or low power mode before setting the I flag to 1.

Figure 24.30 Example of Setting Before Transition to Wait Mode

24.9 Operational Status

The LIN module detects six types of operational statuses.

Statuses that can generate an interrupt request are: frame/wake-up transmit completion, frame/wake-up receive completion, input signal low detection, and error detection.

Table 24.8 lists the types of operational statuses.

Table 24.8 Operational Statuses (j = 0, 1)

Operational Status	Detecting Condition	Detectable Operational Modes	Corresponding Bits in the LST Register
Frame/wake-up transmit completion	When a response frame transmission has been successfully completed in the bit setting as follows: RFT bit in the LRFC register is 1 (transmission selected)	LIN operation mode	FTC
	When a wake-up transmission has been successfully completed	LIN wake-up mode	
Frame/wake-up receive completion	When a response frame reception has been successfully completed in the bit setting as follows: RFT bit in the LRFC register is 0 (reception selected)	LIN operation mode	FRC
	When the low time of the input signal reaches the preset value by bits WURL3 to WURL0 in the LWUP register	LIN wake-up mode	
Input signal low detection	When the falling edge of input signal at the LINjIN pin is detected with the setting of the LDE bit in the LMD0 register to 1 (Input signal low detection enabled), or when setting the LDE bit to 1 while the LINjIN pin is low	All LIN modes	LD
Error detection	When any bit from 0 to 5 in the LEST register becomes 1 (error detected)	LIN operation mode LIN wake-up mode	ERR
Operation mode	When LIN module has entered the operational mode which was set by bits OM1 and OM0 in the LSC register	All LIN modes	OMM1 and OMM0
Data 1 receive completion	When the reception of the first byte of a response frame has been completed in the bit setting as follows: RFT bit in the LRFC register is 0 (reception selected) (1)	LIN operation mode	D1RC

Note:

1. This status is detected except when bits RFDL3 to RFDL0 in the LRFC register are set to 0000b (0 bytes + checksum).

24.10 Error Status

24.10.1 Error Status Types

The LIN module detects six types of error statuses. These error statuses can be checked by bits 0 to 5 in the LEST register.

Table 24.9 lists the types of error statuses.

Table 24.9 Error Statuses

Error Status	Error Detecting Condition	Error Detectable Operational Mode	Communication Processing	Detection Enabled/ Disabled	Corresponding Bit in the LEST Register
Bit error	When the transmitted data does not match with that on the LIN bus monitored by the pin for reception	LIN operation mode LIN wake-up mode	Abort (1)	Selectable	BER
Physical bus error	When LIN bus has already become low just before the break field is transmitted or when it is detected that LIN bus is being fixed to high	LIN operation mode LIN wake-up mode	Abort	Selectable	PBER
Check sum error	When the checksum judgement of the response frame reception processing results in an error	LIN operation mode	—	Not selectable	CSER
Frame timeout error	When the transmit/receive operation is not completed within a specified period of time (2)	LIN operation mode	Abort	Selectable	FTER
Framing error	When the stop bit of each data byte is low in response frame reception processing	LIN operation mode	Abort	Selectable	FER
Overrun error	When the first data byte of the next response frame is received while the FRC bit in the LST register is 1 (frame reception successfully completed)	LIN operation mode	Abort	Selectable	OER

Notes:

1. When a bit error is detected, the processing is aborted after the stop bit is transmitted. If it is detected in a non-data space, such as an interbyte space, the transmission processing is aborted after a check in that space is completed. During wake-up transmission, the wake-up transmission is aborted immediately after the error bit is transmitted.
2. The period of time (as time out value) depends on the response field data length set by bits RFDL3 to RFDL0 in the LRFC register as follows:

$$\text{Time out value} = 50 + (\text{data bytes} + 1) \times 14 \qquad [\text{Tbit}]$$

The above period of time will exceed the value of T_{FRAME_MAX} ({(10 × data bytes + 44) + 1} × 1.4) shown in LIN Specification Package Revision 1.3.

24.10.2 LIN Error Detection Targets

Figure 24.31 shows the area which the LIN module monitors for error detection.

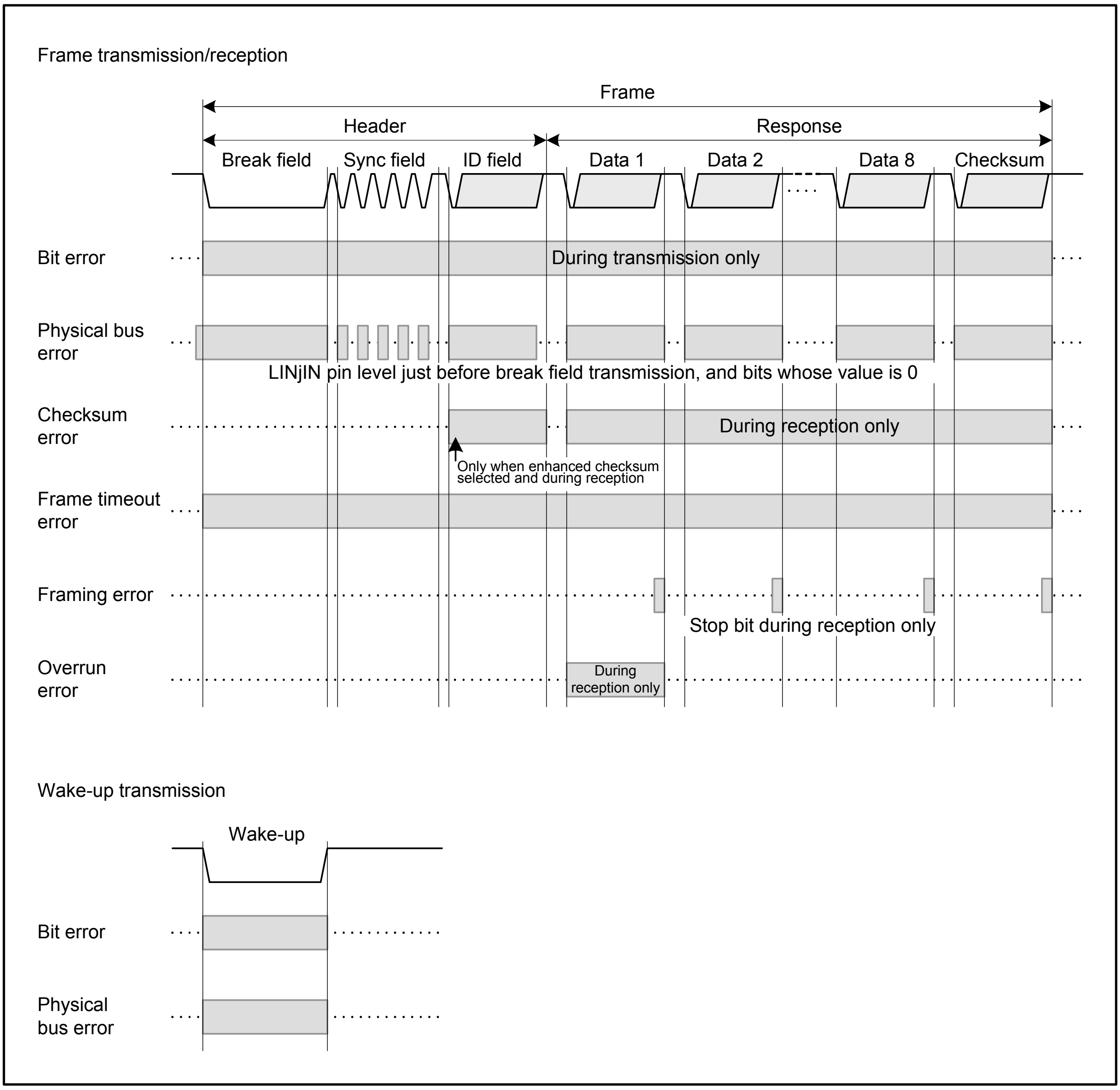


Figure 24.31 LIN Error Detection Targets (j = 0, 1)

24.11 LIN Interrupts

The LIN module generates the LINj interrupt (j = 0, 1) and LIN low detection interrupt.

Each channel has four interrupt sources: frame/wake-up transmit completion, frame/wake-up receive completion, error detection, and input signal low detection.

Interrupt requests by the frame/wake-up transmit completion, frame/wake-up receive completion, and error detection interrupt sources are aggregated in each channel by logical OR as an interrupt request “LINj interrupt”. The input signal low detection interrupt of each channel, on the other hand, is aggregated by logical OR as a LIN low detection interrupt request.

The respective interrupt request is output when the corresponding flag in the LST register becomes 1 while the corresponding bit in the LMD0 register is set to 1 (interrupt enabled). No new interrupt request is generated by the other sources if any of them is 1 since multiple interrupt sources are aggregated.

Figure 24.32 and Figure 24.33 show the block diagram of the LINj interrupt and the LIN low detection interrupt, respectively.

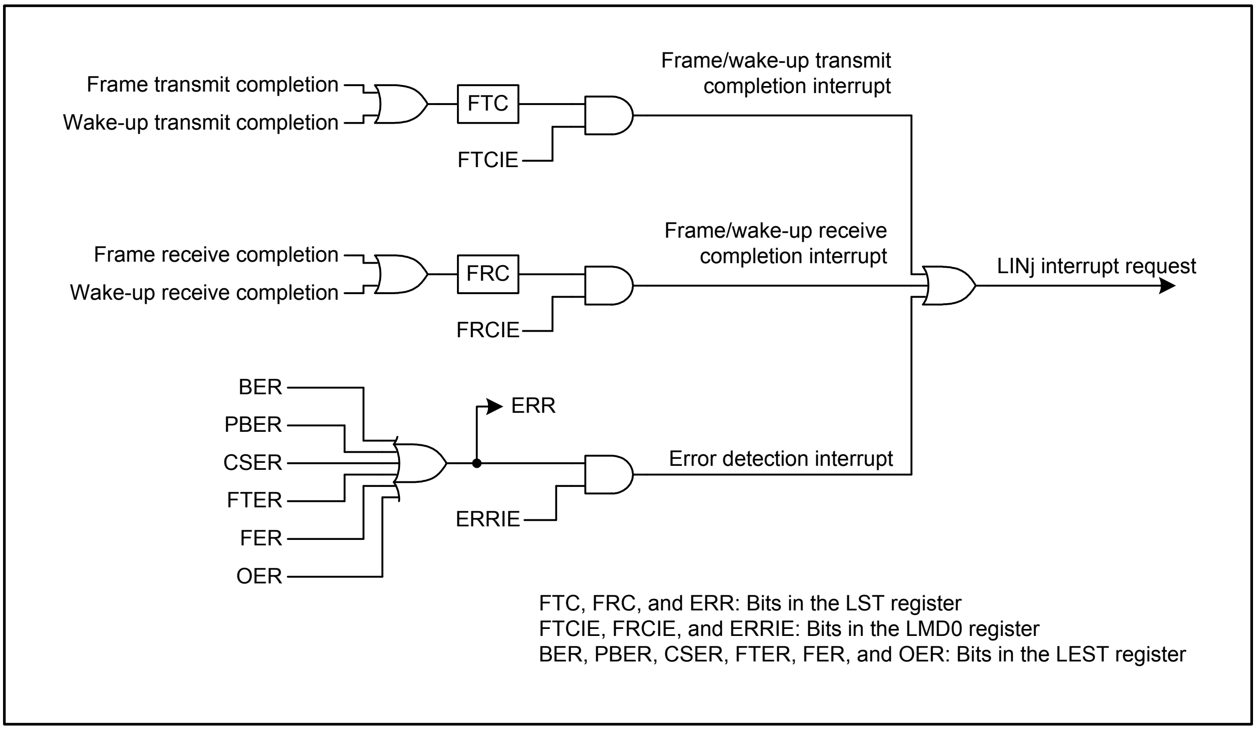


Figure 24.32 LINj Interrupt Block Diagram (j = 0, 1)

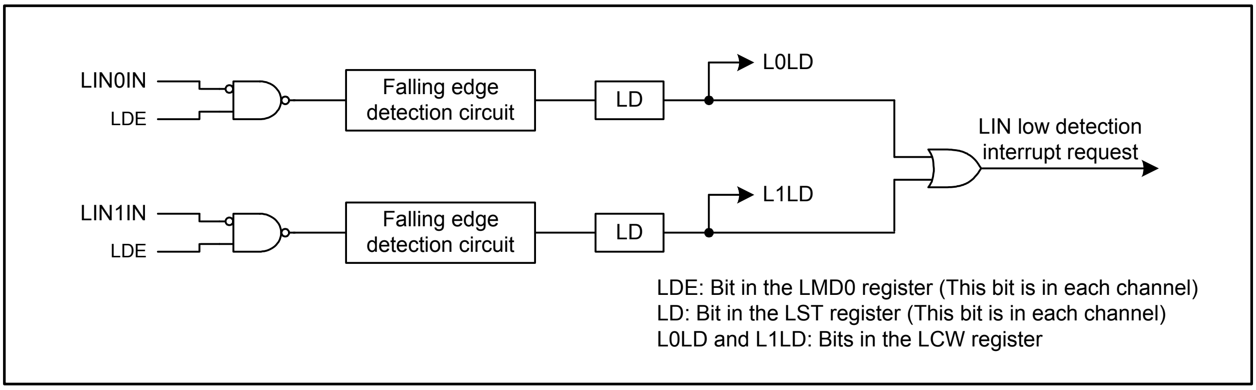


Figure 24.33 LIN Low Detection Interrupt Block Diagram

25. CAN Module

The R32C/142 Group implements three channels (CAN2, CAN3, and CAN5) and the R32C/145 Group implements six channels (CAN0 to CAN5) of the Controller Area Network (CAN) module that complies with the ISO 11898-1 standard. 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).

This chapter covers the specifications for all six channels. Channels CAN0, CAN1, and CAN4 are not available in the R32C/142 Group.

Table 25.1 and Table 25.2 list the CAN module specifications, and Figure 25.1 shows the CAN module block diagram.

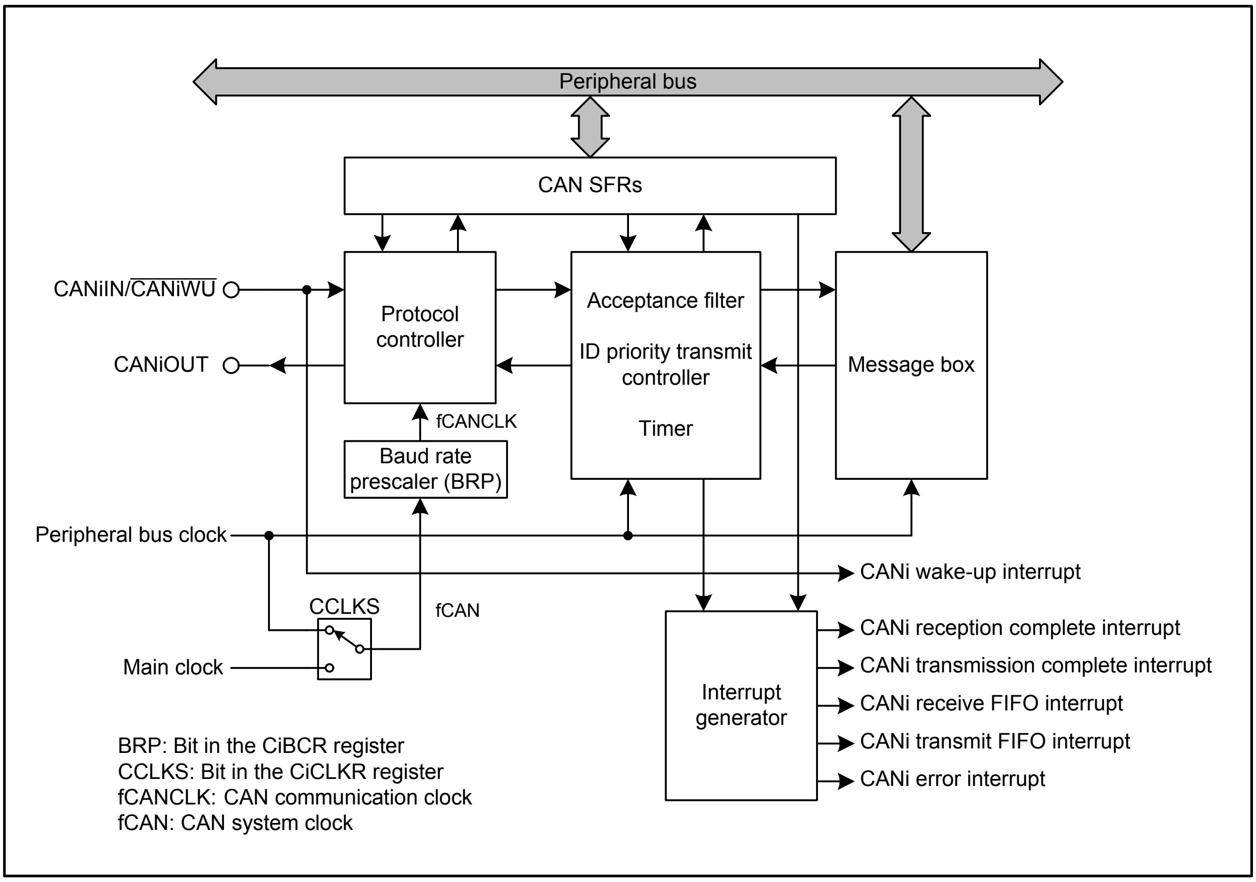
Connect the CAN bus transceiver externally.

Table 25.1 CAN Module Specifications (1/2)

Item	Specification
Protocol	ISO 11898-1 compliant
Bit rate	Maximum 1 Mbps
Message boxes	16 mailboxes: Two selectable mailbox modes: • Normal mailbox mode All 16 mailboxes can be indivisually configured for transmission or reception • FIFO mailbox mode: 8 mailboxes can be indivisually configured for transmission or reception. 4 of the remaining mailboxes can be configured for transmit FIFO and the other 4 mailboxes for receive FIFO
Reception	• Data frames and remote frames can be received • Selectable receiving ID format (standard ID only, extended ID only, or both IDs) • 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 • Received frames in specified mailboxes can be automatically transferred to the CAN gateway module when the gateway function is enabled
Acceptance filtering	4 acceptance masks: 1 mask for every 4 mailboxes The mask can be enabled or disabled for each mailbox
Transmission	• Data frames and remote frames can be transmitted • Selectable transmitting ID format (standard ID only, extended ID only, or both IDs) • Programmable one-shot transmission function • Selectable ID priority transmit mode or mailbox number priority transmit mode • Transmission request can be aborted (A completed abort operation 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 recovering from the bus-off state can be selected: • ISO 11898-1 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 to the error-active state by a program

Table 25.2 CAN Module Specifications (2/2)

Item	Specification
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 among 1, 2, 4, and 8 bit times
Interrupt sources	6 types: • Reception complete • Transmission complete • Receive FIFO • Transmit FIFO • Error • Wake-up
CAN sleep mode	Current consumption can be reduced by stopping the CAN clock
Software support units	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	Peripheral bus clock or main clock selectable
Test modes	3 test modes available for user evaluation: • Listen only mode • Self test mode 0 (external loop back) • Self test mode 1 (internal loop back)



BRP: Bit in the CiBCR register
CCLKS: Bit in the CiCLKR register
fCANCLK: CAN communication clock
fCAN: CAN system clock

Figure 25.1 CAN Module Block Diagram (i = 0 to 5)

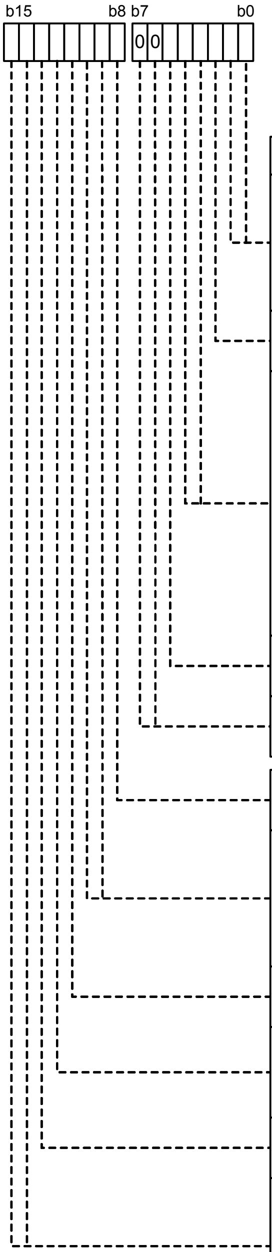
- CANiIN/CANiOUT (i = 0 to 5): CAN I/O pins
- Protocol controller: Handles CAN protocol processing such as bus arbitration, bit timing at transmission and reception, stuffing, error handling, etc.
- Message box: Consists of 16 mailboxes which can be individually configured as either a transmit or receive mailbox. Each mailbox has its own ID, data length code, an 8-byte data field, and a time stamp. In routing mode of the gateway module (the OM bit in the GMR register is set to 1), each channel has exclusive mailboxes for the gateway reception. Received frames are automatically transferred to the CAN gateway module.
- Acceptance filter: Filters received messages. Registers CiMKR0 to CiMKR3 are used for the filtering process.
- Timer: Used for the time stamp function. The timer value when storing a message into a mailbox is written as the time stamp value.
- Wake-up function: Generates a CANi wake-up interrupt request when a message is detected on the CAN bus.
- 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
- CAN SFRs: CAN-associated registers. Refer to 25.1 “CAN SFRs” for details.

25.1 CAN SFRs

CAN-associated registers are shown in Figures 25.2 to 25.11, 25.13, 25.14, 25.16 to 25.20, 25.22, and 25.24 to 25.30.

25.1.1 CANi Control Register (CiCTLR) (i = 0 to 5)

CANi Control Register (i = 0 to 5)



Symbol	Address	Reset Value
C0CTLR, C1CTLR	47F41h-47F40h, 47D41h-47D40h	0000 0000 0000 0101b
C2CTLR, C3CTLR	47B41h-47B40h, 47941h-47940h	0000 0000 0000 0101b
C4CTLR, C5CTLR	47741h-47740h, 47541h-47540h	0000 0000 0000 0101b

Bit Symbol	Bit Name	Function	RW
CANM	CAN Operating Mode Select Bit (1)	b1 b0 0 0 : CAN operation mode 0 1 : CAN reset mode 1 0 : CAN halt mode 1 1 : Do not use this combination	RW
SLPM	CAN Sleep Mode Bit (1, 2)	0: Mode other than CAN sleep mode 1: CAN sleep mode	RW
BOM	Bus-off Recovery Mode Select Bit (3)	b4 b3 0 0 : Normal mode (ISO 11898-1 compliant) 0 1: Entry to CAN halt mode automatically at bus-off entry 1 0 : Entry to CAN halt mode automatically at bus-off end 1 1 : Entry to CAN halt mode (during bus-off recovery period) by a program request	RW
RBOC	Forced Recovery From Bus-off Bit (4)	0: Nothing occurred 1: Forced recovery from bus-off (5)	RW
— (b7-b6)	Reserved	Should be written with 0	RW
MBM	CAN Mailbox Mode Select Bit (3)	0: Normal mailbox mode 1: FIFO mailbox mode	RW
IDFM	ID Format Mode Select Bit (3)	b10b9 0 0 : Standard ID mode 0 1 : Extended ID mode 1 0 : Mixed ID mode 1 1 : Do not use this combination	RW
MLM	Message Lost Mode Select Bit (3, 7)	0: Overwrite mode 1: Overrun mode	RW
TPM	Transmit Priority Mode Select Bit (3)	0: ID priority transmit mode 1: Mailbox number priority transmit mode	RW
TSRC	Time Stamp Counter Reset Bit (6)	0: Not reset 1: Reset (5)	RW
TSPS	Time Stamp Prescaler Select Bit (3)	b15b14 0 0 : Every bit time 0 1 : Every 2-bit time 1 0 : Every 4-bit time 1 1 : Every 8-bit time	RW

Notes:

1. When bits CANM and SLPM are changed, read the CiSTR register to ensure that the mode has been switched. Do not change bits CANM and SLPM until the mode has been switched.
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.
3. Write to bits BOM, MBM, IDFM, MLM, TPM, and TSPS in CAN reset mode.
4. Set the RBOC bit to 1 in bus-off state.
5. Bits RBOC and TSRC are automatically set back to 0 after being set to 1. They are read as 0.
6. Set the TSRC bit to 1 in CAN operation mode.
7. Set the MLM bit to 0 (overwrite mode) when enabling the gateway function.

Figure 25.2 Registers C0CTLR to C5CTLR

25.1.1.1 CANM Bit

Set the CANM bit to select one of the following modes for the CAN module: CAN operation mode, CAN reset mode, or CAN halt mode. Refer to 25.2 “Operating Modes” for details.
Set the SLPM bit to 1 to select CAN sleep mode.
Do not set the CANM bit to 11b.
When the CAN module enters CAN halt mode according to the setting of the BOM bit, the CANM bit automatically becomes 10b.

25.1.1.2 SLPM Bit

When the SLPM bit is set to 1, the CAN module enters CAN sleep mode.
When this bit is set to 0, the CAN module exits CAN sleep mode.
Refer to 25.2 “Operating Modes” for details.

25.1.1.3 BOM Bit

Set the BOM bit to select bus-off recovery mode.

When the BOM bit is 00b, the recovery from bus-off is ISO 11898-1 compliant, i.e. the CAN module reenters 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 01b, as soon as the CAN module enters the bus-off state, the CANM bit in the CiCTLR register (i = 0 to 5) becomes 10b (CAN halt mode) and the CAN module enters CAN halt mode. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR become 00h.

When the BOM bit is 10b, the CANM bit becomes 10b as soon as the CAN module enters the bus-off state. The CAN module enters CAN halt mode after recovering 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 become 00h.

When the BOM bit is 11b, the CAN module enters CAN halt mode by setting the CANM bit to 10b while the CAN module is still in the bus-off state. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR become 00h. However, if the CAN module recovers from bus-off after detecting 11 consecutive recessive bits 128 times before the CANM bit is set to10b, 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 01b, or at bus-off end when the BOM bit is 10b), then the CPU request to enter CAN reset mode has higher priority.

25.1.1.4 RBOC Bit

When the RBOC bit is set to 1 (forced recovery from bus-off) in the bus-off state, the CAN module forcibly recovers from the bus-off state. This bit automatically becomes 0. The error state changes from bus-off to error-active.

When the RBOC bit is set to 1, registers CiRECR and CiTECR become 00h and the BOST bit in the CiSTR register becomes 0 (CAN module is not in bus-off state). The other registers do not change. No bus-off recovery interrupt request is generated by recovering from the bus-off state.
Use the RBOC bit only when the BOM bit is 00b (normal mode).

25.1.1.5 MBM Bit

When the MBM bit is 0 (normal mailbox mode), mailboxes [0] to [15] are configured as transmit or receive mailboxes.

When this bit is 1 (FIFO mailbox mode), mailboxes [0] to [7] are configured as transmit or receive mailboxes, mailboxes [8] to [11] are configured as transmit FIFO, and mailboxes [12] to [15] are as receive FIFO.

Transmit data is written into mailbox [8] (mailbox [8] is a window mailbox for the transmit FIFO).

Receive data is read from mailbox [12] (mailbox [12] is a window mailbox for the receive FIFO).

Table 25.3 and Table 25.4 list the mailbox configurations when the gateway function is disabled and when it is enabled, respectively.

Table 25.3 Mailbox Configuration (1/2): When Gateway Function is Disabled

Mailbox	MBM Bit is 0 (Normal mailbox mode)	MBM Bit is 1 (1) (FIFO mailbox mode)
Mailboxes [0] to [7]	Normal mailbox	Normal mailbox
Mailboxes [8] to [11]		Transmit FIFO
Mailboxes [12] to [15]		Receive FIFO

Note:

1. When the MBM bit is set to 1, note the following:
    - Transmit FIFO is controlled by the CiTFCR register (i = 0 to 5).
      The CiMCTLj register (j = 0 to 15) for mailboxes [8] to [11] is disabled.
      Registers CiMCTL8 to CiMCTL11 cannot be used.
    - Receive FIFO is controlled by the CiRFCR register.
      The CiMCTLj register for mailboxes [12] to [15] is disabled.
      Registers CiMCTL12 to CiMCTL15 cannot be used.
    - Refer to the CiMIER register for the FIFO interrupts.
    - The corresponding bits in the CiMKIVLR register for mailboxes [8] to [15] are disabled. Set 0 to these bits.
    - Transmit/receive FIFOs can be used for both data frames and remote frames.

Table 25.4 Mailbox Configuration (2/2): When Gateway Function is Enabled (1)

Mailbox	MBM bit is 0 (Normal mailbox mode)	MBM bit is 1 (FIFO mailbox mode)
Mailboxes [0] to [7]	Normal mailbox	Normal mailbox
Mailboxes [8] to [11]		Transmit FIFO
Mailboxes [12] to [15]	Gateway receive mailbox (2)	Gateway receive FIFO (3)

Notes:

1. The gateway function can be enabled both in normal mailbox mode and FIFO mailbox mode. When it is enabled, mailboxes [12] to [15] are exclusively used as the gateway receive mailboxes in both modes.
2. When the MBM bit is set to 0 and the gateway function is enabled, set as follows:
   - Set mailboxes [12] to [15] for reception.
   - Enable interrupts for each mailbox. However, the reception complete interrupt is not generated. This setting is necessary to notify the CAN gateway module that the reception is completed.
   - Set bits IDE and RTR of each mailbox to values from 00b to 11b when receiving all four types of frames.
   - Set all the corresponding CANi mask registers (CiMKR3) and the CANi mask invalid registers (CiMKIVLR) to 0 (mask valid).
   - Set the message lost mode to overwrite mode (the MLM bit is 0).
3. When the MBM bit is set to 1 and the gateway function is enabled, set as follows:
   - Enable the receive FIFO interrupt and set it to be generated every time reception is completed. However, the receive FIFO interrupt is not generated. This setting is necessary to notify the CAN gateway module that the reception is completed.
   - Set bits EID and SID of the corresponding CANi mask register (CiMKR3) to 0. Also, set types of a receive frame to bits IDE and RTR in CANi FIFO received compare register 0 (CiFIDCR0) and CANi FIFO received compare register 1 (CiFIDCR1).
   - Set the message lost mode to overwrite mode (the MLM bit is 0).

25.1.1.6 IDFM Bit

Set the IDFM bit to specify the ID format.
When this bit is 00b, all mailboxes (including FIFO mailboxes) handle standard IDs only.
When this bit is 01b, all mailboxes (including FIFO mailboxes) handle extended IDs only.
When this bit is 10b, all mailboxes (including FIFO mailboxes) handle both standard IDs and extended IDs. Standard IDs or extended IDs are specified by setting the IDE bit in the corresponding mailbox in normal mailbox mode. In FIFO mailbox mode, the IDE bit in the corresponding mailbox is used for mailboxes [0] to [7], the IDE bit in registers CiFIDCR0 and CiFIDCR1 is used for the receive FIFO, and the IDE bit in mailbox [8] is used for the transmit FIFO.
Do not set 11b to the IDFM bit.

25.1.1.7 MLM Bit

Set the MLM bit to specify the operation when a new message is captured in an 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 overwrites the old message.
When this bit is 1, all mailboxes are set to overrun mode and the new message is discarded.

25.1.1.8 TPM Bit

Set the TPM bit to specify the priority of modes when transmitting messages.
ID priority transmit mode or mailbox number transmit mode can be selected.
All mailboxes are set to 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 specified in the ISO 11898-1 standard. In ID priority transmit mode, mailboxes [0] to [15] (in normal mailbox mode), mailboxes [0] to [7] (in FIFO mailbox mode), and the transmit FIFO are compared with 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 [7]).

25.1.1.9 TSRC Bit

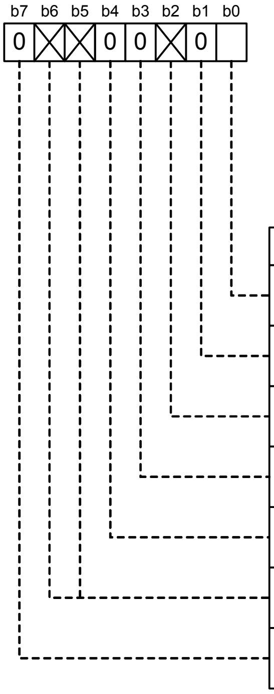
Set the TSRC bit to reset the time stamp counter.
When this bit is set to 1, the CiTSR register (i = 0 to 5) becomes 0000h. It automatically becomes 0.

25.1.1.10 TSPS Bit

Set the TSPS bit to select the prescaler for the time stamp.
The reference clock for the time stamp can be selected among 1, 2, 4, and 8 bit times.

25.1.2 CANi Clock Select Register (CiCLKR) (i = 0 to 5)

CANi Clock Select Register (i = 0 to 5)



Symbol	Address	Reset Value
C0CLKR	47F47h	000X 0000b
C1CLKR	47D47h	000X 0000b
C2CLKR	47B47h	000X 0000b
C3CLKR	47947h	000X 0000b
C4CLKR	47747h	000X 0000b
C5CLKR	47547h	000X 0000b

Bit Symbol	Bit Name	Function	RW
CCLKS	CAN Clock Source Select Bit [1]	0: Peripheral bus clock 1: Main clock [2]	RW
— (b1)	Reserved	Should be written with 0	RW
— (b2)	No register bit; should be written with 0 and read as 0		—
— (b3)	Reserved	Should be written with 0	RW
— (b4)	Reserved	Should be written with 0 and read as undefined value	RW
— (b6-b5)	No register bits; should be written with 0 and read as 0		—
— (b7)	Reserved	Should be written with 0	RW

Notes:

1. Write to the CCLKS bit in CAN reset mode.
2. To set the CCLKS bit to 1, the frequency of the peripheral bus clock should be equal to or higher than the frequency of the main clock.

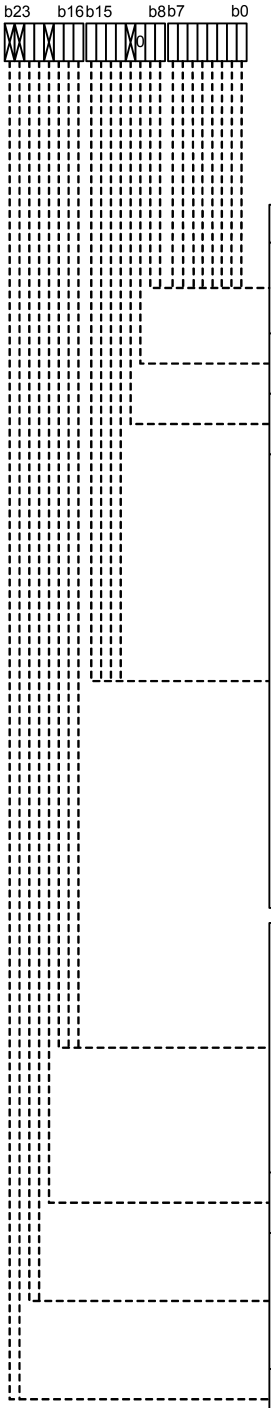
Figure 25.3 Registers C0CLKR to C5CLKR

25.1.2.1 CCLKS Bit

When the CCLKS bit is set to 0, the CAN clock source (fCAN) originates from the PLL frequency synthesizer.
When this bit is set to 1, fCAN originates directly from the external XIN pin bypassing the PLL.

25.1.3 CANi Bit Configuration Register (CiBCR) (i = 0 to 5)

CANi Bit Configuration Register (i = 0 to 5) (1, 2)



Symbol	Address	Reset Value
C0BCR	47F46h-47F44h	00 0000h
C1BCR	47D46h-47D44h	00 0000h
C2BCR	47B46h-47B44h	00 0000h
C3BCR	47946h-47944h	00 0000h
C4BCR	47746h-47744h	00 0000h
C5BCR	47546h-47544h	00 0000h

Bit Symbol	Bit Name	Function	RW
BRP	Prescaler Division Ratio Set Bit (10 bits)	If the setting value is P (0 to 1023), the baud rate prescaler divides fCAN by P + 1	RW
— (b10)	Reserved	Should be written with 0	RW
— (b11)	No register bit; should be written with 0 and read as 0		—
TSEG1	Time Segment 1 Control Bit	b15b14b13b12 0 0 0 0 : Do not use this combination 0 0 0 1 : Do not use this combination 0 0 1 0 : Do not use this combination 0 0 1 1 : 4 Tq 0 1 0 0 : 5 Tq 0 1 0 1 : 6 Tq 0 1 1 0 : 7 Tq 0 1 1 1 : 8 Tq 1 0 0 0 : 9 Tq 1 0 0 1 : 10 Tq 1 0 1 0 : 11 Tq 1 0 1 1 : 12 Tq 1 1 0 0 : 13 Tq 1 1 0 1 : 14 Tq 1 1 1 0 : 15 Tq 1 1 1 1 : 16 Tq	RW
TSEG2	Time Segment 2 Control Bit	b18b17b16 0 0 0 : Do not use this combination 0 0 1 : 2 Tq 0 1 0 : 3 Tq 0 1 1 : 4 Tq 1 0 0 : 5 Tq 1 0 1 : 6 Tq 1 1 0 : 7 Tq 1 1 1 : 8 Tq	RW
— (b19)	No register bit; should be written with 0 and read as 0		—
SJW	Resynchronization Jump Width Control Bit	b21b20 0 0 : 1 Tq 0 1 : 2 Tq 1 0 : 3 Tq 1 1 : 4 Tq	RW
— (b23-b22)	No register bits; should be written with 0 and read as 0		—

Notes:

1. Set this register before entering CAN halt mode from CAN reset mode or CAN operation mode from CAN reset mode. Once this register is set, it can be rewritten in CAN reset mode or CAN halt mode.
2. This register consists of 24 bits. A 32-bit read/write access should be performed carefully as to not rewrite the CiCLKR register.

Figure 25.4 Registers C0BCR to C5BCR

Refer to 25.3 “CAN Communication Speed Configuration” for the bit timing configuration.

25.1.3.1 BRP Bit

The BRP bit sets the frequency of the CAN communication clock (fCANCLK).
One fCANCLK cycle is measured as Time Quantum (Tq).

25.1.3.2 TSEG1 Bit

Set the TSEG1 bit 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 Tq can be set.

25.1.3.3 TSEG2 Bit

Set the TSEG2 bit to specify the length of phase buffer segment TSEG2 (PHASE_SEG2) with the value of Tq.
A value from 2 to 8 Tq can be set.
Set the value smaller than that of the TSEG1 bit.

25.1.3.4 SJW Bit

Set the SJW bit to specify the resynchronization jump width with the value of Tq.
A value from 1 to 4 Tq can be set.
Set the value smaller than or equal to that of the TSEG2 bit.

25.1.4 CANi Mask Register k (CiMKRk) (i = 0 to 5; k = 0 to 3)

CANi Mask Register k (i = 0 to 5; k = 0 to 3) (1, 2)

Symbol	Address	Reset Value
C0MKR0, C0MKR1	47F13h-47F10h, 47F17h-47F14h	Undefined
C0MKR2, C0MKR3	47F1Bh-47F18h, 47F1Fh-47F1Ch	Undefined
C1MKR0, C1MKR1	47D13h-47D10h, 47D17h-47D14h	Undefined
C1MKR2, C1MKR3	47D1Bh-47D18h, 47D1Fh-47D1Ch	Undefined
C2MKR0, C2MKR1	47B13h-47B10h, 47B17h-047B14h	Undefined
C2MKR2, C2MKR3	47B1Bh-47B18h, 47B1Fh-047B1Ch	Undefined
C3MKR0, C3MKR1	47913h-47910h, 47917h-047914h	Undefined
C3MKR2, C3MKR3	4791Bh-47918h, 4791Fh-04791Ch	Undefined
C4MKR0, C4MKR1	47713h-47710h, 47717h-47714h	Undefined
C4MKR2, C4MKR3	4771Bh-47718h, 4771Fh-4771Ch	Undefined
C5MKR0, C5MKR1	47513h-47510h, 47517h-47514h	Undefined
C5MKR2, C5MKR3	4751Bh-47518h, 4751Fh-4751Ch	Undefined

Bit Symbol	Bit Name	Function	RW
EID	Extended ID Bit	0: Corresponding EID bit is not compared 1: Corresponding EID bit is compared	RW
SID	Standard ID Bit	0: Corresponding SID bit is not compared 1: Corresponding SID bit is compared	RW
— (b31-b29)	Reserved	Should be written with 0	RW

Notes:
1. Write to registers CiMKR0 to CiMKR3 in CAN reset mode or CAN halt mode.
2. Set as follows when the gateway function is enabled (the OM bit in the GMR register is 1):
   - Set the CiMKR3 register to 0000 0000h in normal mailbox mode (the MBM bit in the CiCTLR register is 0).
   - Set registers CiMKR2 and CiMKR3 register to 0000 0000h in FIFO mailbox mode (the MBM bit is 1).

Figure 25.5 Registers C0MKR0 to C5MKR3

Refer to 25.5 “Acceptance Filtering and Masking Function” for the masking function in FIFO mailbox mode.

25.1.4.1 EID Bit

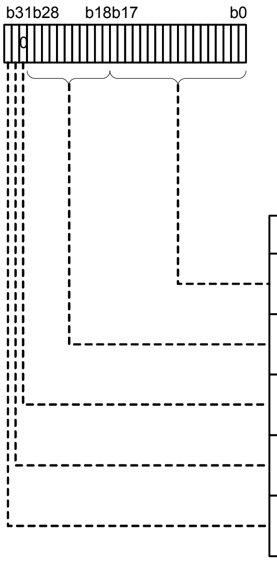
The EID bit is the filter mask bit corresponding to the CAN extended ID bit. This bit is used to receive extended ID messages.
When the EID bit is 0, the corresponding EID bit is not compared for the received ID and the mailbox ID.
When this bit is 1, the corresponding EID bit is compared for the received ID and the mailbox ID.

25.1.4.2 SID Bit

The SID bit is the filter mask bit corresponding to the CAN standard ID bit. This bit is used to receive both standard ID and extended ID messages.
When the SID bit is 0, the corresponding SID bit is not compared for the received ID and the mailbox ID.
When this bit is 1, the corresponding SID bit is compared for the received ID and the mailbox ID.

25.1.5 CANi FIFO Received ID Compare Register n (CiFIDCR0 and CiFIDCR1) (i = 0 to 5; n = 0, 1)

CANi FIFO Received ID Compare Register n (i = 0 to 5; n = 0, 1) (1, 3)



Symbol	Address	Reset Value
C0FIDCR0, C0FIDCR1	47F23h-47F20h, 47F27h-47F24h	Undefined
C1FIDCR0, C1FIDCR1	47D23h-47D20h, 47D27h-47D24h	Undefined
C2FIDCR0, C2FIDCR1	47B23h-47B20h, 47B27h-47B24h	Undefined
C3FIDCR0, C3FIDCR1	47923h-47920h, 47927h-47924h	Undefined
C4FIDCR0, C4FIDCR1	47723h-47720h, 47727h-47724h	Undefined
C5FIDCR0, C5FIDCR1	47523h-47520h, 47527h-47524h	Undefined

Bit Symbol	Bit Name	Function	RW
EID	Extended ID Bit	0: Corresponding EID bit is 0 1: Corresponding EID bit is 1	RW
SID	Standard ID Bit	0: Corresponding SID bit is 0 1: Corresponding SID bit is 1	RW
— (b29)	Reserved	Should be written with 0	RW
RTR	Remote Frame Request Bit	0: Data frame 1: Remote frame	RW
IDE	ID Extension Bit (2)	0: Standard ID 1: Extended ID	RW

Notes:

1. Write to registers CiFIDCR0 and CiFIDCR1 in CAN reset mode or CAN halt mode.
2. The IDE bit is enabled when the IDFM bit in the CiCTLR register is 10b (mixed ID mode). When the IDFM bit is either 00b (standard ID mode) or 01b (extended ID mode), the IDE bit should be written with 0.
3. Set as follows when the gateway function is enabled (the OM bit in the GMR register is 1):
   - Set 0 to bits EID17 to EID0 and bits SID10 to SID0 in registers CiFIDCR0 and CiFIDCR1.
   - The ID format and frame format of a receive frame can be selected by setting bits IDE and RTR in each register.

Figure 25.6 Registers C0FIDCR0 to C5FIDCR1

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 CiMB12 to CiMB15 are disabled.
Refer to 25.5 “Acceptance Filtering and Masking Function” for details on using these registers.

25.1.5.1 EID Bit

The EID bit sets the extended ID of data frames and remote frames. This bit is used to receive extended ID messages.

25.1.5.2 SID Bit

The SID bit sets the standard ID of data frames and remote frames. This bit is used to receive both standard ID and extended ID messages.

25.1.5.3 RTR Bit

The RTR bit sets the specified frame format of data frames or remote frames.

This bit specifies the following operations:

- When both RTR bits in registers CiFIDCR0 and CiFIDCR1 (i = 0 to 5) 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 with different values, both data frames and remote frames can be received.

25.1.5.4 IDE Bit

The IDE bit sets the ID format of standard ID or extended ID.

This bit is enabled when the IDFM bit in the CiCTLR register is 10b (mixed ID mode).

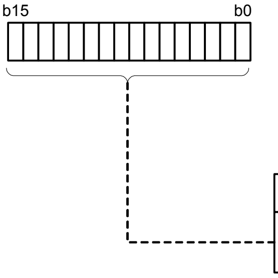
When the IDFM bit is 10b, the IDE bit specifies the following operations:

- 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 with different values, both standard ID and extended ID frames can be received.

25.1.6 CANi Mask Invalid Register (CiMKIVLR) (i = 0 to 5)

CANi Mask Invalid Register (i = 0 to 5) (1)

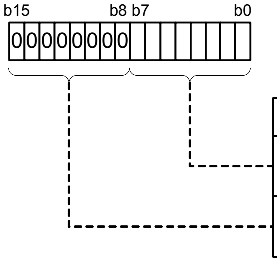
Normal mailbox mode



Symbol	Address	Reset Value
C0MKIVLR	47F2Bh-47F2Ah	Undefined
C1MKIVLR	47D2Bh-47D2Ah	Undefined
C2MKIVLR	47B2Bh-47B2Ah	Undefined
C3MKIVLR	4792Bh-4792Ah	Undefined
C4MKIVLR	4772Bh-4772Ah	Undefined
C5MKIVLR	4752Bh-4752Ah	Undefined

Bit Symbol	Bit Name	Function	RW
— (b15-b0)	Mask Invalid Bit	0: Mask valid 1: Mask invalid	RW

FIFO mailbox mode



Bit Symbol	Bit Name	Function	RW
— (b7-b0)	Mask Invalid Bit	0: Mask valid 1: Mask invalid	RW
— (b15-b8)	Reserved	Should be written with 0	RW

Note:

1. Write to the CiMKIVLR register in CAN reset mode or CAN halt mode.

Figure 25.7 Registers C0MKIVLR to C5MKIVLR

Each bit corresponds to the mailbox with the same number. When each bit is 1, the acceptance mask for the mailbox corresponding to the bit number is disabled. In this case, a received message is stored in the mailbox only if its ID matches bits SID and EID in the CiMBj register (j = 0 to 15).

25.1.7 CANi Mailbox (CiMBj) (i = 0 to 5; j = 0 to 15)

Table 25.5 lists the CANi mailbox memory mapping, and Table 25.6 lists the CAN data frame structure. The reset value of the CANi mailbox is undefined.

Table 25.5 CANi Mailbox Memory Mapping (i = 0 to 5)

Address	Message Content
CAN0 to CAN5	Memory mapping
base + j × 16 + 0	EID7 to EID0
base + j × 16 + 1	EID15 to EID8
base + j × 16 + 2	SID5 to SID0, EID17, EID16
base + j × 16 + 3	IDE, RTR, SID10 to SID6
base + j × 16 + 4	—
base + j × 16 + 5	Data length code (DLC)
base + j × 16 + 6	Data byte 0
base + j × 16 + 7 : : : base + j × 16 + 13	Data byte 1 : : : Data byte 7
base + j × 16 + 14	Time stamp lower byte
base + j × 16 + 15	Time stamp upper byte

base: CAN0 is 47E00h, CAN1 is 47C00h, CAN2 is 47A00h, CAN3 is 47800h, CAN4 is 47600h, and CAN5 is 47400h

j: Mailbox number (j = 0 to 15)

Table 25.6 CAN Data Frame Structure

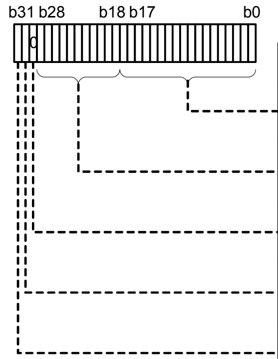
SID10 to SID6	SID5 to SID0	EID17 to EID16	EID15 to EID8	EID7 to EID0	DLC3 to DLC0	DATA0	DATA1	....	DATA7

Note:

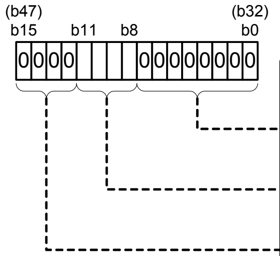
1. Refer to Table 25.4 for the setting in routing mode of the gateway module.

CANi Mailbox Register j (i = 0 to 5; j = 0 to 15) (1)

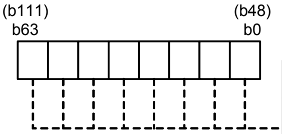
Symbol	Address (2)	Reset Value
C0MB0 to C0MB15	47E00h to 47EFFh	Undefined
C1MB0 to C1MB15	47C00h to 47CFFh	Undefined
C2MB0 to C2MB15	47A00h to 47AFFh	Undefined
C3MB0 to C3MB15	47800h to 478FFh	Undefined
C4MB0 to C4MB15	47600h to 476FFh	Undefined
C5MB0 to C5MB15	47400h to 474FFh	Undefined



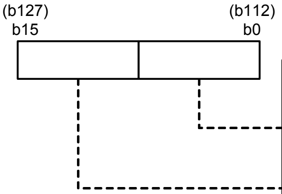
Bit Symbol	Bit Name	Function	RW
EID	Extended ID (3)	0: Corresponding EID bit is 0 1: Corresponding EID bit is 1	RW
SID	Standard ID	0: Corresponding SID bit is 0 1: Corresponding SID bit is 1	RW
— (b29)	Reserved	Should be written with 0	RW
RTR	Remote Frame Request Bit	0: Data frame 1: Remote frame	RW
IDE	ID Extension Bit (4)	0: Standard ID 1: Extended ID	RW



Bit Symbol	Bit Name	Setting Range	RW
— (b7-b0)	Reserved	Should be written with 0	RW
DLC	Data Length Code (5)	0h to Fh	RW
— (b15-b12)	Reserved	Should be written with 0	RW



Symbol	Name	Setting Range	RW
DATA0 to DATA7	Data Bytes 0 to 7 (5, 6)	00h to FFh	RW



Symbol	Name	Setting Range	RW
TSL	Time Stamp Lower Byte	00h to FFh	RW
TSH	Time Stamp Upper Byte	00h to FFh	RW

Notes:

1. Write to the CiMBj register only when the associated CiMCTLj register is 00h and the corresponding mailbox is not processing an abort request.
2. Refer to the memory mapping table for CANi mailbox on the previous page for detailed addresses.
3. If the mailbox has received a standard ID message, the EID bit in the mailbox is undefined.
4. The IDE bit is enabled when the IDFM bit in the CiCTLR register is 10b (mixed ID mode). When the IDFM bit is either 00b (standard ID mod) or 01b (extended ID mode), it should be written with 0.
5. 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.
6. If the mailbox has received a remote frame, the previous values of DATA0 to DATA7 in the mailbox are retained.

Figure 25.8 Registers C0MBj to C5MBj

The previous value of each mailbox is retained unless a new message is received.

25.1.7.1 EID Bit

The EID bit sets the extended ID of data frames and remote frames. This bit is used to transmit or receive extended ID messages.

25.1.7.2 SID Bit

The SID bit sets the standard ID of data frames and remote frames. This bit is used to transmit or receive both standard ID and extended ID messages.

25.1.7.3 RTR Bit

The RTR bit sets the frame format of data frames or remote frames.
This bit specifies the following operations:

- The receive mailbox receives only frames with the format specified by the RTR bit.
- The transmit mailbox transmits according to the frame format specified by the RTR bit.
- The receive FIFO mailbox receives the data frame, remote frame, or both frames specified by the RTR bit in registers CiFIDCR0 and CiFIDCR1 (i = 0 to 5).
- The transmit FIFO mailbox transmits the data frame or remote frame specified by the RTR bit in the relevant transmitting message.

25.1.7.4 IDE Bit

The IDE bit sets the ID format of standard IDs or extended IDs.
This bit is enabled when the IDFM bit in the CiCTLR register is 10b (mixed ID mode).
When the IDFM bit is 10b, the IDE bit specifies the following operations:

- The receive mailbox receives only the ID format specified by the IDE bit.
- The transmit mailbox transmits according to the ID format specified by the IDE bit.
- The receive FIFO mailbox receives messages with the standard ID, extended ID, or both IDs specified by the IDE bit in registers CiFIDCR0 and CiFIDCR1.
- The transmit FIFO mailbox transmits messages with the standard ID or extended ID specified by the IDE bit in the relevant transmitting message.

25.1.7.5 DLC

The DLC sets 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.

Table 25.7 lists the data length corresponding to the DLC.

Table 25.7 Data Length Corresponding to the DLC

DLC[3]	DLC[2]	DLC[1]	DLC[0]	Data Length
0	0	0	0	0 bytes
0	0	0	1	1 byte
0	0	1	0	2 bytes
0	0	1	1	3 bytes
0	1	0	0	4 bytes
0	1	0	1	5 bytes
0	1	1	0	6 bytes
0	1	1	1	7 bytes
1	X	X	X	8 bytes

X: Any value

25.1.7.6 DATA0 to DATA7

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.

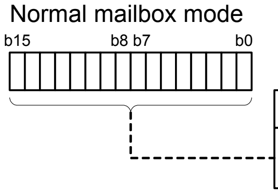
25.1.7.7 TSL and TSH

TSL and TSH store the counter value of the time stamp when received messages are stored in the mailbox.

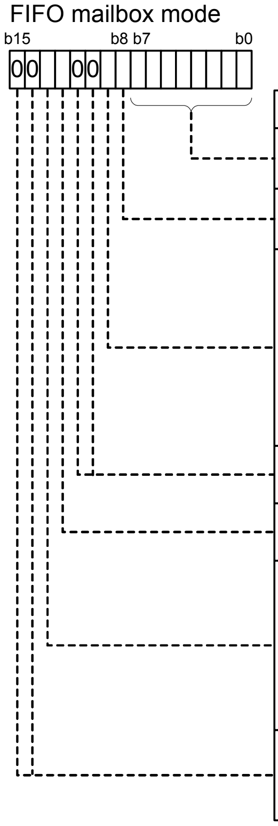
25.1.8 CANi Mailbox Interrupt Enable Register (CiMIER) (i = 0 to 5)

CANi Mailbox Interrupt Enable Register (i = 0 to 5) (1, 2, 4)

Symbol	Address	Reset Value
C0MIER, C1MIER	47F2Fh-47F2Eh, 47D2Fh-47D2Eh	Undefined
C2MIER, C3MIER	47B2Fh-47B2Eh, 4792Fh-4792Eh	Undefined
C4MIER, C5MIER	4772Fh-4772Eh, 4752Fh-4752Eh	Undefined



Bit Symbol	Bit Name	Function	RW
— (b15-b0)	Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW



Bit Symbol	Bit Name	Function	RW
— (b7-b0)	Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
— (b8)	Transmit FIFO Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
— (b9)	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	RW
— (b11-b10)	Reserved	Should be written with 0	RW
— (b12)	Receive FIFO Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
— (b13)	Receive FIFO Interrupt Generation Timing Control Bit	Receive FIFO interrupt request is generated 0: Every time reception is completed 1: When receive FIFO becomes buffer warning by completion of reception (3)	RW
— (b15-b14)	Reserved	Should be written with 0	RW

Notes:

1. Write to the CiMIER register only when the associated CiMCTLj register (j = 0 to 15) is 00h and the corresponding mailbox is not processing a transmission or reception abort request.
2. In FIFO mailbox mode, change the bits in the CiMIER 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.
3. No interrupt request is generated when the receive FIFO becomes buffer warning from full.
4. Set as follows when the gateway function is enabled (the OM bit in the GMR register is 1):
   - Set bits 15 to 12 to 1 (interrupt enabled) in normal mailbox mode (the MBM bit in the CiCTLR register is 0). However, the reception complete interrupt is not generated.
   - Set the 12 bit to 1 and the 13 bit to 0 (every time reception is completed) in FIFO mailbox mode (the MBM bit is 1). However, the receive FIFO interrupt is not generated.

Figure 25.9 Registers C0MIER to C5MIER

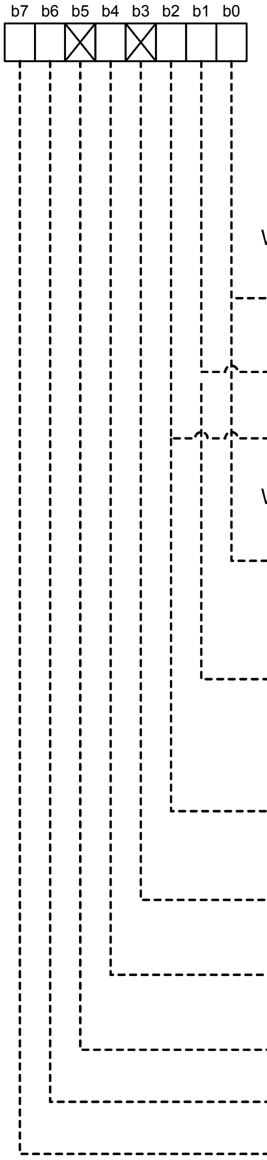
Interrupts can be individually enabled for each mailbox.

In normal mailbox mode (bits 0 to 15) and in FIFO mailbox mode (bits 0 to 7), each bit corresponds to the mailbox with the same number. These bits enable or disable transmission/reception complete interrupts for the corresponding mailboxes.
In FIFO mailbox mode, bits 8, 9, 12, and 13 specify whether transmit/receive FIFO interrupts are enabled/disabled and timing when interrupt requests are generated.
"Buffer warning" indicates a state in which the third unread message is stored in the receive FIFO.

25.1.9 CANi Message Control Register j (CiMCTLj) (i = 0 to 5; j = 0 to 15)

CANi Message Control Register j (i = 0 to 5; j = 0 to 15) (1, 2)

Symbol	Address	Reset Value
C0MCTL0 to C0MCTL15	47F30h to 47F3Fh	00h
C1MCTL0 to C1MCTL15	47D30h to 47D3Fh	00h
C2MCTL0 to C2MCTL15	47B30h to 47B3Fh	00h
C3MCTL0 to C3MCTL15	47930h to 4793Fh	00h
C4MCTL0 to C4MCTL15	47730h to 4773Fh	00h
C5MCTL0 to C5MCTL15	47530h to 4753Fh	00h



Bit Symbol	Bit Name	Function	RW
When the TRMREQ bit is 0 and the RECREQ bit is 1			
NEWDATA	Reception Complete Flag (3, 4)	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	RW
INVALDATA	Reception-in-progress Status Flag	0: Message valid 1: Message being updated	RO
MSGLOST	Message Lost Flag (3, 4)	0: Message is not overwritten or overrun 1: Message is overwritten or overrun	RW
When the TRMREQ bit is 1 and the RECREQ bit is 0			
SENTDATA	Transmission Complete Flag (3, 4)	0: Transmission is not completed (pending) 1: Transmission is completed (successful)	RW
TRMACTIVE	Transmission-in-progress Status Flag	0: Transmission is pending or transmission is not requested 1: From acceptance of transmission request to completion of transmission, or error/arbitration lost	RO
TRMABT	Transmission Abort Complete Flag (3, 4)	0: Transmission has started, transmission abort failed because transmission is completed, or transmission abort is not requested 1: Transmission abort is completed	RW
— (b3)	No register bit; should be written with 0 and read as 0		—
ONESHOT	One-shot Enable Bit (5, 8)	0: One-shot reception or one-shot transmission disabled 1: One-shot reception or one-shot transmission enabled	RW
— (b5)	No register bit; should be written with 0 and read as 0		—
RECREQ	Receive Mailbox Set Bit (4, 6, 7, 8)	0: Not configured for reception 1: Configured for reception	RW
TRMREQ	Transmit Mailbox Set Bit (4, 6, 8)	0: Not configured for transmission 1: Configured for transmission	RW

Notes:

1. Write to the CiMCTLj register in CAN operation mode or CAN halt mode.
2. Do not use registers CiMCTL8 to CiMCTL15 in FIFO mailbox mode.
3. It can only be set to 0. This setting is unchanged even if it is set to 1.
4. 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.
5. 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 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.
6. Do not set both the RECREQ and TRMREQ bits to 1.
7. When setting the RECREQ bit to 0, set bits MSGLOST, NEWDATA, RECREQ to 0 simultaneously.
8. Set as follows when the gateway function is enabled (the OM bit in the GMR register is 1):
   - Set the ONESHOT bit to 0.
   - Set 1 to the RECREQ bit for the gateway receive mailboxes [12] to [15] in normal mailbox mode (the MBM bit in the CiCTLR register is 0).
   - Set 0 to the TRMREQ bit for the gateway receive mailboxes [12] to [15].

Figure 25.10 Registers C0MCTLj to C5MCTLj

25.1.9.1 NEWDATA Bit

The NEWDATA bit becomes 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.
Do not set this bit to 0 by a program while the related INVALDATA bit is 1.

25.1.9.2 SENTDATA Bit

The SENTDATA bit becomes 1 when data transmission from the corresponding mailbox is completed.
This bit is set to 0 by writing 0 by a program.
When setting 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.

25.1.9.3 INVALDATA Bit

After a message has been received, the INVALDATA bit becomes 1 while the received message is updated into the corresponding mailbox.
This bit becomes 0 immediately after the message has been stored. If the mailbox is read while this bit is 1, the data is undefined.

25.1.9.4 TRMACTIVE Bit

The TRMACTIVE bit becomes 1 when the corresponding mailbox of the CAN module begins transmitting a message.
This bit becomes 0 when the CAN module loses CAN bus arbitration, a CAN bus error occurs, or data transmission is completed.

25.1.9.5 MSGLOST Bit

While the NEWDATA bit is 1, the MSGLOST bit becomes 1 when the mailbox is overwritten or overrun by a newly received message. This bit becomes 1 at the end of the sixth bit of EOF.
This bit is set to 0 by writing 0 by a program.
In both overwrite and overrun modes, during five cycles of fCAN (CAN system clock) following the sixth bit of EOF, the MSGLOST bit does not become 0 even if it is set to 0 by a program .

25.1.9.6 TRMABT Bit

The TRMABT bit becomes 1 in the following cases:

- Following a transmission abort request, the transmission abort is completed before starting transmission.
- Following a transmission abort request, the CAN module detects CAN bus arbitration lost or a CAN bus error.
- In one-shot transmission mode (RECREQ bit is 0, TRMREQ bit is 1, and ONESHOT bit is 1), the CAN module detects CAN bus arbitration lost or a CAN bus error.

The TRMABT bit does not become 1 when data transmission is completed. In this case, the SENTDATA bit becomes 1.
The TRMABT bit can be set to 0 by a program.

25.1.9.7 ONESHOT Bit

The ONESHOT bit can be used in receive mode and transmit mode.

(1) One-shot Receive Mode

When the ONESHOT bit is set to 1 in receive mode (RECREQ bit is 1 and TRMREQ bit is 0), the mailbox receives a message only once. The mailbox does not behave as a receive mailbox after having received a message once. The behavior of bits NEWDATA and INVALDATA is the same as in normal reception mode. In one-shot receive mode, the MSGLOST bit does not become 1.
To set the ONESHOT bit to 0, first write 0 to the RECREQ bit and ensure that it has been set to 0.

(2) One-shot Transmit Mode

When the ONESHOT bit is set to 1 in transmit mode (RECREQ bit is 0 and TRMREQ bit is 1), the CAN module transmits a message only once. The CAN module does not transmit the message again if a CAN bus error or CAN bus arbitration lost occurs. When the transmission is completed, the SENTDATA bit becomes 1. If the transmission cannot be completed due to a CAN bus error or CAN bus arbitration lost, the TRMABT bit becomes 1.
Set the ONESHOT bit to 0 after the SENTDATA or TRMABT bit is set to 1.

