User's Manual

16

M16C/6S1 Group

User's Manual: Hardware

RENESAS MCU
M16C Family / M16C/60 Series

Renesas Electronics
www.renesas.com

Rev.1.10 Oct 2013

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation 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.
2. 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.
3. 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.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

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

   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). 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 for which it is not intended. 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 which the product is not intended by Renesas Electronics.
6. 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.
7. 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 systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. 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.
9. 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. You should not use Renesas Electronics products or 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. When exporting the Renesas Electronics 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.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
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.

(2012.4)

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 document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

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 a product with a different part number, confirm that the change will not lead to problems.

   — The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the hardware functions and electrical characteristics of the MCU. It is intended for users designing application systems incorporating the MCU. A basic knowledge of electric circuits, logical circuits, and MCUs is necessary in order to use this manual.
The manual comprises an overview of the product; descriptions of the CPU, system control functions, peripheral functions, and electrical characteristics; and usage notes.

Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the locations of revisions and additions. It does not list all revisions. For details, see the text of the manual.

The following documents apply to the M16C/6S1 Group. Make sure to see the latest versions of these documents. The newest versions of the documents listed may be obtained from the Renesas Electronics Web site.

Document Type	Description	Document Title	Document No.
Datasheet	Hardware overview and electrical characteristics	M16C/6S1 Group Datasheet	
User's manual: Hardware	Hardware specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, timing charts) and operation description Note: For details on using peripheral functions, see the application notes.	M16C/6S1 Group User's Manual: Hardware	This User's manual
User's manual: Software	Description of CPU instruction set	M16C/60, M16C/20, M16C/Tiny Series User's Manual: Software	REJ09B0137
Application note	Information on using peripheral functions and application examples Sample programs Information on writing programs in assembly language and C	Available from Renesas Electronics Web site.	
Renesas technical update	Product specifications, updates on documents, etc.		

2. Notation of Numbers and Symbols

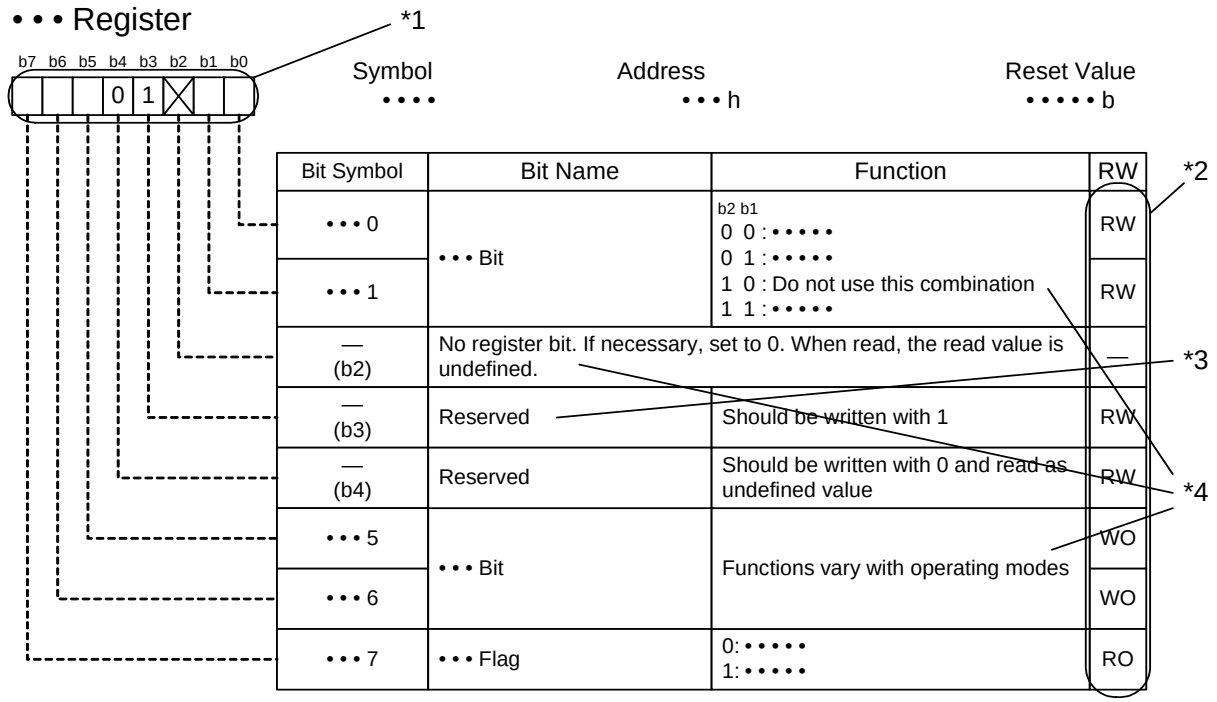
The notation conventions for register names, bit names, numbers, and symbols used in this manual are described below.

(1) Register Names, Bit Names, and Pin Names
Registers, bits, and pins are referred to in the text by symbols. The symbol is accompanied by the word “register,” “bit,” or “pin” to distinguish the three categories.
Examples the SRST bit in the PM0 register
P3_5 pin, VCC pin

(2) Notation of Numbers
The indication “b” is appended to numeric values given in binary format. However, nothing is appended to the values of single bits. The indication “h” is appended to numeric values given in hexadecimal format. Nothing is appended to numeric values given in decimal format.
Examples Binary: 11b
Hexadecimal: EFA0h
Decimal: 1234

3. Register Notation

The symbols and terms used in register diagrams are described below.



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. List of Abbreviations and Acronyms

Abbreviation	Full Form
ADC	Analog-to-digital converter
AES	Advanced Encryption Standard
AFE	Analog Front End
ARIB	Association of Radio Industries and Businesses
BGR	Band Gap Reference
BPF	Band-pass filter
bps	bits per second
CCITT	Comite Consultatif International Telegraphique et Telephonique
CENELEC	Comite Europeen de Normalisation Electrotechnique
CRC	Cyclic Redundancy Check
DAC	Digital-to-analog converter
DCSK	Differential Code Shift Keying
DLL	Data Link Layer
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
FCC	Federal Communications Commission
Hi-Z	High Impedance
I/O	Input/Output
IEBus	Inter Equipment Bus
LPF	Low Pass Filter
LSB	Least Significant Bit
MODEM	Modulator-demodulator
MSB	Most Significant Bit
NC	Non-Connect
PHY	Physical Layer
PLC	Power Line Communications
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
SIM	Subscriber Identity Module
UART	Universal Asynchronous Receiver/Transmitter
VGA	Variable Gain Amplifier

Table of Contents

SFR Quick Reference

Address	Register	Symbol	Page
0000h			
0001h			
0002h			
0003h			
0004h	Processor Mode Register 0	PM0	40, 60, 108
0005h	Processor Mode Register 1	PM1	109
0006h	System Clock Control Register 0	CM0	61
0007h	System Clock Control Register 1	CM1	63
0008h			
0009h			
000Ah	Protect Register	PRCR	35
000Bh			
000Ch	Oscillation Stop Detection Register	CM2	65
000Dh			
000Eh			
000Fh			
0010h	Program 2 Area Control Register	PRG2C	110
0011h			
0012h	Peripheral Clock Select Register	PCLKR	67, 201, 250, 311, 379, 395
0013h	Sub Clock Division Control Register	SCM0	71
0014h			
0015h	Clock Prescaler Reset Flag	CPSRF	201, 250
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	71, 202, 251, 276, 312, 380, 396
0017h			
0018h	Reset Source Determine Register	RSTFR	41
0019h			
001Ah	Voltage Detector Operation Enable Register	VCR2	52
001Bh			
001Ch	PLL Control Register 0	PLC0	68
001Dh	PLL Function Lock Control Register	PLCF	70
001Eh	Processor Mode Register 2	PM2	72, 138
001Fh			
0020h			
0021h			
0022h			
0023h			
0024h			
0025h			
0026h			
0027h			
0028h			
0029h			
002Ah	Voltage Monitor 0 Control Register	VW0C	53
002Bh			
002Ch	Watchdog Timer Detector Register	VW2C	169
002Dh			
002Eh			
002Fh			
0030h			
0031h			
0032h			
0033h			
0034h			
0035h			
0036h			
0037h			
0038h			
0039h			
003Ah			
003Bh			
003Ch			
003Dh			
003Eh			
003Fh			

Address	Register	Symbol	Page
0040h			
0041h			
0042h	$\overline{INT7}$ Interrupt Control Register	INT7IC	140
0043h	$\overline{INT6}$ Interrupt Control Register	INT6IC	140
0044h	$\overline{INT3}$ Interrupt Control Register	INT3IC	140
0045h	Timer B5 Interrupt Control Register	TB5IC	139
0046h	Timer B4 Interrupt Control Register UART1 Bus Collision Detection Interrupt Control Register	TB4IC U1BCNIC	139
0047h	Timer B3 Interrupt Control Register UART0 Bus Collision Detection Interrupt Control Register	TB3IC U0BCNIC	139
0048h	$\overline{SI/O4}$ Interrupt Control Register $\overline{INT5}$ Interrupt Control Register	S4IC INT5IC	140
0049h	$\overline{SI/O3}$ Interrupt Control Register $\overline{INT4}$ Interrupt Control Register	S3IC INT4IC	140
004Ah	UART2 Bus Collision Detection Interrupt Control Register	BCNIC	139
004Bh	DMA0 Interrupt Control Register	DM0IC	139
004Ch	DMA1 Interrupt Control Register	DM1IC	139
004Dh	Key Input Interrupt Control Register	KUPIC	140
004Eh	A/D Conversion Interrupt Control Register	ADIC	139
004Fh	UART2 Transmit Interrupt Control Register	S2TIC	139
0050h	UART2 Receive Interrupt Control Register	S2RIC	139
0051h	UART0 Transmit Interrupt Control Register	S0TIC	139
0052h	UART0 Receive Interrupt Control Register	S0RIC	139
0053h	UART1 Transmit Interrupt Control Register	S1TIC	139
0054h	UART1 Receive Interrupt Control Register	S1RIC	139
0055h	Timer A0 Interrupt Control Register	TA0IC	139
0056h	Timer A1 Interrupt Control Register	TA1IC	139
0057h	Timer A2 Interrupt Control Register	TA2IC	139
0058h	Timer A3 Interrupt Control Register	TA3IC	139
0059h	Timer A4 Interrupt Control Register	TA4IC	139
005Ah	Timer B0 Interrupt Control Register	TB0IC	139
005Bh	Timer B1 Interrupt Control Register	TB1IC	139
005Ch	Timer B2 Interrupt Control Register	TB2IC	139
005Dh	$\overline{INT0}$ Interrupt Control Register	INT0IC	140
005Eh	$\overline{INT1}$ Interrupt Control Register	INT1IC	140
005Fh	$\overline{INT2}$ Interrupt Control Register	INT2IC	140
0060h			
0061h			
0062h			
0063h			
0064h			
0065h			
0066h			
0067h			
0068h			
0069h	DMA2 Interrupt Control Register	DM2IC	139
006Ah	DMA3 Interrupt Control Register	DM3IC	139
006Bh	UART5 Bus Collision Detection Interrupt Control Register	U5BCNIC	139
006Ch	UART5 Transmit Interrupt Control Register	S5TIC	139
006Dh	UART5 Receive Interrupt Control Register	S5RIC	139
006Eh	UART6 Bus Collision Detection Interrupt Control Register Real-Time Clock Periodic Interrupt Control Register	U6BCNIC RTCTIC	139
006Fh	UART6 Transmit Interrupt Control Register Real-Time Clock Alarm Interrupt Control Register	S6TIC RTCCIC	139
0070h	UART6 Receive Interrupt Control Register	S6RIC	139
0071h	UART7 Bus Collision Detection Interrupt Control Register	U7BCNIC	139
0072h	UART7 Transmit Interrupt Control Register	S7TIC	139
0073h	UART7 Receive Interrupt Control Register	S7RIC	139
0074h			
0075h			
0076h			
0077h			
0078h			
0079h			

Note: 1. Blank columns are all reserved space. No access is allowed.

Address	Register	Symbol	Page
007Ah			
007Bh	I2C-bus Interface Interrupt Control Register	IICIC	139
007Ch	SCL/SDA Interrupt Control Register	SCLDAIC	139
007Dh			
007Eh			
007Fh			
0080h to 017Fh			
0180h	DMA0 Source Pointer	SAR0	181
0181h			
0182h			
0183h			
0184h	DMA0 Destination Pointer	DAR0	181
0185h			
0186h			
0187h			
0188h	DMA0 Transfer Counter	TCR0	182
0189h			
018Ah			
018Bh			
018Ch	DMA0 Control Register	DM0CON	183
018Dh			
018Eh			
018Fh			
0190h	DMA1 Source Pointer	SAR1	181
0191h			
0192h			
0193h			
0194h	DMA1 Destination Pointer	DAR1	181
0195h			
0196h			
0197h			
0198h	DMA1 Transfer Counter	TCR1	182
0199h			
019Ah			
019Bh			
019Ch	DMA1 Control Register	DM1CON	183
019Dh			
019Eh			
019Fh			
01A0h	DMA2 Source Pointer	SAR2	181
01A1h			
01A2h			
01A3h			
01A4h	DMA2 Destination Pointer	DAR2	181
01A5h			
01A6h			
01A7h			
01A8h	DMA2 Transfer Counter	TCR2	182
01A9h			
01AAh			
01ABh			
01ACh	DMA2 Control Register	DM2CON	183
01ADh			
01AEh			
01AFh			
01B0h	DMA3 Source Pointer	SAR3	181
01B1h			
01B2h			
01B3h			
01B4h	DMA3 Destination Pointer	DAR3	181
01B5h			
01B6h			
01B7h			
01B8h	DMA3 Transfer Counter	TCR3	182
01B9h			

Address	Register	Symbol	Page
01BAh			
01BBh			
01BCh	DMA3 Control Register	DM3CON	183
01BDh			
01BEh			
01BFh			
01C0h	Timer B0-1 Register	TB01	253
01C1h			
01C2h	Timer B1-1 Register	TB11	253
01C3h			
01C4h	Timer B2-1 Register	TB21	253
01C5h			
01C6h	Pulse Period/Pulse Width Measurement Mode Function Select Register 1	PPWFS1	254
01C7h			
01C8h	Timer B Count Source Select Register 0	TBCS0	255
01C9h	Timer B Count Source Select Register 1	TBCS1	255
01CAh			
01CBh			
01CCh			
01CDh			
01CEh			
01CFh			
01D0h	Timer A Count Source Select Register 0	TACS0	203
01D1h	Timer A Count Source Select Register 1	TACS1	203
01D2h	Timer A Count Source Select Register 2	TACS2	203
01D3h			
01D4h	16-Bit Pulse Width Modulation Mode Function Select Register	PWMFS	204
01D5h	Timer A Waveform Output Function Select Register	TAPOFS	205
01D6h			
01D7h			
01D8h	Timer A Output Waveform Change Enable Register	TAOW	206
01D9h			
01DAh			
01DBh			
01DCh			
01DDh			
01DEh			
01DFh			
01E0h	Timer B3-1 Register	TB31	253
01E1h			
01E2h	Timer B4-1 Register	TB41	253
01E3h			
01E4h	Timer B5-1 Register	TB51	253
01E5h			
01E6h	Pulse Period/Pulse Width Measurement Mode Function Select Register 2	PPWFS2	254
01E7h			
01E8h	Timer B Count Source Select Register 2	TBCS2	255
01E9h	Timer B Count Source Select Register 3	TBCS3	255
01EAh			
01EBh			
01ECh			
01EDh			
01EEh			
01EFh			
01F0h			
01F1h			
01F2h			
01F3h			
01F4h			
01F5h			
01F6h			
01F7h			
01F8h			
01F9h			

Note: 1. Blank columns are all reserved space. No access is allowed.

Address	Register	Symbol	Page
01FAh			
01FBh			
01FCh			
01FDh			
01FEh			
01FFh			
0200h			
0201h			
0202h			
0203h			
0204h			
0205h	Interrupt Source Select Register 3	IFSR3A	141
0206h	Interrupt Source Select Register 2	IFSR2A	142
0207h	Interrupt Source Select Register	IFSR	143
0208h			
0209h			
020Ah			
020Bh			
020Ch			
020Dh			
020Eh	Address Match Interrupt Enable Register	AIER	144
020Fh	Address Match Interrupt Enable Register 2	AIER2	144
0210h	Address Match Interrupt Register 0	RMAD0	145
0211h			
0212h			
0213h			
0214h	Address Match Interrupt Register 1	RMAD1	145
0215h			
0216h			
0217h			
0218h	Address Match Interrupt Register 2	RMAD2	145
0219h			
021Ah			
021Bh			
021Ch	Address Match Interrupt Register 3	RMAD3	145
021Dh			
021Eh			
021Fh			
0220h	Flash Memory Control Register 0	FMR0	88, 484
0221h	Flash Memory Control Register 1	FMR1	487
0222h	Flash Memory Control Register 2	FMR2	89, 488
0223h	Flash Memory Control Register 3	FMR3	489
0224h			
0225h			
0226h			
0227h			
0228h			
0229h			
022Ah			
022Bh			
022Ch			
022Dh			
022Eh			
022Fh			
0230h	Flash Memory Control Register 6	FMR6	490
0231h	Flash Memory Control Register 7	FMR7	491
0232h			
0233h			
0234h			
0235h			
0236h			
0237h			
0238h			
0239h			

Address	Register	Symbol	Page
023Ah			
023Bh			
023Ch			
023Dh			
023Eh			
023Fh			
0240h			
0241h			
0242h			
0243h			
0244h	UART0 Special Mode Register 4	U0SMR4	321
0245h	UART0 Special Mode Register 3	U0SMR3	323
0246h	UART0 Special Mode Register 2	U0SMR2	324
0247h	UART0 Special Mode Register	U0SMR	325
0248h	UART0 Transmit/Receive Mode Register	U0MR	313
0249h	UART0 Bit Rate Register	U0BRG	314
024Ah	UART0 Transmit Buffer Register	U0TB	314
024Bh			
024Ch	UART0 Transmit/Receive Control Register 0	U0C0	315
024Dh	UART0 Transmit/Receive Control Register 1	U0C1	317
024Eh	UART0 Receive Buffer Register	U0RB	318
024Fh			
0250h	UART Transmit/Receive Control Register 2	UCON	320
0251h			
0252h			
0253h			
0254h	UART1 Special Mode Register 4	U1SMR4	321
0255h	UART1 Special Mode Register 3	U1SMR3	323
0256h	UART1 Special Mode Register 2	U1SMR2	324
0257h	UART1 Special Mode Register	U1SMR	325
0258h	UART1 Transmit/Receive Mode Register	U1MR	313
0259h	UART1 Bit Rate Register	U1BRG	314
025Ah	UART1 Transmit Buffer Register	U1TB	314
025Bh			
025Ch	UART1 Transmit/Receive Control Register 0	U1C0	315
025Dh	UART1 Transmit/Receive Control Register 1	U1C1	317
025Eh	UART1 Receive Buffer Register	U1RB	318
025Fh			
0260h			
0261h			
0262h			
0263h			
0264h	UART2 Special Mode Register 4	U2SMR4	321
0265h	UART2 Special Mode Register 3	U2SMR3	323
0266h	UART2 Special Mode Register 2	U2SMR2	324
0267h	UART2 Special Mode Register	U2SMR	325
0268h	UART2 Transmit/Receive Mode Register	U2MR	313
0269h	UART2 Bit Rate Register	U2BRG	314
026Ah	UART2 Transmit Buffer Register	U2TB	314
026Bh			
026Ch	UART2 Transmit/Receive Control Register 0	U2C0	315
026Dh	UART2 Transmit/Receive Control Register 1	U2C1	317
026Eh	UART2 Receive Buffer Register	U2RB	318
026Fh			
0270h	SI/O3 Transmit/Receive Register	S3TRR	380
0271h			
0272h	SI/O3 Control Register	S3C	381
0273h	SI/O3 Bit Rate Register	S3BRG	382
0274h	SI/O4 Transmit/Receive Register	S4TRR	380
0275h			
0276h	SI/O4 Control Register	S4C	381
0277h	SI/O4 Bit Rate Register	S4BRG	382
0278h	SI/O3, 4 Control Register 2	S34C2	382
0279h			

Note: 1. Blank columns are all reserved space. No access is allowed.

Address	Register	Symbol	Page
027Ah			
027Bh			
027Ch			
027Dh			
027Eh			
027Fh			
0280h			
0281h			
0282h			
0283h			
0284h	UART5 Special Mode Register 4	U5SMR4	321
0285h	UART5 Special Mode Register 3	U5SMR3	323
0286h	UART5 Special Mode Register 2	U5SMR2	324
0287h	UART5 Special Mode Register	U5SMR	325
0288h	UART5 Transmit/Receive Mode Register	U5MR	313
0289h	UART5 Bit Rate Register	U5BRG	314
028Ah	UART5 Transmit Buffer Register	U5TB	314
028Bh			
028Ch	UART5 Transmit/Receive Control Register 0	U5C0	315
028Dh	UART5 Transmit/Receive Control Register 1	U5C1	317
028Eh	UART5 Receive Buffer Register	U5RB	318
028Fh			
0290h			
0291h			
0292h			
0293h			
0294h	UART6 Special Mode Register 4	U6SMR4	321
0295h	UART6 Special Mode Register 3	U6SMR3	323
0296h	UART6 Special Mode Register 2	U6SMR2	324
0297h	UART6 Special Mode Register	U6SMR	325
0298h	UART6 Transmit/Receive Mode Register	U6MR	313
0299h	UART6 Bit Rate Register	U6BRG	314
029Ah	UART6 Transmit Buffer Register	U6TB	314
029Bh			
029Ch	UART6 Transmit/Receive Control Register 0	U6C0	315
029Dh	UART6 Transmit/Receive Control Register 1	U6C1	317
029Eh	UART6 Receive Buffer Register	U6RB	318
029Fh			
02A0h			
02A1h			
02A2h			
02A3h			
02A4h	UART7 Special Mode Register 4	U7SMR4	321
02A5h	UART7 Special Mode Register 3	U7SMR3	323
02A6h	UART7 Special Mode Register 2	U7SMR2	324
02A7h	UART7 Special Mode Register	U7SMR	325
02A8h	UART7 Transmit/Receive Mode Register	U7MR	313
02A9h	UART7 Bit Rate Register	U7BRG	314
02AAh	UART7 Transmit Buffer Register	U7TB	314
02ABh			
02ACh	UART7 Transmit/Receive Control Register 0	U7C0	315
02ADh	UART7 Transmit/Receive Control Register 1	U7C1	317
02AEh	UART7 Receive Buffer Register	U7RB	318
02AFh			
02B0h	I2C0 Data Shift Register	S00	397
02B1h			
02B2h	I2C0 Address Register 0	S0D0	398
02B3h	I2C0 Control Register 0	S1D0	399
02B4h	I2C0 Clock Control Register	S20	402
02B5h	I2C0 Start/Stop Condition Control Register	S2D0	404
02B6h	I2C0 Control Register 1	S3D0	405
02B7h	I2C0 Control Register 2	S4D0	409
02B8h	I2C0 Status Register 0	S10	411
02B9h	I2C0 Status Register 1	S11	416

Address	Register	Symbol	Page
02BAh	I2C0 Address Register 1	S0D1	398
02BBh	I2C0 Address Register 2	S0D2	398
02BCh			
02BDh			
02BEh			
02BFh			
02C0h to 02FFh			
0300h	Timer B3/B4/B5 Count Start Flag	TBSR	256
0301h			
0302h	Timer A1-1 Register	TA11	208
0303h			
0304h	Timer A2-1 Register	TA21	208
0305h			
0306h	Timer A4-1 Register	TA41	208
0307h			
0308h			
0309h			
030Ah			
030Bh			
030Ch			
030Dh			
030Eh			
030Fh			
0310h	Timer B3 Register	TB3	252
0311h			
0312h	Timer B4 Register	TB4	252
0313h			
0314h	Timer B5 Register	TB5	252
0315h			
0316h			
0317h			
0318h			
0319h			
031Ah			
031Bh	Timer B3 Mode Register	TB3MR	257
031Ch	Timer B4 Mode Register	TB4MR	257
031Dh	Timer B5 Mode Register	TB5MR	257
031Eh			
031Fh			
0320h	Count Start Flag	TABSR	209, 256
0321h			
0322h	One-Shot Start Flag	ONSF	210
0323h	Trigger Select Register	TRGSR	211
0324h	Up/Down Flag	UDF	212
0325h			
0326h	Timer A0 Register	TA0	207
0327h			
0328h	Timer A1 Register	TA1	207
0329h			
032Ah	Timer A2 Register	TA2	207
032Bh			
032Ch	Timer A3 Register	TA3	207
032Dh			
032Eh	Timer A4 Register	TA4	207
032Fh			
0330h	Timer B0 Register	TB0	252
0331h			
0332h	Timer B1 Register	TB1	252
0333h			
0334h	Timer B2 Register	TB2	252
0335h			
0336h	Timer A0 Mode Register	TA0MR	213
0337h	Timer A1 Mode Register	TA1MR	213
0338h	Timer A2 Mode Register	TA2MR	213
0339h	Timer A3 Mode Register	TA3MR	213

Note: 1. Blank columns are all reserved space. No access is allowed.

Address	Register	Symbol	Page
033Ah	Timer A4 Mode Register	TA4MR	213
033Bh	Timer B0 Mode Register	TB0MR	257
033Ch	Timer B1 Mode Register	TB1MR	257
033Dh	Timer B2 Mode Register	TB2MR	257
033Eh			
033Fh			
0340h	Second Data Register	TRHSEC	277
0341h	Minute Data Register	TRHMIN	278
0342h	Hour Data Register	TRHHR	279
0343h	Day-of-the-Week Data Register	TRHWK	280
0344h	Date Data Register	TRHDY	281
0345h	Month Data Register	TRHMON	282
0346h	Year Data Register	TRHYR	283
0347h	Timer RH Control Register	TRHCR	284
0348h	Timer RH Count Source Select Register	TRHCSR	286
0349h	Clock Error Correction Register	TRHADJ	287
034Ah	Timer RH Interrupt Flag Register	TRHIFR	288
034Bh	Timer RH Interrupt Enable Register	TRHIER	289
034Ch	Alarm Minute Register	TRHAMN	291
034Dh	Alarm Hour Register	TRHAHR	292
034Eh	Alarm Day-of-the-Week Register	TRHAWK	293
034Fh	Timer RH Protect Register	TRHPRC	294
0350h			
0351h			
0352h			
0353h			
0354h			
0355h			
0356h			
0357h			
0358h			
0359h			
035Ah			
035Bh			
035Ch			
035Dh			
035Eh			
035Fh			
0360h	Pull-Up Control Register 0	PUR0	124
0361h	Pull-Up Control Register 1	PUR1	125
0362h	Pull-Up Control Register 2	PUR2	126
0363h			
0364h			
0365h			
0366h	Port Control Register	PCR	127, 146, 447
0367h			
0368h			
0369h	$\overline{\text{NMI}}$ Digital Filter Register	NMIDF	130, 147
036Ah			
036Bh			
036Ch			
036Dh			
036Eh			
036Fh			
0370h			
0371h			
0372h			
0373h			
0374h			
0375h			
0376h			
0377h			
0378h			
0379h			

Address	Register	Symbol	Page
037Ah			
037Bh			
037Ch	Count Source Protection Mode Register	CSPR	170
037Dh	Watchdog Timer Refresh Register	WDTR	171
037Eh	Watchdog Timer Start Register	WDTS	171
037Fh	Watchdog Timer Control Register	WDC	172
0380h			
0381h			
0382h			
0383h			
0384h			
0385h			
0386h			
0387h			
0388h			
0389h			
038Ah			
038Bh			
038Ch			
038Dh			
038Eh			
038Fh			
0390h	DMA2 Source Select Register	DM2SL	184
0391h			
0392h	DMA3 Source Select Register	DM3SL	184
0393h			
0394h			
0395h			
0396h			
0397h			
0398h	DMA0 Source Select Register	DM0SL	184
0399h			
039Ah	DMA1 Source Select Register	DM1SL	184
039Bh			
039Ch			
039Dh			
039Eh			
039Fh			
03A0h			
03A1h			
03A2h	Open-Circuit Detection Assist Function Register	AINRST	448
03A3h			
03A4h			
03A5h			
03A6h			
03A7h			
03A8h			
03A9h			
03AAh			
03ABh			
03ACh			
03ADh			
03AEh			
03AFh			
03B0h			
03B1h			
03B2h			
03B3h			
03B4h	SFR Snoop Address Register	CRCSAR	477
03B5h			
03B6h	CRC Mode Register	CRCMR	478
03B7h			
03B8h			
03B9h			

Note: 1. Blank columns are all reserved space. No access is allowed.

Address	Register	Symbol	Page
03BAh			
03BBh			
03BCh	CRC Data Register	CRCD	478
03BDh			
03BEh	CRC Input Register	CRCIN	478
03BFh			
03C0h	A/D Register 0	AD0	449
03C1h			
03C2h	A/D Register 1	AD1	449
03C3h			
03C4h	A/D Register 2	AD2	449
03C5h			
03C6h	A/D Register 3	AD3	449
03C7h			
03C8h	A/D Register 4	AD4	449
03C9h			
03CAh	A/D Register 5	AD5	449
03CBh			
03CCh	A/D Register 6	AD6	449
03CDh			
03CEh	A/D Register 7	AD7	449
03CFh			
03D0h			
03D1h			
03D2h			
03D3h			
03D4h	A/D Control Register 2	ADCON2	450
03D5h			
03D6h	A/D Control Register 0	ADCON0	451
03D7h	A/D Control Register 1	ADCON1	453
03D8h			
03D9h			
03DAh			
03DBh			
03DCh			
03DDh			
03DEh			
03DFh			
03E0h	Port P0 Register	P0	128
03E1h	Port P1 Register	P1	128
03E2h	Port P0 Direction Register	PD0	129
03E3h	Port P1 Direction Register	PD1	129
03E4h	Port P2 Register	P2	128
03E5h	Port P3 Register	P3	128
03E6h	Port P2 Direction Register	PD2	129
03E7h	Port P3 Direction Register	PD3	129
03E8h	Port P4 Register	P4	128
03E9h	Port P5 Register	P5	128
03EAh	Port P4 Direction Register	PD4	129
03EBh	Port P5 Direction Register	PD5	129
03ECh	Port P6 Register	P6	128
03EDh	Port P7 Register	P7	128
03EEh	Port P6 Direction Register	PD6	129
03EFh	Port P7 Direction Register	PD7	129

Address	Register	Symbol	Page
03F0h	Port P8 Register	P8	128
03F1h	Port P9 Register	P9	128
03F2h	Port P8 Direction Register	PD8	129
03F3h	Port P9 Direction Register	PD9	129
03F4h	Port P10 Register	P10	128
03F5h			
03F6h	Port P10 Direction Register	PD10	129
03F7h			
03F8h			
03F9h			
03FAh			
03FBh			
03FCh			
03FDh			
03FEh			
03FFh			
D000h to D07Fh			
D080h			
D081h			
D082h			
D083h			
D084h			
D085h			
D086h			
D087h			
D088h			
D089h			
D08Ah			
D08Bh			
D08Ch			
D08Dh			
D08Eh			
D08Fh			
D090h			
D091h			
D092h			
D093h			
D094h			
D095h			
D096h			
D097h			
D098h			
D099h			
D09Ah			
D09Bh			
D09Ch			
D09Dh			
D09Eh			
D09Fh			
D0A0h to D7FFh			

FFFFFh	Optional Function Select Address 1	OFS1	42, 54, 173, 493

Note: 1. OFS1 address is not an SFR.

1. Overview

1.1 Features

The M16C/6S1 Group incorporates the M16C/60 Series CPU core that enables a high level of code efficiency and high-speed operation, PLC (Power Line Communication) modem core developed by Yitran Communications Ltd, and AFE (Analog Front End).
The implementation of DCSK (Differential Code Shift Keying) and DCSK turbo spectrum modulation technique in the PLC modem core enables extremely robust communication over the existing electrical wiring, with data rates up to 500 kbps.
In addition, the M16C/6S1 Group complies with worldwide regulations (FCC part 15, ARIB, and CENELEC) and is suitable for a variety of narrowband applications like smart metering and home networking.

1.1.1 Functions

- MCU
  - CPU
    Operating frequency: Up to 30.72 MHz
  - Memory
    Flash memory: 128 KB to 256 KB
    Data flash: 8 KB
    RAM: 20 KB to 31 KB
  - Peripheral functions
    Timer A: 16-bit timer × 5, timer B: 16-bit timer × 6, watchdog timer: Real-time clock
    UART: 6 channels, SIO: 2 channels, I^2C: 1 channel
    A/D: 10 bits, 18 channels
    I/O: 56 pins
- PLC
  - Operating frequency: 46.08 MHz
  - Data rate & Frequency band
    M16C/6S compatibility mode (DCSK)
    FCC&ARIB: 1.25 K, 5 K, 7.5 kbps
    CENELEC A Band: 0.625 K, 2.5 kbps
    CENELEC B Band: 0.625 K, 2.5 kbps
    High-speed mode (DCSK turbo)
    FCC & ARIB: Up to 500 kbps@PHY/up to 300 kbps@MAC payload rate
    CENELEC A Band (outdoor): Up to 150 kbps@PHY/up to 90 kbps@MAC payload rate
    CENELEC B Band (indoor): Up to 50 kbps@PHY/up to 30 kbps@MAC payload rate
- Package: 100 pins, HTQFP
- Supply voltage: 3.3 V

1.2 Specifications

Tables 1.1 to 1.4 outline the Specifications.

Table 1.1 Specifications (1/2)

Item	Function	Description
CPU	Central processing unit	M16C/60 Series core (multiplier: 16-bit × 16-bit → 32-bit, multiply and accumulate instruction: 16-bit × 16-bit + 32-bit → 32-bit) • Number of basic instructions: 91 • Minimum instruction execution time: 32.6 ns (f(BCLK) = 30.72 MHz, VCC1 = VCC2 = 3.0 to 3.6 V) 41.7 ns (f(BCLK) = 24 MHz, VCC1 = VCC2 = 2.7 to 3.0 V)
Memory	ROM, RAM, data flash	See Table 1.5 “Product List”.
Clock	Clock generator	• 4 circuits: Main clock, sub clock, low-speed on-chip oscillator (125 kHz), PLL frequency synthesizer • Oscillation stop detection: Main clock oscillation stop/reoscillation detection function • Frequency divider circuit: Divide ratio selectable from 1, 2, 4, 8, and 16 • Power saving features: Wait mode, stop mode • Real-time clock
I/O Ports	Programmable I/O ports	• CMOS I/O ports: 53 (selectable pull-up resistors) • N-channel open drain ports: 3
Interrupts		• Interrupt vectors: 70 • External interrupt inputs: 14 ($\overline{\text{NMI}}$, $\overline{\text{INT}}$ × 5, key input × 8) • Interrupt priority levels: 7
Watchdog Timer		15-bit timer × 1 (with prescaler) Automatic reset start function selectable
DMA	DMAC	• 4 channels, cycle steal mode • Trigger sources: 43 • Transfer modes: 2 (single transfer, repeat transfer)
Timers	Timer A	16-bit timer × 5 Timer mode, event counter mode, one-shot timer mode, pulse width modulation (PWM) mode, Event counter two-phase pulse signal processing (two-phase encoder input) × 3, programmable output mode × 3
	Timer B	16-bit timer × 6 Timer mode, event counter mode, pulse period measurement mode, pulse width measurement mode
	Real-time clock	Count: seconds, minutes, hours, days of the week, months, years
Serial Interface	UART0 to UART2, UART5 to UART7	Clock synchronous/asynchronous × 5 channels, PLC connection × 1 channel, I^2C-bus, IEBus, special mode 2, SIM (UART2)
	SI/O3, SI/O4	Clock synchronization only × 2 channels
Multi-master I^2C-bus Interface		1 channel
A/D Converter		10-bit resolution × 18 channels, including sample and hold function Conversion time: 2.8 μs
CRC Calculator		CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$), CRC-16 ($X^{16} + X^{15} + X^2 + 1$) compliant
Encryption	AES	AES Encryption (Key length: 128 bits)
Flash Memory		• Erase/write power supply voltage: 2.7 V to 3.6 V • Erase/write cycles: 1,000 times (program ROM 1, program ROM 2), 10,000 times (data flash) • Program security: ROM code protect, ID code check

Table 1.2 Specifications (2/2)

Item	Description
Debug Functions	On-chip debug (2-wire), on-board flash rewrite, address match interrupt × 4
Operation Frequency/Supply Voltage	30.72 MHz/VCC1 = VCC2 = 3.0 V to 3.6 V 24 MHz/VCC1 = VCC2 = 2.7 V to 3.0 V Operation supply voltage range of PLC block/VCCA = 3.0 V to 3.6 V Set VCC1 = VCC2 = VCCA
Current Consumption	Described in Electrical Characteristics
Operating Temperature	-20°C to 85°C, -40°C to 85°C (1)
Package	100-pin HTQFP: PTQP100KE-E (Previous package code: 100PFW-E)

Note:
1. See Table 1.5 “Product List” for the operating temperature.

Table 1.3 PLC Block Specifications in DCSK Mode

Item		Specification
Modulation technique		DCSK
Error Detection/Correction		Short-block error correction, CRC-16
Compliant worldwide regulations		FCC, ARIB, EN50065-1-CENELEC
Data transfer rate and Frequency band	FCC and ARIB	120 kHz to 400 kHz 7.5 kbps, Standard mode (SM) 5.0 kbps, Robust mode (RM) 1.25 kbps, Extremely Robust mode (ERM)
	CENELEC	A Band: 20 kHz to 80 kHz B Band: 95 kHz to 125 kHz 2.5 kbps, Robust mode (RM) 0.625 kbps, Extremely Robust mode (ERM)
Internal AFE		D/A converter, LPF, line driver amplifier A/D converter, VGA, BGR, VDC

Table 1.4 PLC Block Specifications in DCSK Turbo Mode

Item		Specification
Modulation technique		DCSK turbo
Error Detection/Correction		Short-block error correction, CRC-16
Compliant worldwide regulations		FCC, ARIB, EN50065-1-CENELEC
Data transfer rate and Frequency band	FCC and ARIB	120 kHz to 400 kHz Up to 500 kbps@PHY/up to 300 kbps@MAC payload rate (communication packet length: 1760-byte data)
	CENELEC	A Band: 20 kHz to 80 kHz Up to 150 kbps@PHY, up to 90 kbps@MAC payload rate (communication packet length: 1760-byte data)
		B Band: 95 kHz to 125 kHz Up to 50 kbps@PHY, up to 30 kbps@MAC payload rate (communication packet length: 1760-byte data)
Internal AFE		D/A converter, LPF, line driver amplifier A/D converter, VGA, BGR, VDC

1.3 Product List

Table 1.5 lists Product List. Figure 1.1 shows the Part No., with Memory Size and Package.

Table 1.5 Product List **Current of Oct 2013**

Part No.	ROM Capacity			RAM Capacity	Package Code	Remarks
	Program ROM 1	Program ROM 2	Data flash			
R5F36S16NFB	128 KB	16 KB	4 KB × 2 blocks	20 KB	PTQP0100KE-E	Operating temperature -20°C to 85°C
R5F36S16DFB						Operating temperature -40°C to 85°C
R5F36S1ENFB	256 KB			31 KB		Operating temperature -20°C to 85°C
R5F36S1EDFB						Operating temperature -40°C to 85°C

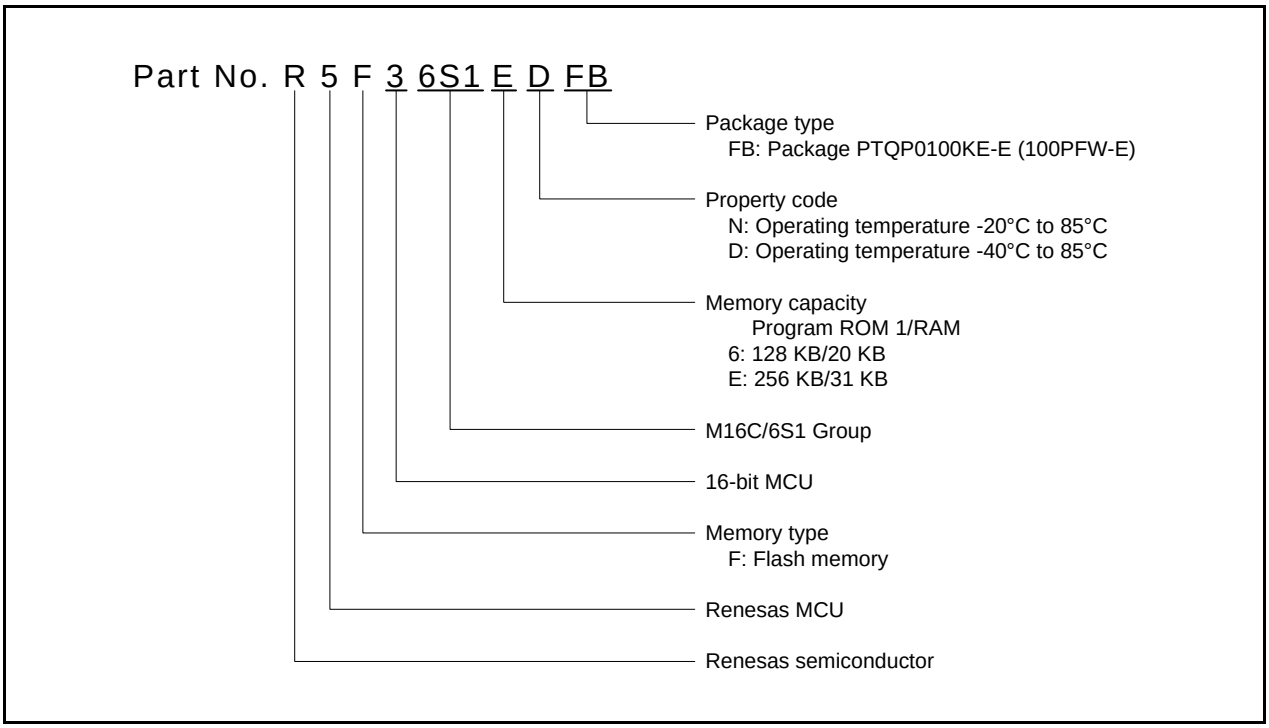


Figure 1.1 Part No., with Memory Size and Package

1.4 Block Diagram

Figure 1.2 shows Block Diagram.

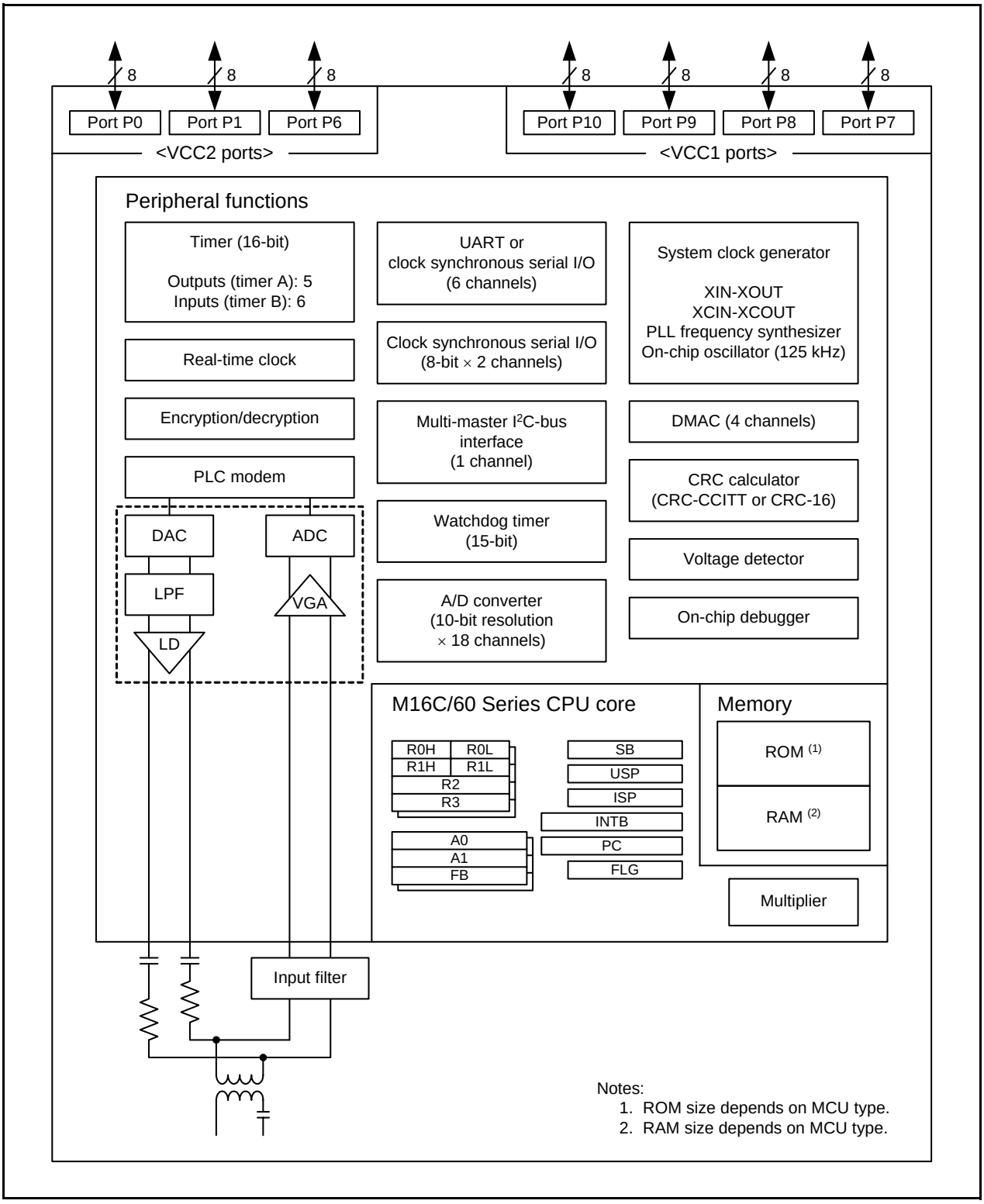
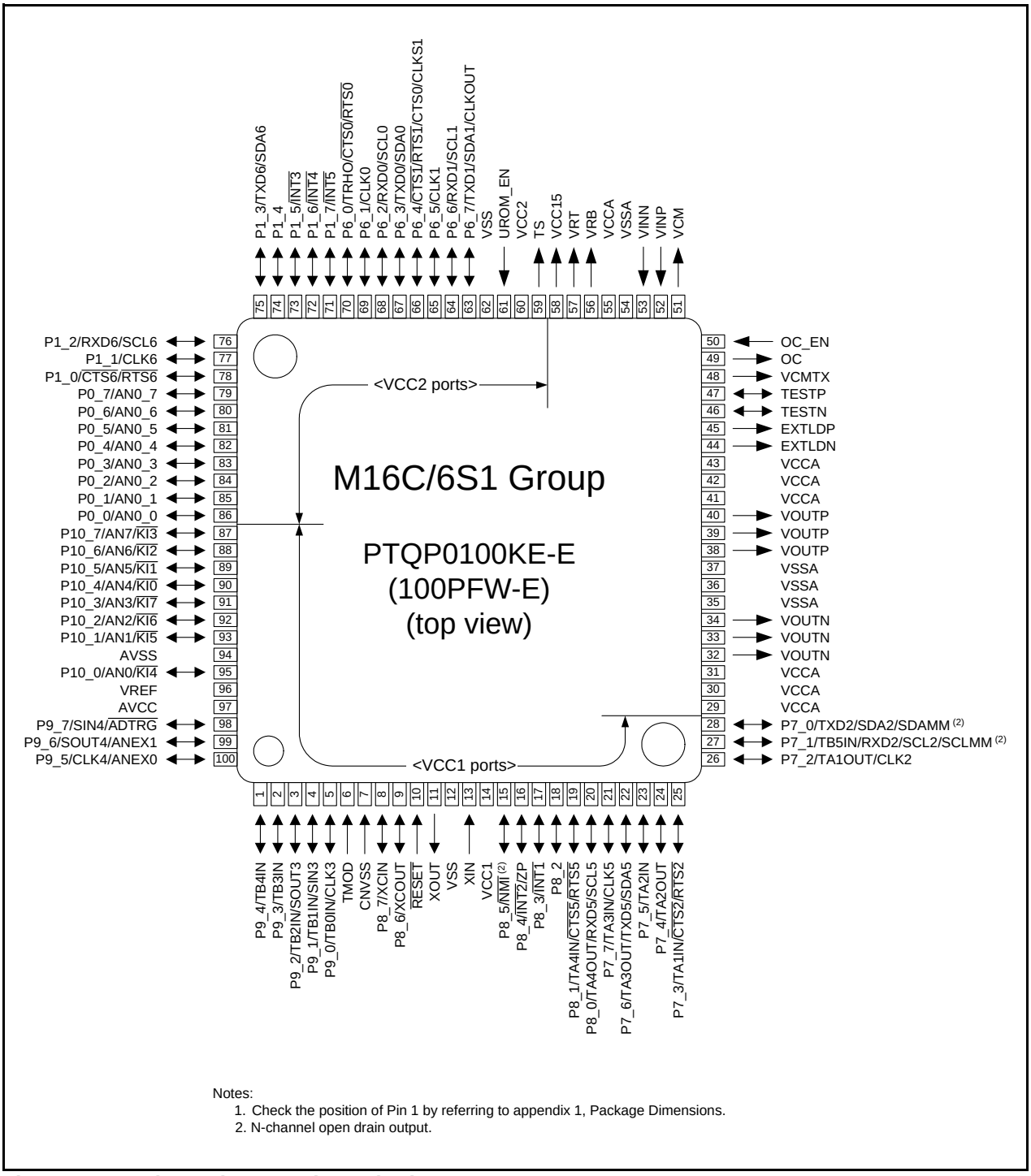


Figure 1.2 Block Diagram

1.5 Pin Assignments

Figure 1.3 shows Pin Assignment (Top View). Tables 1.6 to 1.8 list the Pin Names.



Notes:
1. Check the position of Pin 1 by referring to appendix 1, Package Dimensions.
2. N-channel open drain output.

Figure 1.3 Pin Assignment (Top View)

Table 1.6 Pin Names (1/3)

Pin No.	Power.	Clock	Control Pin	Port	I/O Pin for Peripheral Function				
					Interrupt	Timer	Serial interface	PLC	ADC
1				P9_4		TB4IN			
2				P9_3		TB3IN			
3				P9_2		TB2IN	SOUT3		
4				P9_1		TB1IN	SIN3		
5				P9_0		TB0IN	CLK3		
6			TMOD						
7			CNVSS						
8		XCIN		P8_7					
9		XCOUT		P8_6					
10			$\overline{\text{RESET}}$						
11		XOUT							
12	VSS								
13		XIN							
14	VCC1								
15				P8_5	$\overline{\text{NMI}}$				
16				P8_4	$\overline{\text{INT2}}$	ZP			
17				P8_3	$\overline{\text{INT1}}$				
18				P8_2					
19				P8_1		TA4IN	$\overline{\text{CTS5}}$/$\overline{\text{RTS5}}$		
20				P8_0		TA4OUT	RXD5/SCL5		
21				P7_7		TA3IN	CLK5		
22				P7_6		TA3OUT	TXD5/SDA5		
23				P7_5		TA2IN			
24				P7_4		TA2OUT			
25				P7_3		TA1IN	$\overline{\text{CTS2}}$/$\overline{\text{RTS2}}$		
26				P7_2		TA1OUT	CLK2		
27				P7_1		TB5IN	RXD2/SCL2/SCLMM		
28				P7_0			TXD2/SDA2/SDAMM		
29	VCCA								
30	VCCA								
31	VCCA								
32								VOUTN	
33								VOUTN	
34								VOUTN	
35	VSSA								
36	VSSA								
37	VSSA								
38								VOUTP	
39								VOUTP	
40								VOUTP	
41	VCCA								
42	VCCA								
43	VCCA								
44								EXTLDN	
45								EXTLDP	
46								TESTN	
47								TESTP	
48								VCMTX	

Table 1.7 Pin Names (2/3)

Pin No.	Power.	Clock	Control Pin	Port	I/O Pin for Peripheral Function				
					Interrupt	Timer	Serial interface	PLC	ADC
49								OC	
50								OC_EN	
51								VCM	
52								VINP	
53								VINN	
54	VSSA								
55	VCCA								
56								VRB	
57								VRT	
58								VCC15	
59								TS	
60	VCC2								
61			UROM_EN						
62	VSS								
63		CLKOUT		P6_7			TXD1/SDA1		
64				P6_6			RXD1/SCL1		
65				P6_5			CLK1		
66				P6_4			$\overline{\text{CTS1}}$/$\overline{\text{RTS1}}$/CTS0/CLKS1		
67				P6_3			TXD0/SDA0		
68				P6_2			RXD0/SCL0		
69				P6_1			CLK0		
70				P6_0		TRHO	$\overline{\text{CTS0}}$/$\overline{\text{RTS0}}$		
71				P1_7	$\overline{\text{INT5}}$				
72				P1_6	$\overline{\text{INT4}}$				
73				P1_5	$\overline{\text{INT3}}$				
74				P1_4					
75				P1_3			TXD6/SDA6		
76				P1_2			RXD6/SCL6		
77				P1_1			CLK6		
78				P1_0			$\overline{\text{CTS6}}$/$\overline{\text{RTS6}}$		
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	$\overline{\text{KI3}}$				AN7
88				P10_6	$\overline{\text{KI2}}$				AN6
89				P10_5	$\overline{\text{KI1}}$				AN5
90				P10_4	$\overline{\text{KI0}}$				AN4
91				P10_3	$\overline{\text{KI7}}$				AN3
92				P10_2	$\overline{\text{KI6}}$				AN2
93				P10_1	$\overline{\text{KI5}}$				AN1
94	AVSS								
95				P10_0	$\overline{\text{KI4}}$				AN0

Table 1.8 Pin Names (3/3)

Pin No.	Power.	Clock	Control Pin	Port	I/O Pin for Peripheral Function				
					Interrupt	Timer	Serial interface	PLC	ADC
96									VREF
97	AVCC								
98				P9_7			SIN4		$\overline{\text{ADTRG}}$
99				P9_6			SOUT4		ANEX1
100				P9_5			CLK4		ANEX0

1.6 Pin Functions

Table 1.9 Pin Functions (1/2)

Signal Name	Pin Name	I/O	Power Supply	Description
Digital power supply input	VCC1, VCC2, VSS	I	—	Apply 2.7 V to 3.6 V to pins VCC1 and VCC2, and 0 V to the VSS pin under the condition VCC1 = VCC2.
MCU ADC power supply input	AVCC, AVSS	I	VCC1	Power supply input for the A/D converter. Connect the AVCC pin to VCC1 and the AVSS pin to VSS.
PLC analog power supply input	VCCA, VSSA	I	—	Power supply input for AFE. Connect the VCCA pin to VCC1 and the VSSA pin to VSS.
Reset input	$\overline{\text{RESET}}$	I	VCC1	Driving this pin Low resets the MCU.
Mode setting input	CNVSS, TMOD	I	VCC1	Pins to set an operating mode. Connect both the CNVSS and TMOD pins to VSS via resistors.
	UROM_EN	I	VCC2	Connect to VSS via a resistor.
Main clock input	XIN	I	VCC1	I/O for the main clock oscillation circuit. Connect a crystal oscillator between pins XIN and XOUT.
Main clock output	XOUT	O	VCC1	
Sub clock input	XCIN	I	VCC1	I/O for the sub clock oscillation circuit. Connect a crystal oscillator between pins XCIN and XCOUT.
Sub clock output	XCOUT	O	VCC1	
Clock output	CLKOUT	O	VCC2	This pin outputs the clock having the same frequency as fC, f1, f8, or f32.
$\overline{\text{INT}}$ interrupt input	$\overline{\text{INT1}}$, $\overline{\text{INT2}}$	I	VCC1	Input for the $\overline{\text{INT}}$ interrupt.
	$\overline{\text{INT3}}$ to $\overline{\text{INT5}}$	I	VCC2	
$\overline{\text{NMI}}$ interrupt input	$\overline{\text{NMI}}$	I	VCC1	Input for the $\overline{\text{NMI}}$ interrupt.
Key input interrupt input	$\overline{\text{KI0}}$ to $\overline{\text{KI7}}$	I	VCC1	Input for the key input interrupt.
Timer A	TA1OUT to TA4OUT	I/O	VCC1	I/O for timers A1 to A4.
	TA1IN to TA4IN	I	VCC1	Input for timers A1 to A4.
	ZP	I	VCC1	Input for Z-phase.
Timer B	TB0IN to TB5INT	I	VCC1	Input for timers B0 to B5.
Real-time clock output	TRHO	O	VCC2	Output for the real-time clock.
Serial interface UART0 to UART2, UART5, UART6	$\overline{\text{CTS2}}$, $\overline{\text{CTS5}}$	I	VCC1	Input to control data transmission.
	$\overline{\text{CTS0}}$, $\overline{\text{CTS1}}$, $\overline{\text{CTS6}}$	I	VCC2	
	$\overline{\text{RTS2}}$, $\overline{\text{RTS5}}$	O	VCC1	Output to control data reception.
	$\overline{\text{RTS0}}$, $\overline{\text{RTS1}}$, $\overline{\text{RTS6}}$	O	VCC2	
	CLK2, CLK5	I/O	VCC1	Transmit/receive clock I/O.
	CLK0, CLK1, CLK6	I/O	VCC2	
	RXD2, RXD5	I	VCC1	Serial data input.
	RXD0, RXD1, RXD6	I	VCC2	
	TXD2, TXD5	O	VCC1	Serial data output.
	TXD0, TXD1, TXD6	O	VCC2	
	CLKS1	O	VCC1	Output for the transmit/receive clock multiple-pin output function.

Note:
1. TXD2 is an N-channel open drain output pin. TXDi (i = 0, 1, 5, 6) can be selected as a CMOS output pin or N-channel open drain output pin by a program.

Table 1.10 Pin Functions (2/2)

Signal Name	Pin Name	I/O	Power Supply	Description
UART0 to UART2, UART5, UART6 I^2C mode	SDA2, SDA5	I/O	VCC1	Serial data I/O for I^2C mode.
	SDA0, SDA1, SDA6	I/O	VCC2	
	SCL2, SCL5	I/O	VCC1	Transmit/receive clock I/O for I^2C mode.
	SCL0, SCL1, SCL6	I/O	VCC2	
Serial interface SI/O3, SI/O4	CLK3, CLK4	I/O	VCC1	Transmit/receive clock I/O.
	SIN3, SIN4	I	VCC1	Serial data input.
	SOUT3, SOUT4	O	VCC1	Serial data output.
Multi-master I^2C-bus interface	SDAMM	I/O	VCC1	Serial data I/O. (Output is N-channel open drain)
	SCLMM	I/O	VCC1	Transmit/receive clock I/O. (Output is N-channel open drain)
Reference voltage input	VREF	I	VCC1	Reference voltage input for the A/D converter.
A/D converter	AN0 to AN7	I	VCC1	Analog input for the A/D converter.
	AN0_0 to AN0_7	I	VCC2	
	$\overline{\text{ADTRG}}$	I	VCC1	External activation source input.
	ANEX0, ANEX1	I	VCC1	Extended analog input for the AD converter.
I/O ports	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7	I/O	VCC2	8-bit CMOS I/O ports. Each port has a direction register, allowing each pin in the port to be directed for input or output individually. A pull-up resistor may be enabled or disabled for input ports in 4-bit units.
	P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7	I/O	VCC1	8-bit I/O ports having equivalent functions to P0. However, P7_0, P7_1, and P8_5 are N-channel open drain output ports. No pull-up resistor is provided. P8_5 is an input port for verifying the $\overline{\text{NMI}}$ pin level and shares a pin with $\overline{\text{NMI}}$.

Note:
1. TXD, SDA2, SCL2, SDAMM, and SCLMM are N-channel open drain output pins. TXDi (i = 0, 1, 3, 6), SDAi (i = 0, 1, 5, 6), and SCLi (i = 0, 1, 5, 6) can be selected as CMOS output pins or N-channel open drain output pins by a program.

Table 1.11 Internal Connection Pin Functions

Signal Name	Pin Name	I/O	Description
Internal interrupt input	$\overline{\text{INT0}}$, $\overline{\text{INT6}}$, $\overline{\text{INT7}}$	I	$\overline{\text{INT}}$ interrupt input from PLC modem.
Internal timer A connection	TA0OUT	I/O	Timer A0 I/O from/to PLC modem.
	TA0IN	I	Timer A0 input from PLC modem.
Internal serial connection	TXD7	O	Serial data output to PLC modem.
	RXD7	I	Serial data input from PLC modem.
	CLK7	I/O	Transmit/receive clock I/O from/to PLC modem.
Internal ports	P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7	I/O	Ports to connect PLC modem. Used as addresses, data, control signals, and status signals to access registers of PLC modem and AFE.

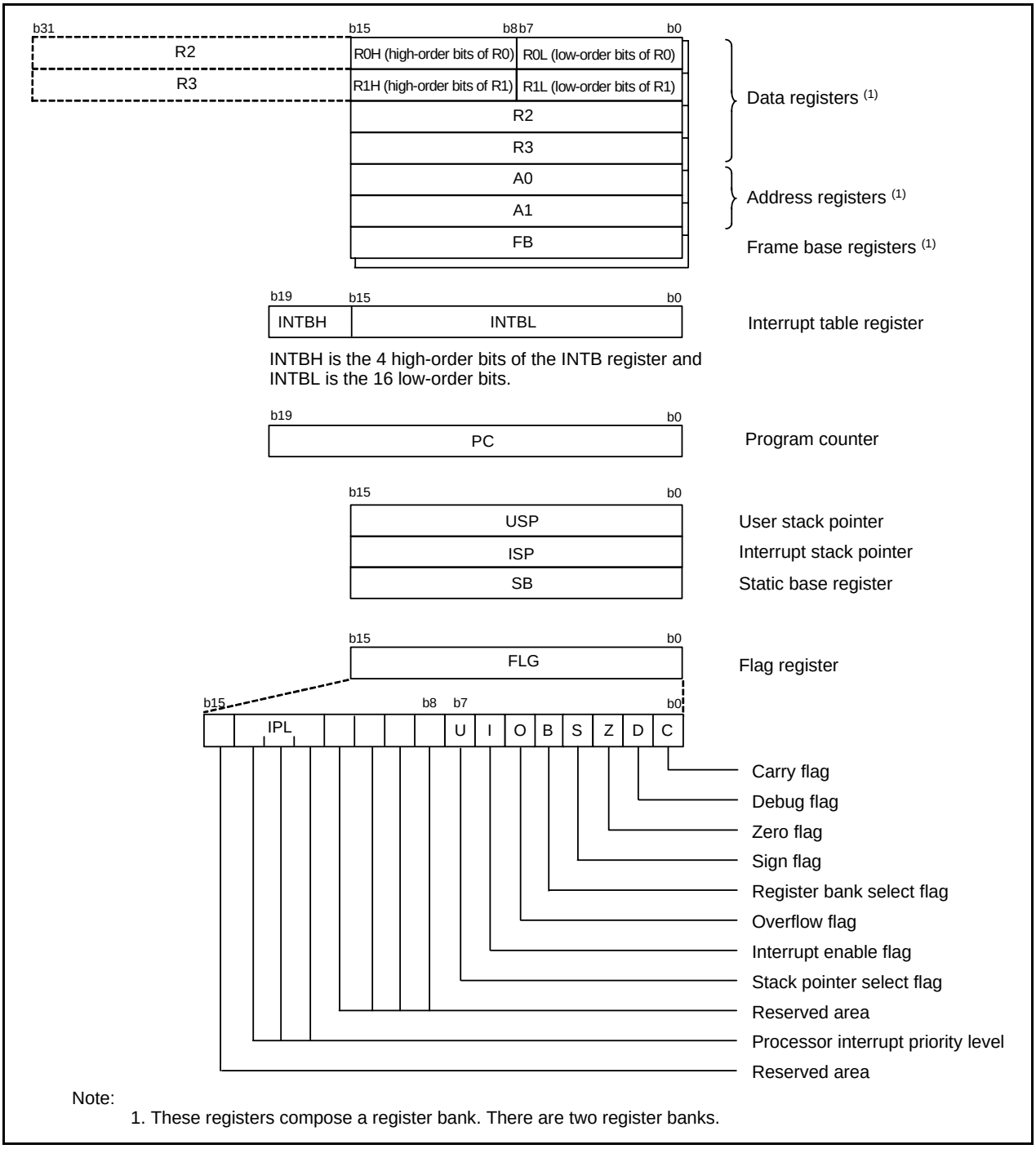
Note:
1. On-chip PLC modem should be controlled by DLL software released by Renesas. Do not access the PLC control registers directly by any user program.

Table 1.12 PLC Block Pin Functions

Signal Name	Pin Name	I/O	Description
RX signal	VINP	Analog input	Input pins for differential reception signals.
	VINN		
TX signal	VOUTP	Analog output	Output pins for differential transmission signals.
	VOUTN		
Analog pins	EXTLDP	Analog output	Differential transmission output pins for the external line driver as optional specifications.
	EXTLDN		
	VCMTX	Analog output	Reference voltage output pin for the analog circuit of the transmitter block. Connect to a bypass capacitor for VSSA.
	VRT	Analog output	Reference voltage output pins for ADC of PLC block. Connect to a bypass capacitor for VSSA.
	VRB		
	VCM	Analog output	Reference voltage output pin for the analog circuit of receiver block. Connect to a bypass capacitor for VSSA.
Testing pins	TESTP	Analog I/O	I/O for testing. Leave open.
	TESTN		
Digital pins	OC_EN	I	Connect to VCCA via a register when using internal line driver for enabling over current protection. Connect to VSSA via register when using external line driver for disabling over current protection.
	OC	O	Status output pin for the overcurrent protection circuit.
	TS	O	Pin to control turning off/on the external line driver.
Regulator output	VCC15	O	Regulator output pin (1.5 V) for the digital circuit of PLC block. Connect only to a bypass capacitor for VSS. Do not use this pin to provide power supply to other circuits.

2. Central Processing Unit (CPU)

Figure 2.1 shows the CPU registers. Seven registers (R0, R1, R2, R3, A0, A1, and FB) out of 13 compose a register bank, and there are two register banks.



Note:
1. These registers compose a register bank. There are two register banks.

Figure 2.1 CPU Registers

2.1 Data Registers (R0, R1, R2, and R3)

R0, R1, R2, and R3 are 16-bit registers used for transfer, arithmetic, and logic operations. R0 and R1 can be split into high-order (R0H/R1H) and low-order (R0L/R1L) bits to be used separately as 8-bit data registers.

R0 can be combined with R2, and R3 can be combined with R1 and be used as 32-bit data registers R2R0 and R3R1, respectively.

2.2 Address Registers (A0 and A1)

A0 and A1 are 16-bit registers used for indirect addressing, relative addressing, transfer, arithmetic, and logic operations. A0 can be combined with A1 and used as a 32-bit address register (A1A0).

2.3 Frame Base Register (FB)

FB is a 16-bit register that is used for FB relative addressing.

2.4 Interrupt Table Register (INTB)

INTB is a 20-bit register that indicates the start address of a relocatable interrupt vector table.

2.5 Program Counter (PC)

The PC is 20 bits wide and indicates the address of the next instruction to be executed.

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

The USP and ISP stack pointers (SP) are each comprised of 16 bits. The U flag is used to switch between USP and ISP.

2.7 Static Base Register (SB)

SB is a 16-bit register used for SB relative addressing.

2.8 Flag Register (FLG)

FLG is an 11-bit register that indicates the CPU state.

2.8.1 Carry Flag (C Flag)

The C flag retains a carry, borrow, or shift-out bit generated by the arithmetic/logic unit.

2.8.2 Debug Flag (D Flag)

The D flag is for debugging only. Set it to 0.

2.8.3 Zero Flag (Z Flag)

The Z flag becomes 1 when an arithmetic operation results in 0. Otherwise, it becomes 0.

2.8.4 Sign Flag (S Flag)

The S flag becomes 1 when an arithmetic operation results in a negative value. Otherwise, it becomes 0.

2.8.5 Register Bank Select Flag (B Flag)

Register bank 0 is selected when the B flag is 0. Register bank 1 is selected when this flag is 1.

2.8.6 Overflow Flag (O Flag)

The O flag becomes 1 when an arithmetic operation results in an overflow. Otherwise, it becomes 0.

2.8.7 Interrupt Enable Flag (I Flag)

The I flag enables maskable interrupts.
Maskable interrupts are disabled when the I flag is 0, and enabled when it is 1. The I flag becomes 0 when an interrupt request is accepted.

2.8.8 Stack Pointer Select Flag (U Flag)

ISP is selected when the U flag is 0. USP is selected when the U flag is 1.
The U flag becomes 0 when a hardware interrupt request is accepted, or the INT instruction of software interrupt number 0 to 31 is executed.

2.8.9 Processor Interrupt Priority Level (IPL)

IPL is 3 bits wide and assigns processor interrupt priority levels from 0 to 7.
If a requested interrupt has higher priority than IPL, the interrupt request is enabled.

2.8.10 Reserved Areas

Only set these bits to 0. The read value is undefined.

3. Address Space

3.1 Address Space

The M16C/6S1 Group has a 1 MB address space from 00000h to FFFFFh.

3.2 Memory Map

Special function registers (SFRs) are allocated from address 00000h to 003FFh and from 0D000h to 0D7FFh. Peripheral function control registers are located here. All blank areas within SFRs are reserved. Do not access these areas.

Internal RAM is allocated from address 00400h and higher, with 20 KB of internal RAM allocated from 00400h to 053FFh. Internal RAM is used not only for data storage, but also for the stack area when subroutines are called or when an interrupt request is accepted.

The internal ROM is flash memory. Three internal ROM areas are available: data flash, program ROM 1, and program ROM 2.

The data flash is allocated from 0E000h to 0FFFFh. This data flash area is mostly used for data storage, but can also store programs.

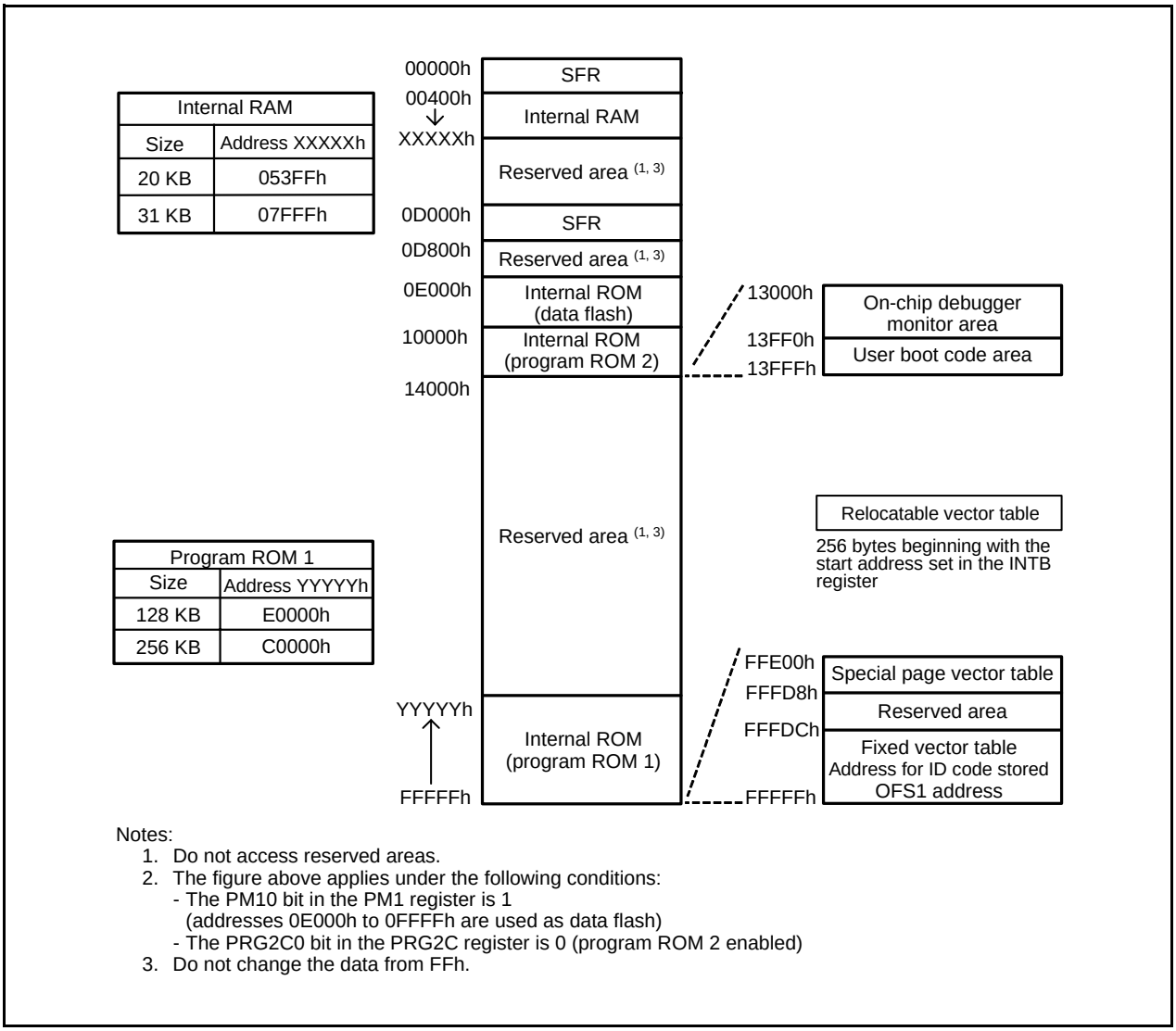
Program ROM 2 is allocated from 10000h to 13FFFh. Program ROM 1 is allocated from FFFFFh and lower, with the 128-KB program ROM 1 area allocated from address E0000h to FFFFFh.

The special page vectors are allocated from FFE00h to FFFD7h. They are used for the JMPS and JSRS instructions. Refer to the M16C/60, M16C/20, M16C/Tiny Series User's Manual: Software for details.

The fixed vector table for interrupts is allocated from FFFDCh to FFFFFh.

The 256 bytes beginning with the start address set in the INTB register compose the relocatable vector table for interrupts.

Figure 3.1 shows the Memory Map.



Notes:
1. Do not access reserved areas.
2. The figure above applies under the following conditions:
   - The PM10 bit in the PM1 register is 1
     (addresses 0E000h to 0FFFFh are used as data flash)
   - The PRG2C0 bit in the PRG2C register is 0 (program ROM 2 enabled)
3. Do not change the data from FFh.

Figure 3.1 Memory Map

4. Special Function Registers (SFRs)

4.1 SFRs

An SFR is a control register for a peripheral function. Tables 4.1 to 4.16 list the SFR information.

Table 4.1 SFR Information (1/16) (1)

Address	Register	Symbol	Reset Value
0000h			
0001h			
0002h			
0003h			
0004h	Processor Mode Register 0	PM0	0000 0000b
0005h	Processor Mode Register 1	PM1	0000 1000b
0006h	System Clock Control Register 0	CM0	0100 1000b
0007h	System Clock Control Register 1	CM1	0010 0000b
0008h			
0009h			
000Ah	Protect Register	PRCR	00h
000Bh			
000Ch	Oscillation Stop Detection Register	CM2	0X00 0010b (2)
000Dh			
000Eh			
000Fh			
0010h	Program 2 Area Control Register	PRG2C	XXXX XX00b
0011h			
0012h	Peripheral Clock Select Register	PCLKR	0000 0011b
0013h	Sub Clock Division Control Register	SCM0	XXXX X000b
0014h			
0015h	Clock Prescaler Reset Flag	CPSRF	0XXX XXXXb
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
0017h			
0018h	Reset Source Determine Register	RSTFR	XX00 001Xb (3)
0019h			
001Ah	Voltage Detector Operation Enable Register	VCR2	0000 0000b (4) 001X 0000b (4)
001Bh			
001Ch	PLL Control Register 0	PLC0	0001 X010b
001Dh	PLL Function Lock Control Register	PLCF	
001Eh	Processor Mode Register 2	PM2	XX00 0X01b
001Fh			

Notes: X: Undefined

1. The blank areas are reserved. No access is allowed.
2. Oscillator stop detect reset does not affect bits CM20, CM21, and CM27.
3. The state of bits in the RSTFR register depends on the reset type.
4. This is the reset value after hardware reset. Refer to the explanation of each register for details.

Table 4.2 SFR Information (2/16) (1)

Address	Register	Symbol	Reset Value
0020h			
0021h			
0022h			
0023h			
0024h			
0025h			
0026h			
0027h			
0028h			
0029h			
002Ah	Voltage Monitor 0 Control Register	VW0C	1000 XX10b (2, 3) 1100 XX11b (2)
002Bh			
002Ch	Watchdog Timer Detector Register	VW2C	1000 0X10b
002Dh			
002Eh			
002Fh			
0030h			
0031h			
0032h			
0033h			
0034h			
0035h			
0036h			
0037h			
0038h			
0039h			
003Ah			
003Bh			
003Ch			
003Dh			
003Eh			
003Fh			
0040h			
0041h			
0042h	$\overline{\text{INT7}}$ Interrupt Control Register	INT7IC	XX00 X000b
0043h	$\overline{\text{INT6}}$ Interrupt Control Register	INT6IC	XX00 X000b
0044h	$\overline{\text{INT3}}$ Interrupt Control Register	INT3IC	XX00 X000b
0045h	Timer B5 Interrupt Control Register	TB5IC	XXXX X000b
0046h	Timer B4 Interrupt Control Register UART1 Bus Collision Detection Interrupt Control Register	TB4IC U1BCNIC	XXXX X000b
0047h	Timer B3 Interrupt Control Register UART0 Bus Collision Detection Interrupt Control Register	TB3IC U0BCNIC	XXXX X000b
0048h	SI/O4 Interrupt Control Register $\overline{\text{INT5}}$ Interrupt Control Register	S4IC INT5IC	XX00 X000b
0049h	SI/O3 Interrupt Control Register $\overline{\text{INT4}}$ Interrupt Control Register	S3IC INT4IC	XX00 X000b
004Ah	UART2 Bus Collision Detection Interrupt Control Register	BCNIC	XXXX X000b
004Bh	DMA0 Interrupt Control Register	DM0IC	XXXX X000b
004Ch	DMA1 Interrupt Control Register	DM1IC	XXXX X000b
004Dh	Key Input Interrupt Control Register	KUPIC	XX00 X000b
004Eh	A/D Conversion Interrupt Control Register	ADIC	XXXX X000b
004Fh	UART2 Transmit Interrupt Control Register	S2TIC	XXXX X000b

Notes: X: Undefined

1. The blank areas are reserved. No access is allowed.
2. This is the reset value after hardware reset. Refer to the explanation of each register for details.
3. This is the reset value when the LVDAS bit of address OFS1 is 1 during hardware reset

Table 4.3 SFR Information (3/16) (1)

Address	Register	Symbol	Reset Value
0050h	UART2 Receive Interrupt Control Register	S2RIC	XXXX X000b
0051h	UART0 Transmit Interrupt Control Register	S0TIC	XXXX X000b
0052h	UART0 Receive Interrupt Control Register	S0RIC	XXXX X000b
0053h	UART1 Transmit Interrupt Control Register	S1TIC	XXXX X000b
0054h	UART1 Receive Interrupt Control Register	S1RIC	XXXX X000b
0055h	Timer A0 Interrupt Control Register	TA0IC	XXXX X000b
0056h	Timer A1 Interrupt Control Register	TA1IC	XXXX X000b
0057h	Timer A2 Interrupt Control Register	TA2IC	XXXX X000b
0058h	Timer A3 Interrupt Control Register	TA3IC	XXXX X000b
0059h	Timer A4 Interrupt Control Register	TA4IC	XXXX X000b
005Ah	Timer B0 Interrupt Control Register	TB0IC	XXXX X000b
005Bh	Timer B1 Interrupt Control Register	TB1IC	XXXX X000b
005Ch	Timer B2 Interrupt Control Register	TB2IC	XXXX X000b
005Dh	$\overline{\text{INT0}}$ Interrupt Control Register	INT0IC	XX00 X000b
005Eh	$\overline{\text{INT1}}$ Interrupt Control Register	INT1IC	XX00 X000b
005Fh	$\overline{\text{INT2}}$ Interrupt Control Register	INT2IC	XX00 X000b
0060h			
0061h			
0062h			
0063h			
0064h			
0065h			
0066h			
0067h			
0068h			
0069h	DMA2 Interrupt Control Register	DM2IC	XXXX X000b
006Ah	DMA3 Interrupt Control Register	DM3IC	XXXX X000b
006Bh	UART5 Bus Collision Detection Interrupt Control Register	U5BCNIC	XXXX X000b
006Ch	UART5 Transmit Interrupt Control Register	S5TIC	XXXX X000b
006Dh	UART5 Receive Interrupt Control Register	S5RIC	XXXX X000b
006Eh	UART6 Bus Collision Detection Interrupt Control Register Real-Time Clock Periodic Interrupt Control Register	U6BCNIC RTCTIC	XXXX X000b
006Fh	UART6 Transmit Interrupt Control Register Real-Time Clock Alarm Interrupt Control Register	S6TIC RTCCIC	XXXX X000b
0070h	UART6 Receive Interrupt Control Register	S6RIC	XXXX X000b
0071h	UART7 Bus Collision Detection Interrupt Control Register	U7BCNIC	XXXX X000b
0072h	UART7 Transmit Interrupt Control Register	S7TIC	XXXX X000b
0073h	UART7 Receive Interrupt Control Register	S7RIC	XXXX X000b
0074h			
0075h			
0076h			
0077h			
0078h			
0079h			
007Ah			
007Bh	I2C-bus Interface Interrupt Control Register	IICIC	XXXX X000b
007Ch	SCL/SDA Interrupt Control Register	SCLDAIC	XXXX X000b
007Dh			
007Eh			
007Fh			
0080h to 017Fh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.4 SFR Information (4/16) (1)

Address	Register	Symbol	Reset Value
0180h	DMA0 Source Pointer	SAR0	XXh
0181h			XXh
0182h			0Xh
0183h			
0184h	DMA0 Destination Pointer	DAR0	XXh
0185h			XXh
0186h			0Xh
0187h			
0188h	DMA0 Transfer Counter	TCR0	XXh
0189h			XXh
018Ah			
018Bh			
018Ch	DMA0 Control Register	DM0CON	0000 0X00b
018Dh			
018Eh			
018Fh			
0190h	DMA1 Source Pointer	SAR1	XXh
0191h			XXh
0192h			0Xh
0193h			
0194h	DMA1 Destination Pointer	DAR1	XXh
0195h			XXh
0196h			0Xh
0197h			
0198h	DMA1 Transfer Counter	TCR1	XXh
0199h			XXh
019Ah			
019Bh			
019Ch	DMA1 Control Register	DM1CON	0000 0X00b
019Dh			
019Eh			
019Fh			
01A0h	DMA2 Source Pointer	SAR2	XXh
01A1h			XXh
01A2h			0Xh
01A3h			
01A4h	DMA2 Destination Pointer	DAR2	XXh
01A5h			XXh
01A6h			0Xh
01A7h			
01A8h	DMA2 Transfer Counter	TCR2	XXh
01A9h			XXh
01AAh			
01ABh			
01ACh	DMA2 Control Register	DM2CON	0000 0X00b
01ADh			
01AEh			
01AFh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.5 SFR Information (5/16) (1)

Address	Register	Symbol	Reset Value
01B0h	DMA3 Source Pointer	SAR3	XXh
01B1h			XXh
01B2h			0Xh
01B3h			
01B4h	DMA3 Destination Pointer	DAR3	XXh
01B5h			XXh
01B6h			0Xh
01B7h			
01B8h	DMA3 Transfer Counter	TCR3	XXh
01B9h			XXh
01BAh			
01BBh			
01BCh	DMA3 Control Register	DM3CON	0000 0X00b
01BDh			
01BEh			
01BFh			
01C0h	Timer B0-1 Register	TB01	XXh
01C1h			XXh
01C2h	Timer B1-1 Register	TB11	XXh
01C3h			XXh
01C4h	Timer B2-1 Register	TB21	XXh
01C5h			XXh
01C6h	Pulse Period/Pulse Width Measurement Mode Function Select Register 1	PPWFS1	XXXX X000b
01C7h			
01C8h	Timer B Count Source Select Register 0	TBCS0	00h
01C9h	Timer B Count Source Select Register 1	TBCS1	X0h
01CAh			
01CBh			
01CCh			
01CDh			
01CEh			
01CFh			
01D0h	Timer A Count Source Select Register 0	TACS0	00h
01D1h	Timer A Count Source Select Register 1	TACS1	00h
01D2h	Timer A Count Source Select Register 2	TACS2	X0h
01D3h			
01D4h	16-Bit Pulse Width Modulation Mode Function Select Register	PWMFS	0XX0 X00Xb
01D5h	Timer A Waveform Output Function Select Register	TAPOFS	XXX0 0000b
01D6h			
01D7h			
01D8h	Timer A Output Waveform Change Enable Register	TAOW	XXX0 X00Xb
01D9h			
01DAh			
01DBh			
01DCh			
01DDh			
01DEh			
01DFh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.6 SFR Information (6/16) (1)

Address	Register	Symbol	Reset Value
01E0h	Timer B3-1 Register	TB31	XXh
01E1h			XXh
01E2h	Timer B4-1 Register	TB41	XXh
01E3h			XXh
01E4h	Timer B5-1 Register	TB51	XXh
01E5h			XXh
01E6h	Pulse Period/Pulse Width Measurement Mode Function Select Register 2	PPWFS2	XXXX X000b
01E7h			
01E8h	Timer B Count Source Select Register 2	TBCS2	00h
01E9h	Timer B Count Source Select Register 3	TBCS3	X0h
01EAh			
01EBh			
01ECh			
01EDh			
01EEh			
01EFh			
01F0h			
01F1h			
01F2h			
01F3h			
01F4h			
01F5h			
01F6h			
01F7h			
01F8h			
01F9h			
01FAh			
01FBh			
01FCh			
01FDh			
01FEh			
01FFh			
0200h			
0201h			
0202h			
0203h			
0204h			
0205h	Interrupt Source Select Register 3	IFSR3A	00h
0206h	Interrupt Source Select Register 2	IFSR2A	00h
0207h	Interrupt Source Select Register	IFSR	00h
0208h			
0209h			
020Ah			
020Bh			
020Ch			
020Dh			
020Eh	Address Match Interrupt Enable Register	AIER	XXXX XX00b
020Fh	Address Match Interrupt Enable Register 2	AIER2	XXXX XX00b

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.7 SFR Information (7/16) (1)

Address	Register	Symbol	Reset Value
0210h	Address Match Interrupt Register 0	RMAD0	00h
0211h			00h
0212h			X0h
0213h			
0214h	Address Match Interrupt Register 1	RMAD1	00h
0215h			00h
0216h			X0h
0217h			
0218h	Address Match Interrupt Register 2	RMAD2	00h
0219h			00h
021Ah			X0h
021Bh			
021Ch	Address Match Interrupt Register 3	RMAD3	00h
021Dh			00h
021Eh			X0h
021Fh			
0220h	Flash Memory Control Register 0	FMR0	0000 0001b (Other than user boot mode) 0010 0001b (User boot mode)
0221h	Flash Memory Control Register 1	FMR1	00X0 XX0Xb
0222h	Flash Memory Control Register 2	FMR2	XXXX 0000b
0223h	Flash Memory Control Register 3	FMR3	XXXX 0000b
0224h			
0225h			
0226h			
0227h			
0228h			
0229h			
022Ah			
022Bh			
022Ch			
022Dh			
022Eh			
022Fh			
0230h	Flash Memory Control Register 6	FMR6	XX0X XX00b
0231h	Flash Memory Control Register 7	FMR7	1000 0000b
0232h			
0233h			
0234h			
0235h			
0236h			
0237h			
0238h			
0239h			
023Ah			
023Bh			
023Ch			
023Dh			
023Eh			
023Fh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.8 SFR Information (8/16) (1)

Address	Register	Symbol	Reset Value
0240h			
0241h			
0242h			
0243h			
0244h	UART0 Special Mode Register 4	U0SMR4	00h
0245h	UART0 Special Mode Register 3	U0SMR3	000X 0X0Xb
0246h	UART0 Special Mode Register 2	U0SMR2	X000 0000b
0247h	UART0 Special Mode Register	U0SMR	X000 0000b
0248h	UART0 Transmit/Receive Mode Register	U0MR	00h
0249h	UART0 Bit Rate Register	U0BRG	XXh
024Ah	UART0 Transmit Buffer Register	U0TB	XXh
024Bh			XXh
024Ch	UART0 Transmit/Receive Control Register 0	U0C0	0000 1000b
024Dh	UART0 Transmit/Receive Control Register 1	U0C1	00XX 0010b
024Eh	UART0 Receive Buffer Register	U0RB	XXh
024Fh			XXh
0250h	UART Transmit/Receive Control Register 2	UCON	X000 0000b
0251h			
0252h			
0253h			
0254h	UART1 Special Mode Register 4	U1SMR4	00h
0255h	UART1 Special Mode Register 3	U1SMR3	000X 0X0Xb
0256h	UART1 Special Mode Register 2	U1SMR2	X000 0000b
0257h	UART1 Special Mode Register	U1SMR	X000 0000b
0258h	UART1 Transmit/Receive Mode Register	U1MR	00h
0259h	UART1 Bit Rate Register	U1BRG	XXh
025Ah	UART1 Transmit Buffer Register	U1TB	XXh
025Bh			XXh
025Ch	UART1 Transmit/Receive Control Register 0	U1C0	0000 1000b
025Dh	UART1 Transmit/Receive Control Register 1	U1C1	00XX 0010b
025Eh	UART1 Receive Buffer Register	U1RB	XXh
025Fh			XXh
0260h			
0261h			
0262h			
0263h			
0264h	UART2 Special Mode Register 4	U2SMR4	00h
0265h	UART2 Special Mode Register 3	U2SMR3	000X 0X0Xb
0266h	UART2 Special Mode Register 2	U2SMR2	X000 0000b
0267h	UART2 Special Mode Register	U2SMR	X000 0000b
0268h	UART2 Transmit/Receive Mode Register	U2MR	00h
0269h	UART2 Bit Rate Register	U2BRG	XXh
026Ah	UART2 Transmit Buffer Register	U2TB	XXh
026Bh			XXh
026Ch	UART2 Transmit/Receive Control Register 0	U2C0	0000 1000b
026Dh	UART2 Transmit/Receive Control Register 1	U2C1	0000 0010b
026Eh	UART2 Receive Buffer Register	U2RB	XXh
026Fh			XXh

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.9 SFR Information (9/16) (1)

Address	Register	Symbol	Reset Value
0270h	SI/O3 Transmit/Receive Register	S3TRR	XXh
0271h			
0272h	SI/O3 Control Register	S3C	0100 0000b
0273h	SI/O3 Bit Rate Register	S3BRG	XXh
0274h	SI/O4 Transmit/Receive Register	S4TRR	XXh
0275h			
0276h	SI/O4 Control Register	S4C	0100 0000b
0277h	SI/O4 Bit Rate Register	S4BRG	XXh
0278h	SI/O3, 4 Control Register 2	S34C2	00XX X0X0b
0279h			
027Ah			
027Bh			
027Ch			
027Dh			
027Eh			
027Fh			
0280h			
0281h			
0282h			
0283h			
0284h	UART5 Special Mode Register 4	U5SMR4	00h
0285h	UART5 Special Mode Register 3	U5SMR3	000X 0X0Xb
0286h	UART5 Special Mode Register 2	U5SMR2	X000 0000b
0287h	UART5 Special Mode Register	U5SMR	X000 0000b
0288h	UART5 Transmit/Receive Mode Register	U5MR	00h
0289h	UART5 Bit Rate Register	U5BRG	XXh
028Ah	UART5 Transmit Buffer Register	U5TB	XXh
028Bh			XXh
028Ch	UART5 Transmit/Receive Control Register 0	U5C0	0000 1000b
028Dh	UART5 Transmit/Receive Control Register 1	U5C1	0000 0010b
028Eh	UART5 Receive Buffer Register	U5RB	XXh
028Fh			XXh
0290h			
0291h			
0292h			
0293h			
0294h	UART6 Special Mode Register 4	U6SMR4	00h
0295h	UART6 Special Mode Register 3	U6SMR3	000X 0X0Xb
0296h	UART6 Special Mode Register 2	U6SMR2	X000 0000b
0297h	UART6 Special Mode Register	U6SMR	X000 0000b
0298h	UART6 Transmit/Receive Mode Register	U6MR	00h
0299h	UART6 Bit Rate Register	U6BRG	XXh
029Ah	UART6 Transmit Buffer Register	U6TB	XXh
029Bh			XXh
029Ch	UART6 Transmit/Receive Control Register 0	U6C0	0000 1000b
029Dh	UART6 Transmit/Receive Control Register 1	U6C1	0000 0010b
029Eh	UART6 Receive Buffer Register	U6RB	XXh
029Fh			XXh

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.10 SFR Information (10/16) (1)

Address	Register	Symbol	Reset Value
02A0h			
02A1h			
02A2h			
02A3h			
02A4h	UART7 Special Mode Register 4	U7SMR4	00h
02A5h	UART7 Special Mode Register 3	U7SMR3	000X 0X0Xb
02A6h	UART7 Special Mode Register 2	U7SMR2	X000 0000b
02A7h	UART7 Special Mode Register	U7SMR	X000 0000b
02A8h	UART7 Transmit/Receive Mode Register	U7MR	00h
02A9h	UART7 Bit Rate Register	U7BRG	XXh
02AAh	UART7 Transmit Buffer Register	U7TB	XXh
02ABh			XXh
02ACh	UART7 Transmit/Receive Control Register 0	U7C0	0000 1000b
02ADh	UART7 Transmit/Receive Control Register 1	U7C1	0000 0010b
02AEh	UART7 Receive Buffer Register	U7RB	XXh
02AFh			XXh
02B0h	I2C0 Data Shift Register	S00	XXh
02B1h			
02B2h	I2C0 Address Register 0	S0D0	0000 000Xb
02B3h	I2C0 Control Register 0	S1D0	00h
02B4h	I2C0 Clock Control Register	S20	00h
02B5h	I2C0 Start/Stop Condition Control Register	S2D0	0001 1010b
02B6h	I2C0 Control Register 1	S3D0	0011 0000b
02B7h	I2C0 Control Register 2	S4D0	00h
02B8h	I2C0 Status Register 0	S10	0001 000Xb
02B9h	I2C0 Status Register 1	S11	XXXX X000b
02BAh	I2C0 Address Register 1	S0D1	0000 000Xb
02BBh	I2C0 Address Register 2	S0D2	0000 000Xb
02BCh			
02BDh			
02BEh			
02BFh			
02C0h to 02FFh			
0300h	Timer B3/B4/B5 Count Start Flag	TBSR	000X XXXXb
0301h			
0302h	Timer A1-1 Register	TA11	XXh
0303h			XXh
0304h	Timer A2-1 Register	TA21	XXh
0305h			XXh
0306h	Timer A4-1 Register	TA41	XXh
0307h			XXh
0308h			
0309h			
030Ah			
030Bh			
030Ch			
030Dh			
030Eh			
030Fh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.11 SFR Information (11/16) (1)

Address	Register	Symbol	Reset Value
0310h	Timer B3 Register	TB3	XXh
0311h			XXh
0312h	Timer B4 Register	TB4	XXh
0313h			XXh
0314h	Timer B5 Register	TB5	XXh
0315h			XXh
0316h			
0317h			
0318h			
0319h			
031Ah			
031Bh	Timer B3 Mode Register	TB3MR	00XX 0000b
031Ch	Timer B4 Mode Register	TB4MR	00XX 0000b
031Dh	Timer B5 Mode Register	TB5MR	00XX 0000b
031Eh			
031Fh			
0320h	Count Start Flag	TABSR	00h
0321h			
0322h	One-Shot Start Flag	ONSF	00h
0323h	Trigger Select Register	TRGSR	00h
0324h	Up/Down Flag	UDF	00h
0325h			
0326h	Timer A0 Register	TA0	XXh
0327h			XXh
0328h	Timer A1 Register	TA1	XXh
0329h			XXh
032Ah	Timer A2 Register	TA2	XXh
032Bh			XXh
032Ch	Timer A3 Register	TA3	XXh
032Dh			XXh
032Eh	Timer A4 Register	TA4	XXh
032Fh			XXh
0330h	Timer B0 Register	TB0	XXh
0331h			XXh
0332h	Timer B1 Register	TB1	XXh
0333h			XXh
0334h	Timer B2 Register	TB2	XXh
0335h			XXh
0336h	Timer A0 Mode Register	TA0MR	00h
0337h	Timer A1 Mode Register	TA1MR	00h
0338h	Timer A2 Mode Register	TA2MR	00h
0339h	Timer A3 Mode Register	TA3MR	00h
033Ah	Timer A4 Mode Register	TA4MR	00h
033Bh	Timer B0 Mode Register	TB0MR	00XX 0000b
033Ch	Timer B1 Mode Register	TB1MR	00XX 0000b
033Dh	Timer B2 Mode Register	TB2MR	00XX 0000b
033Eh			
033Fh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.12 SFR Information (12/16) (1)

Address	Register	Symbol	Reset Value
0340h	Second Data Register	TRHSEC	0000 0000b
0341h	Minute Data Register	TRHMIN	0000 0000b
0342h	Hour Data Register	TRHHR	0000 0000b
0343h	Day-of-the-Week Data Register	TRHWK	0000 0000b
0344h	Date Data Register	TRHDY	0000 0001b
0345h	Month Data Register	TRHMON	0000 0001b
0346h	Year Data Register	TRHYR	0000 0000b
0347h	Timer RH Control Register	TRHCR	0000 0100b
0348h	Timer RH Count Source Select Register	TRHCSR	0000 1000b
0349h	Clock Error Correction Register	TRHADJ	0000 0000b
034Ah	Timer RH Interrupt Flag Register	TRHIFR	XXX0 0000b
034Bh	Timer RH Interrupt Enable Register	TRHIER	0000 0000b
034Ch	Alarm Minute Register	TRHAMN	0000 0000b
034Dh	Alarm Hour Register	TRHAHR	0000 0000b
034Eh	Alarm Day-of-the-Week Register	TRHAWK	0XXX X000b
034Fh	Timer RH Protect Register	TRHPRC	00XX XXXXb
0350h			
0351h			
0352h			
0353h			
0354h			
0355h			
0356h			
0357h			
0358h			
0359h			
035Ah			
035Bh			
035Ch			
035Dh			
035Eh			
035Fh			
0360h	Pull-Up Control Register 0	PUR0	00h
0361h	Pull-Up Control Register 1	PUR1	0000 0000b
0362h	Pull-Up Control Register 2	PUR2	00h
0363h			
0364h			
0365h			
0366h	Port Control Register	PCR	0000 0XX0b
0367h			
0368h			
0369h	$\overline{\text{NMI}}$ Digital Filter Register	NMIDF	XXXX X000b
036Ah			
036Bh			
036Ch			
036Dh			
036Eh			
036Fh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.13 SFR Information (13/16) (1)

Address	Register	Symbol	Reset Value
0370h			
0371h			
0372h			
0373h			
0374h			
0375h			
0376h			
0377h			
0378h			
0379h			
037Ah			
037Bh			
037Ch	Count Source Protection Mode Register	CSPR	00h (2)
037Dh	Watchdog Timer Refresh Register	WDTR	XXh
037Eh	Watchdog Timer Start Register	WDTS	XXh
037Fh	Watchdog Timer Control Register	WDC	00XX XXXXb
0380h			
0381h			
0382h			
0383h			
0384h			
0385h			
0386h			
0387h			
0388h			
0389h			
038Ah			
038Bh			
038Ch			
038Dh			
038Eh			
038Fh			
0390h	DMA2 Source Select Register	DM2SL	00h
0391h			
0392h	DMA3 Source Select Register	DM3SL	00h
0393h			
0394h			
0395h			
0396h			
0397h			
0398h	DMA0 Source Select Register	DM0SL	00h
0399h			
039Ah	DMA1 Source Select Register	DM1SL	00h
039Bh			
039Ch			
039Dh			
039Eh			
039Fh			

Notes: X: Undefined

1. The blank areas are reserved. No access is allowed.
2. When the CSPROINI bit in the OFS1 address is 0, the reset value is 10000000b.

Table 4.14 SFR Information (14/16) (1)

Address	Register	Symbol	Reset Value
03A0h			
03A1h			
03A2h	Open-Circuit Detection Assist Function Register	AINRST	XX00 XXXXb
03A3h			
03A4h			
03A5h			
03A6h			
03A7h			
03A8h			
03A9h			
03AAh			
03ABh			
03ACh			
03ADh			
03AEh			
03AFh			
03B0h			
03B1h			
03B2h			
03B3h			
03B4h	SFR Snoop Address Register	CRCSAR	XXXX XXXXb
03B5h			00XX XXXXb
03B6h	CRC Mode Register	CRCMR	0XXX XXX0b
03B7h			
03B8h			
03B9h			
03BAh			
03BBh			
03BCh	CRC Data Register	CRCD	XXh
03BDh			XXh
03BEh	CRC Input Register	CRCIN	XXh
03BFh			
03C0h	A/D Register 0	AD0	XXXX XXXXb
03C1h			0000 00XXb
03C2h	A/D Register 1	AD1	XXXX XXXXb
03C3h			0000 00XXb
03C4h	A/D Register 2	AD2	XXXX XXXXb
03C5h			0000 00XXb
03C6h	A/D Register 3	AD3	XXXX XXXXb
03C7h			0000 00XXb
03C8h	A/D Register 4	AD4	XXXX XXXXb
03C9h			0000 00XXb
03CAh	A/D Register 5	AD5	XXXX XXXXb
03CBh			0000 00XXb
03CCh	A/D Register 6	AD6	XXXX XXXXb
03CDh			0000 00XXb
03CEh	A/D Register 7	AD7	XXXX XXXXb
03CFh			0000 00XXb

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.15 SFR Information (15/16) (1)

Address	Register	Symbol	Reset Value
03D0h			
03D1h			
03D2h			
03D3h			
03D4h	A/D Control Register 2	ADCON2	0000 X00Xb
03D5h			
03D6h	A/D Control Register 0	ADCON0	0000 0XXXb
03D7h	A/D Control Register 1	ADCON1	0000 0000b
03D8h			
03D9h			
03DAh			
03DBh			
03DCh			
03DDh			
03DEh			
03DFh			
03E0h	Port P0 Register	P0	XXh
03E1h	Port P1 Register	P1	XXh
03E2h	Port P0 Direction Register	PD0	00h
03E3h	Port P1 Direction Register	PD1	00h
03E4h	Port P2 Register	P2	XXh
03E5h	Port P3 Register	P3	XXh
03E6h	Port P2 Direction Register	PD2	00h
03E7h	Port P3 Direction Register	PD3	00h
03E8h	Port P4 Register	P4	XXh
03E9h	Port P5 Register	P5	XXh
03EAh	Port P4 Direction Register	PD4	00h
03EBh	Port P5 Direction Register	PD5	00h
03ECh	Port P6 Register	P6	XXh
03EDh	Port P7 Register	P7	XXh
03EEh	Port P6 Direction Register	PD6	00h
03EFh	Port P7 Direction Register	PD7	00h
03F0h	Port P8 Register	P8	XXh
03F1h	Port P9 Register	P9	XXh
03F2h	Port P8 Direction Register	PD8	00h
03F3h	Port P9 Direction Register	PD9	00h
03F4h	Port P10 Register	P10	XXh
03F5h			
03F6h	Port P10 Direction Register	PD10	00h
03F7h			
03F8h			
03F9h			
03FAh			
03FBh			
03FCh			
03FDh			
03FEh			
03FFh			
0400h to D07Fh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

Table 4.16 SFR Information (16/16) (1)

Address	Register	Symbol	Reset Value
D080h			
D081h			
D082h			
D083h			
D084h			
D085h			
D086h			
D087h			
D088h			
D089h			
D08Ah			
D08Bh			
D08Ch			
D08Dh			
D08Eh			
D08Fh			
D090h			
D091h			
D092h			
D093h			
D094h			
D095h			
D096h			
D097h			
D098h			
D099h			
D09Ah			
D09Bh			
D09Ch			
D09Dh			
D09Eh			
D09Fh			

Note: X: Undefined

1. The blank areas are reserved. No access is allowed.

4.2 Notes on SFRs

4.2.1 Register Settings

Table 4.17 lists Registers with Write-Only Bits and registers whose function differs between reading and writing. Set these registers with immediate values. Do not use read-modify-write instructions. When establishing the next value by altering the existing value, write the existing value to the RAM as well as to the register. Transfer the next value to the register after making changes in the RAM.

Table 4.17 Registers with Write-Only Bits

Register	Symbol	Address
Watchdog Timer Refresh Register	WDTR	037Dh
Watchdog Timer Start Register	WDTS	037Eh
Timer A0 Register	TA0	0327h to 0326h
Timer A1 Register	TA1	0329h to 0328h
Timer A2 Register	TA2	032Bh to 032Ah
Timer A3 Register	TA3	032Dh to 032Ch
Timer A4 Register	TA4	032Fh to 032Eh
Timer A1-1 Register	TA11	0303h to 0302h
Timer A2-1 Register	TA21	0305h to 0304h
Timer A4-1 Register	TA41	0307h to 0306h
UART0 Bit Rate Register	U0BRG	0249h
UART1 Bit Rate Register	U1BRG	0259h
UART2 Bit Rate Register	U2BRG	0269h
UART5 Bit Rate Register	U5BRG	0289h
UART6 Bit Rate Register	U6BRG	0299h
UART7 Bit Rate Register	U7BRG	02A9h
UART0 Transmit Buffer Register	U0TB	024Bh to 024Ah
UART1 Transmit Buffer Register	U1TB	025Bh to 025Ah
UART2 Transmit Buffer Register	U2TB	026Bh to 026Ah
UART5 Transmit Buffer Register	U5TB	028Bh to 028Ah
UART6 Transmit Buffer Register	U6TB	029Bh to 029Ah
UART7 Transmit Buffer Register	U7TB	02ABh to 02AAh
SI/O3 Bit Rate Register	S3BRG	0273h
SI/O4 Bit Rate Register	S4BRG	0277h
I2C0 Control Register 1	S3D0	02B6h
I2C0 Status Register 0	S10	02B8h

Table 4.18 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, DEC, DIV, DIVU, DIVX, EXTS, INC, MUL, MULU, NEG, SBB, and SUB
Decimal operation	DADC, DADD, DSBB, and DSUB
Logical operation	AND, NOT, OR, and XOR
Jump	ADJNZ, SBJNZ

5. Protection

5.1 Introduction

In the event that a program runs out of control, this function protects the important registers listed below so that they will not be rewritten easily.

5.2 Register

Table 5.1 Registers

Address	Register	Symbol	Reset Value
000Ah	Protect Register	PRCR	00h

5.2.1 Protect Register (PRCR)

Protect Register

b7	b6	b5	b4	b3	b2	b1	b0
0		0	0				

Symbol	Address	After Reset
PRCR	000Ah	00h

Bit Symbol	Bit Name	Function	RW
PRC0	Protect bit 0	Enable write to registers CM0, CM1, CM2, PLC0, PLCF, PCLKR, SCM0, and PCLKSTP1 0: Write protected 1: Write enabled	RW
PRC1	Protect bit 1	Enable write to registers PM0, PM1, and PM2 0: Write protected 1: Write enabled	RW
PRC2	Protect bit 2	Enable write to registers PD9, S3C, and S4C 0: Write protected 1: Write enabled	RW
PRC3	Protect bit 3	Enable write to registers VCR2 and VW0C 0: Write protected 1: Write enabled	RW
— (b5-b4)	Reserved bits	Set to 0.	RW
PRC6	Protect bit 6	Enable write to the PRG2C register 0: Write protected 1: Write enabled	RW
— (b7)	Reserved bit	Set to 0.	RW

PRC6, PRC3, PRC1, PRC0 (Protect bits 6, 3, 1, 0) (b6, b3, b1, b0)

When setting bits PRC6, PRC3, PRC1, and PRC0 to 1 (write enabled), the bits remain 1 (write enabled). To change registers protected by these bits, follow these steps:

(1) Set the PRCi bit to 1. (i = 0, 1, 3, 6)
(2) Write to the register protected by the PRCi bit.
(3) Set the PRCi bit to 0 (write protected).

PRC2 (Protect bit 2) (b2)

After setting the PRC2 bit to 1 (write enabled), by writing to a given SFR, the PRC2 bit becomes 0. Change the registers protected by the PRC2 bit in the next instruction after setting the PRC2 bit to 1. The steps are shown below. Make sure there are no interrupts or DMA transfers between steps (1) and (2).

(1) Set the PRC2 bit to 1.
(2) Write to the register protected by the PRC2 bit.

5.3 Notes on Protection

After setting the PRC2 bit to 1 (write enabled), by writing to a given SFR, the PRC2 bit becomes 0 (write disabled). Change the registers protected by the PRC2 bit in the next instruction after setting the PRC2 bit to 1. Make sure there are no interrupts or DMA transfers between the instruction that sets the PRC2 bit to 1 and the next instruction.

6. Resets

6.1 Introduction

The following resets can be used to reset the MCU: hardware reset, voltage monitor 0 reset, oscillator stop detect reset, watchdog timer reset, and software reset.
Table 6.1 lists the Types of Resets, Figure 6.1 shows the Reset Circuit Block Diagram, and Table 6.3 lists the I/O Pins.

Table 6.1 Types of Resets

Reset Name	Trigger	Registers and Bits Not to Reset
Hardware reset	A low-level signal is applied to the $\overline{\text{RESET}}$ pin.	(A)
Voltage monitor 0 reset	The drop in voltage on VCC1 (reference voltage: Vdet0)	N/A
Oscillator stop detect reset	A stop in the main clock oscillator is detected.	(A) (B) (C)
Watchdog timer reset	The watchdog timer underflows.	(A) (B)
Software reset	Setting the PM03 bit in the PM0 register to 1	(A) (B)

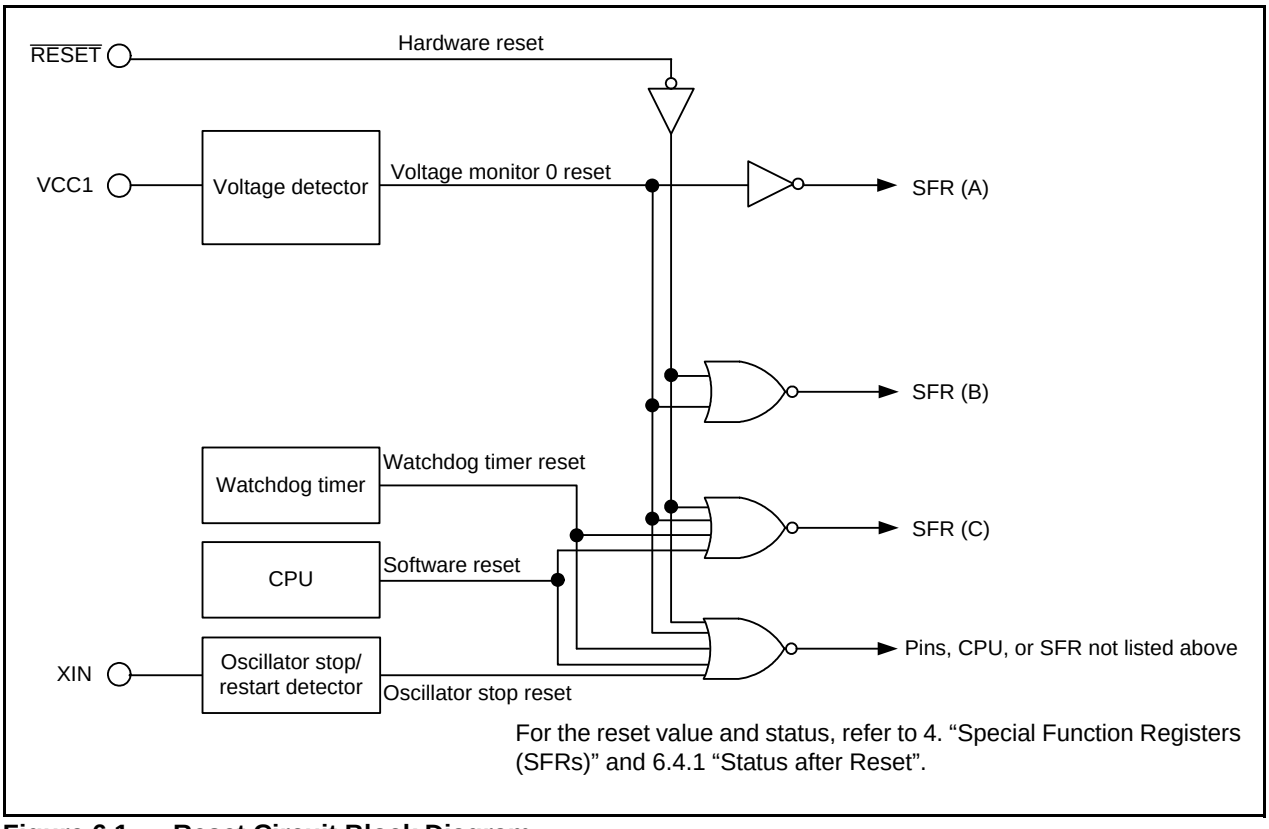


Figure 6.1 Reset Circuit Block Diagram

Table 6.2 Classification of SFRs Which are Reset

SFR	Register and Bit
SFR(A)	Bits OSDR and CWR in the RSTFR register CWR bit in the RSTFR register Registers VCR2 and VW0C VW2C3 bit in the VW2C register Bits PM00 and PM01 in the PM0 register
SFR(B)	—
SFR(C)	Bits CM20, CM21, and CM27 in the CM2 register

Table 6.3 I/O Pins

Pin	I/O	Function
$\overline{\text{RESET}}$	Input	Hardware reset input
VCC1	Input	Power input. The voltage monitor 0 reset is generated by monitoring VCC1.
XIN	Input	Main clock input. The oscillator stop detect reset is generated by monitoring the main clock.

6.2 Registers

Refer to 7. “Voltage Detector” for registers used with the voltage monitor 0 reset. Refer to 14. “Watchdog Timer” for registers used with the watchdog timer reset. Refer to 8.7 “Oscillator Stop/Restart Detect Function” for registers used with the oscillator stop detect reset.

Table 6.4 Registers

Address	Register	Symbol	Reset Value
0004h	Processor Mode Register 0	PM0	0000 0000b
0018h	Reset Source Determine Register	RSTFR	XX00 001Xb (1)

Note:
1. Refer to 6.2.2 “Reset Source Determine Register (RSTFR)”.

6.2.1 Processor Mode Register 0 (PM0)

Processor Mode Register 0

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

Symbol	Address	After Reset
PM0	0004h	0000 0000b

Bit Symbol	Bit Name	Function	RW
PM00	Processor mode bit	b1 b0 0 0: Single-chip mode 0 1: Do not set 1 0: Do not set 1 1: Do not set	RW
PM01			RW
— (b2)	Reserved bit	Set to 0.	RW
PM03	Software reset bit	Setting this bit to 1 resets the MCU. When read, the read value is 0.	RW
— (b7-b4)	Reserved bits	Set to 0.	RW

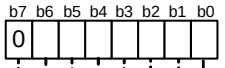
Write to this register after setting the PRC1 bit in the PRCR register to 1 (write enabled).

PM03 (Software reset bit) (b3)

A software reset is generated by setting the PM03 bit to 1.

6.2.2 Reset Source Determine Register (RSTFR)

Reset Source Determine Register



Symbol: RSTFR | Address: 0018h | After Reset: See Table 6.5.

Bit Symbol	Bit Name	Function	RW
CWR	Cold start-up/warm start-up discrimination flag	0: Cold start 1: Warm start	RW
HWR	Hardware reset detection flag	0: Not detected 1: Detected	RO
SWR	Software reset detection flag	0: Not detected 1: Detected	RO
WDR	Watchdog timer reset detect flag	0: Not detected 1: Detected	RO
— (b5-b4)	Reserved bits	The read value is 0.	RO
OSDR	Oscillator stop detect reset detect flag	0: Not detected 1: Detected	RW
— (b7)	Reserved bit	If necessary, set to 0. When read, the read value is undefined.	RW

Table 6.5 RSTFR Register Reset Value

Reset	Bits in the RSTFR Register						
	OSDR	LVD2R	LVD1R	WDR	SWR	HWR	CWR
Hardware reset	No change	0	0	0	0	1	No change
Voltage monitor 0 reset	0	0	0	0	0	0	0
Oscillator stop detect reset	1	0	0	0	0	0	No change
Watchdog timer reset	0	0	0	1	0	0	No change
Software reset	0	0	0	0	1	0	No change

CWR (Cold/warm start discrimination flag) (b0)

Conditions to become 0:

- Power-on

Condition to become 1:

- Setting this bit to 1

OSDR (Oscillator stop detect reset detection flag) (b6)

Conditions to become 0:

- Power-on
- Setting this bit to 0

This bit will not become 1 even when written to 1.

6.3 Optional Function Select Area

In the optional function select area, the MCU state after reset and the function to prevent rewrite in parallel I/O mode are selected.
The optional function select area is not an SFR, and therefore cannot be rewritten by a program. Set an appropriate value when writing a program to flash memory. The entire optional function select area becomes FFh when the block including the optional function select area is erased.
In blank products, the OFS1 address value is FFh when shipped. After a value is written by the user, this register takes on the written value. In programmed products, the OFS1 address is the value set in the user program prior to shipping.

6.3.1 Optional Function Select Address 1 (OFS1)

Optional Function Select Address 1

b7	b6	b5	b4	b3	b2	b1	b0
			1			1	

Symbol: OFS1
Address: FFFFFh

Bit Symbol	Bit Name	Function
WDTON	Watchdog timer start select bit	0: Watchdog timer starts automatically after reset 1: Watchdog timer is stopped after reset
— (b1)	Reserved bit	Set to 1.
ROMCR	ROM code protect cancel bit	0: ROM code protection cancelled 1: ROMCP1 bit enabled
ROMCP1	ROM code protect bit	0: ROM code protection enabled 1: ROM code protection disabled
— (b4)	Reserved bit	Set to 1.
VDSEL1	Vdet0 select bit 1	0: Vdet0_2 1: Vdet0_0
LVDAS	Voltage detector 0 start bit	0: Voltage monitor 0 reset enabled after hardware reset 1: Voltage monitor 0 reset disabled after hardware reset
CSPROINI	After-reset count source protection mode select bit	0: Count source protection mode enabled after reset 1: Count source protection mode disabled after reset

WDTON (Watchdog timer start select bit) (b0)
CSPROINI (After-reset count source protection mode select bit) (b7)

These bits select the state of the watchdog timer after reset.
Set the WDTON bit to 0 (watchdog timer starts automatically after reset) when setting the CSPROINI bit to 0 (count source protection mode enabled after reset).
Refer to 14. “Watchdog Timer” for details on the watchdog timer and count source protection mode.

ROMCR (ROM code protect cancel bit) (b2)
ROMCP1 (ROM code protect bit) (b3)

These bits prevent the flash memory from being read or changed in parallel I/O mode.

Table 6.6 ROM Code Protection

Bit Setting		ROM Code Protection
ROMCR bit	ROMCP1 bit	
0	0	Cancelled
0	1	
1	0	Enabled
1	1	Cancelled

VDSEL1 (Vdet0 select bit 1) (b5)

Set this bit to 0 (Vdet0 is 2.85 V) when using the voltage monitor 0 reset. Refer to 6.4.7 “Cold/Warm Start Discrimination”.
This bit is disabled in boot mode.

LVDAS (Voltage detector 0 start bit) (b6)

This bit is disabled in boot mode.

6.4 Operations

6.4.1 Status after Reset

The status of SFRs after reset depends on the reset type. See the Reset Value chapter in 4. “Special Function Registers (SFRs)”. Table 6.7 lists the Pin Status When $\overline{\text{RESET}}$ Pin Level is Low, Figure 6.2 shows the CPU Register Status after Reset, and Figure 6.3 shows the Reset Sequence.

Table 6.7 Pin Status When $\overline{\text{RESET}}$ Pin Level is Low

Pin Name	Status (1)	
	Single-chip mode (CNVSS = VSS)	Boot mode (CNVSS = VCC1)
P0	Input port	Input port
P1	Input port	Input port
P2, P3, P4_0~P4_3	Input port	Input port
P4_4	Input port	Input port
P4_5~P4_7	Input port	Input port
P6, P7, P8, P9, P10	Input port	Input port

Note:

1. These two columns show the valid pin state when the internal power supply voltage has stabilized after power on. The pin status is undefined until td(P-R) has elapsed after power-on.

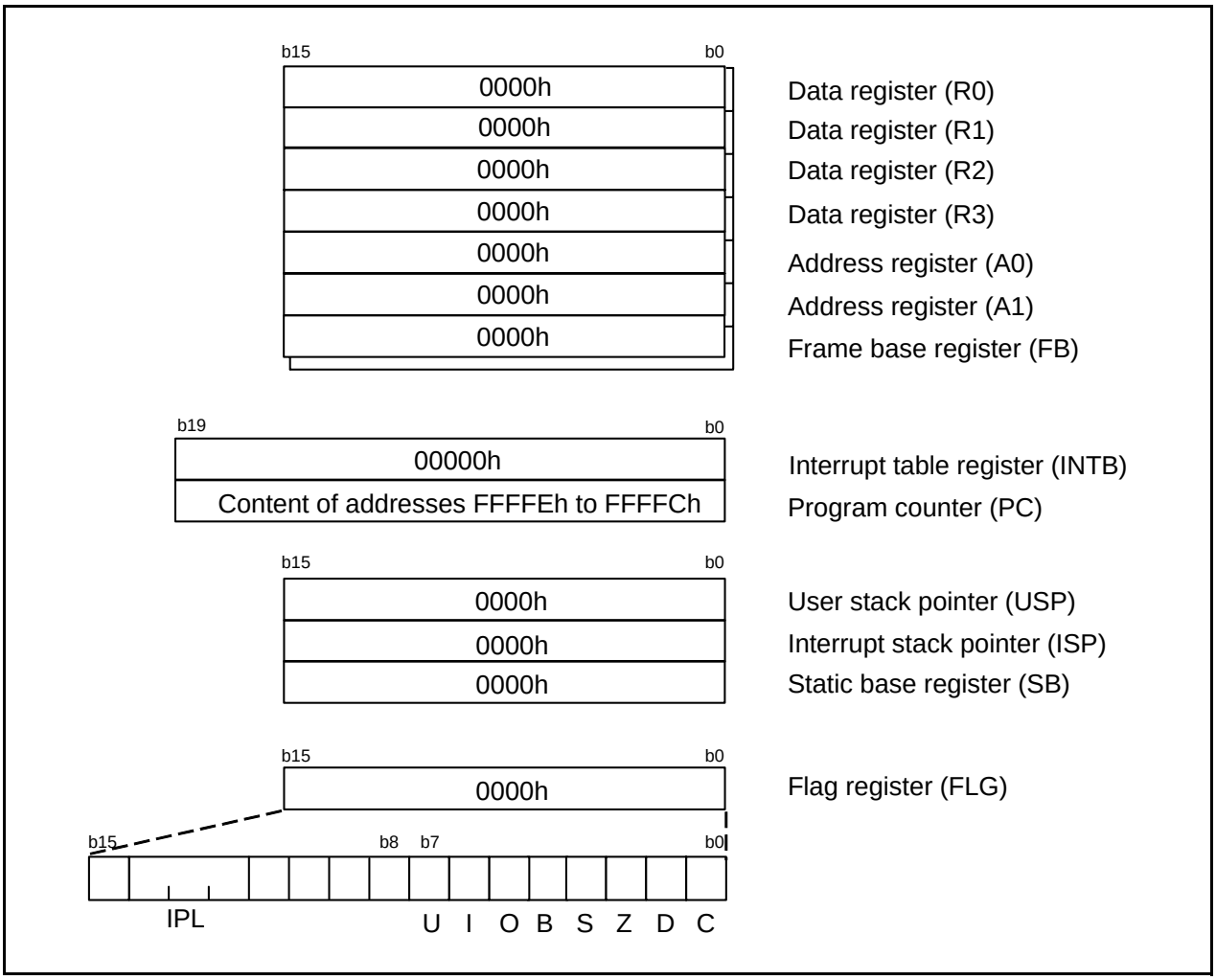


Figure 6.2 CPU Register Status after Reset

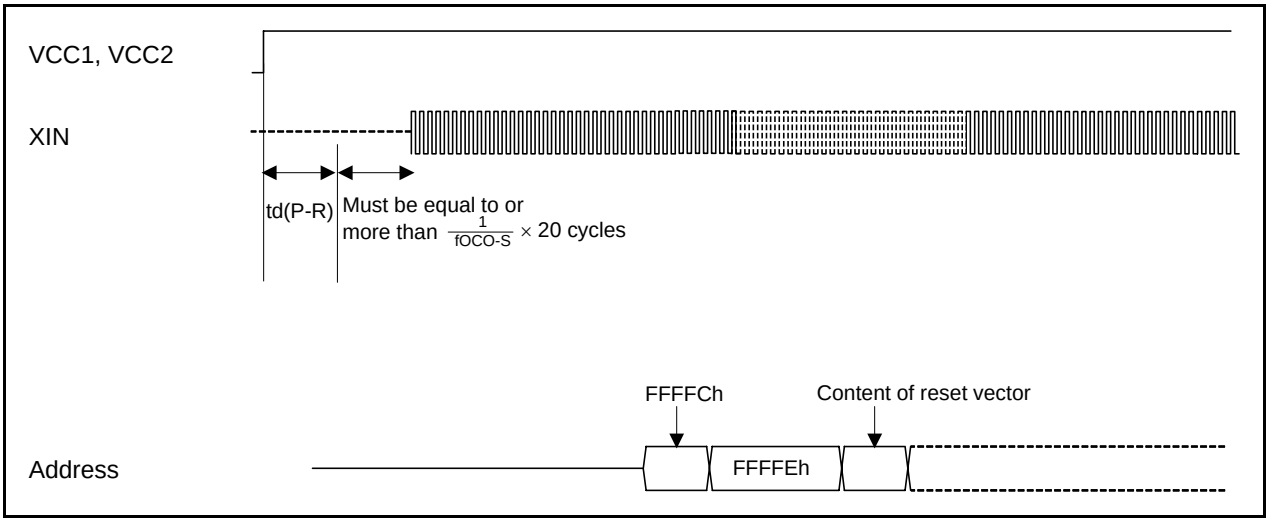


Figure 6.3 Reset Sequence

6.4.2 Hardware Reset

This reset is triggered by the $\overline{\text{RESET}}$ pin. When the power supply voltage meets the recommended operating conditions, the MCU resets the pins, CPU, and SFRs when a low-level signal is applied to the $\overline{\text{RESET}}$ pin.

When changing the signal applied to the $\overline{\text{RESET}}$ pin from low to high, the MCU executes the program at the address indicated by the reset vector. fOCO-S divided by 8 is automatically selected as the CPU clock after reset.

The HWR bit in the RSTFR register becomes 1 (hardware reset detected) after hardware reset. Refer to 4. “Special Function Registers (SFRs)” for the rest of the SFR states after reset.

The internal RAM is not reset. When a low-level signal is applied to the $\overline{\text{RESET}}$ pin while writing data to the internal RAM, the internal RAM becomes undefined.

The procedures for generating a hardware reset are as follows:

When the power supply is stable

(1) Apply a low-level signal to the $\overline{\text{RESET}}$ pin.
(2) Wait for tw(RSTL).
(3) Apply a high-level signal to the $\overline{\text{RESET}}$ pin.

When the power is turned on

(1) Apply a low-level signal to the $\overline{\text{RESET}}$ pin.
(2) Raise the power supply voltage to the recommended operating level.
(3) Wait for td(P-R) until the internal voltage stabilizes.
(4) Wait for $\frac{1}{fOCO-S} \times 20$ cycles.
(5) Apply a high-level signal to the $\overline{\text{RESET}}$ pin.

Figure 6.4 shows an Reset Circuit Example.

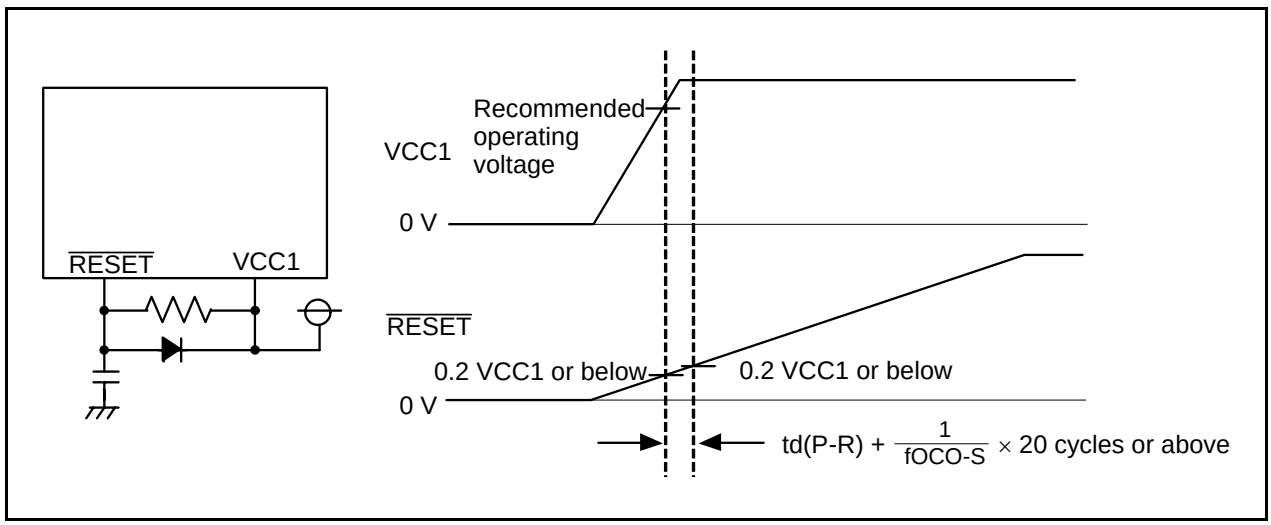


Figure 6.4 Reset Circuit Example

6.4.3 Voltage Monitor 0 Reset

This reset is triggered by the MCU's on-chip voltage detector 0. The voltage detector 0 monitors the voltage applied to the VCC1 pin (Vdet0).

The MCU resets the pins, CPU, and SFRs when the voltage applied to the VCC1 pin drops to Vdet0 or below.

Then, the fOCO-S count starts when the voltage applied to the VCC1 pin rises to Vdet0 or above. The internal reset signal becomes high after 32 cycles of fOCO-S, and then the MCU executes the program at the address indicated by the reset vector. fOCO-S divided by 8 is automatically selected as the CPU clock after reset.

The CWR bit in the RSTFR register becomes 0 (cold start) after voltage monitor 0 reset. Refer to 4. "Special Function Registers (SFRs)" for the remaining SFR states after reset.

The internal RAM is not reset. When the voltage applied to the VCC1 pin drops to Vdet0 or below while writing data to the internal RAM, the internal RAM becomes undefined.

Refer to 7. "Voltage Detector" for details of the voltage monitor 0 reset.

6.4.4 Oscillator Stop Detect Reset

The MCU resets and stops the pins, CPU, and SFRs when the CM27 bit in the CM2 register is 0 (reset when oscillator stop detected), if it detects that the main clock oscillator has stopped.

The OSDR bit in the RSTFR register becomes 1 (oscillator stop detect reset detected) after oscillator stop detect reset. Some SFRs are not reset at oscillator stop detect reset. Refer to 4. "Special Function Registers (SFRs)" for details.

The internal RAM is not reset. When the main clock oscillator stop is detected while writing data to the internal RAM, the internal RAM becomes undefined.

Oscillator stop detect reset is canceled by hardware reset or voltage monitor 0 reset.

Refer to 8.7 "Oscillator Stop/Restart Detect Function" for details.

6.4.5 Watchdog Timer Reset

The MCU resets the pins, CPU, and SFRs when the PM12 bit in the PM1 register is 1 (reset when watchdog timer underflows) and the watchdog timer underflows. Then the MCU executes the program at the address determined by the reset vector. fOCO-S divided by 8 is automatically selected as the CPU clock after reset.

The WDR bit in the RSTFR register becomes 1 (watchdog timer reset detected) after watchdog timer reset. Some SFRs are not reset at watchdog timer reset. Refer to 4. "Special Function Registers (SFRs)" for details.

The internal RAM is not reset. When the watchdog timer underflows while writing data to the internal RAM, the internal RAM becomes undefined.

Refer to 14. "Watchdog Timer" for details.

6.4.6 Software Reset

The MCU resets the pins, CPU, and SFRs when the PM03 bit in the PM0 register is 1 (MCU reset). Then the MCU executes the program at the address determined by the reset vector. fOCO-S divided by 8 is automatically selected as the CPU clock after reset.

The SWR bit in the RSTFR register becomes 1 (software reset detected) after software reset. Some SFRs are not reset at software reset. Refer to 4. "Special Function Registers (SFRs)" for details.

The internal RAM is not reset.

6.4.7 Cold/Warm Start Discrimination

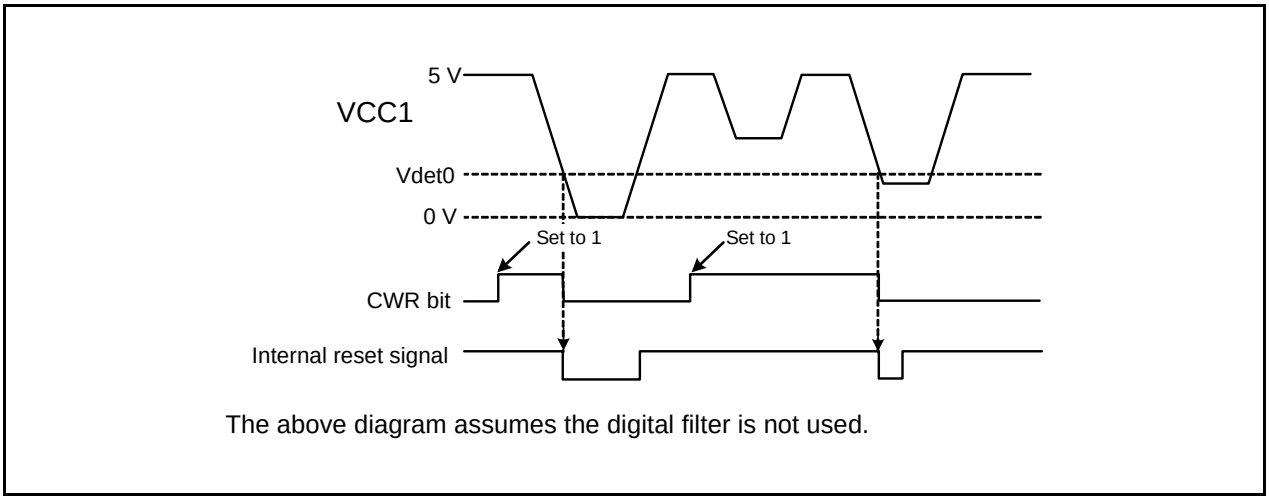
The cold/warm start discrimination detects whether or not voltage applied to the VCC1 pin drops to the RAM hold voltage or below. The reference voltage is Vdet0. Therefore, the voltage monitor 0 reset is used for cold/warm start discrimination. Follow 7.4.1.1 “Voltage Monitor 0 Reset” to set the bits related to the voltage monitor 0 reset.

The CWR bit in the RSTFR register is 0 (cold start) when power is turned on. The CWR bit also becomes 0 after voltage monitor 0 reset. The CWR bit becomes 1 (warm start) by writing 1, and remains unchanged at hardware reset, oscillator stop detect reset, watchdog timer reset, or software reset.

In the cold/warm start discrimination, the Vdet0 level can be selected by setting the VDSEL1 bit in the OFS1 address.

- When voltage monitor 0 reset is used
  Set the VDSEL1 bit to 0 (Vdet0 = 2.85 V (Vdet0_2)).
- When voltage monitor 0 reset is not required as the user system
  Set the VDSEL1 bit to 1 (Vdet0_0). In this case, voltage monitor 0 reset and its cancellation are based on Vdet0_0. Therefore, execute hardware reset after cancelling the voltage monitor 0 reset.

Figure 6.5 shows the Cold/Warm Start Discrimination Example.



The above diagram assumes the digital filter is not used.

Figure 6.5 Cold/Warm Start Discrimination Example

6.5 Notes on Resets

6.5.1 Power Supply Rising Gradient

When supplying power to the MCU, make sure that the power supply voltage applied to the VCC1 pin meets the SVCC conditions.

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
SVCC	Power supply VCC1 rising gradient (Voltage range: 0 V to 2.0 V)	0.05			V/ms

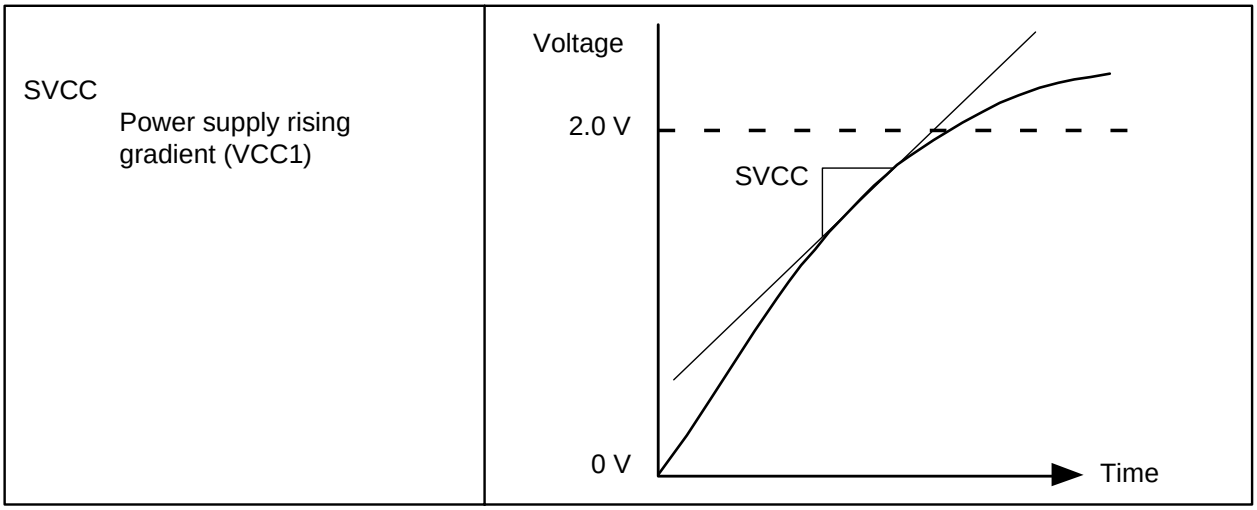


Figure 6.6 SVCC Timing

6.5.2 OSDR Bit (Oscillation Stop Detect Reset Detection Flag)

When an oscillation stop detect reset is generated, the MCU is reset and then stopped. This state is canceled by hardware reset or voltage monitor 0 reset.

Note that the OSDR bit in the RSTFR register value is not affected by a hardware reset, but becomes 0 (not detected) from a voltage monitor 0 reset.

6.5.3 Hardware Reset When VCC1 < Vdet0

If a hardware reset is executed when the LVDAS bit in the OFS1 address is 0 (voltage monitor 0 reset enabled after hardware reset) and VCC1 < Vdet0, the MCU executes the program at the address indicated by the reset vector when changing the signal applied to the RESET pin from low to high. A voltage monitor 0 reset is not generated.

6.5.4 Starting PLL Clock Oscillation

6.5.4.1 When Using Voltage Detector 0

Do not change the PLC07 bit in the PLC0 register from 0 to 1 when the VC25 bit in the VCR2 register is 1.

To change the PLC07 bit from 0 to 1 while using a voltage detector, use the following procedure:

(1) Set the VC25 bit to 0 (voltage detector off).
(2) Change the PLC07 bit from 0 to 1.
(3) Wait for 1 ms.
(4) Set the VC25 bit to 1 (voltage detector on).

6.5.4.2 When Using 125 kHz On-chip Oscillator Mode or 125 kHz On-chip Oscillator Low Power Mode

Change the PLC07 bit in the PLC0 register from 0 to 1 while dividing the clock by 8 or 16 (selectable by setting the CM06 bit in the CM0 register and bits CM17 to CM16 in the CM1 register).

6.5.4.3 Count Source for Timer A and Timer B

When using the PLL clock, do not use fOCO-S as the count source for timer A and timer B.

6.5.4.4 When Using fOCO-S as the Count Source for the Watchdog Timer

Change the PLC07 bit in the PLC0 register from 0 to 1 using the following procedure:

(1) Write 00h to the WDTR register, then write FFh (watchdog timer refresh).
(2) Change the PLC07 bit from 0 to 1.
(3) Wait for 1 ms.
(4) Write 00h to the WDTR register, then write FFh (watchdog timer refresh).

7. Voltage Detector

7.1 Introduction

The voltage detector monitors the voltage applied to the VCC1 pin. Voltage monitor 0 reset can be used by user program.

Table 7.1 lists the Voltage Detector Specifications and Figure 7.1 shows Voltage Detector Block Diagram.

Table 7.1 Voltage Detector Specifications

Item		Voltage Detector 0
VCC1 monitor	Voltage to monitor	Vdet0
	Detection target	Whether rises through or falls through Vdet0
	Voltage to detect	Selectable from two levels in the OFS1 address
	Software monitor	None
Process when voltage is detected	Reset	Voltage monitor 0 reset
		Reset when Vdet0 > VCC1; restart CPU operation when VCC1 > Vdet0
	Interrupt	None

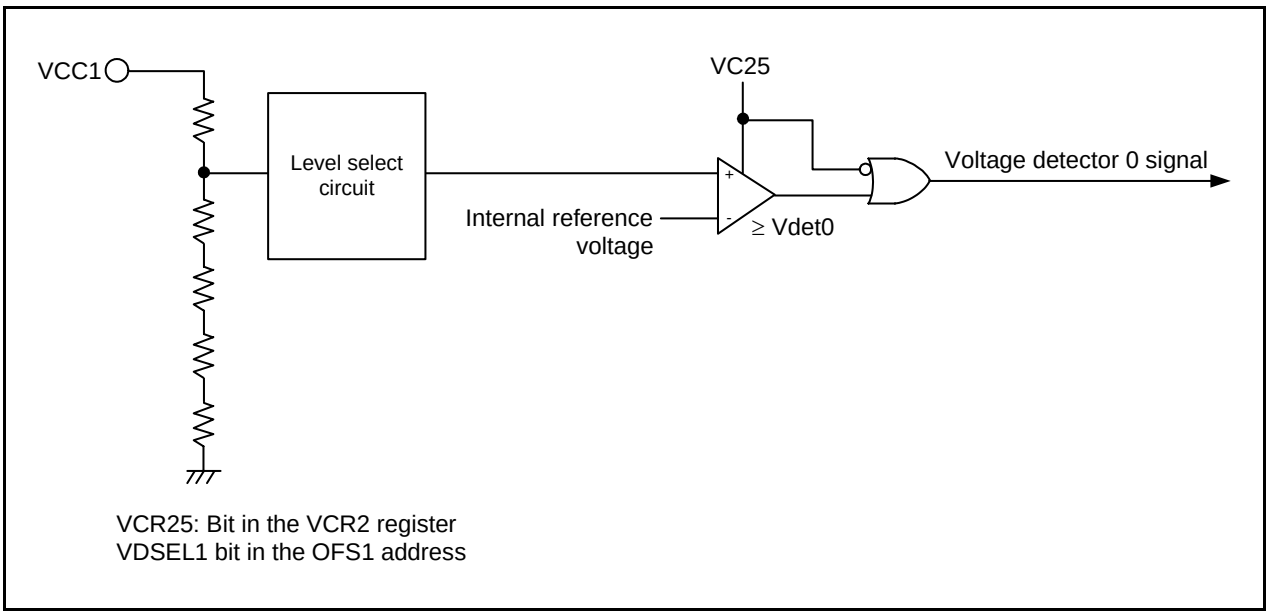


Figure 7.1 Voltage Detector Block Diagram

7.2 Registers

Table 7.2 shows the registers of the voltage detector. The reset value shows the values after hardware reset.

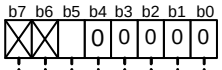
Refer to the each register explanation for details.

Table 7.2 Registers

Address	Register	Symbol	Reset Value
001Ah	Voltage Detector Operation Enable Register	VCR2	0000 0000b 001X 0000b
002Ah	Voltage Monitor 0 Control Register	VW0C	1000 XX10b 1100 XX11b

7.2.1 Voltage Detector Operation Enable Register (VCR2)

Voltage Detector Operation Enable Register



Symbol	Address	After Reset
VCR2	001Ah	0000 0000b (1) 0010 0000b (2)

Bit Symbol	Bit Name	Function	RW
— (b4-b0)	Reserved bits	Set to 0.	RW
VC25	Voltage detector 0 enable bit	0: Voltage detector 0 disabled 1: Voltage detector 0 enabled	RW
— (b7-b6)	Reserved bits	Set to 0.	RW

Notes:
1. This is the reset value when the LVDAS bit of address OFS1 is 1 during hardware reset
2. This is the reset value after voltage monitor 0 reset or when the LVDAS bit of address OFS1 is 0 during hardware reset.

Set the PRC3 bit in the PRCR register to 1 (write enabled) before rewriting this register.

This register does not change at oscillator stop detect reset, watchdog timer reset, or software reset.

VC25 (Voltage detector 0 enable bit) (b5)

To use voltage monitor 0 reset, set the VC25 bit to 1 (voltage detector 0 enabled). After changing the VC25 bit to 1, the detector starts operating when the td(E-A) elapses.

7.2.2 Voltage Monitor 0 Control Register (VW0C)

Voltage Monitor 0 Control Register

b7	b6	b5	b4	b3	b2	b1	b0
1	1				0		

Symbol	Address	After Reset
VW0C	002Ah	1000 XX10b (1) 1100 XX11b (2)

Bit Symbol	Bit Name	Function	RW
VW0C0	Voltage monitor 0 reset enable bit	0: Disabled 1: Enabled	RW
— (b1)	Reserved bit	Set to 1.	RW
— (b2)	Reserved bit	Set to 0. When read, the read value is undefined.	RW
— (b3)	Reserved bit	When read, the read value is undefined.	RO
— (b5-b4)	Reserved bits	Set to 0.	RW
— (b7-b6)	Reserved bits	Set to 1.	RW

Notes:
1. This is the reset value when the LVDAS bit of address OFS1 is 1 during hardware reset.
2. This is the reset value after voltage monitor 0 reset or when the LVDAS bit of address OFS1 is 0 during hardware reset.

Set the PRC3 bit in the PRCR register to 1 (write enabled) before rewriting to this register.
This register does not change at oscillator stop detect reset, watchdog timer reset, or software reset.

VW0C0 (Voltage monitor 0 reset enable bit) (b0)

The VW0C0 bit is enabled when the VC25 bit in the VCR2 register is 1 (voltage detector 0 enabled). Set the VW0C0 bit to 0 (disabled) when the VC25 bit is 0 (voltage detector 0 disabled).

Bit 6

When the LVDAS bit in the OFS1 address is 1, this bit becomes 0 after hardware reset. When using voltage monitor 0 reset, set this bit to 1.

7.3 Optional Function Select Area

In the optional function select area, the MCU state after reset and the function to prevent rewrite in parallel I/O mode are selected.
The optional function select area is not an SFR, and therefore cannot be rewritten by a program. Set an appropriate value when writing a program to flash memory. The entire optional function select area becomes FFh when the block including the optional function select area is erased.
In blank products, the OFS1 address value is FFh when shipped. After a value is written by the user, this register takes on the written value.
In programmed products, the OFS1 address is the value set in the user program prior to shipping.

7.3.1 Optional Function Select Address 1 (OFS1)

Optional Function Select Address 1

b7	b6	b5	b4	b3	b2	b1	b0
			1			1	

Symbol: OFS1
Address: FFFFFh

Bit Symbol	Bit Name	Function
WDTON	Watchdog timer start select bit	0: Watchdog timer starts automatically after reset 1: Watchdog timer is stopped after reset
— (b1)	Reserved bit	Set to 1.
ROMCR	ROM code protect cancel bit	0: ROM code protection cancelled 1: ROMCP1 bit enabled
ROMCP1	ROM code protect bit	0: ROM code protection enabled 1: ROM code protection disabled
— (b4)	Reserved bit	Set to 1.
VDSEL1	Vdet0 select bit 1	0: Vdet0_2 1: Vdet0_0
LVDAS	Voltage detector 0 start bit	0: Voltage monitor 0 reset enabled after hardware reset 1: Voltage monitor 0 reset disabled after hardware reset
CSPROINI	After-reset count source protection mode select bit	0: Count source protection mode enabled after reset 1: Count source protection mode disabled after reset

VDSEL1 (Vdet0 select bit 1) (b5)

The Vdet0 level used in voltage detector 0 is selectable. Voltage detector 0 operates based on Vdet0. Set the VDSEL1 bit to 0 (Vdet0 is 2.85 V) when using voltage monitor 0 reset. Refer to 6.4.7 “Cold/Warm Start Discrimination”.
This bit is disabled in boot mode.

LVDAS (Voltage detector 0 start bit) (b6)

This bit is disabled in boot mode.

7.4 Operations

7.4.1 Voltage Detector 0

When the VC25 bit in the VCR2 register is 1 (voltage detector 0 enabled), voltage detector 0 monitors the voltage applied to the VCC1 pin and detects whether the voltage rises through or falls through Vdet0. The Vdet0 level can be selected by the VDSEL1 bit in the OFS1 address.

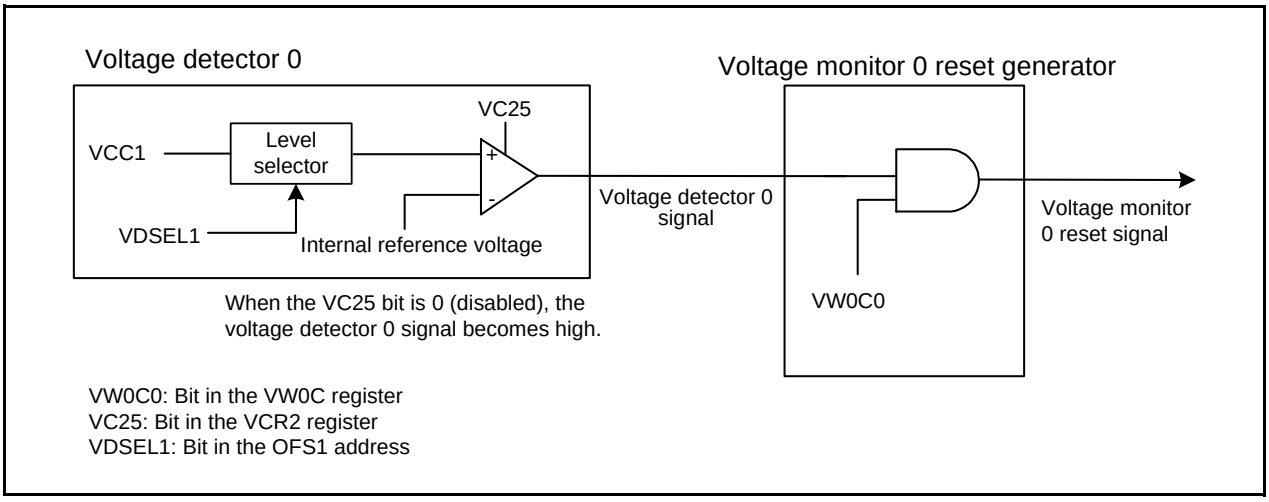


Figure 7.2 Voltage Monitor 0 Reset Generator Block Diagram

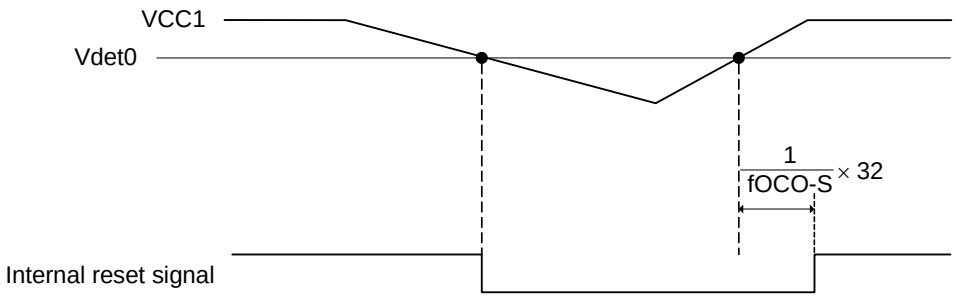
7.4.1.1 Voltage Monitor 0 Reset

When using voltage monitor 0 reset, set the VDSEL1 bit in the OFS1 address to 0 (Vdet0_2).
When the LVDAS bit in the OFS1 address is 1 (voltage monitor 0 reset disabled after hardware reset), set the related bits according to the procedure listed in Table 7.3. When the LVDAS bit in the OFS1 address is 0 (voltage monitor 0 reset enabled after hardware reset), the procedure listed in Table 7.3 is unnecessary.

Table 7.3 Procedures for Setting Voltage Monitor 0 Reset Related Bits

Step	Processing
1	Set the VC25 bit in the VCR2 register to 1 (voltage detector 0 enabled).
2	Wait for td(E-A).
3	Set the VW0C0 bit in the VW0C register to 1 (voltage monitor 0 reset enabled).

When voltage monitor 0 reset is generated, the CWR bit in the RSTFR register becomes 0 (cold start). Refer to 6.4.3 “Voltage Monitor 0 Reset” for status after reset.
Figure 7.3 shows Voltage Monitor 0 Reset Operation Example.



The above diagram assumes the following:
- The VC25 bit in the VCR2 register is 1 (voltage detector 0 enabled).
- The VW0C0 bit in the VW0C register is 1 (voltage monitor 0 reset enabled).

The pins, CPU, and SFRs are initialized when the internal reset signal goes low.
The MCU executes the program at the address indicated by the reset vector when the internal reset signal changes from low to high.
Refer to 4. “Special Function Registers (SFRs)” for the SFR status after reset.

Note:
1. Make sure that VCC1 does not drop to recommended operating condition VCC1 during sampling time.

Figure 7.3 Voltage Monitor 0 Reset Operation Example

8. Clock Generator

8.1 Introduction

The clock generator generates operating clocks for the CPU and peripheral functions. Four circuits are incorporated to generate the system clock signals.

- Main clock oscillation circuit
- PLL frequency synthesizer
- 125 kHz on-chip oscillator
- Sub clock oscillation circuit

Table 8.1 lists Clock Generator Specifications and Figure 8.1 shows System Clock Generator.

Table 8.1 Clock Generator Specifications

Item	Main Clock Oscillation Circuit	PLL Frequency Synthesizer	125 kHz On-Chip Oscillator	Sub Clock Oscillation Circuit
Application	• CPU clock source • Peripheral function clock source	• CPU clock source • Peripheral function clock source	• CPU clock source • Peripheral function clock source • CPU and peripheral function clock sources when the main clock stops oscillating • Watchdog timer count source when the CPU clock is stopped	• CPU clock source • Peripheral function clock source
Clock frequency	15.36 MHz	CPU: 11.52 to 30.72 MHz PLC: 46.08 MHz	Approx. 125 kHz	32.768 kHz
Connectable oscillators	Crystal (2)	— (1)	—	Crystal
Pins connecting to oscillator	XIN, XOUT	— (1)	—	XCIN, XCOUT
Oscillator start/stop function	Enabled	Enabled	Enabled	Enabled
Oscillator status after reset	Oscillating	Stopped	Oscillating	Stopped

Notes:
1. The PLL frequency synthesizer uses the main clock oscillation circuit as a reference clock source. The items above are based on the main clock oscillation circuit.
2. Use 15.36 MHz (with an accuracy of less than ±75 ppm) for the crystal.

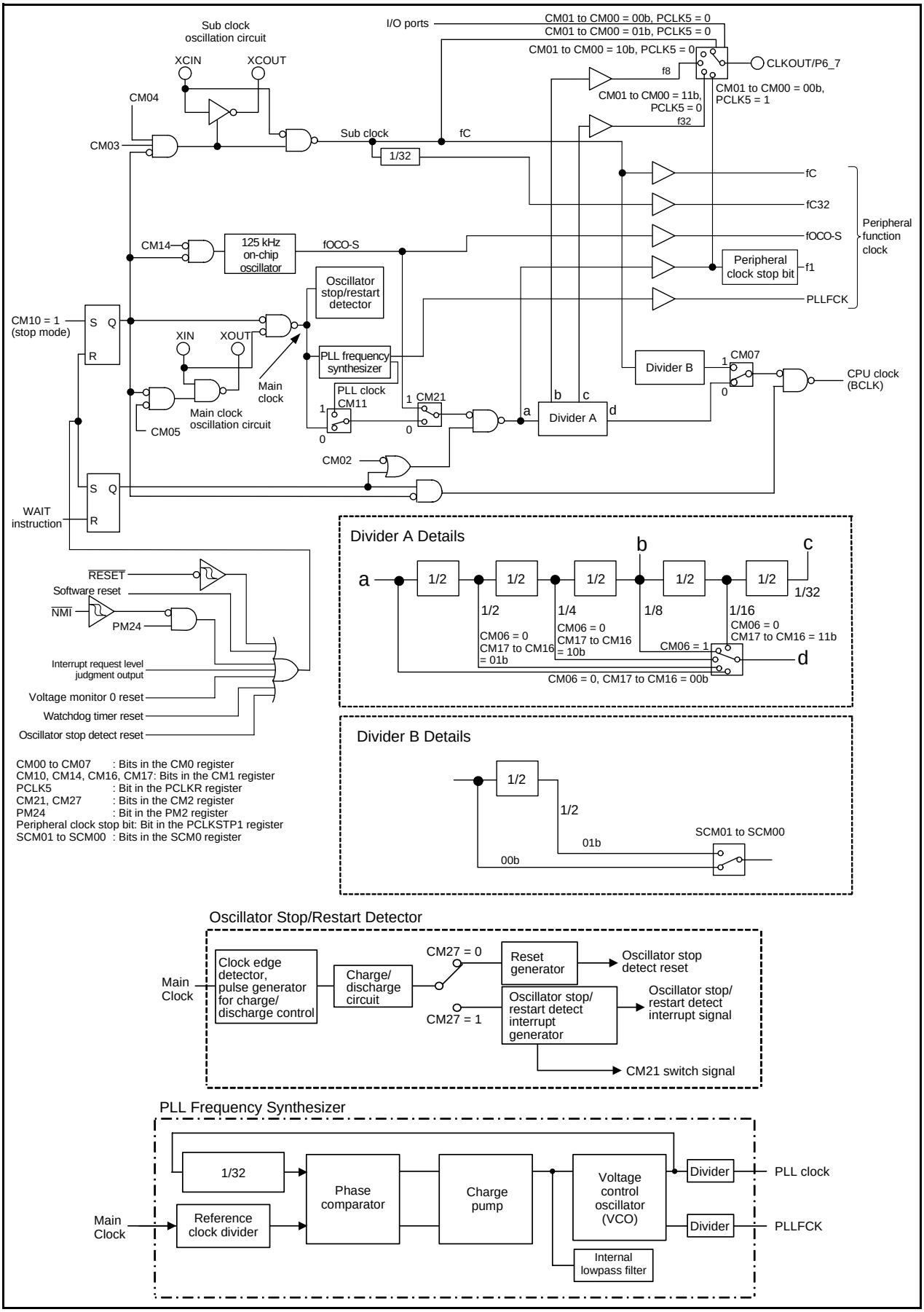


Figure 8.1 System Clock Generator

Table 8.2 I/O Pins

Pin Name	I/O	Function
XIN	Input	I/O pins for the main clock oscillation circuit
XOUT	Output	
XCIN	Input (1)	I/O pins for a sub clock oscillation circuit
XCOUT	Output (1)	
CLKOUT	Output	Clock output

Note:
1. Set the port direction bits which share pins to 0 (input mode).

8.2 Registers

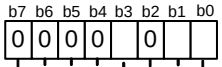
Table 8.3 Registers

Address	Register	Symbol	Reset Value
0004h	Processor Mode Register 0	PM0	0000 0000b
0006h	System Clock Control Register 0	CM0	0100 1000b
0007h	System Clock Control Register 1	CM1	0010 0000b
000Ch	Oscillation Stop Detection Register	CM2	0X00 0010b (1)
0012h	Peripheral Clock Select Register	PCLKR	0000 0011b
0013h	Sub Clock Division Control Register	SCM0	XXXX X000b
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
001Ch	PLL Control Register 0	PLC0	0001 X010b
001Dh	PLL Function Lock Control Register	PLCF	00h
001Eh	Processor Mode Register 2	PM2	XX00 0X01b

Note:
1. Bits CM20, CM21, and CM27 remain unchanged at oscillator stop detect reset.

8.2.1 Processor Mode Register 0 (PM0)

Processor Mode Register 0



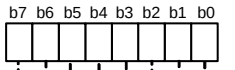
Symbol	Address	After Reset
PM0	0004h	0000 0000b

Bit Symbol	Bit Name	Function	RW
PM00	Processor mode bit	b1 b0 0 0: Single-chip mode 0 1: Do not set 1 0: Do not set 1 1: Do not set	RW
PM01			RW
— (b2)	Reserved bit	Set to 0.	RW
PM03	Software reset bit	Setting this bit to 1 resets the MCU. When read, the read value is 0.	RW
— (b7-b4)	Reserved bits	Set to 0.	RW

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

8.2.2 System Clock Control Register 0 (CM0)

System Clock Control Register 0



Symbol	Address	After Reset
CM0	0006h	0100 1000b

Bit Symbol	Bit Name	Function	RW
CM00	Clock output function select bit	b1 b0 0 0: I/O port 0 1: Output fC 1 0: Output f8 1 1: Output f32	RW
CM01			
CM02	Wait mode peripheral function clock stop bit	0: Peripheral function clock f1 does not stop in wait mode 1: Peripheral function clock f1 stops in wait mode	RW
CM03	XCIN clock stop bit	0: On 1: Off	RW
CM04	Port XC select bit	0: I/O ports 1: XCIN-XCOUT oscillation function	RW
CM05	Main clock stop bit	0: On 1: Off	RW
CM06	Main clock division select bit 0	0: Bits CM16 and CM17 in the CM1 register enabled 1: Divide-by-8 mode	RW
CM07	System clock select bit	0: Main clock, PLL clock, or on-chip oscillator clock 1: Sub clock	RW

Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register.
See Table 9.3 “Clock-Related Bit Setting and Modes” to select a clock and mode.