25.1.9.8 RECREQ Bit

Set the RECREQ bit to select one of the receive modes shown in Table 25.12.
When the RECREQ bit is set to 1, the corresponding mailbox is configured to receive a data frame or a remote frame.
When this 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 a program during the following period:
Hardware protection is started

- from the acceptance filter procedure (the beginning of the 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 the CRC field to the end of the seventh 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

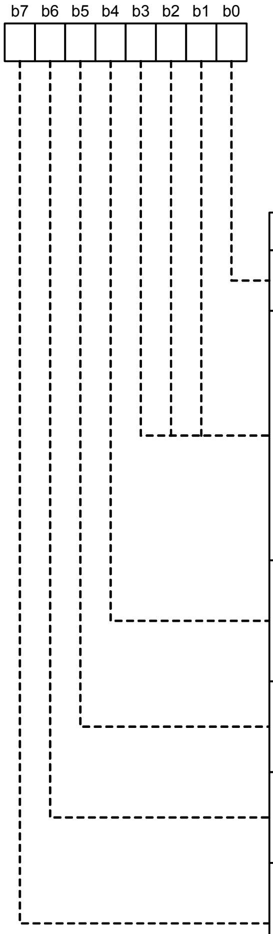
When setting the RECREQ bit to 1, do not set 1 to the TRMREQ bit.
To change the configuration of a mailbox from transmit to receive, first abort the transmission and then set bits SENTDATA and TRMABT to 0 before changing to receive.

25.1.9.9 TRMREQ Bit

The TRMREQ bit selects transmit modes shown in Table 25.12.
When this bit is set to 1, the corresponding mailbox is configured to transmit a data frame or a remote frame.
When this bit is set to 0, the corresponding mailbox is not configured to transmit 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 receive to transmit, first abort the reception and then set bits NEWDATA and MSGLOST to 0 before changing to transmit.

25.1.10 CANi Receive FIFO Control Register (CiRFCR) (i = 0 to 5)

CANi Receive FIFO Control Register (i = 0 to 5) (1)



Symbol	Address	Reset Value
C0RFCR	47F48h	1000 0000b
C1RFCR	47D48h	1000 0000b
C2RFCR	47B48h	1000 0000b
C3RFCR	47948h	1000 0000b
C4RFCR	47748h	1000 0000b
C5RFCR	47548h	1000 0000b

Bit Symbol	Bit Name	Function	RW
RFE	Receive FIFO Enable Bit (2)	0: Receive FIFO disabled 1: Receive FIFO enabled	RW
RFUST	Receive FIFO Unread Message Number Status Bit	b3 b2 b1 0 0 0 : No unread messages 0 0 1 : 1 unread message 0 1 0 : 2 unread messages 0 1 1 : 3 unread messages 1 0 0 : 4 unread messages 1 0 1 : Reserved 1 1 0 : Reserved 1 1 1 : Reserved	RO
RFMLF	Receive FIFO Message Lost Flag (3)	0: No receive FIFO message lost has occurred 1: Receive FIFO message lost has occurred	RW
RFFST	Receive FIFO Full Status Bit	0: Receive FIFO is not full 1: Receive FIFO is full (4 unread messages)	RO
RFWST	Receive FIFO Buffer Warning Status Bit	0: Receive FIFO is not buffer warning 1: Receive FIFO is buffer warning (3 unread messages)	RO
RFEST	Receive FIFO Empty Status Bit	0: Unread message in receive FIFO 1: No unread message in receive FIFO	RO

Notes:

1. Write to the CiRFCR register in CAN operation mode or CAN halt mode.
2. When setting the RFE bit to 0, set the RFMLF bit to 0 as well.
3. It can only be set to 0. This setting is unchanged even if it is set to 1.

Figure 25.11 Registers C0RFCR to C5RFCR

25.1.10.1 RFE 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 is 1).
Do not set this bit to 1 in normal mailbox mode (MBM bit in the CiCTLR register (i = 0 to 5) is 0).
Due to hardware protection, the RFE bit is not set to 0 by a program during the following period:
Hardware protection is started

- from the acceptance filter procedure (the beginning of the CRC field)

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. a maximum period of hardware protection is from the beginning of the CRC field to the end of the seventh bit of EOF).
- if the receive FIFO is not specified to receive the message, after the acceptance filter procedure.

25.1.10.2 RFUST Bit

The RFUST bit indicates the number of unread messages in the receive FIFO.
The value of this bit is initialized to 000b when the RFE bit is set to 0.

25.1.10.3 RFMLF Bit

The RFMLF bit becomes 1 (receive FIFO message lost has occurred) when the receive FIFO receives a new message and the receive FIFO is full. This bit becomes 1 at the end of the sixth bit of EOF.
The RFMLF bit is set to 0 by a program.
In both overwrite and overrun modes, this bit cannot be set to 0 (no receive FIFO message lost has occurred) by writing 0 by a program due to hardware protection during the five cycles of fCAN following the sixth bit of EOF, if the receive FIFO is full and determined to receive the message.

25.1.10.4 RFFST Bit

The RFFST bit becomes 1 (receive FIFO is full) when there are four unread messages in the receive FIFO. This bit becomes 0 (receive FIFO is not full) when there are less than four unread messages in the receive FIFO. This bit becomes 0 when the RFE bit is 0.

25.1.10.5 RFWST Bit

The RFWST bit becomes 1 (receive FIFO is buffer warning) when there are three unread messages in the receive FIFO. This bit becomes 0 (receive FIFO is not buffer warning) when there are less than three or equal to four unread messages in the receive FIFO. This bit becomes 0 when the RFE bit is 0.

25.1.10.6 RFEST Bit

The RFEST bit becomes 1 (no unread message in receive FIFO) when there are no unread messages in the receive FIFO. This bit becomes 1 when the RFE bit is set to 0. The RFEST bit becomes 0 (unread message in receive FIFO) when there is one or more unread messages in the receive FIFO.

Figure 25.12 shows receive FIFO mailbox operation.

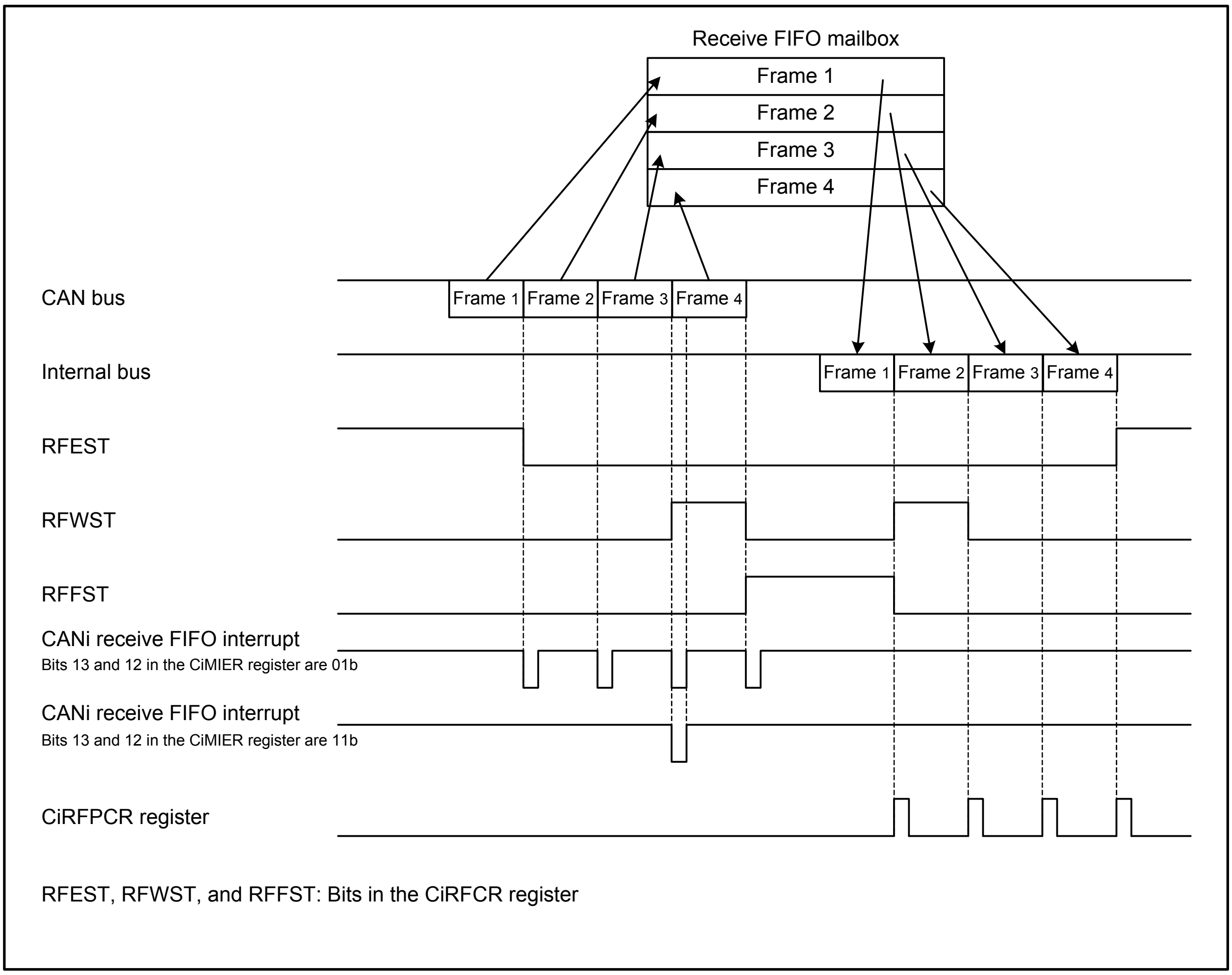
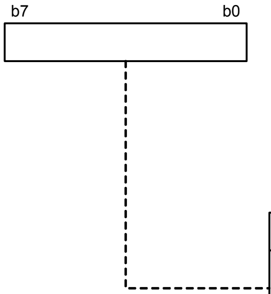


Figure 25.12 Receive FIFO Mailbox Operation (Bits 13 and 12 in CiMIER Register are 01b and 11b) (i = 0 to 5)

25.1.11 CANi Receive FIFO Pointer Control Register (CiRFPCR) (i = 0 to 5)

CANi Receive FIFO Pointer Control Register (i = 0 to 5)



Symbol	Address	Reset Value
C0RFPCR	47F49h	Undefined
C1RFPCR	47D49h	Undefined
C2RFPCR	47B49h	Undefined
C3RFPCR	47949h	Undefined
C4RFPCR	47749h	Undefined
C5RFPCR	47549h	Undefined

Function	Setting Value	RW
The CPU-side pointer for the receive FIFO increments by writing FFh	FFh	WO

Figure 25.13 Registers C0RFPCR to C5RFPCR

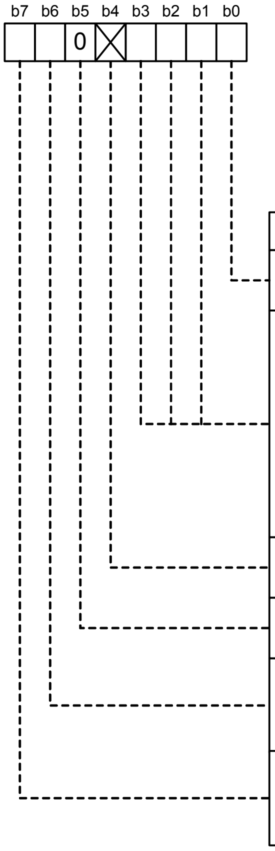
When there are messages in the receive FIFO, write FFh to the CiRFPCR register by a program to increment the CPU-side pointer for the receive FIFO to the next mailbox location.

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 increment when a new message is received and 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 does not increment by writing to the CiRFPCR register by a program.

25.1.12 CANi Transmit FIFO Control Register (CiTFCR) (i = 0 to 5)

CANi Transmit FIFO Control Register (i = 0 to 5) (1)



Symbol	Address	Reset Value
C0TFCR	47F4Ah	1000 0000b
C1TFCR	47D4Ah	1000 0000b
C2TFCR	47B4Ah	1000 0000b
C3TFCR	4794Ah	1000 0000b
C4TFCR	4774Ah	1000 0000b
C5TFCR	4754Ah	1000 0000b

Bit Symbol	Bit Name	Function	RW
TFE	Transmit FIFO Enable Bit	0: Transmit FIFO disabled 1: Transmit FIFO enabled	RW
TFUST	Transmit FIFO Unsent Message Number Status Bit	b3 b2 b1 0 0 0 : No unsent messages 0 0 1 : 1 unsent message 0 1 0 : 2 unsent messages 0 1 1 : 3 unsent messages 1 0 0 : 4 unsent messages 1 0 1 : Reserved 1 1 0 : Reserved 1 1 1 : Reserved	RO
— (b4)	No register bit; should be written with 0 and read as 0		—
— (b5)	Reserved	Should be written with 0 and read as undefined value	RO
TFFST	Transmit FIFO Full Status Bit	0: Transmit FIFO is not full 1: Transmit FIFO is full (4 unsent messages)	RO
TFEST	Transmit FIFO Empty Status Bit	0: Unsent message in transmit FIFO 1: No unsent message in transmit FIFO	RO

Note:

1. Write to the CiTFCR register in CAN operation mode or CAN halt mode.

Figure 25.14 Registers C0TFCR to C5TFCR

25.1.12.1 TFE Bit

When the TFE bit is set to 1, the transmit FIFO is enabled.

When this bit is set to 0, the transmit FIFO becomes empty (TFEST bit is 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 a 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 1 again, ensure that the TFEST bit is 1.

After setting the TFE bit to 1, write transmit data to the CiMB8 register.

Do not set this bit to 1 in normal mailbox mode (MBM bit in the CiCTLR register is 0).

25.1.12.2 TFUST 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 000b when transmission abort or transmission is completed.

25.1.12.3 TFFST Bit

The TFFST bit becomes 1 (transmit FIFO is full) when there are four unsent messages in the transmit FIFO. This bit becomes 0 (transmit FIFO is not full) when there are less than four unsent messages in the transmit FIFO. This bit becomes 0 when a transmission from the transmit FIFO has been aborted.

25.1.12.4 TFEST Bit

The TFEST bit becomes 1 (no unsent message in transmit FIFO) when there are no unsent messages in the transmit FIFO. This bit becomes 1 when transmission from the transmit FIFO has been aborted. The TFEST bit becomes 0 (unsent message in transmit FIFO) when there is at least one unsent messages in the transmit FIFO.

Figure 25.15 shows transmit FIFO mailbox operation.

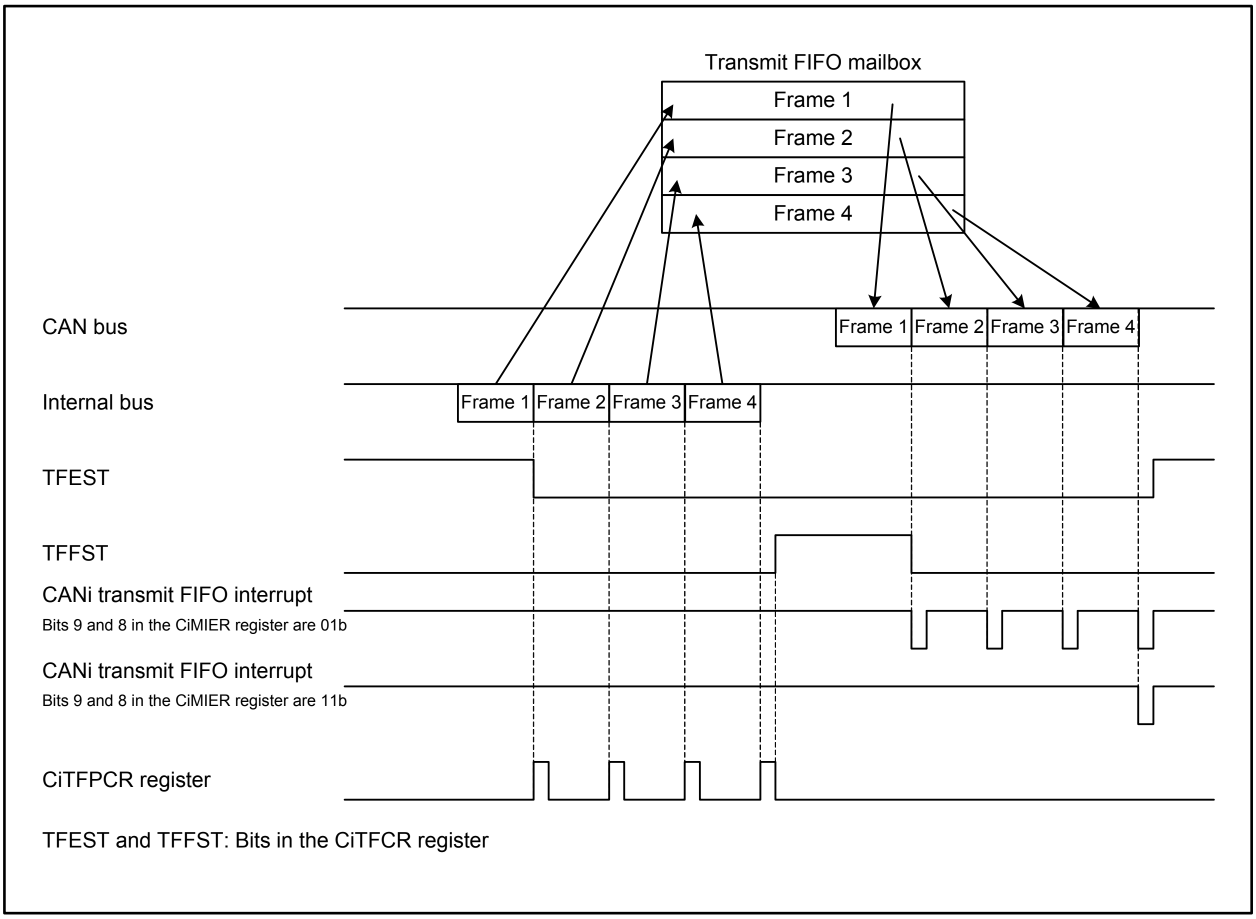
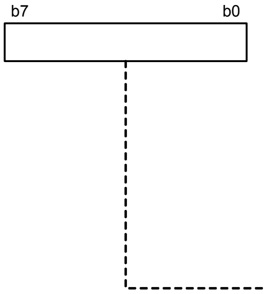


Figure 25.15 Transmit FIFO Mailbox Operation (Bits 9 and 8 in CiMIER Register are 01b and 11b) (i = 0 to 5)

25.1.13 CANi Transmit FIFO Pointer Control Register (CiTFPCR) (i = 0 to 5)

CANi Transmit FIFO Pointer Control Register (i = 0 to 5)



Symbol	Address	Reset Value
C0TFPCR	47F4Bh	Undefined
C1TFPCR	47D4Bh	Undefined
C2TFPCR	47B4Bh	Undefined
C3TFPCR	4794Bh	Undefined
C4TFPCR	4774Bh	Undefined
C5TFPCR	4754Bh	Undefined

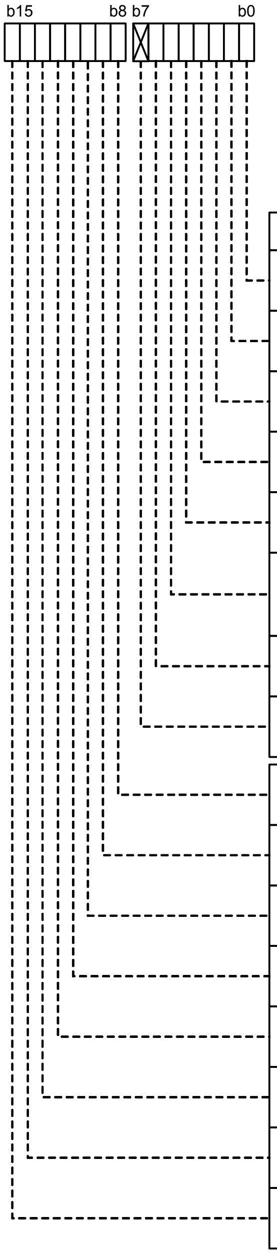
Function	Setting Value	RW
The CPU-side pointer for the transmit FIFO increments by writing FFh	FFh	WO

Figure 25.16 Registers C0TFPCR to C5TFPCR

When the transmit FIFO is not full, write FFh 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 CiTFCR register is 0 (transmit FIFO disabled).

25.1.14 CANi Status Register (CiSTR) (i = 0 to 5)

CANi Status Register (i = 0 to 5)



Symbol	Address	Reset Value
C0STR	47F43h-47F42h	0000 0000 0000 0101b
C1STR	47D43h-47D42h	0000 0000 0000 0101b
C2STR	47B43h-47B42h	0000 0000 0000 0101b
C3STR	47943h-47942h	0000 0000 0000 0101b
C4STR	47743h-47742h	0000 0000 0000 0101b
C5STR	47543h-47542h	0000 0000 0000 0101b

Bit Symbol	Bit Name	Function	RW
RSTST	CAN Reset Status Flag	0: Not in CAN reset mode 1: In CAN reset mode	RO
HLTST	CAN Halt Status Flag	0: Not in CAN halt mode 1: In CAN halt mode	RO
SLPST	CAN Sleep Status Flag	0: Not in CAN sleep mode 1: In CAN sleep mode	RO
EPST	Error-passive Status Flag	0: Not in error-passive state 1: In error-passive state	RO
BOST	Bus-off Status Flag	0: Not in bus-off state 1: In bus-off state	RO
TRMST	Transmit Status Flag (transmitter)	0: Bus idle or reception in progress 1: Transmission in progress or in bus-off state	RO
RECST	Receive Status Flag (receiver)	0: Bus idle or transmission in progress 1: Reception in progress	RO
— (b7)	No register bit; this bit is read as 0		—
NDST	NEWDATA Status Flag	0: No mailbox with NEWDATA bit is 1 1: Mailbox with NEWDATA bit is 1	RO
SDST	SENTDATA Status Flag	0: No mailbox with SENTDATA bit is 1 1: Mailbox with SENTDATA bit is 1	RO
RFST	Receive FIFO Status Flag	0: No message in receive FIFO 1: Message in receive FIFO	RO
TFST	Transmit FIFO Status Flag	0: Transmit FIFO is full 1: Transmit FIFO is not full	RO
NMLST	Normal Mailbox Message Lost Status Flag	0: No mailbox with MSGLOST bit is 1 1: Mailbox with MSGLOST bit is 1	RO
FMLST	FIFO Mailbox Message Lost Status Flag	0: RFMLF bit is 0 1: RFMLF bit is 1	RO
TABST	Transmission Abort Status Flag	0: No mailbox with TRMABT bit is 1 1: Mailbox with TRMABT bit is 1	RO
EST	Error Status Flag	0: No error occurred 1: Error occurred	RO

Figure 25.17 Registers C0STR to C5STR

25.1.14.1 RSTST Bit

The RSTST bit becomes 1 when the CAN module enters CAN reset mode.
This 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.

25.1.14.2 HLTST Bit

The HLTST bit becomes 1 when the CAN module enters CAN halt mode.
This bit is 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.

25.1.14.3 SLPST Bit

The SLPST bit becomes 1 when the CAN module enters CAN sleep mode.
This bit is 0 when the CAN module is not in CAN sleep mode.

25.1.14.4 EPST Bit

The EPST bit becomes 1 when the value of the CiTECR or CiRECR register (i = 0 to 5) exceeds 127 and the CAN module enters the error-passive state ($128 \leq TEC < 256$ or $128 \leq REC < 256$). This bit is 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).

25.1.14.5 BOST Bit

The BOST bit becomes 1 when the value of the CiTECR register exceeds 255 and the CAN module enters the bus-off state ($TEC \geq 256$). This bit is 0 when the CAN module is not in the bus-off state.

25.1.14.6 TRMST Bit

The TRMST bit becomes 1 when the CAN module performs as a transmitter node or enters the bus-off state.
This bit becomes 0 when the CAN module performs as a receiver node or enters the bus-idle state.

25.1.14.7 RECST Bit

The RECST bit becomes 1 when the CAN module performs as a receiver node.
This bit becomes 0 when the CAN module performs as a transmitter node or enters the bus-idle state.

25.1.14.8 NDST Bit

The NDST bit becomes 1 when at least one NEWDATA bit in the CiMCTLj register (j = 0 to 15) is 1 regardless of the value of the CiMIER register.
The NDST bit becomes 0 when all NEWDATA bits are 0.

25.1.14.9 SDST Bit

The SDST bit becomes 1 when at least one SENTDATA bit in the CiMCTLj register (i = 0 to 5; j = 0 to 15) is 1 regardless of the value of the CiMIER register.
The SDST bit becomes 0 when all SENTDATA bits are 0.

25.1.14.10 RFST Bit

The RFST bit is 1 when there are messages in the receive FIFO.
This bit is 0 when the receive FIFO is empty.
This bit becomes 0 when normal mailbox mode is selected.

25.1.14.11 TFST Bit

The TFST bit is 1 when the transmit FIFO is not full.
This bit is 0 when the transmit FIFO is full.
This bit becomes 0 when normal mailbox mode is selected.

25.1.14.12 NMLST Bit

The NMLST bit becomes 1 when at least one MSGLOST bit in the CiMCTLj register is 1 regardless of the value of the CiMIER register.
The NMLST bit becomes 0 when all MSGLOST bits are 0.

25.1.14.13 FMLST Bit

The FMLST bit becomes 1 when the RFMLF bit in the CiRFCR register is 1 regardless of the value of the CiMIER register.
The FMLST bit becomes 0 when the RFMLF bit is 0.

25.1.14.14 TABST Bit

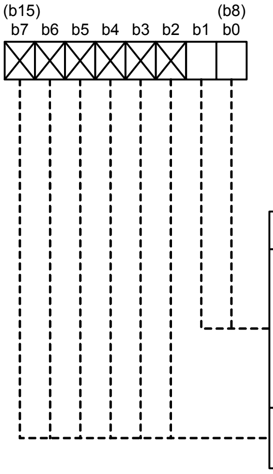
The TABST bit becomes 1 when at least one TRMABT bit in the CiMCTLj register is 1 regardless of the value of the CiMIER register.
The TABST bit becomes 0 when all TRMABT bits are 0.

25.1.14.15 EST Bit

The EST bit becomes 1 when at least one error is detected by the CiEIFR register regardless of the value of the CiEIER register.
This bit becomes 0 when no error is detected by the CiEIFR register.

25.1.15 CANi Mailbox Search Mode Register (CiMSMR) (i = 0 to 5)

CANi Mailbox Search Mode Register (i = 0 to 5) [1]



Symbol	Address	Reset Value
C0MSMR	47F53h	0000 0000b
C1MSMR	47D53h	0000 0000b
C2MSMR	47B53h	0000 0000b
C3MSMR	47953h	0000 0000b
C4MSMR	47753h	0000 0000b
C5MSMR	47553h	0000 0000b

Bit Symbol	Bit Name	Function	RW
MBSM	Mailbox Search Mode Select Bit	b1 b0 0 0 : Receive mailbox search mode 0 1 : Transmit mailbox search mode 1 0 : Message lost search mode 1 1 : Channel search mode	RW
— (b7-b2)	No register bits; should be written with 0 and read as 0		—

Note:

1. Write to the CiMSMR register in CAN operation mode or CAN halt mode.

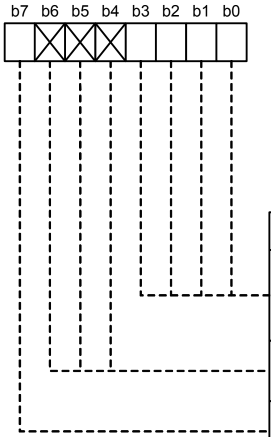
Figure 25.18 Registers C0MSMR to C5MSMR

25.1.15.1 MBSM Bit

Set the MBSM bit to select the search mode for the mailbox search function.
When this bit is 00b, receive mailbox search mode is selected. In this mode, the search targets are the NEWDATA bit in the CiMCTLj register (j = 0 to 15) for the normal mailbox and the RFEST bit in the CiRFCR register.
When the MBSM bit is 01b, transmit mailbox search mode is selected. In this mode, the search target is the SENTDATA bit in the CiMCTLj register.
When the MBSM bit is 10b, message lost search mode is selected. In this mode, the search targets are the MSGLOST bit in the CiMCTLj register for the normal mailbox and the RFMLF bit in the CiRFCR register.
When the MBSM bit is 11b, channel search mode is selected. In this mode, the search target is the CiCSSR register.
Refer to 25.1.17 “CANi Channel Search Support Register (CiCSSR) (i = 0 to 5)”.

25.1.16 CANi Mailbox Search Status Register (CiMSSR) (i = 0 to 5)

CANi Mailbox Search Status Register (i = 0 to 5)



Symbol	Address	Reset Value
C0MSSR	47F52h	1000 0000b
C1MSSR	47D52h	1000 0000b
C2MSSR	47B52h	1000 0000b
C3MSSR	47952h	1000 0000b
C4MSSR	47752h	1000 0000b
C5MSSR	47552h	1000 0000b

Bit Symbol	Bit Name	Function	RW
MBNST	Search Result Mailbox Number Status Bit	Output of search result in each search mode Output number: 0 to 15	RO
— (b6-b4)	No register bits; these bits are read as 0		—
SEST	Search Result Status Bit	0: Search result found 1: No search result	RO

Figure 25.19 Registers C0MSSR to C5MSSR

25.1.16.1 MBNST Bit

The MBNST bit outputs the smallest mailbox number that is searched in each mode of the CiMSMR register (i = 0 to 5).

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 follows:

- 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 [12]) is output when there are messages in the receive FIFO, and there are no unread received messages or lost messages in any of the normal mailboxes (mailboxes [0] to [7]).

In transmit mailbox search mode, the transmit FIFO (mailbox [8]) is not output.

Table 25.8 lists the operation of MBNST bit in FIFO mailbox mode.

Table 25.8 Operation of MBNST Bit in FIFO Mailbox Mode

MBSM Bit	Mailbox [8] (Transmit FIFO)	Mailbox [12] (Receive FIFO)
00b	Mailbox [8] is not output	Mailbox [12] is output when no NEWDATA bit for the normal mailbox is set to 1 and there are messages in the receive FIFO
01b		Mailbox [12] is not output
10b		Mailbox [12] is output when no MSGLOST bit for the normal mailbox is set to 1 and the RFMLF bit is set to 1 in the receive FIFO
11b		Mailbox [12] is not output

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.

25.1.16.2 SEST Bit

The SEST bit becomes 1 when no corresponding mailbox is found after searching all mailboxes.

For example, in transmit mailbox search mode, the SEST bit becomes 1 when no SENTDATA bit for mailboxes is 1. The SEST bit becomes 0 when at lease one SENTDATA bit is 1.

When the SEST bit is 1, the value of the MBNST bit is undefined.

25.1.17 CANi Channel Search Support Register (CiCSSR) (i = 0 to 5)

CANi Channel Search Support Register (i = 0 to 5) [1, 2]

b7 b0

Symbol	Address	Reset Value
C0CSSR	47F51h	Undefined
C1CSSR	47D51h	Undefined
C2CSSR	47B51h	Undefined
C3CSSR	47951h	Undefined
C4CSSR	47751h	Undefined
C5CSSR	47551h	Undefined

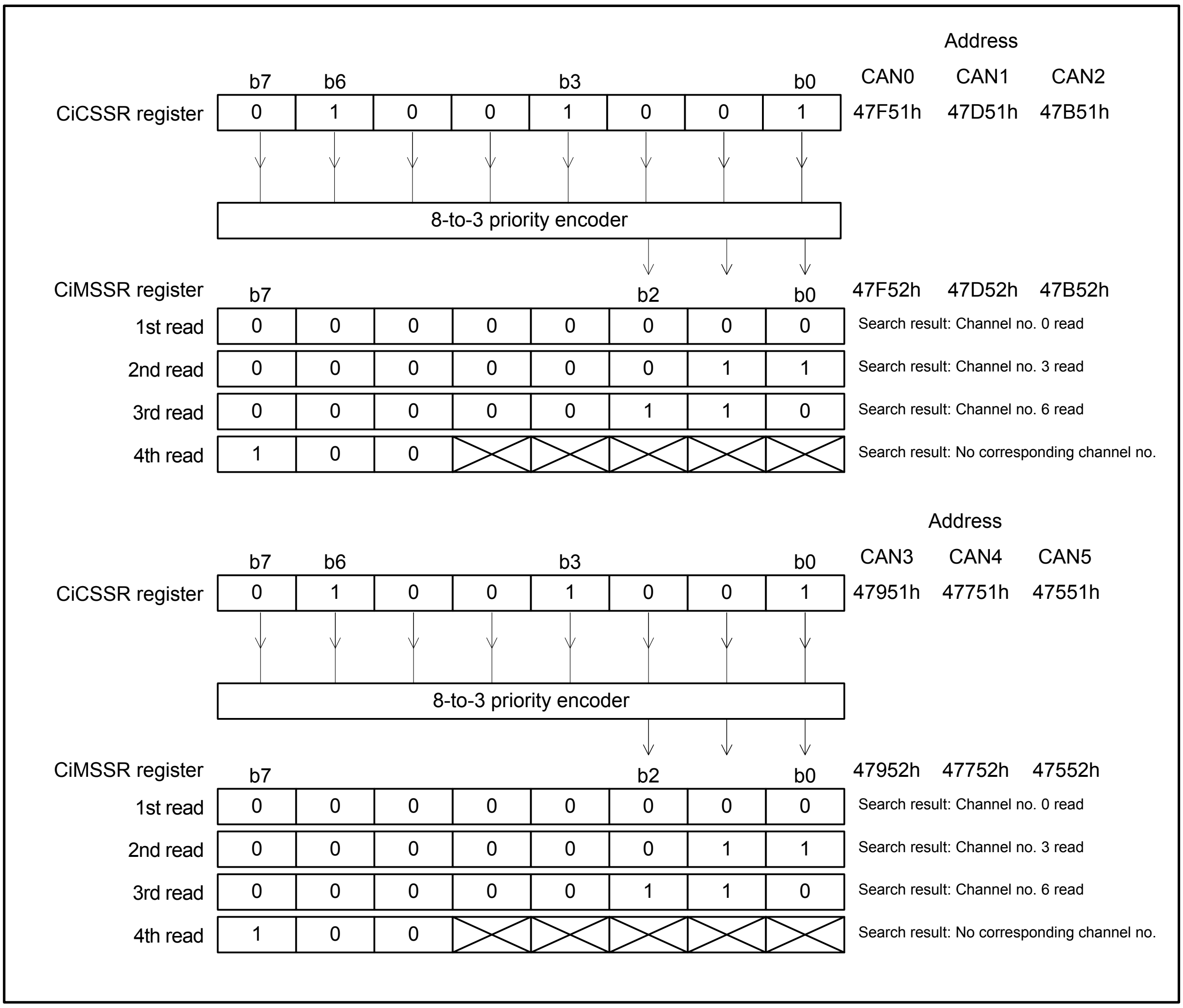
Function	Setting Value	RW
When the value of the channel search is input, the channel number is output to the CiMSSR register	Channel value	RW

Notes:

1. Write to the CiCSSR register only when the MBSM bit in the CiMSMR register is 11b (channel search mode).
2. Write to the CiCSSR register in CAN operation mode or CAN halt mode.

Figure 25.20 Registers C0CSSR to C5CSSR

The bits in the CiCSSR register, which are set to 1, are encoded by an 8-to-3 priority encoder (the lower bit position, the higher priority) and output to the MBNST bits in the CiMSSR register.
The value of the CiMSSR register is updated whenever the CiMSSR register is read.
Figure 25.21 shows the write and read of registers CiCSSR and CiMSSR.



Address

	CAN0	CAN1	CAN2
CiCSSR	47F51h	47D51h	47B51h
CiMSSR	47F52h	47D52h	47B52h

	b7	b6			b3			b0
CiCSSR register	0	1	0	0	1	0	0	1

CiMSSR register	b7					b2		b0	
1st read	0	0	0	0	0	0	0	0	Search result: Channel no. 0 read
2nd read	0	0	0	0	0	0	1	1	Search result: Channel no. 3 read
3rd read	0	0	0	0	0	1	1	0	Search result: Channel no. 6 read
4th read	1	0	0	☒	☒	☒	☒	☒	Search result: No corresponding channel no.

Address

	CAN3	CAN4	CAN5
CiCSSR	47951h	47751h	47551h
CiMSSR	47952h	47752h	47552h

	b7	b6			b3			b0
CiCSSR register	0	1	0	0	1	0	0	1

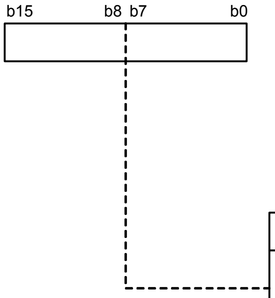
CiMSSR register	b7					b2		b0	
1st read	0	0	0	0	0	0	0	0	Search result: Channel no. 0 read
2nd read	0	0	0	0	0	0	1	1	Search result: Channel no. 3 read
3rd read	0	0	0	0	0	1	1	0	Search result: Channel no. 6 read
4th read	1	0	0	☒	☒	☒	☒	☒	Search result: No corresponding channel no.

Figure 25.21 Write and Read of Registers CiCSSR and CiMSSR (i = 0 to 5)

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-to-3 priority encoder conversion is read.

25.1.18 CANi Acceptance Filter Support Register (CiAFSR) (i = 0 to 5)

CANi Acceptance Filter Support Register (i = 0 to 5) (1)



Symbol	Address	Reset Value
C0AFSR	47F57h-47F56h	Undefined
C1AFSR	47D57h-47D56h	Undefined
C2AFSR	47B57h-47B56h	Undefined
C3AFSR	47957h-47956h	Undefined
C4AFSR	47757h-47756h	Undefined
C5AFSR	47557h-47556h	Undefined

Function	Setting Value	RW
After the standard ID of a received message is written, the value converted for data table search can be read	Standard ID/ converted value	RW

Note:

1. Write to the CiAFSR register in CAN operation mode or CAN halt mode.

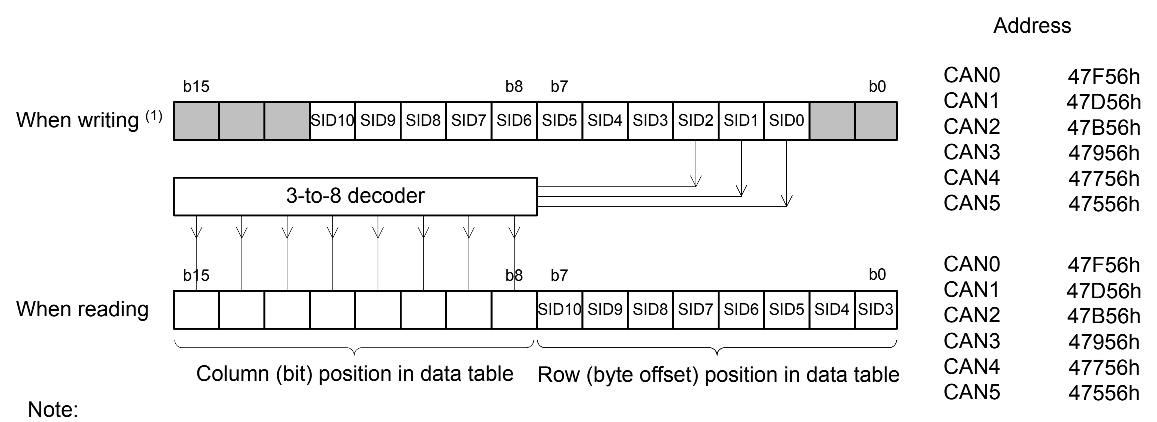
Figure 25.22 C0AFSR to C5AFSR

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 enabled/disabled in bit units. When the CiAFSR register is written with 16-bit unit data including the SID bit in the CiMBj register (j = 0 to 15), 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: 078h, 087h, 111h
- When there are too many IDs to receive and software filtering time is expected to be shortened.

Figure 25.23 shows the write and read of CiAFSR register.



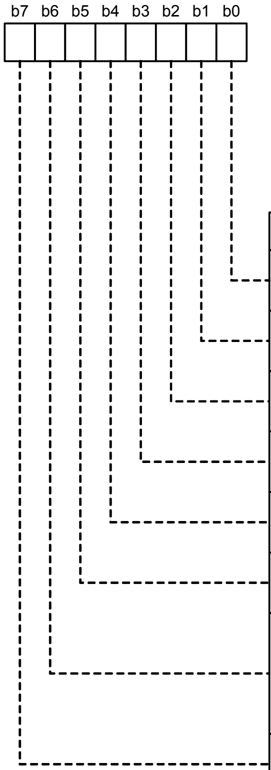
Note:

1. Write the same value as the 16-bit unit data including the SID bit in the CiMBj register.

Figure 25.23 Write and Read of CiAFSR Register (i = 0 to 5; j = 0 to 15)

25.1.19 CANi Error Interrupt Enable Register (CiEIER) (i = 0 to 5)

CANi Error Interrupt Enable Register (i = 0 to 5) (1)



Symbol	Address	Reset Value
C0EIER	47F4Ch	00h
C1EIER	47D4Ch	00h
C2EIER	47B4Ch	00h
C3EIER	4794Ch	00h
C4EIER	4774Ch	00h
C5EIER	4754Ch	00h

Bit Symbol	Bit Name	Function	RW
BEIE	Bus Error Interrupt Enable Bit	0: Bus error interrupt disabled 1: Bus error interrupt enabled	RW
EWIE	Error Warning Interrupt Enable Bit	0: Error warning interrupt disabled 1: Error warning interrupt enabled	RW
EPIE	Error Passive Interrupt Enable Bit	0: Error passive interrupt disabled 1: Error passive interrupt enabled	RW
BOEIE	Bus-off Entry Interrupt Enable Bit	0: Bus-off entry interrupt disabled 1: Bus-off entry interrupt enabled	RW
BORIE	Bus-off Recovery Interrupt Enable Bit	0: Bus-off recovery interrupt disabled 1: Bus-off recovery interrupt enabled	RW
ORIE	Receive Overrun Interrupt Enable Bit	0: Receive overrun interrupt disabled 1: Receive overrun interrupt enabled	RW
OLIE	Overload Frame Transmit Interrupt Enable Bit	0: Overload frame transmit interrupt disabled 1: Overload frame transmit interrupt enabled	RW
BLIE	Bus Lock Interrupt Enable Bit	0: Bus lock interrupt disabled 1: Bus lock interrupt enabled	RW

Note:

1. Write to the CiEIER register in CAN reset mode.

Figure 25.24 Registers C0EIER to C5EIER

The CiEIER register is used to set the error interrupt enabled/disabled individually for each error interrupt source in the CiEIFR register.

25.1.19.1 BEIE Bit

When the BEIE bit is 0, no error interrupt request is generated even if the BEIF bit in the CiEIFR register (i = 0 to 5) is set to 1.
When the BEIE bit is 1, an error interrupt request is generated if the BEIF bit is set to 1.

25.1.19.2 EWIE 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.

25.1.19.3 EPIE 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.

25.1.19.4 BOEIE 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.

25.1.19.5 BORIE 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 1, an error interrupt request is generated if the BORIF bit is set to 1.

25.1.19.6 ORIE Bit

When the ORIE bit is 0, no error interrupt request is 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.

25.1.19.7 OLIE Bit

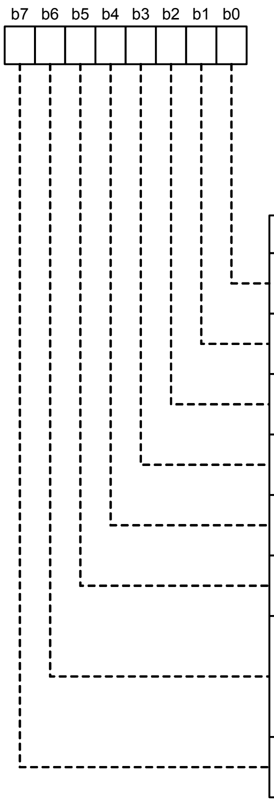
When the OLIE bit is 0, no error interrupt request is generated even if the OLIF bit in the CiEIFR register is set to 1.
When the OLIE bit is 1, an error interrupt request is generated if the OLIF bit is set to 1.

25.1.19.8 BLIE 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.

25.1.20 CANi Error Interrupt Factor Judge Register (CiEIFR) (i = 0 to 5)

CANi Error Interrupt Factor Judge Register (i = 0 to 5) (1)



Symbol	Address	Reset Value
C0EIFR	47F4Dh	00h
C1EIFR	47D4Dh	00h
C2EIFR	47B4Dh	00h
C3EIFR	4794Dh	00h
C4EIFR	4774Dh	00h
C5EIFR	4754Dh	00h

Bit Symbol	Bit Name	Function	RW
BEIF	Bus Error Detect Flag	0: No bus error detected 1: Bus error detected	RW
EWIF	Error Warning Detect Flag	0: No error warning detected 1: Error warning detected	RW
EPIF	Error Passive Detect Flag	0: No error passive detected 1: Error passive detected	RW
BOEIF	Bus-off Entry Detect Flag	0: No bus-off entry detected 1: Bus-off entry detected	RW
BORIF	Bus-off Recovery Detect Flag	0: No bus-off recovery detected 1: Bus-off recovery detected	RW
ORIF	Receive Overrun Detect Flag	0: No receive overrun detected 1: Receive overrun detected	RW
OLIF	Overload Frame Transmission Detect Flag	0: No overload frame transmission detected 1: Overload frame transmission detected	RW
BLIF	Bus Lock Detect Flag	0: No bus lock detected 1: Bus lock detected	RW

Note:

1. When writing 0 to these bits 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 on these bit values.

Figure 25.25 Registers C0EIFR to C5EIFR

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.

25.1.20.1 BEIF Bit

The BEIF bit becomes 1 when a bus error is detected.

25.1.20.2 EWIF Bit

The EWIF bit becomes 1 when the value of the receive error counter (REC) or transmit error counter (TEC) exceeds 95.
This bit becomes 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, this bit does not become 1 until the REC and the TEC go below 95 and then exceed 95 again.

25.1.20.3 EPIF Bit

The EPIF bit becomes 1 when the CAN error state enters the error-passive state (the REC or TEC value exceeds 127).
This bit becomes 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, this bit does not become 1 until the REC and the TEC go below 127 and then exceed 127 again.

25.1.20.4 BOEIF Bit

The BOEIF bit becomes 1 when the CAN error state enters the bus-off state (the TEC value exceeds 255).
This bit also becomes 1 when the BOM bit in the CiCTLR register (i = 0 to 5) is 01b (entry to CAN halt mode automatically at bus-off entry) and the CAN module enters the bus-off state.

25.1.20.5 BORIF Bit

The BORIF bit becomes 1 when the CAN module recovers from the bus-off state normally by detecting 11 consecutive bits 128 times in the following conditions:

(1) When the BOM bit in the CiCTLR register is 00b
(2) When the BOM bit is 10b
(3) When the BOM bit is 11b

The BORIF bit does not become 1 if the CAN module recovers from the bus-off state in the following conditions:

(1) When the CANM bit in the CiCTLR register is set to 01b (CAN reset mode)
(2) When the RBOC bit in the CiCTLR register is set to 1 (forced recovery from bus-off)
(3) When the BOM bit is 01b
(4) When the BOM bit is 11b and the CANM bit is set to 10b (CAN halt mode) before normal recovery occurs

Table 25.9 lists the operation of bits BOEIF and BORIF according to BOM bit setting value.

Table 25.9 Operation of Bits BOEIF and BORIF According to BOM Bit Setting Value

BOM Bit	BOEIF Bit	BORIF Bit
00b	Becomes 1 when entering the bus-off state	Becomes 1 when recovering from the bus-off state
01b		Does not become 1
10b		Becomes 1 when recovering from the bus-off state
11b		Becomes 1 if normal bus-off recovery occurs before the CANM bit is set to 10b (CAN halt mode)

25.1.20.6 ORIF Bit

The ORIF bit becomes 1 when a receive overrun occurs.
This bit does not become 1 in overwrite mode. In overwrite mode, a reception complete interrupt request is generated if an overwrite condition occurs, thus this bit does not become 1.
In normal mailbox mode, if an overrun occurs in any mailbox from [0] to [15] in overrun mode, this bit is set to 1.
In FIFO mailbox mode, if an overrun occurs in any mailbox from [0] to [7] or the receive FIFO in overrun mode, this bit becomes 1.

25.1.20.7 OLIF Bit

The OLIF bit becomes 1 if the transmitting condition of an overload frame is detected when the CAN module performs transmission or reception.

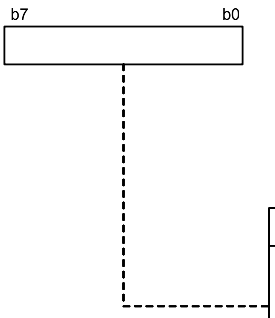
25.1.20.8 BLIF Bit

The BLIF bit becomes 1 if 32 consecutive dominant bits are detected on the CAN bus while the CAN module is in CAN operation mode.
After the BLIF bit becomes 1, 32 consecutive dominant bits are detected again 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.

25.1.21 CANi Receive Error Count Register (CiRECR) (i = 0 to 5)

CANi Receive Error Count Register (i = 0 to 5)



Symbol	Address	Reset Value
C0RECR	47F4Eh	00h
C1RECR	47D4Eh	00h
C2RECR	47B4Eh	00h
C3RECR	4794Eh	00h
C4RECR	4774Eh	00h
C5RECR	4754Eh	00h

Function	Counter Value	RW
Receive error count function The CiRECR register increments or decrements the counter value according to the error status of the CAN module during reception	00h to FFh (1)	RO

Note:

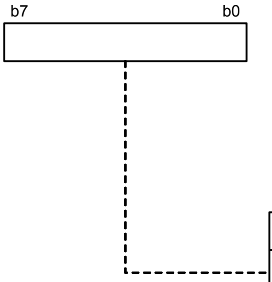
1. The value in the bus-off state is undefined.

Figure 25.26 Registers C0RECR to C5RECR

The CiRECR register indicates the value of the receive error counter.
Refer to the CAN Specification (ISO 11898-1) for the increment/decrement conditions of the receive error counter.

25.1.22 CANi Transmit Error Count Register (CiTECR) (i = 0 to 5)

CANi Transmit Error Count Register (i = 0 to 5)



Symbol	Address	Reset Value
C0TECR	47F4Fh	00h
C1TECR	47D4Fh	00h
C2TECR	47B4Fh	00h
C3TECR	4794Fh	00h
C4TECR	4774Fh	00h
C5TECR	4754Fh	00h

Function	Counter Value	RW
Transmit error count function The CiTECR register increments or decrements the counter value according to the error status of the CAN module during transmission	00h to FFh (1)	RO

Note:

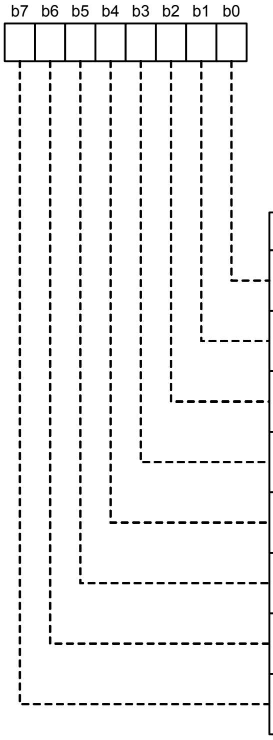
1. The value in the bus-off state is undefined.

Figure 25.27 C0TECR to C5TECR

The CiTECR register indicates the value of the TEC error counter.
Refer to the CAN Specification (ISO 11898-1) for the increment/decrement conditions of the transmit error counter.

25.1.23 CANi Error Code Store Register (CiECSR) (i = 0 to 5)

CANi Error Code Store Register (i = 0 to 5)



Symbol	Address	Reset Value
C0ECSR	47F50h	00h
C1ECSR	47D50h	00h
C2ECSR	47B50h	00h
C3ECSR	47950h	00h
C4ECSR	47750h	00h
C5ECSR	47550h	00h

Bit Symbol	Bit Name	Function	RW
SEF	Stuff Error Flag [1, 2]	0: No stuff error detected 1: Stuff error detected	RW
FEF	Form Error Flag [1, 2]	0: No form error detected 1: Form error detected	RW
AEF	ACK Error Flag [1, 2]	0: No ACK error detected 1: ACK error detected	RW
CEF	CRC Error Flag [1, 2]	0: No CRC error detected 1: CRC error detected	RW
BE1F	Bit Error (recessive) Flag [1, 2]	0: No bit error detected 1: Bit error (recessive) detected	RW
BE0F	Bit Error (dominant) Flag [1, 2]	0: No bit error detected 1: Bit error (dominant) detected	RW
ADEF	ACK Delimiter Error Bit [1, 2]	0: No ACK delimiter error detected 1: ACK delimiter error detected	RW
EDPM	Error Display Mode Select Bit [3]	0: Output of first detected error code [4] 1: Output of accumulated error code	RW

Notes:

1. Writing 1 has no effect on these bit values.
2. 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.
3. Write to the EDPM bit in CAN reset mode or CAN halt mode.
4. If more than one error condition is detected simultaneously, all corresponding bits are set to 1.

Figure 25.28 Registers C0ECSR to C5ECSR

The CiECSR register can be used to monitor whether an error has occurred on the CAN bus. Refer to the CAN Specification (ISO 11898-1) 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 is written by a program are the same, the relevant bit is set to 1.

25.1.23.1 SEF Bit

The SEF bit becomes 1 when a stuff error is detected.

25.1.23.2 FEF Bit

The FEF bit becomes 1 when a form error is detected.

25.1.23.3 AEF Bit

The AEF bit becomes 1 when an ACK error is detected.

25.1.23.4 CEF Bit

The CEF bit becomes 1 when a CRC error is detected.

25.1.23.5 BE1F Bit

The BE1F bit becomes 1 when a recessive bit error is detected.

25.1.23.6 BE0F Bit

The BE0F bit becomes 1 when a dominant bit error is detected.

25.1.23.7 ADEF Bit

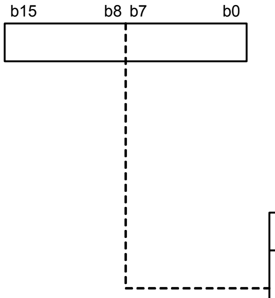
The ADEF bit becomes 1 when a form error is detected with the ACK delimiter during transmission.

25.1.23.8 EDPM Bit

The EDPM bit selects the output mode of the CiECSR register (i = 0 to 5).
When this bit is set to 0, the CiECSR register outputs the first error code.
When this bit is set to 1, the CiECSR register outputs the accumulated error code.

25.1.24 CANi Time Stamp Register (CiTSR) (i = 0 to 5)

CANi Time Stamp Register (i = 0 to 5) [1]



Symbol	Address	Reset Value
C0TSR	47F55h-47F54h	0000h
C1TSR	47D55h-47D54h	0000h
C2TSR	47B55h-47B54h	0000h
C3TSR	47955h-47954h	0000h
C4TSR	47755h-47754h	0000h
C5TSR	47555h-47554h	0000h

Function	Counter Value	RW
Free-running counter value for the time stamp function	0000h to FFFFh	RO

Note:

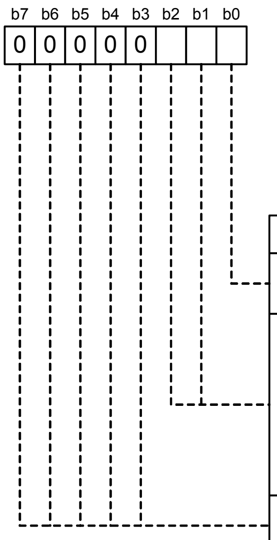
1. Read the CiTSR register in 16-bit units.

Figure 25.29 Registers C0TSR to C5TSR

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 (j = 0 to 15) when a received message is stored in a receive mailbox.

25.1.25 CANi Test Control Register (CiTCR) (i = 0 to 5)

CANi Test Control Register (i = 0 to 5) (1)



Symbol	Address	Reset Value
C0TCR	47F58h	00h
C1TCR	47D58h	00h
C2TCR	47B58h	00h
C3TCR	47958h	00h
C4TCR	47758h	00h
C5TCR	47558h	00h

Bit Symbol	Bit Name	Function	RW
TSTE	CAN Test Mode Enable Bit	0: CAN test mode disabled 1: CAN test mode enabled	RW
TSTM	CAN Test Mode Select Bit	b2 b1 0 0 : Other than CAN test mode 0 1 : Listen only mode 1 0 : Self test mode 0 (external loop back) 1 1 : Self test mode 1 (internal loop back)	RW
— (b7-b3)	Reserved	Should be written with 0	RW

Note:

1. Write to the CiTCR register in CAN halt mode.

Figure 25.30 Registers C0TCR to C5TCR

25.1.25.1 TSTE Bit

When the TSTE bit is set to 0, CAN test mode is disabled.
When this bit is set to 1, CAN test mode is enabled.

25.1.25.2 TSTM Bit

The TSTM bit selects the CAN test mode.
The details of each CAN test mode are described below.

25.1.25.3 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 25.31 shows the connection when listen only mode is selected.

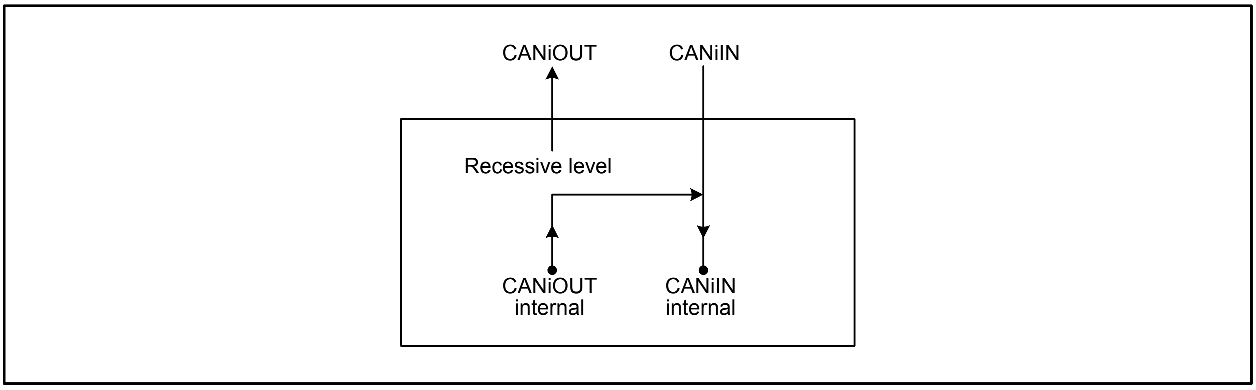


Figure 25.31 Connection when Listen Only Mode is Selected (i = 0 to 5)

25.1.25.4 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 a receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit.
Connect the CANiOUT/CANiIN pins (i = 0 to 5) to the transceiver.
Figure 25.32 shows the connection when self test mode 0 is selected.

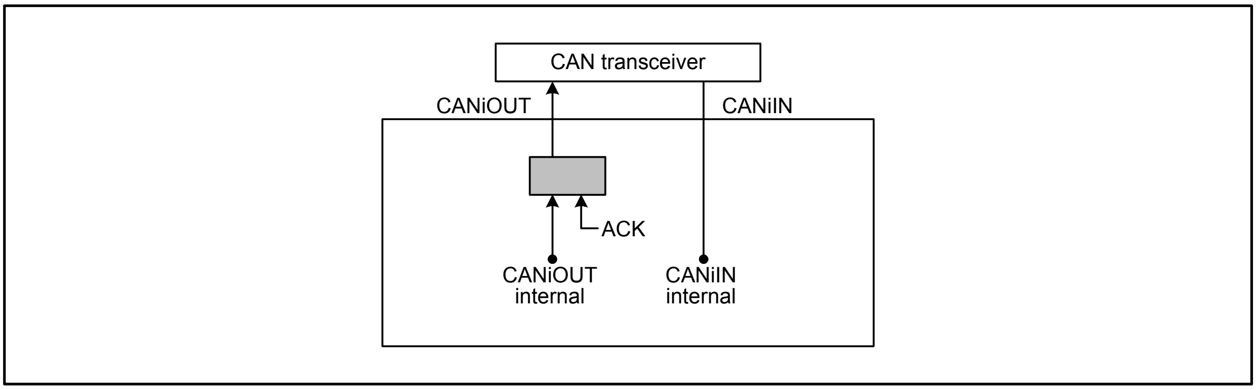


Figure 25.32 Connection when Self Test Mode 0 is Selected (i = 0 to 5)

25.1.25.5 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 a 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 CANiOUT pin (i = 0 to 5) to the internal CANiIN pin. The input value of the external CANiIN pin is ignored. The external CANiOUT pin outputs only recessive bits. The CANiOUT/CANiIN pins do not need to be connected to the CAN bus or any external device.

Figure 25.33 shows the connection when self test mode 1 is selected.

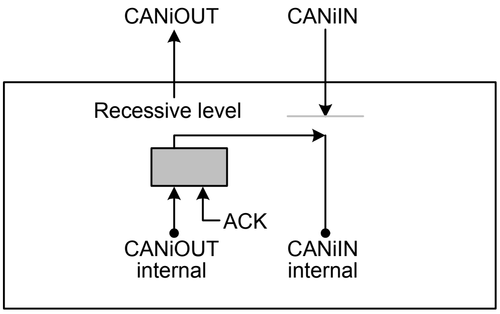


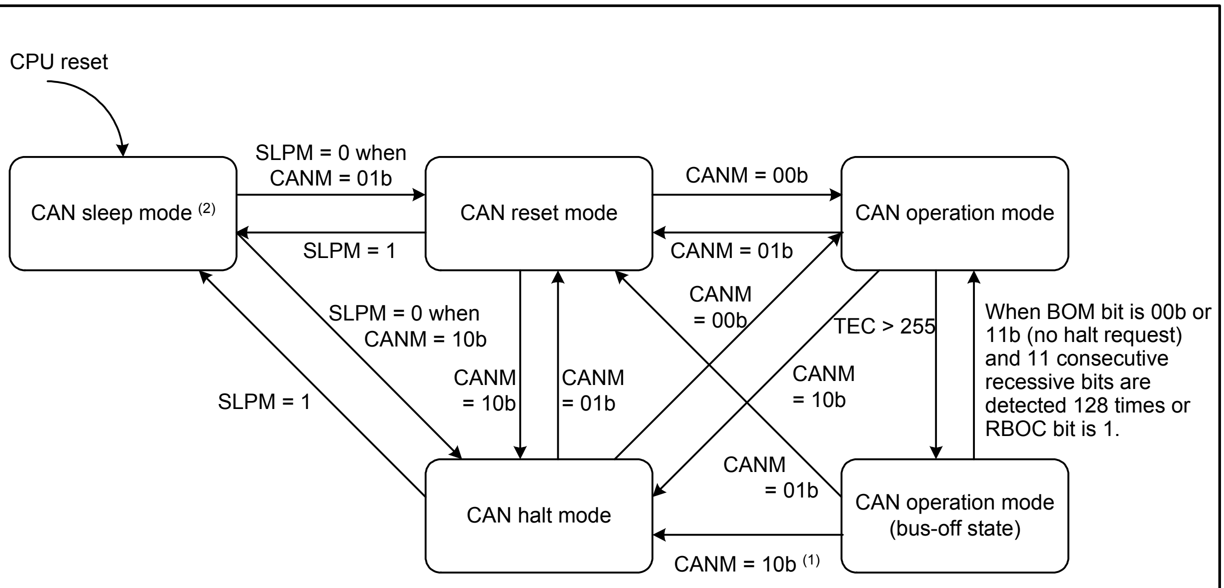
Figure 25.33 Connection when Self Test Mode 1 is Selected (i = 0 to 5)

25.2 Operating Modes

The CAN module has the following four operating modes:

- CAN reset mode
- CAN halt mode
- CAN operation mode
- CAN sleep mode

Figure 25.34 shows the transition between CAN operating modes.



CANM, SLPM, BOM, and 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 01b, the state transition timing is immediately after entering the bus-off state.
   - When the BOM bit is 10b, the state transition timing is at the end of the bus-off state.
   - When the BOM bit is 11b, the state transition timing is at the setting of the CANM bit to 10b (CAN halt mode).
2. Write only to the SLPM bit to exit/set CAN sleep mode.

Figure 25.34 Transition between CAN Operating Modes (i = 0 to 5)

25.2.1 CAN Reset Mode

CAN reset mode is provided for CAN communication configuration.
When the CANM bit in the CiCTLR register (i = 0 to 5) is set to 01b, the CAN module enters CAN reset. Then the RSTST bit in the CiSTR register becomes 1. Do not change the CANM bit until the RSTST bit becomes 1.
Configure the CiBCR register before exiting CAN reset mode and entering any other mode.

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 (j = 0 to 15)
- 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)

The previous values of the following registers are retained after entering CAN reset mode:

- CiCLKR register
- CiCTLR register
- CiSTR register (bits SLPST and TFST)
- CiMIER register
- CiEIER register
- CiBCR register
- CiCSSR register
- CiECSR register (EDPM bit only)
- CiMBj register
- Registers CiMKR0 to CiMKR3
- Registers CiFIDCR0 and CiFIDCR1
- CiMKIVLR register
- CiAFSR register
- CiRFPCR register
- CiTFPCR register

25.2.2 CAN Halt Mode

CAN halt mode is used for mailbox configuration and test mode setting.

When the CANM bit in the CiCTLR register (i = 0 to 5) is set to 10b, CAN halt mode is selected. Then the HLTST bit in the CiSTR register becomes 1. Do not change the CANM bit until the HLTST bit becomes 1.

Refer to Table 25.10 "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 with automatic bit rate detection.

Table 25.10 Operation in CAN Reset Mode and CAN Halt Mode

Mode	Receiver	Transmitter	Bus-off
CAN reset mode	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 00b A halt request from a program will be acknowledged only after bus-off recovery - When the BOM bit is 01b CAN module automatically enters 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 10b CAN module automatically enters 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 11b 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

BOM bit: Bit in the CiCTLR register (i = 0 to 5)

Notes:

1. If several messages are requested to be transmitted, mode transition occurs after the completion of the first message transmission. When 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.

25.2.3 CAN Sleep Mode

CAN sleep mode is used for reducing current consumption by stopping the clock supply to the CAN module. After a MCU reset, the CAN module starts from CAN sleep mode.

When the SLPM bit in the CiCTLR register (i = 0 to 5) is set to 1, the CAN module enters CAN sleep mode. Then the SLPST bit in the CiSTR register becomes 1. Do not change the value of the SLPM bit until the SLPST bit becomes 1. 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 other registers (except the SLPM bit) during CAN sleep mode. Read operations are 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.

25.2.4 CAN Operation Mode (Excluding Bus-off State)

CAN operation mode is used for CAN communication.
When the CANM bit in the CiCTLR register (i = 0 to 5) is set to 00b, the CAN module enters CAN operation mode.
Then bits RSTST and HLTST in the CiSTR register become 0. Do not change the value of the CANM bit until these bits become 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 submodes, 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 bits in the CiTCR register are 10b) or self test mode 1 (TSTM bits are 11b) is selected.

Figure 25.35 shows the submode in CAN operation mode.

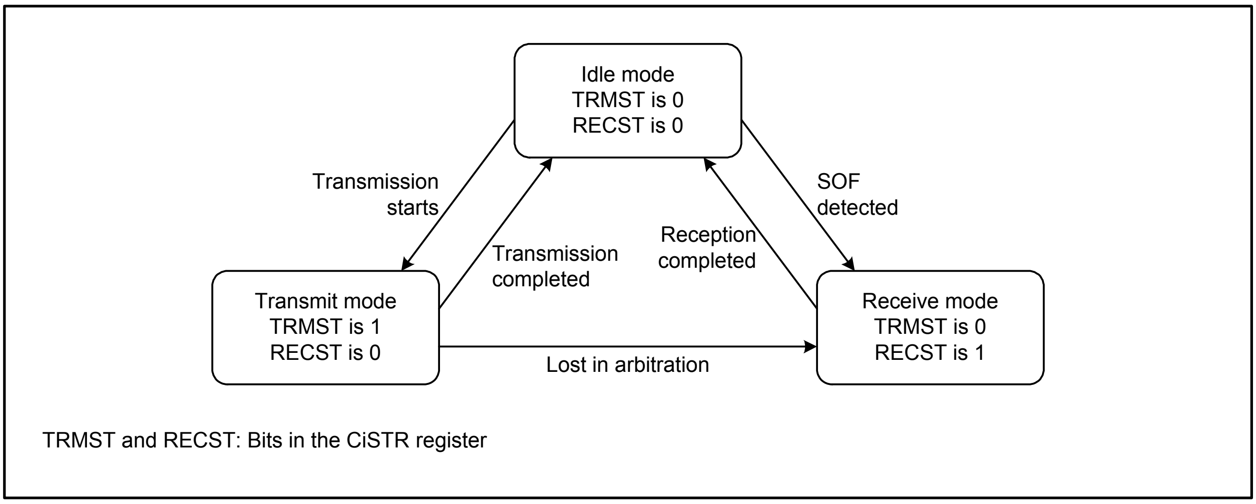


Figure 25.35 Submode in CAN Operation Mode (i = 0 to 5)

25.2.5 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 the bus-off state, the values of the associated registers, except registers CiSTR, CiEIFR, CiRECR, CiTECR, and CiTSR (i = 0 to 5), remain unchanged.

(1) When the BOM bit in the CiCTLR register is 00b (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. The BORIF bit in the CiEIFR register becomes 1 (bus-off recovery detected) at this time.

(2) When the RBOC bit in the CiCTLR register is set to 1 (forced recovery from bus-off)

The CAN module enters the error-active state when it is in the 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 does not become 1 at this time.

(3) When the BOM bit is 01b (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 does not become 1 at this time.

(4) When the BOM bit is 10b (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 becomes 1 at this time.

(5) When the BOM bit is 11b (entry to CAN halt mode by a program) and the CANM bit in the CiCTLR register is set to 10b (CAN halt mode) during the bus-off state

The CAN module enters CAN halt mode when it is in the bus-off state and the CANM bit is set to 10b (CAN halt mode). The BORIF bit does not become 1 at this time.
If the CANM bit is not set to 10b during bus-off, the same behavior as (1) applies.

25.3 CAN Communication Speed Configuration

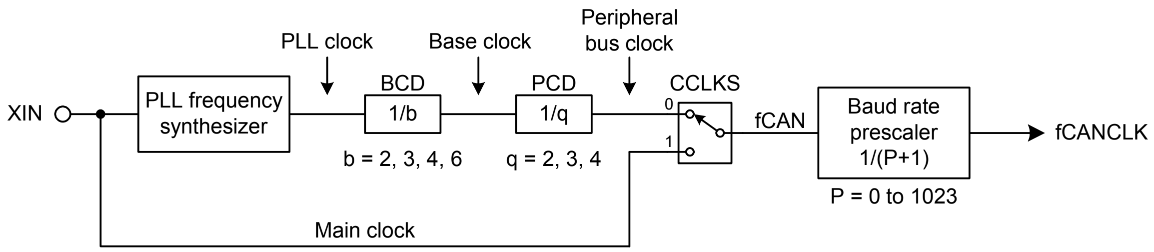
The following description explains about the CAN communication speed configuration.

25.3.1 CAN Clock Configuration

This group has a CAN clock selector.

The CAN clock can be configured by setting the CCLKS bit in the CiCLKR register (i = 0 to 5) and the BRP bit in the CiBCR register.

Figure 25.36 shows the block diagram of CAN clock generator.



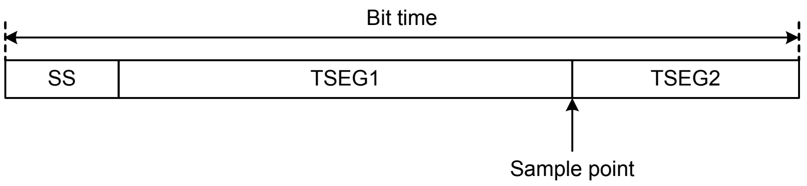
BCD and PCD: Bits in the CCR register
CCLKS: Bit in the CiCLKR register
fCAN: CAN system clock
P: Setting value of the BRP bit in the CiBCR register, P = 0 to 1023
fCANCLK: CAN communication clock, fCANCLK = fCAN/(P+1)

Figure 25.36 Block Diagram of the CAN Clock Generator (i = 0 to 5)

25.3.2 Bit Timing Configuration

The bit time is a single bit time for transmitting/receiving a message and consists of the three segments in the figure below.

Figure 25.37 shows the bit timing.



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 25.37 Bit Timing

25.3.3 Bit rate

The bit rate depends on the CAN clock (fCAN), the divisor of the baud rate prescaler, and the number of Tq of 1 bit time.

$$\text{Bit rate[bps]} = \frac{fCAN}{\text{Baud rate prescaler division value}^{(1)} \times \text{number of Tq of 1 bit time}} = \frac{fCANCLK}{\text{Number of Tq of 1 bit time}}$$

Note:

1. Divisor 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 5)

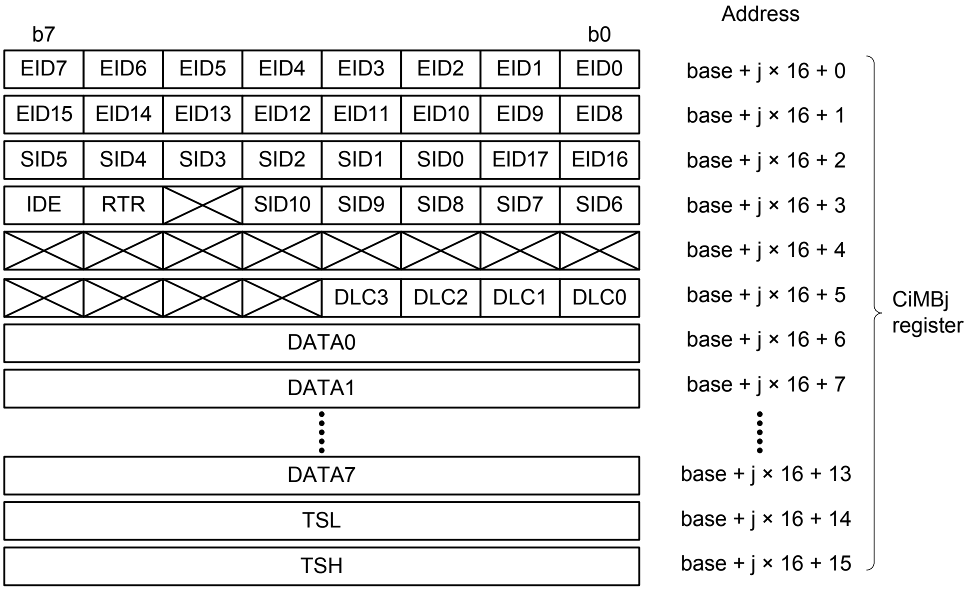
Table 25.11 lists bit rate examples.

Table 25.11 Bit Rate Examples

fCAN	32 MHz		24 MHz		20 MHz		16 MHz		8 MHz	
Bit Rate	No. of Tq	P+1	No. of Tq	P+1	No. of Tq	P+1	No. of Tq	P+1	No. of Tq	P+1
1 Mbps	8 Tq 16 Tq	4 2	8 Tq	3	10 Tq 20 Tq	2 1	8 Tq 16 Tq	2 1	8 Tq	1
500 kbps	8 Tq 16 Tq	8 4	8 Tq 16 Tq	6 3	10 Tq 20 Tq	4 2	8 Tq 16 Tq	4 2	8 Tq 16 Tq	2 1
250 kbps	8 Tq 16 Tq	16 8	8 Tq 16 Tq	12 6	10 Tq 20 Tq	8 4	8 Tq 16 Tq	8 4	8 Tq 16 Tq	4 2
83.3 kbps	8 Tq 16 Tq	48 24	8 Tq 16 Tq	36 18	8 Tq 10 Tq 16 Tq 20 Tq	30 24 15 12	8 Tq 16 Tq	24 12	8 Tq 16 Tq	12 6
33.3 kbps	8 Tq 10 Tq 16 Tq 20 Tq	120 96 60 48	8 Tq 10 Tq 16 Tq 20 Tq	90 72 45 36	8 Tq 10 Tq 20 Tq	75 60 30	8 Tq 10 Tq 16 Tq 20 Tq	60 48 30 24	8 Tq 10 Tq 16 Tq 20 Tq	30 24 15 12

25.4 Mailbox and Mask Register Structure

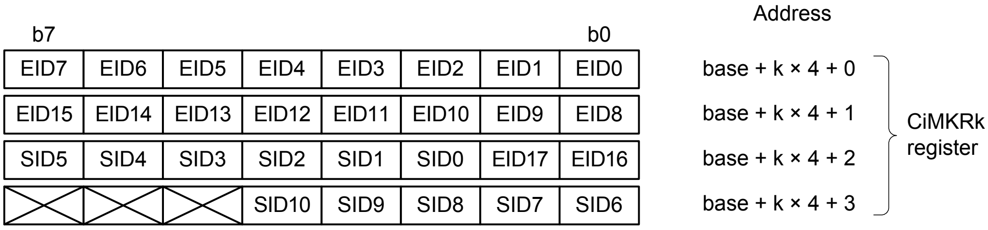
There are 16 mailboxes with the same structure.
Figure 25.38 shows the structure of registers C0MBj to C5MBj (j = 0 to 15).



base: CAN0 is 47E00h, CAN1 is 47C00h, CAN2 is 47A00h, CAN3 is 47800h, CAN4 is 47600h, and CAN5 is 47400h

Figure 25.38 Structure of Registers C0MBj to C5MBj (i = 0 to 5; j = 0 to 15)

There are four mask registers with the same structure.
Figure 25.39 shows the structure of registers C0MKRk to C5MKRk (k = 0 to 3).



base: CAN0 is 47F10h, CAN1 is 47D10h, CAN2 is 47B10h, CAN3 is 47910h, CAN4 is 47710h, and CAN5 is 47510h

Figure 25.39 Structure of Registers C0MKRk to C5MKRk (i = 0 to 5; k = 0 to 3)

There are two FIFO received ID compare registers with the same structure.
Figure 25.40 shows the structure of registers C0FIDCRn to C5FIDCRn (n = 0, 1).

b7							b0	Address
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	base + n × 4 + 0
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	base + n × 4 + 1
SID5	SID4	SID3	SID2	SID1	SID0	EID17	EID16	base + n × 4 + 2
IDE	RTR		SID10	SID9	SID8	SID7	SID6	base + n × 4 + 3

CiFIDCRn register

base: CAN0 is 47F20h, CAN1 is 47D20h, CAN2 is 47B20h, CAN3 is 47920h, CAN4 is 47720h, and CAN5 is 47520h

Figure 25.40 Structure of Registers C0FIDCRn to C5FIDCRn (i = 0 to 5; n = 0, 1)

25.5 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 CiMKR3 (i = 0 to 5) can perform masking of the standard ID and the extended ID of 29 bits.

- The CiMKR0 register corresponds to mailboxes [0] to [3].
- The CiMKR1 register corresponds to mailboxes [4] to [7].
- The CiMKR2 register corresponds to mailboxes [8] to [11] in normal mailbox mode, and mailboxes [12] to [15] in receive FIFO of FIFO mailbox mode.
- The CiMKR3 register corresponds to mailboxes [12] to [15] in normal mailbox mode and receive FIFO of FIFO mailbox mode.

The CiMKIVLR register disables acceptance filtering individually for each mailbox.

The IDE bit in the CiMBj register (j = 0 to 15) is enabled when the IDFM bit in the CiCTLR register is 10b (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 [7]) use the single corresponding register among registers CiMKR0 to CiMKR1 for acceptance filtering. Receive FIFO mailboxes (mailboxes [12] to [15]) use two registers CiMKR2 and CiMKR3 for acceptance filtering.
Also, the receive FIFO uses registers CiFIDCR0 and CiFIDCR1 for ID comparison. Bits EID, SID, RTR, and IDE in registers CiMB12 to CiMB15 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.
The CiMKIVLR register is disabled for the receive FIFO.

If both the settings for 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 the data and remote frames are received.
When a combination of 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 25.41 shows the mask registers and their corresponding mailboxes, and Figure 25.42 shows acceptance filtering.

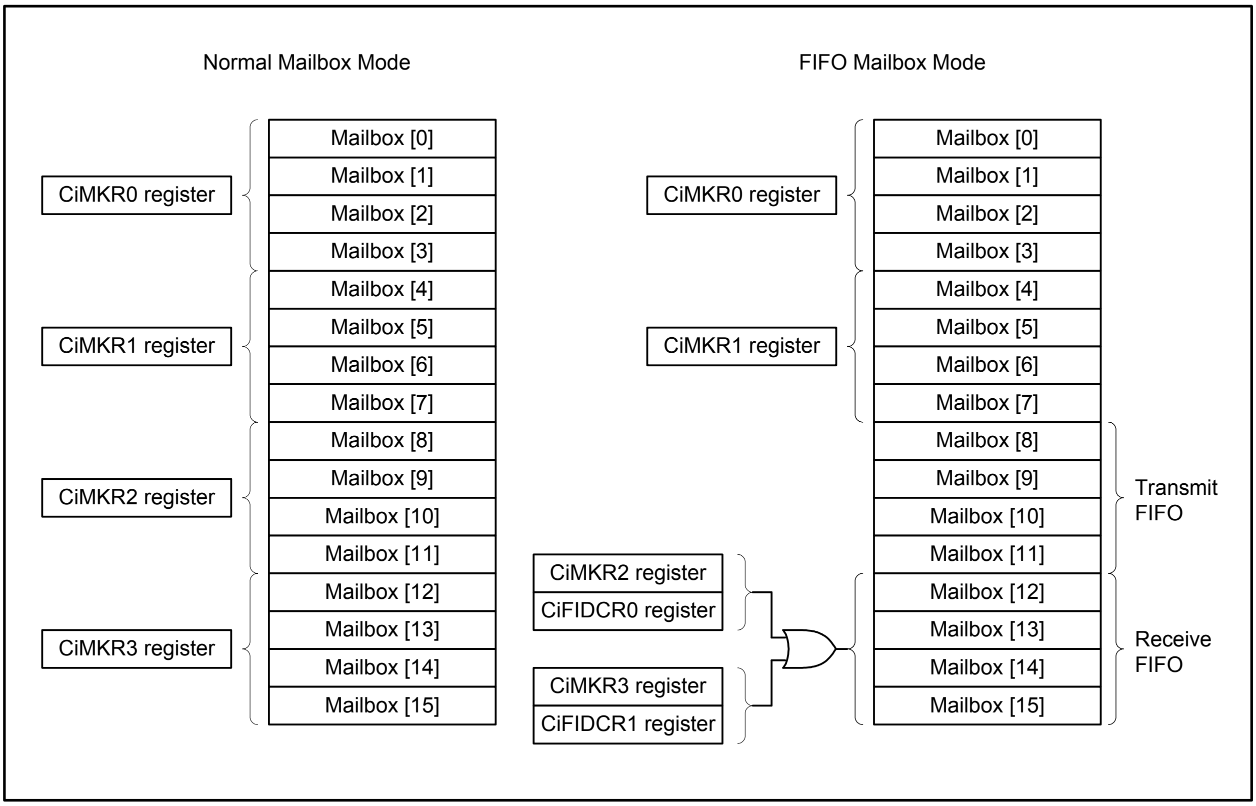
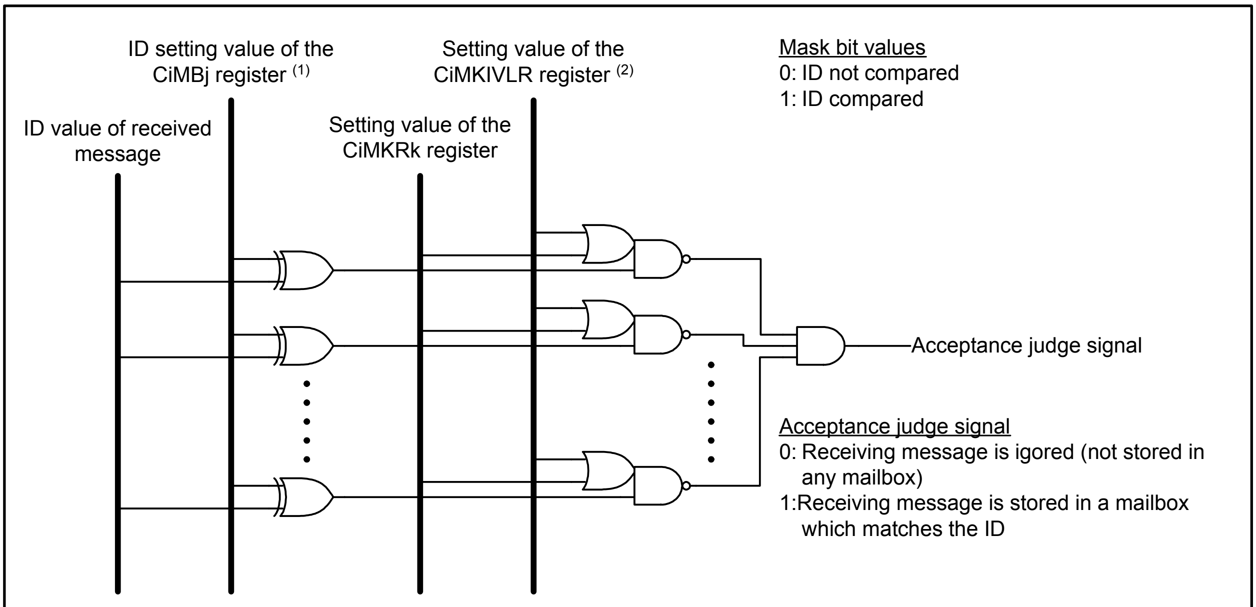


Figure 25.41 Mask Registers and Their Corresponding Mailboxes (i = 0 to 5)



Notes:

1. The values set in registers CiFIDCR0 and CiFIDCR1 are used in FIFO mailbox mode.
2. Invalid in FIFO mailboxes.

Figure 25.42 Acceptance Filtering (i = 0 to 5; j = 0 to 15; k = 0 to 3)

25.6 Reception and Transmission

Table 25.12 lists the CAN communication mode configuration.

Table 25.12 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

TRMREQ, RECREQ, and ONESHOT: Bits in the CiMCTLj register (i = 0 to 5; j = 0 to 15)

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 (i = 0 to 5; j = 0 to 15) to 00h.

(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 which mailbox 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.

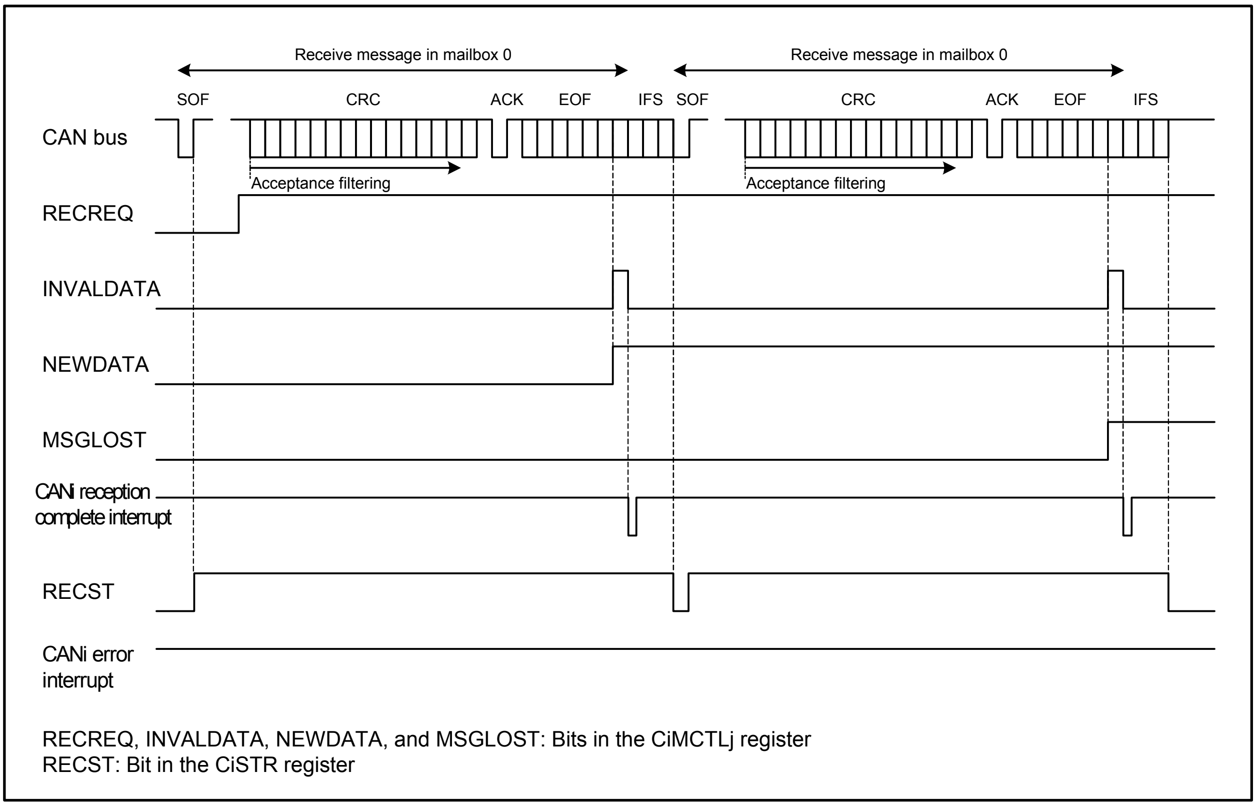
When a mailbox is configured as a transmit mailbox or a one-shot transmit mailbox, note the following:

(1) Before a mailbox is configured as a transmit mailbox or one-shot transmit mailbox, ensure that the CiMCTLj register is 00h and that there is no pending abort process.

25.6.1 Reception

Figure 25.43 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 match the receiving conditions of the CiMCTL0 register (i = 0 to 5).



RECREQ, INVALDATA, NEWDATA, and MSGLOST: Bits in the CiMCTLj register
RECST: Bit in the CiSTR register

Figure 25.43 Operation Example of Data Frame Reception in Overwrite Mode (i = 0 to 5; j = 0 to 15)

(1) When an SOF is detected on the CAN bus, the RECST bit in the CiSTR register becomes 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 (j = 0 to 15) for the receive mailbox becomes 1 (new data being updated/stored in the mailbox). Simultaneously, the INVALDATA bit in the CiMCTLj register becomes 1 (message is being updated), and then the INVALDATA bit becomes 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 becomes 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 becomes 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 25.44 shows an operation example of data frame reception in overrun mode.
This example shows the operation of overrunning the second message when the CAN module receives two consecutive CAN messages that match the receiving conditions of the CiMCTL0 register (i = 0 to 5).

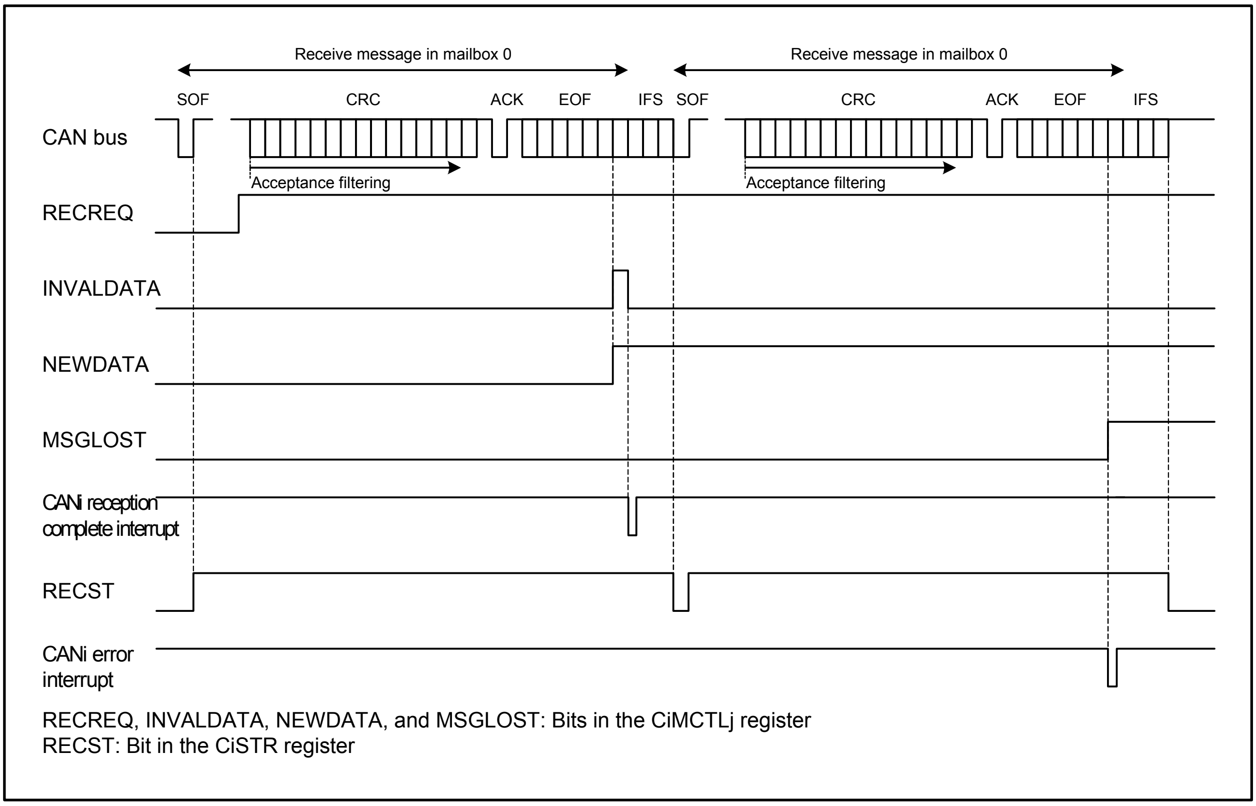


Figure 25.44 Operation Example of Data Frame Reception in Overrun Mode (i = 0 to 5; j = 0 to 15)

(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 (j = 0 to 15) becomes 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 1 (interrupt enabled).

25.6.2 Transmission

Figure 25.45 shows an operation example of data frame transmission. This example shows an operation of transmitting messages that have been set in registers CiMCTL0 and CiMCTL1 (i = 0 to 5).

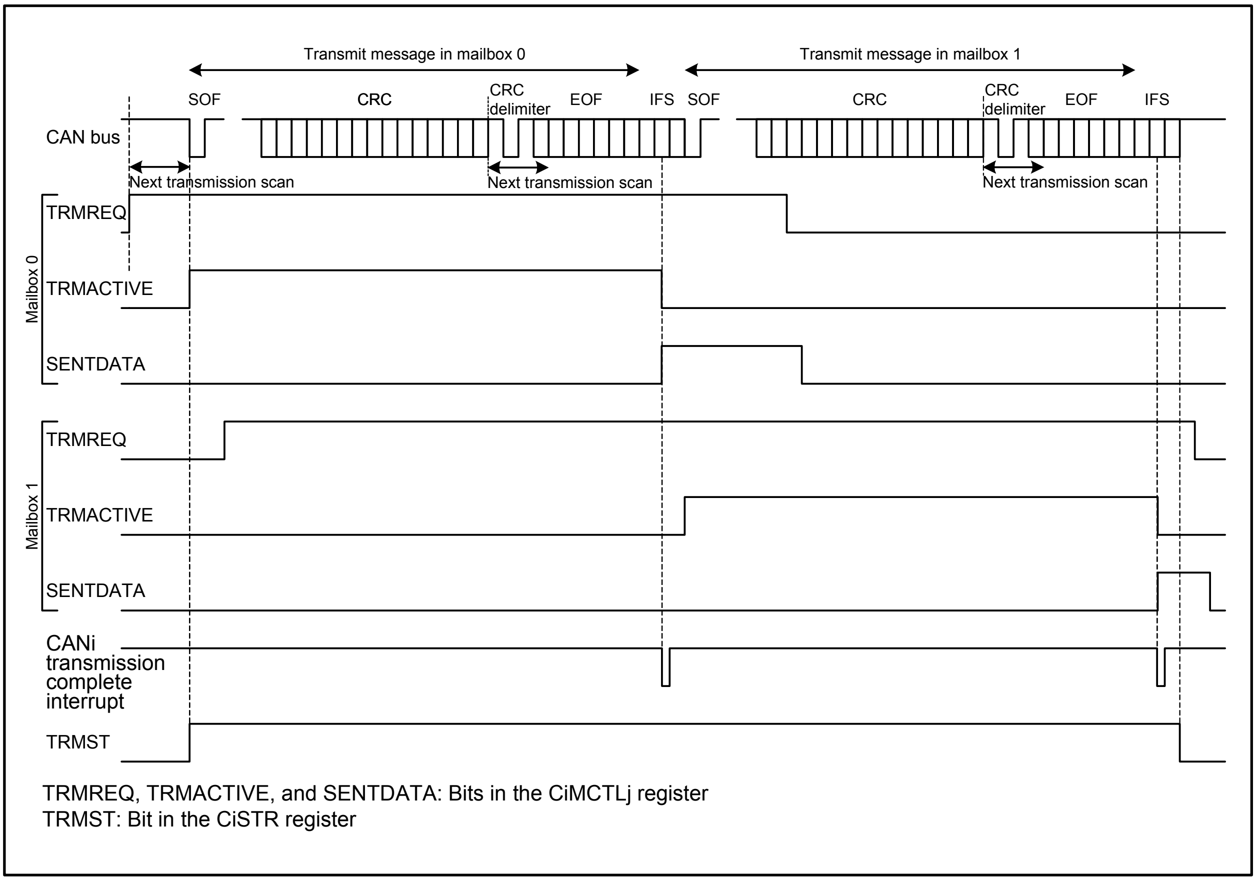


Figure 25.45 Operation Example of Data Frame Transmission (i = 0 to 5; j = 0 to 15)

(1) When the TRMREQ bit in the CiMCTLj register (j = 0 to 15) is set to 1 (transmit mailbox) in the 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 becomes 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 becomes 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 becomes 1 (transmission completed) and the TRMACTIVE bit becomes 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 becomes 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 loss of arbitration, the transmission scan procedure is performed again from the start of the error delimiter to search for the highest-priority transmit mailbox.

25.7 CAN Interrupts

The CAN module provides the following CAN interrupts:

- CANi wakeup interrupt (i = 0 to 5)
- CANi reception complete interrupt
- CANi transmission complete interrupt
- CANi receive FIFO interrupt
- CANi transmit FIFO interrupt
- CANi error interrupt

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

26. CAN Gateway Module

The CAN gateway module relays frames between the CAN modules. Frames received from the CAN are stored into the transmit FIFO corresponding to each CAN module according to the routing table.

The R32C/142 Group has three channels (CAN2, CAN3, and CAN5), and the R32C/145 Group has six channels (CAN0 to CAN5) of the CAN module.

In this chapter, variables i and j indicate a channel number and transmit FIFO number, respectively. The ranges of the variables are as follows:

- R32C/142 Group: i = 2, 3, 5; j = 0, 1
- R32C/145 Group: i = 0 to 5; j = 0, 1

Table 26.1 lists the CAN gateway module specifications, and Figure 26.1 and Figure 26.2 show the CAN gateway module block diagrams.

Table 26.1 CAN Gateway Module Specifications

Item	Specification
Modules	CAN2, CAN3, and CAN5 (R32C/142 Group) CAN0 to CAN5 (R32C/145 Group)
Number of CAN channels for reception	1 to 3 channels (R32C/142 Group) 1 to 6 channels (R32C/145 Group)
Number of CAN destination channels for transmission	1 to 3 channels (R32C/142 Group) 1 to 6 channels (R32C/145 Group)
Message ID	Standard ID or extended ID
Routing table	384 entries The masking function can group multiple IDs together with one entry
Transmit FIFO	Each CAN channel has two rows of 32 blocks
Time stamp function	This function adds the counter value of a dedicated free-running 16-bit counter to a message frame
Error detection	• Routing table checksum calculation function • Transmit FIFO check function • Routing table parity check function • Transmit FIFO parity check function
Interrupt sources	• Transmit FIFO critical level interrupt • Transmit FIFO overflow interrupt • Routing error interrupt • Hardware error interrupt

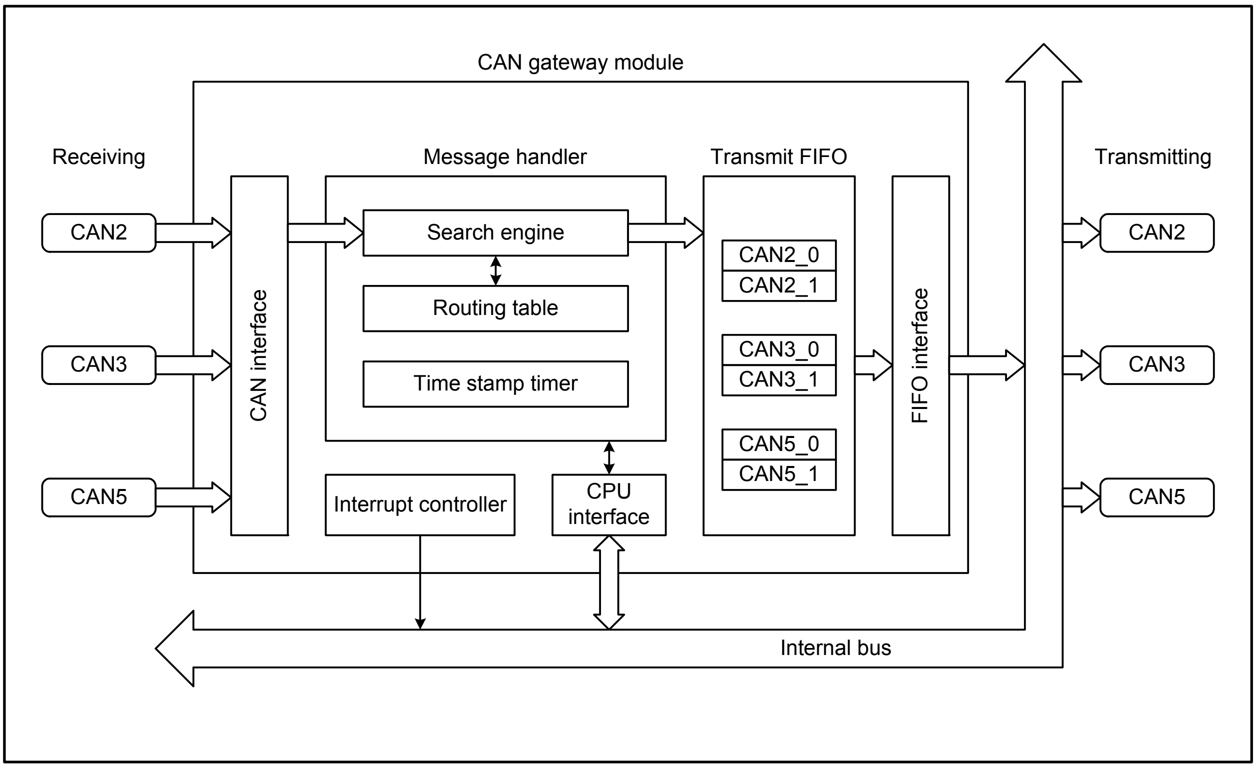


Figure 26.1 CAN Gateway Module Block Diagram for the R32C/142 Group

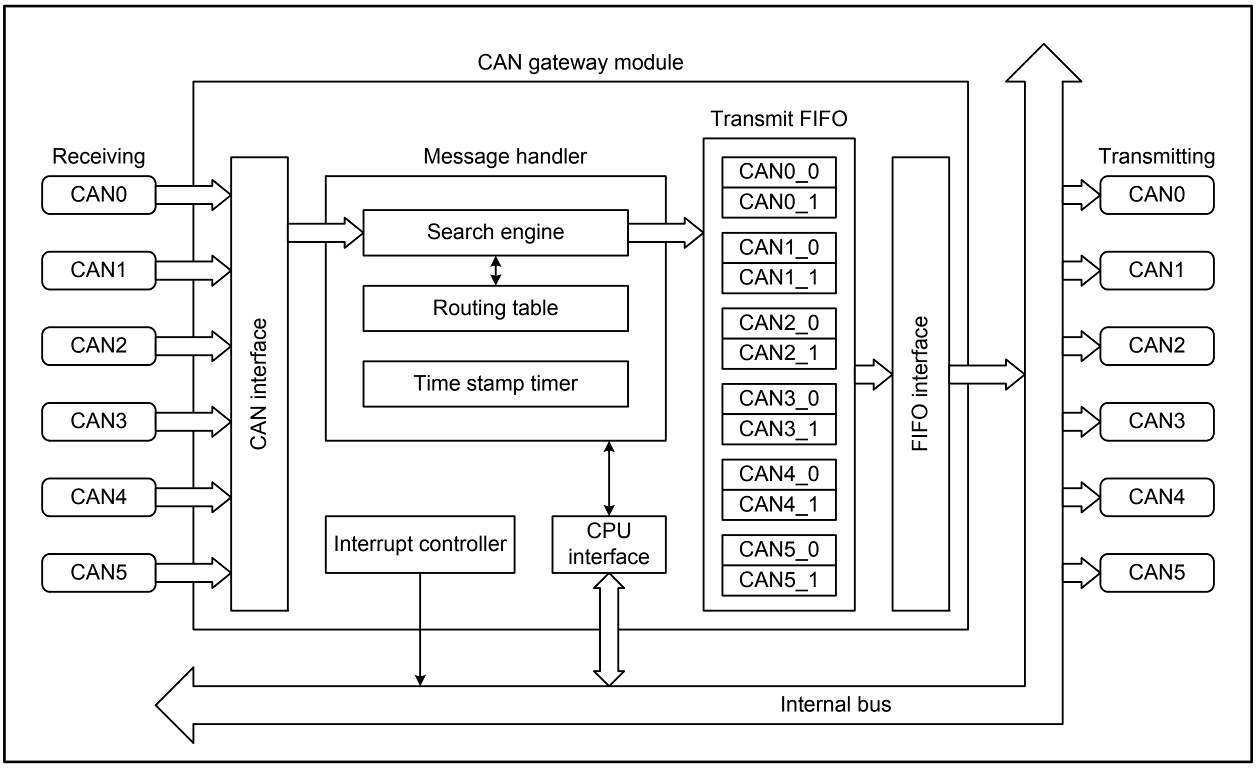
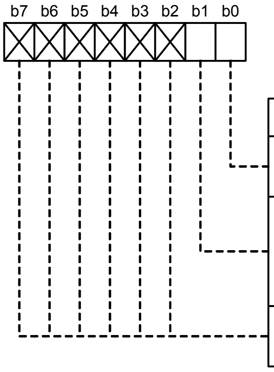


Figure 26.2 CAN Gateway Module Block Diagram for the R32C/145 Group

26.1 Registers Associated with the CAN Gateway Module

26.1.1 Gateway Mode Register (GMR)

Gateway Mode Register



Symbol	Address	Reset Value
GMR	47200h	0000 0000b

Bit Symbol	Bit Name	Function	RW
OM	Operating Mode Select Bit (1)	0: Configuration mode 1: Routing mode	RW
RST	Reset Bit (2)	Set this bit to 1 to start resetting the CAN gateway module. It automatically becomes 0 when the reset is completed	RW
— (b7-b2)	No register bits; should be written with 0 and read as 0		—

Notes:
1. This bit cannot be set to 1 during the checksum calculation and parity check. Verify that the GCSR register is 00h to see if the MCU has entered configuration mode after setting this bit to 0.
2. This bit can be read as 1 while being reset; otherwise, it is read as 0.

Figure 26.3 GMR Register

This register controls the transition between the following three CAN gateway module modes:

- Gateway reset mode
  The gateway registers are initialized in this mode.
- Configuration mode
  The gateway module is configured in this mode.
- Routing mode
  The gateway module operates in this mode.

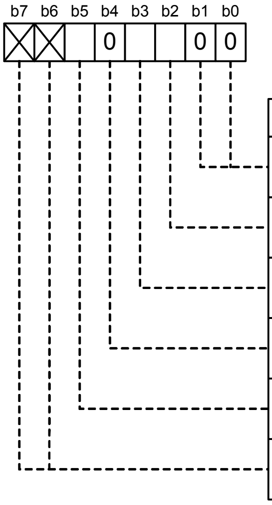
Table 26.2 Operations in Each Mode

	Gateway Reset Mode	Configuration Mode	Routing Mode
Checksum calculation on routing table	Not available	Available	Not available
Transmit FIFO check	Not available	Available	Not available
Data setting for routing table	Initialized (setting disabled)	Setting enabled	Setting disabled
Routing	Not available	Not available	Available
Processing for a frame being routed (1)	Discarding the frame	Transferring to FIFO	Transferring to FIFO

Note:
1. This is the processing for a frame being routed when entering any mode from routing mode.

26.1.2 Gateway Channel Control Register (GCCR)

Gateway Channel Control Register



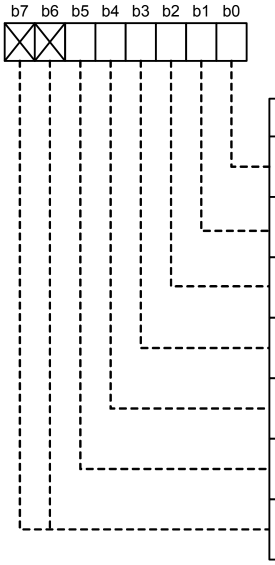
Symbol	Address	Reset Value
GCCR	4720Ch	0000 0000b

Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved	Should be written with 0	RW
CH2E	Channel 2 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
CH3E	Channel 3 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
— (b4)	Reserved	Should be written with 0	RW
CH5E	Channel 5 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
— (b7-b6)	No register bits; should be written with 0 and read as 0		—

Notes:
1. The data being routed are transferred to the transmit FIFO when this bit is set to 0 from 1 during routing.
2. After setting this register, read the CHiS bit in the GCSR register to verify that the mode transition is completed. Routing is enabled when the module is in routing mode and the corresponding CHiS bit is 1.

Figure 26.4 GCCR Register in the R32C/142 Group

Gateway Channel Control Register



Symbol	Address	Reset Value
GCCR	4720Ch	0000 0000b

Bit Symbol	Bit Name	Function	RW
CH0E	Channel 0 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
CH1E	Channel 1 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
CH2E	Channel 2 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
CH3E	Channel 3 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
CH4E	Channel 4 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
CH5E	Channel 5 Routing Enable Bit (1, 2)	0: Routing disabled 1: Routing enabled	RW
— (b7-b6)	No register bits; should be written with 0 and read as 0		—

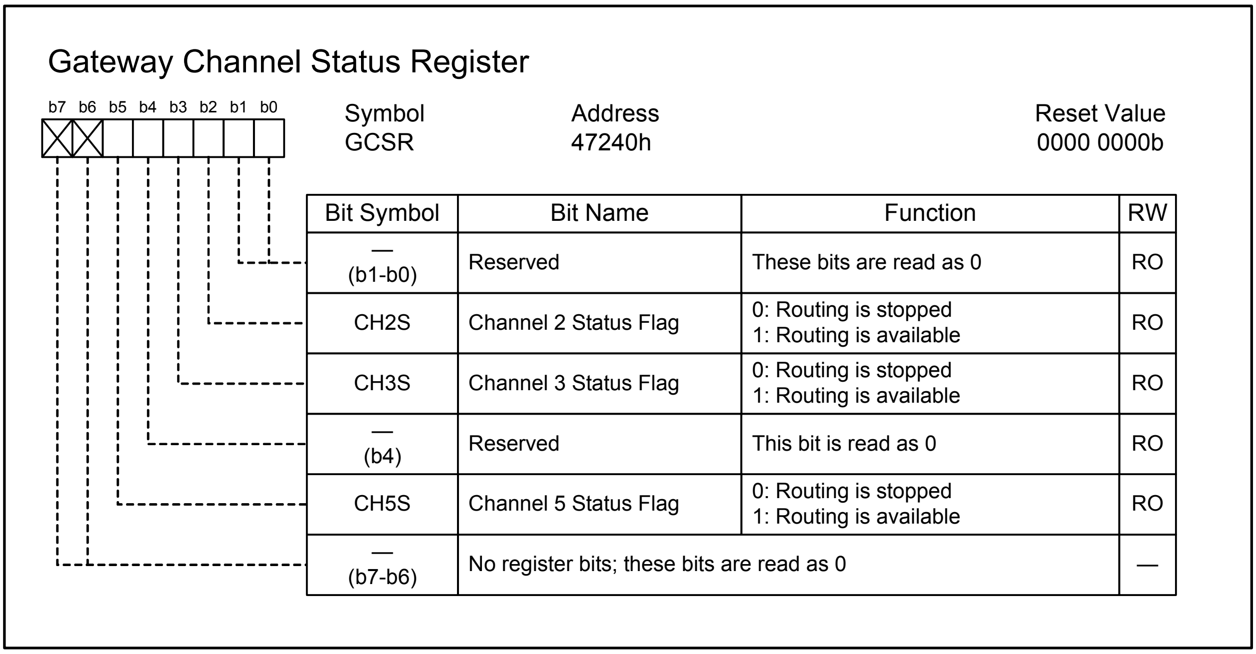
Notes:

1. The data being routed are transferred to the transmit FIFO when this bit is set to 0 from 1 during routing.
2. After setting this register, read the CHiS bit in the GCSR register to verify that the mode transition is completed. Routing is enabled when the module is in routing mode and the corresponding CHiS bit is 1.

Figure 26.5 GCCR Register in the R32C/145 Group

This register enables/disables routing of the message handler for each channel.

26.1.3 Gateway Channel Status Register (GCSR)



Gateway Channel Status Register

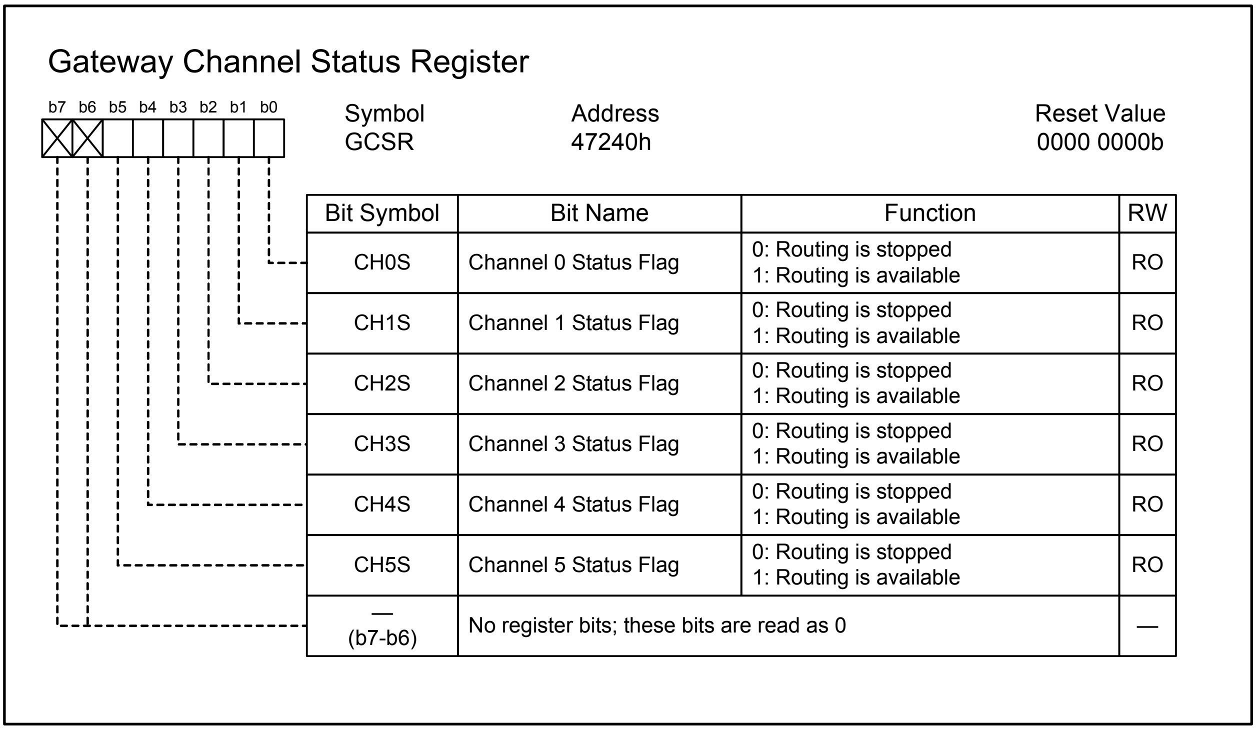
Symbol GCSR

Address 47240h

Reset Value 0000 0000b

Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved	These bits are read as 0	RO
CH2S	Channel 2 Status Flag	0: Routing is stopped 1: Routing is available	RO
CH3S	Channel 3 Status Flag	0: Routing is stopped 1: Routing is available	RO
— (b4)	Reserved	This bit is read as 0	RO
CH5S	Channel 5 Status Flag	0: Routing is stopped 1: Routing is available	RO
— (b7-b6)	No register bits; these bits are read as 0		—

Figure 26.6 GCSR Register in the R32C/142 Group



Gateway Channel Status Register

Symbol GCSR

Address 47240h

Reset Value 0000 0000b

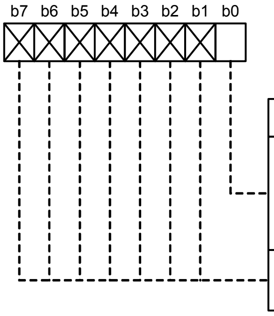
Bit Symbol	Bit Name	Function	RW
CH0S	Channel 0 Status Flag	0: Routing is stopped 1: Routing is available	RO
CH1S	Channel 1 Status Flag	0: Routing is stopped 1: Routing is available	RO
CH2S	Channel 2 Status Flag	0: Routing is stopped 1: Routing is available	RO
CH3S	Channel 3 Status Flag	0: Routing is stopped 1: Routing is available	RO
CH4S	Channel 4 Status Flag	0: Routing is stopped 1: Routing is available	RO
CH5S	Channel 5 Status Flag	0: Routing is stopped 1: Routing is available	RO
— (b7-b6)	No register bits; these bits are read as 0		—

Figure 26.7 GCSR Register in the R32C/145 Group

This register indicates the status of the routing for each channel in the CAN gateway module is either stopped or available.

26.1.4 Gateway Routing Table Checksum Control Register (GRMCC)

Gateway Routing Table Checksum Control Register



Symbol	Address	Reset Value
GRMCC	47204h	0000 0000b

Bit Symbol	Bit Name	Function	RW
CALC	Routing Table Start Calculating Checksum Bit	Set this bit to 1 to start the checksum calculation. It automatically becomes 0 when the calculation is completed (1, 2)	RW
— (b7-b1)	No register bits; should be written with 0 and read as 0		—

Notes:

1. Set this bit to 1 only in configuration mode.
2. Set this bit to 1 when the GCSR register is 00h.

Figure 26.8 GRMCC Register

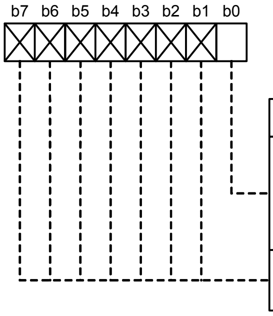
This register starts the checksum calculation on the routing table.

This function is for checking whether the routing table memory is damaged. The calculation starts by setting the CALC bit to 1. Read bits SCS1 and SCS0 in the GSCFC register to verify that the calculation is completed. The calculation result is reflected to the GRMSR register.

When the RMPCE bit in the GPCCR register is 1 (parity check enabled), the parity check for the routing table is also performed. The result of the parity check is reflected to the RMPER bit in the GRESR register.

26.1.5 Gateway Transmit FIFO Check Control Register (GTFCC)

Gateway Transmit FIFO Check Control Register



Symbol	Address	Reset Value
GTFCC	47205h	0000 0000b

Bit Symbol	Bit Name	Function	RW
CHK	Transmit FIFO Check Start Bit (1)	Set this bit to 1 to start the transmit FIFO check. It automatically becomes 0 when the check is completed (1, 2)	RW
— (b7-b1)	No register bits; should be written with 0 and read as 0		—

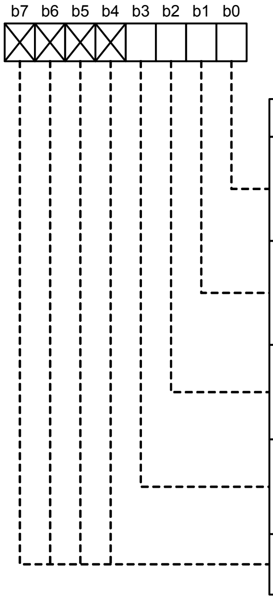
Notes:
1. Set this bit to 1 only in configuration mode.
2. Set this bit to 1 when the GCSR register is 00h.

Figure 26.9 GTFCC Register

This register starts a check on the transmit FIFO.
This function is for checking whether the transmit FIFO memory is damaged. The memory check starts by setting the CHK bit to 1. The result is reflected to bits FCS1 and FCS0 in the GSCFC register.
Messages in the transmit FIFO are deleted once the check is performed. Therefore, read the important messages left in the transmit FIFO before the check starts.

26.1.6 Gateway Checksum Calculation/FIFO Check Status Register (GSCFC)

Gateway Checksum Calculation/FIFO Check Status Register



Symbol	Address	Reset Value
GSCFC	47241h	0000 0000b

Bit Symbol	Bit Name	Function	RW
SCS0	Routing Table Checksum Calculation Status Flag	b1 b0 0 0 : Checksum calculation is not yet performed (1) 0 1 : Checksum calculation is in progress (1) 1 0 : Checksum calculation is completed (2) 1 1 : Reserved	RO
SCS1			RO
FCS0	Transmit FIFO Check Status Flag	b3 b2 0 0 : FIFO check is not yet performed 0 1 : FIFO check is in progress 1 0 : FIFO check is completed successfully 1 1 : FIFO check is completed in error	RO
FCS1			RO
— (b7-b4)	No register bits; these bits are read as 0		—

Notes:

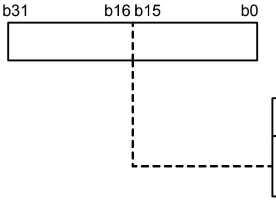
1. The value of the GRMSR register is undefined when these bits are 00b or 01b.
2. The value of the GRMSR register is the result of the checksum calculation when these bits are 10b.

Figure 26.10 GSCFC Register

This register indicates the status of the checksum calculation and the transmit FIFO check.

26.1.7 Gateway Routing Table Checksum Register (GRMSR)

Gateway Routing Table Checksum Register



Symbol	Address	Reset Value
GRMSR	47247h-47244h	0000 0000h

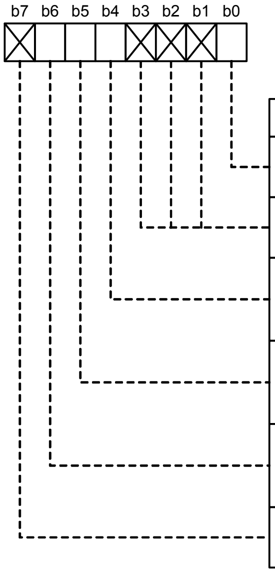
Function	RW
The result of the checksum calculation on the routing table is stored in this register	RO

Figure 26.11 GRMSR Register

This register stores the result of the checksum calculation on the routing table.

26.1.8 Gateway Transmit FIFO Read Control Register (GTFRC)

Gateway Transmit FIFO Read Control Register (1)



Symbol GTFRC

Address 47220h

Reset Value 0000 0000b

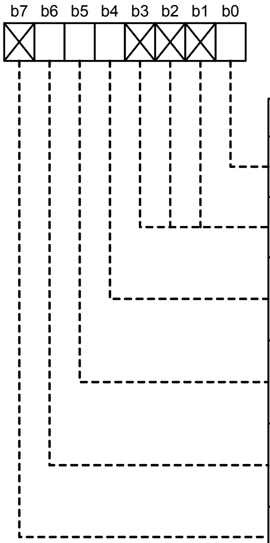
Bit Symbol	Bit Name	Function	RW
FIFO	FIFO Select Bit (2)	0: FIFO0 1: FIFO1	RW
— (b3-b1)	No register bits; should be written with 0 and read as 0		—
CH0	Channel Select Bit (2)	b6 b5 b4 0 0 0 : Do not use this combination 0 0 1 : Do not use this combination 0 1 0 : Channel 2 0 1 1 : Channel 3 1 0 0 : Do not use this combination 1 0 1 : Channel 5 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
CH1			
CH2			
— (b7)	No register bit; should be written with 0 and read as 0		—

Notes:

1. A write operation to this register is ignored during the transmit FIFO check.
2. Data transfer is not performed from the transmit FIFO to registers GFRR0 to GFRR3 when there is no data in the specified FIFO (when the GFijFL register is 0).

Figure 26.12 GTFRC Register in the R32C/142 Group

Gateway Transmit FIFO Read Control Register (1)



Symbol GTFRC

Address 47220h

Reset Value 0000 0000b

Bit Symbol	Bit Name	Function	RW
FIFO	FIFO Select Bit (2)	0: FIFO0 1: FIFO1	RW
— (b3-b1)	No register bits; should be written with 0 and read as 0		—
CH0	Channel Select Bit (2)	b6 b5 b4 0 0 0 : Channel 0 0 0 1 : Channel 1 0 1 0 : Channel 2 0 1 1 : Channel 3 1 0 0 : Channel 4 1 0 1 : Channel 5 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
CH1			
CH2			
— (b7)	No register bit; should be written with 0 and read as 0		—

Notes:

1. A write operation to this register is ignored during the transmit FIFO check.
2. Data transfer is not performed from the transmit FIFO to registers GFRR0 to GFRR3 when there is no data in the specified FIFO (when the GFijFL register is 0).

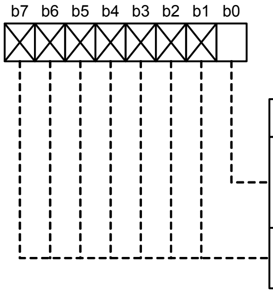
Figure 26.13 GTFRC Register in the R32C/145 Group

This register sets the conditions required for transferring frames stored in the transmit FIFO to registers GFRR0 to GFRR3.

The first message frame stored in the corresponding transmit FIFO is transferred to registers GFRR0 to GFRR3 by setting the channel and FIFO numbers. The value of the GFijFL register does not change at this point; however, it is updated when reading the GFRR3 register. After reading the GFRR3 register, the next message frame continues being transferred to registers GFRR0 to GFRR3 when there is data left in the transmit FIFO that is specified in the GTFRC register.

26.1.9 Gateway Transmit FIFO Read Status Register (GTFRS)

Gateway Transmit FIFO Read Status Register



Symbol	Address	Reset Value
GTFRS	47221h	0000 0000b

Bit Symbol	Bit Name	Function	RW
BUSY	Busy Flag (1)	0: Frames are not yet transmitted/ frame transmission is completed 1: Frames are being transmitted	RO
— (b7-b1)	No register bits; these bits are read as 0		—

Note:

1. Read registers GFRR0 to GFRR3 when this bit is 0. They are read as undefined values when this bit is 1.

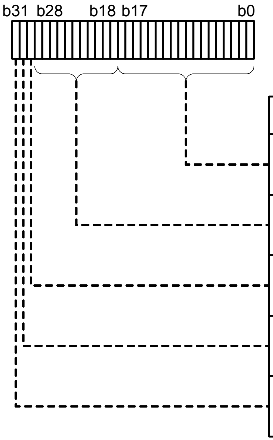
Figure 26.14 GTFRS Register

This register indicates the status of a frame transfer from the transmit FIFO to registers GFRR0 to GFRR3.

26.1.10 Gateway Transmit FIFO Read Register k (GFRRk) (k = 0 to 3)

The message frames stored in the transmit FIFO are read using this register. After the transmit FIFO to be read is specified in the GTFRC register, the first frame of the corresponding FIFO is transferred to registers GFRR0 to GFRR3. Read registers GFRR0 to GFRR3 in the following order: GFRR0, GFRR1, GFRR2, and GFRR3. When the next message frame exists in the transmit FIFO after reading the GFRR3 register, it is transferred to registers GFRR0 to GFRR3.

Gateway Transmit FIFO Read Register 0 (1)



Symbol	Address	Reset Value
GFRR0	47293h-47290h	0000 0000h

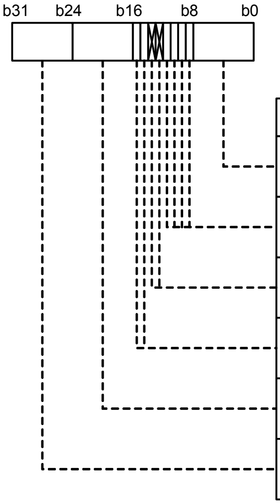
Bit Symbol	Bit Name	Function	RW
EID	Extended ID	The extended ID value is stored	RO
SID	Standard ID	The standard ID value is stored	RO
EPC	Error Parity Check Bit (2)	0: No parity error 1: Parity error occurred	RO
RTR	Remote Frame Request Bit	The value of the remote frame request bit is stored	RO
IDE	ID Extension Bit	The value of the ID extension bit is stored	RO

Notes:

1. This register value is valid when the BUSY bit in the GTFRS register is 0.
2. The value is set when a message frame is transferred from the transmit FIFO to this register.