CM01 to CM00 (Clock output function select bit) (b1 to b0)

The CLKOUT pin outputs can be selected. These bits are enabled when the PCLK5 bit in the PCLKR register is set to 0 (selected by bits CM01 to CM00). When the PCLK5 bit is 1, set bits CM01 to CM00 to 00b. Table 8.4 lists CLKOUT Pin Functions.

Table 8.4 CLKOUT Pin Functions

PCLKR Register	CM0 Register		CLKOUT Pin Output
PCLK5 bit	CM01 bit	CM00 bit	
0	0	0	I/O port
0	0	1	fC is output
0	1	0	f8 is output
0	1	1	f32 is output
1	0	0	f1 is output

Only set the combinations listed above.

CM02 (Wait mode peripheral function clock stop bit) (b2)

This bit is used to stop the f1 peripheral function clock in wait mode. The fC, fC32, and fOCO-S are not affected by the CM02 bit.
When the PM21 bit in the PM2 register becomes 1 (clock change disabled), the CM02 bit remains unchanged even when written to.

CM03 (XCIN clock stop Bit) (b3)

The CM03 bit becomes 1 (off) while the CM04 bit is 0 (P8_6 and P8_7 are I/O ports).

CM04 (Port XC select bit) (b4)

The CM03 bit becomes 1 (off) while the CM04 bit is 0 (P8_6 and P8_7 are I/O ports).

CM05 (Main clock stop bit) (b5)

This bit is used to stop the main clock. The main clock is allowed to stop in the following cases.

- Entering low power mode
- Entering 125 kHz on-chip oscillator low power mode

This bit cannot be used to detect if the main clock is stopped or not. Refer to 8.7 “Oscillator Stop/Restart Detect Function” for details on main clock stop detection.
When the PM21 bit in the PM2 register is 1 (clock change disabled), this bit remains unchanged even when written to.

CM06 (Main clock division select bit) (b6)

The CM06 bit becomes 1 (divide-by-8 mode) under the following conditions:

- When entering stop mode
- When the CM21 bit in the CM2 register is 0 (main clock or PLL clock) and the CM05 bit is 1 (main clock off)

CM07 (System clock select bit) (b7)

The CPU clock source and the peripheral function clock f1 depend on combinations of the bit status of the CM07 bit, the CM11 bit in the CM1 register, and the CM21 bit in the CM2 register. When the CM07 bit is 0 (main clock, PLL clock, or on-chip oscillator clock used as CPU clock), the CPU clock source and the peripheral function clock f1 can be selected by combinations of the bit status of the CM11 bit and the CM21 bit. When the CM07 bit is 1 (sub clock used as CPU clock), the CPU clock source is fC, and the peripheral function clock f1 can be selected by combinations of the bit status of bits CM11 and CM21.
When setting the PM21 bit in the PM2 register to 1 (clock change disabled), set the CM07 bit to 0 (main clock) before setting the PM21 bit to 1. When the PM21 bit is set to 1, this bit remains unchanged even when written to.

8.2.3 System Clock Control Register 1 (CM1)

System Clock Control Register 1

b7	b6	b5	b4	b3	b2	b1	b0
					0		

Symbol	Address	After Reset
CM1	0007h	0010 0000b

Bit Symbol	Bit Name	Function	RW
CM10	All clock stop control bit	0: Clock on 1: All clocks off (stop mode)	RW
CM11	System clock select bit 1	0: Main clock 1: PLL clock	RW
— (b2)	Reserved bit	Set to 0.	RW
CM13	XIN-XOUT feedback resistor select bit	0: Internal feedback resistor connected 1: Internal feedback resistor not connected	RW
CM14	125 kHz on-chip oscillator stop bit	0: 125 kHz on-chip oscillator on 1: 125 kHz on-chip oscillator off	RW
CM15	XIN-XOUT drive capacity select bit	0: Low 1: High	RW
CM16	Main clock division select bit 1	b7 b6 0 0: No division mode 0 1: Divide-by-2 mode 1 0: Divide-by-4 mode 1 1: Divide-by-16 mode	RW
CM17			

Rewrite the CM1 register after setting the PRC0 bit in the PRCR register to 1 (write enabled).
See Table 9.3 “Clock-Related Bit Setting and Modes” to select a clock and a mode.

CM10 (All clock stop control bit) (b0)

When the CM11 bit is 1 (PLL clock), or the CM20 bit in the CM2 register is 1 (oscillator stop detect function enabled), do not set the CM10 bit to 1.
In the following cases, this bit remains unchanged even when written to (The MCU does not enter stop mode).

- The PM21 bit in the PM2 register is 1 (clock change disabled).
- The CSPRO bit in the CSPR register is 1 (watchdog timer count source protection mode enabled).
- A low is input to the $\overline{\text{NMI}}$ pin.

CM11 (System clock select bit) (b1)

The CM11 bit is valid when the CM21 bit in the CM2 register is set to 0 (main clock or PLL clock).
The CPU clock source and the peripheral function clock f1 can be selected by the CM11 bit when the CM07 bit is 0 (main clock, PLL clock, or on-chip oscillator clock used as CPU clock). The peripheral function clock f1 can be selected by the CM11 bit when the CM07 bit is 1 (sub clock used as CPU clock).
When the PM21 bit in the PM2 register is 1 (clock change disabled), the CM11 bit remains unchanged even when written to.

CM13 (XIN-XOUT feedback resistor select bit) (b3)

The CM13 bit can be used when the main clock is not used at all. When connecting a crystal between pins XIN and XOUT, set the CM13 bit to 0 (internal feedback resistor connected). Do not set this bit to 1.

When the CM10 bit is 1 (stop mode), the feedback resistor is not connected regardless of the CM13 bit status.

CM14 (125 kHz on-chip oscillator stop bit) (b4)

The CM14 bit can be set to 1 (125 kHz on-chip oscillator off) when the CM21 bit is 0 (main clock or PLL clock). When the CM21 bit is set to 1 (on-chip oscillator clock), the CM14 bit is automatically set to 0 (125 kHz on-chip oscillator on) and remains unchanged even when 1 is written to this bit. Note that the 125 kHz on-chip oscillator does not stop.

When the CSPRO bit in the CSPR register is 1 (watchdog timer count source protection mode), the CM14 bit is automatically set to 0 (125 kHz on-chip oscillator on) and remains unchanged even when 1 is written to this bit. Note that the 125 kHz on-chip oscillator does not stop.

CM15 (XIN-XOUT drive capacity select bit) (b5)

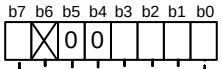
The CM15 bit becomes 1 (drive capacity high) when entering stop mode or when the CM21 bit in the CM2 register is 0 (main clock or PLL clock) and the CM05 bit in the CM0 register is set to 1 (main clock stopped).

CM17 to CM16 (Main clock division select bit 1) (b7 to b6)

Bits CM17 to CM16 are enabled when the CM06 bit becomes 0 (bits CM16 and CM17 enabled).

8.2.4 Oscillation Stop Detection Register (CM2)

Oscillation Stop Detection Register



Symbol: CM2 | Address: 000Ch | After Reset: 0X00 0010b

Bit Symbol	Bit Name	Function	RW
CM20	Oscillator stop/restart detect enable bit	0: Oscillator stop/restart detect function disabled 1: Oscillator stop/restart detect function enabled	RW
CM21	System clock select bit 2	0: Main clock or PLL clock 1: On-chip oscillator clock	RW
CM22	Oscillator stop/restart detect flag	0: Main clock stop/restart not detected 1: Main clock stop/restart detected	RW
CM23	XIN monitor flag	0: Main clock oscillating 1: Main clock stopped	RO
— (b5-b4)	Reserved bits	Set to 0.	RW
— (b6)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
CM27	Operation select bit (when an oscillator stop/restart is detected)	0: Oscillator stop detect reset 1: Oscillator stop/restart detect interrupt	RW

Rewrite the CM2 register after setting the PRC0 bit in the PRCR register to 1 (write enabled). Bits CM20, CM21, and CM27 do not change at oscillator stop detect reset.
See Table 9.3 “Clock-Related Bit Setting and Modes” to select a clock and a mode.

CM20 (Oscillator stop/restart detect enable bit) (b0)

Set the CM20 bit to 0 (oscillator stop/restart detect function disabled) to enter stop mode. Set the CM20 bit back to 1 (enabled) after exiting stop mode.
When the PM21 bit in the PM2 register is 1 (clock change disabled), the CM20 bit remains unchanged even when being written.

CM21 (System clock select bit 2) (b1)

When the CM07 bit is 0 (main clock, PLL clock, or on-chip oscillator clock used as CPU clock source), the CPU clock source and the peripheral function clock f1 can be selected by the CM21 bit. When the CM07 bit is 1 (sub clock used as CPU clock source), the peripheral function clock f1 can be selected by the CM21 bit.
When the CM20 bit is 1 (oscillator stop/restart detect function enabled) and the CM23 bit is 1 (main clock stopped), do not set the CM21 bit to 0 (main clock or PLL clock).
When the CM20 bit is 1 (oscillator stop/restart detect function enabled), the CM27 bit is 1 (oscillator stop/restart detect interrupt), and the main clock is used as a CPU clock source, the CM21 bit becomes 1 (on-chip oscillator clock) if the main clock stop is detected. Refer to 8.7 “Oscillator Stop/Restart Detect Function” for details.

CM22 (Oscillator stop/restart detect flag) (b2)

Condition to become 0:

- Set it to 0.

Conditions to become 1:

- Main clock stop is detected.
- Main clock restart is detected.

  (The CM22 bit remains unchanged even if 1 is written.)

When the CM22 bit changes state from 0 to 1, an oscillator stop/restart detect interrupt is generated. Use this bit in an interrupt routine to determine the factors of interrupts between the oscillator stop/restart detect interrupt and other interrupts.

When the CM22 bit is 1 and oscillator stop or restart is detected, an oscillator stop/restart detect interrupt is not generated. The bit does not become 0 even if an oscillator stop/restart detect interrupt request is accepted.

CM23 (XIN monitor flag) (b3)

Determine the main clock status by reading the CM23 bit several times in the oscillator stop/restart detect interrupt routine.

8.2.5 Peripheral Clock Select Register (PCLKR)

Peripheral Clock Select Register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
0	0		0	0	0			PCLKR	0012h	0000 0011b

Bit Symbol	Bit Name	Function	RW
PCLK0	Timers A and B clock select bit (clock source for timers A and B, and multi-master I^2C-bus interface)	0: f2TIMAB/f2IIC 1: f1TIMAB/f1IIC	RW
PCLK1	SI/O clock select bit (clock source for UART0 to UART2, UART5 to UART7, SI/O3, and SI/O4)	0: f2SIO 1: f1SIO	RW
— (b4-b2)	Reserved bits	Set to 0.	RW
PCLK5	Clock output function expansion bit	0: Selected by bits CM01 to CM00 in the CM0 register 1: Output f1	RW
— (b7-b6)	Reserved bits	Set to 0.	RW

Write to the PCLKR register after setting the PRC0 bit in the PRCR register to 1 (write enabled).

PCLK5 (Clock output function extension bit) (b5)

Output from the CLKOUT pin is selectable. When the PCLK5 bit is 1, set bits CM01 to CM00 to 00b. See Table 8.4 “CLKOUT Pin Functions”.

8.2.6 PLL Control Register 0 (PLC0)

PLL Control Register 0

b7 b6 b5 b4 b3 b2 b1 b0	Symbol	Address	After Reset
	PLC0	001Ch	0001 X010b

Bit Symbol	Bit Name	Function	RW
PLC00	PLL multiplying factor select bit	b2 b1 b0 0 0 0: Do not set 0 0 1: Multiply-by-2 0 1 0: Multiply-by-4 0 1 1: Multiply-by-6 1 0 0: Multiply-by-8 1 0 1: 1 1 0: 1 1 1: Do not set these values	RW
PLC01			RW
PLC02			RW
— (b3)	Reserved bit	When read, the read value is undefined.	RO
PLC04	Reference frequency counter set bit	b5 b4 0 0: No division 0 1: Divide-by-2 1 0: Divide-by-4 1 1: Do not set	RW
PLC05			RW
PLC06	PLLFCK generation enable bit	0: PLLFCK generation disabled 1: PLLFCK generation enabled	RW
PLC07	Operation enable bit	0: PLL off 1: PLL on	RW

Rewrite the PLC0 register after setting the PRC0 bit in the PRCR register to 1 (write enabled).

PLC02 to PLC00 (PLL multiplying factor select bit) (b2 to b0)

Write to bits PLC00 to PLC02 when the PLC07 bit is 0 (PLL off).
When the PM21 bit in the PM2 register is 1 (clock change disabled), writing to bits PLC02 to PLC00 has no effect.

PLC05 to PLC04 (Reference frequency counter set bit) (b5 to b4)

Write to bits PLC05 to PLC04 when the PLC07 bit is 0 (PLL off).
When the PM21 bit in the PM2 register is 1 (clock change disabled), writing to bits PLC05 to PLC04 has no effect.

PLC06 (PLLFCK generation enable bit) (b6)

Write to the PLC06 bit when the PLC07 bit is 0 (PLL off).
Set the PLC06 bit to 1 (PLLFCK generation enabled) when using PLC functions. Table 8.5 lists PLC06 Bit Functions.
When the PM21 bit in the PM2 register is 1 (clock change disabled), writing to the PLC06 bit has no effect.

Table 8.5 PLC06 Bit Functions

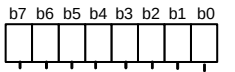
Item	PLC06 Bit	
	0	1
PLLFCK generation	Disabled	Enabled
Bits PLC02 to PLC00 in the PLC0 register	Enabled	Disabled
PLCF register	Disabled	Enabled

PLC07 (Operation enable bit) (b7)

When the PM21 bit in the PM2 register is 1 (clock change disabled), writing to the PLC07 bit has no effect.

8.2.7 PLL Function Lock Control Register (PLCF)

PLLFCK Control Register



Symbol	Address	After Reset
PLCF	001Dh	00h

Bit Symbol	Bit Name	Function	RW
PLCF0	PLLFCK multiplying factor select bit	b1 b0 0 0: Multiply by 8 0 1: Multiply by 12 1 0: Multiply by 16 1 1: Multiply by 24	RW
PLCF1			RW
PLCF2	PLL clock select bit during PLLFCK enabled	b1 b0 0 0: Do not use PLL clock 0 1: Divide-by-3 1 0: Divide-by-4 1 1: Divide-by-8	RW
PLCF3			RW
— (b7-b4)	Reserved bits	Set to 0.	RW

Write to the PLCF register after setting the PRC0 bit in the PRCR register to 1 (write enabled).
The PLCF register is enabled when the PLC06 bit in the PLC0 register is 1 (PLLFCK generation enabled).

PLCF1 to PLCF0 (PLLFCK multiplying factor select bit) (b1-b0)

Set bits PLCF1 to PLCF0 so that PLLFCK is 46.08 MHz.
Write to these bits when the PLC07 bit is 0 (PLL off).
When the PM21 bit in the PM2 register is 1 (clock change disabled), writing to bits PLCF1 to PLCF0 has no effect.

PLCF3 to PLCF2 (PLLFCK clock select bit during PLLFCK enabled) (b3-b2)

When using the PLC MODEM function and PLL clock, set one of the following: 01b (divide-by-3), 10b (divide-by-4), and 11b (divide-by-8). 00b can be set when the PLL clock is not used.
Write to these bits when the PLC07 bit is 0 (PLL off).
When the PM21 bit in the PM2 register is 1 (clock change disabled), writing to bits PLCF 3 to PLCF 2 has no effect.

8.2.8 Sub Clock Division Control Register (SCM0)

Sub Clock Division Control Register

b7	b6	b5	b4	b3	b2	b1	b0
×	×	×	×	×	0		

Symbol	Address	After Reset
SCM0	0013h	XXXXX000b

Bit Symbol	Bit Name	Function	RW
SCM00	Sub clock division ratio select bit	b1 b0 0 0: No division 0 1: Divided-by-2 1 0: Do not set 1 1: Do not set	RW
SCM01			
— (b2)	Reserved bit	Set to 0.	RW
— (b7-b3)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register.

8.2.9 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1

b7	b6	b5	b4	b3	b2	b1	b0
×				×			

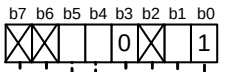
Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I^2C peripheral clock stop bit (multi-master I^2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register

8.2.10 Processor Mode Register 2 (PM2)

Processor Mode Register 2



Symbol	Address	After Reset
PM2	001Eh	XX00 0X01b

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved bit	Set to 1.	RW
PM21	System clock protection bit	0: Clock is protected by PRCR register 1: Clock change disabled	RW
— (b2)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
— (b3)	Reserved bit	Set to 0.	RW
PM24	$\overline{\text{NMI}}$ interrupt enable bit	0: $\overline{\text{NMI}}$ interrupt disabled 1: $\overline{\text{NMI}}$ interrupt enabled	RW
PM25	Peripheral clock fC provide bit	0: Not provided 1: Provided	RW
— (b7-b6)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

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

PM21 (System clock protection bit) (b1)

The PM21 bit is used to protect the CPU clock. (Refer to 8.6 “System Clock Protection Function”).
When the PM21 bit is set to 1, writing to the following bits has no effect:

- Bits CM02, CM05, and CM07 in the CM0 register
- Bit CM10 and CM11 in the CM1 register
- The CM20 bit in the CM2 register
- All bits in the PLC0 register

Do not execute the WAIT instruction when the PM21 bit is 1.
Once the PM21 bit is set to 1, it cannot be set to 0 by a program (writing 0 has no effect).

PM25 (Peripheral clock fC provide bit) (b5)

The PM25 bit provides fC to the real-time clock. (See Figure 8.5 “Peripheral Function Clocks”.)

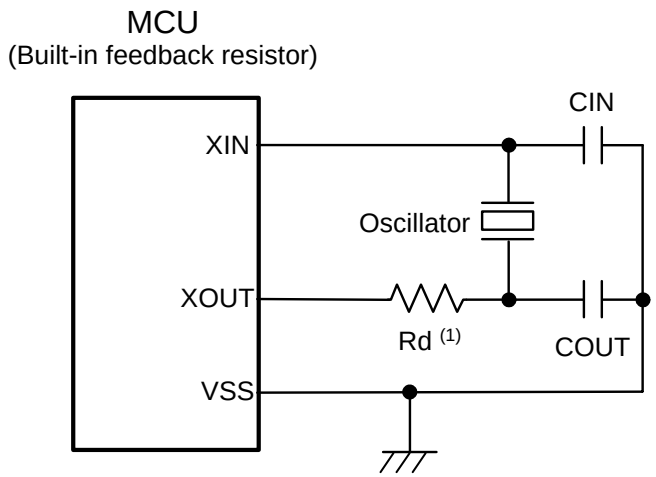
8.3 Clocks Generated by Clock Generators

Clocks generated by the clock generators are described below.

8.3.1 Main Clock

This clock is supplied by the main clock oscillator circuit and used as a clock source for the CPU and peripheral function clocks. After reset, the main clock is running, but is not used as a clock source for the CPU.

The main clock oscillator circuit is configured by connecting a crystal between pins XIN and XOUT. The main clock oscillator circuit contains a feedback resistor, which is separated from the oscillator circuit in stop mode in order to reduce the amount of power consumed by the chip. Figure 8.2 shows Main Clock Connection Examples.



Note:

1. Insert a damping resistor if required. The resistance will vary depending on the oscillator and the oscillation drive capacity setting. Use the value recommended by the oscillator manufacturer.
   When the oscillation drive capacity is set to low, check if oscillation is stable in low status. Also, place a feedback resistor between XIN and XOUT if the oscillator manufacturer recommends placing the resistor externally.

Figure 8.2 Main Clock Connection Examples

The XOUT becomes high by setting the CM05 bit in the CM0 register to 1 (main clock oscillator circuit turned off) after switching the clock source for the CPU clock to the sub clock (fC) or on-chip oscillator clock (fOCO-S). In this case, the XIN is pulled high to the XOUT via the feedback resistor because the internal feedback resistor remains connected.

When the main clock oscillator circuit is not used, setting the CM13 bit in the CM1 register to 1 enables to select the internal feedback resistor not connected.

Perform the following steps to start or stop the main clock. Refer to 8.2 “Registers” for access to register and bit.

Main clock oscillator start

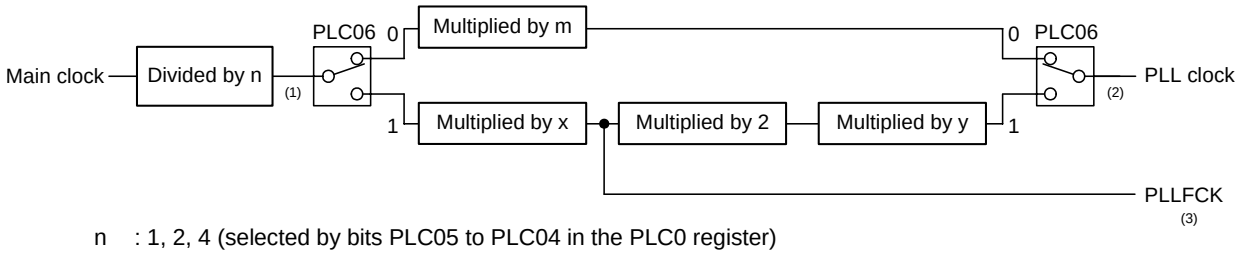
(1) Set the CM15 bit to 1 (drive capacity high) when a crystal is connected between pins XIN and XOUT.
(2) Set the CM05 bit to 0 (main clock oscillating).
(3) Wait until main clock oscillation stabilizes.

Main clock oscillator stop

(1) Set the CM20 bit in the CM2 register to 0 (oscillator stop/restart detect function disabled).
(2) Set the CM05 bit to 1 (stop).

8.3.2 PLL Clock

The PLL clock is generated by the PLL frequency synthesizer. This clock is used as the clock source for the CPU and peripheral function clocks.
After reset, the PLL frequency synthesizer is stopped.
The PLL clock is the main clock divided by bits PLC05 to PLC04 in the PLC0 register and then multiplied by bits PLC02 to PLC00. Set bits PLC05 to PLC04 so that the divided frequency will be between 2 MHz and 6 MHz. Figure 8.3 shows Relation between Main Clock and PLL Clock.



n : 1, 2, 4 (selected by bits PLC05 to PLC04 in the PLC0 register)
m : 2, 4, 6, 8 (selected by bits PLC02 to PLC00 in the PLC0 register)
x : 8, 12, 16, 24 (selected by bits PLCF1 to PLCF0 in the PLCF register)
y : 3, 4, 8 (selected by bits PLCF3 to PLCF2 in the PLCF register)

Notes:
1. Set the frequency divided by n to between 2 MHz and 6 MHz.
2. Set 10 MHz ≤ PLL clock frequency ≤ 32 MHz
3. Set the frequency to 46.08 MHz to use the PLC MODEM function.

Figure 8.3 Relation between Main Clock and PLL Clock

Bits PLC05 to PLC04, PLC02 to PLC00 can be set only once after reset.
Table 8.6 and 8.7 list Example Settings for PLL Clock Frequencies.

Table 8.6 Example Settings for PLL Clock Frequencies

Main Clock	Setting Value		PLL Clock
	Bits PLC05 to PLC04	Bits PLC02 to PLC00	
15.36 MHz	10b (divide-by-4)	100b (multiply-by-8)	30.72 MHz

Table 8.7 Example Settings for PLL Clock Frequencies (When the PLC06 Bit in the PLC0 Register is 1 (PLLFCK Enabled))

Frequency Divided by n	Bits in the PLCF Register		PLLFCK	PLL Clock
	PLCF1 to PLCF0	PLCF3 to PLCF2		
3.84 MHz	01b (multiplied by 12)	01b (divided by 3)	46.08 MHz	30.72 MHz
		10b (divided by 4)	46.08 MHz	23.04 MHz
		11b (divided by 8)	46.08 MHz	11.52 MHz

8.3.3 125 kHz On-Chip Oscillator Clock (fOCO-S)

This clock is approximately 125 kHz, and is supplied by the 125 kHz on-chip oscillator. It is used as the clock source for the CPU and peripheral function clocks. In addition, when the CSPRO bit in the CSPR register is 1 (count source protection mode enabled), this clock is used as the count source for the watchdog timer (refer to 14.4.2 “Count Source Protection Mode Enabled”).

After reset, fOCO-S divided by 8 becomes the CPU clock.

If the main clock stops oscillating, when the CM20 bit in the CM2 register is 1 (oscillator stop/restart detect function enabled) and the CM27 bit is 1 (oscillator stop/restart detect interrupt), the 125 kHz on-chip oscillator automatically starts operating and supplying the necessary clock for the MCU.

Follow the steps below to start or stop the fOCO-S. Refer to 8.2 “Registers” for access to register and bit.

fOCO-S start

(1) Set the CM14 bit in the CM1 register to 0 (125 kHz on-chip oscillator on).

(2) Wait for tsu(fOCO-S).

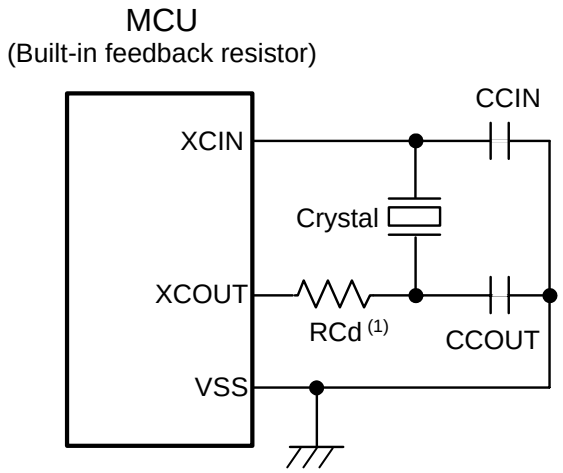
fOCO-S stop

(1) Set the CM14 bit in the CM1 register to 1 (125 kHz on-chip oscillator off).

When the CM21 bit is 1 (on-chip oscillator used as the clock source for the CPU), the CM14 bit becomes 0 (125 kHz on-chip oscillator on).

8.3.4 Sub Clock (fC)

The sub clock is supplied by the sub clock oscillator circuit. This clock is the clock source for count sources of the CPU clock, timer A, timer B, real-time clock.
The sub clock oscillator circuit is configured by connecting a crystal between pins XCIN and XCOUT. The sub clock oscillator circuit contains a feedback resistor, which is disconnected from the oscillation circuit in stop mode in order to reduce the amount of power consumed by the chip. Figure 8.4 shows Sub Clock Connection Examples.



Note:
1. Insert a damping resistor if required. The resistance will vary depending on the oscillator. Use the value recommended by the oscillator manufacturer.
Also, place a feedback resistor between XCIN and XCOUT if the oscillator manufacturer recommends placing the resistor externally.

Figure 8.4 Sub Clock Connection Examples

After reset, the sub clock is stopped. At this time, the feedback resistor is disconnected from the oscillator circuit.
Follow the steps below to start the sub clock. Refer to 8.2 “Registers” for access to registers and bits.

When connecting a crystal between pins XCIN and XCOUT:
(1) Set the PU21 bit in the PUR2 register to 0 (P8_4, P8_6, and P8_7 not pulled high).
(2) Set bits PD8_6 and PD8_7 in the PD8 register to 0 (P8_6, P8_7 function as input ports).
(3) Set the CM04 bit to 1 (XCIN-XCOUT oscillation function).
(4) Set the CM03 bit to 0 (sub clock on).
(5) Wait until sub clock oscillation stabilizes.

8.4 CPU Clock and Peripheral Function Clocks

The CPU is run by the CPU clock, and the peripheral functions are run by the peripheral function clocks.

8.4.1 CPU Clock and BCLK

The CPU clock is an operating clock for the CPU and watchdog timer. It is also used as a sampling clock for the $\overline{\text{NMI}}$ digital filter.

The main clock, PLL clock, fOCO-S, or fC can be selected as the clock source for the CPU clock. (See Table 9.2 “Clocks in Normal Operating Mode”.)

When the main clock, PLL clock, or fOCO-S is selected as the clock source for the CPU clock, the selected clock divided by 1, 2, 4, 8 or 16 becomes the CPU clock. Use the CM06 bit in the CM0 register and bits CM17 to CM16 in the CM1 register to select a frequency-divided value.

When fC is selected as the clock source for the CPU clock, fC divided by 1 (no division) or 2 is used as the CPU clock. The divisor is selected by bits SCM01 to SCM00 in the SCM0 register.

After reset, fOCO-S divided by 8 becomes the CPU clock. Note that when entering stop mode or when the CM21 bit in the CM2 register is set to 0 (main clock or PLL clock) and the CM05 bit in the CM0 register is set to 1 (stop) in low-speed mode, the CM06 bit in the CM0 register becomes 1 (divide-by-8 mode).

BCLK is a bus reference clock.

8.4.2 Peripheral Function Clocks (f1, fOCO-S, fC32, fC)

f1, fOCO-S, and fC32 are operating clocks for the peripheral functions.

f1 is produced from one of the following:

- Main clock divided by 1 (no division)
- PLL clock divided by 1 (no division)
- fOCO-S divided by 1 (no division)

f1 is used for timers A and B, real-time clock, UART0 to UART2, UART5 to UART7, SI/O3, SI/O4, multi-master I^2C-bus interface, and the A/D converter. The f1 clock provided to each peripheral function can be disabled in the PCLKSTP1 register.

When the WAIT instruction is executed after setting the CM02 bit in the CM0 register to 1 (peripheral function clock f1 turned off during wait mode), the f1 clock is stopped.

fOCO-S is used for timers A and B. It is also used for reset, voltage detector, and watchdog timer. fOCO-S is also used when the CM14 bit in the CM1 register is set to 0 (125 kHz on-chip oscillator on).

fC is divided by 32 to produce fC32. fC32 is used for timers A and B, and can be used when the sub clock is on.

fC is used as the count source for the real-time clock when the PM25 bit in the PM2 register is 1 (peripheral clock fC provided). fC can be used when the sub clock is on.

Figure 8.5 shows Peripheral Function Clocks.

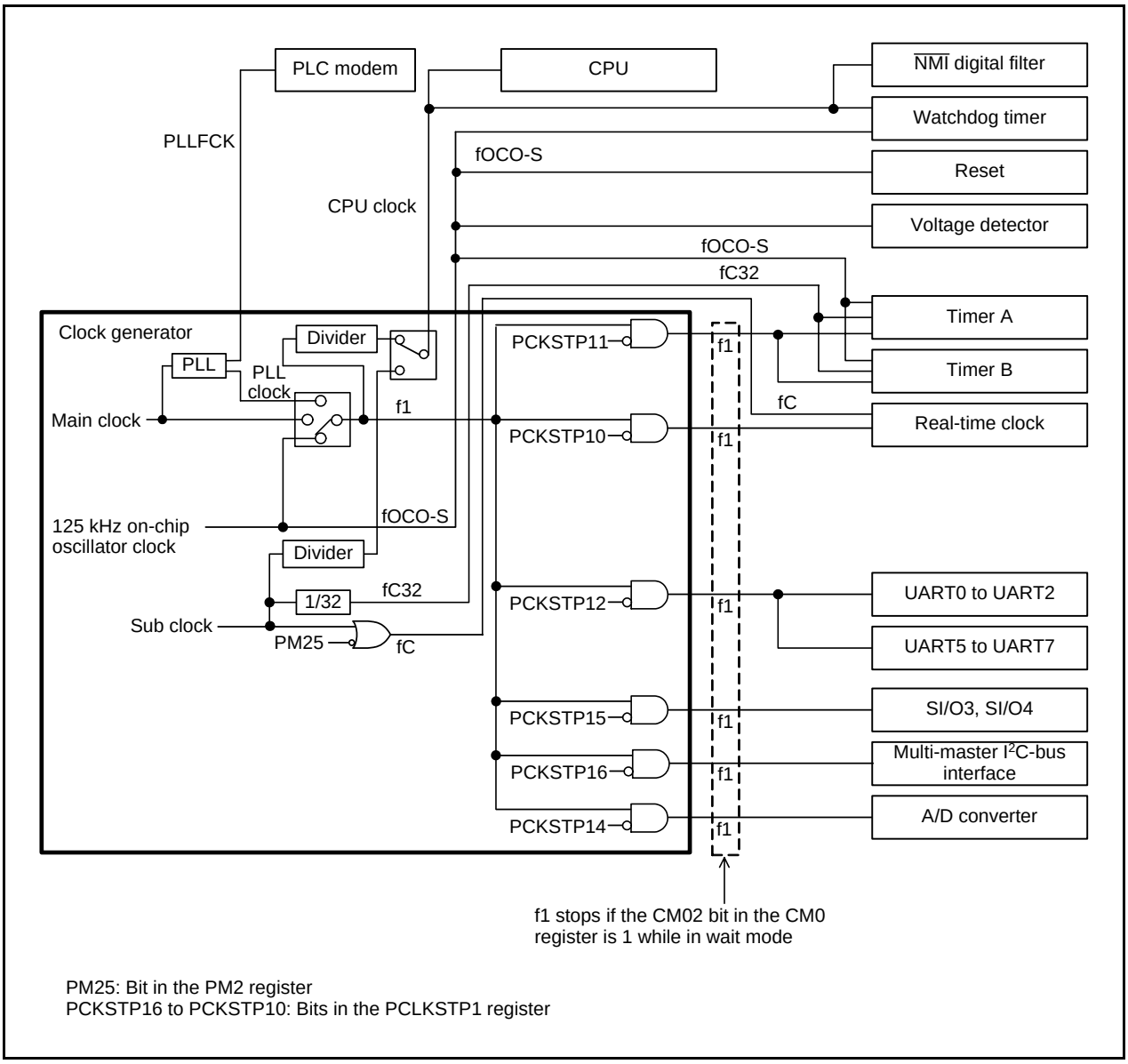


Figure 8.5 Peripheral Function Clocks

8.5 Clock Output Function

The f1, f8, f32 or fC clock can be output from the CLKOUT pin. Use bits CM01 to CM00 in the CM0 register, and the PCLK5 bit in the PCLKR register to select a clock. f8 has the same frequency as f1 divided by 8, and f32 has the same frequency as f1 divided by 32.
Set the frequency of the clock output from the CLKOUT pin to 25 MHz or below.

8.6 System Clock Protection Function

The system clock protection function prohibits the CPU clock from changing clock sources when the main clock is selected as the CPU clock source. This is to prevent the CPU clock from stopping due to an unexpected program operation.
When the PM21 bit in the PM2 register is set to 1 (clock change disabled), the following bits remain unchanged even if they are written to:

- The CM02 bit in the CM0 register (peripheral function clock f1 in wait mode)
- The CM05 bit in the CM0 register (to prevent the main clock from being stopped)
- The CM07 bit in the CM0 register (clock source of the CPU clock)
- The CM10 bit in the CM1 register (MCU does not enter stop mode)
- The CM11 bit in the CM1 register (clock source of the CPU clock)
- The CM20 bit in the CM2 register (oscillator stop/restart detect function set)
- All bits in the PLC0 register (PLL frequency synthesizer set)

To use the system clock protect function, set the CM05 bit in the CM0 register to 0 (main clock oscillation) and CM07 bit to 0 (main clock as CPU clock source), and then follow the steps below.

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

When the PM21 bit is 1, do not execute the WAIT instruction.

8.7 Oscillator Stop/Restart Detect Function

This function detects a stop/restart of the main clock oscillation circuit. The oscillator stop/restart detect function can be enabled and disabled with the CM20 bit in the CM2 register.
A reset or oscillator stop/restart detect interrupt is generated when an oscillator stop or restart is detected. Set the CM27 bit in the CM2 register to select the reset or interrupt.
Table 8.8 lists Oscillator Stop/Restart Detect Function Specifications.

Table 8.8 Oscillator Stop/Restart Detect Function Specifications

Item	Specification
Oscillator stop detectable clock and frequency bandwidth	f(XIN) ≥ 2 MHz
Enabling condition for the oscillator stop/restart detect function	Set the CM20 bit to 1 (enabled)
Operation when oscillator stop/restart detected	When CM27 bit is 0: Oscillator stop detect reset generated When CM27 bit is 1: Oscillator stop/restart detect interrupt generated

8.7.1 Operation When CM27 Bit is 0 (Oscillator Stop Detect Reset)

When main clock stop is detected while the CM20 bit is 1 (oscillator stop/restart detect function enabled), the MCU is initialized, and then stops (oscillator stop reset). (Refer to 4. “Special Function Registers (SFRs)” and 6. “Resets”.)
The status is cancelled at hardware reset or voltage monitor 0 reset. The MCU can also be initialized and stopped when a restart is detected, but do not use the MCU in this manner (during main clock stop, do not set the CM20 bit to 1 and the CM27 bit to 0).

8.7.2 Operation When CM27 Bit is 1 (Oscillator Stop/Restart Detect Interrupt)

When the CM20 bit is 1 (oscillator stop/restart detect function enabled), the system is placed in the state shown in Table 8.9 if the main clock detects oscillator stop or restart.

The CM21 bit becomes 1 in high-speed, medium-speed, or low-speed mode. Thus, high-speed and medium-speed mode become 125 kHz on-chip oscillator mode. Because the CM07 bit does not change, low-speed mode remains in low-speed mode, but fOCO-S becomes the clock source for the peripheral functions.

When the CM21 bit is set to 1, the CM14 bit is set to 0 (125 kHz on-chip oscillator on).

Since the CM21 bit remains unchanged in PLL operating mode, select 125 kHz on-chip oscillator mode inside the interrupt routine.

Table 8.9 State after Oscillator Stop/Restart Detect When CM27 Bit is 1

Condition		After Detection
Main clock oscillator stop detected	High-speed mode Medium-speed mode Low-speed mode 125 kHz on-chip oscillator mode	• Oscillator stop/restart detect interrupt is generated • CM14 bit is 0 (125 kHz on-chip oscillator on) • CM21 bit is 1 (fOCO-S is used as the clock source for the CPU and peripheral function clocks) (1) • CM22 bit is 1 (main clock stop detected) • CM23 bit is 1 (main clock stopped)
	PLL operating mode	• Oscillator stop/restart detect interrupt is generated • CM14 bit is 0 (125 kHz on-chip oscillator on) • CM21 bit remains unchanged • CM22 bit is 1 (main clock stop detected) • CM23 bit is 1 (main clock stopped)
Main clock oscillator restart detected	—	• Oscillator stop/restart detect interrupt is generated • CM14 bit is 0 (125 kHz on-chip oscillator on) • CM21 bit does not change • CM22 bit is 1 (main clock stop detected) • CM23 bit is 0 (main clock oscillating)

CM14 bit: Bit in the CM1 register

Bits CM21, CM22, CM23: Bits in the CM2 register

Note:

1. fC is used as the CPU clock in low-speed mode.

8.7.3 Using the Oscillator Stop/Restart Detect Function

After oscillator stop is detected, if the main clock re-oscillates, set the main clock back to the clock source for the CPU clock and peripheral functions by a program. Figure 8.6 shows the Switching from On-Chip Oscillator Clock to Main Clock.

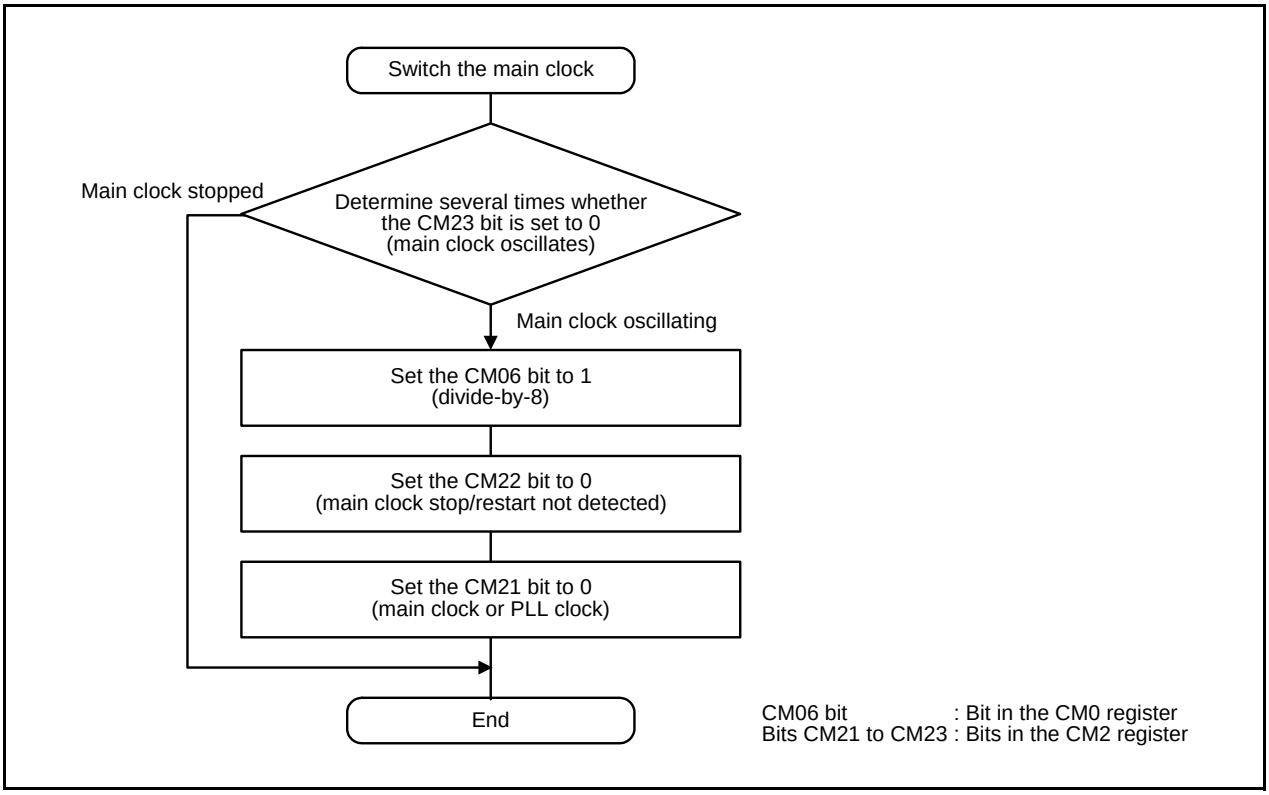


Figure 8.6 Switching from On-Chip Oscillator Clock to Main Clock

The CM22 bit becomes 1 at the same time an oscillator stop/restart detect interrupt is generated. When the CM22 bit is 1, the oscillator stop/restart detect interrupt is disabled. When setting the CM22 bit to 0 by a program, the oscillator stop/restart detect interrupt is enabled.

8.8 Interrupt

The oscillator stop/restart detect interrupt is a non-maskable interrupt.

The watchdog timer interrupt and oscillator stop/restart detect interrupt share the same vector. When using multiple interrupts together, read the detect flags of the events in the interrupt processing program, and determine the source of the interrupt.

The detect flag for oscillator stop/restart detect is the CM22 bit in the CM2 register. After the interrupt source is determined, set the CM22 bit to 0 (not detected).

8.9 Notes on Clock Generator

8.9.1 Oscillation Circuit Using an Oscillator

The following items should be observed when connecting an oscillator:

- The oscillation characteristics are tied closely to the user's board design. Perform a careful evaluation of the board before connecting an oscillator.
- Oscillation circuit structure depends on the oscillator. The M16C/6S1 Group MCU contains a feedback resistor, but an additional external feedback resistor may be required. Contact the oscillator manufacturer regarding circuit constants, as they are dependent on the oscillator or stray capacitance of the mounted circuit.
- Check output from the CLKOUT pin to confirm that the clock generated by the oscillation circuit is properly transmitted to the MCU.

The procedure for outputting a clock from the CLKOUT pin is listed below. Set the clock output from the CLKOUT pin to 25 MHz or lower.

Outputting the main clock

(1) Set the PRC0 bit in the PRCR register to 1 (write enabled).
(2) Set the CM11 bit in the CM1 register, the CM07 bit in the CM0 register, and the CM21 bit in the CM2 register all to 0 (main clock selected).
(3) Select the clock output from the CLKOUT pin (see the table below).
(4) Set the PRC0 bit in the PRCR register to 0 (write disabled).

Table 8.10 Output from CLKOUT Pin When Selecting Main Clock

Bit Setting		Output from the CLKOUT Pin
PCLKR register	CM0 register	
PCLK5 bit	Bits CM01 to CM00	
1	00b	Clock with the same frequency as the main clock
0	10b	Main clock divided by 8
0	11b	Main clock divided by 32

Outputting the sub clock

(1) Set the PRC0 bit in the PRCR register to 1 (write enabled).
(2) Set the CM07 bit in the CM0 register to 1 (sub clock selected).
(3) Set the PCLK5 bit in the PCLKR register to 0, and bits CM01 to CM00 in the CM0 register to 01b (fC output from CLKOUT pin).
(4) Set the PRC0 bit in the PRCR register to 0 (write disabled).

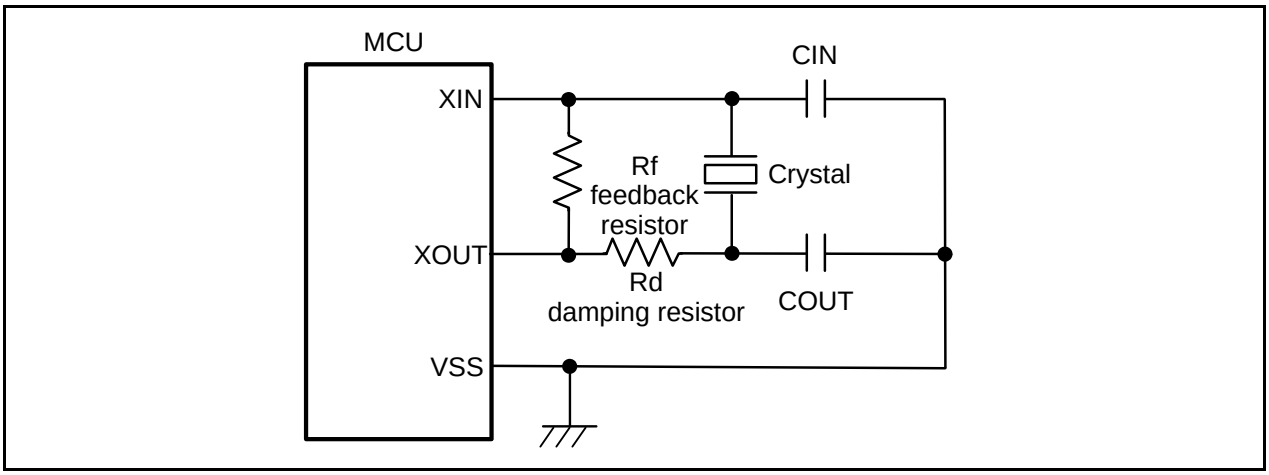


Figure 8.7 Oscillation Circuit Example

8.9.2 Noise Countermeasure

8.9.2.1 Clock I/O Pin Wiring

- Connect the shortest possible wiring to the clock I/O pin.
- Connect (a) the capacitor's ground lead connected to the oscillator, and (b) the MCU's VSS pin, with the shortest possible wiring (maximum 20 mm).

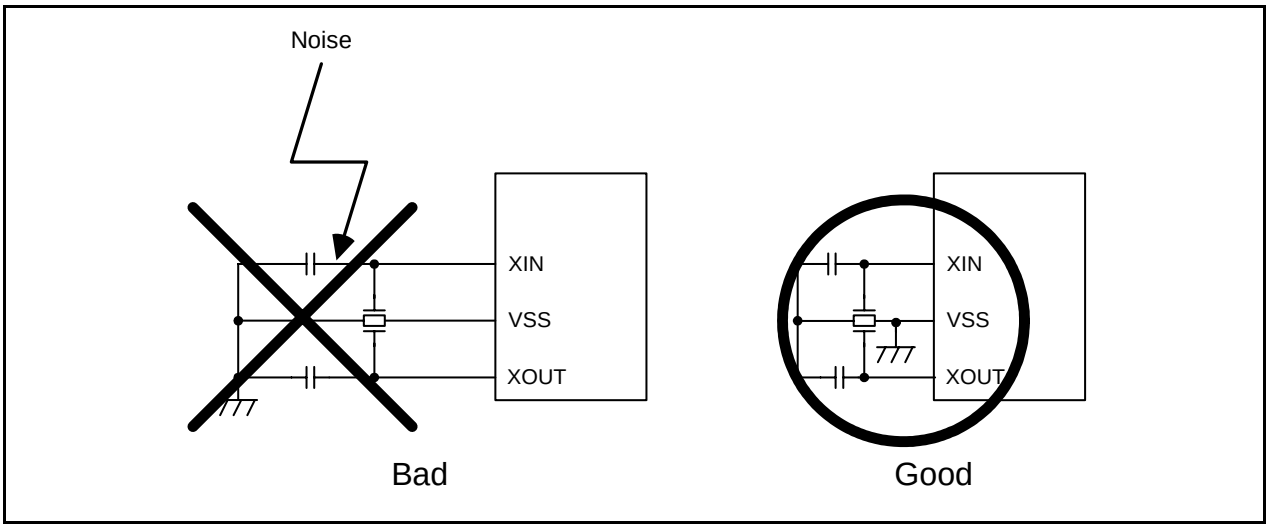


Figure 8.8 Clock I/O Pin Wiring

Reason:
If noise enters the clock I/O pin, the clock waveform becomes unstable, which causes an error in operation or a program runaway. Also, if a potential difference attributed to the noise occurs between the VSS level of the MCU and the VSS level of the oscillator, an accurate clock is not input to the MCU.

8.9.2.2 Large Current Signal Line

For large currents that exceed the MCU's current range, wire the signal lines as far away from the MCU as possible (especially the oscillator).

Reason:
In the system using the MCU, there are signal lines for controlling motors, LEDs, and thermal heads. When a large current flows through these signal lines, noise is generated due to mutual inductance.

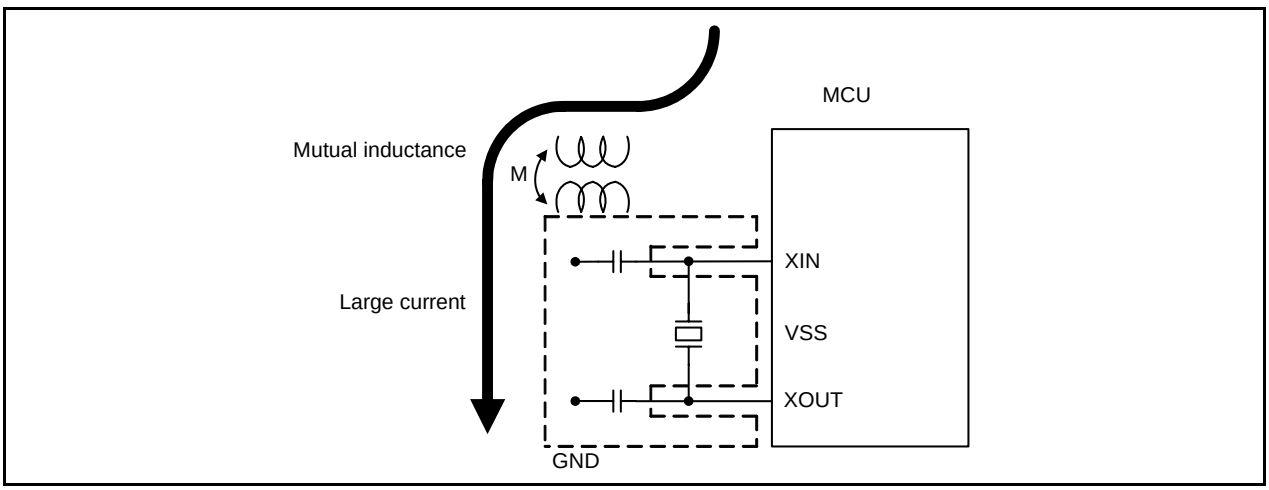


Figure 8.9 Large Current Signal Line Wiring

8.9.2.3 Signal Line Whose Level Changes at a High-Speed

For a signal line whose level changes at a high-speed, wire it as far away from the oscillator and the oscillator wiring pattern as possible. Do not wire it across or extend it parallel to a clock-related signal line or other signal lines which are sensitive to noise.

Reason:
A signal whose level changes at a high-speed (such as the signal from the TAiOUT pin) affects other signal lines due to the level change at rising or falling edges. Specifically, when the signal line crosses the clock-related signal line, the clock waveform becomes unstable, which causes an error in operation or a program runaway.

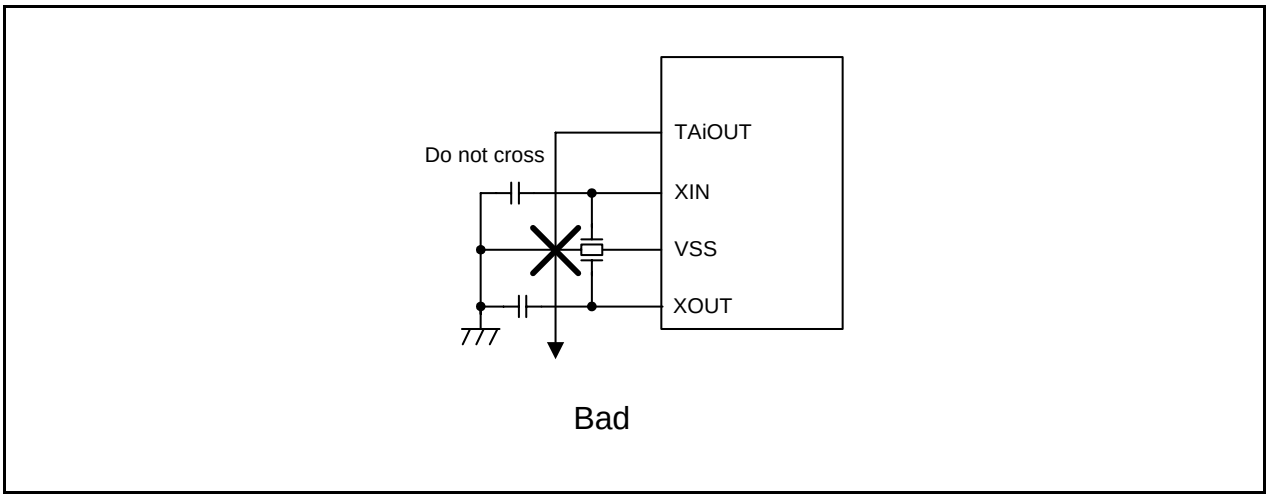


Figure 8.10 Wiring of Signal Line Whose Level Changes at High-Speed

8.9.3 Oscillation Stop/Restart Detect Function

- In the following cases, set the CM20 bit to 0 (oscillation stop/restart detect function disabled), and then change the setting of each bit.
  - When the CM05 bit is set to 1 (main clock stopped)
  - When the CM10 bit is set to 1 (stop mode)
- To enter wait mode while using the oscillation stop/restart detect function, set the CM02 bit to 0 (peripheral function clock f1 not turned off during wait mode).
- This function cannot be used if the main clock frequency is 2 MHz or lower. In that case, set the CM20 bit to 0 (oscillation stop/restart detect function disabled).

8.9.4 PLL Frequency Synthesizer

To use the PLL frequency synthesizer, stabilize the supply voltage within to the acceptable range of power supply ripple.

Table 8.11 Acceptable Range of Power Supply Ripple

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
$f_{(ripple)}$	Power supply ripple allowable frequency (VCC1)			10	kHz
$V_{P-P(ripple)}$	Power supply ripple allowable amplitude voltage			0.3	V
$V_{CC(\lvert\Delta V/\Delta T\rvert)}$	Power supply ripple rising/falling gradient			0.3	V/ms

f(ripple)
Power supply ripple allowable frequency (VCC1)

Vp-p(ripple)
Power supply ripple allowable amplitude voltage

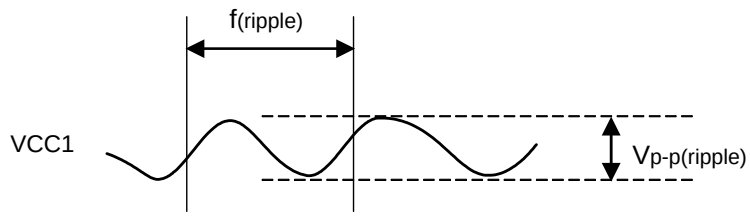


Figure 8.11 Voltage Fluctuation Timing

9. Power Control

9.1 Introduction

This chapter describes how to reduce the amount of current consumption.

9.2 Registers

Refer to 8. “Clock Generator” for the clock-related registers.

Table 9.1 Registers

Address	Register	Symbol	Reset Value
0220h	Flash Memory Control Register 0	FMR0	0000 0001b (Other than user boot mode) 0010 0001b (User boot mode)
0222h	Flash Memory Control Register 2	FMR2	XXXX 0000b

9.2.1 Flash Memory Control Register 0 (FMR0)

Flash Memory Control Register 0

b7	b6	b5	b4	b3	b2	b1	b0
			0				

Symbol	Address	After Reset
FMR0	0220h	0000 0001b (other than user boot mode) 0010 0001b (user boot mode)

Bit Symbol	Bit Name	Function	RW
FMR00	RY/$\overline{BY}$ status flag	0: Busy (being written or erased) 1: Ready	RO
FMR01	CPU rewrite mode select bit	0: CPU rewrite mode disabled 1: CPU rewrite mode enabled	RW
FMR02	Lock bit disable select bit	0: Lock bit enabled 1: Lock bit disabled	RW
FMSTP	Flash memory stop bit	0: Flash memory operation enabled 1: Flash memory operation stopped (low power-mode, flash memory initialized)	RW
— (b4)	Reserved bit	Set to 0.	RW
— (b5)	Reserved bit	Set to 0 in other than user boot mode. Set to 1 in user boot mode.	RW
FMR06	Program status flag	0: Completed as expected 1: Completed in error	RO
— (b7)	Reserved bit	Read as undefined value.	RO

FMR01 (CPU rewrite mode select bit) (b1)

Commands can be accepted by setting the FMR01 bit to 1 (CPU rewrite mode enabled).
To set the FMR01 bit to 1, write 0 and then 1 in succession. Do not generate any interrupts or DMA transfers between setting 0 and 1.
Change the FMR01 bit when the PM24 bit in the PM2 register is 0 (NMI interrupt disabled) or high is input to the NMI pin.
While in EW0 mode, write to this bit from a program in an area other than flash memory.
Enter read array mode, and then set this bit to 0.

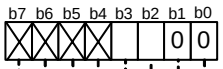
FMSTP (Flash memory stop bit) (b3)

The FMSTP bit resets the flash memory control circuits and minimizes current consumption in the flash memory. Access to the internal flash memory is disabled when the FMSTP bit is set to 1 (flash memory operation stopped). Set the FMSTP bit by a program located in an area other than the flash memory.
Set the FMSTP bit to 1 under the following condition:

- A flash memory access error occurs while erasing or programming in EW0 mode (the FMR00 bit does not revert to 1 (ready)).

9.2.2 Flash Memory Control Register 2 (FMR2)

Flash Memory Control Register 2



Symbol: FMR2 | Address: 0222h | After Reset: XXXX 0000b

Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved bits	Set to 0.	RW
FMR22	Slow read mode enable bit	0: Disabled 1: Enabled	RW
FMR23	Low current consumption read mode enable bit	0: Disabled 1: Enabled	RW
— (b7-b4)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

FMR22 (Slow read mode enable bit) (b2)

This bit enables mode which reduces the amount of current consumption when reading the flash memory. When rewriting the flash memory (CPU rewrite mode), set the FMR22 bit to 0 (slow read mode disabled).

To set the FMR22 bit to 1, write 0 and then 1 in succession. Make sure no interrupts or DMA transfers occur between writing 0 and 1.

Set the FMR23 bit to 1 (low current consumption read mode enabled) after the FMR22 bit is set to 1 (slow read mode enabled). Also, set the FMR22 bit to 0 (slow read mode disabled) after the FMR23 bit is set to 0 (slow read mode disabled). Do not change the FMR22 bit and FMR23 bit at the same time.

FMR23 (Low current consumption read mode enable bit) (b3)

This bit enables the mode which reduces the amount of current consumption when reading the flash memory. When rewriting the flash memory (CPU rewrite mode), set the FMR23 bit to 0 (low current consumption read mode disabled).

Low current consumption read mode can be used when the CM07 bit in the CM0 register is 1 (sub clock used as CPU clock).

To set the FMR23 bit to 1, write 0 and then 1 in succession. Make sure no interrupts or DMA transfers occur between writing 0 and 1.

Set the FMR23 bit to 1 (low current consumption read mode enabled) after the FMR22 bit is set to 1 (slow read mode enabled). Also, set the FMR22 bit to 0 (slow read mode disabled) after the FMR23 bit is set to 0 (low current consumption read mode disabled). Do not change bits FMR22 and FMR23 at the same time.

Do not set the FMR23 bit to 1 (low current consumption read mode enabled) when any of the following occurs:

- When the CM07 bit is 0 (main clock or on-chip oscillator clock selected as CPU clock source).
- When the FMR22 bit is 0 (slow read mode disabled)
- When the FMSTP bit is 1 (flash memory stopped)
- During the wake up operation when the FMSTP bit is changed from 1 to 0 (tps)

9.3 Clock

The amount of current consumption correlates with the number of operating clocks and frequency. If there are fewer operating clocks and a lower frequency, current consumption will be low.
Normal operating mode, wait mode, and stop mode are provided to control power consumption. All mode states, except wait mode and stop mode, are referred to as normal operating mode in this document.

9.3.1 Normal Operating Mode

In normal operating mode, because both the CPU clock and the peripheral function clocks are supplied, the CPU and the peripheral functions are operating. Power control is exercised by controlling the CPU clock frequency. The higher the CPU clock frequency, the higher the processing capability. The lower the CPU clock frequency, the lower the power consumption in the chip. If unnecessary oscillator circuits are stopped, power consumption is further reduced.

9.3.1.1 High-Speed Mode and Medium-Speed Mode

In high-speed mode, the main clock divided by 1 (no division) is used as the CPU clock.
In medium-speed mode, the main clock divided by 2, 4, 8 or 16 is used as the CPU clock.
f1 with the same frequency of the main clock divided by 1 is used as the peripheral function clocks in both high-speed and medium-speed modes. When fC is supplied, fC and fC32 can be used as the peripheral function clocks. When fOCO-S is supplied, it can be used as the peripheral function clocks.

9.3.1.2 PLL Operating Mode

The PLL clock divided by 1 (no division), 2, 4, 8 or 16 is used as the CPU clock. f1 with the same frequency of the PLL clock divided by 1 (no division) is used as the peripheral function clocks.
When fC is supplied, fC and fC32 can be used as the peripheral function clocks. When fOCO-S is supplied, it can be used as the peripheral function clocks.
PLL operating mode can be entered and exited from high-speed mode or medium-speed mode. To enter other modes including wait mode and stop mode, enter high-speed mode or medium-speed mode first, and then enter the intended mode (see Figure 9.1 “Clock Mode Transition”).

9.3.1.3 125 kHz On-Chip Oscillator Mode

The fOCO-S clock divided by 1 (no division), 2, 4, 8 or 16 is used as the CPU clock. f1 with the same frequency of the fOCO-S clock divided by 1 is used as the peripheral function clocks.
When fC is supplied, fC and fC32 can be used as the peripheral function clocks. fOCO-S can be used as the peripheral function clocks.

9.3.1.4 125 kHz On-Chip Oscillator Low Power Mode

The main clock is turned off after the MCU enters 125 kHz on-chip oscillator mode. The fOCO-S clock divided by 1 (no division), 2, 4, 8 or 16 is used as the CPU clock. f1 with the same frequency of the fOCO-S clock divided by 1 is used as the peripheral function clocks.
When fC is supplied, fC and fC32 can be used as the peripheral function clocks. fOCO-S can be used as the peripheral function clocks.

9.3.1.5 Low-Speed Mode

fC divided by 1 (no division) or 2 is used as the CPU clock.
When the CM21 bit is 0 and the CM11 bit is 0 (main clock), f1 with the same frequency of the main clock divided by 1 is used as the peripheral function clocks. When the CM21 bit is 0 and the CM11 bit is 1 (PLL clock), f1 with the same frequency of the PLL clock divided by 1 is used as the peripheral function clocks. When the CM21 bit is 1 (on-chip oscillator clock), f1 with the same frequency as the fOCO-S clock divided by 1 is used as the peripheral function clocks.
fC and fC32 can be used as the peripheral function clocks. When fOCO-S is supplied, it can be used as the peripheral function clocks.

9.3.1.6 Low Power Mode

The main clock is stopped after the MCU enters low-speed mode. fC divided by 1 (no division) or 2 is used as the CPU clock.
fC and fC32 can be used as the peripheral function clocks. When fOCO-S is supplied, it can be used as the peripheral function clocks.

Table 9.2 Clocks in Normal Operating Mode

Mode	CPU Clock	Peripheral Clocks (2)		
		f1	fC, fC32	fOCO-S
High-speed mode	Main clock divided by 1 (1)	Main clock divided by 1	Enabled	Enabled
Medium-speed mode	Main clock divided by n (1)			
PLL operating mode	PLL clock divided by n (1)	PLL clock divided by 1	Enabled	Enabled
125 kHz on-chip oscillator mode	fOCO-S divided by n (1)	fOCO-S divided by 1	Enabled	Enabled
125 kHz on-chip oscillator low power mode	fOCO-S divided by n (1)	fOCO-S divided by 1	Enabled	Enabled
Low-speed mode	fC divided by n	Any of the following: Main clock divided by 1 (when the CM21 is 0 and the CM11 is 0) PLL clock divided by 1 (when the CM21 is 0 and the CM11 is 0) fOCO-S divided by 1 (when the CM21 is 1)	Enabled	Enabled
Low power mode	fC divided by n	fOCO-S divided by 1 (when the CM21 is 1)	Enabled	Enabled

CM11 : Bit in the CM1 register
CM21 : Bit in the CM2 register
Notes:
1. Select by setting the CM06 bit in the CM0 register and bits CM17 to CM16 in the CM1 register.
2. The peripheral clock is enabled when each clock is supplied. Refer to 8. “Clock Generator” for the clock supply method.

Table 9.3 Clock-Related Bit Setting and Modes

Mode	CM2 Register	CM1 Register		CM0 Register			
	CM21	CM14	CM11	CM07	CM05	CM04	CM03
High-speed mode, medium-speed mode	0	—	0	0	0	—	—
PLL operating mode	0	—	1	0	0	—	—
125 kHz on-chip oscillator mode	1	0	0	0	0 (1)	—	—
125 kHz on-chip oscillator low power mode	1	0	0	0	1	—	—
Low-speed mode	—	—	0	1	0 (1)	1	0
Low power mode	—	—	0	1	1	1	0

Note:
1. The main clock is oscillated.

Table 9.4 Selecting Clock Division Related Bits (1)

Division	CM1 Register	CM0 Register
	Bits CM17 to CM16	CM16 bit
No division (2)	00b	0
Divide-by-2	01b	0
Divide-by-4	10b	0
Divide-by-8	-	1
Divide-by-16	11b	0

Notes:
1. While in high-speed mode, medium-speed mode, PLL operating mode, 125 kHz on-chip oscillator mode, or 125 kHz on-chip oscillator low power mode.
2. Select divide-by-1 (no division) in high-speed mode.

Table 9.5 Example Settings for Low-Speed Mode and Low-Power Mode Division Related Bits

Division	SCM0 Register
	SCM01 and SCM00
No division	00b
Divide-by-2	01b

9.3.2 Clock Mode Transition Procedure

Figure 9.1 shows Clock Mode Transition. Arrows indicate possible mode transitions.

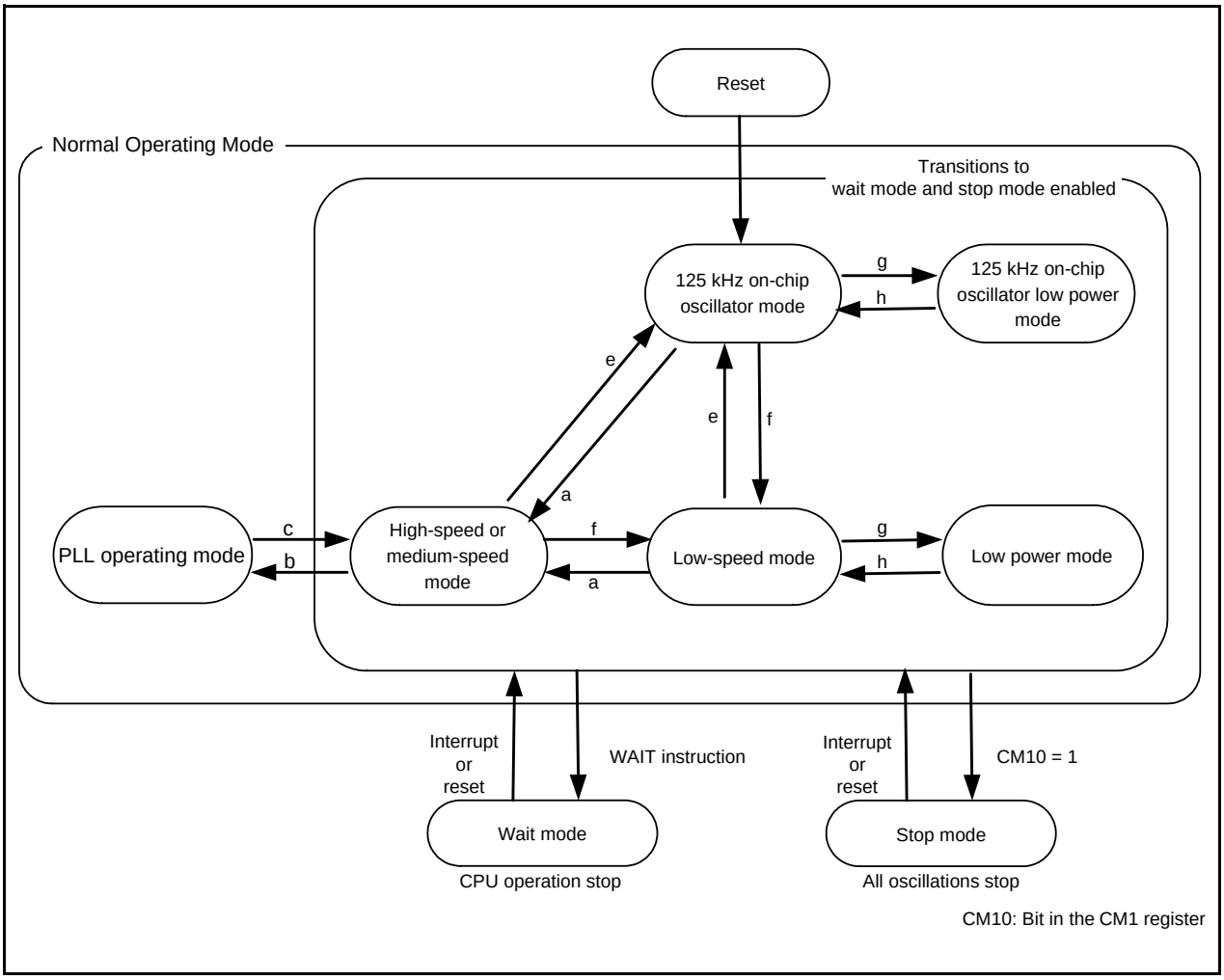


Figure 9.1 Clock Mode Transition

To start or stop clock oscillations, or to change modes in normal operating mode, follow the instructions below.

- Enter a different mode after the clock for that mode stabilizes completely.
- When stopping a clock, do it after mode transition is completed. Do not stop the clock at the same time as mode transition.
- To change the mode, follow procedures a to c and e to h listed below. To access registers and bits, refer to 9.2 “Registers”. Letters a to c and e to h correspond to those in Figure 9.1 “Clock Mode Transition”.
- For oscillator start and stop, refer to 8.3.1 “Main Clock” to 8.3.4 “Sub Clock (fC)”.
- When entering a new mode from PLL operating mode, high-speed or medium-speed mode, or 125 kHz on-chip oscillator mode, or entering one of these modes from another mode, select divide by 8 or divide by 16.
- When the clock division ratio is switched in PLL operating mode, or high-speed or medium-speed mode, the ratio changes in the order shown in Figure 9.2.

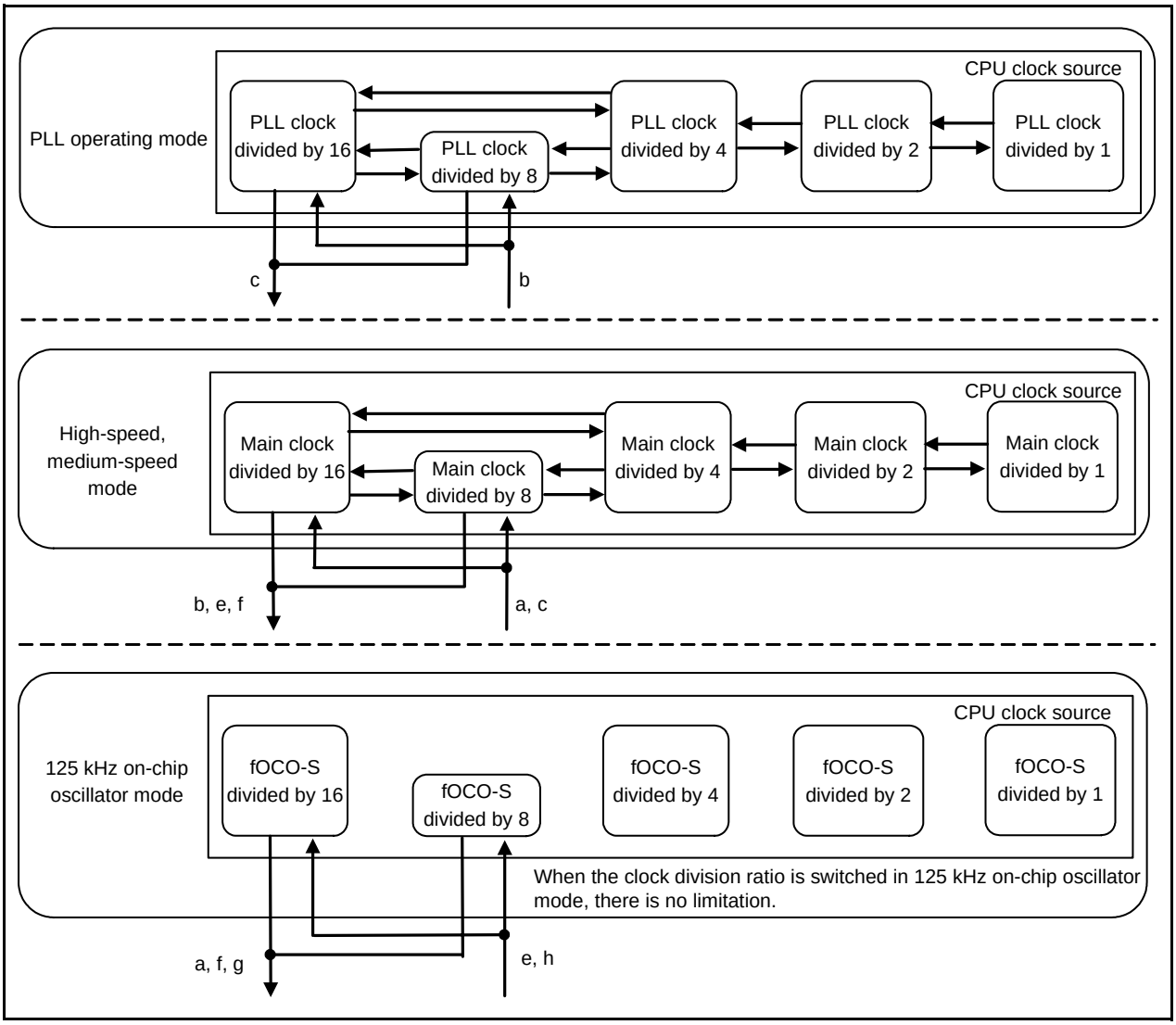


Figure 9.2 Clock Divide Transition

a. Entering high-speed mode or medium-speed mode from 125 kHz on-chip oscillator mode or low-speed mode
   (1) Select the main clock and wait until the oscillation stabilizes. Refer to 8.3.1 “Main Clock”.
   (2) Set the CM06 bit to 1 (divide-by-8 mode).
   (3) Set the CM21 bit to 0 and the CM07 bit to 0 (main clock selected as CPU clock source).

b. Entering PLL operating mode from high-speed mode or medium-speed mode
   (1) Set a multiplying factor and reference frequency counter by using bits PLC05 to PLC04 and bits PLC02 to PLC00 in the PLC0 register.
   (2) Set the PLC07 bit to 1 (PLL on).
   (3) Wait for tsu(PLL) until the PLL clock stabilizes.
   (4) Select divide-by-8 or divide-by-16 mode by setting the CM06 bit and bits CM17 to CM16.
   (5) Set the CM11 bit to 1, the CM21 bit to 0, and the CM07 bit to 0 (PLL clock selected as CPU clock source).

c. Entering high-speed mode or medium-speed mode from PLL operating mode
   (1) Select divide-by-8 or divide-by-16 mode by setting the CM06 bit and bits CM17 to CM16.
   (2) Set the CM11 bit to 0, the CM21 bit to 0, and the CM07 bit to 0 (main clock selected as CPU clock source).
   (3) Set the PLC07 bit to 0 (PLL off).

e. Entering 125 kHz on-chip oscillator mode from high-speed mode, medium-speed mode, or low-speed mode
   (1) Select the 125 kHz on-chip oscillator and wait until the oscillation stabilizes. Refer to 8.3.3 “125 kHz On-Chip Oscillator Clock (fOCO-S)”.
   (2) Set the CM21 bit to 1 (on-chip oscillator clock selected as CPU clock source).
   (3) Set the CM07 bit to 0 (main clock, PLL clock, or on-chip oscillator clock selected as CPU clock source).

f. Entering low-speed mode from high-speed mode, medium-speed mode, or 125 kHz on-chip oscillator mode
   (1) Select the sub clock and wait until the oscillation stabilizes. Refer to 8.3.4 “Sub Clock (fC)”.
   (2) Select the division ratio by bits SCM10 to SCM00 in the SCM0 register.
   (3) Set the CM07 bit to 1 (sub clock selected as CPU clock source).

g. Entering 125 kHz on-chip oscillator low power mode from 125 kHz on-chip oscillator mode.
   Entering low power mode from low-speed mode.
   Follow the procedure below.
   • Stop the main clock. Refer to 8.3.1 “Main Clock”.

h. Entering 125 kHz on-chip oscillator mode from 125 kHz on-chip oscillator low power mode.
   Entering low-speed mode from low power mode.
   Follow both or either of the procedures below (in no particular order).
   • Select the main clock and wait until the oscillation stabilizes. Refer to 8.3.1 “Main Clock”.

9.3.3 Wait Mode

In wait mode, the CPU clock, CPU, watchdog timer, and $\overline{\text{NMI}}$ digital filter are turned off as they are operated by the CPU clock. However, if the CSPRO bit in the CSPR register is 1 (count source protection enabled), the watchdog timer remains active. Because the clock generator does not stop, peripheral functions supplied by a peripheral clock keep operating.

9.3.3.1 Peripheral Function Clock Stop Function

When the CM02 bit is 1 (peripheral function clock f1 turned off during wait mode), the f1 clock is turned off while in wait mode, and power consumption is reduced. However, all the peripheral clocks except f1 (i.e. fOCO-S, fC, and fC32) do not stop.

9.3.3.2 Entering Wait Mode

The MCU enters wait mode by executing a WAIT instruction.
When the CM11 bit is 1 (PLL clock selected as CPU clock source), set the CM11 bit to 0 (main clock selected as CPU clock source) before entering wait mode. Chip power consumption can be reduced by setting the PLC07 bit to 0 (PLL off).

When using wait mode, set the following:

(1) Set the I flag to 0.
(2) Set the interrupt priority level of bits ILVL2 to ILVL0 in the interrupt control register for the peripheral function interrupt which is used to exit wait mode. Start the peripheral function which is used to exit wait mode if it is stopped.
(3) Set 000b (interrupt disabled) to bits ILVL2 to ILVL0 in the interrupt control registers for the peripheral function interrupts not used to exit wait mode.
(When using any of the following resets or interrupts to exit wait mode, set 000b to bits ILVL2 to ILVL0 in all interrupt control registers for peripheral function interrupts: hardware reset, voltage monitor 0 reset, voltage monitor 1 reset, voltage monitor 2 reset, watchdog timer reset, $\overline{\text{NMI}}$ interrupt, voltage monitor 1 interrupt, or voltage monitor 2 interrupt).
(4) Set the I flag to 1.
(5) Execute the WAIT instruction.

9.3.3.3 Pin Status in Wait Mode

Table 9.6 lists Pin Status in Wait Mode.

Table 9.6 Pin Status in Wait Mode

Pin		Single-Chip Mode
I/O ports		Retains the status just prior to entering wait mode
CLKOUT	fC selected	Does not stop
	f1, f8, f32 selected	Does not stop when the CM02 bit is 0. When the CM02 bit is 1, the status immediately prior to entering wait mode is retained.

9.3.3.4 Exiting Wait Mode

The MCU exits wait mode by a reset or interrupt. Table 9.7 lists Resets and Interrupts to Exit Wait Mode and Conditions for Use.

The peripheral function interrupts are affected by the CM02 bit. When the CM02 bit is 0 (peripheral function clock f1 not turned off in wait mode), peripheral function interrupts can be used to exit wait mode. When the CM02 bit is 1 (peripheral function clock f1 turned off in wait mode), the peripheral functions using the peripheral function clock f1 stop operating, so that the peripheral functions activated by external signals and the peripheral function clocks except f1 (fOCO-S, fC, fC32) can be used to exit wait mode.

fOCO-S is also used for the digital filter in the voltage detector, so the MCU exits wait mode when the digital filter is disabled or when fOCO-S is supplied.

Table 9.7 Resets and Interrupts to Exit Wait Mode and Conditions for Use

Interrupt, Reset			Conditions for Use	
			CM02 = 0	CM02 = 1
Interrupt	Peripheral function interrupt	$\overline{\text{INT}}$	Usable	Usable
		Key input	Usable	Usable
		Timer A, timer B	Usable in all modes	Usable when fOCO-S or fC32 is supplied and is used as count source. Usable when counting external signals in event counter mode.
		Serial interface	Usable in external clock	Usable in external clock
		Multi-master I^2C-bus interface	Both I^2C-bus interface interrupt and SCL/SDA interrupt are usable	SCL/SDA interrupt is usable
		Real-time clock	Usable when fC is supplied	
	$\overline{\text{NMI}}$		Usable when the digital filter is disabled (bits NMIDF2 to NMIDF0 in the NMIDF register are 000b)	
Reset	Hardware reset		Usable	
	Voltage detection 0 reset		Usable when fOCO-S is supplied or digital filter is disabled	
	Watchdog timer		Usable when count source protection mode is enabled (the CSPRO bit in the CSPR register is 1).	

When exiting wait mode by means of an interrupt, an interrupt routine is performed after an interrupt request is generated, and then the CPU clock is supplied again.

When the MCU exits wait mode by an interrupt, the CPU clock is the same CPU clock used while executing the WAIT instruction.

9.3.4 Stop Mode

In stop mode, all oscillator circuits, the CPU clock, and peripheral function clocks are stopped. Therefore, the CPU and the peripheral functions using these clocks stop operating. The least amount of power is consumed in this mode. If the voltage applied to pins VCC1 and VCC2 is VRAM or greater, the contents of internal RAM are retained. When applying 2.7 V or less to pins VCC1 and VCC2, make sure VCC1 = VCC2 ≥ VRAM.

However, the peripheral functions activated by external signals keep operating.

9.3.4.1 Entering Stop Mode

The MCU enters stop mode by setting the CM10 bit in the CM1 register to 1 (all clocks turned off). At the same time, the CM06 bit in the CM0 register becomes 1 (divide-by-8 mode), and the CM15 bit in the CM1 register becomes 1 (main clock oscillator circuit drive capability high).

Before entering stop mode, set the CM20 bit to 0 (oscillator stop/restart detect function disabled).

Also, when the CM11 bit is 1 (PLL clock used as the CPU clock source), set the CM11 bit to 0 (main clock used as the CPU clock source), and then the PLC07 bit to 0 (PLL turned off) before entering stop mode.

When using stop mode, set the following:

(1) Set the I flag to 0.

(2) Set the interrupt priority level of bits ILVL2 to ILVL0 in the interrupt control register for the peripheral function interrupt which is used to stop mode. Start the peripheral function which is used to stop mode if it is stopped.

(3) Set 000b (interrupt disabled) to bits ILVL2 to ILVL0 in the interrupt control registers for the peripheral function interrupts not used to exit stop mode.
(When using any of the following resets or interrupts to exit stop mode, set 000b to bits ILVL2 to ILVL0 in all interrupt control registers for peripheral function interrupts: hardware reset, voltage monitor 0 reset, $\overline{\text{NMI}}$ interrupt, voltage monitor 1 interrupt, or voltage monitor 2 interrupt)

(4) Set the I flag to 1.

(5) Set the CM10 bit in the CM1 register to 1.

9.3.4.2 Pin Status in Stop Mode

Table 9.8 lists Pin Status in Stop Mode.

Table 9.8 Pin Status in Stop Mode

Pin		Single-Chip Mode
I/O ports		Retains status just prior to stop mode
CLKOUT	f1, f8, f32, fC selected	Retains status just prior to stop mode
XOUT		High
XCIN, XCOUT		High-impedance

9.3.4.3 Exiting Stop Mode

Use a reset or an interrupt to exit stop mode. Table 9.9 lists Resets and Interrupts to Exit Stop Mode and Conditions for Use.

Table 9.9 Resets and Interrupts to Exit Stop Mode and Conditions for Use

Interrupt, Reset			Conditions for Use
Interrupt	Peripheral function interrupt	$\overline{\text{INT}}$	Usable
		Key input	Usable
		Timer A, timer B	Usable when counting external signals in event counter mode
		Serial interface	Usable when an external clock is selected
		Multi-master I^2C-bus interface	SCL/SDA interrupt is usable
	$\overline{\text{NMI}}$		Usable when the digital filter is disabled (bits NMIDF2 to NMIDF0 in the NMIDF register are 000b)
Reset	Hardware reset		Usable
	Voltage detection 0 reset		Usable when the digital filter is disabled (VW0C1 bit in the VW0C register is 1)

To exit stop mode by using hardware reset, voltage monitor 0 reset, $\overline{\text{NMI}}$ interrupt, set bits ILVL2 to ILVL0 in the interrupt control registers for the peripheral function interrupt to 000b (interrupt disabled) before setting the CM10 bit to 1.

When exiting stop mode by means of a peripheral function interrupt, an interrupt routine is performed after an interrupt request is generated and then the CPU clock is supplied again.
When stop mode is exited by means of an interrupt, the CPU clock source varies depending on the CPU clock source setting before the MCU had entered stop mode. Table 9.10 lists CPU Clock After Exiting Stop Mode.

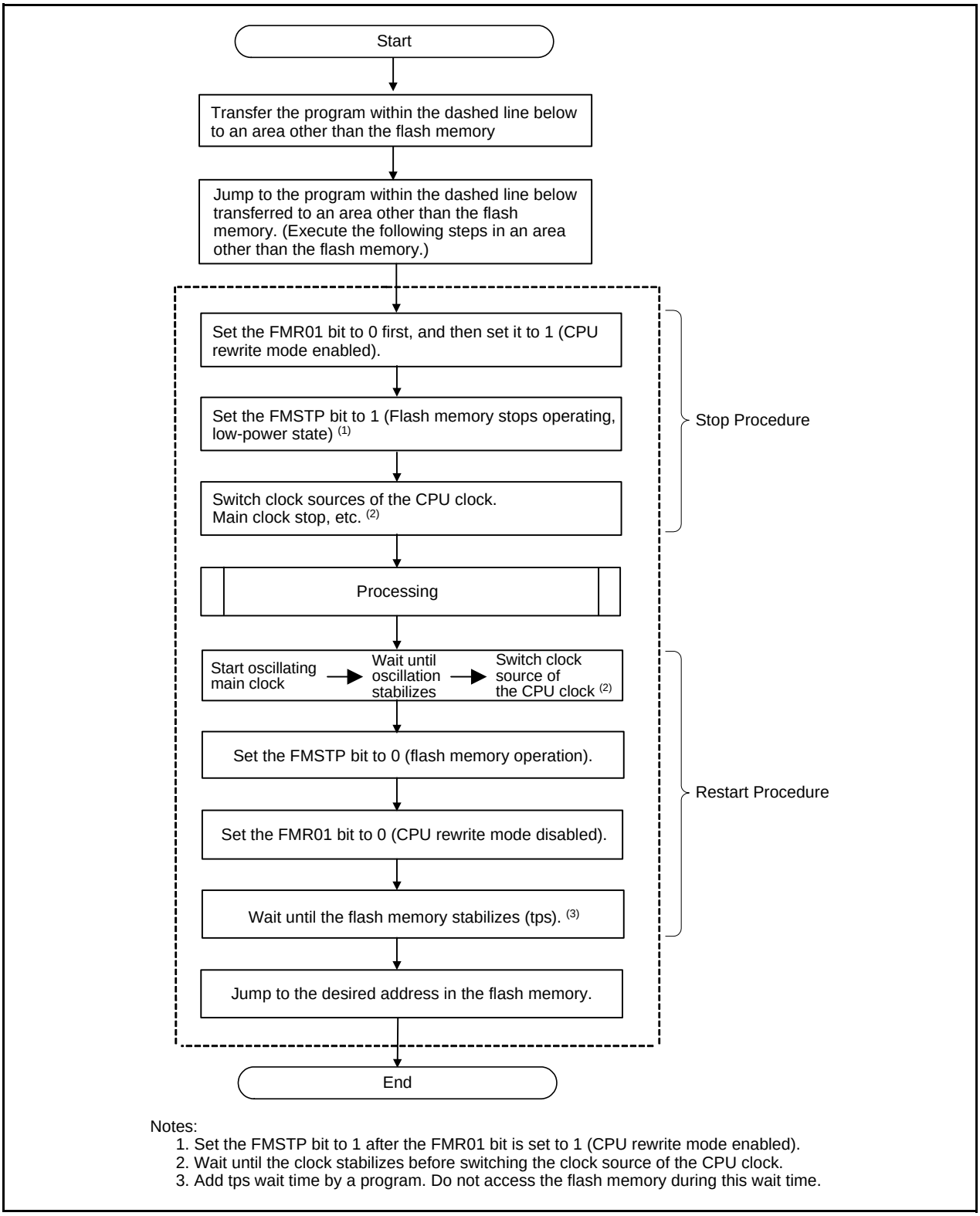
Table 9.10 CPU Clock After Exiting Stop Mode

CPU Clock Before Entering Stop Mode	CPU Clock After Exiting Stop Mode
Main clock divided by 1 (no division), 2, 4, 8 or 16	Main clock divided by 8
fOCO-S divided by 1 (no division), 2, 4, 8 or 16	fOCO-S divided by 8
fC divided by 1 (no division), 2, or 4	fC divided by 1 (no division), 2, or 4

9.4 Power Control in Flash Memory

9.4.1 Stopping Flash Memory

When the flash memory is stopped, current consumption is reduced. Execute a program in any area other than the flash memory. Figure 9.3 shows Stop and Restart of the Flash Memory. Follow the flowchart of Figure 9.3.



Notes:
1. Set the FMSTP bit to 1 after the FMR01 bit is set to 1 (CPU rewrite mode enabled).
2. Wait until the clock stabilizes before switching the clock source of the CPU clock.
3. Add tps wait time by a program. Do not access the flash memory during this wait time.

Figure 9.3 Stop and Restart of the Flash Memory

9.4.2 Reading Flash Memory

Current consumption while reading the flash memory can be reduced by using bits FMR22 and FMR23.

9.4.2.1 Slow Read Mode

Slow read mode can be used when f(BCLK) is below or equal to f(SLOW_R). Figure 9.4 shows Setting and Canceling Slow Read Mode.

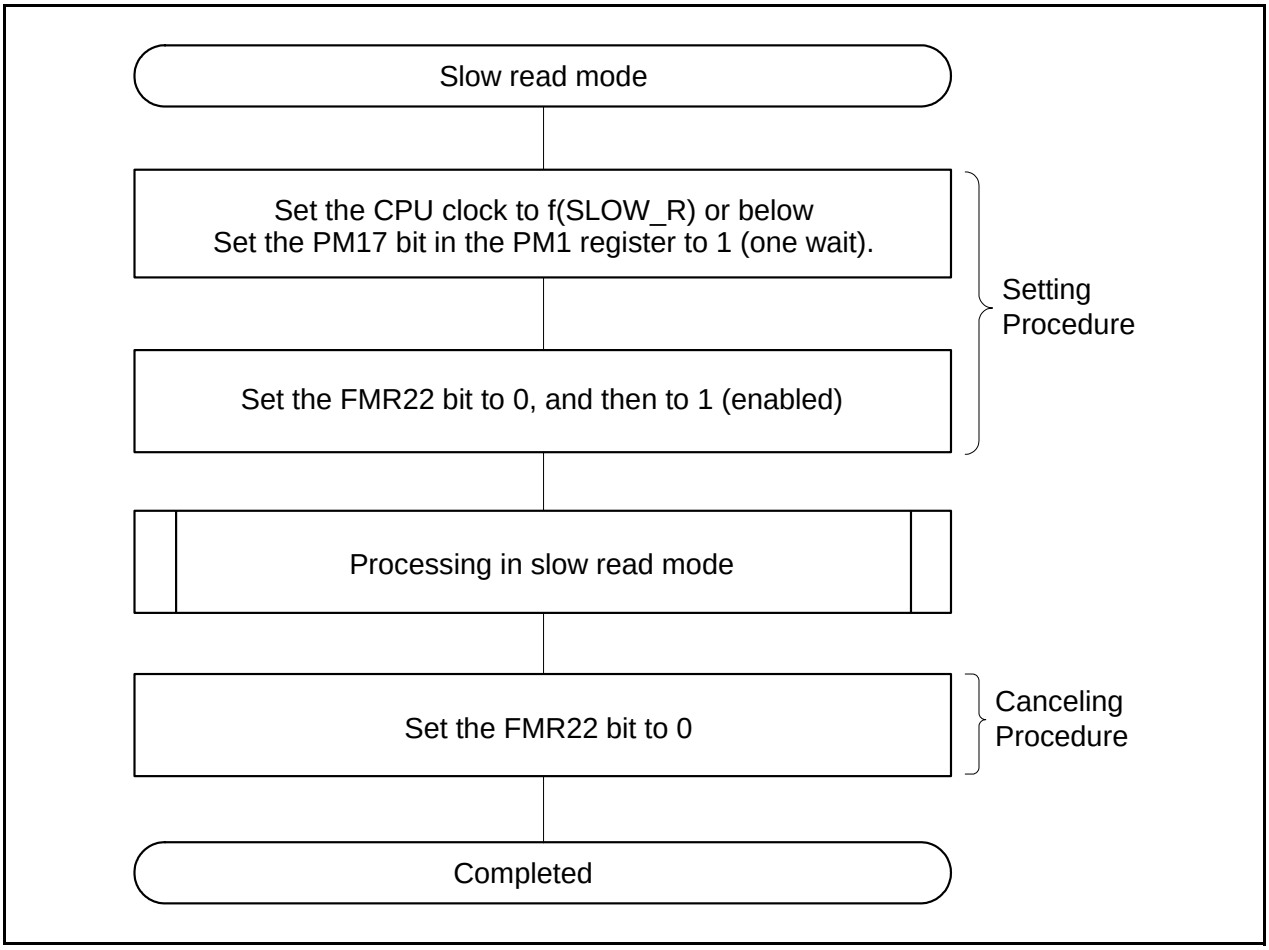


Figure 9.4 Setting and Canceling Slow Read Mode

9.4.2.2 Low Current Consumption Read Mode

Low current consumption read mode can be used when the CM07 bit in the CM0 register is 1 (sub clock used as CPU clock). Figure 9.5 shows Setting and Canceling Low Current Consumption Read Mode.

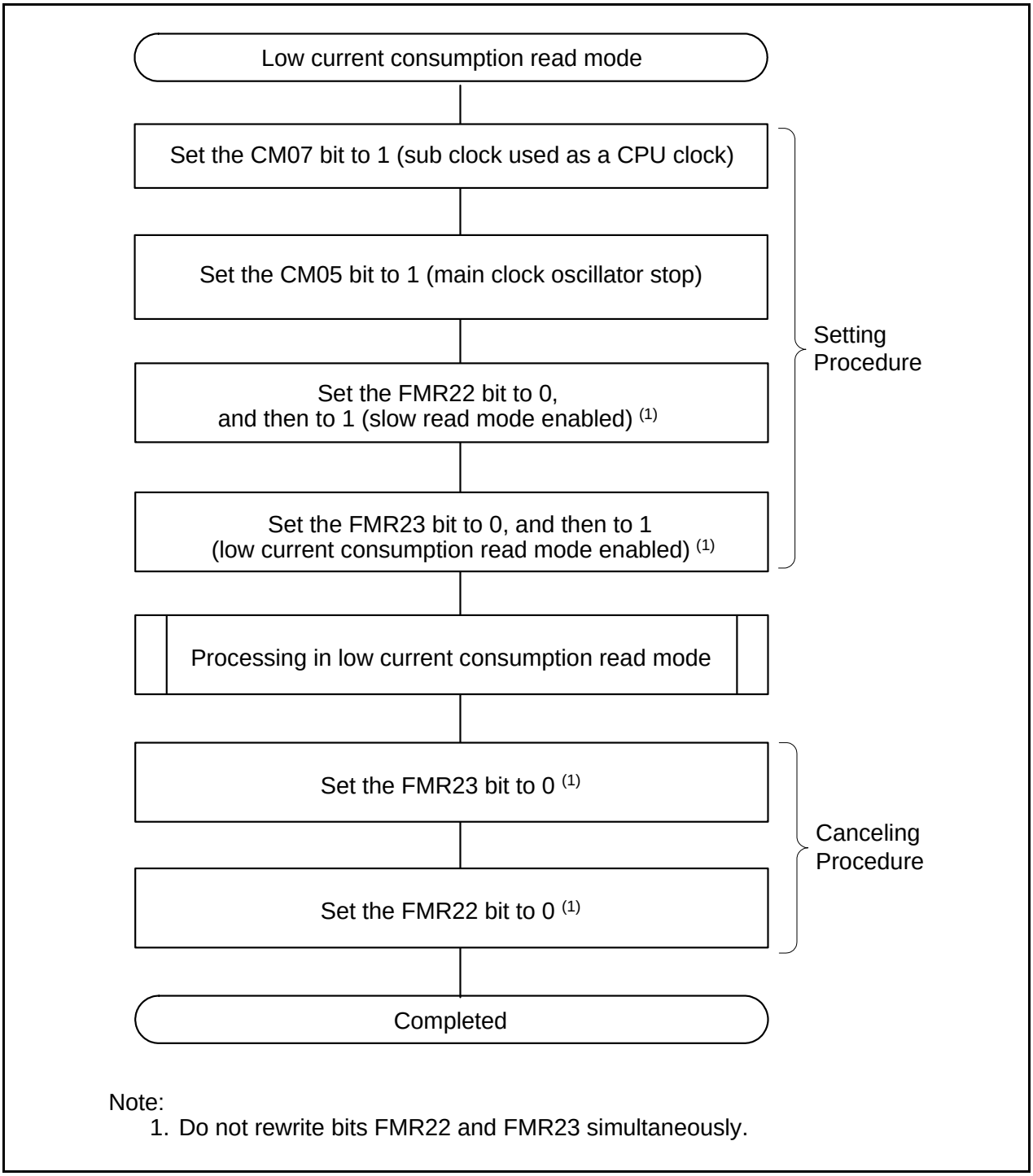


Figure 9.5 Setting and Canceling Low Current Consumption Read Mode

9.5 Reducing Power Consumption

To reduce power consumption, refer to the following descriptions when designing a system or writing a program.

9.5.1 Ports

The MCU retains the state of each I/O port even when it enters wait mode or stop mode. A current flows in the active output ports. A shoot-through current flows to the input ports in the high-impedance state. When entering wait mode or stop mode, set unused ports to input and stabilize the potential.

9.5.2 A/D Converter

When not performing A/D conversion, set the ADSTBY bit in the ADCON1 register to 0 (A/D operation stop).

9.5.3 Stopping Peripheral Functions

Use the PCLKSTP1 register to stop providing f1 to the peripheral functions not using f1.
Use the CM02 bit in the CM0 register to stop the unnecessary peripheral functions while in wait mode.

9.5.4 Switching the Oscillation-Driving Capacity

Set the driving capacity to low when oscillation is stable.

9.6 Notes on Power Control

9.6.1 CPU Clock

When switching the CPU clock source, wait until oscillation of the switched clock source is stable. After exiting stop mode, wait until oscillation stabilizes before changing the division.

9.6.2 Wait Mode

- Insert four or more NOP instructions following the WAIT instruction. When entering wait mode, because the instruction queue prefetches instructions that follow the WAIT instruction, prefetched instructions are sometimes executed prior to the interrupt routine used to exit wait mode. As shown below, when the instruction to set the I flag to 1 is allocated just before the WAIT instruction, interrupt requests are not accepted before the WAIT instruction is executed.

  The following is an example program for entering wait mode:

```
Program Example:  FSET   I          ;
                  WAIT              ; Enter wait mode
                  NOP               ; Insert at least four NOP instructions
                  NOP
                  NOP
                  NOP
```

- Do not enter wait mode from PLL operating mode. To enter wait mode from PLL operating mode, first enter medium-speed mode, then set the PLC07 bit to 0 (PLL off).
- Do not enter wait mode from low current consumption read mode. To enter wait mode from low current consumption read mode, set the FMR23 bit in the FMR2 register to 0 (low current consumption read mode disabled).
- Do not enter wait mode from CPU rewrite mode. To enter wait mode from CPU rewrite mode, first set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled), then disable the DMA transfer.
- Set the PLC07 bit in the PLC0 register to 0 (PLL off). When the PLC07 bit is 1 (PLL on), current consumption cannot be reduced even in wait mode.

9.6.3 Stop Mode

- When exiting stop mode by a hardware reset, drive the $\overline{\text{RESET}}$ pin low for 20 fOCO-S cycles or more.
- Set the MR0 bit in the TAiMR register (i = 0 to 4) to 0 (pulse not output) when using timer A to exit stop mode.
- When entering stop mode, insert a JMP.B instruction immediately after executing an instruction that sets the CM10 bit in the CM1 register to 1 (stop mode), and then insert at least four NOP instructions. When entering stop mode, the instruction queue reads ahead the instructions following the instruction which sets the CM10 bit to 1. Thus, some of the instructions may be executed before the MCU enters stop mode or before the interrupt routine for returning from stop mode. As shown below, when the instruction to set the I flag to 1 is allocated just before the instruction to set the CM10 bit to 1, interrupt requests are not accepted before entering stop mode.

  The following is an example program for entering stop mode:

```
Program Example:  FSET    I
                  BSET    0, CM1   ; Enter stop mode
                  JMP.B   L2       ; Insert a JMP.B instruction
          L2:
                  NOP              ; At least four NOP instructions
                  NOP
                  NOP
                  NOP
```

- Do not enter stop mode from PLL operating mode. To enter stop mode from PLL operating mode, first enter medium-speed mode, then set the PLC07 bit to 0 (PLL off).
- Do not enter stop mode from low current consumption read mode. To enter stop mode from low current consumption read mode, set the FMR23 bit in the FMR2 register to 0 (low current consumption read mode disabled).
- Do not enter stop mode from CPU rewrite mode. To enter stop mode from CPU rewrite mode, first set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled), then disable the DMA transfer.
- Do not enter stop mode when the oscillator stop/restart detect function is enabled. To enter stop mode, set the CM20 bit in the CM2 register to 0 (oscillator stop/restart detect function disabled).
- Entering stop mode is disabled when the FMR01 bit is 1 (CPU rewrite mode enabled). Therefore, do not enter stop mode when the flash memory is stopped (bits FMR01 and FMSTP are 1).

9.6.4 Low Current Consumption Read Mode

- Enter low current consumption read mode through slow read mode (see Figure 9.5 “Setting and Canceling Low Current Consumption Read Mode”).
- When the FMR23 bit in the FMR2 register is 1 (low current consumption read mode enabled), do not set the FMSTP bit to 1 (flash memory stopped). Also, when the FMSTP bit is 1, do not set the FMR23 bit to 1.
- When the FMR01 bit in the FMR0 register to 1 (CPU rewrite mode enabled), do not set the FMR23 bit in the FMR2 register to 1 (low current consumption read mode enable).

9.6.5 Slow Read Mode

When the FMR01 bit in the FMR0 register to 1 (CPU rewrite mode enabled), do not set the FMR22 bit in the FMR2 register to 1 (slow read mode enabled).

10. Processor Mode

10.1 Introduction

Single-chip mode can be selected.

Table 10.1 Processor Mode Features

Processor Mode	Access Space	Pins Assigned as I/O Ports
Single-chip mode	SFR, internal RAM, internal ROM	All pins are I/O ports or peripheral function I/O pins

Table 10.2 I/O Pins

Pin Name	I/O	Function
CNVSS	Input	Applies a low level

10.2 Registers

Table 10.3 Registers

Address	Register	Symbol	Reset Value
0004h	Processor Mode Register 0	PM0	0000 0000b
0005h	Processor Mode Register 1	PM1	0000 1000b
0010h	Program 2 Area Control Register	PRG2C	XXXX XX00b

10.2.1 Processor Mode Register 0 (PM0)

Processor Mode Register 0

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

Symbol	Address	After Reset
PM0	0004h	0000 0000b

Bit Symbol	Bit Name	Function	RW
PM00	Processor mode bit	b1 b0 0 0: Single-chip mode 0 1: Do not set 1 0: Do not set 1 1: Do not set	RW
PM01			RW
— (b2)	Reserved bit	Set to 0.	RW
PM03	Software reset bit	Setting this bit to 1 resets the MCU. When read, the read value is 0.	RW
— (b7-b4)	Reserved bits	Set to 0.	RW

Rewrite this register after setting the PRC1 bit in the PRCR register to 1 (write enabled).

PM01 to PM00 (Processor mode bit) (b1 to b0)

Set these bits to 00.

10.2.2 Processor Mode Register 1 (PM1)

Processor Mode Register 1

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

Symbol	Address	After Reset
PM1	0005h	0000 1000b

Bit Symbol	Bit Name	Function	RW
PM10	Data flash enable bit	0: Disabled 1: Enabled	RW
— (b1)	Reserved bit	Set to 0.	RW
PM12	Watchdog timer function select bit	0: Watchdog timer interrupt 1: Watchdog timer reset	RW
— (b3)	Reserved bit	Set to 1.	RW
— (b6-b4)	Reserved bits	Set to 0.	RW
PM17	Wait bit	0: No wait state 1: Wait state (1 wait)	RW

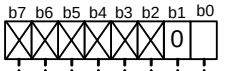
Rewrite this register after setting the PRC1 bit in the PRCR register to 1 (write enabled).
The PM12 bit becomes 1 by setting it to 1. Setting it to 0 has no effect.

PM10 (Data flash enable bit) (b0)

This bit is used to map data flash to addresses 0E000h to 0FFFFh.
Data flash includes block A (addresses 0E000h to 0EFFFh) and block B (addresses 0F000h to 0FFFFh). When data flash is enabled by the setting of the PM10 bit, both block A and block B can be used.
The PM10 bit becomes 1 while the FMR01 bit in the FMR0 register is 1 (CPU rewrite mode).

10.2.3 Program 2 Area Control Register (PRG2C)

Program 2 Area Control Register



Symbol	Address	After Reset
PRG2C	0010h	XXXX XX00b

Bit Symbol	Bit Name	Function	RW
PRG2C0	Program ROM 2 disable bit	0: Enable program ROM 2 1: Disable program ROM 2	RW
(b1)	Reserved bit	Set to 0.	RW
— (b7-b2)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

Rewrite this register after setting the PRC6 bit in the PRCR register to 1 (write enabled).

PRG2C0 (Program ROM 2 disable bit) (b0)

This bit is used to map program ROM 2 to addresses 10000h to 13FFFh.

11. Bus

11.1 Introduction

The following describes the internal bus in the MCU.

Table 11.1 Bus Specifications

Item	Specification
Internal bus	• Separate bus • 16-bit data bus width • 0 or 1 software waits can be inserted

11.2 Registers

Table 11.2 lists bus related registers. Refer to 10. “Processor Mode” for registers PM0 and PM1. Refer to 24. “Flash Memory” for the FMR1 register.

Table 11.2 Registers

Address	Register	Symbol	Reset Value
0004h	Processor Mode Register 0	PM0	0000 0000b
0005h	Processor Mode Register 1	PM1	0000 1000b
0221h	Flash Memory Control Register 1	FMR1	00X0 XX0Xb

11.3 Operations

11.3.1 Internal Bus

11.3.1.1 Reference Clock

The internal bus operate based on the BCLK. However, the area accessed and wait states affect bus operation. Refer to 11.3.1.3 “Software Wait States of the Internal Bus” for details.

11.3.1.2 Bus Hold

The internal bus is in a hold state under the following conditions:

- Rewriting the flash memory in EW1 mode while auto-programming or auto-erasing

When the bus is in hold state, the following occur:

- CPU is stopped
- DMAC is stopped
- Watchdog timer is stopped when the CSPRO bit in the CSPR register is 0 (count source protection mode disabled)

Bus use priority is given to bus hold, DMAC, and CPU in descending order. However, if the CPU is accessing an odd address in word units, DMAC cannot gain control of the bus between two separate accesses.

Bus Hold > DMAC > CPU

Figure 11.1 Bus Use Priority

11.3.1.3 Software Wait States of the Internal Bus

The PM17 bit in the PM1 register, which is a software-wait-related bit, affects both the internal memory and the external area. Table 11.3 lists Bits and Bus Cycles Related to Software Wait States (SFR and Internal Memory).
The data flash of the internal ROM is affected by both the PM17 bit in the PM1 register and the FMR17 bit in the FMR1 register.

Table 11.3 Bits and Bus Cycles Related to Software Wait States (SFR and Internal Memory)

Area		Setting of Software-Wait-Related Bits		Software Wait States	Bus Cycle
		FMR1 register FMR17 bit	PM1 register PM17 bit		
SFR		0 or 1	0 or 1	1	2 BCLK cycles (1)
Internal RAM		0 or 1	0	None	1 BCLK cycle (1)
			1	1	2 BCLK cycles
Internal ROM	Program ROM 1 Program ROM 2	0 or 1	0	None	1 BCLK cycle (1)
			1	1	2 BCLK cycles
	Data flash	0	0 or 1	1	2 BCLK cycles (1)
		1	0	None	1 BCLK cycle
			1	1	2 BCLK cycle

Note:
1. Status after reset.

11.4 Notes on Bus

11.4.1 Reading Data Flash

When 2.7 V ≤ VCC1 ≤ 3.0 V, one wait must be inserted to read the data flash. Use the PM17 bit or the FMR17 bit to insert one wait.

12. Programmable I/O Ports

Note

P2 to P5 have no external connections. These ports are connected to PLC modem internally.

12.1 Introduction

Table 12.1 lists Programmable I/O Ports Specifications (hereafter referred to as I/O ports).
Each pin functions as an I/O port, a peripheral function input/output.
To set peripheral functions, refer to the description for the individual function. To use ports as peripheral function input/output pins, refer to 12.4 “Peripheral Function I/O”.

Table 12.1 Programmable I/O Ports Specifications

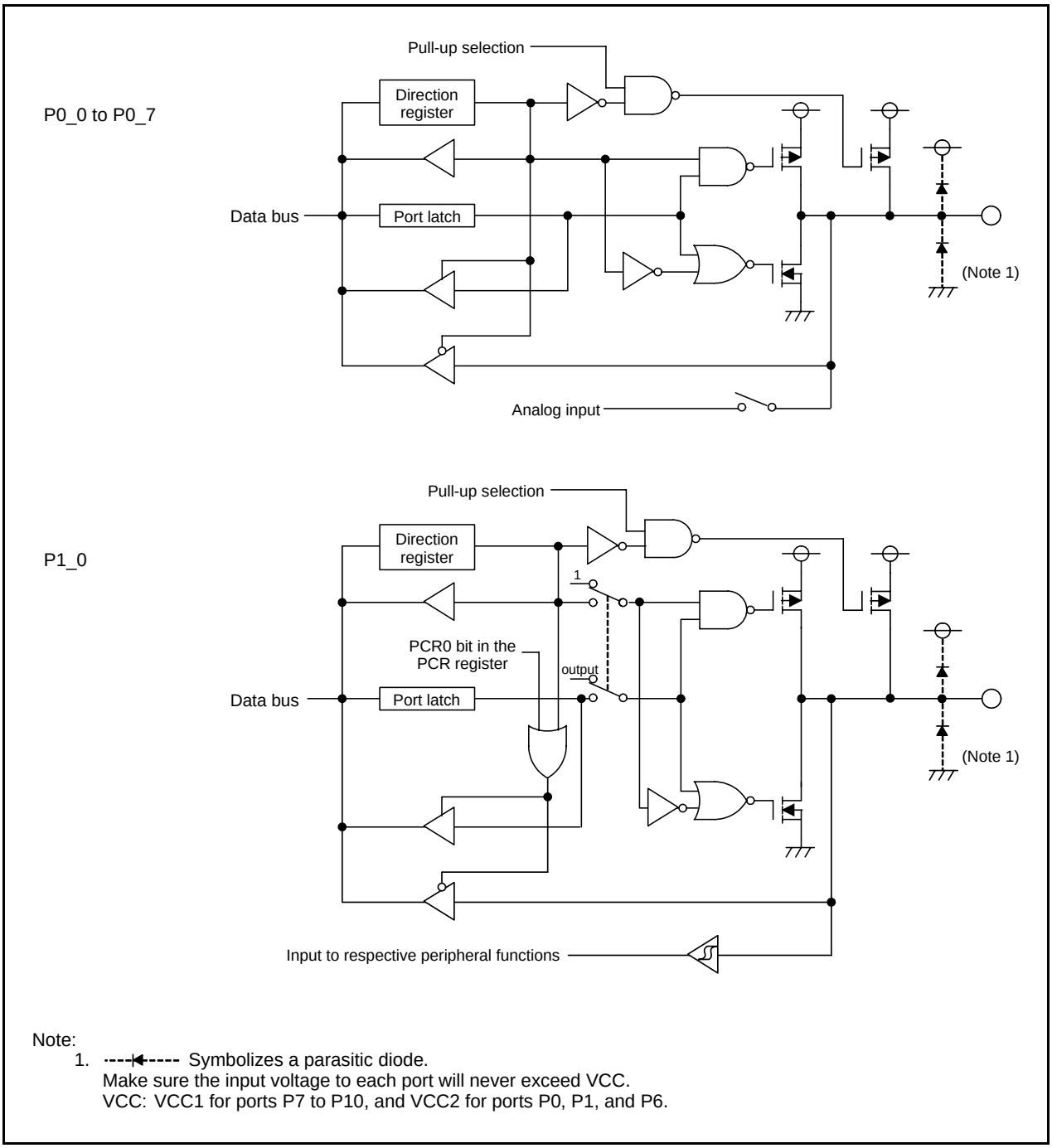
Item		Specification
Number of ports	Total	56
	CMOS output	53
	N-channel open drain output	3
Input/output	VCC2 level	P0, P1, P6
	VCC1 level	P7 to P10
Input/output level		Select input or output for each individual port by a program.
Select function		Select a pull-up resistor in 4-bit units.

Table 12.2 I/O Pins

Pin Name	I/O	Function
P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7	I/O	Input/output port CMOS output, pull-up resistor selectable
P7_0 to P7_7	I/O	Input/output port P7_0 to P7_1: N-channel open drain output, no pull-up resistor P7_2 to P7_7: CMOS output, pull-up resistor selectable
P8_0 to P8_7	I/O	Input/output port P8_0 to P8_4, P8_6, P8_7: CMOS output, pull-up resistor selectable P8_5: N-channel open drain output, no pull-up resistor
P9_0 to P9_7, P10_0 to P10_7	I/O	Input/output port CMOS output, pull-up resistor selectable
P2_0 to P2_7 P3_0 to P3_7 P4_0 to P4_7 P5_0 to P5_7 (1)	I/O	Input/output port Internally connected to PLC modem

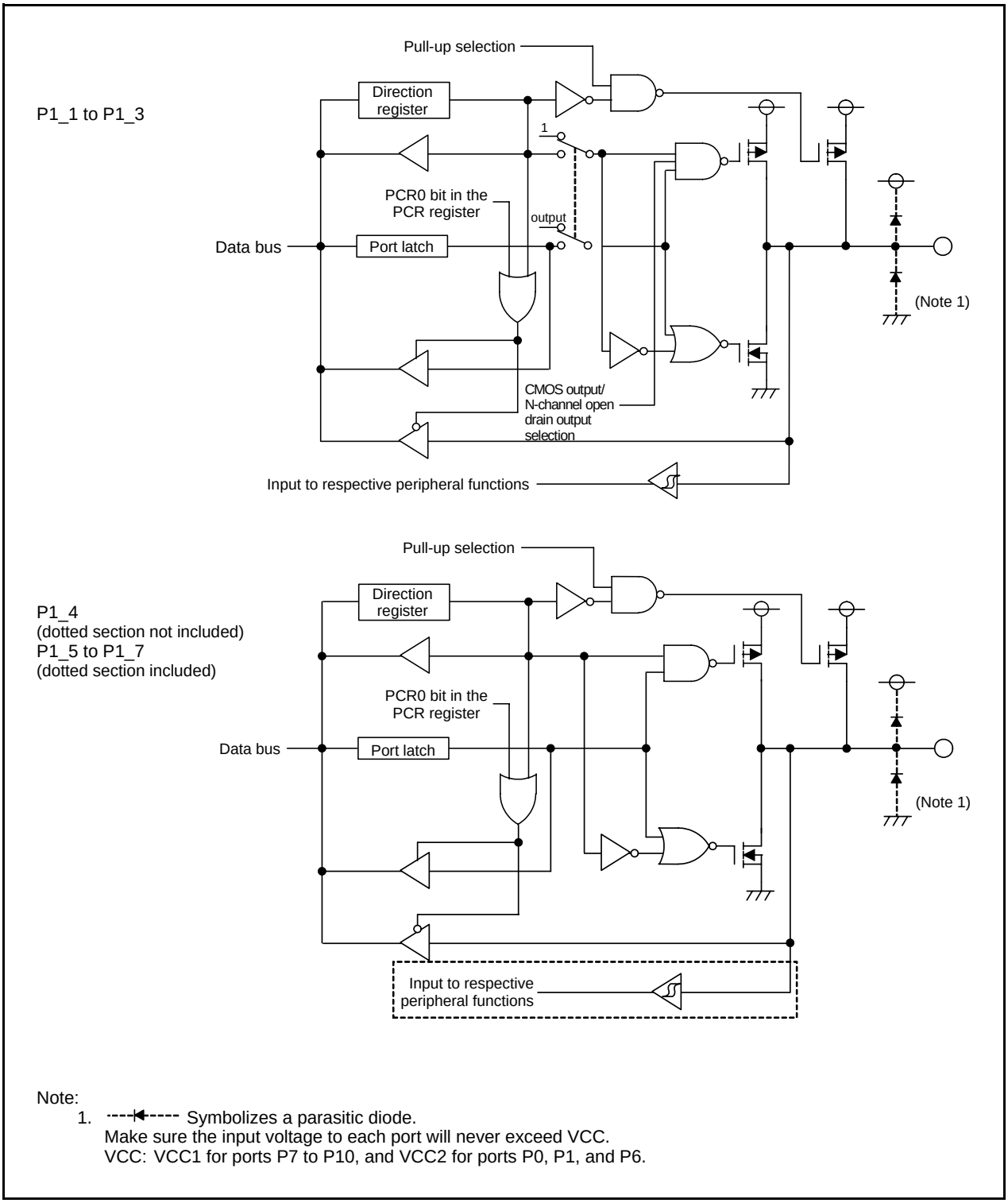
Note:
1. P2 to P5 are connected to the internal PLC modem. No external pin is provided.
Control them via provided DLL software and do not control them directly via user software.

12.2 I/O Ports and Pins



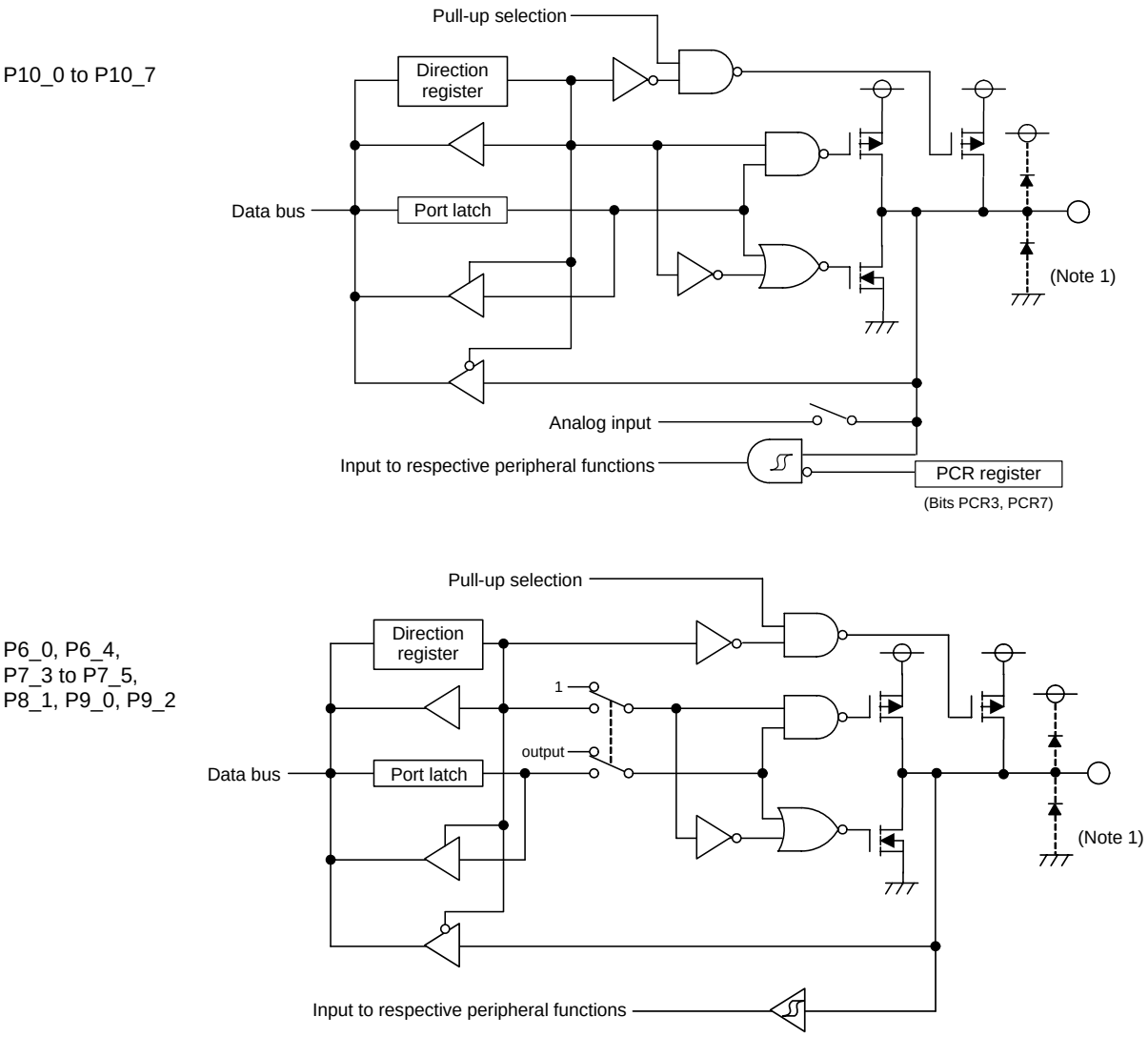
Note:
1. ----◄----- Symbolizes a parasitic diode.
Make sure the input voltage to each port will never exceed VCC.
VCC: VCC1 for ports P7 to P10, and VCC2 for ports P0, P1, and P6.

Figure 12.1 I/O Ports (1/8)



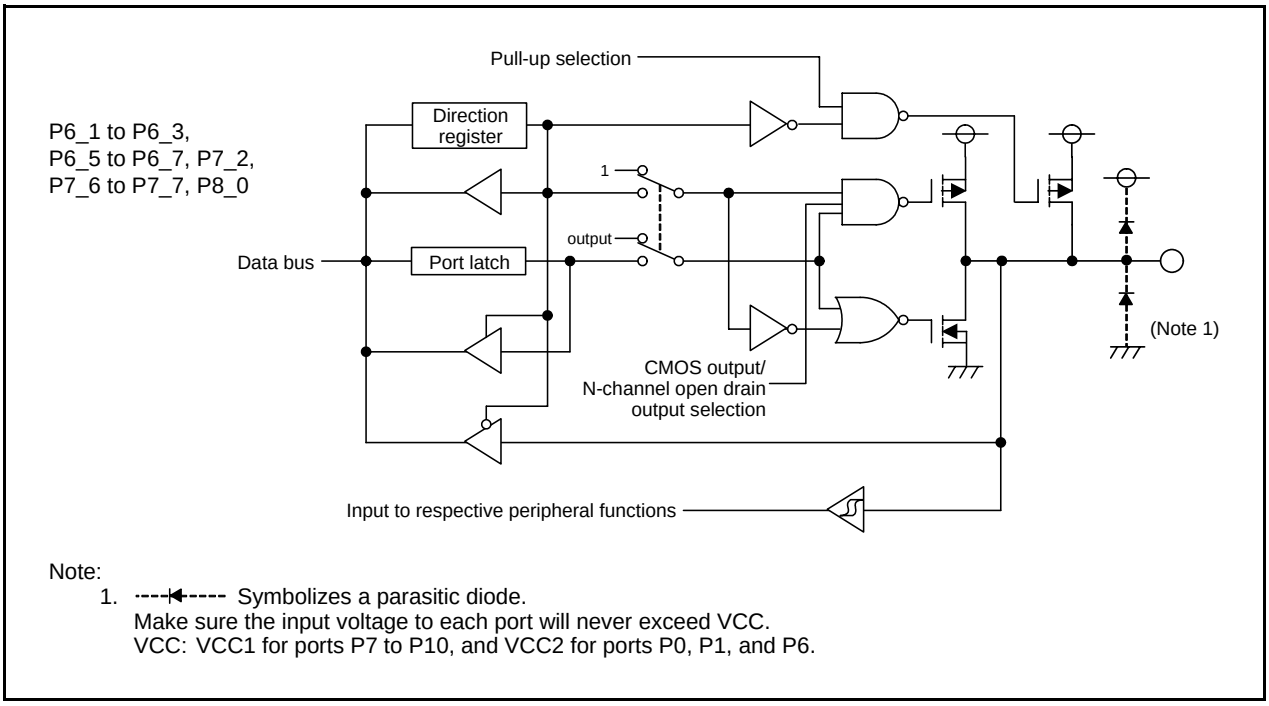
Note:
1. ----|◄|---- Symbolizes a parasitic diode.
Make sure the input voltage to each port will never exceed VCC.
VCC: VCC1 for ports P7 to P10, and VCC2 for ports P0, P1, and P6.

Figure 12.2 I/O Ports (2/8)



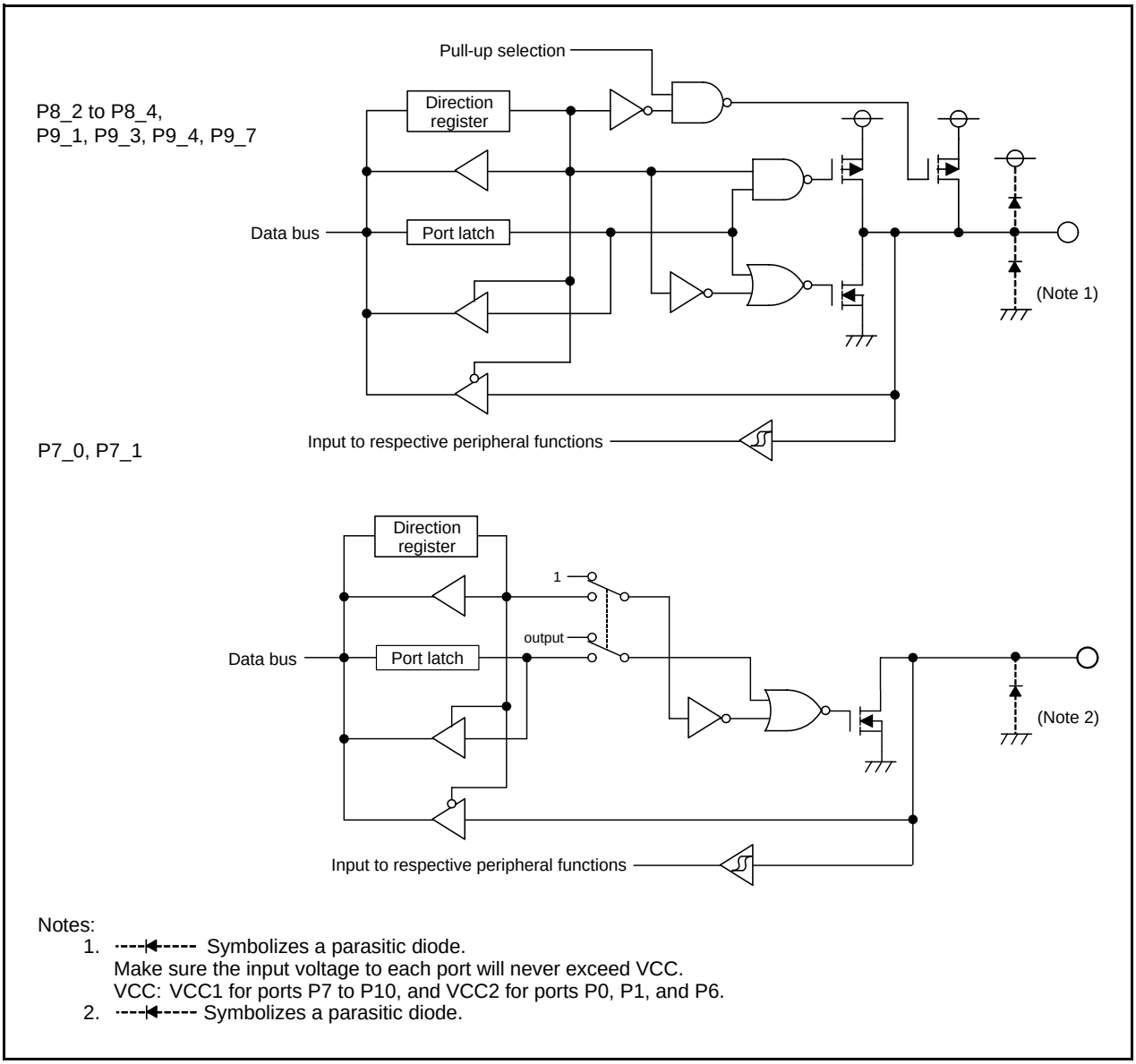
Note:
1. ----◄----- Symbolizes a parasitic diode.
Make sure the input voltage to each port will never exceed VCC.
VCC: VCC1 for ports P7 to P10, and VCC2 for ports P0, P1, and P6.

Figure 12.3 I/O Ports (3/8)



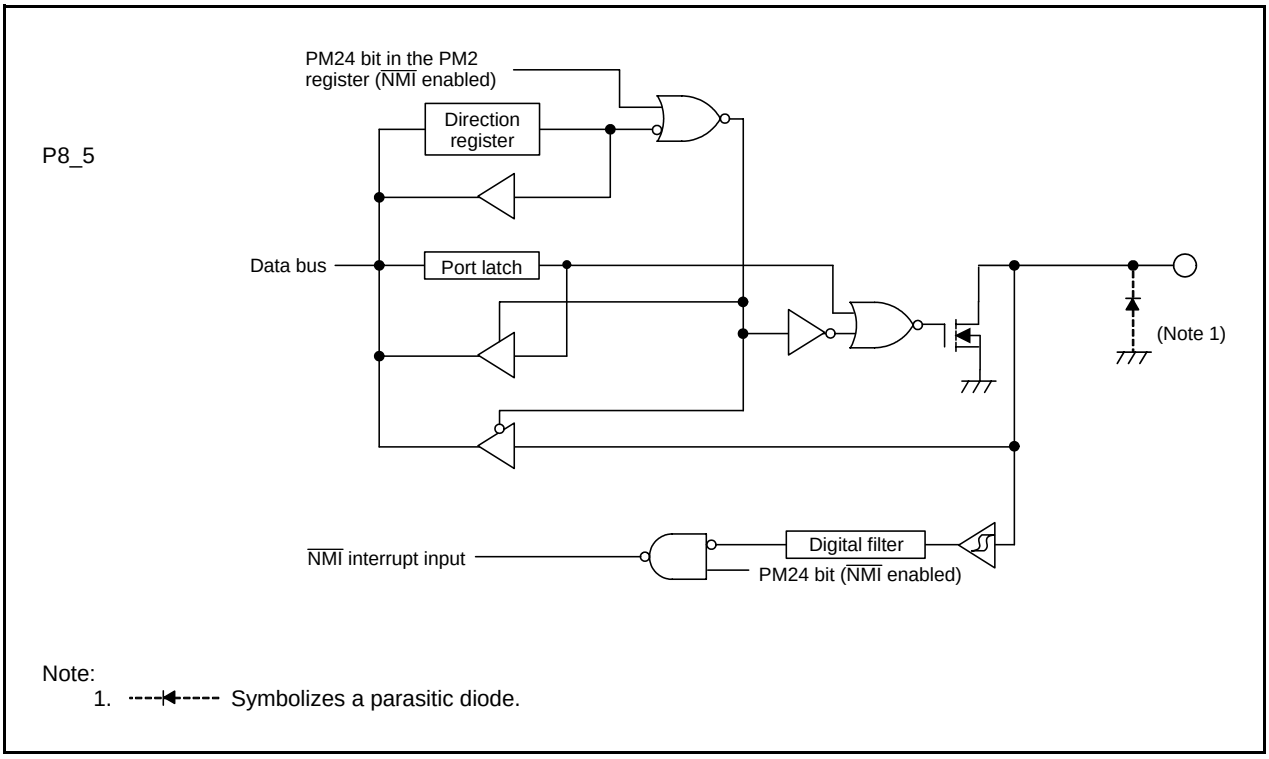
Note:
1. ----|◄|---- Symbolizes a parasitic diode.
Make sure the input voltage to each port will never exceed VCC.
VCC: VCC1 for ports P7 to P10, and VCC2 for ports P0, P1, and P6.

Figure 12.4 I/O Ports (4/8)



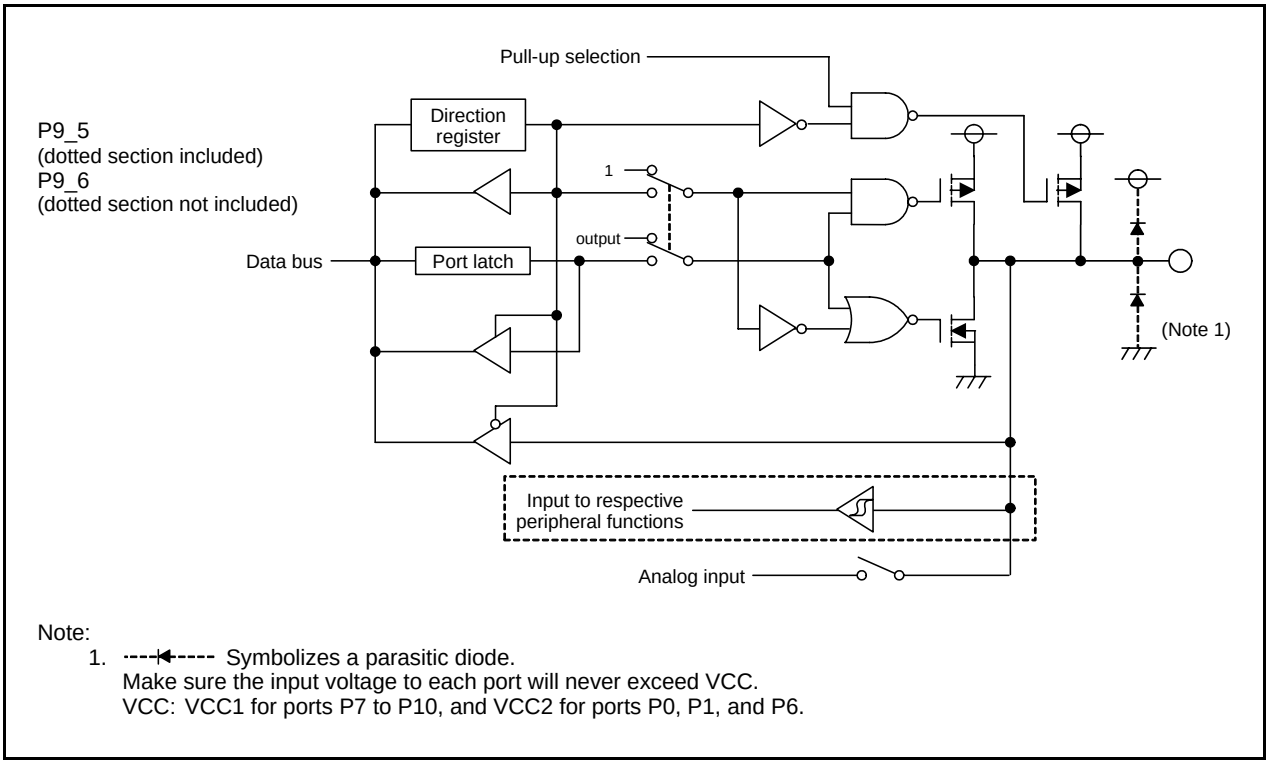
Notes:
1. ----◄----- Symbolizes a parasitic diode.
   Make sure the input voltage to each port will never exceed VCC.
   VCC: VCC1 for ports P7 to P10, and VCC2 for ports P0, P1, and P6.
2. ----◄----- Symbolizes a parasitic diode.

Figure 12.5 I/O Ports (5/8)



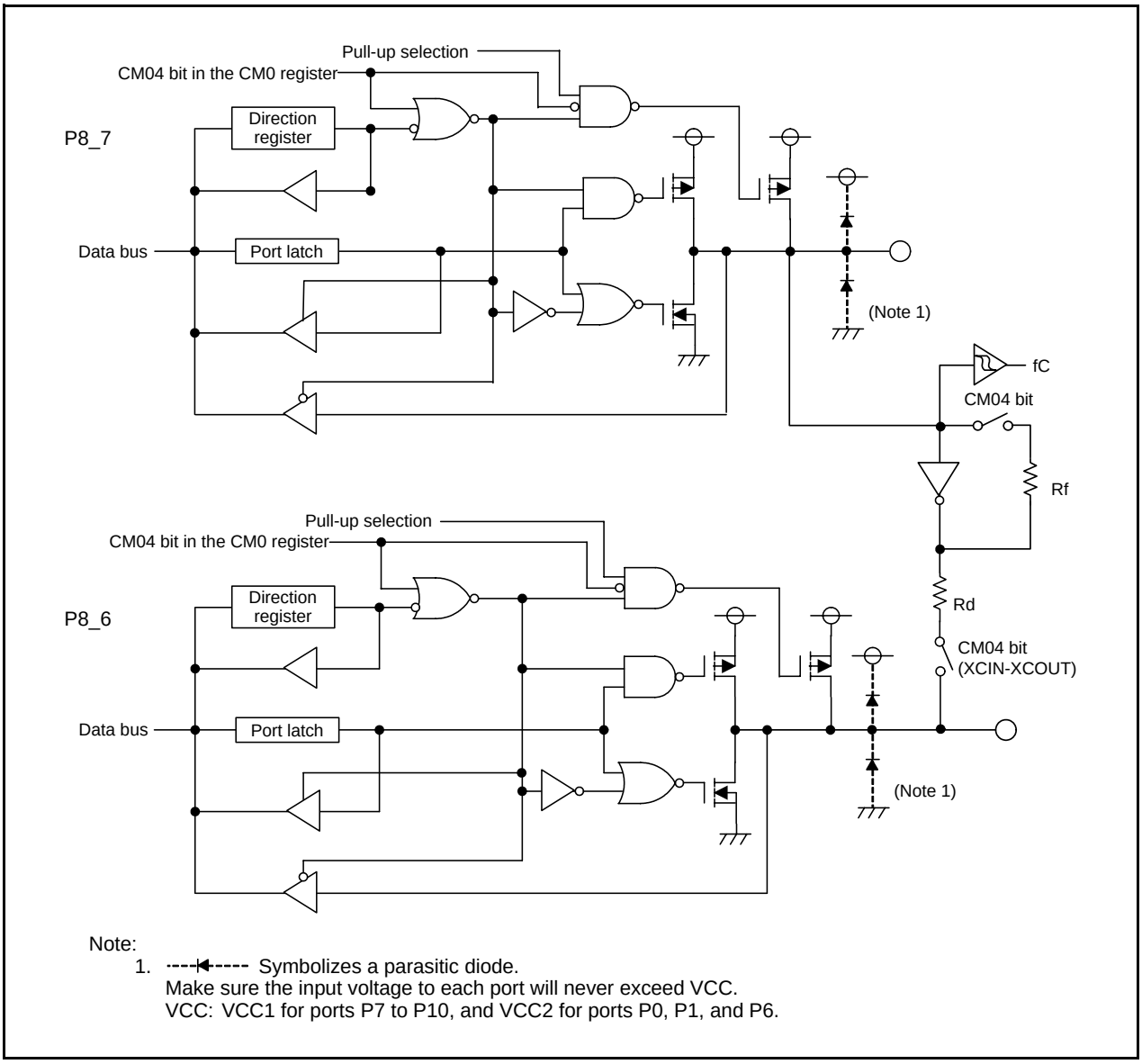
Note:
1. ----|◄----- Symbolizes a parasitic diode.

Figure 12.6 I/O Ports (6/8)



Note:
1. ----|◄----- Symbolizes a parasitic diode.
Make sure the input voltage to each port will never exceed VCC.
VCC: VCC1 for ports P7 to P10, and VCC2 for ports P0, P1, and P6.

Figure 12.7 I/O Ports (7/8)



Note:
1. ----◄----- Symbolizes a parasitic diode.
Make sure the input voltage to each port will never exceed VCC.
VCC: VCC1 for ports P7 to P10, and VCC2 for ports P0, P1, and P6.

Figure 12.8 I/O Ports (8/8)

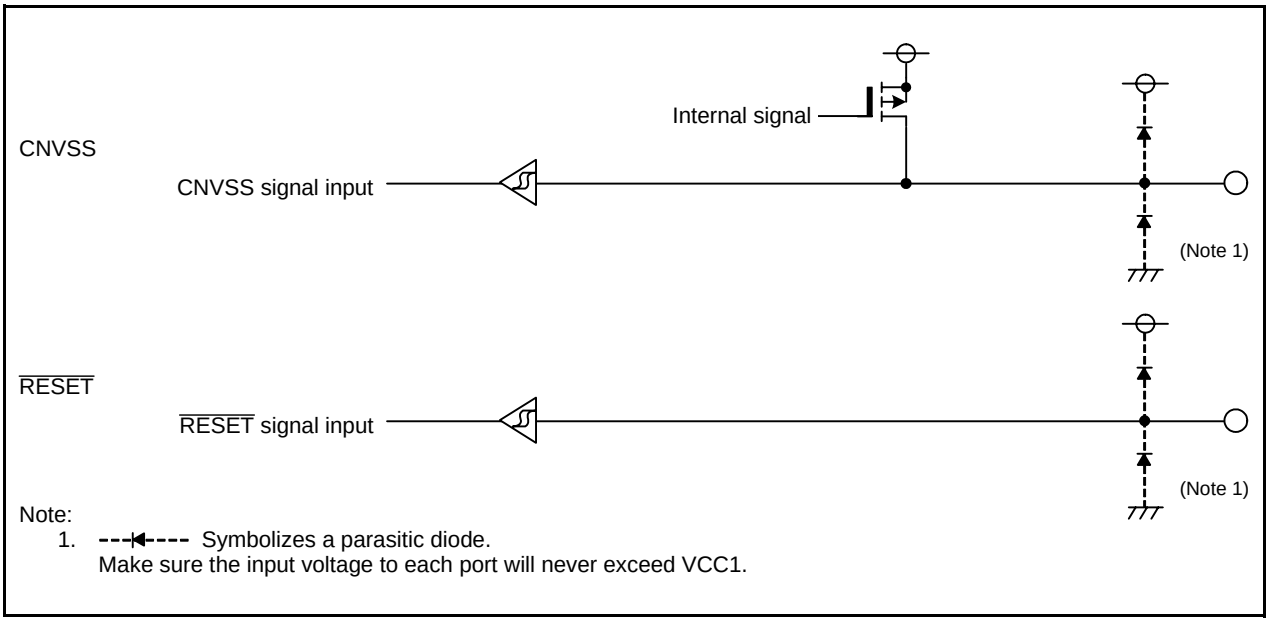


Figure 12.9 I/O Pins

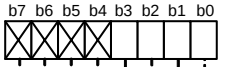
12.3 Registers

Table 12.3 Registers

Address	Register	Symbol	Reset Value
0360h	Pull-Up Control Register 0	PUR0	00h
0361h	Pull-Up Control Register 1	PUR1	0000 0000b
0362h	Pull-Up Control Register 2	PUR2	00h
0366h	Port Control Register	PCR	0000 0XX0b
0369h	$\overline{\text{NMI}}$ Digital Filter Register	NMIDF	XXXX X000b
03E0h	Port P0 Register	P0	XXh
03E1h	Port P1 Register	P1	XXh
03E2h	Port P0 Direction Register	PD0	00h
03E3h	Port P1 Direction Register	PD1	00h
03E4h	Port P2 Register	P2	XXh
03E5h	Port P3 Register	P3	XXh
03E6h	Port P2 Direction Register	PD2	00h
03E7h	Port P3 Direction Register	PD3	00h
03E8h	Port P4 Register	P4	XXh
03E9h	Port P5 Register	P5	XXh
03EAh	Port P4 Direction Register	PD4	00h
03EBh	Port P5 Direction Register	PD5	00h
03ECh	Port P6 Register	P6	XXh
03EDh	Port P7 Register	P7	XXh
03EEh	Port P6 Direction Register	PD6	00h
03EFh	Port P7 Direction Register	PD7	00h
03F0h	Port P8 Register	P8	XXh
03F1h	Port P9 Register	P9	XXh
03F2h	Port P8 Direction Register	PD8	00h
03F3h	Port P9 Direction Register	PD9	00h
03F4h	Port P10 Register	P10	XXh
03F6h	Port P10 Direction Register	PD10	00h

12.3.1 Pull-Up Control Register 0 (PUR0)

Pull-Up Control Register 0



Symbol: PUR0 | Address: 0360h | After Reset: 00h

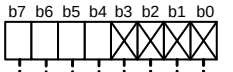
Bit Symbol	Bit Name	Function	RW
PU00	P0_0 to P0_3 pull-up	0: Not pulled high 1: Pulled high	RW
PU01	P0_4 to P0_7 pull-up		RW
PU02	P1_0 to P1_3 pull-up		RW
PU03	P1_4 to P1_7 pull-up		RW
— (b7-b4)	No register bits. Read as undefined.		—

PU0i Bit (b3 to b0) (i = 0 to 3)

The pin for which the PU0i bit is 1 (pulled high) and the direction bit is 0 (input mode) is pulled high.

12.3.2 Pull-Up Control Register 1 (PUR1)

Pull-Up Control Register 1



Symbol: PUR1 | Address: 0361h | After Reset: 0000 0000b

Bit Symbol	Bit Name	Function	RW
— (b3-b0)	No register bits. Read as undefined.		—
PU14	P6_0 to P6_3 pull-up	0: Not pulled high 1: Pulled high	RW
PU15	P6_4 to P6_7 pull-up		RW
PU16	P7_2 to P7_3 pull-up		RW
PU17	P7_4 to P7_7 pull-up		RW

PU14 (P6_0 to P6_3 pull-up) (b4)
PU15 (P6_4 to P6_7 pull-up) (b5)
PU17 (P7_4 to P7_7 pull-up) (b7)

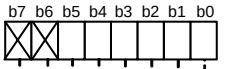
The pin for which the bit in the PU1i bit (i = 4, 5, 7) is 1 (pulled high) and the direction bit is 0 (input mode) is pulled high.

PU16 (P7_2 to P7_3 pull-up) (b6)

The pin for which the bit in the PU16 bit is 1 (pulled high) and the direction bit is 0 (input mode) is pulled high.
Pins P7_0 and P7_1 are not pulled high.

12.3.3 Pull-Up Control Register 2 (PUR2)

Pull-Up Control Register 2



Symbol: PUR2 | Address: 0362h | After Reset: 00h

Bit Symbol	Bit Name	Function	RW
PU20	P8_0 to P8_3 pull-up	0: Not pulled high 1: Pulled high	RW
PU21	P8_4 to P8_7 pull-up		RW
PU22	P9_0 to P9_3 pull-up		RW
PU23	P9_4 to P9_7 pull-up		RW
PU24	P10_0 to P10_3 pull-up		RW
PU25	P10_4 to P10_7 pull-up		RW
— (b7-b6)	No register bits. If necessary, set to 0. Read as 0.		—

The pin for which the bit in the PUR2 register is 1 (pulled high) and the direction bit is 0 (input mode) is pulled high.

PU20 (P8_0 to P8_3 pull-up) (b0)
PU22 (P9_0 to P9_3 pull-up) (b2)
PU23 (P9_4 to P9_7 pull-up) (b3)
PU24 (P10_0 to P10_3 pull-up) (b4)
PU25 (P10_4 to P10_7 pull-up) (b5)

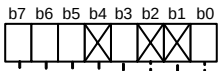
When the PU2i (i = 0, 2 to 5) bit is 1 (pulled high) and the direction bit is 0 (input mode), the corresponding pin is pulled high.

PU21 (P8_4, P8_6, P8_7 pull-up) (b1)

When the PU21 bit is 1 (pulled high) and the direction bit is 0 (input mode), the corresponding pin is pulled high.
The P8_5 pin is not pulled high.

12.3.4 Port Control Register (PCR)

Port Control Register



Symbol: PCR | Address: 0366h | After Reset: 0000 0XX0b

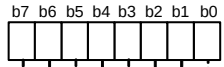
Bit Symbol	Bit Name	Function	RW
PCR0	Port P1 control bit	Operation performed when the P1 register is read 0: When the port is set to input, the input levels of pins P1_0 to P1_7 are read. When set to output, the port latch is read. 1: The port latch is read regardless of whether the port is set to input or output.	RW
— (b2-b1)	No register bits. If necessary, set to 0. Read as undefined.		—
PCR3	Key input enable bit ($\overline{KI4}$ to $\overline{KI7}$)	0: Enabled 1: Disabled	RW
— (b4)	No register bits. Read as undefined.		—
PCR5	$\overline{INT6}$ input enable bit	0: Enabled 1: Disabled	RW
PCR6	$\overline{INT7}$ input enable bit	0: Enabled 1: Disabled	RW
PCR7	Key input enable bit ($\overline{KI0}$ to $\overline{KI3}$)	0: Enabled 1: Disabled	RW

PCR0 (Port P1 control bit) (b0)

When the P1 register is read after the PCR0 bit is set to 1, the corresponding port latch is read regardless of the PD1 register setting.

12.3.5 Port Pi Register (Pi) (i = 0 to 10)

Port Pi Register (i = 0 to 10)



Symbol	Address	After Reset
P0 to P3	03E0h, 03E1h, 03E4h, 03E5h	XXh
P4 to P7	03E8h, 03E9h, 03ECh, 03EDh	XXh
P8 to P10	03F0h, 03F1h, 03F4h	XXh

Bit Symbol	Bit Name	Function	RW
Pi_0	Port Pi_0 bit	The pin level of any I/O port which is set to input mode can be read by reading the corresponding bit in this register. The pin level of any I/O port which is set to output mode can be controlled by writing to the corresponding bit in this register. 0: Low level 1: High level	RW
Pi_1	Port Pi_1 bit		RW
Pi_2	Port Pi_2 bit		RW
Pi_3	Port Pi_3 bit		RW
Pi_4	Port Pi_4 bit		RW
Pi_5	Port Pi_5 bit		RW
Pi_6	Port Pi_6 bit		RW
Pi_7	Port Pi_7 bit		RW

Data input/output to and from external devices are accomplished by reading and writing to the Pi register. Each bit of the Pi register consists of a port latch to hold the output data and a circuit to read the pin status.

For ports set to input mode, the input level of the pin can be read by reading the corresponding Pi register, and data can be written to the port latch by writing to the Pi register.

For ports set to output mode, the port latch can be read by reading the corresponding Pi register, and data can be written to the port latch by writing to the Pi register. The data written to the port latch is output from the pin. Each bit in the Pi register corresponds to one port.

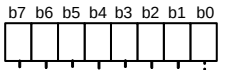
Since P7_0, P7_1, and P8_5 are N-channel open drain ports, when set to 1, the pin status becomes high-impedance.

When the CM04 bit in the CM0 register is 1 (XCIN-XCOUT oscillation function) and bits PD8_6 and PD8_7 in the PD8 register are 0 (input mode), values of bits P8_6 and P8_7 in the P8 register are undefined.

There is no external pin for P2 to P5.

12.3.6 Port Pi Direction Register (PDi) (i = 0 to 10)

Port Pi Direction Register (i = 0 to 10)



Symbol	Address	After Reset
PD0 to PD3	03E2h, 03E3h, 03E6h, 03E7h	00h
PD4 to PD7	03EAh, 03EBh, 03EEh, 03EFh	00h
PD8 to PD10	03F2h, 03F3h, 03F6h	00h

Bit Symbol	Bit Name	Function	RW
PDi_0	Port Pi_0 direction bit	0: Input mode (functions as an input port) 1: Output mode (functions as an output port)	RW
PDi_1	Port Pi_1 direction bit		RW
PDi_2	Port Pi_2 direction bit		RW
PDi_3	Port Pi_3 direction bit		RW
PDi_4	Port Pi_4 direction bit		RW
PDi_5	Port Pi_5 direction bit		RW
PDi_6	Port Pi_6 direction bit		RW
PDi_7	Port Pi_7 direction bit		RW

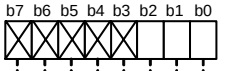
Write to the PD9 register in the next instruction after setting the PRC2 bit in the PRCR register to 1 (write enabled).

Select whether I/O ports are to be used for input or output by the PDi register. Each bit in the PDi register has its corresponding port.

There is no external pin for P2 to P5.

12.3.7 $\overline{\text{NMI}}$ Digital Filter Register (NMIDF)

$\overline{\text{NMI}}$ Digital Filter Register



Symbol	Address	After Reset
NMIDF	0369h	XXXX X000b

Bit Symbol	Bit Name	Function	RW
NMIDF0	$\overline{\text{NMI}}$ filter sampling clock select bit	b2 b1 b0 0 0 0: No filter 0 0 1: CPU clock divided by 2 0 1 0: CPU clock divided by 4 0 1 1: CPU clock divided by 8 1 0 0: CPU clock divided by 16 1 0 1: CPU clock divided by 32 1 1 0: CPU clock divided by 64 1 1 1: CPU clock divided by 128	RW
NMIDF1			RW
NMIDF2			RW
— (b7-b3)	No register bits. If necessary, set to 0. Read as undefined value.		—

Change the NMIDF register under the following conditions:

- The PM24 bit in the PM2 register is 0 ($\overline{\text{NMI}}$ interrupt disabled)

Once the PM24 bit is set to 1 ($\overline{\text{NMI}}$ interrupt enabled), it cannot be set to 0 by a program. Change the NMIDF register before setting the PM24 bit to 1.

12.4 Peripheral Function I/O

12.4.1 Peripheral Function I/O and Port Direction Bits

Programmable I/O ports can share pins with peripheral function I/O. (See Tables 1.6 to 1.8 “Pin Names”.) Table 12.4 lists The Setting of Direction Bits Functioning as Peripheral Function I/O. For peripheral function settings, see descriptions of each function.

Table 12.4 The Setting of Direction Bits Functioning as Peripheral Function I/O

Peripheral Function I/O	The Setting of the Port Direction Bit Sharing the Same Pin
Input	Set to 0 (input mode).
Output	Set to either 0 or 1. (Outputs regardless of the direction bit setting)

12.4.2 Priority Level of Peripheral Function I/O

Multiple peripheral functions can share the same pin.
For example, when peripheral function A and peripheral function B share a pin, input and output are as follows:

- When the pin functions as input for peripheral functions A and B
  The same signal is input as each input signal. However, the timing of accepting the signal differs depending on conditions (e.g. internal delay) of functions A and B.
- When the pin functions as output for peripheral function A and as input for peripheral function B
  Peripheral function A outputs a signal from the pin, and peripheral function B inputs the signal.

12.4.3 $\overline{\text{NMI}}$ Digital Filter

The $\overline{\text{NMI}}$ input function includes a digital filter. A sampling clock can be selected by bits NMIDF2 to NMIDF0 in the NMIDF register. The $\overline{\text{NMI}}$ level is sampled for every sampling clock. When the same sampled level is detected three times in a row, the level is transferred to the internal circuit.
When using the $\overline{\text{NMI}}$ digital filter, do not enter wait mode or stop mode.
Port P8_5 is not affected by the digital filter.
Figure 12.10 shows NMI Digital Filter, and Figure 12.11 shows NMI Digital Filter Operation Example.

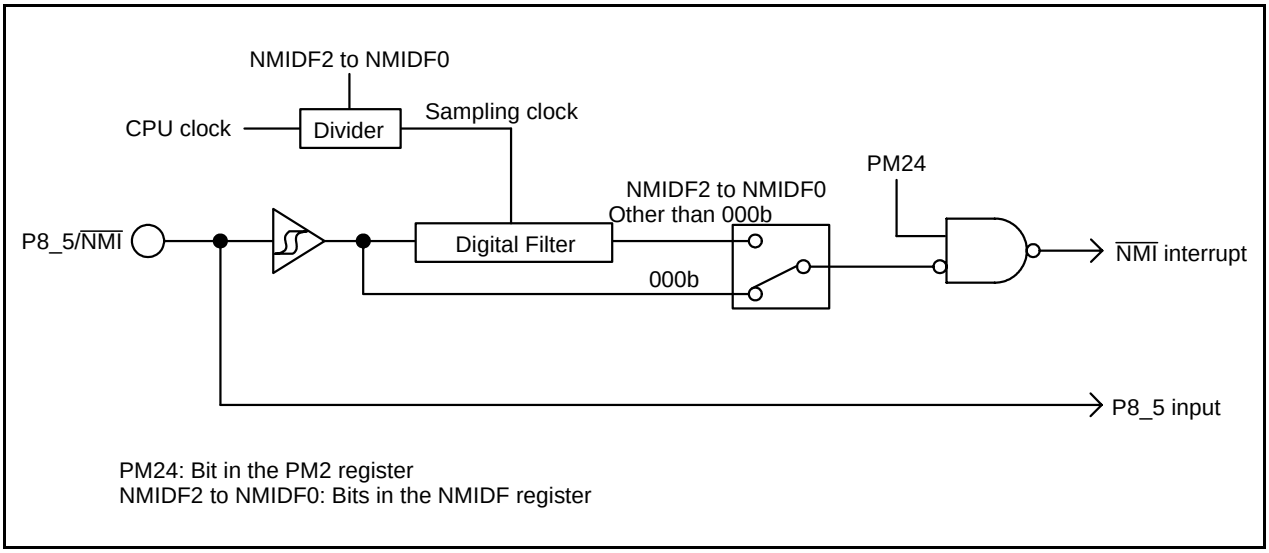


Figure 12.10 $\overline{\text{NMI}}$ Digital Filter

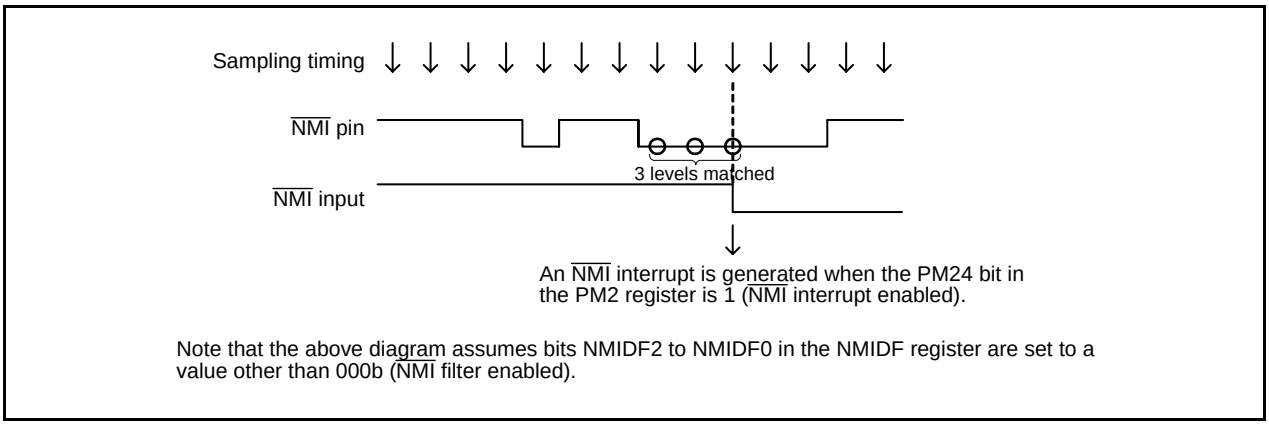


Figure 12.11 $\overline{\text{NMI}}$ Digital Filter Operation Example

12.4.4 CNVSS Pin

The built-in pull-up resistor of the CNVSS pin is activated after watchdog timer reset, hardware reset, or voltage monitor 0 reset. Thus, the CNVSS pin outputs a high-level signal up to two cycles of the fOCO-S. Always connect the CNVSS pin to VSS via a resistor.