Figure 26.15 GFRR0 Register

Gateway Transmit FIFO Read Register 1 (1)



Symbol	Address	Reset Value
GFRR1	47297h-47294h	0000 0000h

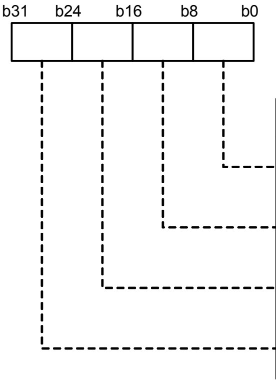
Bit Symbol	Bit Name	Function	RW
LA1	Label 1	The label 1 value added during routing is stored	RO
DLC	Data Length Code	The value of the data length code is stored	RO
— (b13-b12)	No register bits; these bits are read as 0		—
LA0	Label 0	The label 0 value added during routing is stored	RO
DATA0	Data 0	The data 0 value is stored	RO
DATA1	Data 1	The data 1 value is stored	RO

Note:

1. This register value is valid when the BUSY bit in the GTFRS register is 0.

Figure 26.16 GFRR1 Register

Gateway Transmit FIFO Read Register 2 (1)



Symbol	Address	Reset Value
GFRR2	4729Bh-47298h	0000 0000h

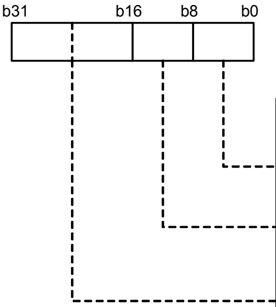
Bit Symbol	Bit Name	Function	RW
DATA2	Data 2	The data 2 value is stored	RO
DATA3	Data 3	The data 3 value is stored	RO
DATA4	Data 4	The data 4 value is stored	RO
DATA5	Data 5	The data 5 value is stored	RO

Note:

1. This register value is valid when the BUSY bit in the GTFRS register is 0.

Figure 26.17 GFRR2 Register

Gateway Transmit FIFO Read Register 3 (1, 2)



Symbol	Address	Reset Value
GFRR3	4729Fh-4729Ch	0000 0000h

Bit Symbol	Bit Name	Function	RW
DATA6	Data 6	The data 6 value is stored	RO
DATA7	Data 7	The data 7 value is stored	RO
TS	Time Stamp	The time stamp value added during routing is stored. The lowest bit is always read as 0	RO

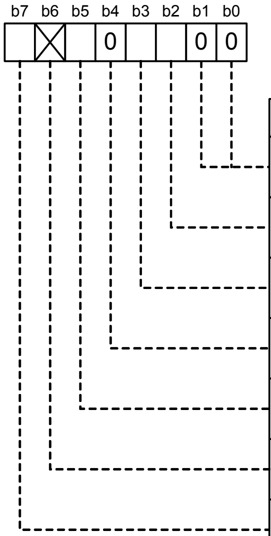
Notes:

1. The fill level of the corresponding transmit FIFO is reduced by 1 when this register is read.
2. This register value is valid when the BUSY bit in the GTFRS register is 0.

Figure 26.18 GFRR3 Register

26.1.11 Gateway Parity Check Control Register (GPCCR)

Gateway Parity Check Control Register [1]



Symbol	Address	Reset Value
GPCCR	47214h	0000 0000b

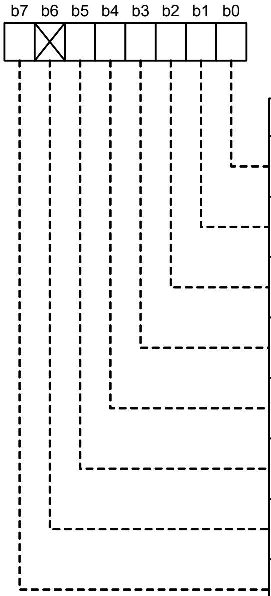
Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved	Should be written with 0	RW
PCE2	Channel 2 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
PCE3	Channel 3 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
— (b4)	Reserved	Should be written with 0	RW
PCE5	Channel 5 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
— (b6)	No register bit; should be written with 0 and read as 0		—
RMPCE	Routing Table Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW

Note:

1. Set this register in configuration mode.

Figure 26.19 GPCCR Register in the R32C/142 Group

Gateway Parity Check Control Register (1)



Symbol	Address	Reset Value
GPCCR	47214h	0000 0000b

Bit Symbol	Bit Name	Function	RW
PCE0	Channel 0 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
PCE1	Channel 1 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
PCE2	Channel 2 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
PCE3	Channel 3 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
PCE4	Channel 4 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
PCE5	Channel 5 Transmit FIFO Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW
— (b6)	No register bit; should be written with 0 and read as 0		—
RMPCE	Routing Table Parity Check Enable Bit	0: Parity check disabled 1: Parity check enabled	RW

Note:

1. Set this register in configuration mode.

Figure 26.20 GPCCR Register in the R32C/145 Group

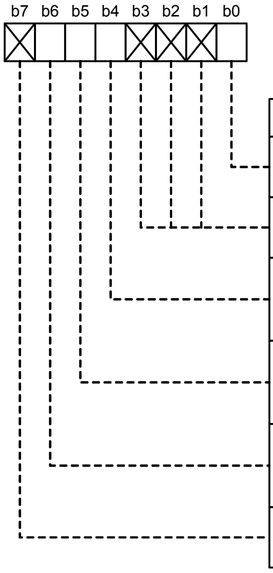
This register enables/disables the parity check on the routing table memory and message frames. The type of parity is even parity.

When the RMPCE bit is 1, the parity check is performed during the checksum calculation by setting the CALC bit in the GRMCC register to 1. Read the RMPER bit in the GRESR register to verify the result of the parity check.

When the PCEi bit is 1, a parity is added when a message frame is transferred to the transmit FIFO. The parity check is performed when a message frame is transferred from the transmit FIFO to registers GFRR0 to GFRR3. The result is stored in the EPC bit in the GFRR0 register.

26.1.12 Gateway Time Stamp Timer Control Register (GTSCR)

Gateway Time Stamp Timer Control Register



Symbol GTSCR

Address 47218h

Reset Value 0000 0000b

Bit Symbol	Bit Name	Function	RW
CST	Time Stamp Timer Count Start Bit	0: Stop counter (1) 1: Start counter	RW
— (b3-b1)	No register bits; should be written with 0 and read as 0		—
DIV0	Count Source Divide Ratio Select Bit (2)	b6 b5 b4 0 0 0 : Divide-by-128 0 0 1 : Divide-by-256 0 1 0 : Divide-by-512 0 1 1 : Divide-by-1024 1 0 0 : Divide-by-2048 1 0 1 : Divide-by-4096 1 1 0 : Divide-by-8192 1 1 1 : Divide-by-16384	RW
DIV1			
DIV2			
— (b7)	No register bit; should be written with 0 and read as 0		—

Notes:

1. The GTSTR register becomes 0000h when setting this bit to 0.
2. The setting values of bits DIV2 to DIV0 are valid only when the CST bit is 0.

Figure 26.21 GTSCR Register

This register starts/stops the counter of the time stamp timer (16-bit free-running timer) and controls the divide ratio of the count source.

The operating clock for the CAN gateway module which is input to the CAN gateway module has the same frequency as the peripheral bus clock.

Bits DIV2 to DIV0

Bits DIV2 to DIV0 divide the operating clock of the CAN gateway module to determine the count source cycle of the time stamp timer.

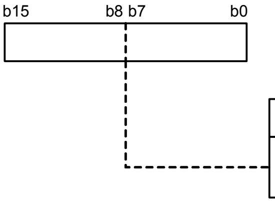
Table 26.3 lists the count source cycles and the overflow periods which are calculated according to the divide ratios and the clocks for timer operations.

Table 26.3 Count Source Cycle and Overflow Period for the Time Stamp Timer

Divide Ratio	Operating Clock (Peripheral Bus Clock)	Count Source Cycle	Overflow Period
1/128	8 MHz	16.0 μs	Approximately 1.0 s
	16 MHz	8.0 μs	Approximately 0.5 s
	20 MHz	6.4 μs	Approximately 0.4 s
	32 MHz	4.0 μs	Approximately 0.3 s
1/256	8 MHz	32.0 μs	Approximately 2.1 s
	16 MHz	16.0 μs	Approximately 1.0 s
	20 MHz	12.8 μs	Approximately 0.8 s
	32 MHz	8.0 μs	Approximately 0.5 s
1/512	8 MHz	64.0 μs	Approximately 4.2 s
	16 MHz	32.0 μs	Approximately 2.1 s
	20 MHz	25.6 μs	Approximately 1.7 s
	32 MHz	16.0 μs	Approximately 1.0 s
1/1024	8 MHz	128.0 μs	Approximately 8.4 s
	16 MHz	64.0 μs	Approximately 4.2 s
	20 MHz	51.2 μs	Approximately 3.4 s
	32 MHz	32.0 μs	Approximately 2.1 s
1/2048	8 MHz	256.0 μs	Approximately 16.8 s
	16 MHz	128.0 μs	Approximately 8.4 s
	20 MHz	102.4 μs	Approximately 6.7 s
	32 MHz	64.0 μs	Approximately 4.2 s
1/4096	8 MHz	512.0 μs	Approximately 33.6 s
	16 MHz	256.0 μs	Approximately 16.8 s
	20 MHz	204.8 μs	Approximately 13.4 s
	32 MHz	128.0 μs	Approximately 8.4 s
1/8192	8 MHz	1024.0 μs	Approximately 67.1 s
	16 MHz	512.0 μs	Approximately 33.6 s
	20 MHz	409.6 μs	Approximately 26.8 s
	32 MHz	256.0 μs	Approximately 16.8 s
1/16384	8 MHz	2048.0 μs	Approximately 134.2 s
	16 MHz	1024.0 μs	Approximately 67.1 s
	20 MHz	819.2 μs	Approximately 53.7 s
	32 MHz	512.0 μs	Approximately 33.6 s

26.1.13 Gateway Time Stamp Timer Register (GTSTR)

Gateway Time Stamp Timer Register (1)



Symbol	Address	Reset Value
GTSTR	47269h-47268h	0000h

Function	RW
The bits are read as the current count value	RO

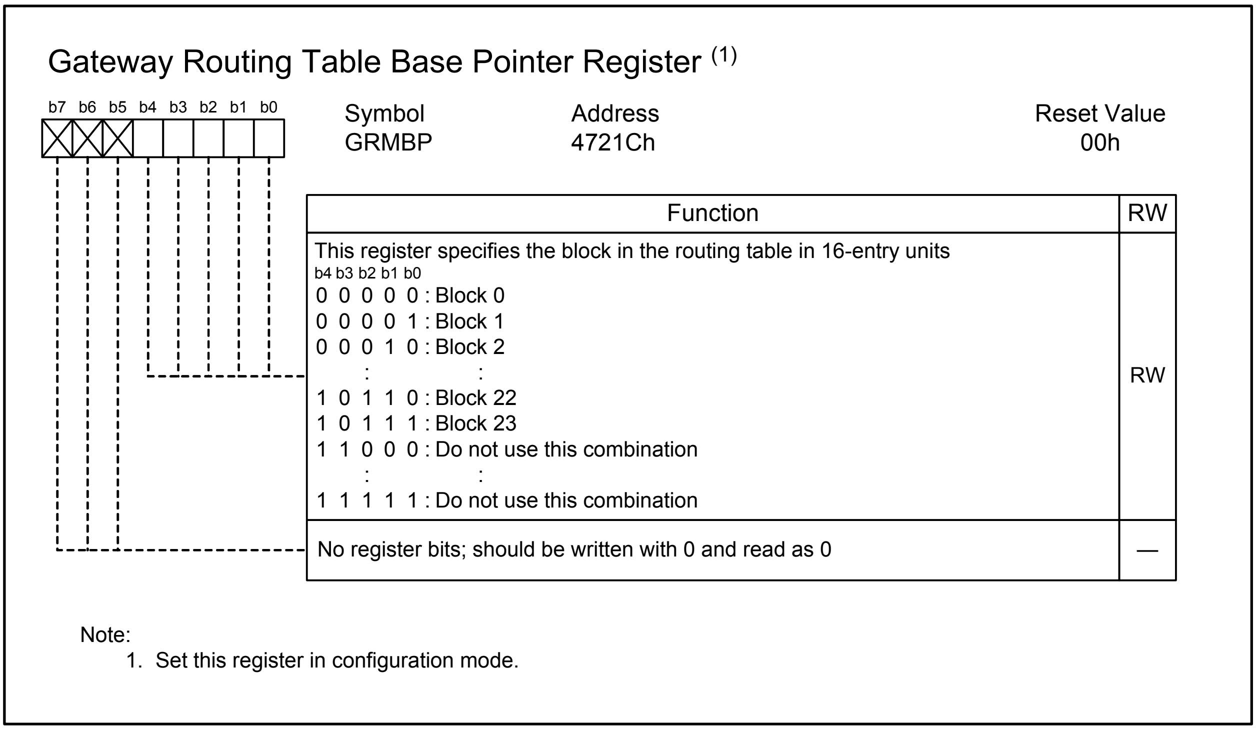
Note:

1. This register becomes 0000h when the CST bit in the GTSCR register is set to 0.

Figure 26.22 GTSTR Register

This register is a 16-bit free-running timer for the time stamp.
When transferring a message frame to the transmit FIFO, upper 15 bits of this register are added to the TS bit of the corresponding message frame.
When transmitting frames, read this register to measure the data retention period from when a frame is stored in the transmit FIFO until it is transmitted.

26.1.14 Gateway Routing Table Base Pointer Register (GRMBP)



Gateway Routing Table Base Pointer Register [(1)]

Symbol	Address	Reset Value
GRMBP	4721Ch	00h

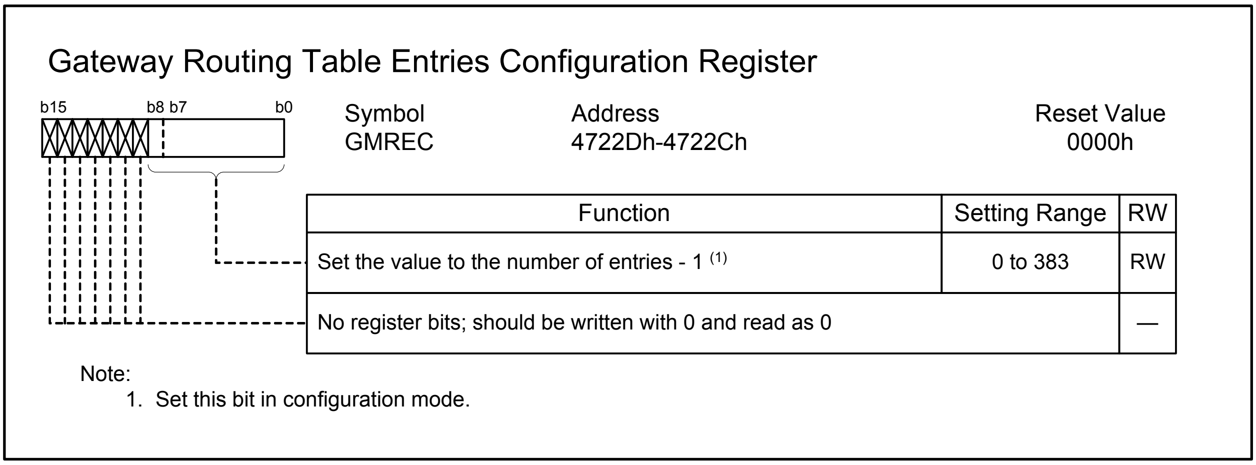
Function	RW
This register specifies the block in the routing table in 16-entry units b4 b3 b2 b1 b0 0 0 0 0 0 : Block 0 0 0 0 0 1 : Block 1 0 0 0 1 0 : Block 2 : : 1 0 1 1 0 : Block 22 1 0 1 1 1 : Block 23 1 1 0 0 0 : Do not use this combination : : 1 1 1 1 1 : Do not use this combination	RW
No register bits; should be written with 0 and read as 0	—

Note:
1. Set this register in configuration mode.

Figure 26.23 GRMBP Register

This register specifies the block in the routing table.
The routing table consists of 384 entries which are divided into 24 blocks (i.e. 16 entries per block). A write/read access to the routing table should be performed in 1-block units.

26.1.15 Gateway Routing Table Entries Configuration Register (GMREC)



Gateway Routing Table Entries Configuration Register

Symbol	Address	Reset Value
GMREC	4722Dh-4722Ch	0000h

Function	Setting Range	RW
Set the value to the number of entries - 1 [(1)]	0 to 383	RW
No register bits; should be written with 0 and read as 0		—

Note:
1. Set this bit in configuration mode.

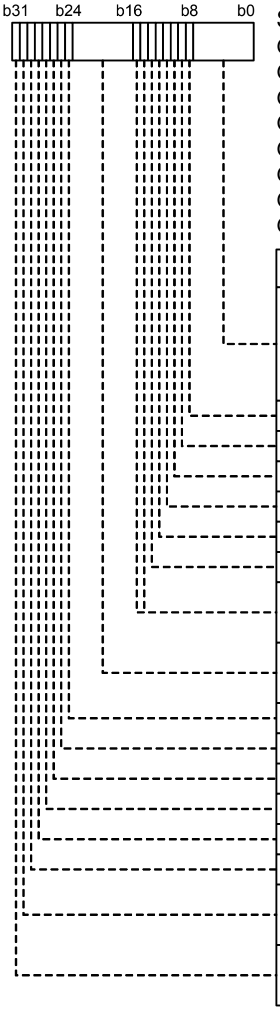
Figure 26.24 GMREC Register

This register specifies the number of entries in the routing table.
The CAN gateway module determines the search area in the routing table and the area for calculating the checksum based on this register value.

26.1.16 Gateway Routing Table Register mH/L (GRMmH/GRMmL) (m = 0 to 15)

This register performs a write/read access to the routing table in 1-block units (16 entries). Set the GRMBP register to specify the block number (0 to 23).

Gateway Routing Table Register mH (m = 0 to 15) (1, 2, 3)



Symbol	Address	Reset Value
GRM0H, GRM1H	47307h-47304h, 4730Fh-4730Ch	Undefined
GRM2H, GRM3H	47317h-47314h, 4731Fh-4731Ch	Undefined
GRM4H, GRM5H	47327h-47324h, 4732Fh-4732Ch	Undefined
GRM6H, GRM7H	47337h-47334h, 4733Fh-4733Ch	Undefined
GRM8H, GRM9H	47347h-47344h, 4734Fh-4734Ch	Undefined
GRM10H, GRM11H	47357h-47354h, 4735Fh-4735Ch	Undefined
GRM12H, GRM13H	47367h-47364h, 4736Fh-4736Ch	Undefined
GRM14H, GRM15H	47377h-47374h, 4737Fh-4737Ch	Undefined

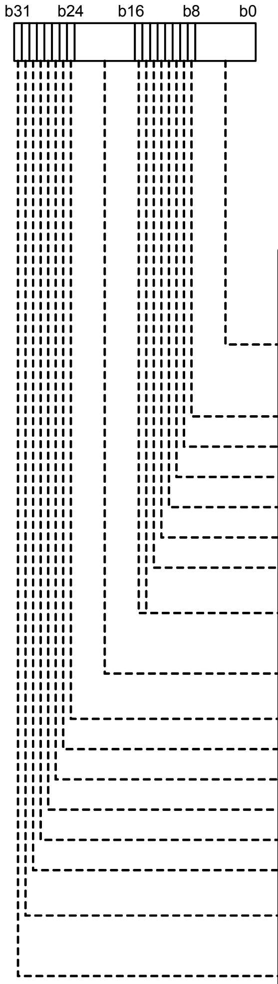
Bit Symbol	Bit Name	Function	RW
IDM	Message ID Mask	Sets whether to apply a mask to the lower 8 bits of the message ID 0: Not applied 1: Applied	RW
— (b8)	Processing Channel Setting Bit x (x = 2, 3, 5) (4)	Selects whether or not frames received from channel x are processed 0: Not processed 1: Processed Bits 8, 9, and 12 are reserved. Should be written with 0	RW
— (b9)			RW
PCH2			RW
PCH3			RW
— (b12)			RW
PCH5			RW
LA0	Label Area 0	Sets the label 0 value which is added to a frame during routing	RW
LA1	Label Area 1	Sets the label 1 value which is added to a frame during routing	RW
— (b24)	Destination Channel Setting Bit y (y = 2, 3, 5) (4)	Selects whether or not frames are transferred to the transmit FIFO of channel y 0: Not transferred 1: Transferred Bits 24, 25, and 28 are reserved. Should be written with 0	RW
— (b25)			RW
DCH2			RW
DCH3			RW
— (b28)			RW
DCH5			RW
FIFO	Transmit FIFO Number Setting Bit (5)	0: Transmit FIFO0 1: Transmit FIFO1	RW
PAR	Parity Bit (6)	0: Parity is 0 1: Parity is 1	RO

Notes:

1. Access this register in configuration mode and when the GCSR register is 00h (routing is stopped for all channels).
2. A 32-bit write/read access should be performed to this register.
3. Except for the PAR bit, all other 31 bits are targeted for the checksum calculation.
4. Multiple channels can be set to 1. A routing error occurs when bits PCH5, PCH3, and PCH2 or bits DCH5, DCH3, and DCH2 are all 0.
5. All channels must have the same transmit FIFO number.
6. Write 0 to this bit when setting the routing table. This bit is automatically calculated and added when writing.

Figure 26.25 Registers GRM0H to GRM15H in the R32C/142 Group

Gateway Routing Table Register mH (m = 0 to 15) (1, 2, 3)



Symbol	Address	Reset Value
GRM0H, GRM1H	47307h-47304h, 4730Fh-4730Ch	Undefined
GRM2H, GRM3H	47317h-47314h, 4731Fh-4731Ch	Undefined
GRM4H, GRM5H	47327h-47324h, 4732Fh-4732Ch	Undefined
GRM6H, GRM7H	47337h-47334h, 4733Fh-4733Ch	Undefined
GRM8H, GRM9H	47347h-47344h, 4734Fh-4734Ch	Undefined
GRM10H, GRM11H	47357h-47354h, 4735Fh-4735Ch	Undefined
GRM12H, GRM13H	47367h-47364h, 4736Fh-4736Ch	Undefined
GRM14H, GRM15H	47377h-47374h, 4737Fh-4737Ch	Undefined

Bit Symbol	Bit Name	Function	RW
IDM	Message ID Mask	Sets whether to apply a mask to the lower 8 bits of the message ID 0: Not applied 1: Applied	RW
PCH0	Processing Channel Setting Bit x (x = 0 to 5) (4)	Selects whether or not frames received from channel x are processed 0: Not processed 1: Processed	RW
PCH1			RW
PCH2			RW
PCH3			RW
PCH4			RW
PCH5			RW
LA0	Label Area 0	Sets the label 0 value which is added to a frame during routing	RW
LA1	Label Area 1	Sets the label 1 value which is added to a frame during routing	RW
DCH0	Destination Channel Setting Bit y (y = 0 to 5) (4)	Selects whether or not frames are transferred to the transmit FIFO of channel y 0: Not transferred 1: Transferred	RW
DCH1			RW
DCH2			RW
DCH3			RW
DCH4			RW
DCH5			RW
FIFO	Transmit FIFO Number Setting Bit (5)	0: Transmit FIFO0 1: Transmit FIFO1	RW
PAR	Parity Bit (6)	0: Parity is 0 1: Parity is 1	RO

Notes:
1. Access this register in configuration mode and when the GCSR register is 00h (routing is stopped for all channels).
2. A 32-bit write/read access should be performed to this register.
3. Except for the PAR bit, all other 31 bits are targeted for the checksum calculation.
4. Multiple channels can be set to 1. A routing error occurs when bits PCH5 to PCH0 or bits DCH5 to DCH0 are all 0.
5. All channels must have the same transmit FIFO number.
6. Write 0 to this bit when setting the routing table. This bit is automatically calculated and added when writing.

Figure 26.26 Registers GRM0H to GRM15H in the R32C/145 Group

IDM Bit (bits b7 to b0)

Set this bit to enable routing for multiple message IDs with one entry. The masking information is applied to the lower 8 bits of the message IDs. Set the corresponding bit of the IDM bit (message ID mask) to 1 for masking.

Table 26.4 lists a setting example of the masking information.

Table 26.4 Masking Information Setting Example

Message ID (ID) (29 Bits)		Message ID Mask (IDM) (8 Bits)	Message ID for Routing
Extended ID	0F00 0700h	FFh	0F00 0700h to 0F00 07FFh
	1FFF 00F0h	07h	1FFF 00F0h to 1FFF 00F7h
	1FFF 00F0h	02h	1FFF 00F0h, 1FFF 00F2h
Standard ID	0000 0700h	FFh	700h to 7FFh
	0000 05F0h	03h	5F0h to 5F3h
	0000 05F0h	04h	5F0h, 5F4h

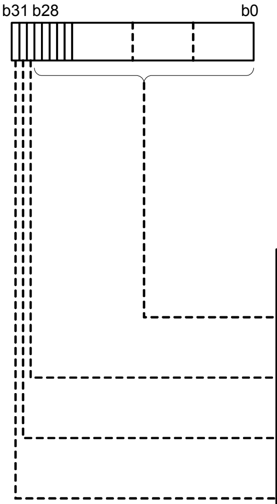
LA0 Bit (bits b15 and b14)

These bits are added to a message frame when routing is being processed.
This bit information is not used in the processing and can be defined by the users.

LA1 Bit (bits b23 to b16)

These bits are added to a message frame when routing is being processed.
This bit information is not used in the processing and can be defined by the users.

Gateway Routing Table Register mL (m = 0 to 15) [1, 2]



Symbol	Address	Reset Value
GRM0L, GRM1L	47303h-47300h, 4730Bh-47308h	Undefined
GRM2L, GRM3L	47313h-47310h, 4731Bh-47318h	Undefined
GRM4L, GRM5L	47323h-47320h, 4732Bh-47328h	Undefined
GRM6L, GRM7L	47333h-47330h, 4733Bh-47338h	Undefined
GRM8L, GRM9L	47343h-47340h, 4734Bh-47348h	Undefined
GRM10L, GRM11L	47353h-47350h, 4735Bh-47358h	Undefined
GRM12L, GRM13L	47363h-47360h, 4736Bh-47368h	Undefined
GRM14L, GRM15L	47373h-47370h, 4737Bh-47378h	Undefined

Bit Symbol	Bit Name	Function	RW
ID	Message ID [3]	Message ID	RW
RTR	Remote Frame Request Bit	0: Data Frame 1: Remote Frame	RW
IDE	ID Extension Bit	0: Standard ID 1: Extended ID	RW
PAR	Parity Bit [4]	0: Parity is 0 1: Parity is 1	RO

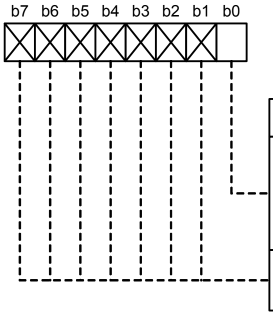
Notes:
1. Access this register in configuration mode and when the GCSR register is 00h (routing is stopped for all channels).
2. A 32-bit write/read access should be performed to this register.
3. The value must be LSB-justified for the standard ID.
4. Write 0 to this bit when setting the routing table. This bit is automatically calculated and added when writing.

Figure 26.27 Registers GRM0L to GRM15L

When setting the routing table, note that the value of the lower 31 bits (IDE + RTR + ID) for each entry should differ from the other entries and should be set in ascending order.

26.1.17 Gateway Echo-back Control Register (GEBCR)

Gateway Echo-back Control Register



Symbol	Address	Reset Value
GEBCR	4722Eh	0000 0000b

Bit Symbol	Bit Name	Function	RW
ECHO	Echo-back Enable Bit [1]	0: Transfer to the same channel is disabled 1: Transfer to the same channel is enabled	RW
— (b7-b1)	No register bits; should be written with 0 and read as 0		—

Note:
1. Set this bit in configuration mode.

Figure 26.28 GEBCR Register

This register enables/disables the transfer of a routable frame to the same channel from which the frame is received.

ECHO Bit

Table 26.5 lists the ECHO bit setting and the routing behavior. When received frames are to be routed, the module behaves according to the setting value of the ECHO bit.

- When transfer to the same channel is enabled

  Message frames are transferred to all transmit FIFOs of the channels whose DCHi bit of the entry data is set to 1.
- When transfer to the same channel is disabled

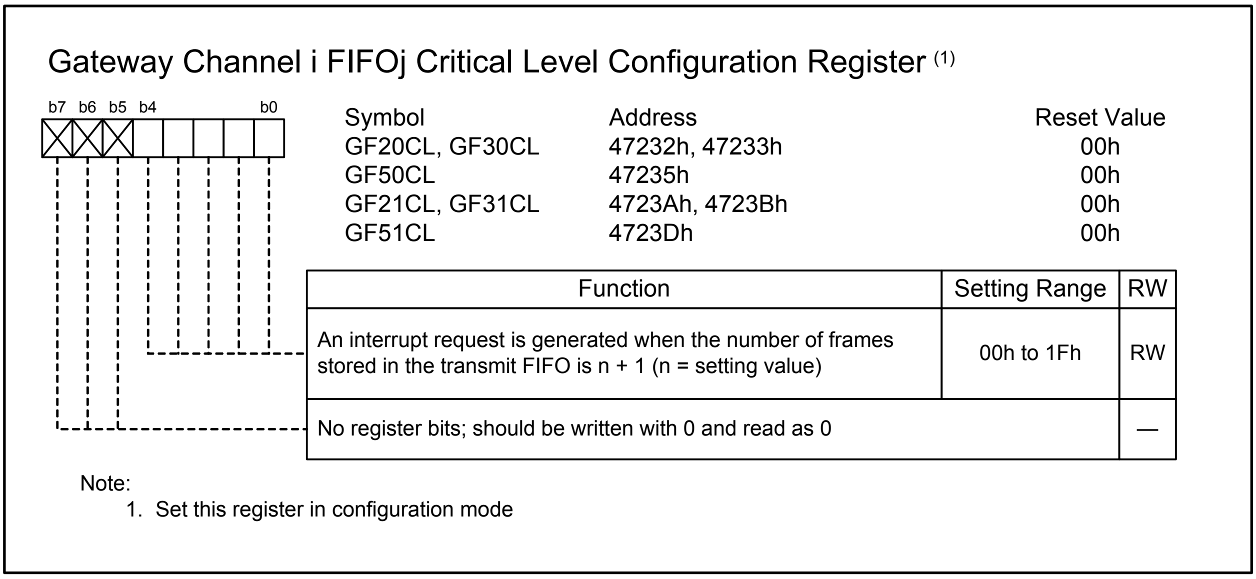
  Message frames are not transferred to the transmit FIFO of the same channel as the receive channel, whose the DCHi bit of the entry data is set to 1. They are transferred to the other channels.

  When there are no channels set as destinations other than the receive channel, a routing error occurs, and message frames are discarded instead of being transferred.

Table 26.5 ECHO Bit Setting and Routing Behavior in the R32C/145 Group

Entry Data in Routing Table												Receive Channel	CAN Gateway Behavior	
PCH0 to PCH5 (channels to be processed)						DCH0 to DCH5 (destination channels)							ECHO bit is 1 (transfer to the same channel is enabled)	ECHO bit is 0 (transfer to the same channel is disabled)
0	1	2	3	4	5	0	1	2	3	4	5			
1	0	0	0	0	0	0	1	1	1	1	1	0	Transferring to channels 1 to 5	Transferring to channels 1 to 5
0	0	1	0	0	0	0	0	1	0	0	1	2	Transferring to channels 2 and 5	Transferring to channel 5
0	0	0	0	1	0	0	0	0	0	1	0	4	Transferring to channel 4	Routing error (destination is not specified), frames are discarded

26.1.18 Gateway Channel i FIFOj Critical Level Configuration Register (GFijCL)



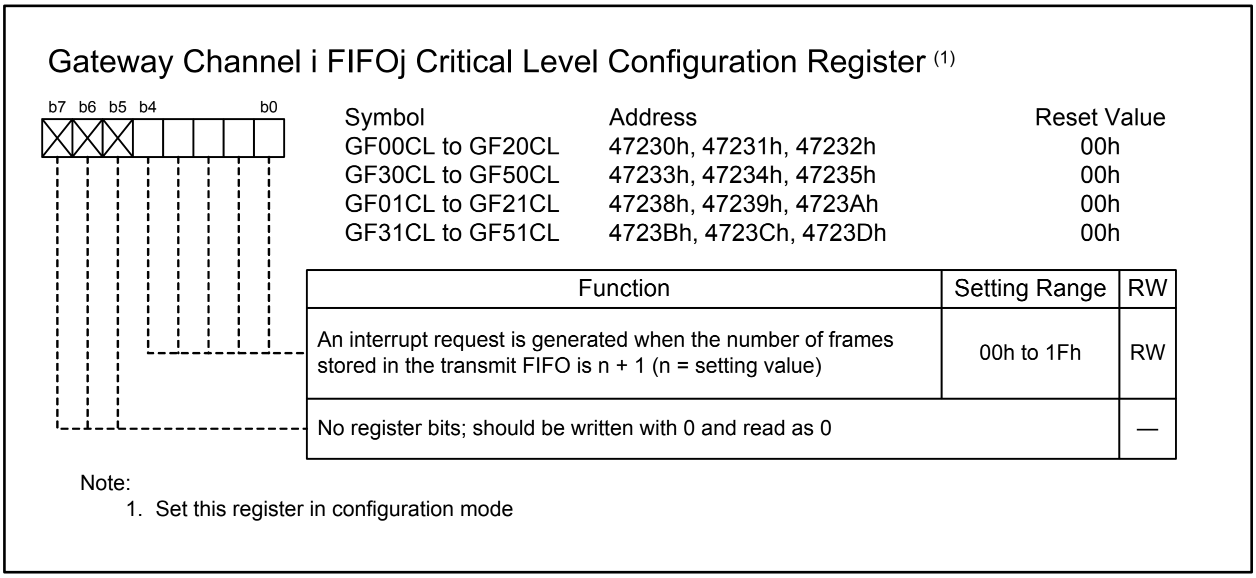
Gateway Channel i FIFOj Critical Level Configuration Register (1)

Symbol	Address	Reset Value
GF20CL, GF30CL	47232h, 47233h	00h
GF50CL	47235h	00h
GF21CL, GF31CL	4723Ah, 4723Bh	00h
GF51CL	4723Dh	00h

Function	Setting Range	RW
An interrupt request is generated when the number of frames stored in the transmit FIFO is n + 1 (n = setting value)	00h to 1Fh	RW
No register bits; should be written with 0 and read as 0		—

Note:
1. Set this register in configuration mode

Figure 26.29 Registers GF20CL, GF30CL, GF50CL, GF21CL, GF31CL, and GF51CL in the R32C/142 Group



Gateway Channel i FIFOj Critical Level Configuration Register (1)

Symbol	Address	Reset Value
GF00CL to GF20CL	47230h, 47231h, 47232h	00h
GF30CL to GF50CL	47233h, 47234h, 47235h	00h
GF01CL to GF21CL	47238h, 47239h, 4723Ah	00h
GF31CL to GF51CL	4723Bh, 4723Ch, 4723Dh	00h

Function	Setting Range	RW
An interrupt request is generated when the number of frames stored in the transmit FIFO is n + 1 (n = setting value)	00h to 1Fh	RW
No register bits; should be written with 0 and read as 0		—

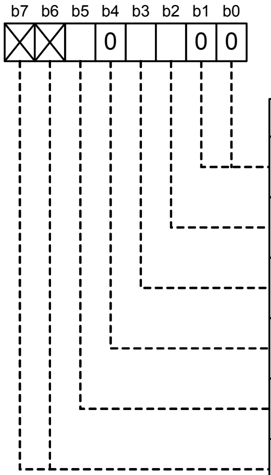
Note:
1. Set this register in configuration mode

Figure 26.30 Registers GF00CL to GF51CL in the R32C/145 Group

This register sets the critical level (the number of frames stored in the transmit FIFO) to generate a CANi gateway transmit FIFO interrupt request.
When the value of the GFijFL register exceeds that of the GFijCL register, a CANi gateway transmit FIFO interrupt request is generated.
Registers GFi0CL and GFi1CL are for FIFO0 and FIFO1, respectively.

26.1.19 Gateway Transmit FIFO Clear Register (GTFCR)

Gateway Transmit FIFO Clear Register

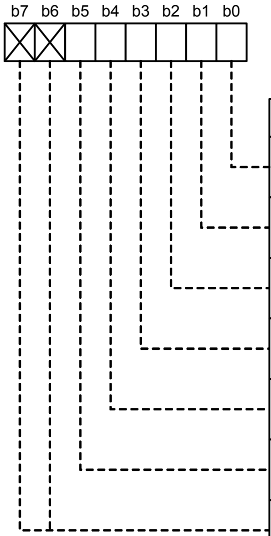


Symbol	Address	Reset Value
GTFCR	47208h	0000 0000b

Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved	Should be written with 0	RW
FCL2	Channel 2 Transmit FIFO Fill Level Clear Bit	The fill levels of the both transmit FIFO0 and FIFO1 become 0 by setting this bit to 1. It automatically becomes 0 when the levels of the both FIFOs become 0 Bit 4 should be written with 0	RW
FCL3	Channel 3 Transmit FIFO Fill Level Clear Bit		RW
— (b4)	Reserved		RW
FCL5	Channel 5 Transmit FIFO Fill Level Clear Bit		RW
— (b7-b6)	No register bits; should be written with 0 and read as 0		—

Figure 26.31 GTFCR Register in the R32C/142 Group

Gateway Transmit FIFO Clear Register



Symbol	Address	Reset Value
GTFCR	47208h	0000 0000b

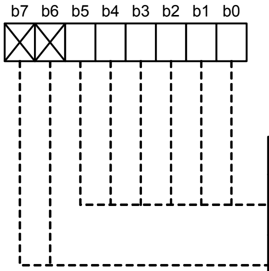
Bit Symbol	Bit Name	Function	RW
FCL0	Channel 0 Transmit FIFO Fill Level Clear Bit	The fill levels of the both transmit FIFO0 and FIFO1 become 0 by setting this bit to 1. It automatically becomes 0 when the levels of the both FIFOs become 0	RW
FCL1	Channel 1 Transmit FIFO Fill Level Clear Bit		RW
FCL2	Channel 2 Transmit FIFO Fill Level Clear Bit		RW
FCL3	Channel 3 Transmit FIFO Fill Level Clear Bit		RW
FCL4	Channel 4 Transmit FIFO Fill Level Clear Bit		RW
FCL5	Channel 5 Transmit FIFO Fill Level Clear Bit		RW
— (b7-b6)	No register bits; should be written with 0 and read as 0		—

Figure 26.32 GTFCR Register in the R32C/145 Group

This register clears the fill levels of the transmit FIFOs to 0. The messages in the transmit FIFOs are discarded when the fill levels are cleared to 0.

26.1.20 Gateway Channel i FIFOj Fill Level Register (GFijFL)

Gateway Channel i FIFOj Fill Level Register (1, 2, 3)



Symbol	Address	Reset Value
GF20FL, GF21FL	47250h, 47252h	00h
GF30FL, GF31FL	47254h, 47256h	00h
GF50FL, GF51FL	4725Ch, 4725Eh	00h

Function	RW
These bits indicate the number of frames stored in the transmit FIFO	RO
No register bits; these bits are read as 0	—

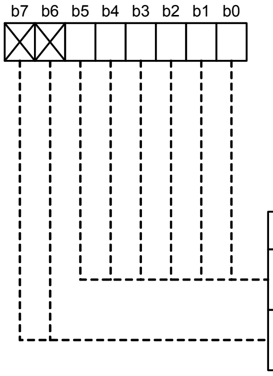
Notes:

1. Depending on the timing of reading this register, the number of frames being stored in the transmit FIFO may exceed the number of frames actually transferred (i.e. when a new frame is stored into the transmit FIFO after reading this register).
2. The value of this register is updated in the following cases:
   - When a frame transfer to the transmit FIFO is completed
   - When reading the GFRR3 register
   - When the fill level of the transmit FIFO is cleared

   The value does not change when frames in the transmit FIFO are transferred to registers GFRR0 to GFRR3.
3. This register becomes 00h when the transmit FIFO check is completed or after a CAN gateway reset.

Figure 26.33 Registers GF20FL, GF30FL, GF50FL, GF21FL, GF31FL, and GF51FL in the R32C/142 Group

Gateway Channel i FIFOj Fill Level Register (1, 2, 3)



Symbol	Address	Reset Value
GF00FL, GF01FL	47248h, 4724Ah	00h
GF10FL, GF11FL	4724Ch, 4724Eh	00h
GF20FL, GF21FL	47250h, 47252h	00h
GF30FL, GF31FL	47254h, 47256h	00h
GF40FL, GF41FL	47258h, 4725Ah	00h
GF50FL, GF51FL	4725Ch, 4725Eh	00h

Function	RW
These bits indicate the number of frames stored in the transmit FIFO	RO
No register bits; these bits are read as 0	—

Notes:

1. Depending on the timing of reading this register, the number of frames being stored in the transmit FIFO may exceed the number of frames actually transferred (i.e. when a new frame is stored into the transmit FIFO after reading this register).
2. The value of this register is updated in the following cases:
   - When a frame transfer to the transmit FIFO is completed
   - When reading the GFRR3 register
   - When the fill level of the transmit FIFO is cleared

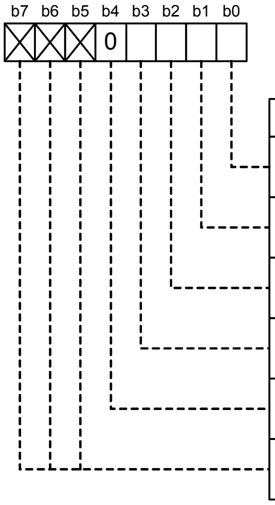
   The value does not change when frames in the transmit FIFO are transferred to registers GFRR0 to GFRR3.
3. This register becomes 00h when the transmit FIFO check is completed or after a CAN gateway reset.

Figure 26.34 Registers GF00FL to GF51FL in the R32C/145 Group

This register indicates the fill level of the transmit FIFO (the number of frames stored in the transmit FIFO). The value of this register indicates the number of frames that can be read from the transmit FIFO.

26.1.21 Gateway Channel i Transmit FIFO Interrupt Enable Register (GCiIE)

Gateway Channel i Transmit FIFO Interrupt Enable Register

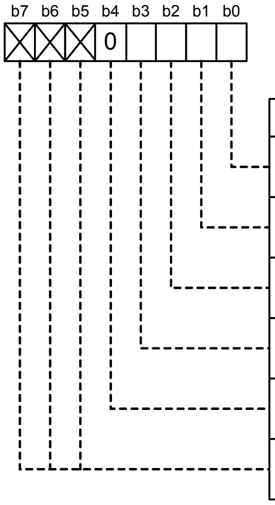


Symbol	Address	Reset Value
GC2IE, GC3IE	47272h, 47273h	0000 0000b
GC5IE	47275h	0000 0000b

Bit Symbol	Bit Name	Function	RW
F0CLIE	Transmit FIFO0 Critical Level Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
F1CLIE	Transmit FIFO1 Critical Level Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
F0OE	Transmit FIFO0 Overflow Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
F1OE	Transmit FIFO1 Overflow Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
— (b4)	Reserved	Should be written with 0	RW
— (b7-b5)	No register bits; should be written with 0 and read as 0		—

Figure 26.35 Registers GC2IE, GC3IE, and GC5IE in the R32C/142 Group

Gateway Channel i Transmit FIFO Interrupt Enable Register



Symbol	Address	Reset Value
GC0IE to GC2IE	47270h, 47271h, 47272h	0000 0000b
GC3IE to GC5IE	47273h, 47274h, 47275h	0000 0000b

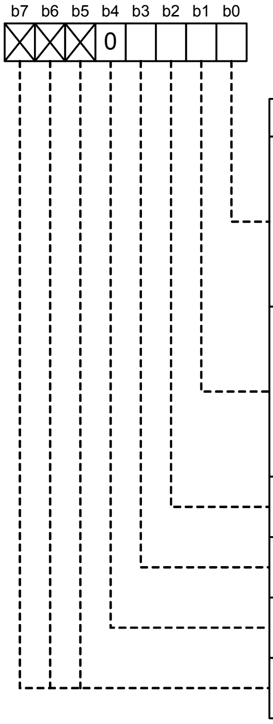
Bit Symbol	Bit Name	Function	RW
F0CLIE	Transmit FIFO0 Critical Level Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
F1CLIE	Transmit FIFO1 Critical Level Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
F0OE	Transmit FIFO0 Overflow Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
F1OE	Transmit FIFO1 Overflow Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	RW
— (b4)	Reserved	Should be written with 0	RW
— (b7-b5)	No register bits; should be written with 0 and read as 0		—

Figure 26.36 Registers GC0IE to GC5IE in the R32C/145 Group

This register enables/disables the transmit FIFOj critical level interrupt and the transmit FIFOj overflow interrupt.

26.1.22 Gateway Channel i Transmit FIFO Status Register (GCiSR)

Gateway Channel i Transmit FIFO Status Register



Symbol	Address	Reset Value
GC2SR, GC3SR	4727Ah, 4727Bh	0000 0000b
GC5SR	4727Dh	0000 0000b

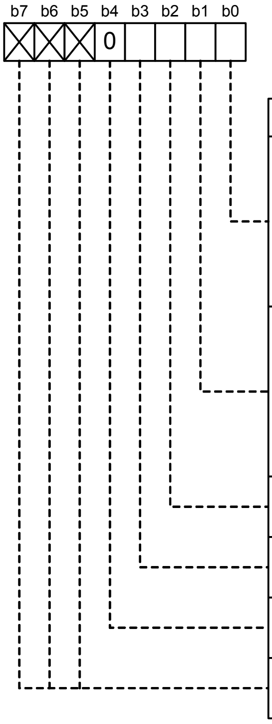
Bit Symbol	Bit Name	Function	RW
F0CL	Transmit FIFO0 Critical Level Flag (1)	0: The number of frames in the transmit FIFO0 is less than the critical level 1: The number of frames in the transmit FIFO0 is equal to or greater than the critical level	RO
F1CL	Transmit FIFO1 Critical Level Flag (1)	0: The number of frames in the transmit FIFO1 is less than the critical level 1: The number of frames in the transmit FIFO1 is equal to or greater than the critical level	RO
F0O	Transmit FIFO0 Overflow Flag (2)	0: No overflow of the transmit FIFO0 1: Overflow of the transmit FIFO0 (3)	RW
F1O	Transmit FIFO1 Overflow Flag (2)	0: No overflow of the transmit FIFO1 1: Overflow of the transmit FIFO1 (3)	RW
— (b4)	Reserved	Should be written with 0 and read as undefined value	RW
— (b7-b5)	No register bits; should be written with 0 and read as 0		—

Notes:

1. This flag becomes 1 when the value of the GFijFL register exceeds the value set in the GFijCL register. If not, it becomes 0.
2. It can only be set to 0. This setting is unchanged even if it is set to 1.
3. The data in the transmit FIFO is read as undefined when an overflow occurs.

Figure 26.37 Registers GC2SR, GC3SR, and GC5SR in the R32C/142 Group

Gateway Channel i Transmit FIFO Status Register



Symbol	Address	Reset Value
GC0SR to GC2SR	47278h, 47279h, 4727Ah	0000 0000b
GC3SR to GC5SR	4727Bh, 4727Ch, 4727Dh	0000 0000b

Bit Symbol	Bit Name	Function	RW
F0CL	Transmit FIFO0 Critical Level Flag (1)	0: The number of frames in the transmit FIFO0 is less than the critical level 1: The number of frames in the transmit FIFO0 is equal to or greater than the critical level	RO
F1CL	Transmit FIFO1 Critical Level Flag (1)	0: The number of frames in the transmit FIFO1 is less than the critical level 1: The number of frames in the transmit FIFO1 is equal to or greater than the critical level	RO
F0O	Transmit FIFO0 Overflow Flag (2)	0: No overflow of the transmit FIFO0 1: Overflow of the transmit FIFO0 (3)	RW
F1O	Transmit FIFO1 Overflow Flag (2)	0: No overflow of the transmit FIFO1 1: Overflow of the transmit FIFO1(3)	RW
— (b4)	Reserved	Should be written with 0 and read as undefined value	RW
— (b7-b5)	No register bits; should be written with 0 and read as 0		—

Notes:

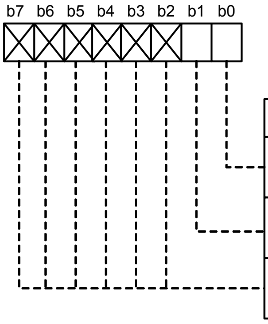
1. This flag becomes 1 when the value of the GFijFL register exceeds the value set in the GFijCL register. If not, it becomes 0.
2. It can only be set to 0. This setting is unchanged even if it is set to 1.
3. The data in the transmit FIFO is read as undefined when an overflow occurs.

Figure 26.38 Registers GC0SR to GC5SR in the R32C/145 Group

This register indicates the state of the transmit FIFO for each channel.
A CANi gateway transmit FIFO interrupt is generated when one or more flags are 1 and also when the corresponding interrupt is enabled.

26.1.23 Gateway Error Interrupt Enable Register (GIER)

Gateway Error Interrupt Enable Register (1)



Symbol	Address	Reset Value
GIER	47280h	0000 0000b

Bit Symbol	Bit Name	Function	RW
RERE	Routing Error Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled (2)	RW
HERE	Hardware Error Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled (2)	RW
— (b7-b2)	No register bits; should be written with 0 and read as 0		—

Notes:

1. Set this register in configuration mode.
2. An interrupt request is generated when the corresponding error occurs while this bit is 1.

Figure 26.39 GIER Register

This register enables/disables the routing error interrupt and the hardware error interrupt.

RERE Bit

Set this bit to enable/disable the routing error interrupt.

Incorrect setting of the routing table and parity error detection on the table cause a routing error. When a routing error occurs, frames being processed are discarded.

An interrupt request is generated by detecting any of the following when the routing error interrupt is enabled:

- When the destination channel is not specified (bits DCH5 to DCH0 are all 0).
- When the channel to be processed is not specified (bits PCH5 to PCH0 are all 0).
- The ECHO bit in the GEBCR register is 0 (transfer to the same channel is disabled) and the receive channel is the only destination set.
- There is a parity error on the routing table.

HERE Bit

Set this bit to enable/disable the hardware error interrupt.

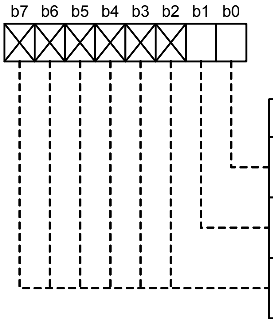
When a hardware error occurs, frames being processed are discarded.

An interrupt request is generated by detecting any of the following when the hardware error interrupt is enabled:

- When an error is detected in the message handler
- A search request is generated from the same channel during routing.

26.1.24 Gateway Error Status Register (GSR)

Gateway Error Status Register



Symbol	Address	Reset Value
GSR	47284h	0000 0000b

Bit Symbol	Bit Name	Function	RW
RES	Routing Error Status Flag (1)	0: No routing error occurred 1: Routing error occurred (2)	RW
HES	Hardware Error Status Flag (1)	0: No hardware error occurred 1: Hardware error occurred (2)	RW
— (b7-b2)	No register bits; should be written with 0 and read as 0		—

Notes:

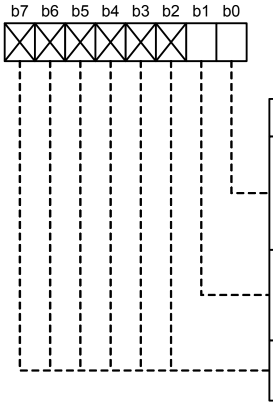
1. It can only be set to 0. This setting is unchanged even if it is set to 1.
2. Frames are discarded when this bit is 1.

Figure 26.40 GSR Register

This register indicates whether a routing error or a hardware error has occurred. A CAN gateway error interrupt request is generated either when the RERE bit in the GIER register and the RES bit are both 1, or when the HERE bit in the GIER register and the HES bit are both 1.

26.1.25 Gateway Routing Error Status Register (GRESR)

Gateway Routing Error Status Register



Symbol	Address	Reset Value
GRESR	47260h	0000 0000b

Bit Symbol	Bit Name	Function	RW
CHUC	Channel Unspecified Error Flag (1)	0: No error occurred 1: Destination and reception channels for the entry searched are not specified	RW
RMPER	Routing Table Parity Error Flag (1)	0: No error occurred 1: A parity error is detected on an entry by the message handler	RW
— (b7-b2)	No register bits; should be written with 0 and read as 0		—

Note:

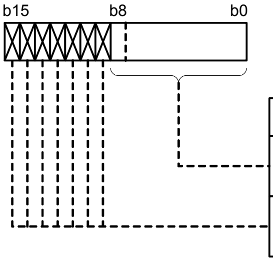
1. It can only be set to 0. This setting is unchanged even if it is set to 1.

Figure 26.41 GRESR Register

This register indicates that a routing error is occurred.

26.1.26 Gateway Error Entry Indication Register (GEEIR)

Gateway Error Entry Indication Register (1, 2)



Symbol	Address	Reset Value
GEEIR	47263h-47262h	0000h

Function	RW
These bits indicate the entry number -1 for which a channel unspecified error or a routing table parity error has occurred.	RO
No register bits; these bits are read as 0	—

Notes:

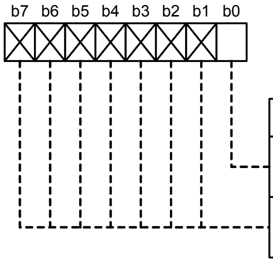
1. This register is enabled only when a routing error occurs.
2. A parity check on the routing table is performed during the ID search. Therefore, a parity error may be detected on an entry which does not match the received ID.

Figure 26.42 GEEIR Register

This register stores the value of the entry number - 1 for which a channel unspecified error or a routing error has occurred.

26.1.27 Gateway Bit Search Control Register n (GBSCn) (n = 0, 1)

Gateway Bit Search Control Register n (n = 0, 1) [(1)]



Symbol	Address	Reset Value
GBSC0, GBSC1	47383h, 47387h	0000 0000b

Bit Symbol	Bit Name	Function	RW
SMD	Search Mode Setting Bit	0: Auto search mode 1: Manual search mode	RW
— (b7-b1)	No register bits; should be written with 0 and read as 0		—

Note:

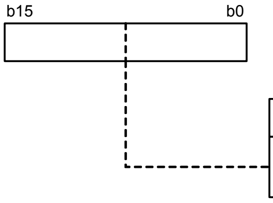
1. This register is not reset even if the RST bit in the GMR register is set to 1.

Figure 26.43 Registers GBSC0 and GBSC1

This register selects the bit search mode.

26.1.28 Gateway Bit Search Support Register n (GBSRn) (n = 0, 1)

Gateway Bit Search Support Register n (n = 0, 1) [(1)]



Symbol	Address	Reset Value
GBSR0	47381h-47380h	0000h
GBSR1	47385h-47384h	0000h

Function	RW
Sets a bit sequence for the bit search in 16-bit units. The search starts as soon as the data is set [(2, 3)]	RW

Notes:

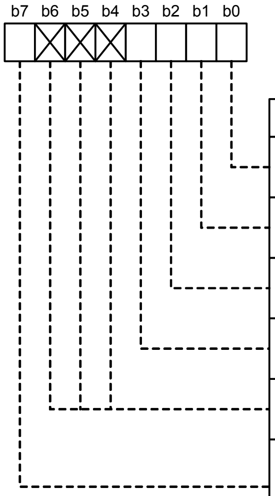
1. This register is not reset even if the RST bit in the GMR register is set to 1.
2. Bits which have been examined become 0. This register becomes 0000h once the search is completed for all bits.
3. Do not set 0000h to this register.

Figure 26.44 Registers GBSR0 and GBSR1

This register sets a bit sequence for the bit search.
Registers GBSR0 and GBSR1 have the same function.

26.1.29 Gateway Bit Search Status Register n (GBSSn) (n = 0, 1)

Gateway Bit Search Status Register n (n = 0, 1) (1, 2)



Symbol	Address	Reset Value
GBSS0, GBSS1	47382h, 47386h	1000 0000b

Bit Symbol	Bit Name	Function	RW
BP0	Bit Search Result	Indicates the bit position of the least significant 1 in the bit sequence set in the GBSRn register	RO
BP1			RO
BP2			RO
BP3			RO
— (b6-b4)	No register bits; these bits are read as 0		—
SCPL	Search Completion Flag	0: Search is in progress 1: Search is completed/not yet searched	RO

Notes:
1. This register is not reset even if the RST bit in the GMR register is set to 1.
2. Although a value should be written to this register in order to continue a search in manual search mode, the written value is ignored.

Figure 26.45 Registers GBSS0 and GBSS1

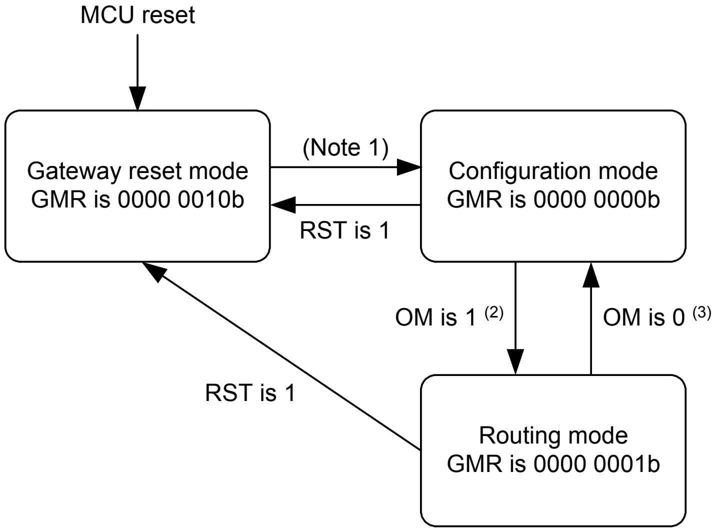
This register indicates the result of the search on the bit sequence set in the GBSRn register.

26.2 Operating Modes

The CAN gateway module has the following three operating modes:

- Gateway reset mode
- Configuration mode
- Routing mode

Figure 26.46 shows the mode transition.



OM and RST: Bits in the GMR register

Notes:

1. The module automatically enters configuration mode after resetting the CAN gateway module.
2. The OM bit cannot be set to 1 during the transmit FIFO check and the checksum calculation on the routing table.
3. When setting the OM bit to 0, make sure that the GCSR register has become 00h in order to move onto the next process.

Figure 26.46 Mode Transition

(1) Gateway Reset Mode

Setting the RST bit in the GMR register to 1 starts resetting the CAN gateway module. After a reset, the RST bit becomes 0, and the CAN gateway module enters configuration mode.

(2) Configuration Mode

The CAN gateway module enters configuration mode automatically after resetting the CAN gateway module or by setting the OM bit in the GMR register to 0 in routing mode.

The CAN gateway module is configured in this mode. Only the checksum calculation on the routing table and the transmit FIFO check are enabled in this mode. Routing cannot be performed in this mode.

(3) Routing Mode

While in configuration mode, set the OM bit to 1 to enter routing mode.

Routing of the CAN message frames is performed in this mode. Routing is enabled/disabled for each channel by setting the GCCR register. The checksum calculation on the routing table and the transmit FIFO check cannot be performed.

26.3 Message Handler (Routing Unit)

26.3.1 Operational Overview

According to the routing table, the message handler routes messages received by the CAN module.

Major processing of the message handler are:

- Searching for entries from the routing table which match the IDE, RTR, and ID of a received message.
- Transferring the message to the transmit FIFO according to the content of the matched entry.

The following is the routing procedure for the message handler:

(1) Obtains the IDE, RTR, and ID from a received frame

(2) Performs a parity check on entry data when the RMPCE bit in the GPCCR register is 1 (1)

(3) Obtains the IDE, RTR, ID, and IDM from the entry data

(4) Compares the IDE, RTR, and ID with those of the received frame (proceed to the next entry and return to (2) if they do not match) (2)

(5) Obtains the information on the transmit FIFO of the destination (destination channel and transmit FIFO number) from the entry data

(6) Obtains the information of the channel to be processed from the entry data and verifies whether the receive channel is to be processed (3)

(7) Determines whether or not to transfer to the same channel

- When transfer to the same channel is enabled
  Transfers messages to all the corresponding transmit FIFOs according to the transmit FIFO information obtained
- When transfer to the same channel is disabled
  Transfers messages to transmit FIFOs other than the receive channel based on the obtained information on the transmit FIFO (4)

Notes:

1. Messages being processed are discarded when a parity error is detected.
2. Messages are discarded when there is no matched entry.
3. Messages are discarded when the message-received channel is not designated for processing.
4. The RES bit in the GSR register becomes 1 (routing error occurred) when there are no destination channels besides the receive channel.

When messages are received from multiple channels, they are processed from the channel with smaller channel number.

26.3.2 Routing Table

The information required for transferring a message frame is defined in the routing table. The routing table becomes undefined after a reset, and therefore needs to be configured after every reset.
A write/read access to the routing table is performed using registers GRMBP and GRMmH/GRMmL (m = 0 to 15). The GRMBP register specifies the block, and registers GRMmH/GRMmL access each entry. The routing table has 24 blocks (384 entries), and there are 16 entries per block.

Figure 26.47 shows a block diagram of accessing entries, and Table 26.6 lists the setting example of the routing table.

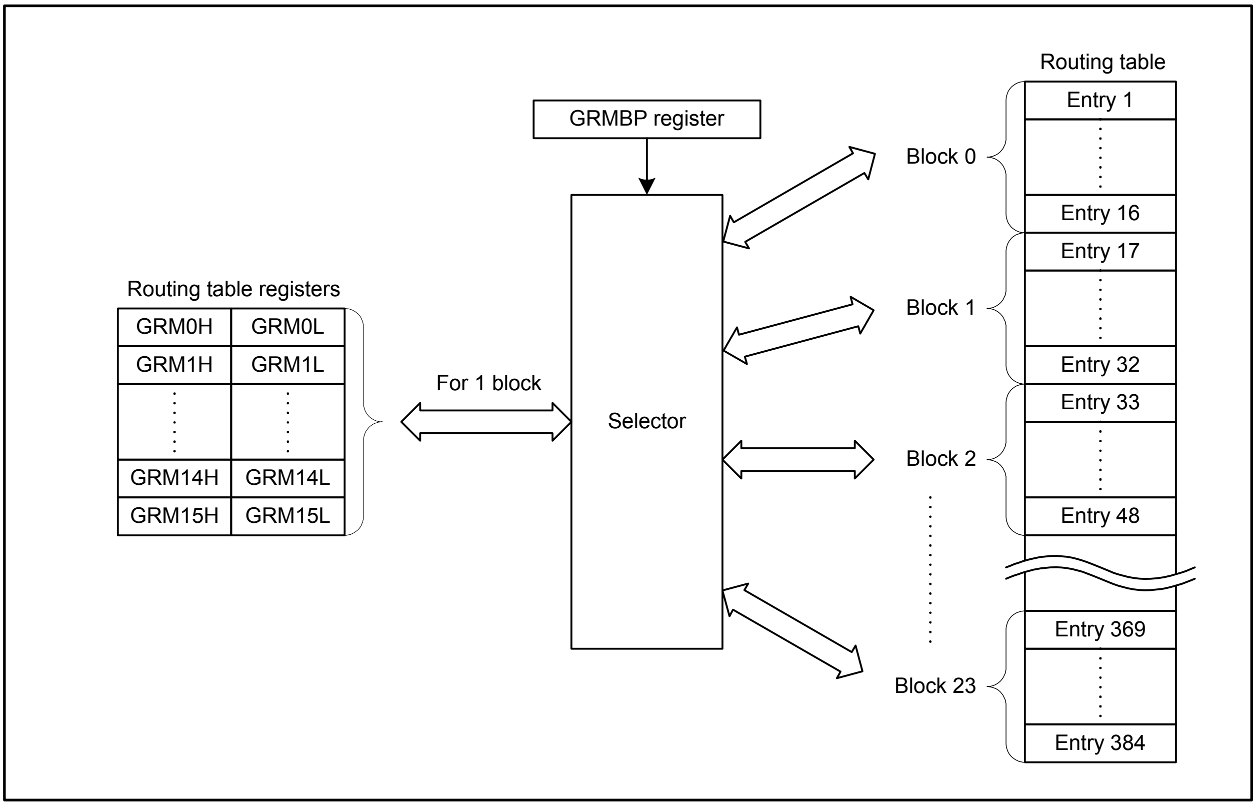


Figure 26.47 Block Diagram for Accessing Entries

Table 26.6 Routing Table Setting Example in the R32C/145 Group (m = 0 to 15)

	GRMmH Register							GRMmL Register			
	PAR 1 bit (1)	FIFO 1 bit (2)	DCH5 to DCH0 6 bits	LA1 8 bits	LA0 2 bits	PCH5 to PCH0 6 bits	IDM 8 bits (3)	PAR 1 bit (1)	IDE 1 bit (4)	RTR 1 bit (4)	ID 29 bits (4, 5)
Entry 1	0	1	11 1111b	00h	00b	11 1110b	00h	0	0	0	0000 0001h
Entry 2	0	1	00 1111b	01h	00b	11 0000b	00h	0	0	0	0000 0002h
Entry 3	0	1	00 0001b	02h	01b	00 0010b	00h	0	0	0	0000 0010h
Entry 4	0	1	00 0010b	03h	01b	00 0001b	00h	0	0	0	0000 0011h
Entry 5	0	0	00 0011b	04h	01b	00 1000b	00h	0	0	0	0000 0020h
Entry 6	0	1	00 0001b	05h	10b	00 1000b	00h	0	0	0	0000 0150h
Entry 7	0	1	00 0111b	06h	10b	11 1000b	00h	0	0	0	0000 0256h
Entry 8	0	0	00 0100b	07h	01b	00 0011b	00h	0	0	0	0000 0333h
Entry 9	0	1	00 1010b	08h	01b	00 0101b	00h	0	0	0	0000 0366h
Entry 10	0	0	00 0010b	09h	01b	00 0001b	0Fh	0	0	0	0000 0400h
Entry 11	0	1	00 0001b	10h	00b	00 0010b	00h	0	0	0	0000 0621h
Entry 12	0	0	10 1100b	11h	00b	00 0011b	00h	0	0	0	0000 0622h
Entry 13	0	1	01 0010b	12h	10b	00 0100b	00h	0	0	0	0000 0623h
Entry 14	0	1	00 0110b	13h	10b	00 0001b	00h	0	0	0	0000 0651h
Entry 15	0	0	00 0110b	14h	11b	00 1000b	00h	0	0	0	0000 0660h
Entry 16	0	1	00 1010b	15h	11b	01 0000b	00h	0	0	0	0000 0681h
Entry 17	0	1	00 1011b	16h	01b	00 0100b	00h	0	0	0	0000 06F3h
Entry 18	0	1	00 0110b	17h	10b	01 1000b	7Fh	0	0	0	0000 0700h
:	:	:	:	:	:	:	:	:	:	:	:
Entry 379	0	0	01 1111b	23h	10b	10 0000b	00h	0	1	0	10B0 00F6h
Entry 380	0	0	00 0101b	24h	10b	11 1010b	00h	0	1	0	130A 0005h
Entry 381	0	0	10 0000b	25h	11b	01 1111b	00h	0	1	0	1F00 1000h
Entry 382	0	0	00 1110b	26h	00b	11 0001b	00h	0	1	1	12B0 000Fh
Entry 383	0	0	11 0001b	27h	01b	00 0110b	00h	0	1	1	1F00 0030h
Entry 384	0	1	00 0001b	28h	11b	00 0110b	00h	0	1	1	1F00 F000h

Notes:

1. The parity (PAR) is calculated when a value is set in registers GRMmH and GRMmL.
2. The transmit FIFO number for an entry can be either 0 or 1.
3. The masking information can be applied to the lower 8 bits of the IDs. Set the minimum ID value to the ID field when multiple IDs are destined to be selected according to the masking information.
4. Set the lower 31 bits (IDE + RTR + ID) in ascending order, from entry 1 to entry 384, in the routing table.
5. Do not set the same value to other IDs.

Entry 10 is the setting example of the transfer to transmit FIFO0 of channel 1 when receiving a message frame of the data frame, standard ID, and message IDs from 400h to 40Fh on channel 0.
Entry 18 is the setting example of the transfer to transmit FIFO1 of channel 1 or channel 2 when receiving a message frame of the data frame, standard ID, and message IDs from 700h to 77Fh on channel 3 or channel 4.

26.3.3 Checksum Calculation for the Routing Table

After setting the routing table, the checksum value can be calculated. Set the CALC bit in the GRMCC register to 1 to start calculating the checksum value. Read bits SCS1 and SCS0 in the GSCFC register for the current calculation status and the GRMSR register for the calculation result (the checksum value for the routing table).

The following is the procedure for calculating the checksum value:

(1) Add routing table data for the number of entries set in the GMREC register. Each entry is 64-bit data, which is divided into 32-bit units for calculating. The value of the summation can be used to calculate the sum (A) and the carry (B). Note that the PAR bit is replaced with 0 for the calculation.

(2) Add the sum (A) and the carry (B) to calculate the sum (C). The carry that occurs in this calculation is ignored.

(3) The checksum value is defined to be the inverted value of the sum (C).

Figure 26.48 shows the checksum calculation for the routing table.

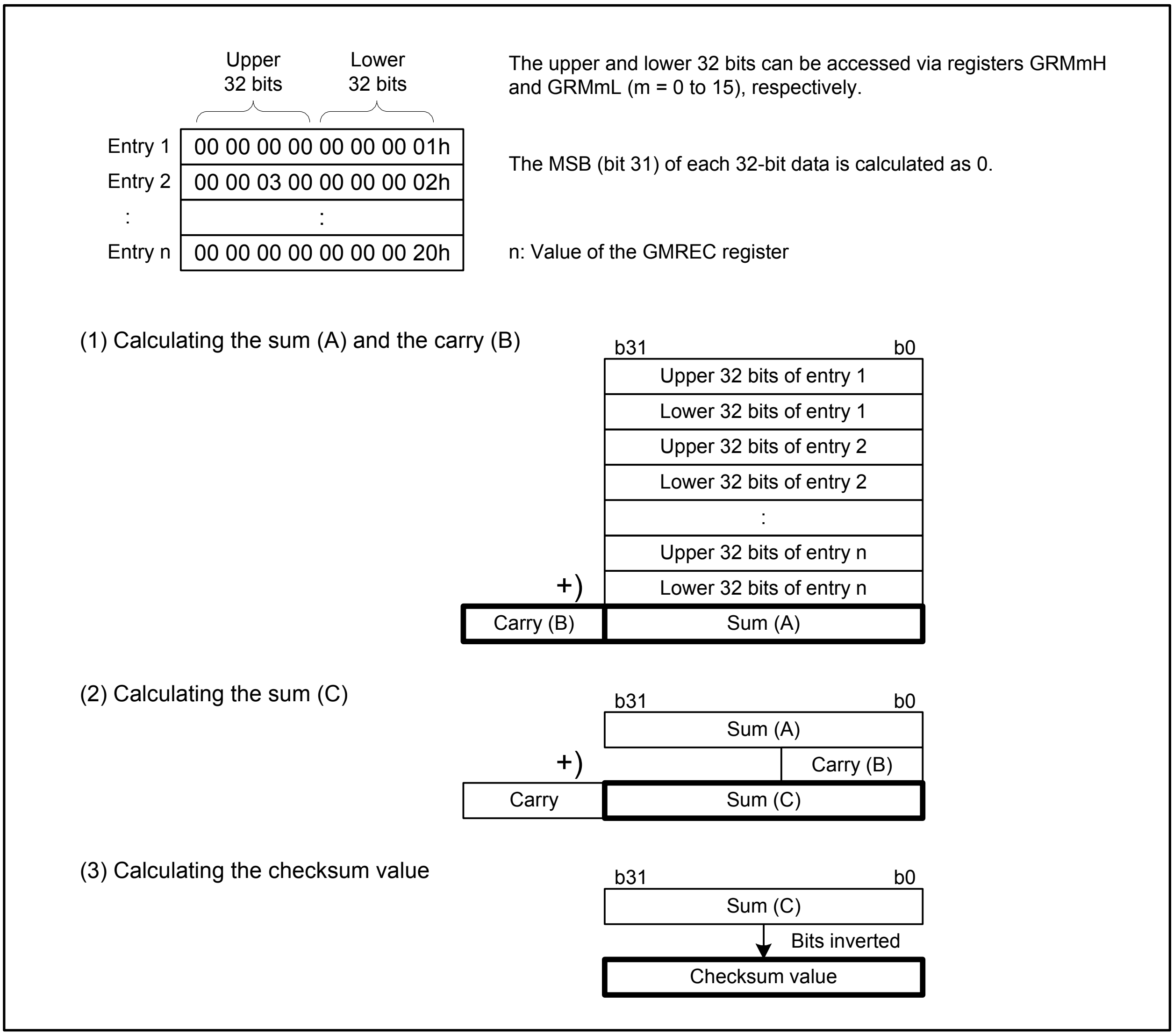


Figure 26.48 Checksum Calculation for the Routing Table

26.4 Transmit FIFO

Each CAN channel has two 32 blocks of transmit FIFOs (FIFO0 and FIFO1), and each block consists of 16 bytes (one frame).

When there are messages in the transmit FIFOs, the first frame stored is transferred from the transmit FIFOs to registers GFRR0 to GFRR3 by setting bits FIFO and CH2 to CH0 in the GTFRC register, or by reading the GFRR3 register. The BUSY bit in the GTFRS register becomes 1 during the transfer. The next frame transfer starts if there are any messages left in the transmit FIFOs when reading the GFRR3 register. Read the GFijFL register to verify the number of frames stored in transmit FIFOj of channel i.

Figure 26.49 shows the transmit FIFO block structure.

Standard ID

IDE is 0	RTR	EPC	Standard ID	0	Data	LA0	Reserved	DLC	LA1	Data	TSL	Data
1 bit	1 bit	1 bit	11 bits	18 bits	16 bits	2 bits	2 bits	4 bits	8 bits	32 bits	16 bits	16 bits

Extended ID

IDE is 1	RTR	EPC	Extended ID	Data	LA0	Reserved	DLC	LA1	Data	TSL	Data
1 bit	1 bit	1 bit	29 bits	16 bits	2 bits	2 bits	4 bits	8 bits	32 bits	16 bits	16 bits

Figure 26.49 Transmit FIFO Block Structure

26.4.1 Transmit FIFO Check

This function checks a read/write access to the transmit FIFO.

The transmit FIFO check starts by setting the CHK bit in the GTFCC register to 1 when the GCSR register is 00h and also when the module is in configuration mode.

Read bits FCS1 and FCS0 in the GSCFC register for the result of the transmit FIFO check.

26.4.2 Parity Check

The parity information is added when a message frame is transferred to the transmit FIFO. The parity check function can be enabled/disabled by setting the GPCCR register.

Read the EPC bit in the GFRR0 register for the result of the parity check.

26.4.3 Transmit FIFO Associated Interrupts

26.4.3.1 Transmit FIFO Critical Level Interrupt

When the number of frames stored in the transmit FIFOj of channel i exceeds the number set in the GFijCL register, a transmit FIFO critical level interrupt request is generated.

Figure 26.50 shows the timing for the transmit FIFO critical level interrupt request to be generated.

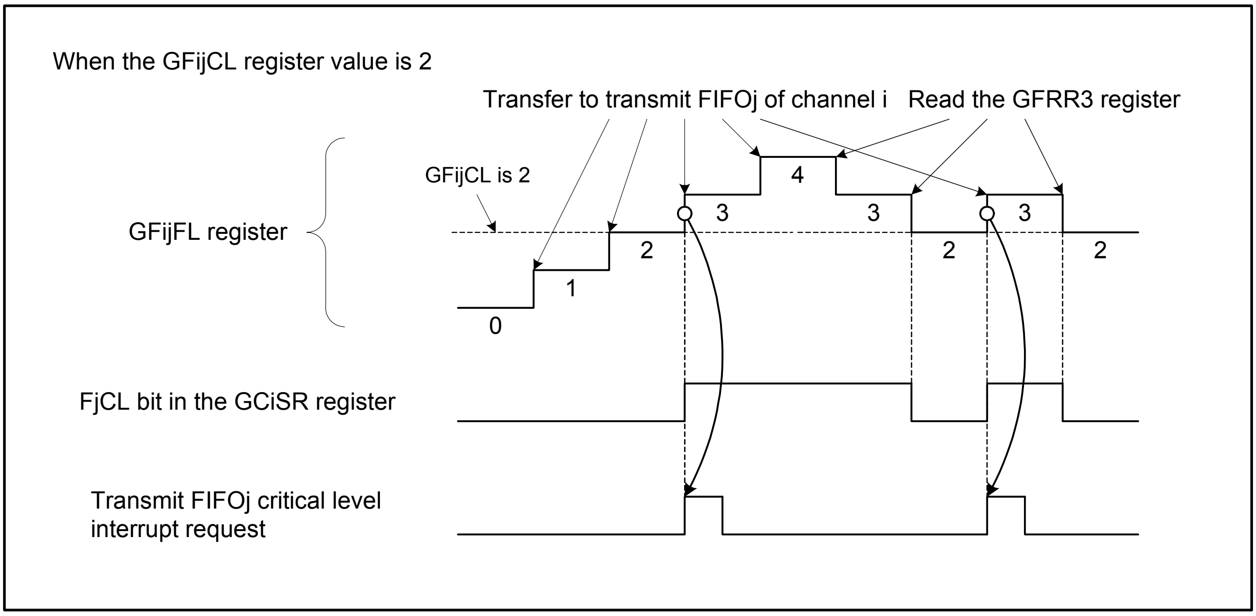


Figure 26.50 Transmit FIFO Critical Level Interrupt Request Generation Timing

26.4.3.2 Transmit FIFO Overflow Interrupt

If there are already 32 message frames stored in the corresponding FIFO when transferring message frames to the transmit FIFO, the most recent frame is stored after discarding the frame which is stored first. The FjO bit becomes 1 (overflow of the transmit FIFOj) and then an interrupt request is generated.

When an overflow occurs, the number of discarded frames cannot be verified. Moreover, the data in the transmit FIFO in which an overflow occurs is not guaranteed.

26.5 CAN Gateway Interrupts

The CAN gateway module has two interrupts listed below.

(1) CANi Gateway Transmit FIFO Interrupt

There are four interrupt sources for this interrupt. Read the GCiSR register to check which interrupt source has generated an interrupt request.

- Channel i transmit FIFO0 critical level
- Channel i transmit FIFO1 critical level
- Channel i transmit FIFO0 overflow
- Channel i transmit FIFO1 overflow

(2) CAN Gateway Error Interrupt

There are two interrupt sources for this interrupt. Read the GSR register to check which interrupt source has generated an interrupt request.

- Routing error
- Hardware error

26.6 Setting Procedure

26.6.1 Setting the CAN Gateway Module

The following procedure is an example of the initial setting for the CAN gateway module:

(1) Entering configuration mode

Write 0000 0000b to the GMR register when it is 0000 0001b (routing mode). After it becomes 0000 0000b, verify that the GCSR register becomes 00h (routing is stopped for all channels).
When the GMR register is 0000 0010b (gateway reset mode), wait until it becomes 0000 0000b.

(2) Setting the routing table

(a) Selecting the base pointer of the routing table
Set the block number to the GRMBP register.

(b) Setting the entry data (within the same block)
Set the necessary number of entry data repeatedly from the same block to registers GRMmL and GRMmH (m = 0 to 15).
Start again from (a) when setting entry data from different blocks.

(c) Setting the number of entries in the routing table
Set the number of entries - 1 to the GMREC register.

(3) Performing the checksum calculation (if needed)

(a) Starting the checksum calculation
Write 1 to the CALC bit in the GRMCC register.

(b) Waiting for the checksum calculation to be completed
Wait until bits SCS1 and SCS0 in the GSCFC register become 10b.

(c) Confirming the result of the checksum calculation
Confirm that the result stored in the GRMSR register is the same as the value expected.

(4) Performing the transmit FIFO check (if needed)

(a) Starting the transmit FIFO check
Write 1 to the CHK bit in the GTFCC register.

(b) Waiting for and then confirming that the transmit FIFO check is completed
Wait until bits FCS1 and FCS0 in the GSCFC register become 10b or 11b.
Fail-safe operations must be performed when bits FCS1 and FCS0 become 11b, which indicates a failure on the transmit FIFO check.

(5) Setting the transmit FIFO interrupts

(a) Setting the FIFO critical level
Set the value of the FIFO critical level - 1 to the GFijCL register.

(b) Enabling/disabling the transmit FIFO interrupts
Enable/disable the transmit FIFOj critical level interrupt with the FjCLIE bit in the GCiIE register and the transmit FIFOj overflow interrupt with the FjOE bit.

(6) Enabling/disabling the error interrupts

Enable/disable the routing error interrupt with the RERE bit in the GIER register and the hardware error interrupt with the HERE bit.

(7) Enabling/disabling the parity check

Enable/disable the parity check on channel i with the PCEi bit in the GPCCR register and on the routing table with the RMPCE bit.

(8) Enabling/disabling the routing

Enable/disable routing channel i with the CHiE bit in the GCCR register.

(9) Enabling/disabling the echo-back
Enable/disable transfer to the same channel with the ECHO bit in the GEBCR register.

(10) Setting the time stamp timer
(a) Setting the count source for the time stamp timer
Set bits DIV2 to DIV0 in the GTSCR register to select the divide ratio for the count source.
(b) Starting the time stamp timer
Write 1 (start counter) to the CST bit in the GTSCR register.

(11) Transiting to routing mode
Write 1 (routing mode) to the OM bit in the GMR register.

(12) Verifying the state of the routing-enabled channel
Verify that the routing-enabled channel, set in (8), is enabled with the CHiS bit in the GCSR register.

(13) Initializing the CAN module
Initialize CAN modules of the channels to be used.
After the initialization, the CAN gateway module starts routing by receiving frames.

26.6.2 CAN Module Setting

This section describes the CAN module settings in order to use the gateway function.

The CAN modules automatically transfer CAN frames received into mailboxes [12] to [15] to the CAN gateway module by setting the corresponding bits to be routing enabled.

The following describes the settings of the CAN associated registers required when applying the CAN gateway module.

CAN associated registers not specified in this section can be arbitrarily set.

26.6.2.1 CANi Control Register (CiCTLR) Setting

Set the CANi control register as follows in order to apply the gateway function.

Table 26.7 CANi Control Register Setting Values

Register	Bit	Function
CiCTLR	CANM	Set this bit to 00b (CAN operation mode) (1)
	SLPM	Set this bit to 0 (mode other than CAN sleep mode) (1)
	BOM	Select the bus-off recovery mode
	RBOC	Set this bit to 1 for the forced recovery from bus-off
	b7 and b6	Set these bits to 00b
	MBM	Select mailbox mode (2)
	IDFM	Select the ID format
	MLM	Set this bit to 0 (overwrite mode) (3)
	TPM	Select the transmit priority mode
	TSRC	Set this bit to 1 when resetting the time stamp counter
	TSPS	Select the time stamp prescaler

Notes:

1. CAN operating modes can be changed anytime. However, transfer to the CAN gateway module stops when the mode selected is not CAN operation mode.
2. The MBM bit setting may change the settings of the message control registers and the mask registers.
3. This setting is required to enable the continuos message transfer to the gateway mailbox.

26.6.2.2 Mailbox Setting

The gateway function can be applied in both normal mailbox mode and FIFO mailbox mode. When it is applied, mailboxes [12] to [15] are exclusively used for gateway reception in both modes.

Table 26.8 Mailbox Setting

Mailbox	MBM Bit is 0 (Normal Mailbox Mode)	MBM Bit is 1 (FIFO Mailbox Mode)
Mailboxes [0] to [7]	Normal mailbox	Normal mailbox
Mailboxes [8] to [11]		Transmit FIFO
Mailboxes [12] to [15]	Gateway receive mailbox	Gateway receive FIFO

26.6.2.3 Register Setting in Normal Mailbox Mode (When the MBM Bit is 0)

Set the registers as shown in Table 26.9 in normal mailbox mode (when the MBM bit is 0).

Table 26.9 Registers and Setting Value Used in Normal Mailbox Mode (When the MBM Bit is 0)

Register	Bit	Function
CiMKR3	EID	Set all of these bits to 0 (corresponding EID bit is not compared)
	SID	Set all of these bits to 0 (corresponding SID bit is not compared)
	b31 to b29	Set these bits to 000b
CiMKIVLR	b11 to b0	Set these bits according to mailbox usage
	b15 to b12	Set all these bits to 0 (mask valid)
CiMB12 to CiMB15	EID	The lower 18 bits of the extended ID received are stored
	SID	The lower 11 bits of the extended ID received or standard ID received are stored
	RTR (1)	Set this bit to 1 when selecting the remote frame
	IDE (1)	Set this bit to 1 when selecting the extended ID
CiMIER	b11 to b0	Set these bits according to mailbox usage
	b15 to b12	Set this bit to 1 (interrupt enabled) (2)
CiMCTL12 to CiMCTL15	NEWDATA	Reception complete flag
	INVALDATA	Reception-in-progress status flag
	MSGLOST	Message lost flag
	b3	Set this bit to 0
	ONESHOT	Set this bit to 0 (one-shot reception or one-shot transmission disabled)
	b5	Set this bit to 0
	RECREQ	Set this bit to 1 (configured for reception)
	TRMREQ	Set this bit to 0 (not configured for transmission)

Notes:

1. The ID format (standard ID or extended ID) and the frame format (data frame or remote frame) for the receive frame can be selected by setting bits RTR and IDE. Set four individual mailboxes with different values in order to receive all four types of frames.
2. A receive complete interrupt is not generated. This is the setting for the CAN gateway module to be notified that the reception is completed.

26.6.2.4 Register Setting in FIFO Mailbox Mode (When the MBM Bit is 1)

Set the registers as shown in Table 26.10 in FIFO mailbox mode (when the MBM bit is 1).

Table 26.10 Registers and Setting Value Used in FIFO Mailbox Mode (When the MBM Bit is 1)

Register	Bit	Function
CiMKR2 and CiMKR3	EID	Set all of these bits to 0 (corresponding EID bit is not compared)
	SID	Set all of these bits to 0 (corresponding SID bit is not compared)
	b31 to b29	Set these bits to 000b
CiFIDCR0 and CiFIDCR1	EID	Set all of these bits to 0 (corresponding EID bit is 0)
	SID	Set all of these bits to 0 (corresponding SID bit is 0)
	b29	Set this bit to 0
	RTR (1)	Set this bit to 1 when selecting the remote frame
	IDE (1)	Set this bit to 1 when selecting the extended ID
CiMKIVLR	b11 to b0	Set these bits according to mailbox usage
	b15 to b12	Set all these bits to 0 (mask valid)
CiMIER	b7 to b0	Set these bits according to mailbox usage
	b8	Set this bit to 1 when enabling the transmit FIFO interrupt
	b9	Set the transmit FIFO interrupt generation timing
	b11 and b10	Set these bits to 00b
	b12	Set this bit to 1 (interrupt enabled) (2)
	b13	Set this bits to 0 (every time reception is completed)
	b15 and b14	Set these bits to 00b
CiRFCR	RFE	Set this bit to 1 (receive FIFO enabled)
	RFUST	Receive FIFO unread message number status flag
	RFMLF	Receive FIFO message lost flag
	RFFST	Receive FIFO full status flag
	RFWST	Receive FIFO buffer warning status flag
	RFEST	Receive FIFO empty status flag

Notes:

1. The ID format (standard ID or extended ID) and the frame format (data frame or remote frame) for the receive frame can be selected by setting bits RTR and IDE. Two types of frames out of four combinations can be selected.
2. A receive complete interrupt is not generated. This is the setting for the CAN gateway module to be notified that the reception is completed.

26.7 Bit Search Support Function

The CAN gateway module has a function to support searching 1 in a bit sequence. This function is independent from other functions of the CAN gateway module; therefore, associated registers (registers GBSCn, GBSRn, and GBSSn (n = 0, 1)) are not reset even if the RST bit in the GMR register is set to 1.

26.7.1 Operation of the Bit Search Support Function

Once a 16-bit sequence is set to the GBSRn register after setting the bit search mode with the SMD bit in the GBSCn register, the SCPL bit in the GBSSn register becomes 0 (search is in progress) and the bit position of the least significant 1 is stored in bits BP3 to BP0 in the GBSSn register. The least significant 1 in the GBSRn register is rewritten to 0 at this point.

The bit search is repeated when there are multiple 1s in the GBSRn register. When the SMD bit in the GBSCn register is 0 (auto search mode), the next search starts when reading this register. When the SMD bit is 1 (manual search mode), the next search starts by writing a value to this register after reading this register. The value written to this register is ignored.

When the last 1 is detected, the bit number is stored in bits BP3 to BP0 and the SCPL bit becomes 1 (search is completed). The last 1 is rewritten to 0, and the GBSRn register becomes 0000h.

26.7.2 Example of Using the Bit Search Support Function

The following is an operation example of the bit search on 0108h (0000 0001 0000 1000b) using the GBSR0 register.

(1) The first search starts when writing 0108h to the GBSR0 register.

(2) The search starts from the LSB. Once 1 is detected, the search stops and the GBSS0 register becomes 03h (bit 3 is 1, search is in progress) as a search result. Since bits which have been detected become 0, the GBSR0 register becomes 0100h.

(3) In auto search mode, the next search starts by reading the GBSS0 register. In manual search mode, however, the next search starts by writing a value to the GBSS0 register after reading the GBSS0 register. The value written to the GBSS0 register is ignored in this case.

(4) Bit 8 is the only bit that is 1 at this point; therefore, the search is completed on bit 8. The values of the registers are 0000h for the GBSR0 register and 88h (bit 8 is 1, search is completed) for the GBSS0 register.

27. I/O Pins

Each pin of the MCU functions as a programmable I/O port or an I/O pin for internal peripheral functions. These functions can be switched by the function select registers. The pull-up resistors are enabled for every group of four pins. However, a pull-up resistor is separated from other peripheral functions even if it is enabled, when a pin functions as an output pin.
Figure 27.1 shows a block diagram of typical I/O pin.

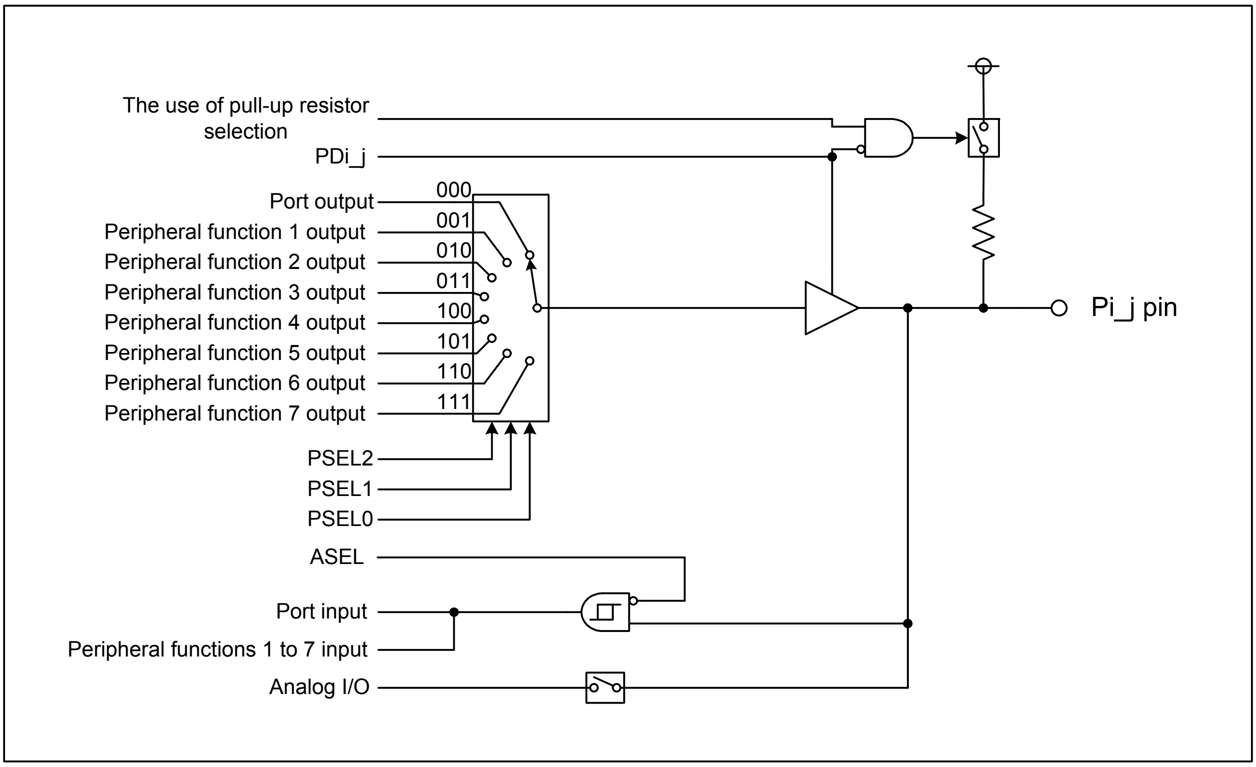


Figure 27.1 Typical I/O Pin Block Diagram (i = 0 to 10; j = 0 to 7)

The registers to control I/O pins are as follows: port Pi direction register (PDi register), output function select register, and pull-up control register. The PDi register selects the input or output state of pins. The output function select register which selects an output function consists of bits PSEL2 to PSEL0, and ASEL. Bits PSEL2 to PSEL0 select a function as a programmable I/O or peripheral function output (except analog output). The ASEL bit prevents the increase in power consumption of input buffer caused by an intermediate potential when a pin functions as an analog I/O pin. The pull-up control register enables/disables the pull-up resistors.
To use a pin as an analog I/O pin, the PDi_j bit should be set to 0 (input), and bits PSEL2 to PSEL0 should be set to 000b and the ASEL bit should be set to 1.
The input-only port P8_5, which shares a pin with $\overline{\text{NMI}}$ has neither bit 5 of the function select register nor the PDi register. Port P9_1 also functions as an input-only port. Bit 1 of the function select register and the PDi register are reserved. Port P9 is protected from unexpected write accesses by the PRC2 bit in the PRCR register. Ports P3, P7, and P8 are protected from unexpected write accesses by the PRC30 bit in the PRCR3 register (refer to 9. “Protection”).

27.1 Port Pi Direction Register (PDi Register, i = 0 to 10)

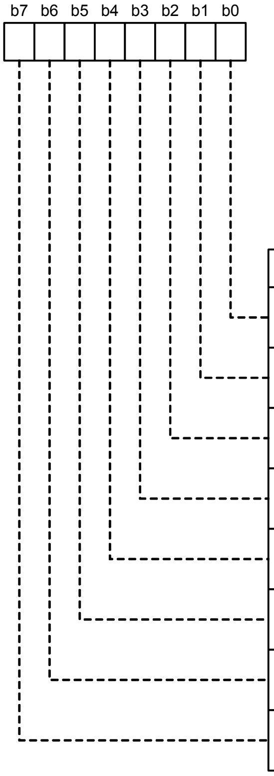
The PDi register selects the input or output state of pins. Bits in this register correspond to respective pins.

Figure 27.2 shows the PDi register.

No register bit is provided for port P8_5. For port P9_1, a reserved bit is provided.

The PD9 register is protected from unexpected write accesses by setting the PRC2 bit in the PRCR register. Registers PD3, PD7, and PD8 are protected by the PRC30 bit in the PRCR3 register (refer to 9. “Protection”).

Port Pi Direction Register (i = 0 to 10)



Symbol	Address	Reset Value
PD0, PD1, PD2	03C2h, 03C3h, 03C6h	0000 0000b
PD3 (1)	03C7h	0000 0000b
PD4, PD5, PD6	03CAh, 03CBh, 03CEh	0000 0000b
PD7 (1)	03CFh	0000 0000b
PD8 (1, 2)	03D2h	00X0 0000b
PD9 (3, 4)	03D3h	0000 0000b
PD10	03D6h	0000 0000b

Bit Symbol	Bit Name	Function	RW
PDi_0	Port Pi_0 Direction Bit (3)	0: Input port 1: Output port	RW
PDi_1	Port Pi_1 Direction Bit (3)	0: Input port 1: Output port	RW
PDi_2	Port Pi_2 Direction Bit (3)	0: Input port 1: Output port	RW
PDi_3	Port Pi_3 Direction Bit	0: Input port 1: Output port	RW
PDi_4	Port Pi_4 Direction Bit	0: Input port 1: Output port	RW
PDi_5	Port Pi_5 Direction Bit (2)	0: Input port 1: Output port	RW
PDi_6	Port Pi_6 Direction Bit	0: Input port 1: Output port	RW
PDi_7	Port Pi_7 Direction Bit	0: Input port 1: Output port	RW

Notes:

1. Registers PD3, PD7, and PD8 should be rewritten after the PRC30 bit in the PRCR3 register is set to 1 (write enabled).
2. The PD8_5 bit in the PD8 register is unavailable on this MCU. This bit should be written with 0 and read as an undefined value.
3. Bits PD9_0 to PD9_2 in the PD9 register are reserved. These bits should be written with 0.
4. Set the PRC2 bit in the PRCR register to 1 (write enabled) just before rewriting the PD9 register. No interrupt handling or DMA transfers should be inserted between these two instructions.

Figure 27.2 Registers PD0 to PD10

27.2 Output Function Select Register

When a programmable I/O port and peripheral function output share a pin, this register selects the output function of the pin. Regardless of the register setting, a signal is input to all the connected peripheral functions.

The output function select register consists of bits PSEL2 to PSEL0, and ASEL. Bits PSEL2 to PSEL0 select a function as programmable I/O or peripheral function output (except analog output). The ASEL bit prevents the increase in power consumption caused by an intermediate potential generated when a pin functions as an analog I/O pin.

Table 27.1 shows the peripheral functions assigned to each PSEL2 to PSEL0 bit combination, and Figure 27.3 to Figure 27.18 show the function select registers.

Note that ports P8_5 and P9_1 (input only) have no output function select registers.

The P9_iS register is protected from unexpected write accesses by setting the PRC2 bit in the PRCR register and registers P3_iS, P7_iS, and P8_iS are protected by the PRC30 bit in the PRCR3 register (refer to 9. “Protection”).

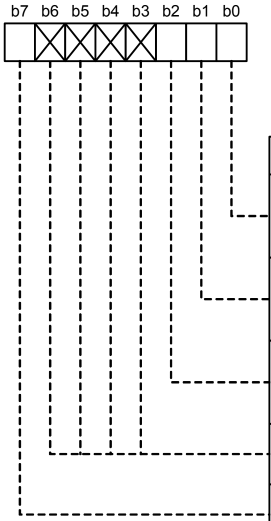
Table 27.1 Peripheral Function Assignment

Bits PSEL2 to PSEL0	Peripheral Function
001b	Timer
010b	Three-phase motor control timers, Serial bus interfaces 0 and 1
011b	UART
100b	UART special function, CAN channels 2, 3, 4, and 5 (1)
101b	Intelligent I/O group 0, CAN channel 0 (1), LIN module
110b	Intelligent I/O group 1, CAN channel 1 (1)
111b	

Note:

1. Channels CAN0, CAN1, and CAN4 are not available in the R32C/142 Group.

Port P0_i Function Select Register (i = 0 to 7)



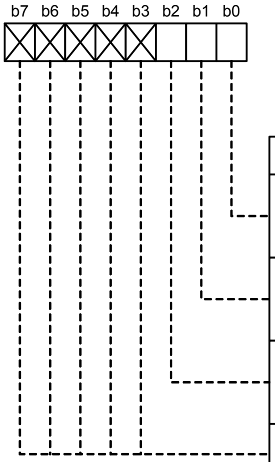
Symbol	Address	Reset Value
P0_0S to P0_2S	400A0h, 400A2h, 400A4h	0XXX X000b
P0_3S to P0_5S	400A6h, 400A8h, 400AAh	0XXX X000b
P0_6S, P0_7S	400ACh, 400AEh	0XXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P0_i Output Function Select Bit	b2 b1 b0 0 0 0 : I/O port P0_i 0 0 1 : Do not use this combination 0 1 0 : Do not use this combination 0 1 1 : Do not use this combination 1 0 0 : Do not use this combination 1 0 1 : Do not use this combination 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b6-b3)	No register bits; should be written with 0 and read as undefined value		—
ASEL	Port P0_i Analog Function Select Bit	0: Function other than AN0_i 1: AN0_i	RW

Figure 27.3 Registers P0_0S to P0_7S

Port P0_i (i = 0 to 7) shares a pin with the AN0_i input pin for the A/D converter.
To use it as a programmable I/O port, the P0_iS register should be set to 00h. To use it as an A/D converter input pin, this register should be set to 80h and the PD0_i bit should be set to 0 (port P0_i functions as an input port).

Port P1_i Function Select Register (i = 0 to 7)



Symbol	Address	Reset Value
P1_0S to P1_2S	400A1h, 400A3h, 400A5h	XXXX X000b
P1_3S to P1_5S	400A7h, 400A9h, 400ABh	XXXX X000b
P1_6S, P1_7S	400ADh, 400AFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P1_i Output Function Select Bit (1)	b2 b1 b0 0 0 0 : I/O port P1_i 0 0 1 : Do not use this combination 0 1 0 : SBI0 output 0 1 1 : Do not use this combination 1 0 0 : CAN3/CAN5 output 1 0 1 : IIO0_i output 1 1 0 : IIO1_i output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

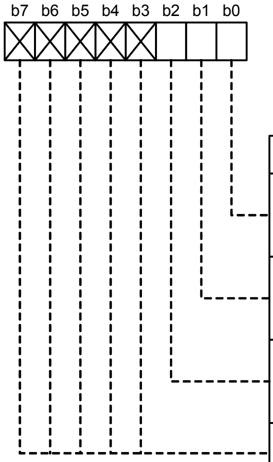
1. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P1_0	P1_0	— (2)	SSO0 output	— (2)	— (2)	IIO0_0 output	IIO1_0 output	— (2)
P1_1	P1_1	— (2)	SSCK0 output	— (2)	CAN3OUT	IIO0_1 output	IIO1_1 output	— (2)
P1_2	P1_2	— (2)	SSI0 output	— (2)	— (2)	IIO0_2 output	IIO1_2 output	— (2)
P1_3	P1_3	— (2)	$\overline{\text{SCS0}}$ output	— (2)	— (2)	IIO0_3 output	IIO1_3 output	— (2)
P1_4	P1_4	— (2)	— (2)	— (2)	— (2)	IIO0_4 output	IIO1_4 output	— (2)
P1_5	P1_5	— (2)	— (2)	— (2)	CAN5OUT	IIO0_5 output	IIO1_5 output	— (2)
P1_6	P1_6	— (2)	— (2)	— (2)	— (2)	IIO0_6 output	IIO1_6 output	— (2)
P1_7	P1_7	— (2)	— (2)	— (2)	— (2)	IIO0_7 output	IIO1_7 output	— (2)

2. Do not use this combination.

Figure 27.4 Registers P1_0S to P1_7S in the R32C/142 Group

Port P1_i Function Select Register (i = 0 to 7)



Symbol	Address	Reset Value
P1_0S to P1_2S	400A1h, 400A3h, 400A5h	XXXX X000b
P1_3S to P1_5S	400A7h, 400A9h, 400ABh	XXXX X000b
P1_6S, P1_7S	400ADh, 400AFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P1_i Output Function Select Bit [1]	b2 b1 b0 0 0 0 : I/O port P1_i 0 0 1 : Do not use this combination 0 1 0 : SBI0 output 0 1 1 : Do not use this combination 1 0 0 : CAN3/CAN4/CAN5 output 1 0 1 : IIO0_i output 1 1 0 : IIO1_i output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. Refer to the following table for each pin setting.

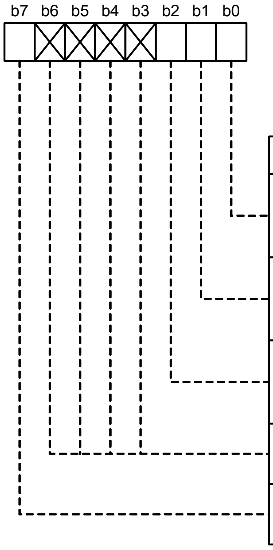
Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P1_0	P1_0	— [2]	SSO0 output	— [2]	— [2]	IIO0_0 output	IIO1_0 output	— [2]
P1_1	P1_1	— [2]	SSCK0 output	— [2]	CAN3OUT	IIO0_1 output	IIO1_1 output	— [2]
P1_2	P1_2	— [2]	SSI0 output	— [2]	— [2]	IIO0_2 output	IIO1_2 output	— [2]
P1_3	P1_3	— [2]	$\overline{\text{SCS0}}$ output	— [2]	CAN4OUT	IIO0_3 output	IIO1_3 output	— [2]
P1_4	P1_4	— [2]	— [2]	— [2]	— [2]	IIO0_4 output	IIO1_4 output	— [2]
P1_5	P1_5	— [2]	— [2]	— [2]	CAN5OUT	IIO0_5 output	IIO1_5 output	— [2]
P1_6	P1_6	— [2]	— [2]	— [2]	— [2]	IIO0_6 output	IIO1_6 output	— [2]
P1_7	P1_7	— [2]	— [2]	— [2]	— [2]	IIO0_7 output	IIO1_7 output	— [2]

2. Do not use this combination.

Figure 27.5 Registers P1_0S to P1_7S in the R32C/145 Group

Port P1_i (i = 0 to 7) shares a pin with the serial bus interface (SBI0), intelligent I/O groups 0 and 1 (IIO0 and IIO1), CAN module, and external interrupt input pin.
To use it as an output pin, the PD1_i bit should be set to 1 (port P1_i functions as an output port) and a function should be selected according to Figures 27.4 and 27.5. To use it as an input pin, the PD1_i bit should be set to 0 (port P1_i functions as an input port).

Port P2_i Function Select Register (i = 0 to 7)



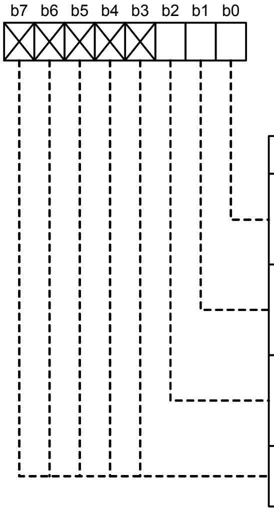
Symbol	Address	Reset Value
P2_0S to P2_2S	400B0h, 400B2h, 400B4h	0XXX X000b
P2_3S to P2_5S	400B6h, 400B8h, 400BAh	0XXX X000b
P2_6S, P2_7S	400BCh, 400BEh	0XXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P2_i Output Function Select Bit	b2 b1 b0 0 0 0 : I/O port P2_i 0 0 1 : Do not use this combination 0 1 0 : Do not use this combination 0 1 1 : Do not use this combination 1 0 0 : Do not use this combination 1 0 1 : Do not use this combination 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b6-b3)	No register bits; should be written with 0 and read as undefined value		—
ASEL	Port P2_i Analog Function Select Bit	0: Function other than AN2_i 1: AN2_i	RW

Figure 27.6 Registers P2_0S to P2_7S

Port P2_i (i = 0 to 7) shares a pin with the AN2_i pin for the A/D converter.
To use it as a programmable I/O port, the P2_iS register should be set to 00h. To use it as an A/D converter input pin, this register should be set to 80h and the PD2_i bit should be set to 0 (port P2_i functions as an input port).

Port P3_i Function Select Register (i = 0 to 7) (1)



Symbol	Address	Reset Value
P3_0S to P3_2S	400B1h, 400B3h, 400B5h	XXXX X000b
P3_3S to P3_5S	400B7h, 400B9h, 400BBh	XXXX X000b
P3_6S, P3_7S	400BDh, 400BFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P3_i Output Function Select Bit (2)	b2 b1 b0 0 0 0 : I/O port P3_i 0 0 1 : Timer output 0 1 0 : Three-phase motor control output 0 1 1 : UART4 output 1 0 0 : Do not use this combination 1 0 1 : LIN0/LIN1 output 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. These registers should be rewritten after the PRC30 bit in the PRCR3 register is set to 1 (write enabled).
2. Refer to the following table for each pin setting.

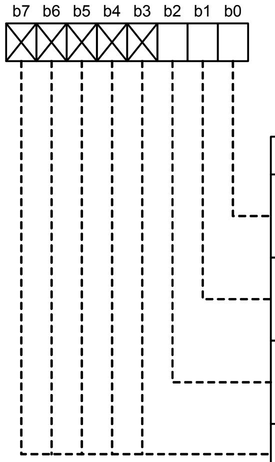
Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P3_0	P3_0	TA0OUT output	— (3)	— (3)	— (3)	LIN0OUT	— (3)	— (3)
P3_1	P3_1	TA3OUT output	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P3_2	P3_2	TA1OUT output	V	— (3)	— (3)	LIN1OUT	— (3)	— (3)
P3_3	P3_3	— (3)	$\overline{\text{V}}$	— (3)	— (3)	— (3)	— (3)	— (3)
P3_4	P3_4	TA2OUT output	W	CLK4 output	— (3)	— (3)	— (3)	— (3)
P3_5	P3_5	— (3)	$\overline{\text{W}}$	— (3)	— (3)	— (3)	— (3)	— (3)
P3_6	P3_6	TA4OUT output	U	TXD4	— (3)	— (3)	— (3)	— (3)
P3_7	P3_7	— (3)	$\overline{\text{U}}$	$\overline{\text{RTS4}}$	— (3)	— (3)	— (3)	— (3)

3. Do not use this combination.

Figure 27.7 Registers P3_0S to P3_7S

Port P3_i (i = 0 to 7) shares a pin with the timer output, three-phase motor control output, serial interface (UART4), and LIN module.
To use it as an output pin, the PD3_i bit should be set to 1 (port P3_i functions as an output port) and a function should be selected according to Figure 27.7. To use it as an input pin, the PD3_i bit should be set to 0 (port P3_i functions as an input port).

Port P4_i Function Select Register (i = 0 to 7)



Symbol	Address	Reset Value
P4_0S to P4_2S	400C0h, 400C2h, 400C4h	XXXX X000b
P4_3S to P4_5S	400C6h, 400C8h, 400CAh	XXXX X000b
P4_6S, P4_7S	400CCh, 400CEh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P4_i Output Function Select Bit (1)	b2 b1 b0 0 0 0 : I/O port P4_i 0 0 1 : Do not use this combination 0 1 0 : SBI1 output 0 1 1 : UART3 output 1 0 0 : Do not use this combination 1 0 1 : LIN0/LIN1 output 1 1 0 : IIO1_i output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P4_0	P4_0	— (2)	— (2)	$\overline{\text{RTS3}}$	— (2)	— (2)	IIO1_0 output	— (2)
P4_1	P4_1	— (2)	— (2)	CLK3 output	— (2)	— (2)	IIO1_1 output	— (2)
P4_2	P4_2	— (2)	— (2)	— (2)	— (2)	— (2)	IIO1_2 output	— (2)
P4_3	P4_3	— (2)	— (2)	TXD3	— (2)	— (2)	IIO1_3 output	— (2)
P4_4	P4_4	— (2)	SSO1 output	— (2)	— (2)	LIN0OUT	IIO1_4 output	— (2)
P4_5	P4_5	— (2)	SSCK1 output	— (2)	— (2)	— (2)	IIO1_5 output	— (2)
P4_6	P4_6	— (2)	SSI1 output	— (2)	— (2)	LIN1OUT	IIO1_6 output	— (2)
P4_7	P4_7	— (2)	$\overline{\text{SCS1}}$ output	— (2)	— (2)	— (2)	IIO1_7 output	— (2)

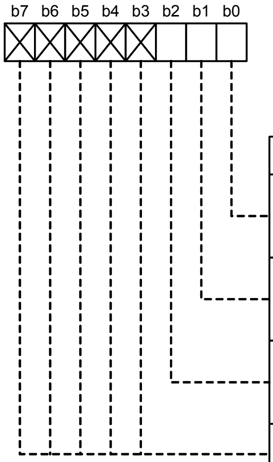
2. Do not use this combination.

Figure 27.8 Registers P4_0S to P4_7S

Port P4_i (i = 0 to 7) shares a pin with the serial interface (UART3), serial bus interface (SBI1), LIN module, and intelligent I/O group 1 (IIO1).

To use it as an output pin, the PD4_i bit should be set to 1 (port P4_i functions as an output port) and a function should be selected according to Figure 27.8. To use it as an input pin, the PD4_i bit should be set to 0 (port P4_i functions as an input port).

Port P5_i Function Select Register (i = 0 to 7)



Symbol	Address	Reset Value
P5_0S to P5_2S	400C1h, 400C3h, 400C5h	XXXX X000b
P5_3S to P5_5S	400C7h, 400C9h, 400CBh	XXXX X000b
P5_6S, P5_7S	400CDh, 400CFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P5_i Output Function Select Bit (1)	b2 b1 b0 0 0 0 : I/O port P5_i 0 0 1 : Do not use this combination 0 1 0 : Do not use this combination 0 1 1 : Do not use this combination 1 0 0 : CAN3/CAN5 output 1 0 1 : IIO0_i output 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

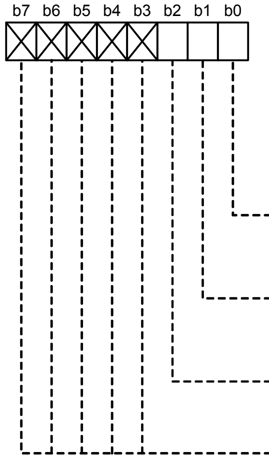
1. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P5_0	P5_0	— (2)	— (2)	— (2)	— (2)	IIO0_0 output	— (2)	— (2)
P5_1	P5_1	— (2)	— (2)	— (2)	— (2)	IIO0_1 output	— (2)	— (2)
P5_2	P5_2	— (2)	— (2)	— (2)	CAN5OUT	IIO0_2 output	— (2)	— (2)
P5_3	P5_3	— (2)	— (2)	— (2)	— (2)	IIO0_3 output	— (2)	— (2)
P5_4	P5_4	— (2)	— (2)	— (2)	— (2)	IIO0_4 output	— (2)	— (2)
P5_5	P5_5	— (2)	— (2)	— (2)	— (2)	IIO0_5 output	— (2)	— (2)
P5_6	P5_6	— (2)	— (2)	— (2)	CAN3OUT	IIO0_6 output	— (2)	— (2)
P5_7	P5_7	— (2)	— (2)	— (2)	— (2)	IIO0_7 output	— (2)	— (2)

2. Do not use this combination.

Figure 27.9 Registers P5_0S to P5_7S in the R32C/142 Group

Port P5_i Function Select Register (i = 0 to 7)



Symbol	Address	Reset Value
P5_0S to P5_2S	400C1h, 400C3h, 400C5h	XXXX X000b
P5_3S to P5_5S	400C7h, 400C9h, 400CBh	XXXX X000b
P5_6S, P5_7S	400CDh, 400CFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P5_i Output Function Select Bit (1)	b2 b1 b0 0 0 0 : I/O port P5_i 0 0 1 : Do not use this combination 0 1 0 : Do not use this combination 0 1 1 : Do not use this combination 1 0 0 : CAN3/CAN4/CAN5 output 1 0 1 : IIO0_i output 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. Refer to the following table for each pin setting.

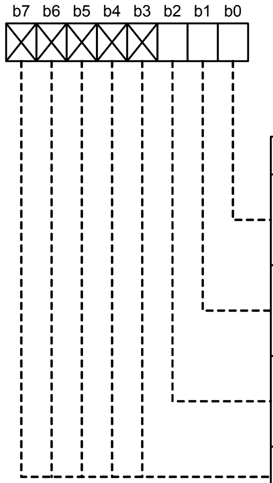
Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P5_0	P5_0	— (2)	— (2)	— (2)	— (2)	IIO0_0 output	— (2)	— (2)
P5_1	P5_1	— (2)	— (2)	— (2)	— (2)	IIO0_1 output	— (2)	— (2)
P5_2	P5_2	— (2)	— (2)	— (2)	CAN5OUT	IIO0_2 output	— (2)	— (2)
P5_3	P5_3	— (2)	— (2)	— (2)	— (2)	IIO0_3 output	— (2)	— (2)
P5_4	P5_4	— (2)	— (2)	— (2)	CAN4OUT	IIO0_4 output	— (2)	— (2)
P5_5	P5_5	— (2)	— (2)	— (2)	— (2)	IIO0_5 output	— (2)	— (2)
P5_6	P5_6	— (2)	— (2)	— (2)	CAN3OUT	IIO0_6 output	— (2)	— (2)
P5_7	P5_7	— (2)	— (2)	— (2)	— (2)	IIO0_7 output	— (2)	— (2)

2. Do not use this combination.

Figure 27.10 Registers P5_0S to P5_7S in the R32C/145 Group

Port P5_i (i = 0 to 7) shares a pin with intelligent I/O group 0 (IIO0) and the CAN module.
To use it as an output pin, the PD5_i bit should be set to 1 (port P5_i functions as an output port) and a function should be selected according to Figures 27.9 and 27.8. To use it as an input pin, the PD5_i bit should be set to 0 (port P5_i functions as an input port).

Port P6_i Function Select Register (i = 0 to 7)



Symbol	Address	Reset Value
P6_0S to P6_2S	400D0h, 400D2h, 400D4h	XXXX X000b
P6_3S to P6_5S	400D6h, 400D8h, 400DAh	XXXX X000b
P6_6S, P6_7S	400DCh, 400DEh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P6_i Output Function Select Bit (1)	b2 b1 b0 0 0 0 : I/O port P6_i 0 0 1 : Do not use this combination 0 1 0 : SBI0/SBI1 output 0 1 1 : UART0/UART1 output 1 0 0 : UART0/UART1 special function output/CAN 2 output 1 0 1 : Do not use this combination 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. Refer to the following table for each pin setting.

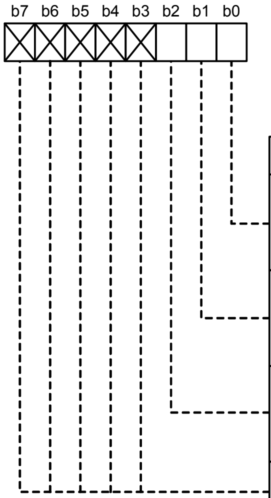
Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P6_0	P6_0	— (2)	$\overline{\text{SCS0}}$ output	$\overline{\text{RTS0}}$	CAN2OUT	— (2)	— (2)	— (2)
P6_1	P6_1	— (2)	SSCK0 output	CLK0 output	— (2)	— (2)	— (2)	— (2)
P6_2	P6_2	— (2)	SSI0 output	SCL0 output	STXD0	— (2)	— (2)	— (2)
P6_3	P6_3	— (2)	SSO0 output	TXD0 SDA0 output	— (2)	— (2)	— (2)	— (2)
P6_4	P6_4	— (2)	$\overline{\text{SCS1}}$ output	$\overline{\text{RTS1}}$	— (2)	— (2)	— (2)	— (2)
P6_5	P6_5	— (2)	SSCK1 output	CLK1 output	— (2)	— (2)	— (2)	— (2)
P6_6	P6_6	— (2)	SSI1 output	SCL1 output	STXD1	— (2)	— (2)	— (2)
P6_7	P6_7	— (2)	SSO1 output	TXD1 SDA1 output	— (2)	— (2)	— (2)	— (2)

2. Do not use this combination.

Figure 27.11 Registers P6_0S to P6_7S

Port P6_i (i = 0 to 7) shares a pin with the serial bus interface (SBI0 and SBI1), serial interface (UART0 and UART1), and CAN module (CAN2).
To use it as an output pin, the PD6_i bit should be set to 1 (port P6_i functions as an output port) and a function should be selected according to Figure 27.11. To use it as an input pin, the PD6_i bit should be set to 0 (port P6_i functions as an input port).

Port P7_i Function Select Register (i = 0 to 7) (1)



Symbol	Address	Reset Value
P7_0S to P7_2S	400D1h, 400D3h, 400D5h	XXXX X000b
P7_3S to P7_5S	400D7h, 400D9h, 400DBh	XXXX X000b
P7_6S, P7_7S	400DDh, 400DFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P7_i Output Function Select Bit (2)	b2 b1 b0 0 0 0 : I/O port P7_i 0 0 1 : Timer output 0 1 0 : Three-phase motor control output 0 1 1 : UART2/UART3 output 1 0 0 : UART2 special function output 1 0 1 : LIN0 output 1 1 0 : IIO1 output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

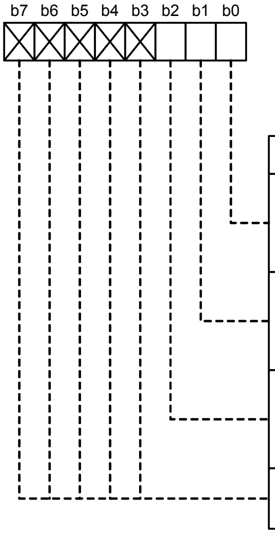
1. These registers should be rewritten after the PRC30 bit in the PRCR3 register is set to 1 (write enabled).
2. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P7_0	P7_0	TA0OUT output	— (3)	TXD2 SDA2 output	— (3)	— (3)	IIO1_6 output	— (3)
P7_1	P7_1	— (3)	— (3)	SCL2 output	STXD2	— (3)	IIO1_7 output	— (3)
P7_2	P7_2	TA1OUT output	V	CLK2 output	— (3)	— (3)	— (3)	— (3)
P7_3	P7_3	— (3)	$\overline{V}$	$\overline{RTS2}$	— (3)	— (3)	IIO1_0 output	— (3)
P7_4	P7_4	TA2OUT output	W	— (3)	— (3)	LIN0OUT	IIO1_1 output	— (3)
P7_5	P7_5	— (3)	$\overline{W}$	— (3)	— (3)	— (3)	IIO1_2 output	— (3)
P7_6	P7_6	TA3OUT output	— (3)	TXD3	— (3)	— (3)	IIO1_3 output	— (3)
P7_7	P7_7	— (3)	— (3)	CLK3 output	— (3)	— (3)	IIO1_4 output	— (3)

3. Do not use this combination.

Figure 27.12 Registers P7_0S to P7_7S in the R32C/142 Group

Port P7_i Function Select Register (i = 0 to 7) (1)



Symbol	Address	Reset Value
P7_0S to P7_2S	400D1h, 400D3h, 400D5h	XXXX X000b
P7_3S to P7_5S	400D7h, 400D9h, 400DBh	XXXX X000b
P7_6S, P7_7S	400DDh, 400DFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P7_i Output Function Select Bit (2)	b2 b1 b0 0 0 0 : I/O port P7_i 0 0 1 : Timer output 0 1 0 : Three-phase motor control output 0 1 1 : UART2/UART3 output 1 0 0 : UART2 special function output 1 0 1 : CAN0/LIN0 output 1 1 0 : IIO1 output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. These registers should be rewritten after the PRC30 bit in the PRCR3 register is set to 1 (write enabled).
2. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P7_0	P7_0	TA0OUT output	— (3)	TXD2 SDA2 output	— (3)	— (3)	IIO1_6 output	— (3)
P7_1	P7_1	— (3)	— (3)	SCL2 output	STXD2	— (3)	IIO1_7 output	— (3)
P7_2	P7_2	TA1OUT output	V	CLK2 output	— (3)	— (3)	— (3)	— (3)
P7_3	P7_3	— (3)	$\overline{\text{V}}$	$\overline{\text{RTS2}}$	— (3)	— (3)	IIO1_0 output	— (3)
P7_4	P7_4	TA2OUT output	W	— (3)	— (3)	LIN0OUT	IIO1_1 output	— (3)
P7_5	P7_5	— (3)	$\overline{\text{W}}$	— (3)	— (3)	— (3)	IIO1_2 output	— (3)
P7_6	P7_6	TA3OUT output	— (3)	TXD3	— (3)	CAN0OUT	IIO1_3 output	— (3)
P7_7	P7_7	— (3)	— (3)	CLK3 output	— (3)	— (3)	IIO1_4 output	— (3)

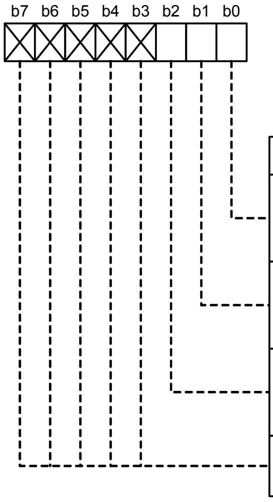
3. Do not use this combination.

Figure 27.13 Registers P7_0S to P7_7S in the R32C/145 Group

Port P7_i (i = 0 to 7) shares a pin with the timer, three-phase motor control, serial interface (UART2 and UART3), intelligent I/O group 1 (IIO1), LIN module, and CAN module.

To use it as an output pin, the PD7_i bit should be set to 1 (port P7_i functions as an output port) and a function should be selected according to Figures 27.10 and 27.11. To use it as an input pin, the PD7_i bit should be set to 0 (port P7_i functions as an input port).

Port P8_i Function Select Register (i = 0 to 4, 6, 7) (1)



Symbol	Address	Reset Value
P8_0S to P8_2S	400E0h, 400E2h, 400E4h	XXXX X000b
P8_3S, P8_4S	400E6h, 400E8h	XXXX X000b
P8_6S, P8_7S	400ECh, 400EEh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P8_i Output Function Select Bit (2)	b2 b1 b0 0 0 0 : I/O port P8_i 0 0 1 : Timer output 0 1 0 : Three-phase motor control output 0 1 1 : UART3 output 1 0 0 : Do not use this combination 1 0 1 : LIN1 output 1 1 0 : IIO1 output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

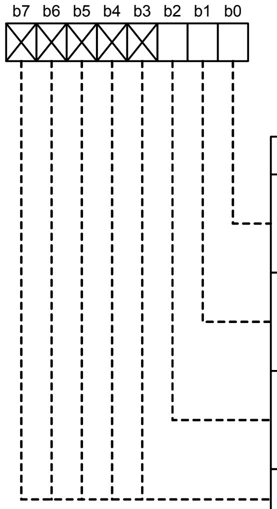
1. These registers should be rewritten after the PRC30 bit in the PRCR3 register is set to 1 (write enabled).
2. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P8_0	P8_0	TA4OUT output	U	— (3)	— (3)	— (3)	— (3)	— (3)
P8_1	P8_1	— (3)	$\overline{U}$	$\overline{RTS3}$	— (3)	LIN1OUT	IIO1_5 output	— (3)
P8_2	P8_2	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P8_3	P8_3	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P8_4	P8_4	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P8_6	P8_6	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P8_7	P8_7	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)

2. Do not use this combination.

Figure 27.14 Registers P8_0S to P8_4S, P8_6S, and P8_7S in the R32C/142 Group

Port P8_i Function Select Register (i = 0 to 4, 6, 7) (1)



Symbol	Address	Reset Value
P8_0S to P8_2S	400E0h, 400E2h, 400E4h	XXXX X000b
P8_3S, P8_4S	400E6h, 400E8h	XXXX X000b
P8_6S, P8_7S	400ECh, 400EEh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P8_i Output Function Select Bit (2)	b2 b1 b0 0 0 0 : I/O port P8_i 0 0 1 : Timer output 0 1 0 : Three-phase motor control output 0 1 1 : UART3 output 1 0 0 : Do not use this combination 1 0 1 : CAN0/LIN1 output 1 1 0 : IIO1/CAN1 output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. These registers should be rewritten after the PRC30 bit in the PRCR3 register is set to 1 (write enabled).
2. Refer to the following table for each pin setting.

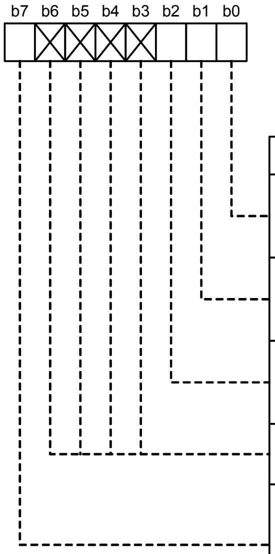
Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P8_0	P8_0	TA4OUT output	U	— (3)	— (3)	— (3)	— (3)	— (3)
P8_1	P8_1	— (3)	$\overline{U}$	$\overline{RTS3}$	— (3)	LIN1OUT	IIO1_5 output	— (3)
P8_2	P8_2	— (3)	— (3)	— (3)	— (3)	CAN0OUT	CAN1OUT	— (3)
P8_3	P8_3	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P8_4	P8_4	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P8_6	P8_6	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P8_7	P8_7	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)

3. Do not use this combination.

Figure 27.15 Registers P8_0S to P8_4S, P8_6S, and P8_7S in the R32C/145 Group

Port P8_i (i = 0 to 4, 6, 7) shares a pin with the timer, three-phase motor control, serial interface (UART3), intelligent I/O group 1 (IIO1), LIN module, CAN module, and external interrupt input pin.
To use it as an output pin, the PD8_i bit should be set to 1 (port P8_i functions as an output port) and a function should be selected according to Figures 27.12 and 27.13. To use it as an input pin, the PD8_i bit should be set to 0 (port P8_i functions as an input port).