12.5 Unassigned Pin Handling

Table 12.5 Unassigned Pin Handling

Pin Name	Connection (2)
Ports P0, P1, P6	One of the following: • Set to input mode and connect a pin to VSS via a resistor (pull-down) • Set to input mode and connect a pin to VCC2 via a resistor (pull-up) • Set to output mode and leave the pins open (1)
Ports P7 to P10	One of the following: • Set to input mode and connect a pin to VSS via a resistor (pull-down) • Set to input mode and connect a pin to VCC1 via a resistor (pull-up) • Set to output mode and leave the pins open (1, 3)
AVCC	Connect to VCC1
AVSS, VREF	Connect to VSS

Notes:
1. When setting a port to output mode and leaving it open, be aware that the port remains in input mode until it is switched to output mode by a program after reset. For this reason, the voltage level on the pin becomes undefined, causing the power supply current to increase while the port remains in input mode.
   Furthermore, since the contents of the direction registers could be changed by noise or noise-induced loss of control, it is recommended that the contents of the direction registers be regularly reset in software to improve the reliability of the program.
2. Make sure the unassigned pins are connected with the shortest possible wiring from the MCU pins (maximum 2 cm).
3. Ports P7_0, P7_1 and P8_5 are N-channel open drain outputs.
   When ports P7_0, P7_1 and P8_5 are set to output mode, make sure a low-level signal is output from the pins.

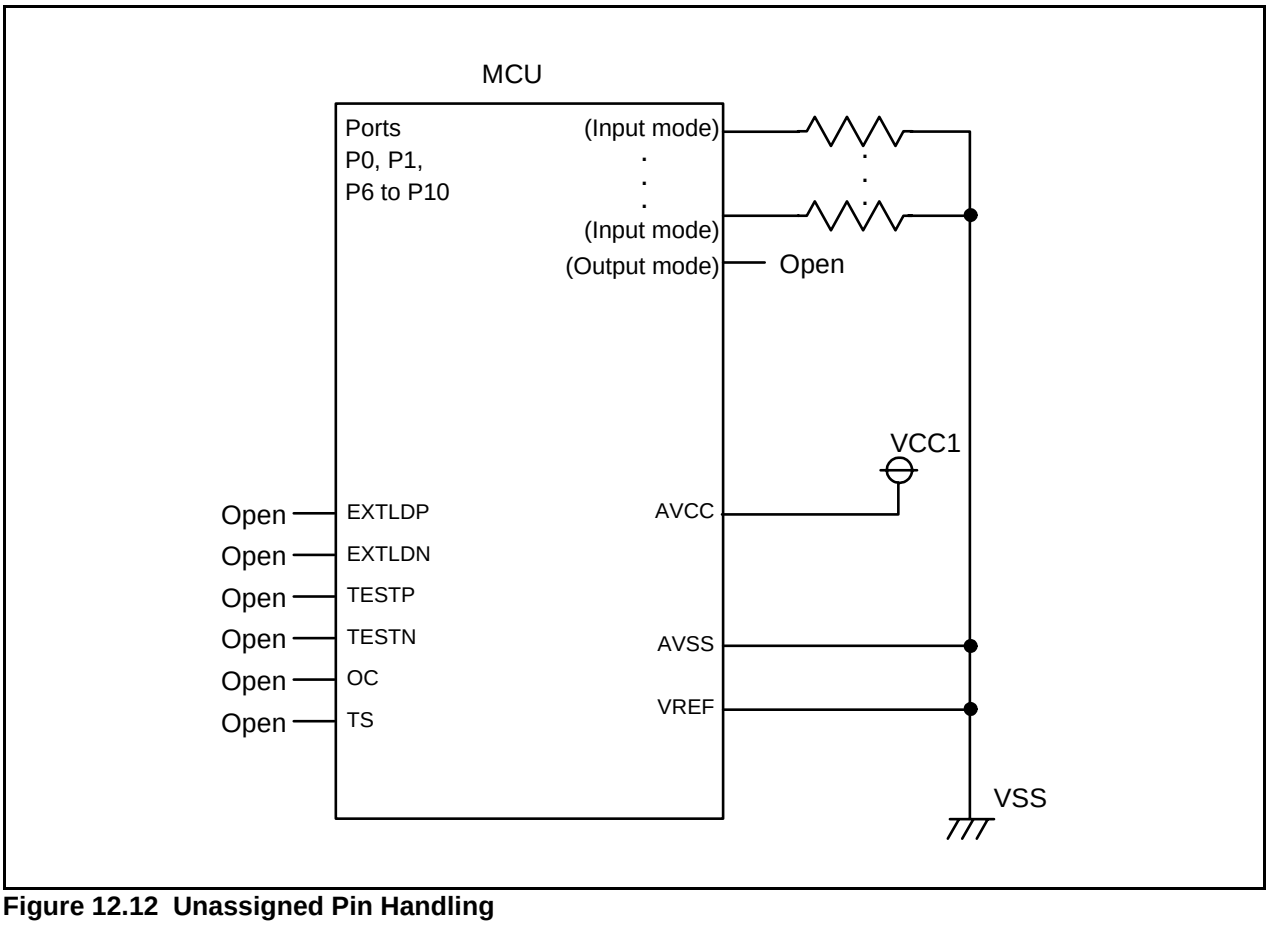


Figure 12.12 Unassigned Pin Handling

12.6 Notes on Programmable I/O Ports

12.6.1 Influence of SI/O3 and SI/O4

Setting the SM32 bit in the S3C register to 1 causes the P9_2 pin to become high-impedance. Similarly, setting the SM42 bit in the S4C register to 1 causes the P9_6 pin to become high-impedance.

13. Interrupts

Note

No external pin is provided for $\overline{\text{INT0}}$, $\overline{\text{INT6}}$ and $\overline{\text{INT7}}$ because it is internally connected to the PLC modem.

13.1 Introduction

Table 13.1 lists Types of Interrupts, and Table 13.2 lists I/O Pins. The pins shown in Table 13.2 are external interrupt input pins. Refer to the peripheral functions for the pins related to the peripheral functions.

Table 13.1 Types of Interrupts

Type		Interrupt	Function
Software		Undefined instruction (UND instruction) Overflow (INTO instruction) BRK instruction INT instruction	An interrupt is generated by executing an instruction. Non-maskable interrupt (2)
Hardware	Specific	$\overline{\text{NMI}}$ Watchdog timer Oscillator stop/restart detect Address match Single step (1) $\overline{\text{DBC}}$ (1)	Interrupt by the MCU hardware Non-maskable interrupt (2)
	Peripheral function	$\overline{\text{INT}}$, timers, etc. (Refer to 13.6.2 “Relocatable Vector Tables”.)	Interrupt by the peripheral functions in the MCU Maskable interrupt (interrupt priority level: 7 levels) (2)

Notes:
1. This interrupt is provided exclusively for developers and should not be used.
2. Maskable interrupt: Interrupt status (enabled or disabled) can be selected by the interrupt enable flag (I flag).
   Interrupt priority can be changed by the interrupt priority level.
   Non-maskable interrupt: Interrupt status (enabled or disabled) cannot be selected by the interrupt enable flag (I flag).
   Interrupt priority cannot be changed by the interrupt priority level.

Table 13.2 I/O Pins

Pin Name	I/O	Function
$\overline{\text{NMI}}$	Input	$\overline{\text{NMI}}$ interrupt input
$\overline{\text{INTi}}$	Input (1)	$\overline{\text{INTi}}$ interrupt input
$\overline{\text{KI0}}$ to $\overline{\text{KI7}}$	Input (1)	Key input

i = 0 to 7

Note:
1. Set the port direction bits which share pins to 0 (input mode).
2. $\overline{\text{INT0}}$, $\overline{\text{INT6}}$, and $\overline{\text{INT7}}$ are connected to the internal PLC modem. No external pin is provided. Control them via provided DLL software and do not control them directly via user software.

13.2 Registers

Table 13.3 Registers (1/2)

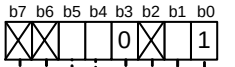
Address	Register	Symbol	Reset Value
001Eh	Processor Mode Register 2	PM2	XX00 0X01b
0042h	$\overline{\text{INT7}}$ Interrupt Control Register	INT7IC	XX00 X000b
0043h	$\overline{\text{INT6}}$ Interrupt Control Register	INT6IC	XX00 X000b
0044h	$\overline{\text{INT3}}$ Interrupt Control Register	INT3IC	XX00 X000b
0045h	Timer B5 Interrupt Control Register	TB5IC	XXXX X000b
0046h	Timer B4 Interrupt Control Register UART1 Bus Collision Detection Interrupt Control Register	TB4IC, U1BCNIC	XXXX X000b
0047h	Timer B3 Interrupt Control Register, UART0 Bus Collision Detection Interrupt Control Register	TB3IC, U0BCNIC	XXXX X000b
0048h	SI/O4 Interrupt Control Register, $\overline{\text{INT5}}$ Interrupt Control Register	S4IC, INT5IC	XX00 X000b
0049h	SI/O3 Interrupt Control Register, $\overline{\text{INT4}}$ Interrupt Control Register	S3IC, INT4IC	XX00 X000b
004Ah	UART2 Bus Collision Detection Interrupt Control Register	BCNIC	XXXX X000b
004Bh	DMA0 Interrupt Control Register	DM0IC	XXXX X000b
004Ch	DMA1 Interrupt Control Register	DM1IC	XXXX X000b
004Dh	Key Input Interrupt Control Register	KUPIC	XX00 X000b
004Eh	A/D Conversion Interrupt Control Register	ADIC	XXXX X000b
004Fh	UART2 Transmit Interrupt Control Register	S2TIC	XXXX X000b
0050h	UART2 Receive Interrupt Control Register	S2RIC	XXXX X000b
0051h	UART0 Transmit Interrupt Control Register	S0TIC	XXXX X000b
0052h	UART0 Receive Interrupt Control Register	S0RIC	XXXX X000b
0053h	UART1 Transmit Interrupt Control Register	S1TIC	XXXX X000b
0054h	UART1 Transmit Interrupt Control Register	S1TIC	XXXX X000b
0055h	Timer A0 Interrupt Control Register	TA0IC	XXXX X000b
0056h	Timer A1 Interrupt Control Register	TA1IC	XXXX X000b
0057h	Timer A2 Interrupt Control Register	TA2IC	XXXX X000b
0058h	Timer A3 Interrupt Control Register	TA3IC	XXXX X000b
0059h	Timer A4 Interrupt Control Register	TA4IC	XXXX X000b
005Ah	Timer B0 Interrupt Control Register	TB0IC	XXXX X000b
005Bh	Timer B1 Interrupt Control Register	TB1IC	XXXX X000b
005Ch	Timer B2 Interrupt Control Register	TB2IC	XXXX X000b
005Dh	$\overline{\text{INT0}}$ Interrupt Control Register	INT0IC	XX00 X000b
005Eh	$\overline{\text{INT1}}$ Interrupt Control Register	INT1IC	XX00 X000b
005Fh	$\overline{\text{INT2}}$ Interrupt Control Register	INT2IC	XX00 X000b
0069h	DMA2 Interrupt Control Register	DM2IC	XXXX X000b
006Ah	DMA3 Interrupt Control Register	DM3IC	XXXX X000b
006Bh	UART5 Bus Collision Detection Interrupt Control Register	U5BCNIC	XXXX X000b
006Ch	UART5 Transmit Interrupt Control Register	S5TIC	XXXX X000b
006Dh	UART5 Receive Interrupt Control Register	S5RIC	XXXX X000b

Table 13.4 Registers (2/2)

Address	Register	Symbol	Reset Value
006Eh	UART6 Bus Collision Detection Interrupt Control Register, Real-Time Clock Periodic Interrupt Control Register	U6BCNIC, RTCTIC	XXXX X000b
006Fh	UART6 Transmit Interrupt Control Register, Real-Time Clock Alarm Interrupt Control Register	S6TIC, RTCCIC	XXXX X000b
0070h	UART6 Receive Interrupt Control Register	S6RIC	XXXX X000b
0071h	UART7 Bus Collision Detection Interrupt Control Register	U7BCNIC	XXXX X000b
0072h	UART7 Transmit Interrupt Control Register	S7TIC	XXXX X000b
0073h	UART7 Receive Interrupt Control Register	S7RIC	XXXX X000b
007Bh	I2C-bus Interface Interrupt Control Register	IICIC	XXXX X000b
007Ch	SCL/SDA Interrupt Control Register	SCLDAIC	XXXX X000b
0205h	Interrupt Source Select Register 3	IFSR3A	00h
0206h	Interrupt Source Select Register 2	IFSR2A	00h
0207h	Interrupt Source Select Register	IFSR	00h
020Eh	Address Match Interrupt Enable Register	AIER	XXXX XX00b
020Fh	Address Match Interrupt Enable Register 2	AIER2	XXXX XX00b
0210h	Address Match Interrupt Register 0	RMAD0	00h
0211h			00h
0212h			X0h
0214h	Address Match Interrupt Register 1	RMAD1	00h
0215h			00h
0216h			X0h
0218h	Address Match Interrupt Register 2	RMAD2	00h
0219h			00h
021Ah			X0h
021Ch	Address Match Interrupt Register 3	RMAD3	00h
021Dh			00h
021Eh			X0h
0366h	Port Control Register	PCR	0000 0XX0b
0369h	$\overline{\text{NMI}}$ Digital Filter Register	NMIDF	XXXX X000b

13.2.1 Processor Mode Register 2 (PM2)

Processor Mode Register 2



Symbol	Address	After Reset
PM2	001Eh	XX00 0X01b

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved bit	Set to 1.	RW
PM21	System clock protection bit	0: Clock is protected by PRCR register 1: Clock change disabled	RW
— (b2)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
— (b3)	Reserved bit	Set to 0.	RW
PM24	$\overline{\text{NMI}}$ interrupt enable bit	0: $\overline{\text{NMI}}$ interrupt disabled 1: $\overline{\text{NMI}}$ interrupt enabled	RW
PM25	Peripheral clock fC provide bit	0: Not provided 1: Provided	RW
— (b7-b6)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

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

PM24 ($\overline{\text{NMI}}$ interrupt enable bit) (b4)

Once this bit is set to 1, it cannot be set to 0 by a program (writing a 0 has no effect).

13.2.2 Interrupt Control Register 1 (TB5IC, TB4IC/U1BCNIC, TB3IC/U0BCNIC, BCNIC, DM0IC to DM3IC, ADIC, S0TIC to S2TIC, S0RIC to S2RIC, TA0IC to TA4IC, TB0IC to TB2IC, U5BCNIC, S5TIC, S5RIC to S7RIC, U6BCNIC/RTCTIC, S6TIC/RTCCIC, U7BCNIC, S7TIC, IICIC, SCLDAIC)

Interrupt Control Register 1

Symbol	Address	After Reset
Refer to the table below for symbols and addresses.		XXXX X000b

b7 b6 b5 b4 b3 b2 b1 b0

Bit Symbol	Bit Name	Function	RW
ILVL0	Interrupt priority 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 bit	0: Interrupt not requested 1: Interrupt requested	RW
— (b7-b4)	No register bits. If necessary, set to 0. Read as undefined value.		—

Symbol	Address
TB5IC	0045h
TB4IC/U1BCNIC	0046h
TB3IC/U0BCNIC	0047h
BCNIC	004Ah
DM0IC	004Bh
DM1IC	004Ch
DM2IC	0069h
DM3IC	006Ah
ADIC	004Eh
S0TIC	0051h
S1TIC	0053h
S2TIC	004Fh
S0RIC	0052h
S1RIC	0054h
S2RIC	0050h
TA0IC	0055h
TA1IC	0056h

Symbol	Address
TA2IC	0057h
TA3IC	0058h
TA4IC	0059h
TB0IC	005Ah
TB1IC	005Bh
TB2IC	005Ch
U5BCNIC/CEC1IC	006Bh
S5TIC/CEC2IC	006Ch
S5RIC	006Dh
S6RIC	0070h
S7RIC	0073h
U6BCNIC/RTCTIC	006Eh
S6TIC/RTCCIC	006Fh
U7BCNIC/PMC0IC	0071h
S7TIC/PMC1IC	0072h
IICIC	007Bh
SCLDAIC	007Ch

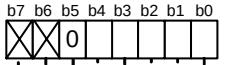
Rewrite this register at a point that does not generate an interrupt request.
When multiple interrupt sources share the register, select an interrupt source in registers IFSR2A and IFSR3A.

IR (Interrupt request bit) (b3)

Do not set the IR bit to 1 when it is 0.

13.2.3 Interrupt Control Register 2 (INT7IC, INT6IC, INT3IC, S4IC/INT5IC, S3IC/INT4IC, KUPIC, INT0IC to INT2IC)

Interrupt Control Register 2



Symbol Address After Reset

Refer to the table below for symbols and addresses. XX00 X000b

Bit Symbol	Bit Name	Function	RW
ILVL0	Interrupt priority 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 bit	0: Interrupt not requested 1: Interrupt requested	RW (1)
POL	Polarity select bit	0: Select falling edge 1: Select rising edge	RW
— (b5)	Reserved bit	Set to 0.	RW
— (b7-b6)	No register bits. If necessary, set to 0. Read as undefined value.		—

Symbol	Address
INT7IC	0042h
INT6IC	0043h
INT3IC	0044h
S4IC/INT5IC	0048h
S3IC/INT4IC	0049h

Symbol	Address
KUPIC	004Dh
INT0IC	005Dh
INT1IC	005Eh
INT2IC	005Fh

Rewrite this register at a point that does not generate an interrupt request.
When multiple interrupt sources share the register, select an interrupt source in the IFSR register.

IR (Interrupt request bit) (b3)

Do not set the IR bit to 1 when it is 0.

POL (Polarity select bit) (b4)

When the IFSRi bit in the IFSR register is 1 (both edges), set the POL bit in the INTiIC register to 0 (falling edge) (i = 0 to 5). Similarly, when bits IFSR30 and IFSR31 in the IFSR3A register are 1 (both edges), set the POL bit in registers INT6IC and INT7IC to 0 (falling edge).
Set the POL bit in the S3IC register to 0 (falling edge) when the IFSR6 bit in the IFSR register is 0 (SI/O3 selected). Set the POL bit in the S4IC register to 0 (falling edge) when the IFSR7 bit is 0 (SI/O4 selected).

13.2.4 Interrupt Source Select Register 3 (IFSR3A)

Interrupt Source Select Register 3

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
0			0	0	0			IFSR3A	0205h	00h

Bit Symbol	Bit Name	Function	RW
IFSR30	$\overline{\text{INT6}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
IFSR31	$\overline{\text{INT7}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
— (b4-b2)	Reserved bits	Set to 0.	RW
IFSR35	Interrupt request source select bit	0: UART6 start/stop condition detection, bus collision detection 1: Real-time clock cycle	RW
IFSR36	Interrupt request source select bit	0: UART6 transmission, NACK 1: Real-time clock alarm	RW
IFSR37	Key input interrupt polarity select bit	0: One edge 1: Both edges	RW

IFSR31 and IFSR30 ($\overline{\text{INT7}}$ and $\overline{\text{INT6}}$ interrupt polarity select bit) (b1-b0)

When setting this bit to 1 (both edges), make sure the corresponding POL bit in registers INT6IC and INT7IC is set to 0 (falling edge).

13.2.5 Interrupt Source Select Register 2 (IFSR2A)

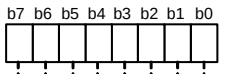
Interrupt Source Select Register 2

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
		0	0			0	0	IFSR2A	0206h	00h

Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved bits	Set to 0.	RW
IFSR22	Interrupt request source select bit	0: Not used 1: I^2C-bus interface	RW
IFSR23	Interrupt request source select bit	0: Not used 1: SCL/SDA	RW
— (b5-b4)	Reserved bits	Set to 0.	RW
IFSR26	Interrupt request source select bit	0: Timer B3 1: UART0 start/stop condition detection, bus collision detection	RW
IFSR27	Interrupt request source select bit	0: Timer B4 1: UART1 start/stop condition detection, bus collision detection	RW

13.2.6 Interrupt Source Select Register (IFSR)

Interrupt Source Select Register



Symbol	Address	After Reset
IFSR	0207h	00h

Bit Symbol	Bit Name	Function	RW
IFSR0	$\overline{\text{INT0}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
IFSR1	$\overline{\text{INT1}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
IFSR2	$\overline{\text{INT2}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
IFSR3	$\overline{\text{INT3}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
IFSR4	$\overline{\text{INT4}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
IFSR5	$\overline{\text{INT5}}$ interrupt polarity select bit	0: One edge 1: Both edges	RW
IFSR6	Interrupt request source select bit	0: SI/O3 1: $\overline{\text{INT4}}$	RW
IFSR7	Interrupt request source select bit	0: SI/O4 1: $\overline{\text{INT5}}$	RW

IFSR5 to IFSR0 ($\overline{\text{INT5}}$ to $\overline{\text{INT0}}$ interrupt polarity select bit) (b5 to b0)

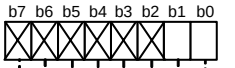
When setting this bit to 1 (both edges), make sure the POL bit in registers INT0IC to INT5IC are set to 0 (falling edge).

IFSR7, IFSR6 (Interrupt request source select bit) (b7, b6)

When setting this bit to 0 (SI/O3, SI/O4), make sure the POL bit in registers S3IC and S4IC are set to 0 (falling edge).

13.2.7 Address Match Interrupt Enable Register (AIER)

Address Match Interrupt Enable Register

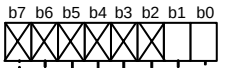


Symbol	Address	After Reset
AIER	020Eh	XXXX XX00b

Bit Symbol	Bit Name	Function	RW
AIER0	Address match interrupt 0 enable bit	0: Interrupt disabled 1: Interrupt enabled	RW
AIER1	Address match interrupt 1 enable bit	0: Interrupt disabled 1: Interrupt enabled	RW
— (b7-b2)	No register bits. If necessary, set to 0. Read as undefined value.		—

13.2.8 Address Match Interrupt Enable Register 2 (AIER2)

Address Match Interrupt Enable Register 2

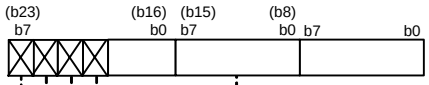


Symbol	Address	After Reset
AIER2	020Fh	XXXX XX00b

Bit Symbol	Bit Name	Function	RW
AIER20	Address match interrupt 2 enable bit	0: Interrupt disabled 1: Interrupt enabled	RW
AIER21	Address match interrupt 3 enable bit	0: Interrupt disabled 1: Interrupt enabled	RW
— (b7-b2)	No register bits. If necessary, set to 0. Read as undefined value.		—

13.2.9 Address Match Interrupt Register i (RMADi) (i = 0 to 3)

Address Match Interrupt Register i (i = 0 to 3)

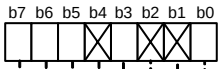


Symbol	Address	After Reset
RMAD0	0212h to 0210h	X0 0000h
RMAD1	0216h to 0214h	X0 0000h
RMAD2	021Ah to 0218h	X0 0000h
RMAD3	021Eh to 021Ch	X0 0000h

Function	Setting Range	RW
Address setting register for address match interrupt (b19 to b0)	00000h to FFFFFh	RW
No register bits. If necessary, set to 0. Read as undefined value.		—

13.2.10 Port Control Register (PCR)

Port Control Register



Symbol: PCR | Address: 0366h | After Reset: 0000 0XX0b

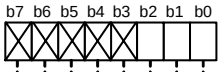
Bit Symbol	Bit Name	Function	RW
PCR0	Port P1 control bit	Operation performed when the P1 register is read 0: When the port is set to input, the input levels of pins P1_0 to P1_7 are read. When set to output, the port latch is read. 1: The port latch is read regardless of whether the port is set to input or output.	RW
— (b2-b1)	No register bits. If necessary, set to 0. Read as undefined.		—
PCR3	Key input enable bit ($\overline{KI4}$ to $\overline{KI7}$)	0: Enabled 1: Disabled	RW
— (b4)	No register bits. Read as undefined.		—
PCR5	$\overline{INT6}$ input enable bit	0: Enabled 1: Disabled	RW
PCR6	$\overline{INT7}$ input enable bit	0: Enabled 1: Disabled	RW
PCR7	Key input enable bit ($\overline{KI0}$ to $\overline{KI3}$)	0: Enabled 1: Disabled	RW

PCR3 (Key input enable bit) (b3)
PCR7 (Key input enable bit) (b7)

Set the PCR7 bit to 1 (key input disabled) when using pins AN4 to AN7 for analog input. Set the PCR3 bit to 1 (key input disabled) when using pins AN0 to AN3 for analog input.

13.2.11 $\overline{\text{NMI}}$ Digital Filter Register (NMIDF)

$\overline{\text{NMI}}$ Digital Filter Register



Symbol	Address	After Reset
NMIDF	0369h	XXXX X000b

Bit Symbol	Bit Name	Function	RW
NMIDF0	$\overline{\text{NMI}}$ filter sampling clock select bit	b2 b1 b0 0 0 0: No filter 0 0 1: CPU clock divided by 2 0 1 0: CPU clock divided by 4 0 1 1: CPU clock divided by 8 1 0 0: CPU clock divided by 16 1 0 1: CPU clock divided by 32 1 1 0: CPU clock divided by 64 1 1 1: CPU clock divided by 128	RW
NMIDF1			RW
NMIDF2			RW
— (b7-b3)	No register bits. If necessary, set to 0. Read as undefined value.		—

Change the NMIDF register under the following condition:

- The PM24 bit in the PM2 register is 0 ($\overline{\text{NMI}}$ interrupt disabled)

Once the PM24 bit is set to 1 ($\overline{\text{NMI}}$ interrupt enabled), it cannot be set to 0 by a program. Change the NMIDF register before setting the PM24 bit to 1.

13.3 Types of Interrupt

Figure 13.1 shows Types of Interrupt.

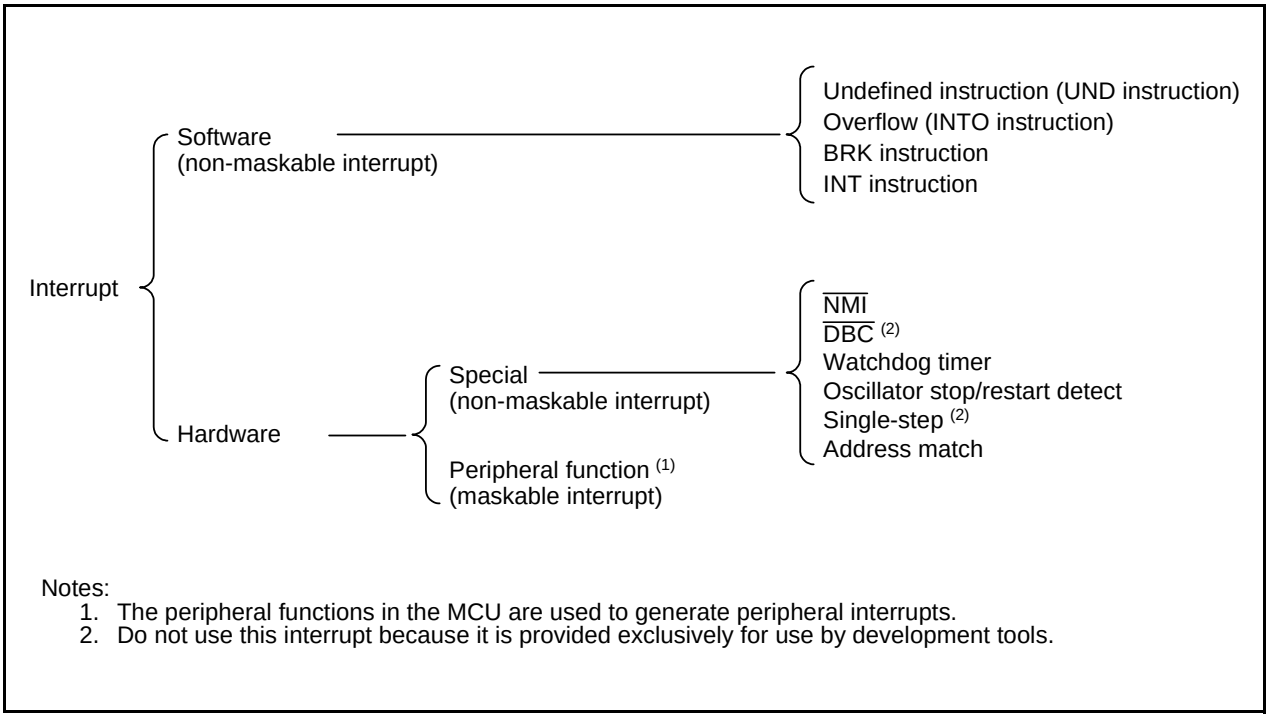


Figure 13.1 Types of Interrupt

- Maskable interrupt: The interrupt priority **can be changed** by enabling (disabling) an interrupt with the interrupt enable flag (I flag) or by using interrupt priority levels.
- Non-maskable interrupt: The interrupt priority **cannot be changed** by enabling (disabling) an interrupt with the interrupt enable flag (I flag) or by using interrupt priority levels.

13.4 Software Interrupts

A software interrupt occurs when executing instructions. Software interrupts are non-maskable interrupts.

13.4.1 Undefined Instruction Interrupt

An undefined instruction interrupt occurs when executing the UND instruction.

13.4.2 Overflow Interrupt

An overflow interrupt occurs when executing the INTO instruction with the O flag in the FLG register set to 1 (the operation resulted in an overflow). The following are instructions whose O flag changes by an arithmetic operation:
ABS, ADC, ADCF, ADD, CMP, DIV, DIVU, DIVX, NEG, RMPA, SBB, SHA, and SUB

13.4.3 BRK Interrupt

A BRK interrupt occurs when the BRK instruction is executed.

13.4.4 INT Instruction Interrupt

An INT instruction interrupt occurs when the INT instruction is executed. Software interrupt numbers 0 to 63 can be specified for the INT instruction. Because software interrupt numbers 2 to 31, 41 to 51, 59, and 60 are assigned to peripheral function interrupts, the same interrupt routine used for peripheral function interrupts can be executed by executing the INT instruction.
For software interrupt numbers 0 to 31, the U flag is saved on the stack during instruction execution and is cleared to 0 (ISP selected) before executing an interrupt sequence. The U flag is restored from the stack when returning from the interrupt routine. For software interrupt numbers 32 to 63, the U flag does not change state during instruction execution, and the SP selected at the time is used.

13.5 Hardware Interrupts

Hardware interrupts are classified into two types: special interrupts and peripheral function interrupts.

13.5.1 Special Interrupts

Special interrupts are non-maskable interrupts.

13.5.1.1 $\overline{\text{NMI}}$ Interrupt

An $\overline{\text{NMI}}$ interrupt is generated when input on the $\overline{\text{NMI}}$ pin changes state from high to low. For details about the $\overline{\text{NMI}}$ interrupt, refer to 13.9 “$\overline{\text{NMI}}$ Interrupt”.

13.5.1.2 $\overline{\text{DBC}}$ Interrupt

Do not use this interrupt because it is provided exclusively for use by development tools.

13.5.1.3 Watchdog Timer Interrupt

The interrupt is generated by the watchdog timer. Once a watchdog timer interrupt is generated, be sure to refresh the watchdog timer. For details about the watchdog timer, refer to 14. “Watchdog Timer”.

13.5.1.4 Oscillator Stop/Restart Detect Interrupt

The interrupt is generated by the oscillator stop/restart detect function. For details about this function, refer to 8. “Clock Generator”.

13.5.1.5 Single-Step Interrupt

Do not use this interrupt because it is provided exclusively for use by development tools.

13.5.1.6 Address Match Interrupt

When the AIER0 or AIER1 bit in the AIER register, or the AIER20 or AIER21 bit in the AIER2 register is 1 (address match interrupt enabled), an address match interrupt is generated immediately before executing an instruction at the address indicated by the corresponding registers RMAD0 to RMAD3. For details about the address match interrupt, refer to 13.11 “Address Match Interrupt”.

13.5.2 Peripheral Function Interrupts

A peripheral function interrupt occurs when a request from a peripheral function in the MCU is acknowledged. Peripheral function interrupts are maskable interrupts. See Table 13.6 and Table 13.7 “Relocatable Vector Tables”. Refer to the descriptions of each function for details on how the corresponding peripheral function interrupt is generated.

13.6 Interrupts and Interrupt Vectors

One interrupt vector consists of 4 bytes. Set the start address of each interrupt routine in the respective interrupt vectors. When an interrupt request is accepted, the CPU branches to the address set in the corresponding interrupt vector. Figure 13.2 shows an Interrupt Vector.

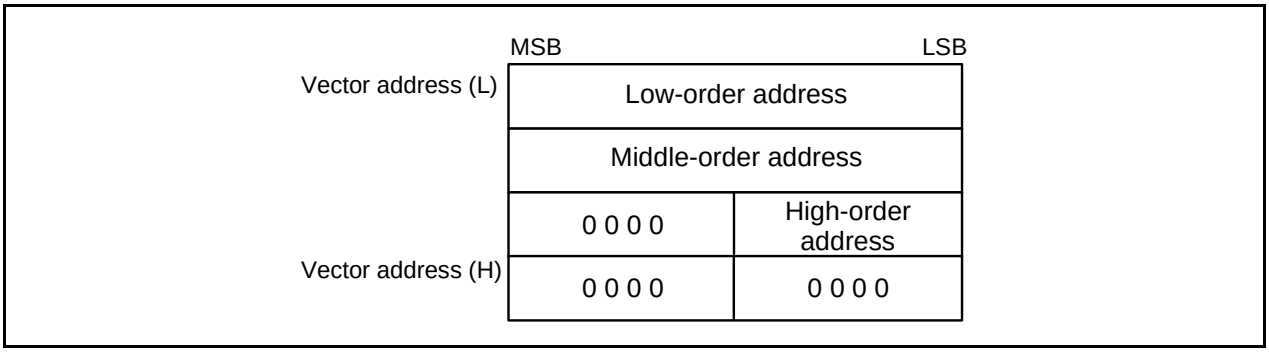


Figure 13.2 Interrupt Vector

13.6.1 Fixed Vector Tables

The fixed vector tables are allocated to addresses from FFFDCh to FFFFFh. Table 13.5 lists the Fixed Vector Tables. In the flash memory MCU version, the vector addresses (H) of fixed vectors are used for the ID code check function and OFS1 address. For details, refer to 24. “Flash Memory”.

Table 13.5 Fixed Vector Tables

Interrupt Source	Vector Table Addresses Address (L) to Address (H)	Reference
Undefined instruction (UND instruction)	FFFDCh to FFFDFh	M16C/60, M16C/20, M16C/Tiny Series User’s Manual: Software
Overflow (INTO instruction)	FFFE0h to FFFE3h	
BRK instruction (2)	FFFE4h to FFFE7h	
Address match	FFFE8h to FFFEBh	13.11 “Address Match Interrupt”
Single-step (1)	FFFECh to FFFEFh	—
Watchdog timer, oscillator stop/restart detect	FFFF0h to FFFF3h	14. “Watchdog Timer” 8. “Clock Generator”
$\overline{\text{DBC}}$ (1)	FFFF4h to FFFF7h	—
$\overline{\text{NMI}}$	FFFF8h to FFFFBh	13.9 “$\overline{\text{NMI}}$ Interrupt”
Reset	FFFFCh to FFFFFh	6. “Resets”

Notes:
1. Do not use this interrupt because it is provided exclusively for use by development tools.
2. If the content of address FFFE6h is FFh, program execution starts from the address shown by the vector in the relocatable vector table.

13.6.2 Relocatable Vector Tables

The 256 bytes beginning with the start address set in the INTB register compose a relocatable vector table area. Setting an even address in the INTB register results in the interrupt sequence being executed faster than setting an odd address.

Table 13.6 Relocatable Vector Tables (1/2)

Interrupt Source	Vector Address (1) Address (L) to Address (H)	Software Interrupt Number	Reference
INT instruction interrupt (5)	+0 to +3 (0000h to 0003h) to +252 to +255 (00FCh to 00FFh)	0 to 63	M16C/60, M16C/20, M16C/Tiny Series User's Manual: Software
BRK instruction (5)	+0 to +3 (0000h to 0003h)	0	
$\overline{\text{INT7}}$	+8 to +11 (0008h to 000Bh)	2	13.8 "$\overline{\text{INT}}$ Interrupt"
$\overline{\text{INT6}}$	+12 to +15 (000Ch to 000Fh)	3	
$\overline{\text{INT3}}$	+16 to +19 (0010h to 0013h)	4	
Timer B5	+20 to +23 (0014h to 0017h)	5	17. "Timer B"
Timer B4, UART1 start/stop condition detection, bus collision detection (4)	+24 to +27 (0018h to 001Bh)	6	17. "Timer B" 19. "Serial Interface UARTi (i = 0 to 2, 5 to 7)"
Timer B3, UART0 start/stop condition detection, bus collision detection (4)	+28 to +31 (001Ch to 001Fh)	7	
SI/O4, $\overline{\text{INT5}}$ (2)	+32 to +35 (0020h to 0023h)	8	13.8 "$\overline{\text{INT}}$ Interrupt" 20. "Serial Interface SI/O3 and SI/O4"
SI/O3, $\overline{\text{INT4}}$ (2)	+36 to +39 (0024h to 0027h)	9	
UART2 start/stop condition detection, bus collision detection (4)	+40 to +43 (0028h to 002Bh)	10	19. "Serial Interface UARTi (i = 0 to 2, 5 to 7)"
DMA0	+44 to +47 (002Ch to 002Fh)	11	15. "DMAC"
DMA1	+48 to +51 (0030h to 0033h)	12	
Key input interrupt	+52 to +55 (0034h to 0037h)	13	13.10 "Key Input Interrupt"
A/D converter	+56 to +59 (0038h to 003Bh)	14	22. "A/D Converter"
UART2 transmit, NACK2 (3)	+60 to +63 (003Ch to 003Fh)	15	19. "Serial Interface UARTi (i = 0 to 2, 5 to 7)"
UART2 receive, ACK2 (3)	+64 to +67 (0040h to 0043h)	16	
UART0 transmit, NACK0 (3)	+68 to +71 (0044h to 0047h)	17	
UART0 receive, ACK0 (3)	+72 to +75 (0048h to 004Bh)	18	
UART1 transmit, NACK1 (3)	+76 to +79 (004Ch to 004Fh)	19	
UART1 receive, ACK1 (3)	+80 to +83 (0050h to 0053h)	20	
Timer A0	+84 to +87 (0054h to 0057h)	21	16. "Timer A"
Timer A1	+88 to +91 (0058h to 005Bh)	22	
Timer A2	+92 to +95 (005Ch to 005Fh)	23	
Timer A3	+96 to +99 (0060h to 0063h)	24	
Timer A4	+100 to +103 (0064h to 0067h)	25	

Notes:
1. Address relative to address in INTB.
2. Use bits IFSR6 and IFSR7 in the IFSR register to select a source.
3. In I^2C mode, NACK and ACK are interrupt sources.
4. Use bits IFSR26 and IFSR27 in the IFSR2A register to select a source.
5. These interrupts cannot be disabled using the I flag.

Table 13.7 Relocatable Vector Tables (2/2)

Interrupt Source	Vector Address (1) Address (L) to Address (H)	Software Interrupt Number	Reference
Timer B0	+104 to +107 (0068h to 006Bh)	26	17. “Timer B”
Timer B1	+108 to +111 (006Ch to 006Fh)	27	
Timer B2	+112 to +115 (0070h to 0073h)	28	
$\overline{\text{INT0}}$	+116 to +119 (0074h to 0077h)	29	13.8 “$\overline{\text{INT}}$ Interrupt”
$\overline{\text{INT1}}$	+120 to +123 (0078h to 007Bh)	30	
$\overline{\text{INT2}}$	+124 to +127 (007Ch to 007Fh)	31	
DMA2	+164 to +167 (00A4h to 00A7h)	41	15. “DMAC”
DMA3	+168 to +171 (00A8h to 00ABh)	42	
UART5 start/stop condition detection, bus collision detection	+172 to +175 (00ACh to 0AFh)	43	19. “Serial Interface UARTi (i = 0 to 2, 5 to 7)”
UART5 transmit, NACK5 (2)	+176 to +179 (00B0h to 00B3h)	44	
UART5 receive, ACK5 (2)	+180 to +183 (00B4h to 00B7h)	45	
UART6 start/stop condition detection, bus collision detection, real-time clock period (3)	+184 to +187 (00B8h to 00BBh)	46	18. “Real-Time Clock” 19. “Serial Interface UARTi (i = 0 to 2, 5 to 7)”
UART6 transmit, NACK6, real-time clock alarm (2, 3)	+188 to +191 (00BCh to 00BFh)	47	
UART6 receive, ACK6 (2)	+192 to +195 (00C0h to 00C3h)	48	
UART7 start/stop condition detection, bus collision detection	+196 to +199 (00C4h to 00C7h)	49	19. “Serial Interface UARTi (i = 0 to 2, 5 to 7)”
UART7 transmit, NACK7 (2)	+200 to +203 (00C8h to 00CBh)	50	
UART7 receive, ACK7 (2)	+204 to +207 (00CCh to 00CFh)	51	
I^2C-bus interface interrupt (4)	+236 to +239 (00ECh to 00EFh)	59	21. “Multi-Master I^2C-bus Interface”
SCL/SDA interrupt (4)	+240 to +243 (00F0h to 00F3h)	60	

Notes:
1. Address relative to address in INTB.
2. In I^2C mode, NACK and ACK are the interrupt sources.
3. Use bits IFSR35 and IFSR36 in the IFSR3A register to select a source.
4. Use bits IFSR22 and IFSR23 in the IFSR2A register to select a source.

13.7 Interrupt Control

13.7.1 Maskable Interrupt Control

The settings of enabling/disabling the maskable interrupts and of the acceptance priority are explained below. Note that these explanations do not apply to non-maskable interrupts.
Use the I flag in the FLG register, IPL, and bits ILVL2 to ILVL0 in the corresponding interrupt control register to enable or disable a maskable interrupt. Whether an interrupt is requested or not is indicated by the IR bit in the corresponding interrupt control register.

13.7.1.1 I Flag

The I flag enables or disables maskable interrupts. Setting the I flag to 1 (enabled) enables maskable interrupts. Setting the I flag to 0 (disabled) disables all maskable interrupts.

13.7.1.2 IR Bit

The IR bit is set to 1 (interrupt requested) when an interrupt request is generated. Then, when the interrupt request is accepted, the IR bit is automatically set to 0 (interrupt not requested).
The IR bit can be set to 0 by a program. Do not write 1 to this bit.

13.7.1.3 Bits ILVL2 to ILVL0 and IPL

Interrupt priority levels can be set using bits ILVL2 to ILVL0.
Table 13.8 lists the Settings of Interrupt Priority Levels and Table 13.9 lists the Interrupt Priority Levels Enabled by IPL.

An interrupt request is accepted under the following conditions.

- I flag = 1
- IR bit = 1
- Interrupt priority level > IPL

The I flag, IR bit, bits ILVL2 to ILVL0 and IPL are independent each other. In no case do they affect one another.

Table 13.8 Settings of Interrupt Priority Levels

Bits ILVL2 to ILVL0	Interrupt Priority Level	Priority
000b	Level 0 (interrupt disabled)	—
001b	Level 1	Low
010b	Level 2	↓
011b	Level 3	↓
100b	Level 4	↓
101b	Level 5	↓
110b	Level 6	↓
111b	Level 7	High

Table 13.9 Interrupt Priority Levels Enabled by IPL

IPL	Enabled Interrupt Priority Levels
000b	Level 1 and above are enabled
001b	Level 2 and above are enabled
010b	Level 3 and above are enabled
011b	Level 4 and above are enabled
100b	Level 5 and above are enabled
101b	Level 6 and above are enabled
110b	Level 7 and above are enabled
111b	All maskable interrupts are disabled

13.7.2 Interrupt Sequence

The interrupt sequence is explained here. The sequence starts when an interrupt request is accepted and ends when the interrupt routine is executed.
If an interrupt request occurs during execution of an instruction, the processor determines its priority after the execution of the instruction is completed, and transfers control to the interrupt sequence from the next cycle. However, if an interrupt occurs during execution of either the SMOVB, SMOVF, SSTR, or RMPA instruction, the processor temporarily suspends the instruction being executed, and transfers control to the interrupt sequence.
The CPU behavior during the interrupt sequence is described below. Figure 13.3 shows Time Required for Executing Interrupt Sequence.

(1) The CPU obtains interrupt information (interrupt number and interrupt request level) by reading address 00000h. Then, the IR bit applicable to the interrupt information is set to 0 (interrupt not requested).
(2) The FLG register, prior to the interrupt sequence, is saved to a temporary register (1) within the CPU.
(3) Flags I, D, and U in the FLG register are set as follows:
The I flag is set to 0 (interrupt disabled)
The D flag is set to 0 (single-step interrupt disabled).
The U flag is set to 0 (ISP selected).
Note that the U flag does not change states when an INT instruction for software interrupt numbers 32 to 63 is executed.
(4) The temporary register (1) within the CPU is saved on the stack.
(5) The PC is saved on the stack.
(6) The interrupt priority level of the acknowledged interrupt is set in the IPL.
(7) The start address of the relevant interrupt routine set in the interrupt vector is stored in the PC.

After the interrupt sequence is completed, an instruction is executed from the starting address of the interrupt routine.

Note:
1. Temporary registers cannot be modified by users.

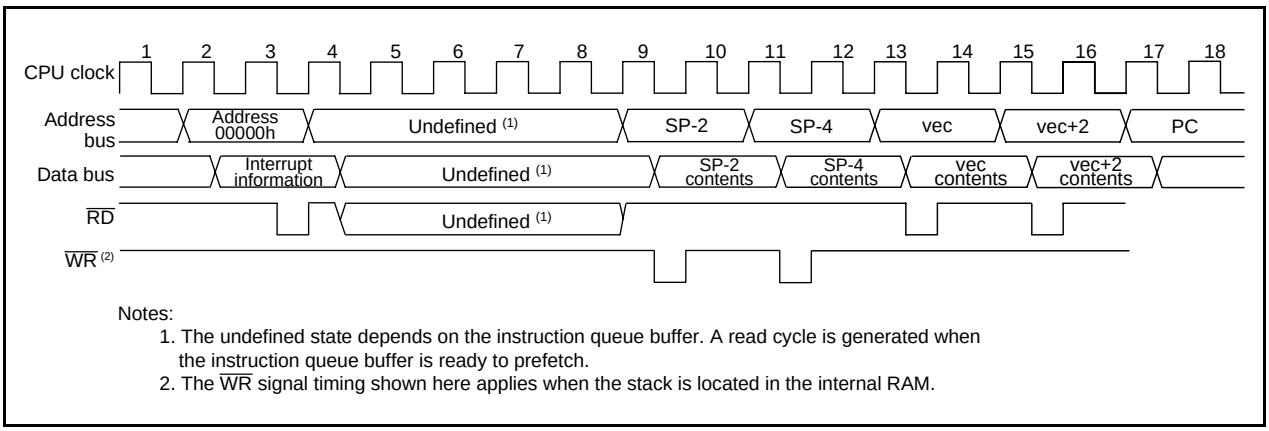
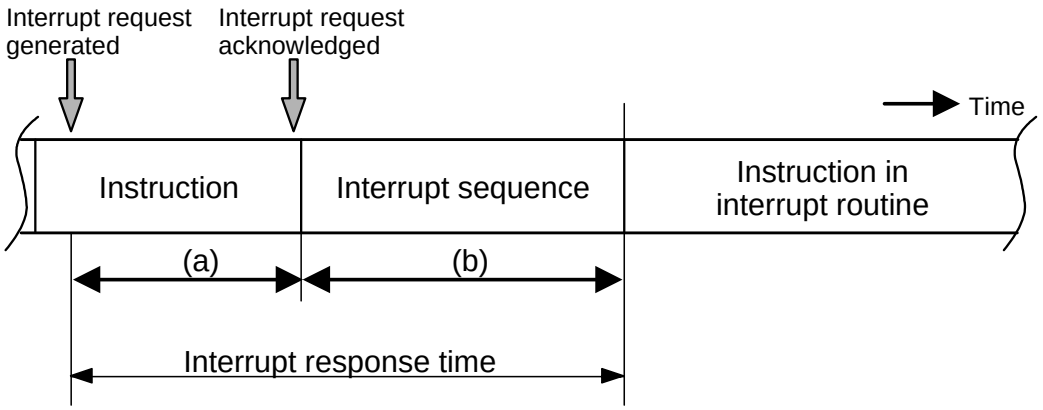


Figure 13.3 Time Required for Executing Interrupt Sequence

13.7.3 Interrupt Response Time

Figure 13.4 shows the Interrupt Response Time. The interrupt response or interrupt acknowledge time denotes the time from when an interrupt request is generated until the first instruction in the interrupt routine is executed. Specifically, it consists of the time from when an interrupt request is generated until the executing instruction is completed ((a) in Figure 13.4) and the time during which the interrupt sequence is executed ((b) in Figure 13.4).



(a) The time from when an interrupt request is generated until the instruction currently executing is completed. The length of this time varies with the instruction being executed. The DIVX instruction requires the longest time, which is equal to 30 cycles (no wait state, and when the divisor is a register).

(b) The time during which the interrupt sequence is executed. For details, see the table below. Note, however, that the values in this table must be increased by two cycles for the $\overline{\text{DBC}}$ interrupt and by one cycle for the address match and single-step interrupts.

Interrupt Vector Address	SP Value	16-Bit Bus, No Wait States	8-Bit Bus, No Wait States
Even	Even	18 cycles	20 cycles
Even	Odd	19 cycles	20 cycles
Odd	Even	19 cycles	20 cycles
Odd	Odd	20 cycles	20 cycles

Figure 13.4 Interrupt Response Time

13.7.4 Variation of IPL When Interrupt Request is Accepted

When a maskable interrupt request is accepted, the interrupt priority level of the accepted interrupt is set in the IPL.

When a software interrupt or special interrupt request is accepted, one of the interrupt priority levels listed in Table 13.10 is set in the IPL. Table 13.10 lists the IPL Level Set in IPL When Software or Special Interrupt is Accepted.

Table 13.10 IPL Level Set in IPL When Software or Special Interrupt is Accepted

Interrupt Source	Level Set in IPL
Watchdog timer, $\overline{\text{NMI}}$, oscillator stop/restart detect	7
Software, address match, $\overline{\text{DBC}}$, single-step	Not changed

13.7.5 Saving Registers

In the interrupt sequence, the FLG register and PC are saved on the stack.
At this time, the 4 high-order bits of the PC and the 4 high-order (IPL) and 8 low-order bits in the FLG register, 16 bits in total, are saved on the stack first. Next, the 16 low-order bits of the PC are saved. Figure 13.5 shows the Stack Status Before and After Acceptance of Interrupt Request.
The other necessary registers must be saved by a program at the beginning of the interrupt routine. Use the PUSHM instruction, and all registers except SP can be saved with a single instruction.

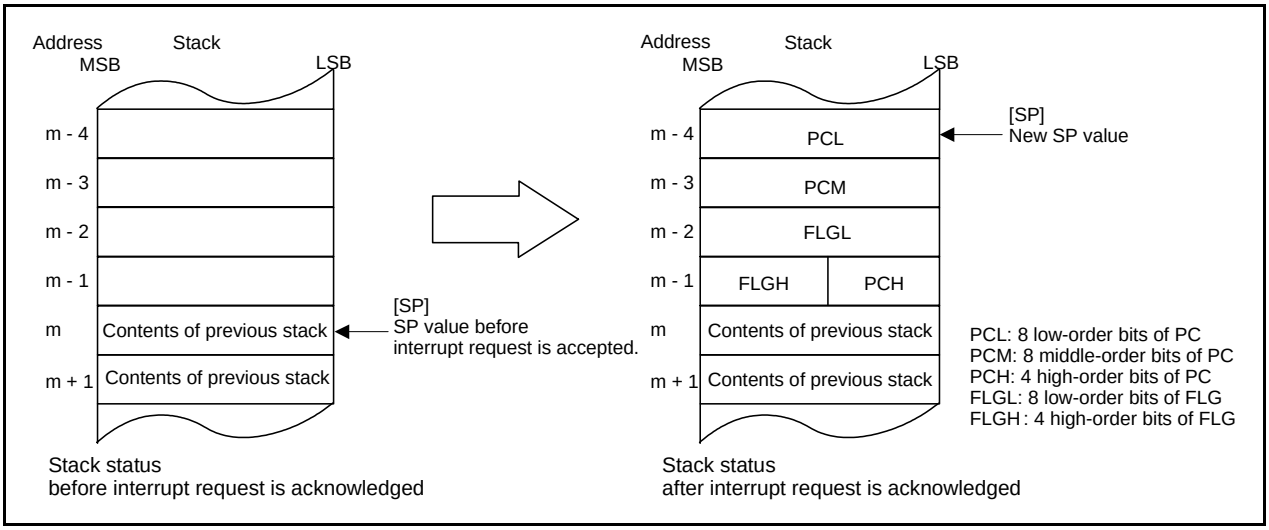


Figure 13.5 Stack Status Before and After Acceptance of Interrupt Request

The register save operation carried out in the interrupt sequence is dependent on whether the SP (1), at the time of acceptance of an interrupt request, is even or odd. If the SP (1) is even, the FLG register and the PC are saved 16 bits at a time. If odd, they are saved in two steps, 8 bits at a time. Figure 13.6 shows the Register Save Operation.

Note:
1. When an INT instruction with software numbers 32 to 63 has been executed, it is the SP indicated by the U flag. Otherwise, it is the ISP.

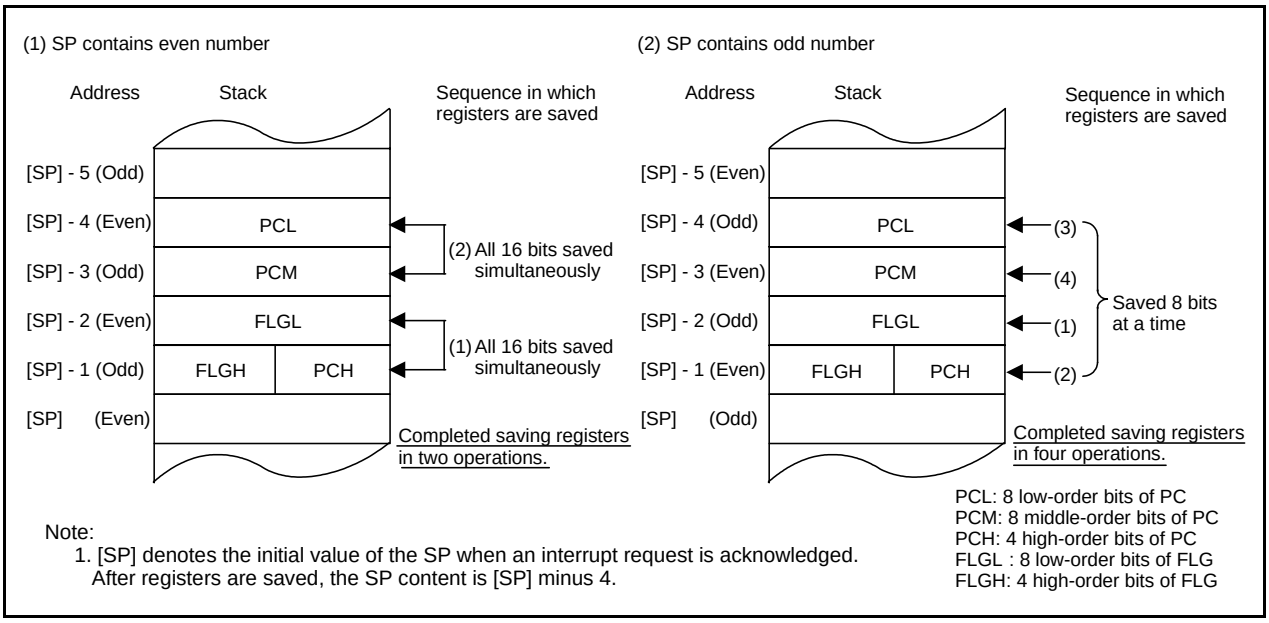


Figure 13.6 Register Save Operation

13.7.6 Returning from an Interrupt Routine

The FLG register and PC saved in the stack immediately before entering the interrupt sequence are restored from the stack by executing the REIT instruction at the end of the interrupt routine. Then, the CPU returns to the program which was being executed before the interrupt request was accepted.
Restore the other registers saved by a program within the interrupt routine using the POPM or another instruction before executing the REIT instruction.
The register bank is switched back to the bank used prior to the interrupt sequence by the REIT instruction.

13.7.7 Interrupt Priority

If two or more interrupt requests occur at the same sampling points (the point in time at which interrupt requests are detected), the interrupt with the highest priority is acknowledged.
For maskable interrupts (peripheral function interrupts), any priority level can be selected using bits ILVL2 to ILVL0. However, if two or more maskable interrupts have the same priority level, their interrupt priority is selected by hardware, with the highest priority interrupt accepted.
The watchdog timer interrupt and other special interrupts have their priority levels set in hardware. Figure 13.7 shows the Hardware Interrupt Priority.
Software interrupts are not affected by the interrupt priority. When an instruction is executed, control always branches to the interrupt routine.

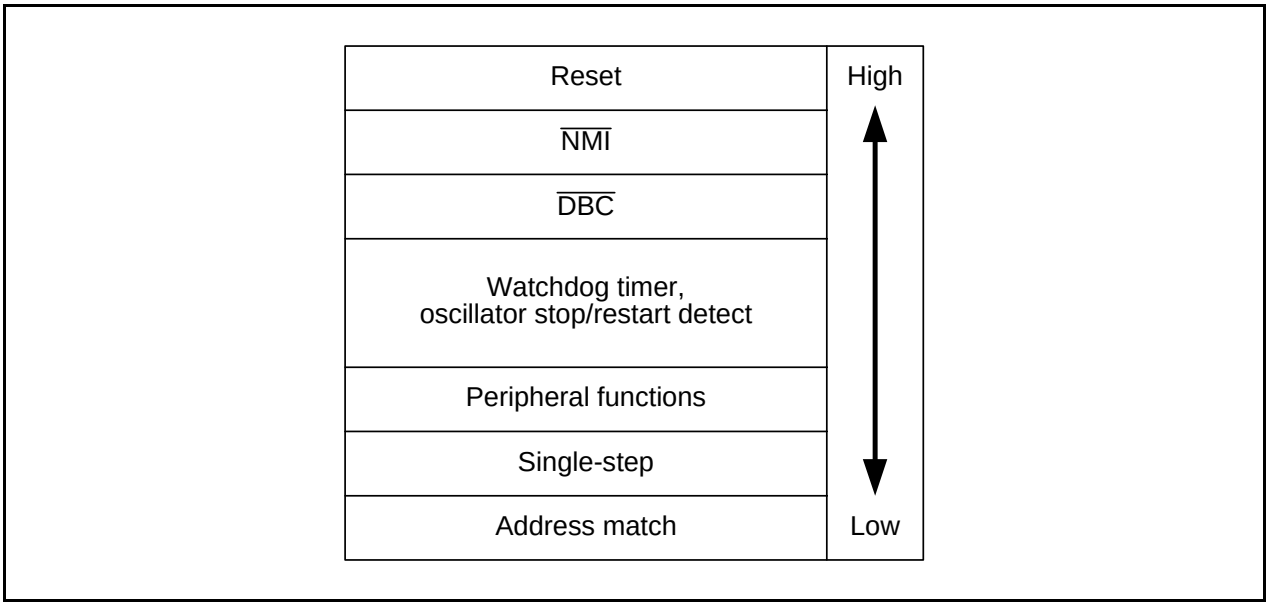


Figure 13.7 Hardware Interrupt Priority

13.7.8 Interrupt Priority Level Select Circuit

The interrupt priority level select circuit selects the highest priority interrupt among sampled interrupt requests at the same sampling point.
Figure 13.8 shows the Interrupt Priority Select Circuit 1, and Figure 13.9 shows the Interrupt Priority Select Circuit 2.

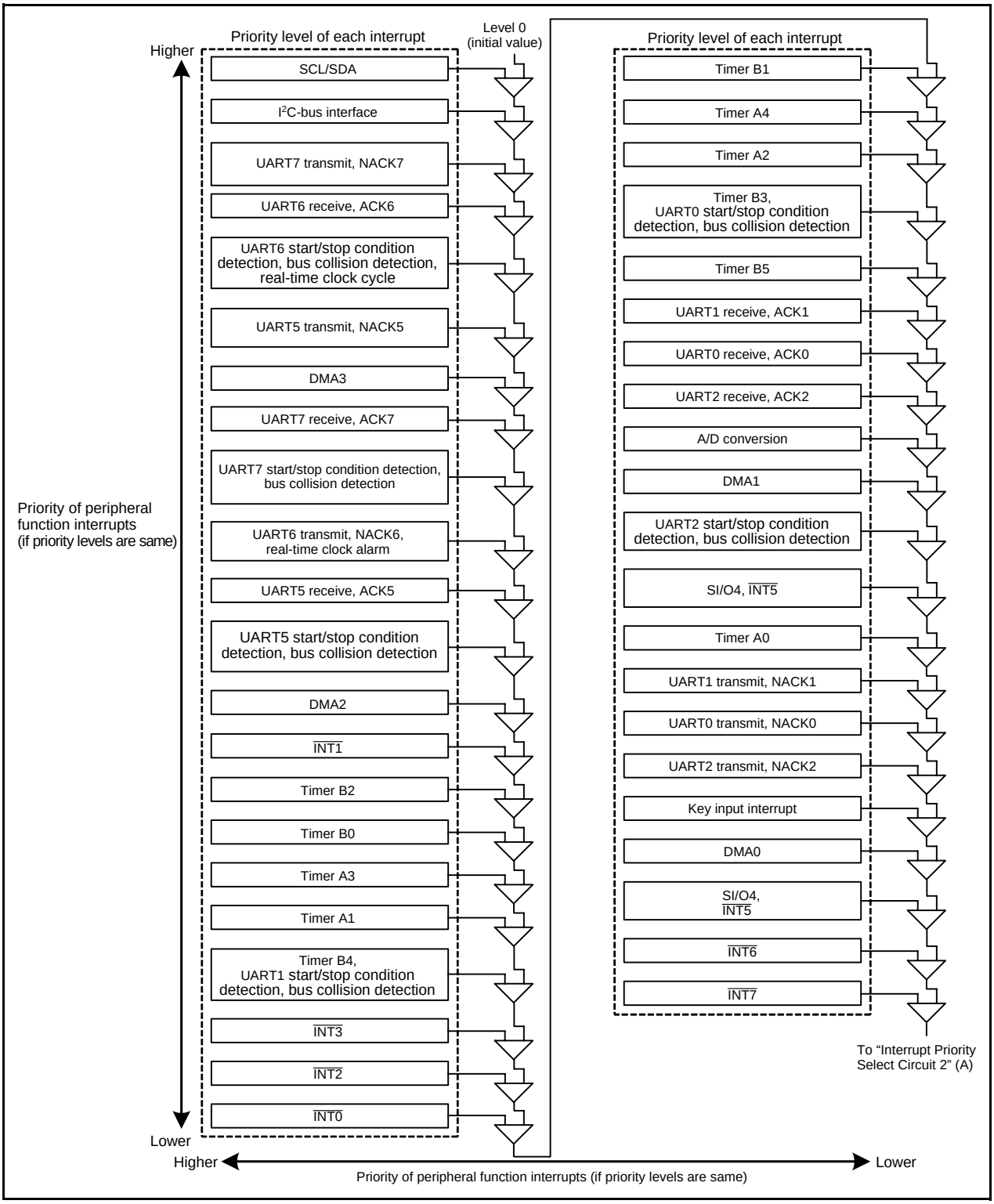


Figure 13.8 Interrupt Priority Select Circuit 1

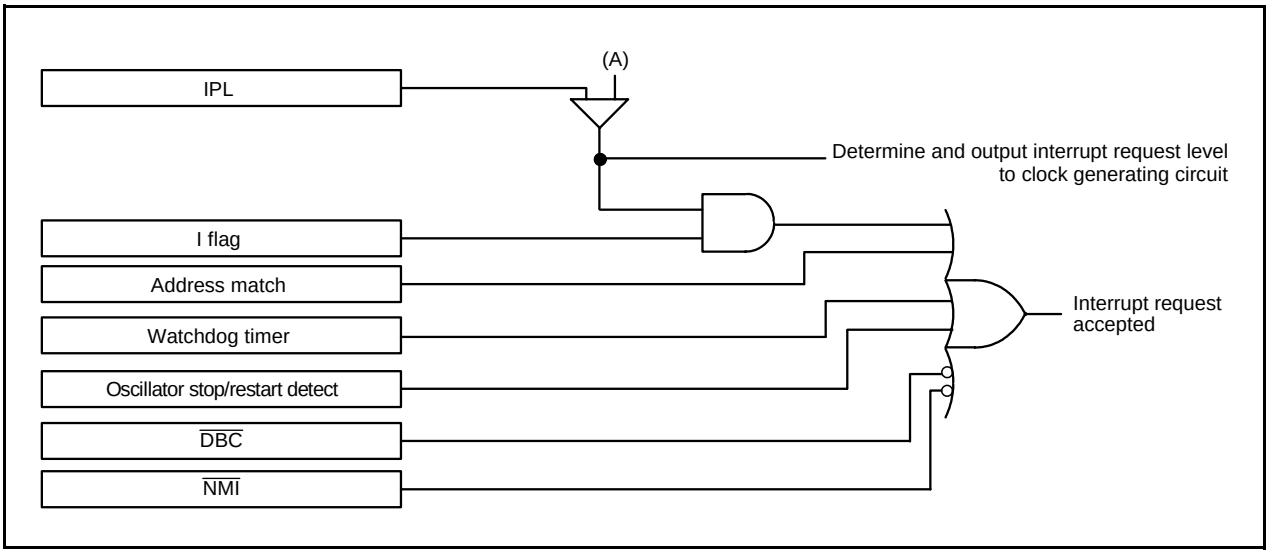


Figure 13.9 Interrupt Priority Select Circuit 2

13.7.9 Multiple Interrupts

The following shows the internal bit states when control has branched to an interrupt routine.

- I flag = 0 (interrupt disabled)
- IR bit = 0 (interrupt not requested)
- Interrupt priority level = IPL

By setting the I flag to 1 (interrupt enabled) in the interrupt routine, an interrupt request with higher priority than the IPL can be acknowledged.

The interrupt requests not acknowledged because of their low interrupt priority level are kept pending. When the IPL is restored by an REIT instruction and interrupt priority is resolved against it, the pending interrupt request is acknowledged if the following condition is met:

Interrupt priority level of pending interrupt request > Restored IP

13.8 $\overline{\text{INT}}$ Interrupt

The $\overline{\text{INTi}}$ interrupt (i = 1 to 5) is triggered by the edges of external inputs. The edge polarity is selected using the IFSRi bit in the IFSR register, or the IFSR30 or IFSR31 bit in the IFSR3A register.

The $\overline{\text{INT4}}$ and $\overline{\text{INT5}}$ each share an interrupt vector and interrupt control register with SI/O3 and SI/O4, respectively. To use the $\overline{\text{INT4}}$ interrupt, set the IFSR6 bit in the IFSR register to 1 ($\overline{\text{INT4}}$). To use the $\overline{\text{INT5}}$ interrupt, set the IFSR7 bit in the IFSR register to 1 ($\overline{\text{INT5}}$).

After modifying the IFSR6 or IFSR7 bit, set the corresponding IR bit to 0 (interrupt not requested) before enabling the interrupt.

$\overline{\text{INT0}}$, $\overline{\text{INT6}}$, and $\overline{\text{INT7}}$ are connected to the internal PLC modem. No external pin is provided.

13.9 $\overline{\text{NMI}}$ Interrupt

An $\overline{\text{NMI}}$ interrupt is generated when input to the $\overline{\text{NMI}}$ pin changes state from high to low. The $\overline{\text{NMI}}$ interrupt is a non-maskable interrupt. To use the $\overline{\text{NMI}}$ interrupt, set the PM24 bit in the PM2 register to 1 ($\overline{\text{NMI}}$ interrupt enabled). The $\overline{\text{NMI}}$ input uses the digital filter. Refer to 12. “Programmable I/O Ports” for the digital filter. Figure 13.10 shows $\overline{\text{NMI}}$ Interrupt Block Diagram.

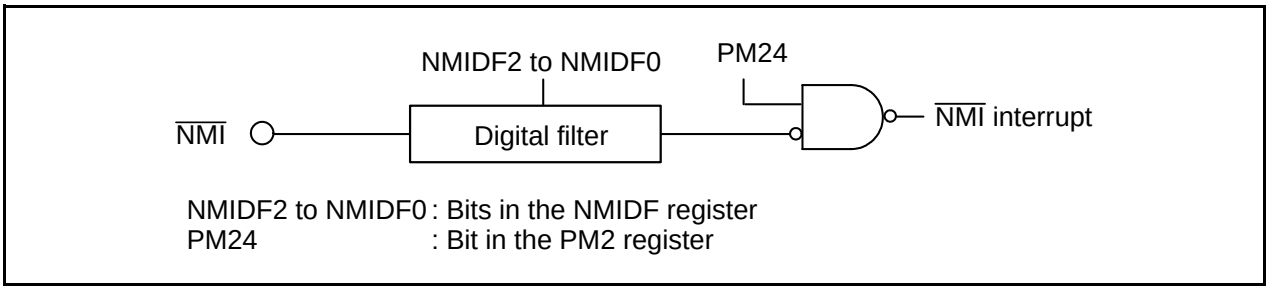


Figure 13.10 $\overline{\text{NMI}}$ Interrupt Block Diagram

13.10 Key Input Interrupt

An interrupt is generated by input to any pins of multiple pins.

When the PCR7 bit in the PCR register is 0 ($\overline{\text{KI0}}$ to $\overline{\text{KI3}}$ key input enabled), any pins which are set to input by setting bits PD10_4 to PD10_7 in the PD10 register to 0 (input) are used for the key input interrupt.

When using any pins from $\overline{\text{KI0}}$ to $\overline{\text{KI3}}$ for the key input interrupt, do not use all four pins AN4 to AN7 as analog input pins.

Also, when the PCR3 bit in the PCR register is 0 ($\overline{\text{KI4}}$ to $\overline{\text{KI7}}$ key input enabled), any pins which are set to input by setting bits P10_0 to P10_3 in the PD10 register to 0 (input) are used for the key input interrupt.

When using any pin from $\overline{\text{KI4}}$ to $\overline{\text{KI7}}$ for the key input interrupt, do not use all four pins AN0 to AN3 as analog input pins.

When waveform input to the pins for the key input interrupt matches waveform selected by the IFSR37 bit in the IFSR3A register and the POL bit in the KUPIC register, the IR bit in the KUPIC register becomes 1 (key input interrupt requested).

Key input interrupts can be used as a key-on wake up function for getting the MCU out of wait or stop mode.

Figure 13.11 shows Block Diagram of Key Input Interrupt.

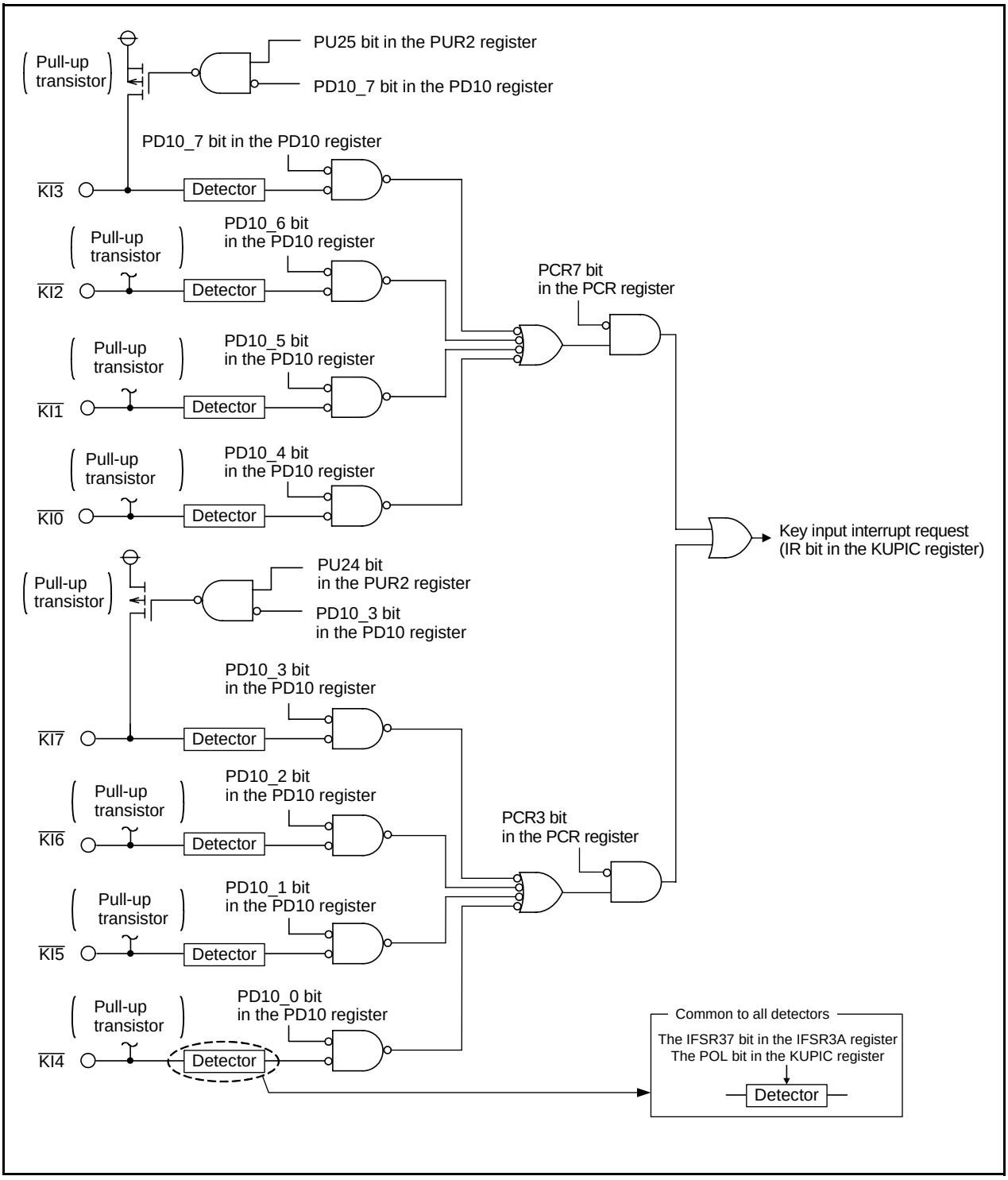


Figure 13.11 Block Diagram of Key Input Interrupt

13.11 Address Match Interrupt

An address match interrupt is generated immediately before executing the instruction at the address indicated by the RMADi register (i = 0 to 3). Set the start address of any instruction in the RMADi register. Use bits AIER0 and AIER1 in the AIER register, and bits AIER20 and AIER21 in the AIER2 register to enable or disable the interrupt. Note that the address match interrupt is unaffected by the I flag and IPL. When an address match interrupt request is acknowledged, the value of the PC that is saved to the stack area (refer to 13.7.5 “Saving Registers”) varies depending on the instruction at the address indicated by the RMADi register. (The value of the PC that is saved to the stack area is not the correct return address.) Therefore, follow one of the methods described below to return from the address match interrupt.

- Rewrite the contents of the stack and then use the REIT instruction to return.
- Restore the stack to its previous state by using the POP or other instructions before the interrupt request was accepted and then use a jump instruction to return.

Table 13.11 Value of PC Saved on Stack Area When Address Match Interrupt Request Accepted

Instruction at the Address Indicated by the RMADi Register	Value of the PC That is Saved to the Stack Area
• 16-bit operation code instructions • Instruction shown below among 8-bit operation code instructions ADD.B:S #IMM8, dest SUB.B:S #IMM8, dest AND.B:S #IMM8, dest OR.B:S #IMM8, dest MOV.B:S #IMM8, dest STZ #IMM8, dest STNZ #IMM8, dest STZX #IMM81, #IMM82,dest CMP.B:S #IMM8, dest PUSHM src POPM dest JMPS #IMM8 JSRS #IMM8 MOV.B:S #IMM, dest (however, dest = A0 or A1)	The address indicated by the RMADi register +2
Instructions not listed above	The address indicated by the RMADi register +1

Refer to 13.7.5 “Saving Registers” for PC values saved to the stack area.

Table 13.12 Relationship between Address Match Interrupt Sources and Associated Registers

Address Match Interrupt Sources	Address Match Interrupt Enable Bit	Address Match Interrupt Register
Address match interrupt 0	AIER0	RMAD0
Address match interrupt 1	AIER1	RMAD1
Address match interrupt 2	AIER20	RMAD2
Address match interrupt 3	AIER21	RMAD3

13.12 Non-Maskable Interrupt Source Discrimination

The watchdog timer interrupt and oscillator stop/restart detect interrupt share the same interrupt vector. When using some functions together, read the detect flags of the events in an interrupt processing program, and determine the source of the interrupt. Table 13.13 lists Bits Used for Non-Maskable Interrupt Source Discrimination.

Table 13.13 Bits Used for Non-Maskable Interrupt Source Discrimination

Interrupt	Detect Flag	
	Bit Position	Function
Watchdog timer	VW2C3 bit in the VW2C register (watchdog timer underflow detected)	0: not detected 1: detected
Oscillator stop/restart detect	CM22 bit in the CM2 register (oscillator stop/restart detected)	

13.13 Notes on Interrupts

13.13.1 Reading Address 00000h

Do not read address 00000h by a program. When a maskable interrupt request is accepted, the CPU reads interrupt information (interrupt number and interrupt request priority level) from address 00000h during the interrupt sequence. At this time, the IR bit of the accepted interrupt is cleared to 0 (interrupt not requested).

If address 00000h is read by a program, the IR bit for the interrupt which has the highest priority among the enabled interrupts becomes 0. Thus, some problems may be caused: interrupts may be canceled, or an unexpected interrupt request may be generated.

13.13.2 SP Setting

Set a value in the SP (USP, ISP) before accepting an interrupt. The SP (USP, ISP) is set to 0000h after reset. Therefore, if an interrupt is accepted before setting a value in the SP (USP, ISP), the program may go out of control.

Set a value in the ISP at the beginning of the program. For the first instruction after reset only, all interrupts including the $\overline{\text{NMI}}$ interrupt are disabled.

13.13.3 $\overline{\text{NMI}}$ Interrupt

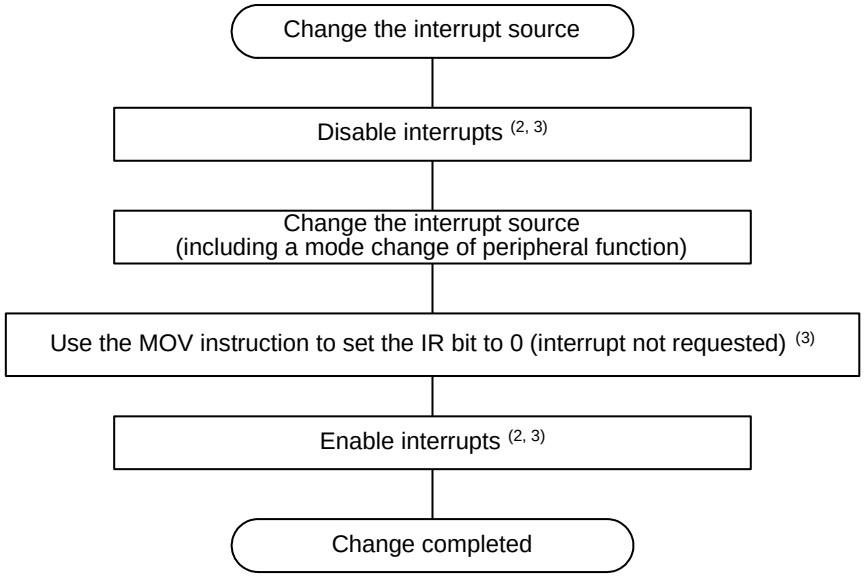
- When not using the $\overline{\text{NMI}}$ interrupt, set the PM24 bit in the PM2 register to 0 ($\overline{\text{NMI}}$ interrupt disabled).
- The $\overline{\text{NMI}}$ interrupt is disabled after reset. The $\overline{\text{NMI}}$ interrupt is enabled by setting the PM24 bit in the PM2 register to 1. Set the PM24 bit to 1 when a high-level signal is applied to the $\overline{\text{NMI}}$ pin. When the PM24 bit is set to 1 while a low-level signal is applied, an $\overline{\text{NMI}}$ interrupt is generated. Once the $\overline{\text{NMI}}$ interrupt is enabled, it cannot be disabled until the MCU is reset.
- Stop mode cannot be entered while the PM24 bit is 1 ($\overline{\text{NMI}}$ interrupt enabled) and input on the $\overline{\text{NMI}}$ pin is low. When input on the $\overline{\text{NMI}}$ pin is low, the CM10 bit in the CM1 register is fixed to 0.
- Do not enter wait mode while the PM24 bit is 1 ($\overline{\text{NMI}}$ interrupt enabled) and input on the $\overline{\text{NMI}}$ pin is low because the CPU clock remains active even though the CPU stops, and therefore, current consumption of the chip does not drop. In this case, normal condition is restored by the next interrupt generated.
- Set the low- and high-level durations of the input signal to the $\overline{\text{NMI}}$ pin to 2 CPU clock cycles + 300 ns or more.

13.13.4 Changing an Interrupt Source

When the interrupt source is changed, the IR bit in the interrupt control register may inadvertently become 1 (interrupt requested). To use an interrupt, change the interrupt source, and then set the IR bit to 0 (interrupt not requested).

In this section, the changing of an interrupt source refers to all elements (e.g. changing the mode of a peripheral function) used in changing the interrupt source, polarity, and timing assigned to each software interrupt number. When using an element to change the interrupt source, polarity, or timing, make the change before setting the IR bit to 0 (interrupt not requested). Refer to the descriptions of the individual peripheral functions for details of the peripheral function interrupts.

Figure 13.12 shows the Procedure for Changing the Interrupt Generate Factor.



IR bit: A bit in the interrupt control register

Notes:
1. Execute the above settings individually. Do not use one instruction to execute two or more settings simultaneously.
2. Use the I flag for the $\overline{\text{INTi}}$ interrupt (i = 0 to 7).
   For the interrupts from peripheral functions other than the $\overline{\text{INTi}}$ interrupt, disable the peripheral function that is the source of the interrupt so as not to generate an interrupt request before changing the interrupt source. In this case, if the maskable interrupts can all be disabled without causing a problem, use the I flag. Otherwise, use corresponding bits ILVL2 to ILVL0 in the interrupt control register whose interrupt source is to be changed.
3. Refer to 13.13.5 “Rewriting the Interrupt Control Register” below for details about the instructions to be used and the notes to be taken for instruction execution.

Figure 13.12 Procedure for Changing the Interrupt Generate Factor

13.13.5 Rewriting the Interrupt Control Register

To modify the interrupt control register, follow either of the procedures below:

- Modify in places where no requests for the interrupt control register may occur.
- If an interrupt request can be generated, disable that interrupt and then rewrite the interrupt control register.

When using the I flag to disable an interrupt, set the I flag as shown in the sample program code below. (Refer to 13.13.6 “Instruction to Rewrite the Interrupt Control Register” for rewriting the interrupt control registers using the sample program code.)

Examples 1 through 3 show how to prevent the I flag from becoming 1 (interrupt enabled) before the contents of the interrupt control register is rewritten, owing to the effects of the internal bus and the instruction queue buffer.

Example 1: Using the NOP instruction to pause the program until the interrupt control register is modified

```
INT_SWITCH1:
    FCLR    I               ; Disable interrupts.
    AND.B   #00H, 0055H     ; Set the TA0IC register to 00h.
    NOP                     ;
    NOP
    FSET    I               ; Enable interrupts.
```

Example 2: Using a dummy read to delay the FSET instruction

```
INT_SWITCH2:
    FCLR    I               ; Disable interrupts.
    AND.B   #00H, 0055H     ; Set the TA0IC register to 00h.
    MOV.W   MEM, R0         ; Dummy read.
    FSET    I               ; Enable interrupts.
```

Example 3: Using the POPC instruction to change the I flag

```
INT_SWITCH3:
    PUSHC   FLG
    FCLR    I               ; Disable interrupts.
    AND.B   #00H, 0055H     ; Set the TA0IC register to 00h.
    POPC    FLG             ; Enable interrupts.
```

13.13.6 Instruction to Rewrite the Interrupt Control Register

- Do not use the BTSTC and BTSTS instructions to rewrite the interrupt control registers.
- Use the AND, OR, BCLR, BSET, or MOV instruction to rewrite interrupt control registers.
  When an interrupt request is generated for the register being rewritten while executing an AND, OR, BCLR, BSET, or MOV instruction, the IR bit becomes 1 (interrupt requested) and remains 1.

13.13.7 $\overline{\text{INT}}$ Interrupt

- Either a low level of at least tw(INL) width or a high level of at least tw(INH) width is necessary for the signal input to pins $\overline{\text{INT1}}$ through $\overline{\text{INT5}}$ regardless of the CPU operation clock.
- If the POL bit in registers INT0IC to INT7IC, bits IFSR7 to IFSR0 in the IFSR register, or bits IFSR31 to IFSR30 in the IFSR3A register are changed, the IR bit may inadvertently become 1 (interrupt requested). Be sure to set the IR bit to 0 (interrupt not requested) after changing any of these register bits.

14. Watchdog Timer

14.1 Introduction

The watchdog timer contains a 15-bit counter, and the count source protection mode (enabled/disabled) can be set.
Table 14.1 lists Watchdog Timer Specifications.
Refer to 6.4.5 “Watchdog Timer Reset” for details of watchdog timer reset.
Figure 14.1 shows Watchdog Timer Block Diagram.

Table 14.1 Watchdog Timer Specifications

Item	Count Source Protection Mode Disabled	Count Source Protection Mode Enabled
Count source	CPU clock	fOCO-S
Count operation	Decrement	
Count start conditions	Either of the following can be selected (selected by the WDTON bit in the OFS1 address) • Count automatically starts after reset. • Count starts by writing to the WDTS register.	
Count stop condition	Stop mode, wait mode, bus hold	None
Watchdog timer counter refresh timing	• Reset (refer to 6. “Resets”) • Write 00h, and then FFh to the WDTR register. • Underflow	
Operation when the timer underflows	Watchdog timer interrupt or watchdog timer reset	Watchdog timer reset
Selectable functions	• Prescaler divide ratio Divide-by-16 or divide-by-128 (selected by the WDC7 bit in the WDC register) However, divide-by-2 is selected when the CM07 bit in the CM0 register is 1 (sub clock). • Count source protection mode Enabled or disabled (selected by the CSPROINI bit in the OFS1 address and the CSPRO bit in the CSPR register)	

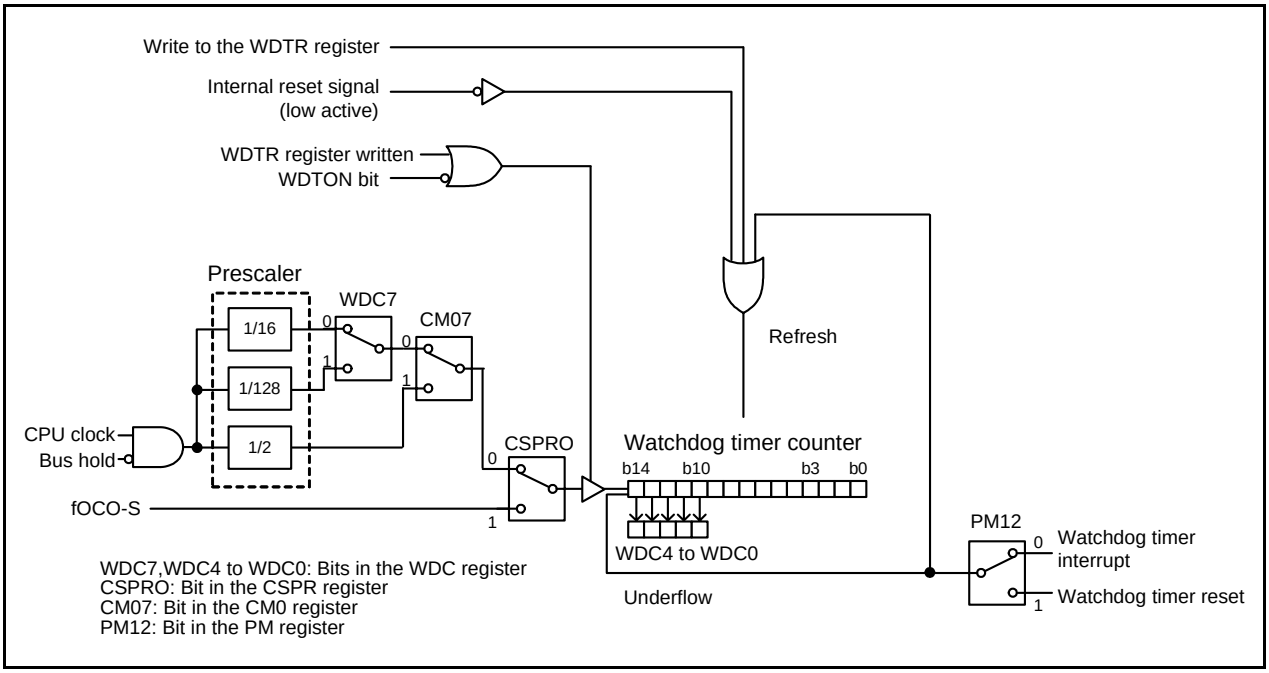


Figure 14.1 Watchdog Timer Block Diagram

14.2 Registers

Table 14.2 Registers

Address	Register	Symbol	Reset Value
002Ch	Watchdog Timer Detector Register	VW2C	1000 0X10b
037Ch	Count Source Protection Mode Register	CSPR	00h (1)
037Dh	Watchdog Timer Refresh Register	WDTR	XXh
037Eh	Watchdog Timer Start Register	WDTS	XXh
037Fh	Watchdog Timer Control Register	WDC	00XX XXXXb

Note:
1. When the CSPROINI bit in the OFS1 address is 0, the reset value becomes 1000 0000b.

14.2.1 Watchdog Timer Detector Register (VW2C)

Watchdog Timer Detector Register

b7	b6	b5	b4	b3	b2	b1	b0
1	0						0

Symbol	Address	After Reset
VW2C	002Ch	1000 0X10b

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved bit	Set to 0.	RW
VW2C3	WDT detection flag	0 : Not detected 1 : Watchdog timer underflow detected	RW
— (b7-b4)	Reserved bit	Set to 0.	RW

Set the PRC3 bit in the PRCR register to 1 (write enabled) before rewriting the VW2C register.
VW2C3 bit do not change at oscillator stop detect reset, watchdog timer reset, or software reset.

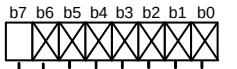
VW2C3 (WDT detection flag) (b3)

Use this bit in an interrupt routine to determine the source of the interrupts from the watchdog timer, the oscillator stop/restart detect.
Conditions to become 0:
- Writing 0 by a program

Condition to become 1:
- Watchdog timer underflow detected
  (This flag remains unchanged even if 1 is written by a program.)

14.2.2 Count Source Protection Mode Register (CSPR)

Count Source Protection Mode Register



Symbol	Address	After Reset
CSPR	037Ch	0000 0000b (When the CSPROINI bit in the OFS1 address is 1) 1000 0000b (When the CSPROINI bit in the OFS1 address is 0)

Bit Symbol	Bit Name	Function	RW
— (b6-b0)	No register bits. If necessary, set to 0. The read value is 0.		RW
CSPRO	Count source protection mode select bit	0: Count source protection mode disabled 1: Count source protection mode enabled	RW

CSPRO (Count source protection mode select bit) (b7)

Select the CSPRO bit before the watchdog timer starts counting. Once counting starts, do not change the CSPRO bit.

Condition to become 0:

- Reset when the CSPROINI bit in the OFS1 address is 1.
  (This flag remains unchanged even if 0 is written by a program.)

Condition to become 1:

- When the CSPROINI bit in the OFS1 address is 0
- Write 0, and then write 1.
  Make sure no interrupts or DMA transfers will occur between setting the bit to 0 and setting it to 1.

14.2.3 Watchdog Timer Refresh Register (WDTR)

Watchdog Timer Refresh Register

b7 b0	Symbol	Address	After Reset
	WDTR	037Dh	XXh

Function	RW
Setting 00h and then FFh refreshes the watchdog timer.	WO

After the watchdog timer interrupt occurs, refresh the watchdog timer by setting the WDTR register.

14.2.4 Watchdog Timer Start Register (WDTS)

Watchdog Timer Start Register

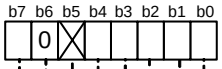
b7 b0	Symbol	Address	After Reset
	WDTS	037Eh	XXh

Function	RW
The watchdog timer starts counting after a write instruction to this register.	WO

The WDTS register is enabled when the WDTON bit in the OFS1 address is 1 (watchdog timer is in a stopped state after reset).

14.2.5 Watchdog Timer Control Register (WDC)

Watchdog Timer Control Register



Symbol: WDC
Address: 037Fh
After Reset: 00XX XXXXb

Bit Symbol	Bit Name	Function	RW
WDC0	Higher-order bits (b14 to b10) of the watchdog timer can be read.		RO
WDC1			RO
WDC2			RO
WDC3			RO
WDC4			RO
— (b5)	No register bit. If necessary, set to 0. Read as 0.		—
— (b6)	Reserved bit	Set to 0.	RW
WDC7	Prescaler select bit	0: Divide-by-16 1: Divide-by-128	RW

WDC4 to WDC0 (b4 to b0)

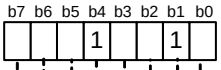
When reading the watchdog timer value while the CSPRO bit in the CSPR register is 1 (count source protection mode enabled), read bits WDC4 to WDC0 more than three times to determine the values.

14.3 Optional Function Select Area

In the optional function select area, the MCU state after reset and the function to prevent rewrite in parallel I/O mode are selected.
The optional function select area is not an SFR, and therefore cannot be rewritten by a program. Set an appropriate value when writing a program to flash memory. The entire optional function select area becomes FFh when the block including the optional function select area is erased.
In blank products, the OFS1 address value is FFh when shipped. After a value is written by the user, this register takes on the written value. In programmed products, the OFS1 address is the value set in the user program prior to shipping.

14.3.1 Optional Function Select Address 1 (OFS1)

Optional Function Select Address 1



Symbol: OFS1
Address: FFFFFh

Bit Symbol	Bit Name	Function
WDTON	Watchdog timer start select bit	0: Watchdog timer starts automatically after reset 1: Watchdog timer is stopped after reset
— (b1)	Reserved bit	Set to 1.
ROMCR	ROM code protect cancel bit	0: ROM code protection cancelled 1: ROMCP1 bit enabled
ROMCP1	ROM code protect bit	0: ROM code protection enabled 1: ROM code protection disabled
— (b4)	Reserved bit	Set to 1.
VDSEL1	Vdet0 select bit 1	0: Vdet0_2 1: Vdet0_0
LVDAS	Voltage detector 0 start bit	0: Voltage monitor 0 reset enabled after hardware reset 1: Voltage monitor 0 reset disabled after hardware reset
CSPROINI	After-reset count source protection mode select bit	0: Count source protection mode enabled after reset 1: Count source protection mode disabled after reset

WDTON (Watchdog timer start select bit) (b0)

CSPROINI (After-reset count source protection mode select bit) (b7)

Set the WDTON bit to 0 (watchdog timer starts automatically after reset) when setting the CSPROINI bit to 0 (count source protection mode enabled after reset).

14.4 Operations

14.4.1 Count Source Protection Mode Disabled

The CPU clock is used as the watchdog timer count source when count source protection mode is disabled.
Table 14.3 lists Watchdog Timer Specifications (Count Source Protection Mode Disabled).

Table 14.3 Watchdog Timer Specifications (Count Source Protection Mode Disabled)

Item	Specification
Count source	CPU clock
Count operation	Decrement
Cycles	When the CM07 bit in the CM0 register is 0 (main clock, PLL clock, fOCO-S): $\frac{\text{Prescaler divide value (n)} \times \text{watchdog timer count value (32768)}}{\text{CPU clock}}$ (1) n: 16 or 128 (selected by the WDC7 bit in the WDC register) ex.) When CPU clock frequency is 16 MHz and the prescaler division rate is 16, the watchdog timer cycle is approximately 32.8 ms. When the CM07 bit is 1 (sub clock): $\frac{\text{Prescaler divide value (2)} \times \text{watchdog timer count value (32768)}}{\text{CPU clock}}$ (1)
Watchdog timer counter refresh timing	• Reset (refer to 6. “Resets”) • Write 00h, and then FFh to the WDTR register. • Underflow
Count start conditions	Set the WDTON bit in the OFS1 address to select the watchdog timer operation after reset. • WDTON bit is 1 (watchdog timer is in stop state after reset) The watchdog timer counter and prescaler stop after reset and count starts by writing to the WDTS register. • WDTON bit is 0 (watchdog timer starts automatically after reset) The watchdog timer counter and prescaler start counting automatically after reset.
Count stop conditions	• Stop mode • Wait mode • Bus hold (Count resumes from the hold value after exiting.)
Operation when timer underflows	• PM12 bit in the PM1 register is 0 Watchdog timer interrupt • PM12 bit in the PM1 register is 1 Watchdog timer reset (Refer to 6.4.5 “Watchdog Timer Reset”.)

Note:
1. When writing 00h and then FFh to the WDTR register, the watchdog timer is refreshed, but the prescaler is not initialized. Thus, some errors in the watchdog timer period may be caused by the prescaler. The prescaler is initialized after reset.

14.4.2 Count Source Protection Mode Enabled

The fOCO-S is used as the watchdog timer count source when the count source protection mode is enabled.
Table 14.4 lists the Watchdog Timer Specifications (Count Source Protection Mode Enabled).

Table 14.4 Watchdog Timer Specifications (Count Source Protection Mode Enabled)

Item	Specification
Count source	fOCO-S (The 125 kHz on-chip oscillator clock automatically starts oscillating.)
Count operation	Decrement
Cycle	$\frac{\text{Watchdog timer count value (4096)}}{\text{fOCO-S}}$ (The watchdog timer cycle is approximately 32.8 ms.)
Watchdog timer counter refresh timing	• Reset (refer to 6. “Resets”) • Write 00h, and then FFh to the WDTR register. • Underflow
Count start conditions	Set the WDTON bit in the OFS1 address to select the watchdog timer operation after reset. • WDTON bit is 1 (watchdog timer is stopped after reset) The watchdog timer counter and prescaler stop after reset and count starts by writing to the WDTS register. • WDTON bit is 0 (watchdog timer starts automatically after reset) The watchdog timer counter and prescaler start counting automatically after reset.
Count stop condition	None (The count does not stop in wait mode or by bus hold once started. The MCU does not enter stop mode.)
Operation when timer underflows	Watchdog timer reset (Refer to 6.4.5 “Watchdog Timer Reset”).

When the CSPRO bit in the CSPR register is 1 (count source protection mode enabled), the watchdog timer counter underflows every 4096 cycles because three low-order bits are not used.

Also when the CSPRO bit is set to 1 (count source protection mode enabled), the following bits change:

- The CM14 bit in the CM1 register becomes 0 (125 kHz on-chip oscillator on). It remains unchanged even if 1 is written, and the 125 kHz on-chip oscillator does not stop.
- The PM12 bit in the PM1 register becomes 1 (watchdog timer reset when watchdog timer counter underflows).
- The CM10 bit in the CM1 register remains unchanged even if 1 is written, and the MCU does not enter stop mode.

14.5 Interrupts

Watchdog timer interrupts are non-maskable interrupts.

The watchdog timer interrupt and oscillator stop/restart detect interrupt share an vector. When using multiple functions, read the detect flag in an interrupt process program to determine the source of the interrupt.

The VW2C3 bit in the VW2C register is the detect flag for the watchdog timer. After the interrupt factor is determined, set the VW2C3 bit to 0 (not detected) by a program.

14.6 Notes on Watchdog Timer

After a watchdog timer interrupt is generated, use the WDTR register to refresh the watchdog timer counter.

15. DMAC

15.1 Introduction

The direct memory access controller (DMAC) allows data to be transferred without CPU intervention. Four DMAC channels are included. Each time a DMA request occurs, the DMAC transfers one (8- or 16-bit) unit of data from the source address to the destination address. The DMAC uses the same data bus used by the CPU. Because the DMAC has higher priority for bus control than the CPU, and because it makes use of a cycle steal method, it can transfer one word (16 bits) or one byte (8 bits) of data within a very short time after a DMA request is generated. Figure 15.1 shows the DMAC Block Diagram. Table 15.1 lists DMAC Specifications, and Figure 15.1 shows DMAC Block Diagram.

Table 15.1 DMAC Specifications

Item		Specification
Number of channels		4 (cycle steal method)
Transfer memory spaces		• From a given address in the 1-MB space to a fixed address • From a fixed address to a given address in the 1-MB space • From a fixed address to a fixed address
Maximum number of bytes transferred		128 KB (with 16-bit transfers) or 64 KB (with 8-bit transfers)
DMA request factors (1)		43 factors Falling edge of $\overline{INT0}$ to $\overline{INT7}$ (8) Both edges of $\overline{INT0}$ to $\overline{INT7}$ (8) Timer A0 to timer A4 interrupt requests (5) Timer B0 to timer B5 interrupt requests (6) UART0 to 2, UART5 to 7 transmission interrupt requests (6) UART0 to 2, UART5 to 7 reception/ACK interrupt requests (6) SI/O3, SI/O4 interrupt requests (2) A/D conversion interrupt requests (1) Software triggers (1)
Channel priority		DMA0 > DMA1 > DMA2 > DMA3 (DMA0 takes precedence)
Transfers		8 bits or 16 bits
Transfer address direction		Forward or fixed (The source and destination addresses cannot both be in the forward direction.)
Transfer mode	Single transfer	Transfer is completed when the DMAi transfer counter underflows.
	Repeat transfer	When the DMAi transfer counter underflows, it is reloaded with the value of the DMAi transfer counter reload register and DMA transfer continues.
DMA interrupt request generation timing		When the DMAi transfer counter underflows
DMA transfer start		Data transfer is initiated each time a DMA request is generated when the DMAE bit in the DMiCON register is 1 (enabled).
DMA transfer stop	Single transfer	• When the DMAE bit is set to 0 (disabled) • After the DMAi transfer counter underflows
	Repeat transfer	When the DMAE bit is set to 0 (disabled)
Reload timing for forward address pointer and DMAi transfer counter		When a data transfer is started after setting the DMAE bit to 1 (enabled), the forward address pointer is reloaded with the value of the SARi or DARi register, whichever is specified to be in the forward direction, and the DMAi transfer counter is reloaded with the value of the DMAi transfer counter reload register.
DMA transfer cycles		Minimum 3 cycles between SFR and internal RAM

i = 0 to 3

Note:

1. The selectable sources of DMA requests differ for each channel.

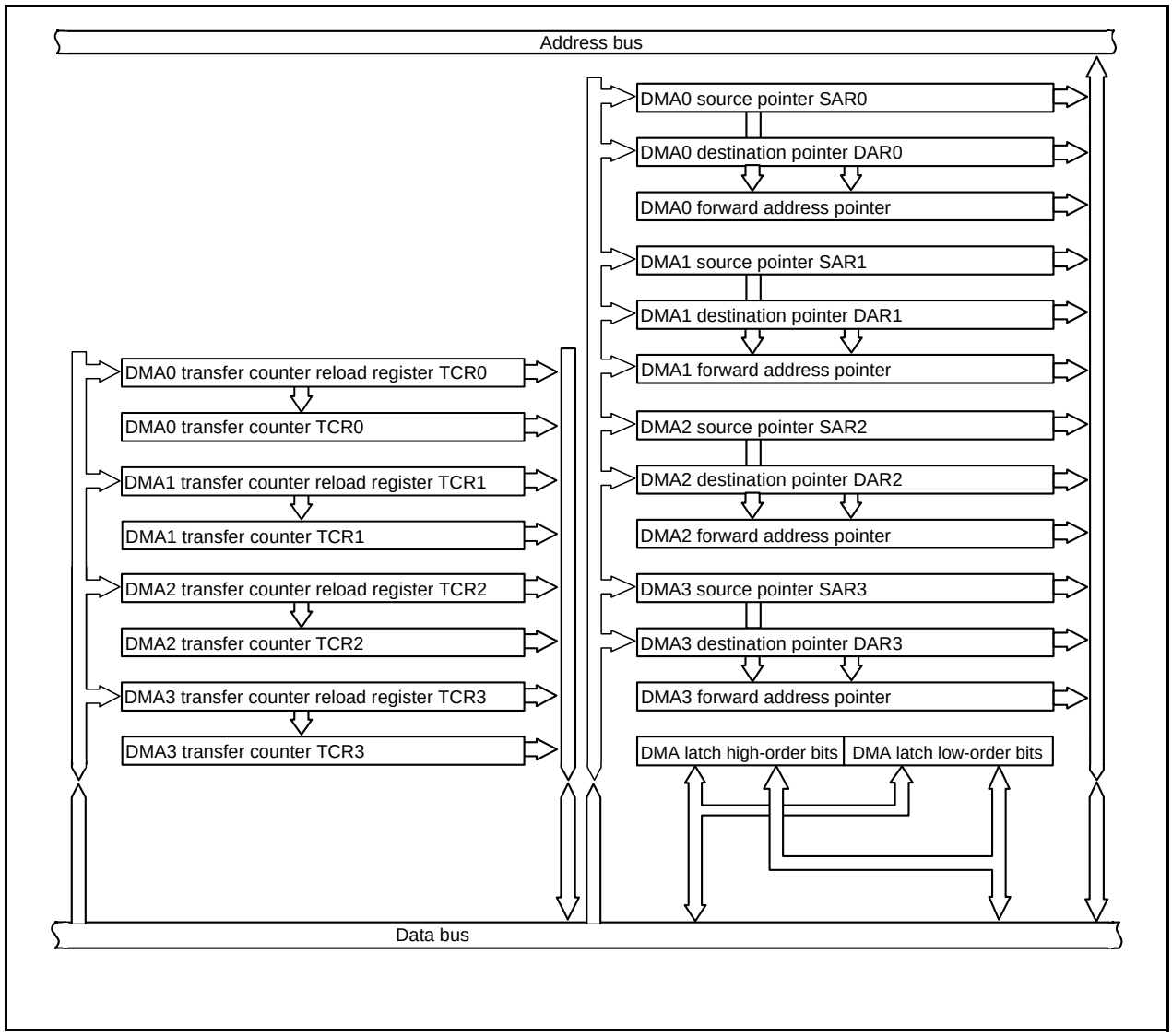


Figure 15.1 DMAC Block Diagram

15.2 Registers

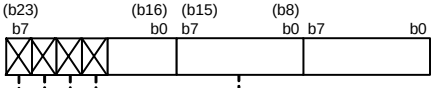
Table 15.2 lists Registers. Do not access these registers during DMAC operation.

Table 15.2 Registers

Address	Register	Symbol	Reset Value
0180h	DMA0 Source Pointer	SAR0	XXh
0181h			XXh
0182h			0Xh
0184h	DMA0 Destination Pointer	DAR0	XXh
0185h			XXh
0186h			0Xh
0188h	DMA0 Transfer Counter	TCR0	XXh
0189h			XXh
018Ch	DMA0 Control Register	DM0CON	0000 0X00b
0190h	DMA1 Source Pointer	SAR1	XXh
0191h			XXh
0192h			0Xh
0194h	DMA1 Destination Pointer	DAR1	XXh
0195h			XXh
0196h			0Xh
0198h	DMA1 Transfer Counter	TCR1	XXh
0199h			XXh
019Ch	DMA1 Control Register	DM1CON	0000 0X00b
01A0h	DMA2 Source Pointer	SAR2	XXh
01A1h			XXh
01A2h			0Xh
01A4h	DMA2 Destination Pointer	DAR2	XXh
01A5h			XXh
01A6h			0Xh
01A8h	DMA2 Transfer Counter	TCR2	XXh
01A9h			XXh
01ACh	DMA2 Control Register	DM2CON	0000 0X00b
01B0h	DMA3 Source Pointer	SAR3	XXh
01B1h			XXh
01B2h			0Xh
01B4h	DMA3 Destination Pointer	DAR3	XXh
01B5h			XXh
01B6h			0Xh
01B8h	DMA3 Transfer Counter	TCR3	XXh
01B9h			XXh
01BCh	DMA3 Control Register	DM3CON	0000 0X00b
0390h	DMA2 Source Select Register	DM2SL	00h
0392h	DMA3 Source Select Register	DM3SL	00h
0398h	DMA0 Source Select Register	DM0SL	00h
039Ah	DMA1 Source Select Register	DM1SL	00h

15.2.1 DMAi Source Pointer (SARi) (i = 0 to 3)

DMAi Source Pointer (i = 0 to 3)



Symbol	Address	After Reset
SAR0	0182h to 0180h	0X XXXXh
SAR1	0192h to 0190h	0X XXXXh
SAR2	01A2h to 01A0h	0X XXXXh
SAR3	01B2h to 01B0h	0X XXXXh

Function	Setting Range	RW
Set the source address of transfer.	00000h to FFFFFh	RW
No register bits. If necessary, set to 0. Read as 0.		—

If the DSD bit in the DMiCON register is 0 (fixed), write to SARi register when the DMAE bit in the DMiCON register is 0 (DMA disabled).
If the DSD bit is 1 (forward direction), this register can be written to at any time.
If the DSD bit is 1 and the DMAE bit is 1 (DMA enabled), the DMAi forward address pointer can be read from this register. Otherwise, the value written to it can be read.
The forward address pointer is incremented when a DMA request is accepted.

15.2.2 DMAi Destination Pointer (DARi) (i = 0 to 3)

DMAi Destination Pointer (i = 0 to 3)

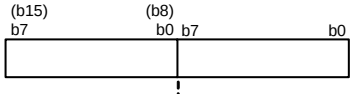
Symbol	Address	After Reset
DAR0	0186h to 0184h	0X XXXXh
DAR1	0196h to 0194h	0X XXXXh
DAR2	01A6h to 01A4h	0X XXXXh
DAR3	01B6h to 01B4h	0X XXXXh

Function	Setting Range	RW
Set the destination address of transfer.	00000h to FFFFFh	RW
No register bits. If necessary, set to 0. Read as 0.		—

If the DAD bit in the DMiCON register is 0 (fixed), write to DARi register when the DMAE bit in the DMiCON register is 0 (DMA disabled).
If the DAD bit is 1 (forward direction), this register can be written to at any time.
If the DAD bit is 1 and the DMAE bit is 1 (DMA enabled), the DMAi forward address pointer can be read from this register. Otherwise, the value written to it can be read.
The forward address pointer is incremented on accepting a DMA request.

15.2.3 DMAi Transfer Counter (TCRi) (i = 0 to 3)

DMAi Transfer Counter (i = 0 to 3)



Symbol	Address	After Reset
TCR0	0189h to 0188h	XXh
TCR1	0199h to 0198h	XXh
TCR2	01A9h to 01A8h	XXh
TCR3	01B9h to 01B8h	XXh

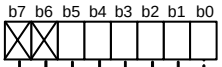
Function	Setting Range	RW
Set the transfer count minus 1.	0000h to FFFFh	RW

The written value in the TCRi register is stored in the DMAi transfer counter reload register.
The value of the DMAi transfer counter reload register is transferred to the DMAi transfer counter in either of the following cases:

- When the DMAE bit in the DMiCON register is set to 1 (DMA enabled) (single transfer mode, repeat transfer mode)
- When the DMAi transfer counter underflows (repeat transfer mode)

15.2.4 DMAi Control Register (DMiCON) (i = 0 to 3)

DMAi Control Register (i = 0 to 3)



Symbol	Address	After Reset
DM0CON	018Ch	0000 0X00b
DM1CON	019Ch	0000 0X00b
DM2CON	01ACh	0000 0X00b
DM3CON	01BCh	0000 0X00b

Bit Symbol	Bit Name	Function	RW
DMBIT	Transfer unit bit select bit	0: 16 bits 1: 8 bits	RW
DMASL	Repeat transfer mode select bit	0: Single transfer 1: Repeat transfer	RW
DMAS	DMA request bit	0: DMA not requested 1: DMA requested	RW
DMAE	DMA enable bit	0: Disabled 1: Enabled	RW
DSD	Source address direction select bit	0: Fixed 1: Forward	RW
DAD	Destination address direction select bit	0: Fixed 1: Forward	RW
— (b7-b6)	No register bits. If necessary, set to 0. Read as 0.		—

DMAS (DMA request bit) (b2)

Conditions to become 0:

- Set the bit to 0.
- Starting data transfer

Condition to become 1:

- Set the bit to 1.

DMAE (DMA enable bit) (b3)

Conditions to become 0:

- Set the bit to 0.
- The DMA transfer counter underflows (single transfer mode)

Condition to become 1:

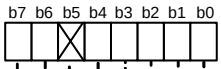
- Set the bit to 1.

DSD (Source address direction select bit) (b4)
DAD (Destination address direction select bit) (b5)

Set at least one of the DAD bit and DSD bit to 0 (address direction fixed).

15.2.5 DMAi Source Select Register (DMiSL) (i = 0 to 3)

DMAi Source Select Register (i = 0 to 3)



Symbol	Address	After Reset
DM0SL	0398h	00h
DM1SL	039Ah	00h
DM2SL	0390h	00h
DM3SL	0392h	00h

Bit Symbol	Bit Name	Function	RW
DSEL0	DMA request source select bit	See the bit explanation below.	RW
DSEL1			RW
DSEL2			RW
DSEL3			RW
DSEL4			RW
— (b5)	No register bit. If necessary, set to 0. Read as 0.		—
DMS	DMA request source expansion select bit	0: Basic request source 1: Extended request source	RW
DSR	Software DMA request bit	A DMA request is generated by setting this bit to 1 when the DMS bit is 0 (basic source) and bits DSEL4 to DSEL0 are 00001b (software trigger). Read as 0.	RW

DSEL4 to DSEL0 (DMA request source select bit) (b4 to b0)

The sources of DMAi requests can be selected by a combination of the DMS bit and bits DSEL4 to DSEL0 in the manner shown in Tables 15.3 to 15.6. Tables 15.3 to 15.6 list the Sources of DMA Requests.

Table 15.3 Source of DMA Request (DMA0)

DSEL4 to DSEL0	DMS = 0 (Basic Source of Request)	DMS = 1 (Extended Source of Request)
0 0 0 0 0 b	Falling edge of $\overline{\text{INT0}}$ pin	–
0 0 0 0 1 b	Software trigger	–
0 0 0 1 0 b	Timer A0	–
0 0 0 1 1 b	Timer A1	–
0 0 1 0 0 b	Timer A2	–
0 0 1 0 1 b	Timer A3	–
0 0 1 1 0 b	Timer A4	Both edges of $\overline{\text{INT0}}$ pin
0 0 1 1 1 b	Timer B0	Timer B3
0 1 0 0 0 b	Timer B1	Timer B4
0 1 0 0 1 b	Timer B2	Timer B5
0 1 0 1 0 b	UART0 transmission	–
0 1 0 1 1 b	UART0 reception	–
0 1 1 0 0 b	UART2 transmission	–
0 1 1 0 1 b	UART2 reception	–
0 1 1 1 0 b	A/D conversion	–
0 1 1 1 1 b	UART1 transmission	–
1 0 0 0 0 b	UART1 reception	Falling edge of $\overline{\text{INT4}}$ pin
1 0 0 0 1 b	UART5 transmission	Both edges of $\overline{\text{INT4}}$ pin
1 0 0 1 0 b	UART5 reception	–
1 0 0 1 1 b	UART6 transmission	–
1 0 1 0 0 b	UART6 reception	–
1 0 1 0 1 b	UART7 transmission	–
1 0 1 1 0 b	UART7 reception	–
1 0 1 1 1 b	–	–
1 1 X X X b	–	–

X indicates 0 or 1. – indicates no setting.

Table 15.4 Source of DMA Request (DMA1)

DSEL4 to DSEL0	DMS = 0 (Basic Source of Request)	DMS = 1 (Extended Source of Request)
0 0 0 0 0 b	Falling edge of $\overline{\text{INT1}}$ pin	–
0 0 0 0 1 b	Software trigger	–
0 0 0 1 0 b	Timer A0	–
0 0 0 1 1 b	Timer A1	–
0 0 1 0 0 b	Timer A2	–
0 0 1 0 1 b	Timer A3	SI/O3
0 0 1 1 0 b	Timer A4	SI/O4
0 0 1 1 1 b	Timer B0	Both edges of $\overline{\text{INT1}}$ pin
0 1 0 0 0 b	Timer B1	–
0 1 0 0 1 b	Timer B2	–
0 1 0 1 0 b	UART0 transmission	–
0 1 0 1 1 b	UART0 reception/ACK0	–
0 1 1 0 0 b	UART2 transmission	–
0 1 1 0 1 b	UART2 reception/ACK2	–
0 1 1 1 0 b	A/D conversion	–
0 1 1 1 1 b	UART1 reception/ACK1	–
1 0 0 0 0 b	UART1 transmission	Falling edge of $\overline{\text{INT5}}$ pin
1 0 0 0 1 b	UART5 transmission	Both edges of $\overline{\text{INT5}}$ pin
1 0 0 1 0 b	UART5 reception/ACK5	–
1 0 0 1 1 b	UART6 transmission	–
1 0 1 0 0 b	UART6 reception/ACK6	–
1 0 1 0 1 b	UART7 transmission	–
1 0 1 1 0 b	UART7 reception/ACK7	–
1 0 1 1 1 b	–	–
1 1 X X X b	–	–

X indicates 0 or 1. – indicates no setting.

Table 15.5 Source of DMA Request (DMA2)

DSEL4 to DSEL0	DMS = 0 (Basic Source of Request)	DMS = 1 (Extended Source of Request)
0 0 0 0 0 b	Falling edge of $\overline{\text{INT2}}$ pin	–
0 0 0 0 1 b	Software trigger	–
0 0 0 1 0 b	Timer A0	–
0 0 0 1 1 b	Timer A1	–
0 0 1 0 0 b	Timer A2	–
0 0 1 0 1 b	Timer A3	–
0 0 1 1 0 b	Timer A4	Both edges of $\overline{\text{INT2}}$ pin
0 0 1 1 1 b	Timer B0	Timer B3
0 1 0 0 0 b	Timer B1	Timer B4
0 1 0 0 1 b	Timer B2	Timer B5
0 1 0 1 0 b	UART0 transmission	–
0 1 0 1 1 b	UART0 reception	–
0 1 1 0 0 b	UART2 transmission	–
0 1 1 0 1 b	UART2 reception	–
0 1 1 1 0 b	A/D conversion	–
0 1 1 1 1 b	UART1 transmission	–
1 0 0 0 0 b	UART1 reception	Falling edge of $\overline{\text{INT6}}$ pin
1 0 0 0 1 b	UART5 transmission	Both edges of $\overline{\text{INT6}}$ pin
1 0 0 1 0 b	UART5 reception	–
1 0 0 1 1 b	UART6 transmission	–
1 0 1 0 0 b	UART6 reception	–
1 0 1 0 1 b	UART7 transmission	–
1 0 1 1 0 b	UART7 reception	–
1 0 1 1 1 b	–	–
1 1 X X X b	–	–

X indicates 0 or 1. – indicates no setting.

Table 15.6 Source of DMA Request (DMA3)

DSEL4 to DSEL0	DMS = 0 (Basic Source of Request)	DMS = 1 (Extended Source of Request)
0 0 0 0 0 b	Falling edge of $\overline{\text{INT3}}$ pin	–
0 0 0 0 1 b	Software trigger	–
0 0 0 1 0 b	Timer A0	–
0 0 0 1 1 b	Timer A1	–
0 0 1 0 0 b	Timer A2	–
0 0 1 0 1 b	Timer A3	SI/O3
0 0 1 1 0 b	Timer A4	SI/O4
0 0 1 1 1 b	Timer B0	Both edges of $\overline{\text{INT3}}$ pin
0 1 0 0 0 b	Timer B1	–
0 1 0 0 1 b	Timer B2	–
0 1 0 1 0 b	UART0 transmission	–
0 1 0 1 1 b	UART0 reception/ACK0	–
0 1 1 0 0 b	UART2 transmission	–
0 1 1 0 1 b	UART2 reception/ACK2	–
0 1 1 1 0 b	A/D conversion	–
0 1 1 1 1 b	UART1 reception/ACK1	–
1 0 0 0 0 b	UART1 transmission	Falling edge of $\overline{\text{INT7}}$ pin
1 0 0 0 1 b	UART5 transmission	Both edges of $\overline{\text{INT7}}$ pin
1 0 0 1 0 b	UART5 reception/ACK5	–
1 0 0 1 1 b	UART6 transmission	–
1 0 1 0 0 b	UART6 reception/ACK6	–
1 0 1 0 1 b	UART7 transmission	–
1 0 1 1 0 b	UART7 reception/ACK7	–
1 0 1 1 1 b	–	–
1 1 X X X b	–	–

X indicates 0 or 1. – indicates no setting.

15.3 Operations

15.3.1 DMA Enabled

When data transfer starts after setting the DMAE bit in the DMiCON register (i = 0 to 3) to 1 (enabled), the DMAC operates as listed below. If 1 is written to the DMAE bit when it is already set to 1, the DMAC also performs the following operation.

- The forward address pointer is reloaded with the SARi register value when the DSD bit in the DMiCON register is 1 (forward), or the DARi register value when the DAD bit in the DMiCON register is 1 (forward).
- The DMAi transfer counter is reloaded with the DMAi transfer counter reload register value.

15.3.2 DMA Request

The DMAC can generate a DMA request as triggered by the request source that is selected with the DMS bit and bits DSEL4 to DSEL0 in the DMiSL register (i = 0 to 3) on each channel. Table 15.7 lists the Timing at Which the DMAS Bit Changes State.

Whenever a DMA request is generated, the DMAS bit is set to 1 (DMA requested) regardless of whether or not the DMAE bit is set. If the DMAE bit is set to 1 (enabled) when this occurs, the DMAS bit is set to 0 (DMA not requested) immediately before a data transfer starts. This bit cannot be set to 1 by a program (writing a 1 has no effect).

If the DMAE bit is 1, the DMAS bit in almost all cases is 0 when read in a program, because a data transfer starts immediately after a DMA request is generated. Read the DMAE bit to determine whether the DMAC is enabled. When a DMA request transfer cycle is shorter than a DMA transfer cycle, the number of transfer requests and the number of transfers do not match.

When the peripheral function is selected as a DMA source, relations with interrupts are as follows:

- DMA transfers are not affected by the I flag or interrupt control registers. DMA requests are always accepted even when interrupt requests are not accepted.
- The IR bit in the interrupt control register retains its value when a DMA transfer is accepted.

Table 15.7 Timing at Which the DMAS Bit Changes State

DMA Source	DMAS Bit in the DMiCON Register	
	Timing at Which the Bit is Set to 1	Timing at Which the Bit is Set to 0
Software trigger	When the DSR bit in the DMiSL register is set to 1	• Immediately before a data transfer starts • When set by writing a 0 by a program
External factor	When an input edge of pins $\overline{INT0}$ to $\overline{INT7}$ matches with what is selected by bits DSEL4 to DSEL0 and DMS in the DMiSL register.	
Peripheral function	When an interrupt request of the peripheral function selected by bits DSEL4 to DSEL0 and DMS bit in the DMiSL register is generated. (If the IR bit in an interrupt control register is 0, the timing is when 0 is changed to 1.)	

i = 0 to 3

15.3.3 Transfer Cycles

A transfer cycle is composed of a bus cycle to read data from a source address (source read), and a bus cycle to write data to a destination address (destination write). The number of read and write bus cycles depends on the source and destination addresses.

Figure 15.2 shows Transfer Cycles for Source Read Operations. For convenience, the destination write cycle is shown as one bus cycle and the source read cycles for the different conditions are shown. In reality, the destination write cycle is subject to the same conditions as the source read cycle, with the transfer cycle changing accordingly. When calculating transfer cycles, take into consideration each condition for the source read and the destination write cycle. For example, when data is transferred in 16-bit units, and the source address of transfer is an odd address ((2) in Figure 15.2), two source read bus cycles and two destination write bus cycles are required.

15.3.3.1 Effect of Source and Destination Addresses

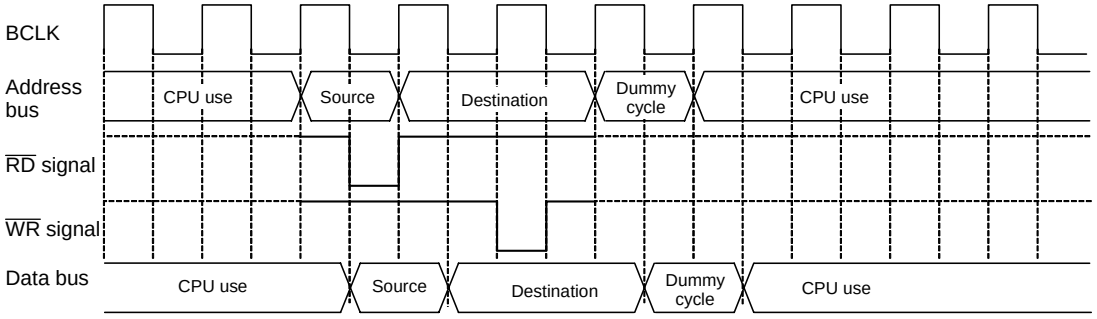
When a 16-bit unit of data is transferred with a 16-bit data bus and the source address starts with an odd address, the source-read cycle is incremented by one bus cycle, compared to a source address starting with an even address.

When a 16-bit unit of data is transferred with a 16-bit data bus and the destination address starts with an odd address, the destination-write cycle is incremented by one bus cycle, compared to a destination address starting with an even address.

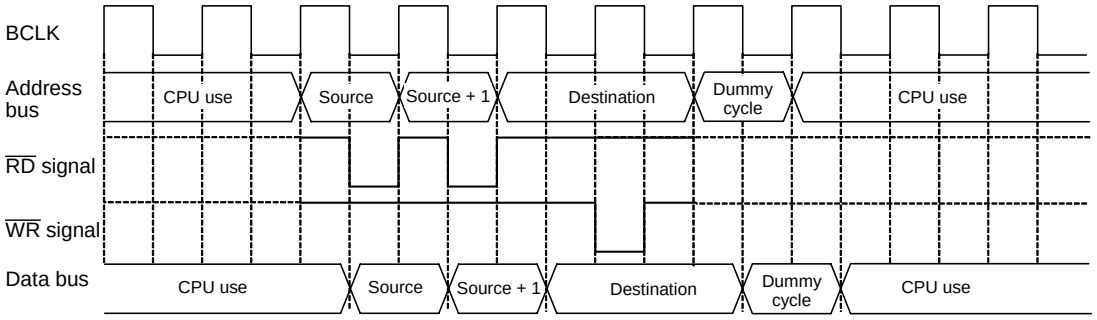
15.3.3.2 Effect of Software Wait

For memory or SFR accesses in which one or more software wait states are inserted, the number of bus cycles required increases by an amount equal to the number of software wait states.

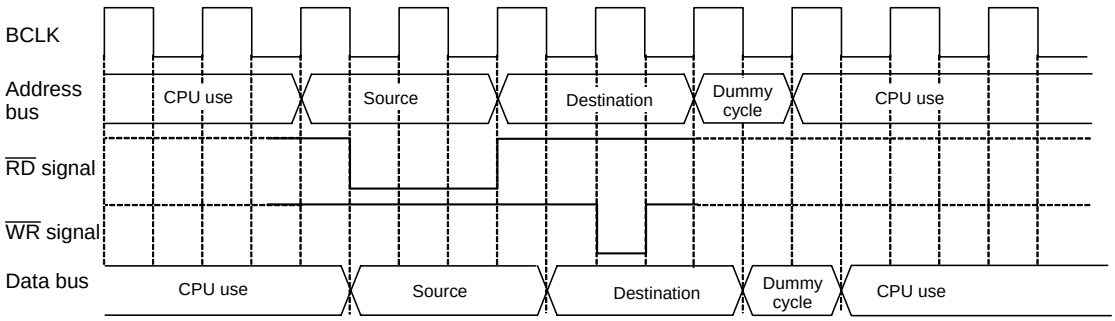
(1) Transfers are performed in 8-bit or 16-bit units, and source of transfer is an even address.



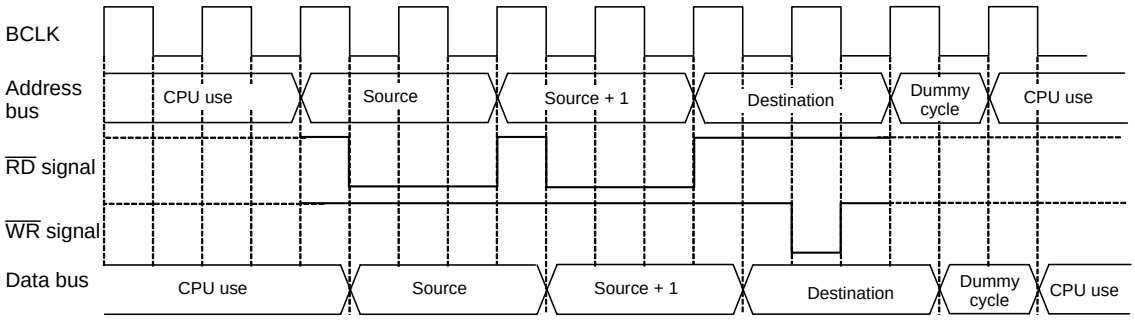
(2) Transfers are performed in 16-bit units, and the source address of transfer is an odd address.



(3) Source read cycle under condition (1) with one wait state inserted



(4) Source read cycle under condition (2) with one wait state inserted



Note:
1. The same timing changes occur with the respective conditions at the destination as at the source.

Figure 15.2 Transfer Cycles for Source Read Operations

15.3.4 DMAC Transfer Cycles

The number of DMAC transfer cycles can be calculated as shown below.

Number of transfer cycles per transfer unit = Number of read cycles × j + Number of write cycles × k

Table 15.8 DMAC Transfer Cycles

Transfer Unit	Bus Width	Access Address	Single-Chip Mode	
			No. of Read Cycles	No. of Write Cycles
8-bit transfers (DMBIT = 1)	16-bit (BYTE = low)	Even	1	1
		Odd	1	1
	8-bit (BYTE = high)	Even	N/A	N/A
		Odd	N/A	N/A
16-bit transfers (DMBIT = 0)	16-bit (BYTE = low)	Even	1	1
		Odd	2	2
	8-bit (BYTE = high)	Even	N/A	N/A
		Odd	N/A	N/A

DMBIT: Bit in the DMiCON register (i = 0 to 3)

Table 15.9 Coefficients j and k (1/2)

	Internal Area		
	Internal ROM, RAM		SFR
	No wait states	Wait states	1 wait state
j	1	2	2
k	1	2	2

15.3.5 Single Transfer Mode

In single transfer mode, the transfer stops when the DMAi transfer counter underflows. Figure 15.3 shows Operation Example in Single Transfer Mode.

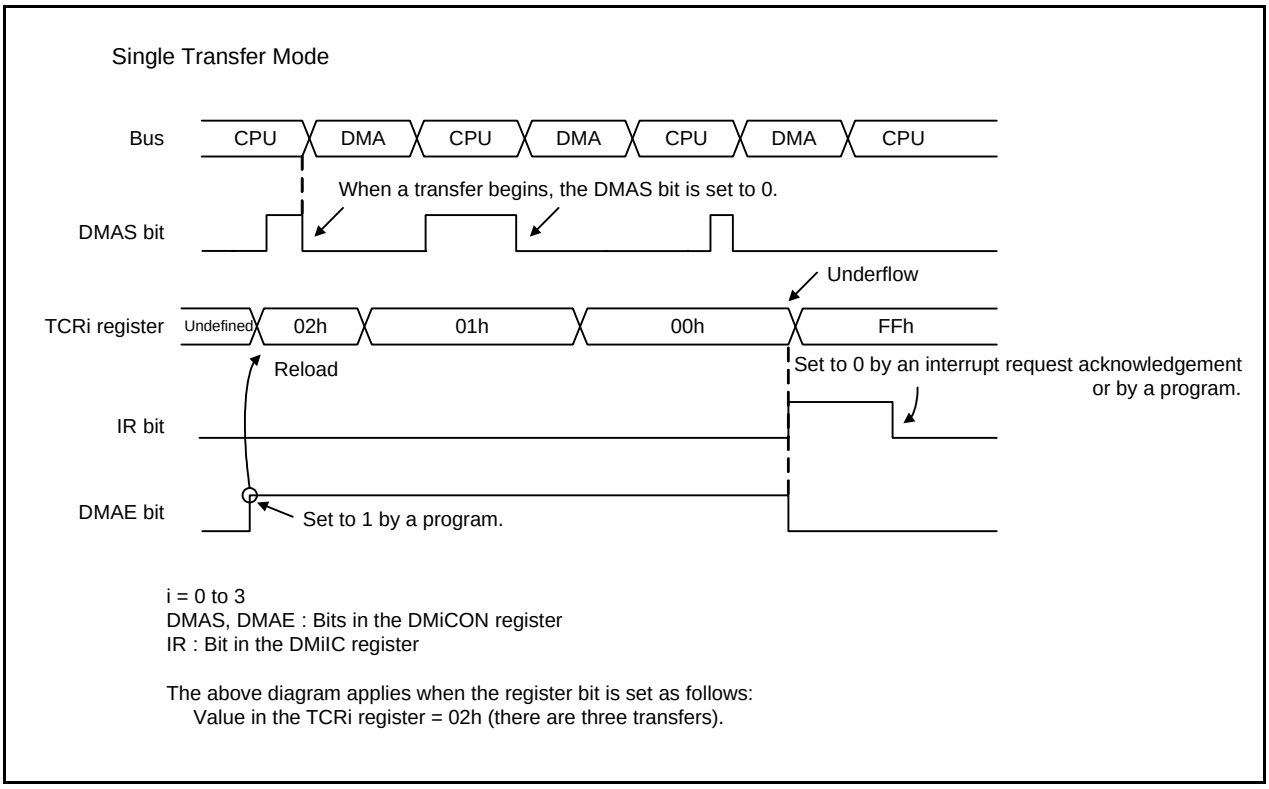


Figure 15.3 Operation Example in Single Transfer Mode

15.3.6 Repeat Transfer Mode

In repeat transfer mode, when the DMAi transfer counter underflows, it is reloaded with the value of the DMAi transfer counter reload register and DMA transfer continues. Figure 15.4 shows Operation Example in Repeat Transfer Mode.

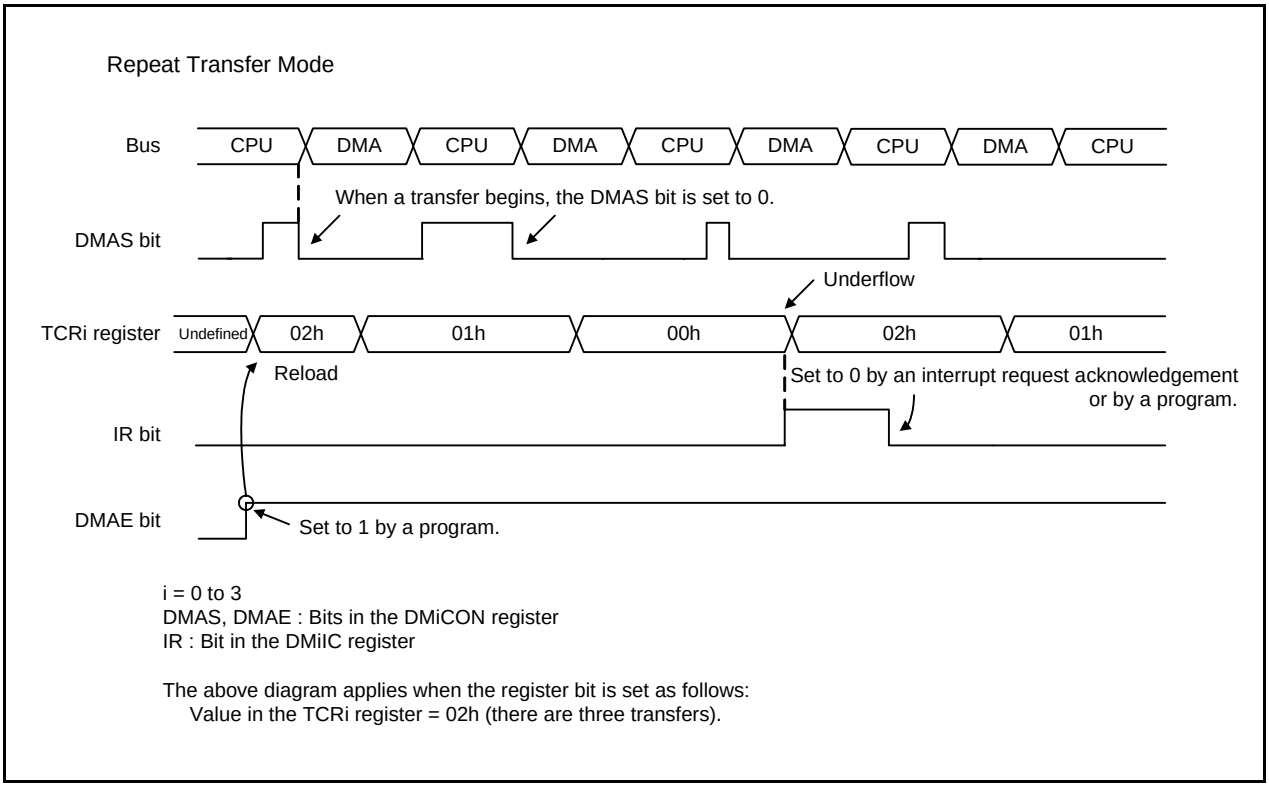


Figure 15.4 Operation Example in Repeat Transfer Mode

15.3.7 Channel Priority and DMA Transfer Timing

If multiple channels among DMA0 to DMA3 are enabled and DMA transfer request signals are detected active in the same sampling period (one period from a falling edge to the next falling edge of BCLK), the DMAS bit on each channel is set to 1 (DMA requested) at the same time. In this case, the DMA requests are arbitrated according to the following channel priority: DMA0 > DMA1 > DMA2 > DMA3. The DMAC operation when DMA0 and DMA1 requests are detected active in the same sampling period is described below. Figure 15.5 shows an example of DMA Transfer by External Sources.

In Figure 15.5, DMA0, which has a high channel priority, is received first to start a transfer when DMA0 and DMA1 requests are generated simultaneously. After one DMA0 transfer is completed, the bus access privilege is returned to the CPU. When the CPU has completed one bus access, a DMA1 transfer starts. After one DMA1 transfer is completed, the bus access privilege is again returned to the CPU.

In addition, DMA requests cannot be incremented since each channel has one DMAS bit. Therefore, when DMA requests, such as DMA1 in Figure 15.5, occur more than once, the DMAS bit is set to 0 after receiving the bus access privilege. The bus access privilege is returned to the CPU when one transfer is completed.

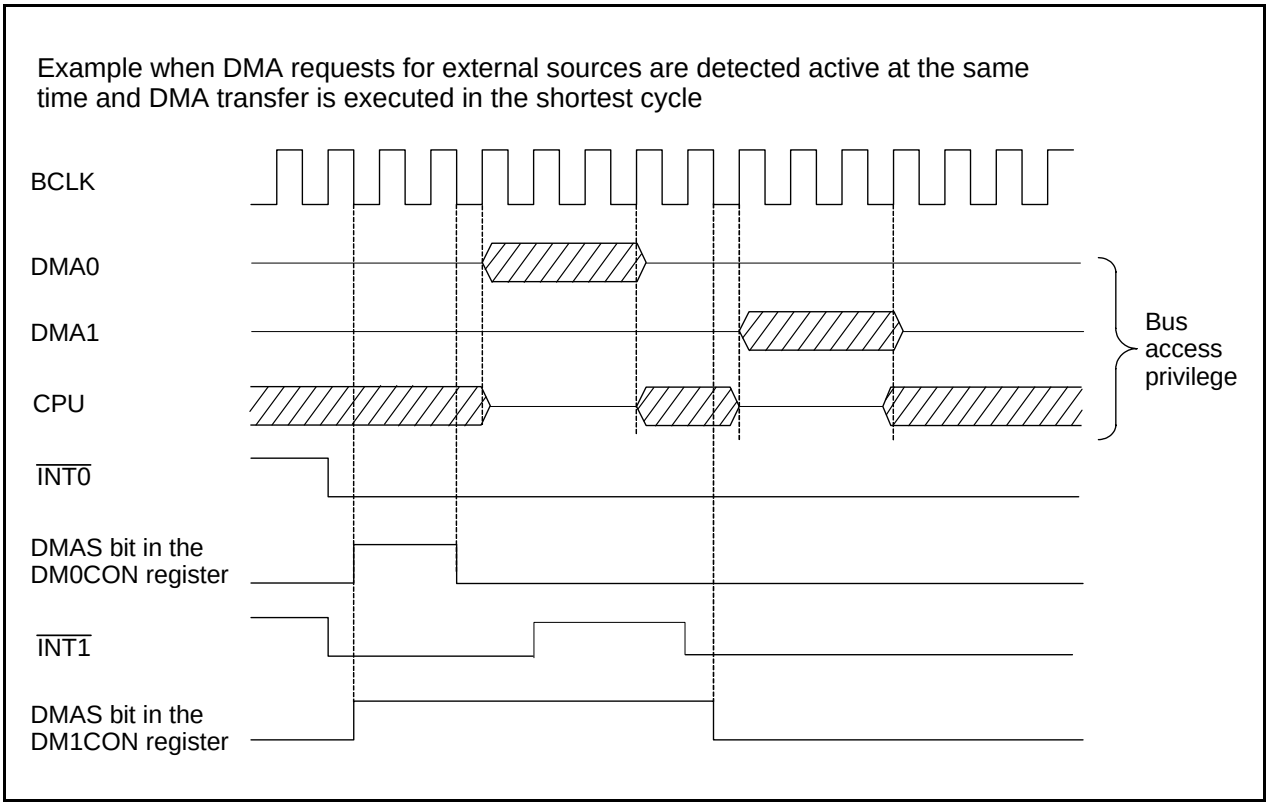


Figure 15.5 DMA Transfer by External Sources

15.4 Interrupts

Refer to operation examples for interrupt request generation timing.
For the details of interrupt control, refer to 13.7 “Interrupt Control”.

Table 15.10 DMAC Interrupt Related Registers

Address	Register	Symbol	Reset Value
004Bh	DMA0 Interrupt Control Register	DM0IC	XXXX X000b
004Ch	DMA1 Interrupt Control Register	DM1IC	XXXX X000b
0069h	DMA2 Interrupt Control Register	DM2IC	XXXX X000b
006Ah	DMA3 Interrupt Control Register	DM3IC	XXXX X000b

When the DMS bit or bits DSEL4 to DSEL0 in the DMiSL register are changed, the DMAS bit in the DMiCON sometimes becomes 1 (DMA requested). Therefore, set the DMAS bit to 0 (DMA not requested) after the DMS bit or bits DSEL4 to DSEL0 in the DMiSL register are changed. Refer to 13.13 “Notes on Interrupts”.

15.5 Notes on DMAC

15.5.1 Write to the DMAE Bit in the DMiCON Register (i = 0 to 3)

When both of following conditions are met, follow steps (1) and (2) below.

- Write a 1 (DMAi is in active state) to the DMAE bit when it is 1.
- A DMA request may be generated at the same time the DMAE bit is being written.

Steps

(1) Set bits DMAE and DMAS in the DMiCON register to 1 simultaneously (1).

(2) Make sure that the DMAi circuit is in an initialized state (2) in a program.
If the DMAi is not in an initialized state, repeat these two steps.

Notes:

1. The DMAS bit remains unchanged even if set to 1. However, it becomes 0 when set to 0 (DMA not requested). To prevent the DMAS bit from being modified to 0, 1 should be written to the DMAS bit when 1 is written to the DMAE bit. This setting allows the DMAS bit to retain its value previous to being rewritten.
Similarly, when writing to the DMAE bit with a read-modify-write instruction, set the DMAS bit to 1 to retain the DMA request that was generated while executing the instruction.
2. Read the TCRi register to verify whether the DMAi is in an initialized state.
If the read value is equal to a value that was written to the TCRi register before DMA transfer starts, the DMAi is in an initialized state. (When a DMA request is generated after writing to the DMAE bit, the read value is a value written to the TCRi register minus 1.) If the read value is a value in the middle of a transfer, the DMAi is not in an initialized state.

15.5.2 Changing the DMA Request Source

When the DMS bit or bits DSEL4 to DSEL0 in the DMiSL register are changed, the DMAS bit in the DMiCON sometimes becomes 1 (DMA requested). Set the DMAS bit to 0 (DMA not requested) after the DMS bit or bits DSEL4 to DSEL0 in the DMiSL register are changed.

16. Timer A

Note

Pins TA0IN and TA0OUT are not provided for timer A0 because it is internally connected to the PLC modem.

16.1 Introduction

Timers A consists of timers A0 to A4. Each timer operates independently of the others. Table 16.1 lists Timer A Specifications, Table 16.2 lists Differences in Timer A Mode, Figure 16.1 shows Timer A and B Count Sources, Figure 16.2 shows Timer A Configuration, Figure 16.3 shows Timer A Block Diagram, and Table 16.3 lists I/O Ports.

Table 16.1 Timer A Specifications

Item	Specification
Configuration	16-bit timer × 5
Operating modes	• Timer mode The timer counts an internal count source. • Event counter mode The timer counts pulses from an external device or overflows and underflows of other timers. • One-shot timer mode The timer outputs a pulse only once before it reaches the count 0000h. • Pulse width modulation mode (PWM mode) The timer outputs pulses of given width and cycle successively. • Programmable output mode The timer outputs a given pulse width of a high-/low- level signal (timers A1, A2, and A4).
Interrupt sources	Overflow/underflow × 5

Table 16.2 Differences in Timer A Mode

Item	Timer				
	A0	A1	A2	A3	A4
Event counter mode (two-phase pulse signal processing)	No	No	Yes	Yes	Yes
Programmable output mode	No	Yes	Yes	No	Yes

Note:
1. Timer A0 pins are connected to the internal PLC modem. No external pin is provided.
Control them via provided DLL software and do not control them directly via user software.

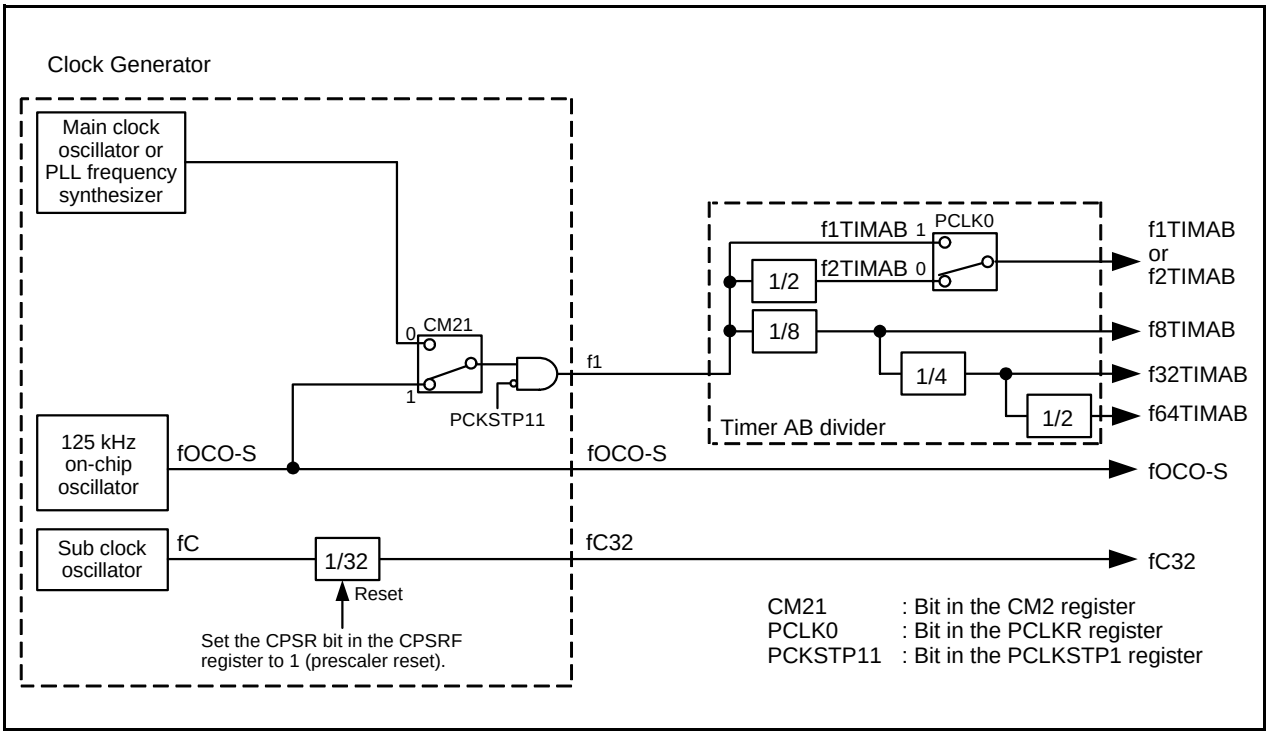


Figure 16.1 Timer A and B Count Sources

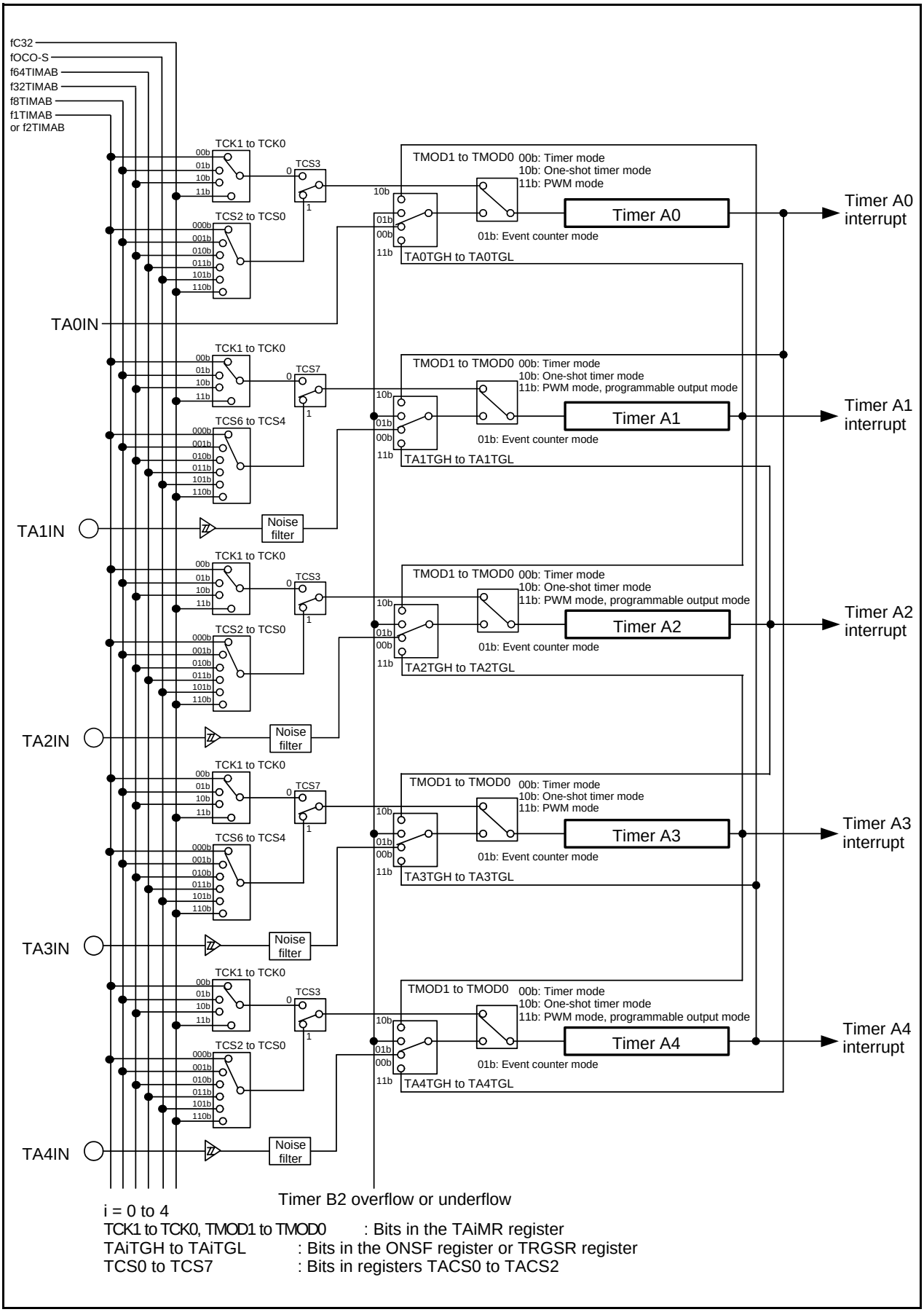
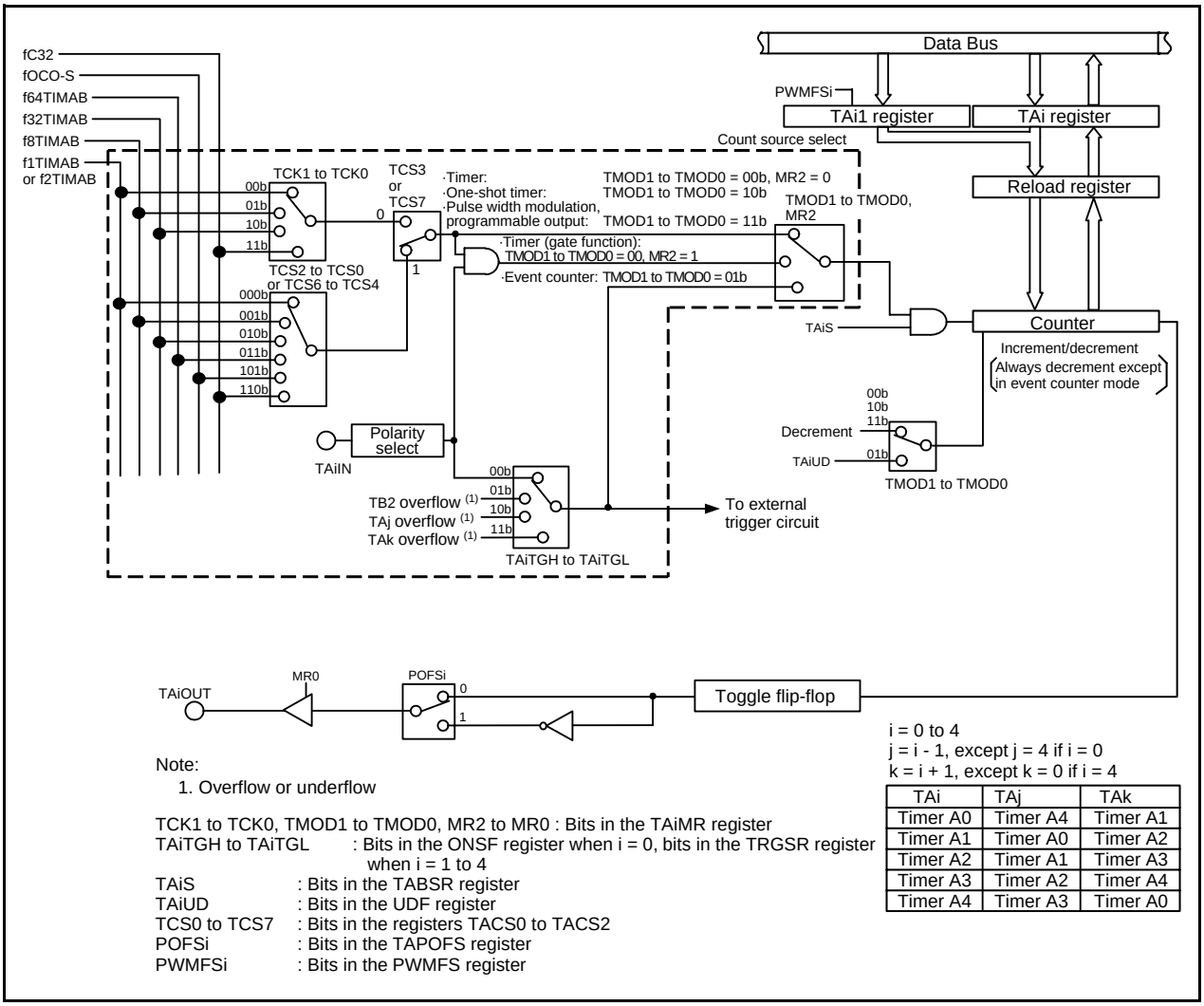


Figure 16.2 Timer A Configuration



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

Figure 16.3 Timer A Block Diagram

Table 16.3 I/O Ports

Pin Name	I/O	Function
TAiIN	Input (1)	Gate input (timer mode) Count source input (event counter mode) Two-phase signal input (event counter mode (two-phase pulse signal processing)) Trigger input (one-shot timer mode, PWM mode, programmable output mode)
TAiOUT	Output	Pulse output (timer mode, event counter mode, one-shot timer mode, PWM mode, and programmable output mode)
	Input (1)	Two-phase pulse input (event counter mode (two-phase pulse signal processing))
ZP	Input (1)	Z-phase (counter initialization) input (event counter mode (two-phase pulse signal processing))

i = 0 to 4; however, i = 2, 3, 4 for two-phase pulse input, and i = 1, 2, 4 in programmable output mode

Notes:

1. When using pins TAiIN, TAiOUT, and ZP for input, set the port direction bits corresponding to the pins to 0 (input mode).

16.2 Registers

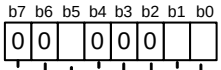
Table 16.4 lists registers associated with timer A.
Refer to “registers and the setting” in each mode for registers and bit settings.

Table 16.4 Registers

Address	Register	Symbol	Reset Value
0012h	Peripheral Clock Select Register	PCLKR	0000 0011b
0015h	Clock Prescaler Reset Flag	CPSRF	0XXX XXXXb
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
01D0h	Timer A Count Source Select Register 0	TACS0	00h
01D1h	Timer A Count Source Select Register 1	TACS1	00h
01D2h	Timer A Count Source Select Register 2	TACS2	X0h
01D4h	16-Bit Pulse Width Modulation Mode Function Select Register	PWMFS	0XX0 X00Xb
01D5h	Timer A Waveform Output Function Select Register	TAPOFS	XXX0 0000b
01D8h	Timer A Output Waveform Change Enable Register	TAOW	XXX0 X00Xb
0302h	Timer A1-1 Register	TA11	XXh
0303h			XXh
0304h	Timer A2-1 Register	TA21	XXh
0305h			XXh
0306h	Timer A4-1 Register	TA41	XXh
0307h			XXh
0320h	Count Start Flag	TABSR	00h
0322h	One-Shot Start Flag	ONSF	00h
0323h	Trigger Select Register	TRGSR	00h
0324h	Up/Down Flag	UDF	00h
0326h	Timer A0 Register	TA0	XXh
0327h			XXh
0328h	Timer A1 Register	TA1	XXh
0329h			XXh
032Ah	Timer A2 Register	TA2	XXh
032Bh			XXh
032Ch	Timer A3 Register	TA3	XXh
032Dh			XXh
032Eh	Timer A4 Register	TA4	XXh
032Fh			XXh
0336h	Timer A0 Mode Register	TA0MR	00h
0337h	Timer A1 Mode Register	TA1MR	00h
0338h	Timer A2 Mode Register	TA2MR	00h
0339h	Timer A3 Mode Register	TA3MR	00h
033Ah	Timer A4 Mode Register	TA4MR	00h

16.2.1 Peripheral Clock Select Register (PCLKR)

Peripheral Clock Select Register



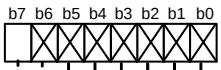
Symbol	Address	After Reset
PCLKR	0012h	0000 0011b

Bit Symbol	Bit Name	Function	RW
PCLK0	Timers A and B clock select bit (clock source for timers A and B, and multi-master I^2C-bus interface)	0: f2TIMAB/f2IIC 1: f1TIMAB/f1IIC	RW
PCLK1	SI/O clock select bit (clock source for UART0 to UART2, UART5 to UART7, SI/O3, and SI/O4)	0: f2SIO 1: f1SIO	RW
— (b4-b2)	Reserved bits	Set to 0.	RW
PCLK5	Clock output function expansion bit	0: Selected by bits CM01 to CM00 in the CM0 register 1: Output f1	RW
— (b7-b6)	Reserved bits	Set to 0.	RW

Set the PCLKR register after the PRC0 bit in the PRCR register is set to 1 (write enabled).

16.2.2 Clock Prescaler Reset Flag (CPSRF)

Clock Prescaler Reset Flag

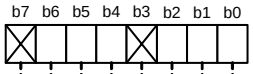


Symbol	Address	After Reset
CPSRF	0015h	0XXX XXXXb

Bit Symbol	Bit Name	Function	RW
— (b6-b0)	No register bits. If necessary, set to 0. Read as undefined value.		—
CPSR	Clock prescaler reset flag	Setting this bit to 1 initializes the prescaler for the timekeeping clock. (Read as 0.)	RW

16.2.3 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1



Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I^2C peripheral clock stop bit (multi-master I^2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

Set the PRC0 bit in the PRCR register to 1 (write enabled) before the PCLKSTP1 register is rewritten.

PCKSTP11 (Timer peripheral clock stop bit) (b1)

Set the PCKSTP11 bit to 0 (f1 provide enabled) when using the f1 as the clock source.

16.2.4 Timer A Count Source Select Register i (TACSi) (i = 0 to 2)

Timer A Count Source Select Register 0, Timer A Count Source Select Register 1

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TACS0 to TACS1	01D0h to 01D1h	00h

Bit Symbol	Bit Name	Function	RW
TCS0	TAi count source select bit	b2 b1 b0 0 0 0: f1TIMAB or f2TIMAB 0 0 1: f8TIMAB 0 1 0: f32TIMAB 0 1 1: f64TIMAB 1 0 0: Do not set 1 0 1: fOCO-S 1 1 0: fC32 1 1 1: Do not set	RW
TCS1			RW
TCS2			RW
TCS3	TAi count source option specified bit	0: TCK0, TCK1 enabled, TCS0 to TCS2 disabled 1: TCK0, TCK1 disabled, TCS0 to TCS2 enabled	RW
TCS4	TAj count source select bit	b6 b5 b4 0 0 0: f1TIMAB or f2TIMAB 0 0 1: f8TIMAB 0 1 0: f32TIMAB 0 1 1: f64TIMAB 1 0 0: Do not set 1 0 1: fOCO-S 1 1 0: fC32 1 1 1: Do not set	RW
TCS5			RW
TCS6			RW
TCS7	TAj count source option specified bit	0: TCK0, TCK1 enabled, TCS4 to TCS6 disabled 1: TCK0, TCK1 disabled, TCS4 to TCS6 enabled	RW

TACS0 register: i = 0, j = 1 TACS1 register: i = 2, j = 3

Timer A Count Source Select Register 2

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TACS2	01D2h	X0h

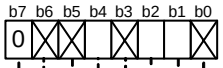
Bit Symbol	Bit Name	Function	RW
TCS0	TA4 count source select bit	b2 b1 b0 0 0 0: f1TIMAB or f2TIMAB 0 0 1: f8TIMAB 0 1 0: f32TIMAB 0 1 1: f64TIMAB 1 0 0: Do not set 1 0 1: fOCO-S 1 1 0: fC32 1 1 1: Do not set	RW
TCS1			RW
TCS2			RW
TCS3	TA4 count source option specified bit	0: TCK0 to TCK1 enabled, TCS0 to TCS2 disabled 1: TCK0 to TCK1 disabled, TCS0 to TCS2 enabled	RW
— (b7-b4)	No register bits. If necessary, set to 0. Read as undefined value.		—

TCS2 to TCS0 (TAi count source select bit) (b2 to b0) (i = 0, 2, 4)
TCS6 to TCS4 (TAj count source select bit) (b6 to b4) (i = 1, 3)

Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.

16.2.5 16-Bit Pulse Width Modulation Mode Function Select Register (PWMFS)

16-Bit Pulse Width Modulation Mode Function Select Register



Symbol	Address	After Reset
PWMFS	01D4h	0XX0 X00Xb

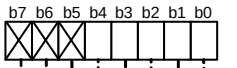
Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit. If necessary, set to 0. Read as undefined value.		—
PWMFS1	Timer A1 programmable output mode select bit	0: PWM mode 16-bit PWM 1: Programmable output mode	RW
PWMFS2	Timer A2 programmable output mode select bit	0: PWM mode 16-bit PWM 1: Programmable output mode	RW
— (b3)	No register bit. If necessary, set to 0. Read as undefined value.		—
PWMFS4	Timer A4 programmable output mode select bit	0: PWM mode 16-bit PWM 1: Programmable output mode	RW
— (b6-b5)	No register bits. If necessary, set to 0. Read as undefined value.		—
— (b7)	Reserved bit	Set to 0.	RW

PWMFS1 (Timer A1 programmable output mode select bit) (b1)
PWMFS2 (Timer A2 programmable output mode select bit) (b2)
PWMFS4 (Timer A4 programmable output mode select bit) (b4)

The bits are enabled when bits TMOD1 to TMOD0 in the TAiMR register are 11b (PWM mode or programmable output mode), and the MR3 bit in the TAiMR register is 0 (16-bit PWM mode).

16.2.6 Timer A Waveform Output Function Select Register (TAPOFS)

Timer A Waveform Output Function Select Register

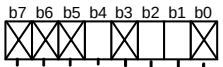


Symbol	Address	After Reset
TAPOFS	01D5h	XXX0 0000b

Bit Symbol	Bit Name	Function	RW
POFS0	TA0OUT output polar control bit	0: Output waveform high-level active 1: Output waveform high-level active (output reversed)	RW
POFS1	TA1OUT output polar control bit		RW
POFS2	TA2OUT output polar control bit		RW
POFS3	TA3OUT output polar control bit		RW
POFS4	TA4OUT output polar control bit		RW
— (b7-b5)	No register bits. If necessary, set to 0. Read as undefined value.		—

16.2.7 Timer A Output Waveform Change Enable Register (TAOW)

Timer A Output Waveform Change Enable Register



Symbol	Address	After Reset
TAOW	01D8h	XXX0 X00Xb

Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit. If necessary, set to 0. Read as undefined value.		—
TA1OW	Timer A1 output waveform change enable bit	0: Change disabled 1: Change enabled	RW
TA2OW	Timer A2 output waveform change enable bit	0: Change disabled 1: Change enabled	RW
— (b3)	No register bit. If necessary, set to 0. Read as undefined value.		—
TA4OW	Timer A4 output waveform change enable bit	0: Change disabled 1: Change enabled	RW
— (b7-b5)	No register bit. If necessary, set to 0. Read as undefined value.		—

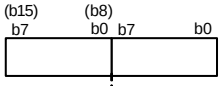
The TAOW register is enabled in programmable output mode.
To change cycles or width of the output waveform, follow the instructions below.

(1)Set the TAiOW bit to 0 (output waveform change disabled). (i = 1, 2, 4)
(2)Write to the TAi register and/or the TAi1 register.
(3)Set the TAiOW bit to 1 (output waveform change enabled).

The updated value is reloaded when the TAiOW bit is 1 (output waveform change enabled) at one cycle before the rising edge of the TAiOUT output (the falling edge when the POFSi bit is 1). The value before the update is reloaded when the TAiOW bit is 0 (output waveform change disabled).

16.2.8 Timer Ai Register (TAi) (i = 0 to 4)

Timer Ai Register (i = 0 to 4)



Symbol	Address	After Reset
TA0	0327h to 0326h	XXh
TA1	0329h to 0328h	XXh
TA2	032Bh to 032Ah	XXh
TA3	032Dh to 032Ch	XXh
TA4	032Fh to 032Eh	XXh

Mode	Function	Setting Range	RW
Timer mode	When n = set value, count cycle: $\frac{(n + 1)}{fj}$	0000h to FFFFh	RW
Event counter mode	When n = set value, FFFFh - n + 1 count (at increment) n + 1 count (at decrement)	0000h to FFFFh	RW
One-shot timer mode	When n = set value, pulse width: $\frac{n}{fj}$	0000h to FFFFh	WO
Pulse width modulation mode (16-bit PWM mode)	When n = set value, PWM period: $\frac{(2^{16} - 1)}{fj}$ PWM pulse width: $\frac{n}{fj}$	0000h to FFFEh	WO
Pulse width modulation mode (8-bit PWM mode)	When n = high-order address set value, m = low-order address set value, PWM period: $\frac{(2^8 - 1) \times (m + 1)}{fj}$ PWM pulse width: $\frac{(m + 1)n}{fj}$	00h to FEh (High-order address) 00h to FFh (Low-order address)	WO
Programmable output mode	When n = set value of TAi1 register, m = set value of TAi register, high-level duration: $\frac{m}{fj}$ low-level duration: $\frac{n}{fj}$	0000h to FFFFh	WO

fj : Count source frequency

Access the register in 16-bit units. Use the MOV instruction to write to the TAi register.

Event counter mode

The timer counts pulses from an external device, or the overflows/underflows of other timers.

One-shot timer mode

If the TAi register is set to 0000h, the counter does not work and timer Ai interrupt requests are not generated. Furthermore, if pulse output is selected, no pulses are output from the TAiOUT pin.

Pulse width modulation mode (16-bit PWM mode)

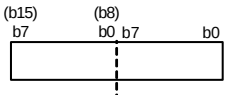
If the TAi register is set to 0000h, the counter does not work, the output level on the TAiOUT pin remains low, and timer Ai interrupt requests are not generated.

Pulse width modulation mode (8-bit PWM mode)

This mode operates as 8-bit prescaler (eight low-order bits) and 8-bit pulse width modulator (eight high-order bits). When the eight high-order bits of the TAi register are set to 00h, the counter does not work, the output level on the TAiOUT pin remains low, and timer Ai interrupt requests are not generated.

16.2.9 Timer Ai-1 Register (TAi1) (i = 1, 2, 4)

Timer Ai-1 Register (i = 1, 2, 4)



Symbol	Address	After Reset
TA11	0303h to 0302h	XXh
TA21	0305h to 0304h	XXh
TA41	0307h to 0306h	XXh

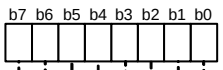
Mode	Function	Setting Range	RW
Programmable output mode	When n = set value of TAi1 register, m = set value of TAi register, high-level duration: $\frac{m}{fj}$ low-level duration: $\frac{n}{fj}$	0001h to FFFFh	WO

fj: Count source frequency

Access the register in 16-bit units. Use the MOV instruction to write to the TAi1 register.

16.2.10 Count Start Flag (TABSR)

Count Start Flag

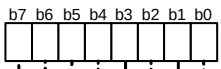


Symbol	Address	After Reset
TABSR	0320h	00h

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 count start flag	0: Count stopped 1: Count started	RW
TA1S	Timer A1 count start flag		RW
TA2S	Timer A2 count start flag		RW
TA3S	Timer A3 count start flag		RW
TA4S	Timer A4 count start flag		RW
TB0S	Timer B0 count start flag		RW
TB1S	Timer B1 count start flag		RW
TB2S	Timer B2 count start flag		RW

16.2.11 One-Shot Start Flag (ONSF)

One-Shot Start Flag



Symbol	Address	After Reset
ONSF	0322h	00h

Bit Symbol	Bit Name	Function	RW
TA0OS	Timer A0 one-shot start flag	The timer starts counting by setting this bit to 1. Read as 0.	RW
TA1OS	Timer A1 one-shot start flag		RW
TA2OS	Timer A2 one-shot start flag		RW
TA3OS	Timer A3 one-shot start flag		RW
TA4OS	Timer A4 one-shot start flag		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: Input on TA0IN pin selected 0 1: Timer B2 selected 1 0: Timer A4 selected 1 1: Timer A1 selected	RW
TA0TGH			RW

TAiOS (Timer Ai one-shot start flag) (i = 0 to 4) (b4 to b0)

This bit is enabled in one-shot timer mode. When the MR2 bit in the TAi register is 0 (TAiOS bit enabled), the timer Ai count starts by setting the TAiOS bit to 1 after setting the TAiS bit in the TABSR register to 1 (start counting).

TAZIE (Z-phase input enable bit) (b5)

This bit is used in event counter mode (two-phase pulse signal processing) of timer A3. Refer to 16.3.4.3 “Counter Initialization by Two-Phase Pulse Signal Processing” for details.

TA0TGH to TA0TGL (Timer A0 event/trigger select bit) (b7 to b6)

This bit is used to select an event or a trigger in the following modes:

- An event in event counter mode (not using two-phase pulse signal processing)
- A trigger in one-shot timer mode or PWM mode

  The above applies when the MR2 bit in the TA0MR register is 1 (trigger selected by bits TA0TGH to TA0TGL).

The active edge of input signals can be selected by the MR1 bit in the TA0MR register when bits TA0TGH to TA0TGL are 00b.

When bits TA0TGH to TA0TGL are set to 01b, 10b, or 11b, an event or a trigger occurs when an interrupt request of the selected timer is generated. (An event or trigger occurs while an interrupt is disabled because bits TA0TGH to TA0TGL are not influenced by the I flag, IPL, or the interrupt control registers.)

16.2.12 Trigger Select Register (TRGSR)

Trigger Select Register

b7 b6 b5 b4 b3 b2 b1 b0	Symbol	Address	After Reset
	TRGSR	0323h	00h

Bit Symbol	Bit Name	Function	RW
TA1TGL	Timer A1 event/trigger select bit	b1 b0 0 0: Input on TA1IN selected 0 1: TB2 selected 1 0: TA0 selected 1 1: TA2 selected	RW
TA1TGH			RW
TA2TGL	Timer A2 event/trigger select bit	b3 b2 0 0: Input on TA2IN selected 0 1: TB2 selected 1 0: TA1 selected 1 1: TA3 selected	RW
TA2TGH			RW
TA3TGL	Timer A3 event/trigger select bit	b5 b4 0 0: Input on TA3IN selected 0 1: TB2 selected 1 0: TA2 selected 1 1: TA4 selected	RW
TA3TGH			RW
TA4TGL	Timer A4 event/trigger select bit	b7 b6 0 0: Input on TA4IN selected 0 1: TB2 selected 1 0: TA3 selected 1 1: TA0 selected	RW
TA4TGH			RW

TA1TGH to TA1TGL (Timer A1 event/trigger select bit) (b1 to b0)
TA2TGH to TA2TGL (Timer A2 event/trigger select bit) (b3 to b2)
TA3TGH to TA3TGL (Timer A3 event/trigger select bit) (b5 to b4)
TA4TGH to TA4TGL (Timer A4 event/trigger select bit) (b7 to b6)

These bits are used to select an event or a trigger of the following modes:

- Event in event counter mode (not using two-phase pulse signal processing)
- Trigger in one-shot timer mode, PWM mode, or programmable output mode
  The above applies when the MR2 bit in the TAiMR register is 1 (trigger selected by bits TAiTGH to TAiTGL).

The active edge of input signals can be selected by the MR1 bit in the TAiMR register when bits TAiTGH to TAiTGL are 00b.

When bits TAiTGH to TAiTGL are set to 01b, 10b, or 11b, an event or a trigger occurs when an interrupt request of the selected timer is generated. (An event or a trigger occurs while an interrupt is disabled because bits TAiTGH to TAiTGL are not influenced by the I flag, IPL, or the interrupt control registers.)

16.2.13 Up/Down Flag (UDF)

Up/Down Flag

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: UDF
Address: 0324h
After Reset: 00h

Bit Symbol	Bit Name	Function	RW
TA0UD	Timer A0 up/down flag	0: Decrement 1: Increment	RW
TA1UD	Timer A1 up/down flag		RW
TA2UD	Timer A2 up/down flag		RW
TA3UD	Timer A3 up/down flag		RW
TA4UD	Timer A4 up/down flag		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	RW
TA3P	Timer A3 two-phase pulse signal processing select bit		RW
TA4P	Timer A4 two-phase pulse signal processing select bit		RW

TAiUD (Timer Ai up/down flag) (i = 0 to 4) (b4 to b0)

Enabled in event counter mode (when not using two-phase pulse signal processing).

TA2P (Timer A2 two-phase pulse signal processing select bit) (b5)
TA3P (Timer A3 two-phase pulse signal processing select bit) (b6)
TA4P (Timer A4 two-phase pulse signal processing select bit) (b7)

Set these bits to 0 when not using two-phase pulse signal processing.

16.2.14 Timer Ai Mode Register (TAiMR) (i = 0 to 4)

Timer Ai Mode Register (i = 0 to 4)

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	After Reset
TA0MR to TA4MR	0336h to 033Ah	00h

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation 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 (PWM) mode or programmable output mode	RW
TMOD1			RW
MR0	Function varies with the operation mode.		RW
MR1			RW
MR2			RW
MR3			RW
TCK0	Function varies with operation mode.		RW
TCK1			RW

16.3 Operations

16.3.1 Common Operations

16.3.1.1 Operating Clock

The count source for each timer acts as a clock, controlling such timer operations as counting and reloading.

If the conditions to start counting are met, the counter not operating starts counting at the count timing of the first count source. For this reason, a delay exists between when the count start conditions are met and the counter starts counting. Figure 16.4 shows Output Example of One-Shot Timer Mode.

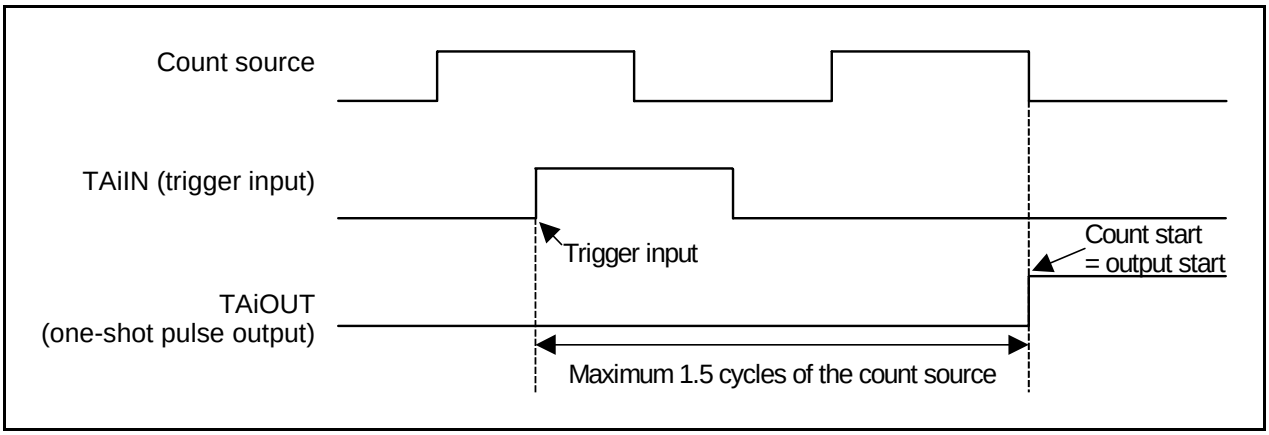


Figure 16.4 Output Example of One-Shot Timer Mode

16.3.1.2 Counter Reload Timing

Timer Ai starts counting from the value (n) set in the TAi register. The TAi register consists of a counter and a reload register. The counter starts decrementing the count source from n, reloads a value in the reload register at the next count source after the value becomes 0000h, and continues decrementing. (When incrementing, the counter reloads a value in the reload register at the next count source after the value becomes FFFFh.)

The value written in the TAi register is reflected in the counter and the reload register at the timings below.

- When the count is stopped
- Between when the count starts and the first count source is input
  A value written to the TAi register is immediately written to the counter and the reload register.

- After the count starts and the first count source is input
  A value written to the TAi register is immediately written to the reload register. The counter continues counting and reloads the value in the reload register at the next count source after the value becomes 0000h (or FFFFh).

16.3.1.3 Count Source

Internal clocks are counted in timer mode, one-shot timer mode, PWM mode, and programmable output mode. (See Figure 16.1 “Timer A and B Count Sources”.) Table 16.5 lists Timer A Count Source.

f1 is any of the clocks listed below. (Refer to 8. “Clock Generator”.) Set thePCKSTP11 bit in the PCLKSTP1 register to 0 (f1 provide enabled) when using the f1.

- Main clock divided by 1 (no division)
- PLL clock divided by 1 (no division)
- fOCO-S divided by 1 (no division)

Table 16.5 Timer A Count Source

Count Source	Bit Set Value				Remarks
	PCLK0	TCS3 TCS7	TCS2 to TCS0 TCS6 to TCS4	TCK1 to TCK0	
f1TIMAB	1	0	—	00b	f1
		1	000b	—	
f2TIMAB	0	0	—	00b	f1 divided by 2
		1	000b	—	
f8TIMAB	—	0	—	01b	f1 divided by 8
	—	1	001b	—	
f32TIMAB	—	0	—	10b	f1 divided by 32
	—	1	010b	—	
f64TIMAB	—	1	011b	—	f1 divided by 64
fOCO-S	—	1	101b	—	fOCO-S
fC32	—	0	—	11b	fC32
	—	1	110b	—	

PCLK0: Bit in the PCLKR register
TCS7 to TCS0: Bits in registers TACS0 to TACS2
TCK1 to TCK0: Bits in the TAiMR register (i = 0 to 4)

16.3.2 Timer Mode

In timer mode, the timer counts a count source generated internally. Table 16.6 lists Timer Mode Specifications, Table 16.7 lists Registers and the Setting in Timer Mode, and Figure 16.5 shows Operation Example in Timer Mode.

Table 16.6 Timer Mode Specifications

Item	Specification
Count source	f1TIMAB, f2TIMAB, f8TIMAB, f32TIMAB, f64TIMAB, fOCO-S, fC32
Count operation	• Decrement • When the timer underflows, it reloads the reload register contents and continues counting.
Counter cycles	$\frac{(n + 1)}{fj}$ n: set value of TAi register 0000h to FFFFh fj: frequency of count source
Count start condition	Set the TAiS bit in the TABSR register to 1 (start counting).
Count stop condition	Set the TAiS bit to 0 (stop counting).
Interrupt request generation timing	Timer underflow
TAiIN pin function	I/O port or gate input
TAiOUT pin function	I/O port or pulse output
Read from timer	Count value can be read by reading the TAi register.
Write to timer	• When not counting Value written to the TAi register is written to both reload register and counter. • When counting Value written to the TAi register is written to only reload register (transferred to counter when reloaded next).
Selectable functions	• Gate function Counting can be started and stopped by an input signal to the TAiIN pin. • Pulse output function Whenever the timer underflows, the output polarity of the TAiOUT pin is inverted. When the TAiS bit is set to 0 (stop counting), the pin outputs a low-level signal. • Output polarity control While the output polarity of the TAiOUT pin is inverted (the TAiS bit is set to 0 (stop counting)), the pin outputs a high-level signal.

i = 0 to 4

Table 16.7 Registers and Their Setting in Timer Mode (1)

Register	Bit	Setting
PCLKR	PCLK0	Select the count source.
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	Set to 0 when using f1.
PWMFS	PWMFSi	Set to 0.
TACS0 to TACS2	7 to 0	Select the count source.
TAPOFS	POFSi	Select the output polarity when the MR0 bit in the TAiMR register is 1 (pulse output).
TAOW	TAiOW	Set to 0.
TAi1	15 to 0	- (does not need to be set)
TABSR	TAiS	Set to 1 when starting counting. Set to 0 when stopping counting.
ONSF	TAiOS	Set to 0.
	TAZIE	Set to 0.
	TA0TGH to TA0TGL	Set to 00b.
TRGSR	TAiTGH to TAiTGL	Set to 00b.
UDF	TAiUD	Set to 0.
	TAiP	Set to 0.
TAi	15 to 0	Set the counter value.
TAiMR	7 to 0	Refer to the TAiMR register below

i = 0 to 4

Note:

1. This table does not describe a procedure.

Timer Mode

Timer Ai Mode Register (i = 0 to 4)

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	0

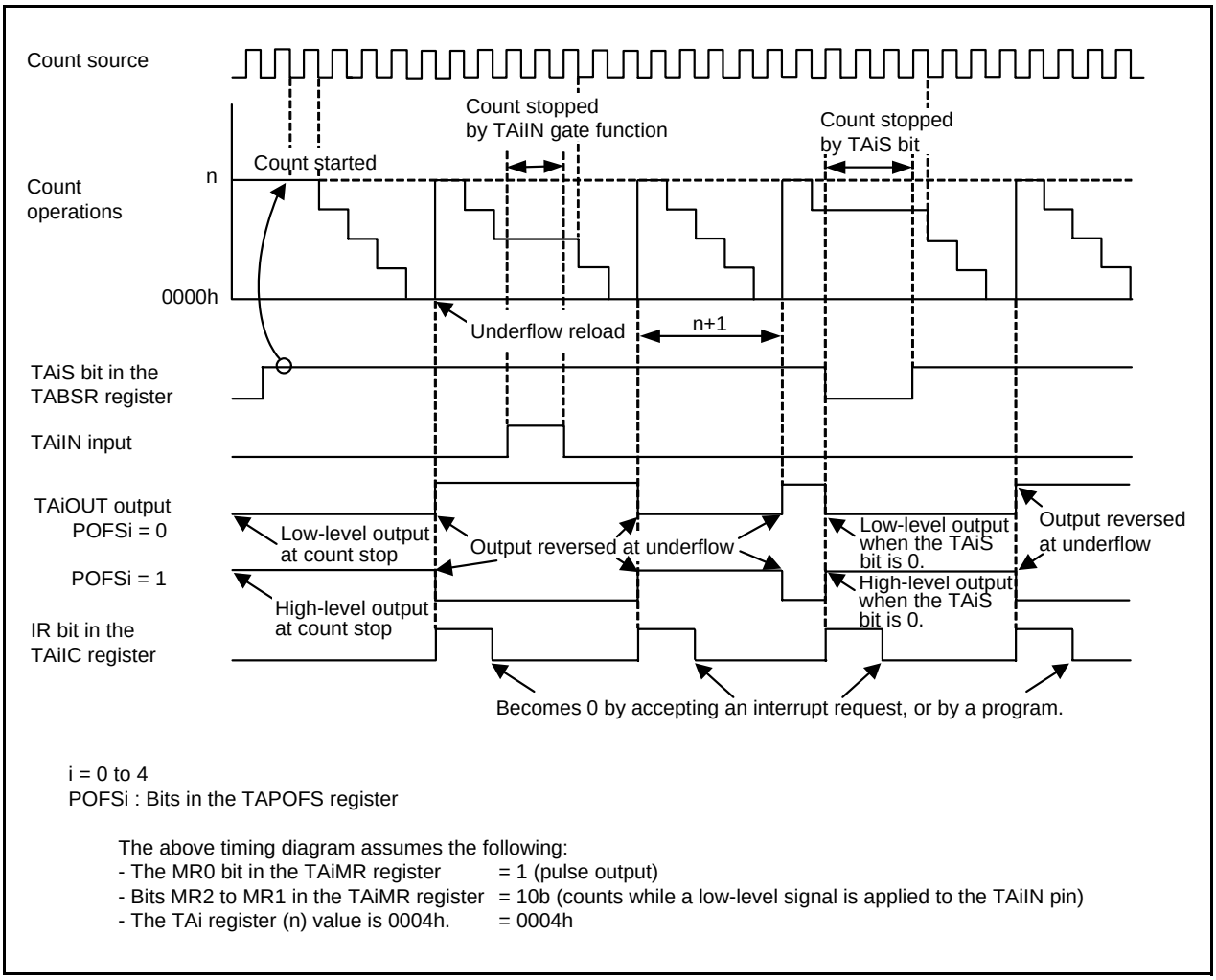
Symbol	Address	After Reset
TA0MR to TA4MR	0336h to 033Ah	00h

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode	RW
TMOD1			RW
MR0	Pulse output function select bit	0 : No pulse output (TAiOUT pin functions as I/O port) 1 : Pulse output (TAiOUT pin functions as a pulse output pin)	RW
MR1	Gate function select bit	b4 b3 0 0 : 0 1 : } Gate function not available (TAiIN pin functions as I/O port) 1 0 : Counts while input on the TAiIN pin is low 1 1 : Counts while input on the TAiIN pin is high	RW
MR2			RW
MR3	Set to 0 in timer mode.		RW
TCK0	Count source select bit	b7 b6 0 0 : f1TIMAB or f2TIMAB 0 1 : f8TIMAB 1 0 : f32TIMAB 1 1 : fC32	RW
TCK1			

TCK1 to TCK0 (Count source select bit) (b7 to b6)

These bits are enabled when the TCS3 bit or TCS7 bit in registers TACS0 to TACS2 is set to 0 (TCK0 to TCK1 enabled).

Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.



i = 0 to 4
POFSi : Bits in the TAPOFS register

The above timing diagram assumes the following:
- The MR0 bit in the TAiMR register = 1 (pulse output)
- Bits MR2 to MR1 in the TAiMR register = 10b (counts while a low-level signal is applied to the TAiIN pin)
- The TAi register (n) value is 0004h. = 0004h

Figure 16.5 Operation Example in Timer Mode

16.3.3 Event Counter Mode (When Not Processing Two-Phase Pulse Signal)

In event counter mode, the timer counts pulses from an external device, or overflows/underflows of other timers. Timers A2, A3, and A4 can count two-phase external signals (refer to 16.3.4 “Event Counter Mode (When Processing Two-Phase Pulse Signal)”). Table 16.8 lists Event Counter Mode Specifications (When Not Processing Two-Phase Pulse Signal). Table 16.9 lists Registers and the Setting in Event Counter Mode (When Not Processing Two-Phase Pulse Signal). Figure 16.6 shows Operation Example in Event Counter Mode.

Table 16.8 Event Counter Mode Specifications (When Not Processing Two-Phase Pulse Signal)

Item	Specification
Count source	• External signals input to the TAiIN pin (active edge can be selected by a program) • Timer B2 overflows or underflows • Timer Aj overflows or underflows (j = i - 1, except j = 4 if i = 0) • Timer Ak overflows or underflows (k = i + 1, except k=0 if i = 4)
Count operations	• Increment or decrement can be selected by a program. • When the timer overflows or underflows, it reloads the reload register contents and continues counting. When selecting free-run type, the timer continues counting without reloading.
Number of counts	When selecting reload type: • FFFFh - n + 1 for increment • n + 1 for decrement n: set value of the TAi register 0000h to FFFFh
Count start condition	Set the TAiS bit in the TABSR register to 1 (start counting).
Count stop condition	Set the TAiS bit to 0 (stop counting).
Interrupt request generation timing	Timer overflow or underflow
TAiIN pin function	I/O port or count source input
TAiOUT pin function	I/O port or pulse output
Read from timer	Count value can be read by reading the TAi register.
Write to timer	• When not counting Value written to the TAi register is written to both reload register and counter. • When counting Value written to the TAi register is written to only reload register (transferred to counter when reloaded next).
Selectable functions	• Free-run count function Even when the timer overflows or underflows, the reload register content is not reloaded. • Pulse output function Whenever the timer underflows or underflows, the output polarity of the TAiOUT pin is inverted. When the TAiS bit is set to 0 (stop counting), the pin outputs a low-level signal. • Output polarity control While the output polarity of the TAiOUT pin is inverted (the TAiS bit is set to 0 (stop counting)), the pin outputs a high-level signal.

i = 0 to 4

Table 16.9 Registers and Settings in Event Counter Mode (When Not Processing Two-Phase Pulse Signal) (1)

Register	Bit	Setting
PCLKR	PCLK0	- (setting unnecessary)
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	- (setting unnecessary)
PWMFS	PWMFSi	Set to 0.
TACS0 to TACS2	7 to 0	- (setting unnecessary)
TAPOFS	POFSi	Select the output polarity when the MR0 bit in the TAiMR register is 1 (pulse output).
TAOW	TAiOW	Set to 0.
TAi1	15 to 0	- (setting unnecessary)
TABSR	TAiS	Set to 1 when starting counting. Set to 0 when stopping counting.
ONSF	TAiOS	Set to 0.
	TAZIE	Set to 0.
	TA0TGH to TA0TGL	Select a count source.
TRGSR	TAiTGH to TAiTGL	Select a count source.
UDF	TAiUD	Select a count operation.
	TAiP	Set to 0.
TAi	15 to 0	Set the counter value.
TAiMR	7 to 0	Refer to the TAiMR register below.

i = 0 to 4

Note:

1. This table does not describe a procedure.

Event Counter Mode (When Not Using Two-Phase Pulse Signal Processing)
Timer Ai Mode Register (i = 0 to 4)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
		0	0			0	1	TA0MR to TA4MR	0336h to 033Ah	00h

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	RW
TMOD1			RW
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin functions as I/O port) 1 : Pulse is output (TAiOUT pin functions as pulse output pin)	RW
MR1	Count polarity select bit	0 : Counts falling edge of external signal 1 : Counts rising edge of external signal	RW
MR2	Set to 0 in event counter mode.		RW
MR3	Set to 0 in event counter mode.		RW
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	RW
TCK1	Can be 0 or 1 when not using two-phase pulse signal processing.		RW

MR1 (Count polarity select bit) (b3)

This bit is enabled when bits TAiTGH to TAiTGL in the ONSF or TRGSR register are 00b (TAiIN pin input).

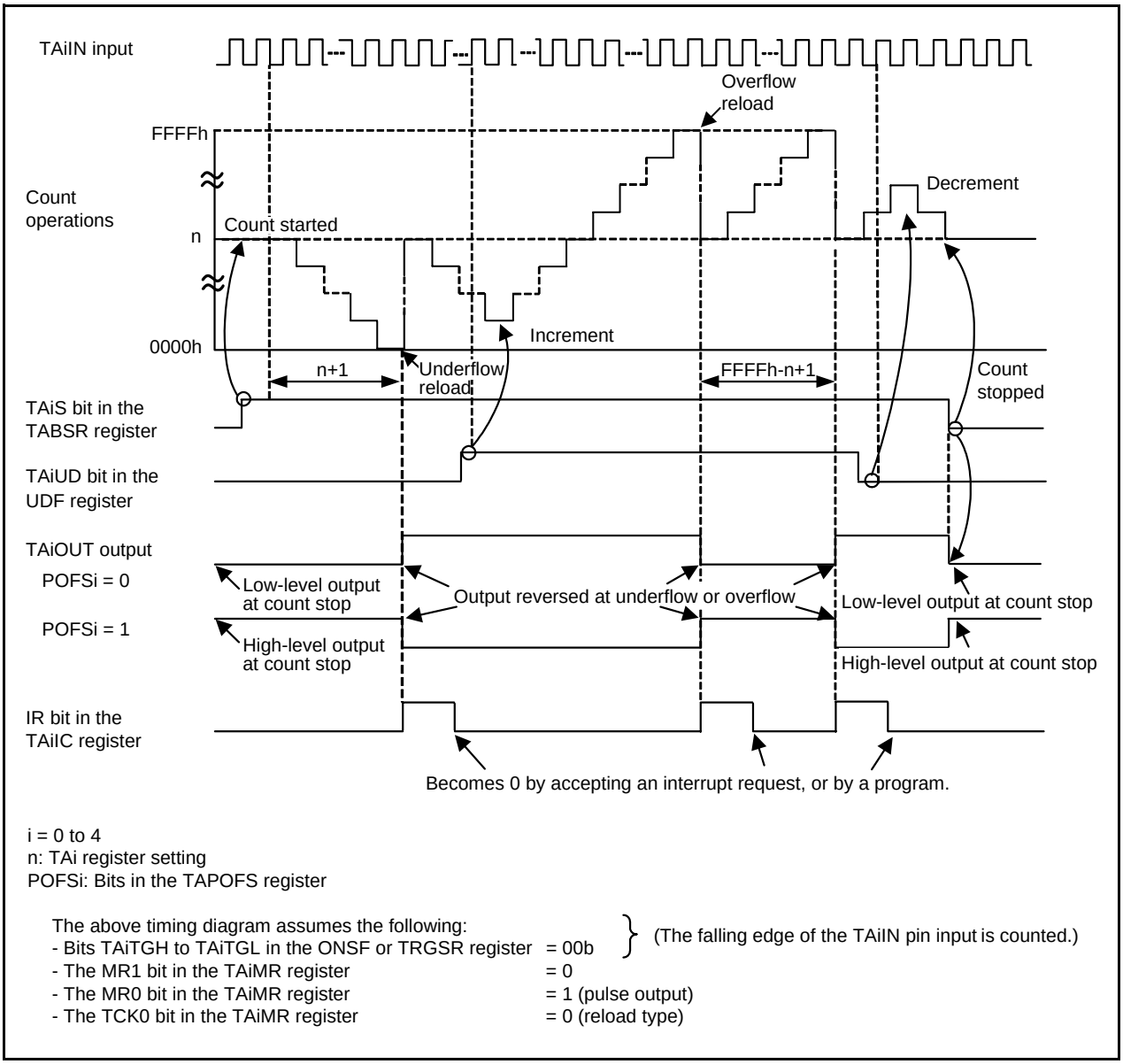


Figure 16.6 Operation Example in Event Counter Mode

16.3.4 Event Counter Mode (When Processing Two-Phase Pulse Signal)

Timers A2, A3, and A4 can be used to count two-phase pulse signals. Table 16.10 lists Event Counter Mode Specifications (When Processing Two-Phase Pulse Signal with Timers A2, A3, and A4). Table 16.11 lists Registers and the Setting in Event Counter Mode (When Processing Two-Phase Pulse Signal).

Table 16.10 Event Counter Mode Specifications (When Processing Two-Phase Pulse Signal with Timers A2, A3, and A4)

Item	Specification
Count source	Two-phase pulse signals input to the TAiIN or TAiOUT pin
Count operations	• Increment or decrement can be selected by a two-phase pulse signal. • When the timer overflows or underflows, it reloads the reload register contents and continues counting. When selecting free-run type, the timer continues counting without reloading.
Number of counts	When selecting reload type: • FFFFh - n + 1 for increment • n + 1 for decrement n: set value of the TAi register 0000h to FFFFh
Count start condition	Set the TAiS bit in the TABSR register to 1 (start counting).
Count stop condition	Set the TAiS bit to 0 (stop counting).
Interrupt request generation timing	Timer overflow or underflow
TAiIN pin function	Two-phase pulse input
TAiOUT pin function	Two-phase pulse input
Read from timer	Count value can be read by reading timer A2, A3, or A4 register.
Write to timer	• When not counting Value written to the TAi register is written to both reload register and counter. • When counting Value written to the TAi register is written to only reload register (transferred to counter when reloaded next).
Selectable functions	• Select normal or multiply-by-4 processing operation (timer A3). • Counter initialization by Z-phase input (timer A3) The timer count value is initialized to 0 by Z-phase input.

i = 2 to 4

Table 16.11 Registers and Settings in Event Counter Mode (When Processing Two-Phase Pulse Signal) (1)

Register	Bit	Setting
PCLKR	PCLK0	Set to 1.
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	- (setting unnecessary)
PWMFS	PWMFSi	Set to 0.
TACS0 to TACS2	7 to 0	Set to 00h.
TAPOFS	POFSi	Set to 0.
TAOW	TAiOW	Set to 0.
TAi1	15 to 0	- (setting unnecessary)
TABSR	TAiS	Set to 1 when starting counting. Set to 0 when stopping counting.
ONSF	TAiOS	Set to 0.
	TAZIE	Set to 1 when using Z-phase input at timer A3.
	TA0TGH to TA0TGL	Set to 00b.
TRGSR	TAiTGH to TAiTGL	Set to 00b.
UDF	TAiUD	Set to 0.
	TAiP	Set to 1.
TAi	15 to 0	Set the counter value.
TAiMR	7 to 0	Refer to the TAiMR register below.

i = 2 to 4

Note:

1. This table does not describe a procedure.

Event Counter Mode (When Using Two-Phase Pulse Signal Processing)
Timer Ai Mode Register (i = 2 to 4)

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

Symbol	Address	After Reset
TA2MR to TA4MR	0338h to 033Ah	00h

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	RW
TMOD1			RW
MR0	Set to 0 to use two-phase signal processing.		RW
MR1	Set to 0 to use two-phase signal processing.		RW
MR2	Set to 1 to use two-phase signal processing.		RW
MR3	Set to 0 to use two-phase signal processing.		RW
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	RW
TCK1	Two-phase pulse signal processing operation type select bit	0 : Normal processing operation 1 : Multiply-by-4 processing operation	RW

TCK1 (Two-phase pulse signal processing operation type select bit) (b7)

The TCK1 bit can be set only for timer A3 mode register. No matter how this bit is set, timers A2 and A4 always operate in normal processing mode and multiply-by-4 processing mode, respectively.

16.3.4.1 Normal Processing

The timer increments rising edges or decrements falling edges on the TAjIN pin when input signals to the TAjOUT (j = 2, 3) pin is high level.

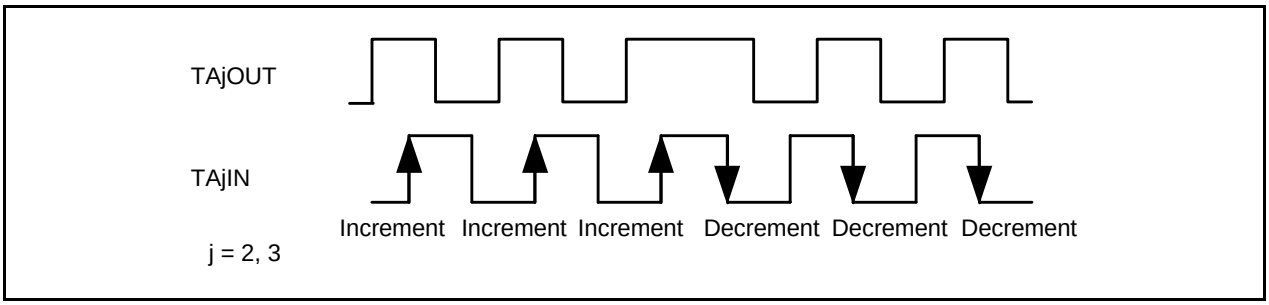


Figure 16.7 Normal Processing

16.3.4.2 Multiply-by-4 Processing

If the phase relationship is such that the input signal to the TAkIN pin goes high when the input signal to the TAkOUT pin (k = 3, 4) is high, the timer increments rising and falling edges of the input signal to pins TAkOUT and TAkIN. If the phase relationship is such that the input signal to the TAkIN pin goes low when the input signal to the TAkOUT pin is high, the timer decrements rising and falling edges of the input signal to pins TAkOUT and TAkIN.

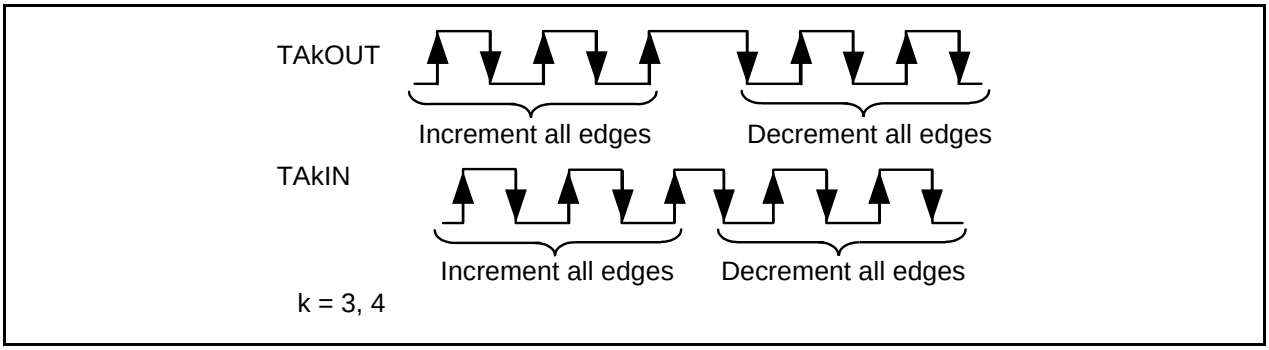


Figure 16.8 Multiply-by-4 Processing

16.3.4.3 Counter Initialization by Two-Phase Pulse Signal Processing

This function initializes the timer count value to 0000h by Z-phase (counter initialization) input during two-phase pulse signal processing.

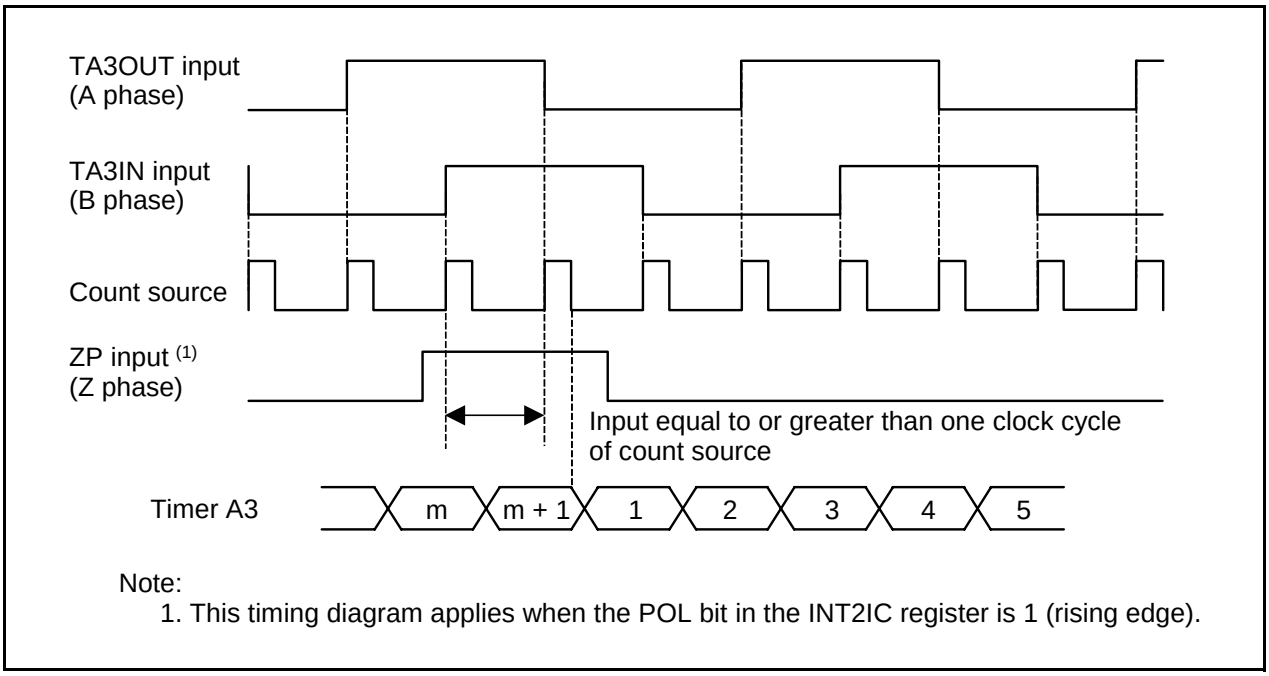
This function can only be used in timer A3 event counter mode during two-phase pulse signal processing, free-running type, multiply-by-4 processing, with Z-phase entered from the ZP pin.

Counter initialization by Z-phase input is enabled by writing 0000h to the TA3 register and setting the TAZIE bit in the ONSF register to 1 (Z-phase input enabled).

Counter initialization is accomplished by Z-phase input edge detection. The rising or falling edge can be selected as the active edge by using the POL bit in the INT2IC register. The Z-phase pulse width applied to the ZP pin must be equal to or greater than one clock cycle of timer A3 count source.

The counter is initialized at the next count timing after accepting Z-phase input. Figure 16.9 shows the Relationship between the Two-Phase Pulse (A-Phase and B-Phase) and the Z-Phase.

When timer A3 overflow or underflow coincides with counter initialization by Z-phase input, a timer A3 interrupt request is generated twice in succession. Do not use the timer A3 interrupt when using this function.



Note:
1. This timing diagram applies when the POL bit in the INT2IC register is 1 (rising edge).

Figure 16.9 Relationship between the Two-Phase Pulse (A-Phase and B-Phase) and the Z-Phase

16.3.5 One-Shot Timer Mode

In one-shot timer mode, the timer is activated only once by one trigger. When the trigger occurs, the timer starts and continues operating for a given period. Table 16.12 lists One-Shot Timer Mode Specifications. Table 16.13 lists Registers and the Setting in One-Shot Timer Mode. Figure 16.10 shows Operation Example in One-Shot Timer Mode.

Table 16.12 One-Shot Timer Mode Specifications

Item	Specification
Count source	f1TIMAB, f2TIMAB, f8TIMAB, f32TIMAB, f64TIMAB, fOCO-S, fC32
Count operations	• Decrement • When the counter reaches 0000h, it stops counting after reloading a new value. • When a trigger occurs while counting, the timer reloads a new value and restarts counting.
Pulse width	$\frac{n}{fj}$  n: set value of the TAi register 0000h to FFFFh However, the counter does not work if 0000h is set. fj: count source frequency
Count start condition	The TAiS bit in the TABSR register is 1 (start counting) and one of the following triggers occurs: • External trigger input from the TAiIN pin • Timer B2 overflow or underflow • Timer Aj overflow or underflow (j = i - 1, except j = 4 if i = 0) • Timer Ak overflow or underflow (k = i + 1, except k = 0 if i = 4) • The TAiOS bit in the ONSF register is set to 1 (one-shot timer start).
Count stop condition	• When the counter is reloaded after reaching 0000h • The TAiS bit is set to 0 (stop counting)
Interrupt request generation timing	When the counter reaches 0000h
TAiIN pin function	I/O port or trigger input
TAiOUT pin function	I/O port or pulse output
Read from timer	An undefined value is read by reading the TAi register.
Write to timer	• When not counting and until the first count source is input after counting starts, the value written to the TAi register is written to both reload register and counter. • When counting (after first count source input), the value written to the TAi register is written to only the reload register (transferred to the counter when reloaded next).
Selectable functions	• Pulse output function The timer outputs a low-level signal when not counting and a high-level signal when counting. • Output polarity control The output polarity of TAiOUT pin is inverted. (While the TAiS bit is set to 0 (stop counting), the pin outputs a high-level signal.)

i = 0 to 4

Table 16.13 Registers and Settings in One-Shot Timer Mode (1)

Register	Bit	Setting
PCLKR	PCLK0	Select the count source.
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	Set to 0 when using f1.
PWMFS	PWMFSi	Set to 0.
TACS0 to TACS2	7 to 0	Select the count source.
TAPOFS	POFSi	Select the output polarity when the MR0 bit in the TAiMR register is 1 (pulse output).
TAOW	TAiOW	Set to 0.
TAi1	15 to 0	- (setting unnecessary)
TABSR	TAiS	Set to 1 when starting counting. Set to 0 when stopping counting.
ONSF	TAiOS	Set to 1 when starting counting while the MR2 bit is 0.
	TAZIE	Set to 0.
	TA0TGH to TA0TGL	Select a count trigger.
TRGSR	TAiTGH to TAiTGL	Select a count trigger.
UDF	TAiUD	Set to 0.
	TAiP	Set to 0.
TAi	15 to 0	Set a high-level pulse width. (2)
TAiMR	7 to 0	Refer to the TAiMR register below.

i = 0 to 4

Notes:

1. This table does not describe a procedure.
2. This applies when the POFSi bit in the TAPOFS register is 0.

One-Shot Timer Mode

Timer Ai Mode Register (i = 0 to 4)

b7	b6	b5	b4	b3	b2	b1	b0
		0				1	0

Symbol	Address	After Reset
TA0MR to TA4MR	0336h to 033Ah	00h

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 1 0 : One-shot timer mode	RW
TMOD1			RW
MR0	Pulse output function select bit	0 : No pulse output (TAiOUT pin functions as I/O port) 1 : Pulse output (TAiOUT pin functions as a pulse output pin)	RW
MR1	External trigger select bit	0 : Falling edge of input signal to TAiIN pin 1 : Rising edge of input signal to TAiIN pin	RW
MR2	Trigger select bit	0 : TAiOS bit enabled 1 : Selected by bits TAiTGH and TAiTGL	RW
MR3	Set to 0 in one-shot timer mode.		RW
TCK0	Count source select bit	b7 b6 0 0 : f1TIMAB or f2TIMAB 0 1 : f8TIMAB 1 0 : f32TIMAB 1 1 : fC32	RW
TCK1			

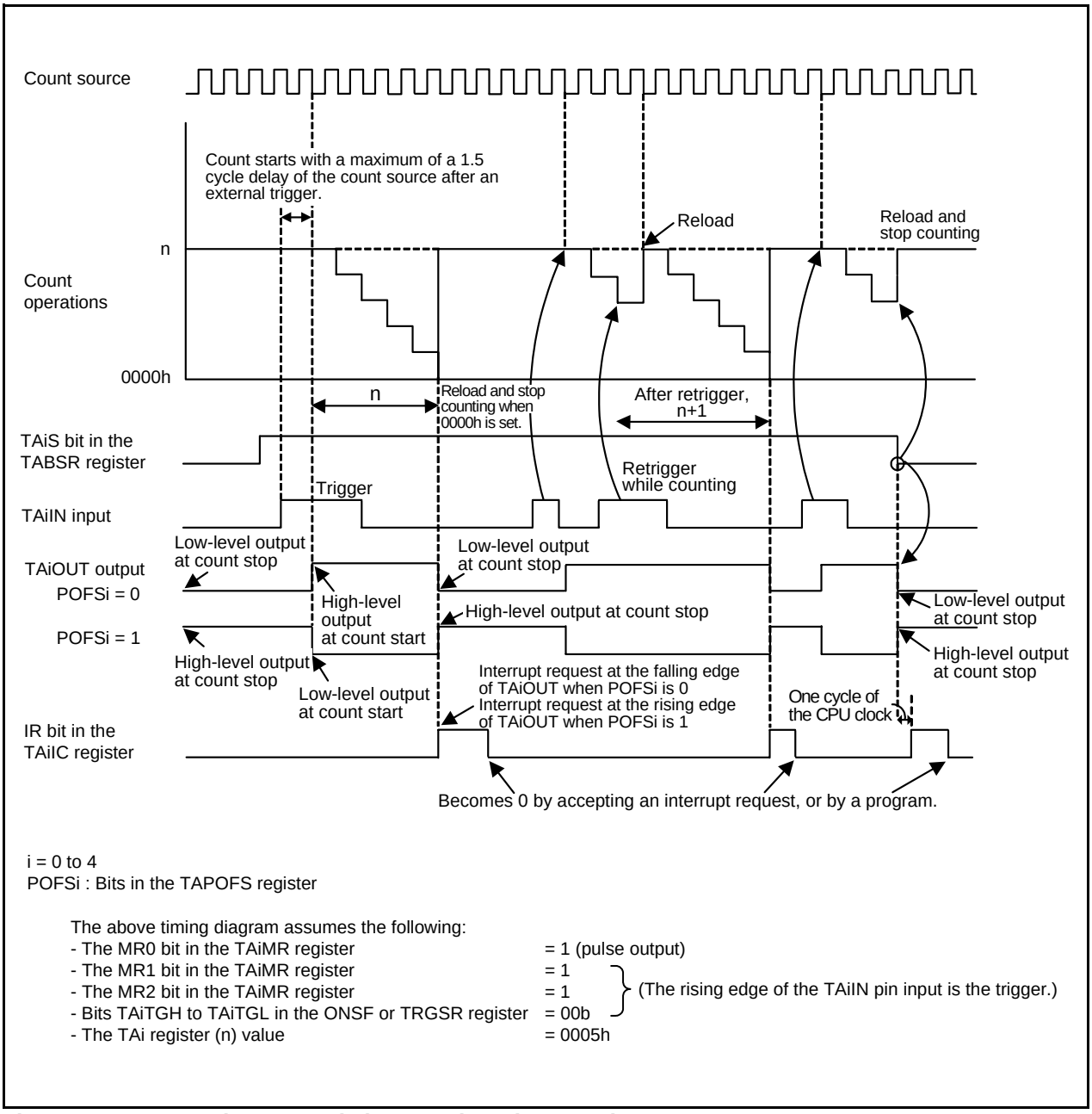
MR1 (External trigger select bit) (b3)

This bit is enabled when the MR2 bit is set to 1 and bits TAiTGH to TAiTGL in the ONSF register or TRGSR register are set to 00b (TAiIN pin input).

TCK1 to TCK0 (Count source select bit) (b7 to b6)

These bits are enabled when the TCS3 bit or TCS7 bit in registers TACS0 to TACS2 is set to 0 (TCK0 to TCK1 enabled).

Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.



i = 0 to 4
POFSi : Bits in the TAPOFS register

The above timing diagram assumes the following:

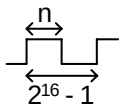
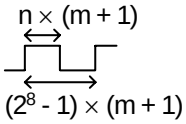
- The MR0 bit in the TAiMR register	= 1 (pulse output)	
- The MR1 bit in the TAiMR register	= 1	(The rising edge of the TAiIN pin input is the trigger.)
- The MR2 bit in the TAiMR register	= 1	
- Bits TAiTGH to TAiTGL in the ONSF or TRGSR register	= 00b	
- The TAi register (n) value	= 0005h	

Figure 16.10 Operation Example in One-Shot Timer Mode

16.3.6 Pulse Width Modulation (PWM) Mode

In PWM mode, the timer outputs pulses of a given width in succession. The counter functions as either 16-bit pulse width modulator or 8-bit pulse width modulator. Table 16.14 lists PWM Mode Specifications. Table 16.15 lists Registers and the Setting in PWM Mode. Figure 16.11 and Figure 16.12 show Operation Example in 16-Bit Pulse Width Modulation Mode and Operation Example in 8-Bit Pulse Width Modulation Mode, respectively.

Table 16.14 PWM Mode Specifications

Item	Specification
Count source	f1TIMAB, f2TIMAB, f8TIMAB, f32TIMAB, f64TIMAB, fOCO-S, fC32
Count operations	• Decrement (operating as an 8-bit or a 16-bit pulse width modulator) • The timer reloads a new value at a rising edge of PWM pulse and continues counting. • The timer is not affected by a trigger that occurs during counting.
16-bit PWM	• Pulse width $\frac{n}{fj}$  • Cycle time $\frac{(2^{16}-1)}{fj}$ n: set value of the TAi register fj: count source frequency
8-bit PWM	• Pulse width $\frac{n \times (m+1)}{fj}$  • Cycle time $\frac{(2^8-1) \times (m+1)}{fj}$ m: set value of the TAi register low-order address n: set value of the TAi register high-order address fj: count source frequency
Count start condition	• The TAiS bit of the TABSR register is set to 1 (start counting). • The TAiS bit is 1 and external trigger input from the TAiIN pin • The TAiS bit is 1 and one of the following external triggers occurs Timer B2 overflow or underflow Timer Aj overflow or underflow (j = i - 1, except j = 4 if i = 0) Timer Ak overflow or underflow (k = i + 1, except k = 0 if i = 4)
Count stop condition	The TAiS bit is set to 0 (stop counting).
Interrupt request generation timing	On the falling edge of the PWM pulse
TAiIN pin function	I/O port or trigger input
TAiOUT pin function	Pulse output
Read from timer	An indeterminate value is read by reading the TAi register.
Write to timer	• When not counting Value written to the TAi register is written to both reload register and counter. • When counting Value written to the TAi register is written to only reload register (transferred to counter when reloaded next).
Selectable functions	• Output polarity control The output polarity of TAiOUT pin is inverted. (While the TAiS bit is set to 0 (stop counting), the pin outputs a high-level signal.)

i = 0 to 4

Table 16.15 Registers and Settings in PWM Mode (1)

Register	Bit	Setting
PCLKR	PCLK0	Select the count source.
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	Set to 0 when using f1.
PWMFS	PWMFSi	Set to 0.
TACS0 to TACS2	7 to 0	Select the count source.
TAPOFS	POFSi	Select the output polarity.
TAOW	TAiOW	Set to 0.
TAi1	15 to 0	- (setting unnecessary)
TABSR	TAiS	Set to 1 when starting counting. Set to 0 when stopping counting.
ONSF	TAiOS	Set to 0.
	TAZIE	Set to 0.
	TA0TGH to TA0GL	Select a count trigger.
TRGSR	TAiTGH to TAiTGL	Select a count trigger.
UDF	TAiUD	Set to 0.
	TAiP	Set to 0.
TAi	15 to 0	Select the pulse width and cycles.
TAiMR	7 to 0	Refer to the TAiMR register below.

i = 0 to 4

Note:

1. This table does not describe a procedure.

Pulse Width Modulation (PWM) Mode

Timer Ai Mode Register (i = 0 to 4)

b7	b6	b5	b4	b3	b2	b1	b0
						1	1

Symbol	Address	After Reset
TA0MR to TA4MR	0336h to 033Ah	00h

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 1 1 : PWM mode or programmable output mode	RW
TMOD1			RW
MR0	Pulse output function select bit	0 : No pulse output (TAiOUT pin functions as I/O port) 1 : Pulse output (TAiOUT pin functions as a pulse output pin)	RW
MR1	External trigger select bit	0 : Falling edge of input signal to TAiIN pin 1 : Rising edge of input signal to TAiIN pin	RW
MR2	Trigger select bit	0 : Write 1 to the TAiS bit in the TABSR register 1 : Selected by bits TAiTGH to TAiTGL	RW
MR3	16/8-bit PWM mode select bit	0 : 16-bit PWM mode 1 : 8-bit PWM mode	RW
TCK0	Count source select bit	b7 b6 0 0 : f1TIMAB or f2TIMAB 0 1 : f8TIMAB 1 0 : f32TIMAB 1 1 : fC32	RW
TCK1			

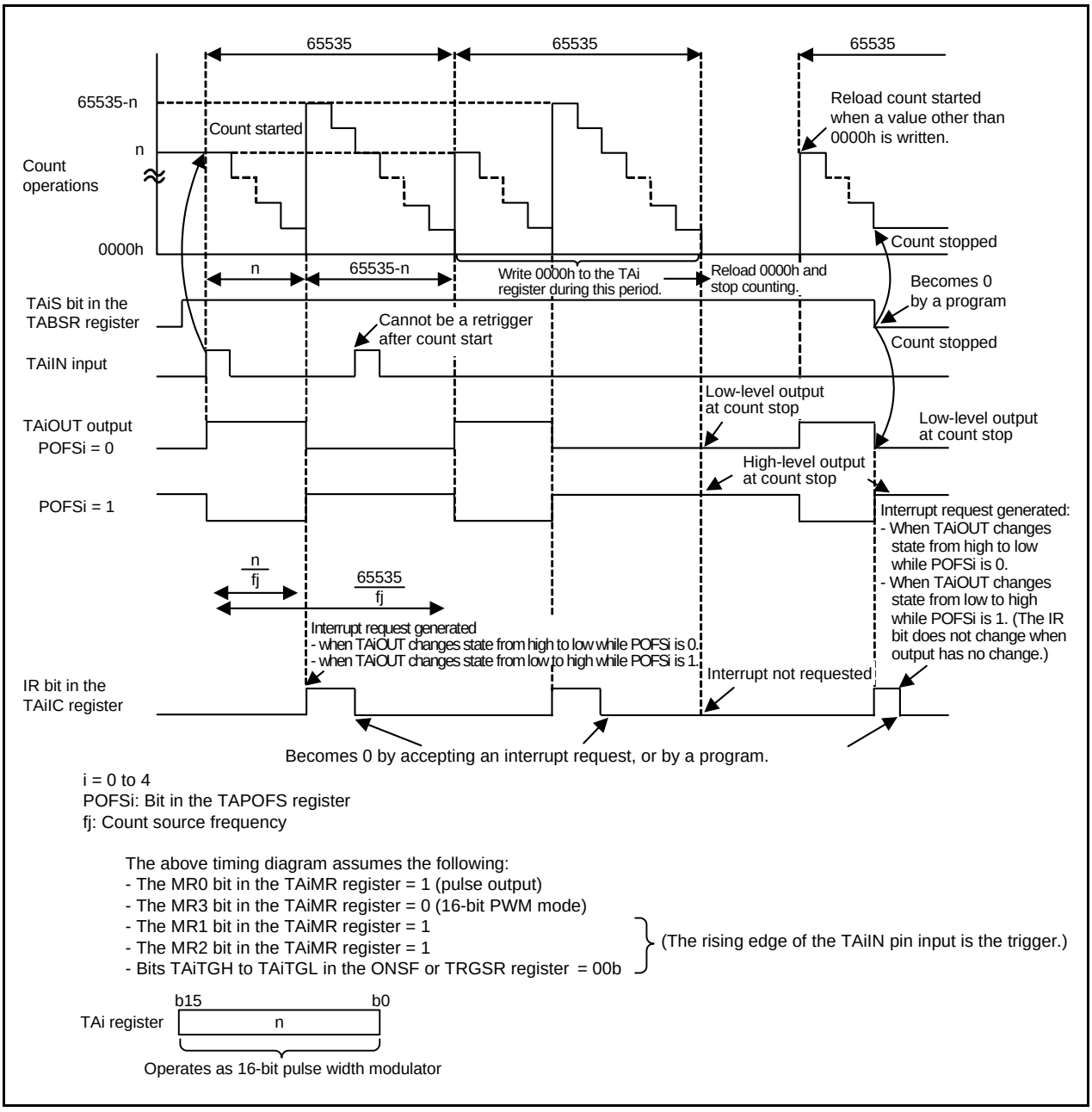
MR1 (External trigger select bit) (b3)

This bit is enabled when the MR2 bit is set to 1 and bits TAiTGH to TAiTGL in the ONSF register or TRGSR register are set to 00b (TAiIN pin input).

TCK1 to TCK0 (Count source select bit) (b7 to b6)

These bits are enabled when the TCS3 bit or TCS7 bit in registers TACS0 to TACS2 is set to 0 (TCK0 to TCK1 enabled).

Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.



i = 0 to 4
POFSi: Bit in the TAPOFS register
fj: Count source frequency

The above timing diagram assumes the following:
- The MR0 bit in the TAiMR register = 1 (pulse output)
- The MR3 bit in the TAiMR register = 0 (16-bit PWM mode)
- The MR1 bit in the TAiMR register = 1
- The MR2 bit in the TAiMR register = 1
- Bits TAiTGH to TAiTGL in the ONSF or TRGSR register = 00b

} (The rising edge of the TAiIN pin input is the trigger.)

Operates as 16-bit pulse width modulator

Figure 16.11 Operation Example in 16-Bit Pulse Width Modulation Mode

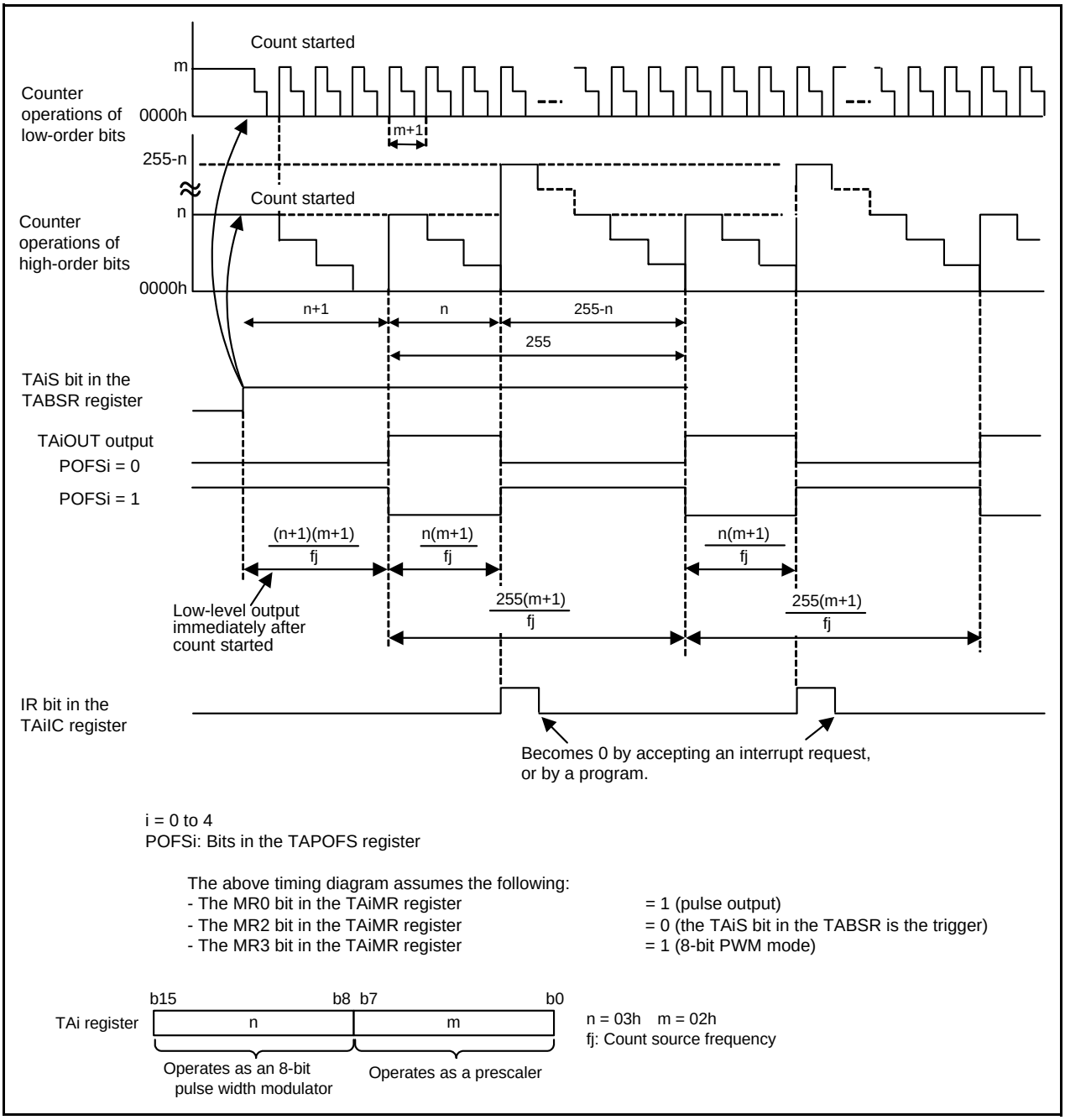
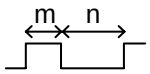


Figure 16.12 Operation Example in 8-Bit Pulse Width Modulation Mode

16.3.7 Programmable Output Mode (Timers A1, A2, and A4)

In programmable output mode, the timer outputs low- and high-levels of pulse width successively. Table 16.16 lists Programmable Output Mode Specifications. Table 16.17 lists Registers and the Setting in Programmable Output Mode. Figure 16.13 shows Operation Example in Programmable Output Mode.

Table 16.16 Programmable Output Mode Specifications

Item	Specification
Count source	f1TIMAB, f2TIMAB, f8TIMAB, f32TIMAB, f64TIMAB, fOCO-S, fC32
Count operations	• Decrement • Reloads on the rising edge of pulse and continues counting • When a trigger occurs while counting, the count is not affected.
Pulse width	• High-level pulse width $\frac{m}{fj}$  • Low-level pulse width $\frac{n}{fj}$ m: set value of the TAi register n: set value of the TAi1 register fj: count source frequency
Count start condition	• The TAiS bit of the TABSR register is set to 1 (start counting). • The TAiS bit is 1 and external trigger input from the TAiIN pin • The TAiS bit is 1 and one of the following external triggers occurs Timer B2 overflow or underflow Timer Aj overflow or underflow (j = i - 1) Timer Ak overflow or underflow (k = i + 1, except k = 0 if i = 4)
Count stop condition	The TAiS bit is set to 0 (stop counting).
Interrupt request generation timing	At the rising edge of pulse
TAiIN pin function	I/O port or trigger input
TAiOUT pin function	Pulse output
Read from timer	An undefined value is read by reading registers TAi and TAi1.
Write to timer	• When writing to registers TAi and TAi1 while not counting, the value is written to both reload register and counter. • When writing to registers TAi and TAi1 while counting, the value is written to the reload register. (transferred to the counter when reloaded next).
Selectable functions	Output polarity control The output polarity of TAiOUT pin is inverted. (While the TAiS bit is set to 0 (stop counting), the pin outputs a high-level signal.)

i = 1, 2, and 4

Table 16.17 Registers and Settings in Programmable Output Mode (1)

Register	Bit	Setting
PCLKR	PCLK0	Select the count source.
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	Set to 0 when using f1.
PWMFS	PWMFSi	Set to 1.
TACS0 to TACS2	7 to 0	Select the count source.
TAPOFS	POFSi	Select the output polarity.
TAOW	TAiOW	Set to 0 to disable output waveform change, and set to 1 to enable output waveform change.
TAi1	15 to 0	Set a low-level pulse width. (2)
TABSR	TAiS	Set to 1 when starting counting. Set to 0 when stopping counting.
ONSF	TAiOS	Set to 0.
	TAZIE	Set to 0.
	TA0TGH to TA0TGL	Select a count trigger.
TRGSR	TAiTGH to TAiTGL	Select a count trigger.
UDF	TAiUD	Set to 0.
	TAiP	Set to 0.
TAi	15 to 0	Set a high-level pulse width. (2)
TAiMR	7 to 0	Refer to the TAiMR register below.

i = 1, 2, and 4

Notes:

1. This table does not describe a procedure.
2. This applies when the POFSi bit in the TAPOFS register is 0.

Programmable Output Mode

Timer Ai Mode Register (i = 1, 2, 4)

b7	b6	b5	b4	b3	b2	b1	b0
		0				1	1

Symbol	Address	After Reset
TA0MR to TA4MR	0336h to 033Ah	00h

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 1 1 : PWM mode or programmable output mode	RW
TMOD1			RW
MR0	Pulse output function select bit	0 : No pulse output (TAiOUT pin functions as I/O port) 1 : Pulse output (TAiOUT pin functions as a pulse output pin)	RW
MR1	External trigger select bit	0 : Falling edge of input signal to TAiIN pin 1 : Rising edge of input signal to TAiIN pin	RW
MR2	Trigger select bit	0 : Write 1 to the TAiS bit in the TABSR register 1 : Selected by bits TAiTGH to TAiTGL	RW
MR3	Set to 0 in programmable output mode.		RW
TCK0	Count source select bit	b1 b0 0 0 : f1TIMAB or f2TIMAB 0 1 : f8TIMAB 1 0 : f32TIMAB 1 1 : fC32	RW
TCK1			

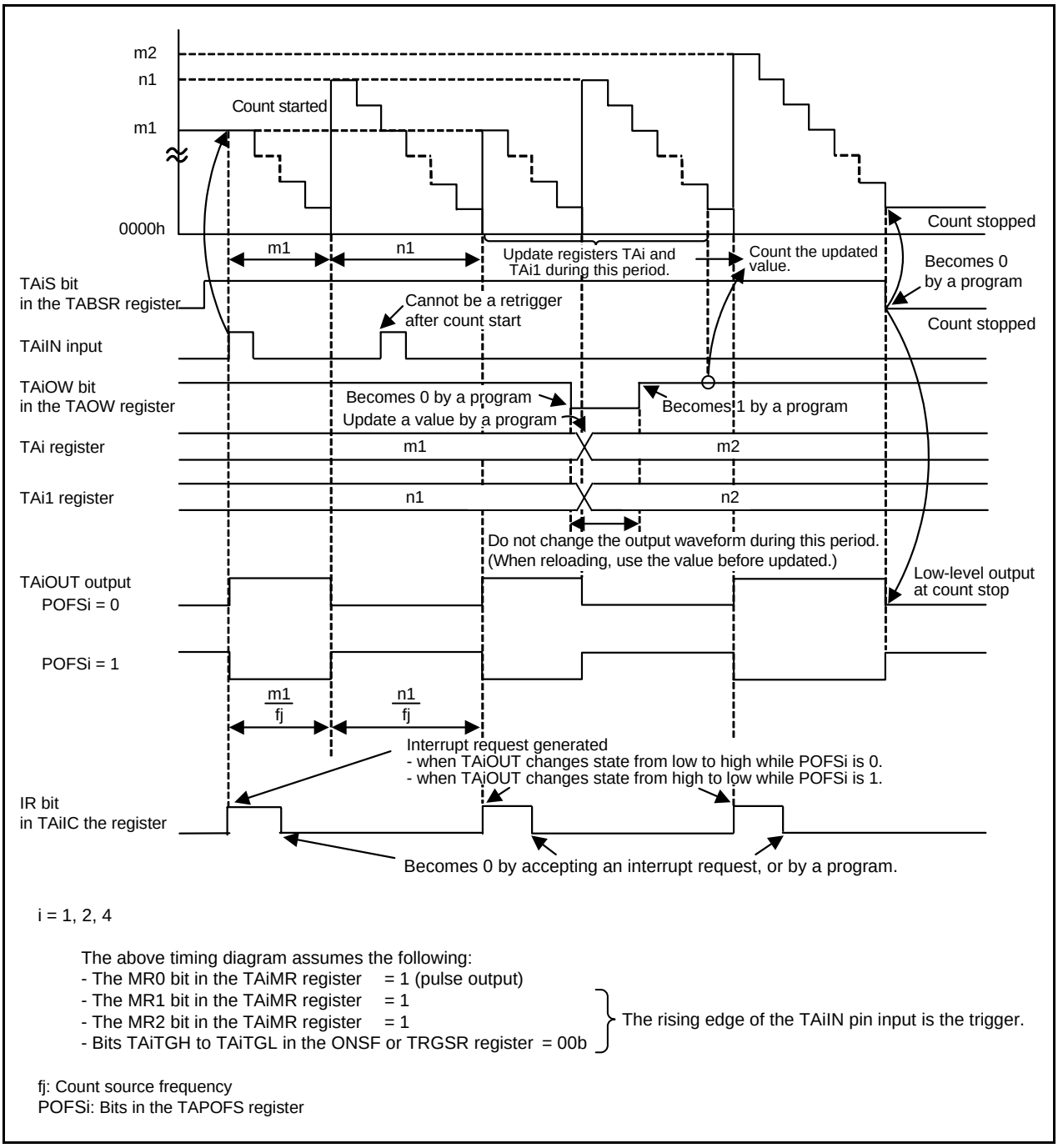
MR1 (External trigger select bit) (b3)

This bit is enabled when the MR2 bit is 1 and bits TAiTGH to TAiTGL in the ONSF register or TRGSR register are set to 00b (TAiIN pin input).

TCK1 to TCK0 (Count source select bit) (b7 to b6)

These bits are enabled when the TCS3 bit or TCS7 bit in registers TACS0 to TACS2 is set to 0 (TCK0 to TCK1 enabled).

Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.



i = 1, 2, 4

The above timing diagram assumes the following:
- The MR0 bit in the TAiMR register = 1 (pulse output)
- The MR1 bit in the TAiMR register = 1
- The MR2 bit in the TAiMR register = 1
- Bits TAiTGH to TAiTGL in the ONSF or TRGSR register = 00b

} The rising edge of the TAiIN pin input is the trigger.

fj: Count source frequency
POFSi: Bits in the TAPOFS register

Figure 16.13 Operation Example in Programmable Output Mode

16.4 Interrupts

Refer to individual operation examples for interrupt request generating timing.
Refer to 13.7 “Interrupt Control” for details of interrupt control. Table 16.18 lists Timer A Interrupt Related Registers.

Table 16.18 Timer A Interrupt Related Registers

Address	Register	Symbol	Reset Value
0055h	Timer A0 Interrupt Control Register	TA0IC	XXXX X000b
0056h	Timer A1 Interrupt Control Register	TA1IC	XXXX X000b
0057h	Timer A2 Interrupt Control Register	TA2IC	XXXX X000b
0058h	Timer A3 Interrupt Control Register	TA3IC	XXXX X000b
0059h	Timer A4 Interrupt Control Register	TA4IC	XXXX X000b

The IR bit in the TAiIC register may become 1 (interrupt requested) when the TMOD1 bit in the TAiMR register is changed from 0 to 1 (change from timer mode or event counter mode to one-shot timer mode, PWM mode, or programmable output mode). Make sure to follow the procedure below when setting the TMOD1 bit to 1. Refer to 13.13 “Notes on Interrupts” as well.

(1)Set bits ILVL2 to ILVL0 in the TAiIC register to 000b (interrupt disabled).
(2)Set the TAiMR register.
(3)Set the IR bit in the TAiIC register to 0 (interrupt not requested).

16.5 Notes on Timer A

16.5.1 Common Notes on Multiple Modes

16.5.1.1 Register Setting

The timer stops after reset. Set the mode, count source, counter value, etc., using registers TAiMR, TAi, TAi1, UDF, TRGSR, PWMFS, TACS0 to TACS2, TAPOFS, PCLKR, and bits TAZIE, TA0TGL, and TA0TGH in the ONSF register before setting the TAiS bit in the TABSR register to 1 (count started) (i = 0 to 4).

Always make sure registers TAiMR, UDF, TRGSR, PWMFS, TACS0 to TACS2, TAPOFS, PCLKR and bits TAZIE, TA0TGL, TA0TGH in the ONSF register are modified while the TAiS bit is 0 (count stopped), regardless of whether after reset or not.

16.5.1.2 Event or Trigger

When bits TAiTGH to TAiTGL in the registers ONSF or TRGSR are 01b, 10b, or 11b, an event or a trigger occurs when an interrupt request of the selected timer is generated. An event or trigger occurs while an interrupt is disabled because an interrupt request signal is generated regardless of the I flag, IPL, or interrupt control registers.

For some modes of the timers selected using bits TAiTGH to TAiTGL, an interrupt request is generated by a source other than overflow or underflow.

For example, when using pulse-period measurement mode or pulse-width measurement mode in timer B2, an interrupt request is generated at an active edge of the measurement pulse. For details, refer to the “Interrupt request generation timing” in each mode’s specification table.

16.5.1.3 Influence of $\overline{SD}$

When a low-level signal is applied to the $\overline{SD}$ pin while the IVPCR1 bit in the TB2SC register is 1 (three-phase output forcible cutoff by input on $\overline{SD}$ pin enabled), the following pins become high-impedance:

P7_2/CLK2/TA1OUT/V, P7_3/$\overline{CTS2}$/$\overline{RTS2}$/TA1IN/V, P7_4/TA2OUT/W,
P7_5/TA2IN/W, P8_0/TA4OUT/RXD5/SCL5/U, P8_1/TA4IN/$\overline{CTS5}$/$\overline{RTS5}$/U

16.5.2 Timer A (Timer Mode)

16.5.2.1 Read from Timer

While counting, the counter value can be read at any time by reading the TAi register. However, if the counter is read at the same time as it is reloaded, the read value is FFFFh. Also, if the counter is read before it starts counting, or after a value is set in the TAi register while not counting, the set value is read.

16.5.3 Timer A (Event Counter Mode)

16.5.3.1 Read from Timer

While counting, the counter value can be read at any time by reading the TAi register. However, while reloading, FFFFh can be read in underflow, and 0000h in overflow. When the counter is read before it starts counting and after a value is set in the TAi register while not counting, the set value is read.

16.5.4 Timer A (One-Shot Timer Mode)

16.5.4.1 Stop While Counting

When setting the TAiS bit to 0 (count stopped), the following occurs:

- The counter stops counting and the contents of the reload register are reloaded.
- The TAiOUT pin outputs a low-level signal when the POFSi bit in the TAPOFS register is 0, and outputs a high-level signal when it is 1.
- After one cycle of the CPU clock, the IR bit in the TAiIC register becomes 1 (interrupt requested).

16.5.4.2 Delay between the Trigger Input and Timer Output

As the one-shot timer output is synchronized with an internally generated count source, when an external trigger is selected, a maximum 1.5 cycle delay of the count source occurs between the trigger input to the TAiIN pin and timer output.

16.5.4.3 Changing Operating Modes

The IR bit becomes 1 when the timer operating mode is set with any of the following:

- Selecting one-shot timer mode after reset
- Changing the operating mode from timer mode to one-shot timer mode
- Changing the operating mode from event counter mode to one-shot timer mode

To use the timer Ai interrupt (IR bit), set the IR bit to 0 after the changes listed above are made.

16.5.4.4 Retrigger

When a trigger occurs while counting, the counter reloads the reload register to continue counting after generating a retrigger and decrementing once. To generate a trigger while counting, generate a retrigger after more than one cycle of the timer count source has elapsed following the previous trigger.

When an external trigger is generated, do not generate a retrigger for 300 ns before the count value becomes 0000h. The one-shot timer may stop counting.

16.5.5 Timer A (Pulse Width Modulation Mode)

16.5.5.1 Changing Operating Modes

The IR bit becomes 1 when setting a timer operating mode with any of the following:
- Selecting PWM mode or programmable output mode after reset
- Changing the operating mode from timer mode to PWM mode or programmable output mode
- Changing the operating mode from event counter mode to PWM mode or programmable output mode

To use the timer Ai interrupt (IR bit), set the IR bit to 0 by a program after the changes listed above are made.

16.5.5.2 Stop While Counting

When setting the TAiS bit to 0 (count stopped) during PWM pulse output, the following occur:
When the POFSi bit in the TAPOFS register is 0:
- Counting stops
- When the TAiOUT pin is high, the output level goes low and the IR bit becomes 1.
- When the TAiOUT pin is low, both the output level and the IR bit remain unchanged.

When the POFSi bit in the TAPOFS register is 1:
- Stop counting.
- If the TAiOUT pin output is low, the output level goes high and the IR bit is set to 1.
- If the TAiOUT pin output is high, both the output level and the IR bit remain unchanged.

16.5.6 Timer A (Programmable Output Mode)

16.5.6.1 Changing the Operating Mode

The IR bit becomes 1 when setting a timer operating mode with any of the following:
- Selecting PWM mode or programmable output mode after reset
- Changing the operating mode from timer mode to PWM mode or programmable output mode
- Changing the operating mode from event counter mode to PWM mode or programmable output mode

To use the timer Ai interrupt (IR bit), set the IR bit to 0 by a program after the changes listed above are made.

16.5.6.2 Stop While Counting

When setting the TAiS bit to 0 (count stopped) during pulse output, the following occur:
When the POFSi bit in the TAPOFS register is 0:
- Counting stops.
- When the TAiOUT pin is high, the output level goes low.
- When the TAiOUT pin is low, the output level remains unchanged.
- The IR bit remains unchanged.

When the POFSi bit in the TAPOFS register is 1:
- Counting stops
- When the TAiOUT pin output is low, the output level goes high.
- When the TAiOUT pin output is high, the output level remains unchanged.
- The IR bit remains unchanged.

17. Timer B

17.1 Introduction

Timers B0 to B5 are provided for timer B. Each timer operates independently of the others. Table 17.1 lists Timer B Specifications, Figure 17.1 shows Timer A and B Count Sources, Figure 17.2 shows the Timer B Configuration, Figure 17.3 shows the Timer B Block Diagram, and Table 17.2 lists the I/O Ports.

Table 17.1 Timer B Specifications

Item	Specification
Configuration	16-bit timer × 6
Operating mode	• Timer mode The timer counts an internal count source. • Event counter mode The timer counts pulses from an external device, or overflows and underflows of other timers. • Pulse period/pulse width measurement modes The timer measures pulse period or pulse width of an external signal.
Interrupt source	Overflow/underflow/active edge of measurement pulse × 6

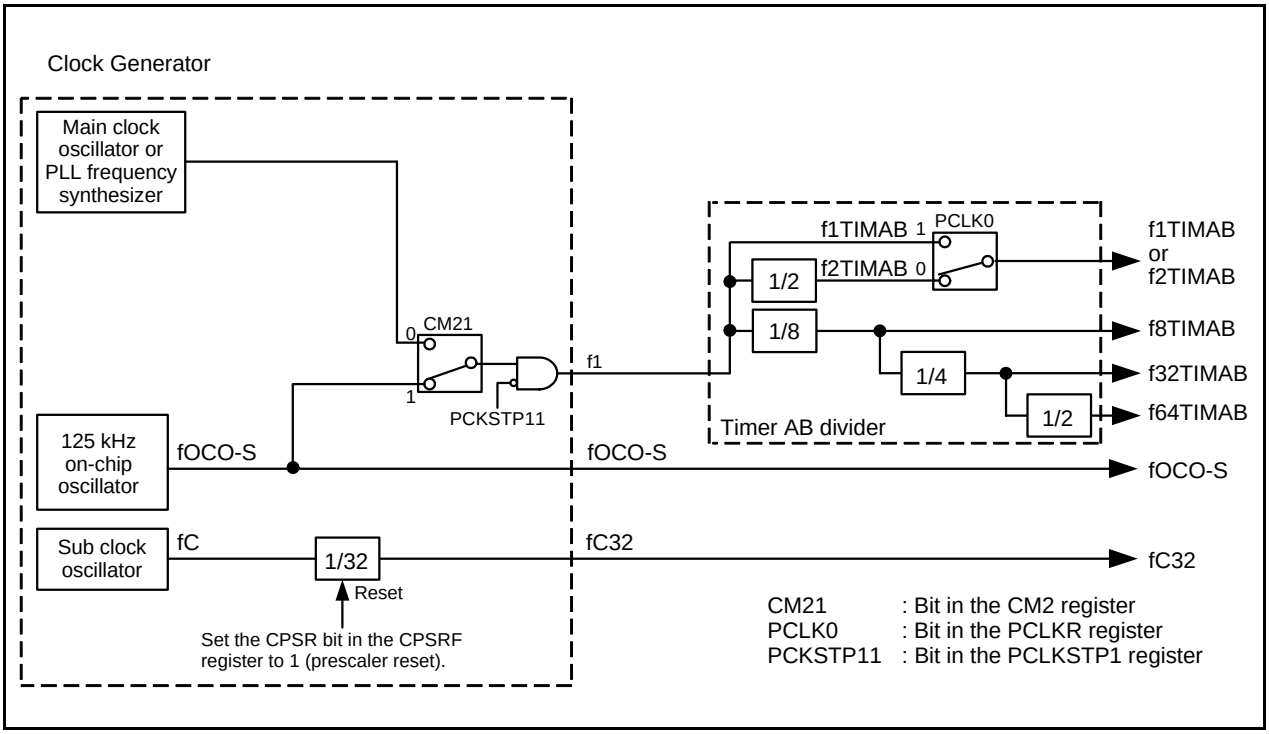


Figure 17.1 Timer A and B Count Sources

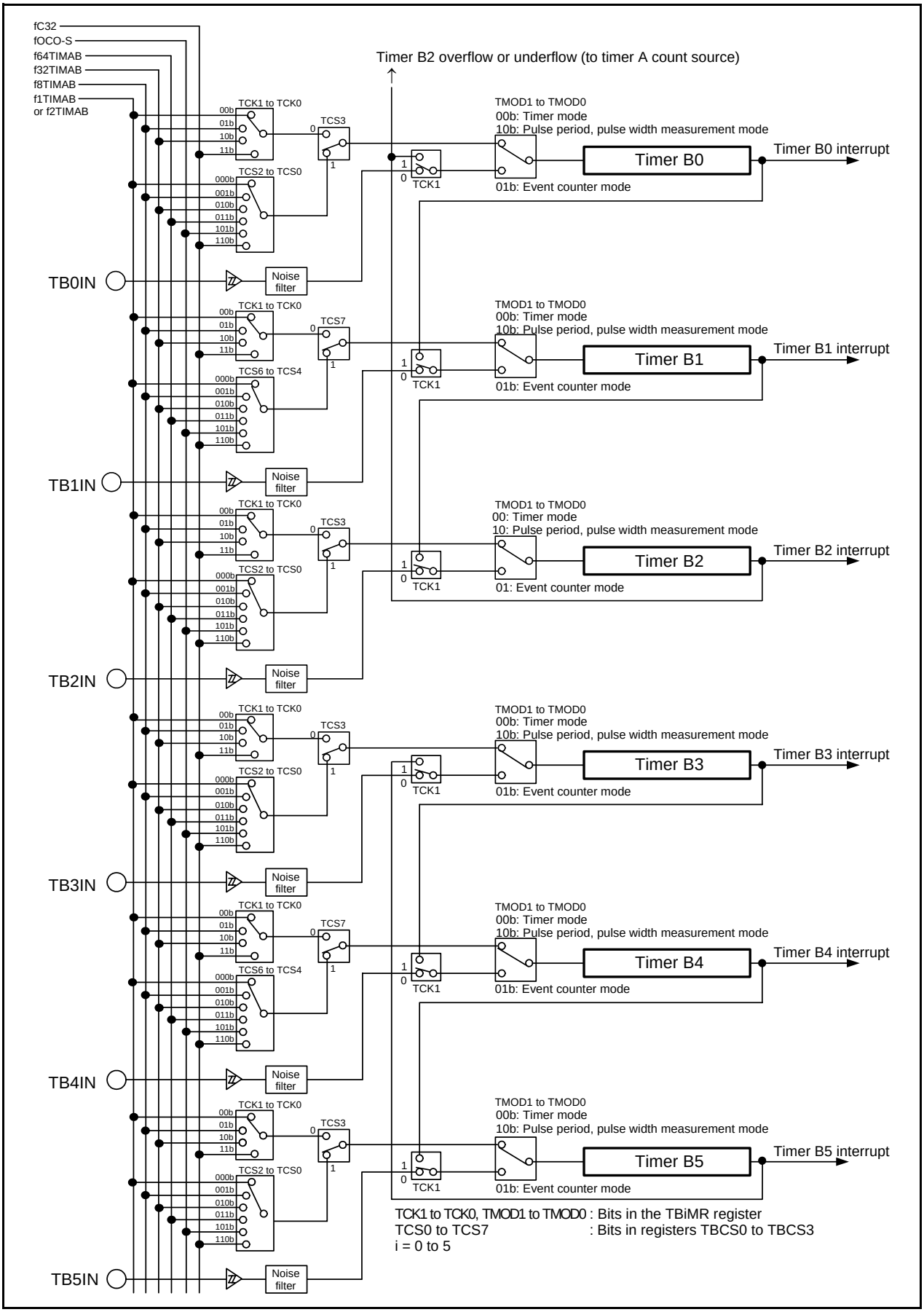


Figure 17.2 Timer B Configuration

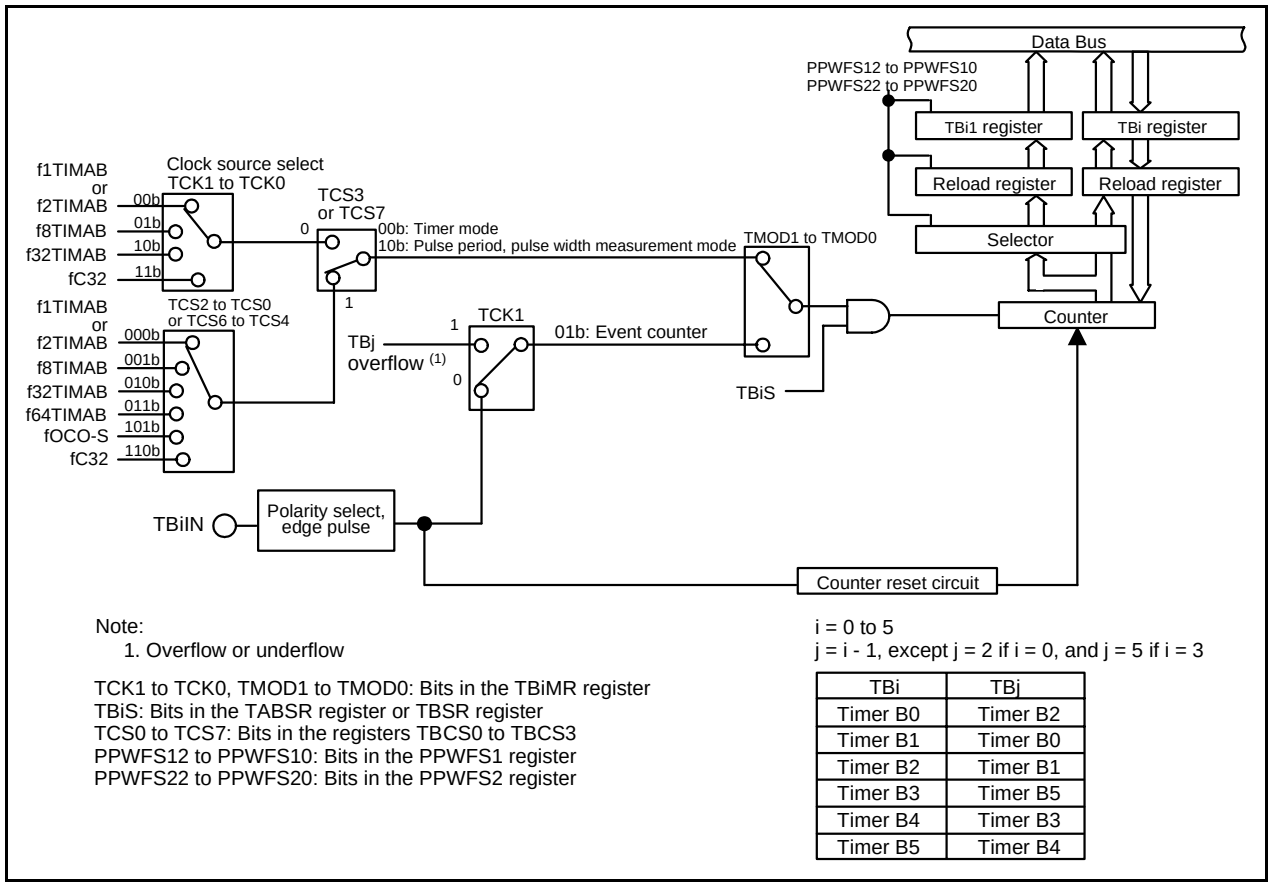


Figure 17.3 Timer B Block Diagram

Table 17.2 I/O Ports

Pin Name	I/O	Function
TBiIN	Input (1)	Count source input (event counter mode) Measurement pulse input (pulse period measurement mode, pulse width measurement mode)

i = 0 to 5

Note:

1. When using the TBiIN pin for input, set the port direction bit corresponding to the pin to 0 (input mode).

17.2 Registers

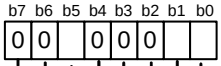
Table 17.3 lists registers associated with timer B.
Refer to “registers and the setting” in each mode for registers and bit settings.

Table 17.3 Registers

Address	Register	Symbol	Reset Value
0012h	Peripheral Clock Select Register	PCLKR	0000 0011b
0015h	Clock Prescaler Reset Flag	CPSRF	0XXX XXXXb
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
01C0h	Timer B0-1 Register	TB01	XXh
01C1h			XXh
01C2h	Timer B1-1 Register	TB11	XXh
01C3h			XXh
01C4h	Timer B2-1 Register	TB21	XXh
01C5h			XXh
01C6h	Pulse Period/Pulse Width Measurement Mode Function Select Register 1	PPWFS1	XXXX X000b
01C8h	Timer B Count Source Select Register 0	TBCS0	00h
01C9h	Timer B Count Source Select Register 1	TBCS1	X0h
01E0h	Timer B3-1 Register	TB31	XXh
01E1h			XXh
01E2h	Timer B4-1 Register	TB41	XXh
01E3h			XXh
01E4h	Timer B5-1 Register	TB51	XXh
01E5h			XXh
01E6h	Pulse Period/Pulse Width Measurement Mode Function Select Register 2	PPWFS2	XXXX X000b
01E8h	Timer B Count Source Select Register 2	TBCS2	00h
01E9h	Timer B Count Source Select Register 3	TBCS3	X0h
0300h	Timer B3/B4/B5 Count Start Flag	TBSR	000X XXXXb
0310h	Timer B3 Register	TB3	XXh
0311h			XXh
0312h	Timer B4 Register	TB4	XXh
0313h			XXh
0314h	Timer B5 Register	TB5	XXh
0315h			XXh
031Bh	Timer B3 Mode Register	TB3MR	00XX 0000b
031Ch	Timer B4 Mode Register	TB4MR	00XX 0000b
031Dh	Timer B5 Mode Register	TB5MR	00XX 0000b
0320h	Count Start Flag	TABSR	00h
0330h	Timer B0 Register	TB0	XXh
0331h			XXh
0332h	Timer B1 Register	TB1	XXh
0333h			XXh
0334h	Timer B2 Register	TB2	XXh
0335h			XXh
033Bh	Timer B0 Mode Register	TB0MR	00XX 0000b
033Ch	Timer B1 Mode Register	TB1MR	00XX 0000b
033Dh	Timer B2 Mode Register	TB2MR	00XX 0000b

17.2.1 Peripheral Clock Select Register (PCLKR)

Peripheral Clock Select Register



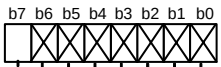
Symbol	Address	After Reset
PCLKR	0012h	0000 0011b

Bit Symbol	Bit Name	Function	RW
PCLK0	Timers A and B clock select bit (clock source for timers A and B, and multi-master I^2C-bus interface)	0: f2TIMAB/f2IIC 1: f1TIMAB/f1IIC	RW
PCLK1	SI/O clock select bit (clock source for UART0 to UART2, UART5 to UART7, SI/O3, and SI/O4)	0: f2SIO 1: f1SIO	RW
— (b4-b2)	Reserved bits	Set to 0.	RW
PCLK5	Clock output function expansion bit	0: Selected by bits CM01 to CM00 in the CM0 register 1: Output f1	RW
— (b7-b6)	Reserved bits	Set to 0.	RW

Write to the PCLKR register after setting the PRC0 bit in the PRCR register to 1 (write enabled).

17.2.2 Clock Prescaler Reset Flag (CPSRF)

Clock Prescaler Reset Flag

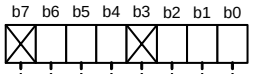


Symbol	Address	After Reset
CPSRF	0015h	0XXX XXXXb

Bit Symbol	Bit Name	Function	RW
— (b6-b0)	No register bits. If necessary, set to 0. Read as undefined value.		—
CPSR	Clock prescaler reset flag	Setting this bit to 1 initializes the prescaler for the timekeeping clock. (Read as 0.)	RW

17.2.3 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1



Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I^2C peripheral clock stop bit (multi-master I^2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

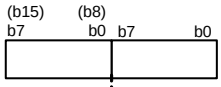
Set the PRC0 bit in the PRCR register to 1 (write enabled) before the PCLKSTP1 register is rewritten.

PCKSTP11 (Timer peripheral clock stop bit) (b1)

Set the PCKSTP11 bit to 0 (f1 provide enabled) when using the f1 as the clock source.

17.2.4 Timer Bi Register (TBi) (i = 0 to 5)

Timer Bi Register (i = 0 to 5)



Symbol	Address	After Reset
TB0	0331h to 0330h	XXXXh
TB1	0333h to 0332h	XXXXh
TB2	0335h to 0334h	XXXXh
TB3	0311h to 0310h	XXXXh
TB4	0313h to 0312h	XXXXh
TB5	0315h to 0314h	XXXXh

Mode	Function	Setting Range	RW
Timer mode	When n = set value, Counter cycle: $\frac{(n + 1)}{fj}$	0000h to FFFFh	RW
Event counter mode	When n = set value, n + 1 count	0000h to FFFFh	RW
Pulse period measurement mode, Pulse width measurement mode	Set an initial value. Measures a pulse period or width. Read the counter value while counting is in progress.	0000h to FFFFh	RW

fj : Count source frequency

Access this register in 16-bit units.

Event Counter Mode

The timer counts pulses from an external device or overflows or underflows of other timers.

Pulse Period Measurement Mode, Pulse Width Measurement Mode

Set these modes when the TBiS bit in the TABSR or TBSR register is set to 0 (count stopped).
Read only (RO) when the TBiS bit in the TABSR or TBSR register is set to 1 (count started).
The counter starts counting the count source at an active edge of the measurement pulse, transfers the count value to a register at the next active edge, and continues counting.
The measurement result can be read by reading the TBi register when bits PPWFS12 to PPWFS10 in the PPWFS1 register and bits PPWFS22 to PPWFS20 in the PPWFS2 register are 0.
While counting is in progress, the counter value can be read by reading the TBi register when bits PPWFS12 to PPWFS10 and bits PPWFS22 to PPWFS20 are 1.

17.2.5 Timer Bi-1 Register (TBi1) (i = 0 to 5)

Timer Bi-1 Register (i = 0 to 5)

(b15) (b8)
b7 b0 b7 b0

Symbol	Address	After Reset
TB01	01C1h to 01C0h	XXXXh
TB11	01C3h to 01C2h	XXXXh
TB21	01C5h to 01C4h	XXXXh
TB31	01E1h to 01E0h	XXXXh
TB41	01E3h to 01E2h	XXXXh
TB51	01E5h to 01E4h	XXXXh

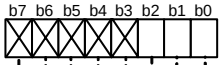
Mode	Function	Setting Range	RW
Pulse period measurement mode Pulse width measurement mode	Measures a pulse period or width	0000h to FFFFh	RO

Access this register in 16-bit units.

When bits PPWFS12 to PPWFS10 in the PPWFS1 register and bits PPWFS22 to PPWFS20 in the PPWFS2 register are 1, the measurement result can be read by reading the TBi-1 register. When these bits are 0, the value in this register is undefined.

17.2.6 Pulse Period/Pulse Width Measurement Mode Function Select Register i (PPWFSi) (i = 1, 2)

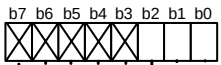
Pulse Period/Pulse Width Measurement Mode Function Select Register 1



Symbol	Address	After Reset
PPWFS1	01C6h	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PPWFS10	Timer B0 pulse period/pulse width measurement mode function select bit	0: Measurement result is stored in the TB0 register. The TB01 register is not used. 1: The counter value is read in the TB0 register. Measurement result is stored in the TB01 register.	RW
PPWFS11	Timer B1 pulse period/pulse width measurement mode function select bit	0: Measurement result is stored in the TB1 register. The TB11 register is not used. 1: The counter value is read in the TB1 register. Measurement result is stored in the TB11 register.	RW
PPWFS12	Timer B2 pulse period/pulse width measurement mode function select bit	0: Measurement result is stored in the TB2 register. The TB21 register is not used. 1: The counter value is read in the TB2 register. Measurement result is stored in the TB21 register.	RW
— (b7-b3)	No register bits. If necessary, set to 0. Read as undefined value.		—

Pulse Period/Pulse Width Measurement Mode Function Select Register 2



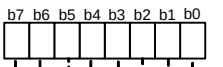
Symbol	Address	After Reset
PPWFS2	01E6h	XXXX X000b

Bit Symbol	Bit Name	Function	RW
PPWFS20	Timer B3 pulse period/pulse width measurement mode function select bit	0: Measurement result is stored in the TB3 register. The TB31 register is not used. 1: The counter value is read in the TB3 register. Measurement result is stored in the TB31 register.	RW
PPWFS21	Timer B4 pulse period/pulse width measurement mode function select bit	0: Measurement result is stored in the TB4 register. The TB41 register is not used. 1: The counter value is read in the TB4 register. Measurement result is stored in the TB41 register.	RW
PPWFS22	Timer B5 pulse period/pulse width measurement mode function select bit	0: Measurement result is stored in the TB5 register. The TB51 register is not used. 1: The counter value is read in the TB5 register. Measurement result is stored in the TB51 register.	RW
— (b7-b3)	No register bits. If necessary, set to 0. Read as undefined value.		—

Enabled in pulse period measurement mode or pulse width measurement mode.

17.2.7 Timer B Count Source Select Register i (TBCSi) (i = 0 to 3)

Timer B Count Source Select Register 0, Timer B Count Source Select Register 2

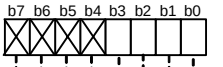


Symbol	Address	After Reset
TBCS0	01C8h	00h
TBCS2	01E8h	00h

Bit Symbol	Bit Name	Function	RW
TCS0	TBi count source select bit	b2 b1 b0 0 0 0: f1TIMAB or f2TIMAB 0 0 1: f8TIMAB 0 1 0: f32TIMAB 0 1 1: f64TIMAB 1 0 0: Do not set 1 0 1: fOCO-S 1 1 0: fC32 1 1 1: Do not set	RW
TCS1			RW
TCS2			RW
TCS3	TBi count source option specified bit	0: TCK0 to TCK1 enabled, TCS0 to TCS2 disabled 1: TCK0 to TCK1 disabled, TCS0 to TCS2 enabled	RW
TCS4	TBj count source select bit	b6 b5 b4 0 0 0: f1TIMAB or f2TIMAB 0 0 1: f8TIMAB 0 1 0: f32TIMAB 0 1 1: f64TIMAB 1 0 0: Do not set 1 0 1: fOCO-S 1 1 0: fC32 1 1 1: Do not set	RW
TCS5			RW
TCS6			RW
TCS7	TBj count source option specified bit	0: TCK0 to TCK1 enabled, TCS4 to TCS6 disabled 1: TCK0 to TCK1 disabled, TCS4 to TCS6 enabled	RW

TBCS0 register: i = 0, j = 1 TBCS2 register: i = 3, j = 4

Timer B Count Source Select Register 1, Timer B Count Source Select Register 3



Symbol	Address	After Reset
TBCS1	01C9h	X0h
TBCS3	01E9h	X0h

Bit Symbol	Bit Name	Function	RW
TCS0	TBi count source select bit	b2 b1 b0 0 0 0: f1TIMAB or f2TIMAB 0 0 1: f8TIMAB 0 1 0: f32TIMAB 0 1 1: f64TIMAB 1 0 0: Do not set 1 0 1: fOCO-S 1 1 0: fC32 1 1 1: Do not set	RW
TCS1			RW
TCS2			RW
TCS3	TBi count source option specified bit	0: TCK0 to TCK1 enabled, TCS0 to TCS2 disabled 1: TCK0 to TCK1 disabled, TCS0 to TCS2 enabled	RW
— (b7-b4)	No register bits. If necessary, set to 0. Read as undefined value.		—

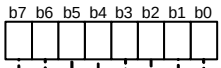
TBCS1 register: i = 2 TBCS3 register: i = 5

TCS2 to TCS0 (TBi count source select bit) (b2 to b0)
TCS6 to TCS4 (TBj count source select bit) (b6 to b4)

Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.

17.2.8 Count Start Flag (TABSR)
Timer B3/B4/B5 Count Start Flag (TBSR)

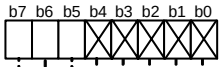
Count Start Flag



Symbol	Address	After Reset
TABSR	0320h	00h

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 count start flag	0: Count stopped 1: Count started	RW
TA1S	Timer A1 count start flag		RW
TA2S	Timer A2 count start flag		RW
TA3S	Timer A3 count start flag		RW
TA4S	Timer A4 count start flag		RW
TB0S	Timer B0 count start flag		RW
TB1S	Timer B1 count start flag		RW
TB2S	Timer B2 count start flag		RW

Timer B3/B4/B5 Count Start Flag



Symbol	Address	After Reset
TBSR	0300h	000X XXXXb

Bit Symbol	Bit Name	Function	RW
— (b4-b0)	No register bits. If necessary, set to 0. Read as undefined value.		—
TB3S	Timer B3 count start flag	0: Stop counting 1: Start counting	RW
TB4S	Timer B4 count start flag		RW
TB5S	Timer B5 count start flag		RW

17.2.9 Timer Bi Mode Register (TBiMR) (i = 0 to 5)

Timer Bi Mode Register (i = 0 to 5)

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TB0MR to TB2MR	033Bh to 033Dh	00XX 0000b
TB3MR to TB5MR	031Bh to 031Dh	00XX 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 0 0: Timer mode 0 1: Event counter mode 1 0: Pulse period measurement mode or pulse width measurement mode 1 1: Do not set	RW
TMOD1			RW
MR0	Function varies with the operation mode.		RW
MR1			RW
— (b4)	No register bit. If necessary, set to 0. Read as undefined value.		—
MR3	Function varies with the operation mode.		RO
TCK0	Count source select bit (function varies with the operation mode)		RW
TCK1			RW

17.3 Operations

17.3.1 Common Operations

17.3.1.1 Operating Clock

The count source for each timer acts as a clock, controlling such timer operations as counting and reloading.

17.3.1.2 Counter Reload Timing

Timer Bi starts counting from the value (n) set in the TBi register. The TBi register consists of a counter and a reload register. The counter starts decrementing the count source from n, reloads a value in the reload register at the next count source after the value becomes 0000h, and continues decrementing. The value written in the TBi register takes effect in the counter and the reload register at the timings below.

- When the count is stopped
- Between the count starts and the first count source is input
  A value written to the TBi register is immediately written to the counter and the reload register.

- After the count starts and the first count source is input
  A value written to the TBi register is immediately written to the reload register.
  The counter continues counting and reloads the value in the reload register at the next count source after the value becomes 0000h.

17.3.1.3 Count Source

Internal clocks are counted in timer mode, pulse period measurement mode, and pulse width measurement mode. (See Figure 17.1 “Timer A and B Count Sources”.) Table 17.4 lists Timer B Count Source.

f1 is any of the clocks listed below. (Refer to 8. “Clock Generator”.) Set thePCKSTP11 bit in the PCLKSTP1 register to 0 (f1 provide enabled) when using the f1.

- Main clock divided by 1 (no division)
- PLL clock divided by 1 (no division)
- fOCO-S divided by 1 (no division)

Table 17.4 Timer B Count Source

Count Source	Bit Set Value				Remarks
	PCLK0	TCS3 / TCS7	TCS2 to TCS0 / TCS4 to TCS6	TCK1 to TCK0	
f1TIMAB	1	0	—	00b	f1
		1	000b	—	
f2TIMAB	0	0	—	00b	f1 divided by 2
		1	000b	—	
f8TIMAB	—	0	—	01b	f1 divided by 8
	—	1	001b	—	
f32TIMAB	—	0	—	10b	f1 divided by 32
	—	1	010b	—	
f64TIMAB	—	1	011b	—	f1 divided by 64
fOCO-S	—	1	101b	—	fOCO-S
fC32	—	0	—	11b	fC32
	—	1	110b	—	

PCLK0: Bit in the PCLKR register
TCS7 to TCS0: Bits in registers TBCS0 toTBCS3
TCK1 to TCK0: Bits in the TBiMR register (i = 0 to 5)

17.3.2 Timer Mode

In timer mode, the timer counts a count source generated internally. Table 17.5 lists Timer Mode Specifications, Table 17.6 lists Registers and the Setting in Timer Mode, and Figure 17.4 shows Operation Example in Timer Mode.

Table 17.5 Timer Mode Specifications

Item	Specification
Count source	f1TIMAB, f2TIMAB, f8TIMAB, f32TIMAB, f64TIMAB, fOCO-S, fC32
Count operations	• Decrement • When the timer underflows, it reloads the reload register contents and continues counting.
Counter cycles	$\frac{1}{(n+1)}$ n: set value of the TBi register 0000h to FFFFh
Count start condition	Set the TBiS bit to 1 (start counting).
Count stop condition	Set the TBiS bit to 0 (stop counting).
Interrupt request generation timing	Timer underflow
TBiIN pin function	I/O port
Read from timer	Count value can be read by reading the TBi register.
Write to timer	• When not counting The value written to the TBi register is written to both the reload register and the counter. • When counting The value written to the TBi register is only written to the reload register (transferred to the counter when reloaded next).

i = 0 to 5
TBiS: Bit in the TABSR or TBSR register

Table 17.6 Registers and Settings in Timer Mode (1)

Register	Bit	Setting
PCLKR	PCLK0	Select the count source.
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	Set to 0 when using f1.
TBi1	15 to 0	- (setting unnecessary)
PPWFS1 to PPWFS2	PPWFS12 to PPWFS10 PPWFS22 to PPWFS20	Set to 0.
TBCS0 to TBCS3	7 to 0	Select the count source.
TABSR TBSR	TAiS	Set to 1 when starting counting. Set to 0 when stopping counting.
TBi	15 to 0	Set the count value.
TBiMR	7 to 0	Refer to the TBiMR register below.

i = 0 to 5
Note:
1. This table does not describe a procedure.

Timer Mode

Timer Bi Mode Register (i = 0 to 5)

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

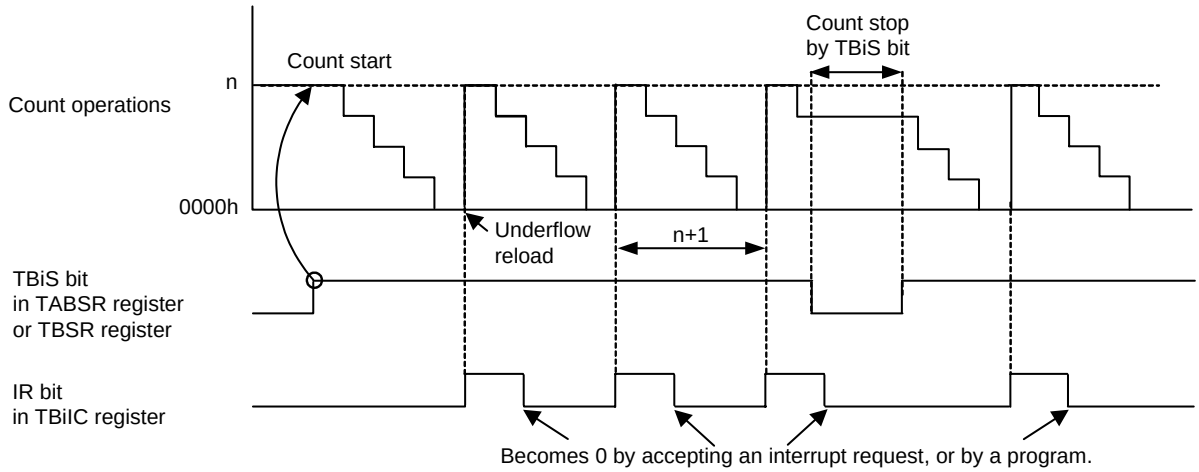
Symbol	Address	After Reset
TB0MR to TB2MR TB3MR to TB5MR	033Bh to 033Dh 031Bh to 031Dh	00XX 0000b 00XX 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode	RW
TMOD1			RW
MR0	Set to 0 in timer mode.		RW
MR1			RW
— (b4)	No register bit. If necessary, set to 0. Read as undefined value.		—
MR3	Write 0 in timer mode. Read as undefined value in timer mode.		RO
TCK0	Count source select bit	b7 b6 0 0 : f1TIMAB or f2TIMAB 0 1 : f8TIMAB 1 0 : f32TIMAB 1 1 : fC32	RW
TCK1			RW

TCK1 to TCK0 (Count source select bit) (b7 to b6)

These bits are enabled when the TCS3 or TCS7 bit in registers TBCS0 to TBCS3 is set to 0 (TCK0 to TCK1 enabled).

Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.



i = 0 to 5

The above timing diagram applies when the register bits are set as follows:
-Value in the TBi register (n) = 0004h

Figure 17.4 Operation Example in Timer Mode

17.3.3 Event Counter Mode

In event counter mode, the timer counts pulses from an external device or overflows and underflows of other timers. Table 17.7 lists Event Counter Mode Specifications, Table 17.8 lists Registers and the Setting in Event Counter Mode, and Figure 17.5 shows Operation Example in Event Counter Mode.

Table 17.7 Event Counter Mode Specifications

Item	Specification
Count source	• External signals input to TBiIN pin (active edge can be selected by a program: rising edge, falling edge, or both rising and falling edges) • Timer Bj overflow or underflow
Count operations	• Decrement • When the timer underflows, it reloads the reload register contents and continues counting.
Number of counts	$\frac{1}{(n+1)}$ n: set value of the TBi register 0000h to FFFFh
Count start condition	Set the TBiS bit to 1 (start counting).
Count stop condition	Set the TBiS bit to 0 (stop counting).
Interrupt request generation timing	Timer underflow
TBiIN pin function	Count source input
Read from timer	Count value can be read by reading the TBi register.
Write to timer	• When not counting Value written to the TBi register is written to both reload register and counter. • When counting Value written to the TBi register is written to only reload register (transferred to counter when reloaded next).

i = 0 to 5 j = i - 1, except j = 2 if i = 0, j = 5 if i = 3
TBiS: Bit in the TABSR or TBSR register

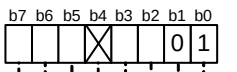
Table 17.8 Registers and Settings in Event Counter Mode (1)

Register	Bit	Setting
PCLKR	PCLK0	- (setting unnecessary)
CPSRF	CPSR	Write a 1 to reset the clock prescaler.
PCLKSTP1	PCKSTP11	- (setting unnecessary)
TBi1	15 to 0	- (setting unnecessary)
PPWFS1 to PPWFS2	PPWFS12 to PPWFS10 PPWFS22 to PPWFS20	Set to 0.
TBCS0 to TBCS3	7 to 0	- (setting unnecessary)
TABSR TBSR	TBiS	Set to 1 when starting counting. Set to 0 when stopping counting.
TBi	15 to 0	Set the count value.
TBiMR	7 to 0	Refer to the TBiMR register below.

i = 0 to 5
Note:
1. This table does not describe a procedure.

Event Counter Mode

Timer Bi Mode Register (i = 0 to 5)



Symbol	Address	After Reset
TB0MR to TB2MR	033Bh to 033Dh	00XX 0000b
TB3MR to TB5MR	031Bh to 031Dh	00XX 0000b

Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	RW
TMOD1			RW
MR0	Count polarity select bit	b3 b2 0 0 : Counts falling edges of external signal 0 1 : Counts rising edges of external signal 1 0 : Counts falling and rising edges of an external signal 1 1 : Do not set	RW
MR1			RW
— (b4)	No register bit. If necessary, set to 0. Read as undefined value.		—
MR3	Write 0 in event counter mode. Read as undefined value in event counter mode.		RO
TCK0	Invalid in event counter mode. Set 0 or 1.		RW
TCK1	Event clock select bit	0 : Input from TBiIN pin 1 : Timer Bj (j = i – 1; however, j = 2 if i = 0, j = 5 if i = 3)	RW

MR1 to MR0 (Count polarity select bit) (b3 to b2)

These bits are enabled when the TCK1 bit is 0 (input from TBiIN pin). If the TCK1 bit is 1 (TBj overflow or underflow), these bits can be set to 0 or 1.

TCK1 (Event clock select bit) (b7)

When the TCK1 bit is 1, an event occurs when an interrupt request of timer Bj (j = i - 1; however, j = 2 if i = 0, j = 5 if i = 3) is generated. An event occurs while an interrupt is disabled because an interrupt request signal is generated regardless of the I flag, IPL, or interrupt control registers

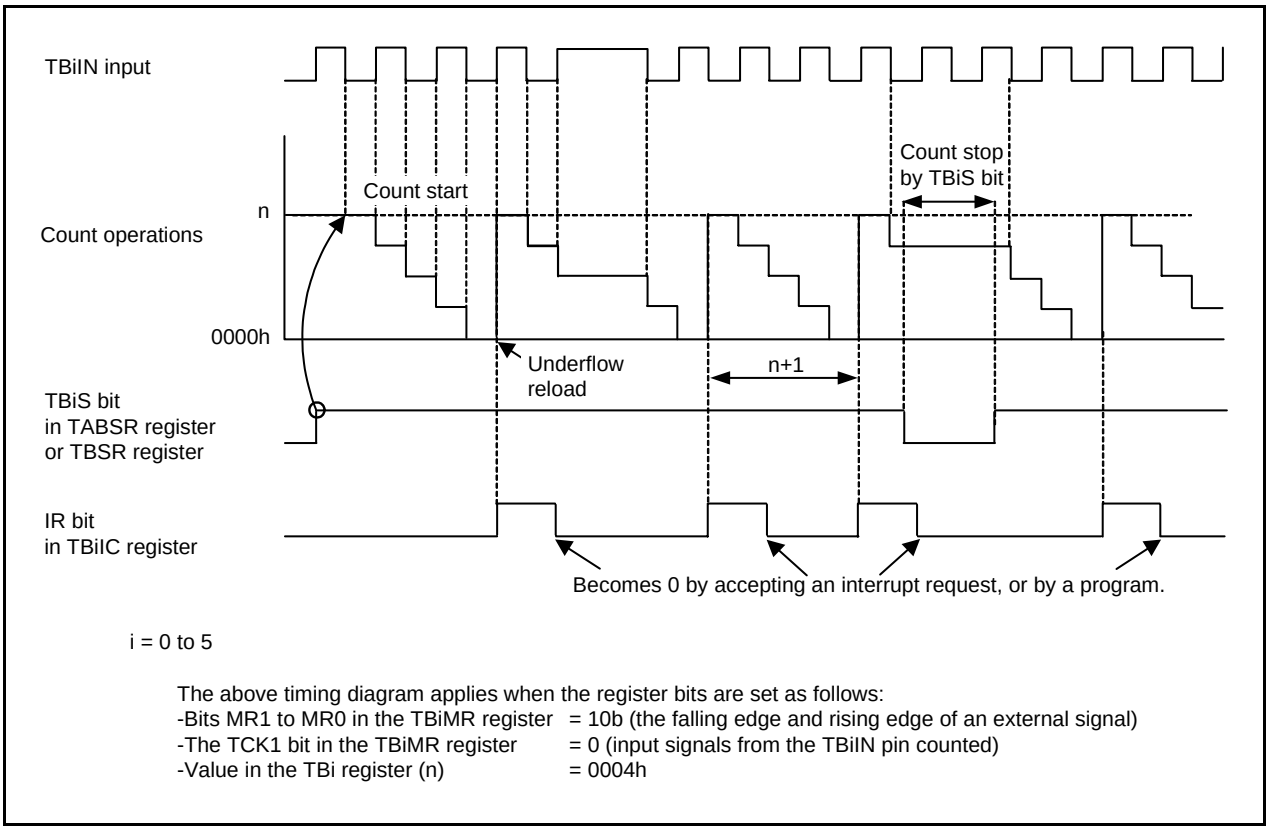


Figure 17.5 Operation Example in Event Counter Mode

17.3.4 Pulse Period/Pulse Width Measurement Modes

In pulse period and pulse width measurement modes, the timer measures pulse period or pulse width of an external signal. Table 17.9 lists Specifications of Pulse Period/Pulse Width Measurement Modes, Table 17.10 lists Registers and the Setting in Pulse Period/Pulse Width Measurement Modes, Figure 17.6 shows Operation Example in Pulse Period Measurement Mode, and Figure 17.7 shows Operation Example in Pulse Width Measurement Mode.

Table 17.9 Specifications of Pulse Period/Pulse Width Measurement Modes

Item	Specification
Count source	f1TIMAB, f2TIMAB, f8TIMAB, f32TIMAB, f64TIMAB, fOCO-S, fC32
Count operations	• Increment • Counter value is transferred to reload register at an active edge of the measurement pulse. The counter value is set to 0000h to continue counting.
Count start condition	Set the TBiS bit to 1 (start counting).
Count stop condition	Set the TBiS bit to 0 (stop counting).
Interrupt request generation timing (3)	• When an active edge of measurement pulse is input (1) • Timer overflow. When an overflow occurs, the MR3 bit in the TBiMR register is set to 1 (overflowed) simultaneously.
TBiIN pin function	Measurement pulse input
Read from timer	When bits PPWFS12 to PPWFS10 and PPWFS22 to PPWFS20 in registers PPWFS1 and PPWFS2 are 0 • Contents of the reload register (measurement result) can be read by reading the TBi register (2) When bits PPWFS12 to PPWFS10 and PPWFS22 to PPWFS20 in registers PPWFS1 and PPWFS2 are 1 • Contents of the counter (counter value) can be read by reading the TBi register • Contents of the reload register (measurement result) can be read by reading the TBi1 register
Write to timer	When not counting, the value written to the TBi register is written to both the reload register and counter.

i = 0 to 5
TBiS: Bit in the TABSR or TBSR register
Notes:
1. No Interrupt request is generated when the first active edge is input after the timer starts counting.
2. Value read from the TBi register is undefined until the second active edge is input after the timer starts counting.
3. When timer Bi in pulse-period measurement mode or pulse-width measurement mode is used as an event or trigger for timer A or timer B other than timer Bi, an event or trigger occurs at both the overflow and active edge of the measurement pulse.

Table 17.10 Registers and Settings in Pulse Period/Pulse Width Measurement Modes (1)

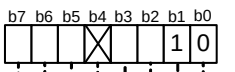
Register	Bit	Setting
PCLKR	PCLK0	Select the count source.
CPSRF	CPSR	Set this bit to reset the clock prescaler.
PCLKSTP1	PCKSTP11	Set to 0 when using f1.
TBi1	15 to 0	Measurement result can be read when the bits in the PPWFS1 or PPWFS2 register corresponding to timer Bi are 1.
PPWFS1 to PPWFS2	PPWFS12 to PPWFS10 PPWFS22 to PPWFS20	Set to 1 to read the counter value while counting.
TBCS0 to TBCS3	7 to 0	Select the count source.
TABSR TBSR	TBiS	Set to 1 when starting counting. Set to 0 when stopping counting.
TBi	15 to 0	Set the initial value. The measurement result can be read when the bits in the PPWFS1 or PPWFS register corresponding to timer Bi are 0. The counter value can be read when the bits in the PPWFS1 or PPWFS2 register corresponding to timer Bi are 1.
TBiMR	7 to 0	Refer to the TBiMR register below.

i = 0 to 5

Note:

1. This table does not describe a procedure.

Pulse Period/Pulse Width Measurement Modes
Timer Bi Mode Register (i = 0 to 5)



Symbol	Address	After Reset
TB0MR to TB2MR TB3MR to TB5MR	033Bh to 033Dh 031Bh to 031Dh	00XX 0000b 00XX 0000b

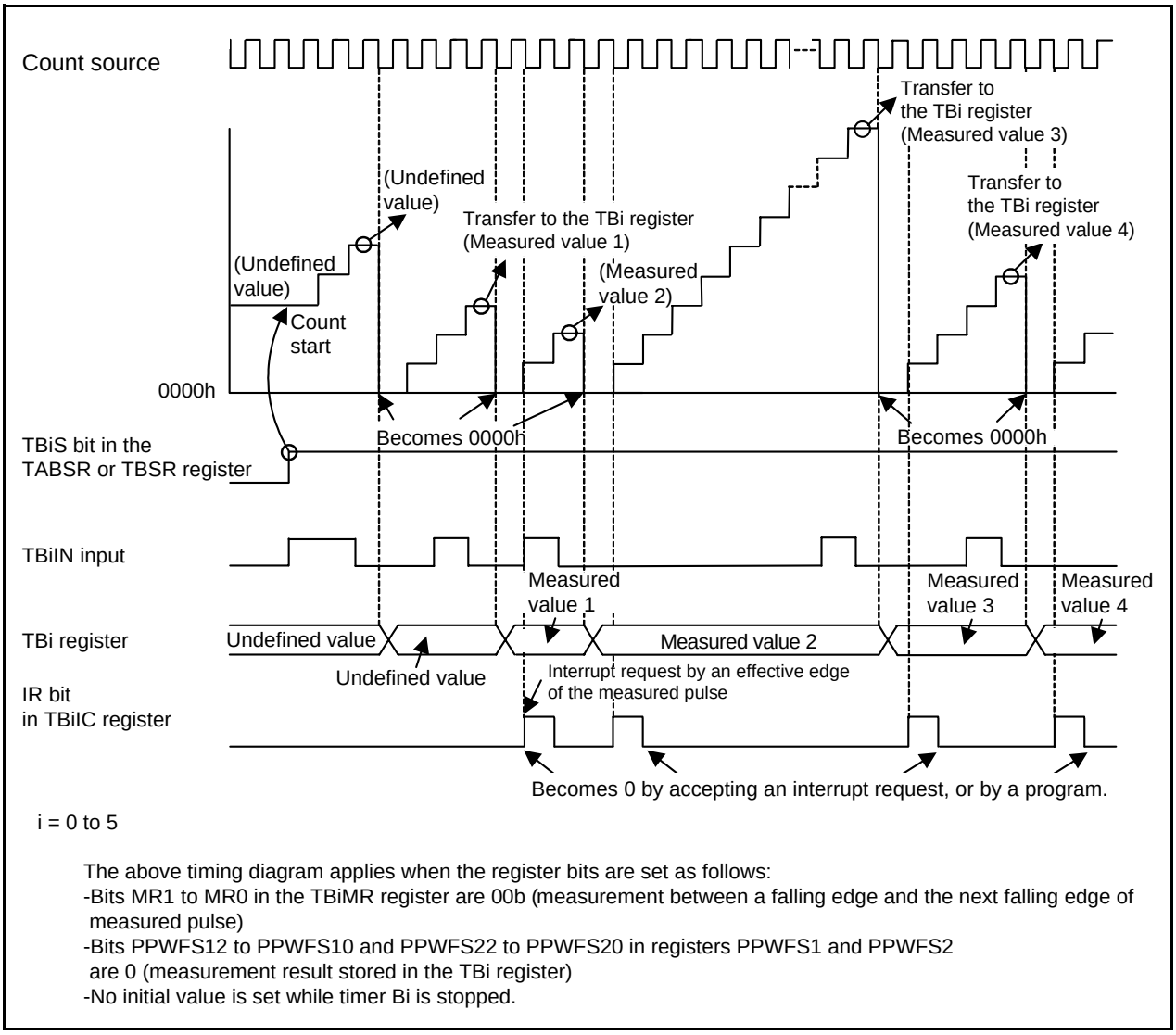
Bit Symbol	Bit Name	Function	RW
TMOD0	Operation mode select bit	b1 b0 1 0 : Pulse period/pulse width measurement modes	RW
TMOD1			RW
MR0	Measurement mode select bit	b3 b2 0 0 : Pulse period measurement (measurement between a falling edge and the next falling edge of measured pulse) 0 1 : Pulse period measurement (measurement between a rising edge and the next rising edge of measured pulse) 1 0 : Pulse width measurement (measurement between a falling edge and the next rising edge of measured pulse and between a rising edge and the next falling edge) 1 1 : Do not set	RW
MR1			RW
— (b4)	No register bit. If necessary, set to 0. Read as undefined value.		—
MR3	Timer Bi overflow flag	0 : No overflow 1 : Overflow	RO
TCK0	Count source select bit	b7 b6 0 0 : f1TIMAB or f2TIMAB 0 1 : f8TIMAB 1 0 : f32TIMAB 1 1 : fC32	RW
TCK1			RW

MR3 (Timer Bi overflow flag) (b5)

This flag is undefined after reset. The MR3 bit is cleared to 0 (no overflow) by writing to the TBiMR register. The MR3 bit cannot be set to 1 by a program.

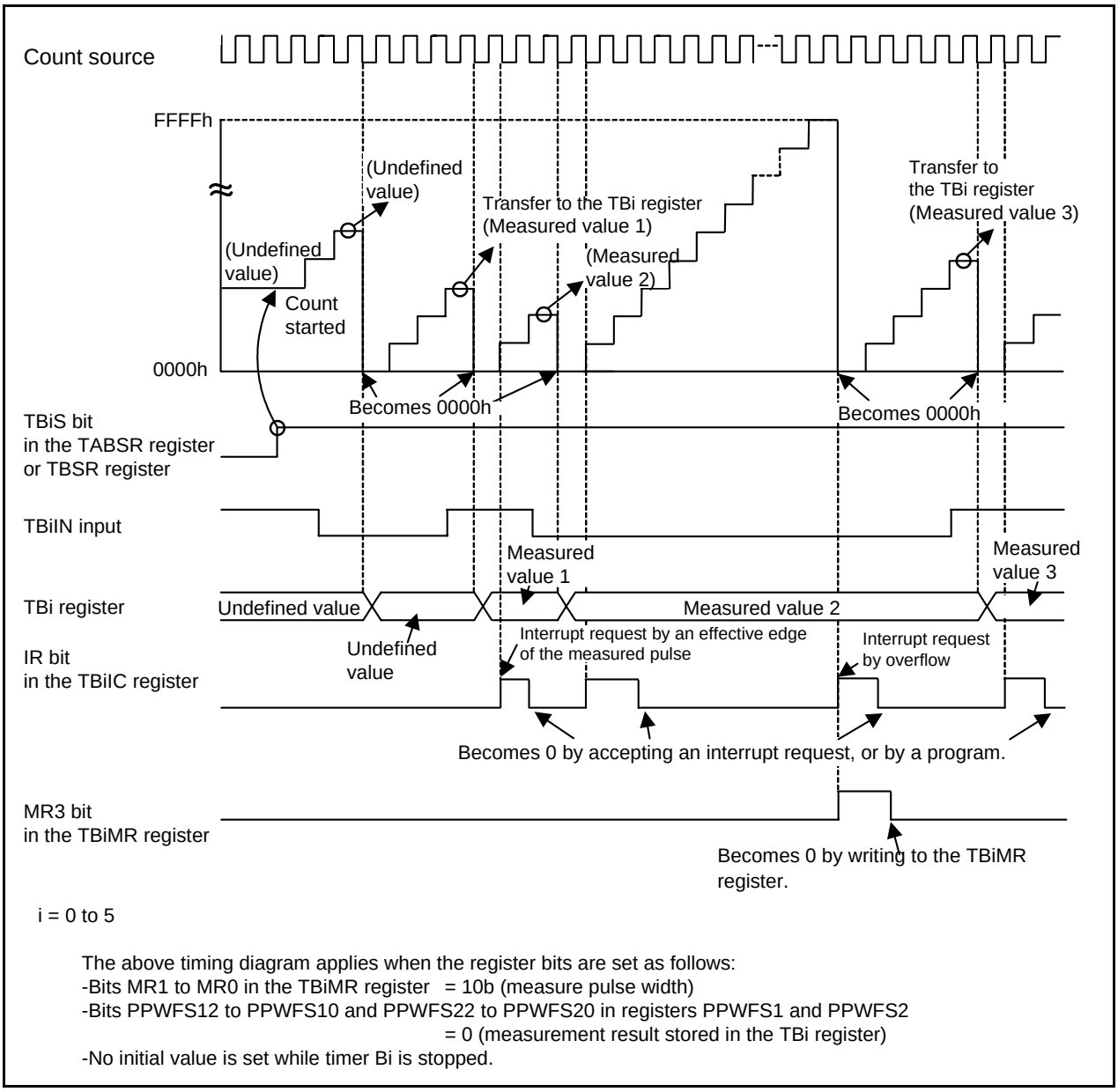
TCK1 to TCK0 (Count source select bit) (b7 to b6)

These bits are enabled when the TCS3 bit or TCS7 bit in registers TBCS0 to TBCS3 is set to 0 (TCK0, TCK1 enabled).
Select f1TIMAB or f2TIMAB by the PCLK0 bit in the PCLKR register.



The above timing diagram applies when the register bits are set as follows:
-Bits MR1 to MR0 in the TBiMR register are 00b (measurement between a falling edge and the next falling edge of measured pulse)
-Bits PPWFS12 to PPWFS10 and PPWFS22 to PPWFS20 in registers PPWFS1 and PPWFS2 are 0 (measurement result stored in the TBi register)
-No initial value is set while timer Bi is stopped.

Figure 17.6 Operation Example in Pulse Period Measurement Mode



i = 0 to 5

The above timing diagram applies when the register bits are set as follows:
-Bits MR1 to MR0 in the TBiMR register = 10b (measure pulse width)
-Bits PPWFS12 to PPWFS10 and PPWFS22 to PPWFS20 in registers PPWFS1 and PPWFS2 = 0 (measurement result stored in the TBi register)
-No initial value is set while timer Bi is stopped.

Figure 17.7 Operation Example in Pulse Width Measurement Mode

17.4 Interrupts

Refer to individual operation examples for interrupt request generating timing.
Refer to 13.7 “Interrupt Control” for details of interrupt control. Table 17.11 lists Timer B Interrupt Related Registers.

Table 17.11 Timer B Interrupt Related Registers

Address	Register	Symbol	Reset Value
0045h	Timer B5 Interrupt Control Register	TB5IC	XXXX X000b
0046h	Timer B4 Interrupt Control Register	TB4IC	XXXX X000b
0047h	Timer B3 Interrupt Control Register	TB3IC	XXXX X000b
005Ah	Timer B0 Interrupt Control Register	TB0IC	XXXX X000b
005Bh	Timer B1 Interrupt Control Register	TB1IC	XXXX X000b
005Ch	Timer B2 Interrupt Control Register	TB2IC	XXXX X000b
0206h	Interrupt Source Select Register 2	IFSR2A	00h

Timers B3 and B4 share interrupt vectors and interrupt control registers with other peripheral functions. When using the timer B3 interrupt, set the IFSR26 bit in the IFSR2A register to 0 (timer B3). When using the timer B4 interrupt, set the IFSR27 bit in the IFSR2A register to 0 (timer B4).

17.5 Notes on Timer B

17.5.1 Common Notes on Multiple Modes

17.5.1.1 Register Setting

The timer is stopped after reset. Set the mode, count source, etc., using registers TBiMR, TBCS0 to TBCS3, TBi, PCLKR, PPWFS1, and PPWFS2 before setting the TBiS bit in the TABSR or TBSR register to 1 (count started) (i = 0 to 5).
Always make sure registers TBiMR, TBCS0 to TBCS3, PCLKR, PPWFS1, and PPWFS2 are modified while the TBiS bit is 0 (count stopped), regardless of whether after reset or not.

17.5.2 Timer B (Timer Mode)

17.5.2.1 Read from Timer

While counting, the counter value can be read at any time by reading the TBi register. However, FFFFh is read while reloading. When the counter is read before it starts counting and after a value is set in the TBi register while not counting, the set value is read.

17.5.3 Timer B (Event Counter Mode)

17.5.3.1 Read from Timer

While counting, the counter value can be read at any time by reading the TBi register. However, FFFFh is read while reloading. When the counter is read before it starts counting and after a value is set in the TBi register while not counting, the set value is read.

17.5.3.2 Event

When the TCK1 bit in the TBiMR register is 1, an event occurs when an interrupt request of the selected timer is generated. An event or trigger occurs while an interrupt is disabled because an interrupt request signal is generated regardless of the I flag, IPL, or interrupt control registers.
When the timer selected by the TCK1 bit uses pulse-period measurement mode or pulse-width measurement mode, an interrupt request is generated at an active edge of the measurement pulse.

17.5.4 Timer B (Pulse Period/Pulse Width Measurement Modes)

17.5.4.1 The MR3 Bit in the TBiMR Register

To clear the MR3 bit to 0 by writing to the TBiMR register while the TBiS bit is 1 (count started), be sure to set the same value as previously set to bits TMOD0, TMOD1, MR0, MR1, TCK0, and TCK1, and set bit 4 to 0.

17.5.4.2 Interrupts

The IR bit in the TBiIC register becomes 1 (interrupt requested) when an active edge of a measurement pulse is input or timer Bi overflows (i = 0 to 5). The source of an interrupt request can be determined using the MR3 bit in the TBiMR register within the interrupt routine.
Use the IR bit in the TBiIC register to detect overflows only. Use the MR3 bit only to determine the interrupt source.

17.5.4.3 Event or Trigger

When timer Bi in pulse-period measurement mode or pulse-width measurement mode is used as an event or trigger for timer A or timer B other than timer Bi, an event or trigger occurs at both the overflow and active edge of the measurement pulse.

17.5.4.4 Operations between Count Start and the First Measurement

When a count is started and the first active edge is input, an undefined value is transferred to the reload register. At this time, a timer Bi interrupt request is not generated.

The value of the counter is undefined after reset. If a count is started in this state, the MR3 bit may become 1 and a timer Bi interrupt request may be generated after the count starts before an active edge is input. When a value is set in the TBi register while the TBiS bit is 0 (count stopped), the same value is written to the counter.

17.5.4.5 Pulse Period Measurement Mode

When active edge and overflow are generated simultaneously, input is not recognized at the active edge because an interrupt request is generated only once. Use this mode so an overflow is not generated, or use pulse width measurement.

17.5.4.6 Pulse Width Measurement Mode

In pulse width measurement, pulse widths are measured successively. Check whether the measurement result is a high-level width or a low-level width in the user program.
When an interrupt request is generated, read the TBiIN pin level inside the interrupt routine, and check whether it is the edge of an input pulse or overflow. The TBiIN pin level can be read from bits in the register of ports sharing a pin.

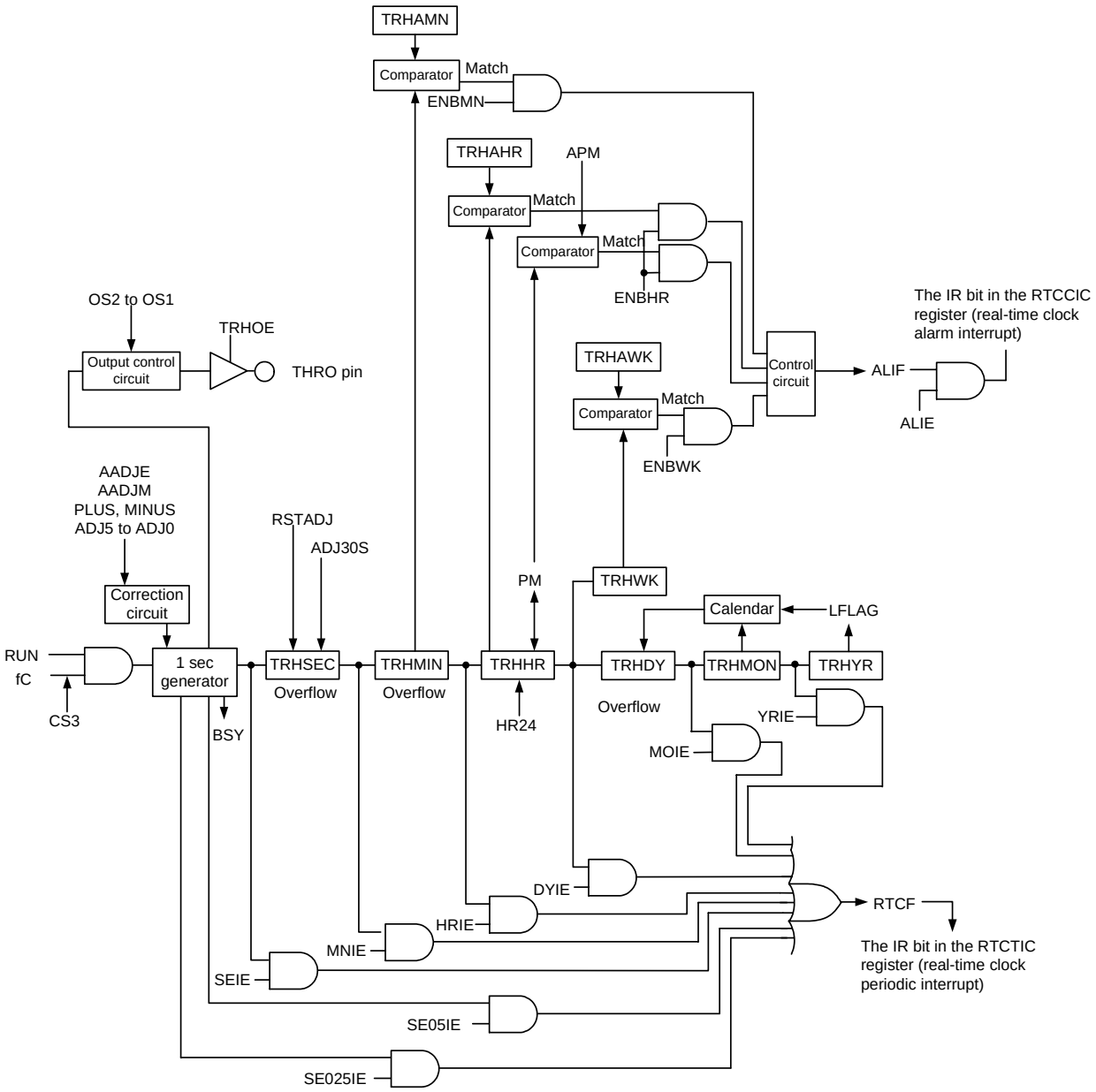
18. Real-Time Clock

18.1 Introduction

The real-time clock is a timer RH. The real-time clock generates a one-second signal from a count source and counts seconds, minutes, hours, a.m./p.m., a date, a day of the week, a week, a month, and a year. Leap years from 2000 to 2099 are automatically set. It also detects matches with specified minutes, hours, and a day of the week.
Table 18.1 lists Real-Time Clock Specifications, Figure 18.1 shows a Real-Time Clock Block Diagram, and Table 18.2 lists the I/O Port.

Table 18.1 Real-Time Clock Specifications

Item	Specification
Count source	fC
Count operation	Increment
Count start condition	1 (count started) is written to the RUN bit in the TRHCR register.
Count stop condition	0 (count stopped) is written to the RUN bit in the TRHCR register.
Interrupt request generation timing	Periodic interrupt Select one of the following: • 250 ms cycles • 500 ms cycles • Update second data • Update minute data • Update hour data • Update date and day-of-the-week data • Update month data • Update year data Alarm interrupt When time data and alarm data match.
TRHO pin function	Programmable I/O port or clock output
Read from timer	When reading the TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, or TRHYR register, the counter value can be read. The values in registers TRHSEC, TRHMIN, TRHHR, TRHDY, TRHMON, and TRHYR are represented by the BCD code.
Write to timer	When the RUN bit in the TRHCR register is 0 (count stopped), registers TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, and TRHYR can be written to. The values written to registers TRHSEC, TRHMIN, TRHHR, TRHDAY, TRHMON, and TRHYR are represented by the BCD code.
Selectable functions	• 12-/24-hour mode switch function • Alarm function Either of following is detected: - Combination of a specified day of the week, minute, and hour - Combination of a specified hour and minute - A specified minute • Second adjustment function • Time error correction function Automatic correction function or correction by software • Clock output



RUN, HR24, PM, LFLAG, TRHOE, AADJE: Bits in the TRHCR register
YRIE, MOIE, DYIE, HRIE, MNIE, SEIE, SE05IE, SE025IE: Bits in the TRHIER register
AADJM, OS2 to OS1, CS3: Bits in the TRHCSR register
BSY: Bit in the TRHSEC register
PLUS, MINUS, ADJ5 to ADJ0: Bits in the TRHADJ register
RTSADJ, ADJ30S, ALIE, RTCF, ALIF: Bits in the TRHIFR register
ENBMN: Bit in the TRHAMN register
ENBHR, APM: Bits in the TRHAHR register
ENBWK: Bit in the TRHAWK register

TRHYR: Bits YR13 to YR10 and YR03 to YR00 in the TRHYR register
TRHMON: Bits MO10 and MO03 to MO00 in the TRHMON register
TRHDY: Bits DY11 to DY10 and DY03 to DY00 in the TRHDY register
TRHWK: Bits WK2 to WK0 in the TRHWK register
TRHHR: Bits HR11 to HR10 and HR03 to HR00 in the TRHHR register
TRHMIN: Bits MN12 to MN10 and MN03 to MN00 in the TRHMIN register
TRHSEC: Bits SC12 to SC10 and SC03 to SC00 in the TRHSEC register

TRHAWK: Bits AWK02 to AWK00 in the TRHAWK register
TRHAHR: Bits AHR5 to AHR4 and AHR3 to AHR0 in the TRHAHR register
TRHAMN: Bits AMN6 to AMN4 and AMN3 to AMN0 in the TRHAMN register

Figure 18.1 Real-Time Clock Block Diagram

Table 18.2 I/O Port

Pin Name	I/O	Function
TRHO	Output	Alarm output

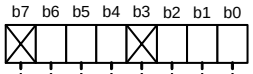
18.2 Registers

Table 18.3 Register Structure

Address	Register	Symbol	Reset Value
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
0340h	Second Data Register	TRHSEC	0000 0000b
0341h	Minute Data Register	TRHMIN	0000 0000b
0342h	Hour Data Register	TRHHR	0000 0000b
0343h	Day-of-the-Week Data Register	TRHWK	0000 0000b
0344h	Date Data Register	TRHDY	0000 0001b
0345h	Month Data Register	TRHMON	0000 0001b
0346h	Year Data Register	TRHYR	0000 0000b
0347h	Timer RH Control Register	TRHCR	0000 0100b
0348h	Timer RH Count Source Select Register	TRHCSR	0000 1000b
0349h	Clock Error Correction Register	TRHADJ	0000 0000b
034Ah	Timer RH Interrupt Flag Register	TRHIFR	XXX0 0000b
034Bh	Timer RH Interrupt Enable Register	TRHIER	0000 0000b
034Ch	Alarm Minute Register	TRHAMN	0000 0000b
034Dh	Alarm Hour Register	TRHAHR	0000 0000b
034Eh	Alarm Day-of-the-Week Register	TRHAWK	0XXX X000b
034Fh	Timer RH Protect Register	TRHPRC	00XX XXXXb

18.2.1 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1



Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I^2C peripheral clock stop bit (multi-master I^2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

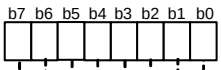
Set the PRC0 bit in the PRCR register to 1 (write enabled) before rewriting this register.

PCKSTP10 (Real-time clock peripheral clock stop bit) (b0)

Set this bit to 0 (f1 provide enabled) when using the real-time clock. Also, set the PM25 bit in the PM2 register to 1 (fC provide enabled).

18.2.2 Second Data Register (TRHSEC)

Second Data Register



Symbol: TRHSEC | Address: 0340h | After Reset: 0000 0000b

Bit Symbol	Bit Name	Function	Setting Range	RW
SC00	First digit of second count bit	Count 0 to 9 every second. When the digit increments, 1 is added to the second digit of second.	0 to 9	RW
SC01				RW
SC02				RW
SC03				RW
SC10	Second digit of second count bit	When counting 0 to 5, 60 seconds are counted.	0 to 5	RW
SC11				RW
SC12				RW
BSY	Timer RH busy flag	This bit is 1 while the TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, or TRHYR register is updated.		RO

Access the register in 8-bit units.
Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this register.

SC03 to SC00 (First digit of second count bit) (b3 to b0)
SC12 to SC10 (Second digit of second count bit) (b6 to b4)

Set values between 00 and 59 by the BCD code.
Write to these bits when the RUN bit in the TRHCR register is 0 (count stopped).
Do not access these bits:

- When the BSY bit is 1 (the data is being updated).
- Until three fC cycles have elapsed after rewriting the RUN bit in the TRHCR register.

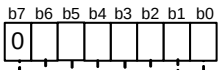
BSY (Timer RH busy flag) (b7)

This bit is 1 while data is updated. Read the following bits when this bit is 0 (not while data is updated):

- Bits SC12 to SC10 and SC03 to SC00 in the TRHSEC register
- Bits MN12 to MN10 and MN03 to MN00 in the TRHMIN register
- Bits HR11 to HR10 and HR03 to HR00 in the TRHHR register
- The PM bit in the TRHCR register
- Bits WK2 to WK0 in the TRHWK register
- Bits DY11 to DY10 and DY03 to DY00 in the TRHDY register
- Bits MO10 and MO03 to MO00 in the TRHMON register
- Bits YR13 to YR10 and YR03 to YR00 in the TRHYR register

18.2.3 Minute Data Register (TRHMIN)

Minute Data Register



Symbol	Address	After Reset
TRHMIN	0341h	0000 0000b

Bit Symbol	Bit Name	Function	Setting Range	RW
MN00	First digit of minute count bit	Count 0 to 9 every minute. When the digit increments, 1 is added to the second digit of minute.	0 to 9	RW
MN01				RW
MN02				RW
MN03				RW
MN10	Second digit of minute count bit	When counting 0 to 5, 60 minutes are counted.	0 to 5	RW
MN11				RW
MN12				RW
$\overline{\text{(b7)}}$	Reserved bit	Set to 0.		RW

Access the register in 8-bit units.
Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this register.

MN03 to MN00 (First digit of minute count bit) (b3 to b0)
MN12 to MN10 (Second digit of minute count bit) (b6 to b4)

Set values between 00 and 59 by the BCD code.
When the digit increments from the TRHSEC register, 1 is added.
Write to these bits when the RUN bit in the TRHCR register is 0 (count stopped).
Read these bits when the BSY bit in the TRHSEC is 0 (not while data is updated).

18.2.4 Hour Data Register (TRHHR)

Hour Data Register

b7	b6	b5	b4	b3	b2	b1	b0
X	X						

Symbol	Address	After Reset
TRHHR	0342h	0000 0000b

Bit Symbol	Bit Name	Function	Setting Range	RW
HR00	First digit of hour count bit	Count 0 to 9 every hour. When the digit increments, 1 is added to the second digit of hour.	0 to 9	RW
HR01				RW
HR02				RW
HR03				RW
HR10	Second digit of hour count bit	Count 0 to 1 when the HR24 bit is 0 (12-hour mode). Count 0 to 2 when the HR24 bit is 1 (24-hour mode).	0 to 2	RW
HR11				RW
— (b7-b6)	No register bits. If necessary, set to 0. When read, the read value is 0.			—

Access the register in 8-bit units.
Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this register.

HR03 to HR00 (First digit of hour count bit) (b3 to b0)
HR11 to HR10 (Second digit of hour count bit) (b5 to b4)

When the HR24 bit in the TRHCR register is 0 (12-hour mode), set values between 00 and 11 by the BCD code. When the HR24 bit is 1 (24-hour mode), set a value between 00 and 23 by the BCD code.
When the digit increments from the TRHMIN register, 1 is added.
Write to these bits when the RUN bit in the TRHCR register is 0 (count stopped).
Read these bits when the BSY bit in the TRHSEC register is 0 (not while data is updated).

18.2.5 Day-of-the-Week Data Register (TRHWK)

Day-of-the-Week Data Register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
0	0	0	0	0				TRHWK	0343h	0000 0000b

Bit Symbol	Bit Name	Function	RW
WK0	Day-of-the-week count bit	b2 b1 b0 0 0 0: Sunday 0 0 1: Monday 0 1 0: Tuesday 0 1 1: Wednesday 1 0 0: Thursday 1 0 1: Friday 1 1 0: Saturday 1 1 1: Do not set	RW
WK1			RW
WK2			RW
— (b7-b3)	Reserved bits	Set to 0.	RW

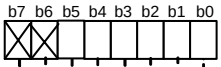
Access the register in 8-bit units.
Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this register.

WK2 to WK0 (Day-of-the-week count bit) (b2 to b0)

A week is counted by counting from 000b (Sunday) to 110b (Saturday) repeatedly. Do not set these bits to 111b.
When the digit increments from the TRHHR register, 1 is added.
Write to these bits when the RUN bit in the TRHCR register is 0 (count stopped).
Read these bits when the BSY bit in the TRHSEC register is 0 (not while data is updated).

18.2.6 Date Data Register (TRHDY)

Date Data Register



Symbol	Address	After Reset
TRHDY	0344h	0000 0001b

Bit Symbol	Bit Name	Function	Setting Range	RW
DY00	First digit of date count bit	Count 0 to 9 every day. When the digit increments, 1 is added to the second digit of date.	0 to 9	RW
DY01				RW
DY02				RW
DY03				RW
DY10	Second digit of date count bit	Count 0 to 3.	0 to 3	RW
DY11				RW
— (b7-b6)	No register bits. If necessary, set to 0. When read, the read value is 0.			—

Access the register in 8-bit units.
Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this register.

DY03 to DY00 (First digit of date count bit) (b3 to b0)
DY11 to DY10 (Second digit of date count bit) (b5 to b4)

Set values between 01 and 31 by the BCD code.
The digit increments from the TRHHR register, 1 is added.
The number of days (28 to 31) in each month including February in a leap year are counted from 2000 to 2099.
Write to these bits when the RUN bit in the TRHDY register is 0 (count stopped).
Read these bits when the BSY bit in the TRHSEC is 0 (not while data is updated).

18.2.7 Month Data Register (TRHMON)

Month Data Register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TRHMON	0345h	0000 0001b

Bit Symbol	Bit Name	Function	Setting Range	RW
MO00	First digit of month count bit	Count 0 to 9 every month. When the digit moves up, 1 is added to the second digit of month.	0 to 9	RW
MO01				RW
MO02				RW
MO03				RW
MO10	Second digit of month count bit	Count 0 to 1.	0 to 1	RW
— (b7-b5)	No register bits. If necessary, set to 0. When read, the read value is 0.			—

Access the register in 8-bit units.
Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this register.

MO03 to MO00 (First digit of month count bit) (b3 to b0)
MO10 (Second digit of month count bit) (b4)

Set values between 01 and 12 by the BCD code.
When the digit increments from the TRHDY register, 1 is added.
Write to these bits when the RUN bit in the TRHCR register is 0 (count stopped).
Read these bits when the BSY bit in the TRHSEC is 0 (not while data is updated).

18.2.8 Year Data Register (TRHYR)

Year Data Register

b7 b6 b5 b4 b3 b2 b1 b0	Symbol	Address	After Reset
	TRHYR	0346h	0000 0000h

Bit Symbol	Bit Name	Function	Setting Range	RW
YR00	First digit of year count bit	Count 0 to 9 every year. When the digit moves up, 1 is added to the second digit of year.	0 to 9	RW
YR01				RW
YR02				RW
YR03				RW
YR10	Second digit of year count bit	Count 0 to 9.	0 to 9	RW
YR11				RW
YR12				RW
YR13				RW

Access the register in 8-bit units.
Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this register.

YR03 to YR00 (First digit of year count bit) (b3 to b0)
YR13 to YR10 (Second digit of year count bit) (b7 to b4)

Set values between 00 and 99 by the BCD code. Fourth digit and third digit of year are fixed to 20.
When the digit increments from the TRHMON register, 1 is added.
Write to these bits when the RUN bit in the TRHCR register are 0 (count stopped).
Read these bits when the BSY bit in the TRHSEC is 0 (not while data is updated).

18.2.9 Timer RH Control Register (TRHCR)

Timer RH Control Register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
			0	0				TRHCR	0347h	0000 0100b

Bit Symbol	Bit Name	Function	RW
AADJE	Automatic correction function enable bit	0: Automatic correction function disabled (correction function by software enabled) 1: Automatic correction function enabled (correction function by software disabled)	RW
TRHOE	Timer RH output enable bit	0: TRHO output disabled 1: TRHO output enabled	RW
LFLAG	Leap year flag	0: Common year 1: Leap year	RO
— (b3)	Reserved bit	Set to 0.	RW
— (b4)	Reserved bit	When read, the read value is 0.	RO
PM	a.m./p.m. bit	0: a.m. 1: p.m.	RW
HR24	Operating mode bit	0: 12-hour mode 1: 24-hour mode	RW
RUN	Timer RH operation start bit	0: Timer RH count stopped 1: Timer RH count started	RW

Access the register in 8-bit units.

TRHOE (Timer RH output enable bit) (b1)

Rewrite this bit when the RUN bit is 0 (count stops).

LFLAG (Leap year flag) (b2)

This bit becomes 1 (leap year) when the values of the TRHYR register are 00 or the multiples of four. When this bit is 1, the number of days in February becomes 29.

PM (a.m./p.m. bit) (b5)

Write to this bit when the RUN bit in the TRHCR register is 0 (count stopped). Set the PROTECT bit in the TRHPRC register to 1 (write enabled) before rewriting this bit.
Read this bit when the BSY bit in the TRHSEC register is 0 (not while data is updated).
This bit is enabled when the HR24 bit is 0 (12-hour mode).
This bit changes as follows while counting.