Port P9_i Function Select Register (i = 3 to 7) (1)



Symbol	Address	Reset Value
P9_3S to P9_6S	400E7h, 400E9h, 400EBh, 400EDh	0XXX X000b
P9_7S	400EFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P9_i Output Function Select Bit (2)	b2 b1 b0 0 0 0 : I/O port P9_i 0 0 1 : Do not use this combination 0 1 0 : Do not use this combination 0 1 1 : UART4 output 1 0 0 : Do not use this combination 1 0 1 : Do not use this combination 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b6-b3)	No register bits; should be written with 0 and read as undefined value		—
ASEL (i = 3 to 6)	Port P9_i (i = 3 to 6) Analog Functions Select Bit	0: Function other than analog pin 1: Analog pin	RW
— (b7) (i = 7)	No register bit; should be written with 0 and read as undefined value		—

Notes:

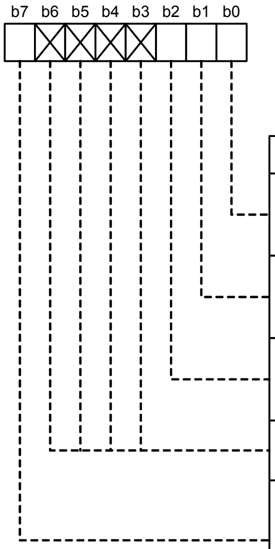
1. Set the PRC2 bit in the PRCR register to 1 (write enabled) just before rewriting this register. No interrupt handling or DMA transfers should be inserted between these two instructions.
2. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P9_3	P9_3	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P9_4	P9_4	— (3)	— (3)	$\overline{RTS4}$	— (3)	— (3)	— (3)	— (3)
P9_5	P9_5	— (3)	— (3)	CLK4 output	— (3)	— (3)	— (3)	— (3)
P9_6	P9_6	— (3)	— (3)	TXD4	— (3)	— (3)	— (3)	— (3)
P9_7	P9_7	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)

3. Do not use this combination.

Figure 27.16 Registers P9_3S to P9_7S in the R32C/142 Group

Port P9_i Function Select Register (i = 3 to 7) (1)



Symbol	Address	Reset Value
P9_3S to P9_6S	400E7h, 400E9h, 400EBh, 400EDh	0XXX X000b
P9_7S	400EFh	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P9_i Output Function Select Bit (2)	b2 b1 b0 0 0 0 : I/O port P9_i 0 0 1 : Do not use this combination 0 1 0 : Do not use this combination 0 1 1 : UART4 output 1 0 0 : Do not use this combination 1 0 1 : Do not use this combination 1 1 0 : CAN1 output 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b6-b3)	No register bits; should be written with 0 and read as undefined value		—
ASEL (i = 3 to 6)	Port P9_i (i = 3 to 6) Analog Functions Select Bit	0: Function other than analog pin 1: Analog pin	RW
— (b7) (i = 7)	No register bit; should be written with 0 and read as undefined value		—

Notes:

1. Set the PRC2 bit in the PRCR register to 1 (write enabled) just before rewriting this register. No interrupt handling or DMA transfers should be inserted between these two instructions.
2. Refer to the following table for each pin setting.

Port	Setting Value of Bits PSEL2 to PSEL0							
	000b	001b	010b	011b	100b	101b	110b	111b
P9_3	P9_3	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)
P9_4	P9_4	— (3)	— (3)	$\overline{\text{RTS4}}$	— (3)	— (3)	— (3)	— (3)
P9_5	P9_5	— (3)	— (3)	CLK4 output	— (3)	— (3)	— (3)	— (3)
P9_6	P9_6	— (3)	— (3)	TXD4	— (3)	— (3)	CAN1OUT	— (3)
P9_7	P9_7	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)	— (3)

3. Do not use this combination.

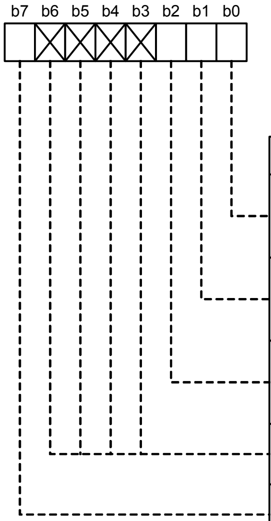
Figure 27.17 Registers P9_3S to P9_7S in the R32C/145 Group

Port P9_i (i = 3 to 7) shares a pin with the serial interface (UART4) and CAN module. In particular, port P9_i (i = 3 to 6) also shares a pin with the A/D converter I/O (ANEX0 and ANEX1) pin and D/A converter output pin.

To use it as the A/D converter pin or the D/A converter pin, the P9_iS register should be set to 80h and the PD9_i bit should be set to 0 (port P9_i functions as an input port) irrespective of the I/O state.

To use it as an output pin for functions other than the A/D converter or the D/A converter, the PD9_i bit should be set to 1 (port P9_i functions as an output port) and a function should be selected according to Figures 27.14 and 27.15. To use it as an input pin of functions other than the A/D converter or the D/A converter, the PD9_i bit should be set to 0 (port P9_i functions as an input port).

Port P10_i Function Select Register (i = 0 to 7)



Symbol	Address	Reset Value
P10_0S to P10_2S	400F0h, 400F2h, 400F4h	0XXX X000b
P10_3S to P10_5S	400F6h, 400F8h, 400FAh	0XXX X000b
P10_6S, P10_7S	400FCh, 400FEh	0XXX X000b

Bit Symbol	Bit Name	Function	RW
PSEL0	Port P10_i Output Function Select Bit	b2 b1 b0 0 0 0 : I/O port P10_i 0 0 1 : Do not use this combination 0 1 0 : Do not use this combination 0 1 1 : Do not use this combination 1 0 0 : Do not use this combination 1 0 1 : Do not use this combination 1 1 0 : Do not use this combination 1 1 1 : Do not use this combination	RW
PSEL1			RW
PSEL2			RW
— (b6-b3)	No register bits; should be written with 0 and read as undefined value		—
ASEL	Port P10_i Analog Functions Select Bit	0: Function other than AN_i 1: AN_i	RW

Figure 27.18 Registers P10_0S to P10_7S

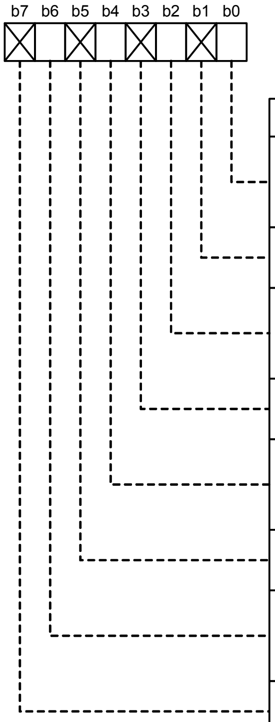
Port P10_i (i = 0 to 7) shares a pin with the AN_i input pin for the A/D converter and key input interrupt pin. To use as it a programmable I/O port, the P10_iS register should be set to 00h. To use it as an input pin (except for the A/D converter), the PD10_i bit should be set to 0 (port P10_i functions as an input port). To use it as an input pin for the A/D converter, the P10_iS register should be set to 80h and the PD10_i bit should be set to 0 (port P10_i functions as an input port).

27.3 Input Function Select Registers

When a peripheral function input is assigned to multiple pins, these registers select which input pin should be connected to the peripheral function.

Figure 27.19 to Figure 27.26 show the input function select registers.

Input Function Select Register 0



Symbol	Address	Reset Value
IFS0	40098h	X0X0 X0X0b

Bit Symbol	Bit Name	Function	RW
IFS00	Timer A Input Pin Switch Bit (1)	Assign timer A input to 0: Port P3 1: Port P7/port P8	RW
— (b1)	No register bit; should be written with 0 and read as undefined value		—
IFS02	UART3 Input Pin Switch Bit (2)	Assign UART3 input to 0: Port P4 1: Port P7/port P8	RW
— (b3)	No register bit; should be written with 0 and read as undefined value		—
IFS04	UART4 Input Pin Switch Bit (3)	Assign UART4 input to 0: Port P9 1: Port P3	RW
— (b5)	No register bit; should be written with 0 and read as undefined value		—
IFS06	Timer B Input Pin Switch Bit (4)	Assign timer B input to 0: Port P6/port P9 1: Port P1/port P3/port P7/port P9	RW
— (b7)	No register bit; should be written with 0 and read as undefined value		—

Notes:

1. Refer to the following table for each pin setting of timer A.

IFS00	TA0OUT input	TA1OUT input	TA1IN	TA2OUT input	TA2IN	TA3OUT input	TA4OUT input	TA4IN
0	P3_0	P3_2	P3_3	P3_4	P3_5	P3_1	P3_6	P3_7
1	P7_0	P7_2	P7_3	P7_4	P7_5	P7_6	P8_0	P8_1

2. Refer to the following table for each pin setting of UART3.

IFS02	CLK3 input	RXD3	$\overline{\text{CTS3}}$
0	P4_1	P4_2	P4_0
1	P7_7	P8_0	P8_1

3. Refer to the following table for each pin setting of UART4.

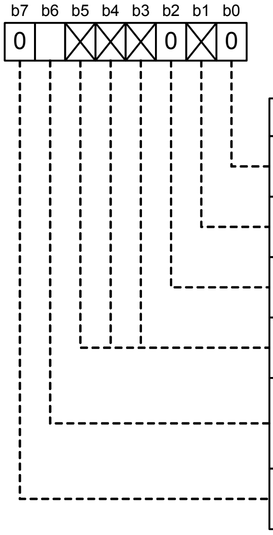
IFS04	CLK4 input	RXD4	$\overline{\text{CTS4}}$
0	P9_5	P9_7	P9_4
1	P3_4	P3_5	P3_7

4. Refer to the following table for each pin setting of timer B.

IFS06	TB0IN	TB1IN	TB2IN	TB3IN	TB4IN	TB5IN
0	P6_0	P6_1	P6_2	P9_3	P9_4	P6_3
1	P1_4	P3_7	P9_7	P9_3	P9_4	P7_1

Figure 27.19 IFS0 Register

Input Function Select Register 1

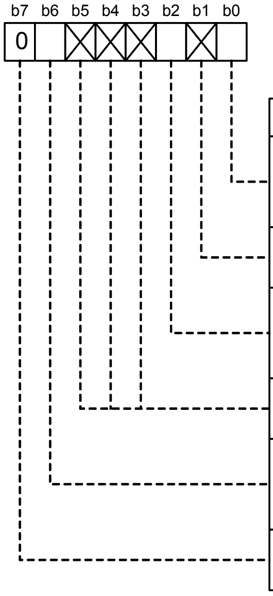


Symbol	Address	Reset Value
IFS1	40099h	00XX X0X0b

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved	Should be written with 0	RW
— (b1)	No register bit; should be written with 0 and read as undefined value		—
— (b2)	Reserved	Should be written with 0	RW
— (b5-b3)	No register bits; should be written with 0 and read as undefined value		—
IFS16	CAN3 Input Pin Switch Bit	Assign CAN3IN/$\overline{\text{CAN3WU}}$ input to 0: Port P5_7 1: Port P1_0	RW
— (b7)	Reserved	Should be written with 0	RW

Figure 27.20 IFS1 Register in the R32C/142 Group

Input Function Select Register 1

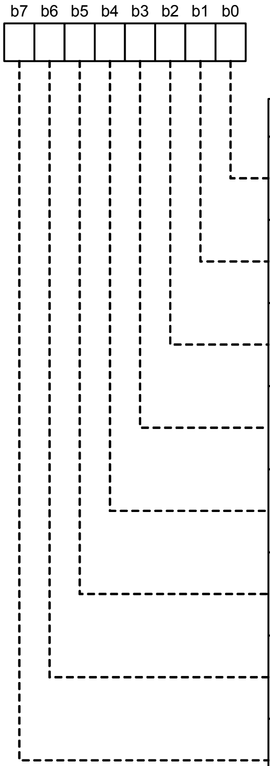


Symbol	Address	Reset Value
IFS1	40099h	00XX X0X0b

Bit Symbol	Bit Name	Function	RW
IFS10	CAN0 Input Pin Switch Bit	Assign CAN0IN/$\overline{\text{CAN0WU}}$ input to 0: Port P7_7 1: Port P8_3	RW
— (b1)	No register bit; should be written with 0 and read as undefined value		—
IFS12	CAN1 Input Pin Switch Bit	Assign CAN1IN/$\overline{\text{CAN1WU}}$ input to 0: Port P9_5 1: Port P8_3	RW
— (b5-b3)	No register bits; should be written with 0 and read as undefined value		—
IFS16	CAN3 Input Pin Switch Bit	Assign CAN3IN/$\overline{\text{CAN3WU}}$ input to 0: Port P5_7 1: Port P1_0	RW
— (b7)	Reserved	Should be written with 0	RW

Figure 27.21 IFS1 Register in the R32C/145 Group

Input Function Select Register 2



Symbol	Address	Reset Value
IFS2	4009Ah	0000 0000b

Bit Symbol	Bit Name	Function	RW
IFS20	Intelligent I/O Group 0 Input Pin Switch Bit (1)	Assign IIO0 input to b1 b0 0 0 : Do not use this combination 0 1 : Port P1 1 0 : Port P5 1 1 : Do not use this combination	RW
IFS21			RW
IFS22	Intelligent I/O Group 0 Two-phase Pulse Input Pin Switch Bit (2)	Assign this input to b3 b2 0 0 : Port P8 and $\overline{\text{INT1}}$ 0 1 : Port P7 and $\overline{\text{INT0}}$ 1 0 : Port P3 and $\overline{\text{INT1}}$ 1 1 : Port P3 and $\overline{\text{INT0}}$	RW
IFS23			RW
IFS24	Intelligent I/O Group 1 Input Pin Switch Bit (3)	Assign IIO1 input to b5 b4 0 0 : Port P7/port P8 0 1 : Do not use this combination 1 0 : Port P1 1 1 : Port P4	RW
IFS25			RW
IFS26	Intelligent I/O Group 1 Two-phase Pulse Input Pin Switch Bit (4)	Assign this input to b7 b6 0 0 : Port P8 and $\overline{\text{INT1}}$ 0 1 : Port P7 and $\overline{\text{INT0}}$ 1 0 : Port P3 and $\overline{\text{INT1}}$ 1 1 : Port P3 and $\overline{\text{INT0}}$	RW
IFS27			RW

Notes:

1. Refer to the following table for each pin setting of intelligent I/O group 0.

IFS21	IFS20	IIO0_0 input	IIO0_1 input	IIO0_2 input	IIO0_3 input	IIO0_4 input	IIO0_5 input	IIO0_6 input	IIO0_7 input
0	1	P1_0	P1_1	P1_2	P1_3	P1_4	P1_5	P1_6	P1_7
1	0	P5_0	P5_1	P5_2	P5_3	P5_4	P5_5	P5_6	P5_7

2. Refer to the following table for each pin setting of intelligent I/O group 0 in two-phase pulse signal processing mode.

IFS23	IFS22	UD0A	UD0B	UD0Z
0	0	P8_0	P8_1	P8_3 ($\overline{\text{INT1}}$)
0	1	P7_6	P7_7	P8_2 ($\overline{\text{INT0}}$)
1	0	P3_0	P3_1	P8_3 ($\overline{\text{INT1}}$)
1	1	P3_0	P3_1	P8_2 ($\overline{\text{INT0}}$)

3. Refer to the following table for each pin setting of intelligent I/O group 1.

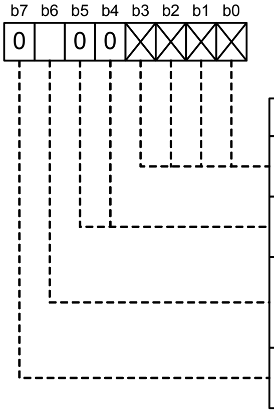
IFS25	IFS24	IIO1_0 input	IIO1_1 input	IIO1_2 input	IIO1_3 input	IIO1_4 input	IIO1_5 input	IIO1_6 input	IIO1_7 input
0	0	P7_3	P7_4	P7_5	P7_6	P7_7	P8_1	P7_0	P7_1
0	1	P1_0	P1_1	P1_2	P1_3	P1_4	P1_5	P1_6	P1_7
1	0	P4_0	P4_1	P4_2	P4_3	P4_4	P4_5	P4_6	P4_7

4. Refer to the following table for each pin setting of intelligent I/O group 1 in two-phase pulse signal processing mode.

IFS27	IFS26	UD1A	UD1B	UD1Z
0	0	P8_0	P8_1	P8_3 ($\overline{\text{INT1}}$)
0	1	P7_6	P7_7	P8_2 ($\overline{\text{INT0}}$)
1	0	P3_0	P3_1	P8_3 ($\overline{\text{INT1}}$)
1	1	P3_0	P3_1	P8_2 ($\overline{\text{INT0}}$)

Figure 27.22 IFS2 Register

Input Function Select Register 3

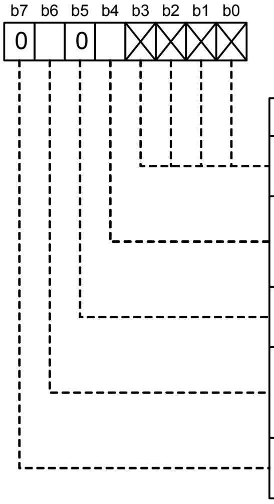


Symbol	Address	Reset Value
IFS3	4009Bh	0000 XXXXb

Bit Symbol	Bit Name	Function	RW
— (b3-b0)	No register bits; should be written with 0 and read as undefined value		—
— (b5-b4)	Reserved	Should be written with 0	RW
IFS36	CAN5 Input Pin Switch Bit	Assign CAN5IN/$\overline{\text{CAN5WU}}$ input to 0: Port P5_3 1: Port P1_4	RW
— (b7)	Reserved	Should be written with 0	RW

Figure 27.23 IFS3 Register in the R32C/142 Group

Input Function Select Register 3

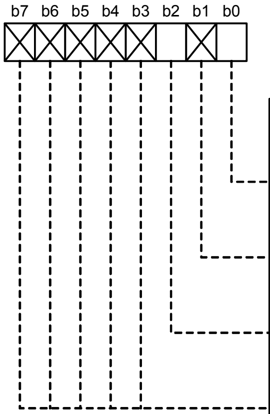


Symbol	Address	Reset Value
IFS3	4009Bh	0000 XXXXb

Bit Symbol	Bit Name	Function	RW
— (b3-b0)	No register bits; should be written with 0 and read as undefined value		—
IFS34	CAN4 Input Pin Switch Bit	Assign CAN4IN/$\overline{\text{CAN4WU}}$ input to 0: Port P5_5 1: Port P1_2	RW
— (b5)	Reserved	Should be written with 0	RW
IFS36	CAN5 Input Pin Switch Bit	Assign CAN5IN/$\overline{\text{CAN5WU}}$ input to 0: Port P5_3 1: Port P1_4	RW
— (b7)	Reserved	Should be written with 0	RW

Figure 27.24 IFS3 Register in the R32C/145 Group

Input Function Select Register 5



Symbol
IFS5

Address
4009Dh

Reset Value
XXXX X0X0b

Bit Symbol	Bit Name	Function	RW
IFS50	SBI0 Input Pin Switch Bit (1)	Assign SBI0 input to 0: Port P1 1: Port P6	RW
— (b1)	No register bit; should be written with 0 and read as undefined value		—
IFS52	SBI1 Input Pin Switch Bit (2)	Assign SBI1 input to 0: Port P4 1: Port P6	RW
— (b7-b3)	No register bits; should be written with 0 and read as undefined value		—

Notes:

1. Refer to the following table for each pin setting of SBI0.

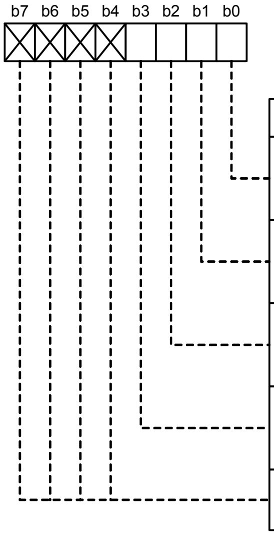
IFS50	SSO0 input	SSI0 input
0	P1_0	P1_2
1	P6_3	P6_2

2. Refer to the following table for each pin setting of SBI1.

IFS52	SSO1 input	SSI1 input
0	P4_4	P4_6
1	P6_7	P6_6

Figure 27.25 IFS5 Register

Input Function Select Register 6



Symbol	Address	Reset Value
IFS6	4009Eh	XXXX 0000b

Bit Symbol	Bit Name	Function	RW
IFS60	LIN0 Input Pin Switch Bit	Assign LIN0IN input to b1 b0 0 0 : Port P3_1 0 1 : Port P4_5 1 0 : Port P7_5 1 1 : Do not use this combination	RW
IFS61			RW
IFS62	LIN1 Input Pin Switch Bit	Assign LIN1IN input to b3 b2 0 0 : Port P3_3 0 1 : Port P4_7 1 0 : Port P8_0 1 1 : Do not use this combination	RW
IFS63			RW
— (b7-b4)	No register bits; should be written with 0 and read as undefined value		—

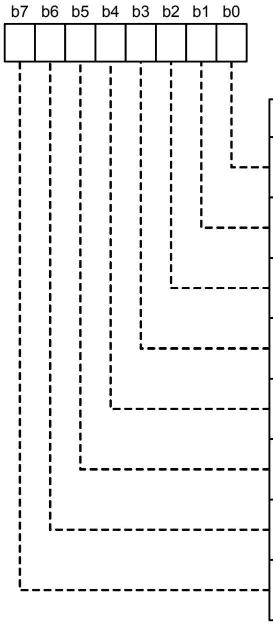
Figure 27.26 IFS6 Register

27.4 Pull-up Control Registers 0 to 3 (Registers PUR0 to PUR3)

Figure 27.27 to Figure 27.30 show registers PUR0 to PUR3.

These registers enable/disable the pull-up resistors for every group of four pins. To enable the pull-up resistors, the corresponding bits in registers PUR0 to PUR3 should be set to 1 (pull-up resistor enabled) and the respective bits in the direction register should be set to 0 (input).

Pull-up Control Register 0

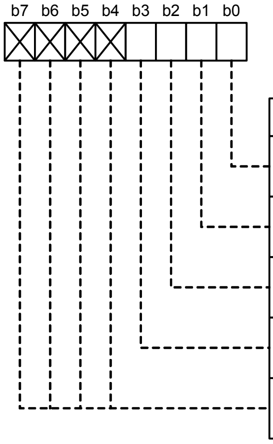


Symbol	Address	Reset Value
PUR0	03F0h	0000 0000b

Bit Symbol	Bit Name	Function	RW
PU00	P0_0 to P0_3 Pull-up Control Bit	Control pull-up setting for corresponding ports 0: Pull-up resistor disabled 1: Pull-up resistor enabled	RW
PU01	P0_4 to P0_7 Pull-up Control Bit		RW
PU02	P1_0 to P1_3 Pull-up Control Bit		RW
PU03	P1_4 to P1_7 Pull-up Control Bit		RW
PU04	P2_0 to P2_3 Pull-up Control Bit		RW
PU05	P2_4 to P2_7 Pull-up Control Bit		RW
PU06	P3_0 to P3_3 Pull-up Control Bit		RW
PU07	P3_4 to P3_7 Pull-up Control Bit		RW

Figure 27.27 PUR0 Register

Pull-up Control Register 1

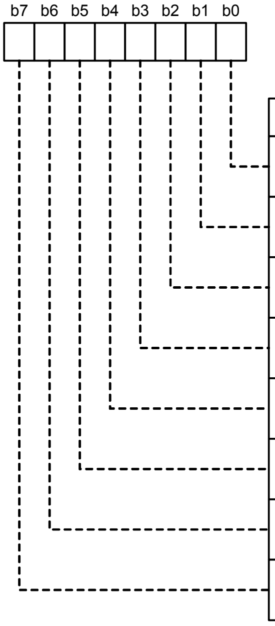


Symbol	Address	Reset Value
PUR1	03F1h	XXXX 0000b

Bit Symbol	Bit Name	Function	RW
PU10	P4_0 to P4_3 Pull-up Control Bit	Control pull-up setting for corresponding ports 0: Pull-up resistor disabled 1: Pull-up resistor enabled	RW
PU11	P4_4 to P4_7 Pull-up Control Bit		RW
PU12	P5_0 to P5_3 Pull-up Control Bit		RW
PU13	P5_4 to P5_7 Pull-up Control Bit		RW
— (b7-b4)	No register bits; should be written with 0 and read as undefined value		—

Figure 27.28 PUR1 Register

Pull-up Control Register 2



Symbol	Address	Reset Value
PUR2	03F2h	0000 0000b

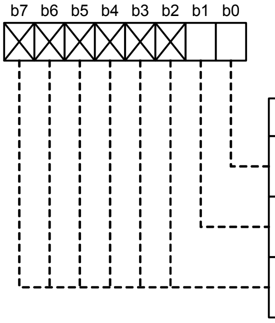
Bit Symbol	Bit Name	Function	RW
PU20	P6_0 to P6_3 Pull-up Control Bit	Control pull-up setting for corresponding ports 0: Pull-up resistor disabled 1: Pull-up resistor enabled	RW
PU21	P6_4 to P6_7 Pull-up Control Bit		RW
PU22	P7_0 to P7_3 Pull-up Control Bit		RW
PU23	P7_4 to P7_7 Pull-up Control Bit		RW
PU24	P8_0 to P8_3 Pull-up Control Bit		RW
PU25	P8_4 to P8_7 Pull-up Control Bit [1]		RW
PU26	P9_1 and P9_3 Pull-up Control Bit		RW
PU27	P9_4 to P9_7 Pull-up Control Bit		RW

Note:

1. Port P8_5 has no pull-up resistor.

Figure 27.29 PUR2 Register

Pull-up Control Register 3



Symbol	Address	Reset Value
PUR3	03F3h	XXXX XX00b

Bit Symbol	Bit Name	Function	RW
PU30	P10_0 to P10_3 Pull-up Control Bit	Control pull-up setting for corresponding ports 0: Pull-up resistor disabled 1: Pull-up resistor enabled	RW
PU31	P10_4 to P10_7 Pull-up Control Bit		RW
— (b7-b2)	No register bits; should be written with 0 and read as undefined value		—

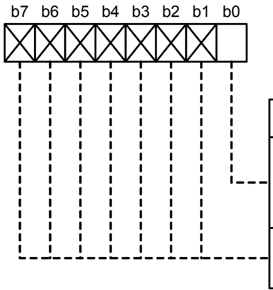
Figure 27.30 PUR3 Register

27.5 Port Control Register (PCR Register)

Figure 27.31 shows the PCR register.

This register selects an output mode for port P1 between push-pull output and pseudo-N-channel open drain output. When the PCR0 bit is set to 1, the P-channel transistor in the output buffer is turned off. Note that port P1 cannot be a perfect open drain output due to remaining parasitic diode. The absolute maximum rating of the input voltage is, therefore, -0.3 V to VCC + 0.3 V (refer to Figure 27.32).

Port Control Register



Symbol	Address	Reset Value
PCR	03FFh	XXXX XXX0b

Bit Symbol	Bit Name	Function	RW
PCR0	Port P1 Output Format Control Bit	0: Push-pull output 1: Pseudo-N-channel open drain output [(1)]	RW
— (b7-b1)	No register bits; should be written with 0 and read as undefined value		—

Note:

1. This function is designated not to make port P1 a full open drain, but to turn off the P-channel transistor in the CMOS output buffer. Therefore, the absolute maximum rating of the input voltage is -0.3 V to VCC + 0.3 V.

Figure 27.31 PCR Register

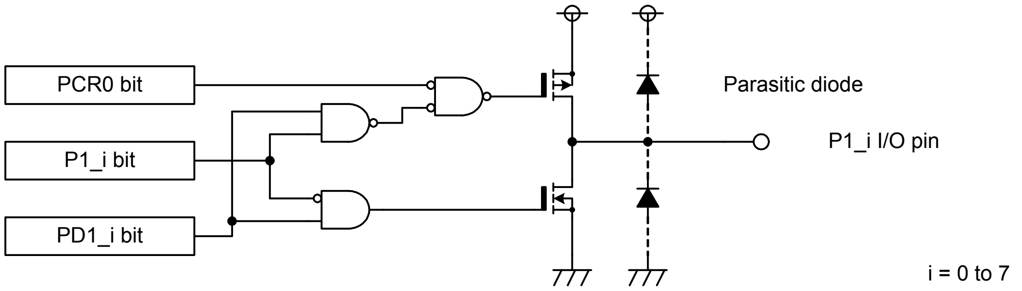


Figure 27.32 Port P1 Output Buffer Configuration

27.6 Configuring Unused Pins

Table 27.2 and Figure 27.34 show examples of configuring unused pins on the board.

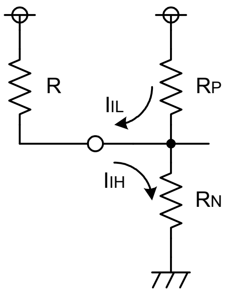
Table 27.2 Unused Pin Configuration in Single-chip Mode (1)

Pin Name	Setting
Ports P0 to P10 (excluding ports P8_5 and P9_1) (2)	Configure as input ports so that each pin is connected to VSS via its own resistor; (3) or configure as output ports to leave the pins open
P9_1	Connect the pin to VSS via a resistor (3)
XOUT (4)	Leave pin open
$\overline{\text{NMI}}$ (P8_5)	Connect the pin to VCC via a resistor (3)
AVCC	Connect the pin to VCC
AVSS, VREF	Connect the pin to VSS
NSD	Connect the pin to VCC via a resistor of 1 to 4.7 kΩ

Notes:

1. Unused pins should be wired within 2 cm of the MCU.
2. When configuring the pins as output ports to leave them open, note that ports as inputs remain unchanged from when the reset is released until the mode transition is completed. During this transition, the power current may increase due to an undefined voltage level of the pins. In addition, the contents of the direction register may change due to noise or program runaway caused by the noise. To avoid these situations, reconfigure the direction register regularly by software, which may achieve higher program reliability.
3. The resistance value appropriate to the system should be designated. The range from 10 to 100 kΩ is recommended.
4. This setting is applicable when an external clock is applied to the XIN pin.

Pull-up/pull-down resistors



The figure shows the equivalent circuit of an input pin.
The equivalent input resistors (R_P and R_N) are calculated using input power current (I_{IL} and I_{IH}).

Example: When VCC = 5.0 V, $I_{IH} = I_{IL} = 5\ \mu A$,

$$R_P = R_N = \frac{5.0}{5 \times 10^{-6}} = 1\ M\Omega$$

Since the voltage (V_{IH}) defined as high is more than 0.7 VCC,
the resistance value R should satisfy the following expression:

$$R//R_P : R_N = 0.3 : 0.7$$

That is,

$$R = \frac{3R_PR_N}{7R_P - 3R_N}$$

Specifically,

Example: When VCC = 5.0 V, $I_{IH} = I_{IL} = 5\ \mu A$,

$$R = \frac{3 \times 10^6 \times 10^6}{7 \times 10^6 - 3 \times 10^6} = 750000$$

The maximum pull-up resistor R is approximately 750 kΩ.
The actual resistance value is the calculated value with some margins.

Figure 27.33 Pull-up/Pull-down Resistors

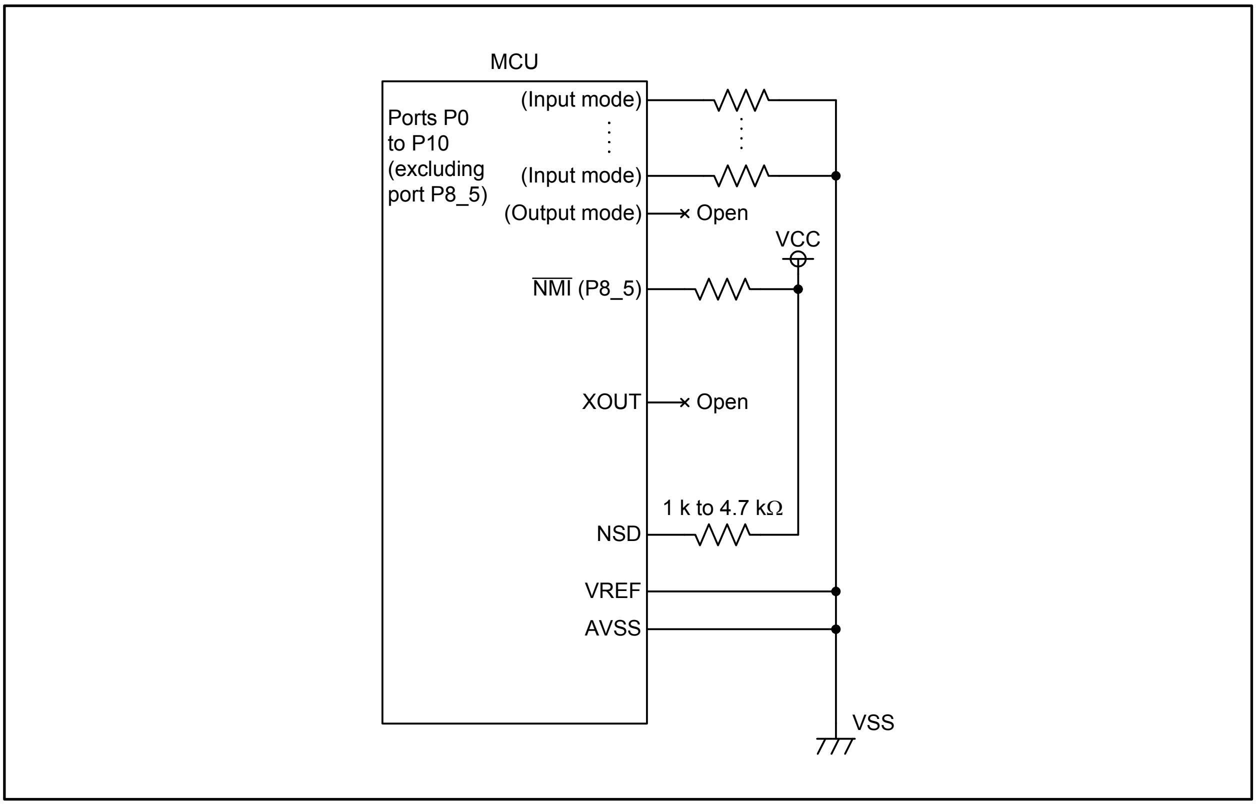


Figure 27.34 Unused Pin Configuration

28. Flash Memory

28.1 Overview

Rewrite operation to the flash memory can be performed in the following three modes: CPU rewrite mode, standard serial I/O mode, and parallel I/O mode.
Table 28.1 lists specifications of the flash memory and Table 28.2 shows the overview of each rewrite mode.

Table 28.1 Flash Memory Specifications

Item	Specification
Rewrite modes	CPU rewrite mode, standard serial I/O mode, parallel I/O mode
Structure	Block architecture. Refer to Figure 28.1
Program operation	8-byte basis
Erase operation	1-block basis
Program and erase control method	Software commands
Protection types	Lock bit protect, ROM code protect, ID code protect
Software commands	9

Table 28.2 Flash Memory Rewrite Mode Overview

Rewrite Mode	CPU Rewrite Mode	Standard Serial I/O Mode	Parallel I/O Mode
Function	CPU executes a software command to rewrite the flash memory EW0 mode: Rewritable in areas other than the on-chip flash memory EW1 mode: Rewritable in areas other than specified blocks to be rewritten	A dedicated serial programmer rewrites the flash memory Standard serial I/O mode 1: Synchronous serial I/O selected Standard serial I/O mode 2: UART selected	A dedicated parallel programmer rewrites the flash memory
CPU operating mode	Single-chip mode	Standard serial I/O mode	Parallel I/O mode
Programmer	—	Serial programmer	Parallel programmer
On-board rewriting	Supported	Supported	Not supported

Figure 28.1 shows the on-chip flash memory structure.
The on-chip flash memory contains program area to store user programs, and data area/data flash to store the result of user programs. The program area consists of blocks 0 to 9, and data area/data flash consists of blocks A and B.
Each block can be individually protected (locked) from programming or erasing by setting the lock bit.

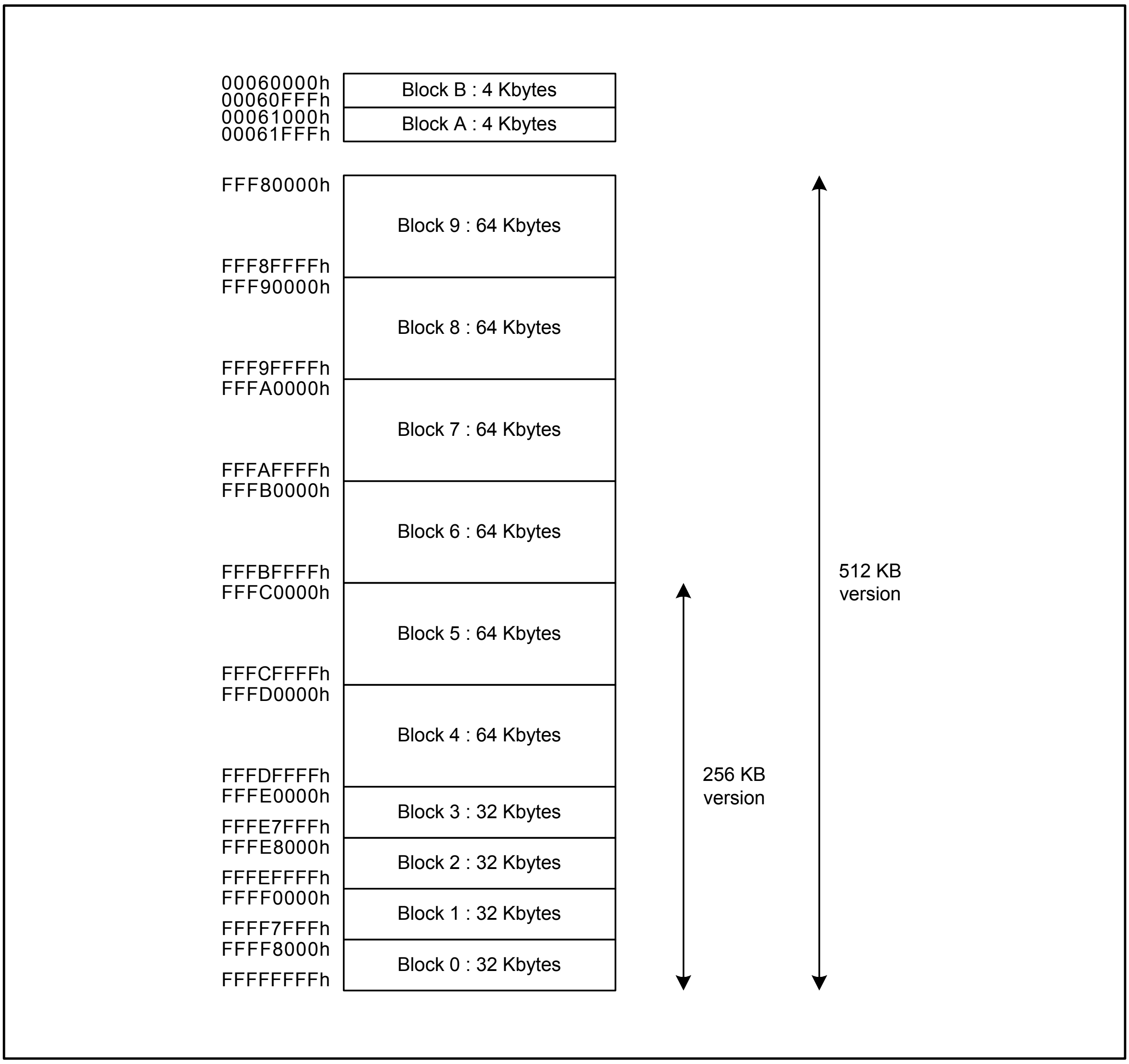


Figure 28.1 On-chip Flash Memory Block Diagram

28.2 Flash Memory Protection

There are three types of protection as shown in Table 28.3. Lock bit protection is intended to prevent accidental write or erase by program runaway. ROM code protection and ID code protection are intended to prevent read or write by a third party.

Table 28.3 Protection Types and Characteristics

Protection Type	Lock Bit Protection	ROM Code Protection	ID Code Protection
Protected operations	Erase, write	Read, write	Read, erase, write
Protection available in	CPU rewrite mode Standard serial I/O mode Parallel I/O mode	Parallel I/O mode	Standard serial I/O mode
Protection available for	Individual blocks	Entire flash memory	Entire flash memory
Protection activated by	Setting 0 to the lock bit of block to be protected	Setting 0 to any protect bit of blocks	Writing the program which has set an ID code to specified address
Protection deactivated by	Setting the LBD bit in the FMR register to 1 (lock bit protection disabled), or by erasing the blocks whose lock bits are set to 0 to permanently deactivate the protection	Erasing all blocks whose protect bits are set to 0	Inputting a proper ID code to the serial programmer

28.2.1 Lock Bit Protection

This protection can be used in all three rewrite modes. When the lock bit protection is activated, all blocks whose lock bits are set to 0 (locked) are protected against programming and erasing.
To set the lock bit to 0, the lock bit program command must be issued.
To temporarily deactivate the protection of all protected blocks, disable the lock bit protection itself by setting the LBD bit in the FMR1 register to 1 (lock bit protection disabled). The protection of a protected block is deactivated permanently and its lock bit becomes 1 (unlocked) if the block is erased.

28.2.2 ROM Code Protection

This protection can only be used in parallel I/O mode. When the ROM code protection is activated, the entire flash memory is protected against reading and writing.
To deactivate the protection, erase all the blocks whose protect bits are set to 0 (protected).
Each block has two protect bits. Setting any protect bit to 0 by a software command activates the protection for the entire flash memory. Table 28.4 lists protect bit addresses.

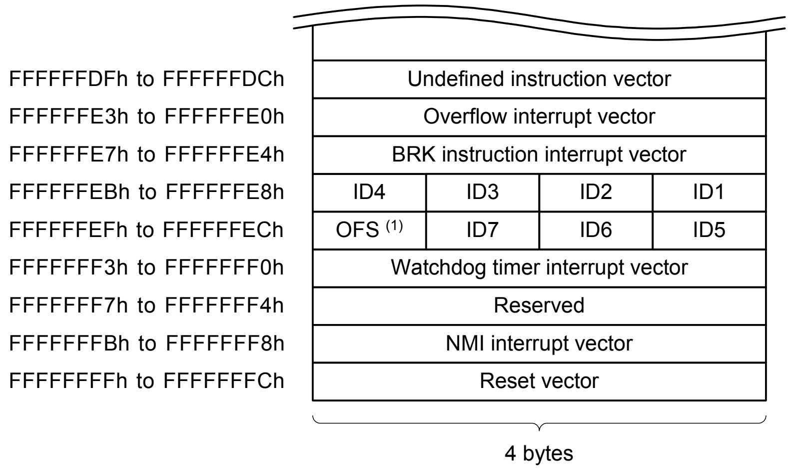
Table 28.4 Protect Bit Addresses

Block	Protect Bit 0	Protect Bit 1
Block B	00060100h	00060300h
Block A	00061100h	00061300h
Block 9	FFF80100h	FFF80300h
Block 8	FFF90100h	FFF90300h
Block 7	FFFA0100h	FFFA0300h
Block 6	FFFB0100h	FFFB0300h
Block 5	FFFC0100h	FFFC0300h
Block 4	FFFD0100h	FFFD0300h
Block 3	FFFE0100h	FFFE0300h
Block 2	FFFE8100h	FFFE8300h
Block 1	FFFF0100h	FFFF0300h
Block 0	FFFF8100h	FFFF8300h

28.2.3 ID Code Protection

This protection can only be used in standard serial I/O mode. When the ID code protection is activated, a command sent from the serial programmer is accepted only if the 7-byte ID code sent from the serial programmer is identical to the ID code programmed in the flash memory. However, if the reset vector is FFFFFFFFh, the ID code check is skipped because the flash memory is considered to be blank. When the reset vector is FFFFFFFFh and the ROM code protection is activated, only the block erase command is accepted.

The ID codes sent from the serial programmer are consecutively numbered as ID1, ID2, ..., and ID7. ID codes programmed in the flash memory, also numbered as ID1, ID2, ..., and ID7, are assigned to addresses FFFFFFE8h, FFFFFFE9h, ..., and FFFFFFEEh as shown in Figure 28.2. The ID code protection is activated when a program which has an ID code set in the corresponding address is written to the flash memory.



Note:
1. Refer to the Watchdog Timer chapter for the OFS area.

Figure 28.2 Addresses for ID Code Stored

28.3 CPU Rewrite Mode

In CPU rewrite mode, the CPU executes software commands to rewrite the flash memory. The CPU accesses the flash memory not via the CPU buses, but via the dedicated flash memory rewrite buses (refer to Figure 28.3).

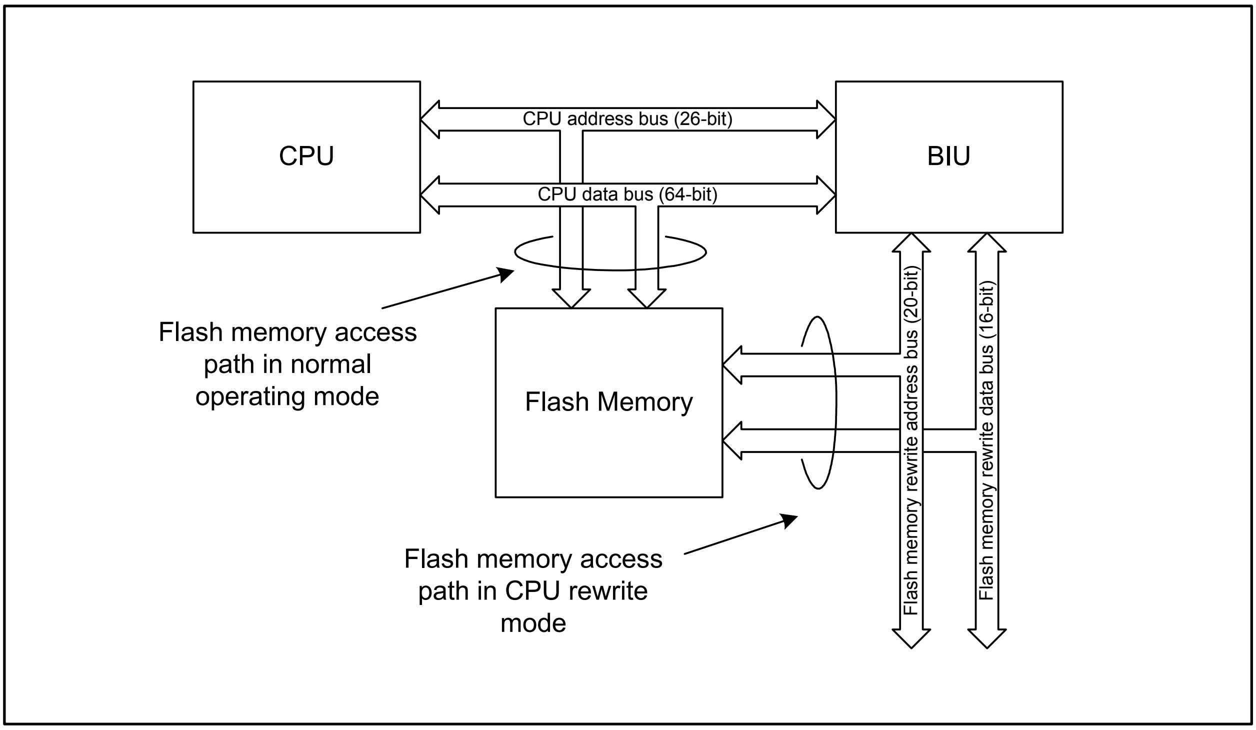


Figure 28.3 Flash Memory Access Path in CPU Rewrite Mode

Bus setting for flash memory rewrite should be performed by the FEBC register. Refer to 28.3.1 “Flash Memory Rewrite Bus Timing” and 29. “Electrical Characteristics” for the appropriate bus setting.

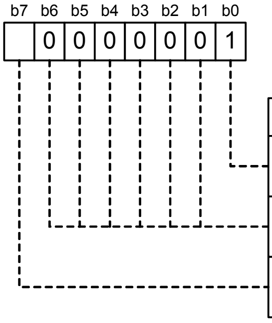
The CPU rewrite mode contains modes EW0 and EW1 as shown in Table 28.5.

Table 28.5 EW0 and EW1 Modes

Item	EW0 Mode	EW1 Mode
Rewrite program executable spaces	Spaces other than the on-chip flash memory	Internal spaces other than specified blocks to be rewritten, internal RAM
Restrictions on software commands	None	• Do not execute either the program command or the block erase command for blocks where the rewrite control programs are written to • Do not execute the enter read status register mode command • Execute the enter read lock bit status mode command in RAM • Execute the enter read protect bit status mode command in RAM
Mode after program/ erase operation	Read status register mode	Read array mode
CPU state during program/erase operation	Operating	In a hold state (I/O ports maintain the state before the command was executed)
Flash memory state detection by	• Reading the FMSR0 register by a program • Executing the enter read status register mode command to read data	• Reading the FMSR0 register by a program
Other restrictions	None	• The watchdog timer is disabled in count source protect mode • Disable interrupts (except NMI) and DMA transfer during program/erase operation

To select CPU rewrite mode, the FEW bit in the FMCR register should be set to 1. Then, EW0 mode/EW1 mode can be selected by setting the EWM bit in the FMR0 register.
Registers FMCR and FMR0 are protected by registers PRR and FPR0, respectively.
Figure 28.4 to Figure 28.11 show associated registers.

Flash Memory Control Register [1]



Symbol
FMCR

Address
0006h

Reset Value
0000 0001b

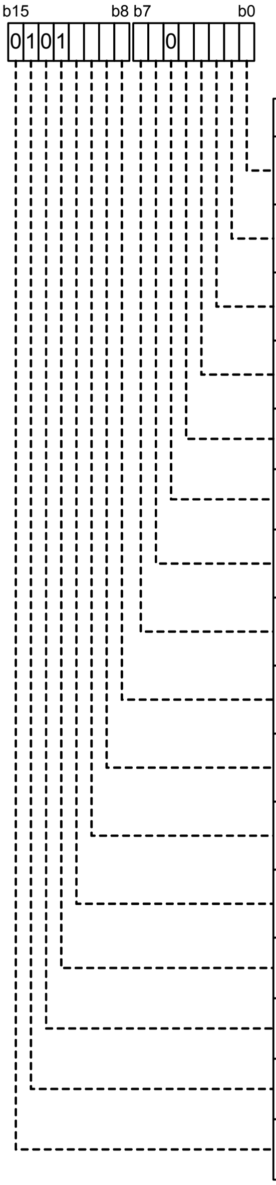
Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved	Should be written with 1	RW
— (b6-b1)	Reserved	Should be written with 0	RW
FEW	CPU Rewrite Mode Setting Bit [2]	0: Normal operating mode 1: CPU rewrite mode	RW

Notes:
1. Set the PRR register to AAh (write enabled) before rewriting this register.
2. When the MRS bit in the VRCR register is 1 (main regulator stopped), this bit should not be set to 1.

Figure 28.4 FMCR Register

Flash Memory Rewrite Bus Control Register [(1)]

Symbol	Address	Reset Value
FEBC	001Dh-001Ch	0000h



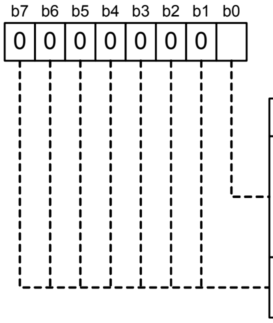
Bit Symbol	Bit Name	Function	RW
FWR0	$\overline{\text{RD}}$ Pulse Width Setting Bit	b3 b2 b1 b0 0 0 0 0 : *wr* = 1 0 0 0 1 : *wr* = 2 0 1 0 1 : *wr* = 3 0 1 1 0 : *wr* = 4 1 0 1 0 : *wr* = 5 1 0 1 1 : *wr* = 6 1 1 1 1 : *wr* = 7 Only use the combinations listed above	RW
FWR1			RW
FWR2			RW
FWR3			RW
FWR4	RD Pulse Width Extension Select Bit	0: No pulse width extension 1: Pulse width extension selected	RW
— (b5)	Reserved	Should be written with 0	RW
MPY0	Multiplied Cycle Setting Bit	b7 b6 0 0 : Do not use this combination 0 1 : Do not use this combination 1 0 : *mpy* = 3 1 1 : *mpy* = 4	RW
MPY1			RW
FSUW0	Address Setup Before WR Setting Bit	b9 b8 0 0 : *suw* = 0 0 1 : *suw* = 1 1 0 : *suw* = 2 1 1 : *suw* = 3	RW
FSUW1			
FWW0	WR Pulse Width Setting Bit	b11 b10 0 0 : *ww* = 1 0 1 : *ww* = 2 1 0 : *ww* = 3 1 1 : *ww* = 4	RW
FWW1			
— (b12)	Reserved	Should be written with 1	RW
— (b13)	Reserved	Should be written with 0	RW
— (b14)	Reserved	Should be written with 1	RW
— (b15)	Reserved	Should be written with 0	RW

Note:

1. Set the PRR register to AAh (write enabled) before rewriting this register.

Figure 28.5 FEBC Register

Flash Register Protection Unlock Register 0

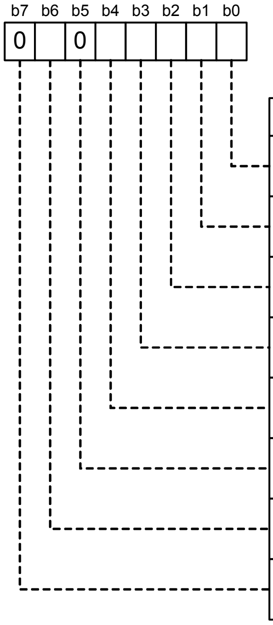


Symbol	Address	Reset Value
FPR0	40008h	0000 0000b

Bit Symbol	Bit Name	Function	RW
PR0	Protection Unlock Bit	For registers FMR0 and FMR1, 0: Write disabled 1: Write enabled	RW
— (b7-b1)	Reserved	Should be written with 0	RW

Figure 28.6 FPR0 Register

Flash Memory Control Register 0 (1, 2)



Symbol	Address	Reset Value
FMR0	40000h	0X01 XX00b

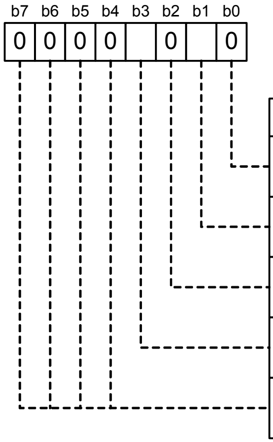
Bit Symbol	Bit Name	Function	RW
EWM	Rewrite Mode Select Bit	0: EW0 mode 1: EW1 mode	RW
LBM	Lock Bit Read Mode Setting Bit	0: Read via data bus (3) 1: Read by the LBS bit (4)	RW
LBS	Lock Bit Status Flag (4)	0: Locked 1: Unlocked	RO
RRDY	Read Ready Flag	0: Busy 1: Ready	RO
FCA	Final Command Accept Busy Flag	0: Final command accept ready 1: Final command accept busy	RO
— (b5)	Reserved	Should be written with 0	RW
— (b6)	Reserved	This bit is read as undefined value	RO
— (b7)	Reserved	Should be written with 0	RW

Notes:

1. Set the PR0 bit in the FPR0 register to 1 (write enabled) before rewriting this register.
2. This register is reset after exiting wait mode or stop mode.
3. After entering read lock bit status mode, if any even address in the corresponding block is read, the lock bit status is indicated in bit 6 of read data.
4. The LBS bit indicates the lock bit status by the read lock bit status command.

Figure 28.7 FMR0 Register

Flash Memory Control Register 1 [1]



Symbol FMR1

Address 40009h

Reset Value 0000 0010b

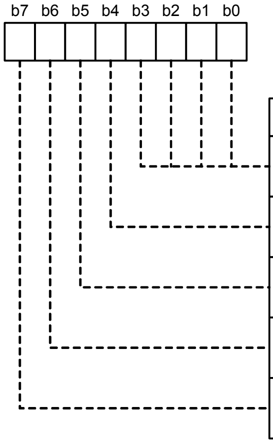
Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved	Should be written with 0	RW
RR	Reset Release Bit	0: Reset 1: Reset released	RW
— (b2)	Reserved	Should be written with 0	RW
LBD	Lock Bit Protect Disable Bit	0: Lock bit protection enabled 1: Lock bit protection disabled	RW
— (b7-b4)	Reserved	Should be written with 0	RW

Note:

1. Set the PR0 bit in the FPR0 register to 1 (write enabled) before rewriting this register.

Figure 28.8 FMR1 Register

Flash Memory Status Register 0



Symbol FMSR0

Address 40001h

Reset Value 1000 0000b

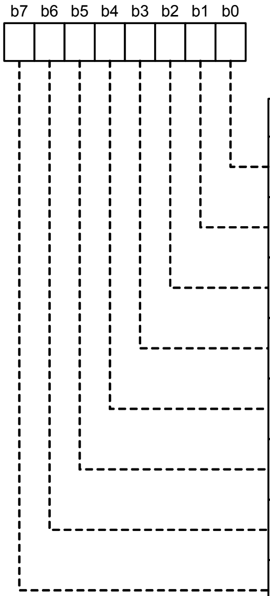
Bit Symbol	Bit Name	Function	RW
— (b3-b0)	Reserved	These bits are read as undefined value	RO
WERR	Program Error Flag	0: No program error 1: Program error occurred	RO
EERR	Erase Error Flag	0: No erase error 1: Erase error occurred [1]	RO
— (b6)	Reserved	This bit is read as undefined value	RO
RDY	Ready Flag	0: Busy 1: Ready	RO

Note:

1. If an erase error has occurred, issue the clear status register command first, then reissue the block erase command repeatedly until no more erase errors occur. After that, execute three more block erase operations.

Figure 28.9 FMSR0 Register

Block Protect Bit Monitor Register 0 (1)



Symbol	Address	Reset Value
FBPM0	4000Ah	??X? ????b (1)

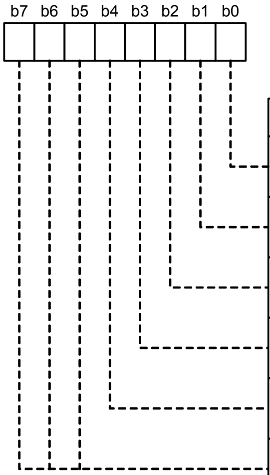
Bit Symbol	Bit Name	Function	RW
BP0	Block 0 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BP1	Block 1 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BP2	Block 2 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BP3	Block 3 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BP4	Block 4 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
— (b5)	Reserved	This bit is read as undefined value	RO
BP5	Block 5 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BP6	Block 6 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO

Note:

1. This register is updated only once after exiting reset mode. The protect bit status at that time is applied as reset value.

Figure 28.10 FBPM0 Register

Block Protect Bit Monitor Register 1 (1)



Symbol	Address	Reset Value
FBPM1	4000Bh	XXX? ????b (1)

Bit Symbol	Bit Name	Function	RW
BP7	Block 7 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BP8	Block 8 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BP9	Block 9 Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BPB	Block B Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
BPA	Block A Protect Bit Monitor Flag	0: Protected 1: Protection unlocked	RO
— (b7-b5)	Reserved	These bits are read as undefined value	RO

Note:

1. This register is updated only once after exiting reset mode. The protect bit status at that time is applied as the reset value.

Figure 28.11 FBPM1 Register

28.3.1 Flash Memory Rewrite Bus Timing

The bus setting for the flash memory rewrite is performed by setting the FEBC register. This section specifically describes the setting of FEBC register.

The reference clock is the base clock set with bits BCD1 and BCD0 in the CCR register. Time duration including tsu, tw, tc, and th are specified by the number of base clock cycles.

Table 28.6 to Table 28.8 show the correlation of the read cycle and setting of bits MPY1, MPY0, and FWR4 to FWR0, according to peripheral bus clock divide ratios. Table 28.9 to Table 28.11 show the correlation of the write cycle and setting of bits MPY1, MPY0, FSUW1, FSUW0, FWW1, and FWW0. Associated read/write timings are illustrated in Figure 28.12 and Figure 28.13, respectively.

Read/write cycle timing is selected from the tables below to meet the timing requirements in the CPU rewrite mode described in the electrical characteristics.

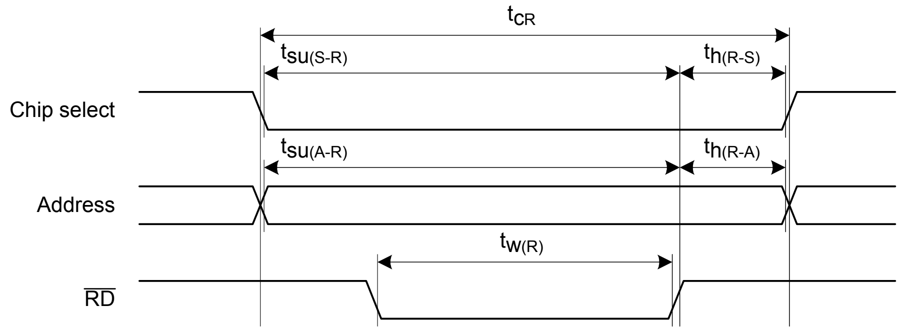


Figure 28.12 Read Timing

Table 28.6 Read Cycle and Bit Settings: MPY1, MPY0, and FWR4 to FWR0, When Peripheral Bus Clock is Divided by 2 (unit: cycles)

FWR3 to FWR0 Bit Settings		FWR4 Bit Settings	MPY1 and MPY0 Bit Settings							
			10b				11b			
			$mpy = 3$				$mpy = 4$			
			tsu(S-R), tsu(A-R)	tw(R)	tcR	th(R-S), th(R-A)	tsu(S-R), tsu(A-R)	tw(R)	tcR	th(R-S), th(R-A)
0000b	$wr = 1$	0	4	3	4	0	6	5	6	0
		1	6	5	6	0	6	5	6	0
0001b	$wr = 2$	0	8	7	8	0	10	9	10	0
		1	8	7	8	0	10	9	10	0
0101b	$wr = 3$	0	10	9	10	0	14	13	14	0
		1	12	11	12	0	14	13	14	0
0110b	$wr = 4$	0	14	13	14	0	18	17	18	0
		1	14	13	14	0	18	17	18	0
1010b	$wr = 5$	0	16	15	16	0	22	21	22	0
		1	18	17	18	0	22	21	22	0
1011b	$wr = 6$	0	20	19	20	0	26	25	26	0
		1	20	19	20	0	26	25	26	0
1111b	$wr = 7$	0	22	21	22	0	30	29	30	0
		1	24	23	24	0	30	29	30	0

Table 28.7 Read Cycle and Bit Settings: MPY1, MPY0, and FWR4 to FWR0, When Peripheral Bus Clock is Divided by 3 (unit: cycles)

FWR3 to FWR0 Bit Settings		FWR4 Bit Settings	MPY1 and MPY0 Bit Settings							
			10b				11b			
			$mpy = 3$				$mpy = 4$			
			tsu(S-R), tsu(A-R)	tw(R)	tcR	th(R-S), th(R-A)	tsu(S-R), tsu(A-R)	tw(R)	tcR	th(R-S), th(R-A)
0000b	$wr = 1$	0	6	4.5	6	0	6	4.5	6	0
		1	6	4.5	6	0	6	4.5	6	0
0001b	$wr = 2$	0	9	7.5	9	0	9	7.5	9	0
		1	9	7.5	9	0	12	10.5	12	0
0101b	$wr = 3$	0	12	10.5	12	0	15	13.5	15	0
		1	12	10.5	12	0	15	13.5	15	0
0110b	$wr = 4$	0	15	13.5	15	0	18	16.5	18	0
		1	15	13.5	15	0	18	16.5	18	0
1010b	$wr = 5$	0	18	16.5	18	0	21	19.5	21	0
		1	18	16.5	18	0	24	22.5	24	0
1011b	$wr = 6$	0	21	19.5	21	0	27	25.5	27	0
		1	21	19.5	21	0	27	25.5	27	0
1111b	$wr = 7$	0	24	22.5	24	0	30	28.5	30	0
		1	24	22.5	24	0	30	28.5	30	0

Table 28.8 Read Cycle and Bit Settings: MPY1, MPY0, and FWR4 to FWR0, When Peripheral Bus Clock is Divided by 4 (unit: cycles)

FWR3 to FWR0 Bit Settings		FWR4 Bit Settings	MPY1 and MPY0 Bit Settings							
			10b				11b			
			$mpy = 3$				$mpy = 4$			
			tsu(S-R), tsu(A-R)	tw(R)	tcR	th(R-S), th(R-A)	tsu(S-R), tsu(A-R)	tw(R)	tcR	th(R-S), th(R-A)
0000b	$wr = 1$	0	4	2	4	0	8	6	8	0
		1	8	6	8	0	8	6	8	0
0001b	$wr = 2$	0	8	6	8	0	12	10	12	0
		1	8	6	8	0	12	10	12	0
0101b	$wr = 3$	0	12	10	12	0	16	14	16	0
		1	12	10	12	0	16	14	16	0
0110b	$wr = 4$	0	16	14	16	0	20	18	20	0
		1	16	14	16	0	20	18	20	0
1010b	$wr = 5$	0	16	14	16	0	24	22	24	0
		1	20	18	20	0	24	22	24	0
1011b	$wr = 6$	0	20	18	20	0	28	26	28	0
		1	20	18	20	0	28	26	28	0
1111b	$wr = 7$	0	24	22	24	0	32	30	32	0
		1	24	22	24	0	32	30	32	0

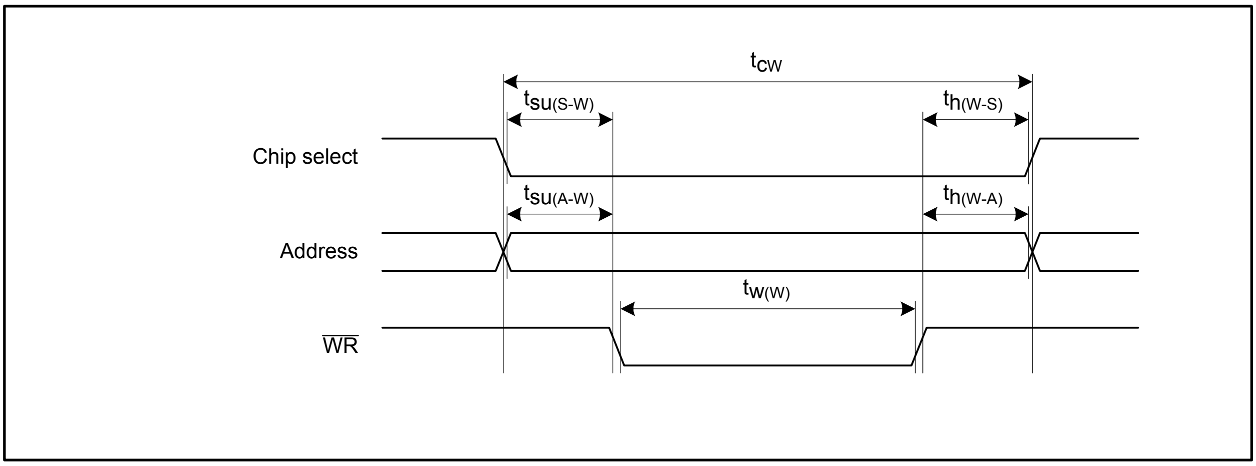


Figure 28.13 Write Timing

Table 28.9 Write Cycle and Bit Settings: MPY1, MPY0, FSUW1, FSUW0, FWW1, and FWW0, When Peripheral Bus Clock is Divided by 2 (unit: cycles)

FSUW1 and FSUW0 Bit Settings		FWW1 and FWW0 Bit Settings		MPY1 and MPY0 Bit Settings							
				10b				11b			
				$mpy = 3$				$mpy = 4$			
				tsu(S-W), tsu(A-W)	tw(W)	tcw	th(W-S), th(W-A)	tsu(S-W), tsu(A-W)	tw(W)	tcw	th(W-S), th(W-A)
00b	$suw = 0$	00b	$ww = 1$	1	3	6	2	1	4	6	1
		01b	$ww = 2$	1	6	8	1	1	8	10	1
		10b	$ww = 3$	1	9	12	2	1	12	14	1
		11b	$ww = 4$	1	12	14	1	1	16	18	1
01b	$suw = 1$	00b	$ww = 1$	4	3	8	1	5	4	10	1
		01b	$ww = 2$	4	6	12	2	5	8	14	1
		10b	$ww = 3$	4	9	14	1	5	12	18	1
		11b	$ww = 4$	4	12	18	2	5	16	22	1
10b	$suw = 2$	00b	$ww = 1$	7	3	12	2	9	4	14	1
		01b	$ww = 2$	7	6	14	1	9	8	18	1
		10b	$ww = 3$	7	9	18	2	9	12	22	1
		11b	$ww = 4$	7	12	20	1	9	16	26	1
11b	$suw = 3$	00b	$ww = 1$	10	3	14	1	13	4	18	1
		01b	$ww = 2$	10	6	18	2	13	8	22	1
		10b	$ww = 3$	10	9	20	1	13	12	26	1
		11b	$ww = 4$	10	12	24	2	13	16	30	1

Table 28.10 Write Cycle and Bit Settings: MPY1, MPY0, FSUW1, FSUW0, FWW1, and FWW0, When Peripheral Bus Clock is Divided by 3 (unit: cycles)

FSUW1 and FSUW0 Bit Settings		FWW1 and FWW0 Bit Settings		MPY1 and MPY0 Bit Settings							
				10b				11b			
				$mpy = 3$				$mpy = 4$			
				$t_{su(S-W)}$, $t_{su(A-W)}$	$t_{w(W)}$	t_{cW}	$t_{h(W-S)}$, $t_{h(W-A)}$	$t_{su(S-W)}$, $t_{su(A-W)}$	$t_{w(W)}$	t_{cW}	$t_{h(W-S)}$, $t_{h(W-A)}$
00b	$suw = 0$	00b	$ww = 1$	1	3	6	2	1	4	6	1
		01b	$ww = 2$	1	6	9	2	1	8	12	3
		10b	$ww = 3$	1	9	12	2	1	12	15	2
		11b	$ww = 4$	1	12	15	2	1	16	18	1
01b	$suw = 1$	00b	$ww = 1$	4	3	9	2	6	3	12	3
		01b	$ww = 2$	4	6	12	2	6	7	15	2
		10b	$ww = 3$	4	9	15	2	6	11	18	1
		11b	$ww = 4$	4	12	18	2	6	15	24	3
10b	$suw = 2$	00b	$ww = 1$	7	3	12	2	9	4	15	2
		01b	$ww = 2$	7	6	15	2	9	8	18	1
		10b	$ww = 3$	7	9	18	2	9	12	24	3
		11b	$ww = 4$	7	12	21	2	9	16	27	2
11b	$suw = 3$	00b	$ww = 1$	10	3	15	2	13	4	18	1
		01b	$ww = 2$	10	6	18	2	13	8	24	3
		10b	$ww = 3$	10	9	21	2	13	12	27	2
		11b	$ww = 4$	10	12	24	2	13	16	30	1

Table 28.11 Write Cycle and Bit Settings: MPY1, MPY0, FSUW1, FSUW0, FWW1, and FWW0, When Peripheral Bus Clock is Divided by 4 (unit: cycles)

FSUW1 and FSUW0 Bit Settings		FWW1 and FWW0 Bit Settings		MPY1 and MPY0 Bit Settings							
				10b				11b			
				$mpy = 3$				$mpy = 4$			
				$t_{su(S-W)}$, $t_{su(A-W)}$	$t_{w(W)}$	t_{cW}	$t_{h(W-S)}$, $t_{h(W-A)}$	$t_{su(S-W)}$, $t_{su(A-W)}$	$t_{w(W)}$	t_{cW}	$t_{h(W-S)}$, $t_{h(W-A)}$
00b	$suw = 0$	00b	$ww = 1$	1	3	8	4	1	4	8	3
		01b	$ww = 2$	1	6	8	1	1	8	12	3
		10b	$ww = 3$	1	9	12	2	1	12	16	3
		11b	$ww = 4$	1	12	16	3	1	16	20	3
01b	$suw = 1$	00b	$ww = 1$	4	3	8	1	5	4	12	3
		01b	$ww = 2$	4	6	12	2	5	8	16	3
		10b	$ww = 3$	4	9	16	3	5	12	20	3
		11b	$ww = 4$	4	12	20	4	5	16	24	3
10b	$suw = 2$	00b	$ww = 1$	8	2	12	2	9	4	16	3
		01b	$ww = 2$	8	5	16	3	9	8	20	3
		10b	$ww = 3$	8	8	20	4	9	12	24	3
		11b	$ww = 4$	8	11	20	1	9	16	28	3
11b	$suw = 3$	00b	$ww = 1$	10	3	16	3	13	4	20	3
		01b	$ww = 2$	10	6	20	4	13	8	24	3
		10b	$ww = 3$	10	9	20	1	13	12	28	3
		11b	$ww = 4$	10	12	24	2	13	16	32	3

28.3.2 Software Commands

In CPU rewrite mode, software commands enable rewrite and erase operations for the flash memory. Writing commands and reading/writing data should be performed in 16-bit units.
Table 28.12 lists the software commands.

Table 28.12 Software Commands

Command	First Command Cycle		Second Command Cycle	
	Address	Data	Address	Data
Enter read array mode	FFFFF800h	00FFh	—	—
Enter read status register mode (1)	FFFFF800h	0070h	—	—
Clear status register	FFFFF800h	0050h	—	—
Program (2)	FFFFF800h	0043h	WA	WD
Block erase	FFFFF800h	0020h	BA	00D0h
Lock bit program	FFFFF800h	0077h	BA	00D0h
Read lock bit status	FFFFF800h	0071h	BA	00D0h
Enter read lock bit status mode (3)	FFFFF800h	0071h	—	—
Protect bit program	FFFFF800h	0067h	PBA	00D0h
Enter read protect bit status mode (3)	FFFFF800h	0061h	—	—

WA: Even address to be written
WD: 16-bit data to be written
BA: Even address within a specific block
PBA: Protect bit address (refer to Table 28.4)

Notes:

1. This command cannot be executed in EW1 mode.
2. The program is performed in 64-bit (4-word) units. A sequence of commands consists of commands from the second to fifth. The upper 29 bits of the address WA should be fixed and the lower 3 bits of respective commands from the second to fifth should be set to 000b, 010b, 100b, and 110b for the addresses 0h, 2h, 4h, and 6h, or 8h, Ah, Ch, and Eh.
3. This command should be executed in RAM.

28.3.3 Mode Transition

CPU rewrite mode supports four flash memory operating modes:

- Read array mode
- Read status register mode
- Read lock bit status mode
- Read protect bit status mode

When reading the flash memory in these modes, the memory content, the status register content, the state of the lock bit in the read block, and the state of the protect bit are individually read. Details are listed in Table 28.13 to Table 28.15.

Table 28.13 Status Register

Bit	Bit Symbol	Bit Name	Definition	
			0	1
b15-b8	—	Disabled bit	—	—
b7	SR7	Sequencer status	BUSY	READY
b6	—	Reserved bit	—	—
b5	SR5	Erase status	Successfully completed	Error
b4	SR4	Program status	Successfully completed	Error
b3	—	Reserved bit	—	—
b2	—	Reserved bit	—	—
b1	—	Reserved bit	—	—
b0	—	Reserved bit	—	—

Table 28.14 Lock Bit Status

Bit	Bit Symbol	Bit Name	Definition	
			0	1
b15-b7	—	Disabled bit	—	—
b6	LBS	Lock bit status	Locked	Unlocked
b5-b0	—	Disabled bit	—	—

Table 28.15 Protect Bit Status

Bit	Bit Symbol	Bit Name	Definition	
			0	1
b15-b7	—	Disabled bit	—	—
b6	PBS	Protect bit status	Protected	Unprotected
b5-b0	—	Disabled bit	—	—

In these operating modes, program or erase operation can be performed by software commands. After an operation is completed, the flash memory module automatically enters read array mode (in EW1 mode) or read status register mode (in EW0 mode).

28.3.4 Issuing Software Commands

This section describes how to issue software commands.
These commands should be issued while the RDY bit in the FMSR0 register is 1 (ready).

28.3.4.1 Enter Read Array Mode Command

Execute this command to enter read array mode.
When 00FFh is written to address FFFFF800h, the flash memory enters read array mode. In this mode, the content stored to a given address in memory can be read.
In EW1 mode, the flash memory is always in read array mode.

28.3.4.2 Enter Read Status Register Mode

Execute this command to enter read status register mode.
When 0070h is written to address FFFFF800h, the status register content is read in any address of the flash memory.
Do not issue this command in EW1 mode.

28.3.4.3 Clear Status Register

Execute this command to reset the status register in the flash memory.
When 0050h is written to address FFFFF800h, bits SR5 and SR4 in the status register become 0 (successfully completed) (refer to Table 28.13). Consequently, bits EERR and WERR in the FMSR0 register become 0 (no errors).

28.3.4.4 Program Command

Execute this command to program the flash memory in 8-byte (4-word) units.

To start automatic programming (program and program-verify operations), write 0043h to address FFFFF800h, then write data to addresses 8n + 0 to 8n + 6. Verify that the FCA bit in the FMR0 register is 0 just before executing the final command.

To monitor the automatic program operation, read the RDY bit in the FMSR0 register. This bit becomes 0 (busy) when the operation is in progress and 1 (ready) when the operation is completed.

The operation result can be verified by the WERR bit in the FMSR0 register (refer to 28.3.5 “Status Check”).

Do not write additional data to an address that is already programmed.

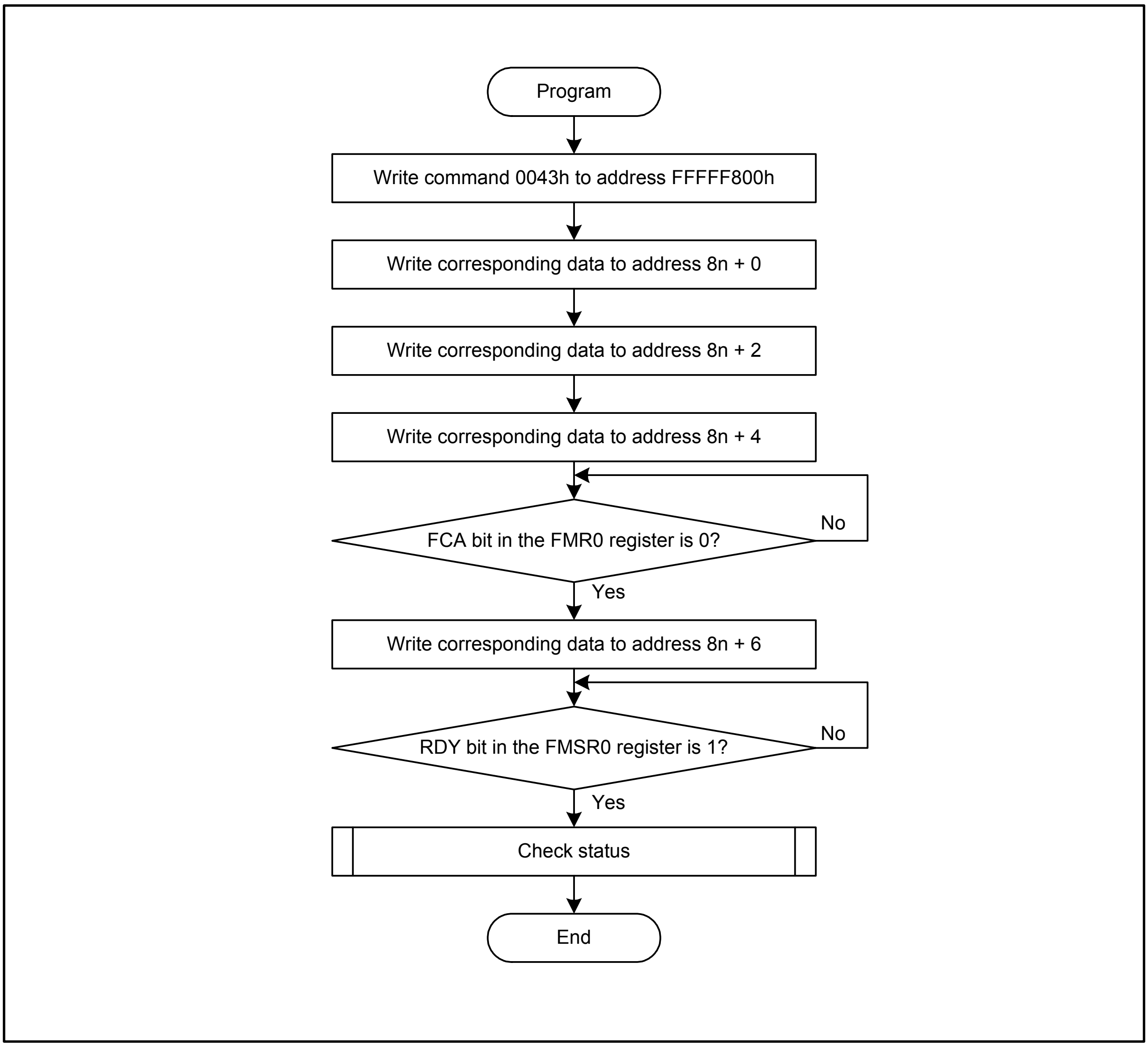


Figure 28.14 Program Command Execution Flowchart

28.3.4.5 Block Erase Command

Execute this command to erase a specified block in the flash memory.

To start automatic erasing of a specified block (erase and erase-verify operations), write 0020h to address FFFFF800h, verify that the FCA bit in the FMR0 register is 0, then write 00D0h to an even address in the corresponding block.

To monitor the automatic erase operation, read the RDY bit in the FMSR0 register. This bit becomes 0 (busy) when the operation is in progress and 1 (ready) when the operation is completed.

The operation result can be verified by the EERR bit in the FMSR0 register (refer to 28.3.5 "Status Check").

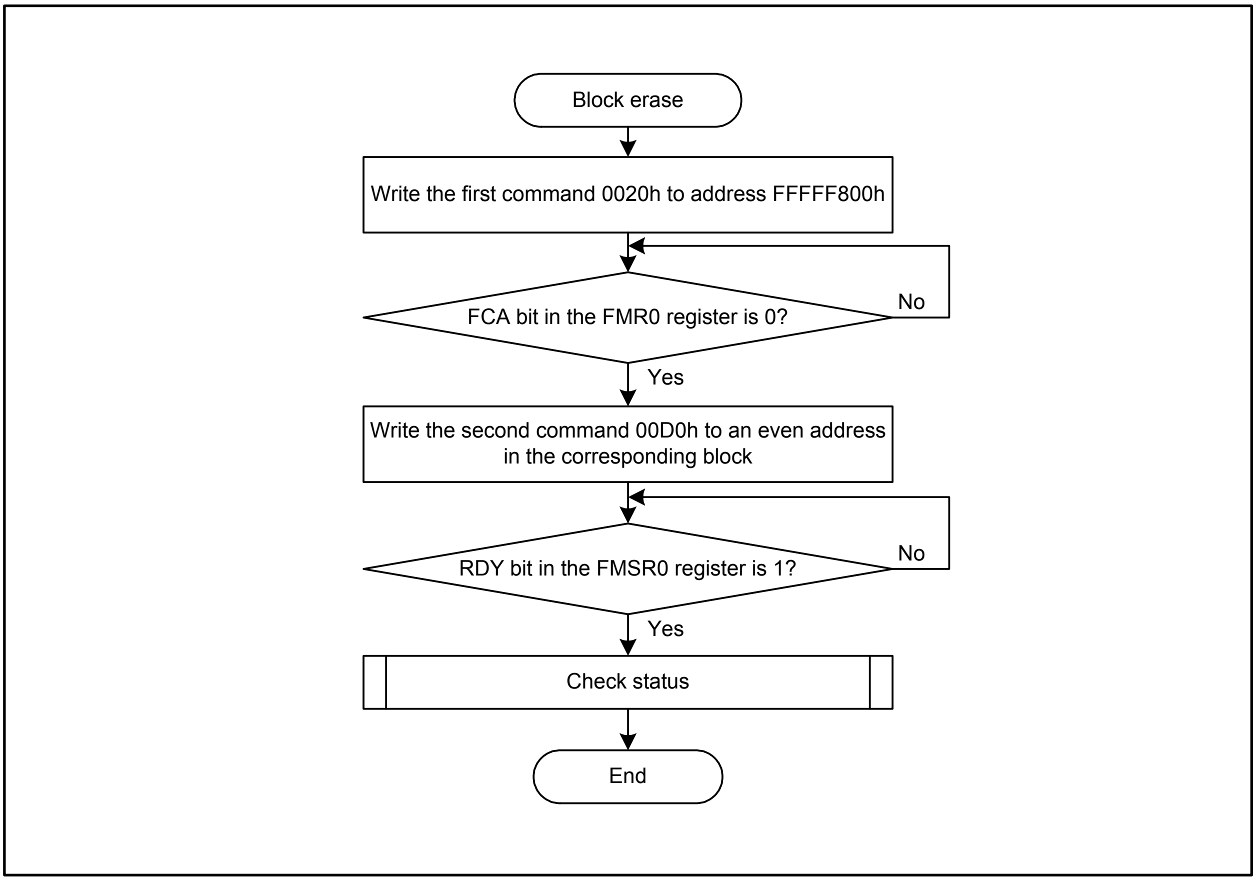


Figure 28.15 Block Erase Command Execution Flowchart

28.3.4.6 Lock Bit Program Command

Execute this command to lock a specified block in the flash memory.

To lock the block, write 0077h to address FFFFF800h, verify that the FCA bit in the FMR0 register is 0, then write 00D0h to an even address in the corresponding block. Then the lock bit of the block becomes 0 (locked).

To monitor the lock bit program, read the RDY bit in the FMSR0 register. This bit becomes 0 (busy) when the operation is in progress and 1 (ready) when the operation is completed.

The state of the lock bit can be verified by the read lock bit status command if the LBM bit in the FMR0 register is 1 (read by the LBS bit) (refer to 28.3.4.7 “Read Lock Bit Status Command”). If the LBM bit is 0 (read via data bus), enter read lock bit status mode (refer to 28.3.4.8 “Enter Read Lock Bit Status Mode Command”).

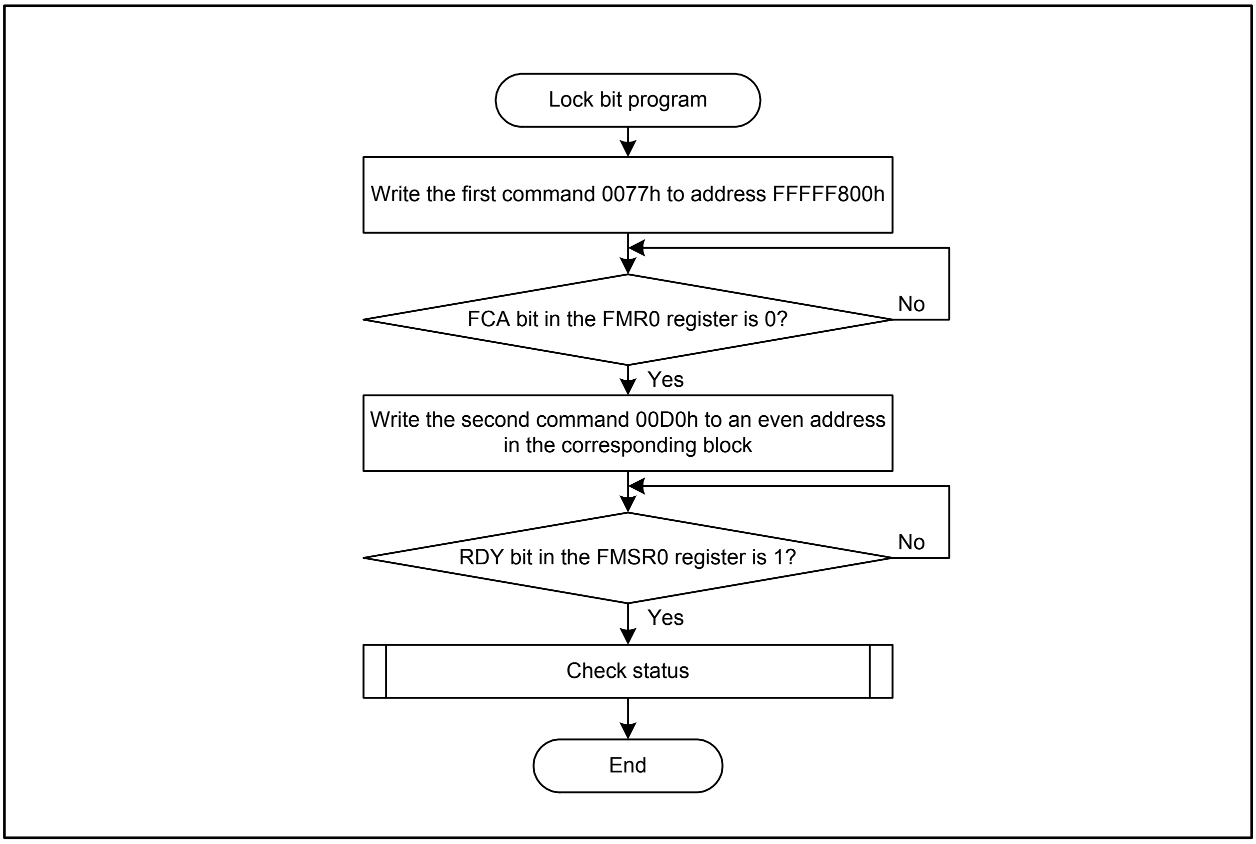


Figure 28.16 Lock Bit Program Command Execution Flowchart

28.3.4.7 Read Lock Bit Status Command

Execute this command to verify if a specified block in the flash memory is locked. This command can be used when the LBM bit in the FMR0 register is 1 (read by the LBS bit).

The LBS bit in the FMSR0 register reflects the lock bit status of the specified block when the following is performed: first write 0071h to address FFFFF800h and verify that the FCA bit in the FMR0 register becomes 0. Then write 00D0h to an even address of the corresponding block.

Read the LBS bit after the RDY bit in the FMSR0 register becomes 1 (ready).

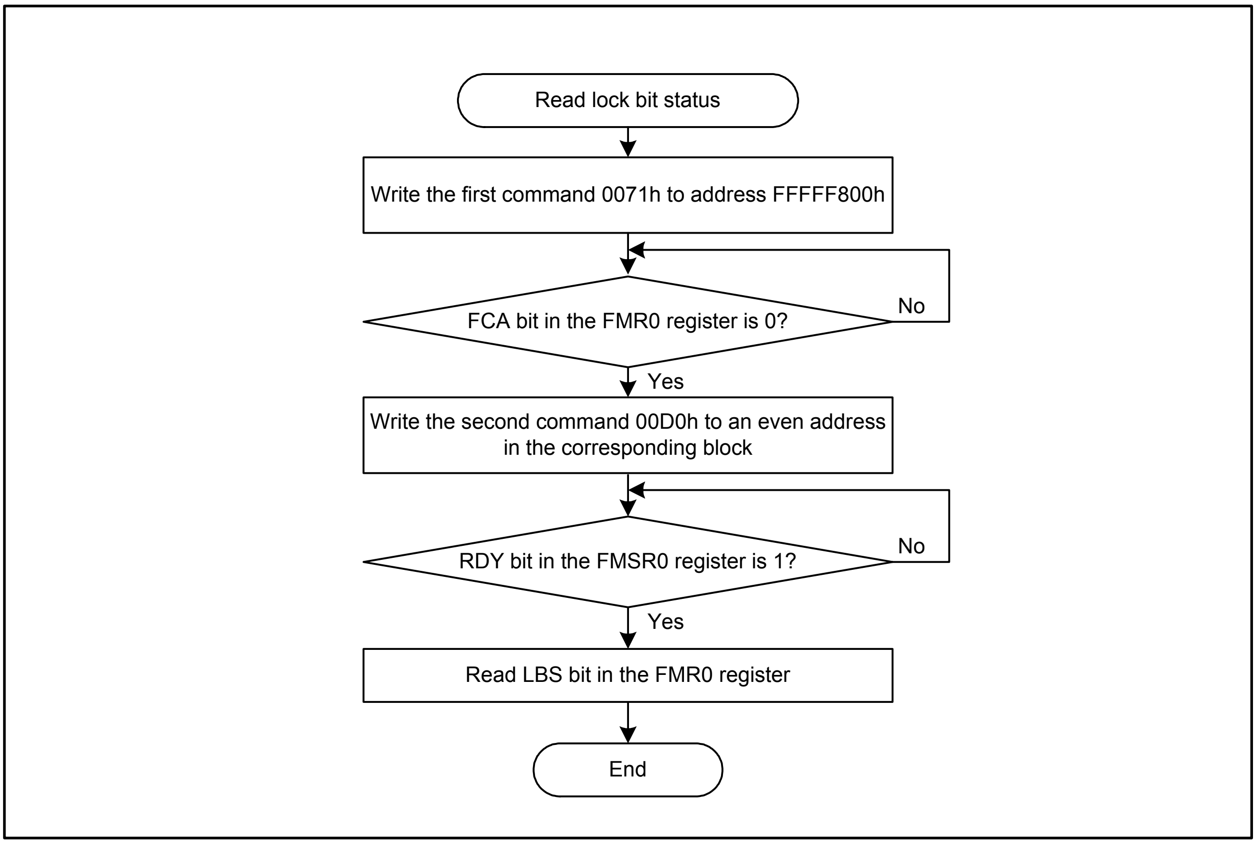


Figure 28.17 Read Lock Bit Status Command Execution Flowchart

28.3.4.8 Enter Read Lock Bit Status Mode Command

Execute this command to enter read lock bit status mode. This command is enabled when the LBM bit in the FMR0 register is 0 (read via data bus).

To read the lock bit status of the read block, write 0071h to address FFFFF800h (refer to Table 28.14). The status is read in any address of the flash memory.

Execute this command in RAM.

28.3.4.9 Protect Bit Program Command

Execute this command to protect a specific block in the flash memory. ROM code protection is enabled by setting one of the protect bits of the block to 0.

To set the protect bit of the designated block to 0 (protected), write 0067h to address FFFFF800h, verify that the FCA bit in the FMR0 register is 0, and then write 00D0h to the protect bit of the corresponding block (refer to Table 28.4).

To monitor the protect bit program, read the RDY bit in the FMSR0 register. This bit becomes 0 (busy) when the operation is in progress and 1 (ready) when the operation is completed.

To verify the state of protect bit, enter read protect bit status mode (refer to 28.3.4.10 "Enter Read Protect Bit Status Mode Command"), then read the flash memory.

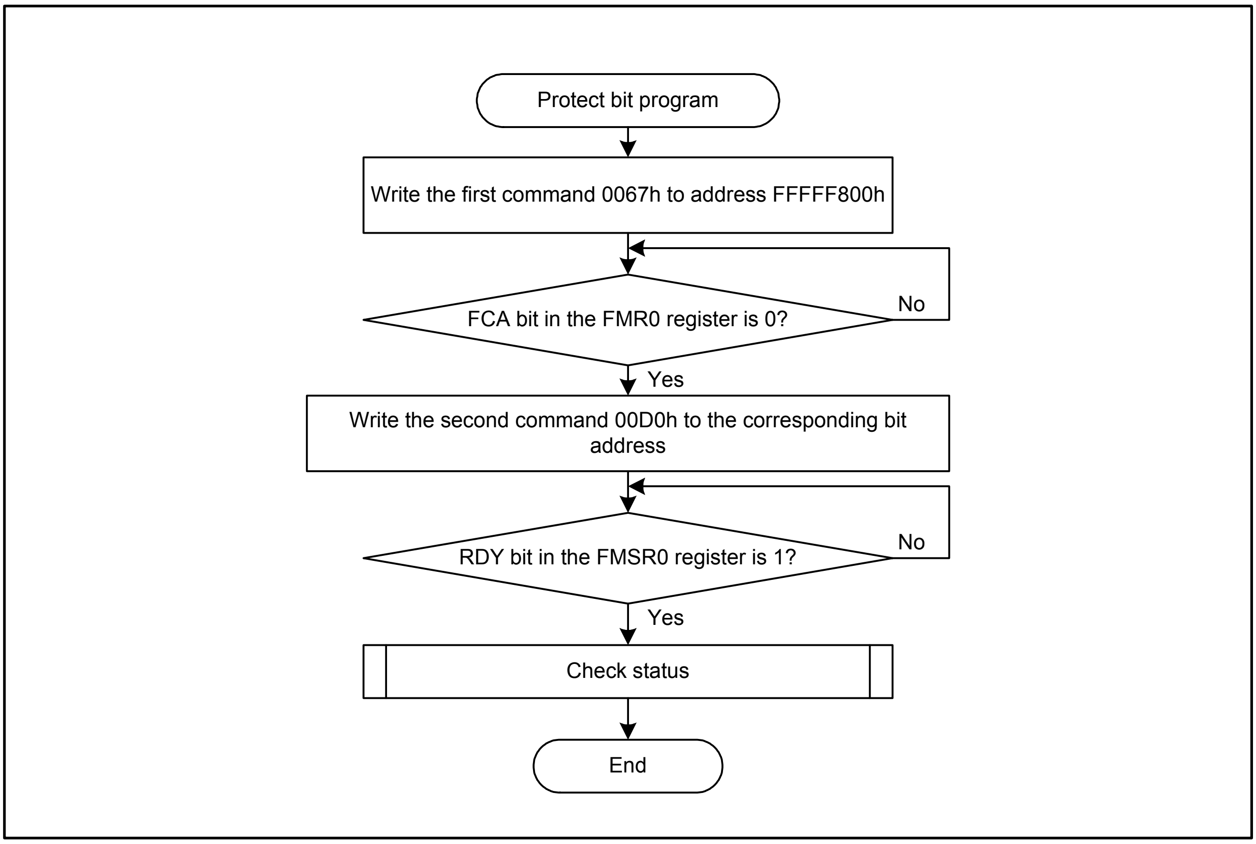


Figure 28.18 Protect Bit Program Command Execution Flowchart

28.3.4.10 Enter Read Protect Bit Status Mode Command

Execute this command to enter read protect bit status mode.

To read the protect bit status of the read block, write 0061h to address FFFFF800h (refer to Table 28.15). The status is read from any address in the flash memory.

Execute this command in RAM.

28.3.5 Status Check

To verify if a software command is successfully executed, read the EERR or WERR bit in the FMSR0 register, or the SR5 bit or SR4 bit in the status register.
Table 28.16 lists status and errors indicated by these bits and Figure 28.19 shows the flowchart of the status check.

Table 28.16 Status and Errors

FMSR0 Register (Status Register)		Error	Source of Error
EERR bit (SR5 bit)	WERR bit (SR4 bit)		
1	1	Command sequence error	• Data other than 00D0h or 00FFh (command to cancel) was written as the last command of two commands • An unavailable address was specified by an address specifying command
1	0	Erase error	• Attempted to erase a locked block • Corresponding block was not erased properly
0	1	Program error	• Attempted to program a locked block • Data was not programmed properly • Lock bit was not programmed properly • Protect bit was not programmed properly
0	0	No error	

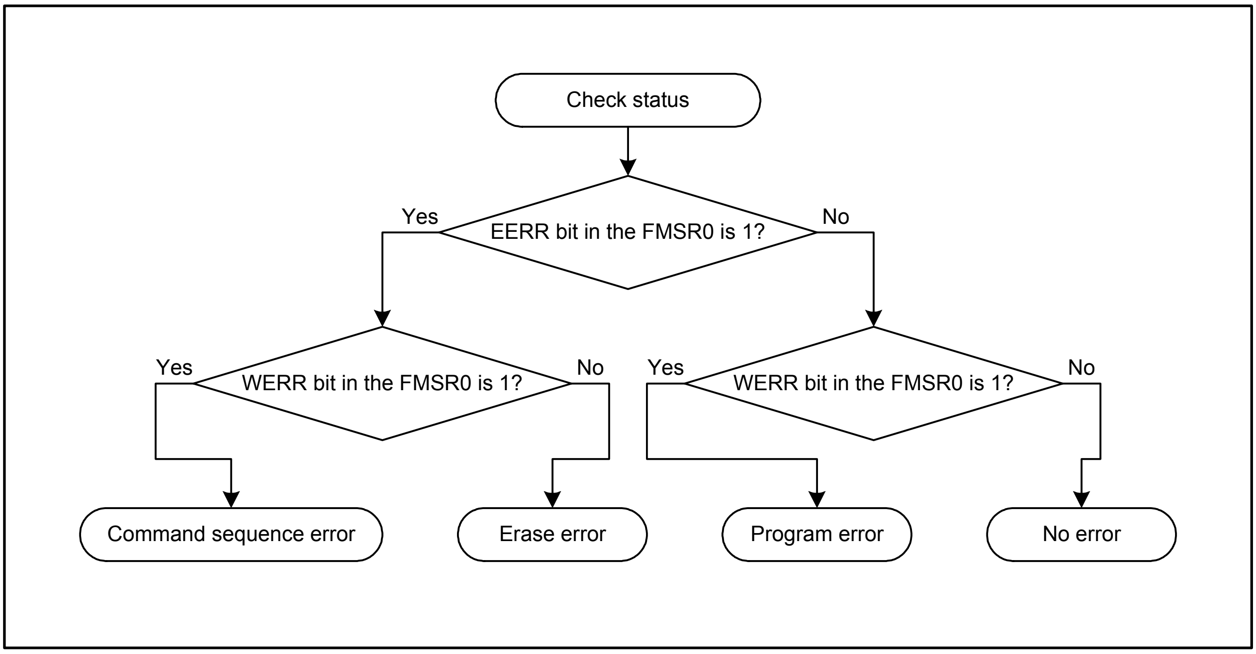


Figure 28.19 Status Check Flowchart

When an error occurs, execute the clear status register command and then handle the error.
If erase errors or program errors occur frequently even though the program is correct, the corresponding block may be disabled.

28.4 Standard Serial I/O Mode

In standard serial I/O mode, an R32C/142 Group and R32C/145 Group compatible serial programmer can be used to rewrite the flash memory while the MCU is mounted on a board.

For further information on the serial programmer, contact your serial programmer manufacturer and refer to the user's manual included with the serial programmer for instructions.

As shown in Table 28.17, this mode provides two types of transmit/receive mode: Standard serial I/O mode 1 which uses a synchronous serial interface, and standard serial I/O mode 2 which uses UART.

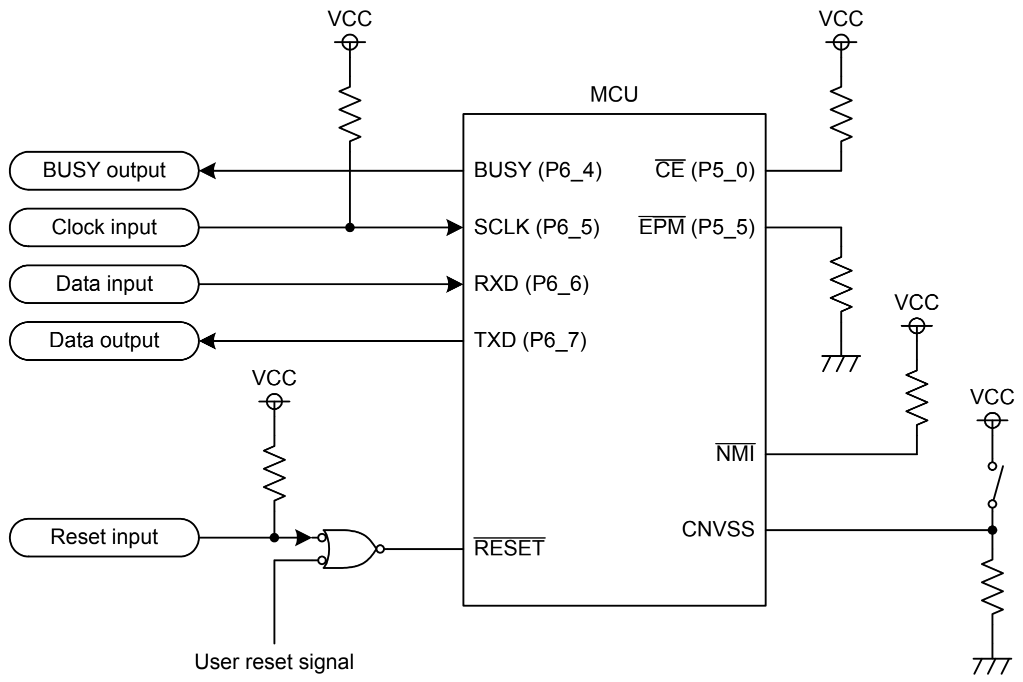
Table 28.17 Standard Serial I/O Mode Specifications

Item		Standard Serial I/O Mode 1	Standard Serial I/O Mode 2
Transmit/receive mode		Synchronous serial I/O	UART
Transmit/receive bit rate		High	Low
Serial interface to be used		UART1	UART1
Pin settings	CNVSS	High	High
	$\overline{\text{CE}}$ (P5_0)	High	High
	$\overline{\text{EPM}}$ (P5_5)	Low	Low
Pin functions	SCLK (P6_5)	In reset: Low In transmission/reception: Transmit/receive clock	In reset: Low In transmission/reception: Unused
	BUSY (P6_4)	BUSY signal	Monitor to check program operation
	RXD (P6_6)	Serial data input	Serial data input
	TXD (P6_7)	Serial data output	Serial data output

Table 28.18 lists the pin definitions and functions in standard serial I/O mode. Figure 28.20 and Figure 28.21 show examples of a circuit application in standard serial I/O modes 1 and 2, respectively. Refer to the serial programmer user manual to handle pins controlled by the serial programmer.

Table 28.18 Pin Definitions and Functions in Standard Serial I/O Mode

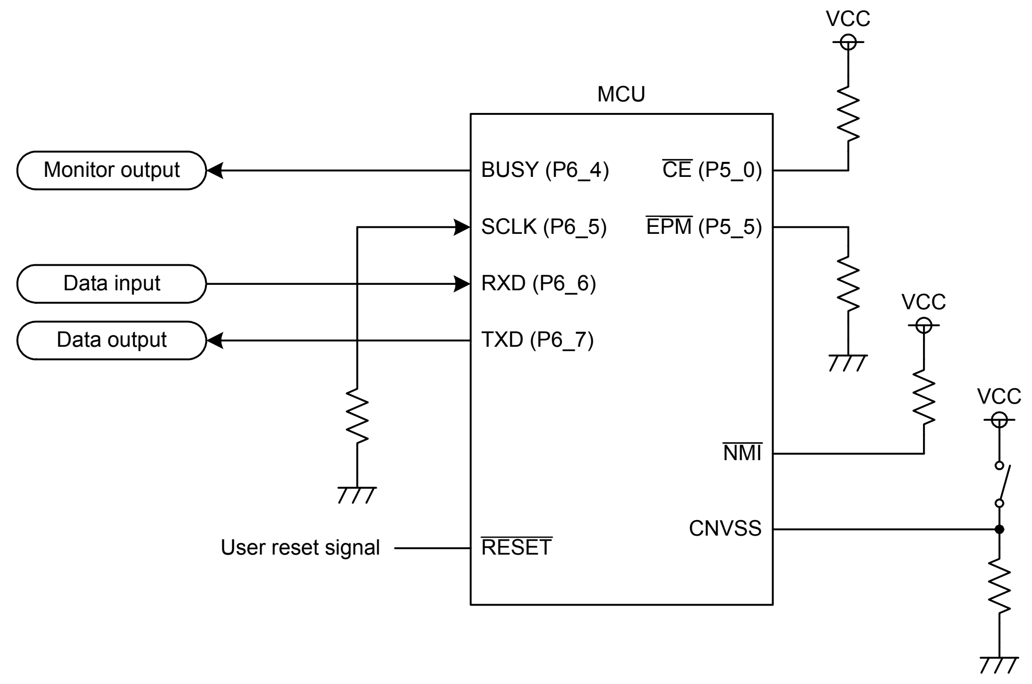
Pin Name	Function	I/O	Description
VCC, VCC0, VSS	Power supply input	I	Applicable as follows: VCC and VCC0 = guaranteed voltage for program/erase operations, VSS = 0 V
VDC1, VDC0	Connecting pins for decoupling capacitor	—	A decoupling capacitor for internal voltage should be connected between VDC0 and VDC1
CNVSS	CNVSS	I	This pin should be connected to VCC via a resistor
$\overline{\text{RESET}}$	Reset input	I	Reset input pin. While the $\overline{\text{RESET}}$ pin is driven low, a clock of 20 cycles or more should be input at the XIN pin
XIN	Main clock input	I	A ceramic resonator or a crystal oscillator should be connected between pins XIN and XOUT. An external clock should be input at XIN while leaving XOUT open
XOUT	Main clock output	O	
NSD	Debug port	I/O	This pin should be connected to VCC via a resistor of 1 to 4.7 kΩ
AVCC, AVSS	Analog power supply	I	AVCC and AVSS should be connected to VCC and VSS, respectively
VREF	Reference voltage input	I	Reference voltage input for the A/D converter and D/A converter
P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7	Input port	I	High or low should be input, or the ports should be left open
P5_0	$\overline{\text{CE}}$ input	I	High should be input
P5_1 to P5_4	Input port	I	High or low should be input, or the ports should be left open
P5_5	$\overline{\text{EPM}}$ input	I	Low should be input
P5_6, P5_7, P6_0 to P6_3	Input port	I	High or low should be input, or the ports should be left open
P6_4	BUSY output	O	Standard serial I/O mode 1: BUSY output pin Standard serial I/O mode 2: Program operation monitor
P6_5	SCLK input	I	Standard serial I/O mode 1: Serial clock input pin Standard serial I/O mode 2: Low should be input
P6_6	Data input RXD	I	Serial data input pin
P6_7	Data output TXD	O	Serial data output pin
P7_0 to P7_7, P8_0 to P8_4	Input port	I	High or low should be input, or the ports should be left open
P8_5	$\overline{\text{NMI}}$ input	I	This pin should be connected to VCC via a resistor
P8_6, P8_7, P9_1, P9_3 to P9_7, P10_0 to P10_7	Input port	I	High or low should be input, or the ports should be left open



Notes:

1. Control pins and external circuitry vary with the serial programmer. Refer to the user's manual included with the serial programmer.
2. In this example, a selector controls the voltage applied to the CNVSS pin to switch between single-chip mode and standard serial I/O mode 1.
3. If the user reset signal becomes low while the MCU is communicating with the serial programmer, cut off the connection between the user reset signal and the $\overline{\text{RESET}}$ pin by, for example, a jumper selector.

Figure 28.20 Circuit Application in Standard Serial I/O Mode 1



Notes:

1. Control pins and external circuitry vary with the serial programmer. Refer to the user’s manual included with the serial programmer.
2. In this example, a selector controls the voltage applied to the CNVSS pin to switch between single-chip mode and standard serial I/O mode 2.
3. If the user reset signal becomes low while the MCU is communicating with the serial programmer, cut off the connection between the user reset signal and the RESET pin by, for example, a jumper selector.

Figure 28.21 Circuit Application in Standard Serial I/O Mode 2

28.5 Parallel I/O mode

In parallel I/O mode, an R32C/142 Group and R32C/145 Group compatible parallel programmer can be used to rewrite the flash memory.

For further information on the parallel programmer, contact your parallel programmer manufacturer and refer to the user’s manual included with your parallel programmer for instructions.

28.6 Notes on Flash Memory Rewriting

28.6.1 Note on Power Supply

- Keep the supply voltage constant within the range specified in the electrical characteristics while a rewrite operation on the flash memory is in progress. If the supply voltage goes beyond the guaranteed value, the device cannot be guaranteed.

28.6.2 Note on Hardware Reset

- Do not perform a hardware reset while a rewrite operation on the flash memory is in progress.

28.6.3 Note on Flash Memory Protection

- If an ID code written in an assigned address has an error, any read/write operation on the flash memory in standard serial I/O mode is disabled.

28.6.4 Notes on Programming

- Do not set the FEW bit in the FMCR register to 1 (CPU rewrite mode) in low speed mode or low power mode.
- The program, block erase, lock bit program, and protect bit program are interrupted by an NMI, a watchdog timer interrupt, or an oscillator stop detection interrupt. If any of the software commands above are interrupted, erase the corresponding block and then execute the same command again. If the block erase command is interrupted, the lock bit and protect bit values become undefined. Therefore, disable the lock bit, and then execute the block erase command again.

28.6.5 Notes on Interrupts

- EW0 mode
  - To use interrupts assigned to the relocatable vector table, the vector table should be addressed in RAM space.
  - If an NMI, watchdog timer interrupt, or oscillator stop detection interrupt is generated, the flash memory module automatically enters read array mode. Therefore, these interrupts are enabled even during a rewrite operation. However, the rewrite operation in progress is aborted by the interrupts and registers FMR0 and FRSR0 are reset. When the interrupt handler has ended, set the LBD bit in the FMR1 register to 1 (lock bit protection disabled) to re-execute the rewrite operation.
  - Instructions BRK, INTO, and UND, which refer to data on the flash memory, cannot be used in this mode.

- EW1 mode
  - Interrupts assigned to the relocatable vector table should not be accepted during program or block erase operation.
  - The watchdog timer should not be used in count source protect mode. The watchdog timer interrupt should not be generated.
  - If an NMI, watchdog timer interrupt, or oscillator stop detection interrupt is generated, the flash memory module automatically enters read array mode. Therefore, these interrupts are enabled even during a rewrite operation. However, the rewrite operation in progress is aborted by the interrupts and registers FMR0 and FRSR0 are reset. When the interrupt handler has ended, set the EWM bit in the FMR0 register to 1 (EW1 mode) and the LBD bit in the FMR1 register to 1 (lock bit protection disabled) to re-execute the rewrite operation.

28.6.6 Notes on Rewrite Control Program

- EW0 mode
  - If the supply voltage drops during the rewrite operation of blocks having the rewrite control program, the rewrite control program may not be successfully rewritten, and the rewrite operation itself may not be performed. In this case, perform the rewrite operation by serial programmer or parallel programmer.

- EW1 mode
  - Do not rewrite blocks having the rewrite control program.

28.6.7 Notes on Number of Program/Erase Operations and Software Command Execution Time

- The time to execute software commands (program, block erase, lock bit program, and protect bit program) increases as the number of program/erase operations increases. If the number of program/erase operations exceeds the endurance value specified in the electrical characteristics, it may take an unpredictable amount of time to execute the software commands. The wait time for executing software commands should be set much longer than the execution time specified in the electrical characteristics.

28.6.8 Other Notes

- The minimum values of program and erase cycles specified in the electrical characteristics are the maximum values that can guarantee the initial performance of the flash memory. The program/erase operation may still be performed even if the number of program/erase oeprations exceeds the guaranteed values.
- Chips repeatedly programmed and erased for debugging should not be used for commercial products.

29. Electrical Characteristics

Table 29.1 Absolute Maximum Ratings (1)

Symbol	Characteristic		Condition	Value	Unit
V_{CC}	Supply voltage		$V_{CC} = AV_{CC}$	-0.3 to 6.0	V
V_{CC0}	Supply voltage		$V_{CC0} \leq V_{CC}$	-0.3 to 6.0	V
AV_{CC}	Analog supply voltage		$V_{CC} = AV_{CC}$	-0.3 to 6.0	V
V_I	Input voltage	XIN, $\overline{\text{RESET}}$, CNVSS, NSD, V_{REF}, P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_1, P9_3 to P9_7, P10_0 to P10_7		-0.3 to V_{CC} + 0.3	V
V_O	Output voltage	XOUT, P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7		-0.3 to V_{CC} + 0.3	V
P_d	Power consumption		T_a = 25°C	500	mW
			$T_a \geq$ 85°C	300	mW
—	Operating temperature range			-40 to 125	°C
T_{stg}	Storage temperature range			-65 to 150	°C

Note:

1. Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Table 29.2 Operating Conditions (1/6) (1)

Symbol	Characteristic		Value			Unit
			Min.	Typ.	Max.	
V_{CC}	Digital supply voltage		4.2	5.0	5.5	V
V_{CC0}	Digital supply voltage		3.0	3.3	V_{CC}	V
AV_{CC}	Analog supply voltage			V_{CC}		V
V_{REF}	Reference voltage		4.2		V_{CC}	V
V_{SS}	Digital ground voltage			0		V
AV_{SS}	Analog ground voltage			0		V
dV_{CC}/dt	V_{CC} ramp up rate (V_{CC} < 2.0 V)		0.05		2.5	V/ms
dV_{CC0}/dt	V_{CC0} ramp up rate (V_{CC0} < 2.0 V)		0.05		2.5	V/ms
V_{IH}	High level input voltage	XIN, $\overline{\text{RESET}}$, CNVSS, NSD	0.8 × V_{CC}		V_{CC}	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7 (2), P9_1, P9_3 to P9_7, P10_0 to P10_7	0.7 × V_{CC}		V_{CC}	V
V_{IL}	Low level input voltage	XIN, $\overline{\text{RESET}}$, CNVSS, NSD	0		0.2 × V_{CC}	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7 (2), P9_1, P9_3 to P9_7, P10_0 to P10_7	0		0.3 × V_{CC}	V
T_{opr}	Operating temperature range	J version	-40		85	°C
		L version	-40		105	°C
		K version	-40		125	°C

Notes:

1. The device is operationally guaranteed under these operating conditions.
2. V_{IH} and V_{IL} for P8_7 are specified for P8_7 as a programmable port. These values are not applicable for P8_7 as XCIN.

Table 29.3 Operating Conditions (2/6)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted) (1)

Symbol	Characteristic		Value (2)			Unit
			Min.	Typ.	Max.	
C_{VDC}	Decoupling capacitance for voltage regulator	Inter-pin voltage: 1.5 V	2.4		10.0	μF

Notes:

1. The device is operationally guaranteed under these operating conditions.
2. This value should be met with due consideration to the following conditions: operating temperature, DC bias, aging, etc.

Table 29.4 Operating Conditions (3/6)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted) (1)

Symbol	Characteristic		Value			Unit
			Min.	Typ.	Max.	
$I_{OH(peak)}$	High level peak output current (2)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7			-10.0	mA
$I_{OH(avg)}$	High level average output current (3)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7			-5.0	mA
$I_{OL(peak)}$	Low level peak output current (2)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7			10.0	mA
$I_{OL(avg)}$	Low level average output current (3)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7			5.0	mA

Notes:
1. The device is operationally guaranteed under these operating conditions.
2. The following conditions should be satisfied:
   - The sum of $I_{OL(peak)}$ of ports P0, P1, P2, P8_6, P8_7, P9, and P10 is 80 mA or less.
   - The sum of $I_{OL(peak)}$ of ports P3, P4, P5, P6, P7, and P8_0 to P8_4 is 80 mA or less.
   - The sum of $I_{OH(peak)}$ of ports P0, P1, and P2 is -40 mA or less.
   - The sum of $I_{OH(peak)}$ of ports P8_6, P8_7, P9, and P10 is -40 mA or less.
   - The sum of $I_{OH(peak)}$ of ports P3, P4, and P5 is -40 mA or less.
   - The sum of $I_{OH(peak)}$ of ports P6, P7, and P8_0 to P8_4 is -40 mA or less.
   - The sum of $I_{OL(peak)}$ of all ports is 80 mA or less.
   - The sum of $I_{OH(peak)}$ of all ports is -80 mA or less.
3. Average value within 100 ms.

Table 29.5 Operating Conditions (4/6)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted) (1)

Symbol	Characteristic	Value			Unit
		Min.	Typ.	Max.	
$f_{(XIN)}$	Main clock oscillator frequency	4		8	MHz
$f_{(XRef)}$	Reference clock frequency	2		4	MHz
$f_{(PLL)}$	PLL clock oscillator frequency	96		128	MHz
$f_{(Base)}$	Base clock frequency			64	MHz
$t_{c(Base)}$	Base clock cycle time	15.625			ns
$f_{(CPU)}$	CPU operating frequency			64	MHz
$t_{c(CPU)}$	CPU clock cycle time	15.625			ns
$f_{(BCLK)}$	Peripheral bus clock operating frequency			32	MHz
$t_{c(BCLK)}$	Peripheral bus clock cycle time	31.25			ns
$f_{(PER)}$	Peripheral clock source frequency			32	MHz
$f_{(XCIN)}$	Sub clock oscillator frequency		32.768	50	kHz

Note:
1. The device is operationally guaranteed under these operating conditions.

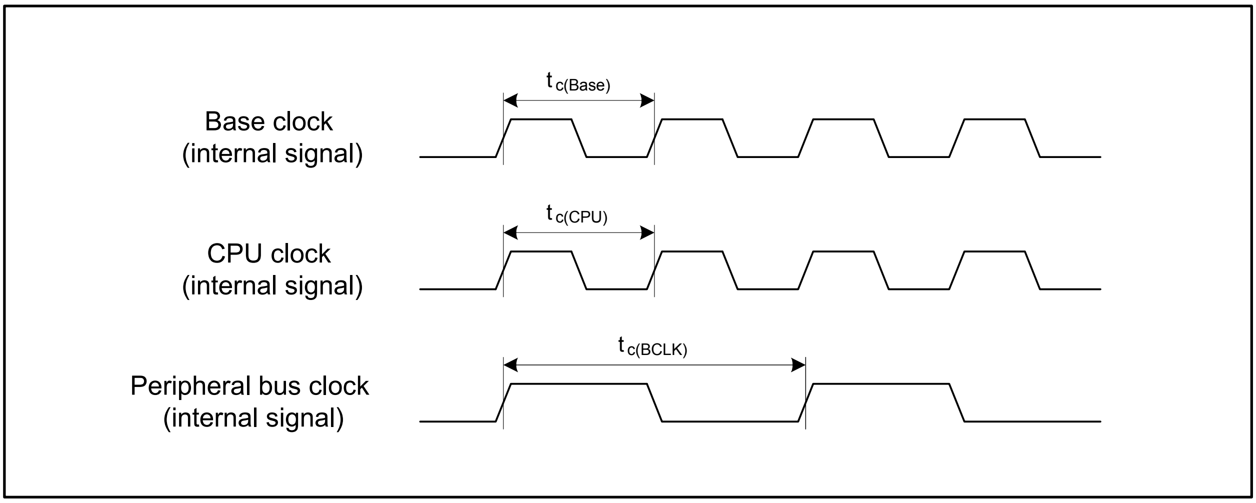


Figure 29.1 Clock Cycle Time

Table 29.6 Operating Conditions (5/6)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted) (1, 2)

Symbol	Characteristic		Measurement Condition	Value			Unit
				Min.	Typ.	Max.	
$I_{IC(H)}$	High input injection current	P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_5, P7_7, P8_0 to P8_4	$V_I > V_{CC}$			0.2	mA
$I_{IC(L)}$	Low input injection current	P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_5, P7_7, P8_0 to P8_4	$V_I < V_{SS}$			-0.2	mA
$\Sigma\|I_{IC}\|$	Total injection current					3.2	mA

Notes:
1. The device is operationally guaranteed under these operating conditions.
2. These conditions are applicable when each port is designated as input.

Table 29.7 Operating Conditions (6/6)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted) (1)

Symbol	Characteristic		Value			Unit
			Min.	Typ.	Max.	
$V_{r(VCC)}$	Allowable ripple voltage	V_{CC} = 5.0 V			0.5	Vp-p
$V_{r(VCC0)}$	Allowable ripple voltage	V_{CC0} = 5.0 V			0.5	Vp-p
		V_{CC0} = 3.3 V			0.3	Vp-p
$dV_{r(VCC)}/dt$	Ripple voltage gradient	V_{CC} = 5.0 V			±0.3	V/ms
$dV_{r(VCC0)}/dt$	Ripple voltage gradient	V_{CC0} = 5.0 V			±0.3	V/ms
		V_{CC0} = 3.3 V			±0.3	V/ms
$f_{r(VCC)}$	Allowable ripple frequency				10	kHz
$f_{r(VCC0)}$	Allowable ripple frequency				10	kHz

Note:
1. The device is operationally guaranteed under these operating conditions.

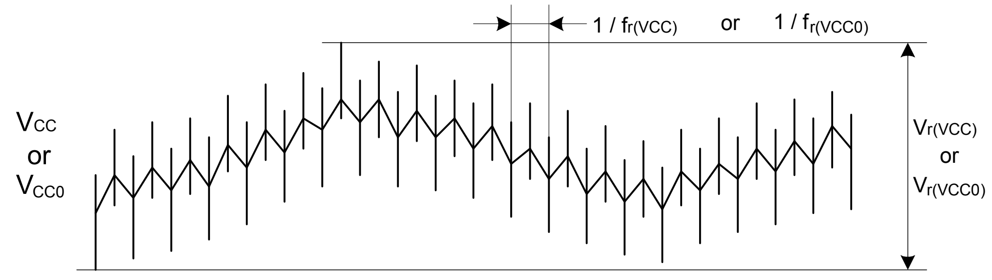


Figure 29.2 Ripple Waveform

Table 29.8 Electrical Characteristics of Flash Memory
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and Ta = T_{opr}, unless otherwise noted)

Symbol	Characteristic		Value			Unit
			Min.	Typ.	Max.	
—	Program and erase cycles (1)	Program area	100			Cycles
		Data area	100			Cycles
—	4-word program time	Program area		150	900	µs
		Data area		300	1700	µs
—	Lock bit-program time	Program area		70	500	µs
		Data area		140	1000	µs
—	Block erasure time	4-Kbyte block		0.12	3.0	s
		32-Kbyte block		0.17	3.0	s
		64-Kbyte block		0.20	3.0	s
—	Data retention (2)	T_a = 55°C (3, 4)	20			Years

Notes:

1. Program/erase definition
   This value represents the number of erasures per block.
   When the number of program and erase cycles is n, each block can be erased n times.
   For example, if a 4-word write is performed in 512 different addresses in the 4-Kbyte block A and then the block is erased, this is counted as a single program/erase operation.
   However, the same address cannot be written to more than once per erasure (overwrite disabled).
2. Data retention includes periods when no supply voltage is applied and no clock is provided.
3. This data retention includes 3000 hours in T_a = 125°C and 7000 hours in T_a = 85°C.
4. Contact a Renesas Electronics sales office for data retention times other than the above condition.

Table 29.9 Power Supply Circuit Timing Characteristics (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Symbol	Characteristic	Measurement Condition	Value			Unit
			Min.	Typ.	Max.	
$t_{d(P-R)}$	Internal power supply start-up stabilization time after the main power supply is turned on				2	ms

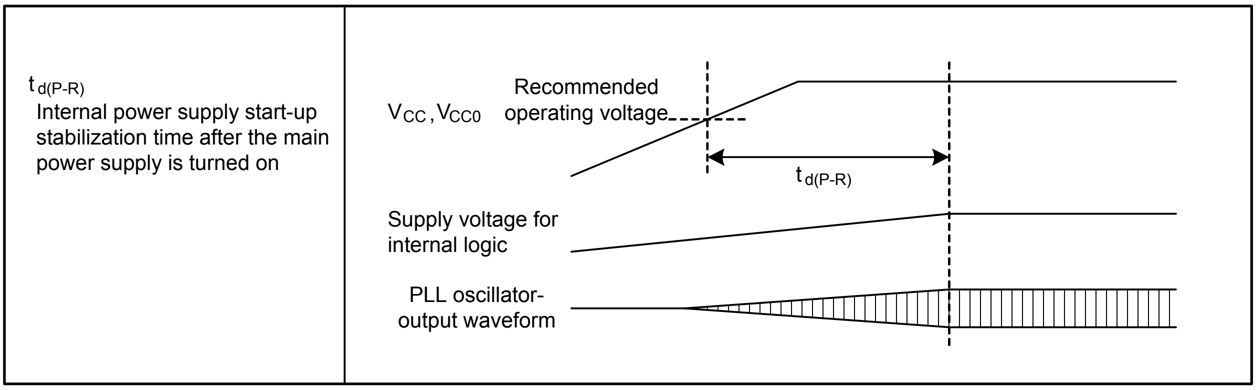


Figure 29.3 Power Supply Circuit Timing

Table 29.10 Electrical Characteristics of Voltage Regulator for Internal Logic (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Symbol	Characteristics	Measurement Condition	Value			Unit
			Min.	Typ.	Max.	
V_{VDC1}	Output voltage			1.5		V

Table 29.11 Electrical Characteristics of Oscillator (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Symbol	Characteristics	Measurement Condition	Value			Unit
			Min.	Typ.	Max.	
$f_{SO(PLL)}$	PLL clock self-oscillation frequency		35	50	65	MHz
$t_{LOCK(PLL)}$	PLL lock time (1)				2	ms
$t_{jitter(p-p)}$	PLL jitter period (p-p)				2.0	ns
$f_{(OCO)}$	On-chip oscillator frequency		94	125	156	kHz

Note:

1. This value is applicable only when the main clock oscillation is stable.

Table 29.12 Electrical Characteristics of Clock Circuitry (V_{CC} **= 4.2 to 5.5 V,** V_{CC0} **= 3.0 V to** V_{CC}**,** V_{SS} **= 0 V, and** T_a **=** T_{opr}**, unless otherwise noted)**

Symbol	Characteristics	Measurement Condition	Value			Unit
			Min.	Typ.	Max.	
$t_{rec(WAIT)}$	Recovery time from wait mode to low power mode				225	µs
$t_{rec(STOP)}$	Recovery time from stop mode (1)				225	µs

Note:

1. The stop mode recovery time does not include the main clock oscillation stabilization time. The CPU starts operating before the oscillator is stabilized.