- Becomes 0 when this bit is 1 (p.m.) and the clock increments from 11:59:59 to 00:00:00.
- Becomes 1 when this bit is 0 (a.m.) and the clock increments from 11:59:59 to 00:00:00.

Figure 18.2 shows Definition of Time Representation.

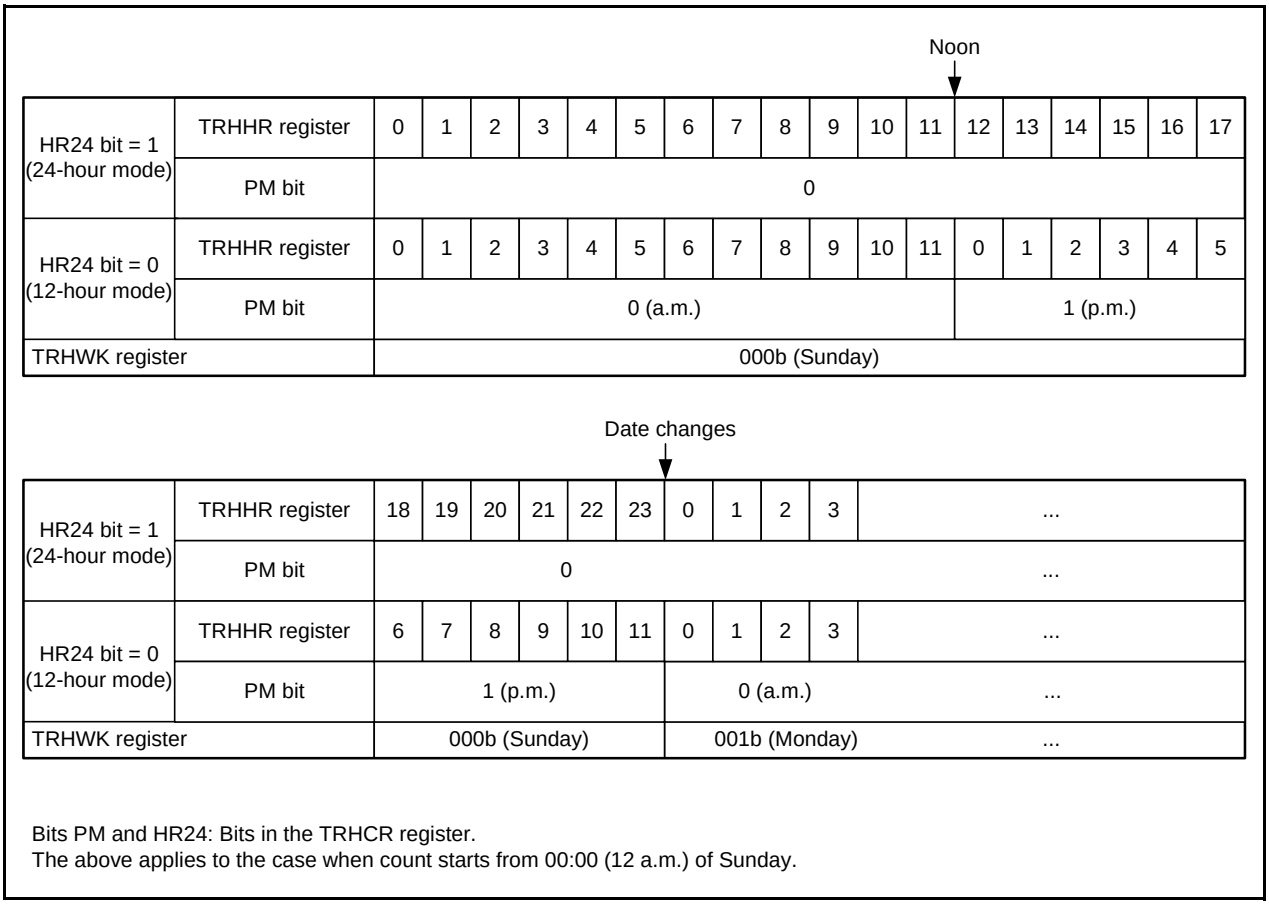


Figure 18.2 Definition of Time Representation

HR24 (Operating mode bit) (b6)

Write to this bit when the RUN bit in the TRHCR register is 0 (counter stopped).

RUN (Timer RH operation start bit) (b7)

When setting the RUN bit to 0 (timer RH count stopped), the internal counter is not reset.

18.2.10 Timer RH Count Source Select Register (TRHCSR)

Timer RH Count Source Select Register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
			1	1	0	0	0	TRHCSR	0348h	0000 1000b

Bit Symbol	Bit Name	Function	RW
— (b2-b0)	Reserved bits	Set to 0.	RW
CS3	Count source select bit	1: fC Set to 1.	RW
— (b4)	Reserved bits	Set to 1.	RW
OS1	Timer RH output select bit	b6 b5 0 0: fC 0 1: 1 Hz (after correction) 1 0: 64 Hz (after correction) 1 1: Do not set	RW
OS2			RW
AADJM	Automatic correction mode select bit	0: Correct every one minute (resolution ±0.5 ppm) 1: Correct every 10 seconds (resolution ±3 ppm)	RW

Access the register in 8-bit units.

CS3 (Count source select bit) (b3)

Set the PM25 bit in the PM2 register to 1 (peripheral clock fC provided). For details of fC, refer to 8. “Clock Generator”.

OS2 to OS1 (Timer RH output select bit) (b6 to b5)

Rewrite these bits when the RUN bit in the TRHCR register is 0 (count stopped).
These bits are enabled when the TRHOE bit in the TRHCR register is 1 (TRHO output enabled). When the TRHOE bit is 0 (TRHO output disabled), bit 4 can be set to 0.

AADJM (Automatic correction mode select bit) (b7)

This bit is enabled when the AADJE bit in the TRHCR register is 1 (automatic correction function enabled).

18.2.11 Clock Error Correction Register (TRHADJ)

Clock Error Correction Register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	After Reset
TRHADJ	0349h	0000 0000b

Bit Symbol	Bit Name	Function	Setting Range	RW
ADJ0	Correction value set bit	Set the correction value	00h to 3Fh (00 to 63)	RW
ADJ1				RW
ADJ2				RW
ADJ3				RW
ADJ4				RW
ADJ5				RW
MINUS	Correction value add/subtract set bit	b7 b6 0 0: Not corrected 0 1: Subtract 1 0: Add 1 1: Do not set		RW
PLUS				RW

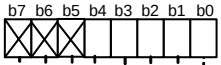
Access the register in 8-bit units.

MINUS and PLUS (Correction value add/subtract set bit) (b7-b6)

This bit adds/subtracts values from the internal counter of the 1-second generator. The value written to bits ADJ5 to ADJ0 is added/subtracted from the internal counter.
To correct a clock that runs fast, set these bits to 01b (subtract). To correct a clock that runs slow, set these bits to 10b (add). Refer to 18.3.4 “Clock Error Correction Function”.

18.2.12 Timer RH Interrupt Flag Register (TRHIFR)

Timer RH Interrupt Flag Register



Symbol	Address	After Reset
TRHIFR	034Ah	XXX0 0000b

Bit Symbol	Bit Name	Function	RW
ALIF	Alarm interrupt flag	0: Interrupt not requested 1: Interrupt requested	RW
RTCF	RTC periodic interrupt flag	0: Interrupt not requested 1: Interrupt requested	RW
ALIE	Alarm interrupt enable bit	0: Alarm interrupt disabled 1: Alarm interrupt enabled	RW
ADJ30S	30 seconds adjust bit	When setting this bit to 1, the values of the TRHSEC register become as follows: 00 when the TRHSEC register values ≤ 29 59 when the TRHSEC register values ≥ 30 When read, the read value is 0.	WO
RSTADJ	Second counter reset adjust bit	When setting this bit to 1, the TRHSEC register becomes 00, and the internal counter is initialized. When read, the read value is 0.	WO
— (b7-b5)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

Access the register in 8-bit units.

ALIF (Alarm interrupt flag) (b0)

Condition to become 0:

- Write 0 after reading this bit. When writing 0 to this bit if the read value is 1, this bit becomes 0.

Condition to become 1:

- Contents of registers TRHAMN, TRHAHR, and TRHAWK match contents of registers TRHMIN, TRHHR, and TRHWK (refer to 18.3.2 “Alarm Function”)

When writing 0 to this bit if the read value is 0, this bit remains unchanged (if this bit changes from 0 to 1 after reading this bit, this bit remains 1 even if writing 0).
Writing 1 has no effect.

RTCF (RTC periodic interrupt flag) (b1)

Condition to become 0:

- Write 0 after reading this bit. When writing 0 to this bit if the read value is 1, this bit becomes 0.

Condition to become 1:

- Interrupt source enabled in the TRHIER register is generated.

When writing 0 to this bit if the read value is 0, this bit remains unchanged (if this bit changes from 0 to 1 after reading this bit, this bit remains 1 even if writing 0).
Writing 1 has no effect.

ADJ30S (30 seconds adjust bit) (b3)

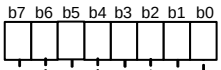
Set the ADJ30S bit to 1 while the BSY bit in the TRHSEC register is 0 (the data is not being updated).

RSTADJ (Second counter reset adjust bit) (b4)

Do not set this bit to 1 when the RTCF bit is 1 (periodic interrupt requested).
Set the RSTADJ bit to 1 while the BSY bit in the TRHSEC register is 0 (the data is not being updated).

18.2.13 Timer RH Interrupt Enable Register (TRHIER)

Timer RH Interrupt Enable Register



Symbol	Address	After Reset
TRHIER	034Bh	0000 0000b

Bit Symbol	Bit Name	Function	RW
SE025IE	Periodic interrupt triggered every 0.25 second enable bit	0: Periodic interrupt triggered every 0.25 second disabled 1: Periodic interrupt triggered every 0.25 second enabled	RW
SE05IE	Periodic interrupt triggered every 0.5 second enable bit	0: Periodic interrupt triggered every 0.5 second disabled 1: Periodic interrupt triggered every 0.5 second enabled	RW
SEIE	Periodic interrupt triggered every second enable bit	0: Periodic interrupt triggered every second disabled 1: Periodic interrupt triggered every second enabled	RW
MNIE	Periodic interrupt triggered every minute enable bit	0: Periodic interrupt triggered every minute disabled 1: Periodic interrupt triggered every minute enabled	RW
HRIE	Periodic interrupt triggered every hour enable bit	0: Periodic interrupt triggered every hour disabled 1: Periodic interrupt triggered every hour enabled	RW
DYIE	Periodic interrupt triggered every day enable bit	0: Periodic interrupt triggered every day disabled 1: Periodic interrupt triggered every day enabled	RW
MOIE	Periodic interrupt triggered every month enable bit	0: Periodic interrupt triggered every month disabled 1: Periodic interrupt triggered every month enabled	RW
YRIE	Periodic interrupt triggered every year enable bit	0: Periodic interrupt triggered every year disabled 1: Periodic interrupt triggered every year enabled	RW

Access the register in 8-bit units.

Write to this register when the RUN bit in the TRHCR register is 0 (counter stops).

An interrupt request can be generated every 0.25 seconds, 0.5 seconds, one second, minute, hour, day, month, or year. To generate an interrupt request, set one of the following bits to 1 (interrupt enabled): SE025IE SE05IE, SEIE, MNIE, HRIE, DYIE, MOIE, and YRIE (be sure to set only one bit to 1). Table 18.4 lists Periodic Interrupt Sources.

Table 18.4 Periodic Interrupt Sources

Factor	Interrupt Source	Interrupt Enable Bit
Periodic interrupt triggered every year	The TRHYR register is updated (one-year period)	YRIE
Periodic interrupt triggered every month	The TRHMON register is updated (one-month period)	MOIE
Periodic interrupt triggered every day	The TRHDY register is updated (one-day period)	DYIE
Periodic interrupt triggered every hour	The TRHHR register is updated (one-hour period)	HRIE
Periodic interrupt triggered every minute	The TRHMIN register is updated (one-minute period)	MNIE
Periodic interrupt triggered every second	The TRHSEC register is updated (one-second period)	SEIE
Periodic interrupt triggered every 500 ms	500 ms cycles	SE05IE
Periodic interrupt triggered every 250 ms	250 ms cycles	SE025IE

When the interrupt is enabled by the above bits, following occurs when the periodic interrupt is generated:

- The RTCF bit in the TRHIFR register becomes 1 (periodic interrupt requested).
- The IR bit in the RTCTIC register becomes 1 (periodic interrupt requested).

18.2.14 Alarm Minute Register (TRHAMN)

Alarm Minute Register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
								TRHAMN	034Ch	0000 0000b

Bit Symbol	Bit Name	Function	Setting Range	RW
AMN0	First digit of minute alarm data bit	Store alarm data	0 to 9	RW
AMN1				RW
AMN2				RW
AMN3				RW
AMN4	Second digit of minute alarm data bit	Store alarm data	0 to 5	RW
AMN5				RW
AMN6				RW
ENBMN	Minute alarm enable bit	0: Minute alarm disabled (not compared with the TRHMIN register) 1: Minute alarm enabled (compared with the TRHMIN register)		RW

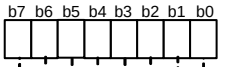
Access the register in 8-bit units.
Write to the register when the BSY bit in the TRHSEC register is 0 (not while data is updated).

AMN3 to AMN0 (First digit of minute alarm data bit) (b3 to b0)
AMN6 to AMN4 (Second digit of minute alarm data bit) (b6 to b4)

Set values between 00 and 59 by the BCD code.

18.2.15 Alarm Hour Register (TRHAHR)

Alarm Hour Register



Symbol	Address	After Reset
TRHAHR	034Dh	0000 0000b

Bit Symbol	Bit Name	Function	Setting Range	RW
AHR0	First digit of hour alarm data bit	Store alarm data	0 to 9	RW
AHR1				RW
AHR2				RW
AHR3				RW
AHR4	Second digit of hour alarm data bit	Store alarm data	0 to 2	RW
AHR5				RW
APM	a.m./p.m. alarm data bit	0: a.m. 1: p.m.		RW
ENBHR	Hour alarm enable bit	0: Hour alarm disabled (not compared with the TRHHR register) 1: Hour alarm enabled (compared with the TRHHR register)		RW

Access the register in 8-bit units.
Write to the register when the BSY bit in the TRHSEC register is 0 (not while data is updated).

AHR3 to AHR0 (First digit of hour alarm data bit) (b3 to b0)
AHR5 to AHR4 (Second digit of hour alarm data bit) (b5 to b4)

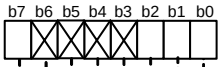
When the HR24 bit in the TRHCR register is 0 (12-hour mode), set values between 00 and 11 by the BCD code. When the HR24 bit is 1 (24-hour mode), set values between 00 and 23 by the BCD code.

APM (a.m./p.m. alarm data bit) (b6)

This bit is disabled when the HR24 bit in the TRHCR register is 1 (24-hour mode).

18.2.16 Alarm Day-of-the-Week Register (TRHAWK)

Alarm Day-of-the-Week Register



Symbol	Address	After Reset
TRHAWK	034Eh	0XXX X000b

Bit Symbol	Bit Name	Function	RW
AWK0	Day-of-the-week alarm data bit	b2 b1 b0 0 0 0: Sunday 0 0 1: Monday 0 1 0: Tuesday 0 1 1: Wednesday 1 0 0: Thursday 1 0 1: Friday 1 1 0: Saturday 1 1 1: Do not set	RW
AWK1			RW
AWK2			RW
— (b6-b3)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—
ENBWK	Day-of-the-week alarm enable bit	0: Day-of-the-week alarm disabled (not compared with the TRHWK register) 1: Day-of-the-week alarm enabled (compared with the TRHWK register)	RW

Access the register in 8-bit units.
Write to the register when the BSY bit in the TRHSEC register is 0 (not while data is updated).

AWK2 to AWK0 (Day-of-the-week alarm data bit) (b2 to b0)

Set 000b (Sunday) to 110b (Saturday).

18.2.17 Timer RH Protect Register (TRHPRC)

Timer RH Protect Register

b7	b6	b5	b4	b3	b2	b1	b0
	0						

Symbol	Address	After Reset
TRHPRC	034Fh	00XX XXXXb

Bit Symbol	Bit Name	Function	RW
— (b5-b0)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—
— (b6)	Reserved bit	Set to 0.	RW
PROTECT	Time data protect bit	Write to the time data register is enabled/ disabled 0: Write disabled 1: Write enabled	RW

Access the register in 8-bit units.

PROTECT (Timer data protect bit) (b7)

Following registers and bits can be changed when this bit is 1 (write enabled):
Registers TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, and TRHYR
The PM bit in the TRHCR register

When writing 1 to this bit by a program, this bit stays 1. Change the registers protected by this bit as follows:

(1) Write 1 to this register
(2) Write a value to the register protected by this bit
(3) Write 0 (write disabled) to this bit

18.3 Operations

18.3.1 Basic Operation

The real-time clock generates a one-second signal from the count source selected in the TRHCSR register and counts seconds, minutes, hours, a.m./p.m., a date, a day of the week, a month, and a year. The day and time to start the count can be set using registers TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, TRHYR and the PM bit in the TRHCR register. Current time and day are read from registers TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, TRHYR and the PM bit in the TRHCR register. However, do not read these registers when the BSY bit in the TRHSEC register is 1 (while data is updated).

An interrupt request can be generated every 0.25 seconds, 0.5 seconds, one second, minute, hour, day, month, or year. When using a periodic interrupt, enable one of the interrupts by the TRHIER register. When the periodic interrupt is generated, the RTCF bit in the TRHIFR register and the IR bit in the RTCTIC register become 1 (interrupt requested).

Figure 18.3 shows Real-Time Clock Basic Operating Example, Figure 18.4 shows Time and Date Change Procedure.

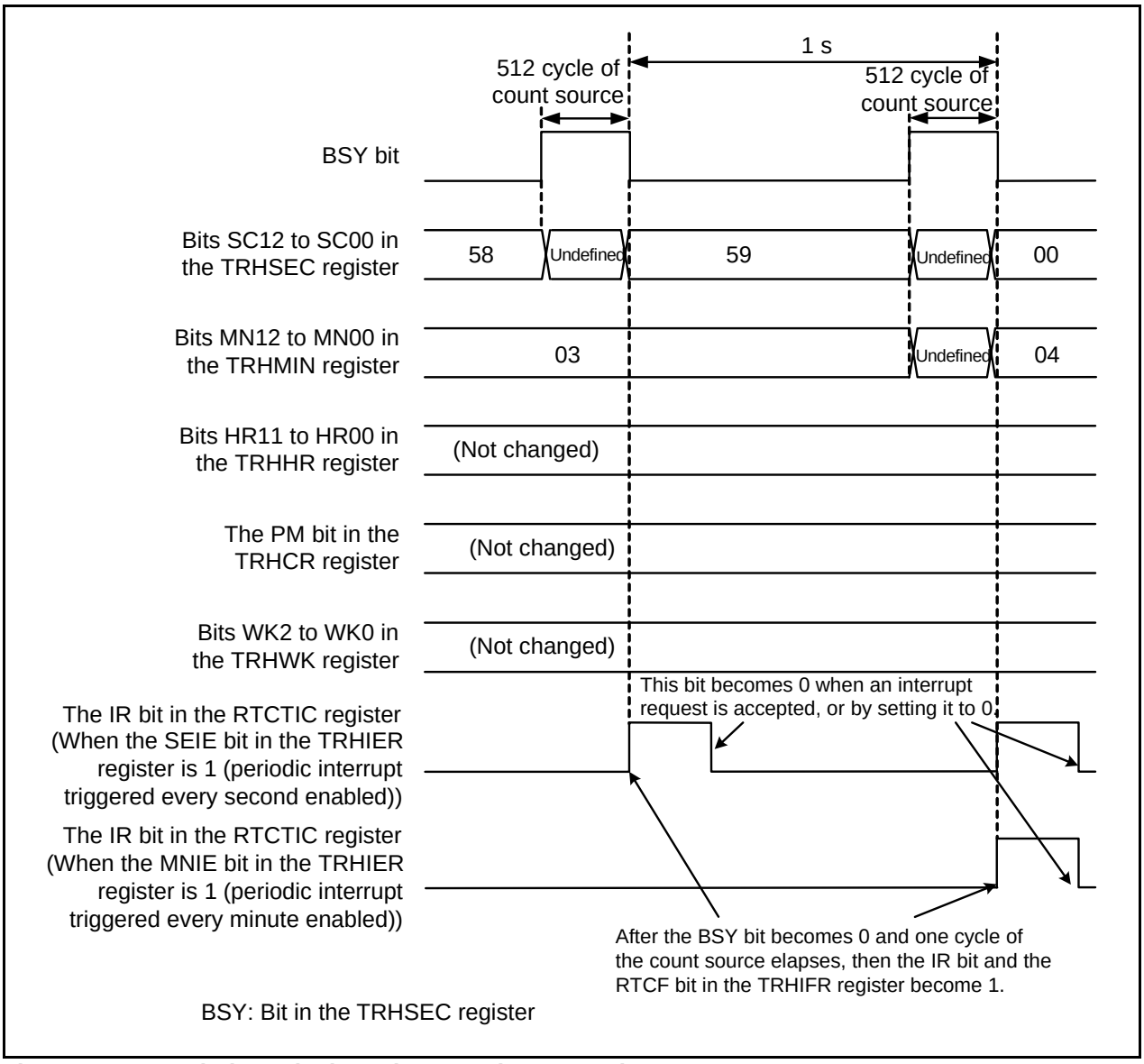


Figure 18.3 Real-Time Clock Basic Operating Example

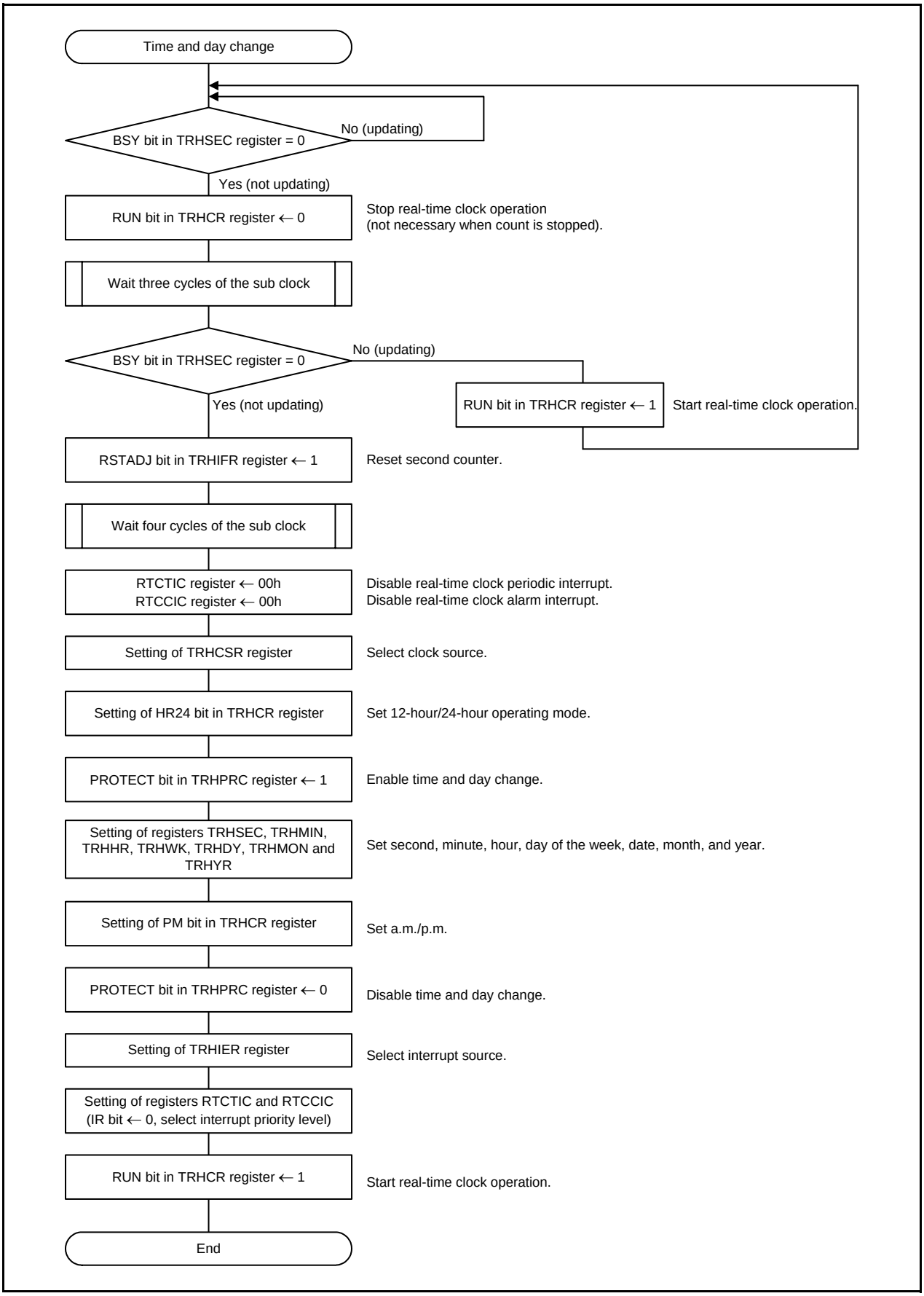


Figure 18.4 Time and Date Change Procedure

18.3.2 Alarm Function

Time data (1) and alarm data (2) are compared, and compare value match is detected.
Either one of the following combinations can be compared. Do not enable the combination other than them. Hour is a.m. or p.m..

- Day of the week, hour, and minute
- Hour and minute
- Minute only

When the comparison result matches, the following occurs:

- The ALIF bit in the TRHIFR register becomes 1 (alarm interrupt requested).
- When the ALIE bit in the TRHIFR register is 1 (alarm interrupt enabled), the IR bit in the RTCCIC register becomes 1 (alarm interrupt requested).

Notes:

1. Bits for time data are as follows:
   Bits MN12 to MN10 and MN03 to MN00 in the TRHMIN register
   Bits HR11 to HR10 and HR03 to HR00 in the TRHHR register
   The PM bit in the TRHCR register
   Bits WK2 to WK0 in the TRHWK register
2. Bits for alarm data are as follows:
   Bits AMN6 to AMN4 and AMN3 to AMN0 in the TRHAMN register
   Bits AHR5 to AHR4 and AHR3 to AHR0 in the TRHAHR register
   The APM bit in the TRHAHR register
   Bits AWK2 to AWK0 in the TRHAWK register

Figure 18.5 shows Alarm Time Setting Procedure.

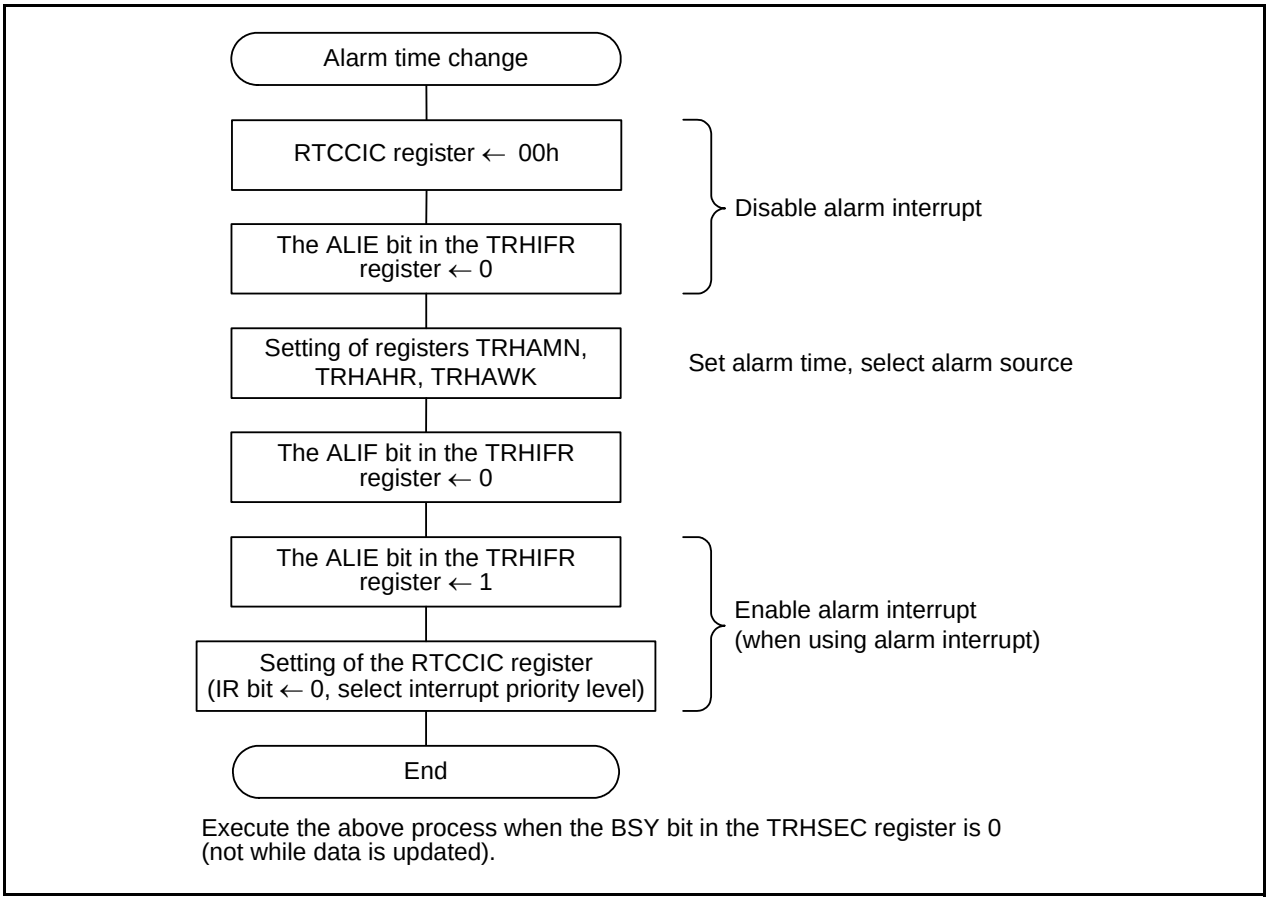


Figure 18.5 Alarm Time Setting Procedure

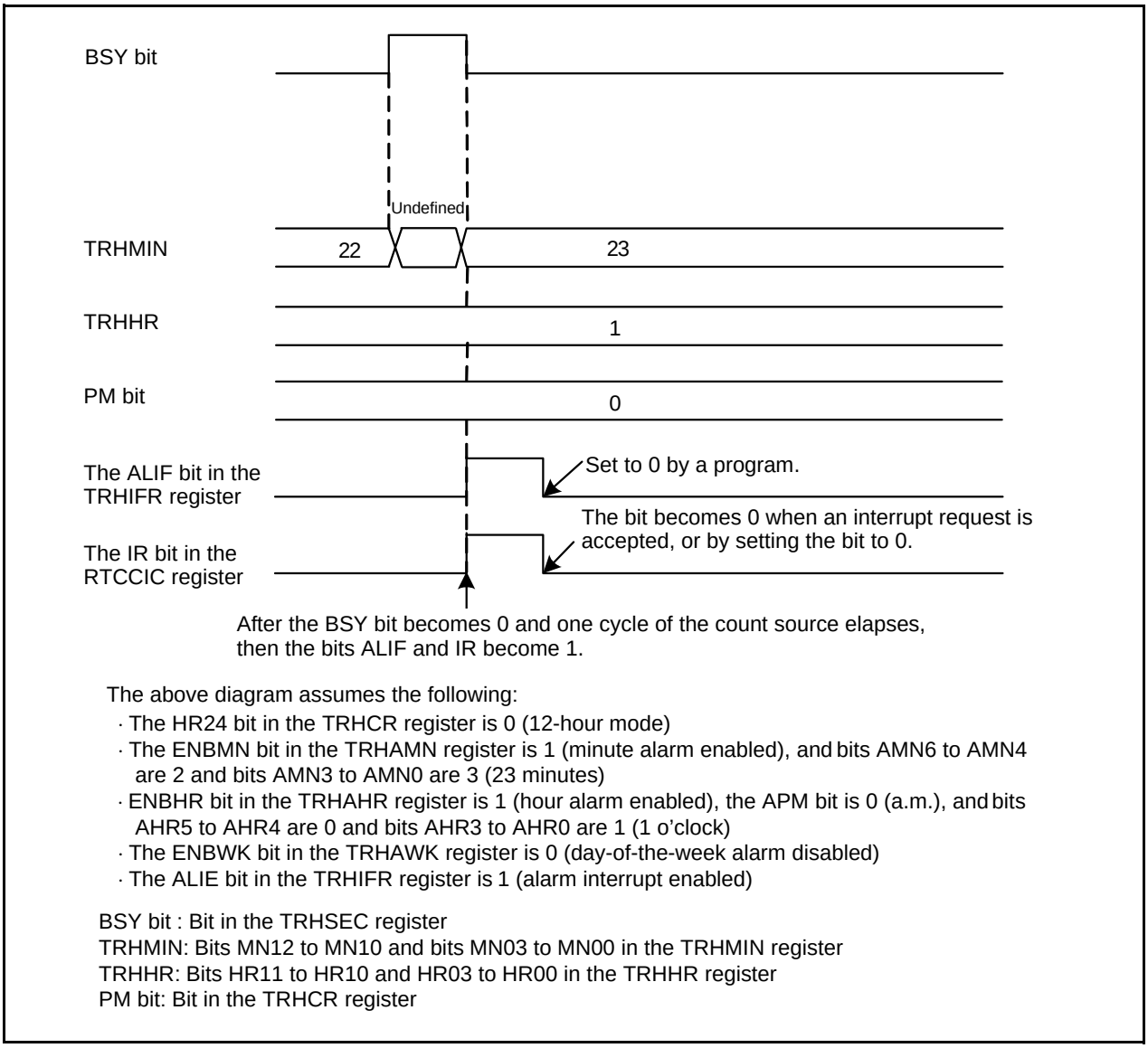


Figure 18.6 Alarm Function

18.3.3 Second Adjustment Function

The TRHSEC register can be adjusted by bits RSTADJ and ADJ30S in the TRHIFR register.
After setting the ADJ30 bit or RSTADJ bit to 1, allow four fC cycles to elapse before accessing the TRHSEC register.

When writing 1 to the RSTADJ bit, the TRHSEC register becomes 00 after a minimum of 2 to 3 cycles of the count source at a minimum, and an internal counter of one second generation circuit is simultaneously initialized.

When writing 1 to the ADJ30S bit, the TRHSEC register value becomes as follows after a minimum of 2 to 3 cycles of the count source at a minimum:

- 00 when the TRHSEC register value ≤ 29
- 59 when the TRHSEC register value ≥ 30

When writing 1 to the ADJ30S bit, the internal counter is not initialized.

18.3.4 Clock Error Correction Function

This function corrects a frequency error of fC. As a basic operation, the internal circuit of the one second generation circuit counts 32.768 kHz 32768 times. When fC is larger or smaller than 32.768 kHz, it can be corrected by increasing or decreasing the number of count.
Select automatic correction or correction by software by the AADJE bit in the TRHCR register.

18.3.4.1 Automatic Correction Function

When the AADJE bit in the TRHCR register is 1, an automatic correction function is enabled.
Select a correction timing by the AADJM bit in the TRHCSR register. Set a correction value and correction content (add/subtract) to the TRHADJ register. The correction value is automatically added or subtracted at the selected correction timing. Examples are as follows:

Ex. 1) fC = 32769 Hz

Error

$$\frac{32769 - 32768}{32768} \times 10^6 = 30.5 \text{ ppm}$$

How to correct
At 32769 Hz, the counter must count 32768 + 1 times to make 1 second. For one minute, the counter must count an additional 60 times. Since the internal counter of the 1-second generator increments the count, subtract 60 from the internal counter every minute to extend the time until overflow occurs.

Register setting
- The AADJM bit in the TRHCSR register: 0 (correct every one minute)
- Set bits PLUS and MINUS in the TRHADJ register to 01b (subtract).
- Bits ADJ5 to ADJ0 in the TRHADJ register: 60

Ex.2) fC = 32770 Hz

Error

$$\frac{32770 - 32768}{32768} \times 10^6 = 61.0 \text{ ppm}$$

How to correct
At 32770 Hz, the counter must count 32768 + 2 times to make 1 second. Since the counter must count an additional 20 times for 10 seconds, subtract 20 from the internal counter every 10 seconds.

Register setting
- The AADJM bit in the TRHCSR register: 1 (correct every 10 seconds)
- Set bits PLUS and MINUS in the TRHADJ register to 01b (subtract).
- Bits ADJ5 to ADJ0 in the TRHADJ register: 20

18.3.4.2 Correction by Software

When the AADJE bit in the TRHCR register is 0, correction by software is enabled. Write a correction value and correction content (add/subtract) to the TRHADJ register at an arbitrary timing. Correction is performed when the write instruction is executed.

Ex.) fC = 32769 Hz

Error

$$\frac{32769 - 32768}{32768} \times 10^6 = 30.5 \text{ ppm}$$

How to correct
At 32769 Hz, the counter must count 32768 + 1 times to make 1 second. Subtract 1 from the internal counter every second.

Register setting
- Set bits PLUS and MINUS in the TRHADJ register to 01b (subtract).
- Bits ADJ5 to ADJ0 in the TRHADJ register: 01
  Write to the TRHADJ register every one second interrupt.

18.3.4.3 Correction Mode Change Procedure

When changing correction mode, set bits PLUS and MINUS in the TRHADJ register to 00b (not corrected) before changing the AADJE bit in the TRHCR register. When rewriting bits PLUS and MINUS and then rewriting these bits again, regardless of changing correction mode or not, wait one or more cycles of the count source to rewrite these bits.

When switching from correction by software to automatic correction:
(1) Set bits PLUS and MINUS bit in the TRHADJ register to 00b (not corrected).
(2) Set the AADJE bit in the TRHCR register to 1 (automatic correction function enabled).
(3) Select a correction cycle by the AADJM bit in the TRHCSR register.
(4) Set add or subtract to bits PLUS and MINUS in the TRHADJ register and a correction value to bits ADJ5 to ADJ0.

Execute (4) after one or more cycles of the count source has elapsed since (1).

When switching from automatic correction to correction by software
(1) Set bits PLUS and MINUS in the TRHADJ register to 00b (not corrected).
(2) Set the AADJE bit in the TRHCR register to 0 (correction function by software enabled)
(3) Correction is performed when setting add or subtract to bits PLUS and MINUS in the TRHADJ register and a correction value to bits ADJ5 to ADJ0 at an arbitrary timing.

Execute (3) after one or more cycles of the count source has elapsed since (1). After (3), correction is performed every time the TRHADJ register is written to.

18.3.5 Clock Output

When the TRHOE bit in the TRHCR register is 1 (TRHO output enabled), clock is output from the TRHO pin. Select clock by bits OS2 to OS1 in the TRHCSR register. When bits OS2 to OS1 are 01b (1 Hz) or 10b (64 Hz), clock corrected by the clock error correction function is output.

18.4 Interrupts

The real-time clock generates the following two types of interrupt.

- Periodic interrupts triggered every 0.25 seconds, 0.5 seconds, one second, minute, hour, day, month, and year
- Alarm interrupt

See Table 18.4 Periodic Interrupt Sources for details of periodic interrupt sources. In the periodic interrupt, when the RTCF bit in the TRHIFR register changes from 0 to 1 (RTC periodic interrupt requested), the IR bit in the RTCTIC register becomes 1 (interrupt requested). Set the RTCF bit to 0 by an interrupt routine.

In the alarm interrupt, when the ALIE bit in the TRHIFR register is 1 (alarm interrupt enabled) and the ALIF bit in the TRHIFR register changes from 0 to 1 (alarm interrupt requested), the IR bit in the TRCCIC register becomes 1 (interrupt requested). Set the ALIF bit to 0 by the interrupt routine.

Refer to specifications and operating examples in each mode for the interrupt request generating timing. Refer to 13.7 “Interrupt Control” for details of interrupt control. Table 18.5 lists Real-Time Clock Interrupt-Associated Registers.

Table 18.5 Real-Time Clock Interrupt-Associated Registers

Address	Register	Symbol	Reset Value
006Eh	Real-Time Clock Periodic Interrupt Control Register	RTCTIC	XXXX X000b
006Fh	Real-Time Clock Alarm Interrupt Control Register	RTCCIC	XXXX X000b
0205h	Interrupt Source Select Register 3	IFSR3A	00h

The real-time clock shares interrupt vectors and interrupt control registers with other peripheral functions. To use periodic interrupts, set the IFSR35 bit in the IFSR3A register to 1 (real-time clock period). To use alarm interrupts, set the IFSR36 bit in the IFSR3A register to 1 (real-time clock alarm).

18.5 Notes on Real-Time Clock

18.5.1 Starting and Stopping the Count

The real-time clock uses the TSTART bit for instructing the count to start or stop, and the TCSTF bit which indicates count started or stopped. Bits TSTART and TCSTF are in the RTCCR1 register.

The real-time clock starts counting and the TCSTF bit becomes 1 (count started) when the TSTART bit is set to 1 (count started). It takes up to two cycles of the count source until the TCSTF bit becomes 1 after setting the TSTART bit to 1. During this time, do not access registers associated with the real-time clock (1) other than the TCSTF bit.

Similarly, when setting the TSTART bit to 0 (count stopped), the real-time clock stops counting and the TCSTF bit becomes 0 (count stopped). It takes up to three cycles of the count source until the TCSTF bit becomes 0 after setting the TSTART bit to 0. During this time, do not access registers associated with the real-time clock other than the TCSTF bit.

Note:

1. Registers associated with the real-time clock: RTCSEC, RTCMIN, RTCHR, RTCWK, RTCCR1, RTCCR2, RTCCSR, RTCCSEC, RTCCMIN, and RTCCHR.

18.5.2 Register Setting (Time Data, etc.)

Write to the following registers/bits when the RUN bit in the TRHCR register is 0 (count stopped):

- Registers TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, TRHYR, and TRHIER
- Bits TRHOE, HR 24, and PM in the TRHCR register
- Bits OS2 to OS1 in the TRHCSR register

Set the TRHIER register after setting other registers and bits mentioned above (immediately before the real-time clock count starts).

18.5.3 Register Setting (Alarm Data)

Write to the following registers when the BSY bit in the TRHSEC register is 0 (not while data is updated).

- Registers TRHAMN, TRHAHR, and TRHAWK

18.5.4 Time Reading Procedure of Real-Time Clock Mode

In real-time clock mode, read time data bits (1) when the BSY bit in the TRHSEC register is 0 (not while data is updated).

When reading multiple registers, if data is rewritten between reading registers, an errant time will be read. To prevent this, use the procedure below when reading:

- Using an interrupt
  Read necessary contents of time data bits in the real-time clock periodic interrupt routine.
- Monitoring by a program 1
  Monitor the IR bit in the RTCTIC register by a program and read necessary contents of time data bits after the IR bit becomes 1 (periodic interrupt requested).
- Monitoring by a program 2
  Read the time data according to Figure 18.7 “Time Data Reading”.

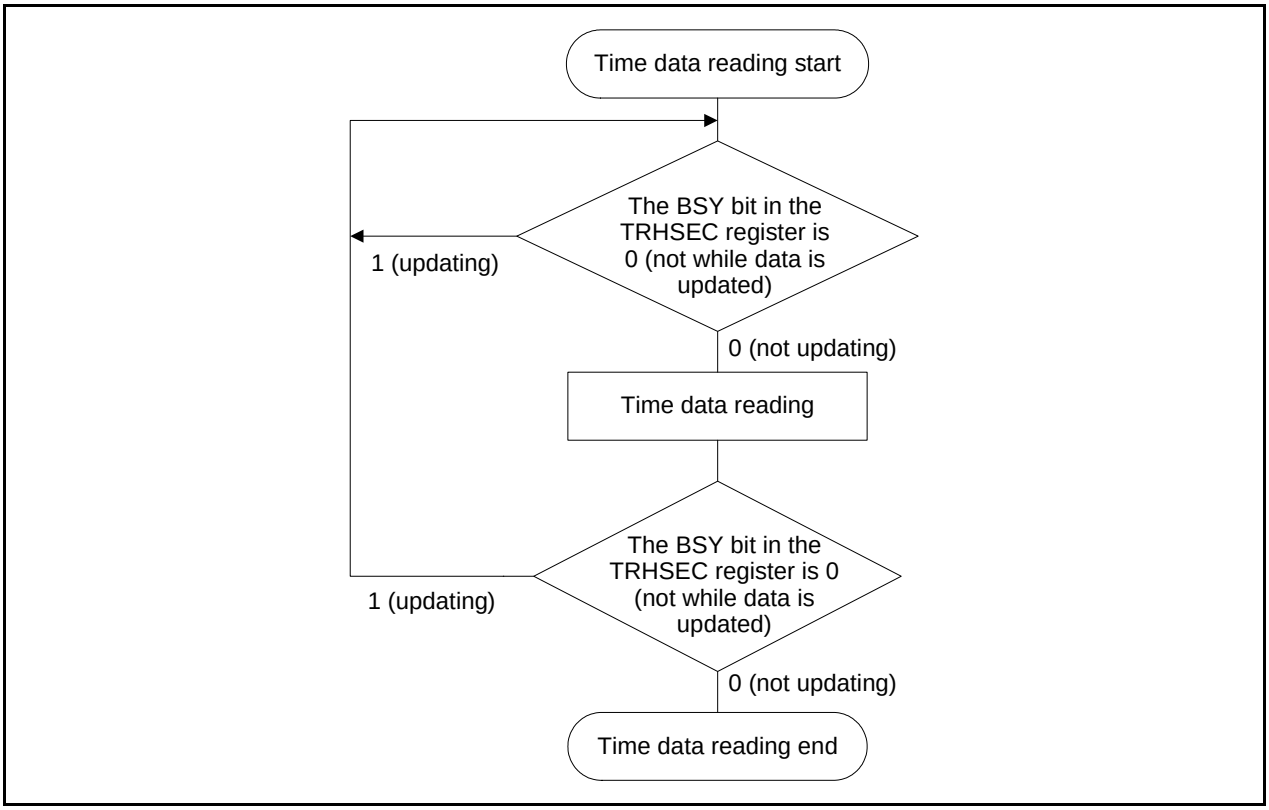


Figure 18.7 Time Data Reading

Also, when reading multiple registers, read them as continuously as possible.

Note:

1. Time data bits are as follows:
   Bits SC12 to SC10 and SC03 to SC00 in the TRHSEC register
   Bits MN12 to MN10 and MN03 to MN00 in the TRHMIN register
   Bits HR11 to HR10 and HR03 to HR00 in the TRHHR register
   Bits WK2 to WK0 in the TRHWK register
   The PM bit in the TRHCR register
   Bits DY11 to DY10 and DY03 to DY00 in the TRHDY register
   Bits MO10 and MO03 to MO00 in the TRHMON register
   Bits YR13 to YR10 and YR03 to YR00 in the TRHYR register

19. Serial Interface UARTi (i = 0 to 2, 5 to 7)

Note

No external pin is provided for UART7 because it is internally connected to the PLC modem.

19.1 Introduction

Each UARTi has a dedicated timer to generate a transmit/receive clock, and operates independently of the others.

Table 19.1 lists UARTi Specifications (i = 0 to 2, 5 to 7), Table 19.2 lists Specification Differences between UART0 to UART2 and UART5 to UART7, Figures 19.1 to 19.3 show UARTi Block Diagram, and Figure 19.4 shows UARTi Transmit/Receive Unit Block Diagram.

Table 19.1 UARTi Specifications (i = 0 to 2, 5 to 7)

Item	Specification
Operational mode	• Clock synchronous serial I/O mode • Clock asynchronous serial I/O mode (UART mode) • Special mode 1 (I^2C mode) • Special mode 2 The simplified I^2C-bus interface is supported. • Special mode 3 (bus collision detection function, IE mode) A 1-byte wave of the UART mode approximates 1-bit of the IEBus. • Special mode 4 (SIM mode) UART2 is available. The SIM interface is supported.

Table 19.2 Specification Differences between UART0 to UART2 and UART5 to UART7

Mode	UART0	UART1	UART2	UART5	UART6	UART7
Clock synchronous serial I/O mode	Available		Available	Available	Available	(1)
Clock asynchronous serial I/O mode (UART mode)	Available		Available	Available	Available	
Special mode 1 (I^2C mode)	Available		Available	Available	Available	
Special mode 2	Available		Available	Available	Available	
Special mode 3 (IE mode)	Available		Available	Available	Available	
Special mode 4 (SIM mode)	Not available		Available	Not available	Not available	

Note:

1. UART7 pins are connected to the internal PLC modem. No external pin is provided.
   Control via DLL software provided from Renesas. Do not control directly via user software.

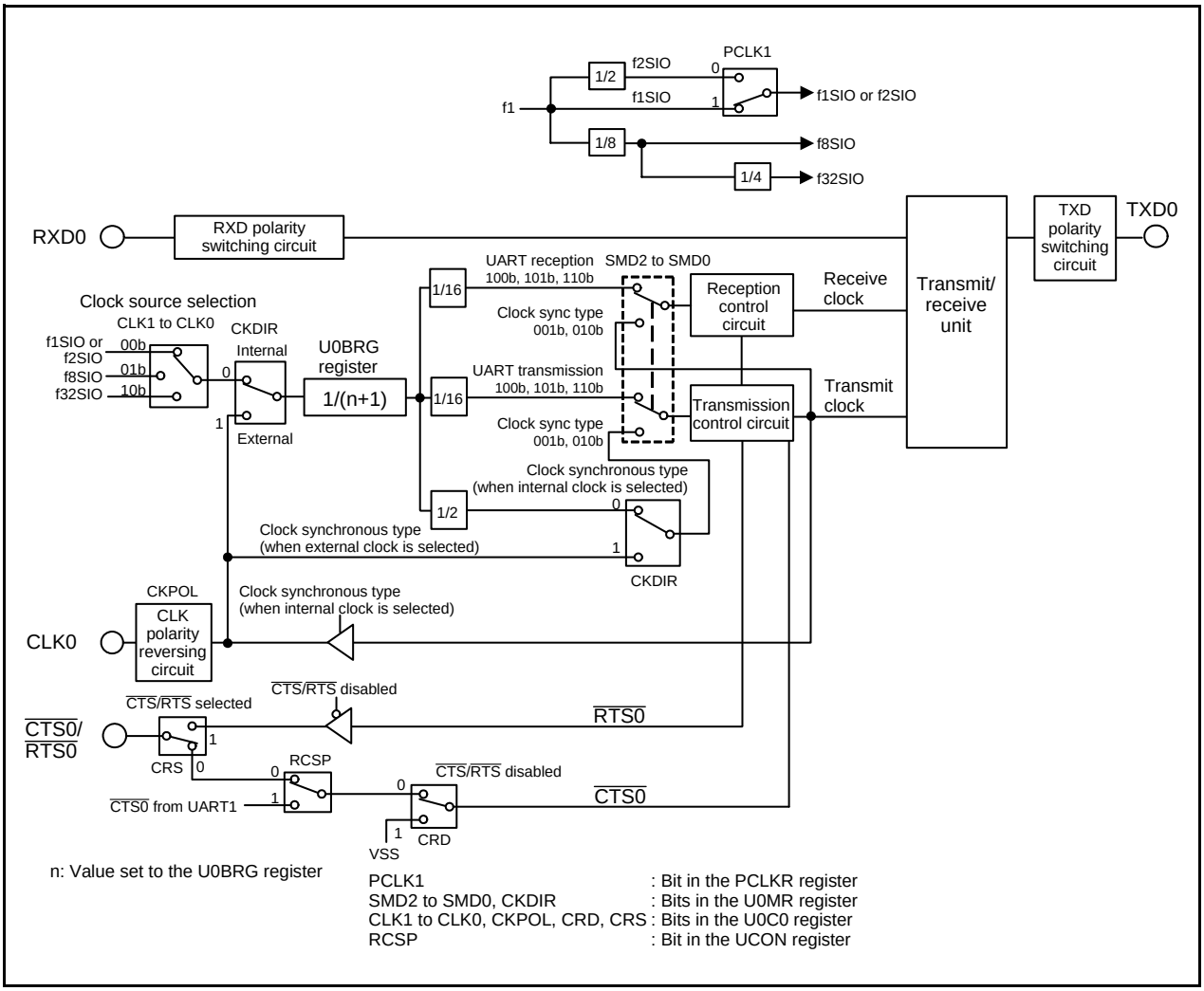


Figure 19.1 UART0 Block Diagram

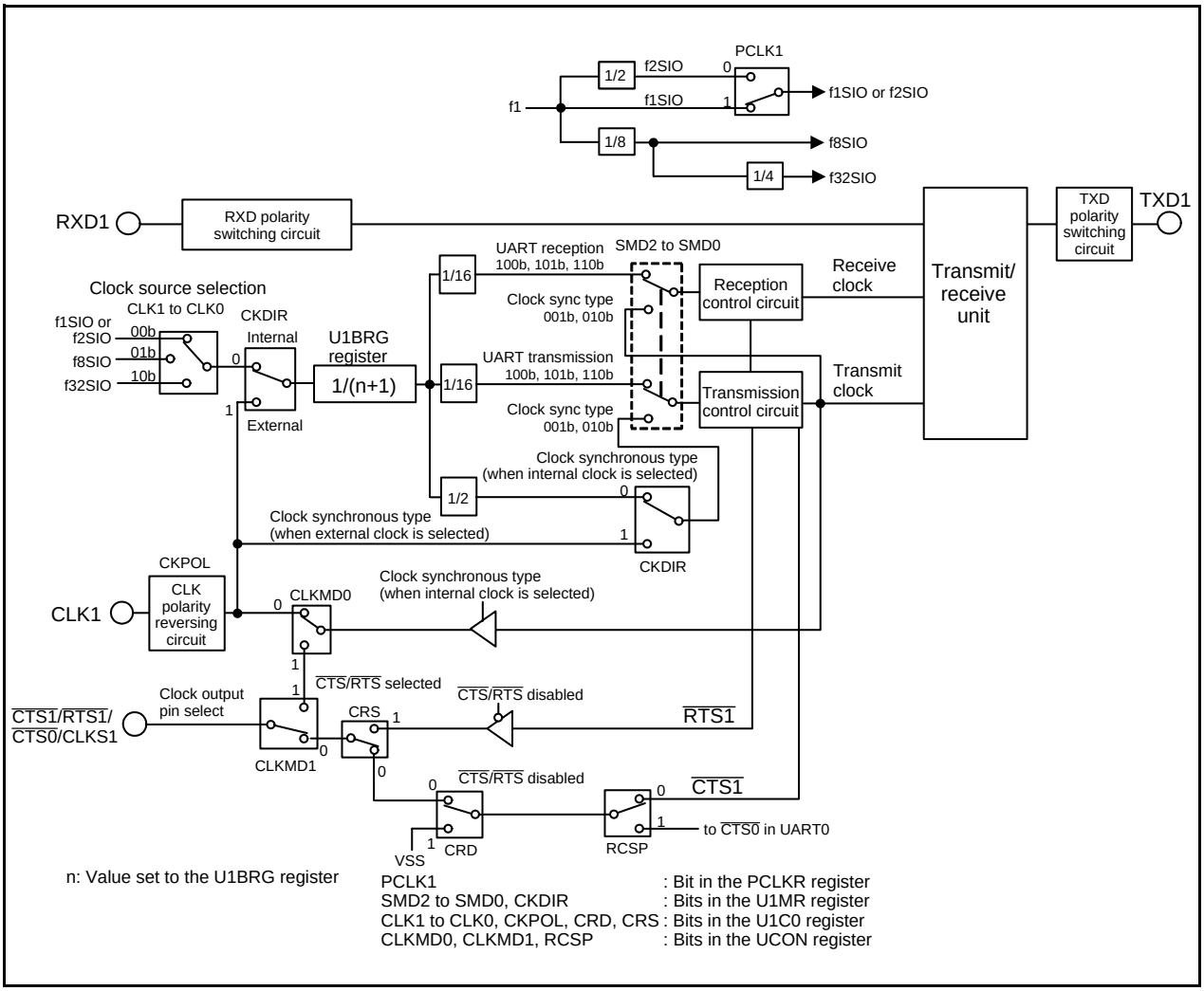
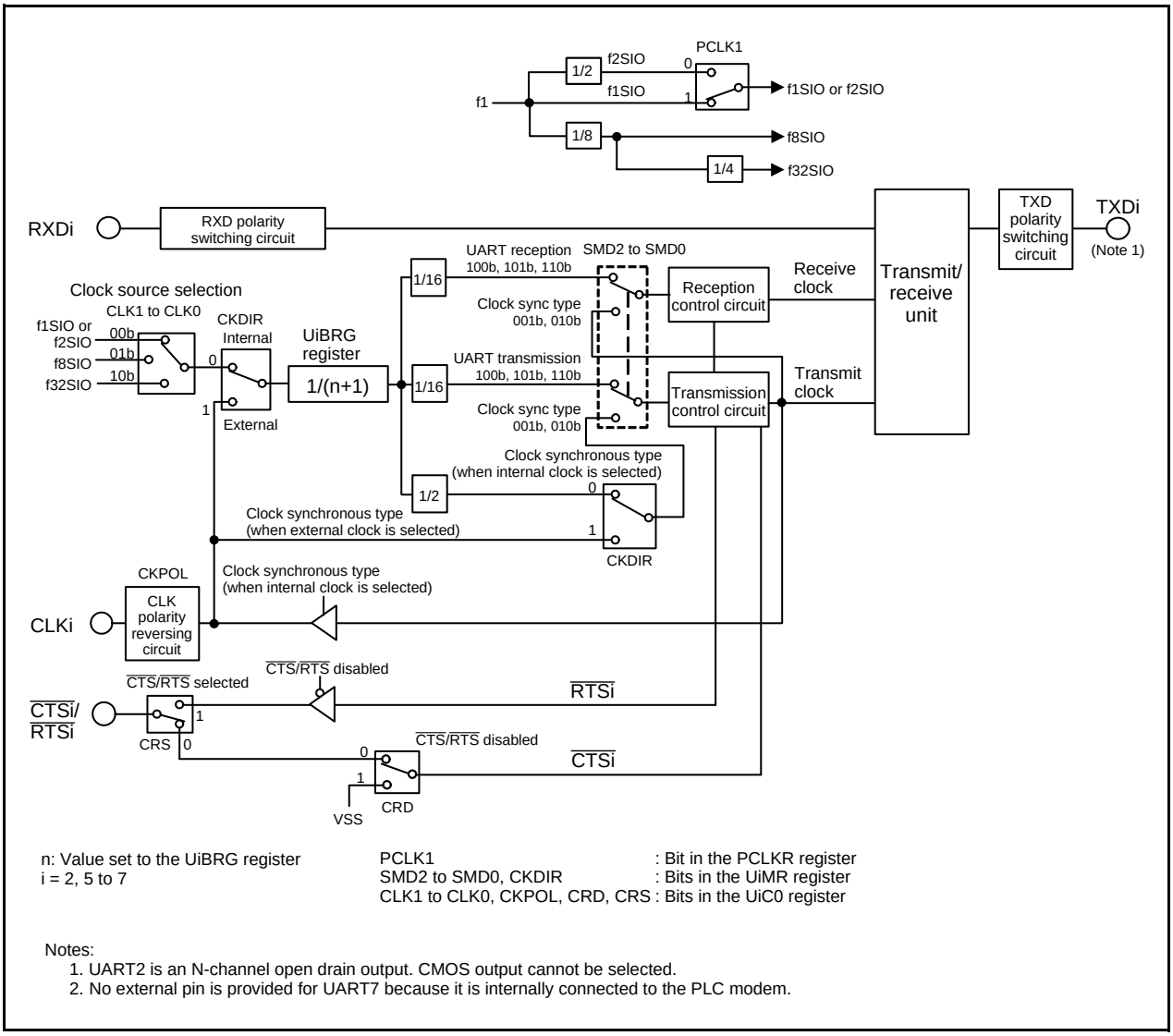


Figure 19.2 UART1 Block Diagram



n: Value set to the UiBRG register
i = 2, 5 to 7

PCLK1 : Bit in the PCLKR register
SMD2 to SMD0, CKDIR : Bits in the UiMR register
CLK1 to CLK0, CKPOL, CRD, CRS : Bits in the UiC0 register

Notes:
1. UART2 is an N-channel open drain output. CMOS output cannot be selected.
2. No external pin is provided for UART7 because it is internally connected to the PLC modem.

Figure 19.3 Block Diagram of UART2, and UART5 to UART7

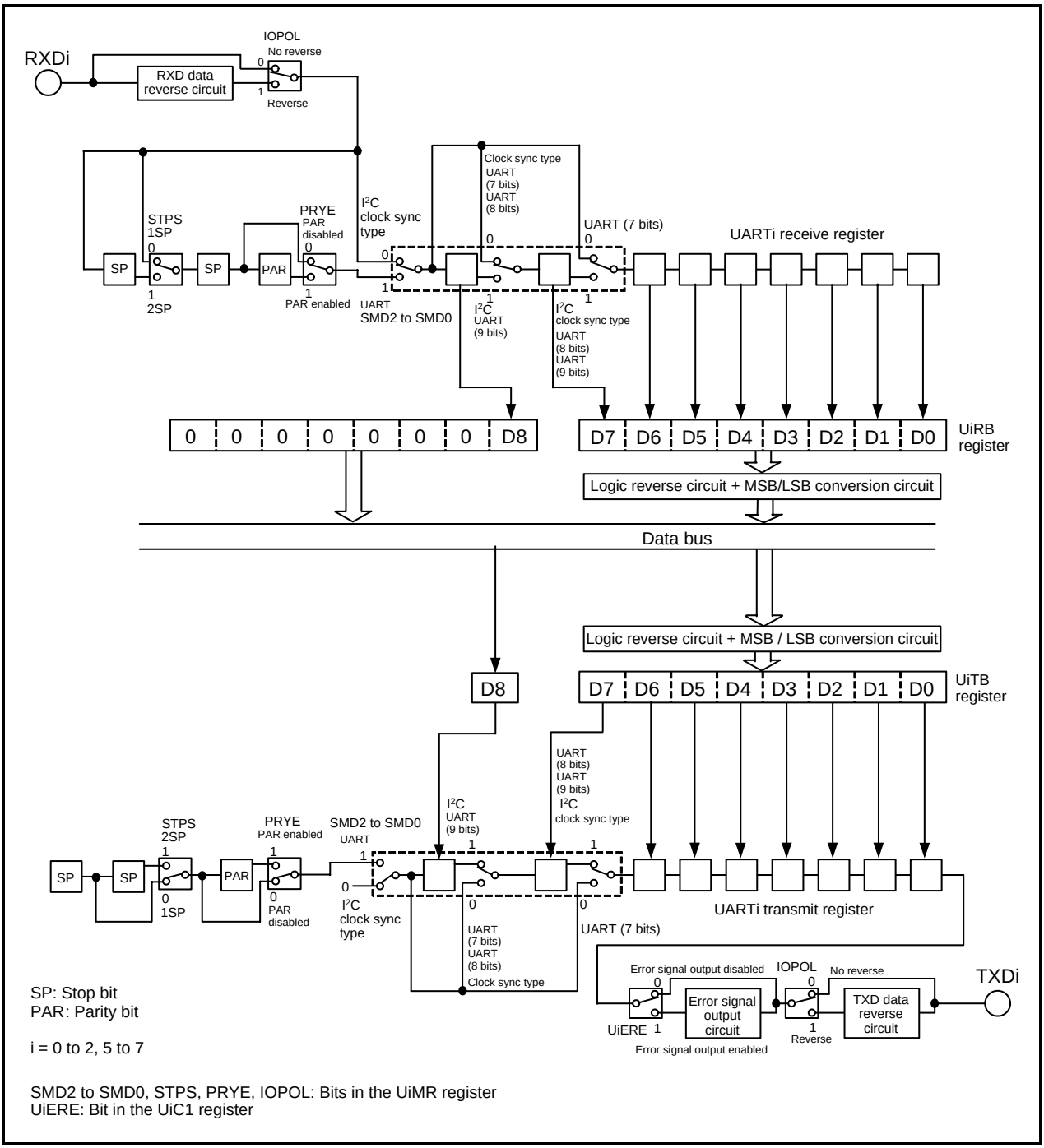


Figure 19.4 UARTi Transmit/Receive Unit Block Diagram

19.2 Registers

Table 19.3 and Table 19.4 list registers associated with UART0 to UART2 and UART5 to UART7. Refer to “Registers Used and Settings” in each mode for the settings of registers and bits.

Table 19.3 Registers (1/2)

Address	Register	Symbol	Reset Value
0012h	Peripheral Clock Select Register	PCLKR	0000 0011b
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
0244h	UART0 Special Mode Register 4	U0SMR4	00h
0245h	UART0 Special Mode Register 3	U0SMR3	000X 0X0Xb
0246h	UART0 Special Mode Register 2	U0SMR2	X000 0000b
0247h	UART0 Special Mode Register	U0SMR	X000 0000b
0248h	UART0 Transmit/Receive Mode Register	U0MR	00h
0249h	UART0 Bit Rate Register	U0BRG	XXh
024Ah	UART0 Transmit Buffer Register	U0TB	XXh
024Bh			XXh
024Ch	UART0 Transmit/Receive Control Register 0	U0C0	0000 1000b
024Dh	UART0 Transmit/Receive Control Register 1	U0C1	00XX 0010b
024Eh	UART0 Receive Buffer Register	U0RB	XXh
024Fh			XXh
0250h	UART Transmit/Receive Control Register 2	UCON	X000 0000b
0254h	UART1 Special Mode Register 4	U1SMR4	00h
0255h	UART1 Special Mode Register 3	U1SMR3	000X 0X0Xb
0256h	UART1 Special Mode Register 2	U1SMR2	X000 0000b
0257h	UART1 Special Mode Register	U1SMR	X000 0000b
0258h	UART1 Transmit/Receive Mode Register	U1MR	00h
0259h	UART1 Bit Rate Register	U1BRG	XXh
025Ah	UART1 Transmit Buffer Register	U1TB	XXh
025Bh			XXh
025Ch	UART1 Transmit/Receive Control Register 0	U1C0	0000 1000b
025Dh	UART1 Transmit/Receive Control Register 1	U1C1	00XX 0010b
025Eh	UART1 Receive Buffer Register	U1RB	XXh
025Fh			XXh
0264h	UART2 Special Mode Register 4	U2SMR4	00h
0265h	UART2 Special Mode Register 3	U2SMR3	000X 0X0Xb
0266h	UART2 Special Mode Register 2	U2SMR2	X000 0000b
0267h	UART2 Special Mode Register	U2SMR	X000 0000b
0268h	UART2 Transmit/Receive Mode Register	U2MR	00h
0269h	UART2 Bit Rate Register	U2BRG	XXh
026Ah	UART2 Transmit Buffer Register	U2TB	XXh
026Bh			XXh
026Ch	UART2 Transmit/Receive Control Register 0	U2C0	0000 1000b
026Dh	UART2 Transmit/Receive Control Register 1	U2C1	0000 0010b
026Eh	UART2 Receive Buffer Register	U2RB	XXh
026Fh			XXh

Table 19.4 Registers (2/2)

Address	Register	Symbol	Reset Value
0284h	UART5 Special Mode Register 4	U5SMR4	00h
0285h	UART5 Special Mode Register 3	U5SMR3	000X 0X0Xb
0286h	UART5 Special Mode Register 2	U5SMR2	X000 0000b
0287h	UART5 Special Mode Register	U5SMR	X000 0000b
0288h	UART5 Transmit/Receive Mode Register	U5MR	00h
0289h	UART5 Bit Rate Register	U5BRG	XXh
028Ah	UART5 Transmit Buffer Register	U5TB	XXh
028Bh			XXh
028Ch	UART5 Transmit/Receive Control Register 0	U5C0	0000 1000b
028Dh	UART5 Transmit/Receive Control Register 1	U5C1	0000 0010b
028Eh	UART5 Receive Buffer Register	U5RB	XXh
028Fh			XXh
0294h	UART6 Special Mode Register 4	U6SMR4	00h
0295h	UART6 Special Mode Register 3	U6SMR3	000X 0X0Xb
0296h	UART6 Special Mode Register 2	U6SMR2	X000 0000b
0297h	UART6 Special Mode Register	U6SMR	X000 0000b
0298h	UART6 Transmit/Receive Mode Register	U6MR	00h
0299h	UART6 Bit Rate Register	U6BRG	XXh
029Ah	UART6 Transmit Buffer Register	U6TB	XXh
029Bh			XXh
029Ch	UART6 Transmit/Receive Control Register 0	U6C0	0000 1000b
029Dh	UART6 Transmit/Receive Control Register 1	U6C1	0000 0010b
029Eh	UART6 Receive Buffer Register	U6RB	XXh
029Fh			XXh
02A4h	UART7 Special Mode Register 4	U7SMR4	00h
02A5h	UART7 Special Mode Register 3	U7SMR3	000X 0X0Xb
02A6h	UART7 Special Mode Register 2	U7SMR2	X000 0000b
02A7h	UART7 Special Mode Register	U7SMR	X000 0000b
02A8h	UART7 Transmit/Receive Mode Register	U7MR	00h
02A9h	UART7 Bit Rate Register	U7BRG	XXh
02AAh	UART7 Transmit Buffer Register	U7TB	XXh
02ABh			XXh
02ACh	UART7 Transmit/Receive Control Register 0	U7C0	0000 1000b
02ADh	UART7 Transmit/Receive Control Register 1	U7C1	0000 0010b
02AEh	UART7 Receive Buffer Register	U7RB	XXh
02AFh			XXh

19.2.1 Peripheral Clock Select Register (PCLKR)

Peripheral Clock Select Register

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

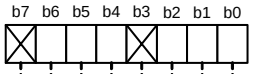
Symbol	Address	After Reset
PCLKR	0012h	0000 0011b

Bit Symbol	Bit Name	Function	RW
PCLK0	Timers A and B clock select bit (clock source for timers A and B, and multi-master I^2C-bus interface)	0: f2TIMAB/f2IIC 1: f1TIMAB/f1IIC	RW
PCLK1	SI/O clock select bit (clock source for UART0 to UART2, UART5 to UART7, SI/O3, and SI/O4)	0: f2SIO 1: f1SIO	RW
— (b4-b2)	Reserved bits	Set to 0.	RW
PCLK5	Clock output function expansion bit	0: Selected by bits CM01 to CM00 in the CM0 register 1: Output f1	RW
— (b7-b6)	Reserved bits	Set to 0.	RW

Set the PRC0 bit in the PRCR register to 1 (write enabled) before the PCLKR register is rewritten.

19.2.2 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1



Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I^2C peripheral clock stop bit (multi-master I^2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

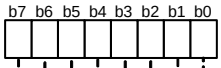
Set the PRC0 bit in the PRCR register to 1 (write enabled) before the PCKSTP1 register is rewritten.

PCKSTP12 (UART peripheral clock stop bit) (b2)

Set the PCKSTP12 bit to 0 (f1 provide enabled) when using the f1 as the clock source of the transmit/receive clock.

19.2.3 UARTi Transmit/Receive Mode Register (UiMR) (i = 0 to 2, 5 to 7)

UARTi Transmit/Receive Mode Register (i = 0 to 2, 5 to 7)



Symbol	Address	After Reset
U0MR, U1MR, U2MR	0248h, 0258h, 0268h	00h
U5MR, U6MR, U7MR	0288h, 0298h, 02A8h	00h

Bit Symbol	Bit Name	Function	RW
SMD0	Serial I/O mode select bit	b2 b1 b0 0 0 0: Serial interface disabled 0 0 1: Clock synchronous serial I/O mode 0 1 0: I^2C mode 1 0 0: UART mode character bit length is 7 bits 1 0 1: UART mode character bit length is 8 bits 1 1 0: UART mode character bit length is 9 bits Only set the values 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: One stop bit 1: Two stop bits	RW
PRY	Odd/even parity select bit	Enabled when PRYE is 1 0: Odd parity 1: Even parity	RW
PRYE	Parity enable bit	0: Parity disabled 1: Parity enabled	RW
IOPOL	TXD, RXD I/O polarity reverse bit	0: No reverse 1: Reverse	RW

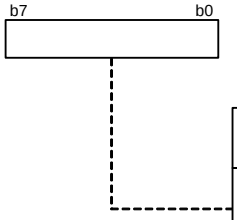
SMD2 to SMD0 (Serial I/O mode select bit) (b2 to b0)

When setting bits SMD2 to SMD0 to 000b (serial interface disabled), set the TE bit in the UiC1 register to 0 (transmission disabled) and the RE bit to 0 (reception disabled).

When using I^2C mode, set the IICM bit in the UiSMR register to 1 (I^2C mode), then set bits SMD2 to SMD0 to 010b (I^2C mode).

19.2.4 UARTi Bit Rate Register (UiBRG) (i = 0 to 2, 5 to 7)

UARTi Bit Rate Register (i = 0 to 2, 5 to 7)



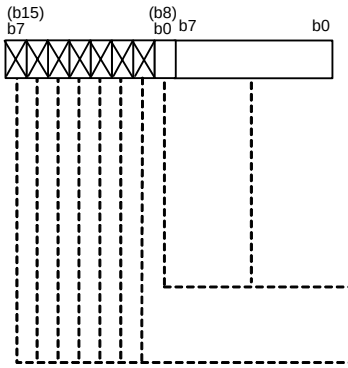
Symbol	Address	After Reset
U0BRG, U1BRG, U2BRG U5BRG, U6BRG, U7BRG	0249h, 0259h, 0269h 0289h, 0299h, 02A9h	XXh XXh

Function	Setting Range	RW
If set value is n, UiBRG divides the count source by n + 1.	00h to FFh (other than I^2C mode) 03h to FFh (I^2C mode)	WO

Write to the UiBRG register while the serial interface is neither transmitting nor receiving.
Use the MOV instruction to write to the UiBRG register.
Write to the UiBRG register after setting bits CLK1 to CLK0 in the UiC0 register.

19.2.5 UARTi Transmit Buffer Register (UiTB) (i = 0 to 2, 5 to 7)

UARTi Transmit Buffer Register (i = 0 to 2, 5 to 7)



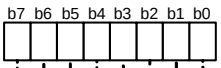
Symbol	Address	After Reset
U0TB	024Bh to 024Ah	XXXXh
U1TB	025Bh to 025Ah	XXXXh
U2TB	026Bh to 026Ah	XXXXh
U5TB	028Bh to 028Ah	XXXXh
U6TB	029Bh to 029Ah	XXXXh
U7TB	02ABh to 02AAh	XXXXh

Function	RW
Transmit data	WO
No register bits. If necessary, set to 0. Read as undefined value.	—

Use the MOV instruction to write to this register.
When character length is 9 bits long or I^2C mode, write to this register in 16-bit units, or in 8-bit units from upper byte to lower byte.

19.2.6 UARTi Transmit/Receive Control Register 0 (UiC0) (i = 0 to 2, 5 to 7)

UARTi Transmit/Receive Control Register 0 (i = 0 to 2, 5 to 7)



Symbol	Address	After Reset
U0C0, U1C0, U2C0 U5C0, U6C0, U7C0	024Ch, 025Ch, 026Ch 028Ch, 029Ch, 02ACh	0000 1000b 0000 1000b

Bit Symbol	Bit Name	Function	RW
CLK0	UiBRG count source select bit	b1 b0 0 0: f1SIO or f2SIO selected 0 1: f8SIO selected 1 0: f32SIO selected 1 1: Do not set	RW
CLK1			RW
CRS	$\overline{CTS}/\overline{RTS}$ function select bit	Enabled when CRD is 0 0: $\overline{CTS}$ function selected 1: $\overline{RTS}$ function selected	RW
TXEPT	Transmit register empty flag	0: Data present in transmit register (transmission in progress) 1: No data present in transmit register (transmission completed)	RO
CRD	$\overline{CTS}/\overline{RTS}$ disable bit	0: $\overline{CTS}/\overline{RTS}$ function enabled 1: $\overline{CTS}/\overline{RTS}$ function disabled	RW
NCH	Data output select bit	0: Pins TXDi/SDAi and SCLi are CMOS output 1: Pins TXDi/SDAi and SCLi are N-channel open-drain output	RW
CKPOL	CLK polarity select bit	0: Transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge 1: Transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge	RW
UFORM	Bit order select bit	0: LSB first 1: MSB first	RW

CLK1 to CLK0 (UiBRG count source select bit) (b1 to b0)

When bits CLK1 to CLK0 are 00b (f1SIO or f2SIO selected), select f1SIO or f2SIO by the PCLK1 bit in the PCLKR register.
Set bits CLK1 to CLK0 after setting the PCLKR register.
If bits CLK1 to CLK0 are changed, set the UiBRG register.

CRS ($\overline{CTS}/\overline{RTS}$ function select bit) (b2)

$\overline{CTS1}/\overline{RTS1}$ can be used when the CLKMD1 bit in the UCON register is 0 (CLK output is only from CLK1) and the RCSP bit in the UCON register is 0 ($\overline{CTS0}/\overline{RTS0}$ not separated).

CRD ($\overline{CTS}/\overline{RTS}$ disable bit) (b4)

When the CRD bit is 1 ($\overline{CTS}/\overline{RTS}$ function disabled), the $\overline{CTSi}/\overline{RTSi}$ pin can be used as an I/O port.

NCH (Data output select bit) (b5)

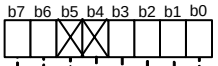
TXD2/SDA2 and SCL2 are N-channel open drain outputs. They cannot be set as CMOS outputs. Nothing is assigned to the NCH bit in the U2C0 register. If necessary, set this bit to 0.
This function is used to set the P-channel transistor of the COMS output buffer always off, but not to change pins TXDi/SDAi and SCLi to open drain output completely.
Check the electrical characteristics for the input voltage range.

UFORM (Bit order select bit) (b7)

The UFORM bit is enabled when bits SMD2 to SMD0 in the UiMR register are 001b (clock synchronous serial I/O mode), or 101b (UART mode, 8-bit character data).
Set the UFORM bit to 1 when bits SMD2 to SMD0 are 010b (I^2C mode), and to 0 when bits SMD2 to SMD0 are 100b (UART mode, 7-bit character data) or 110b (UART mode, 9-bit character data).

19.2.7 UARTi Transmit/Receive Control Register 1 (UiC1) (i = 0 to 2, 5 to 7)

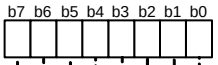
UARTi Transmit/Receive Control Register 1 (i = 0, 1)



Symbol	Address	After Reset
U0C1, U1C1	024Dh, 025Dh	00XX 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 present in UiTB register 1: No data present in UiTB register	RO
RE	Receive enable bit	0: Reception disabled 1: Reception enabled	RW
RI	Receive complete flag	0: No data present in UiRB register 1: Data present in UiRB register	RO
— (b5-b4)	No register bits. If necessary, set to 0. Read as undefined value.		—
UiLCH	Data logic select bit	0: No reverse 1: Reverse	RW
UiERE	Error signal output enable bit	0: Output disabled 1: Output enabled	RW

UARTi Transmit/Receive Control Register 1 (i = 2, 5 to 7)



Symbol	Address	After Reset
U2C1	026Dh	0000 0010b
U5C1, U6C1, U7C1	028Dh, 029Dh, 02ADh	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 present in UiTB register 1: No data present in UiTB register	RO
RE	Receive enable bit	0: Reception disabled 1: Reception enabled	RW
RI	Receive complete flag	0: No data present in UiRB register 1: Data present in UiRB register	RO
UiIRS	UARTi transmit interrupt source select bit	0: UiTB register empty (TI = 1) 1: Transmission completed (TXEPT = 1)	RW
UiRRM	UARTi continuous receive mode enable bit	0: Continuous receive mode disabled 1: Continuous receive mode enabled	RW
UiLCH	Data logic select bit	0: No reverse 1: Reverse	RW
UiERE	Error signal output enable bit	0: Output disabled 1: Output enabled	RW

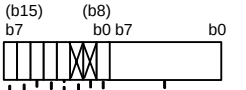
Bits UiIRS and UiRRM of UART0 and UART1 are bits in the UCON register.

UiLCH (Data logic select bit) (b6)

The UiLCH bit is enabled when bits SMD2 to SMD0 in the UiMR register are 001b (clock synchronous serial I/O mode), 100b (UART mode, 7-bit character data), or 101b (UART mode, 8-bit character data). Set this bit to 0 when bits SMD2 to SMD0 are set to 010b (I^2C mode) or 110b (UART mode, 9-bit character data).

19.2.8 UARTi Receive Buffer Register (UiRB) (i = 0 to 2, 5 to 7)

UARTi Receive Buffer Register (i = 0 to 2, 5 to 7)



Symbol	Address	After Reset
U0RB	024Fh to 024Eh	XXXXh
U1RB	025Fh to 025Eh	XXXXh
U2RB	026Fh to 026Eh	XXXXh
U5RB	028Fh to 028Eh	XXXXh
U6RB	029Fh to 029Eh	XXXXh
U7RB	02AFh to 02AEh	XXXXh

Bit Symbol	Bit Name	Function	RW
— (b7-b0)	——	Receive data (D7 to D0)	RO
— (b8)	——	Receive data (D8)	RO
— (b10-b9)	No register bits. If necessary, set to 0. Read as undefined value.		—
ABT	Arbitration lost detect flag	0: Not detected 1: Detected	RW
OER	Overrun error flag	0: No overrun error 1: Overrun error found	RO
FER	Framing error flag	0: No framing error 1: Framing error found	RO
PER	Parity error flag	0: No parity error 1: Parity error found	RO
SUM	Error sum flag	0: No error 1: Error found	RO

When bits SMD2 to SMD0 in the UiMR register are 100b, 101b, or 110b, read this register in 16-bit units, or in 8-bit units from high-order bytes to low-order bytes.
Bits FER and PER arranged in the high-order bytes become 0 when the lower bytes of the UiRB register are read.
If an overrun error occurs, the receive data of the UiRB register is undefined.

ABT (Arbitration lost detect flag) (b11)

The ABT bit is set to 0 by a program. (It remains unchanged even if 1 is written.)

OER (Overrun error flag) (b12)

Conditions to become 0:

- Bits SMD2 to SMD0 in the UiMR register are 000b (serial interface disabled).
- The RE bit in the UiC1 register is 0 (reception disabled).

Condition to become 1:

- The RI bit in the UiC1 register is 1 (data present in UiRB register), and the last bit of the next data is received.

FER (Framing error flag) (b13)

The FER bit is disabled when bits SMD2 to SMD0 are set to 001b (clock synchronous serial I/O mode) or to 010b (I^2C mode). The read value is undefined.

Condition to become 0:

- Bits SMD2 to SMD0 in the UiMR register are 000b (serial interface disabled).
- The RE bit in the UiC1 register is 0 (reception disabled).
- The lower bytes of the UiRB register are read.

Condition to become 1:

- The set number of stop bits is not detected.
  (detected when the received data is transferred from the UARTi receive register to the UiRB register.)

PER (Parity error flag) (b14)

The PER bit is disabled when bits SMD2 to SMD0 are set to 001b (clock synchronous serial I/O mode) or to 010b (I^2C mode). The read value is undefined.

Conditions to become 0:

- Bits SMD2 to SMD0 in the UiMR register are 000b (serial interface disabled).
- The RE bit in the UiC1 register is 0 (reception disabled).
- The lower bytes of the UiRB register are read.

Condition to become 1:

- The number of 1's of the parity bit and character bits do not match the set value of the PRY bit in the UiMR register.
  (detected when the received data is transferred from the UARTi receive register to the UiRB register.)

SUM (Error sum flag) (b15)

The SUM bit is disabled when bits SMD2 to SMD0 are set to 001b (clock synchronous serial I/O mode) or to 010b (I^2C mode). The read value is undefined.

Conditions to become 0:

- Bits SMD2 to SMD0 in the UiMR register are 000b (serial interface disabled).
- The RE bit in the UiC1 register is 0 (reception disabled).
- Bits PER, FER and OER are all 0 (no error).

Condition to become 1:

- At least two bits out of PER, FER, or OER are 1 (error found).

19.2.9 UART Transmit/Receive Control Register 2 (UCON)

UART Transmit/Receive Control Register 2

b7	b6	b5	b4	b3	b2	b1	b0
X							

Symbol	Address	After Reset
UCON	0250h	X000 0000b

Bit symbol	Bit Name	Function	RW
U0IRS	UART0 transmit interrupt source select bit	0: Transmit buffer empty (TI = 1) 1: Transmission completed (TXEPT = 1)	RW
U1IRS	UART1 transmit interrupt source select bit	0: Transmit buffer empty (TI = 1) 1: Transmission completed (TXEPT = 1)	RW
U0RRM	UART0 continuous receive mode enable bit	0: Continuous receive mode disabled 1: Continuous receive mode enabled	RW
U1RRM	UART1 continuous receive mode enable bit	0: Continuous receive mode disabled 1: Continuous receive mode enabled	RW
CLKMD0	UART1CLK, CLKS select bit 0	Enabled when CLKMD1 is 1 0: Clock output from CLK1 1: Clock output from CLKS1	RW
CLKMD1	UART1CLK, CLKS select bit 1	0: CLK output is only from CLK1 1: Transmit/receive clock output from multiple-pin output function selected	RW
RCSP	Separate UART0 $\overline{CTS}/\overline{RTS}$ bit	0: $\overline{CTS}/\overline{RTS}$ shared pin 1: $\overline{CTS}/\overline{RTS}$ separated	RW
— (b7)	No register bit. If necessary, set to 0. Read as undefined value.		—

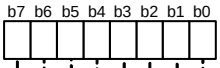
Bits UiIRS and UiRRM of UART2 and UART5 to UART7 are bits in the UiC1 register.

CLKMD1 (UART1CLK, CLKS select bit 1) (b5)

When using multiple transmit/receive clock output pins, make sure that the CKDIR bit in the U1MR register is 0 (internal clock).

19.2.10 UARTi Special Mode Register 4 (UiSMR4) (i = 0 to 2, 5 to 7)

UARTi Special Mode Register 4 (i = 0 to 2, 5 to 7)



Symbol	Address	After Reset
U0SMR4, U1SMR4, U2SMR4	0244h, 0254h, 0264h	00h
U5SMR4, U6SMR4, U7SMR4	0284h, 0294h, 02A4h	00h

Bit Symbol	Bit Name	Function	RW
STAREQ	Start condition generate bit	0: Clear 1: Start	RW
RSTAREQ	Restart condition generate bit	0: Clear 1: Start	RW
STPREQ	Stop condition generate bit	0: Clear 1: Start	RW
STSPSEL	SCL, SDA output select bit	0: Start and stop conditions not output 1: Start and stop conditions output	RW
ACKD	ACK data bit	0: ACK 1: NACK	RW
ACKC	ACK data output enable bit	0: Serial interface data output 1: ACK data output	RW
SCLHI	SCL output stop bit	If stop condition is detected, 0 : Do not stop SCLi output 1 : Stop SCLi output	RW
SWC9	SCL wait auto insert bit 3	0 : No wait-state/wait-state cleared 1 : Hold the SCLi pin low after the ninth bit of the SCLi is received	RW

STAREQ (Start condition generate bit) (b0)

The STAREQ bit becomes 0 when the start condition is generated.

This bit is used in master mode of I^2C mode. To set this bit to 1, preset the IICM bit in the UiSMR register to 1 (I^2C mode). Do not set this bit to 1 when the IICM bit is 0.

RSTAREQ (Restart condition generate bit) (b1)

The RSTAREQ bit becomes 0 when the restart condition is generated.

This bit is used in master mode of I^2C mode. To set this bit to 1, preset the IICM bit in the UiSMR register to 1 (I^2C mode). Do not set this bit to 1 when the IICM bit is 0.

STPREQ (Stop condition generate bit) (b2)

The STPREQ bit becomes 0 when the stop condition is generated.

This bit is used in master mode of I^2C mode. To set this bit to 1, preset the IICM bit in the UiSMR register to 1 (I^2C mode). Do not set this bit to 1 when the IICM bit is 0.

STSPSEL (SCL, SDA output select bit) (b3)

This bit is used in master mode of I^2C mode. To set this bit to 1, preset the IICM bit in the UiSMR register to 1 (I^2C mode). Do not set this bit to 1 when the IICM bit is 0.

Set the STSPSEL bit to 1 (select start condition/stop condition generate circuit) after setting the STARREQ, RSTAREQ, or STPREQ bit to 1 (start).

ACKD (ACK data bit) (b4)
ACKC (ACK data output enable bit) (b5)
SWC9 (SCL wait auto insert bit 3) (b7)

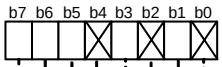
This bit is used in slave mode of I^2C mode. To set this bit to 1, preset the IICM bit in the UiSMR register to 1 (I^2C mode). Do not set this bit to 1 when the IICM bit is 0.

SCLHI (SCL output stop bit) (b6)

This bit is used in master mode of I^2C mode. To set this bit to 1, preset the IICM bit in the UiSMR register to 1 (I^2C mode). Do not set this bit to 1 when the IICM bit is 0.

19.2.11 UARTi Special Mode Register 3 (UiSMR3) (i = 0 to 2, 5 to 7)

UARTi Special Mode Register 3 (i = 0 to 2, 5 to 7)



Symbol	Address	After Reset
U0SMR3, U1SMR3, U2SMR3 U5SMR3, U6SMR3, U7SMR3	0245h, 0255h, 0265h 0285h, 0295h, 02A5h	000X 0X0Xb 000X 0X0Xb

Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit. If necessary, set to 0. Read as undefined value.		—
CKPH	Clock phase set bit	0: No clock delay 1: With clock delay	RW
— (b2)	No register bit. If necessary, set to 0. Read as undefined value.		—
NODC	Clock output select bit	0: CLKi is CMOS output 1: CLKi is N-channel open drain output	RW
— (b4)	No register bit. If necessary, set to 0. Read as undefined value.		—
DL0	SDAi digital delay setup bit	b7 b6 b5 0 0 0: No delay 0 0 1: 1 to 2 cycles of UiBRG count source 0 1 0: 2 to 3 cycles of UiBRG count source 0 1 1: 3 to 4 cycles of UiBRG count source 1 0 0: 4 to 5 cycles of UiBRG count source 1 0 1: 5 to 6 cycles of UiBRG count source 1 1 0: 6 to 7 cycles of UiBRG count source 1 1 1: 7 to 8 cycles of UiBRG count source	RW
DL1			RW
DL2			RW

NODC (Clock output select bit) (b3)

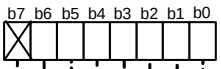
This function is used to set P-channel transistor of the COMS output buffer always off, but not to change the CLKi pin to open drain output completely.
Check the electrical characteristics for the input voltage range.

DL2 to DL0 (SDAi digital delay setup bit) (b7 to b5)

Bits DL2 to DL0 are used to generate a digital delay in SDAi output in I^2C mode. Except in I^2C mode, set these bits to 000b (no delay).
The delay length varies with the load on pins SCLi and SDAi. Also, when using an external clock, the delay length increases by about 100 ns.

19.2.12 UARTi Special Mode Register 2 (UiSMR2) (i = 0 to 2, 5 to 7)

UARTi Special Mode Register 2 (i = 0 to 2, 5 to 7)

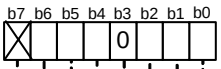


Symbol	Address	After Reset
U0SMR2, U1SMR2, U2SMR2 U5SMR2, U6SMR2, U7SMR2	0246h, 0256h, 0266h 0286h, 0296h, 02A6h	X000 0000b X000 0000b

Bit Symbol	Bit Name	Function	RW
IICM2	I^2C mode select bit 2	0 : Use NACK/ACK interrupt 1 : Use transmit/receive interrupt	RW
CSC	Clock synchronization bit	0 : Clock synchronization disabled 1 : Clock synchronization enabled	RW
SWC	SCL wait auto insert bit	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	When arbitration lost is detected, 0 : Do not stop the SDAi output 1 : Stop the SDAi output	RW
STAC	UARTi auto initialize bit	When the start condition is detected, 0 : Do not initialize the circuit 1 : Initialize the circuit	RW
SWC2	SCL wait output bit 2	0 : Output the transmit/receive clock at the SCLi pin 1 : Hold the SCLi pin low	RW
SDHI	SDA output disable bit	0 : Output data 1 : Stop the output (high-impedance)	RW
— (b7)	No register bit. If necessary, set to 0. Read as undefined value.		—

19.2.13 UARTi Special Mode Register (UiSMR) (i = 0 to 2, 5 to 7)

UARTi Special Mode Register (i = 0 to 2, 5 to 7)



Symbol	Address	After Reset
U0SMR, U1SMR, U2SMR U5SMR, U6SMR, U7SMR	0247h, 0257h, 0267h 0287h, 0297h, 02A7h	X000 0000b X000 0000b

Bit Symbol	Bit Name	Function	RW
IICM	I^2C mode select bit	0: Other than I^2C mode 1: I^2C mode	RW
ABC	Arbitration lost detect flag control bit	0: Update every bit 1: Update every byte	RW
BBS	Bus busy flag	0: Stop-condition detected 1: Start-condition detected (busy)	RW
— (b3)	Reserved bit	Set to 0.	RW
ABSCS	Bus collision detect sampling clock select bit	0: Rising edge of transmit/receive clock 1: Underflow signal of timer Aj	RW
ACSE	Auto clear function select bit of transmit enable bit	0: No auto clear function 1: Auto clear at bus collision	RW
SSS	Transmit start condition select bit	0: Not synchronized to RXDi 1: Synchronized to RXDi	RW
— (b7)	No register bit. If necessary, set to 0. Read as undefined value.		—

BBS (Bus busy flag) (b2)

The BBS bit is set to 0 by a program. (It remains unchanged even if 1 is written.)

ABSCS (Bus collision detect sampling clock select bit) (b4)

When the ABSCS bit is 1, the combinations of UARTi and timer Aj are as follows:

UART0, UART6: Underflow signal of timer A3
UART1, UART7: Underflow signal of timer A4
UART2, UART5: Underflow signal of timer A0

SSS (Transmit start condition select bit) (b6)

When a transmission starts, the SSS bit becomes 0 (not synchronized to RXDi).

19.3 Operations

19.3.1 Clock Synchronous Serial I/O Mode

The clock synchronous serial I/O mode uses a transmit/receive clock to transmit/receive data. Table 19.5 lists the Clock Synchronous Serial I/O Mode Specifications.

Table 19.5 Clock Synchronous Serial I/O Mode Specifications

Item	Specification
Data format	Character length: 8 bits
Transmit/receive clock	• CKDIR bit in the UiMR register = 0 (internal clock): $\frac{fj}{2(n+1)}$ fj = f1SIO, f2SIO, f8SIO, f32SIO n = Setting value of UiBRG register 00h to FFh • CKDIR bit = 1 (external clock): Input from CLKi pin
Transmit/receive control	Selectable from $\overline{CTS}$ function, $\overline{RTS}$ function, or $\overline{CTS}$/$\overline{RTS}$ function disabled
Transmission start conditions	To start transmission, satisfy the following requirements (1) • The TE bit in the UiC1 register = 1 (transmission enabled) • The TI bit in the UiC1 register = 0 (data present in UiTB register) • If $\overline{CTS}$ function is selected, input on the $\overline{CTSi}$ pin is low.
Reception start conditions	To start reception, satisfy the following requirements (1) • The RE bit in the UiC1 register = 1 (reception enabled) • The TE bit in the UiC1 register = 1 (transmission enabled) • The TI bit in the UiC1 register = 0 (data present in the UiTB register)
Interrupt request generation timing	Transmit interrupt: One of the following can be selected. • The UiIRS bit in the UiC1 or UCON register = 0 (transmit buffer empty): When transferring data from the UiTB register to the UARTi transmit register (at start of transmission) • The UiIRS bit = 1 (transmission completed): When the serial interface completes sending data from the UARTi transmit register Receive interrupt: • When transferring data from the UARTi receive register to the UiRB register (at completion of reception)
Error detection	Overrun error (2) This error occurs if the serial interface starts receiving the next unit of data before reading the UiRB register and receiving the seventh bit of the next unit of data
Selectable functions	• CLK polarity selection Data input/output can be selected to occur synchronously with the rising or falling edge of the transmit/receive clock • LSB first, MSB first selection Whether to start sending/receiving data beginning with bit 0 or beginning with bit 7 can be selected • Continuous receive mode selection Reception is enabled immediately by reading the UiRB register • Switching serial data logic This function reverses the logic value of the transmit/receive data • Transmit/receive clock output from multiple pins selection (UART1) The output pin can be selected by a program by setting two UART1 transmit/receive clock pins. • Separate $\overline{CTS}$/$\overline{RTS}$ pins (UART0) $\overline{CTS0}$ and $\overline{RTS0}$ are input/output from separate pins

i = 0 to 2, 5 to 7

Notes:

1. These requirements do not have to be set in any particular order. If transmission/reception is started while an external clock is selected and the TXEPT bit in the UiC0 register is 1 (no data present in transmit register), meet the last requirement at either of the following timings:
   - The CKPOL bit in the UiC0 register is 0 (transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge) and the external clock is high.
   - The CKPOL bit is 1 (transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge) and the external clock is low.
2. If an overrun error occurs, the receive data of the UiRB register will be undefined. The IR bit in the SiRIC register remains unchanged.

Table 19.6 lists Pin Functions in Clock Synchronous Serial I/O Mode (Multiple Transmit/Receive Clock Output Pin Function Not Selected). Table 19.7 lists P6_4 Pin Functions in Clock Synchronous Serial I/O Mode.
Note that for a period from when UARTi operating mode is selected to when transmission starts, the TXDi pin outputs a high-level signal. (If N-channel open drain output is selected, this pin is high-impedance.)
Table 19.8 lists the Registers Used and Settings in Clock Synchronous Serial I/O Mode.

Table 19.6 Pin Functions in Clock Synchronous Serial I/O Mode (Multiple Transmit/Receive Clock Output Pin Function Not Selected)

Pin Name	I/O	Function	Method of Selection
TXDi	Output	Serial data output	(Outputs dummy data only when receiving)
RXDi	Input	Serial data input	Set the port direction bits sharing pins to 0.
	Input	Input port	Set the port direction bits to 0. (can be used as an input port only when transmitting)
CLKi	Output	Transmit/receive clock output	The CKDIR bit in the UiMR register = 0
	Input	Transmit/receive clock input	The CKDIR bit in the UiMR register = 1 Set the port direction bits sharing pins to 0.
$\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$	Input	$\overline{\text{CTS}}$ input	The CRD bit in the UiC0 register = 0 The CRS bit in the UiC0 register = 0 Set the port direction bits sharing pins to 0.
	Output	$\overline{\text{RTS}}$ output	The CRD bit in the UiC0 register = 0 The CRS bit in the UiC0 register = 1
	Input/ output	I/O port	The CRD bit in the UiC0 register = 1

i = 0 to 2, 5 to 7

Table 19.7 P6_4 Pin Functions in Clock Synchronous Serial I/O Mode

Pin Function	Bit Set Value					
	U1C0 register		UCON register			PD6 register
	CRD	CRS	RCSP	CLKMD1	CLKMD0	PD6_4
P6_4	1	—	0	0	—	Input: 0, Output: 1
$\overline{\text{CTS1}}$	0	0	0	0	—	0
$\overline{\text{RTS1}}$	0	1	0	0	—	—
$\overline{\text{CTS0}}$ (1)	0	0	1	0	—	0
CLKS1	—	—	—	1 (2)	1	—

— indicates either 0 or 1

Notes:
1. In addition to these settings, set the CRD bit in the U0C0 register to 0 ($\overline{\text{CTS0}}$/$\overline{\text{RTS0}}$ enabled) and the CRS bit in the U0C0 register to 1 ($\overline{\text{RTS0}}$ selected).
2. When the CLKMD1 bit is 1 and the CLKMD0 bit is 0, the following logic levels are output:
   • High if the CLKPOL bit in the U1C0 register is 0
   • Low if the CLKPOL bit in the U1C0 register is 1

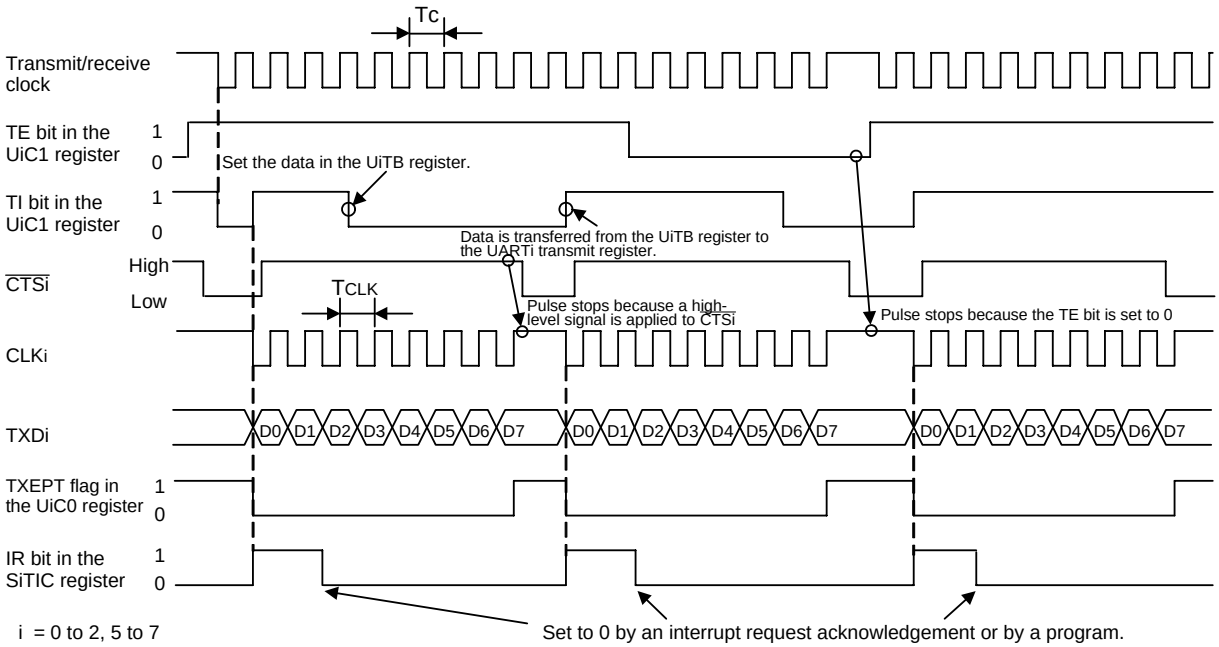
Table 19.8 Registers Used and Settings in Clock Synchronous Serial I/O Mode (1)

Register	Bits	Function
PCLKR	PCLK1	Select the count source for the UiBRG register.
PCLKSTP1	PCKSTP12	Set to 0 when using f1.
UiTB	0 to 7	Set transmission data.
	8	— (does not need to be set) If necessary, set to 0.
UiRB	0 to 7	Reception data can be read.
	8, 11, 13 to 15	When read, the read value is undefined.
	OER	Overrun error flag
UiBRG	0 to 7	Set bit rate.
UiMR	SMD2 to SMD0	Set to 001b.
	CKDIR	Select internal clock or external clock.
	4 to 6	Set to 0.
	IOPOL	Set to 0.
UiC0	CLK1 to CLK0	Select the count source for the UiBRG register.
	CRS	If $\overline{CTS}$ or $\overline{RTS}$ is used, select which function to use.
	TXEPT	Transmit register empty flag
	CRD	Enable or disable the $\overline{CTS}$ or $\overline{RTS}$ function.
	NCH	Select TXDi pin output mode. (2)
	CKPOL	Select the transmit/receive clock polarity.
	UFORM	Select LSB first or MSB first.
UiC1	TE	Set to 1 to enable transmission/reception.
	TI	Transmit buffer empty flag
	RE	Set to 1 to enable reception.
	RI	Reception complete flag
	UjIRS	Select source of UARTj transmit interrupt.
	UjRRM	Set to 1 to use continuous receive mode.
	UiLCH	Set to 1 to use inverted data logic.
	UiERE	Set to 0.
UiSMR	0 to 7	Set to 0.
UiSMR2	0 to 7	Set to 0.
UiSMR3	0 to 2	Set to 0.
	NODC	Select clock output mode.
	4 to 7	Set to 0.
UiSMR4	0 to 7	Set to 0.
UCON	U0IRS	Select source of UART0 transmit interrupt.
	U1IRS	Select source of UART1 transmit interrupt.
	U0RRM	Set to 1 to use continuous receive mode.
	U1RRM	Set to 1 to use continuous receive mode.
	CLKMD0	Select the transmit/receive clock output pin when CLKMD1 is 1.
	CLKMD1	Set to 1 to output UART1 transmit/receive clock from two pins.
	RCSP	Set to 1 to separate the $\overline{CTS0}$/$\overline{RTS}$ signal of UART0.
	7	Set to 0.

i = 0 to 2, 5 to 7 j = 2, 5 to 7

Notes:
1. This table does not describe a procedure.
2. The TXD2 pin is N-channel open drain output. Nothing is assigned to the NCH bit in the U2C0 register. If necessary, set it to 0.

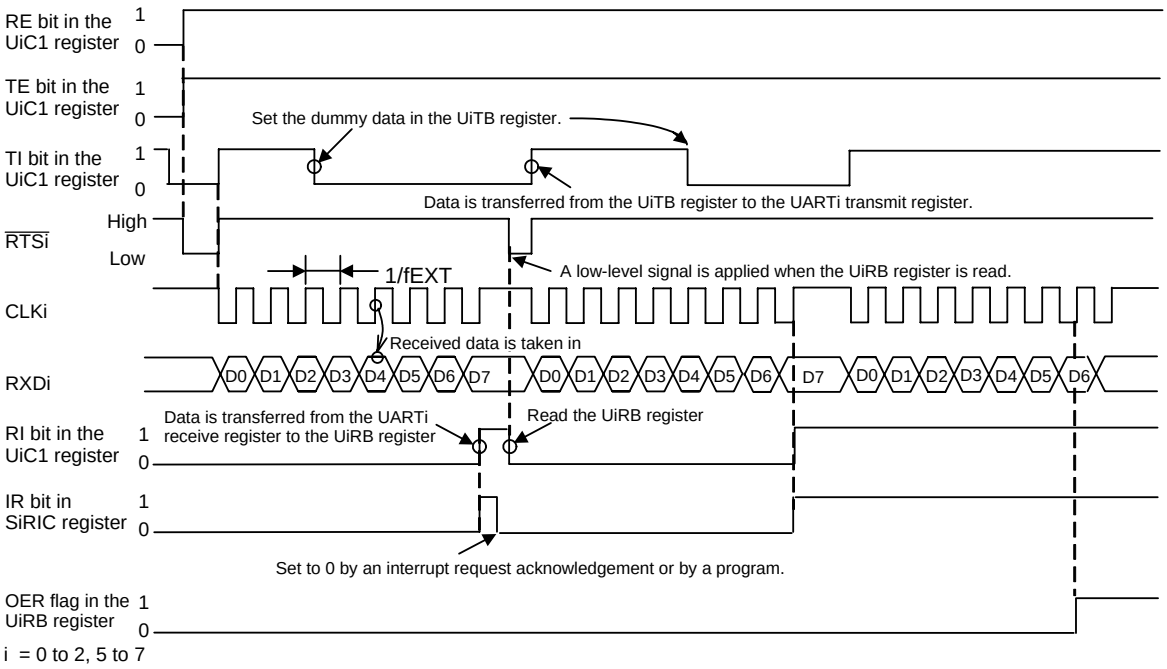
(1) Example of Transmit Timing (Internal Clock Selected)



The above timing diagram applies when the register bits are set as follows:
- The CKDIR bit in the UiMR register = 0 (internal clock)
- The CRD bit in the UiC0 register = 0 (CTS/RTS enabled), the CRS bit = 0 (CTS selected)
- The CKPOL bit in the UiC0 register = 0 (transmit data output at the falling edge and receive data taken in at the rising edge of the transmit/receive clock)
- The UiIRS bit in the UiC1 or UCON register = 0 (an interrupt request occurs when the UiTB register becomes empty)

$T_C = T_{CLK} = 2(n + 1)/fj$
fj: Frequency of UiBRG count source (f1SIO, f2SIO, f8SIO, f32SIO)
n: Value set to the UiBRG register

(2) Example of Receive Timing (External Clock Selected)



The above timing diagram applies to the case where the register bits are set as follows:
- The CKDIR bit in the UiMR register = 1 (external clock)
- The CRD bit in the UiC0 register = 0 (CTS/RTS enabled), the CRS bit = 1 (RTS selected)
- The CKPOL bit in the UiC0 register = 0 (transmit data output at the falling edge and receive data taken in at the rising edge of the transmit/receive clock)

fEXT: Frequency of the external clock

Make sure the following conditions are met when input to the CLKi pin before receiving data is high:
- The TE bit in the UiC1 register = 1 (transmit enabled)
- The RE bit in the UiC1 register = 1 (receive enabled)
- Write dummy data to the UiTB register

Figure 19.5 Transmit/Receive Operation during Clock Synchronous Serial I/O Mode

19.3.1.1 CLK Polarity Select Function

Use the CKPOL bit in the UiC0 register (i = 0 to 2, 5 to 7) to select the transmit/receive clock polarity. Figure 19.6 shows the Transmit/Receive Clock Polarity.

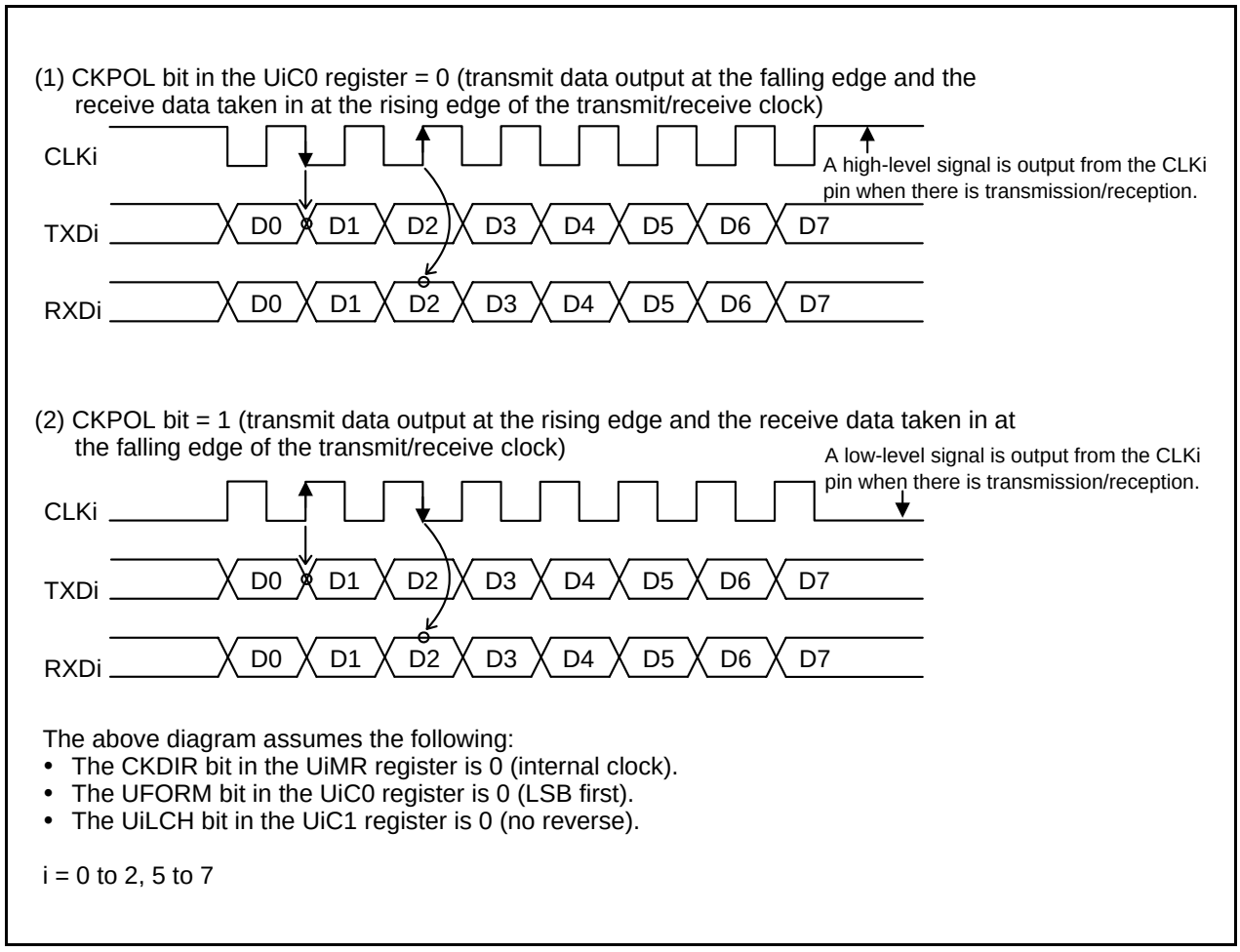
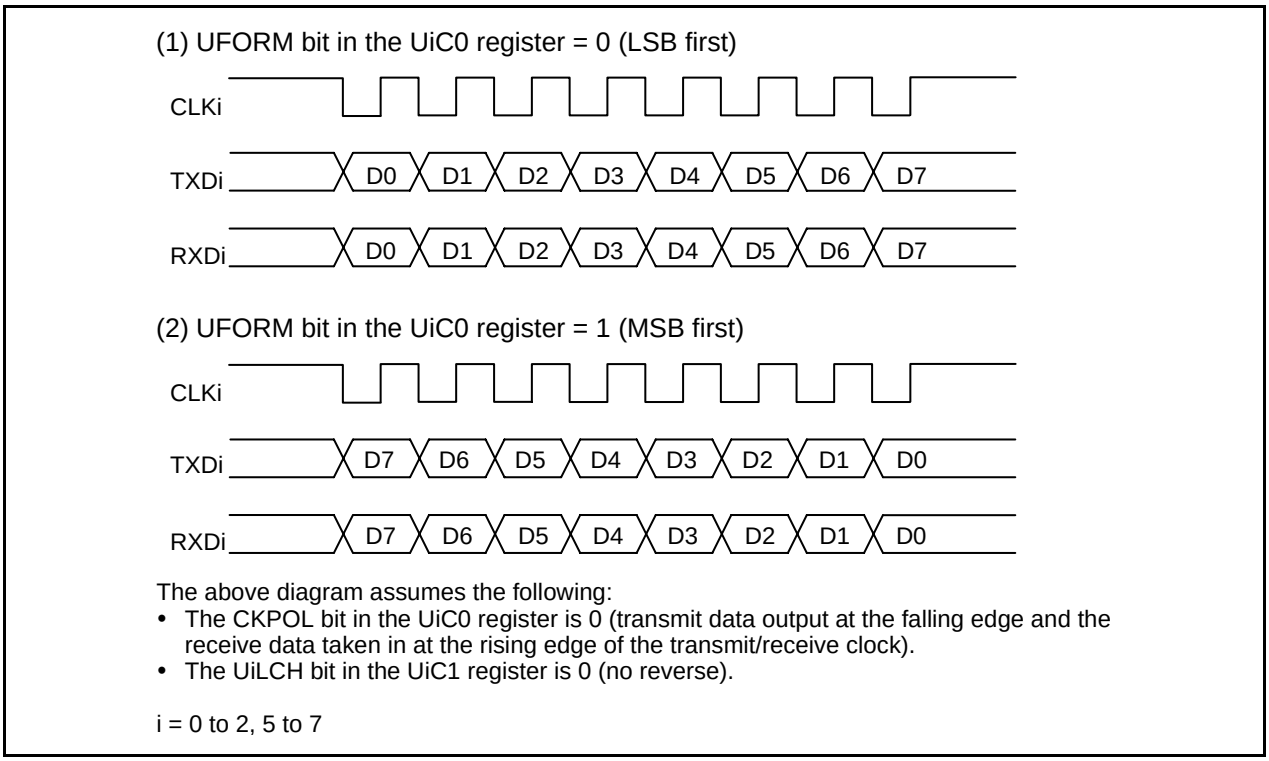


Figure 19.6 Transmit/Receive Clock Polarity

19.3.1.2 LSB First/MSB First Select Function

Use the UFORM bit in the UiC0 register (i = 0 to 2, 5 to 7) to select the bit order. Figure 19.7 shows the Bit Order.



The above diagram assumes the following:
- The CKPOL bit in the UiC0 register is 0 (transmit data output at the falling edge and the receive data taken in at the rising edge of the transmit/receive clock).
- The UiLCH bit in the UiC1 register is 0 (no reverse).

i = 0 to 2, 5 to 7

Figure 19.7 Bit Order

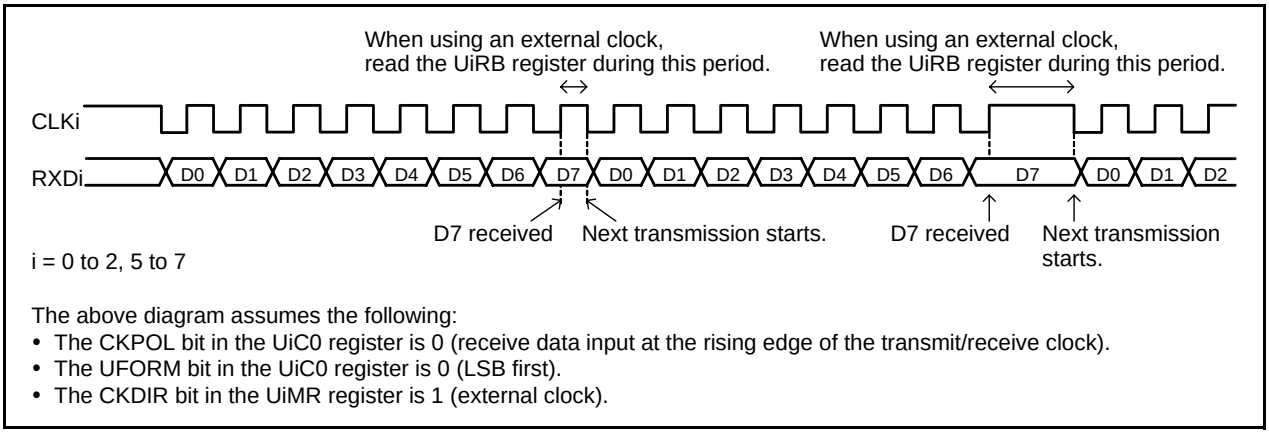
19.3.1.3 Continuous Receive Mode

In continuous receive mode, receive operation is enabled when the receive buffer register is read. It is not necessary to write dummy data to the transmit buffer register to enable receive operation in this mode. However, a dummy read of the receive buffer register is required when starting the operating mode.

When the UiRRM bit in the UiC1 or UCON register (i = 0 to 2, 5 to 7) is 1 (continuous receive mode), the TI bit in the UiC1 register is set to 0 (data present in the UiTB register) by reading the UiRB register. In this case (UiRRM bit = 1), do not write dummy data to the UiTB register by a program.

When using an external clock, read the UiRB register between receiving the eighth bit of data and starting the next transmission.

Figure 19.8 shows Operation Example in Continuous Receive Mode.



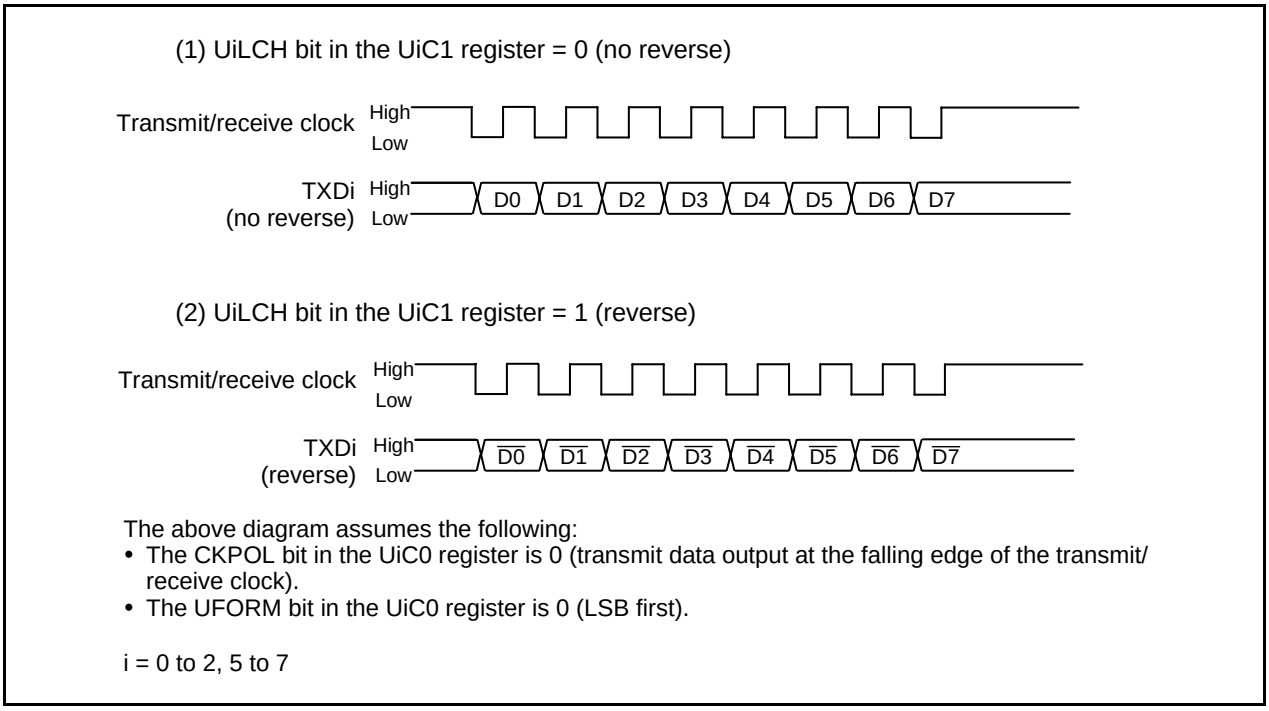
i = 0 to 2, 5 to 7

The above diagram assumes the following:
- The CKPOL bit in the UiC0 register is 0 (receive data input at the rising edge of the transmit/receive clock).
- The UFORM bit in the UiC0 register is 0 (LSB first).
- The CKDIR bit in the UiMR register is 1 (external clock).

Figure 19.8 Operation Example in Continuous Receive Mode

19.3.1.4 Serial Data Logic Switching Function

When the UiLCH bit in the UiC1 register (i = 0 to 2, 5 to 7) is 1 (reverse), data written to the UiTB register has its logic reversed before being transmitted. Similarly, the received data has its logic reversed when read from the UiRB register. Figure 19.9 shows Serial Data Logic.



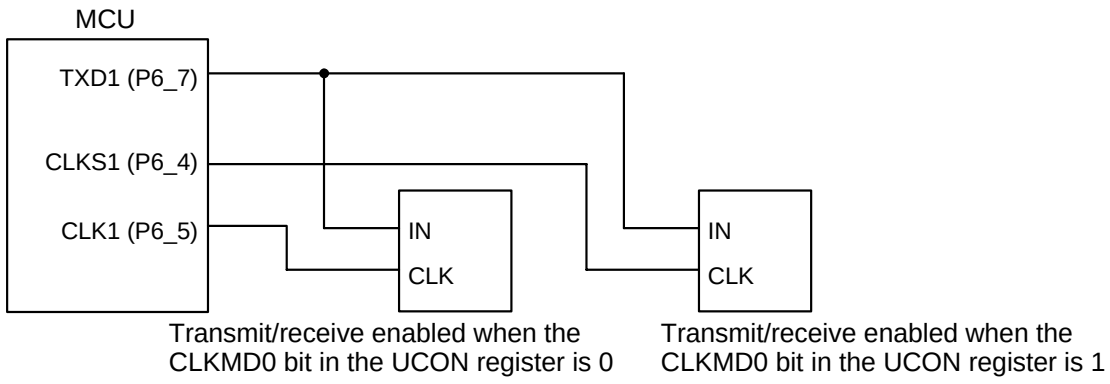
The above diagram assumes the following:
- The CKPOL bit in the UiC0 register is 0 (transmit data output at the falling edge of the transmit/ receive clock).
- The UFORM bit in the UiC0 register is 0 (LSB first).

i = 0 to 2, 5 to 7

Figure 19.9 Serial Data Logic

19.3.1.5 Transmit/Receive Clock Output from Multiple Pins (UART1)

Use bits CLKMD1 to CLKMD0 in the UCON register to select one of the two transmit/receive clock output pins (see Figure 19.10). This function can be used when the selected transmit/receive clock for UART1 is an internal clock.



The above diagram assumes the following:
- The CKDIR bit in the U1MR register is 0 (internal clock).
- The CLKMD1 bit in the UCON register is 1 (transmit/receive clock output from multiple pins).

Figure 19.10 Transmit/Receive Clock Output from Multiple Pins

19.3.1.6 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Function

The $\overline{\text{CTS}}$ function is used to start transmit/receive operation when a low-level signal is applied to the $\overline{\text{CTS}}$i/$\overline{\text{RTS}}$i (i = 0 to 2, 5 to 7) pin. Transmit/receive operation begins when input to the $\overline{\text{CTS}}$i/$\overline{\text{RTS}}$i pin becomes low. If the low-level signal is switched to high during a transmit or receive operation, the operation stops before the next data.

For the $\overline{\text{RTS}}$ function, the $\overline{\text{CTS}}$i/$\overline{\text{RTS}}$i pin outputs a low-level signal when the MCU is ready to receive. The output level becomes high on the first falling edge of the CLKi pin.

See Table 19.6 “Pin Functions in Clock Synchronous Serial I/O Mode (Multiple Transmit/Receive Clock Output Pin Function Not Selected)”.

19.3.1.7 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Separate Function (UART0)

This function separates $\overline{\text{CTS0}}$/$\overline{\text{RTS0}}$, outputs $\overline{\text{RTS0}}$ from the P6_0 pin, and inputs $\overline{\text{CTS0}}$ from the P6_4 pin. To use this function, set the register bits as shown below:

- The CRD bit in the U0C0 register = 0 (enable $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ of UART0)
- The CRS bit in the U0C0 register = 1 (output $\overline{\text{RTS}}$ of UART0)
- The CRD bit in the U1C0 register = 0 (enable $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ of UART1)
- The CRS bit in the U1C0 register = 0 (input $\overline{\text{CTS}}$ of UART1)
- The RCSP bit in the UCON register = 1 (inputs $\overline{\text{CTS0}}$ from the P6_4 pin)
- The CLKMD1 bit in the UCON register = 0 (CLKS1 not used)

Note that when using the $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ separate function, $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ of UART1 function cannot be used.

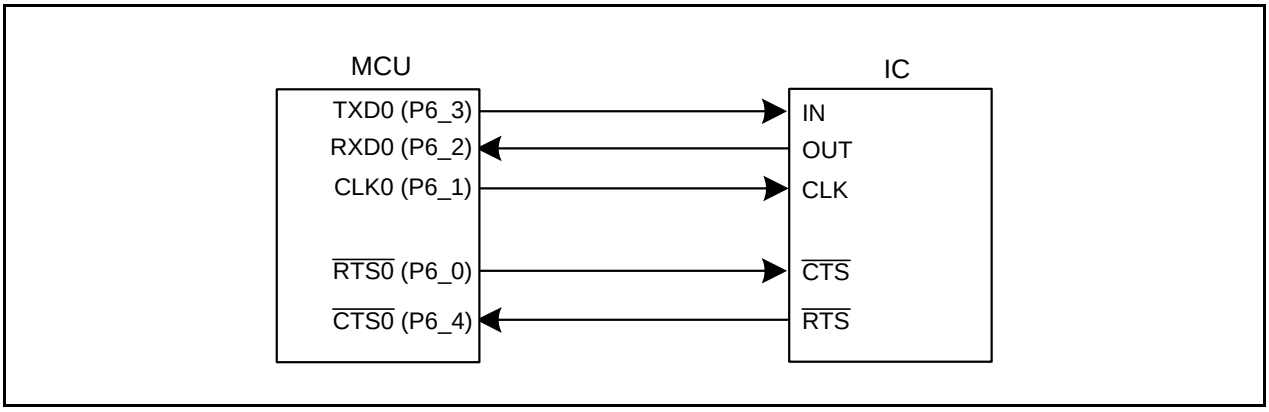


Figure 19.11 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Separate Function

19.3.1.8 Processing When Terminating Communication or When an Error Occurs

When communication is terminated in clock synchronous serial I/O mode, or when a communication error occurs, use the following procedure to reset communication:

(1) Set the TE bit in the UiC1 (i = 0 to 2, 5 to 7) 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 001b (clock synchronous serial I/O mode).
(4) Set the TE bit in the UiC1 register to 1 (transmission enabled) and the RE bit to 1 (reception enabled).

19.3.2 Clock Asynchronous Serial I/O (UART) Mode

The UART mode allows data to be transmitted/received after setting the desired bit rate and bit order. Table 19.9 lists the UART Mode Specifications.

Table 19.9 UART Mode Specifications

Item	Specification
Data format	• Character bits : Selectable from 7, 8, or 9 bits • Start bit : 1 bit • Parity bit : Selectable from odd, even, or none • Stop bit : Selectable from 1 bit or 2 bits
Transmit/receive clock	• The CKDIR bit in the UiMR register = 0 (internal clock): $\frac{fj}{16(n+1)}$ fj = f1SIO, f2SIO, f8SIO, f32SIO n: Setting value of UiBRG register 00h to FFh • CKDIR bit = 1 (external clock): $\frac{fEXT}{16(n+1)}$ fEXT: Input from CLKi pin n: Setting value of UiBRG register 00h to FFh
Transmission and reception control	Selectable from $\overline{CTS}$ function, $\overline{RTS}$ function, or $\overline{CTS}$/$\overline{RTS}$ function disabled
Transmission start conditions	To start transmission, satisfy the following requirements. • The TE bit in the UiC1 register = 1 (transmission enabled) • The TI bit in the UiC1 register = 0 (data present in the UiTB register) • If the $\overline{CTS}$ function is selected, input on the $\overline{CTSi}$ pin is low.
Reception start conditions	To start reception, satisfy the following requirements. • The RE bit in the UiC1 register = 1 (reception enabled) • Start bit detection.
Interrupt request generation timing	Transmit interrupt: One of the following can be selected: • The UiIRS bit in the UiC1 or UCON register = 0 (transmit buffer empty): When transferring data from the UiTB register to the UARTi transmit register (at start of transmission) • The UiIRS bit =1 (transmission completed): When the serial interface completes sending data from the UARTi transmit register Receive interrupt: • When transferring data from the UARTi receive register to the UiRB register (at completion of reception)
Error detection	• Overrun error (1) This error occurs if the serial interface starts receiving the next unit of data before reading the UiRB register and receives the bit before the last stop bit of the next unit of data. • Framing error This error occurs when the number of stop bits set is not detected. • Parity error This error occurs when the number of 1's of the parity bit and character bits does not match the set value of the PRY bit in the UiMR register. • Error sum flag This flag becomes 1 when an overrun, framing, or parity error occurs.
Selectable functions	• LSB first, MSB first selection Whether to start sending/receiving data beginning with bit 0 or beginning with bit 7 can be selected • Serial data logic switch This function reverses the logic of the transmit/receive data. The start and stop bits are not reversed. • TXD, RXD I/O polarity switch This function reverses the polarities of the TXD pin output and RXD pin input. The logic levels of all I/O data are reversed. • Separate $\overline{CTS}$/$\overline{RTS}$ pins (UART0) $\overline{CTS0}$ and $\overline{RTS0}$ are input/output from separate pins.

i = 0 to 2, 5 to 7

Note:

1. If an overrun error occurs, the receive data of the UiRB register will be undefined. The IR bit in the SiRIC register remains unchanged.

Table 19.10 lists I/O Pin Functions in UART Mode. Table 19.11 lists the P6_4 Pin Functions in UART Mode. Note that for a period from when the UARTi operating mode is selected to when transmission starts, the TXDi pin outputs a high-level signal. (If N-channel open drain output is selected, this pin becomes high-impedance.)
Table 19.12 lists the Registers Used and Settings in UART Mode.

Table 19.10 I/O Pin Functions in UART Mode

Pin Name	I/O	Function	Method of Selection
TXDi	Output	Serial data output	(High-level output only when receiving.)
RXDi	Input	Serial data input	Set the port direction bits sharing pins to 0.
	Input	Input port	Set the port direction bits sharing pins to 0. (can be used as an input port only when transmitting.)
CLKi	Input/ output	Input/output port	The CKDIR bit in the UiMR register = 0
	Input	Transmit/receive clock input	The CKDIR bit in the UiMR register = 1 Set the port direction bits sharing pins to 0.
$\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$	Input	$\overline{\text{CTS}}$ input	The CRD bit in the UiC0 register = 0 The CRS bit in the UiC0 register = 0 Set the port direction bits sharing pins to 0.
	Output	$\overline{\text{RTS}}$ output	The CRD bit in the UiC0 register = 0 The CRS bit in the UiC0 register = 1
	Input/ output	I/O port	The CRD bit in the UiC0 register = 1

i = 0 to 2, 5 to 7

Table 19.11 P6_4 Pin Functions in UART Mode

Pin Function	Bit Set Value				
	U1C0 register		UCON register		PD6 register
	CRD	CRS	RCSP	CLKMD1	PD6_4
P6_4	1	—	0	0	Input: 0, Output: 1
$\overline{\text{CTS1}}$	0	0	0	0	0
$\overline{\text{RTS1}}$	0	1	0	0	—
$\overline{\text{CTS0}}$ (1)	0	0	1	0	0

— indicates either 0 or1.
Note:
1. In addition to these settings, set the CRD bit in the U0C0 register to 0 ($\overline{\text{CTS0}}$/$\overline{\text{RTS0}}$ enabled) and the CRS bit in the U0C0 register to 1 ($\overline{\text{RTS0}}$ selected).

Table 19.12 Registers Used and Settings in UART Mode (1)

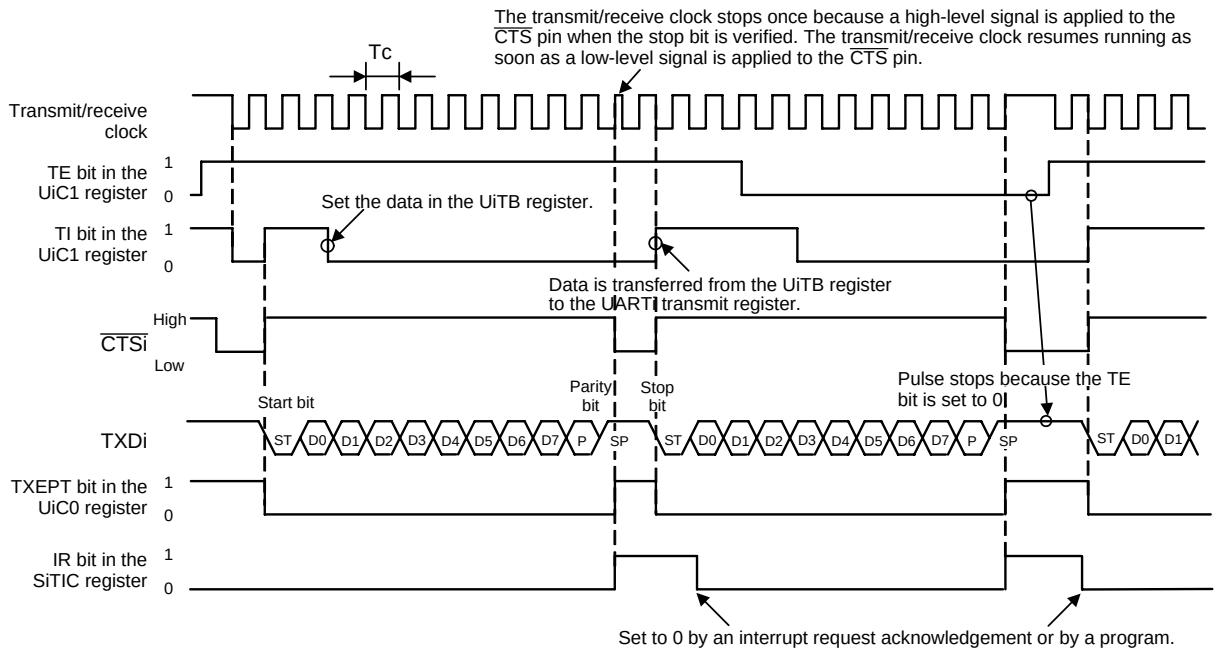
Register	Bits	Function
PCLKR	PCLK1	Select the count source for the UiBRG register.
PCLKSTP1	PCKSTP12	Set to 0 when using f1.
UiTB	0 to 8	Set transmission data. (2)
UiRB	0 to 8	Reception data can be read. (2, 4)
	11	When read, the read value is undefined.
	OER, FER, PER, SUM	Error flag
UiBRG	0 to 7	Set bit rate.
UiMR	SMD2 to SMD0	Set to 100b when character bit length is 7 bits.
		Set to 101b when character bit length is 8 bits.
		Set to 110b when character bit length is 9 bits.
	CKDIR	Select the internal clock or external clock.
	STPS	Select number of stop bits.
	PRY, PRYE	Select whether parity is included and whether odd or even.
	IOPOL	Select the TXD/RXD input/output polarity.
UiC0	CLK0, CLK1	Select the count source for the UiBRG register.
	CRS	If $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ is used, select which function to use.
	TXEPT	Transmit register empty flag
	CRD	Enable or disable the $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ function.
	NCH	Select TXDi pin output mode. (3)
	CKPOL	Set to 0.
	UFORM	LSB first or MSB first can be selected when character bit length is 8 bits. Set to 0 when character bit length is 7 or 9 bits.
UiC1	TE	Set to 1 to enable transmission.
	TI	Transmit buffer empty flag
	RE	Set to 1 to enable reception.
	RI	Reception complete flag
	UjIRS	Select source of UARTj transmit interrupt.
	UjRRM	Set to 0.
	UiLCH	Set to 1 to use reversed data logic.
	UiERE	Set to 0.
UiSMR	0 to 7	Set to 0.
UiSMR2	0 to 7	Set to 0.
UiSMR3	0 to 7	Set to 0.
UiSMR4	0 to 7	Set to 0.
UCON	U0IRS	Select source of UART0 transmit interrupt.
	U1IRS	Select source of UART1 transmit interrupt.
	U0RRM	Set to 0.
	U1RRM	Set to 0.
	CLKMD0	Invalid because CLKMD1 is 0
	CLKMD1	Set to 0.
	RCSP	Set to 1 to input $\overline{\text{CTS0}}$ signal of UART0 from the P6_4 pin.
	7	Set to 0.

i = 0 to 2, 5 to 7 j = 2, 5 to 7

Notes:

1. This table does not describe a procedure.
2. The bits used for transmit/receive data are as follows: Bits 0 to 6 when character bit length is 7 bits; bits 0 to 7 when character bit length is 8 bits; bits 0 to 8 when character bit length is 9 bits.
3. The TXD2 pin is an N-channel open drain output. Nothing is assigned to the NCH bit in the U2C0 register. If necessary, set it to 0.
4. The values of bits 7 and 8 are undefined when character bit length is 7 bits.
   The values of bit 8 is undefined when character bit length is 8 bits.

(1) 8-bit Data Transmit Timing (with a Parity Bit and One Stop Bit)

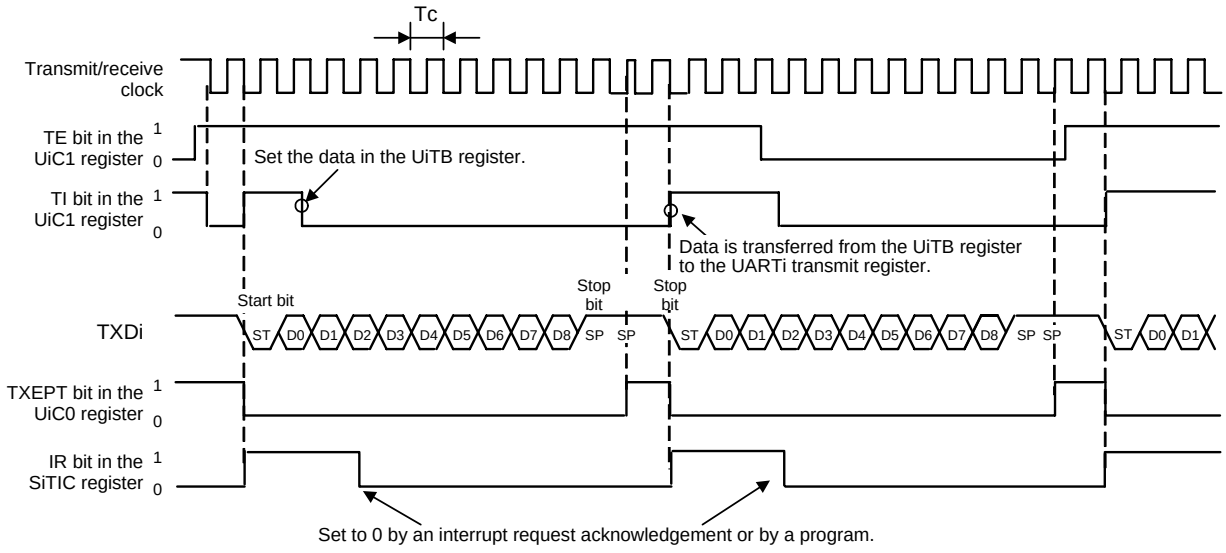


i = 0 to 2, 5 to 7

The above timing diagram applies when the register bits are set as follows:
- The PRYE bit in the UiMR register = 1 (parity enabled)
- The STPS bit in the UiMR register = 0 (one stop bit)
- The CRD bit in the UiC0 register = 0 (CTS/RTS enabled)
- The CRS bit in the UiC0 register = 0 (CTS selected)
- The UiIRS bit in the UiC1 or UCON register = 1 (an interrupt request occurs when transmit completed)

Tc = 16(n + 1)/fj or 16(n + 1)/fEXT
fj: Frequency of UiBRG count source (f1SIO, f2SIO, f8SIO, f32SIO)
fEXT: Frequency of UiBRG count source (external clock)
n: Value set to UiBRG

(2) 9-bit Data Transmit Timing (with No Parity Bit and Two Stop Bits)



i = 0 to 2, 5 to 7

The above timing diagram applies when the register bits are set as follows:
- The PRYE bit in the UiMR register = 0 (parity disabled)
- The STPS bit in the UiMR register = 1 (two stop bits)
- The CRD bit in the UiC0 register = 1 (CTS/RTS disabled)
- The UiIRS bit in the UiC1 or UCON register = 0 (an interrupt request occurs when transmit buffer becomes empty)

TC = 16(n + 1)/fj or 16(n + 1)/fEXT
fj: Frequency of UiBRG count source (f1SIO, f2SIO, f8SIO, f32SIO)
fEXT: Frequency of UiBRG count source (external clock)
n: Value set to UiBRG

Figure 19.12 Transmit Timing in UART Mode

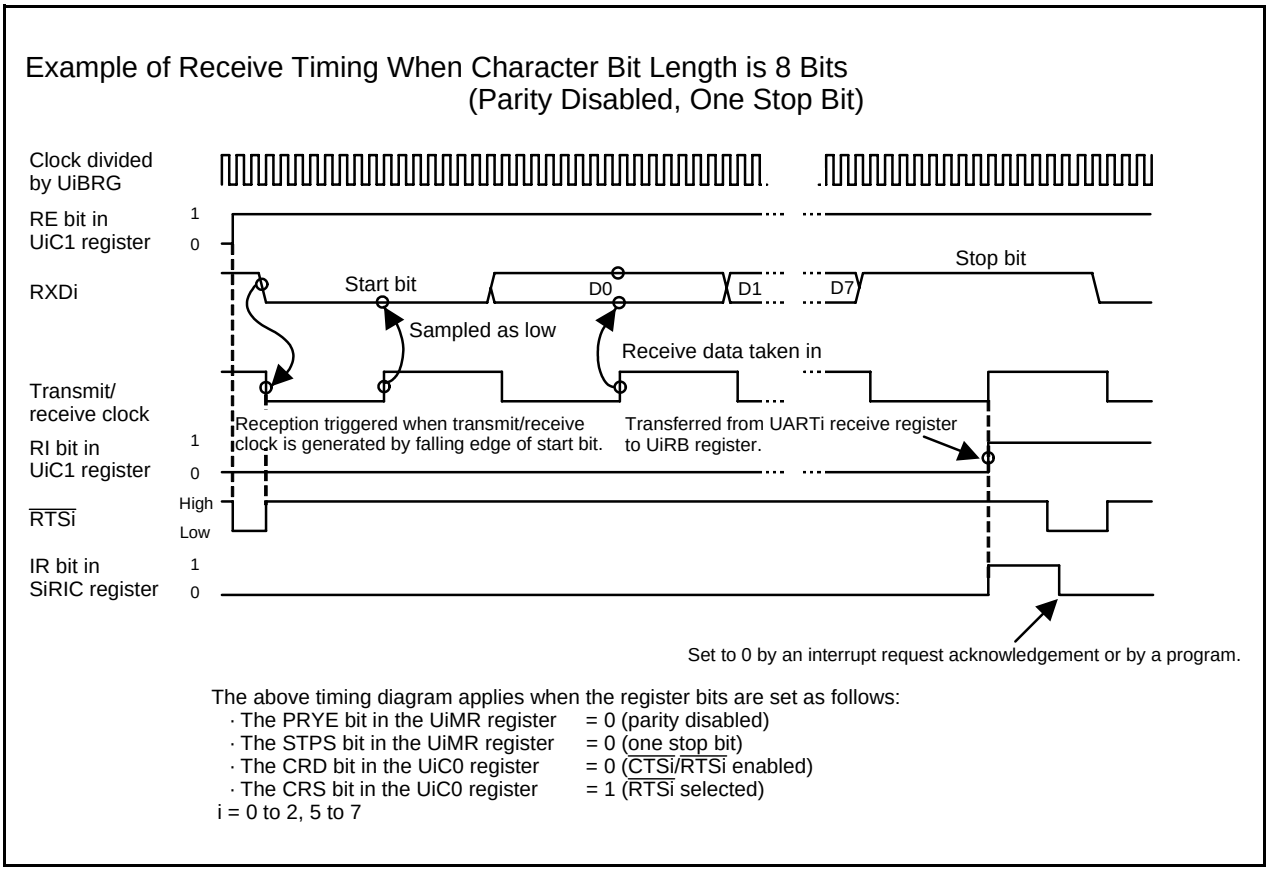


Figure 19.13 Receive Timing in UART Mode

19.3.2.1 Bit Rate

In UART mode, the frequency set by the UiBRG register (i = 0 to 2, 5 to 7) divided by 16 becomes the bit rate.

The setting value (n) of the UiBRG register is calculated by the following formula:

$$n = \frac{fj}{bitrate(bps) \times 16} - 1$$

fj = f1SIO, f2SIO, f8SIO, f32SIO

n = 00h to FFh

Table 19.13 lists Example Bit Rates and Settings.

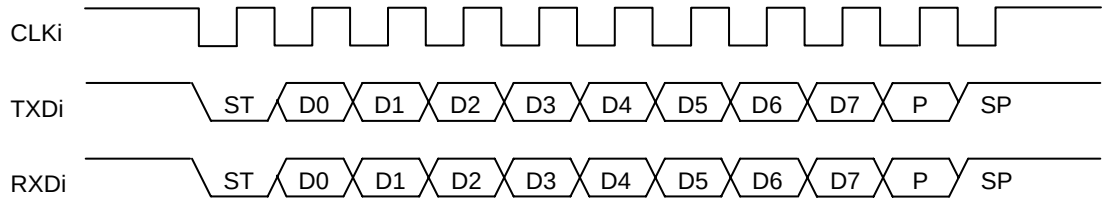
Table 19.13 Example Bit Rates and Settings

Bit Rate (bps)	Count Source of UiBRG	Peripheral Function Clock f1: 15.36 MHz	
		Set value of UiBRG: n	Bit rate (bps)
1200	f8SIO	99 (63h)	1200
2400	f8SIO	49 (31h)	2400
4800	f8SIO	24 (18h)	4800
9600	f1SIO	99 (63h)	9600
19200	f1SIO	49 (31h)	19200
38400	f1SIO	24 (18h)	38400

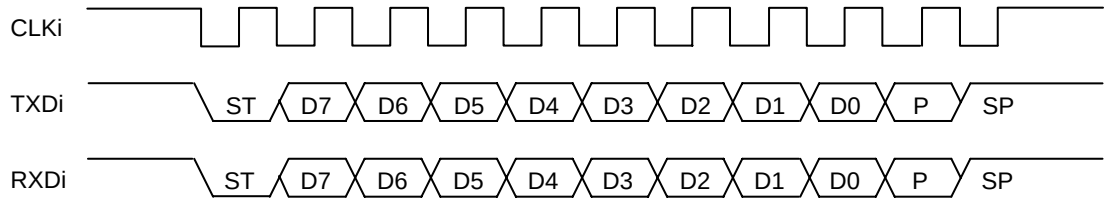
19.3.2.2 LSB First/MSB First Select Function

As shown in Figure 19.14, the bit order can be selected by setting the UFORM bit in the UiC0 register. This function is enabled when the character bit length is 8 bits.

(1) UFORM bit in the UiC0 register = 0 (LSB first)



(2) UFORM bit in the UiC0 register = 1 (MSB first)



ST: Start bit
P: Parity bit
SP: Stop bit
i = 0 to 2, 5 to 7

The above diagram assumes the following:

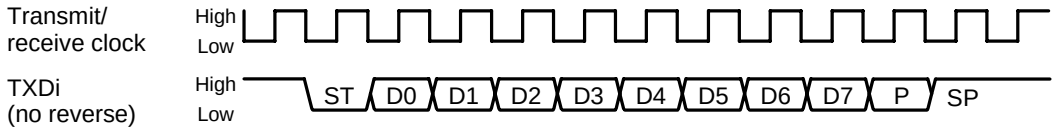
- The CKPOL bit in the UiC0 register is 0 (transmit data output at the falling edge and the receive data taken in at the rising edge of the transmit/receive clock).
- The UiLCH bit in the UiC1 register is 0 (no reverse).
- The STPS bit in the UiMR register is 0 (one stop bit).
- The PRYE bit in the UiMR register is 1 (parity enabled).

Figure 19.14 Bit Order

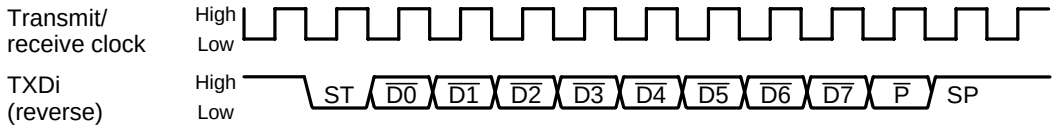
19.3.2.3 Serial Data Logic Switching Function

The logic of the data written to the UiTB register is reversed and then transmitted. Similarly, the reversed logic of the received data is read when the UiRB register is read. Figure 19.15 shows Serial Data Logic.

(1) UiLCH bit in the UiC1 register = 0 (no reverse)



(2) UiLCH bit in the UiC1 register = 1 (reverse)



ST : Start bit
P : Parity bit
SP : Stop bit
i = 0 to 2, 5 to 7

The above diagram assumes the following:
- The CKPOL bit in the UiC0 register is 0 (transmit data output at the falling edge of the transmit/receive clock).
- The UFORM bit in the UiC0 register is 0 (LSB first).
- The STPS bit in the UiMR register is 0 (one stop bit).
- The PRYE bit in the UiMR register is 1 (parity enabled).
- The IOPOL bit in the UiMR register is 0 (TXD, RXD I/O not reversed).

Figure 19.15 Serial Data Logic

19.3.2.4 TXD and RXD I/O Polarity Reverse Function

This function reverses the polarities of the TXDi pin output and RXDi pin input. The logic levels of all input/output data (including bits for start, stop, and parity) are reversed. Figure 19.16 shows TXD and RXD I/O Polarity Reversal.

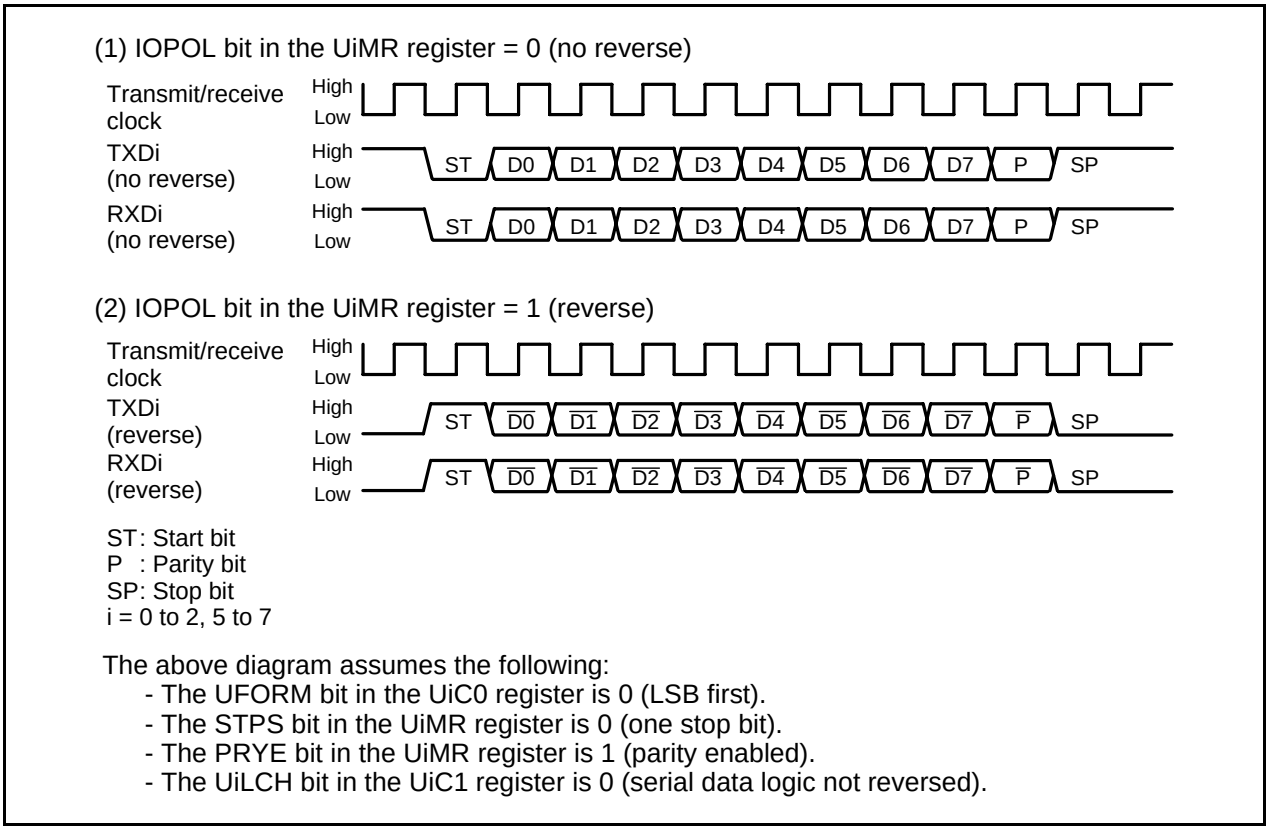


Figure 19.16 TXD and RXD I/O Polarity Reversal

19.3.2.5 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Function

The $\overline{\text{CTS}}$ function is used to start transmit operation when a low-level signal is applied to the $\overline{\text{CTSi}}/\overline{\text{RTSi}}$ (i = 0 to 2, 5 to 7) pin. Transmit operation begins when input to the $\overline{\text{CTSi}}/\overline{\text{RTSi}}$ pin becomes low. If the low-level signal is switched to high during a transmit operation, the operation stops after the ongoing transmit/receive operation is completed.

When the $\overline{\text{RTS}}$ function is used, the $\overline{\text{CTSi}}/\overline{\text{RTSi}}$ pin outputs a low-level signal when the MCU is ready to receive. The output level becomes high when a start bit is detected.

See Table 19.10 “I/O Pin Functions in UART Mode”.

19.3.2.6 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Separate Function (UART0)

This function separates $\overline{\text{CTS0}}$ and $\overline{\text{RTS0}}$, outputs $\overline{\text{RTS0}}$ from the P6_0 pin, and inputs $\overline{\text{CTS0}}$ from the P6_4 pin. To use this function, set the register bits as shown below.

- The CRD bit in the U0C0 register = 0 (enable $\overline{\text{CTS}}/\overline{\text{RTS}}$ of UART0)
- The CRS bit in the U0C0 register = 1 (output $\overline{\text{RTS}}$ of UART0)
- The CRD bit in the U1C0 register = 0 (enable $\overline{\text{CTS}}/\overline{\text{RTS}}$ of UART1)
- The CRS bit in the U1C0 register = 0 (input $\overline{\text{CTS}}$ of UART1)
- The RCSP bit in the UCON register = 1 (inputs $\overline{\text{CTS0}}$ from the P6_4 pin)
- The CLKMD1 bit in the UCON register = 0 (CLKS1 not used)

Note that when using the $\overline{\text{CTS}}/\overline{\text{RTS}}$ separate function, $\overline{\text{CTS}}/\overline{\text{RTS}}$ function of UART1 cannot be used.

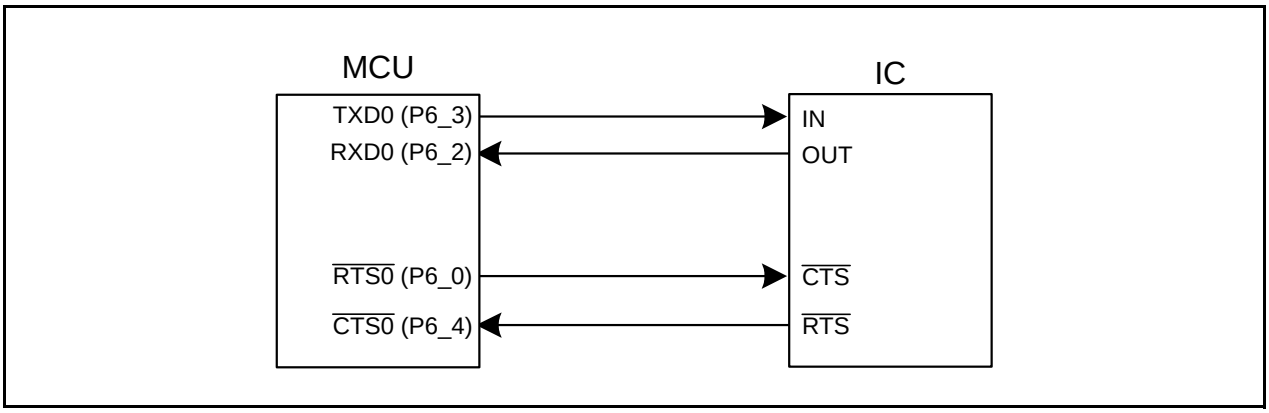


Figure 19.17 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Separate Function

19.3.3 Special Mode 1 (I^2C Mode)

I^2C mode supports the simplified I^2C interface. Table 19.14 lists the specifications of I^2C mode. Table 19.16 and Table 19.17 list the registers used in I^2C mode and the register settings. Table 19.18 lists the I^2C Mode Specifications. Figure 19.18 shows I^2C Mode Block Diagram.

As shown in Table 19.18, the MCU is placed in I^2C mode by setting bits SMD2 to SMD0 in the UiMR register to 010b and the IICM bit in the UiSMR register to 1. Because SDAi transmit output includes a delay circuit, SDAi output changes states after SCLi becomes low and remains stable.

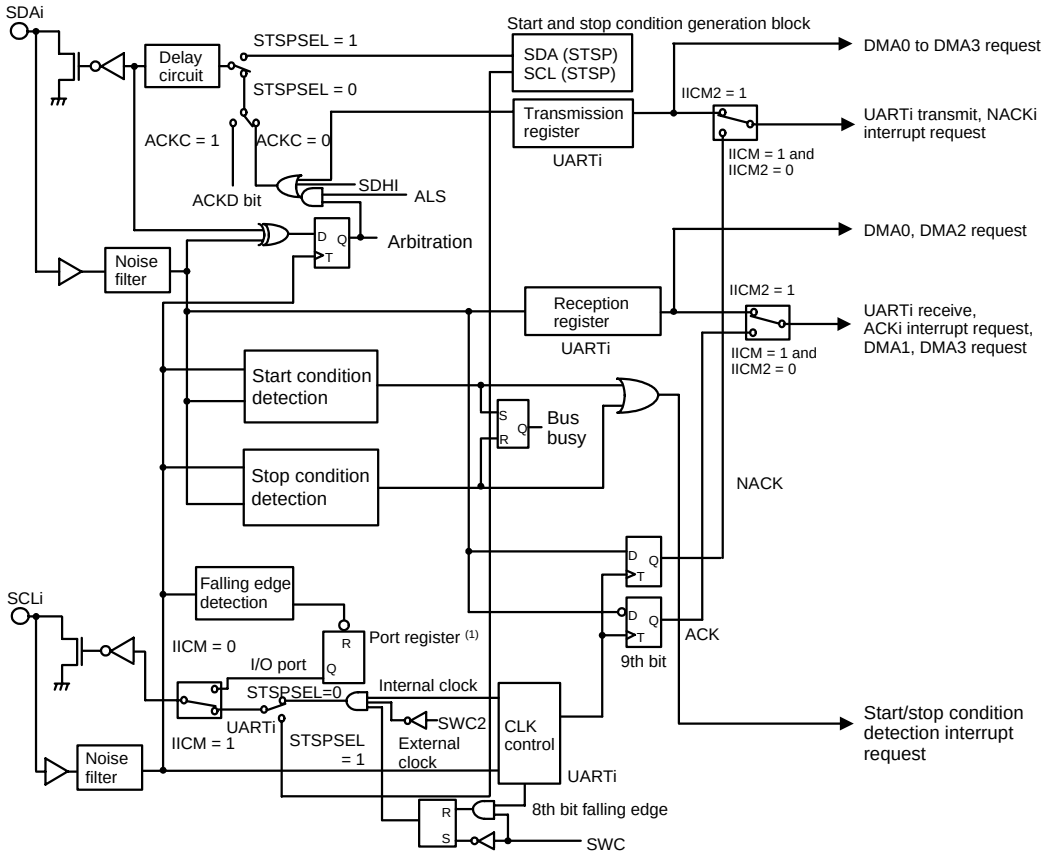
Table 19.14 I^2C Mode Specifications

Item	Specification
Data format	Character bit length: 8 bits
Transmit/receive clock	• Master mode CKDIR bit in the UiMR register = 0 (internal clock): $\frac{fj}{2(n+1)}$ fj = f1SIO, f2SIO, f8SIO, f32SIO n = setting value of the UiBRG register 03h to FFh • Slave mode CKDIR bit = 1 (external clock): Input from the SCLi pin
Transmission start conditions	To start transmission, satisfy the following requirements. (1) • The TE bit in the UiC1 register = 1 (transmission enabled) • The TI bit in the UiC1 register = 0 (data present in UiTB register)
Reception start conditions	To start reception, satisfy the following requirements. (1) • The RE bit in the UiC1 register = 1 (reception enabled) • The TE bit in the UiC1 register = 1 (transmission enabled) • The TI bit in the UiC1 register = 0 (data present in the UiTB register)
Interrupt request generation timing	When a start condition, stop condition, ACK (acknowledge), or NACK (not-acknowledge) is detected.
Error detection	Overrun error (2) This error occurs if the serial interface starts receiving the next unit of data before reading the UiRB register and receives the eighth bit of the unit of next data.
Selectable functions	• Arbitration lost Timing that the ABT bit in the UiRB register is updated can be selected. • SDAi digital delay No digital delay or a delay of 2 to 8 UiBRG count source clock cycles can be selected. • Clock phase setting With or without clock delay can be selected.

i = 0 to 2, 5 to 7

Notes:

1. These requirements do not have to be set in any particular order. When transmission/reception is started as a slave and the TXEPT bit in the UiC0 register is 1 (no data present in transmit register), meet the last requirement when the external clock is high.
2. If an overrun error occurs, the received data of the UiRB register will be undefined.