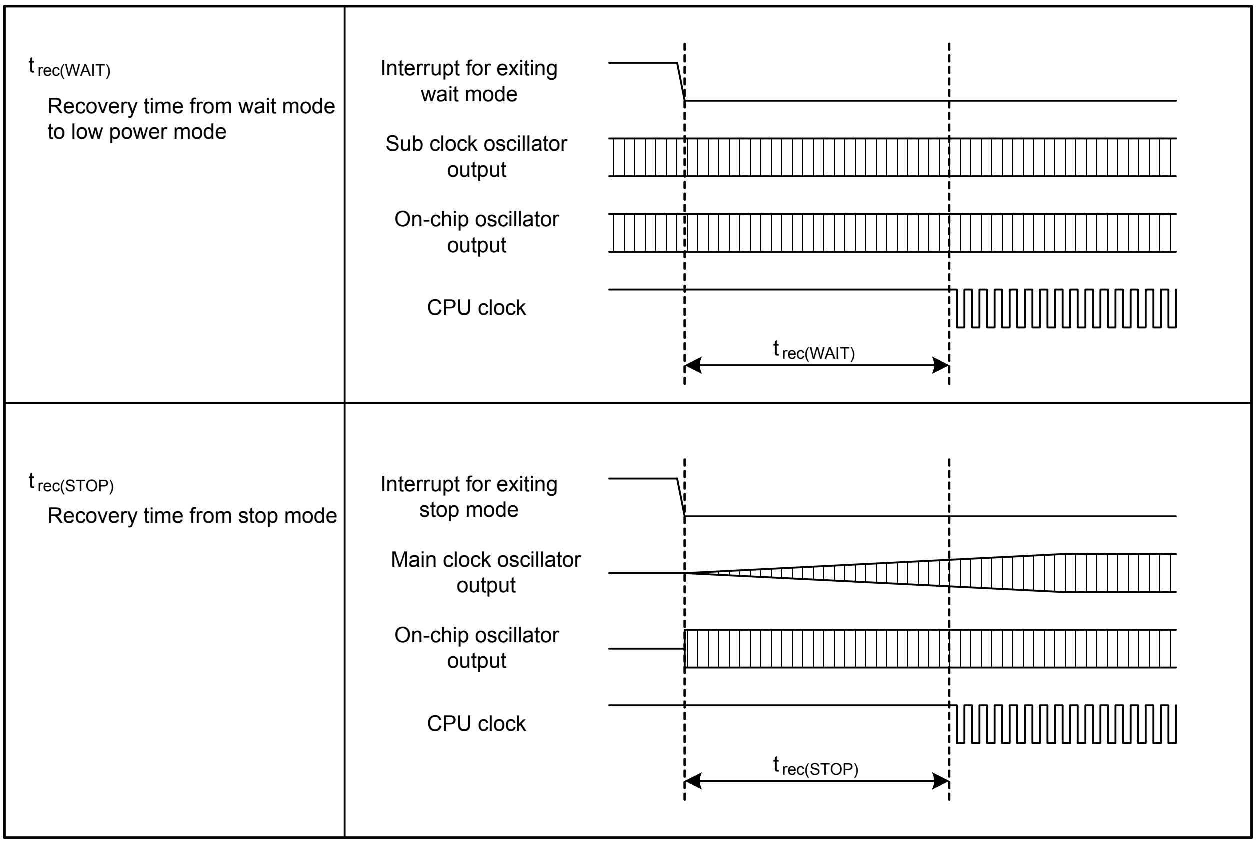


Figure 29.4 Clock Circuit Timing

Timing Requirements (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and Ta = T_{opr}, unless otherwise noted)

Table 29.13 Flash Memory CPU Rewrite Mode Timing

Symbol	Characteristics	Value		Unit
		Min.	Max.	
t_{cR}	Read cycle time	200		ns
$t_{su(S-R)}$	Chip-select setup time before read	200		ns
$t_{h(R-S)}$	Chip-select hold time after read	0		ns
$t_{su(A-R)}$	Address setup time before read	200		ns
$t_{h(R-A)}$	Address hold time after read	0		ns
$t_{w(R)}$	Read pulse width	100		ns
t_{cW}	Write cycle time	200		ns
$t_{su(S-W)}$	Chip-select setup time before write	0		ns
$t_{h(W-S)}$	Chip-select hold time after write	30		ns
$t_{su(A-W)}$	Address setup time before write	0		ns
$t_{h(W-A)}$	Address hold time after write	30		ns
$t_{w(W)}$	Write pulse width	50		ns

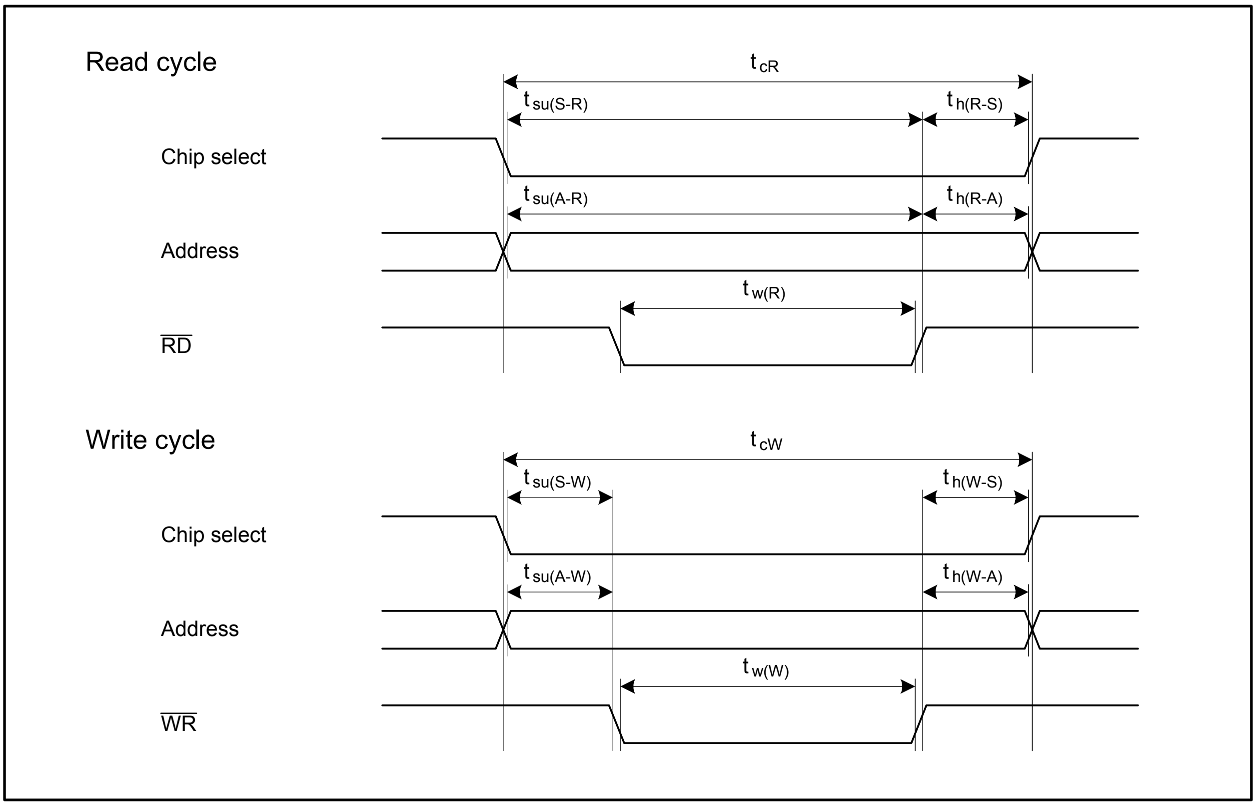


Figure 29.5 Flash Memory CPU Rewrite Mode Timing

V_{CC} = 5 V

Table 29.14 Electrical Characteristics (1/3)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, T_a = T_{opr}, and $f_{(CPU)}$ = 64 MHz, unless otherwise noted)

Symbol	Characteristic		Measurement Condition	Value			Unit
				Min.	Typ.	Max.	
V_{OH}	High level output voltage	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7	I_{OH} = -5 mA	V_{CC} - 2.0		V_{CC}	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7	I_{OH} = -200 μA	V_{CC} - 0.3		V_{CC}	V
V_{OL}	Low level output voltage	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7	I_{OL} = 5 mA			2.0	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_3 to P9_7, P10_0 to P10_7	I_{OL} = 200 μA			0.45	V

V_{CC} = 5 V

Table 29.15 Electrical Characteristics (2/3)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, T_a = T_{opr}, and $f_{(CPU)}$ = 64 MHz, unless otherwise noted)

Symbol	Characteristic		Measurement Condition	Value			Unit
				Min.	Typ.	Max.	
$V_{T+} - V_{T-}$	Hysteresis	$\overline{\text{NMI}}$, $\overline{\text{INT0}}$ to $\overline{\text{INT5}}$, $\overline{\text{KI0}}$ to $\overline{\text{KI3}}$, TA0IN to TA4IN, TA0OUT to TA4OUT, TB0IN to TB5IN, $\overline{\text{CTS0}}$ to $\overline{\text{CTS4}}$, CLK0 to CLK4, RXD0 to RXD4, SCL0 to SCL2, SDA0 to SDA2, $\overline{\text{SS0}}$ to $\overline{\text{SS2}}$, SRXD0 to SRXD2, $\overline{\text{ADTRG}}$, IIO0_0 to IIO0_7, IIO1_0 to IIO1_7, UD0A, UD0B, UD1A, UD1B, $\overline{\text{SCS0}}$, $\overline{\text{SCS1}}$, SSCK0, SSCK1, SSI0, SSI1, SSO0, SSO1, LIN0IN, LIN1IN, CAN0IN to CAN5IN, $\overline{\text{CAN0WU}}$ to $\overline{\text{CAN5WU}}$ (1)		0.2		1.0	V
		$\overline{\text{RESET}}$		0.2		1.8	V
I_{IH}	High level input current	XIN, $\overline{\text{RESET}}$, CNVSS, NSD, P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_1, P9_3 to P9_7, P10_0 to P10_7	V_I = 5 V			1.0	μA
I_{IL}	Low level input current	XIN, $\overline{\text{RESET}}$, CNVSS, NSD, P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_1, P9_3 to P9_7, P10_0 to P10_7	V_I = 0 V			-1.0	μA
R_{PULLUP}	Pull-up resistor	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_1, P9_3 to P9_7, P10_0 to P10_7	V_I = 0 V	30	50	170	kΩ
R_{fXIN}	Feedback resistor	XIN			1.5		MΩ
R_{fXCIN}	Feedback resistor	XCIN			15		MΩ

Note:

1. Pins CAN0IN, CAN1IN, CAN4IN, $\overline{\text{CAN0WU}}$, $\overline{\text{CAN1WU}}$, and $\overline{\text{CAN4WU}}$ are not available in the R32C/142 Group.

V_{CC} = 5 V

Table 29.16 Electrical Characteristics (3/3)
(V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and $T_a = T_{opr}$, unless otherwise noted)

Symbol	Characteristic	Measurement Condition		Value			Unit
				Min.	Typ.	Max.	
I_{CC0} (1)	Power supply current (V_{CC0} pin)	In single-chip mode, output pins are left open and others are connected to V_{SS} XIN-XOUT Drive power: high XCIN-XCOUT Drive power: low	$f_{(CPU)}$ = 64 MHz, $f_{(BCLK)}$ = 32 MHz, $f_{(XIN)}$ = 8 MHz, Active: XIN, PLL, Stopped: XCIN, OCO		36	60	mA
$I_{CC(V + A)}$ (1)	Power supply current (Pins V_{CC} and AV_{cc})		$f_{(CPU)}$ = 64 MHz, $f_{(BCLK)}$ = 32 MHz, $f_{(XIN)}$ = 8 MHz Active: XIN, PLL, Stopped: XCIN, OCO		10		mA
I_{CC}	Power supply current		$f_{(CPU)} = f_{SO(PLL)}/24$ MHz, Active: PLL (self-oscillation), Stopped: XIN, XCIN, OCO		7		mA
			$f_{(CPU)} = f_{(BCLK)} = f_{(XIN)}/256$ MHz, $f_{(XIN)}$ = 8 MHz, Active: XIN, Stopped: PLL, XCIN, OCO		1.2		mA
			$f_{(CPU)} = f_{(BCLK)}$ = 32.768 kHz, Active: XCIN, Stopped: XIN, PLL, OCO, Main regulator: shutdown		220		µA
			$f_{(CPU)} = f_{(BCLK)} = f_{(OCO)}/4$ kHz, Active: OCO, Stopped: XIN, PLL, XCIN, Main regulator: shutdown		230		µA
			$f_{(CPU)} = f_{(BCLK)} = f_{(XIN)}/256$ MHz, $f_{(XIN)}$ = 8 MHz, Active: XIN, Stopped: PLL, XCIN, OCO, T_a = 25°C, Wait mode		1070	2600	µA
			$f_{(CPU)} = f_{(BCLK)}$ = 32.768 kHz, Active: XCIN, Stopped: XIN, PLL, OCO, Main regulator: shutdown, T_a = 25°C, Wait mode		8	140	µA
			$f_{(CPU)} = f_{(BCLK)} = f_{(OCO)}/4$ kHz, Active: OCO, Stopped: XIN, PLL, XCIN, Main regulator: shutdown, T_a = 25°C, Wait mode		10	150	µA
			Stopped: all clocks, Main regulator: shutdown, T_a = 25°C		5	70	µA
			Stopped: all clocks, Main regulator: shutdown, T_a = 85°C			900	µA
			Stopped: all clocks, Main regulator: shutdown, T_a = 105°C			1800	µA
			Stopped: all clocks, Main regulator: shutdown, T_a = 125°C			3500	µA

Note:

1. The sum of $V_{CC0} \times I_{CC0}$ and $V_{CC} \times I_{CC(V+A)}$ should be less than P_d.

V_{CC} = 5 V

Table 29.17 A/D Conversion Characteristics (V_{CC} = AV_{CC} = V_{REF} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = AV_{SS} = 0 V, T_a = T_{opr}, and $f_{(BCLK)}$ = 32 MHz, unless otherwise noted)

Symbol	Characteristic	Measurement Condition		Value			Unit
				Min.	Typ.	Max.	
—	Resolution	V_{REF} = V_{CC}				10	Bits
—	Absolute error	V_{REF} = V_{CC} = 5 V	AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0, ANEX1			±3	LSB
			External op-amp connection mode			±7	LSB
INL	Integral non-linearity error	V_{REF} = V_{CC} = 5 V	AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0, ANEX1			±3	LSB
			External op-amp connection mode			±7	LSB
DNL	Differential non-linearity error					±1	LSB
—	Offset error					±3	LSB
—	Gain error					±3	LSB
R_{LADDER}	Resistor ladder	V_{REF} = V_{CC}		4		20	kΩ
t_{CONV}	Conversion time (10 bits)	ϕ_{AD} = 16 MHz, with sample and hold function		2.06			μs
		ϕ_{AD} = 16 MHz, without sample and hold function		3.69			μs
t_{CONV}	Conversion time (8 bits)	ϕ_{AD} = 16 MHz, with sample and hold function		1.75			μs
		ϕ_{AD} = 16 MHz, without sample and hold function		3.06			μs
t_{SAMP}	Sampling time	ϕ_{AD} = 16 MHz		0.188			μs
V_{IA}	Analog input voltage			0		V_{REF}	V
ϕ_{AD}	Operating clock frequency	Without sample and hold function		0.25		16	MHz
		With sample and hold function		1		16	MHz
$R_{PU(AST)}$	Pull-up resistor for open-circuit detection			5	10	15	kΩ
$R_{PD(AST)}$	Pull-down resistor for open-circuit detection			5	10	15	kΩ

V_{CC} = 5 V

Table 29.18 D/A Conversion Characteristics (V_{CC} = AV_{CC} = V_{REF} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = AV_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Symbol	Characteristic	Measurement Condition	Value			Unit
			Min.	Typ.	Max.	
—	Resolution				8	Bits
—	Absolute precision				1.0	%
t_S	Settling time				3	µs
R_O	Output resistance		4	10	20	kΩ
I_{VREF}	Reference input current	(1)			1.5	mA

Note:

1. One D/A converter is used. The DAi register (i = 0, 1) of the other unused converter is set to 00h. The resistor ladder for the A/D converter is not considered.
   Even when the VCUT bit in the AD0CON1 register is set to 0 (V_{REF} disconnected), I_{VREF} is supplied.

V_{CC} = 5 V

Timing Requirements (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Table 29.19 External Clock Input

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(X)}$	External clock input period	125	250	ns
$t_{w(XH)}$	External clock input high level pulse width	50		ns
$t_{w(XL)}$	External clock input low level pulse width	50		ns
$t_{r(X)}$	External clock input rise time		5	ns
$t_{f(X)}$	External clock input fall time		5	ns
t_w / t_c	External clock input duty	40	60	%

V_{CC} = 5 V

Timing Requirements (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Table 29.20 Timer A Input (counting input in event counter mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input clock cycle time	200		ns
$t_{w(TAH)}$	TAiIN input high level pulse width	80		ns
$t_{w(TAL)}$	TAiIN input low level pulse width	80		ns

Table 29.21 Timer A Input (gating input in timer mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input clock cycle time	400		ns
$t_{w(TAH)}$	TAiIN input high level pulse width	180		ns
$t_{w(TAL)}$	TAiIN input low level pulse width	180		ns

Table 29.22 Timer A Input (external trigger input in one-shot timer mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input clock cycle time	200		ns
$t_{w(TAH)}$	TAiIN input high level pulse width	80		ns
$t_{w(TAL)}$	TAiIN input low level pulse width	80		ns

Table 29.23 Timer A Input (external trigger input in pulse-width modulation mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN input high level pulse width	80		ns
$t_{w(TAL)}$	TAiIN input low level pulse width	80		ns

Table 29.24 Timer A Input (increment/decrement switching input in event counter mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(UP)}$	TAiOUT input clock cycle time	2000		ns
$t_{w(UPH)}$	TAiOUT input high level pulse width	1000		ns
$t_{w(UPL)}$	TAiOUT input low level pulse width	1000		ns
$t_{su(UP-TIN)}$	TAiOUT input setup time	400		ns
$t_{h(TIN-UP)}$	TAiOUT input hold time	400		ns

V_{CC} = 5 V

Timing Requirements (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Table 29.25 Timer B Input (counting input in event counter mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input clock cycle time (one edge counting)	200		ns
$t_{w(TBH)}$	TBiIN input high level pulse width (one edge counting)	80		ns
$t_{w(TBL)}$	TBiIN input low level pulse width (one edge counting)	80		ns
$t_{c(TB)}$	TBiIN input clock cycle time (both edges counting)	200		ns
$t_{w(TBH)}$	TBiIN input high level pulse width (both edges counting)	80		ns
$t_{w(TBL)}$	TBiIN input low level pulse width (both edges counting)	80		ns

Table 29.26 Timer B Input (pulse period measure mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input clock cycle time	400		ns
$t_{w(TBH)}$	TBiIN input high level pulse width	180		ns
$t_{w(TBL)}$	TBiIN input low level pulse width	180		ns

Table 29.27 Timer B Input (pulse-width measure mode)

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input clock cycle time	400		ns
$t_{w(TBH)}$	TBiIN input high level pulse width	180		ns
$t_{w(TBL)}$	TBiIN input low level pulse width	180		ns

V_{CC} = 5 V

Timing Requirements (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Table 29.28 Serial Interface

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi input clock cycle time	200		ns
$t_{w(CKH)}$	CLKi input high level pulse width	80		ns
$t_{w(CKL)}$	CLKi input low level pulse width	80		ns
$t_{su(D-C)}$	RXDi input setup time	80		ns
$t_{h(C-D)}$	RXDi input hold time	90		ns

Table 29.29 A/D Trigger Input

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$t_{w(ADH)}$	$\overline{\text{ADTRG}}$ input high level pulse width Hardware trigger input high level pulse width	$\frac{3}{\phi_{AD}}$		ns
$t_{w(ADL)}$	$\overline{\text{ADTRG}}$ input low level pulse width Hardware trigger input high level pulse width	125		ns

Table 29.30 External Interrupt $\overline{\text{INTi}}$ Input

Symbol	Characteristic		Value		Unit
			Min.	Max.	
$t_{w(INH)}$	$\overline{\text{INTi}}$ input high level pulse width (1)	Edge sensitive	250		ns
		Level sensitive	$t_{c(CPU)}$ + 200		ns
$t_{w(INL)}$	$\overline{\text{INTi}}$ input low level pulse width (1)	Edge sensitive	250		ns
		Level sensitive	$t_{c(CPU)}$ + 200		ns

Note:

1. The values are applied in case the filtering function is disabled.

V_{CC} = 5 V

Timing Requirements (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Table 29.31 Serial Bus Interface

Symbol	Characteristic	Value		Unit
		Min.	Max.	
$f_{(SSCK)}$	SSCKi frequency		4	MHz
$t_{c(SSCK)}$	SSCKi clock cycle time	250		ns
$t_{w(SSCKH)}$	SSCKi input high level pulse width	0.35 × $t_{c(SSCK)}$	0.6 × $t_{c(SSCK)}$	ns
$t_{w(SSCKL)}$	SSCKi input low level pulse width	0.35 × $t_{c(SSCK)}$	0.6 × $t_{c(SSCK)}$	ns
$t_{r(SSCK)}$	SSCKi input rising time		1	μs
$t_{f(SSCK)}$	SSCKi input falling time		1	μs
$t_{su(SCS-SSCK)}$	SCSi input setup time	$t_{c(BCLK)}$ + 50		ns
$t_{h(SSCK-SCS)}$	SCSi input hold time	$t_{c(BCLK)}$ + 50		ns
$t_{su(SSI-SSCK)}$	SSI input setup time	80		ns
$t_{h(SSCK-SSI)}$	SSI input hold time	10		ns
$t_{su(SSO-SSCK)}$	SSO input setup time	80		ns
$t_{h(SSCK-SSO)}$	SSO input hold time	20		ns

V_{CC} = 5 V

Switching Characteristics (V_{CC} = 4.2 to 5.5 V, V_{CC0} = 3.0 V to V_{CC}, V_{SS} = 0 V, and T_a = T_{opr}, unless otherwise noted)

Table 29.32 Serial Interface

Symbol	Characteristic	Measurement Condition	Value		Unit
			Min.	Max.	
$t_{d(C-Q)}$	TXDi output delay time	Refer to Figure 29.6		80	ns
$t_{h(C-Q)}$	TXDi output hold time		0		ns

Table 29.33 Serial Bus Interface

Symbol	Characteristic	Measurement Condition	Value		Unit
			Min.	Max.	
$t_{w(SSCKH)}$	SSCKi output high level pulse width	Refer to Figure 29.6	0.35 × $t_{c(SSCK)}$	0.6 × $t_{c(SSCK)}$	ns
$t_{w(SSCKL)}$	SSCKi output low level pulse width		0.35 × $t_{c(SSCK)}$	0.6 × $t_{c(SSCK)}$	ns
$t_{r(SSCK)}$	SSCKi output rising time			20	ns
$t_{f(SSCK)}$	SSCKi output falling time			20	ns
$t_{d(SCS-SSCK)}$	SSCKi output delay time for SCSi			0.5 × $t_{c(SSCK)}$ + 20	ns
$t_{d(SSCK-SCS)}$	SCSi output delay time for SSCKi		0.5 × $t_{c(SSCK)}$ - 20		ns
$t_{en(SCS-SSO)}$	SSOi output enable time			1.5 × $t_{c(BCLK)}$ + 100	ns
$t_{dis(SCS-SSO)}$	SSOi output disable time			1.5 × $t_{c(BCLK)}$ + 100	ns
$t_{en(SCS-SSI)}$	SSIi output enable time			1.5 × $t_{c(BCLK)}$ + 100	ns
$t_{dis(SCS-SSI)}$	SSIi output disable time			1.5 × $t_{c(BCLK)}$ + 100	ns
$t_{d(SSCK-SSO)}$	SSOi output delay time for SSCKi			30	ns
$t_{d(SSCK-SSI)}$	SSIi output delay time for SSCKi			85	ns
$t_{rec(SCS)}$	SCSi output high level period in continuous transmission			0.625 × $t_{c(SSCK)}$	ns

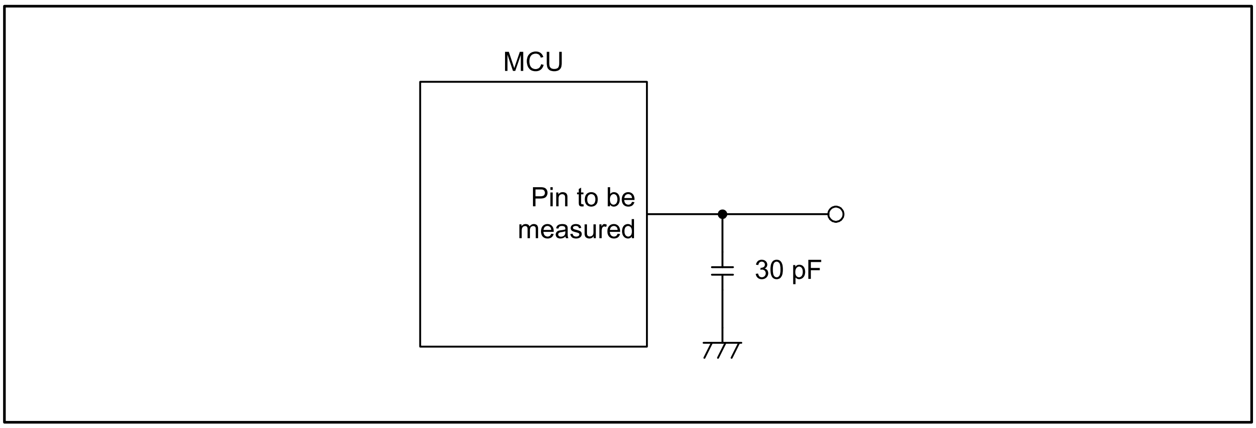


Figure 29.6 Switching Characteristic Measurement Circuit

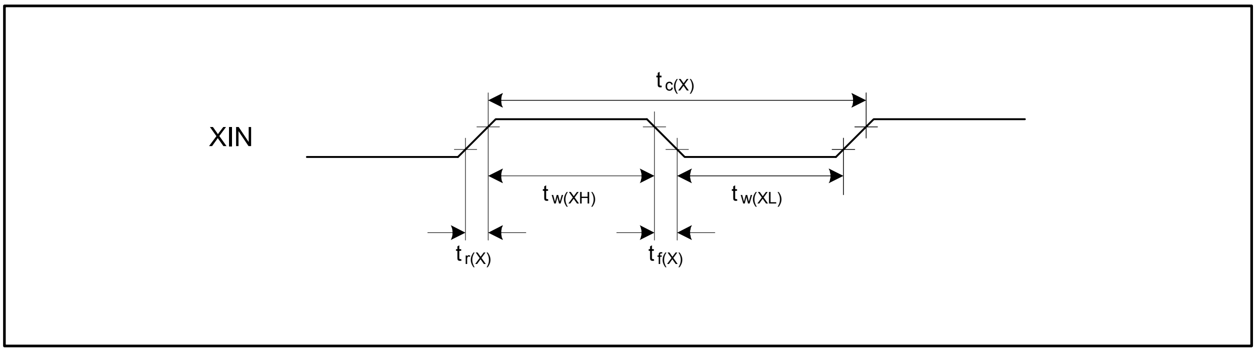


Figure 29.7 External Clock Input Timing

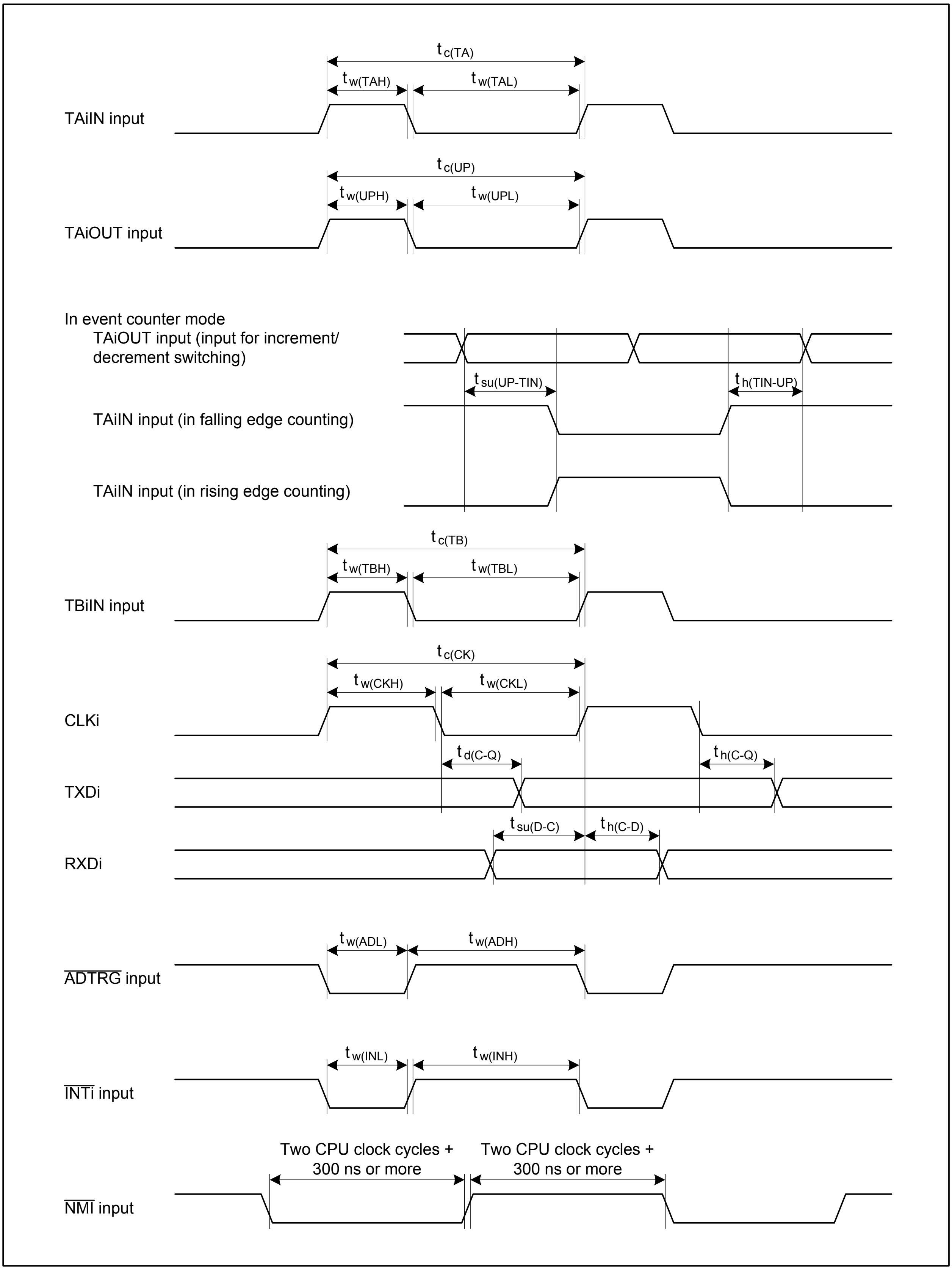


Figure 29.8 Timing of Peripheral Functions

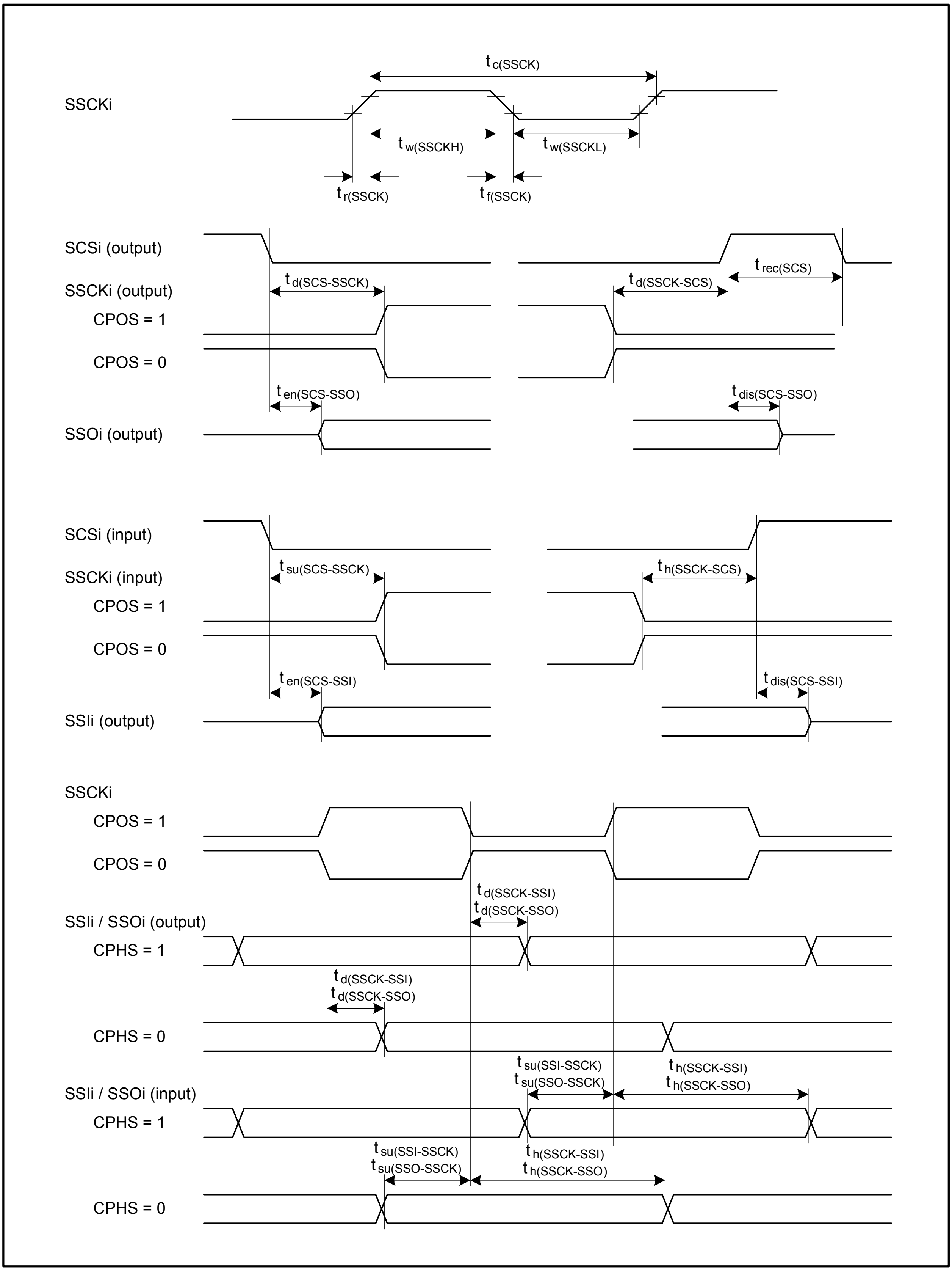


Figure 29.9 Timing of Serial Bus Interface

30. Usage Notes

30.1 Notes on Board Designing

30.1.1 Power Supply Pins

The board should be designed so there is no potential difference between pins with the same name. Note the following points:

- Connect all VSS pins to the same GND. Traces for the pins should be as wide as physically possible so the same voltage can be applied to every VSS pin.

Insert a capacitor between each VCC pin and the VSS pin to prevent operation errors due to noise. The capacitor should be beneficially effective at high and low frequencies and should have a capacitance of approximately 0.1 μF. The traces for the capacitor and the power supply pins should be as short and wide as physically possible.

30.1.2 Supply Voltage

The device is operationally guaranteed under operating conditions specified in electrical characteristics.
Drive the $\overline{\text{RESET}}$ pin low before the supply voltage becomes lower than the recommended value.

30.2 Notes on Register Setting

30.2.1 Registers with Write-only Bits

Read-modify-write instructions cannot be used when setting a register containing write-only bits. Read-modify-write instructions read a value of an address, modify the value, and write the modified value to the same address. Table 30.1 lists read-modify-write instructions, and Table 30.2 lists registers containing write-only bits. To set a new value by modifying the previous one, write the previous value into RAM as well as to the register, change the contents of the RAM and then transfer the new value to the register by the MOV instruction.

Table 30.1 Read-modify-write Instructions

Function	Mnemonic
Transfer	MOV*Dir*
Bit processing	BCLR, BM*Cnd*, BNOT, BSET, BTSTC, and BTSTS
Shifting	ROLC, RORC, ROT, SHA, and SHL
Arithmetic operation	ABS, ADC, ADCF, ADD, ADSF, DEC, DIV, DIVU, DIVX, EXTS, EXTZ, INC, MUL, MULU, NEG, SBB, and SUB
Decimal operation	DADC, DADD, DSBB, and DSUB
Floating-point operation	ADDF, DIVF, MULF, and SUBF
Logical operation	AND, NOT, OR, and XOR

Table 30.2 Registers with Write-only Bits

Module	Register	Symbol	Address
Watchdog timer	Watchdog timer start register	WDTS	04404Eh
Timer A	Timer A0 register (1)	TA0	0347h-0346h
	Timer A1 register (1)	TA1	0349h-0348h
	Timer A2 register (1)	TA2	034Bh-034Ah
	Timer A3 register (1)	TA3	034Dh-034Ch
	Timer A4 register (1)	TA4	034Fh-034Eh
	Increment/decrement select register	UDF	0344h
Three-phase motor control timers	Timer B2 interrupt generating frequency set counter	ICTB2	030Dh
	Timer A1-1 register	TA11	0303h-0302h
	Timer A2-1 register	TA21	0305h-0304h
	Timer A4-1 register	TA41	0307h-0306h
	Timer A1 mirror register	TA1M	0221h-0220h
	Timer A2 mirror register	TA2M	0225h-0224h
	Timer A4 mirror register	TA4M	0229h-0228h
	Timer A1-1 mirror register	TA11M	0223h-0222h
	Timer A2-1 mirror register	TA21M	0227h-0226h
	Timer A4-1 mirror register	TA41M	022Bh-022Ah
	Dead time timer	DTT	030Ch
Serial interface	UART0 bit rate register	U0BRG	0369h
	UART1 bit rate register	U1BRG	02E9h
	UART2 bit rate register	U2BRG	0339h
	UART3 bit rate register	U3BRG	01E1h
	UART4 bit rate register	U4BRG	01E9h
	UART0 transmit buffer register	U0TB	036Bh-036Ah
	UART1 transmit buffer register	U1TB	02EBh-02EAh
	UART2 transmit buffer register	U2TB	033Bh-033Ah
	UART3 transmit buffer register	U3TB	01E3h-01E2h
	UART4 transmit buffer register	U4TB	01EBh-01EAh
CAN module (2)	CAN0 receive FIFO pointer control register	C0RFPCR	047F49h
	CAN0 transmit FIFO pointer control register	C0TFPCR	047F4Bh
	CAN1 receive FIFO pointer control register	C1RFPCR	047D49h
	CAN1 transmit FIFO pointer control register	C1TFPCR	047D4Bh
	CAN2 receive FIFO pointer control register	C2RFPCR	047B49h
	CAN2 transmit FIFO pointer control register	C2TFPCR	047B4Bh
	CAN3 receive FIFO pointer control register	C3RFPCR	046949h
	CAN3 transmit FIFO pointer control register	C3TFPCR	04694Bh
	CAN4 receive FIFO pointer control register	C4RFPCR	047749h
	CAN4 transmit FIFO pointer control register	C4TFPCR	04774Bh
	CAN5 receive FIFO pointer control register	C5RFPCR	047549h
	CAN5 transmit FIFO pointer control register	C5TFPCR	04754Bh

Notes:

1. The register has write-only bits in one-shot timer mode and pulse-width modulation mode.
2. Channels CAN0, CAN1, and CAN4 are not available in the R32C/142 Group.

30.3 Notes on Clock Generator

30.3.1 Sub Clock

30.3.1.1 Oscillator Constant Matching

The constant matching of the sub clock oscillator should be evaluated in both cases when the drive power is high and low.
Contact the oscillator manufacturer for details on the oscillation circuit constant matching.

30.3.2 Power Control

Do not switch the base clock source until the oscillation of the clock to be used has stabilized. However, this does not apply to the on-chip oscillator since it starts running immediately after the CM31 bit in the CM3 register is set to 1.
To switch the base clock source from the PLL clock to a low speed clock, use the MOV.L or OR.L instruction to set the BCS bit in the CCR register to 1.

- Program example in assembly language

```
OR.L      #80h, 0004h
```

- Program example in C language

```
asm("OR.L  #80h, 0004h");
```

30.3.2.1 Stop Mode

- To exit stop mode using a reset, apply a low signal to the $\overline{\text{RESET}}$ pin until the main clock oscillation stabilizes.

30.3.2.2 Suggestions for Power Saving

The followings are suggestions to reduce power consumption when programming or designing systems.

- I/O pins:
  If inputs are floating, both transistors may be conducting. Set unassigned pins to input mode and connect each of them to VSS via a resistor, or set them to output mode and leave them open.
- A/D converter:
  When not performing the A/D conversion, set the VCUT bit in the AD0CON1 register to 0 (VREF disconnected). To perform the A/D conversion, set the VCUT bit to 1 (VREF connected) and wait at least 1 μs before starting conversion.
- D/A converter:
  When not performing the D/A conversion, set the DAiE bit in the DACON register (i = 0, 1) to 0 (output disabled) and the DAi register to 00h.
- Peripheral clock stop:
  When entering wait mode, power consumption can be reduced by setting the CM02 bit in the CM0 register to 1 to stop the peripheral clock source. However, this setting does not stop the fC32.

30.4 Notes on Interrupts

30.4.1 ISP Setting

The interrupt stack pointer (ISP) is initialized to 00000000h after a reset. Set a value to the ISP before an interrupt is accepted, otherwise the program may go out of control. A multiple of 4 should be set to the ISP, which enables faster interrupt sequence due to less memory access.
When using NMI, in particular, since this interrupt cannot be disabled, the PM24 bit in the PM2 register should be set to 1 (NMI enabled) after the ISP is set at the beginning of the program.

30.4.2 NMI

- NMI cannot be disabled once the PM24 bit in the PM2 register is set to 1 (NMI enabled). This bit setting should be done only when using NMI.

- When the PM24 bit in the PM2 register is set to 1 (NMI enabled), the P8_5 bit in the P8 register is enabled just for monitoring the $\overline{\mathrm{NMI}}$ pin state. It is not enabled as a general port.

30.4.3 External Interrupts

- The input signal to the $\overline{\mathrm{INTi}}$ pin (i = 0 to 5) requires the pulse width specified in the electrical characteristics. If the pulse width is narrower than the specification, an external interrupt may not be accepted.

- When the effective level or edge of the $\overline{\mathrm{INTi}}$ pin (i = 0 to 5) is changed by the following bits: bits POL, LVS in the INTiIC register, the IFSR0i bit (i = 0 to 5) in the IFSR0 register, the corresponding IR bit may become 1 (interrupt requested). When setting the above mentioned bits, preset bits ILVL2 to ILVL0 in the INTiIC register to 000b (interrupt disabled). After setting the above mentioned bits, set the corresponding IR bit to 0 (no interrupt requested), then rewrite bits ILVL2 to ILVL0.

30.5 Notes on DMAC

30.5.1 DMAC-associated Register Settings

- Set DMAC-associated registers while bits MDi1 and MDi0 (i = 0 to 3) in the DMDi register are 00b (DMA transfer disabled). Then, set bits MDi1 and MDi0 to 01b (single transfer) or 11b (repeat transfer) at the end of the setup procedure. This procedure also applies when rewriting bits UDAi, USAi, and BWi1 and BWi0 in the DMDi register.

- When rewriting the DMAC-associated registers while DMA transfer is enabled, disable the peripheral function as DMA request source so that no DMA transfer request is generated, then set bits MDi1 and MDi0 in the DMDi register of the corresponding channel to 00b (DMA transfer disabled).

- Once a DMA transfer request is accepted, DMA transfer cannot be disabled even if setting bits MDi1 and MDi0 in the DMDi register to 00b (DMA transfer disabled). Do not change the settings of any DMAC-associated registers other than bits MDi1 and MDi0 until the DMA transfer is completed.

- After setting registers DMiSL and DMiSL2, wait six or more peripheral bus clocks to set bits MDi1 and MDi0 in the DMDi register to 01b (single transfer) or 11b (repeat transfer).

30.5.2 Reading DMAC-associated Registers

- Use the following read order to sequentially read registers DMiSL and DMiSL2:
  DM0SL, DM1SL, DM2SL, and DM3SL
  DM0SL2, DM1SL2, DM2SL2, and DM3SL2

30.6 Notes on Timers

30.6.1 Timer A and Timer B

All timers are stopped after a reset. To restart timers, configure parameters such as operating mode, count source, and counter value, then set the TAiS bit (i = 0 to 4) or TBjS bit (j = 0 to 5) in the TABSR or TBSR register to 1 (count starts).
The following registers and bits should be set while the TAiS bit or TBjS bit is 0 (count stops):

- Registers TAiMR and TBjMR
- UDF register
- Bits TAZIE, TA0TGL, and TA0TGH in the ONSF register
- TRGSR register

30.6.2 Timer A

30.6.2.1 Timer Mode

- While the timer counter is running, the TAi register indicates a counter value at any given time. However, FFFFh is read while reloading is in progress. A set value is read if the TAi register is set while the timer counter is stopped.

30.6.2.2 Event Counter Mode

- While the timer counter is running, the TAi register indicates a counter value at any given time. However, FFFFh is read if the timer counter underflows or 0000h if overflows while reloading is in progress. A set value is read if the TAi register is set while the timer counter is stopped.

30.6.2.3 One-shot Timer Mode

- If the TAiS bit in the TABSR register is set to 0 (count stops) while the timer counter is running, the following operations are performed:
  - The timer counter stops and the setting value of the TAi register is reloaded.
  - A low signal is output at the TAiOUT pin.
  - The IR bit in the TAiIC register becomes 1 (interrupts requested) after one CPU clock cycle.

- The one-shot timer is operated by an internal count source. When the trigger is an input to the TAiIN pin, the signal is output with a maximum one count source clock delay after a trigger input to the TAiIN pin.

- The IR bit becomes 1 by any of the settings below. To use the timer Ai interrupt, set the IR bit to 0 after one of the settings below is done:
  - Select one-shot timer mode after a reset.
  - Switch operating modes from timer mode to one-shot timer mode.
  - Switch operating modes from event counter mode to one-shot timer mode.

- If a retrigger occurs while counting, the timer counter decrements by one, reloads the setting value of the TAi register, and then continues counting. To generate a retrigger while counting, wait one or more count source cycles after the last trigger is generated.

- When an external trigger input is selected to start counting in timer A one-shot mode, do not provide an external retrigger for 300 ns before the timer counter reaches 0000h. Otherwise, it may stop counting.

30.6.2.4 Pulse-width Modulation Mode

- The IR bit becomes 1 by any of the settings below. To use the timer Ai interrupt (i = 0 to 4), set the IR bit to 0 after one of the settings below is done:
  - Select pulse-width modulation mode after a reset.
  - Switch operating modes from timer mode to pulse-width modulation mode.
  - Switch operating modes from event counter mode to pulse-width modulation mode.

- If the TAiS bit in the TABSR register is set to 0 (count stops) while PWM pulse is output, the following operations are performed:
  - The timer counter stops.
  - The output level at the TAiOUT pin changes from high to low. The IR bit becomes 1.
  - When a low signal is output at the TAiOUT pin, it does not change. The IR bit does not change, either.

30.6.3 Timer B

30.6.3.1 Timer Mode and Event Counter Mode

- While the timer counter is running, the TBj register (j = 0 to 5) indicates a counter value at any given time. However, FFFFh is read while reloading is in progress. When a value is set to the TBj register while the timer counter is stopped, if the TBj register is read before the count starts, the set value is read.

30.6.3.2 Pulse Period/Pulse-width Measure Mode

- While the TBjS bit in the TABSR or TBSR register is 1 (start counter), after the MR3 bit becomes 1 (overflow) and at least one count source cycle has elapsed, a write operation to the TBjMR register sets the MR3 bit to 0 (no overflow).

- Use the IR bit in the TBjIC register to detect overflow. The MR3 bit is used only to determine an interrupt request source within the interrupt handler.

- The counter value is undefined when the timer counter starts. Therefore, the timer counter may overflow before a measured pulse is applied on the initial valid edge and cause a timer Bj interrupt request to be generated.

- When the measured pulse is applied on the initial valid edge after the timer counter starts, an undefined value is transferred to the reload register. At this time, a timer Bj interrupt request is not generated.

- The IR bit may become 1 (interrupt requested) by changing bits MR1 and MR0 in the TBjMR register after the timer counter starts. However, if the same value is rewritten to bits MR1 and MR0, the IR bit does not change.

- Pulse width is continuously measured in pulse-width measure mode. Whether the measurement result is high-level width or not is determined by a program.

- When an overflow occurs at the same time a pulse is applied on the valid edge, this pulse is not recognized since an interrupt request is generated only once. Do not let an overflow occur in pulse period measure mode.

- In pulse-width measure mode, determine whether an interrupt source is a pulse applied on the valid edge or an overflow by reading the port level in the timer Bj interrupt handler.

30.7 Notes on Three-phase Motor Control Timers

30.7.1 Shutdown

- When a low signal is applied to the $\overline{\text{NMI}}$ pin with the following bit settings, pins TA1OUT, TA2OUT, and TA4OUT become high-impedance: the PM24 bit in the PM2 register is 1 (NMI enabled), the SDE bit in the IOBC register is 1 (shutdown enabled), the INV02 bit in the INVC0 register is 1 (three-phase motor control timers used), and the INV03 bit is 1 (three-phase motor control timer output enabled).

30.7.2 Register Setting

- Do not write to the TAi1 register (i = 1, 2, 4) before and after timer B2 underflows. Before writing to the TAi1 register, read the TB2 register to verify that sufficient time remains until timer B2 underflows. Then, immediately write to the TAi1 register so no interrupt handling is performed during this write procedure. If the TB2 register indicates little time remains until the underflow, write to the TAi1 register after timer B2 underflows.

30.8 Notes on Serial Interface

30.8.1 Changing the UiBRG Register (i = 0 to 4)

- Set the UiBRG register after setting bits CLK1 and CLK0 in the UiC0 register. When these bits are changed, the UiBRG register must be set again.
- When a clock is input immediately after the UiBRG register is set to 00h, the counter may become FFh. In this case, it requires extra 256 clocks to reload 00h to the register. Once 00h is reloaded, the counter performs the operation without dividing the count source according to the setting.

30.8.2 Synchronous Serial Interface Mode

30.8.2.1 Selecting an External Clock

- If an external clock is selected, the following conditions must be met while the external clock is held high when the CKPOL bit in the UiC0 register (i = 0 to 4) is set to 0 (transmit data output on the falling edge of the transmit/receive clock and receive data input on the rising edge), or while the external clock is held low when the CKPOL bit is set to 1 (transmit data output on the rising edge of the transmit/receive clock and receive data input on the falling edge):
  - The TE bit in the UiC1 register is set to 1 (transmission enabled).
  - The RE bit in the UiC1 register is set to 1 (reception enabled). This bit setting is not required in transmit operation only.
  - The TI bit in the UiC1 register is set to 0 (data held in the UiTB register).

30.8.2.2 Receive Operation

- In synchronous serial interface mode, the transmit/receive clock is controlled by the transmit control circuit. Set UARTi-associated registers (i = 0 to 4) for a transmit operation, even if the MCU is used only for receive operation. Dummy data is output from the TXDi pin while receiving when the TXDi pin is set to output mode.

- When data is received continuously, an overrun error occurs when the RI bit in the UiC1 register is 1 (data held in the UiRB register) and the seventh bit of the next data is received in the UARTi receive shift register. Then, the OER bit in the UiRB register becomes 1 (overrun error occurred). In this case, the UiRB register becomes undefined. If an overrun error occurs, the IR bit in the SiRIC register does not change to 1.

30.8.3 Special Mode 1 (I^{2}C Mode)

- To generate a start condition, stop condition, or restart condition, set the STSPSEL bit in the UiSMR4 register (i = 0 to 2) to 0. Then, wait a half or more clock cycles of the transmit/receive clock to change the condition generate bits (STAREQ, RSTAREQ, or STPREQ bit) from 0 to 1.

30.8.4 Reset Procedure on Communication Error

Operations which result in communication errors such as rewriting function select registers during transmission/reception should not be performed. Follow the procedure below to reset the internal circuit once the communication error occurs in the following cases: when the operation above is performed by a receiver or transmitter or when a bit slip is caused by noise.

A. Synchronous Serial Interface Mode

(1) Set the TE bit in the UiC1 register (i = 0 to 4) to 0 (transmission disabled) and the RE bit to 0 (reception disabled).
(2) Set bits SMD2 to SMD0 in the UiMR register to 000b (serial interface disabled).
(3) Set bits SMD2 to SMD0 in the UiMR register to 001b (synchronous serial interface mode).
(4) Set the TE bit in the UiC1 register to 1 (transmission enabled) and the RE bit to 1 (reception enabled) if necessary.

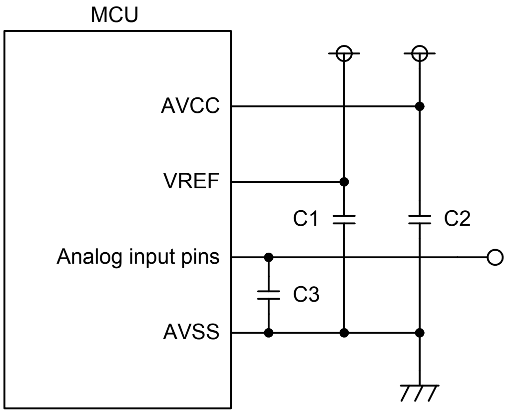
B. UART Mode

(1) Set the TE bit in the UiC1 register to 0 (transmission disabled) and the RE bit to 0 (reception disabled).
(2) Set bits SMD2 to SMD0 in the UiMR register to 000b (serial interface disabled).
(3) Set bits SMD2 to SMD0 in the UiMR register to 100b (UART mode, 7-bit character length), 101b (UART mode, 8-bit character length), or 110b (UART mode, 9-bit character length).
(4) Set the TE bit in the UiC1 register to 1 (transmission enabled) and the RE bit to 1 (reception enabled) if necessary.

30.9 Notes on A/D Converter

30.9.1 Notes on Designing Boards

- Three capacitors should be placed between the AVSS pin and pins such as AVCC, VREF, and analog inputs (AN_0 to AN_7, AN0_0 to AN0_7, and AN2_0 to AN2_7) to avoid erroneous operations caused by noise or latchup, and to reduce conversion errors. Figure 30.1 shows an example of pin configuration for A/D converter.



Notes:
1. C1 ≥ 0.47 μF, C2 ≥ 0.47 μF, and C3 ≥ 100 pF (reference values)
2. The traces for the capacitor and the MCU should be as short and wide as physically possible.

Figure 30.1 Pin Configuration for the A/D Converter

- Do not use AN_4 to AN_7 for analog input if the key input interrupt is to be used. Otherwise, a key input interrupt request occurs when the A/D input voltage becomes VIL or lower.

- When AVCC = VREF = VCC, A/D input voltage for pins AN_0 to AN_7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0, and ANEX1 should be VCC or lower.

30.9.2 Notes on Programming

- The following registers should be written while A/D conversion is stopped. That is, before a trigger occurs: AD0CON0 (except the ADST bit), AD0CON1, AD0CON2, AD0CON3, AD0CON4, and AD0CON5.

- If the VCUT bit in the AD0CON1 register is changed from 0 (VREF connected) to 1 (VREF disconnected), wait for at least 1 µs before starting A/D conversion. When not performing A/D conversion, set the VCUT bit to 0 to reduce power consumption.

- Set the port direction bit for the pin to be used as an analog input pin to 0 (input). Set the ASEL bit of the corresponding port function select register to 1 (port is used as A/D input).

- If the TRG bit in the AD0CON0 register is set to 1 (external trigger or hardware trigger is selected), set the corresponding port direction bit (PD9_7 bit) for the $\overline{\text{ADTRG}}$ pin to 0 (input).

- The ϕAD frequency should be 16 MHz or lower. It should be 1 MHz or higher if the sample and hold function is enabled. If not, it should be 250 kHz or higher.

- If A/D operating mode (bits MD1 and MD0 in the AD0CON0 register or the MD2 bit in the AD0CON1 register) has been changed, reselect analog input pins by setting bits CH2 to CH0 in the AD0CON0 register or bits SCAN1 and SCAN0 in the AD0CON1 register.

- If the AD0i register (i = 0 to 7) is read when the A/D converted result is stored to the register, the stored value may have an error. Read the AD0i register after A/D conversion is completed.
  In one-shot mode or single sweep mode, read the AD0i register after the IR bit in the AD0IC register becomes 1 (interrupt requested).
  In repeat mode, repeat sweep mode 0, or repeat sweep mode 1, an interrupt request can be generated each time A/D conversion is completed if the DUS bit in the AD0CON3 register is set to 1 (DMAC operating mode enabled). Similar to the other modes above, read the AD00 register after the IR bit in the AD0IC register becomes 1 (interrupt requested).

- When an A/D conversion is halted by setting the ADST bit in the AD0CON0 register to 0, the converted result is undefined. In addition, the unconverted AD0i register may also become undefined. Consequently, the AD0i register should not be used just after A/D conversion is halted.

- External triggers cannot be used in DMAC operating mode. When the DMAC is configured to transfer converted results, do not read the AD00 register by a program.

- While in single sweep mode, if A/D conversion is halted by setting the ADST bit in the AD0CON0 register to 0 (A/D conversion is stopped), an interrupt request may be generated even though the sweep is not completed. To halt A/D conversion, disable interrupts before setting the ADST bit to 0.

30.10 Note on Serial Bus Interface

30.10.1 Note on Synchronous Serial Communication Mode and 4-wire Serial Bus Mode

30.10.1.1 Accessing SBI Associated Registers

To read the SBI associated registers (44F06h to 44F1Fh), insert at least three instructions after writing to the registers.

- An example of inserting at least three instructions

  Program example

```
MOV.B   #00h, 44F0Bh ; Set the SS0ER register to 00h.
NOP
NOP
NOP
MOV.B   44F0Bh, R0L
```

30.11 Notes on Flash Memory Rewriting

30.11.1 Note on Power Supply

- Keep the supply voltage constant within the range specified in the electrical characteristics while a rewrite operation on the flash memory is in progress. If the supply voltage goes beyond the guaranteed value, the device cannot be guaranteed.

30.11.2 Note on Hardware Reset

- Do not perform a hardware reset while a rewrite operation on the flash memory is in progress.

30.11.3 Note on Flash Memory Protection

- If an ID code written in an assigned address has an error, any read/write operation on the flash memory in standard serial I/O mode is disabled.

30.11.4 Notes on Programming

- Do not set the FEW bit in the FMCR register to 1 (CPU rewrite mode) in low speed mode or low power mode.
- The program, block erase, lock bit program, and protect bit program are interrupted by an NMI, a watchdog timer interrupt, or an oscillator stop detection interrupt. If any of the software commands above are interrupted, erase the corresponding block and then execute the same command again. If the block erase command is interrupted, the lock bit and protect bit values become undefined. Therefore, disable the lock bit, and then execute the block erase command again.

30.11.5 Notes on Interrupts

- EW0 mode
  - To use interrupts assigned to the relocatable vector table, the vector table should be addressed in RAM space.
  - If an NMI, watchdog timer interrupt, or oscillator stop detection interrupt is generated, the flash memory module automatically enters read array mode. Therefore, these interrupts are enabled even during a rewrite operation. However, the rewrite operation in progress is aborted by the interrupts and registers FMR0 and FRSR0 are reset. When the interrupt handler has ended, set the LBD bit in the FMR1 register to 1 (lock bit protection disabled) to re-execute the rewrite operation.
  - Instructions BRK, INTO, and UND, which refer to data on the flash memory, cannot be used in this mode.

- EW1 mode
  - Interrupts assigned to the relocatable vector table should not be accepted during program or block erase operation.
  - The watchdog timer should not be used in count source protect mode. The watchdog timer interrupt should not be generated.
  - If an NMI, watchdog timer interrupt, or oscillator stop detection interrupt is generated, the flash memory module automatically enters read array mode. Therefore, these interrupts are enabled even during a rewrite operation. However, the rewrite operation in progress is aborted by the interrupts and registers FMR0 and FRSR0 are reset. When the interrupt handler has ended, set the EWM bit in the FMR0 register to 1 (EW1 mode) and the LBD bit in the FMR1 register to 1 (lock bit protection disabled) to re-execute the rewrite operation.

30.11.6 Notes on Rewrite Control Program

- EW0 mode
  - If the supply voltage drops during the rewrite operation of blocks having the rewrite control program, the rewrite control program may not be successfully rewritten, and the rewrite operation itself may not be performed. In this case, perform the rewrite operation by serial programmer or parallel programmer.

- EW1 mode
  - Do not rewrite blocks having the rewrite control program.

30.11.7 Notes on Number of Program/Erase Operations and Software Command Execution Time

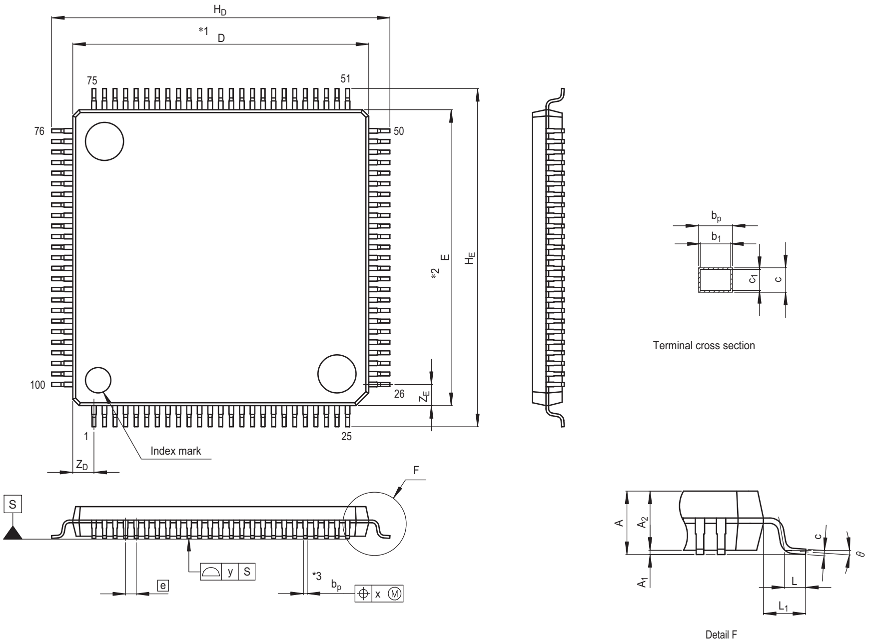
- The time to execute software commands (program, block erase, lock bit program, and protect bit program) increases as the number of program/erase operations increases. If the number of program/erase operations exceeds the endurance value specified in the electrical characteristics, it may take an unpredictable amount of time to execute the software commands. The wait time for executing software commands should be set much longer than the execution time specified in the electrical characteristics.

30.11.8 Other Notes

- The minimum values of program and erase cycles specified in the electrical characteristics are the maximum values that can guarantee the initial performance of the flash memory. The program/ erase operation may still be performed even if the number of program/erase oeprations exceeds the guaranteed values.
- Chips repeatedly programmed and erased for debugging should not be used for commercial products.

Appendix 1. Package Dimensions

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP100-14x14-0.50	PLQP0100KB-A	100P6Q-A / FP-100U / FP-100UV	0.6g



NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	13.9	14.0	14.1
E	13.9	14.0	14.1
A_2	—	1.4	—
H_D	15.8	16.0	16.2
H_E	15.8	16.0	16.2
A	—	—	1.7
A_1	0.05	0.1	0.15
b_p	0.15	0.20	0.25
b_1	—	0.18	—
c	0.09	0.145	0.20
c_1	—	0.125	—
θ	0°	—	8°
$\boxed{e}$	—	0.5	—
x	—	—	0.08
y	—	—	0.08
Z_D	—	1.0	—
Z_E	—	1.0	—
L	0.35	0.5	0.65
L_1	—	1.0	—

INDEX

Revision History	R32C/142 Group and R32C/145 Group User’s Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.10	Sep 12, 2011	—	Initial release

R32C/142 Group and R32C/145 Group User’s Manual: Hardware

Publication Date: Rev.1.10 Sep 12, 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

Colophon 1.1

R32C/142 Group and R32C/145 Group

Renesas Electronics Corporation