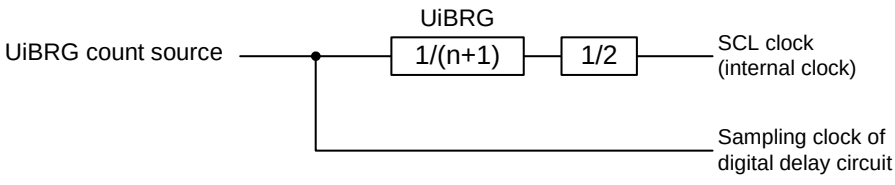
This diagram applies when bits SMD2 to SMD0 in the UiMR register are 010b and the IICM bit in the UiSMR register is 1.

IICM: Bit in the UiSMR register
IICM2, SWC, ALS, SWC2, SDHI: Bits in the UiSMR2 register
STSPSEL, ACKD, ACKC: Bits in the UiSMR4 register

i = 0 to 2, 5 to 7

Note:
1. If the IICM bit is 1, the pin can be read even when the port direction bit corresponding to the SCLi pin is 1 (output mode).

Figure 19.18 I²C Mode Block Diagram



n: UiBRG register setting value

Figure 19.19 Internal Clock Configuration

Table 19.15 I/O Pin Functions in I²C Mode

Pin Name	I/O	Function
SCLi (1, 2)	Input/output	Clock input or output
SDAi (1, 2)	Input/output	Data input or output

Notes:
1. Set the port direction bits sharing pins to 0.
2. Pins CLKi and $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$ are not used (they can be used as I/O ports).

Table 19.16 Registers Used and Settings in I²C Mode (1/2) (1)

Register	Bits	Function	
		Master	Slave
PCLKR	PCLK1	Select the count source for the UiBRG register.	Select the count source for the UiBRG register.
PCLKSTP1	PCKSTP12	Set to 0 when using f1.	Set to 0 when using f1.
UiTB	0 to 7	When transmitting, set the transmission data. When receiving, set FFh.	When transmitting, set the transmission data. When receiving, set FFh.
	8	When transmitting, set to 1. When receiving, set the value in the ACK bit.	When transmitting, set to 1. When receiving, set the value in the ACK bit.
UiRB	0 to 7	Reception data can be read.	Reception data can be read.
	8	ACK or NACK is set in this bit.	ACK or NACK is set in this bit.
	ABT	Arbitration lost detection flag	Disabled
	OER	Overrun error flag	Overrun error flag
	13 to 15	When read, the read value is undefined.	When read, the read value is undefined.
UiBRG	0 to 7	Set a bit rate.	Disabled
UiMR	SMD2 to SMD0	Set to 010b.	Set to 010b.
	CKDIR	Set to 0.	Set to 1.
	4 to 6	Set to 0.	Set to 0.
	IOPOL	Set to 0.	Set to 0.
UiC0	CLK1, CLK0	Select the count source for the UiBRG register.	Disabled
	CRS	Disabled because CRD is 1	Disabled because CRD is 1
	TXEPT	Transmit register empty flag	Transmit register empty flag
	CRD (3)	Set to 1.	Set to 1.
	NCH	Set to 1. (2)	Set to 1. (2)
	CKPOL	Set to 0.	Set to 0.
	UFORM	Set to 1.	Set to 1.
UiC1	TE	Set to 1 to enable transmission.	Set to 1 to enable transmission.
	TI	Transmit buffer empty flag	Transmit buffer empty flag
	RE	Set to 1 to enable reception.	Set to 1 to enable reception.
	RI	Reception complete flag	Reception complete flag
	UjIRS	Set to 1.	Set to 1.
	UjRRM, UiLCH, UiERE	Set to 0.	Set to 0.
UiSMR	IICM	Set to 1.	Set to 1.
	ABC	Select the timing that arbitration lost is detected.	Disabled
	BBS	Bus busy flag	Bus busy flag
	3 to 7	Set to 0.	Set to 0.

i = 0 to 2, 5 to 7 j = 2, 5 to 7

Notes:
1. This table does not describe a procedure.
2. The TXD2 pin is N-channel open drain output. Nothing is assigned to the NCH bit in the U2C0 register. If necessary, set it to 0.
3. When using UART1 in I²C mode, to enable the $\overline{CTS}/\overline{RTS}$ separate function of UART0, set the CRD bit in the U1C0 register to 0 ($\overline{CTS}/\overline{RTS}$ enabled) and the CRS bit to 0 ($\overline{CTS}$ input).

Table 19.17 Registers Used and Settings in I²C Mode (2/2) (1)

Register	Bits	Function	
		Master	Slave
UiSMR2	IICM2	See Table 19.18 "I²C Mode Functions".	See Table 19.18 "I²C Mode Functions".
	CSC	Set to 1 to enable clock synchronization.	Set to 0.
	SWC	Set to 1 to fix SCLi output to low after receiving the eighth bit of the clock.	Set to 1 to fix SCLi output to low after receiving the eighth bit of the clock.
	ALS	Set to 1 to stop SDAi output when arbitration lost is detected.	Set to 0.
	STAC	Set to 0.	Set to 1 to initialize UARTi at start condition detection.
	SWC2	Set to 1 to forcibly pull SCLi output low.	Set to 1 to forcibly pull SCLi output low.
	SDHI	Set to 1 to disable SDAi output.	Set to 1 to disable SDAi output.
	7	Set to 0.	Set to 0.
UiSMR3	0, 2, 4 NODC	Set to 0.	Set to 0.
	CKPH	Set to 1.	Set to 1.
	DL2 to DL0	Set the amount of SDAi digital delay.	Set the amount of SDAi digital delay.
UiSMR4	STAREQ	Set to 1 to generate start condition.	Set to 0.
	RSTAREQ	Set to 1 to generate restart condition.	Set to 0.
	STPREQ	Set to 1 to generate stop condition.	Set to 0.
	STSPSEL	Set to 1 to output each condition.	Set to 0.
	ACKD	Select ACK or NACK.	Select ACK or NACK.
	ACKC	Set to 1 to output ACK data.	Set to 1 to output ACK data.
	SCLHI	Set to 1 to stop SCLi output when stop condition is detected.	Set to 0.
	SWC9	Set to 0.	Set to 1 to set the SCLi to remain low at the falling edge of the ninth bit of clock.
UCON	U0IRS	Set to 1.	Set to 1.
	U1IRS	Set to 1.	Set to 1.
	U0RRM	Set to 0.	Set to 0.
	U1RRM	Set to 0.	Set to 0.
	CLKMD0	Set to 0.	Set to 0.
	CLKMD1	Set to 0.	Set to 0.
	RCSP	Set to 0.	Set to 0.
	7	Set to 0.	Set to 0.

i = 0 to 2, 5 to 7 j = 2, 5 to 7

Note:

1. This table does not describe a procedure.

In I2C mode, functions and timings vary depending on the combination of the IICM2 bit in the UiSMR2 register and CKPH bit in the UiSMR3 register. Figure 19.20 shows Transfer to UiRB Register and Interrupt Timing. See Figure 19.20 for the timing of transferring to the UiRB register, the bit position of the data stored in the UiRB register, types of interrupts, interrupt requests, and DMA request generation timing.

Table 19.18 "I2C Mode Functions" lists comparison of other functions in clock synchronous serial I/O mode with I2C mode.

Table 19.18 I2C Mode Functions

Function	Clock Synchronous Serial I/O Mode (SMD2 to SMD0 = 001b, IICM = 0)	I2C Mode (SMD2 to SMD0 = 010b, IICM = 1)	
		IICM2 = 0 (NACK/ACK Interrupt)	IICM2 = 1 (UART Transmit/Receive Interrupt)
		CKPH = 1 (Clock delay)	CKPH = 1 (Clock delay)
Start and stop condition detect interrupts (3)	—	Start condition detection or stop condition detection (See Figure 19.22 "STSPSEL Bit Functions")	
Transmission, NACK interrupt (2, 3)	UARTi transmission Transmission started or completed (selected by UiIRS)	No acknowledgment detection (NACK) Rising edge of SCLi 9th bit	UARTi transmission Falling edge of SCLi next to the 9th bit
Reception, ACK interrupt (2, 3)	UARTi reception When 8th bit received CKPOL = 0 (rising edge) CKPOL = 1 (falling edge)	Acknowledgment detection (ACK) Rising edge of SCLi 9th bit	UARTi reception Falling edge of SCLi 9th bit
Timing for transferring data from UART reception shift register to UiRB register	CKPOL = 0 (rising edge) CKPOL = 1 (falling edge)	Rising edge of SCLi 9th bit	Falling edges of the 8th bit of SCLi and rising edges of the 9th bit of SCLi
UARTi transmission output delay	Not delayed	Delayed	
Read RXDi and SCLi pin levels	Possible when the corresponding port direction bit = 0	Always possible no matter how the corresponding port direction bit is set	
Initial value of TXDi and SDAi outputs	CKPOL = 0 (high) CKPOL = 1 (low)	The value set in the port register before setting I2C mode (1)	
Initial and end values of SCLi	—	Low	Low
DMA1, DMA3 Factor (2)	UARTi reception	Acknowledgment detection (ACK)	UARTi reception Falling edge of SCLi 9th bit
Read received data	1st to 8th bits of the received data are stored in bits 0 to 7 in the UiRB register.	1st to 8th bits of the received data are stored in bits 7 to 0 in the UiRB register.	When reading by reception interrupt, 1st to 7th bits of the received data are stored in bits 6 to 0 in the UiRB register. 8th bit is stored to bit 8 in the UiRB register. When reading by transmission interrupt, 1st to 8th bits are stored to bits 7 to 0 in the UiRB register.

i = 0 to 2, 5 to 7
SMD2 to SMD0: Bits in the UiMR register
CKPOL: Bit in the UiC0 register
IICM: Bit in the UiSMR register
IICM2: Bit in the UiSMR2 register
CKPH: Bit in the UiSMR3 register
Notes:

1. Set the initial value of SDAi output while bits SMD2 to SMD0 in the UiMR register are 000b (serial interface disabled).
2. See Figure 19.20 "Transfer to UiRB Register and Interrupt Timing".
3. The procedure to change interrupt sources is as follows:
   (1) Disable the interrupt to be changed the source.
   (2) Change the source of interrupt.
   (3) Set the IR bit in the interrupt control register of that interrupt to 0 (no interrupt requested).
   (4) Set bits ILVL2 to ILVL0 in the interrupt control register of that interrupt.

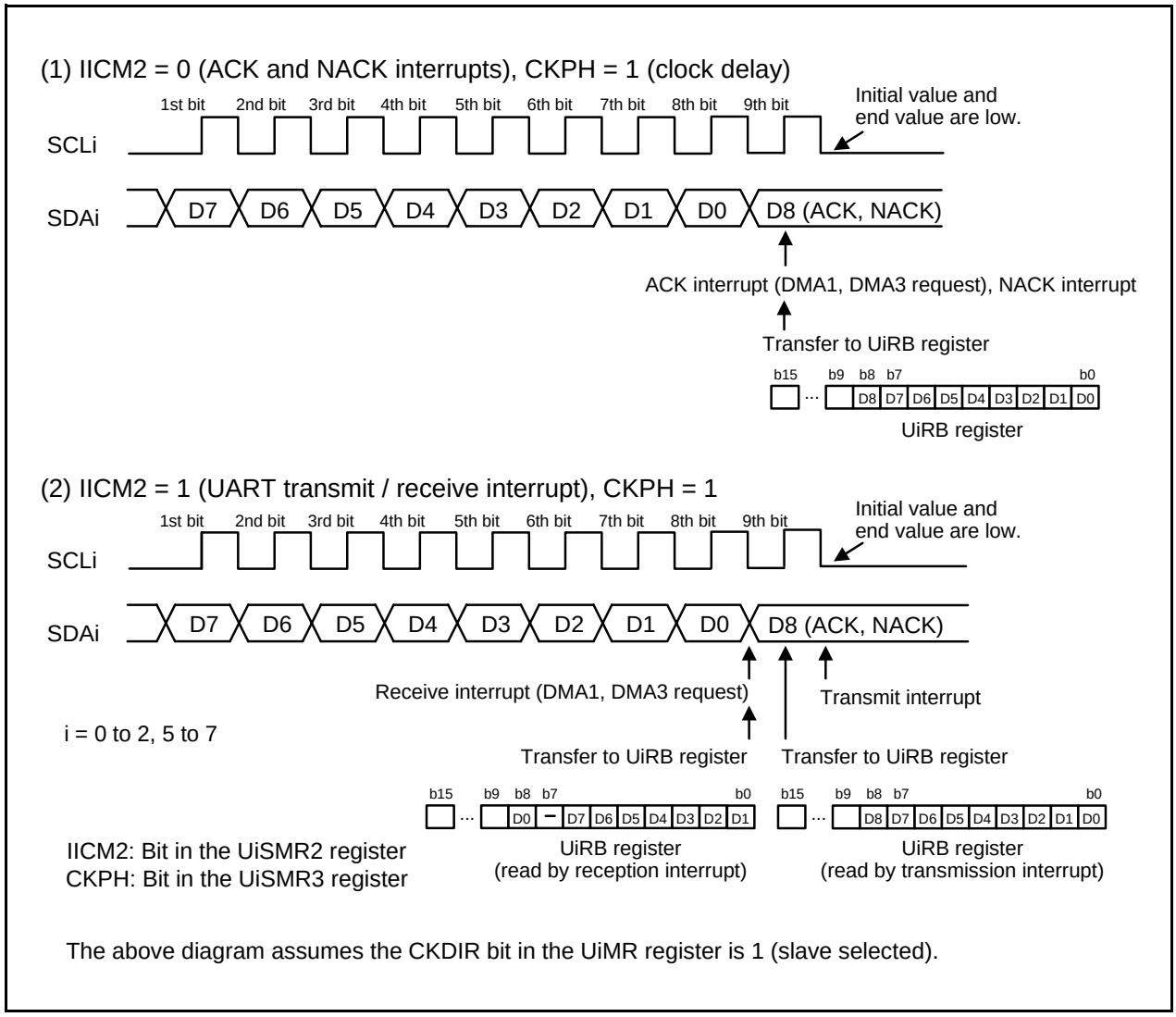


Figure 19.20 Transfer to UiRB Register and Interrupt Timing

19.3.3.1 Detecting Start and Stop Conditions

Start and stop conditions are detected by their respective detectors.
Whether a start or a stop condition has been detected is determined.
This function determines whether a start or stop condition has been detected.
A start condition detect interrupt request is generated when the SDAi pin changes state from high to low while the SCLi pin is in a high state. A stop condition detect interrupt request is generated when the SDAi pin changes state from low to high while the SCLi pin is in a high state.
To detect a start or stop condition, both the set-up and hold times require at least six cycles of the BRGi count source as shown in Figure 19.21. To meet the condition for the Fast-mode specification, the BRGi count source must be at least 10 MHz.

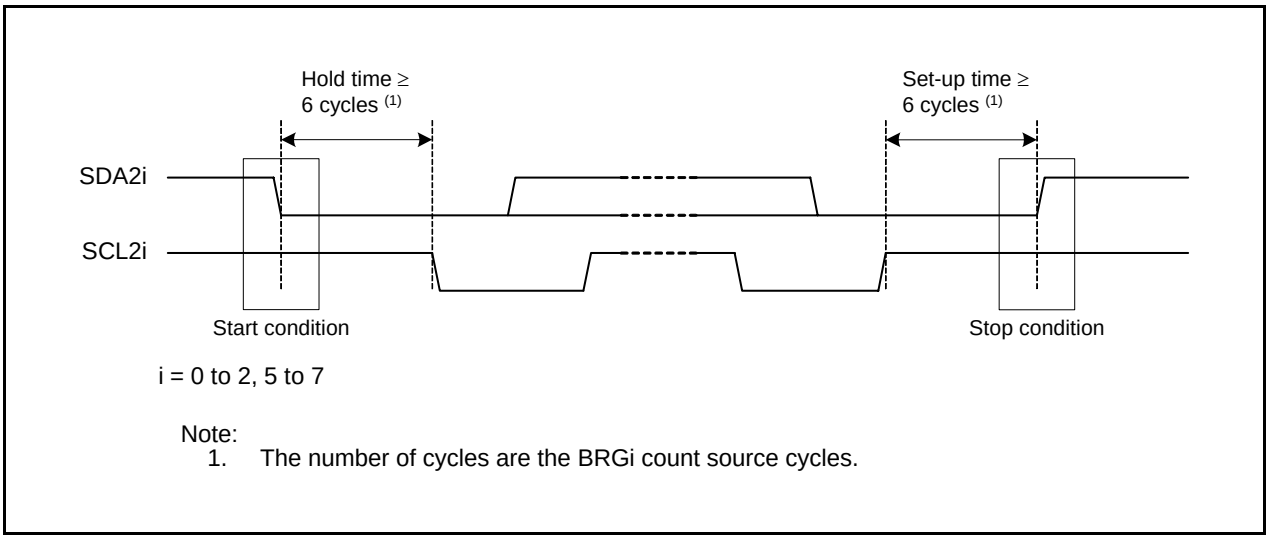


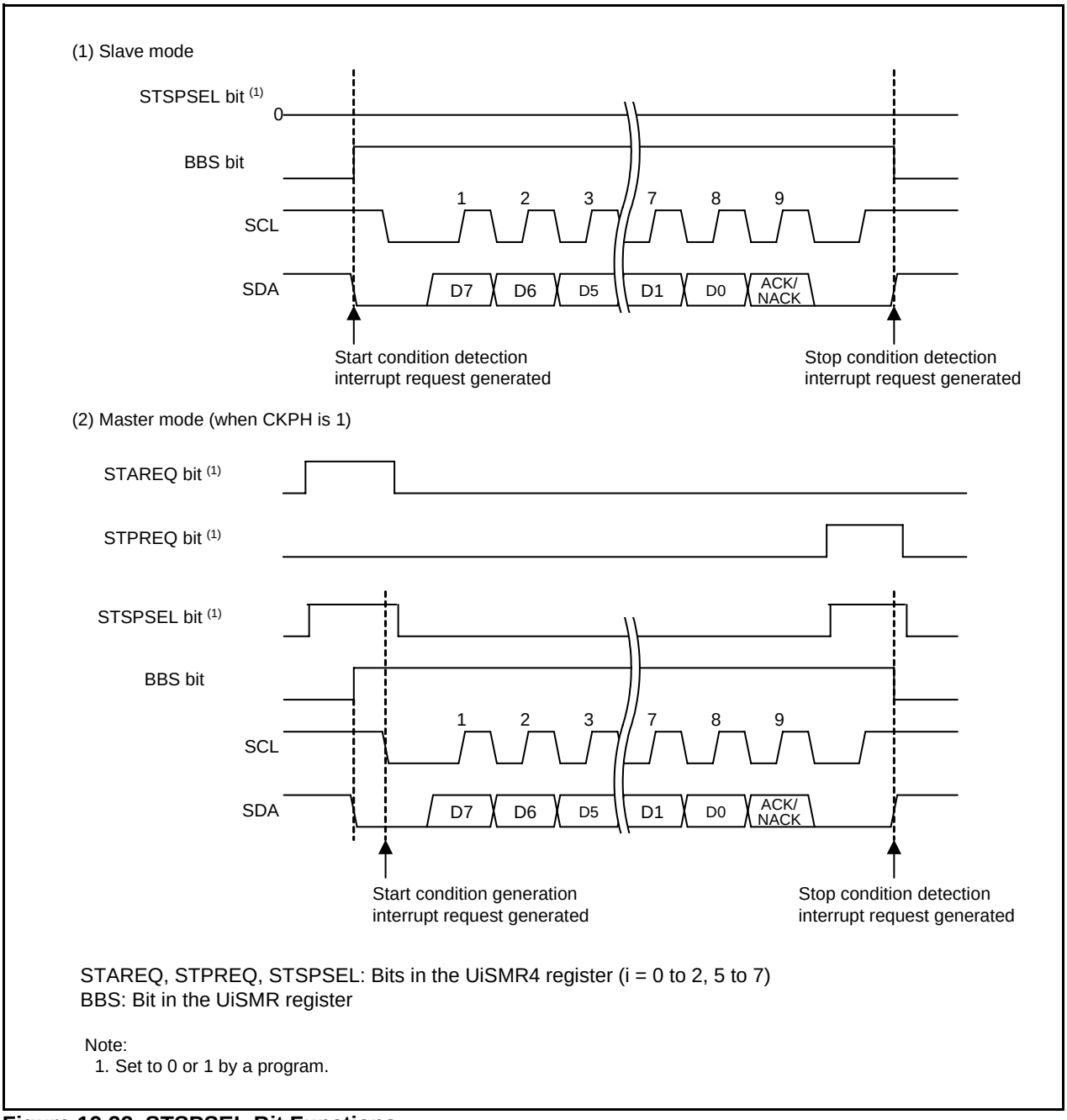
Figure 19.21 Detecting Start and Stop Conditions

19.3.3.2 Generating Start and Stop Conditions

A start condition is generated by setting the STAREQ bit in the UiSMR4 register (i = 0 to 2, 5 to 7) to 1 (start).
A restart condition is generated by setting the RSTAREQ bit in the UiSMR4 register to 1 (start).
A stop condition is generated by setting the STPREQ bit in the UiSMR4 register to 1 (start).
The output procedure is as follows:
(1) Set the STAREQ bit, RSTAREQ bit, or STPREQ bit to 1 (start).
(2) Set the STSPSEL bit in the UiSMR4 register to 1 (output).
The functions of the STSPSEL bit are shown in Table 19.19 and Figure 19.22.

Table 19.19 STSPSEL Bit Functions

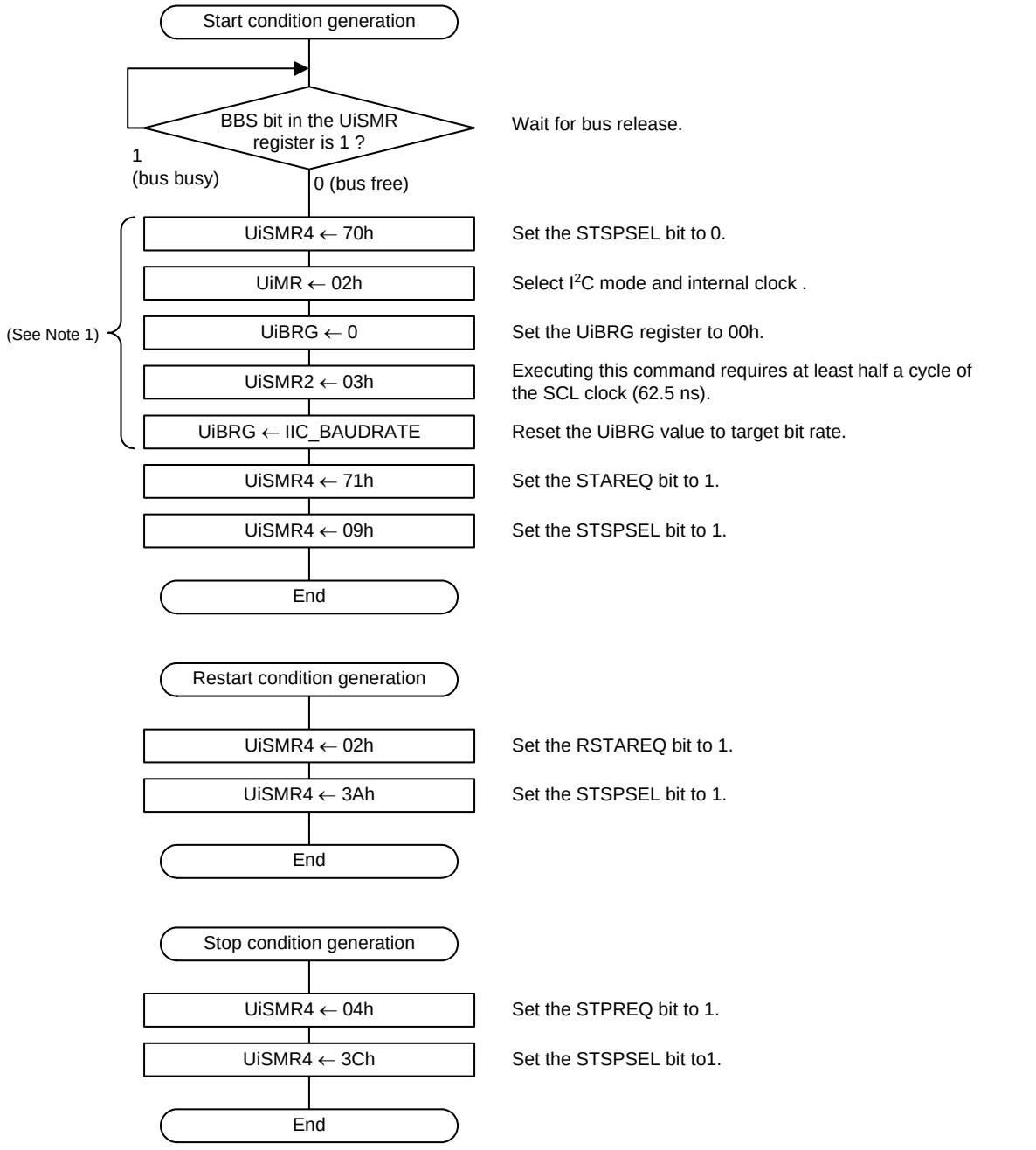
Function	STSPSEL = 0	STSPSEL = 1
Output of pins SCLi and SDAi	Output of transmit/receive clock and data Output of start/stop condition is accomplished by a program using ports (not automatically generated in hardware).	Output of a start/stop condition according to bits STAREQ, RSTAREQ, and STPREQ
Start/stop condition interrupt request generation timing	Detection of start/stop condition	Completion of start/stop condition generation



STAREQ, STPREQ, STSPSEL: Bits in the UiSMR4 register (i = 0 to 2, 5 to 7)
BBS: Bit in the UiSMR register

Note:
1. Set to 0 or 1 by a program.

Figure 19.22 STSPSEL Bit Functions



The above assumes the following:
XIN = 16 MHz, main clock divided by 1 (no division), UiBRG count source = f1

Note:
1. After a stop condition is generated, when generating the next start condition, after setting the STSPSEL bit in the UiSMR4 register to 0 and waiting at least half a cycle of the SCL clock, then set the STAREQ bit to 1.

Technical update number: TN-16C-130A/EA

Figure 19.23 Register Setting Procedures for Condition Generation

19.3.3.3 Arbitration

Unmatching of the transmit data and SDAi pin input data is checked in synchronization with the rising edge of SCLi. Use the ABC bit in the UiSMR register to select the point at which the ABT bit in the UiRB register is updated. If the ABC bit is 0 (update every bit), the ABT bit becomes 1 (unmatching detected) at the same time unmatching is detected during check, and becomes 0 (undetected) when not detected. When setting the ABC bit to 1, if unmatching is ever detected, the ABT bit becomes 1 at the falling edge of the clock pulse of the ninth bit. If the ABT bit needs to be updated every byte, set the ABT bit to 0 after detecting an acknowledge for the first byte, and before transmitting/receiving the next byte.
Setting the ALS bit in the UiSMR2 register to 1 (SDA output stop enabled) causes an arbitration-lost to occur, in which case the SDAi pin becomes high-impedance at the same time the ABT bit becomes 1.

19.3.3.4 SCL Control and Clock Synchronization

Data transmission/reception in I^2C mode uses the transmit/receive clock as shown in Figure 19.20 “Transfer to UiRB Register and Interrupt Timing”. The clock speed increase makes it difficult to secure the required time for ACK generation and data transmit procedure. The I2C 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, 5 to 7) is used to insert a wait-state for ACK generation.
When the SWC bit is set to 1 (the SCLi pin is held low after the eighth bit of SCLi is received), the SCLi pin is held low on the falling edge of the eighth bit of SCLi. When the SWC bit is set to 0 (no waitstate/ wait-state cleared), the SCLi line is released.
When the SWC2 bit in the UiSMR2 register is set to 1 (the SCLi pin is held low), the SCLi pin is forced low even during transmission or reception. When the SWC2 bit is set to 0 (transmit/receive clock is output 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 1 (clock delayed), when the SWC9 bit is set to 1 (the SCLi pin is held low after the ninth bit of the SCLi is received), the SCLi pin is held low on the falling edge of the ninth bit of SCLi. When the SWC9 bit is set to 0 (no wait-state/wait-state cleared), the SCLi line is released.

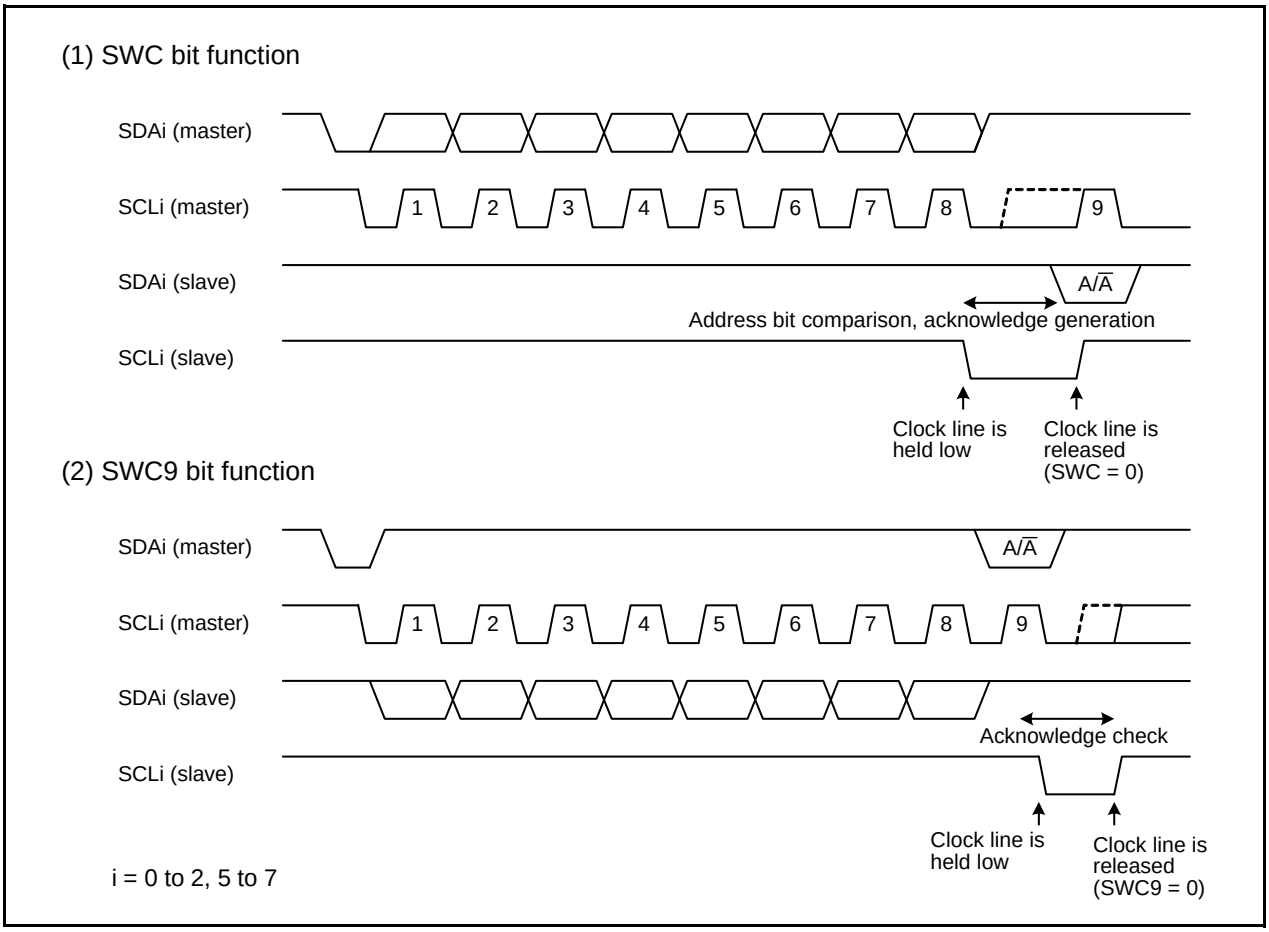


Figure 19.24 Inserting Wait-States Using Bits SWC and SWC9

The CSC bit in the UiSMR2 register synchronizes an internally generated clock with the clock applied to the SCLi pin. For example, if a wait-state is inserted from other devices, the two clocks are not synchronized. While the CSC bit is 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 UiBRG register value and 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 SCLi. The synchronized period starts from one clock prior to an internally generated 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 performing a transmit/receive operation. While 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.

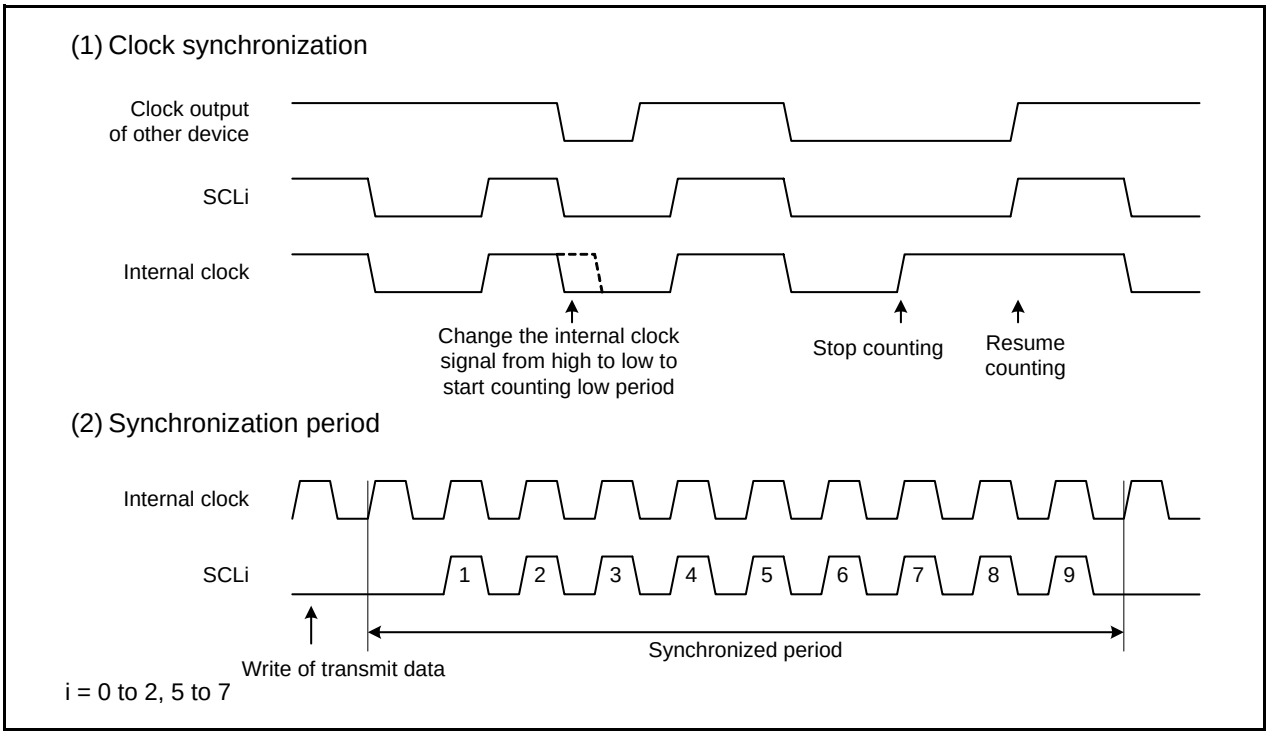


Figure 19.25 Clock Synchronization

19.3.3.5 SCL Clock Frequency

The SCL clock duty generated in I^2C mode is 50%. The low-level width of the SCL clock is 1.25 μs when the I^2C-bus setting is Fast-mode maximum SCL clock (400 kbps). This value does not satisfy the Fast-mode I^2C-bus specification (f_{LOW} = minimum 1.3 μs). Set the SCL clock to 384.6 kbps or less to satisfy the SCL clock low-level width of 1.3 μs or more.

When the clock synchronous function (Figure 19.25 “Clock Synchronization”) is enabled, there is a sampling delay of the noise filter plus 1 to 1.5 cycles of UiBRG count source.

There is also a delay of the SCL clock when high is determined and the SCL clock high width is extended. Therefore, the actual SCL clock becomes slower than SCL clock bit rate setting.

To calculate the effective value of SCL clock, take the SCL clock rise time (t_R) into consideration.

The following is an example of an SCL clock calculation.

Example of an effective value of SCL clock calculation at 384.6 kbps

- UiBRG count source: f1 = 20 MHz
- UiBRG register setting value: n = 26 - 1
- SCL clock rise time: t_R = 100 ns
- SCL clock fall time: t_F = 0 ns
- Noise filter width: t_{NF} = 100 ns (1)
- Sampling delay: t_{SD} = 1 cycle

f_{SCL} (theoretical value) = f1 / (2(n + 1)) = 20 MHz / (2(25 + 1)) = 384.6 kbps

t_{LOW} = 1 / (2fSCL (theoretical value)) = 1 / (2 × 384.6 kbps) = 1.3 μs

t_{HIGH} = 1 / (2fSCL (theoretical value)) + t_{NF} + (t_{SD} × 1 / f1)
= 1 / (2 × 384.6 kbps) + 100 ns + (1 × 1 / 20 MHz)
= 1.45 μs

f_{SCL} (actual value) = 1 / (t_F + t_{LOW} + t_R + t_{HIGH}) = 1 / (0 ns + 1.3 μs + 100 ns + 1.45 μs) 350.8 kbps

Note:
1. Maximum 200 ns.

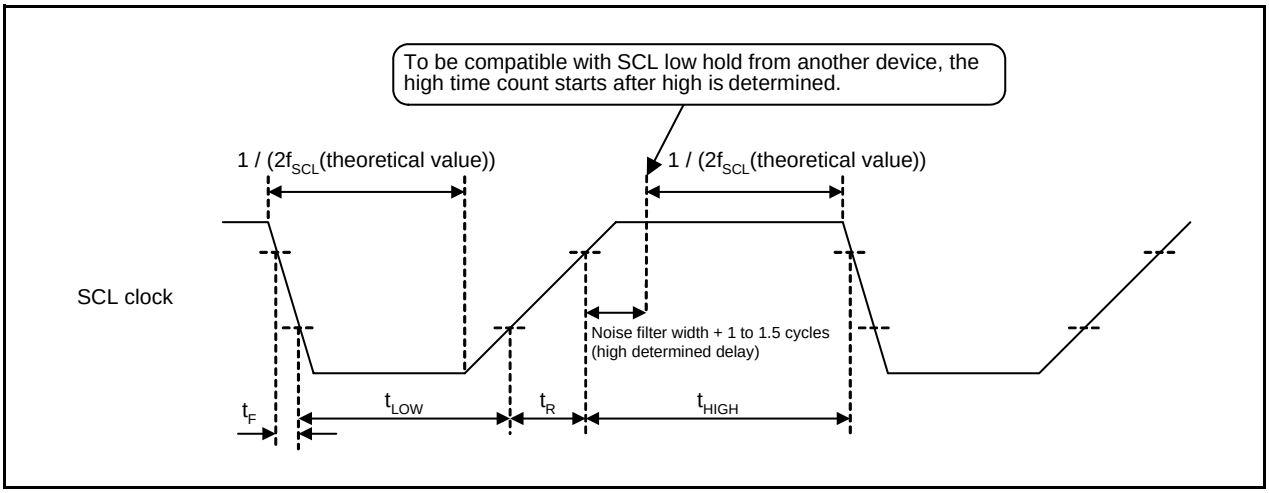


Figure 19.26 SCL Clock

19.3.3.6 SDA Output

When transmitting byte data, the SDAi pin outputs transmit data for the first to eighth bits, and it is released to receive an acknowledgment for the ninth bit.

In I^2C mode, set 9-bit data to the UiTB register. In 9-bit data, set the transmit data to bits b7 to b0 and set b8 to 1. By setting the UFORM bit in the UiC0 register to 1 (MSB first) and 9-bit data to the UiTB register, transmit data is output from the SDAi pin in the following order: b7, b6, b5, b4, b3, b2, b1, b0 and b8. As b8 is 1, the SDAi pin becomes high-impedance at the ninth bit and an acknowledgment can be received.

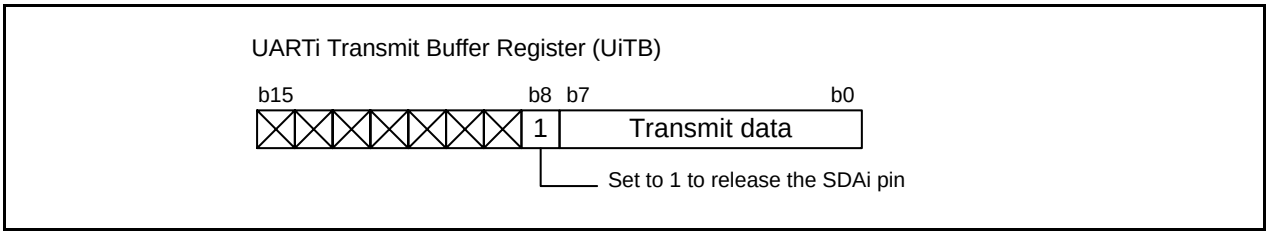


Figure 19.27 UiTB Register Setting (SDA Output)

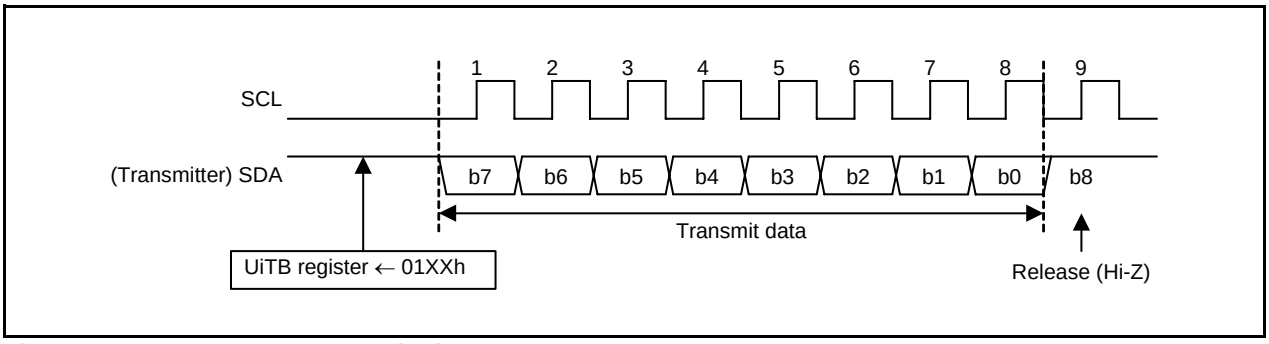


Figure 19.28 Byte Data Transmission

Set bits DL2 to DL0 in the UiSMR3 register to add no delays or a delay of one to eight UiBRG count source clock cycles to SDAi output.

Setting the SDHI bit in the UiSMR2 register to 1 (SDA output disabled) forcibly places the SDAi pin in a high-impedance state. Do not write to the SDHI bit at the rising edge of the UARTi transmit/receive clock as the ABT bit in the UiRB register may inadvertently become 1 (detected).

19.3.3.7 SDA Digital Delay

When transferring data with the I2C-bus, change the data while the SCL clock is low. When SDA is changed while the SCL clock is a high, the change is recognized as one of the corresponding conditions (see 19.5.3.4 “Setup and Hold Times When Generating a Start/Stop Condition”).
This function delays output from the SDAi pin. By delaying the change of the SDA, the data can be changed while the SCL clock is low. This function is enabled by setting bits DL2 to DL0 in the UiSMR3 register to 001b to 111b, and disabled by setting them to 000b.

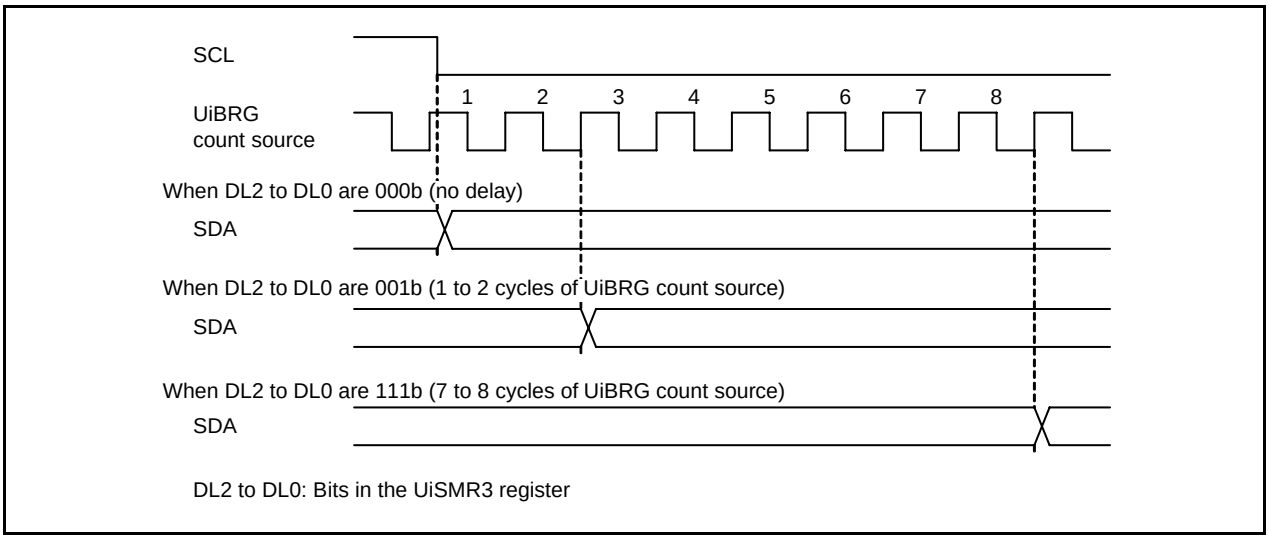


Figure 19.29 SDA Output Selection by Setting Bits DL2 to DL0

19.3.3.8 SDA Input

When the IICM2 bit in the UiSMR2 register (i = 0 to 2, 5 to 7) is set to 0, the first 8 bits of received data (D7 to D0) are stored in bits 7 to 0 in the UiRB register and the ninth bit (ACK/NACK) is stored in bit 8.
When the IICM2 bit in the UiSMR2 register is 0, the first to eighth bits (D7 to D0) of received data are stored in bits 7 to 0 in the UiRB register. The ninth bit (D8) is ACK or NACK.
When the IICM2 bit is 1, the first to seventh bits (D7 to D1) of received data are stored in bits 6 to 0 in the UiRB register and the eighth bit (D0) is stored in bit 8 in the UiRB register. Even when the IICM2 bit is 1, the same data as when the IICM2 bit is 0 can be read, provided the CKPH bit in the UiSMR3 register is 1. To read the data, read the UiRB register after the rising edge of ninth bit of the clock.

When receiving byte data, the SDAi pin is released for the first to eighth bits to receive data, and an acknowledgment is generated for the ninth bit. NACK is generated when the last byte data is received in master mode, or when the slave address does not match in slave mode. In all other cases, ACK is generated.
In I2C mode, set 9-bit data to the UiTB register. In 9-bit data, set FFh to b7 to b0 to release the SDAi pin and set b8 to 0 to generate ACK or 1 to generate NACK.
By setting 00FFh or 01FFh as 9-bit data to the UiTB register, the SDAi pin becomes high-impedance for the first to eighth bits, and data can be received. ACK or NACK is generated at the ninth bit.
Read the received data from the UiRB register. When the clock delay function is used, data transfer to the UiRB register occurs twice and each UiRB register value is different. Refer to Figure 19.20 “Transfer to UiRB Register and Interrupt Timing” for details.

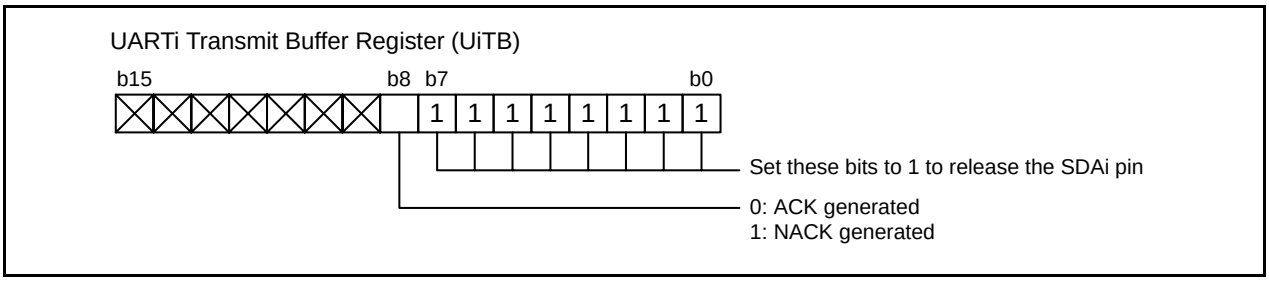


Figure 19.30 UiTB Register Setting (SDA Input)

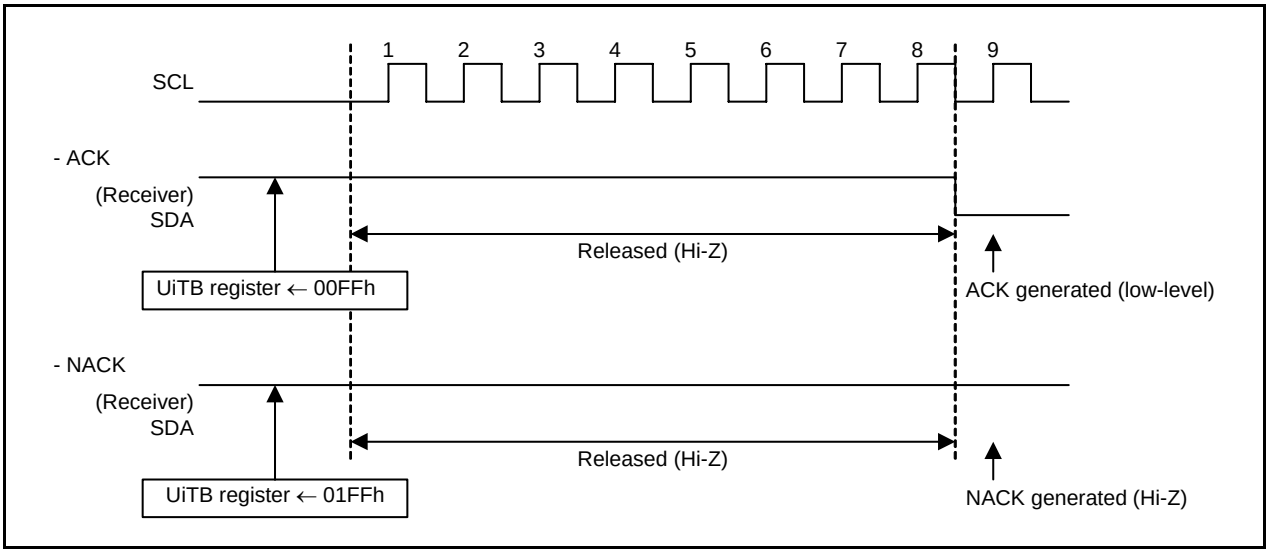


Figure 19.31 Byte Data Reception

19.3.3.9 ACK and NACK

When data is to be received, ACK is output after 8 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, 5 to 7) 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 0, a NACK interrupt request is generated when the SDAi pin is held high at the rising edge of the ninth bit of SCLi. An 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 ACK is detected.

19.3.3.10 Initialization of Transmission/Reception

If a start condition is detected while the STAC bit in the UiSMR2 register is 1 (UARTi initialization enabled), the serial interface operates as described below.

- The transmit shift register is initialized, and the contents of the UiTB register are transferred to the transmit shift register. In this way, the serial interface starts sending data when the next clock pulse is applied. However, the UARTi output value does not change state and remains the same as when a start condition was detected until the first bit of data is output in synchronization with the input clock.
- The receive shift register is initialized, and the serial interface starts receiving data when the next clock pulse is applied.
- The SWC bit in the UiSMR2 register becomes 1 (SCL wait output enabled). Consequently, the SCLi pin is pulled low at the falling edge of the ninth clock pulse.

When UARTi transmission/reception is started using this function, the TI bit does not change.

When the UARTi initializing function is used in slave mode, UARTi is initialized automatically when a start condition is detected. Therefore, an interrupt is unnecessary for detecting a start condition.

19.3.4 Special Mode 2

Special mode 2 supports serial communication between one or multiple master devices and multiple slaves devices. The transmit/receive clock polarity and phase are selectable. Table 19.20 lists the Special Mode 2 Specifications.

Table 19.20 Special Mode 2 Specifications

Item	Specification
Data format	Character data length: 8 bits
Transmit/receive clock	• Master mode The CKDIR bit in the UiMR register = 0 (internal clock): $\frac{fj}{2(n+1)}$ fj = f1SIO, f2SIO, f8SIO, f32SIO n: Setting value of UiBRG register 00h to FFh
Transmit/receive control	Controlled by I/O ports
Transmission start Conditions	To start transmission, satisfy the following requirements. • The TE bit in the UiC1 register = 1 (transmission enabled) • The TI bit in the UiC1 register = 0 (data present in UiTB register)
Reception start Conditions	To start reception, satisfy the following requirements. • The RE bit in the UiC1 register = 1 (reception enabled) • The TE bit = 1 (transmission enabled) • The TI bit = 0 (data present in the UiTB register)
Interrupt request generation timing	Transmit interrupt: One of the following can be selected. • The UiIRS bit in the UiC1 or UCON register = 0 (transmit buffer empty): When transferring data from the UiTB register to the UARTi transmit register (at start of transmission) • The UiIRS bit =1 (transmission completed): When the serial interface completed sending data from the UARTi transmit register Receive interrupt: • When transferring data from the UARTi receive register to the UiRB register (at completion of reception)
Error detection	Overrun error (1) This error occurs if the serial interface starts receiving the next unit of data before reading the UiRB register and receives the 7th bit of the next unit of data.
Selectable functions	• CLK polarity selection Data input/output can be chosen to occur synchronously with the rising or the falling edge of the transmit/receive clock. • LSB first, MSB first selection Whether to start sending/receiving data beginning with bit 0 or beginning with bit 7 can be selected. • Continuous receive mode selection Reception is enabled immediately by reading the UiRB register. • Switching serial data logic This function reverses the logic value of the transmit/receive data. • Clock phase setting Selectable from four combinations of transmit/receive clock polarities and phases

i = 0 to 2, 5 to 7

Note:

1. If an overrun error occurs, the received data of the UiRB register will be undefined. The IR bit in the SiRIC register remains unchanged.

Figure 19.32 shows Serial Bus Communication Control Example (UART2), and Table 19.21 lists I/O Pin Functions in Special Mode 2.

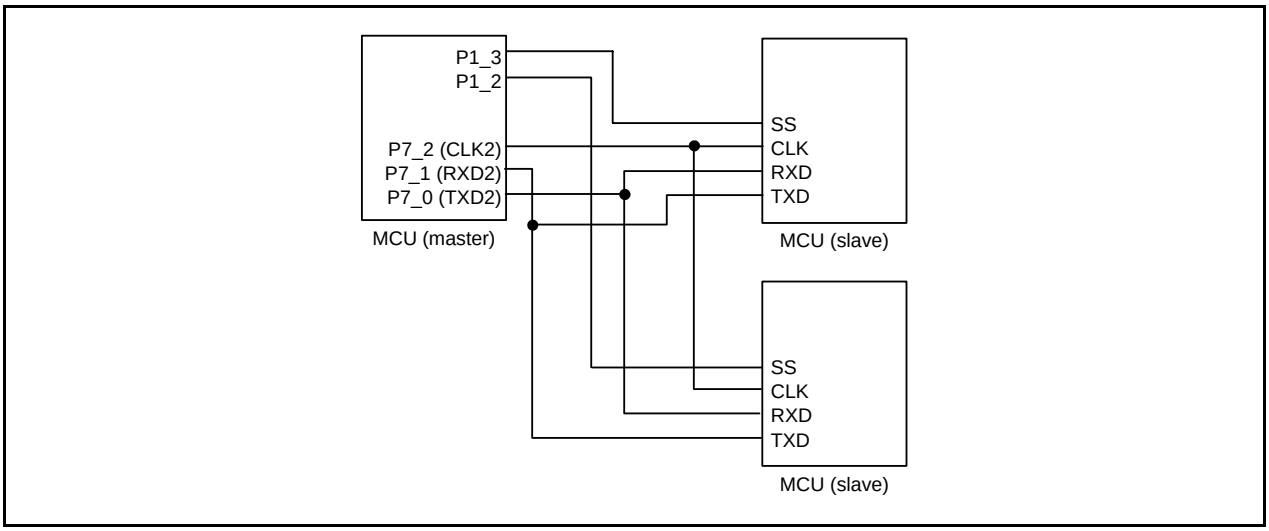


Figure 19.32 Serial Bus Communication Control Example (UART2)

Table 19.21 I/O Pin Functions in Special Mode 2

Pin Name	I/O	Function	Method of Selection
CLKi	Output	Clock output	The CKDIR bit in the UiMR register = 0
TXDi	Output	Serial data output	(Dummy data is output when performing reception only.)
RXDi	Input	Serial data input	Set the port direction bits sharing pins to 0.
	Input	Input port	Set the port direction bits sharing pins to 0. (can be used as an input port only when transmitting)

i = 0 to 2, 5 to 7

Pins CLKi and $\overline{\text{CTSi}}$/$\overline{\text{RTSi}}$ are not used. (They can be used as I/O ports.)

Table 19.22 Registers Used and Settings in Special Mode 2 (1)

Register	Bits	Function
PCLKR	PCLK1	Select the count source for the UiBRG register.
PCLKSTP1	PCKSTP12	Set to 0 when using f1.
UiTB	0 to 7	Set transmission data.
	8	— (does not need to be set) If necessary, set to 0.
UiRB	0 to 7	Reception data can be read.
	OER	Overrun error flag
	8, 11, 13 to 15	When read, the read value is undefined.
UiBRG	0 to 7	Set bit rate.
UiMR	SMD2 to SMD0	Set to 001b.
	CKDIR	Set to 0 in master mode.
	4 to 6	Set to 0.
	IOPOL	Set to 0.
UiC0	CLK0, CLK1	Select the count source for the UiBRG register.
	CRS	Disabled because CRD is 1
	TXEPT	Transmit register empty flag
	CRD	Set to 1.
	NCH	Select TXDi pin output format. (2)
	CKPOL	Clock phases can be set in combination with the CKPH bit in the UiSMR3 register.
	UFORM	Select the LSB first or MSB first.
UiC1	TE	Set to 1 to enable transmission/reception.
	TI	Transmit buffer empty flag
	RE	Set to 1 to enable reception.
	RI	Reception complete flag
	UjIRS	Select UARTj transmit interrupt source.
	UjRRM	Set to 1 to use continuous receive mode.
	UiLCH	Set to 1 to use inverted data logic.
	UiERE	Set to 0.
UiSMR	0 to 7	Set to 0.
UiSMR2	0 to 7	Set to 0.
UiSMR3	CKPH	Clock phases can be set in combination with the CKPOL bit in the UiC0 register.
	NODC	Set to 0.
	0, 2, 4 to 7	Set to 0.
UiSMR4	0 to 7	Set to 0.
UCON	U0IRS	Select UART0 transmit interrupt source.
	U1IRS	Select UART1 transmit interrupt source.
	U0RRM	Set to 1 to use continuous receive mode.
	U1RRM	Set to 1 to use continuous receive mode.
	CLKMD0	Disabled because CLKMD1 is 0
	CLKMD1, RCSP, 7	Set to 0.

i = 0 to 2, 5 to 7 j = 2, 5 to 7

Notes:
1. This table does not describe a procedure.
2. The TXD2 pin is N-channel open drain output. Nothing is assigned to the NCH bit in the U2C0 register. Only write 0 to this bit.

19.3.4.1 Clock Phase Setting Function

One of four combinations of transmit/receive clock phases and polarities can be selected using the CKPH bit in the UiSMR3 register and the CKPOL bit in the UiC0 register.

Make sure the transmit/receive clock polarity and phase are the same for the master and slave devices to be used for communication.

Figure 19.33 shows the Transmit/Receive Timing in Master Mode (Internal Clock).

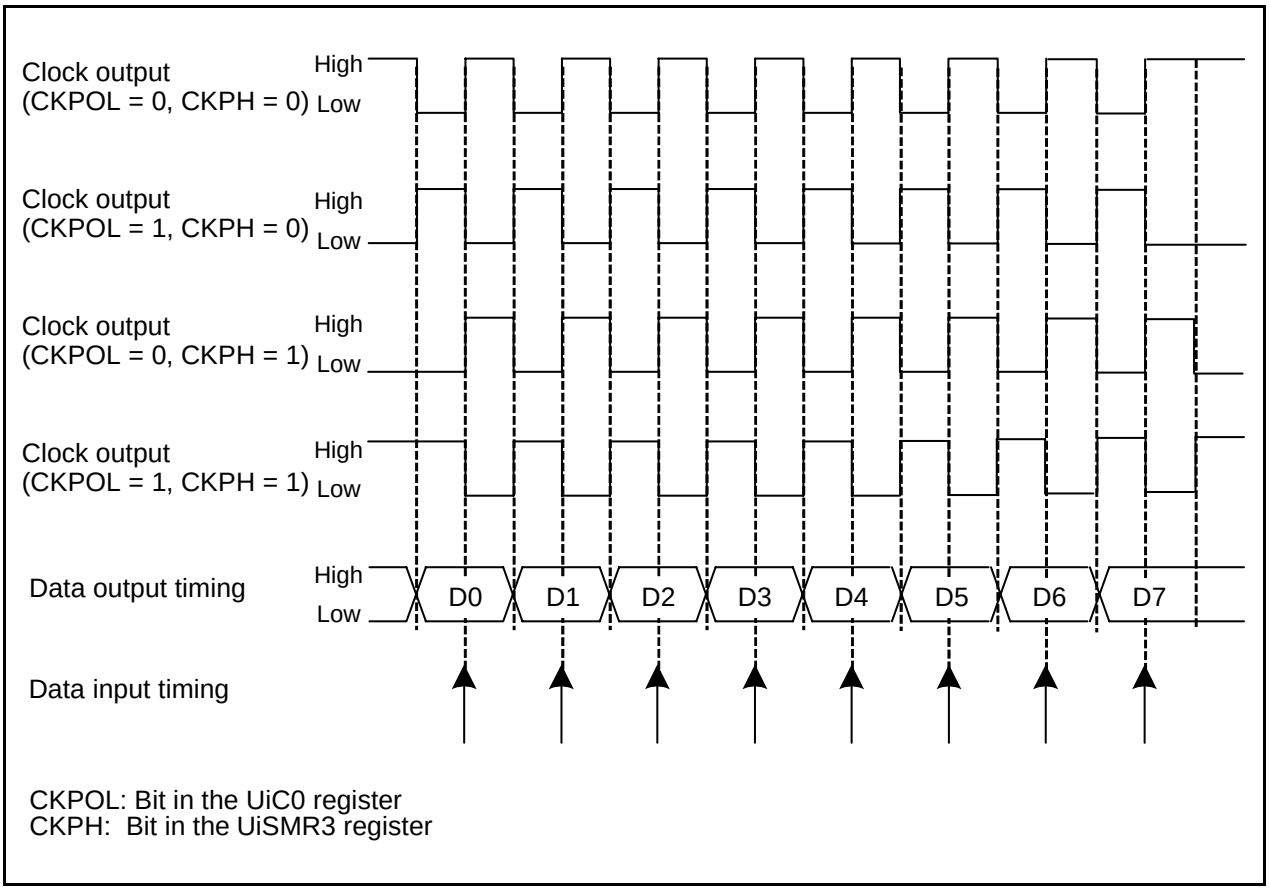


Figure 19.33 Transmit/Receive Timing in Master Mode (Internal Clock)

19.3.5 Special Mode 3 (IE Mode)

In this mode, 1 bit of IEBus is approximated by 1 byte of UART mode waveform.

Table 19.23 lists the Registers Used and Settings in IE Mode (1). Figure 19.34 shows the Bus Collision Detect Function-Related Bits.

If the TXDi pin (i = 0 to 2, 5 to 7) output level and RXDi pin input level do not match, a UARTi bus collision detect interrupt request is generated.

Use bits IFSR26 and IFSR27 in the IFSR2A register to enable the UART0/UART1 bus collision detect function.

Table 19.23 Registers Used and Settings in IE Mode (1)

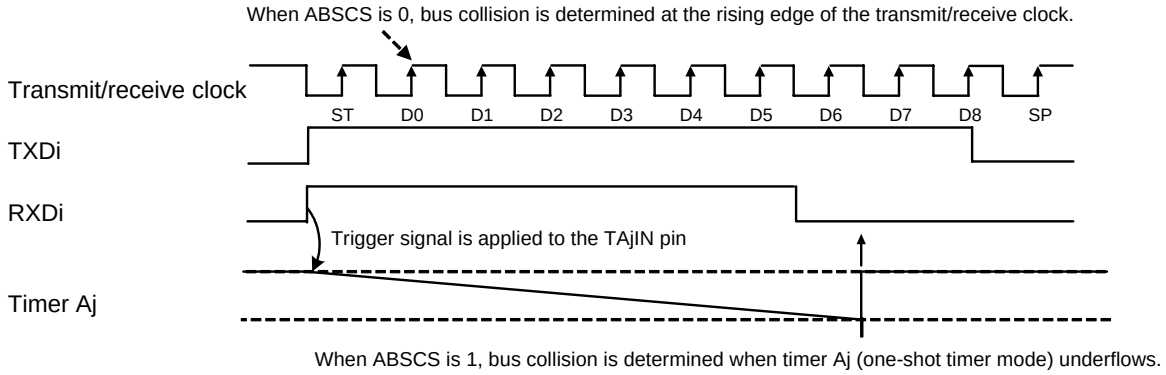
Register	Bits	Function
UiTB	0 to 8	Set transmission data.
UiRB (4)	0 to 8	Reception data can be read.
	OER, FER, PER, SUM	Error flag
UiBRG	0 to 7	Set bit rate.
UiMR	SMD2 to SMD0	Set to 110b.
	CKDIR	Select internal clock or external clock.
	STPS	Set to 0.
	PRY	Disabled because PRYE is 0
	PRYE	Set to 0.
	IOPOL	Select the TXD and RXD input/output polarity.
UiC0	CLK1, CLK0	Select the count source for the UiBRG register.
	CRS	Disabled because CRD is 1
	TXEPT	Transmit register empty flag
	CRD	Set to 1.
	NCH	Select TXDi pin output format. (3)
	CKPOL	Set to 0.
	UFORM	Set to 0.
UiC1	TE	Set to 1 to enable transmission.
	TI	Transmit buffer empty flag
	RE	Set to 1 to enable reception.
	RI	Reception complete flag
	UjIRS (2)	Select the source of UARTj transmit interrupt.
	UjRRM (2), UiLCH, UiERE	Set to 0.
UiSMR	0 to 3, 7	Set to 0.
	ABSCS	Select the sampling timing to detect a bus collision.
	ACSE	Set to 1 to use the auto clear function of transmit enable bit.
	SSS	Select the transmit start condition.
UiSMR2	0 to 7	Set to 0.
UiSMR3	0 to 7	Set to 0.
UiSMR4	0 to 7	Set to 0.
UCON	U0IRS, U1IRS	Select the source of UART0/UART1 transmit interrupt.
	U0RRM, U1RRM	Set to 0.
	CLKMD0	Disabled because CLKMD1 is 0
	CLKMD1, RCSP, 7	Set to 0.

i = 0 to 2, 5 to 7

Notes:

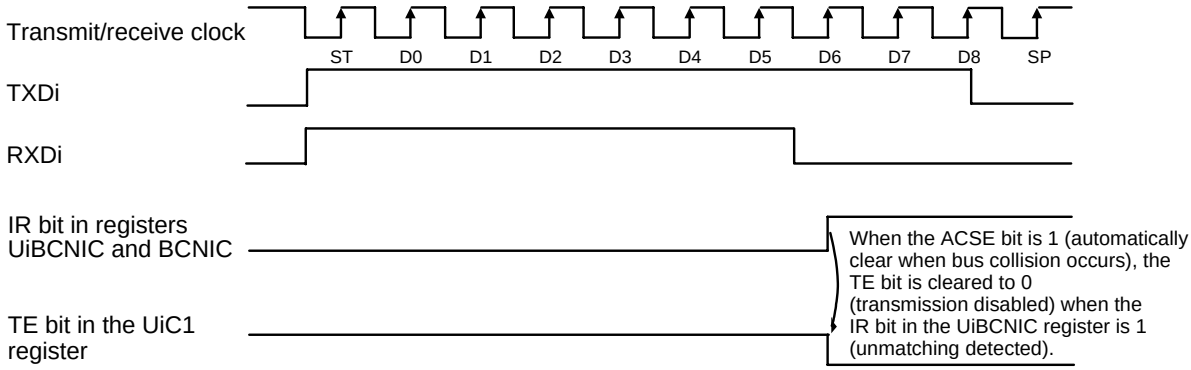
1. This table does not describe a procedure.
2. Set bits 4 and 5 in registers U0C1 and U1C1 to 0. Bits U0IRS, U1IRS, U0RRM, and U1RRM are in the UCON register.
3. The TXD2 pin is N-channel open drain output. Nothing is assigned to the NCH bit in the U2C0 register. If necessary, set it to 0.
4. Set the bits not listed above to 0 when writing to registers in IE mode.

(1) ABSCS bit in UiSMR register (bus collision detect sampling clock select) (i = 0 to 2, 5 to 7)



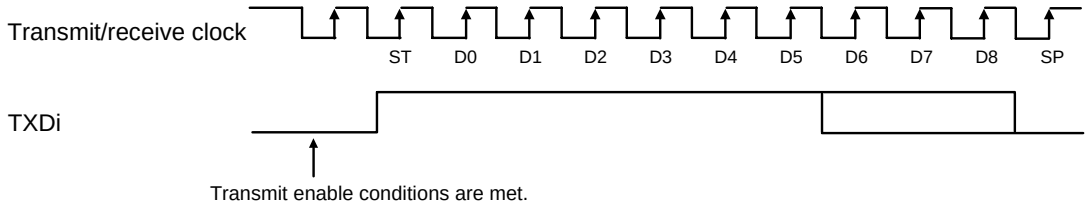
Timer Aj: Timer A3 in UART0; timer A4 in UART1; timer A0 in UART2
timer A0 in UART5; timer A3 in UART6; timer A4 in UART7

(2) ACSE bit in UiSMR register (auto clear of transmit enable bit)

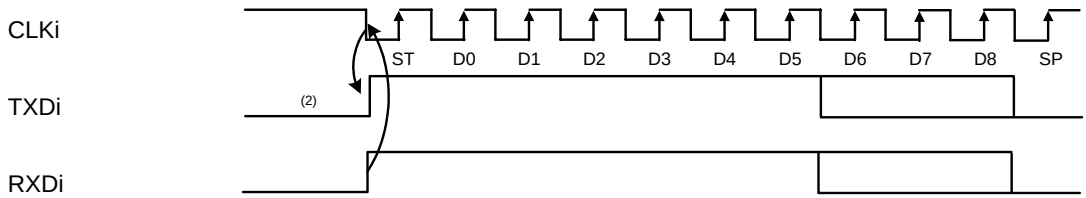


(3) SSS bit in the UiSMR register (transmit start condition select)

When the SSS bit is 0, the serial interface starts sending data one transmit/receive clock cycle after the transmission start condition is met.



When the SSS bit is 1, the serial interface starts sending data at the rising edge of RXDi. (1)



Notes:
1. The falling edge of RXDi when the IOPOL bit in the UiMR register is 0; the rising edge of RXDi when the IOPOL bit is 1.
2. The transmit conditions must be met before the falling edge of RXD.

The above diagram applies when IOPOL is 1 (reversed). i = 0 to 2, 5 to 7

Figure 19.34 Bus Collision Detect Function-Related Bits

19.3.6 Special Mode 4 (SIM Mode) (UART2)

SIM interface devices can communicate in UART mode. Both direct and inverse formats are available. The TXD2 pin outputs a low-level signal when a parity error is detected.

Table 19.24 lists the SIM Mode Specifications. Table 19.25 lists the Registers Used and Settings in SIM Mode (1).

Table 19.24 SIM Mode Specifications

Item	Specification
Data formats	• Direct format • Inverse format
Transmit/receive clock	• The CKDIR bit in the U2MR register = 0 (internal clock): fi/(16(n + 1)) fi = f1SIO, f2SIO, f8SIO, f32SIO n = Setting value of the U2BRG register 00h to FFh • The CKDIR bit = 1 (external clock): fEXT/(16(n + 1)) fEXT = input from the CLK2 pin n = Setting value of the U2BRG register 00h to FFh
Transmission start conditions	To start transmission, satisfy the following requirements. • The TE bit in the U2C1 register = 1 (transmission enabled) • The TI bit in the U2C1 register = 0 (data present in the U2TB register)
Reception start conditions	To start reception, satisfy the following requirements. • The RE bit in the U2C1 register = 1 (reception enabled) • Start bit detection
Interrupt request generation timing (2)	• Transmission When the serial interface completed sending data from the UART2 transmit register (the U2IRS bit =1) • Reception When transferring data from the UART2 receive register to the U2RB register (at completion of reception)
Error detection	• Overrun error (1) This error occurs when the serial interface starts receiving the next unit of data before reading the U2RB register and receives the bit before the last stop bit of the next unit of data. • Framing error (3) This error occurs when the number of stop bits set is not detected. • Parity error (3) During reception, when a parity error is detected, a parity error signal is output from the TXD2 pin. During transmission, a parity error is detected by the level of input to the RXD2 pin when a transmission interrupt occurs. • Error sum flag This flag becomes 1 when an overrun, framing, or parity error occurs.

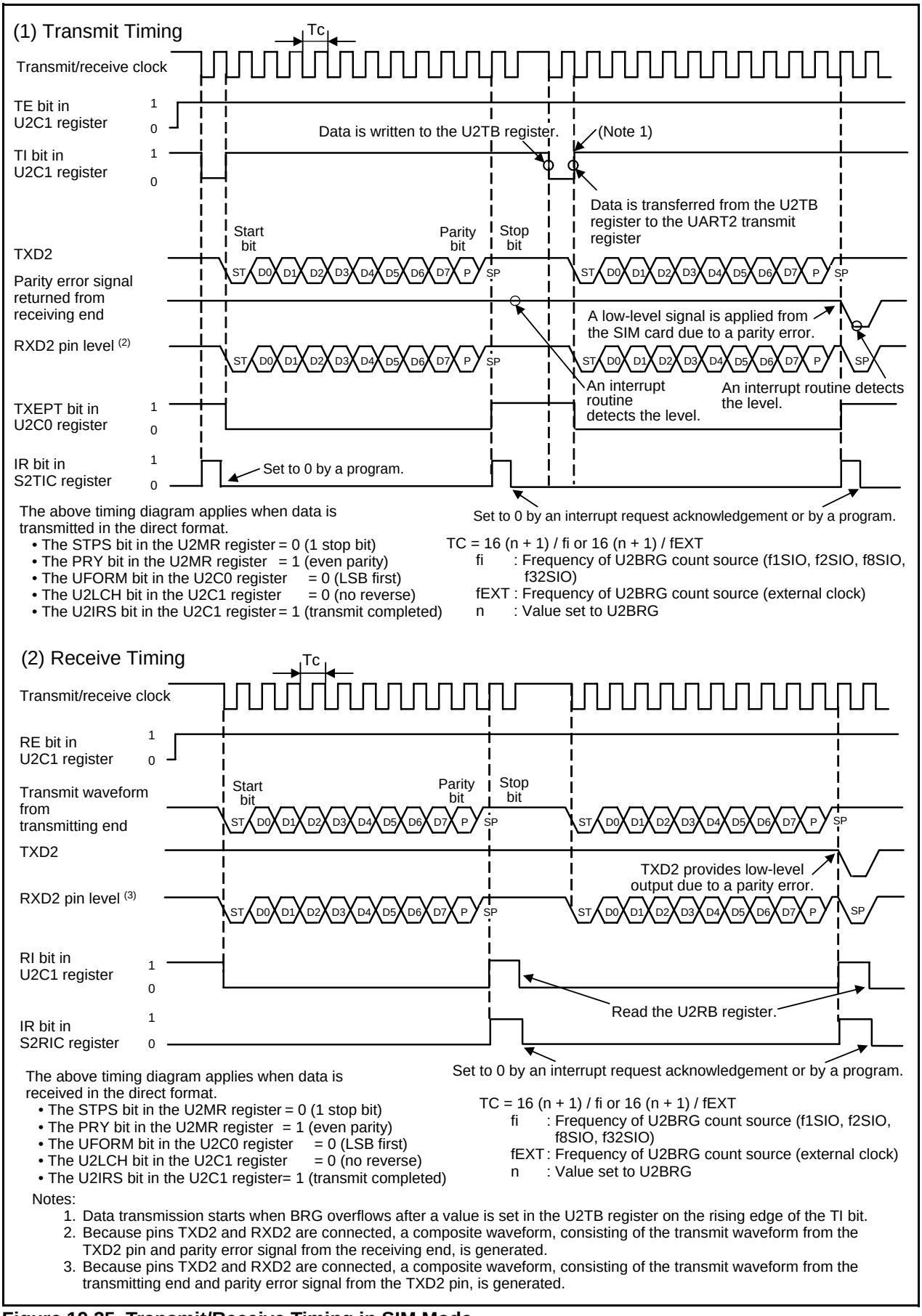
Notes:

1. When an overrun error occurs, the received data of the U2RB register will be undefined. The IR bit in the S2RIC register remains unchanged.
2. After reset is deasserted, a transmit interrupt request is generated by setting bits U2IRS and U2ERE in the U2C1 register to 1 (transmission completed, error signal output), then setting the TE bit to 1 (transmission enabled) and the transmission data to the U2TB register. Therefore, when using SIM mode, make sure to set the IR bit to 0 (interrupt not requested) after setting these bits.
3. The framing error flag and the parity error flag are detected when data is transferred from the UART2 receive register to the U2RB register.

Table 19.25 Registers Used and Settings in SIM Mode (1)

Register	Bits	Function
U2TB (2)	0 to 7	Set transmission data.
U2RB (2)	0 to 7	Reception data can be read.
	OER, FER, PER, SUM	Error flag
U2BRG	0 to 7	Set bit rate.
U2MR	SMD2 to SMD0	Set to 101b.
	CKDIR	Select the internal clock or external clock.
	STPS	Set to 0.
	PRY	Set to 1 in direct format or 0 in inverse format.
	PRYE	Set to 1.
	IOPOL	Set to 0.
U2C0	CLK0, CLK1	Select the count source for the U2BRG register.
	CRS	Disabled because CRD is 1
	TXEPT	Transmit register empty flag
	CRD	Set to 1.
	NCH	Set to 0.
	CKPOL	Set to 0.
	UFORM	Set to 0 in direct format or 1 in inverse format.
U2C1	TE	Set to 1 to enable transmission.
	TI	Transmit buffer empty flag
	RE	Set to 1 to enable reception.
	RI	Reception complete flag
	U2IRS	Set to 1.
	U2RRM	Set to 0.
	U2LCH	Set to 0 in direct format or 1 in inverse format.
	U2ERE	Set to 1.
U2SMR (2)	0 to 3	Set to 0.
U2SMR2	0 to 7	Set to 0.
U2SMR3	0 to 7	Set to 0.
U2SMR4	0 to 7	Set to 0.

Notes:
1. This table does not describe a procedure.
2. Set the bits not listed above to 0 when writing to registers in SIM mode.



Notes:
1. Data transmission starts when BRG overflows after a value is set in the U2TB register on the rising edge of the TI bit.
2. Because pins TXD2 and RXD2 are connected, a composite waveform, consisting of the transmit waveform from the TXD2 pin and parity error signal from the receiving end, is generated.
3. Because pins TXD2 and RXD2 are connected, a composite waveform, consisting of the transmit waveform from the transmitting end and parity error signal from the TXD2 pin, is generated.

Figure 19.35 Transmit/Receive Timing in SIM Mode

Figure 19.36 shows an Example of SIM Interface Connection. Connect TXD2 and RXD2, and then connect a pull-up resistor.

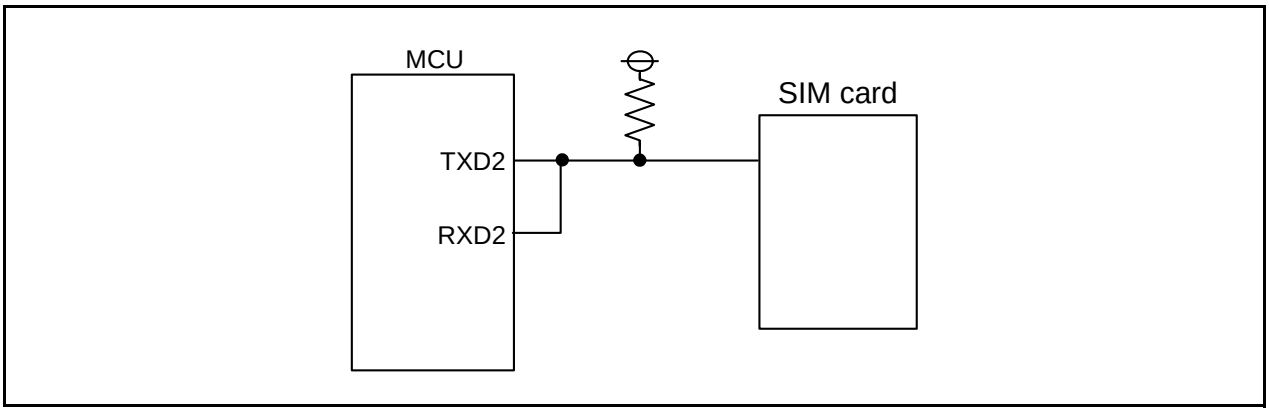


Figure 19.36 Example of SIM Interface Connection

19.3.6.1 Parity Error Signal Output

A parity error signal is enabled by setting the U2ERE bit in the U2C1 register to 1 (error signal output).

The parity error signal is output when a parity error is detected while receiving data. A low-level signal is output from the TXD2 pin in the timing shown in Figure 19.37. If the U2RB register is read while outputting a parity error signal, the PER bit is cleared to 0 (no parity error) and at the same time the TXD2 output again goes high.

When transmitting, a transmission complete interrupt request is generated at the falling edge of the transmit/receive pulse that immediately follows the stop bit. Therefore, whether or not a parity error signal has been returned can be determined by reading the port that shares the RXD2 pin in a transmission complete interrupt routine.

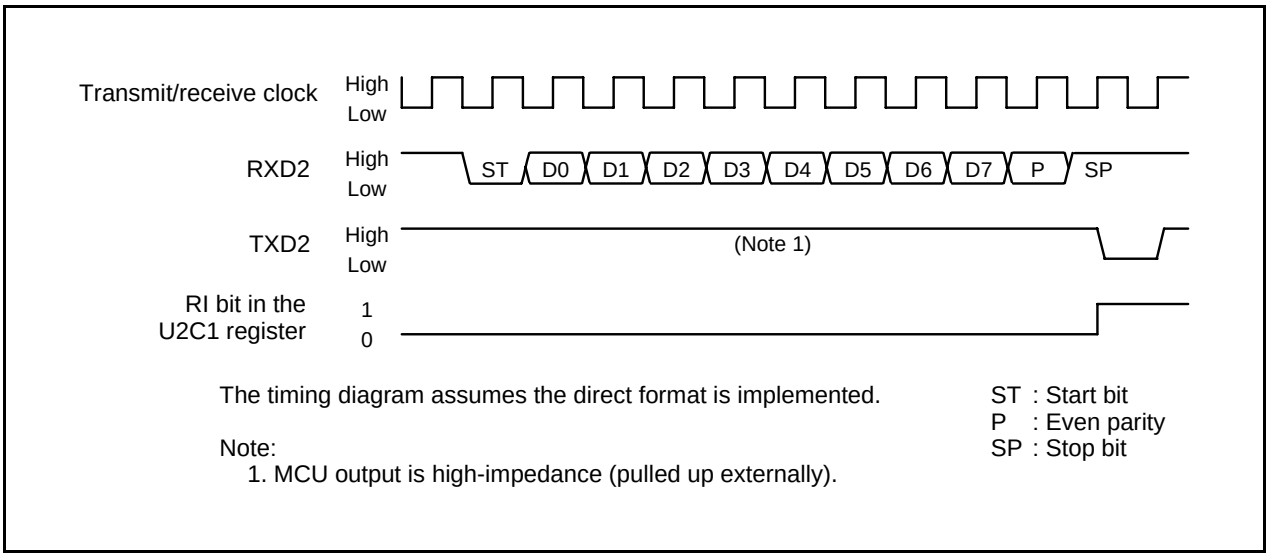


Figure 19.37 Parity Error Signal Output Timing

19.3.6.2 Format

Two formats are available: direct format and inverse format.

For direct format, set the PRYE bit in the U2MR register to 1 (parity enabled), the PRY bit to 1 (even parity), the UFORM bit in the U2C0 register to 0 (LSB first), and the U2LCH bit in the U2C1 register to 0 (not inverted). When data is transmitted, the contents of the U2TB register are transmitted with the even-numbered parity, starting from D0. When data is received, the received data are stored in the U2RB register, starting from D0. The even-numbered parity is used to determine when a parity error occurs.

For inverse format, set the PRYE bit to 1, the PRY bit to 0 (odd parity), the UFORM bit to 1 (MSB first), and the U2LCH bit to 1 (inverted). When data is transmitted, the contents of the U2TB register are logically inverted and are transmitted with odd-numbered parity, starting from D7. When data is received, the receive data is logically inverted and stored in the U2RB register, starting from D7. The odd-numbered parity is used to determine when a parity error occurs.

Figure 19.38 shows SIM Interface Format.

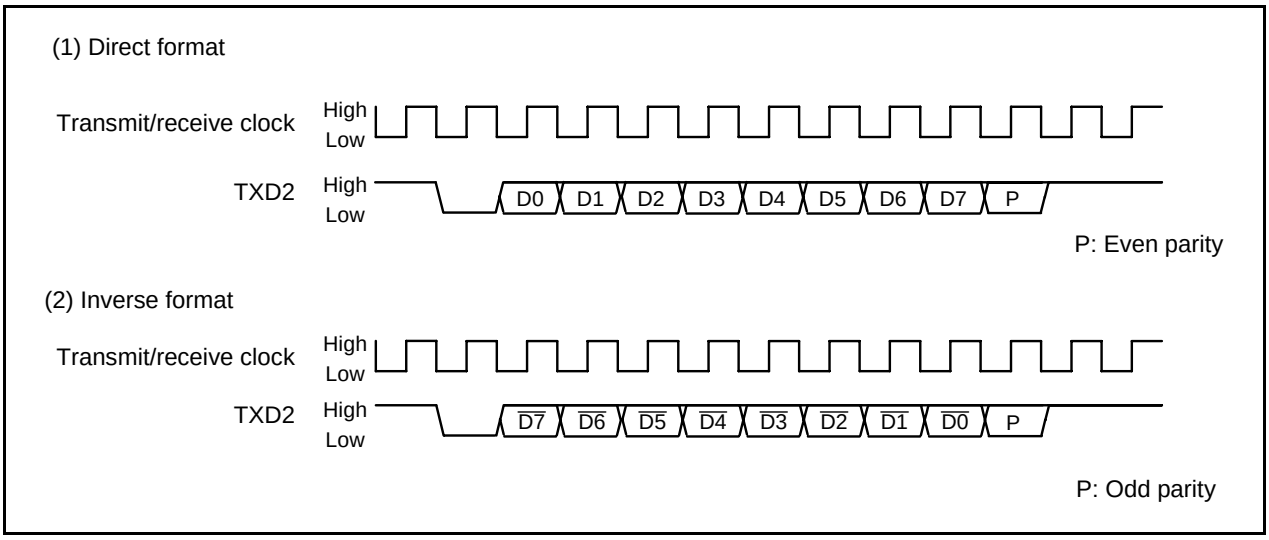


Figure 19.38 SIM Interface Format

19.4 Interrupts

UART0 to UART2 and UART5 to UART7 include interrupts by transmission, reception, ACK, NACK, start/stop condition detection, and bus collision detection.

19.4.1 Interrupt Related Registers

Refer to operation examples in each mode for interrupt sources and interrupt request generation timing. For details of interrupt control, refer to 13.7 “Interrupt Control”. Table 19.26 lists UART0 to UART2, UART5 to UART7 Interrupt Related Registers.

Table 19.26 UART0 to UART2, UART5 to UART7 Interrupt Related Registers

Address	Register	Symbol	Reset Value
0046h	UART1 Bus Collision Detection Interrupt Control Register	U1BCNIC	XXXX X000b
0047h	UART0 Bus Collision Detection Interrupt Control Register	U0BCNIC	XXXX X000b
004Ah	UART2 Bus Collision Detection Interrupt Control Register	BCNIC	XXXX X000b
004Fh	UART2 Transmit Interrupt Control Register	S2TIC	XXXX X000b
0050h	UART2 Receive Interrupt Control Register	S2RIC	XXXX X000b
0051h	UART0 Transmit Interrupt Control Register	S0TIC	XXXX X000b
0052h	UART0 Receive Interrupt Control Register	S0RIC	XXXX X000b
0053h	UART1 Transmit Interrupt Control Register	S1TIC	XXXX X000b
0054h	UART1 Receive Interrupt Control Register	S1RIC	XXXX X000b
006Bh	UART5 Bus Collision Detection Interrupt Control Register	U5BCNIC	XXXX X000b
006Ch	UART5 Transmit Interrupt Control Register	S5TIC	XXXX X000b
006Dh	UART5 Receive Interrupt Control Register	S5RIC	XXXX X000b
006Eh	UART6 Bus Collision Detection Interrupt Control Register	U6BCNIC	XXXX X000b
006Fh	UART6 Transmit Interrupt Control Register	S6TIC	XXXX X000b
0070h	UART6 Receive Interrupt Control Register	S6RIC	XXXX X000b
0071h	UART7 Bus Collision Detection Interrupt Control Register	U7BCNIC	XXXX X000b
0072h	UART7 Transmit Interrupt Control Register	S7TIC	XXXX X000b
0073h	UART7 Receive Interrupt Control Register	S7RIC	XXXX X000b
0205h	Interrupt Source Select Register 3	IFSR3A	00h
0206h	Interrupt Source Select Register 2	IFSR2A	00h

Some interrupts of UART0 to UART2 and UART5 to UART7 share interrupt vectors and interrupt control registers with other peripheral functions. When using these interrupts, select them by interrupt source select registers. Table 19.27 lists Interrupt Selection in UART0 to UART2 and UART6.

Table 19.27 Interrupt Selection in UART0 to UART2 and UART6

Interrupt Source	Interrupt Source Select Register Settings		
	Register	Bit	Setting Value
UART0 start/stop condition detection, bus collision detection	IFSR2A	IFSR26	1
UART1 start/stop condition detection, bus collision detection	IFSR2A	IFSR27	1
UART6 start/stop condition detection, bus collision detection	IFSR3A	IFSR35	0
UART6 transmission, NACK	IFSR3A	IFSR36	0

In the following mode, an interrupt request can be generated by rewriting bit values:

- Special mode 1 (I^2C mode)
  Set the IR bit in the interrupt control register of UARTi to 0 (interrupt not requested), when the following bits are changed:
  Bits SMD2 to SMD0 in the UiMR register, the IICM bit in the UiSMR register, the IICM2 bit in the UiSMR2 register, the CKPH bit in the UiSMR3 register
- Special mode 4 (SIM mode)
  After reset, a transmit interrupt request is generated by setting bits U2IRS and U2ERE in the U2C1 register to 1 (transmission completed, error signal output), then setting the TE bit to 1 (transmission enabled) and the transmission data to the U2TB register. Therefore, when using SIM mode, make sure to set the IR bit to 0 (interrupt not requested) after setting these bits.

19.4.2 Reception Interrupt

- The case that bits SMD2 to SMD0 in the UiMR register are not set to 010b (I^2C mode)
  When the RI bit in the UiC1 register is changed from 0 (no data in the UiRB register) to 1 (data present in the UiRB register), the IR bit in the SiRIC register is automatically set to 1 (interrupt requested).
  If an overrun error occurs (when the RI bit is 1, the next data is received), the RI bit remains 1, and therefore, the IR bit in the SiRIC register remains unchanged.
- The case that bits SMD2 to SMD0 in the UiMR register are set to 010b (I^2C mode)
  When the RI bit in the UiC1 register is changed from 0 (no data in the UiRB register) to 1 (data present in the UiRB register), the IR bit in the SiRIC register is automatically set to 1 (interrupt requested).
  When an overrun error occurs, the IR bit in the SiRIC register also becomes 1.

19.5 Notes on Serial Interface UARTi (i = 0 to 2, 5, 6)

19.5.1 Common Notes on Multiple Modes

19.5.1.1 CLKi Output

(Technical update number: TN-M16C-A178A/E)

When using the N-channel open drain output as an output mode of the CLKi pin, use following procedure to change the pin function:

When changing the pin function from the port to CLKi.

(1) Set bits SMD2 to SMD0 in the UiMR register to a value other than 000b to select serial interface mode.
(2) Set the NODC bit in the UiSMR3 register to 1.

When changing the pin function from CLKi to the port.

(1) Set the NODC bit to 0.
(2) Set bits SMD2 to SMD0 to 000b to disable the serial interface.

19.5.2 Clock Synchronous Serial I/O Mode

19.5.2.1 Transmission/Reception

When the $\overline{\text{RTS}}$ function is used with an external clock, the $\overline{\text{RTSi}}$ pin (i = 0 to 2, 5, 6) outputs a low-level signal, which informs the transmitting side that the MCU is ready for a receive operation. The $\overline{\text{RTSi}}$ pin outputs a high-level signal when a receive operation starts. Therefore, transmit timing and receive timing can be synchronized by connecting the $\overline{\text{RTSi}}$ pin to the $\overline{\text{CTSi}}$ pin on the transmitting side. The $\overline{\text{RTS}}$ function is disabled when an internal clock is selected.

19.5.2.2 Transmission

If the transmission is started while an external clock is selected and the TXEPT bit in the UiC0 register (i = 0 to 4) is 1 (no data present in transmit register), meet the last requirement at either of the following timings:

External clock level:

- The CKPOL bit in the UiC0 register is 0 (transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge) and the external clock is high.
- The CKPOL bit is 1 (transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge) and the external clock is low.

Requirements to start transmission (in no particular order):

- The TE bit in the UiC1 register is 1 (transmission enabled).
- The TI bit in the UiC1 register is 0 (data present in the UiTB register).
- When the CTS function is selected, input on the CTSi pin is low.

19.5.2.3 Reception

In clock synchronous serial I/O mode, a shift clock is generated by activating a transmitter. Set the UARTi-associated registers for a transmit operation even if the MCU is used for a receive operations only. Dummy data is output from the TXDi pin (i = 0 to 2, 5, 6) while receiving.

When an internal clock is selected, a shift clock is generated by setting the TE bit in the UiC1 register to 1 (transmission enabled) and placing dummy data in the UiTB register. When an external clock is selected, set the TE bit to 1 (transmission enabled), set dummy data in the UiTB register, and input an external clock to the CLKi pin to generate a shift clock.

If data is received consecutively, an overrun error occurs when the RI bit in the UiC1 register is 1 (data present in the UiRB register) and the next receive data is received in the UARTi receive register. Then, the OER bit in the UiRB register becomes 1 (overrun error occurred). At this time, the UiRB register is undefined. When an overrun error occurs, program the transmitting and receiving sides to retransmit the previous data. If an overrun error occurs again, the IR bit in the SiRIC register remains unchanged.

To receive data consecutively, set dummy data in the low-order byte in the UiTB register for each receive operation.

External clock level:
- The CKPOL bit in the UiC0 register is 0 (transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge) and the external clock is high.
- The CKPOL bit is 1 (transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge) and the external clock is low.

Requirements to start reception (in no particular order):
- The RE bit in the UiC1 register is 1 (reception enabled).
- The TE bit in the UiC1 register is 1 (transmission enabled).
- The TI bit in the UiC1 register is 0 (data present in the UiTB register).

19.5.3 Special Mode (I^2C Mode)

19.5.3.1 Generating Start and Stop Conditions

When generating start, stop, and restart conditions, set the STSPSEL bit in the UiSMR4 register (i = 0 to 2, 5, 6) to 0 and wait for more than a half cycle of the transmit/receive clock. Then set each condition generation bit (STAREQ, RSTAREQ, and STPREQ) from 0 to 1.

19.5.3.2 IR Bit

Set the following bits first, and then set the IR bit in the UARTi interrupt control registers to 0 (interrupt not requested).

Bits SMD2 to SMD0 in the UiMR register, the IICM bit in the UiSMR register,
the IICM2 bit in the UiSMR2 register, the CKPH bit in the UiSMR3 register

19.5.3.3 Low/High-level Input Voltage and Low-level Output Voltage

The low-level input voltage, high-level input voltage, and low-level output voltage differ from the I^2C-bus specification.
Refer to the recommended operating conditions for I/O ports which share the pins with SCL and SDA.

I^2C-bus specification
High level input voltage (V_{IH}) = min. 0.7 V_{CC}
Low level input voltage (V_{IL}) = max. 0.3 V_{CC}

19.5.3.4 Setup and Hold Times When Generating a Start/Stop Condition

When generating a start condition, the hold time (t_{HD}:STA) is a half cycle of the SCL clock.
When generating a stop condition, the setup time (t_{SU}:STO) is a half cycle of the SCL clock.
When the SDA digital delay function is enabled, take delay time into consideration (see 19.3.3.7 “SDA Digital Delay”).
The following shows a calculation example of hold and setup times when generating a start/stop condition.

Calculation example when setting 100 kbps
- U2iBRG count source: f1 = 20 MHz
- U2iBRG register setting value: n = 100 - 1
- SDA digital delay setting value: DL2 to DL0 are 101b (5 or 6 cycles of U2iBRG count source)

f_{SCL} (theoretical value) = f1 / (2(n+1)) = 20 MHz / (2 × (99 + 1)) = 100 kbps
t_{DL} = delay cycle count / f1 = 6 / 20 MHz = 0.3 μs
$t_{HD:STA}$ (theoretical value) = 1 / (2f_{SCL} (theoretical value)) = 1 / (2 × 100 kbps) = 5 μs
$t_{SU:STO}$ (theoretical value) = 1 / (2f_{SCL} (theoretical value)) = 1 / (2 × 100 kbps) = 5 μs
$f_{HD:STA}$ (actual value) = $t_{HD:STA}$ (theoretical value) - t_{DL} = 5 μs - 0.3 μs = 4.7 μs
$f_{SU:STO}$ (actual value) = $t_{SU:STO}$ (theoretical value) + t_{DL} = 5 μs + 0.3 μs = 5.3 μs

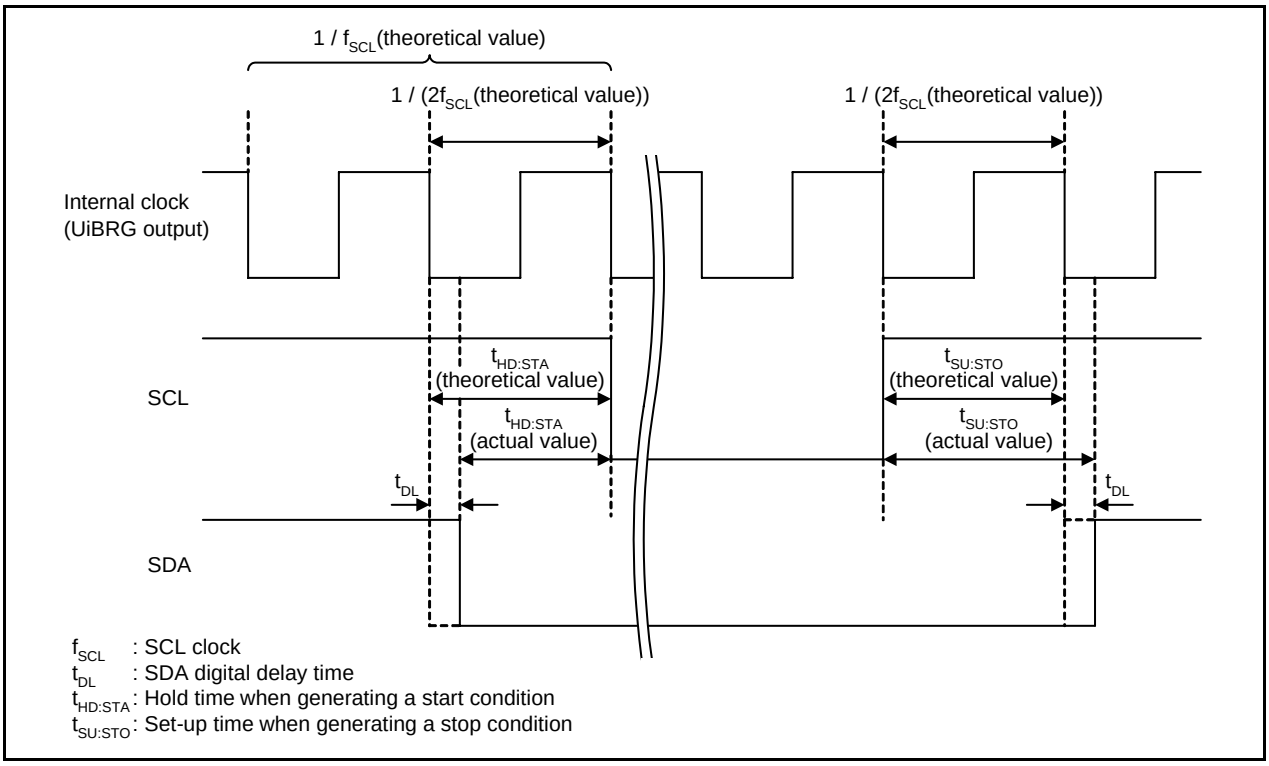


Figure 19.39 Setup and Hold Times When Generating Start and Stop Conditions

19.5.3.5 Restrictions on the Bit Rate When Using the Ui2BRG Count Source

In I^2C mode, set the Ui2BRG register to a value of 03h or greater.

A maximum of three Ui2BRG count source cycles are necessary until the internal circuit acknowledges the SCL clock level. The connectable I^2C-bus bit rate is one-third or less than the Ui2BRG count source speed. If a value between 00h to 02h is set to the Ui2BRG register, bit slippage may occur.

19.5.3.6 Restart Condition in Slave Mode

When a restart condition is detected in slave mode, the successive processes may not be executed correctly. In slave mode, do not use a restart condition.

19.5.3.7 Requirements to Start Transmission/Reception in Slave Mode

When transmission/reception is started in slave mode and the TXEPT bit in the UiC0 register is 1 (no data present in transmit register), meet the last requirement when the external clock is high.

Requirements to start transmission (in no particular order):

- The TE bit in the Ui2C1 register is 1 (transmission enabled).
- The TI bit in the Ui2C1 register is 0 (data present in the UiTB register).

Requirements to start reception (in no particular order):

- The RE bit in the Ui2C1 register is 1 (reception enabled).
- The TE bit in the Ui2C1 register is 1 (transmission enabled).
- The TI bit in the Ui2C1 register is 0 (data present in the UiTB register).

19.5.4 Special Mode 4 (SIM Mode)

After reset, a transmit interrupt request is generated by setting bits U2IRS and U2ERE in the U2C1 register to 1 (transmission completed) and 1 (error signal output), respectively. Therefore, when using SIM mode, make sure to set the IR bit to 0 (interrupt not requested) after setting these bits.

20. Serial Interface SI/O3 and SI/O4

20.1 Introduction

SI/O3 and SI/O4 are dedicated clock-synchronous serial I/O ports.
Table 20.1 lists SI/O3 and SI/O4 Specifications.
Figure 20.1 shows SI/O3 and SI/O4 Block Diagram, and Table 20.2 lists the I/O Ports.

Table 20.1 SI/O3 and SI/O4 Specifications

Item	Specification
Data format	Character length: 8 bits
Transmit/receive clocks	• The SMi6 bit in the SiC register = 1 (internal clock): $\frac{fj}{2(n+1)}$ fj = f1SIO, f2SIO, f8SIO, f32SIO n = setting value of the SiBRG register 00h to FFh The SMi6 bit = 0 (external clock): Input from the CLKi pin (1)
Transmission/reception start condition	Before transmission/reception starts, write transmit data to the SiTRR register. (2)
Interrupt request generation timing	• SMi4 bit in the SiC register = 0 The rising edge of the last transmit/receive clock • the SMi4 bit = 1 The falling edge of the last transmit/receive clock
Selectable functions	• CLK polarity selection Whether data is input/output at the rising or falling edge of the transmit/receive clock can be selected. • LSB first or MSB first selection Whether to start transmitting/receiving data from bit 0 or from bit 7 can be selected. • SOUTi initial value setting function When the SMi6 bit in the SiC register = 0 (external clock), the SOUTi pin output level while not transmitting can be selected. • SOUTi state selection after transmission Whether to set to high-impedance or retain the last bit level can be selected when the SMi6 bit in the SiC register is 1 (internal clock).

i = 3, 4
Notes:
1. The data is shifted every time the external clock is input. When completing data transmission/reception of the eighth bit, read or write to the SiTRR register before inputting the clock for the next data transmission/reception.
2. When the SMi6 bit in the SiC register is 0 (external clock), follow the steps below.
   • When the SMi4 bit in the SiC register is 0, write transmit data to the SiTRR register while input to the CLKi pin is high.
   • When the SMi4 bit is 1, write transmit data to the SiTRR register while input to the CLKi pin is low.

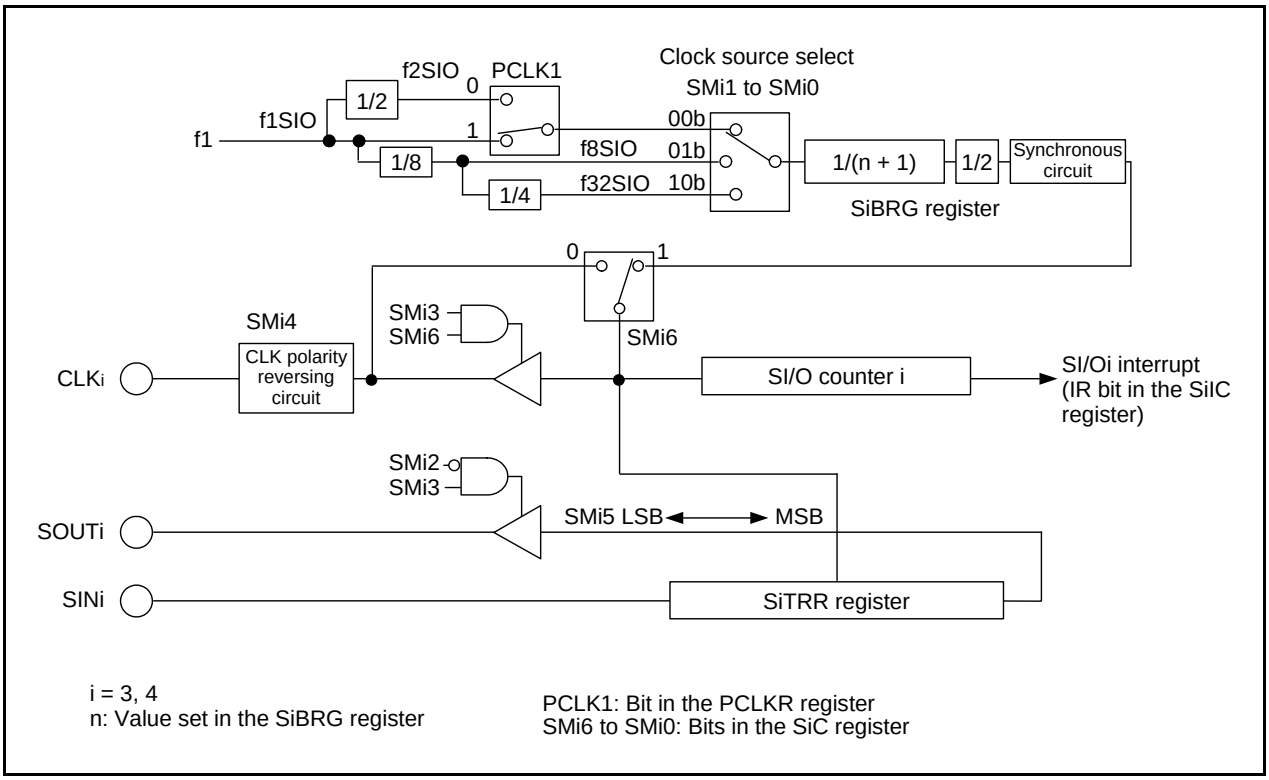


Figure 20.1 SI/O3 and SI/O4 Block Diagram

Table 20.2 I/O Ports

Pin Name	I/O	Function	Selecting Method
CLKi	Output	Transmit/receive clock output	SMi3 bit in the SiC register = 1 SMi6 bit in the SiC register = 1
	Input	Transmit/receive clock input	SMi3 bit in the SiC register = 1 SMi6 bit in the SiC register = 0 Port direction bits sharing pins = 0
SOUTi	Output	Serial data output	SMi3 bit in the SiC register = 1 SMi2 bit in the SiC register = 0
SINi	Input	Serial data input	SMi3 bit in the SiC register = 1 Port direction bits sharing pins = 0 (Dummy data is input only when transmitting.)

i = 3, 4

20.2 Registers

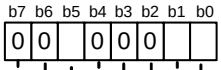
Table 20.3 lists registers associated with SI/O3 and SI/O4.

Table 20.3 Registers

Address	Register	Symbol	Reset Value
0012h	Peripheral Clock Select Register	PCLKR	0000 0011b
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
0270h	SI/O3 Transmit/Receive Register	S3TRR	XXh
0272h	SI/O3 Control Register	S3C	0100 0000b
0273h	SI/O3 Bit Rate Register	S3BRG	XXh
0274h	SI/O4 Transmit/Receive Register	S4TRR	XXh
0276h	SI/O4 Control Register	S4C	0100 0000b
0277h	SI/O4 Bit Rate Register	S4BRG	XXh
0278h	SI/O3, 4 Control Register 2	S34C2	00XX X0X0b

20.2.1 Peripheral Clock Select Register (PCLKR)

Peripheral Clock Select Register



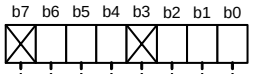
Symbol	Address	After Reset
PCLKR	0012h	0000 0011b

Bit Symbol	Bit Name	Function	RW
PCLK0	Timers A and B clock select bit (clock source for timers A and B, and multi-master I^2C-bus interface)	0: f2TIMAB/f2IIC 1: f1TIMAB/f1IIC	RW
PCLK1	SI/O clock select bit (clock source for UART0 to UART2, UART5 to UART7, SI/O3, and SI/O4)	0: f2SIO 1: f1SIO	RW
— (b4-b2)	Reserved bits	Set to 0.	RW
PCLK5	Clock output function expansion bit	0: Selected by bits CM01 to CM00 in the CM0 register 1: Output f1	RW
— (b7-b6)	Reserved bits	Set to 0.	RW

Rewrite the PCLKR register after setting the PRC0 bit in the PRCR register to 1 (write enabled).

20.2.2 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1



Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I^2C peripheral clock stop bit (multi-master I^2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

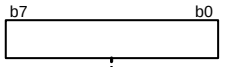
Set the PRC0 bit in the PRCR register to 1 (write enabled) before the PCKSTP1 register is rewritten.

PCKSTP15 (SIO peripheral clock stop bit) (b5)

Set the PCKSTP15 bit to 0 (f1 provide enabled) when using the f1 as the clock source of the transmit/receive clock.

20.2.3 SI/Oi Transmit/Receive Register (SiTRR) (i = 3, 4)

SI/Oi Transmit/Receive Register (i = 3, 4)



Symbol	Address	After Reset
S3TRR	0270h	Undefined
S4TRR	0274h	Undefined

Function	RW
Transmission/reception starts by writing transmit data to this register. After transmission/reception completes, receive data can be read.	RW

Write to the SiTRR register while the serial interface is neither transmitting nor receiving.

Write the value into the SiTRR register each time 1-byte data is received, even when data is only received.

20.2.4 SI/Oi Control Register (SiC) (i = 3, 4)

SI/Oi Control Register (i = 3, 4)

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
S3C	0272h	0100 0000b
S4C	0276h	0100 0000b

Bit symbol	Bit Name	Function	RW
SMi0	Internal synchronous clock select bit	b1 b0 0 0: f1SIO or f2SIO selected 0 1: f8SIO selected 1 0: f32SIO selected 1 1: Do not set this value	RW
SMi1			
SMi2	SOUTi output disable bit	0: SOUTi output enabled 1: SOUTi output disabled (high-impedance)	RW
SMi3	SI/Oi port select bit	0: I/O port serial interface disabled 1: SOUTi output, CLKi function serial interface enabled	RW
SMi4	CLK polarity select bit	0: Transmit data is output at falling edge of transmit/receive clock and receive data is input at rising edge 1: Transmit data is output at rising edge of transmit/receive clock and receive data is input at falling edge	RW
SMi5	Bit order select bit	0: LSB first 1: MSB first	RW
SMi6	Synchronous clock select bit	0: External clock 1: Internal clock	RW
SMi7	SOUTi initial output set bit	Enabled when SMi6 is 0 0: Low output 1: High output	RW

After setting the PRC2 bit in the PRCR register to 1 (write enabled), use the next instruction to write to this register.

SMi1 to SMi0 (Internal synchronous clock select bit) (b1 to b0)

Select f1SIO or f2SIO by the PCLK1 bit in the PCLKR register.
Set the SiBRG register when changing bits SMi1 to SMi0.

SMi2 (SOUTi output disable bit) (b2)

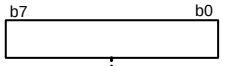
When the SMi2 bit is set to 1 (SOUTi output disabled), the target pin becomes high-impedance regardless of which function of the pin is being used.

SMi7 (SOUTi initial value set bit) (b7)

Set the SMi7 bit when the SMi3 bit is 0 (I/O port, serial interface disabled). The level selected by the SMi7 bit is output from the SOUTi pin by setting the SMi3 bit to 1 and SMi2 bit to 0 (SOUTi output).

20.2.5 SI/Oi Bit Rate Register (SiBRG) (i = 3, 4)

SI/Oi Bit Rate Register (i = 3, 4)



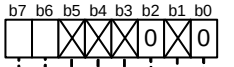
Symbol	Address	After Reset
S3BRG	0273h	Undefined
S4BRG	0277h	Undefined

Function	Setting Range	RW
SiBRG divides the count source by n + 1 where n = set value	00h to FFh	WO

Use the MOV instruction to write into the SiBRG register.
Write into the SiBRG register after setting bits SMi1 to SMi0 in the SiC register, and while the serial interface is neither transmitting nor receiving.

20.2.6 SI/O3, 4 Control Register 2 (S34C2)

SI/O3, 4 Control Register 2



Symbol	Address	After Reset
S34C2	0278h	00XX X0X0b

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved bit	Set to 0.	RW
— (b1)	No register bit. If necessary, set to 0. Read as undefined value		—
— (b2)	Reserved bit	Set to 0.	RW
— (b5-b3)	No register bits. If necessary, set to 0. Read as undefined value.		—
SM26	SOUT3 output control bit	SOUT3 state after transmission 0: High-impedance 1: Last bit level retained	RW
SM27	SOUT4 output control bit	SOUT4 state after transmission 0: High-impedance 1: Last bit level retained	RW

SM26 (SOUT3 output control bit) (b6)
SM27 (SOUT4 output control bit) (b7)

Bits SM26 and SM27 are enabled when the SMi6 bit in the SiC register is 1 (internal clock). Set the SMi3 bit in the SiC register to 1 (serial interface enabled) after setting bits SM26 and SM27.

20.3 Operations

20.3.1 Basic Operations

SI/Oi transmits and receives the data simultaneously. The SiTRR register is not divided into a register for transmission/reception and buffer. Therefore, write transmit data to the SiTRR register while transmission/reception stops. Read receive data from the SiTRR register while transmission/reception stops.

20.3.2 CLK Polarity Selection

Use the SMi4 bit in the SiC register to select the transmit/receive clock polarity. Figure 20.2 shows Polarity of Transmit/Receive Clock.

.

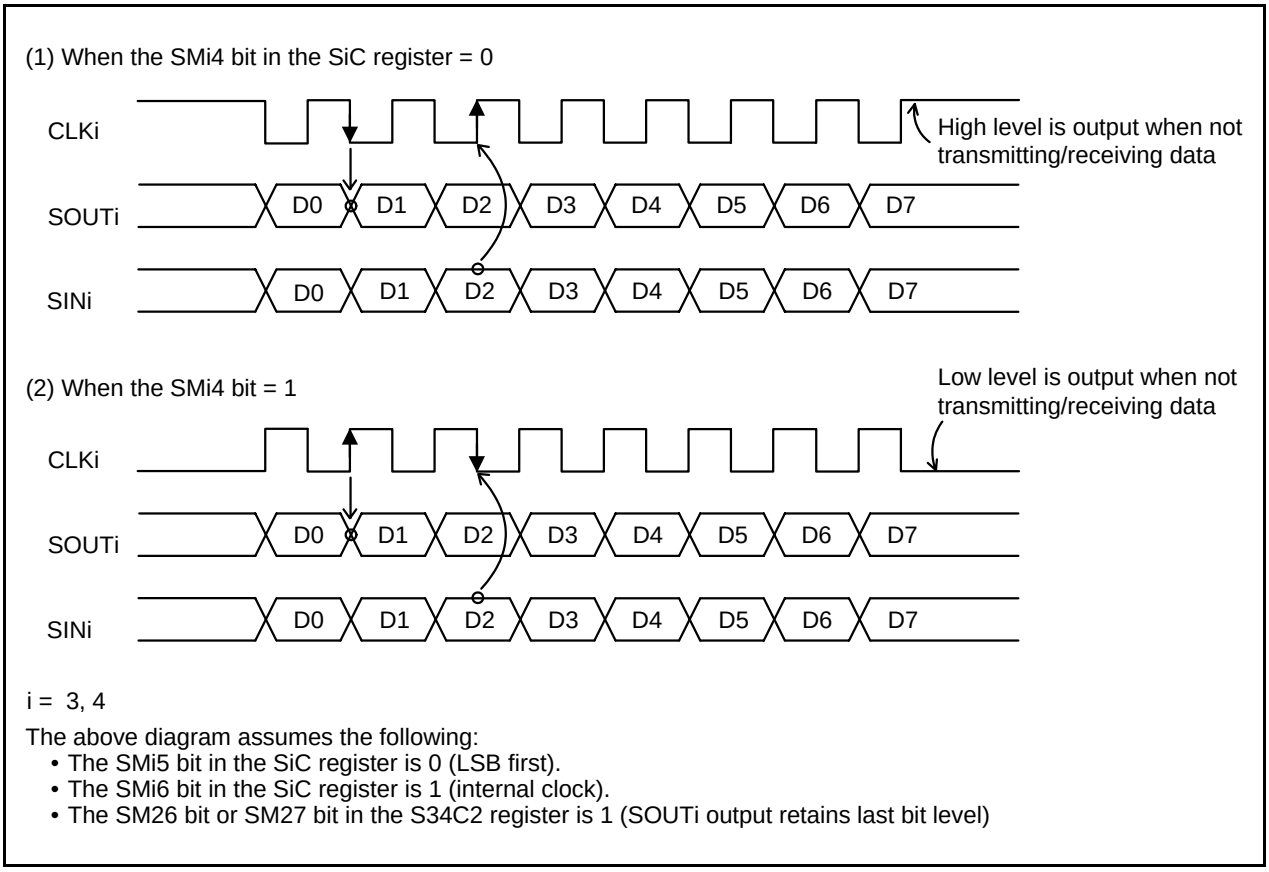


Figure 20.2 Polarity of Transmit/Receive Clock

20.3.3 LSB First or MSB First Selection

Bit order is selected by the SMi5 bit in the SiC register (i = 3, 4). Figure 20.3 shows Bit Order.

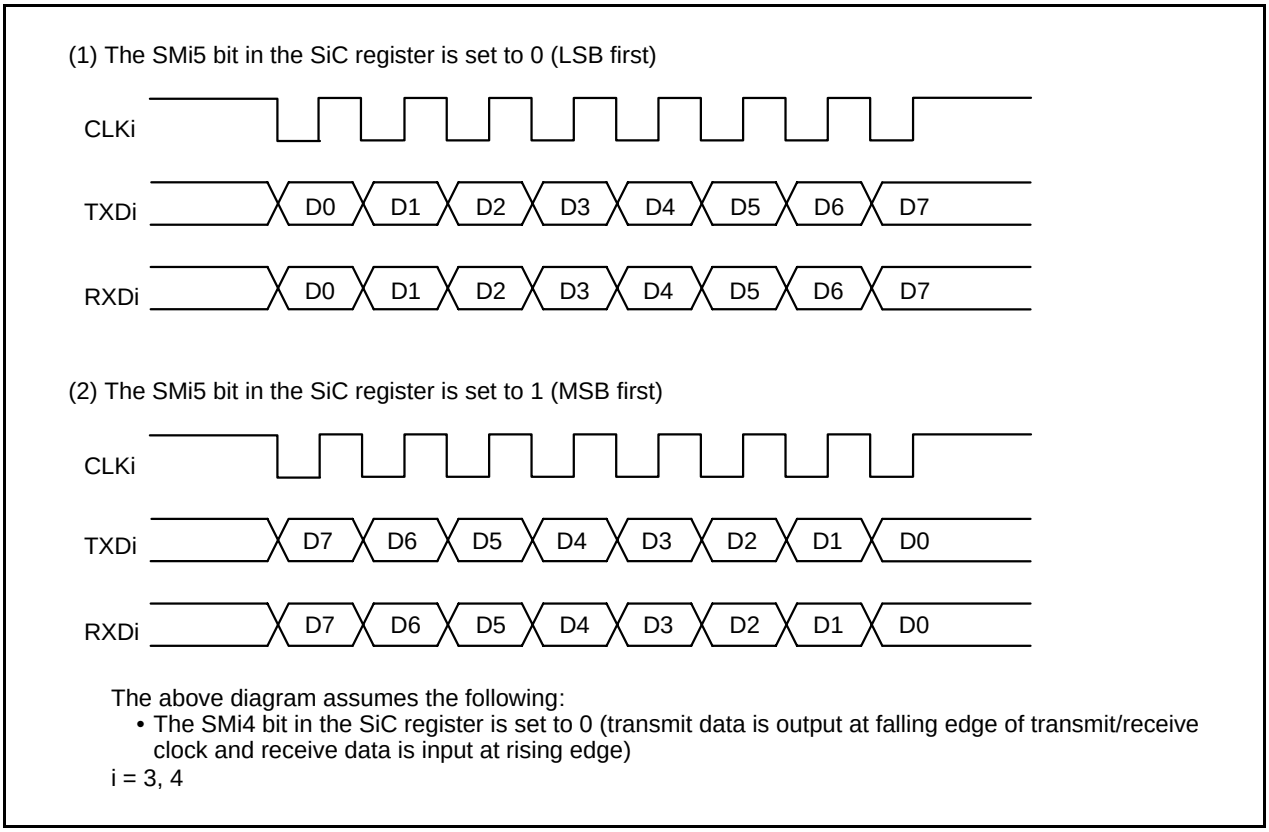


Figure 20.3 Bit Order

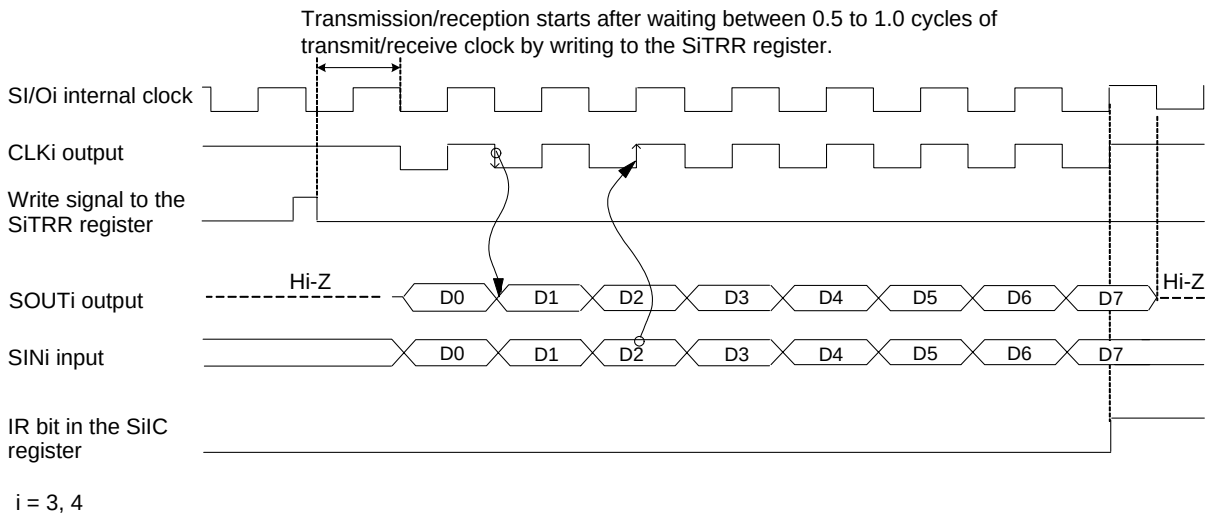
20.3.4 Internal Clock

When the SMi6 bit in the SiC register is 1, data is transmitted/received using the internal clock. The internal clock is selected by the PCLK1 bit in the PLCKR register and bits SMi1 to SMi0 in the SiC register. When using the f1 as the clock source of the internal clock, set the PCKSTP15 bit in the PCLKSTP1 register to 0 (f1 provide enabled).

When the internal clock is used as the transmit/receive clock, the SOUTi pin becomes high-impedance from when the SMi3 bit in the SiC register is set to 1 (SI/Oi enabled) and the SMi2 bit is set to 0 (SOUTi output enabled) to when the first data is output.

When writing transmit data to the SiTRR register, data transmission/reception starts by outputting the transmit/receive clock from the CLKi pin after waiting between 0.5 to 1.0 cycles of the transmit/receive clock. After 8 bits of data have been transmitted/received, the transmit/receive clock from the CLKi pin stops.

Figure 20.4 shows SI/Oi Operation Timing (Internal Clock).



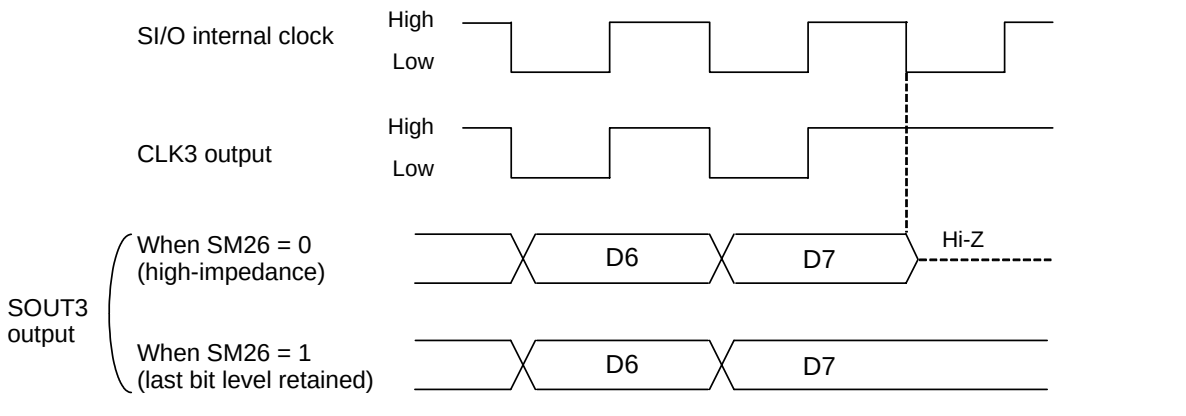
i = 3, 4

The above diagram assumes the following:
In the SiC register, SMi2 = 0 (SOUTi output), SMi3 = 1 (SOUTi output, CLKi function), SMi4 = 0 (transmit data is output at a falling edge of the transmit/receive clock, and receive data is input at a rising edge), SMi5 = 0 (LSB first), SMi6 = 1 (internal clock).
In the S34C2 register, the SM26 bit (SOUT3) or SM27 bit (SOUT4) = 0 (high-impedance after transmission).

Figure 20.4 SI/Oi Operation Timing (Internal Clock)

20.3.5 Function for Selecting SOUTi State after Transmission

The SOUTi pin state after transmission can be selected when the SMi6 bit in the SiC register is set to 1 (internal clock). If bits SM26 and SM27 in the S34C2 register are both set to 1 (last bit level retained), output from the SOUTi pin retains the last bit level after transmission. Figure 20.5 shows SOUT3 Pin Level after Transmission.



The above SOUT3 example assumes the following:
The SM32 bit in the S3C register is 0 (SOUT3 output),
The SM33 bit in the S3C register is 1 (SOUT3 output, CLK3 selected),
The SM34 bit in the S3C register is 0 (transmit data output at the falling edge of the transmit/receive clock and the receive data is input at the rising edge),
The SM35 bit in the S3C register is 0 (LSB first),
The SM36 bit in the S3C register is 1 (internal clock)

Figure 20.5 SOUT3 Pin Level after Transmission

20.3.6 External Clock

When the SMi6 bit in the SiC register is set to 0, data is transmitted/received using the external clock. The external clock is used as transmit/receive clock, the SOUTi output level from when the SMi3 bit in the SiC register is set to 1 (SI/Oi enabled) and SMi2 bit is set to 0 (SOUTi output enabled) to when the first data is output can be selected by the SMi7 bit in the SiC register. Refer to 20.3.8 “Function for Setting SOUTi Initial Value”.

Transmission/reception starts with the external clock after writing the transmit data to the SiTRR register. Data written to the SiTRR register shifts each time the external clock is input. When completing data transmission/reception of the eighth bit, read or write to the SiTRR register before inputting the clock for the next data transmission/reception.

Figure 20.6 shows SI/Oi Operation Timing (External Clock).

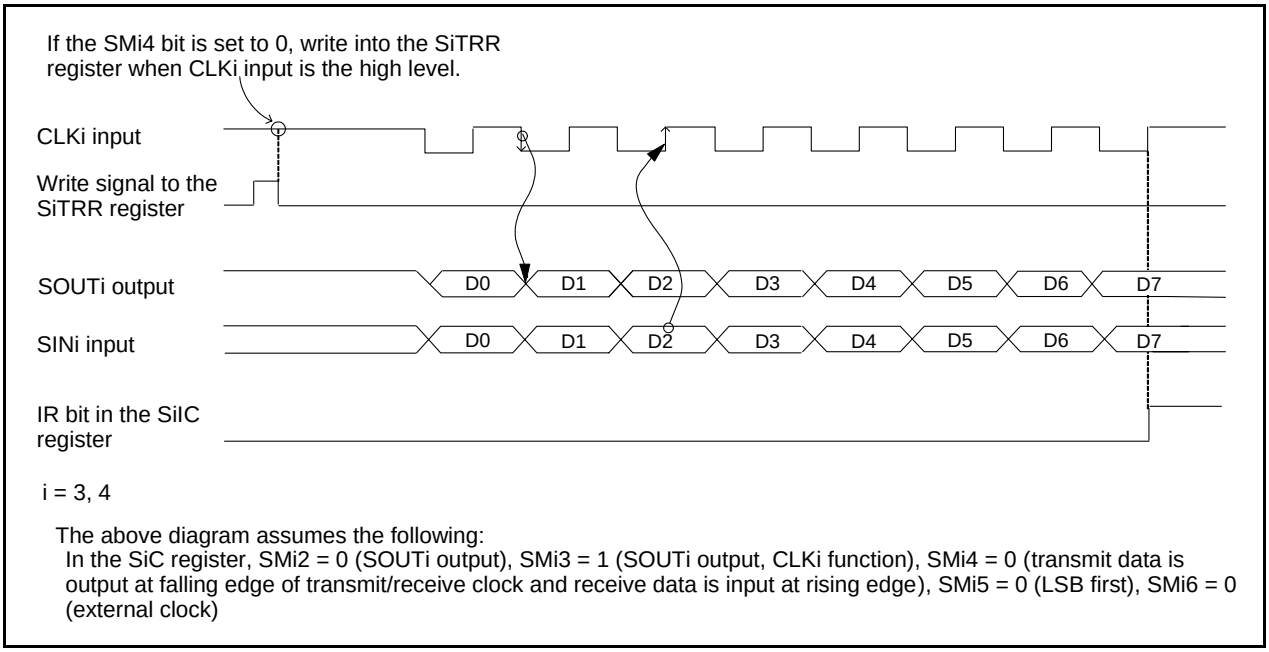


Figure 20.6 SI/Oi Operation Timing (External Clock)

When the SMi6 bit in the SiC register is set to 0 (external clock), write to the SiTRR register and SMi7 bit in the SiC register under the following conditions:

- When the SMi4 bit in the SiC register is set to 0 (transmit data is output at falling edge of transmit/receive clock and receive data is input at rising edge): CLKi input is high.
- When the SMi4 bit is set to 1 (transmit data is output at rising edge of transmit/receive clock and receive data is input at falling edge): CLKi input is low.

20.3.7 SOUTi Pin

The SOUTi pin state can be selected by bits SMi2 and SMi3 in the SiC register.

Table 20.4 lists SOUTi Pin State.

Table 20.4 SOUTi Pin State

Bit Setting		SOUTi Pin State
SiC register		
SMi2	SMi3	
0	0	I/O port or other peripheral function
	1	SOUTi output
1	0/1	High-impedance

20.3.8 Function for Setting SOUTi Initial Value

If the SMi6 bit in the SiC register is set to 0 (external clock), the SOUTi pin output can be fixed high or low when not transmitting/receiving data. High or low can be selected by the SMi7 bit in the SiC register. However, the last bit value of the previous unit of data is retained between adjacent units of data when using the external clock. Figure 20.7 shows Timing Chart for Setting SOUTi Initial Value and How to Set It.

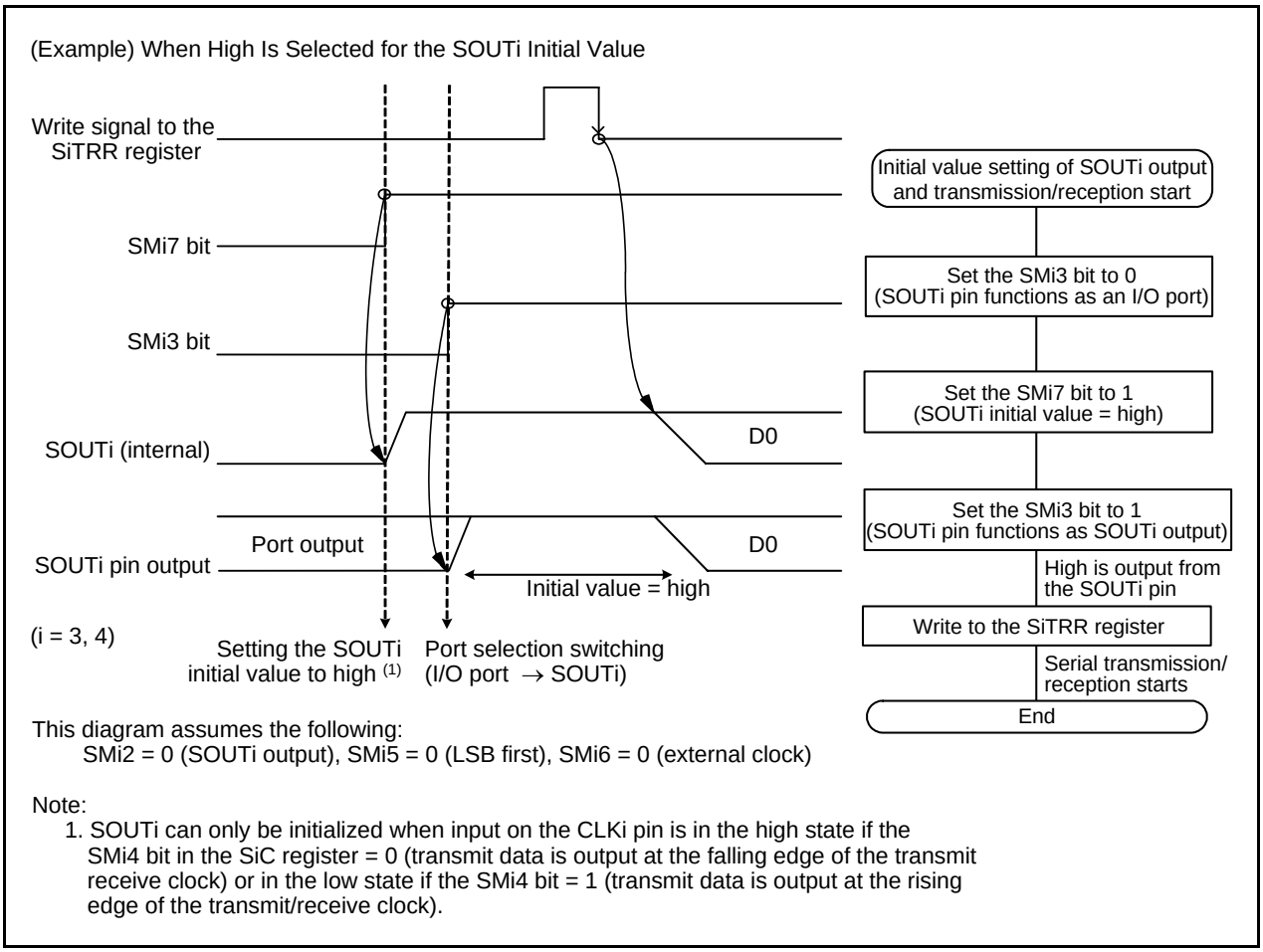


Figure 20.7 Timing Chart for Setting SOUTi Initial Value and How to Set It

20.4 Interrupt

Refer to the operation example for interrupt source or interrupt request generation timing. Refer to 13.7 “Interrupt Control” for interrupt control. Table 20.5 lists Registers Associated with SI/O3 and SI/O4.

Table 20.5 Registers Associated with SI/O3 and SI/O4

Address	Register	Symbol	Reset Value
0048h	SI/O4 Interrupt Control Register	S4IC	XX00 X000b
0049h	SI/O3 Interrupt Control Register	S3IC	XX00 X000b
0207h	Interrupt Source Select Register	IFSR	00h

The interrupts below share the interrupt vector and interrupt control register with other peripheral functions. To use the following interrupts, set bits as follows.

- SI/O3: Set the IFSR6 bit in the IFSR register to 0 (SI/O3).
- SI/O4: Set the IFSR7 bit in the IFSR register to 0 (SI/O4).

Set the POL bit in the SiIC register to 0 (falling edge).

20.5 Notes on Serial Interface SI/O3 and SI/O4

20.5.1 SOUTi Pin Level When SOUTi Output Is Disabled

When the SMi2 bit in the SiC register is set to 1 (SOUTi output disabled), the target pin becomes high-impedance regardless of which pin function being used.

20.5.2 External Clock Control

The data written to the SiTRR register shifts each time the external clock is input. When completing data transmission/reception of the eighth bit, read or write to the SiTRR register before inputting the clock for the next data transmission/reception.

20.5.3 Register Access

Set the SM22 bit in the S34C2 register before setting other registers associated with SI/O3 and SI/O4. After changing the SM22 bit, set other registers associated with SI/O3 and SI/O4 again.

20.5.4 Register Access When Using the External Clock

When the SMi6 bit in the SiC register is 0 (external clock), write to the SMi7 bit in the SiC register and SiTRR register under the following conditions:

- When the SMi4 bit in the SiC register is 0 (transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge): CLKi input is high level.
- When the SMi4 bit in the SiC register is 1 (transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge): CLKi input is low level.

20.5.5 SiTRR Register Access

Write transmit data to the SiTRR register while transmission/reception is stopped. Read receive data from the SiTRR register while transmission/reception is stopped.

The IR bit in the SiIC register becomes 1 (interrupt requested) during output of the eighth bit.

When the SM26 bit (SOUT3) or SM27 bit (SOUT4) in the S34C2 register is 0 (high-impedance after transmission), the SOUTi pin becomes high-impedance when the transmit data is written to the SiTRR register immediately after an interrupt request is generated, and the hold time of the transmit data becomes shorter.

20.5.6 Pin Function Switch When Using the Internal Clock

If the SMi3 bit in the SiC register (i = 3, 4) changes from 0 (I/O port) to 1 (SOUTi output, CLKi function) when setting the SMi2 bit to 0 (SOUTi output) and the SMi6 bit to 1 (internal clock), the SOUTi initial value set to the SOUTi pin by the SMi7 bit may be output for about 10 ns. Then, the SOUTi pin becomes high-impedance.

If the output level from the SOUTi pin when the SMi3 bit changes from 0 to 1 becomes a problem, set the SOUTi initial value by the SMi7 bit.

20.5.7 Operation after Reset When Selecting the External Clock

When the SMi6 bit in the SiC register is 0 (external clock) after reset, the IR bit in the SiIC register becomes 1 (interrupt requested) by inputting the external clock for 8 bits to the CLKi pin. This will also occur even when the SMi3 bit in the SiC register is 0 (serial interface disabled) or before a value is written to the SiTRR register.

21. Multi-Master I2C-bus Interface

21.1 Introduction

The multi-master I2C-bus interface (I2C interface) is a serial communication circuit based on the I2C-bus data transmit/receive format, and is equipped with arbitration lost detect and clock synchronous functions. Table 21.1 lists the Multi-master I2C-bus Interface Specifications, Table 21.2 lists the Detections of I2C Interface, Figure 21.1 shows the Multi-master I2C-bus Interface Block Diagram, and Table 21.3 lists the I/O Ports.

Table 21.1 Multi-master I2C-bus Interface Specifications

Item	Function
Formats	Based on I2C-bus standard: 7-bit addressing format Fast-mode Standard clock mode
Communication modes	Based on I2C-bus standard: Master transmission Master reception Slave transmission Slave reception
Bit rate	16.1 kbps to 400 kbps (fVIIC = 4 MHz)
I/O pins	Serial data line SDAMM (SDA) Serial clock line SCLMM (SCL)
Interrupt request generating sources	• I2C-bus interrupt Completion of transmission Completion of reception Slave address match detection General call detection Stop condition detection Timeout detection • SDA/SCL interrupt Rising or falling edge of the signal of the SDAMM or SCLMM pin
Selectable functions	• I2C-bus interface pin input level select Selectable input level with I2C-bus input level or SMBus input level • SDA/port, SCL/port selection A function to change the SDAMM and SCLMM pins to output ports. • Timeout detection A function that detects when the SCLMM pin is driven high over a certain period of time when the bus is busy. • Free format select A function that generates an interrupt request when receiving the first byte of data, regardless of the slave address value.

fVIIC: I2C-bus system clock

Table 21.2 Detections of I2C Interface

Item	Function
Slave address match	A function to detect a slave address match as a slave transmit or receiver. When own slave address is matched with the calling address sent from a master, ACK is generated automatically. When an address match is not found, no ACK is generated and no more data is transmitted or received. One slave can have up to three slave addresses.
General call	A function to detect a general call in slave reception.
Arbitration lost	A function to detect arbitration lost and stop the output from pins SDAMM and SCLMM.
Bus busy	A function to detect a bus busy state and set/reset the BB bit.

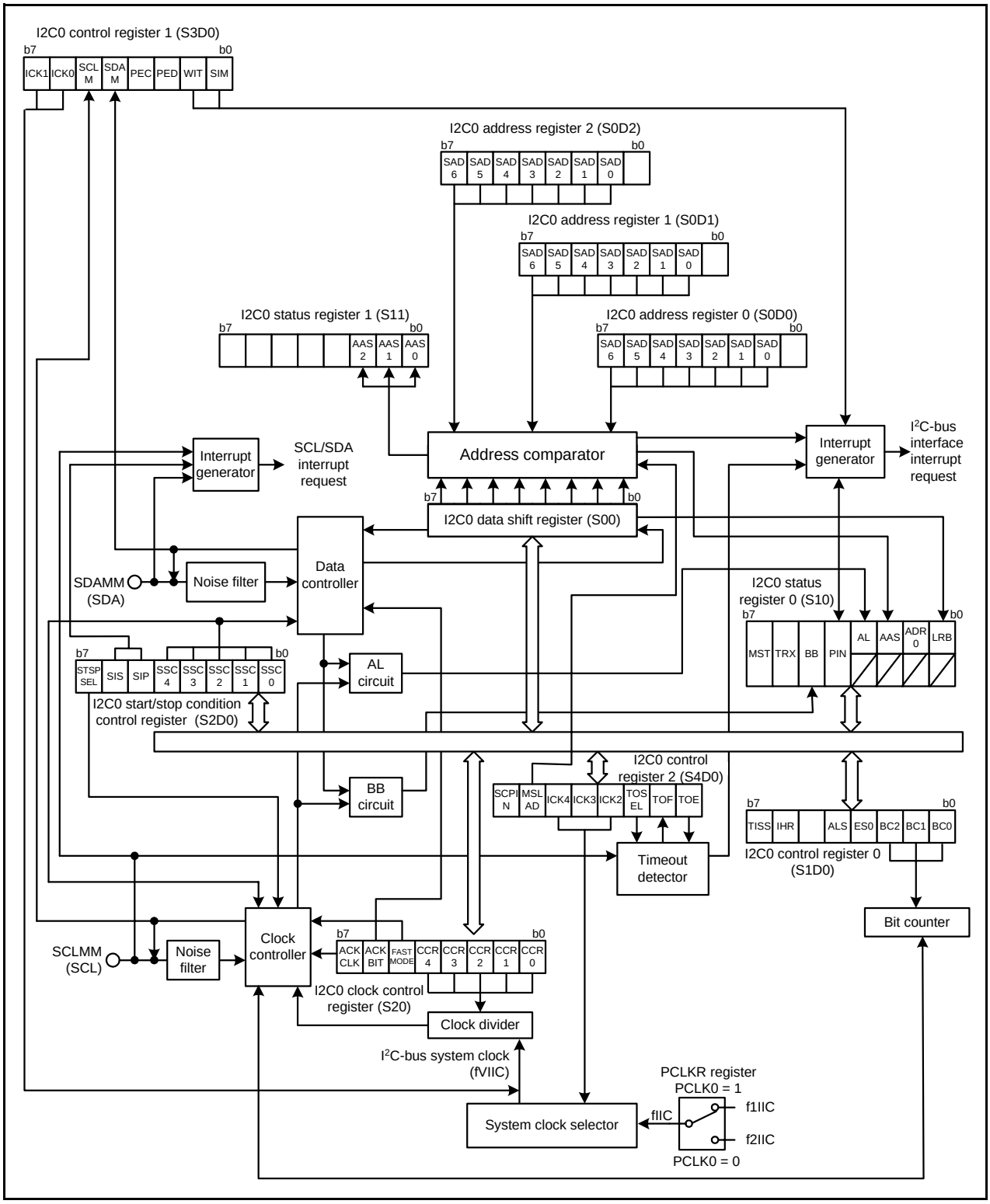


Figure 21.1 Multi-master I2C-bus Interface Block Diagram

Table 21.3 I/O Ports

Pin Name	I/O	Function
SDAMM	I/O	I/O pin for SDA (N-channel open drain output)
SCLMM	I/O	I/O pin for SCL (N-channel open drain output)

21.2 Registers Descriptions

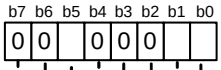
Table 21.4 lists registers associated with multi-master I2C-bus interface. When the CM07 bit in the CM0 register is set to 1 (sub clock is CPU clock), registers listed in Table 21.4 should not be accessed. Set them after the CM07 bit is set to 0 (main clock, PLL clock, or on-chip oscillator clock).

Table 21.4 Register Configuration

Address	Register	Symbol	Reset Value
0012h	Peripheral Clock Select Register	PCLKR	0000 0011b
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
02B0h	I2C0 Data Shift Register	S00	XXh
02B2h	I2C0 Address Register 0	S0D0	0000 000Xb
02B3h	I2C0 Control Register 0	S1D0	00h
02B4h	I2C0 Clock Control Register	S20	00h
02B5h	I2C0 Start/Stop Condition Control Register	S2D0	0001 1010b
02B6h	I2C0 Control Register 1	S3D0	0011 0000b
02B7h	I2C0 Control Register 2	S4D0	00h
02B8h	I2C0 Status Register 0	S10	0001 000Xb
02B9h	I2C0 Status Register 1	S11	XXXX X000b
02BAh	I2C0 Address Register 1	S0D1	0000 000Xb
02BBh	I2C0 Address Register 2	S0D2	0000 000Xb

21.2.1 Peripheral Clock Select Register (PCLKR)

Peripheral Clock Select Register



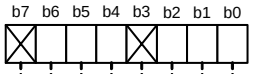
Symbol	Address	After Reset
PCLKR	0012h	0000 0011b

Bit Symbol	Bit Name	Function	RW
PCLK0	Timers A and B clock select bit (clock source for timers A and B, and multi-master I2C-bus interface)	0: f2TIMAB/f2IIC 1: f1TIMAB/f1IIC	RW
PCLK1	SI/O clock select bit (clock source for UART0 to UART2, UART5 to UART7, SI/O3, and SI/O4)	0: f2SIO 1: f1SIO	RW
— (b4-b2)	Reserved bits	Set to 0.	RW
PCLK5	Clock output function expansion bit	0: Selected by bits CM01 to CM00 in the CM0 register 1: Output f1	RW
— (b7-b6)	Reserved bits	Set to 0.	RW

Write to the PCLKR register after setting the PRC0 bit in the PRCR register to 1 (write enabled).

21.2.2 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1



Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I2C peripheral clock stop bit (multi-master I2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

Set the PRC0 bit in the PRCR register to 1 (write enabled) before the PCKSTP1 register is rewritten.

PCKSTP16 (Multi-master I2C peripheral clock stop bit) (b6)

Set the PCKSTP16 bit to 0 (f1 provide enabled) when using multi-master I2C-bus interface.

21.2.3 I2C0 Data Shift Register (S00)

I2C0 Data Shift Register

b7 b0	Symbol	Address	After Reset
	S00	02B0h	Undefined

Function	RW
Transmit/receive data is stored.	RW

When the I2C interface is a transmitter, write transmit data to the S00 register. When the I2C interface is a receiver, received data can be read from the S00 register. In master mode, this register is also used to generate a start condition or stop condition on a bus. (Refer to 21.3.2 “Generating a Start Condition” and 21.3.3 “Generating a Stop Condition”.)

Write to the S00 register when the ES0 bit in the S1D0 register is 1 (I2C interface enabled).

Do not write to the S00 register when transmitting/receiving data.

When the I2C interface is a transmitter, the data in the S00 register is transmitted to other devices. The MSB (bit 7) is transmitted first, synchronizing with the SCLMM clock. Every time 1-bit data is output, the S00 register value is shifted 1 bit to the left.

When the I2C interface is a receiver, data is transferred to the S00 register from other devices. The LSB (bit 0) is input first, synchronizing with the SCLMM clock. Every time 1-bit data is output, S00 register values is shifted 1 bit to the left. Figure 21.2 shows Timing to Store Received Data to the S00 Register.

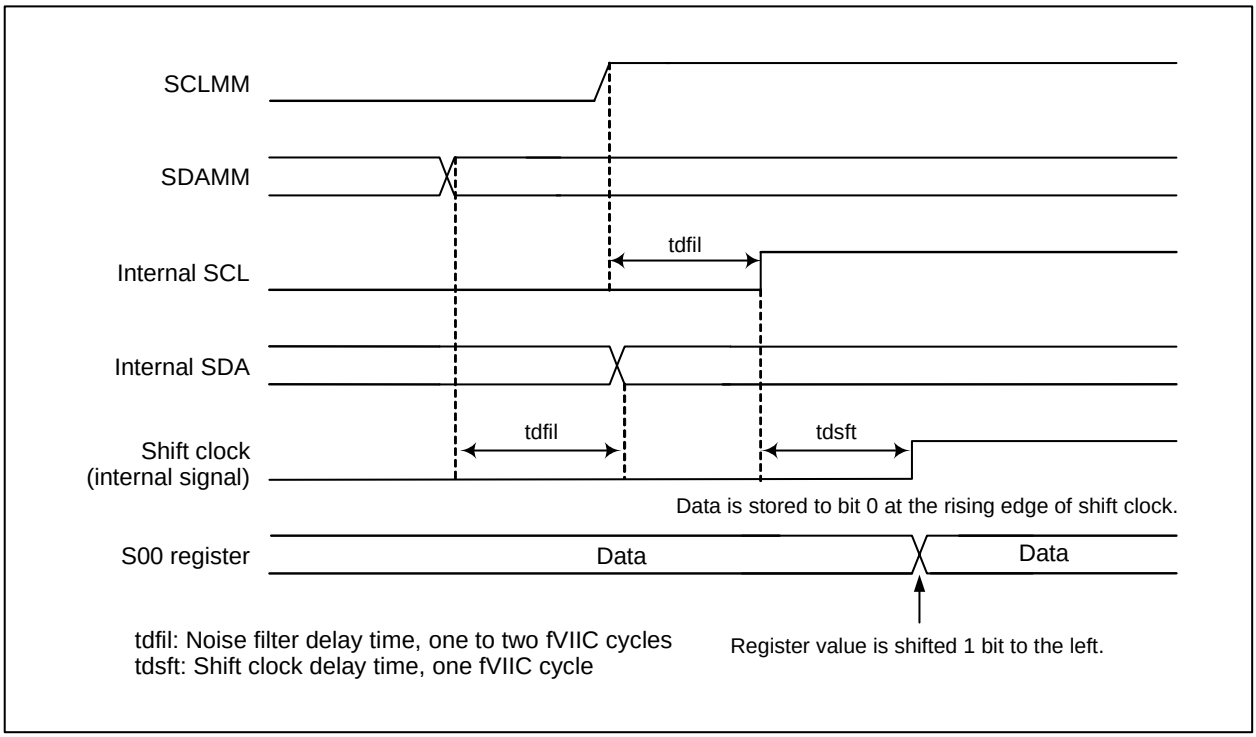
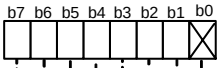


Figure 21.2 Timing to Store Received Data to the S00 Register

21.2.4 I2C0 Address Register i (S0Di) (i = 0 to 2)

I2C0 Address Register i (i = 0 to 2)



Symbol	Address	After Reset
S0D0	02B2h	0000 000Xb
S0D1	02BAh	0000 000Xb
S0D2	02BBh	0000 000Xb

Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit. If necessary, set to 0. Read as undefined value.		—
SAD0	Slave address	Compare with the received address data	RW
SAD1			RW
SAD2			RW
SAD3			RW
SAD4			RW
SAD5			RW
SAD6			RW

SAD6 to SAD0 (Slave address) (b7 to b1)

Bits SAD6 to SAD0 indicate a slave address to be compared for a slave address match detection in slave mode. An I2C interface can have maximum of three slave addresses. Set the S0Di register to 00h when not setting the slave address.

However, when the MSLAD bit in the S4D0 register is 0, registers S0D1 and S0D2 are disabled. Only the slave address set to the S0D0 register is compared with address the data received.

21.2.5 I2C0 Control Register 0 (S1D0)

I2C0 Control Register 0

b7	b6	b5	b4	b3	b2	b1	b0
		0					

Symbol	Address	After Reset
S1D0	02B3h	00X0 0000b

Bit Symbol	Bit Name	Function	RW
BC0	Bit counter (number of transmitted/received bits)	b2 b1 b0 0 0 0: 8 0 0 1: 7 0 1 0: 6 0 1 1: 5 1 0 0: 4 1 0 1: 3 1 1 0: 2 1 1 1: 1	RW
BC1			RW
BC2			RW
ES0	I2C-bus interface enable bit	0: Disabled 1: Enabled	RW
ALS	Data format select bit	0: Addressing format 1: Free data format	RW
— (b5)	Reserved bit	Set to 0.	RW
IHR	I2C-bus interface reset bit	0: Reset is deasserted (automatically) 1: Reset	RW
TISS	I2C-bus interface pin input level select bit	0: I2C-bus input 1: SMBus input	RW

BC2 to BC0 (Bit counter) (b2 to b0)

Bits BC2 to BC0 become 000b (8 bits) when start condition is detected.
When the ACKCLK bit in the S20 register is 0 (no ACK clock), and data for the number of bits selected by bits BC2 to BC0 is transmitted or received, bits BC2 to BC0 become 000b again.
When the ACKCLK bit in the S20 register is 1 (ACK clock), and data for the number of bits selected and an ACK is transmitted or received, bits BC2 to BC0 become 000b again.

ES0 (I2C-bus interface enable bit) (b3)

The ES0 bit enables the I2C interface.

When the ES0 bit is set to 0, the I2C interface status becomes as follows:

- Pins SDAMM and SCLMM: I/O ports or other peripheral pins
- The S00 register is write disabled.
- The I2C-bus system clock (hereinafter called fVIIC) stops.
- S10 register
  - ADR0 bit: 0 (general call not detected)
  - AAS bit: 0 (slave address not matched)
  - AL bit: 0 (arbitration lost not detected)
  - PIN bit: 1 (no I2C-bus interrupt request)
  - BB bit: 0 (bus free)
  - TRX bit: 0 (receive mode)
  - MST bit: 0 (slave mode)
- Bits AAS2 to AAS0 in the S11 register: 0 (slave address not matches)
- The TOF bit in the S4D0 register: 0 (timeout not detected)

ALS (Data format select bit) (b4)

The ALS bit is enabled in slave mode. When the ALS bit is 0 (addressing format), the slave address match detection is performed.

When one of the slave addresses stored to bits SAD6 to SAD0 in the S0Di register (i = 0 to 2) is compared and matched with the calling address by a master, or when a general call address is received, the IR bit in the IICIC register becomes 1 (interrupt requested).

When the ALS bit is 1 (free format), the slave address match detection is not performed. Therefore, the IR bit in the IICIC register becomes 1 (interrupt requested), regardless of the calling address by a master.

IHR (I2C-bus interface reset bit) (b6)

The IHR bit resets the I2C interface if a difficulty in transmission/reception is encountered. When the ES0 bit in the S1D0 register is 1 (I2C interface enabled) and then the IHR bit is set to 1 (reset), the I2C interface becomes as follows:

- S10 register
  - ADR0 bit: 0 (general call not detected)
  - AAS bit: 0 (slave address not matched)
  - AL bit: 0 (arbitration lost not detected)
  - PIN bit: 1 (No I2C-bus interrupt request)
  - BB bit: 0 (bus free)
  - TRX bit: 0 (receive mode)
  - MST bit: 0 (slave mode)
- Bits AAS2 to AAS0 in the S11 register: 0 (slave address not matches)
- TOF bit in the S4D0 register: 0 (timeout not detected)

When the IHR bit is set to 1, the I2C interface is reset and the IHR bit becomes 0 automatically. It takes maximum of 2.5 fVIIC cycles to complete the reset sequence.

Figure 21.3 shows the I2C Interface Reset Timing.

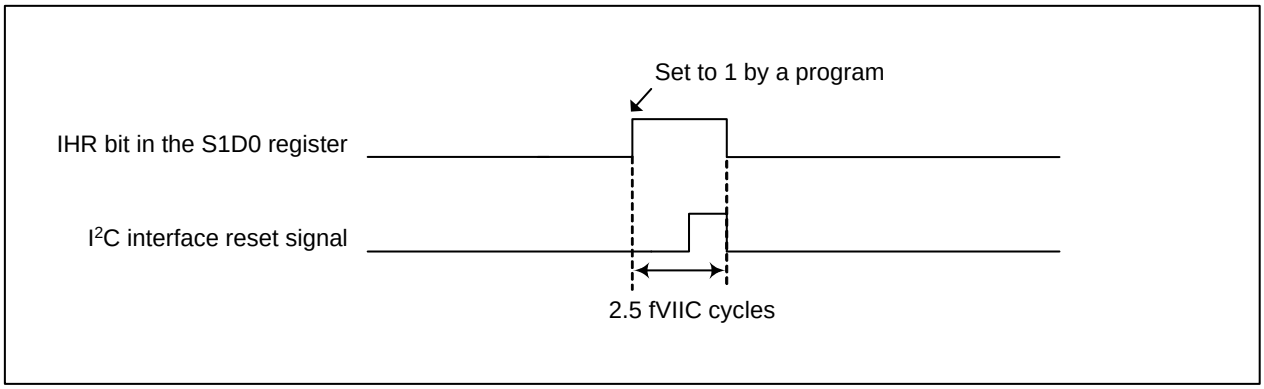


Figure 21.3 I2C Interface Reset Timing

TISS (I2C-bus interface pin input level select bit) (b7)

The TISS bit selects the input level of the SCLMM pin and SDAMM pin for the I2C interface.

21.2.6 I2C0 Clock Control Register (S20)

I2C0 Clock Control Register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
								S20	02B4h	00h

Bit Symbol	Bit Name	Function	RW
CCR0	Bit rate control bit	Refer to bits CCR4 to CCR0 (Bit Rate Control Bit) (b4 to b0) in the next page.	RW
CCR1			RW
CCR2			RW
CCR3			RW
CCR4			RW
FASTMODE	SCL mode select bit	0: Standard-speed clock mode 1: Fast-mode	RW
ACKBIT	ACK bit	0: ACK is returned 1: ACK is not returned	RW
ACKCLK	ACK clock bit	0: No ACK clock present 1: ACK clock present	RW

CCR4 to CCR0 (Bit rate control bit) (b4 to b0)

The setting range of bits CCR4 to CCR0 (CCR value) is 0 to 31. If the value set to bits CCR4 to CCR0 is the CCR value (CCR value: 3 to 31), the bit rate can be determined by the equations below. Refer to 21.3.1.2 “Bit Rate and Duty Cycle” for more details.

In standard-speed clock mode,

$$\text{Bit rate} = \frac{\text{fVIIC}}{8 \times \text{CCR value}} \leq 100 \text{ kbps}$$

When the CCR value is other than 5 in fast-mode,

$$\text{Bit rate} = \frac{\text{fVIIC}}{4 \times \text{CCR value}} \leq 400 \text{ kbps}$$

When the CCR value is 5 in fast-mode, the bit rate is assumed to reach 400 kbps, the maximum bit rate in fast-mode.

$$\text{Bit rate} = \frac{\text{fVIIC}}{2 \times \text{CCR value}} = \frac{\text{fVIIC}}{10} \leq 400 \text{ kbps}$$

Do not set the CCR value from 0 to 2 regardless of the fVIIC frequency.
Rewrite bits CCR4 to CCR0 when the ES0 bit in the S1D0 register is 0 (disabled).

FASTMODE (SCL mode select bit) (b5)

When using the fast-mode, I2C-bus standard (400 kbps maximum), set the FASTMODE bit to 1 (fast-mode) and set fVIIC to 4 MHz or more.
Do not rewrite this bit when transmitting/receiving data.

ACKBIT (ACK bit) (b6)

The ACK bit is enabled in a master reception, slave reception, or slave address reception. When receiving a slave address, the SDAMM pin level during the ACK clock pulse is determined by a combination of bits ALS and ACKBIT in the S1D0 register and the received slave address.
When receiving data, the SDAMM pin level during the ACK clock pulse is determined by the ACKBIT bit. Table 21.5 lists the SDAMM Pin Level during the ACK Clock Pulse.

Table 21.5 SDAMM Pin Level during the ACK Clock Pulse

Received Content	ALS Bit in the S1D0 Register	ACKBIT Bit in the S20 Register	Slave Address Content	SDAMM Pin Level at ACK Clock
Slave Address	0	0	When the MSLAD bit in the S4D0 register is 0: Matched with bits SAD6 to SAD0 in the S0D0 register. When the MSLAD bit is 1: Matched with bits SAD6 to SAD0 in any of registers S0D0 to S0D2.	Low (ACK)
			0000000b	Low (ACK)
			Others	High (NACK)
		1	—	High (NACK)
	1	0	—	Low (ACK)
		1	—	High (NACK)
Data	—	0	—	Low (ACK)
		1	—	High (NACK)

ACKCLK (ACK clock bit) (b7)

When the ACKCLK bit is 1 (ACK clock present), an ACK clock is generated immediately after 1-byte data is transmitted or received (8 clocks).
When the ACKCLK bit is 0 (no ACK clock), no ACK clock is generated after 1-byte data is transmitted or received (8 clocks). At the falling edge of data transmission/reception (the falling edge of the eighth clock), the IR bit in the IICIC register becomes 1 (interrupt requested).
Do not write to this bit when transmitting/receiving data.

21.2.7 I2C0 Start/Stop Condition Control Register (S2D0)

I2C0 Start/Stop Condition Control Register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	After Reset
S2D0	02B5h	0001 1010b

Bit Symbol	Bit Name	Function	RW
SSC0	Start/stop condition setting bit	Refer to SSC4 to SSC0 (Start/Stop Condition Setting Bit) (b4 to b0) in the same page.	RW
SSC1			RW
SSC2			RW
SSC3			RW
SSC4			RW
SIP	SCL/SDA interrupt pin polarity select bit	0: Falling edge 1: Rising edge	RW
SIS	SCL/SDA interrupt pin select bit	0: SDAMM 1: SCLMM	RW
STSPSEL	Start/stop condition generation select bit	0: Short setup/hold time mode 1: Long setup/hold time mode	RW

SSC4 to SSC0 (Start/stop condition setting bit) (b4 to b0)

Bits SSC4 to SSC0 select the start/stop condition detect parameter (SCL open time, setup time, hold time) in standard-speed clock mode. Refer to 21.3.7 “Detecting Start/Stop Conditions”.
Do not set an odd values or 00000b to bits SSC4 to SSC0.

SIP (SCL/SDA interrupt pin polarity select bit) (b5)
SIS (SCL/SDA interrupt pin select bit) (b6)

The IR bit in the SCLDAIC register becomes 1 (interrupt requested) when the I2C interface detects the edge selected by the SIP bit for the pin signal selected by the SIS bit. Refer to 21.4 “Interrupts”.

STSPSEL (Start/stop condition generation select bit) (b7)

See Table 21.13 “Setup/Hold Time for Generating a Start/Stop Condition”.
If the fVIIC frequency is more than 4 MHz, set the STSPSEL bit to 1 (long mode).

21.2.8 I2C0 Control Register 1 (S3D0)

I2C0 Control Register 1

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	After Reset
S3D0	02B6h	0011 0000b

Bit Symbol	Bit Name	Function	RW
SIM	Stop condition detect interrupt enable bit	0: I2C-bus interrupt by stop condition detection is disabled 1: I2C-bus interrupt by stop condition detection is enabled	RW
WIT	Data receive interrupt enable bit	When write, 0: I2C-bus interrupt at 8th clock is disabled 1: I2C-bus interrupt is enabled at 8th clock When read, internal WAIT bit monitor 0: I2C-bus interrupt by falling edge of ACK clock 1: I2C-bus interrupt at 8th clock	RW
PED	SDAMM/port function select bit	0: SDAMM I/O pin 1: Port output pin	RW
PEC	SCLMM/port function select bit	0: SCLMM I/O pin 1: Port output pin	RW
SDAM	Internal SDA output monitor bit	0: Logic 0 output 1: Logic 1 output	RO
SCLM	Internal SCL output monitor bit	0: Logic 0 output 1: Logic 1 output	RO
ICK0	I2C-bus system clock select bit (enabled when bits ICK4 to ICK2 in the S4D0 register are 000b)	b7 b6 0 0: fIIC divided by 2 0 1: fIIC divided by 4 1 0: fIIC divided by 8 1 1: Do not set this value	RW
ICK1			RW

Do not use the bit managing instruction (read-modify-write instruction) to access the S3D0 register. Use the MOV instruction to write to the S3D0 register.

SIM (Stop condition detect interrupt enable bit) (b0)

When the SIM bit is 1 (I2C-bus interrupt by stop condition detection enabled) and a stop condition is detected, the SCPIN bit in the S4D0 register becomes 1 (stop condition detect interrupt requested) and the IR bit in the IICIC register becomes 1 (interrupt requested).

WIT (Data receive interrupt enable bit) (b1)

The WIT bit is enabled in master reception or slave reception.
The WIT bit has two functions.

- Select the I2C-bus interrupt timing when data is received. (write)
- Monitor the state of the internal WAIT flag. (read)

The following describes each function.

The WIT bit can be set so an I2C-bus interrupt request is generated at the eighth clock (before the ACK clock) when receiving data.
When the ACKCLK bit in the S20 register is 1 (ACK clock) and the WIT bit is set to 1 (enable I2C-bus interrupt at 8th clock), an I2C-bus interrupt request is generated at the eighth clock (before the ACK clock). Then, the PIN bit in the S10 register becomes 0 (interrupt requested).
When the ACKCLK bit in the S20 register is 0 (no ACK clock), write a 0 to the WIT bit to disable the I2C-bus interrupt by data reception.
When transmitting data and receiving a slave address, no interrupt requests be generated at the eighth clock (before the ACK clock) regardless of the value written to the WIT bit.

Reading the WIT bit returns the internal WAIT flag status.
An I2C-bus interrupt request is generated at the falling edge of the ninth clock (ACK clock) regardless of the value written to the WIT bit. Then, the PIN bit in the S10 register becomes 0 (interrupt requested).
Therefore, read the internal WAIT flag status to determine whether the I2C-bus interrupt request is generated at the eighth clock (before the ACK clock) or at the falling edge of the ACK clock.
When the WIT bit is set to 1 (I2C-bus interrupt enabled by receiving data), the internal WAIT flag changes under the following condition.
Condition to become 0:

- The S20 register (ACKBIT bit) is written.

Condition to become 1:

- The S00 register is written.

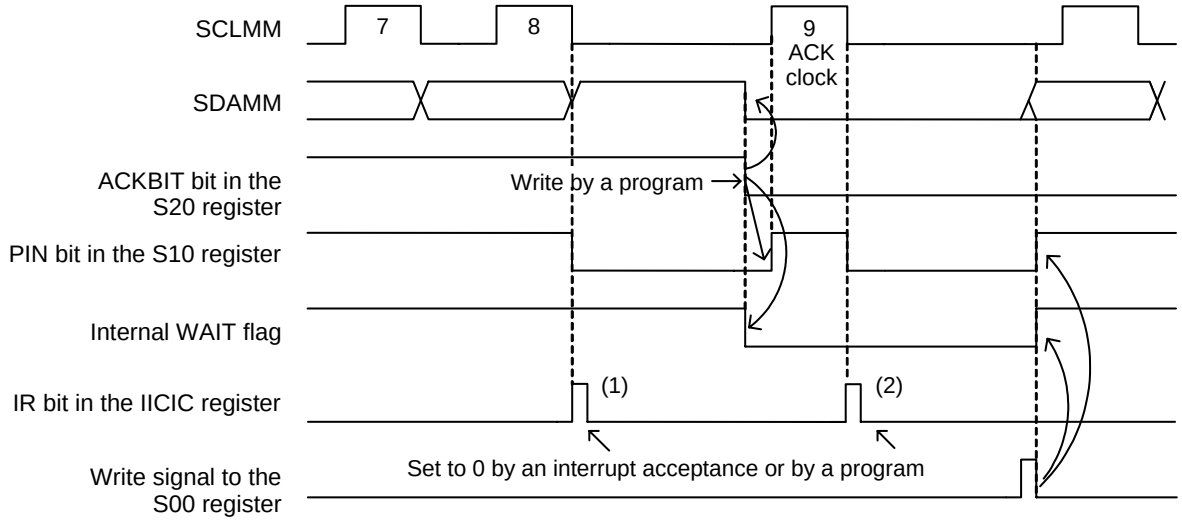
When transmitting data and receiving a slave address, the internal WAIT flag is 0 and an I2C-bus interrupt request will be generated only at the falling edge of the ninth clock (ACK clock), regardless of the value written to the WIT bit.
Table 21.6 lists interrupt request generation timing and the conditions to restart transmission/reception when receiving data. Figure 21.4 shows Interrupt Request Generation Timing in Receive Mode.

Table 21.6 Generating an Interrupt Request and Restarting Transmission/Reception When Receiving Data

I2C-bus Interrupt Request Generation Timing	Internal WAIT Flag Status	Conditions to Restart Transmission/Reception
At the falling edge of the eighth clock (before the ACK clock) (1)	1	Write to the ACKBIT bit in the S20 register (3)
At the falling edge of the ninth clock (ACK clock) (2)	0	Write to the S00 register

Notes:
1. See the timing of (1) on the IR bit in the IICIC register in Figure 21.4.
2. See the timing of (2) on the IR bit in the IICIC register in Figure 21.4.
3. When setting the ACKBIT bit, do not rewrite any other bits and do not set the S00 register.

When setting the WIT bit to 1 in receive mode, and the ACK clock is present:
(I2C-bus interrupt is enabled at eighth clock)



When setting the WIT bit to 0 in receive mode, and the ACK clock is present:
(I2C-bus interrupt is disabled at eighth clock)

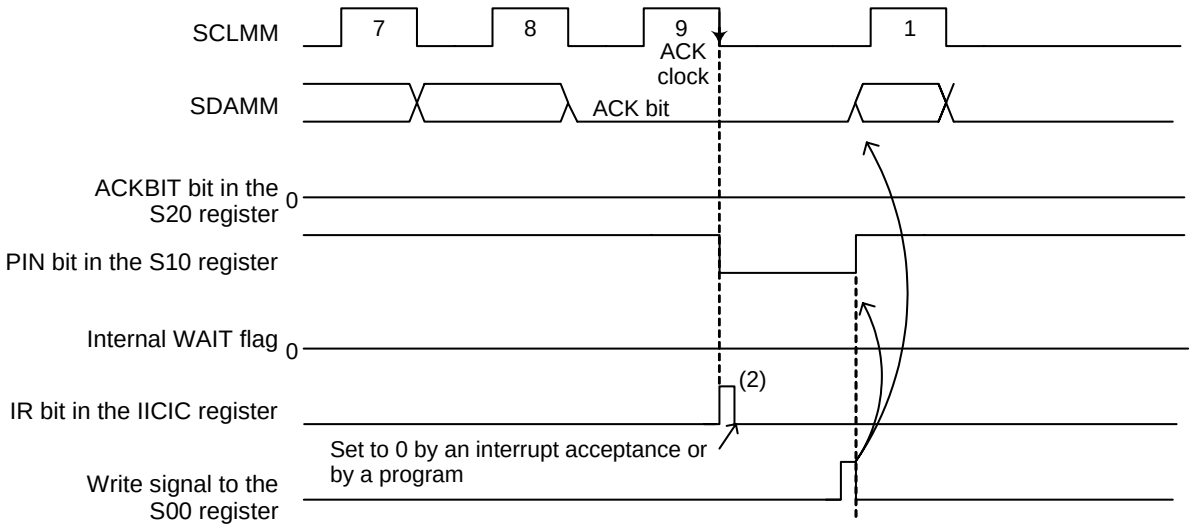


Figure 21.4 Interrupt Request Generation Timing in Receive Mode

PED (SDAMM/port function switch bit) (b2)
PEC (SCLMM/port function switch bit) (b3)

Bits PEC and PED are enabled when the ES0 bit in the S1D0 register is set to 1 (I2C interface enabled).
When the PEC bit is set to 1 (output port), the P7_1 bit value is output from the SCLMM pin regardless of the internal SCL output signal and PD7_1 bit value. When the PED bit is set to 1 (output port), the P7_0 bit value is output from the SDAMM pin regardless of the internal SDA output signal and PD7_0 bit value.
The signal level on the bus is input to the internal SDA and internal SCL.
When bits P7_1 to P7_0 in the P7 register are read after setting bits PD7_1 and PD7_0 in the PD7 register to 0 (input mode), the level on the bus can be read regardless of the values set to bits PED and PEC. Table 21.7 lists SCLMM and SDAMM Pin Functions.

Table 21.7 SCLMM and SDAMM Pin Functions

Pin	S1D0 Register	S3D0 Register		Pin Function
	ES0 bit	PED bit	PEC bit	
P7_1/SCLMM	0	—	—	I/O port or other peripheral pins
	1	—	0	SCLMM (SCL input/output)
		—	1	Output port (output P7_1 bit value)
P7_0/SDAMM	0	—	—	I/O port or other peripheral pins
	1	0	—	SDAMM (SDA input/output)
		1	—	Output port (output P7_0 bit value)

–: 0 or 1

SDAM (Internal SDA output monitor bit) (b4)
SCLM (Internal SCL output monitor bit) (b5)

The internal SDA and SCL output signal levels are the same as the output level of the I2C interface before it has any effect from the external device output. Bits SDAM and SCLM are read only bits. Should be written with 0.

ICK1 to ICK0 (I2C-bus system clock select bit) (b7 to b6)

Bits ICK1 to ICK0 should be rewritten when the ES0 bit in the S1D0 register is 0 (I2C interface disabled). fVIIC is selected by setting all the bits ICK1 to ICK0, bits ICK4 to ICK2 in the S4D0 register, and the PCLK0 bit in the PCLKR register. Refer to 21.3.1.2 “Bit Rate and Duty Cycle”.

Table 21.8 I2C-bus System Clock Select Bits

S4D0 Register			S3D0 Register		fVIIC
ICK4 Bit	ICK3 Bit	ICK2 Bit	ICK1 Bit	ICK0 Bit	
0	0	0	0	0	fIIC divided-by-2
0	0	0	0	1	fIIC divided-by-4
0	0	0	1	0	fIIC divided-by-8
0	0	1	–	–	fIIC divided-by-2.5
0	1	0	–	–	fIIC divided-by-3
0	1	1	–	–	fIIC divided-by-5
1	0	0	–	–	fIIC divided-by-6

–: 0 or 1
Only set the values listed above.

21.2.9 I2C0 Control Register 2 (S4D0)

I2C0 Control Register 2

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
								S4D0	02B7h	00b

Bit Symbol	Bit Name	Function	RW
TOE	Timeout detect function enable bit	0: Disabled 1: Enabled	RW
TOF	Timeout detect flag	0: Not detected 1: Detected	RO
TOSEL	Timeout detect time select bit	0: Long time 1: Short time	RW
ICK2	I2C-bus system clock select bit	b5 b4 b3 0 0 0: Bits ICK1 to ICK0 in the S3D0 register are enabled 0 0 1: fIIC divided by 2.5 0 1 0: fIIC divided by 3 0 1 1: fIIC divided by 5 1 0 0: fIIC divided by 6 Only set the values listed above.	RW
ICK3			RW
ICK4			RW
MSLAD	Slave address control bit	0: S0D0 register only 1: Registers S0D0 to S0D2	RW
SCPIN	Stop condition detect interrupt request bit	0: I2C-bus interrupt not requested 1: I2C-bus interrupt requested	RW

TOE (Timeout detect function enable bit) (b0)

The TOE bit enables the timeout detect function. Refer to 21.3.9 “Timeout Detection” for details.

TOF (Timeout detect flag) (b1)

The TOF bit is enabled when the TOE bit is set to 1. When the TOF bit is set to 1 (detected), the IR bit in the IICIC register is set to 1 (interrupt requested) at the same time.

Condition to become 0:

- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

Condition to become 1:

- The BB bit in the S10 register is set to 1 (bus busy) and the SCLMM high period is greater than the timeout detect period.

TOSEL (Timeout detect time select bit) (b2)

Set the TOSEL bit to select a timeout detection period. The TOSEL bit is enabled when the TOE bit is 1 (timeout detect function enabled).
When long time is selected, the internal counter increments fVIIC as a 16-bit counter. When short time is selected, the internal counter increments fVIIC as a 14-bit counter. Therefore, the timeout detect time is as follows:
When the TOSEL bit is set to 0 (long time),

$$65536 \times \frac{1}{fVIIC}$$

When the TOSEL bit is set to 1 (short time),

$$16384 \times \frac{1}{fVIIC}$$

Table 21.9 lists Timeout Detect Time.

Table 21.9 Timeout Detect Time

fVIIC	Timeout Detect	
	TOSEL bit: 0 (Long time)	TOSEL bit: 1 (Short time)
4 MHz	16.4 ms	4.1 ms
2 MHz	32.8 ms	8.2 ms
1 MHz	65.6 ms	16.4 ms

Rewrite this bit when the TOE bit is 0.

ICK4 to ICK2 (I2C-bus system clock select bit) (b5 to b3)

Bits ICK4 to ICK2 should be rewritten when the ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
fVIIC is selected by setting all the bits ICK4 to ICK2, bits ICK1 to ICK0 in the S3D0 register, and the PCLK0 bit in the PCLKR register. Refer to Table 21.8 “I2C-bus System Clock Select Bits” and 21.3.1.2 “Bit Rate and Duty Cycle”.

MSLAD (Slave address compare bit) (b6)

The MSLAD bit is enabled when the ALS bit in the S1D0 register is set to 0 (addressing format). The MSLAD bit selects the S0Di register (i = 0 to 2) used for address match detection.

SCPIN (Stop condition detect interrupt request bit) (b7)

The SCPIN bit is enabled when the SIM bit in the S3D0 register is set to 1 (enable I2C-bus interrupt by stop condition detection).
Condition to become 0:

- Writing a 0 by a program.

Condition to become 1:

- Stop condition is detected
  (writing a 1 by a program has no effect).

21.2.10 I2C0 Status Register 0 (S10)

I2C0 Status Register 0

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	After Reset
S10	02B8h	0001 000Xb

Bit Symbol	Bit Name	Function	RW
LRB	Last receive bit	When read, 0: Last bit = 0 1: Last bit = 1 When write, see Table 21.10 Functions enabled by S10 register.	RW
ADR0	General call detect flag	When read, 0: Not detected 1: Detected When write, see Table 21.10 Functions enabled by S10 register.	RW
AAS	Slave address compare flag	When read, 0: Address not matched 1: Address matched When write, see Table 21.10 Functions enabled by S10 register.	RW
AL	Arbitration lost detect flag	When read, 0: Not detected 1: Detected When write, see Table 21.10 Functions enabled by S10 register.	RW
PIN	I2C-bus interface interrupt request bit	When read, 0: Interrupt requested 1: Interrupt not requested When write, see Table 21.10 Functions enabled by S10 register.	RW
BB	Bus busy flag	When read, 0: Bus free 1: Bus busy When write, see Table 21.10 Functions enabled by S10 register.	RW
TRX	Communication mode select bit 0	0: Receive mode 1: Transmit mode	RW
MST	Communication mode select bit 1	0: Slave mode 1: Master mode	RW

Do not use the bit managing instruction (read-modify-write instruction) to access the S10 register. Use the MOV instruction to write to the S10 register.

Bit 5 to bit 0 in the S10 register (six lower bits) monitors the state of the I2C interface. The bit values cannot be changed by a program. However, a write to the S10 register, including the six lower bits, is used to generate a start/stop condition.

Bits MST and TRX are read and write bits. To change bits MST or TRX without generating a start/ stop condition, set 1111b to four lower bits in the S10 register.

Table 21.10 lists Functions Enabled by Writing to the S10 Register. Only set the values listed in Table 21.10. If the values listed in Table 21.10 are written to the S10 register, the six lower bits in the S10 register will not be changed.

Table 21.10 Functions Enabled by Writing to the S10 Register

Bit Setting of the S10 Register								Function
MST	TRX	BB	PIN	AL	AAS	ADR0	LRB	
1	1	1	0	0	0	0	0	Sets the I2C interface to start condition standby state in master transmit/receive mode
1	1	0	0	0	0	0	0	Sets the I2C interface to stop condition standby state in master transmit/receive mode
0	0	–	0	1	1	1	1	Slave receive mode
0	1	–	0	1	1	1	1	Slave transmit mode
1	0	–	0	1	1	1	1	Master receive mode
1	1	–	0	1	1	1	1	Master transmit mode

–: 0 or 1

Refer to 21.3.2 “Generating a Start Condition” and 21.3.3 “Generating a Stop Condition” for start/stop conditions.

LRB (Last receive bit) (b0)

When read, the LRB bit functions as described below. See Table 21.10 “Functions Enabled by Writing to the S10 Register” for the bit function in write access.

The LRB bit stores the value of the last bit of the received data. It is used to check if ACK is received. The bit becomes 0 after writing to the S00 register.

ADR0 (General call detect flag) (b1)

The ADR0 bit function in read access is described as follows. See Table 21.10 “Functions Enabled by Writing to the S10 Register” for the bit function in write access.

Condition to become 0:

- Stop condition is detected.
- Start condition is detected.
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

Condition to become 1:

- The ALS bit in the S1D0 register is set to 0 (addressing format) and the received slave address is 0000000b (general call) in slave mode.

AAS (Slave address compare flag) (b2)

The AAS bit function in read access is described as follows. See Table 21.10 “Functions Enabled by Writing to the S10 Register” for the bit function in write access.

Condition to become 0:

- The S00 register is written.
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

Condition to become 1:

- In slave receive mode, the MSLAD bit in the S4D0 register is 1 (registers S0D0 to S0D2), the ALS bit in the S1D0 register is 0 (addressing format), and the received slave address is matched with bits SAD6 to SAD0 in any registers from S0D0 to S0D2.
- In slave receive mode, the MSLAD bit is 0, the ALS bit in the S1D0 register is 0 (addressing format), and the received slave address is matched with bits SAD6 to SAD0 in the S0D0 register.
- In slave receive mode, the ALS bit in the S1D0 register is set to 0 (addressing format) and the received address is 00000000b (general call).

AL (Arbitration lost detect flag) (b3)

The AL bit function in read access is described as follows. See Table 21.10 “Functions Enabled by Writing to the S10 Register” for the bit function in write access.

Condition to become 0:

- The S00 register is written.
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

Condition to become 1:

- In master transmit mode or master receive mode, the SDAMM pin level changes to low by an external device, not by the ACK clock, when slave address is transmitted.
- The SDAMM pin level changes to low by an external device for other than the ACK clock when data is transmitted in master transmit mode.
- In master transmit mode or master receive mode, the SDAMM pin level changes to low by an external device when start condition is transmitted.
- In master transmit mode or master receive mode, the SDAMM pin level changes to low by an external device when stop condition is transmitted.
- The function to avoid start condition overlaps is started.

PIN (I2C-bus interface interrupt request bit) (b4)

The PIN bit function in read access is described as follows. See Table 21.10 “Functions Enabled by Writing to the S10 Register” for the bit function in write access.

Condition to become 0:

- Slave address transmission is completed in master mode (including when detecting arbitration lost).
- 1-byte data transmission is completed (including when detecting arbitration lost).
- 1-byte data reception is completed (the falling edge of eighth clock is detected when the ACKCLK bit in the S20 register is 0, or the falling edge of ACK clock when the ACKCLK bit is 1).
- The WIT bit in the S3D0 register is 1 (I2C-bus interrupt enabled at eighth clock) and 1-byte data is received (before the ACK clock).
- In slave receive mode, the MSLAD bit in the S4D0 register is 1, the ALS bit in the S1D0 register is 0 (addressing format), and any of the slave address stored in bits SAD6 to SAD0 in the S0Di register (i = 0 to 2) is matched with the received slave address (slave address match).
- In slave receive mode, the MSLAD bit is 0, the ALS bit is 0 (addressing format), and the slave address stored in bits SAD6 to SAD0 in the S0D0 register is matched with the received slave address (slave address match).
- In slave receive mode, the ALS bit in the S1D0 register is 0 (addressing format) and the general call address (0000000b) is received.
- In slave receive mode, the ALS bit in the S1D0 register is 1 (free format) and slave address reception is completed.

Condition to become 1:

- The S00 register is written.
- The S20 register is written (when the WIT bit is 1 and the internal WAIT flag is 1).
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

The IR bit in the IICIC register becomes 1 (interrupt requested) as soon as the PIN bit becomes 0 (I2C-bus interrupt requested). When the PIN bit is 0, the SCLMM pin output level is low.

However, the SCLMM pin output level does not become low when all the following conditions are met:

- In master mode, arbitration lost is detected by a slave address or data
- The ALS bit in the S1D0 register is 0 (addressing format)
- The slave address is not 0000000b (general call) and does not match any of the bits from SAD6 to SAD0 in registers S0D0 to S0D2.

BB (Bus busy flag) (b5)

The BB bit function in read access is described below. See Table 21.10 “Functions Enabled by Writing to the S10 Register” for the bit function in write access.

The BB bit indicates the state of the bus system, whether the bus is free or not. The BB bit changes depending on the SCLMM and SDAMM input signals, regardless of master mode or slave mode.

Condition to become 0:

- Stop condition is detected.
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

Condition to become 1:

- Start condition is detected.

TRX (Communication mode select bit 0) (b6)

The TRX selects transmit mode or receive mode.
Condition to become 0:

- The TRX bit is set to 0 by a program.
- Arbitration lost is detected.
- Stop condition is detected.
- Start condition overlap protect function is enabled.
- Start condition is detected when the MST bit in the S10 register is 0 (slave mode).
- No ACK is detected from a receiver when the MST bit in the S10 register is 0 (slave mode).
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

Condition to become 1:

- The TRX bit is set to 1 by a program.
- In slave mode, the ALS bit in the S1D0 register is 0 (addressing format), the AAS bit in the S10 register becomes 1 (address matched) after receiving the slave address, and the received R/W bit is 1.

MST (Communication mode select bit 1) (b7)

The MST bit selects master mode or slave mode.
Condition to become 0:

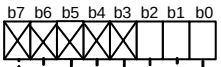
- The MST bit is set to 0 by a program.
- The 1-byte data that lost arbitration is completed transmitting/receiving when arbitration lost is detected.
- Stop condition is detected.
- Start condition overlap protect function is enabled.
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).

Condition to become 1:

- The MST bit is set to 1 by a program.

21.2.11 I2C0 Status Register 1 (S11)

I2C0 Status Register 1



Symbol	Address	After Reset
S11	02B9h	XXXX X000b

Bit Symbol	Bit Name	Function	RW
AAS0	Slave address 0 compare flag	0: No address matched 1: Address matched	RO
AAS1	Slave address 1 compare flag	0: No address matched 1: Address matched	RO
AAS2	Slave address 2 compare flag	0: No address matched 1: Address matched	RO
— (b7-b3)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

AAS0 (Slave address 0 compare flag) (b0)
AAS1 (Slave address 1 compare flag) (b1)
AAS2 (Slave address 2 compare flag) (b2)

The AASi bit indicates an address match when the ALS bit in the S1D0 register is set to 0 (addressing format) and any slave address stored to bits SAD6 to SAD0 in the S0Di register (i = 0 to 2) are compared with the received slave address. The AASi bit becomes 1 when there is an address match or when a general call address is received.
The AAS0 bit is enabled when the MSLAD bit in the S4D0 register is 0 (S0D0 register only). Bits AAS2 to AAS0 are enabled when the MSLAD bit is 1 (registers S0D0 to S0D2).

Bits AAS2 to AAS0 are set to 0 under the following conditions.

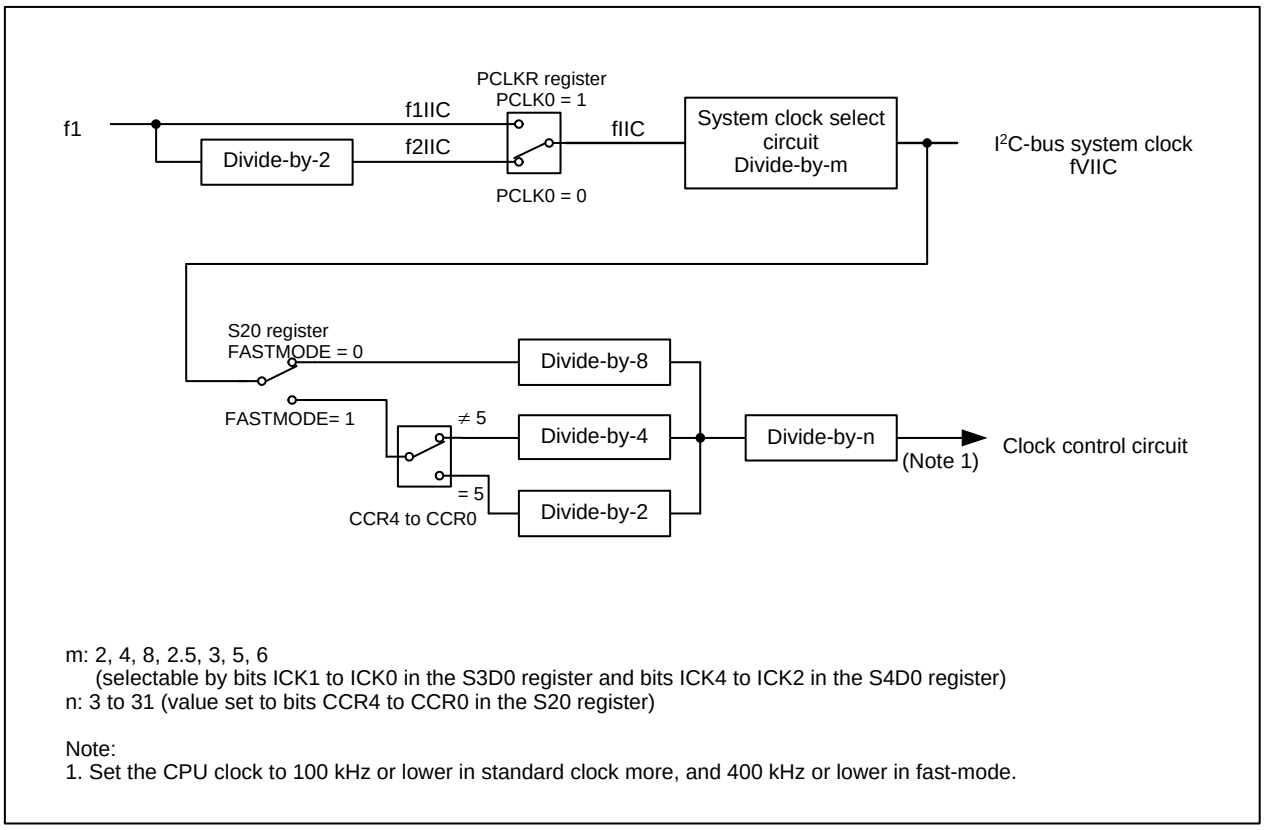
- The ES0 bit in the S1D0 register is set to 0 (I2C interface disabled).
- The IHR bit in the S1D0 register is set to 1 (I2C interface reset).
- The S00 register is written.

21.3 Operations

21.3.1 Clock

Set the PCKSTP16 bit in the PCLKSTP1 register to 0 (f1 provide enabled).

Figure 21.5 shows the I2C-bus Interface Clock.



m: 2, 4, 8, 2.5, 3, 5, 6
(selectable by bits ICK1 to ICK0 in the S3D0 register and bits ICK4 to ICK2 in the S4D0 register)
n: 3 to 31 (value set to bits CCR4 to CCR0 in the S20 register)

Note:
1. Set the CPU clock to 100 kHz or lower in standard clock more, and 400 kHz or lower in fast-mode.

Figure 21.5 I2C-bus Interface Clock

21.3.1.1 fVIIC

fVIIC is determined by setting a combination of the following:

- The frequency of peripheral clock f1
- The PCLK0 bit in the PCLKR register
- Bits ICK1 to ICK0 in the S3D0 register
- Bits ICK4 to ICK2 in the S4D0 register

fVIIC stops when the ES0 bit in the S1D0 register is 0 (I2C interface disabled).

See Table 21.8 “I2C-bus System Clock Select Bits” for details.

21.3.1.2 Bit Rate and Duty Cycle

The bit rate is determined by setting a combination of fVIIC and bits CCR4 to CCR0 in the S20 register.

Table 21.11 lists the Bit Rate of Internal SCL Output and Duty Cycle. Even if there is a change in duty cycle, the bit rate does not change. The bit rate and duty cycle described here are the ones before the I2C interface have any effect from the SCL output of external device.

Table 21.11 Bit Rate of Internal SCL Output and Duty Cycle

Item	Standard Clock Mode (FASTMODE = 0)	Fast-mode (FASTMODE = 1) (CCR value = other than 5)	Fast-mode (FASTMODE = 1) (CCR value = 5)
Bit rate (bps)	$\frac{fVIIC}{8 \times CCR\ value}$	$\frac{fVIIC}{4 \times CCR\ value}$	$\frac{fVIIC}{2 \times CCR\ value} = \frac{fVIIC}{10}$
Duty cycle	50% (1)	50% (2)	35 to 45%

CCR value: Value set to bits CCR4 to CCR0

Notes:
1. Fluctuation of high level: -4 to +2 fVIIC cycles
2. Fluctuation of high level: -2 to +2 fVIIC cycles

When the setting value (CCR value) of bits CCR4 to CCR0 is 5 (00101b) in fast-mode, the maximum bit rate should be 400 kbps in fast-mode.

The bit rate and duty cycle are as follows.

- Bit rate:

$$\frac{fVIIC}{2 \times CCR\ value} = \frac{fVIIC}{10}$$

When fVIIC is 4 MHz, the bit rate is 400 kbps.

- Duty cycle is 35 to 45%

Even if the bit rate is 400 kbps, the minimum low period of SCLMM clock of 1.3 μs (I2C-bus standard) is ensured.

Table 21.12 lists the Bit Setting of Bits CCR4 to CCR0 and Bit Rate (fVIIC = 4 MHz).

Table 21.12 Bit Setting of Bits CCR4 to CCR0 and Bit Rate (fVIIC = 4 MHz)

Bits CCR4 to CCR0 in the S20 Register					Bit Rate (kbps)	
CCR4	CCR3	CCR2	CCR1	CCR0	Standard Clock Mode	Fast-mode
0	0	0	0	0	Do not set (1)	Do not set (1)
0	0	0	0	1	Do not set (1)	Do not set (1)
0	0	0	1	0	Do not set (1)	Do not set (1)
0	0	0	1	1	Do not set (2)	333
0	0	1	0	0	Do not set (2)	250
0	0	1	0	1	100	400
0	0	1	1	0	83.3	166
:	:	:	:	:	:	:
1	1	1	0	1	17.2	34.5
1	1	1	1	0	16.6	33.3
1	1	1	1	1	16.1	32.3

Notes:
1. Bits CCR4 to CCR0 should not be set to 0 to 2 regardless of the fVIIC frequency.
2. Do not exceed the maximum bit rates of 100 kbps in standard clock mode and 400 kbps in fast-mode.

21.3.1.3 Receiving a Slave Address in Wait Mode and Stop Mode

When the CM02 bit in the CM0 register is set to 0 (peripheral clock f1 does not stop in wait mode) and the MCU has entered wait mode, the I2C interface receives the slave address even in wait mode.
When the CM02 bit in the CM0 register is set to 1 (peripheral clock f1 stops in wait mode), and the CPU enters wait mode, stop mode, or low-power consumption mode, the I2C interface stops operating because fVIIC also stops.
The SCL/SDA interrupt can be used in either wait mode or stop mode.

21.3.2 Generating a Start Condition

Follow the procedure below when the ES0 bit in the S1D0 register is 1 (I2C interface enabled) and the BB bit in the S10 register is set to 0 (bus free). Figure 21.6 shows the Procedure to Generate a Start Condition.

(1) Write E0h to the S10 register.
The I2C interface enters the start condition standby state and the SDAMM pin is left open.

(2) Write a slave address to the S00 register.
A start condition is generated. Then, the bit counter becomes 000b, the SCL clock signal is output for 1 byte, and the slave address is transmitted.

After a stop condition is generated and the BB bit becomes 0 (bus free), the S10 register is write disabled for 1.5 cycles of fVIIC. Therefore, when writing E0h to the S10 register and a slave address to the S00 register during the 1.5 fVIIC cycles, start condition standby state is not entered, and a start condition is not generated accordingly. When generating a start condition immediately after the BB bit changes from 1 to 0, confirm that both the TRX and MST bits are 1 after step (1), and then execute step (2).

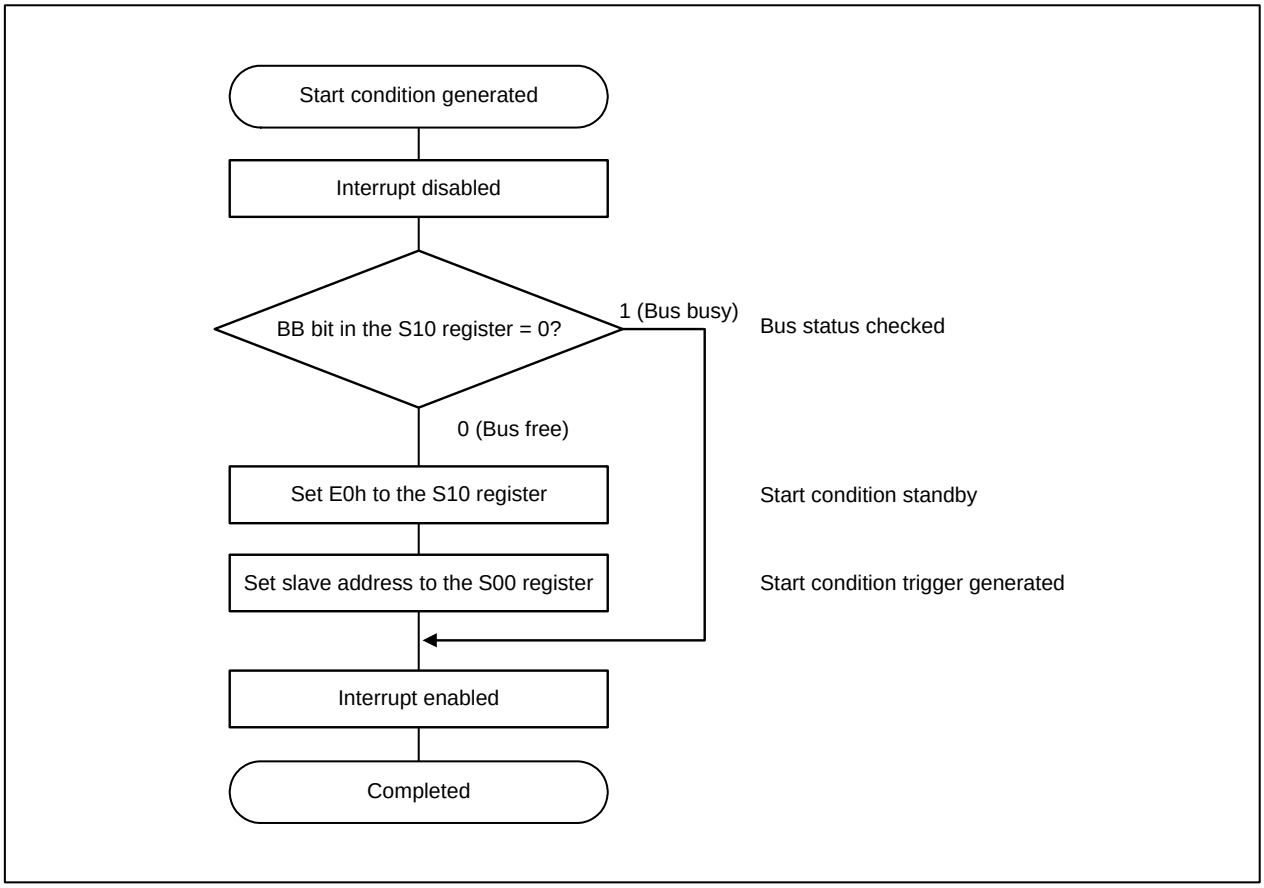


Figure 21.6 Procedure to Generate a Start Condition

The start condition generation timing depends on the modes - standard clock mode or fast-mode. Figure 21.7 shows the Start Condition Generation Timing.
Table 21.13 lists the Setup/Hold Time for Generating a Start/Stop Condition.

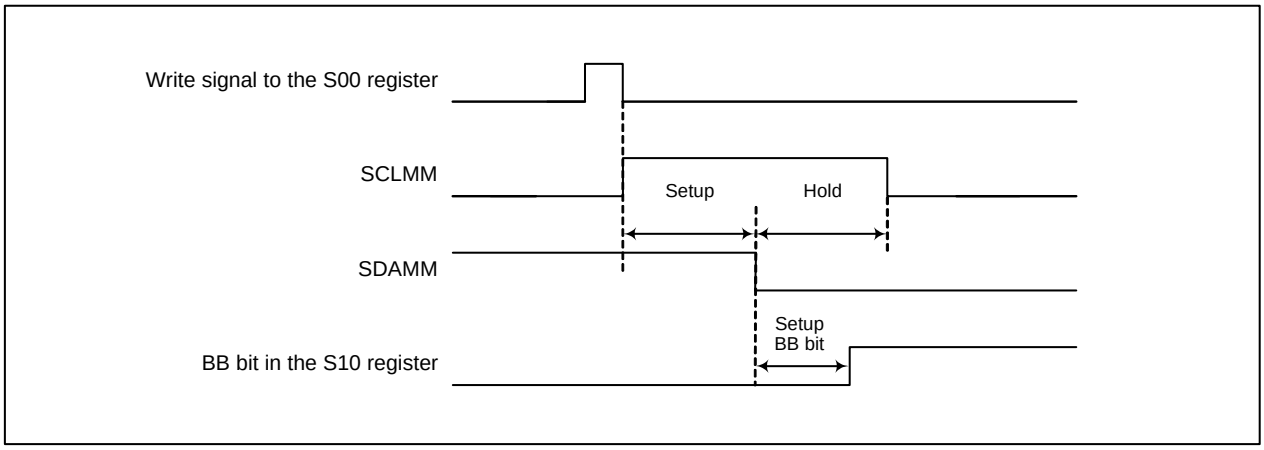


Figure 21.7 Start Condition Generation Timing

Table 21.13 Setup/Hold Time for Generating a Start/Stop Condition

Item	STSPSEL Bit	Standard Clock Mode		Fast-mode	
		fVIIC cycles	fVIIC = 4 MHz	fVIIC cycles	fVIIC = 4 MHz
Setup time	0 (short mode)	20	5.0 µs	10	2.5 µs
	1 (long mode)	52	13.0 µs	26	6.5 µs
Hold time	0 (short mode)	20	5.0 µs	10	2.5 µs
	1 (long mode)	52	13.0 µs	26	6.5 µs
BB bit set/reset time	—	$\frac{\text{SSC value} - 1}{2} + 2$	3.375 µs (1)	3.5	0.875 µs

—: 0 or 1
STSPSEL: Bit in the S2D0 register
SSC value: Value of bits SSC4 to SSC0 in the S2D0 register
Note:
1. Example value when bits SSC4 to SSC0 are 11000b.

21.3.3 Generating a Stop Condition

Use the following procedure when the ES0 bit in the S1D0 register is 1 (I2C interface enabled).

(1) Write C0h to the S10 register.
The I2C interface enters the stop condition standby state and the SDAMM pin is driven low.

(2) Write dummy data to the S00 register.
A stop condition is generated.

The stop condition generation timing depends on the modes - standard clock mode or fast-mode. Figure 21.8 shows the Stop Condition Generation Timing. See Table 21.13 “Setup/Hold Time for Generating a Start/Stop Condition” for setup/hold time.

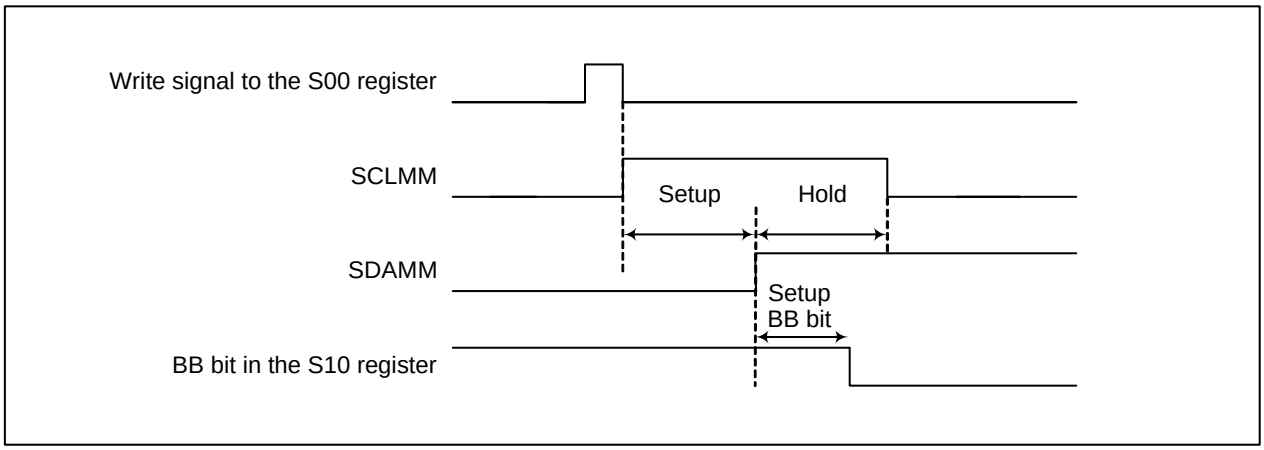


Figure 21.8 Stop Condition Generation Timing

The S10 register or S00 register should not be written until the BB bit in the S10 register becomes 0 (bus free) after the instructions to generate a stop condition (refer to above (2)) are executed.
If the SCLMM pin input signal becomes low until the BB bit in the S10 register becomes 0 (bus free) from the instruction to generate a stop condition is executed and the SCLMM pin becomes high-level, the internal SCL output becomes low. In this case, perform one of the steps below to stop the low signal output from the SCLMM pin (leave the SCLMM pin open).

- Generate a stop condition (perform steps (1) and (2) above).
- Set the ES0 bit in the S1D0 register to 0 (I2C interface disabled).
- Write a 1 to the IHR bit (I2C interface reset).

21.3.4 Generating a Restart Condition

Use the following procedure to generate a restart condition when 1-byte data is transmitted/received.

(1) Write E0h to the S10 register. (Start condition standby state. The SDAMM pin becomes high-impedance.)
(2) Wait until the SDAMM pin level becomes high.
(3) Write a slave address to the S00 register (a start condition trigger is generated)

Figure 21.9 shows the Restart Condition Generation Timing.

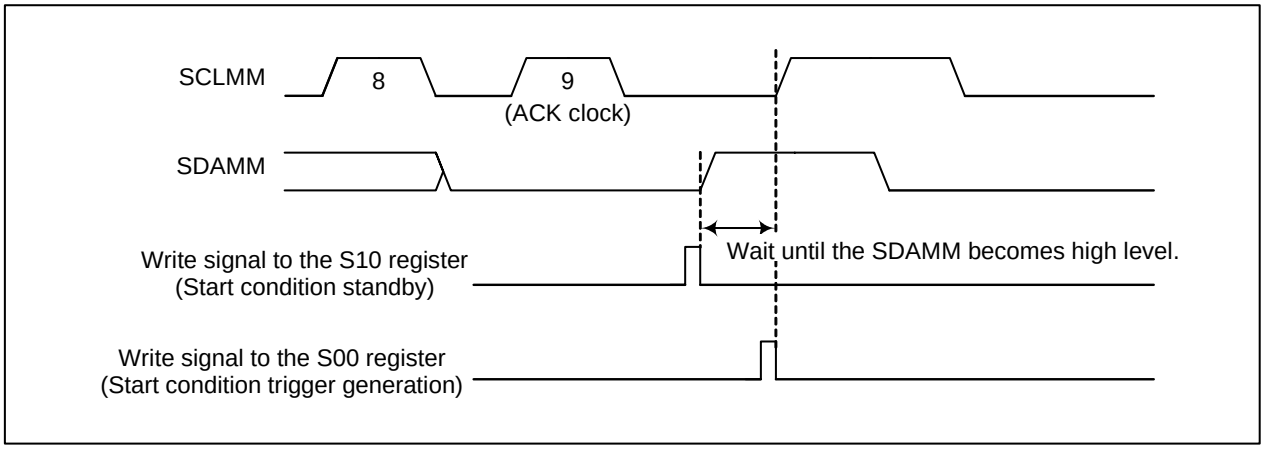


Figure 21.9 Restart Condition Generation Timing

21.3.5 Start Condition Overlap Protect

The I2C interface generates a start condition by setting registers S10 and S00 by a program. The bus system must be free before setting these registers. Check whether the bus is free with the BB bit in the S10 register by a program before setting the registers.

However, even after confirming that the bus is free, other master devices may generate a start condition before setting registers S10 and S00. In this case, when the I2C interface detects a start condition, the BB bit becomes 1 (bus busy) and the start condition overlap protect function is activated. The start condition overlap protect function operates as follows:

- The start condition standby state is not entered even if the S10 register is set to E0h.
- If the I2C interface is in a start condition standby state, exit the state.
- A start condition trigger is not generated even if a data is written to the S00 register by program.
- Bits MST and TRX in the S10 register become 0 (slave receive mode).
- The AL bit in the S10 register becomes 1 (arbitration lost detected).

Figure 21.10 shows the Start Condition Overlap Protect Operation.

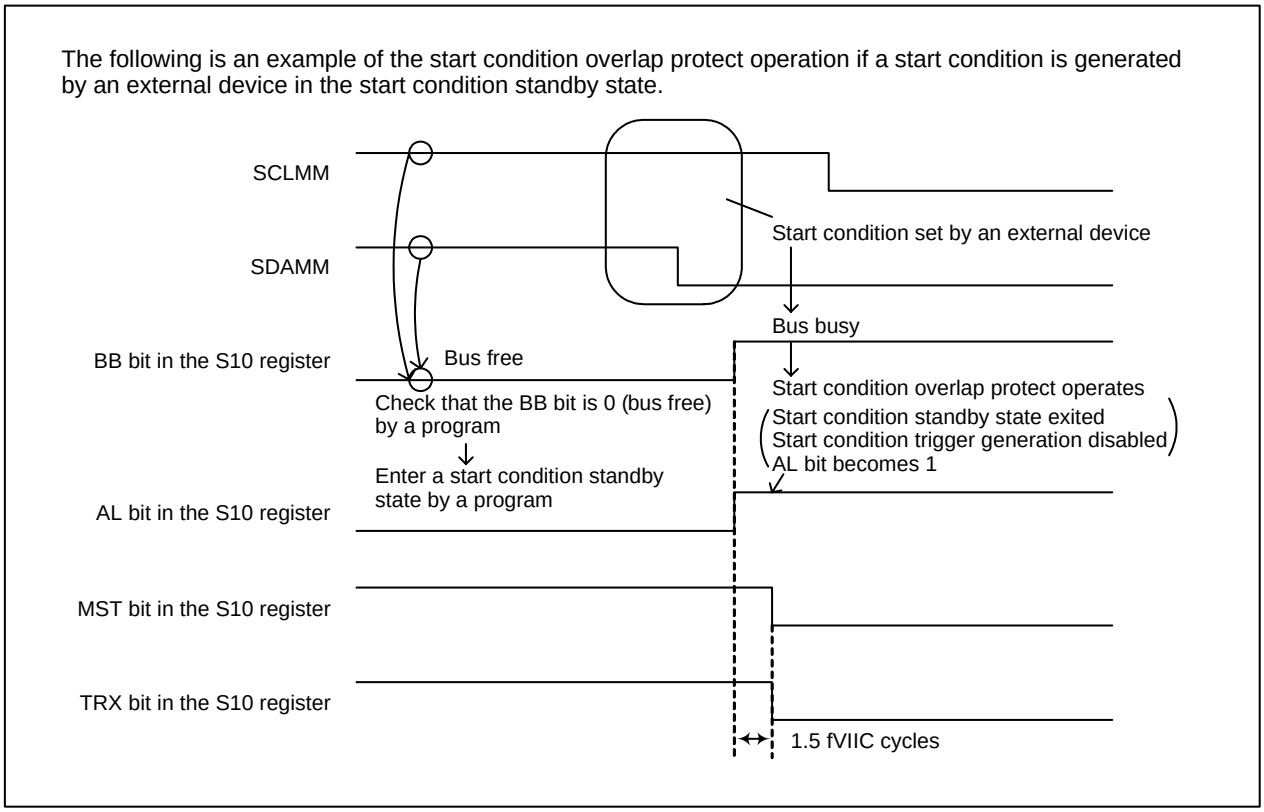


Figure 21.10 Start Condition Overlap Protect Operation

The start condition overlap protect is enabled from the falling edge of SDAMM (start condition) to the completion of the slave address receive. If data is written to registers S10 and S00 during that period, the above operation is performed. Figure 21.11 shows the Start Condition Overlap Protect Function Enable Period.

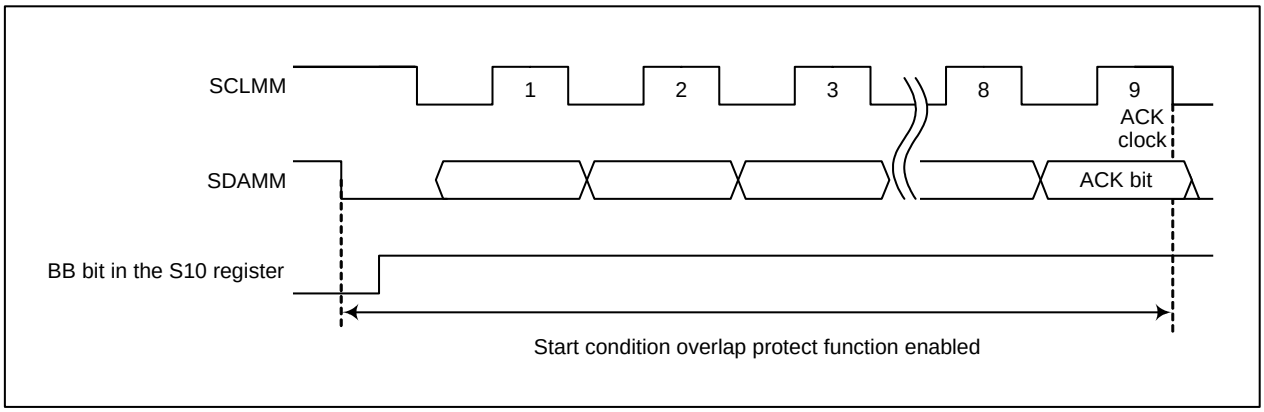


Figure 21.11 Start Condition Overlap Protect Function Enable Period

21.3.6 Arbitration Lost

When all of the conditions below are met, the signal level of SDAMM pin becomes low by an external device and the I2C interface determines that it has lost arbitration.

(a) Transmit/receive (one of the following)
- Slave address transmit (not an ACK clock) in master transmit mode or master receive mode
- Data transmit (not an ACK clock) in master transmit mode
- Start condition generated in master transmit mode or master receive mode
- Stop condition generated in master transmit mode or master receive mode

(b) Internal SDA output: High

(c) SDAMM pin level: Low (sampling at the rising edge of the clock of SCLMM pin.)

Figure 21.12 shows Operation Example When Arbitration Lost is Detected.

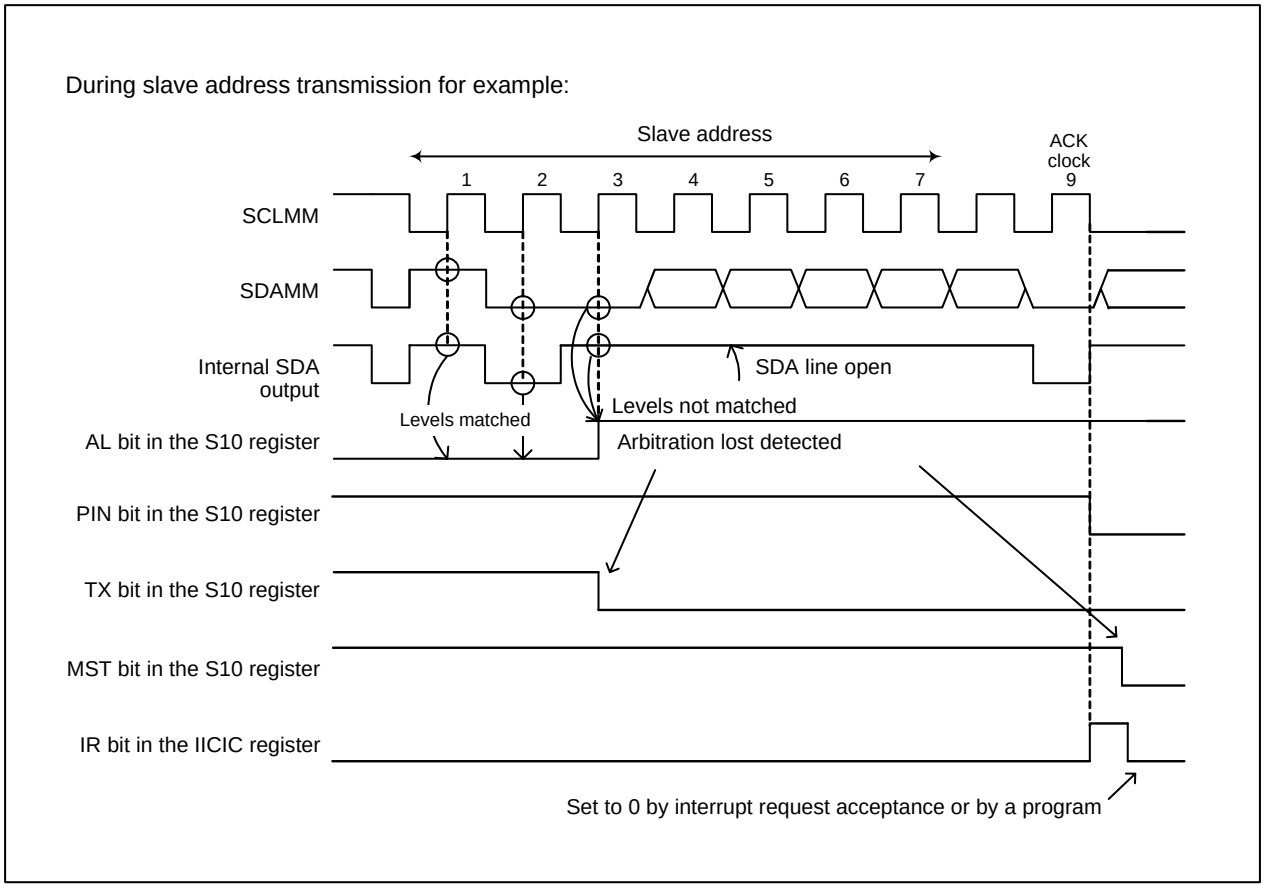


Figure 21.12 Operation Example When Arbitration Lost is Detected

When arbitration lost is detected:

- The AL bit in the S10 register becomes 1 (arbitration lost detected)
- Internal SDA output becomes high. (SDAMM becomes high-impedance)
- Slave receive mode is entered by setting both the TRX and MST bits in the S10 registers to 0 (receive mode and slave mode, respectively).

In order to set the AL bit to 0 again after arbitration lost is detected, set a value to the S00 register.

When arbitration lost is detected during slave address transmission, the I2C interface automatically enters the slave receive mode and receives the slave address sent from another master. When the ALS bit in the S1D0 register is 0 (addressing format), the slave address comparison result is determined by reading bits ADR0 and AAS in the S10 register.

When arbitration lost is detected during data transmission, slave receive mode is automatically entered. Also, when arbitration lost is detected, the TRX bit becomes 0 (receive mode) even when the bit after the slave address is 1 (read). Therefore, after arbitration lost is detected, read the S00 register. When bit 0 in the S00 register is 1, write 4Fh (slave transmit mode) to the S10 register and execute slave transmission.

21.3.7 Detecting Start/Stop Conditions

Figure 21.13 shows Start Condition Detection, Figure 21.14 shows Stop Condition Detection, and Table 21.14 lists Conditions to Detect Start/Stop Condition.

A start/stop condition is detected only when the start/stop condition detect conditions (SCL open time, setup time, and hold time) are selected by bits SSC4 to SSC0 in the S2D0 register, and the signals input to pins SCLMM and SDAMM meet all three conditions (SCLMM open time, setup time, and hold time) listed in Table 21.14.

The BB bit in the S10 register becomes 1 when a start condition is detected and 0 when a stop condition is detected. The set timing and reset timing of the BB bit depends on the mode, standard mode or fast-mode. Refer to the BB bit set/reset times in Table 21.15.

Table 21.15 lists the Recommended Value of Bits SSC4 to SSC0 in Standard Clock Mode.

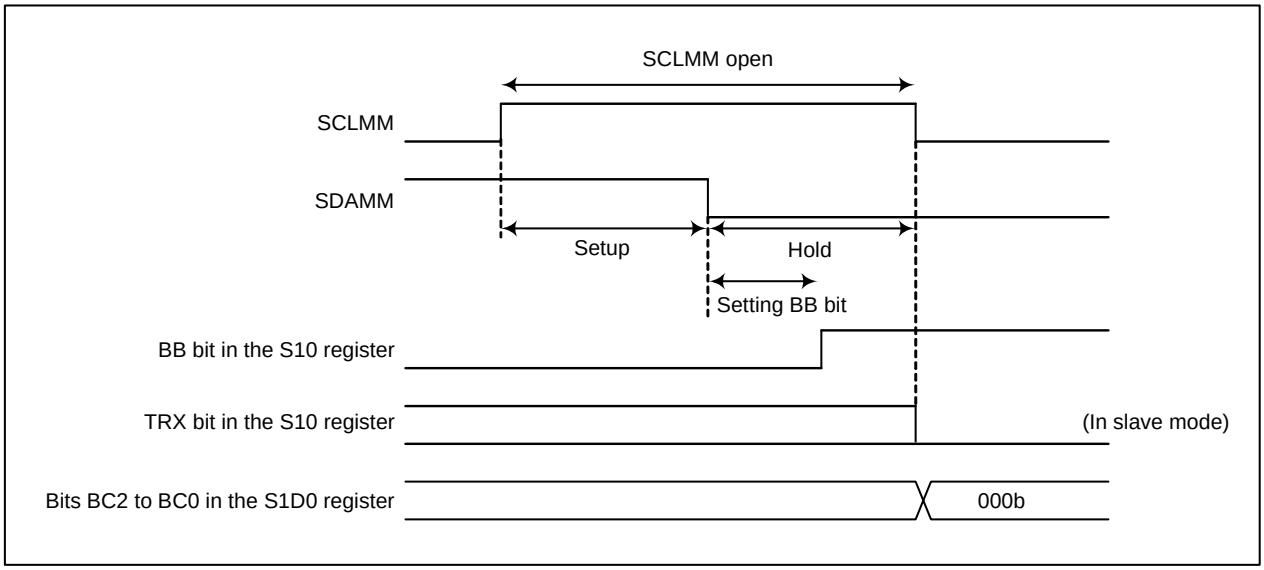


Figure 21.13 Start Condition Detection

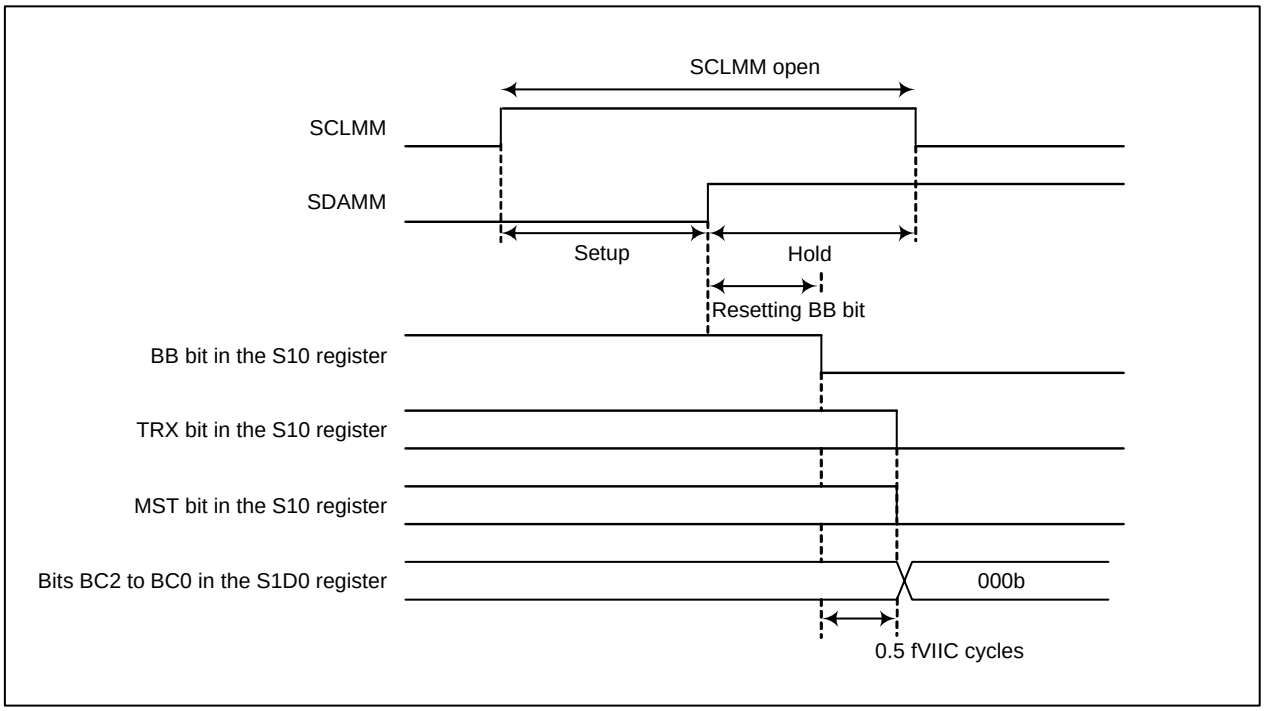


Figure 21.14 Stop Condition Detection

Table 21.14 Conditions to Detect Start/Stop Condition

	Standard Clock Mode	Fast-mode
SCLMM open time	SSC value + 1 cycle	4 cycles
Setup time	$\frac{\text{SSC value}}{2}$ + 1 cycle	2 cycles
Hold time	$\frac{\text{SSC value}}{2}$ cycles	2 cycles
BB bit setting/resetting time	$\frac{\text{SSC value} - 1}{2}$ + 2 cycles	3.5 cycles

Unit: Number of fVIIC cycles
SSC value: Value of bits SSC4 to SSC0 in the S2D0 register

Table 21.15 Recommended Value of Bits SSC4 to SSC0 in Standard Clock Mode

fVIIC	SSC Value (recommended)	Start/Stop Condition			BB Bit Setting/Resetting Time
		SCLMM open time	Setup time	Hold time	
5 MHz	11110b	6.2 μs (31)	3.2 μs (16)	3.0 μs (15)	3.3 μs (16.5)
4 MHz	11010b	6.75 μs (27)	3.5 μs (14)	3.25 μs (13)	3.625 μs (14.5)
	11000b	6.25 μs (25)	3.25 μs (13)	3.0 μs (12)	3.375 μs (13.5)
2 MHz	01100b	6.5 μs (13)	3.5 μs (7)	3.0 μs (6)	3.75 μs (7.5)
	01010b	5.5 μs (11)	3.0 μs (6)	2.5 μs (5)	3.25 μs (6.5)
1 MHz	00100b	5.0 μs (5)	3.0 μs (3)	2.0 μs (2)	3.5 μs (3.5)

SSC value: Value of bits SSC4 to SSC0 in the S2D0 register
( ): Number of fVIIC cycles

21.3.8 Operation after Transmitting/Receiving a Slave Address or Data

After a slave address or 1-byte data has been transmitted/received, the PIN bit in the S10 register becomes 0 (interrupt requested) at the falling edge of the ACK clock. The IR bit in the IICIC register becomes 1 (interrupt requested) at the same time. The value in the S10 register and so on changes depending on the state of the transmit/receive data, and the level of pins SCLMM and SDAMM. Figure 21.15 shows Operation After Transmitting/Receiving a Slave Address or Data.

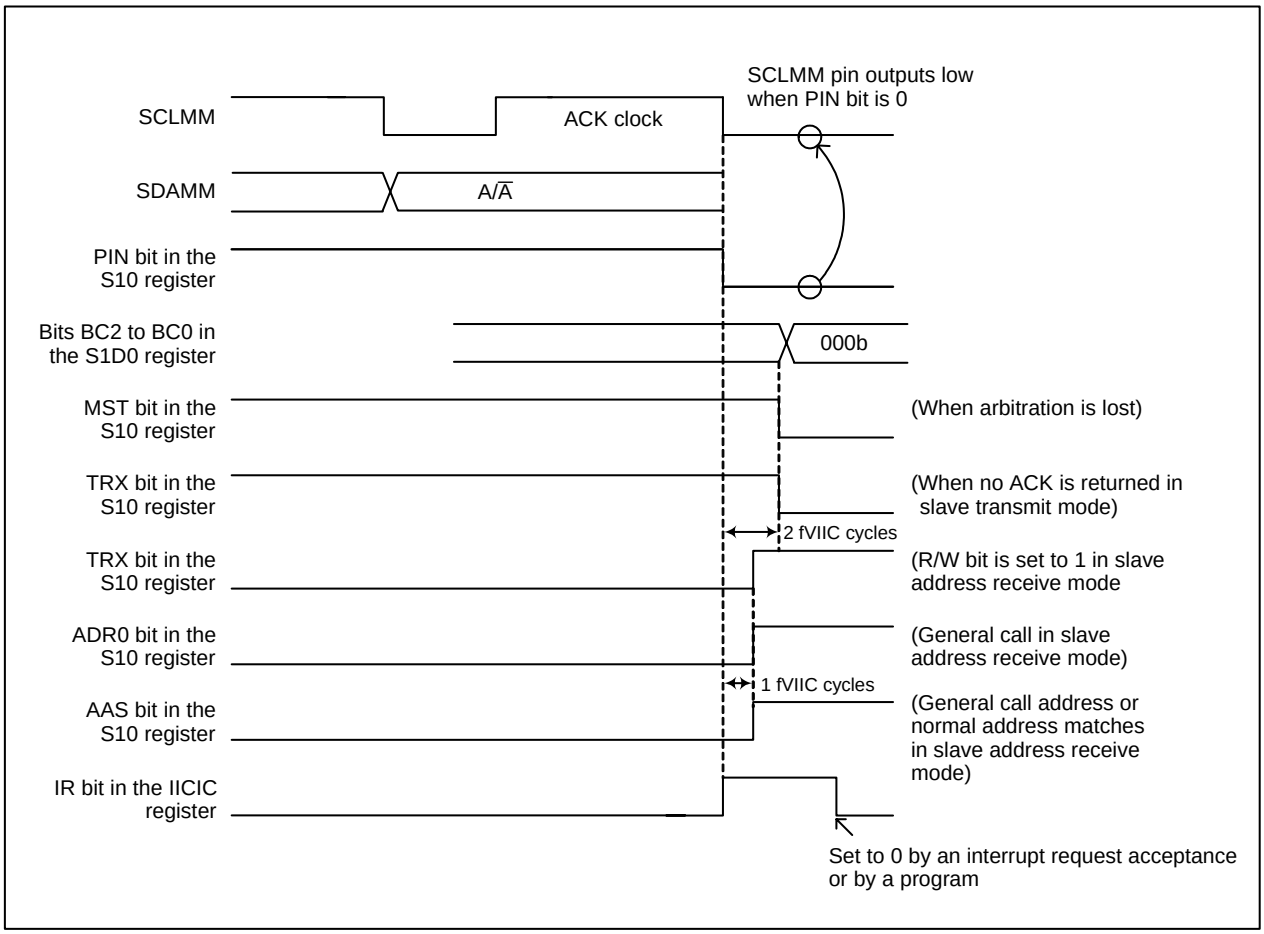


Figure 21.15 Operation After Transmitting/Receiving a Slave Address or Data

21.3.9 Timeout Detection

If the SCL clock is stopped during transmission/reception, each device stops operating, keeping the communication state. The timeout detection function detects timeouts and generates an I2C-bus interrupt request when the SCLMM pin is driven high for more than the selected timeout detection period during transmission/reception. Figure 21.16 shows the Timeout Detection Timing. Refer to "TOSEL (Timeout Detection Period Select Bit) (b2)" in 21.2.9 "I2C0 Control Register 2 (S4D0)" for the timeout detection period.

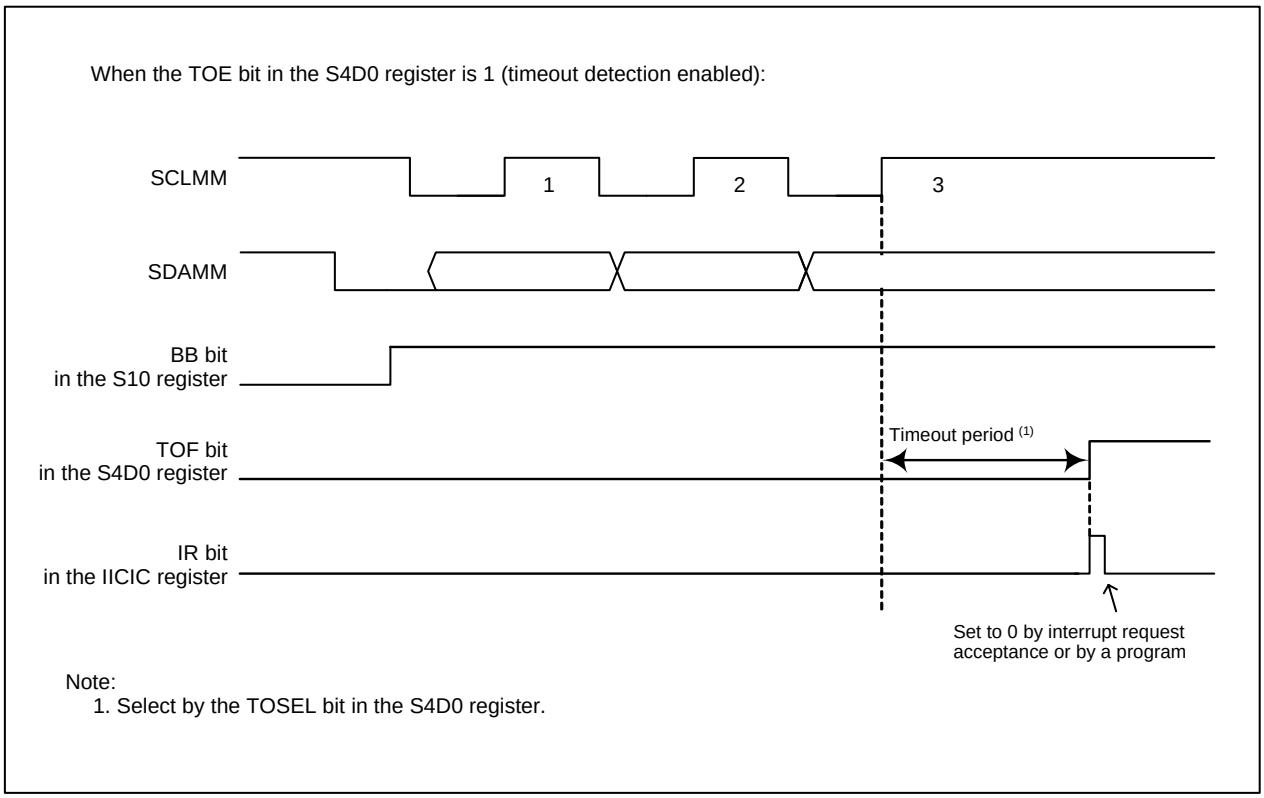


Figure 21.16 Timeout Detection Timing

A timeout is detected when the following conditions are all met:

- The TOE bit in the S4D0 register is set to 1 (timeout detection enabled)
- The BB bit in the S10 register is set to 1 (bus busy)
- The SCLMM pin is driven high for more than the timeout detect period

When a timeout is detected:

- The TOF bit in the S4D0 register becomes 1 (timeout detected)
- The IR bit in the IICIC register becomes 1 (I2C-bus interrupt requested)

When the timeout is detected, perform one of the following:

- Set the ES0 bit in the S1D0 register to 0 (disabled).
- Set the IHR bit in the S1D0 register to 1 (I2C interface reset).

21.3.10 Data Transmit/Receive Examples

The data transmit/receive examples are described in this section. The conditions for the examples are follows.

- Slave address: 7 bits
- Data: 8 bits
- ACK clock
- Standard clock mode, bit rate: 100 kbps (fIIC: 20 MHz; fVIIC: 4 MHz)
  20 MHz (fIIC) divided-by-5 = 4 MHz (fVIIC),
  4 MHz (fVIIC) divided-by-8 and further divided-by-5 = 100 kbps (bit rate)
- In receive mode, an ACK response is sent for data other than the last data. NACK is returned after the last data is received.
- When receiving data, I2C-bus interrupt at the eighth clock (just before ACK clock): disabled
- Stop condition interrupt: enabled
- Timeout detect interrupt: disabled
- Set an own slave address to the S0D0 register (registers S0D1 or S0D2 should not be used)

If an I2C-bus interrupt at 8th clock (just before ACK clock) is enabled in data receive, a receiver generates ACK or NACK after each byte of data has been received.

21.3.10.1 Initial Settings

Follow the initial setting procedures below for 21.3.10.2 to 21.3.10.5.

(1) Write an own slave address to bits SAD6 to SAD0 in the S0D0 register.
(2) Write 85h to the S20 register. (CCR value: 5, standard mode selected, ACK clock presents)
(3) Write 18h to the S4D0 register. (fVIIC: fIIC divided-by-5, timeout interrupt disabled)
(4) Write 01h to the S3D0 register. (stop condition detect interrupt enabled and I2C-bus interrupt at eighth clock is disabled when receiving data)
(5) Write 0Fh to the S10 register. (slave receive mode)
(6) Write 98h to the S2D0 register (SSC value: 18h; start/stop condition generation timing: long mode)
(7) Write 08h to the S1D0 register (bit counter: 8, I2C interface enabled, addressing format, input level: I2C-bus input)

If the MCU uses a single-master system and it is a master, start the initial setting procedures from step (1).

21.3.10.2 Master Transmission

Master transmission is described in this section. The initial settings described in 21.3.10.1 “Initial Settings” are assumed to be completed. Figure 21.17 shows the operation of master transmission. The following programs (A) to (C) are executed at the (A) to (C) in Figure 21.17, respectively.

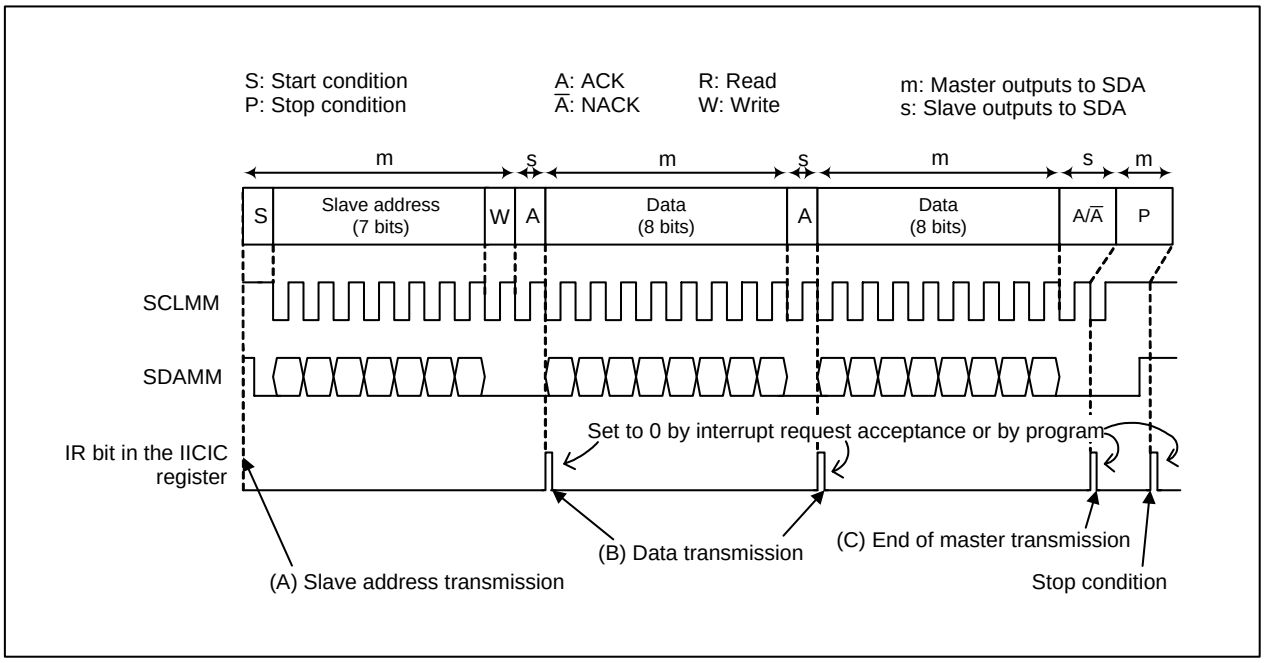


Figure 21.17 Example of Master Transmission

(A) Slave address transmission
- (1) The BB bit in the S10 register must be 0 (bus free).
- (2) Write E0h to the S10 register (start condition standby).
- (3) Write a slave address to the seven most significant bits (MSB) and a 0 to the least significant bit (LSB) (start condition generated, then slave address transmitted).

(B) Data transmission
(in I2C-bus interrupt routine)
- (1) Write transmit data to the S00 register (data transmission).

(C) Completion of Master transmission
(in I2C-bus interrupt routine)
- (1) Write C0h to the S10 register (stop condition standby state).
- (2) Write dummy data to the S00 register (stop condition generated).

When the transmission is completed or ACK is not returned from slave device (NACK returned), master transmission should be completed as shown in the example above.

21.3.10.3 Master Reception

The master reception is described in this section. The initial settings described in 21.3.10.1 “Initial Settings” are assumed to be completed. Figure 21.18 shows the operation example of master reception. The following programs (A) to (D) are executed at the (A) to (D) in Figure 21.18, respectively.

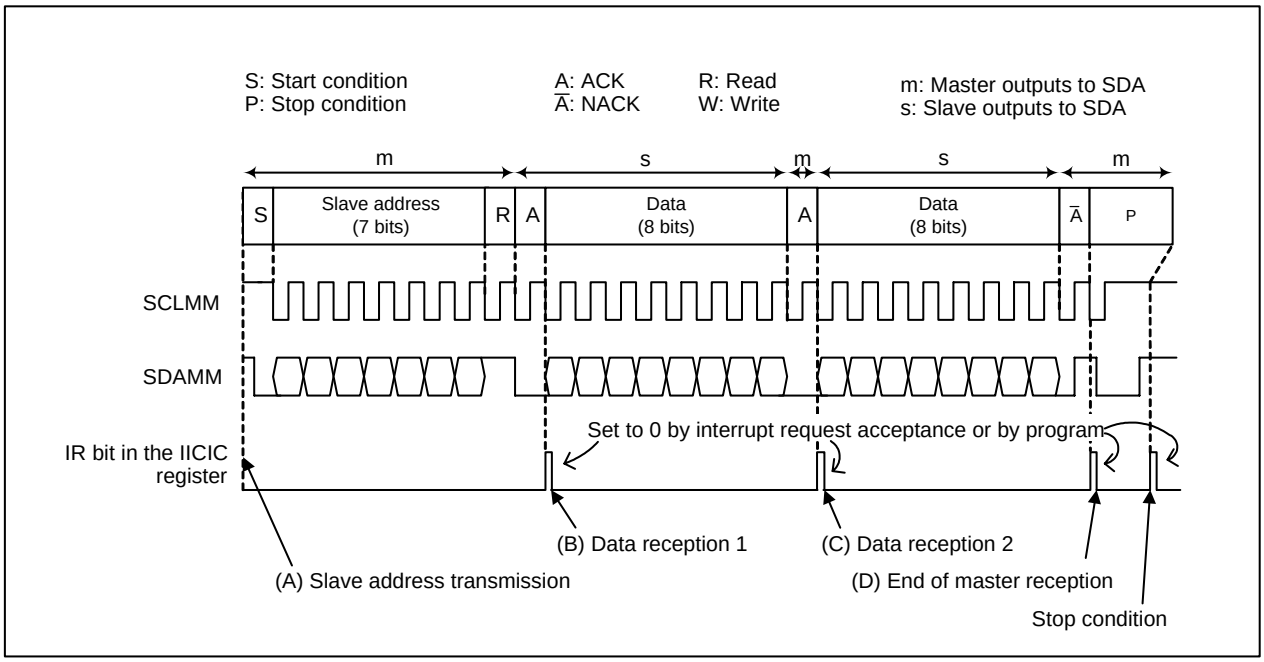


Figure 21.18 Example of Master Reception

(A) Slave address transmission
- (1) The BB bit in the S10 register must be 0 (bus free).
- (2) Write E0h to the S10 register (start condition standby).
- (3) Write a slave address to the seven most significant bits (MSB) and a 1 to the least significant bit (LSB) (start condition generated, then slave address transmitted).

(B) Data reception 1 (after slave address transmission)
(In I2C-bus interrupt routine)
- (1) Write AFh to the S10 register (master receive mode).
- (2) Set the ACKBIT bit in the S20 register to 0 (ACK presents) because the data is not the last one.
- (3) Write dummy data to the S00 register.

(C) Data reception 2 (data reception)
(In I2C-bus interrupt routine)
- (1) Read the received data from the S00 register.
- (2) Set the ACKBIT bit in the S20 register to 1 (no ACK) because the data is the last one.
- (3) Write dummy data to the S00 register.

(D) End of master reception
(In I2C-bus interrupt routine)
- (1) Read the received data from the S00 register.
- (2) Write C0h to the S10 register (stop condition standby state).
- (3) Write dummy data to the S00 register (stop condition generated).

21.3.10.4 Slave Reception

The slave reception is described in this section. The initial settings described in 221.3.10.1 “Initial Settings” are assumed to be completed. Figure 21.19 shows the example of slave reception. The following programs (A) to (C) are executed at the (A) to (C) in Figure 21.19, respectively.

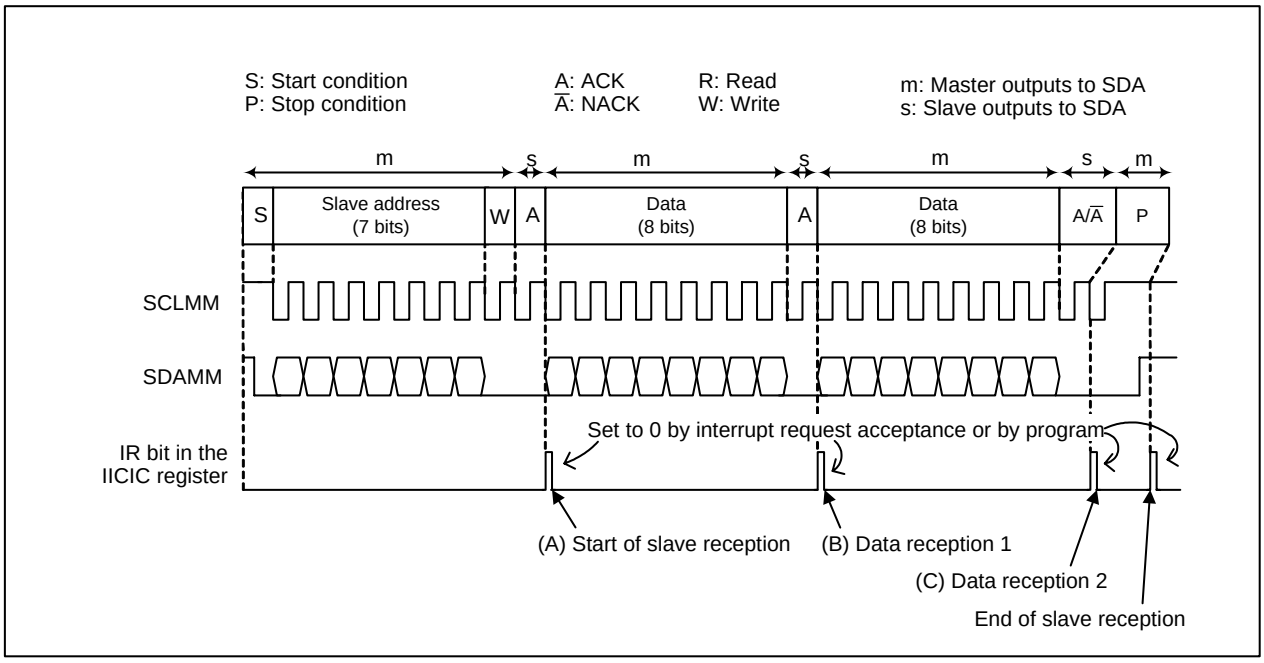


Figure 21.19 Example of Slave Reception

(A) Slave receive is started.

(In I2C-bus interrupt routine)

(1) Check the content of S10 register. When the TRX bit is 0, the I2C interface is in slave receive mode.

(2) Write dummy data to the S00 register.

(B) Data reception 1

(In I2C-bus interrupt routine)

(1) Read the received data from the S00 register.

(2) Set the ACKBIT bit in the S20 register to 0 (ACK presents) because the data is not the last one.

(3) Write dummy data to the S00 register.

(C) Data reception 2

(In I2C-bus interrupt routine)

(1) Read the received data from the S00 register

(2) Set the ACKBIT bit in the S20 register to 1 (no ACK presents) because the data is the last one.

(3) Write dummy data to the S00 register.

21.3.10.5 Slave Transmission

The slave transmission is described in this section. The initial settings described in 21.3.10.1 “Initial Settings” are assumed to be completed. Figure 21.20 shows the example of slave transmission. The following programs (A) to (B) are executed at the (A) and (B) in Figure 21.20, respectively.
When arbitration lost is detected, the TRX bit becomes 0 (receive mode) even when the bit after the slave address is 1 (read). Therefore, after arbitration lost is detected, read the S00 register. When the bit 0 in the S00 register is 1, write 4Fh (slave transmit mode) to the S10 register and execute slave transmission.

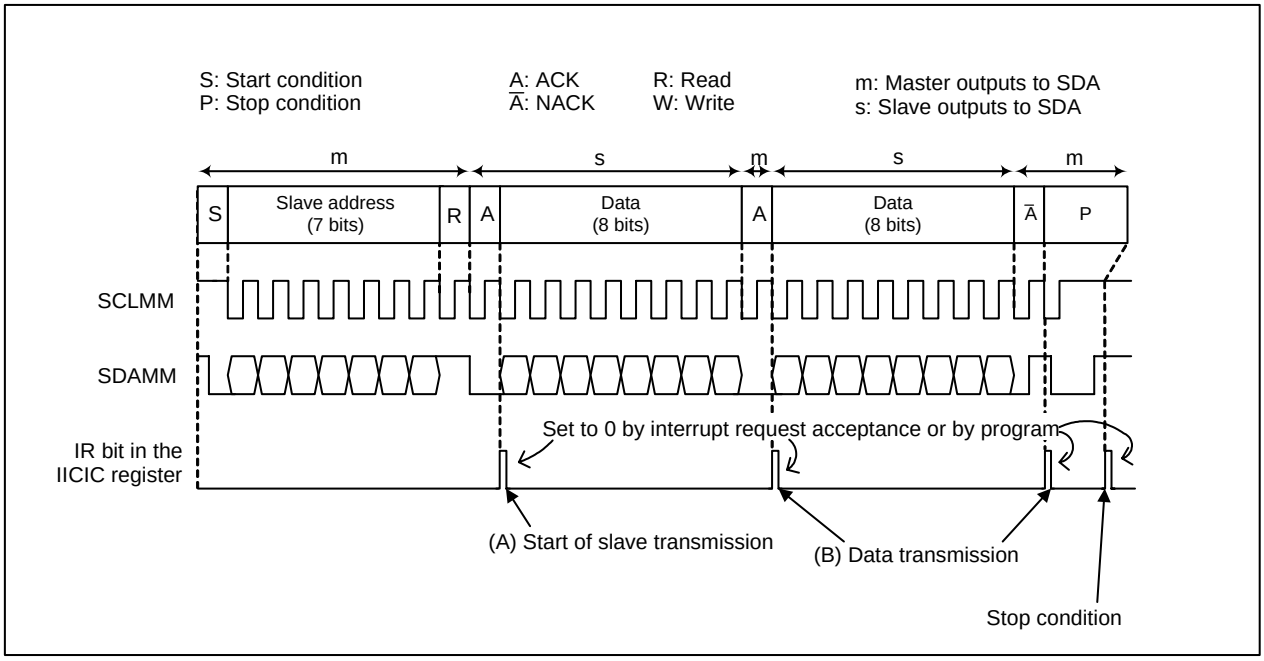


Figure 21.20 Example of Slave Transmission

(A) Start of slave transmission
(In I2C-bus interrupt routine)
(1) Check the content of the S10 register. When the TRX bit is set to 1, the I2C interface is in slave transmit mode.
(2) Write a transmit data to the S00 register.

(B) Data transmission
(In I2C-bus interrupt routine)
(1) Write a transmit data to the S00 register.

Write dummy data to the S00 register even if an interrupt occurs at an ACK clock of the last transmit data. When the S00 register is written, the SCLMM pin becomes high-impedance.

21.4 Interrupts

The I2C interface generates interrupt requests. Figure 21.21 shows I2C Interface Interrupts, and Table 21.16 lists I2C-bus Interrupts.

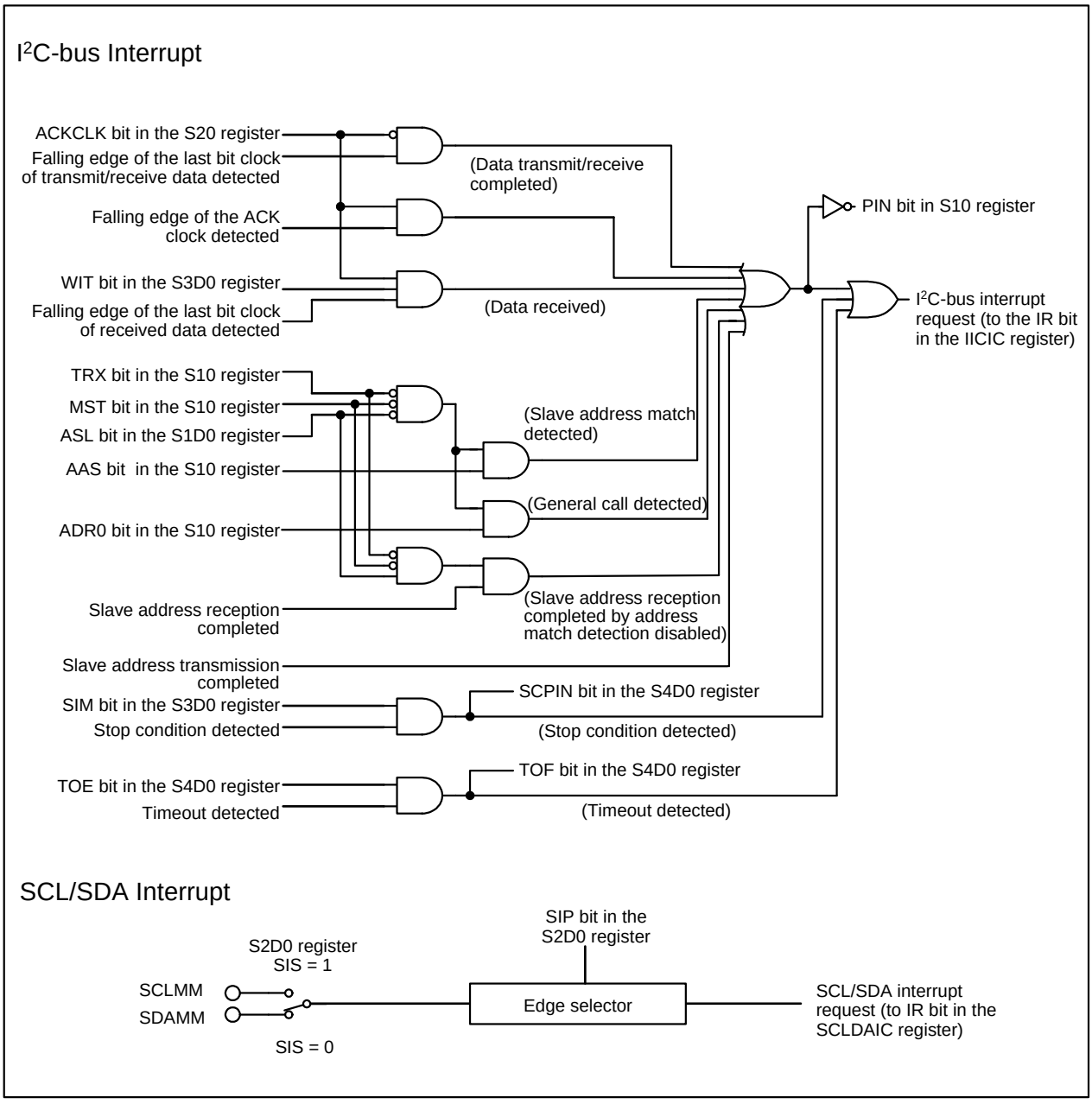


Figure 21.21 I2C Interface Interrupts

Table 21.16 I2C-bus Interrupts

Interrupt	Interrupt Source	Associated Bits (Register)		Interrupt Control Register
		Interrupt enabled	Interrupt request	
I2C-bus Interrupt	Completion of data transmit/receive When the ACKCLK bit in the S20 register is 0: Detection of the falling edge of the last clock of transmit/receive data through SCLMM pin When the ACKCLK bit is 1: Detection of the falling edge of ACK clock through SCLMM pin	—	PIN (S10)	IICIC
	Data reception (before ACK clock) Detection of the falling edge of the last clock of transmit/receive data through SCLMM pin	WIT (S3D0)		
	Detection of slave address match Received slave address matches bits SAD6 to SAD0 in registers S0D0 to S0D2 in slave receive mode with addressing format (AAS bit in the S10 register = 1)	—		
	Detection of general call General call in slave receive mode with addressing format (ADR0 bit in the S10 register = 1)			
	Completion of receiving slave address in slave receive mode with free format			
	Stop condition detected	SIM (S3D0)	SCPIN (S4D0)	
	Timeout detected	TOE (S4D0)	TOF (S4D0)	
SCL/SDA interrupt	Detection of the falling edge or rising edge of I/O signal for the SCLMM or SDAMM pin	—	—	SCLDAIC

Refer to 13.7 “Interrupt Control”. Table 21.17 lists Registers Associated with I2C Interface Interrupts.

Table 21.17 Registers Associated with I2C Interface Interrupts

Address	Register	Symbol	Reset Value
007Bh	I2C-bus Interface Interrupt Control Register	IICIC	XXXX X000b
007Ch	SCL/SDA Interrupt Control Register	SCLDAIC	XXXX X000b
0206h	Interrupt Source Select Register 2	IFSR2A	00h

When using the I2C-bus interface interrupt, set the IFSR22 bit in the IFSR2A register to 1 (I2C-bus interrupt). When using the SCL/SDA interrupt, set the IFSR23 bit in the IFSR2A register to 1 (SCL/SDA interrupt).

The SCL/SDA interrupt is enabled even in wait mode and stop mode.

The IR bit in the SCLDAIC register may become 1 (interrupt requested) when the ES0 bit in the S1D0 register, the SIP bit in the S2D0 register, or the SIS bit in the S2D0 register is changed. Therefore, follow the procedure below to change these bits. Refer to 13.13 “Notes on Interrupts”.

(1) Set bits ILVL2 to ILVL0 in the SCLDAIC register to 000b (interrupt disabled).
(2) Set the ES0 bit in the S1D0 register and bits SIP and SIS in the S2D0 register.
(3) Set the IR bit in the SCLDAIC register to 0 (no interrupt request).

21.5 Notes on Multi-Master I2C-bus Interface

21.5.1 Limitation on CPU Clock

When the CM07 bit in the CM0 register is 1 (CPU clock is a sub clock), do not access the registers listed in Table 21.4 “Register Configuration”. Set the CM07 bit to 0 (main clock, PLL clock, or on-chip oscillator clock) to access these registers.

21.5.2 Register Access

Refer to the notes below when accessing the I2C interface control registers. The period from the rising edge of the first clock of the slave address or 1-byte data transmission/reception to the falling edge of an ACK clock is considered to be the transmission/reception period. When the ACKCLK bit is 0 (no ACK clock), the transmission/reception period is from the rising edge of the first clock of the slave address or 1-byte data transmission/reception to the falling edge of the eighth clock.

21.5.2.1 S00 Register

Do not write to the S00 register during transmission/reception.

21.5.2.2 S1D0 Register

Do not change bits other than the IHR bit in the S1D0 register during transmission/reception.

21.5.2.3 S20 Register

Do not change bits other than the ACKBIT bit in the S20 register during transmission/reception.

21.5.2.4 S3D0 Register

- Do not use the bit managing instruction (read-modify-write instruction) to access the S3D0 register. Use the MOV instruction to write to this register.
- Rewrite bits ICK1 and ICK0 when the ES0 bit in the S1D0 register is 0 (I2C interface disabled).

21.5.2.5 S4D0 Register

Rewrite bits ICK4 to ICK2 when the ES0 bit in the S1D0 register is 0 (I2C interface disabled).

21.5.2.6 S10 Register

- Do not use the bit managing instruction (read-modify-write instruction) to access the S10 register. Use the MOV instruction to write to this register.
- Do not write to the S10 register when bits MST and TRX change their values.

Refer to operation examples in 21.3 “Operations” for bits MST and TRX change.

21.5.3 Generating Stop Condition

Technical update No.: TN-16C-A176A/E

In the multi-master I2C-bus interface, when the slave device and/or other master devices drive the SCLMM line low, no normal stop condition is generated. This is because the SDAMM line is released while the SCLMM line is still driven low.

21.5.4 Low/High-level Input Voltage and Low-level Output Voltage

The low-level input voltage, high-level input voltage, and low-level output voltage differ from the I2C-bus specification.
Refer to the recommended operating conditions for I/O ports which share the pins with SCL and SDA.

I2C-bus specification
High level input voltage (V_{IH}) = min. 0.7 V_{CC}
Low level input voltage (V_{IL}) = max. 0.3 V_{CC}

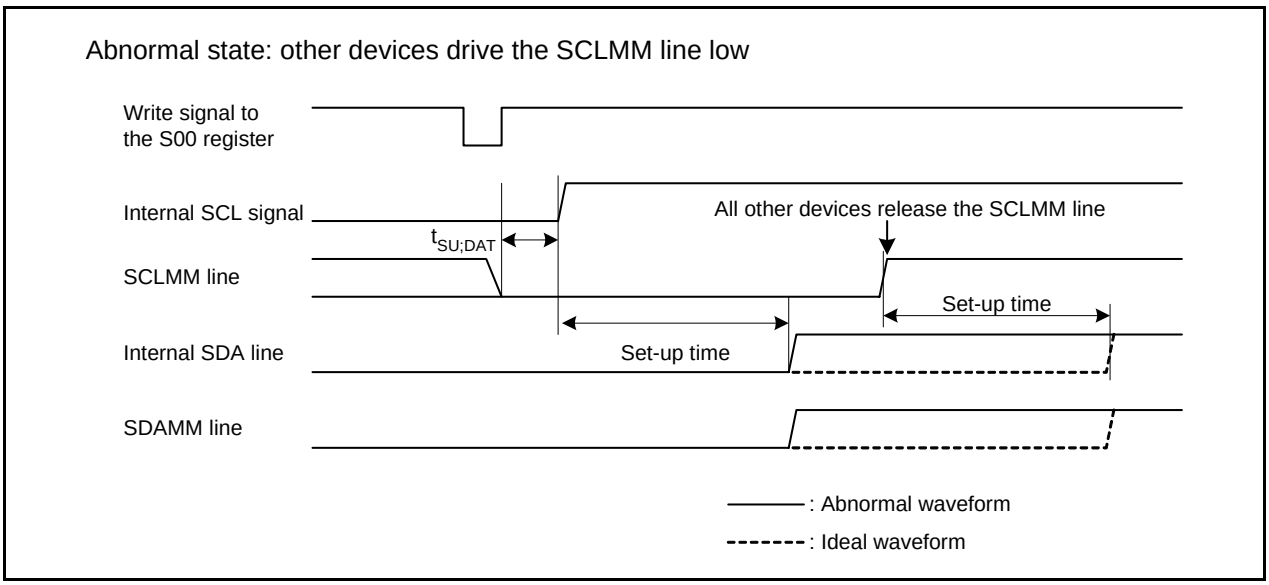


Figure 21.22 Abnormal Waveform

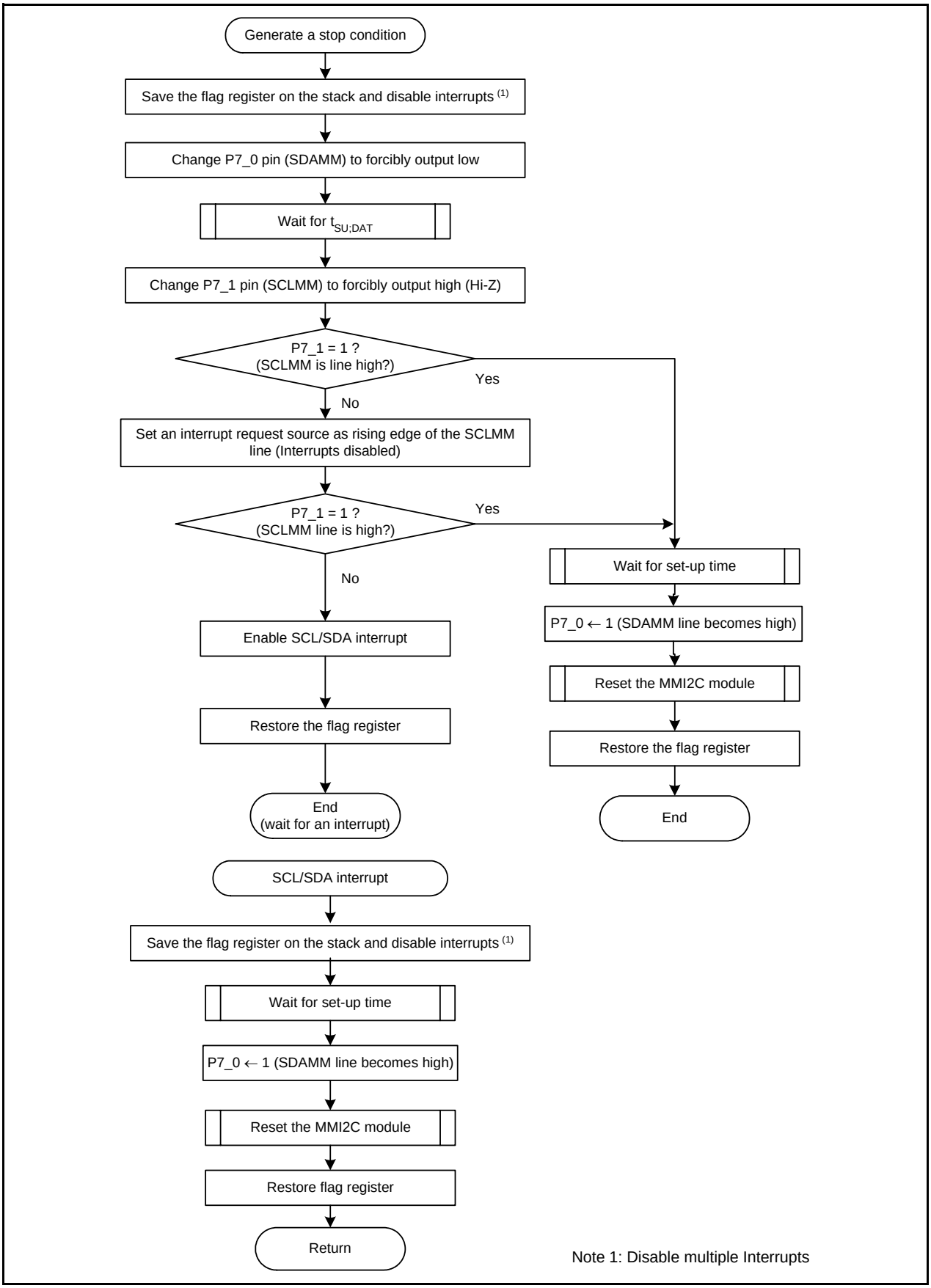


Figure 21.23 Generating a Stop Condition

22. A/D Converter

22.1 Introduction

The A/D converter consists of one 10-bit successive approximation A/D converter.
Table 22.1 lists the A/D Converter Specifications and Figure 22.1 shows an A/D Converter Block Diagram.

Table 22.1 A/D Converter Specifications

Item	Specification
A/D conversion method	Successive approximation
Analog input voltage	0 V to AVCC (VCC1)
Operating clock ϕAD	f1, f1 divided by 2, f1 divided by 3, f1 divided by 4, f1 divided by 6
Resolution	10 bits
Integral nonlinearity error	AVCC = VREF = 3.0 V AN0 to AN7 or AN0_0 to AN0_7 input: ±3 LSB ANEX0 or ANEX1 input: ±3 LSB
Operation modes	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, repeat sweep mode 1
Analog input pins	8 pins (AN0 to AN7) + 2 pins (ANEX0 and ANEX1) + 8 pins (AN0_0 to AN0_7)
A/D conversion start conditions	• Software trigger The ADST bit in the ADCON0 register is set to 1 (A/D conversion start). • External trigger (retrigger is enabled) Input to the $\overline{\text{ADTRG}}$ pin changes from high to low after the ADST bit is set to 1 (A/D conversion start).
Conversion rate per pin	Minimum 43 ϕAD cycles

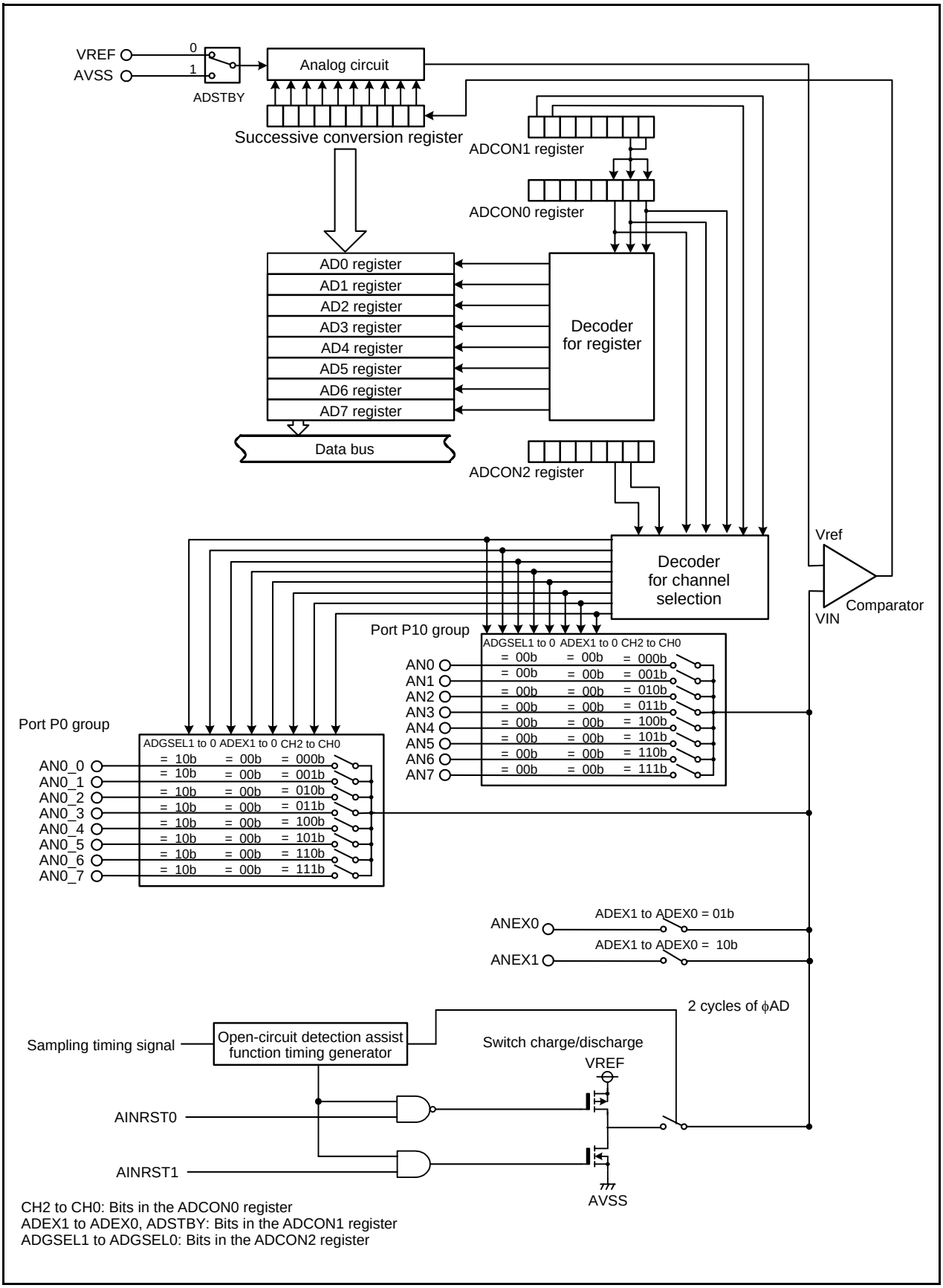


Figure 22.1 A/D Converter Block Diagram

Table 22.2 I/O Ports

Pin Name	I/O	Function
AN0 to AN7	Input	Analog input
ANEX0, ANEX1	Input	Analog input
AN0_0 to AN0_7	Input	Analog input
$\overline{\text{ADTRG}}$	Input	Trigger input

Note:
1. Set the direction bit of the ports sharing a port to 0 (input mode).

22.2 Registers

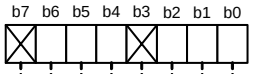
Table 22.3 lists registers associated with A/D converter.

Table 22.3 Registers

Address	Register	Symbol	Reset Value
0016h	Peripheral Clock Stop Register 1	PCLKSTP1	X000 0000b
0366h	Port Control Register	PCR	0000 0XX0b
03A2h	Open-Circuit Detection Assist Function Register	AINRST	XX00 XXXXb
03C0h	A/D Register 0	AD0	XXXX XXXXb
03C1h			0000 00XXb
03C2h	A/D Register 1	AD1	XXXX XXXXb
03C3h			0000 00XXb
03C4h	A/D Register 2	AD2	XXXX XXXXb
03C5h			0000 00XXb
03C6h	A/D Register 3	AD3	XXXX XXXXb
03C7h			0000 00XXb
03C8h	A/D Register 4	AD4	XXXX XXXXb
03C9h			0000 00XXb
03CAh	A/D Register 5	AD5	XXXX XXXXb
03CBh			0000 00XXb
03CCh	A/D Register 6	AD6	XXXX XXXXb
03CDh			0000 00XXb
03CEh	A/D Register 7	AD7	XXXX XXXXb
03CFh			0000 00XXb
03D4h	A/D Control Register 2	ADCON2	0000 X00Xb
03D6h	A/D Control Register 0	ADCON0	0000 0XXXb
03D7h	A/D Control Register 1	ADCON1	0000 0000b

22.2.1 Peripheral Clock Stop Register 1 (PCLKSTP1)

Peripheral Clock Stop Register 1



Symbol	Address	After Reset
PCLKSTP1	0016h	X000 0000b

Bit Symbol	Bit Name	Function	RW
PCKSTP10	Real-time clock peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP11	Timer peripheral clock stop bit (timer A, timer B)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP12	UART peripheral clock stop bit (UART0 to 2, UART5 to 7)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b3)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—
PCKSTP14	AD peripheral clock stop bit	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP15	SIO peripheral clock stop bit (SI/O3, SI/O4)	0: f1 provide enabled 1: f1 provide disabled	RW
PCKSTP16	Multi-master I^2C peripheral clock stop bit (multi-master I^2C-bus interface)	0: f1 provide enabled 1: f1 provide disabled	RW
— (b7)	No register bit. If necessary, set to 0. When read, the read value is undefined.		—

Set the PRC0 bit in the PRCR register to 1 (write enabled) before the PCKSTP1 register is rewritten.

PCKSTP14 (AD peripheral clock stop bit) (b4)

Set the PCKSTP14 bit to 0 (f1 provide enabled) when using the f1 as the clock source of fAD.

22.2.2 Port Control Register (PCR)

Port Control Register

b7 b6 b5 b4 b3 b2 b1 b0	Symbol	Address	After Reset
	PCR	0366h	0000 0XX0b

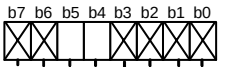
Bit Symbol	Bit Name	Function	RW
PCR0	Port P1 control bit	Operation performed when the P1 register is read 0: When the port is set to input, the input levels of pins P1_0 to P1_7 are read. When set to output, the port latch is read. 1: The port latch is read regardless of whether the port is set to input or output.	RW
— (b2-b1)	No register bits. If necessary, set to 0. Read as undefined.		—
PCR3	Key input enable bit ($\overline{KI4}$ to $\overline{KI7}$)	0: Enabled 1: Disabled	RW
— (b4)	No register bits. Read as undefined.		—
PCR5	$\overline{INT6}$ input enable bit	0: Enabled 1: Disabled	RW
PCR6	$\overline{INT7}$ input enable bit	0: Enabled 1: Disabled	RW
PCR7	Key input enable bit ($\overline{KI0}$ to $\overline{KI3}$)	0: Enabled 1: Disabled	RW

PCR3 (Key input enable bit) (b3)
PCR7 (Key input enable bit) (b7)

Set the PCR7 bit to 1 (key input disabled) when using pins AN4 to AN7 for analog input. Set the PCR3 bit to 1 (key input disabled) when using pins AN0 to AN3 for analog input.

22.2.3 Open-Circuit Detection Assist Function Register (AINRST)

Open-Circuit Detection Assist Function Register



Symbol: AINRST | Address: 03A2h | After Reset: XX00 XXXXb

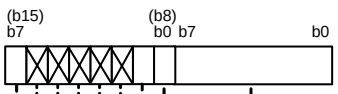
Bit Symbol	Bit Name	Function	RW
— (b3-b0)	No register bits. If necessary, set to 0. Read as undefined value		—
AINRST0	Open-circuit detection assist function enable bit	b5 b4 0 0: Open-circuit detection disabled 0 1: Charge before conversion 1 0: Discharge before conversion 1 1: Do not set	RW
AINRST1			RW
— (b7-b6)	No register bits. If necessary, set to 0. Read as undefined value.		—

AINRST1 to AINRST0 (Open-circuit detection assist function enable bit) (b5 to b4)

To enable the A/D open-circuit detection assist function, set the AINRST0 bit or AINRST1 bit to 1, and then set the ADST bit in the ADCON0 register to 1 (A/D conversion) after waiting for one cycle of ϕAD.

22.2.4 AD Register i (ADi) (i = 0 to 7)

A/D Register (i = 0 to 7)



Symbol	Address	After Reset
AD0	03C1h to 03C0h	0000 00XX XXXX XXXXb
AD1	03C3h to 03C2h	0000 00XX XXXX XXXXb
AD2	03C5h to 03C4h	0000 00XX XXXX XXXXb
AD3	03C7h to 03C6h	0000 00XX XXXX XXXXb
AD4	03C9h to 03C8h	0000 00XX XXXX XXXXb
AD5	03CBh to 03CAh	0000 00XX XXXX XXXXb
AD6	03CDh to 03CCh	0000 00XX XXXX XXXXb
AD7	03CFh to 03CEh	0000 00XX XXXX XXXXb

Function		RW
Eight low-order bits of A/D conversion result		RO
Two high-order bits of A/D conversion result		RO
No register bits. If necessary, set to 0. Read as 0.		—
Reserved bit	Read as 0.	RO

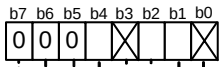
The A/D conversion result is stored in the ADi register corresponding to pins ANi, ANEXi, and AN0_i. Read the ADi register in 16-bit units. Table 22.4 lists Analog Pin and A/D Conversion Result Storing Register.

Table 22.4 Analog Pin and A/D Conversion Result Storing Register

Analog Pin			A/D Conversion Result Storing Register
AN0	ANEX0	AN0_0	AD0 register
AN1	ANEX1	AN0_1	AD1 register
AN2	—	AN0_2	AD2 register
AN3	—	AN0_3	AD3 register
AN4	—	AN0_4	AD4 register
AN5	—	AN0_5	AD5 register
AN6	—	AN0_6	AD6 register
AN7	—	AN0_7	AD7 register

22.2.5 A/D Control Register 2 (ADCON2)

A/D Control Register 2



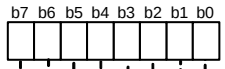
Symbol: ADCON2 | Address: 03D4h | After Reset: 0000 X00Xb

Bit Symbol	Bit Name	Function	RW
— (b0)	No register bit. If necessary, set to 0. Read as undefined value.		—
ADGSEL0	A/D input group select bit	b2 b1 0 0: AN0 to AN7 0 1: Do not set 1 0: AN0_0 to AN0_7 1 1: Do not set	RW
ADGSEL1			RW
— (b3)	No register bit. If necessary, set to 0. Read as undefined value.		—
CKS2	Frequency select bit 2	Refer to the CKS0 bit in the ADCON0 register.	RW
— (b7-b5)	Reserved bits	Set to 0.	RW

If the ADCON2 register is rewritten during A/D conversion, the conversion result is undefined.

22.2.6 A/D Control Register 0 (ADCON0)

A/D Control Register 0



Symbol	Address	After Reset
ADCON0	03D6h	0000 0XXXb

Bit Symbol	Bit Name	Function	RW
CH0	Analog input pin select bit	In one-shot mode or repeat mode b2 b1 b0 0 0 0: AN0 0 0 1: AN1 0 1 0: AN2 0 1 1: AN3 1 0 0: AN4 1 0 1: AN5 1 1 0: AN6 1 1 1: AN7	RW
CH1			RW
CH2			RW
MD0	A/D operation mode select bit 0	b4 b3 0 0: One-shot mode 0 1: Repeat mode 1 0: Single sweep mode 1 1: Repeat sweep mode 0 or repeat sweep mode 1	RW
MD1			RW
TRG	Trigger select bit	0: Software trigger 1: ADTRG trigger	RW
ADST	A/D conversion start flag	0: A/D conversion stop 1: A/D conversion start	RW
CKS0	Frequency select bit 0	Refer to the next page.	RW

If the ADCON0 register is rewritten during A/D conversion, the conversion result is undefined.

CH2 to CH0 (Analog input pin select bit) (b2 to b0)

In one-shot and repeat modes, pins AN0_0 to AN0_7 can be used in the same way as pins AN0 to AN7. Use bits ADGSEL1 to ADGSEL0 in the ADCON2 register to select the desired group.
These bits are disabled in single sweep mode, repeat sweep mode 0, and repeat sweep mode 1.

MD1 to MD0 (A/D operation mode select bit 0) (b4 to b3)

A/D operation mode is selected by a combination of bits MD1 to MD0 and the MD2 bit in the ADCON1 register. Table 22.5 lists A/D Operation Mode.

Table 22.5 A/D Operation Mode

Bit Setting			A/D Operation Mode
ADCON1 Register	ADCON0 Register		
MD2	MD1	MD0	
0	0	0	One-shot mode
0	0	1	Repeat mode
0	1	0	Single sweep mode
0	1	1	Repeat sweep mode 0
1	1	1	Repeat sweep mode 1

Do not set bit combinations not listed above.

CKS0 (Frequency select bit) (b7)

ϕAD frequency is selected by a combination of the CKS0 bit in the ADCON0 register, the CKS1 bit in the ADCON1 register, and the CKS2 bit in the ADCON2 register. Table 22.6 lists ϕAD Frequency.

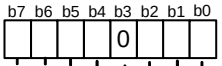
Table 22.6 ϕAD Frequency

CKS2	CKS1	CKS0	ϕAD
0	0	0	fAD (f1) divided by 4
0	0	1	fAD (f1) divided by 2
0	1	0	fAD (f1)
0	1	1	
1	0	1	fAD (f1) divided by 6
1	1	0	fAD (f1) divided by 3
1	1	1	

Only set the values listed above.

22.2.7 A/D Control Register 1 (ADCON1)

A/D Control Register 1



Symbol: ADCON1 | Address: 03D7h | After Reset: 0000 0000b

Bit Symbol	Bit Name	Function	RW
SCAN0	A/D sweep pin select bit	In single sweep mode or repeat sweep mode 0 b1 b0 0 0: AN0 to AN1 (2 pins) 0 1: AN0 to AN3 (4 pins) 1 0: AN0 to AN5 (6 pins) 1 1: AN0 to AN7 (8 pins)	RW
SCAN1		In repeat sweep mode 1 b1 b0 0 0: AN0 (1 pin) 0 1: AN0 to AN1 (2 pins) 1 0: AN0 to AN2 (3 pins) 1 1: AN0 to AN3 (4 pins)	RW
MD2	A/D operation mode select bit 1	0: Any mode other than repeat sweep mode 1 1: Repeat sweep mode 1	RW
— (b3)	Reserved bit	Set to 0.	RW
CKS1	Frequency select bit 1	Refer to the CKS0 bit in the ADCON0 register	RW
ADSTBY	A/D standby bit	0: A/D operation stopped (standby) 1: A/D operation enabled	RW
ADEX0	Extended pin select bit	In one-shot mode or repeat mode b7 b6 0 0: ANEX0 to ANEX1 are not used 0 1: ANEX0 input is A/D converted 1 0: ANEX1 input is A/D converted 1 1: Do not set this value	RW
ADEX1			RW

If the ADCON1 register is rewritten during A/D conversion, the conversion result is undefined.

SCAN1 to SCAN0 (A/D sweep pin select bit) (b1 to b0)

These bits are disabled in one-shot and repeat modes. In single sweep mode, repeat sweep mode 0, and repeat sweep mode 1, pins AN0_0 to AN0_7 can be used in the same way as pins AN0 to AN7. Use bits ADGSEL1 to ADGSEL0 in the ADCON2 register to select the desired group.

MD2 (A/D operation mode select bit 1) (b2)

A/D operation mode is selected by a combination of bits MD1 to MD0 in the ADCON0 register and the MD2 bit. See Table 22.5 “A/D Operation Mode”.

ADSTBY (A/D standby bit) (b5)

If the ADSTBY bit is changed from 0 (A/D operation stopped) to 1 (A/D operation enabled), wait for 1 ϕAD cycle or more before starting A/D conversion.
When the A/D converter is not used, no current flows in the A/D converter by setting the ADSTBY bit to 0 (A/D operation stopped: standby). This helps the power consumption to be reduced.

22.3 Operations

22.3.1 A/D Conversion Cycle

A/D conversion cycle is based on fAD (same as f1) and ϕAD. Divide fAD so ϕAD conforms the standard frequency. Figure 22.2 shows fAD and ϕAD.
Set the PCKSTP14 bit in the PCLKSTP1 register to 0 (f1 provide enabled) when using as the clock source of fAD.

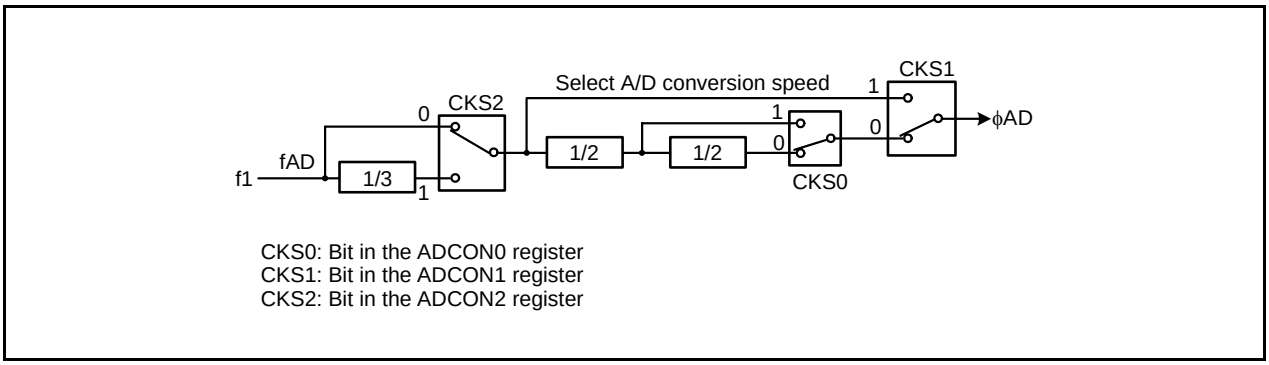


Figure 22.2 fAD and ϕAD

Figure 22.3 shows A/D Conversion Timing.

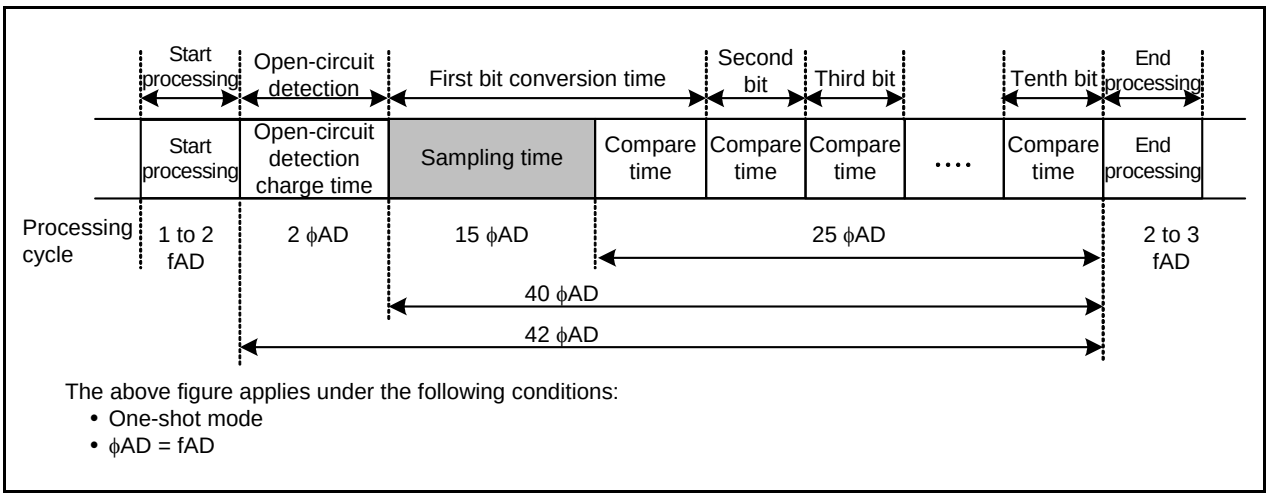


Figure 22.3 A/D Conversion Timing

Table 22.7 lists Cycles of A/D Conversion Item. A/D conversion period is as follows.
Start processing time depends on which ϕAD is selected.
A/D conversion starts after the start processing time elapses by setting the ADST bit in the ADCON0 register to 1 (A/D conversion start). When reading the ADST bit before starting A/D conversion, 0 (A/D conversion stop) is read.
When selecting multiple pins and in A/D conversion repeat mode, between-execution processing time is inserted between A/D conversions.
In one-shot mode and single sweep mode, the ADST bit becomes 0 at the end processing time and the last A/D conversion result is stored in the ADi register.

- One-shot mode:
  Start processing time + A/D conversion execution time + end processing time
- Two pins are selected in single sweep mode:
  Start processing time + (A/D conversion execution time + between-execution processing time + A/D conversion execution time) + end processing time

Table 22.7 Cycles of A/D Conversion Item

A/D Conversion Item		Cycle
Start processing time	ϕAD = fAD	1 to 2 cycles of fAD
	ϕAD = fAD divided by 2	2 to 3 cycles of fAD
	ϕAD = fAD divided by 3	3 to 4 cycles of fAD
	ϕAD = fAD divided by 4	3 to 4 cycles of fAD
	ϕAD = fAD divided by 6	4 to 5 cycles of fAD
	ϕAD = fAD divided by 12	7 to 8 cycles of fAD
A/D conversion execution time	Open-circuit detection disabled	40 cycles of ϕAD
	Open-circuit detection enabled	42 cycles of ϕAD
Between-execution processing time		1 cycle of ϕAD
End processing time		2 to 3 cycles of fAD

22.3.2 A/D Conversion Start Conditions

An A/D conversion start trigger has a software trigger and an external trigger. Figure 22.4 shows A/D Conversion Start Trigger.

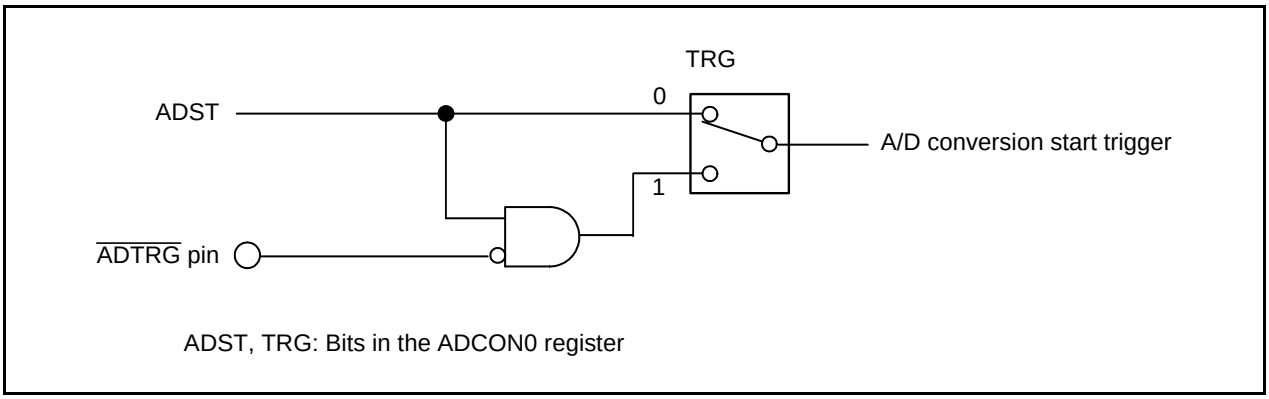


Figure 22.4 A/D Conversion Start Trigger

22.3.2.1 Software Trigger

When the TRG bit in the ADCON0 register is 0 (software trigger), A/D conversion starts by setting the ADST bit in the ADCON0 register to 1 (A/D conversion start).

22.3.2.2 External Trigger

When the TRG bit in the ADCON0 register is 1 ($\overline{\text{ADTRG}}$ trigger), A/D conversion starts if the input level at the $\overline{\text{ADTRG}}$ pin changes from high to low under the following conditions:

- The direction bit of the port which shares the pin with $\overline{\text{ADTRG}}$ 0 (input mode)
- The TRG bit in the ADCON0 register is 1 ($\overline{\text{ADTRG}}$ trigger)
- The ADST bit in the ADCON0 register is 1 (A/D conversion start)

Under the above conditions, when input to the $\overline{\text{ADTRG}}$ pin is changed from high to low, the A/D conversion starts.
Set the high- and low-level durations of the pulse input to the $\overline{\text{ADTRG}}$ pin to two or more cycles of fAD.

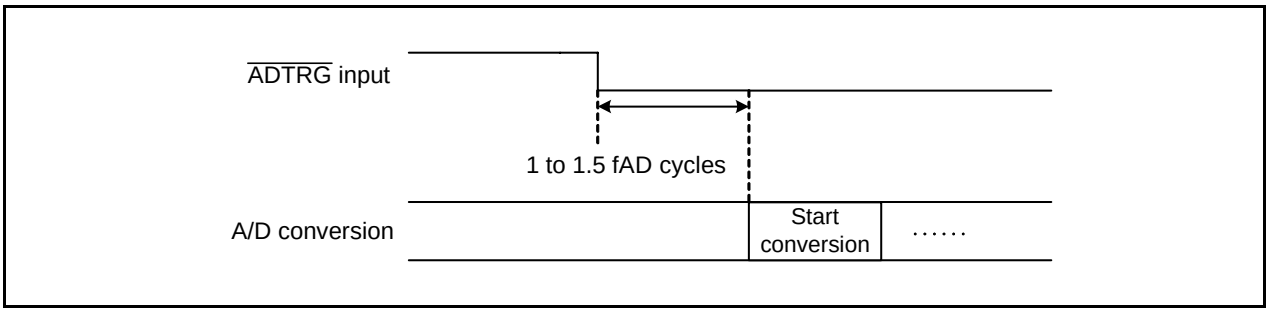


Figure 22.5 A/D conversion Start Timing When External Trigger Input

22.3.3 A/D Conversion Result

When reading the ADi register before A/D conversion is completed, the undefined value is read. Read the ADi register after completing A/D conversion. Use the following procedure to detect the completion of A/D conversion.

- In one-shot mode and single sweep mode:
  The IR bit in the ADIC register becomes 1 (interrupt requested) at the completion of A/D conversion. Ensure that the IR bit becomes 1 to read the ADi register.
  When not using A/D interrupt, set the IR bit to 0 (interrupt not requested) by a program after reading the ADi register.

- In repeat mode, repeat sweep mode 0, and repeat sweep mode 1:
  The IR bit remain unchanged (no interrupt request is generated). At first, read the ADi register after one A/D conversion period elapses (refer to 22.3.1 “A/D Conversion Cycle”). After that, whenever the ADi register is read, the conversion result which has been obtained before reading is read.
  The ADi register is overwritten in every A/D conversion. Read the value before the ADi register is overwritten.

22.3.4 Extended Analog Input Pins

In one-shot mode and repeat modes, pins ANEX0 and ANEX1 can be used as analog input pins by setting bits ADEX1 to ADEX0 in the ADCON1 register.
The A/D conversion result of pins ANEX0 and ANEX1 are stored in registers AD0 and AD1, respectively.

22.3.5 Current Consumption Reduce Function

When the A/D converter is not in use, the power consumption can be reduced by setting the ADSTBY bit in the ADCON1 register to 0 (A/D operation stopped: standby) to shut off any analog circuit current flow.
To use the A/D converter, set the ADSTBY bit to 1 (A/D operation enabled) and wait for one ϕAD cycle or more before setting the ADST bit in the ADCON0 register to 1 (A/D conversion start). Do not set bits ADST and ADSTBY to 1 at the same time.
Also, do not set the ADSTBY bit to 0 (A/D operation stopped: standby) during A/D conversion.

22.3.6 Open-Circuit Detection Assist Function

The A/D converter has a function to set the charge of the sampling capacitor to a predefined state (AVCC or AVSS) before A/D conversion starts. This helps prevent the influence of analog input voltage from the previous conversion and more reliably detect an open-circuit of a trace connected to an analog input pin.
Figure 22.6 shows A/D Open-Circuit Detection Example on AVCC (Preconversion Charge) and Figure 22.8 shows A/D Open-Circuit Detection Example on AVSS (Preconversion Discharge).
The conversion result in open-circuit depends on the external circuit. Use this function only after careful evaluation for the system.

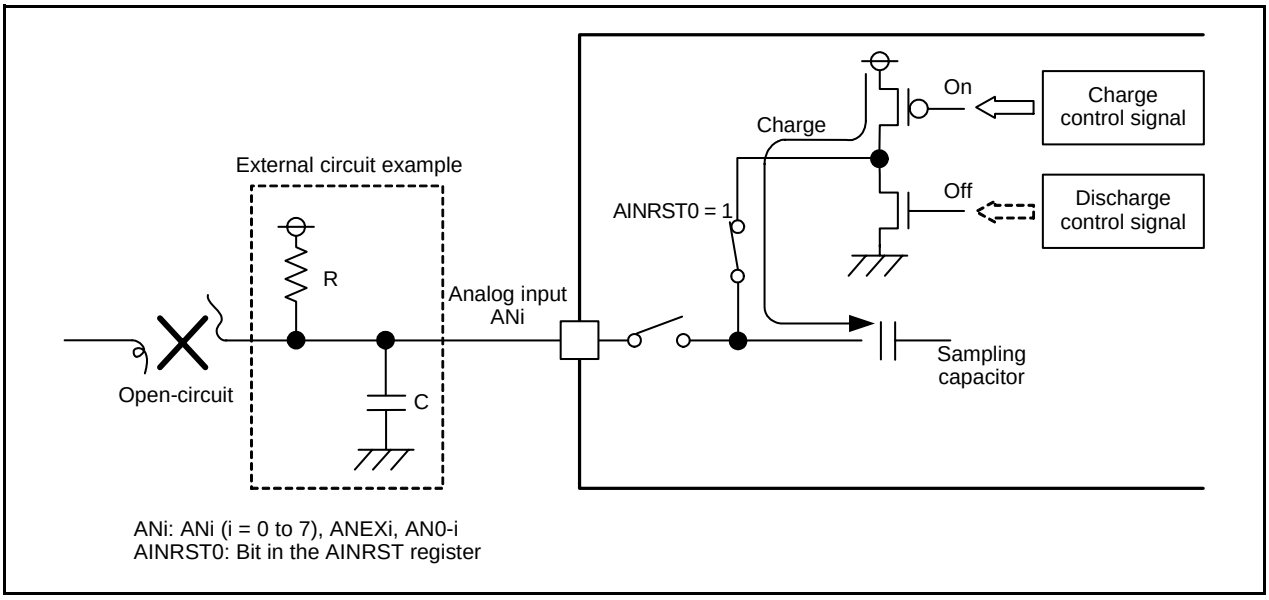
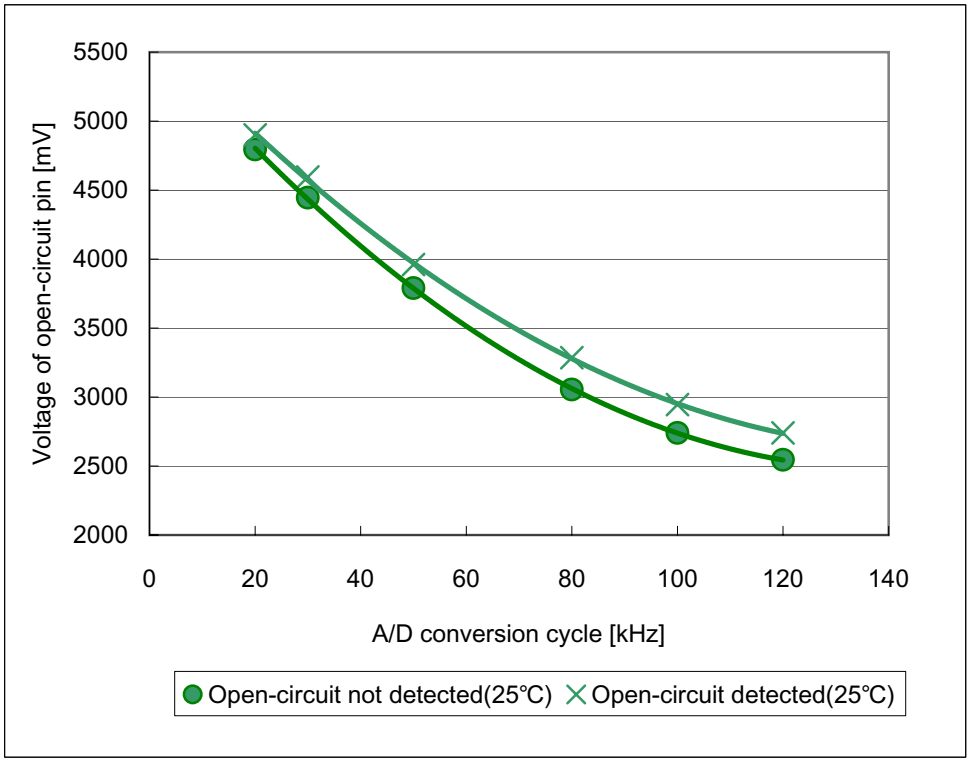


Figure 22.6 A/D Open-Circuit Detection Example on AVCC (Preconversion Charge)



The horizontal axis shows how many times A/D conversion is performed per ms.
For example, 20 kHz indicates A/D conversion is performed every 50 μs.

Figure 22.7 A/D Open-Circuit Detection (Charge) Characteristics (Standard Characteristics)

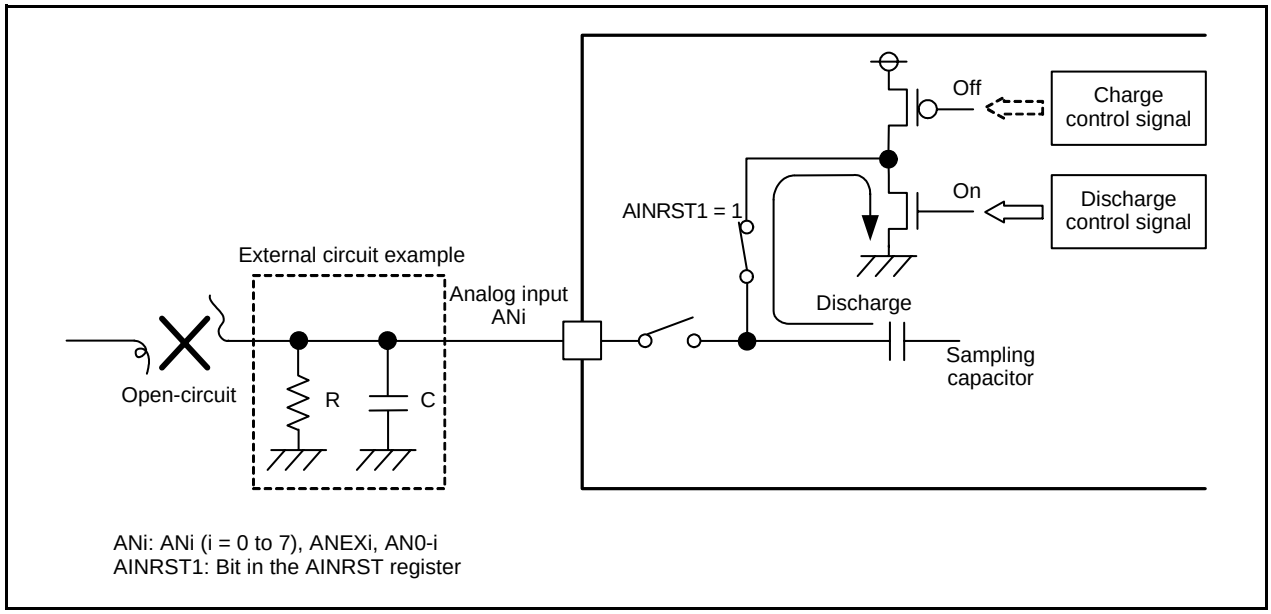
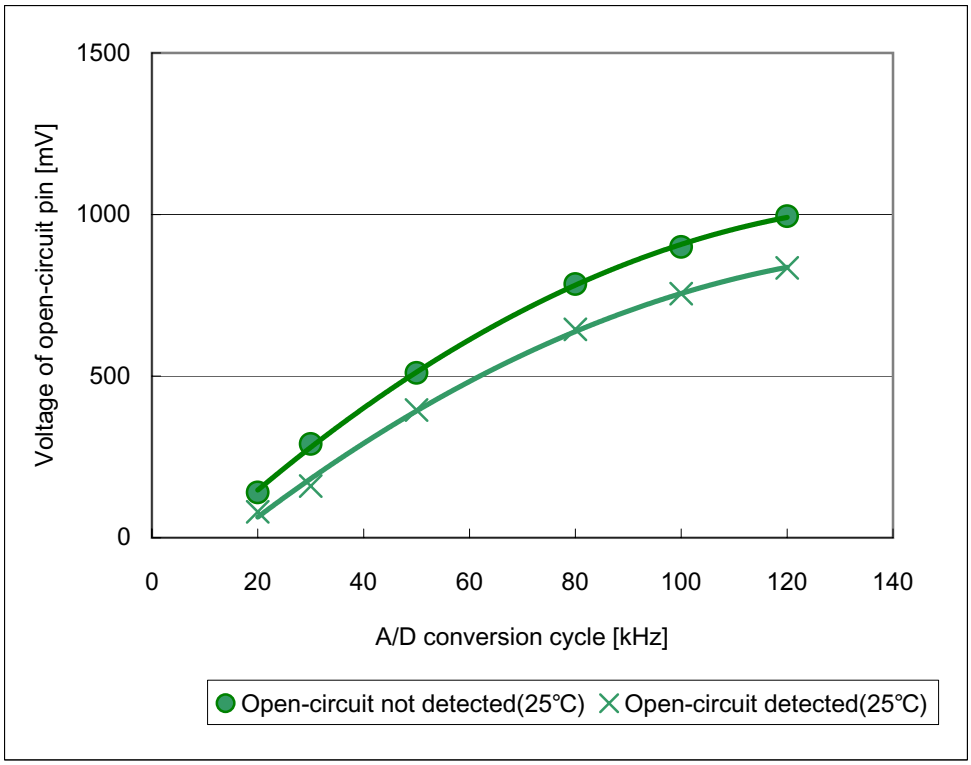


Figure 22.8 A/D Open-Circuit Detection Example on AVSS (Preconversion Discharge)



The horizontal axis shows how many times A/D conversion is performed per ms.
For example, 20 kHz indicates A/D conversion is performed every 50 μs.

Figure 22.9 A/D Open-Circuit Detection (Discharge) Characteristics (Standard Characteristics)

22.4 Operational Modes

22.4.1 One-Shot Mode

In one-shot mode, the analog voltage applied to a selected pin is converted to a digital code once. Table 22.8 lists One-Shot Mode Specifications.

Table 22.8 One-Shot Mode Specifications

Item	Specification
Function	Bits CH2 to CH0 in the ADCON0 register and bits ADGSEL1 to ADGSEL0 in the ADCON2 register, or bits ADEX1 to ADEX0 in the ADCON1 register are used to select a pin. The analog voltage applied to the pin is converted to a digital code once.
A/D conversion start conditions	• When the TRG bit in the ADCON0 register is 0 (software trigger) the ADST bit in the ADCON0 register is set to 1 (A/D conversion starts). • When the TRG bit is 1 ($\overline{\text{ADTRG}}$ trigger) input level at the $\overline{\text{ADTRG}}$ pin changes from high to low after the ADST bit is set to 1 (A/D conversion start).
A/D conversion stop conditions	• Completion of A/D conversion (if a software trigger is selected, the ADST bit becomes 0 (A/D conversion stop)). • Set the ADST bit to 0.
Interrupt request generation timing	Completion of A/D conversion.
Analog input pin	Select one pin from among AN0 to AN7, AN0_0 to AN0_7, ANEX0, and ANEX1.
Reading of A/D conversion result	Read the register among AD0 to AD7 that corresponds to the selected pin.

Table 22.9 Registers and Settings in One-Shot Mode (1)

Register	Bit	Setting
PCLKSTP1	PCKSTP14	Set to 0 when using f1.
PCR	PCR3	Set to 1 (key input disabled) when using pins AN0 to AN3 for analog input.
	PCR7	Set to 1 (key input disabled) when using pins AN4 to AN7 for analog input.
AINRST	AINRST1, AINRST0	Select whether open-circuit detection assist function is used or not.
AD0 to AD7	b9 to b0	A/D conversion result can be read.
ADCON2	ADGSEL1, ADGSEL0	Select analog input pin group.
	CKS2	Select ϕAD frequency.
ADCON0	CH2 to CH0	Select analog input pin.
	MD1 to MD0	Set to 00b.
	TRG	Select a trigger.
	ADST	Set to 1 to start A/D conversion and set to 0 to stop it.
	CKS0	Select ϕAD frequency.
ADCON1	SCAN1, SCAN0	Disabled
	MD2	Set to 0.
	CKS1	Select ϕAD frequency.
	ADSTBY	Set to 1 in A/D conversion.
	ADEX1, ADEX0	Select whether ANEX0 and ANEX1 are used or not

Note:
1. This table does not describe a procedure.

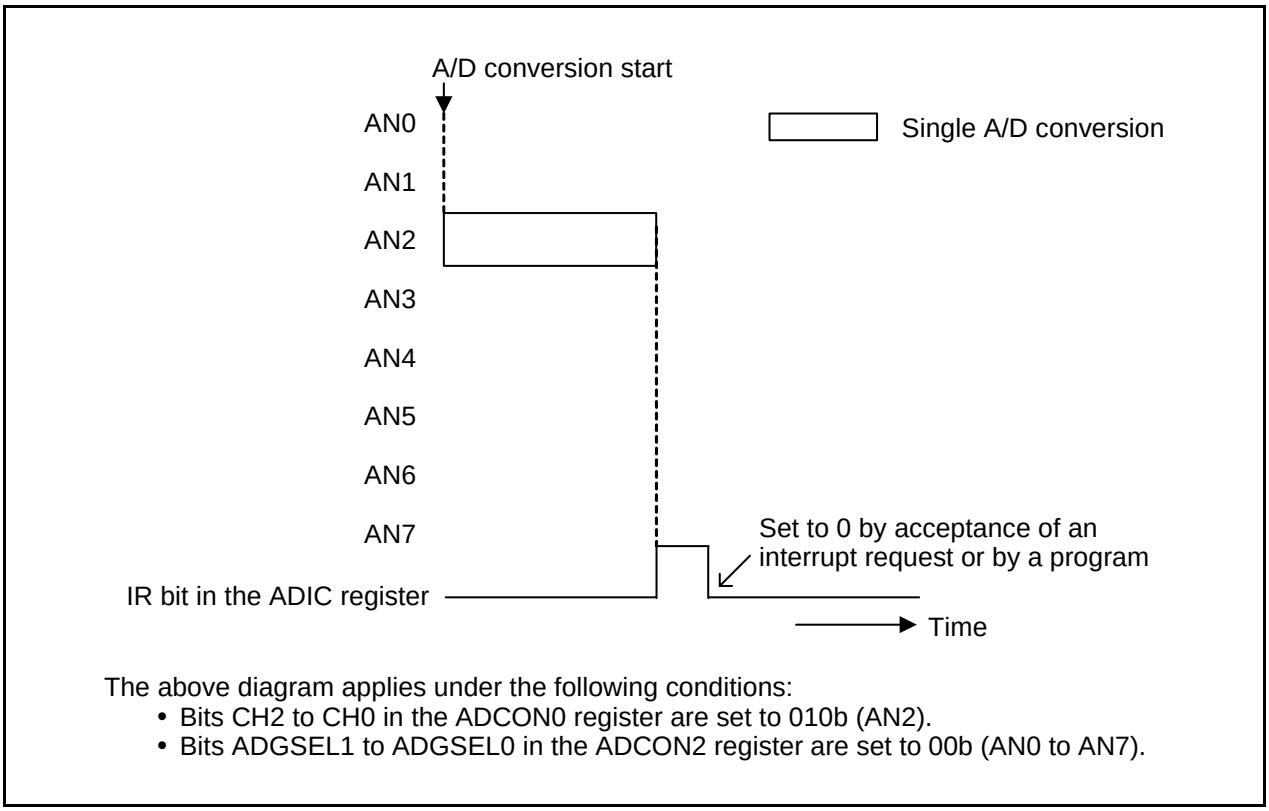


Figure 22.10 Operation Example in One-Shot Mode

22.4.2 Repeat Mode

In repeat mode, the analog voltage applied to a selected pin is repeatedly converted to a digital code. Table 22.10 lists Repeat Mode Specifications.

Table 22.10 Repeat Mode Specifications

Item	Specification
Function	Bits CH2 to CH0 in the ADCON0 register and bits ADGSEL1 to ADGSEL0 in the ADCON2 register, or bits ADEX1 to ADEX0 in the ADCON1 register are used to select a pin. The analog voltage applied to the pin is repeatedly converted to a digital code.
A/D conversion start conditions	• When the TRG bit in the ADCON0 register is 0 (software trigger) the ADST bit in the ADCON0 register is set to 1 (A/D conversion start). • When the TRG bit is 1 ($\overline{\text{ADTRG}}$ trigger) input level at the $\overline{\text{ADTRG}}$ pin changes from high to low after the ADST bit is set to 1 (A/D conversion start).
A/D conversion stop condition	Set the ADST bit to 0 (A/D conversion stop).
Interrupt request generation timing	No interrupt requests generated
Analog input pin	Select one pin from among AN0 to AN7, AN0_0 to AN0_7, ANEX0, and ANEX1.
Reading of A/D conversion result	Read the register among AD0 to AD7 that corresponds to the selected pin.

Table 22.11 Registers and Settings in Repeat Mode (1)

Register	Bit	Setting
PCLKSTP1	PCKSTP14	Set to 0 when using f1.
PCR	PCR3	Set to 1 (key input disabled) when using pins AN0 to AN3 for analog input.
	PCR7	Set to 1 (key input disabled) when using pins AN4 to AN7 for analog input.
AINRST	AINRST1, AINRST0	Select whether open-circuit detection assist function is used or not.
AD0 to AD7	b9 to b0	A/D conversion result can be read.
ADCON2	ADGSEL1, ADGSEL0	Select analog input pin group.
	CKS2	Select ϕAD frequency.
ADCON0	CH2 to CH0	Select analog input pin.
	MD1 to MD0	Set to 01b.
	TRG	Select a trigger.
	ADST	Set to 1 to start A/D conversion and set to 0 to stop it.
	CKS0	Select ϕAD frequency.
ADCON1	SCAN1, SCAN0	Disabled
	MD2	Set to 0.
	CKS1	Select ϕAD frequency.
	ADSTBY	Set to 1 in A/D conversion.
	ADEX1, ADEX0	Select whether ANEX0 and ANEX1 are used or not

Note:
1. This table does not describe a procedure.

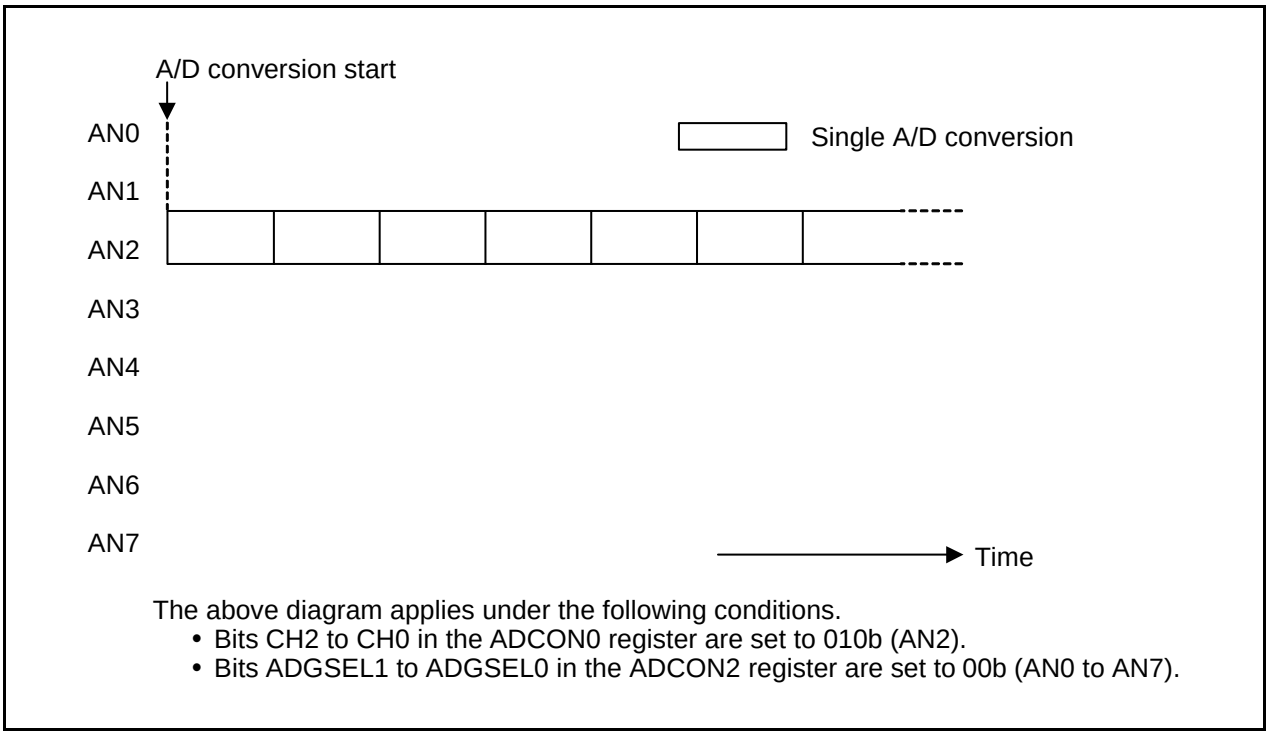


Figure 22.11 Operation Example in Repeat Mode

22.4.3 Single Sweep Mode

In single sweep mode, the analog voltage applied to selected pins is converted one-by-one to a digital code. Table 22.12 lists the Single Sweep Mode Specifications.

Table 22.12 Single Sweep Mode Specifications

Item	Specification
Function	Bits SCAN1 to SCAN0 in the ADCON1 register and bits ADGSEL1 to ADGSEL0 in the ADCON2 register are used to select pins. The analog voltage applied to the pins is converted one-by-one to a digital code.
A/D conversion start conditions	• When the TRG bit in the ADCON0 register is 0 (software trigger) the ADST bit in the ADCON0 register is set to 1 (A/D conversion start). • When the TRG bit is 1 ($\overline{\text{ADTRG}}$ trigger) input level at the $\overline{\text{ADTRG}}$ pin changes from high to low after the ADST bit is set to 1 (A/D conversion start).
A/D conversion stop conditions	• Completion of A/D conversion (if a software trigger is selected, the ADST bit is set to 0 (A/D conversion stop)). • Set the ADST bit to 0.
Interrupt request generation timing	Completion of A/D conversion
Analog input pin	Select from AN0 and AN1 (2 pins), AN0 to AN3 (4 pins), AN0 to AN5 (6 pins), and AN0 to AN7 (8 pins). AN0_0 to AN0_7 can be selected in the same way.
Reading of A/D conversion result	Read the registers among AD0 to AD7 that corresponds to the selected pin.

Table 22.13 Registers and Settings in Single Sweep Mode (1)

Register	Bit	Setting
PCLKSTP1	PCKSTP14	Set to 0 when using f1.
PCR	PCR3	Set to 1 (key input disabled) when using pins AN0 to AN3 for analog input.
	PCR7	Set to 1 (key input disabled) when using pins AN4 to AN7 for analog input.
AINRST	AINRST1, AINRST0	Select whether open-circuit detection assist function is used or not.
AD0 to AD7	b9 to b0	A/D conversion result can be read.
ADCON2	ADGSEL1, ADGSEL0	Select analog input pin group.
	CKS2	Select ϕAD frequency.
ADCON0	CH2 to CH0	Disabled
	MD1 to MD0	Set to 10b.
	TRG	Select a trigger.
	ADST	Set to 1 to start A/D conversion and set to 0 to stop it.
	CKS0	Select ϕAD frequency.
ADCON1	SCAN1, SCAN0	Select analog input pin.
	MD2	Set to 0.
	CKS1	Select ϕAD frequency.
	ADSTBY	Set to 1 in A/D conversion.
	ADEX1, ADEX0	Set to 00b.

Note:
1. This table does not describe a procedure.

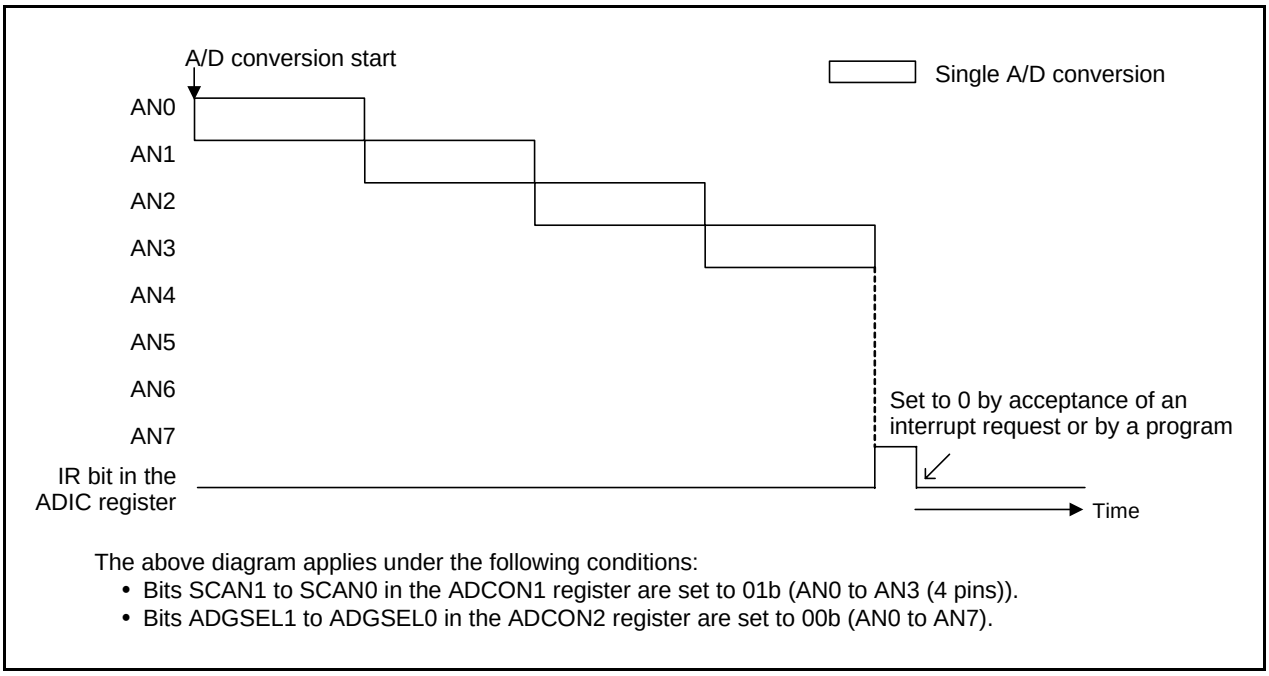


Figure 22.12 Operation Example in Single Sweep Mode

22.4.4 Repeat Sweep Mode 0

In repeat sweep mode 0, the analog voltage applied to selected pins is repeatedly converted to a digital code. Table 22.14 lists the Repeat Sweep Mode 0 Specifications.

Table 22.14 Repeat Sweep Mode 0 Specifications

Item	Specification
Function	Bits SCAN1 to SCAN0 in the ADCON1 register and bits ADGSEL1 to ADGSEL0 in the ADCON2 register are used to select pins. Analog voltage applied to the pins is repeatedly converted to a digital code.
A/D conversion start conditions	• When the TRG bit in the ADCON0 register is 0 (software trigger) • the ADST bit in the ADCON0 register is set to 1 (A/D conversion start). • When the TRG bit is 1 ($\overline{\text{ADTRG}}$ trigger) input level at the $\overline{\text{ADTRG}}$ pin changes from high to low after the ADST bit is set to 1 (A/D conversion start).
A/D conversion stop condition	Set the ADST bit to 0 (A/D conversion stop).
Interrupt request generation timing	No interrupt requests generated
Analog input pin	Select from AN0 and AN1 (2 pins), AN0 to AN3 (4 pins), AN0 to AN5 (6 pins), and AN0 to AN7 (8 pins). AN0_0 to AN0_7 can be selected in the same way.
Reading of A/D conversion result	Read the registers among AD0 to AD7 that corresponds to the selected pins.

Table 22.15 Registers and Settings in Repeat Sweep Mode 0 (1)

Register	Bit	Setting
PCLKSTP1	PCKSTP14	Set to 0 when using f1.
PCR	PCR3	Set to 1 (key input disabled) when using pins AN0 to AN3 for analog input.
	PCR7	Set to 1 (key input disabled) when using pins AN4 to AN7 for analog input.
AINRST	AINRST1, AINRST0	Select whether open-circuit detection assist function is used or not.
AD0 to AD7	b9 to b0	A/D conversion result can be read.
ADCON2	ADGSEL1, ADGSEL0	Select analog input pin group.
	CKS2	Select ϕAD frequency.
ADCON0	CH2 to CH0	Disabled
	MD1 to MD0	Set to 11b.
	TRG	Select a trigger.
	ADST	Set to 1 to start A/D conversion and set to 0 to stop it.
	CKS0	Select ϕAD frequency.
ADCON1	SCAN1, SCAN0	Select analog input pin.
	MD2	Set to 0.
	CKS1	Select ϕAD frequency.
	ADSTBY	Set to 1 in A/D conversion.
	ADEX1, ADEX0	Set to 00b.

Note:
1. This table does not describe a procedure.

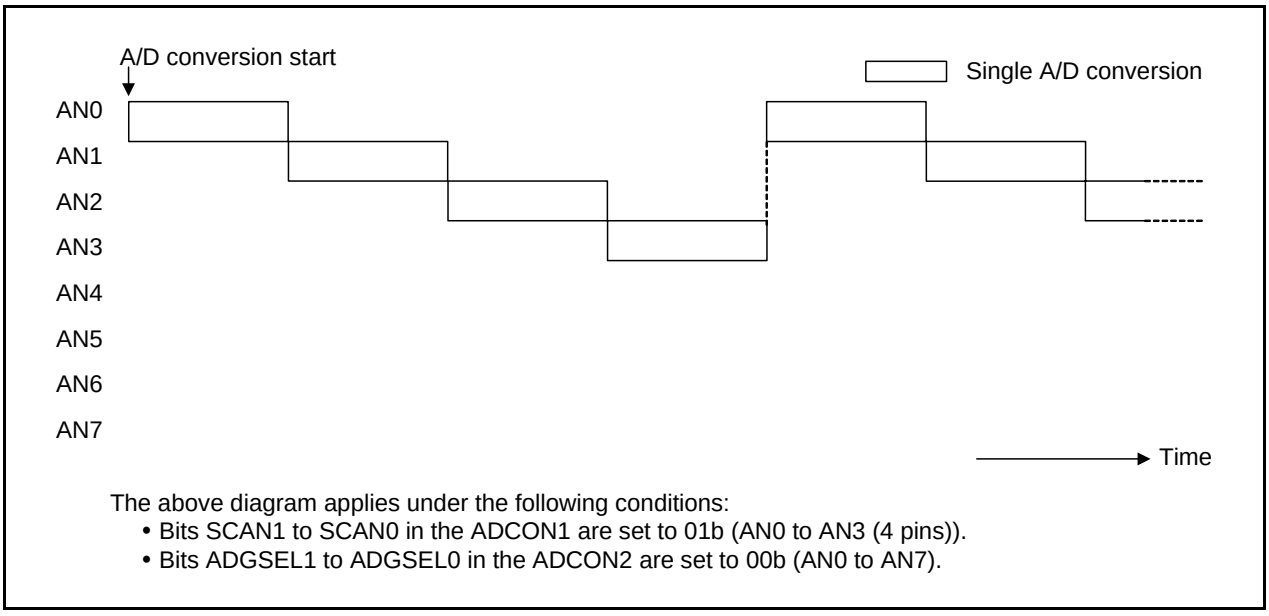


Figure 22.13 Operation Example in Repeat Sweep Mode 0

22.4.5 Repeat Sweep Mode 1

In repeat sweep mode 1, the analog voltage applied to eight selected pins including some prioritized pins is repeatedly converted to a digital code. Table 22.16 lists the Repeat Sweep Mode 1 Specifications.

Table 22.16 Repeat Sweep Mode 1 Specifications

Item	Specification
Function	The input voltage of all eight pins selected by bits ADGSEL1 to ADGSEL0 in the ADCON2 register is repeatedly converted to a digital code. One to four pins selected by SCAN1 to SCAN0 in the ADCON1 register is/are converted by priority. Example: If AN0 is prioritized, input voltage is converted to a digital code in the following order: AN0→AN1→AN0→AN2→AN0→AN3 •••
A/D conversion start conditions	• When the TRG bit in the ADCON0 register is 0 (software trigger), the ADST bit in the ADCON0 register is set to 1 (A/D conversion start). • When the TRG bit is 1 ($\overline{\text{ADTRG}}$ trigger), input level at the $\overline{\text{ADTRG}}$ pin changes from high to low after the ADST bit is set to 1 (A/D conversion start).
A/D conversion stop condition	Set the ADST bit to 0 (A/D conversion stop).
Interrupt request generation timing	No interrupt requests generated
Analog input pins to be given priority when A/D converted	Select from AN0 (1 pin), AN0 and AN1 (2 pins), AN0 to AN2 (3 pins), and AN0 to AN3 (4 pins). AN0_0 to AN0_3 can be selected in the same way.
Reading of A/D conversion result	Read the registers among AD0 to AD7 that corresponds to the selected pins.

Table 22.17 Registers and Settings in Repeat Sweep Mode 1 (1)

Register	Bit	Setting
PCLKSTP1	PCKSTP14	Set to 0 when using f1.
PCR	PCR3	Set to 1 (key input disabled) when using pins AN0 to AN3 for analog input.
	PCR7	Set to 1 (key input disabled) when using pins AN4 to AN7 for analog input.
AINRST	AINRST1, AINRST0	Select whether open-circuit detection assist function is used or not.
AD0 to AD7	b9 to b0	A/D conversion result can be read.
ADCON2	ADGSEL1, ADGSEL0	Select analog input pin group.
	CKS2	Select ϕAD frequency.
ADCON0	CH2 to CH0	Disabled
	MD1 to MD0	Set to 11b.
	TRG	Select a trigger.
	ADST	Set to 1 to start A/D conversion and set to 0 to stop it.
	CKS0	Select ϕAD frequency.
ADCON1	SCAN1, SCAN0	Select a pin to be given priority when A/D converted
	MD2	Set to 1.
	CKS1	Select ϕAD frequency.
	ADSTBY	Set to 1 in A/D conversion.
	ADEX1, ADEX0	Set to 00b.

Note:
1. This table does not describe a procedure.

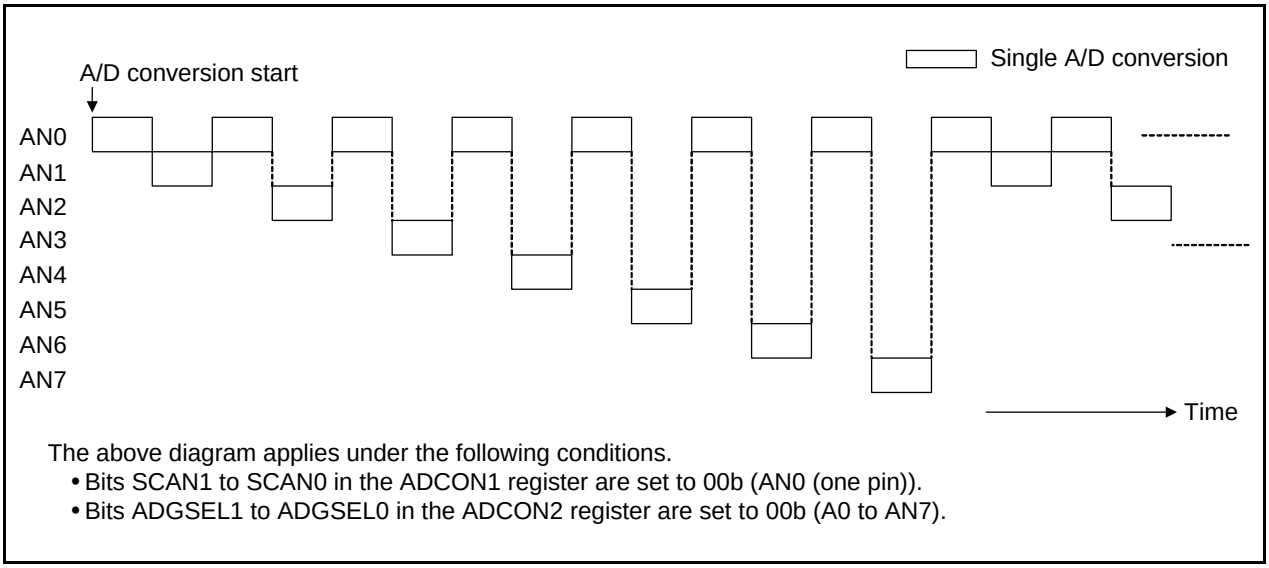


Figure 22.14 Operation Example in Repeat Sweep Mode 1

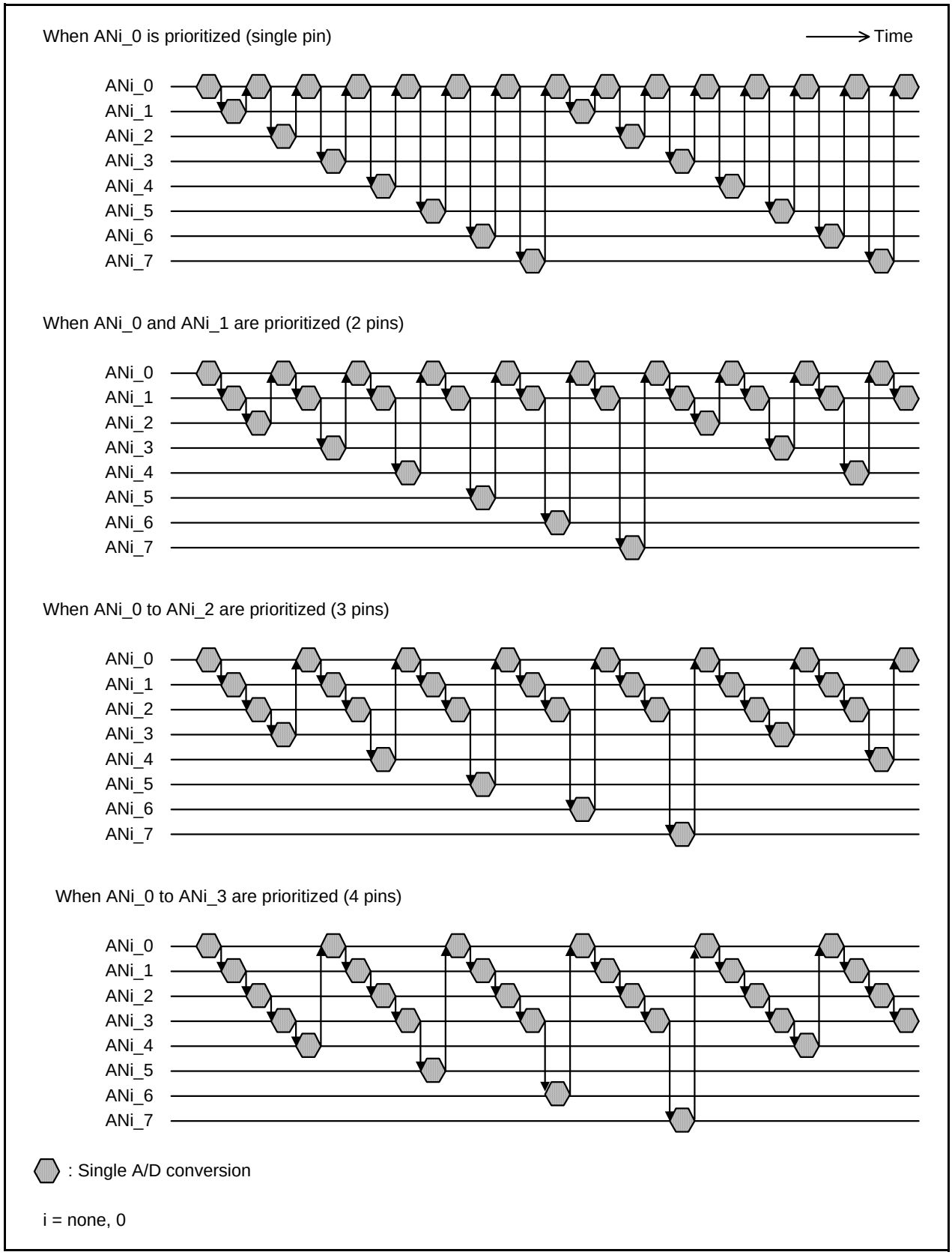


Figure 22.15 Transition Diagram of Pins Used in A/D Conversion in Repeat Sweep Mode 1

22.5 External Sensor

To perform A/D conversion accurately, charging the internal capacitor C shown in Figure 22.16 must be completed within a specified period of time.

T: Specified period of time (sampling time)
R0: Output impedance of sensor equivalent circuit
R: Internal resistance of the MCU
X: Precision (error) of the A/D converter
Y: Resolution of the A/D converter be Y (Y is 1024)

Generally, $VC = VIN\left\{1 - e^{-\frac{1}{C(R0+R)}t}\right\}$

When t = T, $VC = VIN - \frac{X}{Y}VIN = VIN\left(1 - \frac{X}{Y}\right)$

$$e^{-\frac{1}{C(R0+R)}T} = \frac{X}{Y}$$

$$-\frac{1}{C(R0+R)}T = \ln\frac{X}{Y}$$

Therefore, $R0 = -\frac{T}{C \bullet \ln\frac{X}{Y}} - R$

Figure 22.16 shows Analog Input Pin and External Sensor Equivalent Circuit. Impedance R0 by which voltage VC between pins of the capacitor C changes from 0 to VIN - (0.1/1024)VIN in time T when the difference between VIN and VC is 0.1LSB is obtained. (0.1/1024) means that A/D precision drop due to insufficient capacitor charge is kept to 0.1LSB in A/D conversion. Actual error however is the value of absolute accuracy added to 0.1LSB.
When ϕAD is 20 MHz, T is 0.75 μs. Output impedance R0 for charging capacitor C sufficiently within the time T is obtained as follows.

T = 0.75 μs, R = 10 kΩ, C = 6.0 pF, X = 0.1, and Y = 1024. Therefore,

$$R0 = -\frac{0.75 \times 10^{-6}}{6.0 \times 10^{-12} \bullet \ln\frac{0.1}{1024}} - 10 \times 10^{3} \approx 3.5 \times 10^{3}$$

Thus, the output impedance R0 of the sensor equivalent circuit, making the A/D converter precision (error) 0.1LSB or less, is up to 3.5 kΩ.

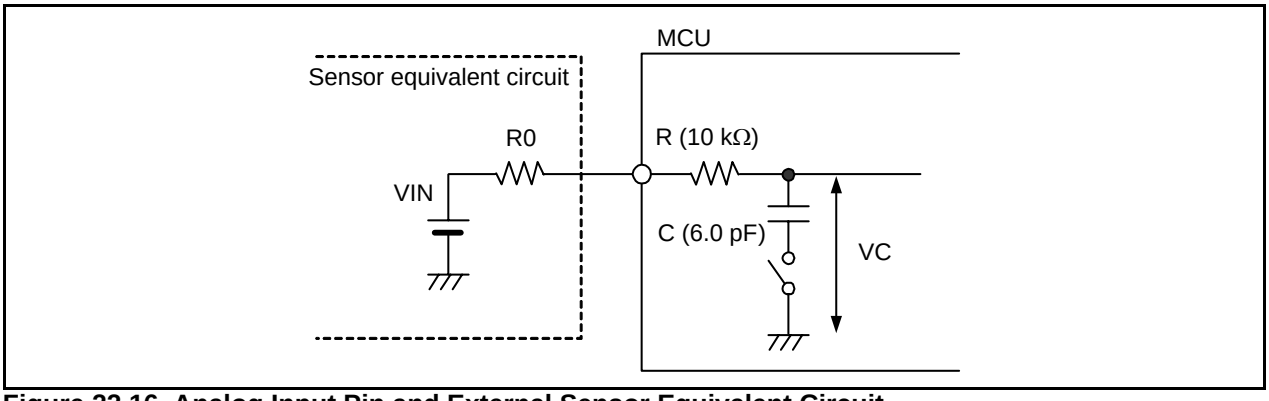


Figure 22.16 Analog Input Pin and External Sensor Equivalent Circuit

22.6 Interrupt

Refer to the operation examples for timing of generating interrupt requests.
Also, refer to 13.7 “Interrupt Control” for details. Table 22.18 lists Registers Associated with A/D Converter Interrupt.

Table 22.18 Registers Associated with A/D Converter Interrupt

Address	Register	Symbol	Reset Value
004Eh	A/D Conversion Interrupt Control Register	ADIC	XXXX X000b

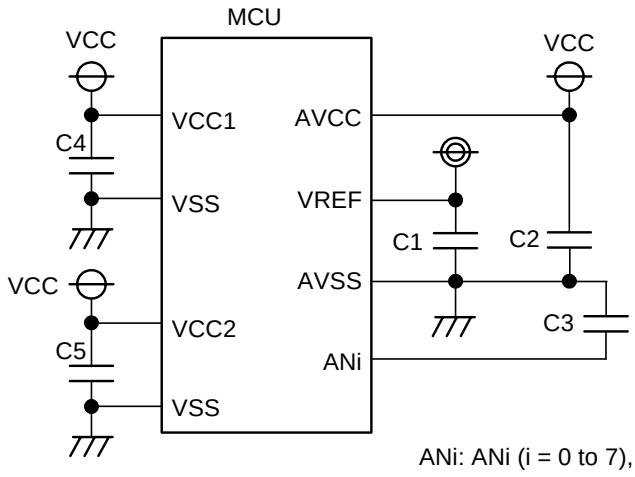
22.7 Notes on A/D Converter

22.7.1 Analog Input Pin

Do not use any of pins AN4 to AN7 as analog input pins if any one of pins $\overline{KI0}$ to $\overline{KI3}$ is used as a key input interrupt. Also, do not use any of four pins AN0 to AN3 as analog input pins if any one of pins $\overline{KI4}$ to $\overline{KI7}$ is used as a key input interrupt.

22.7.2 Pin Configuration

To prevent operation errors due to noise or latchup, and to reduce conversion errors, place capacitors between the AVSS pin and the AVCC pin, the VREF pin, and analog inputs (ANi (i = 0 to 7), ANEXi, AN0_i). Also, place a capacitor between the VCC1 pin and VSS pin.



Notes:
1. C1 ≥ 0.1 μF, C2 ≥ 0.1 μF, C3 ≥ 100 pF, C4 ≥ 0.1 μF, C5 ≥ 0.1 μF (reference values).
2. The traces for the capacitor and MCU should be short and wide as much as physically possible.

Figure 22.17 Example of Pin Configuration

22.7.3 Register Access

Set registers ADCON0 (excluding the ADST bit), ADCON1, and ADCON2 when A/D conversion stops (before a trigger is generated).
After A/D conversion stops, rewrite the ADSTBY bit in the ADCON1 register from 1 to 0.

22.7.4 A/D Conversion Start

When rewriting the ADSTBY bit in the ADCON1 register from 0 (A/D operation stopped) to 1 (A/D operation enabled), wait for one ϕAD cycle or more before starting A/D conversion.

22.7.5 A/D Operation Mode Change

When A/D operation mode has been changed, reselect analog input pins by using bits CH2 to CH0 in the ADCON0 register or bits SCAN1 to SCAN0 in the ADCON1 register.

22.7.6 State When Forcibly Terminated

If A/D conversion in progress is halted by setting the ADST bit in the ADCON0 register to 0, the conversion result is undefined. In addition, the non-converted ADi register may also become undefined. Do not use the ADi register when setting the ADST bit to 0 by a program during A/D conversion.

22.7.7 A/D Open-Circuit Detection Assist Function

The conversion result in open-circuit depends on the external circuit. Use this function only after careful evaluation of the system. When A/D conversion starts after changing the AINRST register, follow these procedures:

(1) Change bits AINRST1 to AINRST0 in the AINRST register.
(2) Wait for one cycle of ϕAD.
(3) Set the ADST bit in the ADCON0 register to 1 (A/D conversion started).

22.7.8 Detecting Completion of A/D Conversion

In one-shot mode and single sweep mode, use the IR bit in the ADIC register to detect completion of A/D conversion. When not using an interrupt, set the IR bit to 0 by a program after the detection. When 1 is written to the ADST bit in the ADCON0 register, the ADST bit becomes 1 (A/D conversion started) after start processing time elapses. (See Table 22.7 “Cycles of A/D Conversion Item”.) When reading the ADST bit shortly after writing 1, 0 (A/D conversion stop) may be read.

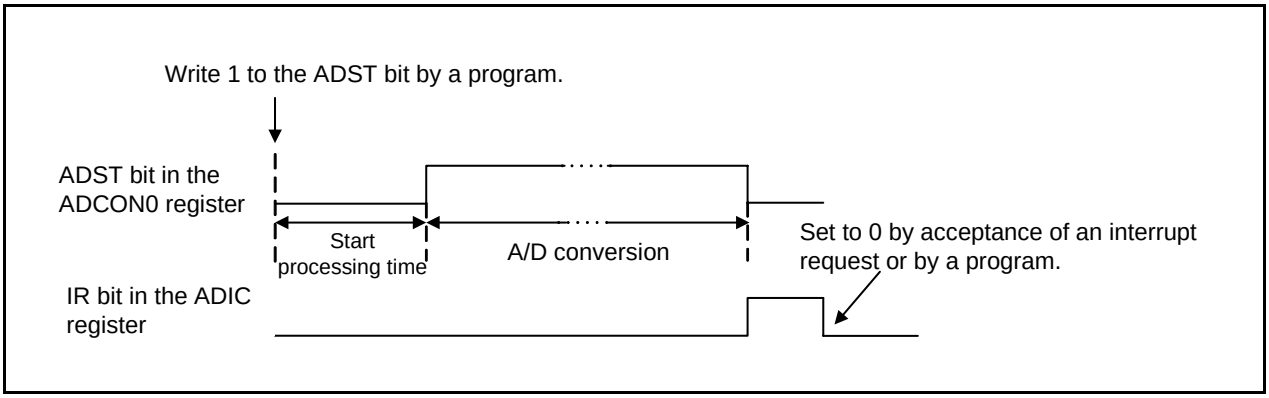


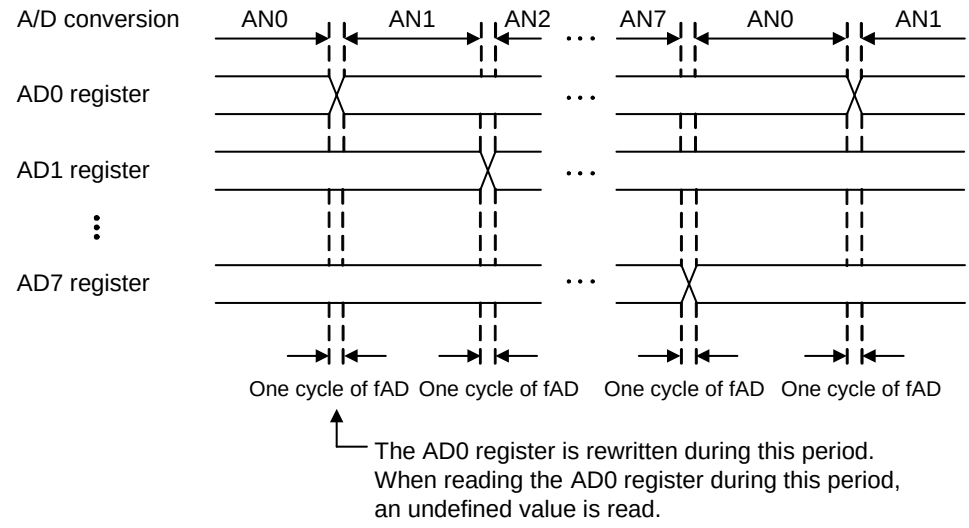
Figure 22.18 ADST Bit Operation

22.7.9 ϕAD

Divide fAD so ϕAD conforms the standard frequency.

22.7.10 Repeat Mode, and Repeat Sweep Mode 0, and Repeat Sweep Mode 1

In repeat mode, and repeat sweep mode 0, and repeat sweep mode 1, when reading the ADi register (i = 0 to 7) during the period when the ADi register value is rewritten, an undefined value may be read. Read the ADi register several times to determine whether the read value is valid. The period for reading an undefined value is one cycle of fAD.



The above diagram assumes the following:

- Bits MD1 to MD0 in the ADCON0 register are 11b, and the MD2 bit in the ADCON1 register is 0 (repeat sweep mode 0).
- Bits ADGSEL1 to ADGSEL0 in the ADCON2 register are 00b (AN0 to AN7 groups selected).
- Bits SCAN1 to SCAN0 in the ADCON1 register are 11b (AN0 to AN7 (8 pins) selected).

Figure 22.19 Period When the ADi Register Value is Rewritten

23. CRC Calculator

23.1 Introduction

The cyclic redundancy check (CRC) calculator detects errors in data blocks. This CRC calculator is enhanced by an additional feature, the CRC snoop, in order to monitor reads from and writes to a certain SFR address, and perform CRC calculations automatically on the data read from and data written to the aforementioned SFR address.

Table 23.1 CRC Calculator Specifications

Item	Specification
Generator polynomial	CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) or CRC-16 ($X^{16} + X^{15} + X^2 + 1$)
Selectable functions	• MSB/LSB selectable • CRC snoop

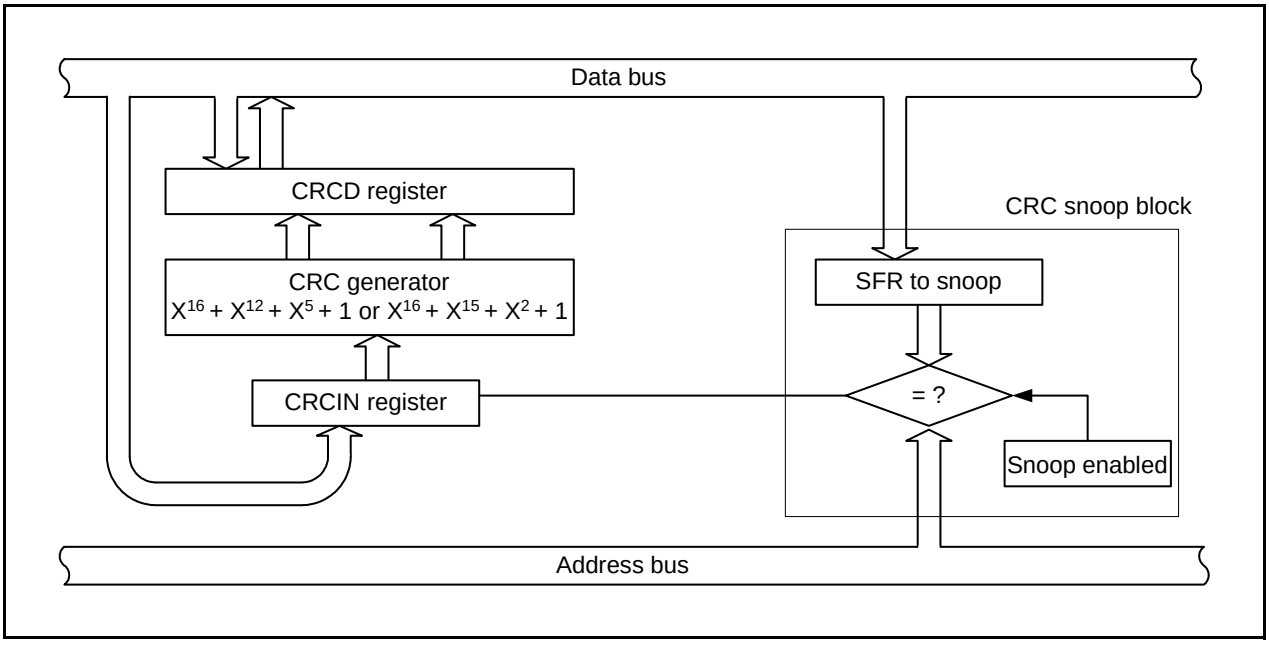


Figure 23.1 CRC Calculator Block Diagram

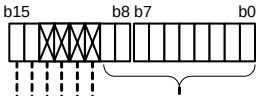
23.2 Registers

Table 23.2 Registers

Address	Register	Symbol	Reset Value
03B4h	SFR Snoop Address Register	CRCSAR	XXXX XXXXb
03B5h			00XX XXXXb
03B6h	CRC Mode Register	CRCMR	0XXX XXX0b
03BCh	CRC Data Register	CRCD	XXh
03BDh			XXh
03BEh	CRC Input Register	CRCIN	XXh

23.2.1 SFR Snoop Address Register (CRCSAR)

SFR Snoop Address Register



Symbol: CRCSAR
Address: 03B5h to 03B4h
After Reset: 00XX XXXX XXXX XXXXb

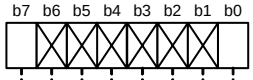
Bit Symbol	Bit Name	Function	RW
CRCSAR9 to CRCSAR0	SFR snoop address bit	Set the SFR address to snoop	RW
— (b13-b10)	No register bits. If necessary, set to 0. Read as undefined value.		—
CRCSR	Snoop-on-read enable bit	0: Disabled 1: Enabled	RW
CRCSW	Snoop-on-write enable bit	0: Disabled 1: Enabled	RW

CRCSR (Snoop-on-read enable bit) (b14)
CRCSW (Snoop-on-write enable bit) (b15)

Do not set bits CRCSR and CRCSW to 1 at the same time. Set the CRCSR bit to 0 when the CRCSW bit is 1. Set the CRCSW bit to 0 when the CRCSR bit is 1.

23.2.2 CRC Mode Register (CRCMR)

CRC Mode Register

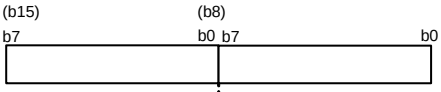


Symbol	Address	After Reset
CRCMR	03B6h	0XXX XXX0b

Bit Symbol	Bit Name	Function	RW
CRCPS	CRC polynomial select bit	0: $X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT) 1: $X^{16} + X^{15} + X^2 + 1$ (CRC-16)	RW
— (b6-b1)	No register bits. If necessary, set to 0. Read as undefined value.		—
CRCMS	CRC mode select bit	0: LSB first 1: MSB first	RW

23.2.3 CRC Data Register (CRCD)

CRC Data Register



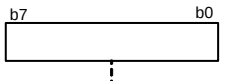
Symbol	Address	After Reset
CRCD	03BDh to 03BCh	Undefined

Function	Setting Range	RW
When data is written to the CRCIN register after setting the initial value in the CRCD register, the CRC can be read out from this register.	0000h to FFFFh	RW

Write 0000h to the CRCD register and then write the first data to the CRCIN register.
Execute this operation every time CRC calculation is performed. Refer to the setting procedures described in Figure 23.2 “CRC Calculation When Using CRC-CCITT” and Figure 23.3 “CRC Calculation When Using CRC-16”.

23.2.4 CRC Input Register (CRCIN)

CRC Input Register



Symbol	Address	After Reset
CRCIN	03BEh	Undefined

Function	Setting Range	RW
Data input	00h to FFh	RW

23.3 Operations

23.3.1 Basic Operation

The CRC (Cyclic Redundancy Check) calculator detects errors in data blocks. The MCU uses two generator polynomials of CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) and CRC-16 ($X^{16} + X^{15} + X^2 + 1$) to generate CRC.

The CRC is 16-bit code generated for a given length of a data block in 8-bit units. After setting the default value in the CRCD register, the CRC is stored in the CRCD register every time 1-byte of data is written to the CRCIN register. CRC generation for 1-byte data is completed in two CPU clock cycles.

23.3.2 CRC Snoop

The CRC snoop monitors reads from and writes to a certain SFR address and performs CRC calculation on the data read from and written to the aforementioned SFR address automatically. Because the CRC snoop recognizes writes to and reads from a certain SFR address as a trigger to automatically perform CRC calculation, there is no need to write data to the CRCIN register. All SFR addresses from 0020h to 03FFh are subject to the CRC snoop. The CRC snoop is useful in monitoring writes to the UART TX buffer, and reads from the UART RX buffer.

To use this function, write a target SFR address to bits CRCSAR9 to CRCSAR0 in the CRCSAR register. Then, set the CRCSW bit in the CRCSAR register to 1 to enable snooping on writes to the target, or set the CRCSR bit in the CRCSAR register to 1 to enable snooping on reads from the target.

When setting the CRCSW bit to 1, and writing data to a target SFR address by CPU or DMA, the CRC calculator stores the data in the CRCIN register and performs CRC calculation. Similarly, when setting the CRCSR bit to 1, and reading data in a target SFR address by CPU or DMA, the CRC calculator stores the data in the CRCIN register and performs CRC calculation.

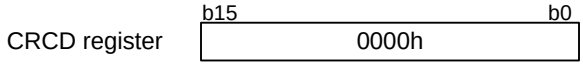
CRC calculation is performed 1-byte at a time. When the target SFR address is accessed in words (16 bits), CRC is generated on the lower byte (1 byte) of data.

When using CRC-CCITT with LSB first:

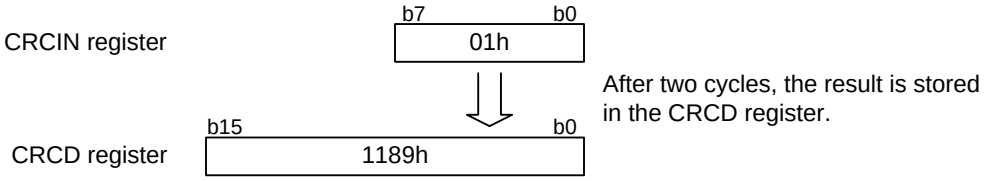
Generator polynomial: $X^{16} + X^{12} + X^5 + 1$ (1 0001 0000 0010 0001b)

Setting procedures

(1) Write 0000h (initial value) to the CRCD register.



(2) Write 01h to the CRCIN register.



(3) Write 23h to the CRCIN register.

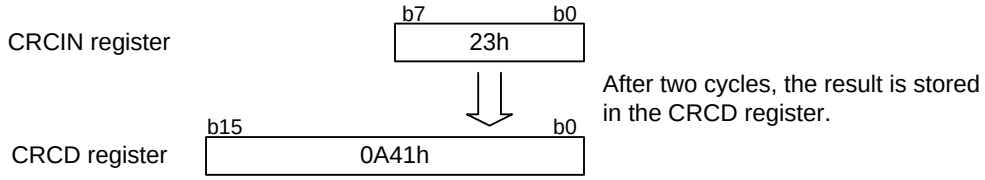


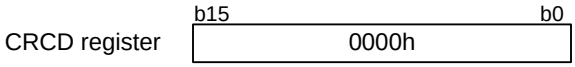
Figure 23.2 CRC Calculation When Using CRC-CCITT

When using CRC-16 with MSB first

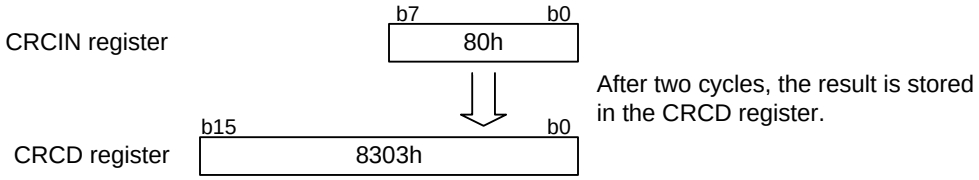
Generator polynomial: $X^{16} + X^{15} + X^2 + 1$ (1 1000 0000 0000 0101b)

Setting procedures

(1) Write 0000h (initial value) to the CRCD register.



(2) Write 80h to the CRCIN register.



(3) Write C4h to the CRCIN register.

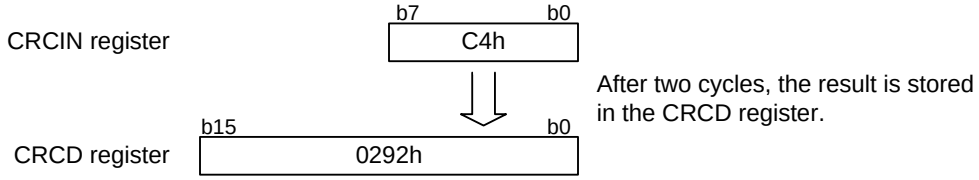


Figure 23.3 CRC Calculation When Using CRC-16

24. Flash Memory

24.1 Introduction

This product uses flash memory as ROM. In this chapter, flash memory refers to the flash memory inside the MCU.
In this product, the flash memory can perform in three rewrite modes: CPU rewrite mode, standard serial I/O mode, and parallel I/O mode.
Table 24.1 lists Flash Memory Specifications (see Tables 1.1 and 1.2 “Specifications” for the items not listed in Table 24.1).

Table 24.1 Flash Memory Specifications

Item		Specification
Flash memory rewrite mode		3 modes (CPU rewrite, standard serial I/O, and parallel I/O)
Erase block	Program ROM 1	See Figure 24.1 “Flash Memory Block Diagram”.
	Program ROM 2	1 block (16 KB)
	Data flash	2 blocks (4 KB each)
Program method		In 2-word (4-byte) units
Erase method		Block erase
Program and erase control method		Program and erase controlled by software commands
Suspend function		Program suspend and erase suspend
Protect method		A lock bit protects each block.
Number of commands		8
Program and erase cycles	Program ROM 1 and program ROM 2	1,000 times (1)
	Data flash	10,000 times (1)
Data retention		20 years
Flash memory rewrite disable function		Parallel I/O mode ROM code protect function Standard serial I/O mode ID code check function, forced erase function, and standard serial I/O mode disable function
User boot function		User boot mode

Note:
1. Definition of program and erase cycles:
The program and erase cycles is the number of erase operations performed on a per-block basis. For example, assume that a 4-KB block is programmed in 1,024 operations, writing two words at a time, and erased thereafter. In this case, the block is considered to have been programmed and erased once.
If the program and erase cycles are 1,000 times, each block can be erased up to 1,000 times.

Table 24.2 Flash Memory Rewrite Modes Overview

Flash Memory Rewrite Mode	CPU Rewrite Mode	Standard Serial I/O Mode	Parallel I/O Mode
Function	The flash memory is rewritten when the CPU executes software commands. EW0 mode: Rewritable in areas other than the flash memory EW1 mode: Rewritable in the flash memory	The flash memory is rewritten using a dedicated serial programmer. Standard serial I/O mode 1: Clock synchronous serial I/O Standard serial I/O mode 2: 2-wire clock asynchronous serial I/O	The flash memory is rewritten using a dedicated parallel programmer.
Areas which can be rewritten	Program ROM 1, program ROM 2, and data flash	Program ROM 1, program ROM 2, and data flash	Program ROM 1, program ROM 2, and data flash
ROM programmer	None	Serial programmer	Parallel programmer
On-board rewrite	Available	Available	Unavailable

24.2 Memory Map

The flash memory is used as ROM in this product. The flash memory is comprised of program ROM 1, program ROM 2, and data flash. Figure 24.1 shows the Flash Memory Block Diagram.

The flash memory is divided into several blocks, each of which can be protected (locked) from programming or erasing. The flash memory can be rewritten in CPU rewrite, standard serial I/O, and parallel I/O modes.

Program ROM 2 can be used when the PRG2C0 bit in the PRG2C register is 0 (program ROM 2 enabled).

Data flash can be used when the PM10 bit in the PM1 register is set to 1 (0E000h to 0FFFFh: data flash). Data flash is divided into block A and block B.

Table 24.3 lists the differences among program ROM 1, program ROM 2, and data flash.

Program can be allocated in program ROM 1, program ROM 2, or data flash.

Table 24.3 Program ROM 1, Program ROM 2, and Data Flash

Item	Flash Memory		
	Program ROM 1	Program ROM 2	Data Flash
Program and erase cycles	1,000 times		10,000 times
Forced erase function	Enabled		Disabled
Frequency limit when reading	No		Yes
User boot program	Do not allocate	Allocatable	Do not allocate

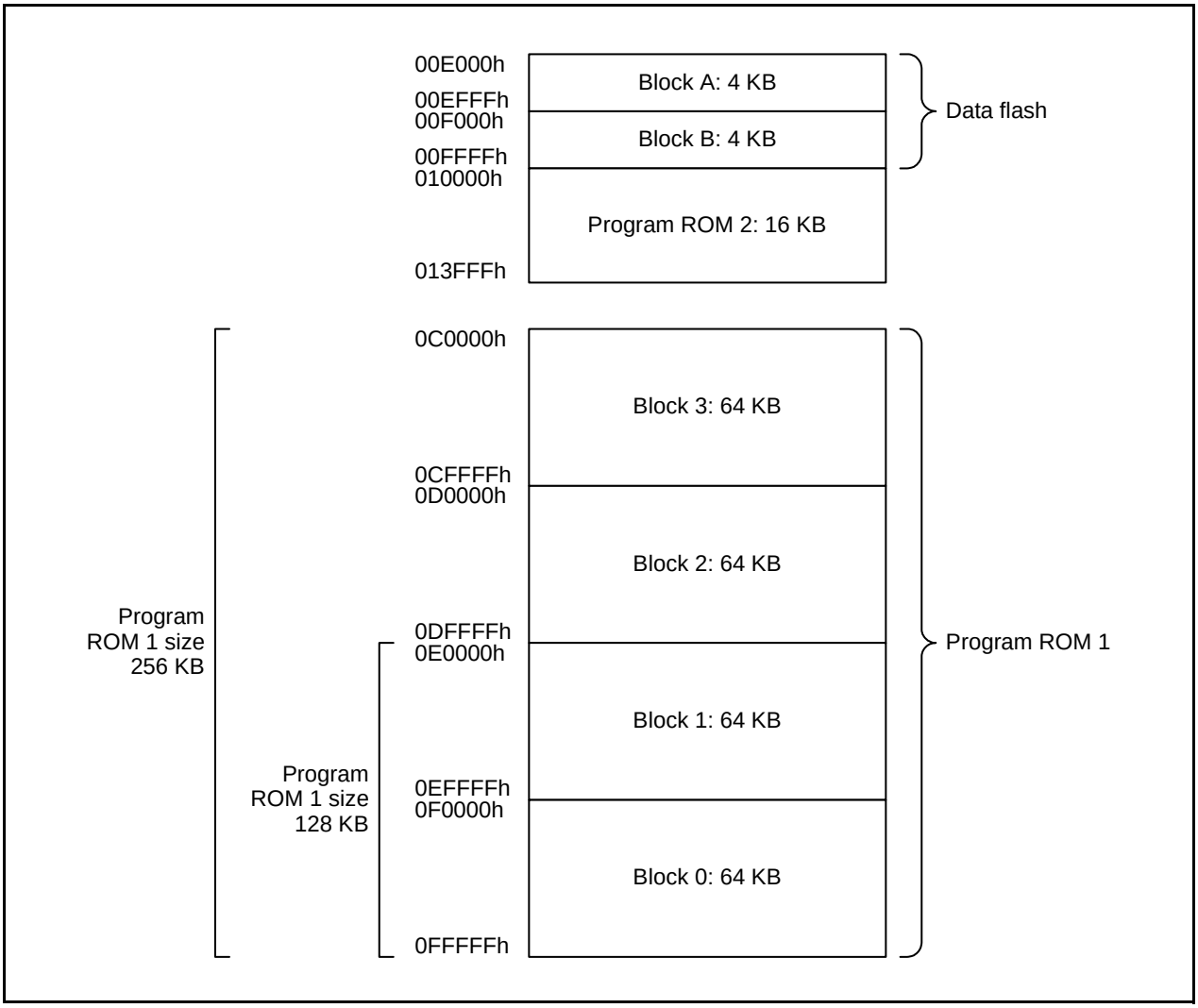


Figure 24.1 Flash Memory Block Diagram

24.3 Registers

Table 24.4 Registers

Address	Register	Symbol	Reset Value
0220h	Flash Memory Control Register 0	FMR0	0000 0001b (Other than user boot mode) 0010 0001b (User boot mode)
0221h	Flash Memory Control Register 1	FMR1	00X0 XX0Xb
0222h	Flash Memory Control Register 2	FMR2	XXXX 0000b
0223h	Flash Memory Control Register 3	FMR3	XXXX 0000b
0230h	Flash Memory Control Register 6	FMR6	XX0X XX00b
0231h	Flash Memory Control Register 7	FMR7	1000 0000b

24.3.1 Flash Memory Control Register 0 (FMR0)

Flash Memory Control Register 0

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset
			0					FMR0	0220h	0000 0001b (other than user boot mode) 0010 0001b (user boot mode)

Bit Symbol	Bit Name	Function	RW
FMR00	RY/$\overline{BY}$ status flag	0: Busy (being written or erased) 1: Ready	RO
FMR01	CPU rewrite mode select bit	0: CPU rewrite mode disabled 1: CPU rewrite mode enabled	RW
FMR02	Lock bit disable select bit	0: Lock bit enabled 1: Lock bit disabled	RW
FMSTP	Flash memory stop bit	0: Flash memory operation enabled 1: Flash memory operation stopped (low power-mode, flash memory initialized)	RW
— (b4)	Reserved bit	Set to 0.	RW
— (b5)	Reserved bit	Set to 0 in other than user boot mode. Set to 1 in user boot mode.	RW
FMR06	Program status flag	0: Completed as expected 1: Completed in error	RO
— (b7)	Reserved bit	Read as undefined value.	RO

FMR00 (RY/$\overline{BY}$ status flag) (b0)

This bit indicates the flash memory operating state.

Conditions to become 0:

During the following commands execution:

Program, block erase, lock bit program, read lock bit status, and block blank check

- When the flash memory stops (the FMSTP bit is 1)
- During the wake up operation when the FMSTP bit is changed from 1 to 0

Condition to become 1:

Other than those above.

FMR01 (CPU rewrite mode select bit) (b1)

Commands can be accepted by setting the FMR01 bit to 1 (CPU rewrite mode enabled).
To set the FMR01 bit to 1, write 0 and then 1 in succession. Make sure no interrupts or DMA transfers will occur before writing 1 after writing 0.
Change FMR01 bit when the PM24 bit in the PM2 register is 0 ($\overline{\text{NMI}}$ interrupt disabled) or high is input to the $\overline{\text{NMI}}$ pin.
While in EW0 mode, write to this bit from a program in an area other than flash memory.
Enter read array mode, and then set this bit to 0.

FMR02 (Lock bit disable select bit) (b2)

The lock bit is disabled by setting the FMR02 bit to 1 (lock bit disabled) (Refer to 24.8.2 “Data Protect Function”).
The FMR02 bit does not change the lock bit data, but disables the lock bit function. If an erase command is executed when the FMR02 bit is set to 1, the lock bit data status changes from 0 (locked) to 1 (unlocked) after command execution is completed.
The FMR02 bit is enabled when the FMR01 bit is 1 (CPU rewrite mode enabled). When setting the FMR02 bit to either 1 or 0, change this bit when the FMR01 bit is 1.
To set the FMR02 bit to 1, write 0 and then 1 to the FMR02 bit in succession when the FMR01 bit is 1. Make sure no interrupts or DMA transfers will occur before writing 1 after writing 0.
Do not change the FMR02 bit while programming, erasing, or suspending.

FMSTP (Flash memory stop bit) (b3)

The FMSTP bit resets the flash memory control circuits and minimizes current consumption in the flash memory. Access to the internal flash memory is disabled when the FMSTP bit is set to 1 (flash memory operation stopped). Set the FMSTP bit by a program located in an area other than the flash memory.
Set the FMSTP bit to 1 under the following condition:

- A flash memory access error occurs while erasing or programming in EW0 mode (the FMR00 bit does not revert to 1 (ready)).

After the FMSTP bit is set to 0 (flash memory operation enabled), wait until the flash memory circuit stabilizes (tps), then perform the next operation.
Also, when the FMSTP bit is set to 0 immediately after this bit is set to 1, wait for tps after the bit is set to 1. The procedure for this case is described below.

(1) Set the FMSTP bit to 1.
(2) Wait until the flash memory circuit stabilizes (tps).
(3) Set the FMSTP bit to 0.
(4) Wait for tps.

The FMSTP bit is enabled when the FMR01 bit is 1 (CPU rewrite mode). When the FMR01 bit is 0, although the FMSTP bit can be set to 1 by writing 1, the flash memory is neither placed in low-power mode nor initialized.
When the FMR22 bit is 1 (slow read mode enabled) or the FMR23 bit is 1 (low-current consumption read mode enabled), do not set the FMSTP bit in the FMR0 register to 1 (flash memory operation stopped). Also, when the FMSTP bit is 1, do not set the FMR23 bit to 1.

FMR06 (Program status flag) (b6)

This bit indicates the auto-program operation state.
Condition to become 0:

- Execute the clear status command.

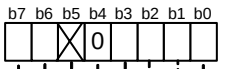
Condition to become 1:

- Refer to 24.8.6.1 “Full Status Check”.

Do not execute the following commands when the FMR06 bit is 1:
Program, block erase, lock bit program, and block blank check.

24.3.2 Flash Memory Control Register 1 (FMR1)

Flash Memory Control Register 1



Symbol: FMR1 | Address: 0221h | After Reset: 00X0 XX0Xb

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved bit	Read as undefined value.	RO
FMR11	Write to FMR6 register enable bit	0: Disabled 1: Enabled	RW
— (b3-b2)	Reserved bits	Read as undefined value.	RO
FMR14	CPU rewrite unit select bit	0: Word 1: Byte	RW
— (b5)	No register bit. If necessary, set to 0. Read as undefined value.		—
FMR16	Lock bit status flag	0: Lock 1: Unlock	RO
FMR17	Data flash wait bit	0: 1 wait 1: Follow the setting of the PM17 bit in the PM1 register	RW

FMR11 (Write to FMR6 register enable bit) (b1)

Change FMR11 bit when the PM24 bit in the PM2 register is 0 ($\overline{\text{NMI}}$ interrupt disabled) or high is input to the $\overline{\text{NMI}}$ pin.

FMR14 (CPU rewrite unit select bit) (b4)

This bit is used to select the CPU rewrite unit. When using the program command in EW1 mode, set this bit to 1 (byte). To set the FMR14 bit to 1, write 0 and then 1 to the FMR14 bit in succession while the FMR01 bit is 1 (CPU rewrite mode enabled). Make sure no interrupts or DMA transfers will occur before writing 1 after writing 0.

FMR16 (Lock bit status flag) (b6)

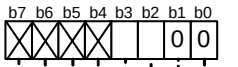
This bit indicates the execution result of the read lock bit status command.

FMR17 (Data flash wait bit) (b7)

This bit is used to select the number of wait states for data flash.
When setting this bit to 0, one wait is inserted to the read cycle of the data flash. The write cycle is not affected.

24.3.3 Flash Memory Control Register 2 (FMR2)

Flash Memory Control Register 2



Symbol: FMR2 | Address: 0222h | After Reset: XXXX 0000b

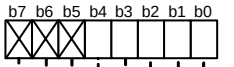
Bit Symbol	Bit Name	Function	RW
— (b1-b0)	Reserved bits	Set to 0.	RW
FMR22	Slow read mode enable bit	0: Disabled 1: Enabled	RW
FMR23	Low current consumption read mode enable bit	0: Disabled 1: Enabled	RW
— (b7-b4)	No register bits. If necessary, set to 0. When read, the read value is undefined.		—

FMR22 (Slow read mode enable bit) (b2)
FMR23 (Low-current consumption read mode enable bit) (b3)

Refer to 9.4 “Power Control in Flash Memory”.

24.3.4 Flash Memory Control Register 3 (FMR3)

Flash Memory Control Register 3



Symbol	Address	After Reset
FMR3	0223h	XXXX 0000b

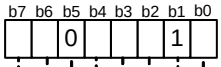
Bit Symbol	Bit Name	Function	RW
FMR30	Suspend function enable bit	0: Disabled 1: Enabled	RW
FMR31	Suspend request bit	0: Command restart 1: Suspend request	RW
FMR32	Erase suspend status flag	0: Erase suspend not accepted 1: Erase suspend accepted	RO
FMR33	Program suspend status flag	0: Program suspend not accepted 1: Program suspend accepted	RO
— (b4)	Reserved bit	Read as undefined value.	RO
— (b7-b5)	No register bits. If necessary, set to 0. Read as undefined value.		—

FMR30 (Suspend function enable bit) (b0)

To set the FMR30 bit to 1, write 0 and then 1 in succession. Make sure no interrupts or DMA transfers will occur before writing 1 after writing 0.

24.3.5 Flash Memory Control Register 6 (FMR6)

Flash Memory Control Register 6



Symbol	Address	After Reset
FMR6	0230h	XX0X XX00b

Bit Symbol	Bit Name	Function	RW
FMR60	EW1 mode select bit	0: EW0 mode 1: EW1 mode	RW
FMR61	Reserved bit	Set to 1.	RW
— (b4-b2)	Reserved bits	Read as undefined value.	RO
— (b5)	Reserved bit	Set to 0.	RW
— (b7-b6)	Reserved bits	Read as undefined value.	RO

When accessing the FMR6 register, set a CPU clock frequency of 10 MHz or less by the CM06 bit in the CM0 register and bits CM17 and CM16 in the CM1 register. Also, set the PM17 bit in the PM1 register to 1 (wait state).

FMR60 (EW1 mode select bit) (b0)

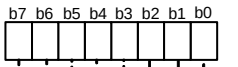
To set the FMR60 bit to 1, write 1 when both FMR01 bit in the FMR0 register and FMR11 bit in the FMR1 register are 1.
Change FMR60 bit when the PM24 bit in the PM2 register is 0 ($\overline{\text{NMI}}$ interrupt disabled) or high is input to the $\overline{\text{NMI}}$ pin. Also, change this bit when the FMR00 bit in the FMR0 register is 1 (ready).

FMR61 (b1)

Set the FMR61 bit to 1 when using CPU rewrite mode.

24.3.6 Flash Memory Control Register 7 (FMR7)

Flash Memory Control Register 7



Symbol	Address	After Reset
FMR7	0231h	1000 0000b

Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved bit	Read as undefined value.	RO
— (b1)	Reserved bit	Read as undefined value.	RO
— (b2)	Reserved bit	Read as undefined value.	RO
— (b3)	Reserved bit	Read as undefined value.	RO
— (b4)	Reserved bits	Read as undefined value.	RO
FMR75	Erase status flag	0: Completed as expected 1: Completed in error	RO
— (b6)	Reserved bit	Read as undefined value.	RO
— (b7)	Reserved bit	Read as undefined value.	RO

FMR75 (Erase status flag) (b5)

This bit indicates the auto-erase operation state.

Condition to become 0:

- Execute the clear status command

Condition to become 1:

- Refer to 24.8.6.1 “Full Status Check”.

Do not execute the following commands when the FMR75 bit is 1:

Program, block erase, lock bit program, and block blank check.

24.4 Optional Function Select Area

In an option function select area, the MCU state after reset and the function to prevent rewrite in parallel I/O mode are selected.

The option function select area is not an SFR, and therefore cannot be rewritten by a program. Set an appropriate value when writing a program to the flash memory. The entire option function select area becomes FFh when the block including the option function select area is erased.

In blank products, the OFS1 address value is FFh when shipped. After a value is written by the user, this register takes on the written value.

In programmed products, the OFS1 address is the value set in the user program prior to shipping.

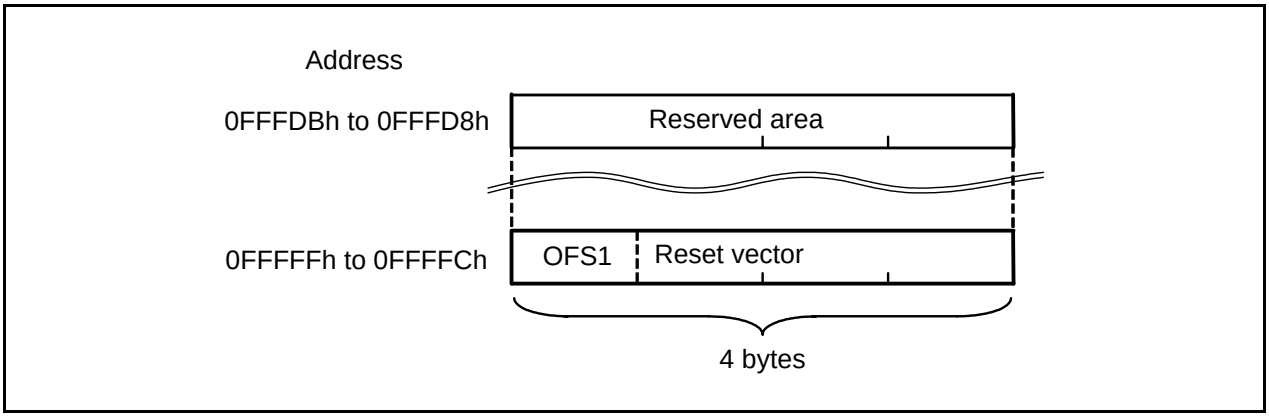
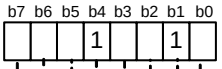


Figure 24.2 Option Function Select Area

24.4.1 Optional Function Select Address 1 (OFS1)

Optional Function Select Address 1



Symbol: OFS1

Address: FFFFFh

Bit Symbol	Bit Name	Function
WDTON	Watchdog timer start select bit	0: Watchdog timer starts automatically after reset 1: Watchdog timer is stopped after reset
— (b1)	Reserved bit	Set to 1.
ROMCR	ROM code protect cancel bit	0: ROM code protection cancelled 1: ROMCP1 bit enabled
ROMCP1	ROM code protect bit	0: ROM code protection enabled 1: ROM code protection disabled
— (b4)	Reserved bit	Set to 1.
VDSEL1	Vdet0 select bit 1	0: Vdet0_2 1: Vdet0_0
LVDAS	Voltage detector 0 start bit	0: Voltage monitor 0 reset enabled after hardware reset 1: Voltage monitor 0 reset disabled after hardware reset
CSPROINI	After-reset count source protection mode select bit	0: Count source protection mode enabled after reset 1: Count source protection mode disabled after reset

ROMCR (ROM code protect disable bit) (b2)
ROMCP1 (ROM code protect bit) (b3)

These bits are used to prohibit the flash memory from being read or rewritten during parallel I/O mode.

Table 24.5 ROM Code Protect

Bit Setting		ROM Code Protect
ROMCR bit	ROMCP1 bit	
0	0	Disabled
0	1	
1	0	Enabled
1	1	Disabled

24.5 Flash Memory Rewrite Disable Function

This function prohibits the flash memory from being read, written, and erased. The details are shown for each mode.

Parallel I/O mode

ROM code protect function

Standard serial I/O mode

ID code check function, forced erase function, and standard serial I/O mode disable function

24.6 Boot Mode

A hardware reset occurs while a high-level signal is applied to pins CNVSS. After reset, the MCU enters boot mode. In boot mode, user boot mode or standard serial I/O mode is selected in accordance with the content of a user boot code area. Refer to 24.9 “Standard Serial I/O Mode” for details on standard serial I/O mode.

The MCU does not enter boot mode after voltage monitor 0 reset.

24.7 User Boot Mode

This mode is used for starting the flash memory rewrite program programmed by a user.

Allocate the flash memory rewrite program to program ROM 2. In user boot mode, the program is executed from address 10000h (starting address of program ROM 2). After starting the program, the flash memory is rewritten according to the program in EW0 or EW1 mode.

24.7.1 User Boot Function

User boot mode can be selected by the status of a port when the MCU starts in boot mode. Table 24.6 lists the User Boot Function Specifications.

Table 24.6 User Boot Function Specifications

Item	Specification
Entry pin	None or select a port from P0 to P10
User boot start level	Select high or low
User boot start address	Address 10000h (program ROM 2 start address)

Set “UserBoot” in ASCII code to addresses 13FF0h to 13FF7h in the user boot code area, select a port for entry from addresses 13FF8h to 13FF9h and 13FFAh, and select the start level with address 13FFBh. After starting boot mode, user boot mode or standard serial I/O mode is selected in accordance with the level of the selected port.

In addition, if addresses 13FF0h to 13FF7h are set to “UserBoot” in ASCII code and addresses 13FF8h to 13FFBh are set to 00h, user boot mode is selected.

In user boot mode, the program of address 10000h (program ROM 2 start address) is executed.

Figure 24.3 shows the User Boot Code Area, Table 24.7 lists Start Mode (When Port Pi_j is Selected for Entry), Table 24.8 lists “UserBoot” in ASCII Code, and Table 24.9 lists Addresses of Selectable Ports for Entry.

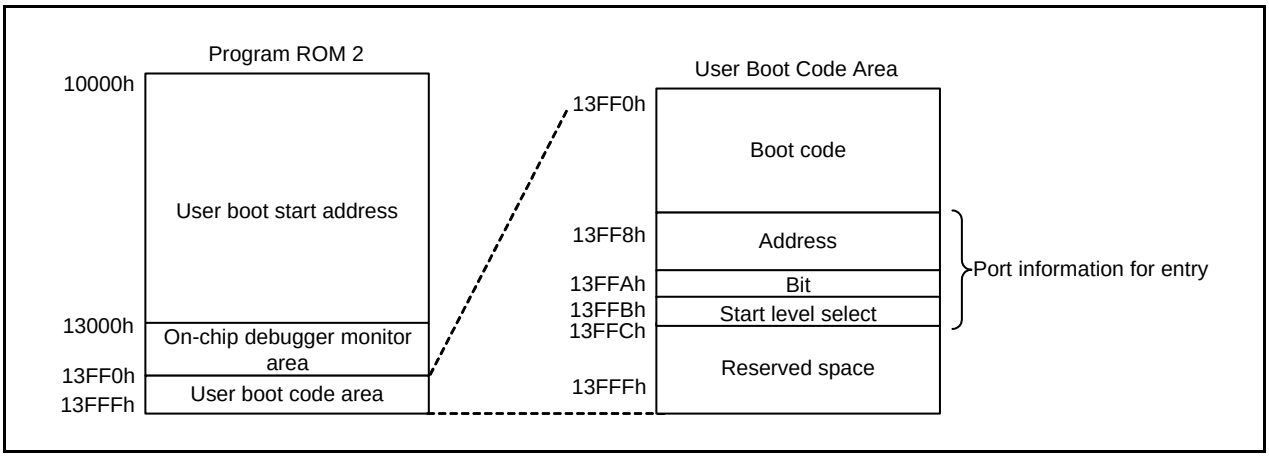


Figure 24.3 User Boot Code Area

Table 24.7 Start Mode (When Port Pi_j is Selected for Entry) (1)

Boot Code (13FF0h to 13FF7h)	Port Information for Entry			Port Pi_j Input Level	Start Mode
	Address (13FF8h to 13FF9h)	Bit (13FFAh)	Start level select (13FFBh)		
"UserBoot" in ASCII code (2)	0000h	00h	00h	—	User boot mode
	Pi register address (3)	00h to 07h (value of j)	00h	High	Standard serial I/O mode
				Low	User boot mode
	Pi register address (3)	00h to 07h (value of j)	01h	High	User boot mode
				Low	Standard serial I/O mode
Other than "UserBoot" in ASCII code	—	—	—	—	Standard serial I/O mode

i = 0, 1, 6 to 10, j = 0 to 7

Notes:
1. Only use the values listed in Table 24.7.
2. See Table 24.8 "UserBoot" in ASCII Code.
3. See Table 24.9 "Addresses of Selectable Ports for Entry".

Table 24.8 "UserBoot" in ASCII Code

Address	ASCII Code
13FF0h	55h (upper-case U)
13FF1h	73h (lower-case s)
13FF2h	65h (lower-case e)
13FF3h	72h (lower-case r)
13FF4h	42h (upper-case B)
13FF5h	6Fh (lower-case o)
13FF6h	6Fh (lower-case o)
13FF7h	74h (lower-case t)

Table 24.9 Addresses of Selectable Ports for Entry

Port	Address	
	13FF9h	13FF98h
P0	03h	E0h
P1	03h	E1h
P6	03h	ECh
P7	03h	EDh
P8	03h	F0h
P9	03h	F1h
P10	03h	F4h

Table 24.10 Example Settings of User Boot Code Area

When starting up in user boot mode while input level of the port P1_5 is low:

Address	Setting Value	Meaning
13FF0h	55h	Upper-case U
13FF1h	73h	Lower-case s
13FF2h	65h	Lower-case e
13FF3h	72h	Lower-case r
13FF4h	42h	Upper-case B
13FF5h	6Fh	Lower-case o
13FF6h	6Fh	Lower-case o
13FF7h	74h	Lower-case t
13FF8h	E1h	Port P1_5
13FF9h	03h	
13FFAh	05h	
13FFBh	00h	Low level

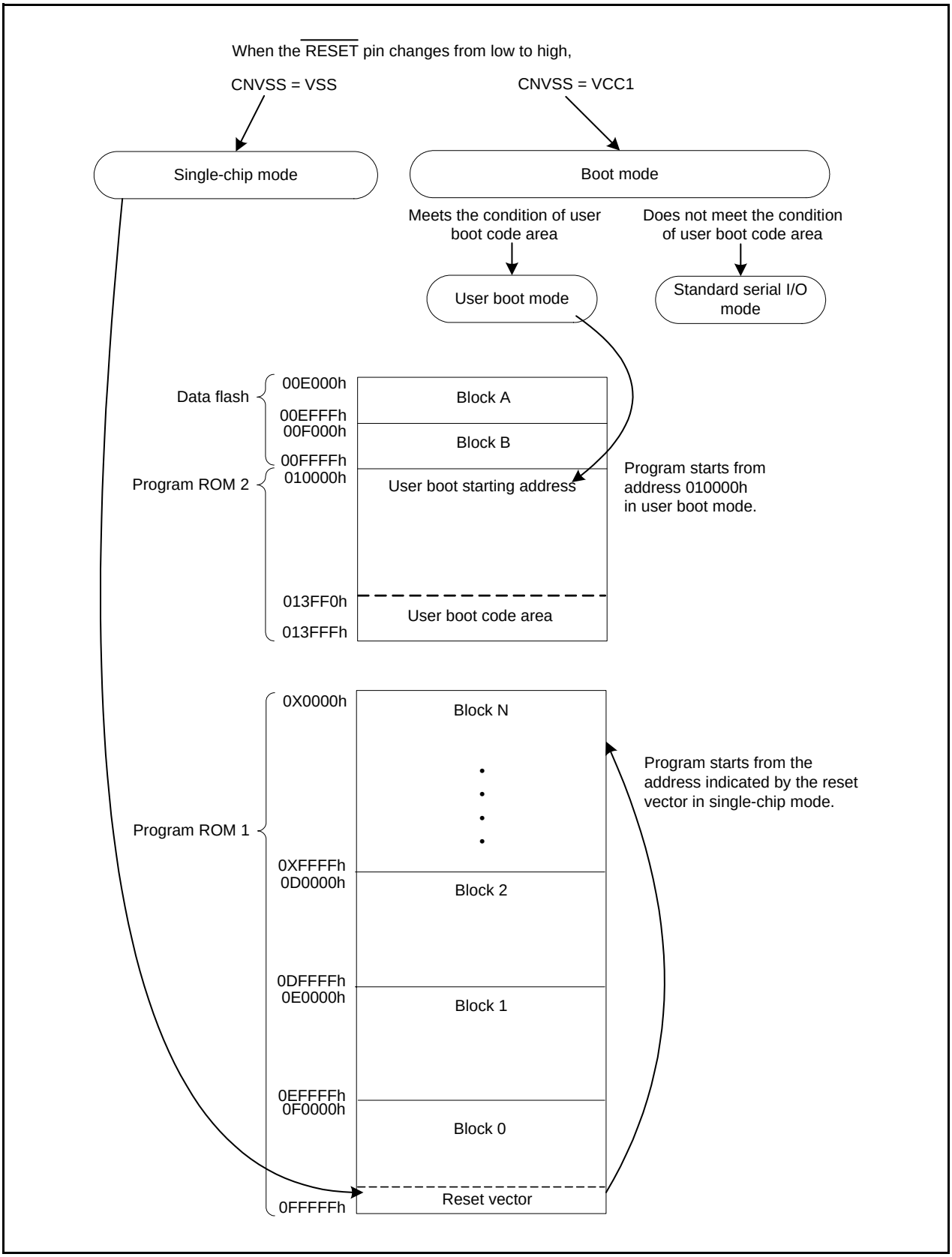


Figure 24.4 Program Starting Address in User Boot Mode

24.8 CPU Rewrite Mode

In CPU rewrite mode, the flash memory can be rewritten when the CPU executes software commands. Program ROM 1, program ROM 2, and data flash can be rewritten with the MCU mounted on the board and without using a ROM programmer.

The program and block erase commands are executed only in individual block areas of program ROM 1, program ROM 2, and data flash.

The flash memory has a suspend function to temporarily suspend operation when erasing or programming in CPU rewrite mode. Refer to 24.8.3 “Suspend Function” for details of the suspend function.

Erase-write 0 mode (EW0 mode) and erase-write 1 mode (EW1 mode) are available in CPU rewrite mode. Table 24.11 lists the differences between EW0 mode and EW1 mode.

Table 24.11 EW0 Mode and EW1 Mode

Item	EW0 Mode	EW1 Mode
Rewrite control Program allocatable area	• Program ROM 1 • Program ROM 2	• Program ROM 1 • Program ROM 2
Rewrite Control Program executable area	The rewrite control program must be transferred to an area other than the flash memory (e.g., RAM) before being executed.	The rewrite control program can be executed in program ROM 1 and program ROM 2.
Rewritable area	• Program ROM 1 • Program ROM 2 • Data flash	• Program ROM 1 • Program ROM 2 • Data flash Excluding blocks with the rewrite control program
Software command restriction	None	• Program and block erase commands Do not execute in a block with the rewrite control program. • Read status register command Do not execute.
Mode after program/erase, or during program/erase suspend	Read status register mode	Read array mode
State during auto write and auto erase	Hold state is not maintained.	Bus is in a hold state. (1)
Flash memory status detection	• Read bits FMR00 and FMR06 in the FMR0 register, FMR75 bit in the FMR7 register, and bits FMR32 and FMR33 in the FMR3 register by a program. • Execute the read status register command, and then read bits SR7, SR5 and SR4 in the status register.	Read bits FMR00 and FMR06 in the FMR0 register, FMR75 bit in the FMR7 register, and bits FMR32 and FMR33 in the FMR3 register by a program.

Note:

1. Refer to 11.3.1.2 “Bus Hold” for detail about the bus hold.

24.8.1 Operating Speed

Set a CPU clock frequency of 10 MHz or less by the CM06 bit in the CM0 register and bits CM17 and CM16 in the CM1 register before entering CPU rewrite mode (EW0 or EW1 mode). Also, set the PM17 bit in the PM1 register to 1 (wait state).

24.8.2 Data Protect Function

Each block in the flash memory has a nonvolatile lock bit. The lock bit is enabled by setting the FMR02 bit to 0 (lock bit enabled). The lock bit allows blocks to be individually protected (locked) against programming and erasure. This prevents data from being inadvertently written to or erased from the flash memory. Table 24.12 lists Lock Bit and Block State.

Table 24.12 Lock Bit and Block State

FMR02 Bit in the FMR0 Register	Lock Bit	Block State
0 (enabled)	0 (locked)	Protected against programming and erasure
	1 (unlocked)	Can be programmed or erased
1 (disabled)	0 (locked)	Can be programmed or erased
	1 (unlocked)	

Condition to become 0:

- Execute the lock bit program command

Condition to become 1:

- Execute the block erase command while the FMR02 bit in the FMR0 register is set to 1 (lock bit disabled).

If the block erase command is executed while the FMR02 bit is set to 1, the target block is erased regardless of lock bit status. The lock bit data can be read by the read lock bit status command.
Refer to 24.8.4 “Software Command (EW0 Mode)”, for details on each command.

24.8.3 Suspend Function

The suspend function suspends automatic programming and erasure. It can be used for an interrupt operation because program ROM 1, program ROM 2, and data flash can be read while automatic programming or erasure is suspended. Enable the interrupts used to enter suspend mode beforehand. The program command, erase command, and lock bit program command are subjects for suspend. Suspend operation is the same for the program command and lock bit program command, so both commands are described together as program suspend.
Do not suspend again in suspend mode. Table 24.13 lists Operation after Command is Issued during Suspend.

Table 24.13 Operation after Command is Issued during Suspend

Suspend	Command	Operation	
		Blocks erased or programmed before suspend	Other blocks
Erase suspend (suspend while executing erase command)	Block erase	The command is not executed. A command sequence error occurs.	
	Program	The command is not executed. A command sequence error occurs.	The command can be executed. Program suspend does not start or an error does not occur even when setting the FMR31 bit to 1 (suspend request).
	Lock bit program	The command is not executed. A command sequence error occurs.	The command can be executed.
	Read array	The command can be executed.	
	Read status register		
	Clear status register		
	Read lock bit status	The command is not executed. A command sequence error occurs.	The command can be executed.
	Block blank check	Do not execute the command.	
Program suspend (suspend while executing program or lock bit program command)	Block erase	The command is not executed. A command sequence error occurs. (1)	
	Program		
	Lock bit program		
	Read array	The command can be executed.	
	Read status register		
	Clear status register	Do not execute the command. (1)	
	Read lock bit status	Do not execute the command.	
	Block blank check		

Note:
1. If the command sequence error occurs after executing block erase, program, or lock bit program commands mistakenly during program suspend. execute the clear status register command, then restart suspend.

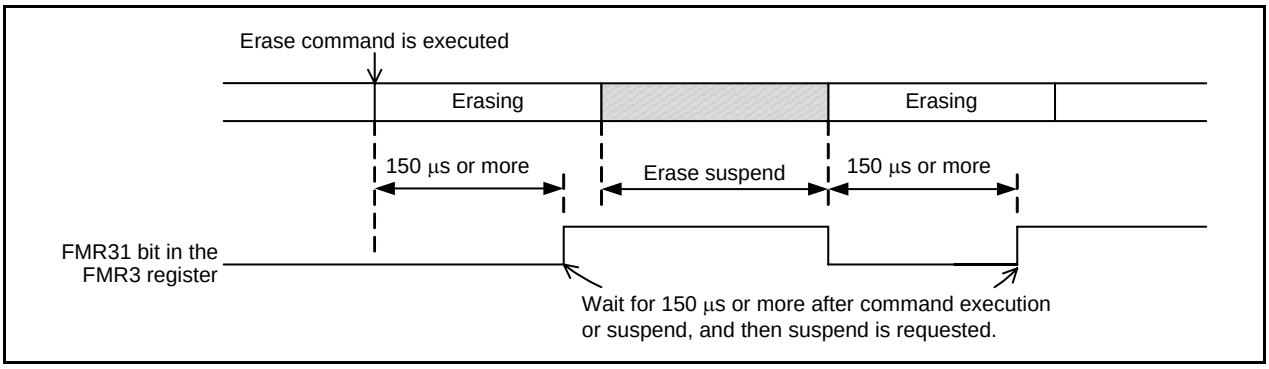


Figure 24.5 Suspend Request

24.8.4 Software Command (EW0 Mode)

Table 24.14 shows Software Commands. Read or write commands and data in 16-bit units. When command code is written, the 8 high-order bits (D15 to D8) are ignored.

Table 24.14 EW0 Mode Software Commands

Command	First Bus Cycle			Second Bus Cycle			Third Bus Cycle		
	Mode	Address	Data (D15 to D0)	Mode	Address	Data (D15 to D0)	Mode	Address	Data (D15 to D0)
Read array	Write	x	xxFFh						
Read status register	Write	x	xx70h	Read	x	SRD			
Clear status register	Write	x	xx50h						
Program	Write	WA	xx41h	Write	WA	WD0	Write	WA	WD1
Block erase	Write	x	xx20h	Write	BA	xxD0h			
Lock bit program	Write	BA	xx77h	Write	BA	xxD0h			
Read lock bit status	Write	x	xx71h	Write	BA	xxD0h			
Block blank check (1)	Write	x	xx25h	Write	BA	xxD0h			

SRD : Data in the status register (D7 to D0)
WA : Write address (Set the end of the address to 0h, 4h, 8h, or Ch)
WD0 : Write data low-order word (16 bits)
WD1 : Write data high-order word (16 bits)
BA : Highest-order block address (even address)
x : Any even address in program ROM 1, program ROM 2, or data flash
xx : Eight high-order bits of command code (ignored)
Note:
1. Block blank check command is designed for programmer manufacturer. Not for customers in general.

Software commands are described below. For symbols shown in flowcharts, refer to those in Table 24.14. Refer to 24.8.3 “Suspend Function” for program, block erase, and lock bit program commands when using suspend function.

24.8.5 Software Command (EW1 Mode)

Table 24.15 shows Software Commands. Read or write commands and data in 16-bit units. When command code is written, the 8 high-order bits (D15 to D8) are ignored.
When using the program command, set the FMR14 bit in the FMR1 register to 1.

Table 24.15 EW1 Mode Software Commands

Command	First Bus Cycle			Second Bus Cycle			Third and Following Bus Cycles		
	Mode	Address	Data (D15 to D0)	Mode	Address	Data (D15 to D0)	Mode	Address	Data (D15 to D0)
Read array	Write	x	xxFFh						
Read status register	Write	x	xx70h	Read	x	SRD			
Clear status register	Write	x	xx50h						
Program	Write	WA	xx41h	Write	WA	FF_{16} + BD0	Write	WA	FF_{16} + BDn (n = 1 to 3)
Block erase	Write	x	xx20h	Write	BA	xxD0h			
Lock bit program	Write	BA	xx77h	Write	BA	xxD0h			
Read lock bit status	Write	x	xx71h	Write	BA	xxD0h			
Block blank check (1)	Write	x	xx25h	Write	BA	xxD0h			

SRD : Data in the status register (D7 to D0)
WA : Write address (Set the end of the address to 0h, 4h, 8h, or Ch)
WD0 : Write data low-order word (16 bits)
WD1 : Write data high-order word (16 bits)
BA : Highest-order block address (even address)
x : Any even address in program ROM 1, program ROM 2, or data flash
xx : Eight high-order bits of command code (ignored)
BD : Write byte data (8 bits) (Set the high-order byte (D8 to D15) to FF_{16}.)
Note:
1. Block blank check command is designed for programmer manufacturer. Not for customers in general.

Software commands are described below. For symbols shown in flowcharts, refer to those in Table 24.12. Refer to 24.8.3 “Suspend Function” for program, block erase, and lock bit program commands when using suspend function.

24.8.5.1 Read Array Command

The read array command is used to read the flash memory.
By writing the command code xxFFh in the first bus cycle, the flash memory enters read array mode. The content of the specified address can be read in 16-bit units by entering the address to be read after the next bus cycle.
The flash memory remains in read array mode until another command is written. Therefore, the contents of multiple addresses can be read consecutively.

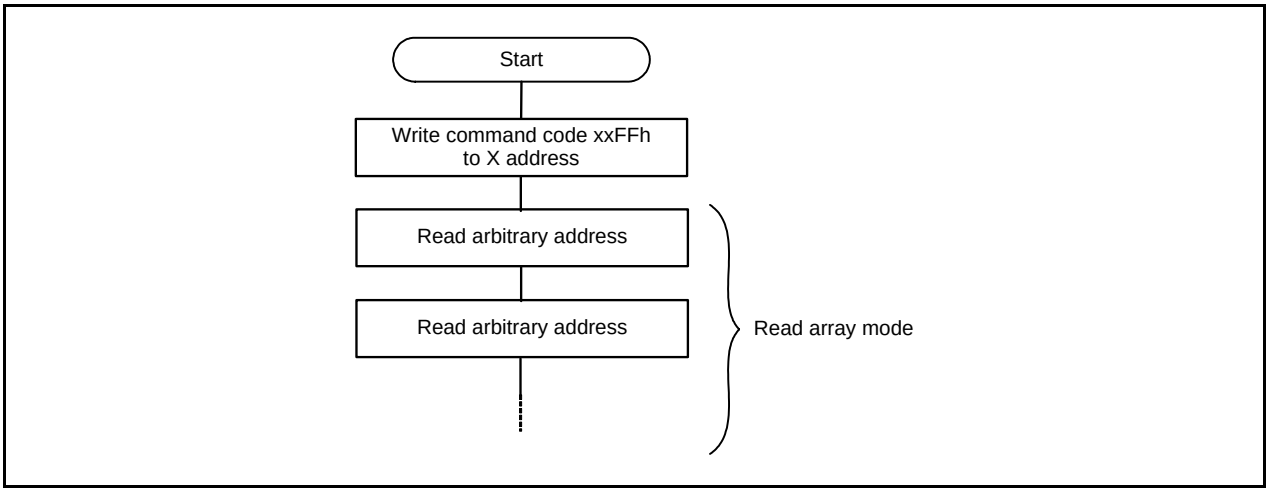


Figure 24.6 Read Array Command

24.8.5.2 Read Status Register Command

The read status register command is used to read the status register.
By writing the command code xx70h in the first bus cycle, the status register can be read in the second bus cycle. (Refer to 24.8.6 “Status Register”). To read the status register, read an even address in the program ROM 1, program ROM 2, or data flash.
Do not execute this command in EW1 mode.

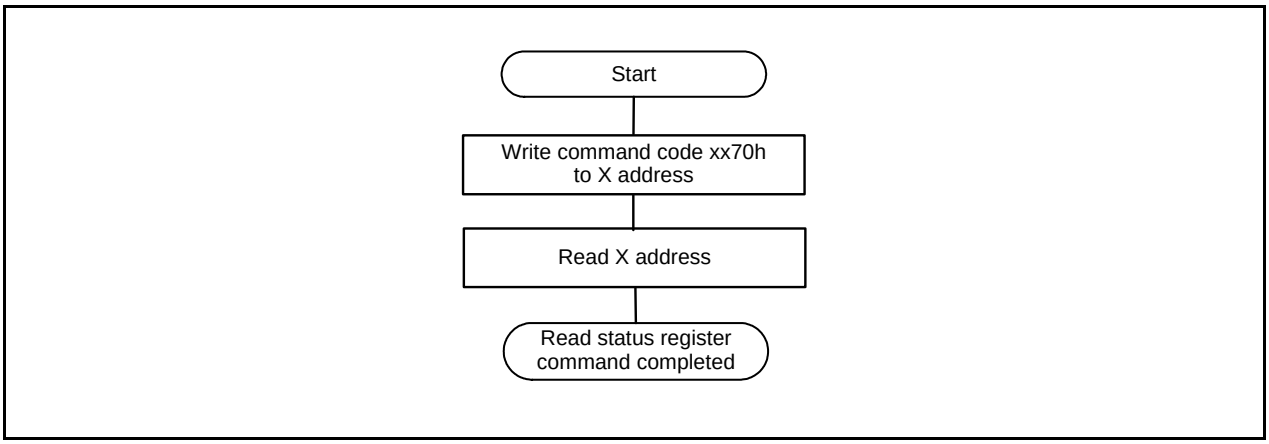


Figure 24.7 Read Status Register Command

24.8.5.3 Clear Status Register Command

The clear status register command is used to clear the status register.
By writing the command code xx50h, bits FMR07 and FMR06 in the FMR0 register (SR5 and SR4 in the status register) become 00b.

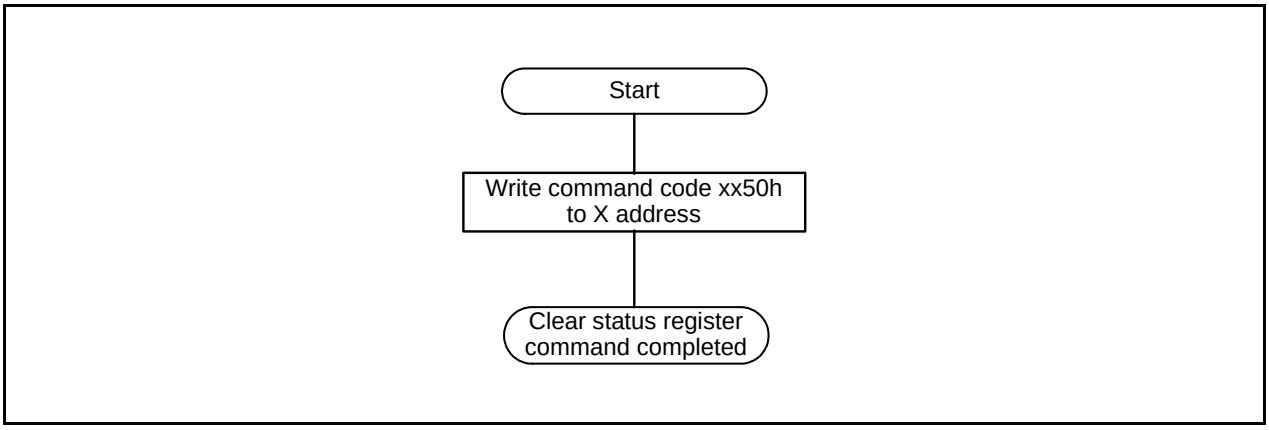


Figure 24.8 Clear Status Register Command

24.8.5.4 Program Command (EW0 Mode)

The program command is used to write two words (4 bytes) of data to the flash memory.
By writing xx41h in the first bus cycle and data to the write address in the second and third bus cycles, auto-program operation (data program and verify) is started. Set the end of the write address to 0h, 4h, 8h, or Ch.
The FMR00 bit in the FMR0 register indicates whether the auto-program operation has been completed. The FMR00 bit is 0 (busy) during the auto-program operation, and 1 (ready) after the auto-program operation is completed.
After the auto-program operation is completed, the FMR06 bit in the FMR0 register indicates whether or not the auto-program operation has been completed as expected. (Refer to 24.8.6.1 “Full Status Check”).
Do not rewrite the addresses already programmed. Figure 24.9 shows a Flow Chart of the Program Command Programming (Suspend Function Disabled).
The lock bit protects individual blocks from being programmed inadvertently. (Refer to 24.8.2 “Data Protect Function”.)

In EW1 mode, do not execute this command on a block to which the rewrite control program is allocated.

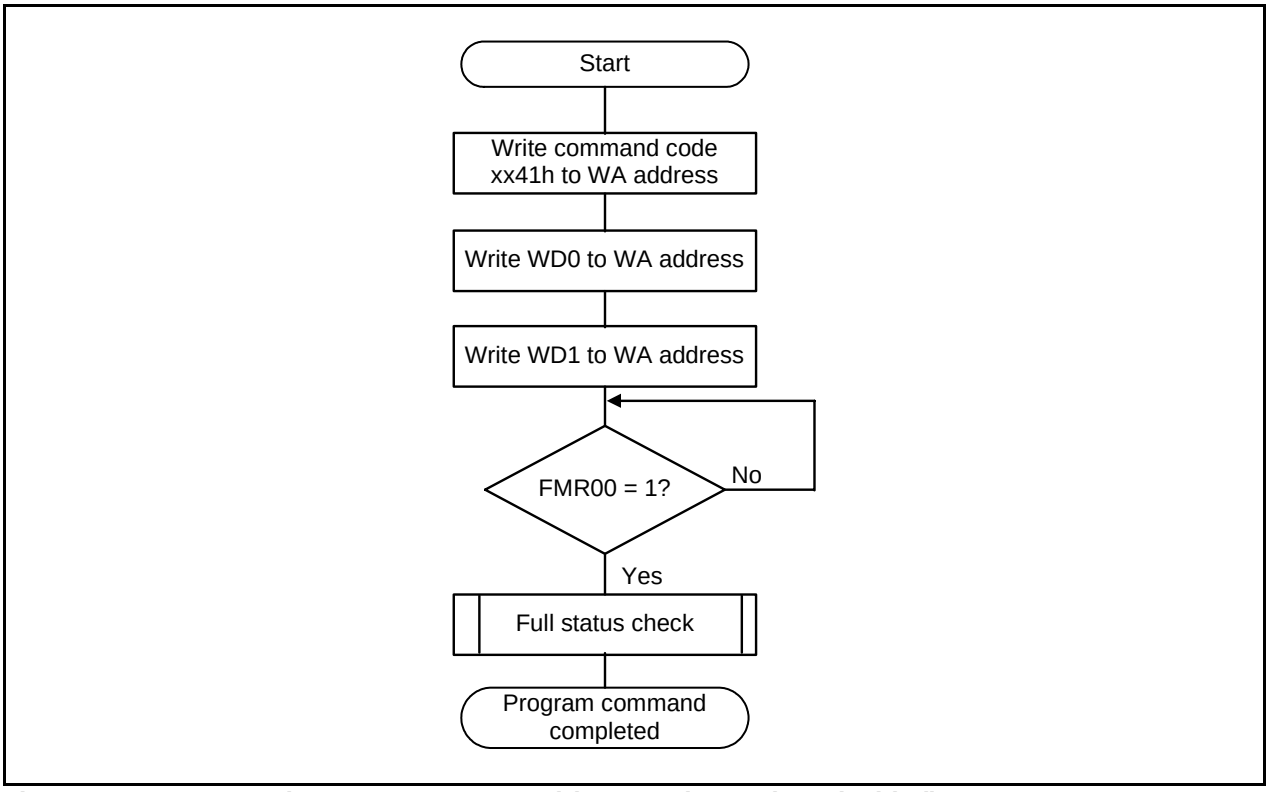


Figure 24.9 EW0 Mode Program Command (Suspend Function Disabled)

24.8.5.5 Program Command (EW1 Mode)

The program command is used to write 4 bytes of data to the flash memory. Se the FMR14 bit in the FMR1 register to 1.

By writing xx41h in the first bus cycle and data to the write address in the second and third to fifth bus cycles, auto-program operation (data program and verify) is started. Set the end of the write address to 0h, 4h, 8h, or Ch.

The FMR00 bit in the FMR0 register indicates whether the auto-program operation has been completed. The FMR00 bit is 0 (busy) during the auto-program operation, and 1 (ready) after the auto-program operation is completed.

After the auto-program operation is completed, the FMR06 bit in the FMR0 register indicates whether or not the auto-program operation has been completed as expected. (Refer to 24.8.6.1 “Full Status Check”).

Do not rewrite the addresses already programmed. Figure 24.9 shows a Flow Chart of the Program Command Programming (Suspend Function Disabled).

The lock bit protects individual blocks from being programmed inadvertently. (Refer to 24.8.2 “Data Protect Function”.)

In EW1 mode, do not execute this command on a block to which the rewrite control program is allocated.

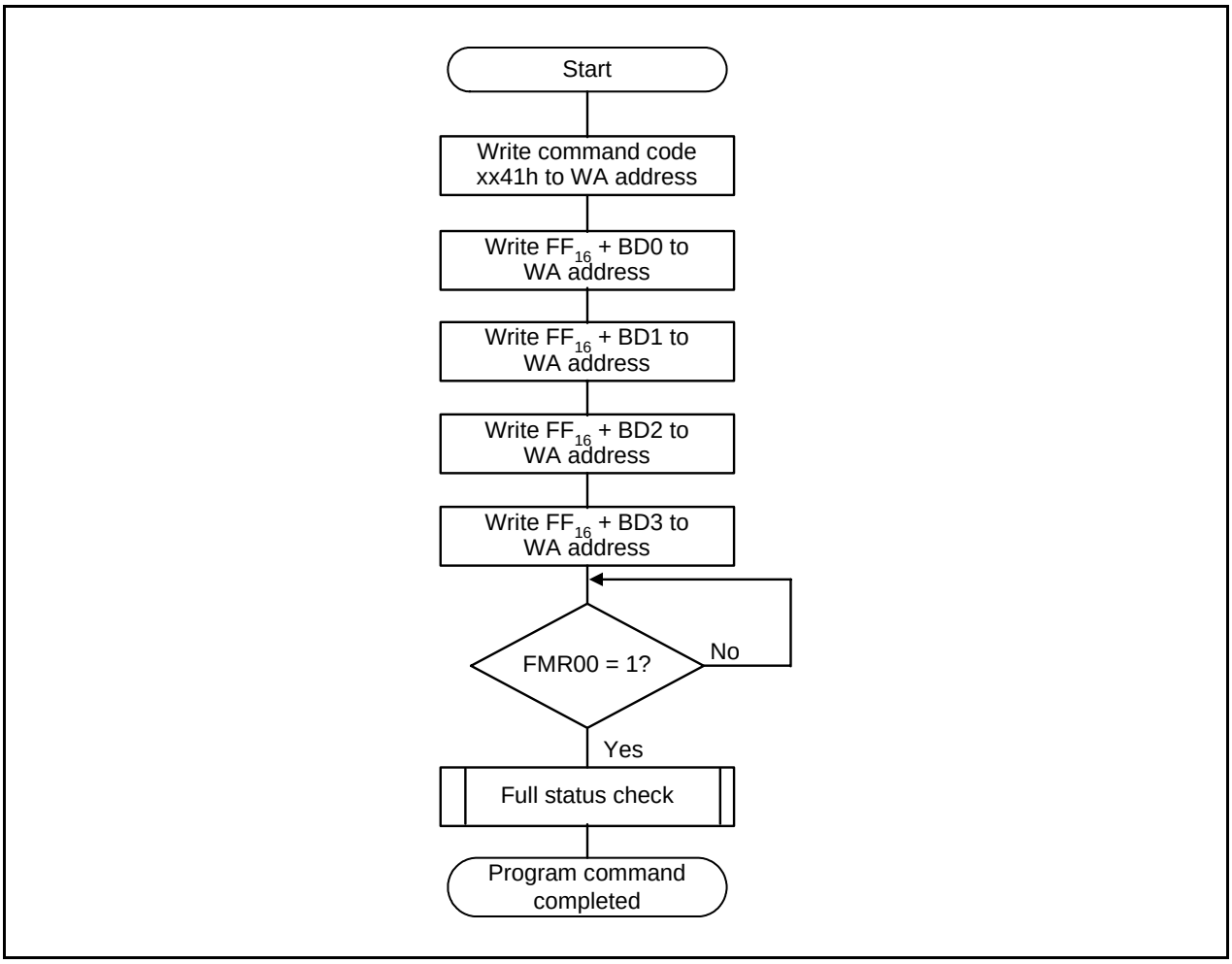


Figure 24.10 EW1 Mode Program Command (Suspend Function Disabled)

24.8.5.6 Block Erase Command

By writing xx20h in the first bus cycle and xxD0h to the highest-order even address of a block in the second bus cycle, an auto-erase operation (erase and verify) is started on the specified block.
The FMR00 bit in the FMR0 register indicates whether the auto-erase operation has been completed. The FMR00 bit is 0 (busy) during the auto-erase operation, and 1 (ready) when the auto-erase operation is completed.
After the auto erase operation is completed, the FMR75 bit in the FMR7 register indicates whether or not the auto erase operation has been completed as expected. (Refer to 24.8.6.1 “Full Status Check”).
Figure 24.11 shows a Flow Chart of the Block Erase Command Programming (Suspend Function Disabled).
The lock bit protects individual blocks from being erased inadvertently. (Refer to 24.8.2 “Data Protect Function”.)
In EW1 mode, do not execute this command on the block to which the rewrite control program is allocated.

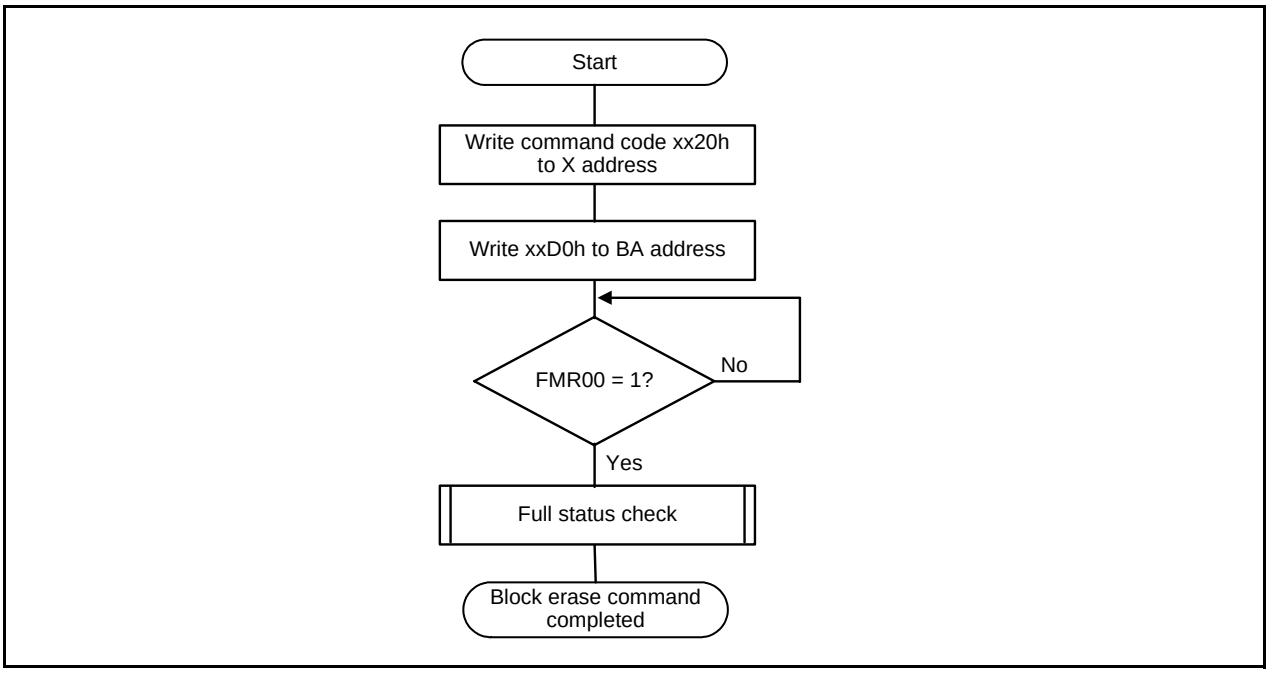


Figure 24.11 Block Erase Command (Suspend Function Disabled)

24.8.5.7 Lock Bit Program Command

The lock bit program command is used to set the lock bit for a specified block to 0 (locked).
By writing xx77h in the first bus cycle and xxD0h to the highest-order even address of a block in the second bus cycle, the lock bit for the specified block is set to 0. The address value specified in the first bus cycle must be the same highest-order address of a block specified in the second bus cycle.
Figure 24.12 shows a Flow Chart of the Lock Bit Program Command Programming (Suspend Function Disabled). Execute the read lock bit status command to read the lock bit state (lock bit data).
The FMR00 bit in the FMR0 register indicates whether a lock bit program operation has been completed.
Refer to 24.8.2 “Data Protect Function”, for details on lock bit functions and how to set it to 1 (unlocked).

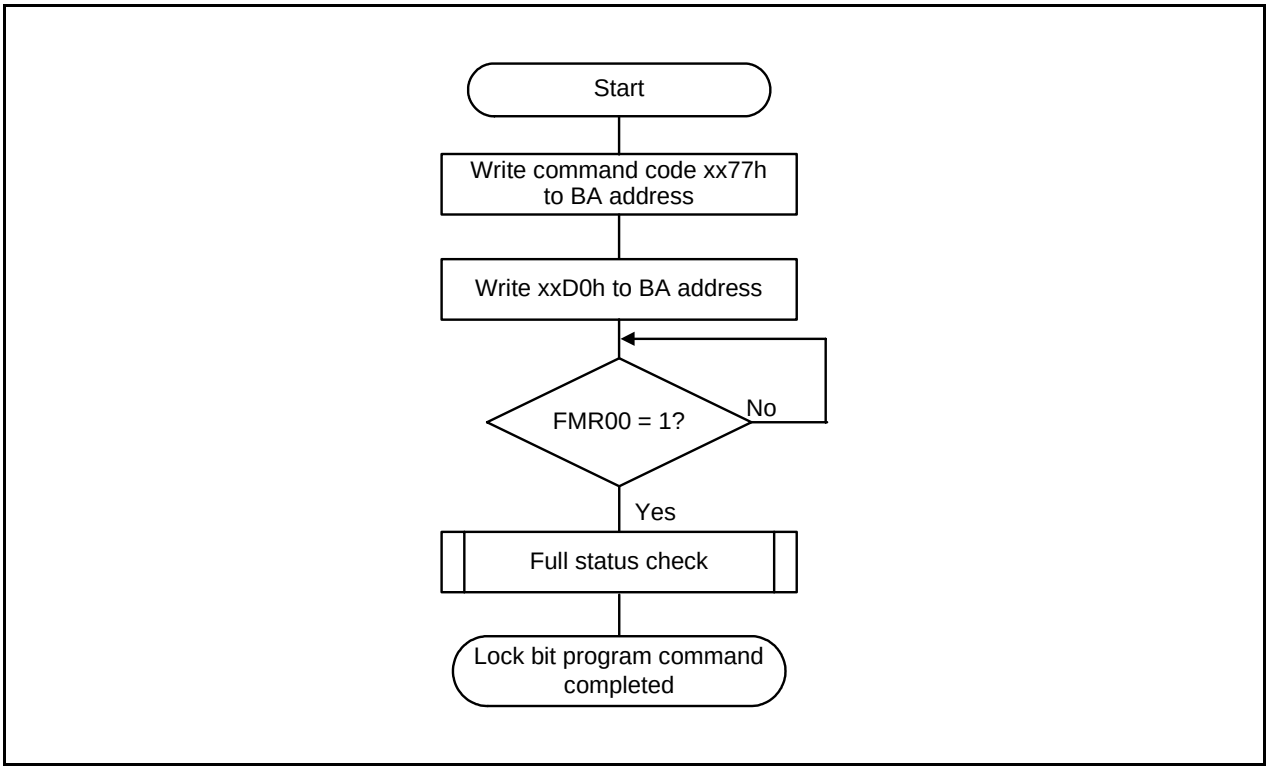


Figure 24.12 Lock Bit Program Command (Suspend Function Disabled)

24.8.5.8 Read Lock Bit Status

The read lock bit status command is used to read the lock bit state of a specified block.
By writing xx71h in the first bus cycle and xxD0h to the highest-order even address of a block in the second bus cycle, the FMR16 bit in the FMR1 register stores information on the lock bit status of a specified block. Read the FMR16 bit after the FMR00 bit in the FMR0 register is set to 1 (ready). Do not execute other commands while the FMR00 bit is 0.
Figure 24.13 shows a Flow Chart of the Read Lock Bit Status Command Programming.

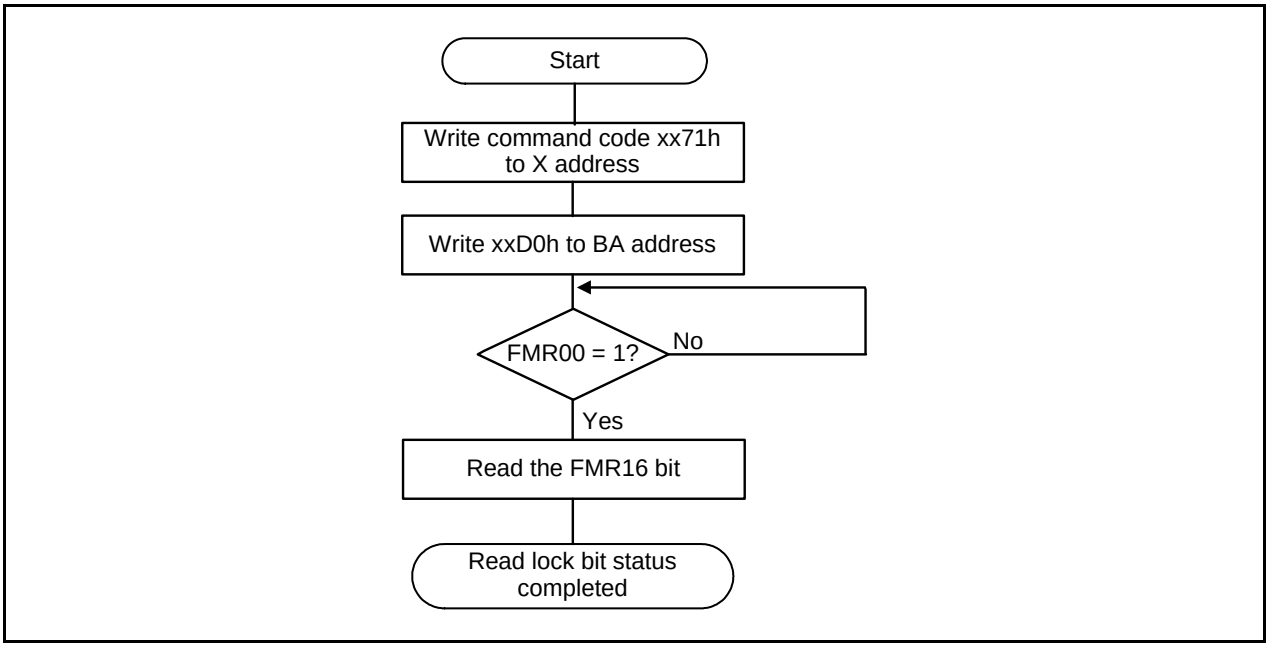


Figure 24.13 Read Lock Bit Status Command

24.8.5.9 Block Blank Check Command

The block blank check command is used to check whether or not a specified block is blank (state after erase).
By writing xx25h in the first bus cycle and xxD0h in the second bus cycle to the highest-order even address of a block, the check result is stored in the FMR75 bit in the FMR7 register. Read the FMR75 bit after the FMR00 bit in the FMR0 register is set to 1 (ready). Do not execute other commands while the FMR00 bit is 0.
The block blank check command is valid for unlocked blocks.
If the block blank check command is executed on a block whose lock bit is 0 (locked), the FMR75 bit (SR5) is set to 1 (not blank) regardless of the FMR02 bit state.
Figure 24.14 shows a Flow Chart of the Block Blank Check Command Programming.

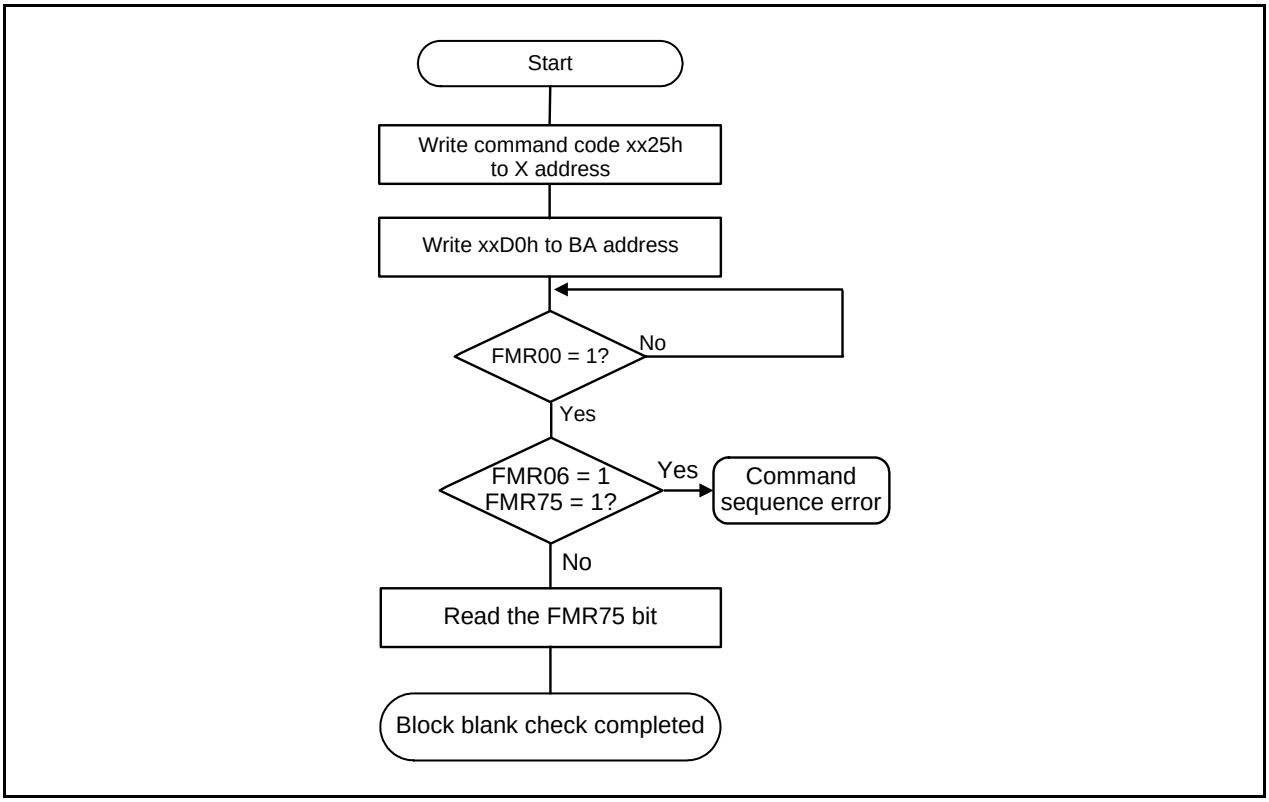


Figure 24.14 Block Blank Check Command

As a result of block blank check, when the block is not blank, execute the clear status register command before executing other software commands.

The block blank check command is designated for use with a programmer. Use this command where instantaneous power failures do not occur. When an instantaneous power failure occurs while the block erase command is executed, execute the block erase command again. The block blank check command cannot be used to check whether the erase operation is successfully completed or not.

Do not execute the block blank check command during suspend.

24.8.6 Status Register

The status register indicates flash memory operating state and whether or not an erase or program operation has completed as expected.
Bits FMR00 and FMR06 in the FMR0 register and the FMR75 bit in the FMR7 register indicate status register states. Refer to 24.3.1 “Flash Memory Control Register 0 (FMR0)” for each bit.

Table 24.16 Difference in Reading of Status Register

Item	Registers FMR0 and FMR7	Command
Condition	No limit	
Reading procedure	Read bits FMR00 and FMR06 in the FMR0 register and the FMR75 bit in the FMR7 register.	• Read any even address in program ROM 1, program ROM 2, or data flash after writing the read status register command. • Read any even address in program ROM 1, program ROM 2, or data flash after executing the program command, block erase command, lock bit program command, or block blank check command before executing the read array command.

Table 24.17 Status Register

Bits in Status Register	Bit in FMR0 and FMR7 Register	Status	Definition		Reset Value
			0	1	
SR0 (D0)	—	Reserved	—	—	—
SR1 (D1)	—	Reserved	—	—	—
SR2 (D2)	—	Reserved	—	—	—
SR3 (D3)	—	Reserved	—	—	—
SR4 (D4)	FMR06	Program status	Completed as expected	Completed in error	0
SR5 (D5)	FMR75	Erase status	Completed as expected	Completed in error	0
SR6 (D6)	—	Reserved	—	—	—
SR7 (D7)	FMR00	Sequencer status	Busy	Ready	1

D0 to D7: The data buses read when the read status register command is executed.

24.8.6.1 Full Status Check

If an error occurs, the FMR06 bit in the FMR0 register or the FMR75 bit in the FMR7 register is set to 1, indicating the occurrence of an error. Therefore, the execution results can be confirmed by checking these status bits (full status check).

Table 24.18 Errors and FMR0 and FMR7 Register States

FMR75 bit in the FMR7 register	FMR06 bit in the FMR0 register	Error	Error Occurrence Conditions
1	1	Command Sequence error	• Command is written incorrectly. • Invalid data (data other than xxD0h or xxFFh) is written in the second bus cycle of the lock bit program or block erase command. (1)
1	0	Erase error	• The block erase command is executed on a locked block. (2) • The block erase command is executed on an unlocked block, but the auto-erase operation is not completed as expected. • The block blank check command is executed, and the check result is not blank.
0	1	Program error	• The program command is executed on a locked block. (2) • The program command is executed on an unlocked block, but auto-program operation is not completed as expected. • The lock bit program command is executed, but the lock bit is not written as expected. (2)

Notes:
1. The flash memory enters read array mode by writing command code xxFFh in the second bus cycle of the command. The command code written in the first bus cycle becomes invalid.
2. When the FMR02 bit is set to 1 (lock bit disabled), no error occurs even under the conditions above.

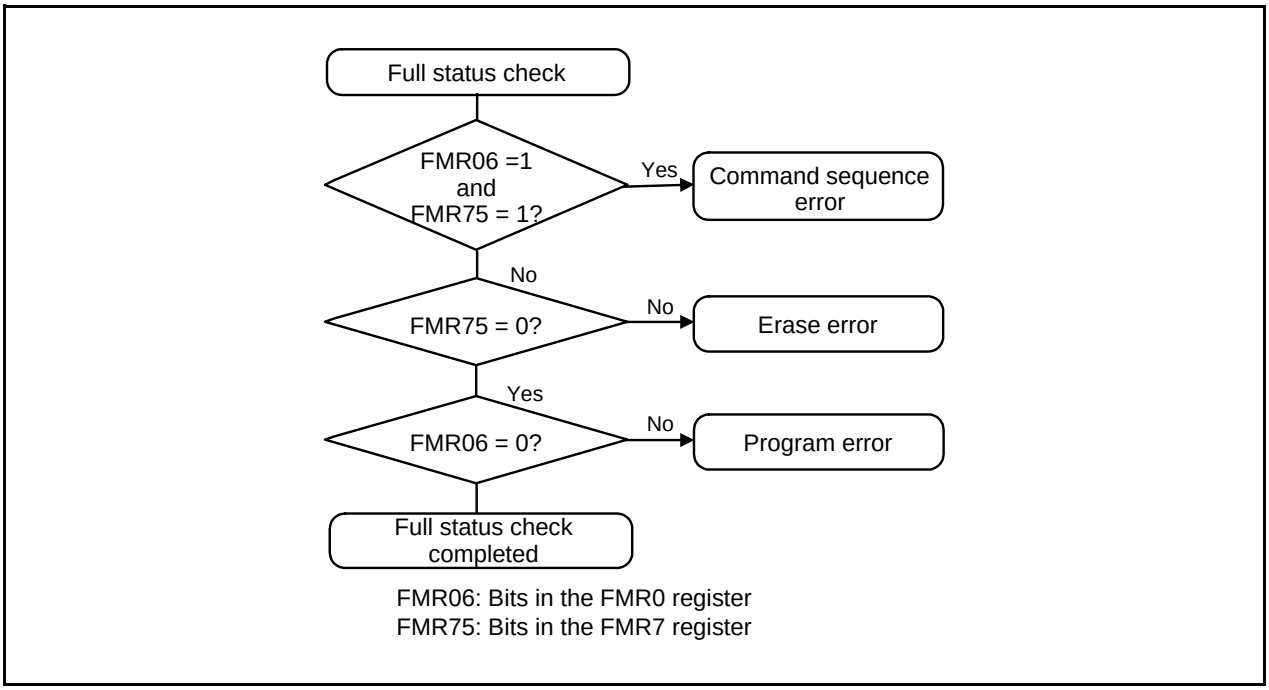


Figure 24.15 Full Status Check

24.8.6.2 Handling Procedure for Errors

When errors occur, follow the procedures below.
Do not execute the program, block erase, lock bit program, and block blank check commands when either FMR06 or FMR75 is 1 (completed in error). Execute each command after executing the clear status register command.

Command sequence error

(1) Execute the clear status register command and set bits FMR06 and FMR75 to 0 (completed as expected).
(2) Check if the command is written correctly and execute the correct command.

Erase error

(1) Execute the clear status register command and set the FMR75 bit to 0 (completed as expected).
(2) Execute the read lock bit status command. Set the FMR02 bit in the FMR register to 1 (lock bit disabled) if the lock bit in the block where the error occurred is set to 0 (locked).
(3) Execute the block erase command again.
(4) Execute (1), (2), and (3) until an erase error is not generated.

If an error still occurs even after repeating three times, do not use that block.
When handling an erase error of the block blank check command and erasing is not necessary, execute (1) only.

Program error

[When a program operation is executed]

(1) Execute the clear status register command and set the FMR06 bit to 0 (completed as expected).
(2) Execute the read lock bit status command. Set the FMR02 bit in the FMR0 register to 1 if the lock bit in the block where the error occurred is set to 0.
(3) Execute the program command again.

If an error still occurs even after repeating three times, do not use that block.
If the lock bit is set to 1 (unlocked), do not use the address in which error has occurred as it is. Execute the block erase command to erase the block, in which the error has occurred, before executing the program command to write to the same address again.

[When a lock bit program operation is executed]

(1) Execute the clear status register command and set the FMR06 bit to 0.
(2) Set the FMR02 bit in the FMR0 register to 1.
(3) Execute the block erase command to erase the block where the error occurred.
(4) Execute the lock bit program command again after writing the data as needed

If an error still occurs, do not use that block.

24.8.7 EW0 Mode

The MCU enters CPU rewrite mode when the FMR01 bit in the FMR0 register is set to 1 (CPU rewrite mode enabled) and is ready to accept commands. EW0 mode is selected by setting the FMR60 bit in the FMR6 register to 0. Figure 24.16 shows Setting and Resetting of EW0 Mode

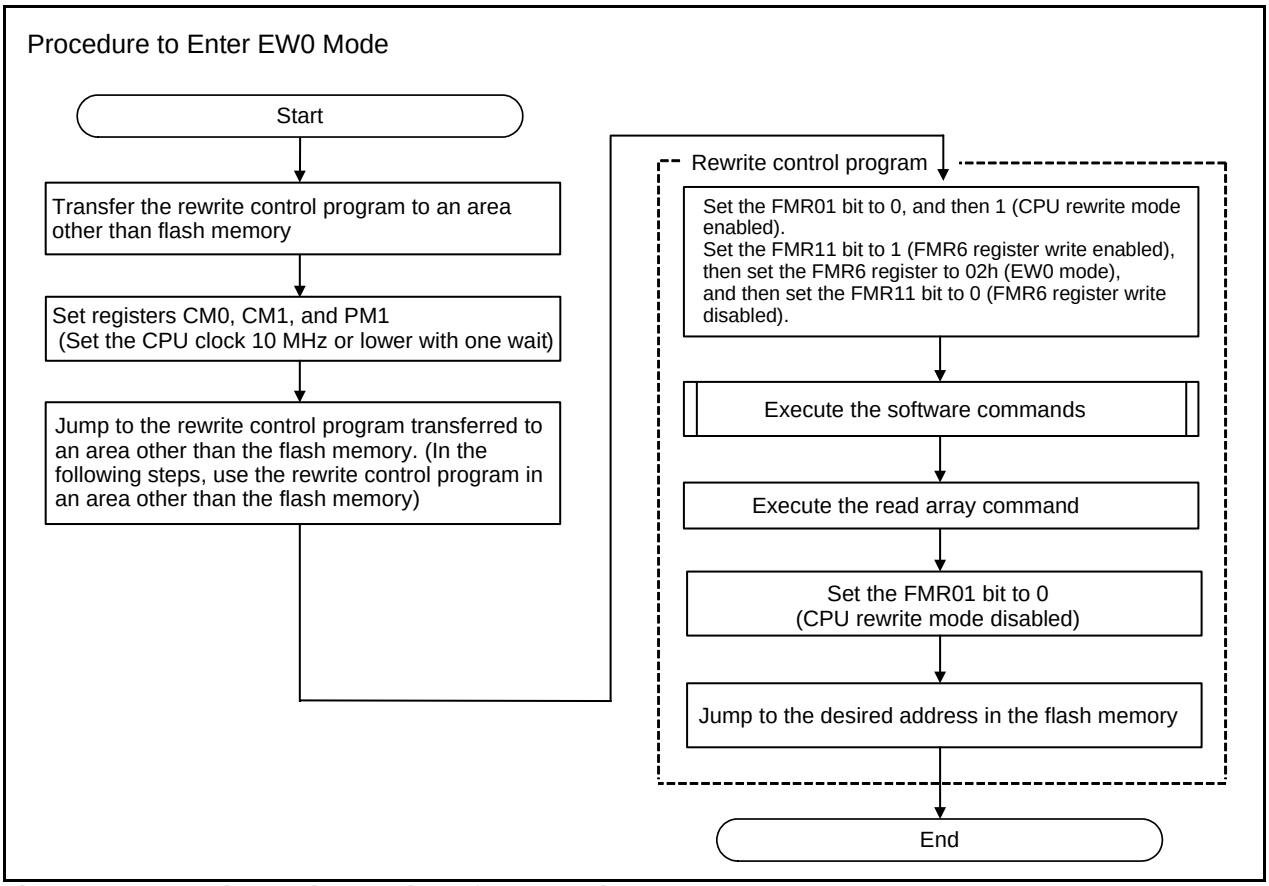


Figure 24.16 Setting and Resetting of EW0 Mode

Do not execute the following instructions:
UND instruction, INTO instruction, JMPS instruction, JSRS instruction, and BRK instruction

The following are interrupts which can be used in EW0 mode and operations when the interrupts occur during auto-erase operation or auto-program operation:

- Maskable interrupt (suspend disabled)
  To use the interrupt, allocate a variable vector table in areas other than the flash memory.
- Maskable interrupt (suspend enabled)
  To use the interrupt, allocate a variable vector table in areas other than the flash memory.
  When the FMR00 bit in the FMR0 register is checked in the interrupt routine and the result is 0 (being written or erased), auto-erase operation or auto-program operation suspends after td(SR-SUS) elapses by setting the FMR31 bit in the FMR3 register to 1 (suspend request). Auto-erase operation or auto-program operation restarts by setting the FMR31 bit to 0 (command restart) at the completion of the interrupt.
- NMI, watchdog timer, oscillator stop/restart detect, voltage detect 1, and voltage detect 2 interrupts
  Auto-erase operation or auto-program operation is forcibly stopped as soon as the interrupt occurs, and then the interrupt process starts.
  After the flash memory restart, execute auto-erase operation again and confirm that it is completed as expected in order to read the correct value.

The watchdog timer operates even in auto erasing or auto programming operation. Refresh the watchdog timer regularly.

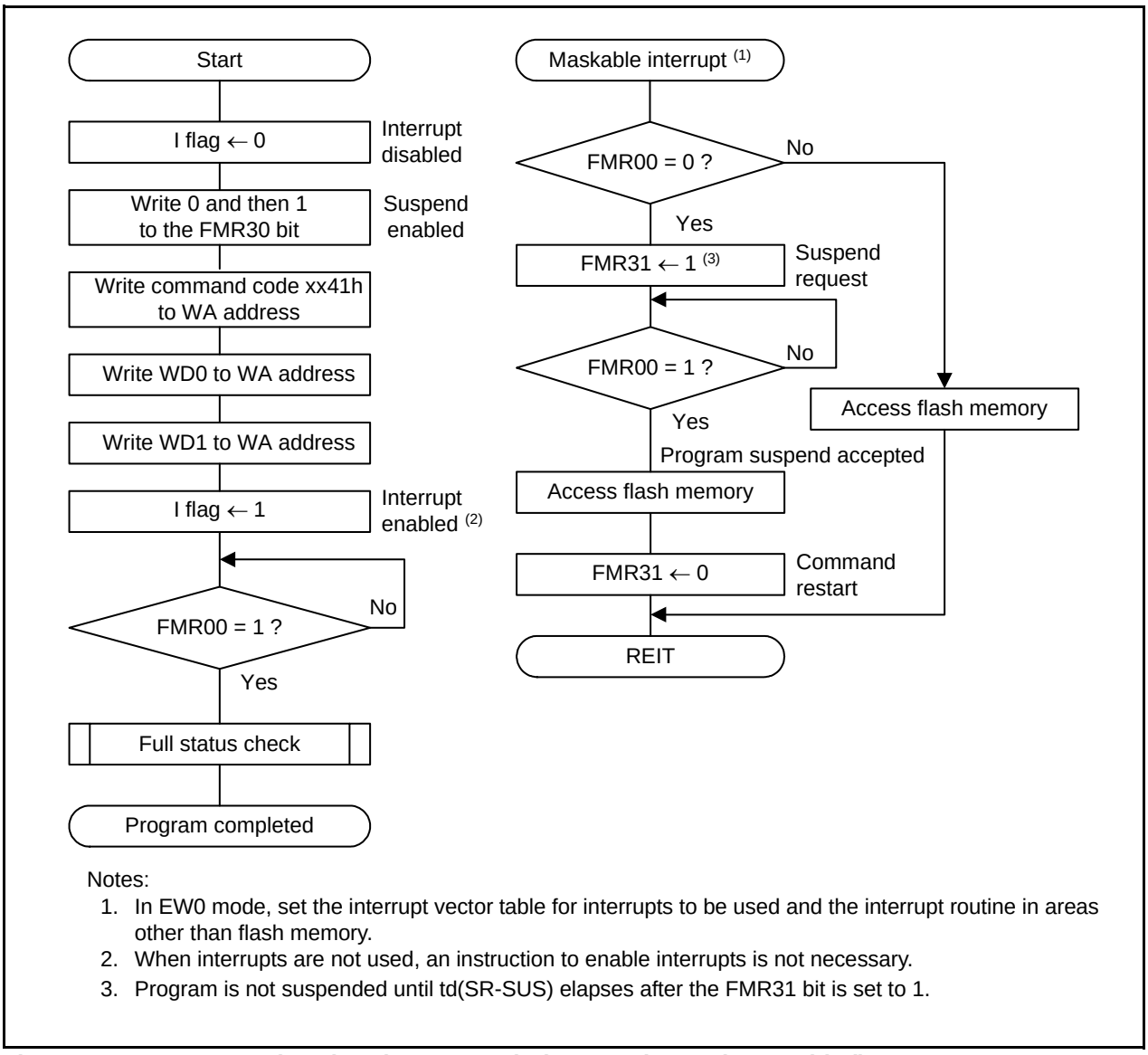
Table 24.19 Modes after Executing Commands (in EW0 Mode)

Command	Mode after Executing Command
Read array	Read array mode
Clear status register	Read array mode
Program	Read status register mode (1)
Block erase	
Lock bit program	
Read lock bit status	Read lock bit status mode (1)
Block blank check	Read status register mode (1)

Note:
1. Flash memory can be read only in read array mode.

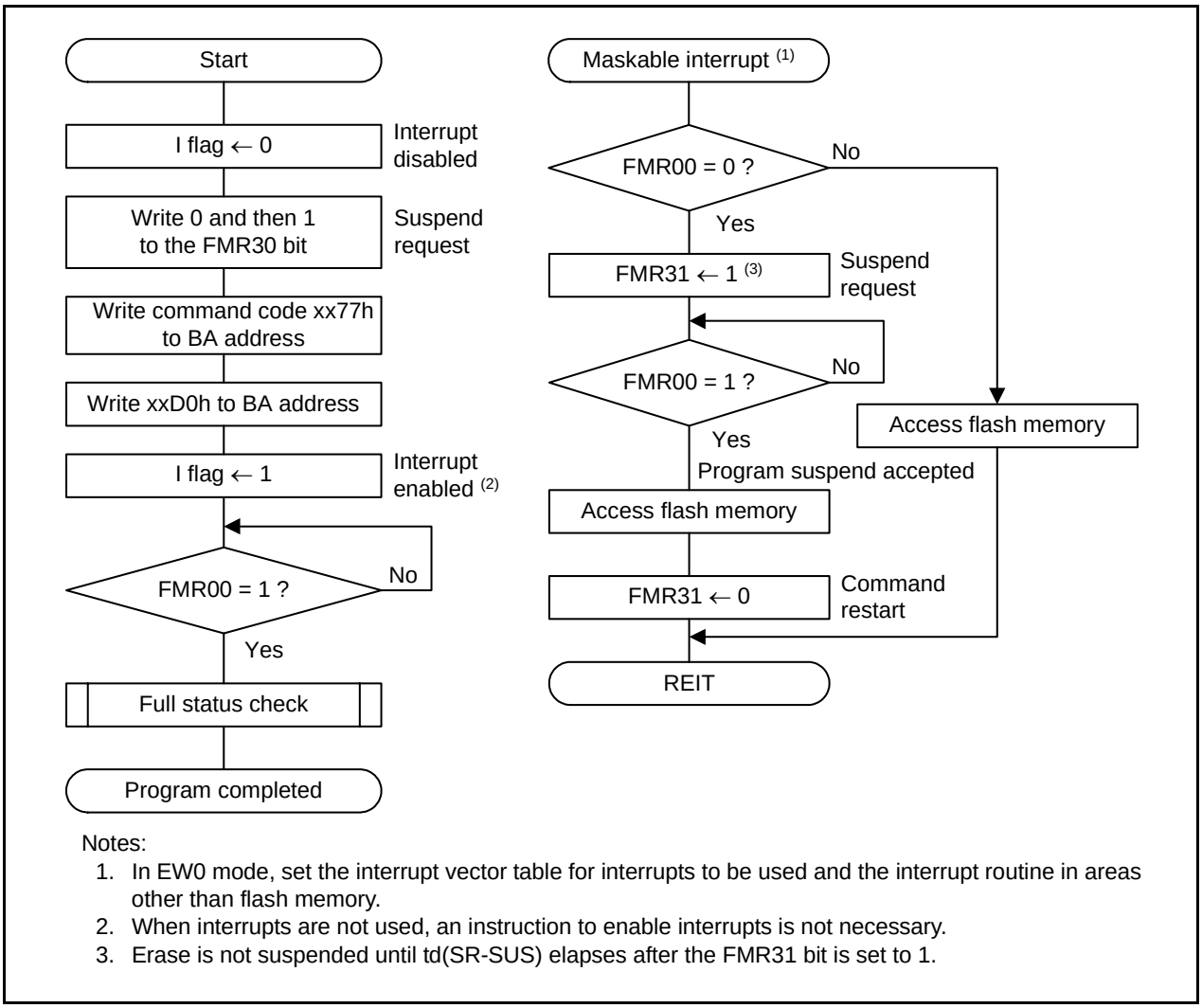
24.8.7.1 Suspend Function (EW0 Mode)

When using suspend function in EW0 mode, check the status of the flash memory in the interrupt routine and enter suspend mode. Program suspend or erase suspend is not accepted until td (SR-SUS) elapses after the FMR31 bit is set to 1. Access to the flash memory after confirming the acceptance of program suspend or erase suspend by the FMR33 or FMR32 bit. Set the FMR31 bit to 0 (command restart) to restart auto-program and auto-erase operations at the completion of the access to the flash memory. Figures 24.17 to 24.19 show a flow chart in EW0 mode when the suspend function is enabled, and Figure 24.20 shows Suspend Operation Example in EW0 Mode.



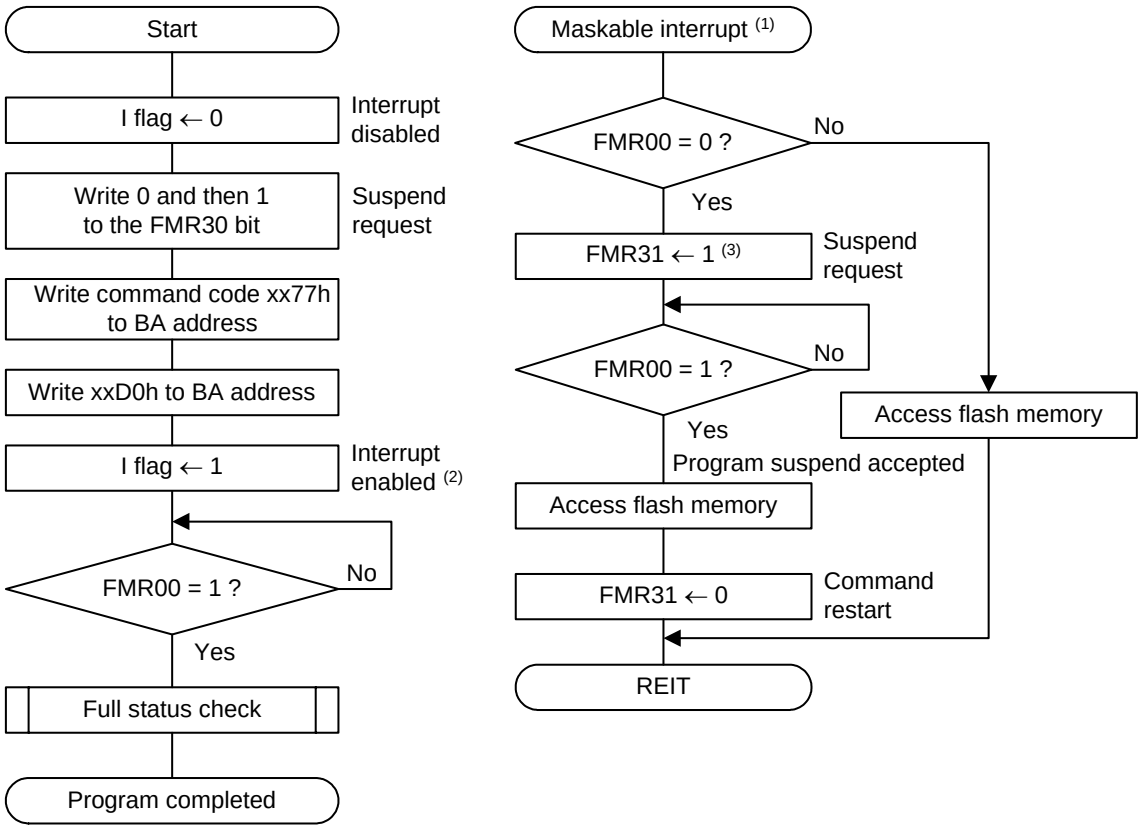
Notes:
1. In EW0 mode, set the interrupt vector table for interrupts to be used and the interrupt routine in areas other than flash memory.
2. When interrupts are not used, an instruction to enable interrupts is not necessary.
3. Program is not suspended until td(SR-SUS) elapses after the FMR31 bit is set to 1.

Figure 24.17 Program Flowchart in EW0 Mode (Suspend Function Enabled)



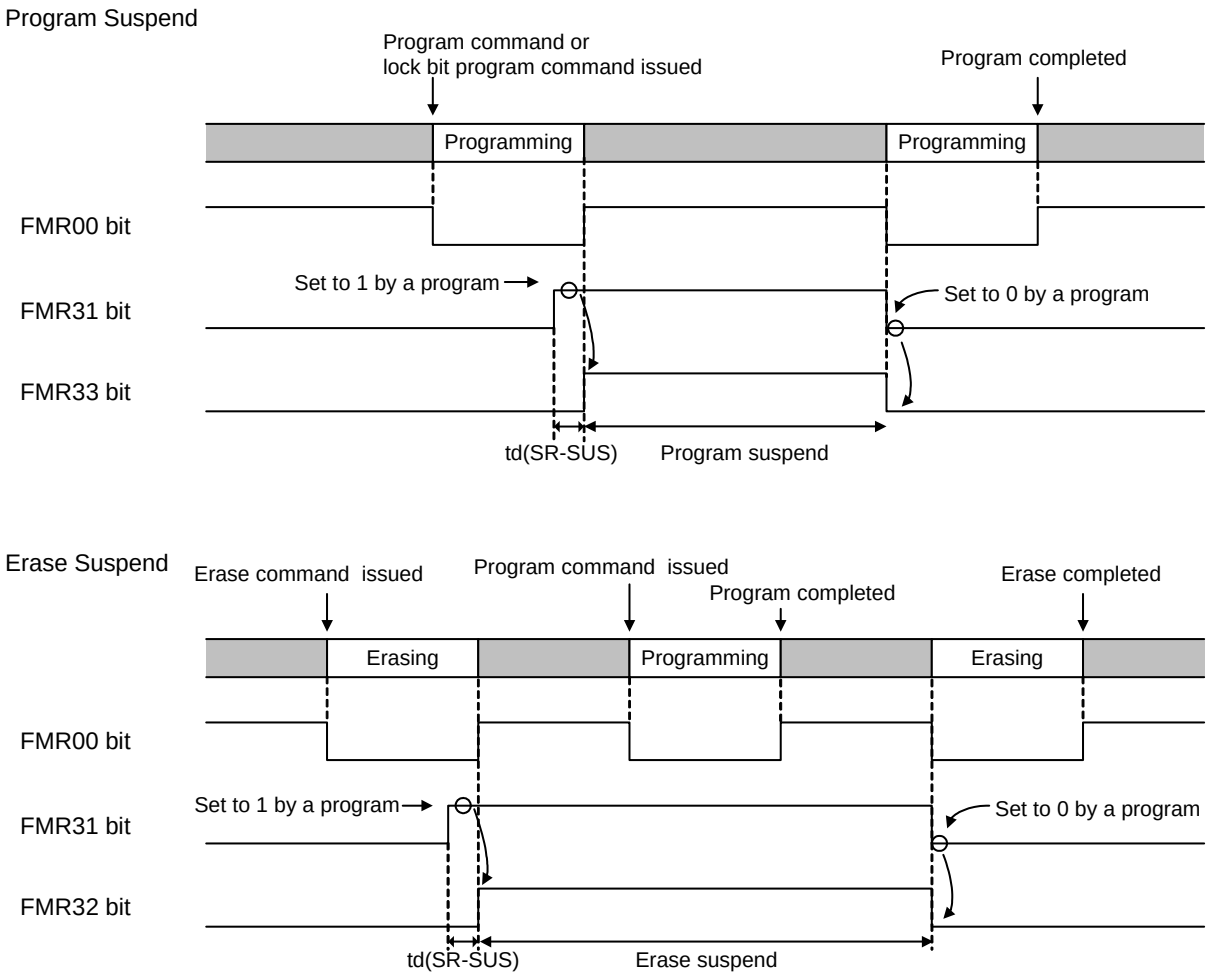
Notes:
1. In EW0 mode, set the interrupt vector table for interrupts to be used and the interrupt routine in areas other than flash memory.
2. When interrupts are not used, an instruction to enable interrupts is not necessary.
3. Erase is not suspended until td(SR-SUS) elapses after the FMR31 bit is set to 1.

Figure 24.18 Block Erase Flowchart in EW0 Mode (Suspend Function Enabled)



Notes:
1. In EW0 mode, set the interrupt vector table for interrupts to be used and the interrupt routine in areas other than flash memory.
2. When interrupts are not used, an instruction to enable interrupts is not necessary.
3. Program is not suspended until td(SR-SUS) elapses after the FMR31 bit is set to 1.

Figure 24.19 Lock Bit Program Flowchart in EW0 Mode (Suspend Function Enabled)



FMR00: Bit in the FMR0 register
FMR33, FMR32, FMR31: Bits in the FMR3 register

The above diagram applies under the following condition.

- The FMR30 bit in the FMR3 register is 1 (suspend enabled).

Figure 24.20 Suspend Operation Example in EW0 Mode

24.8.8 EW1 Mode

EW1 mode is selected by setting the FMR60 bit in the FMR6 register to 1 after setting the FMR01 bit in the FMR0 register to 1. Figure 24.21 shows Setting and Resetting of EW1 Mode.
When a program or erase operation is initiated, the CPU halts all program execution until the operation is completed.

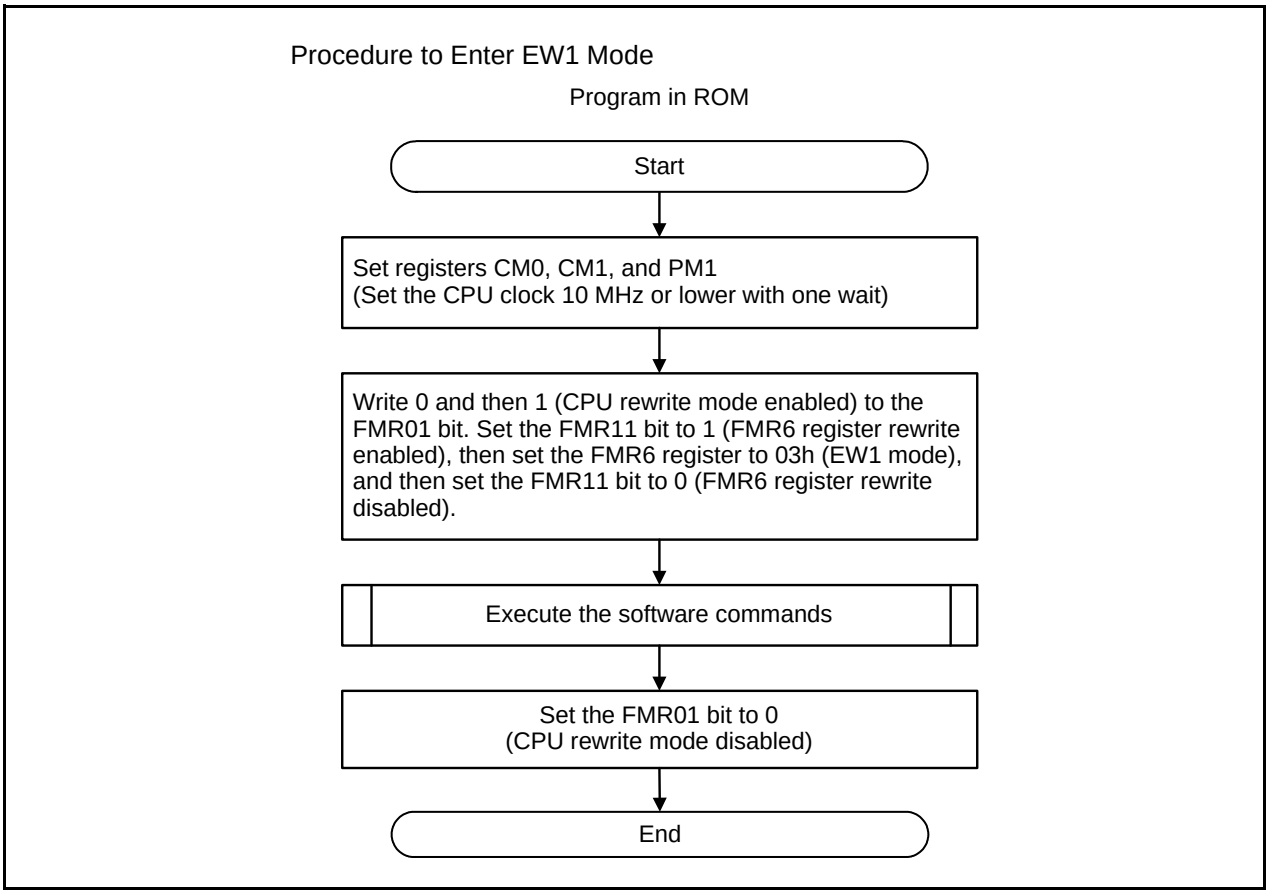


Figure 24.21 Setting and Resetting of EW1 Mode

The following are interrupts which can be used in EW1 mode and operations when the interrupts occur during auto-erase operation or auto-program operation:

- Maskable interrupt (suspend function enabled)
  Auto-erase operation or auto-program operation suspends after td(SR-SUS) elapses and the interrupt process is executed. Auto-erase operation or auto-program operation restarts by setting the FMR31 bit in the FMR3 register to 0 (command restart) after the interrupt process is completed.
- Maskable interrupt (suspend function disabled)
  Auto-erase operation or auto-program operation has a higher priority level and the interrupt request has to wait. The interrupt process is executed after auto-erase operation or auto-program operation is completed.
- $\overline{\text{NMI}}$, watchdog timer, oscillator stop/restart detect, voltage detect 1, and voltage detect 2 interrupts
  Auto-erase operation or auto-program operation forcibly stops as soon as the interrupt occurs, and then the interrupt process starts.
  After the flash memory restart, execute auto-erase operation again and confirm that it is completed as expected in order to read the correct value.

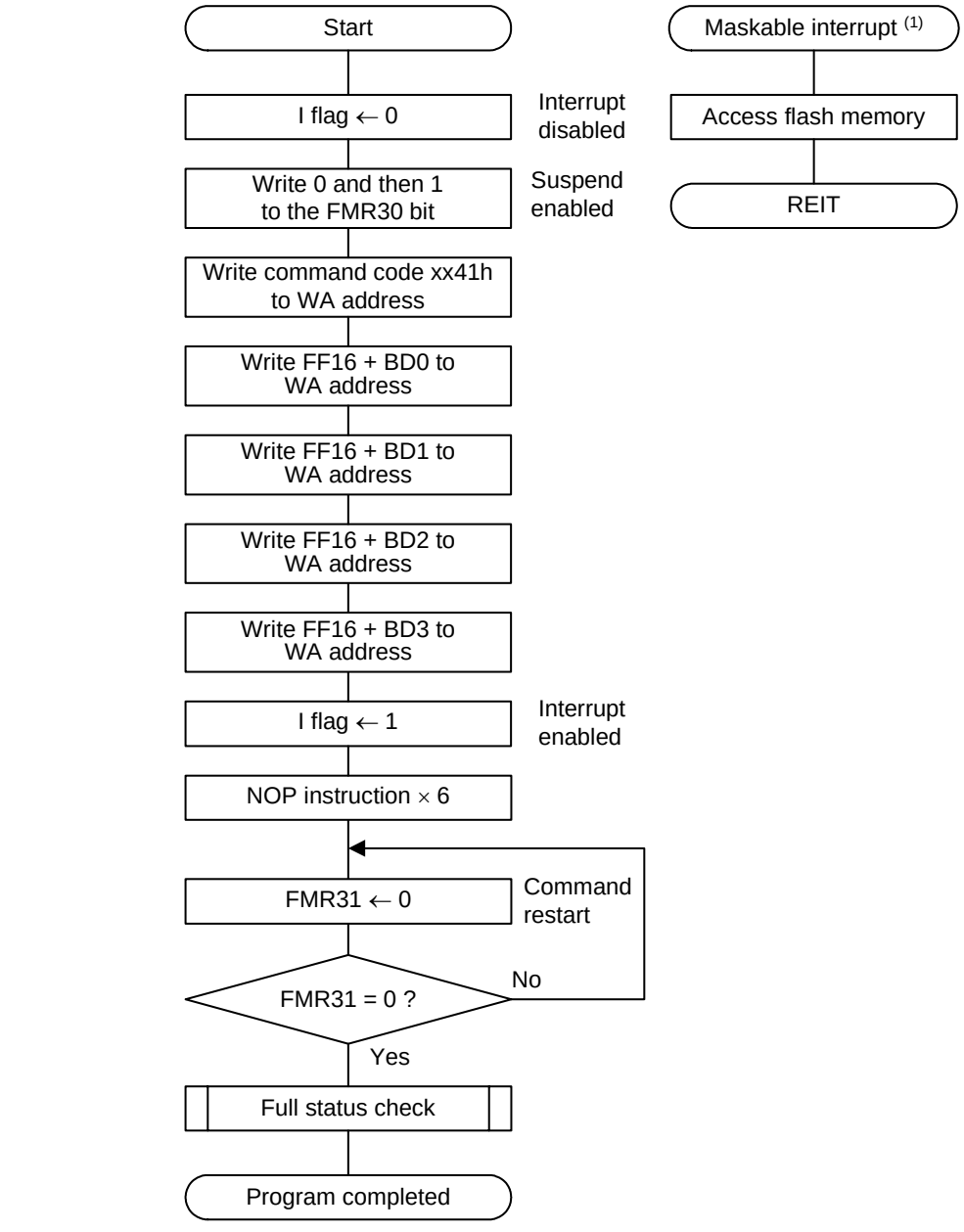
The watchdog timer stops its counting during auto-erase or auto-programming. Do not use EW1 mode while the CSPRO bit in the CSPR register is 1 (count source protection mode enabled). Use EW0 mode. However, the watchdog timer counts during erase suspend or program suspend. The interrupt request can be generated, so initialize the watchdog timer regularly by using the suspend function.

Table 24.20 Modes after Executing Commands (in EW1 Mode)

Command	Mode after Executing Command
Read array	Read array mode
Clear status register	
Program	
Block erase	
Lock bit program	
Read lock bit status	
Block blank check	

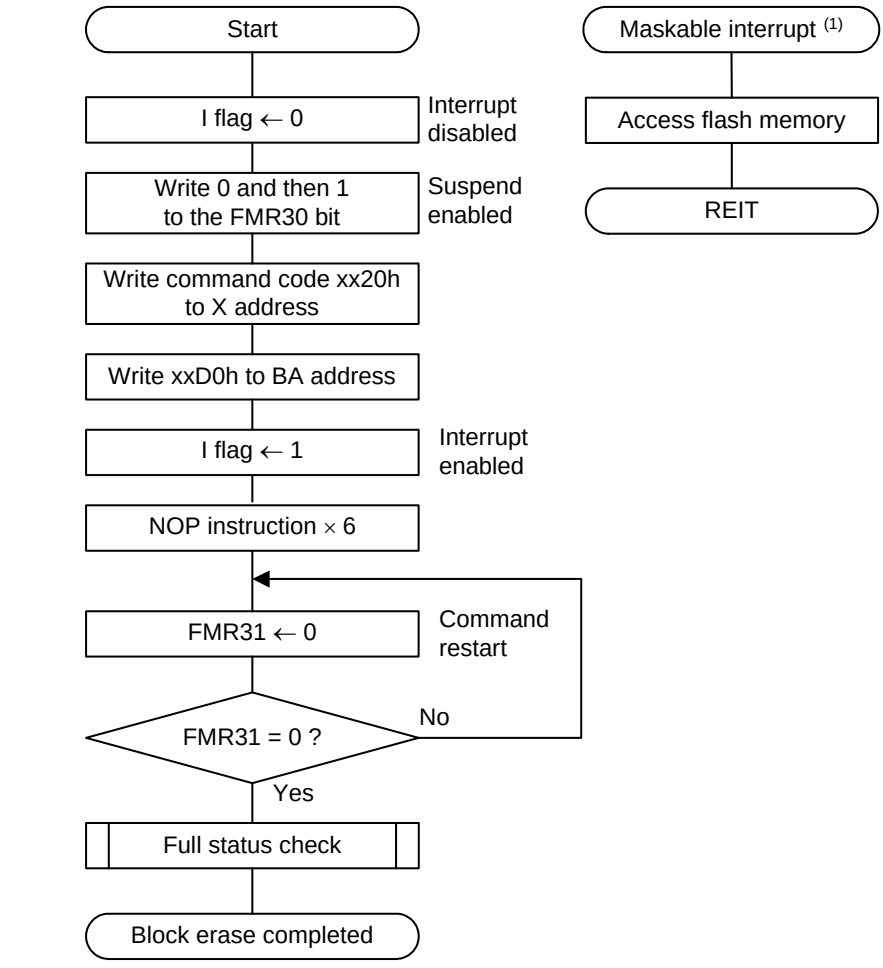
24.8.8.1 Suspend Function (EW1 Mode)

When using suspend function in EW1 mode, an interrupt request is not accepted until td(SR-SUS) elapses after the interrupt request is generated. When the interrupt request is accepted, the flash memory enters erase suspend or program suspend. Set the FMR31 bit to 0 (command restart) to restart automatic program and erase operations at the completion of the interrupt. Figures 24.22 to 24.24 show a flowchart in EW1 mode when the suspend function is enabled, and Figure 24.25 shows Suspend Operation Example in EW1 Mode.



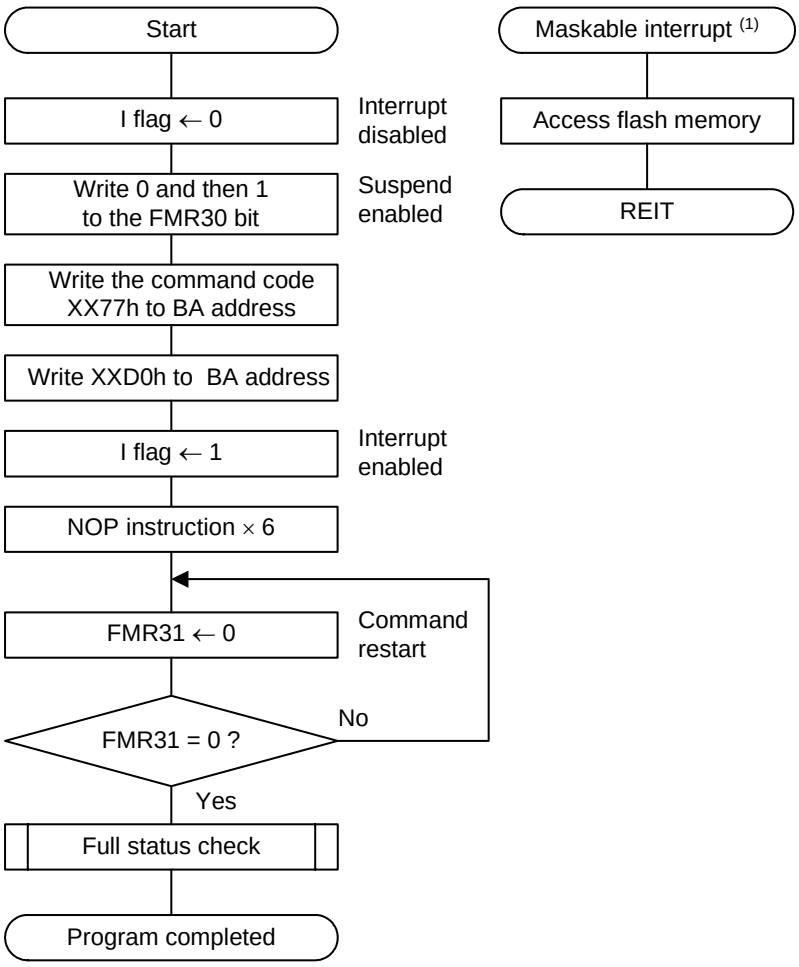
Note:
1. An interrupt request is not accepted until td(SR-SUS) elapses after the interrupt request is generated. Enable interrupts used to enter suspend mode beforehand.

Figure 24.22 Program Flowchart in EW1 Mode (Suspend Function Enabled)



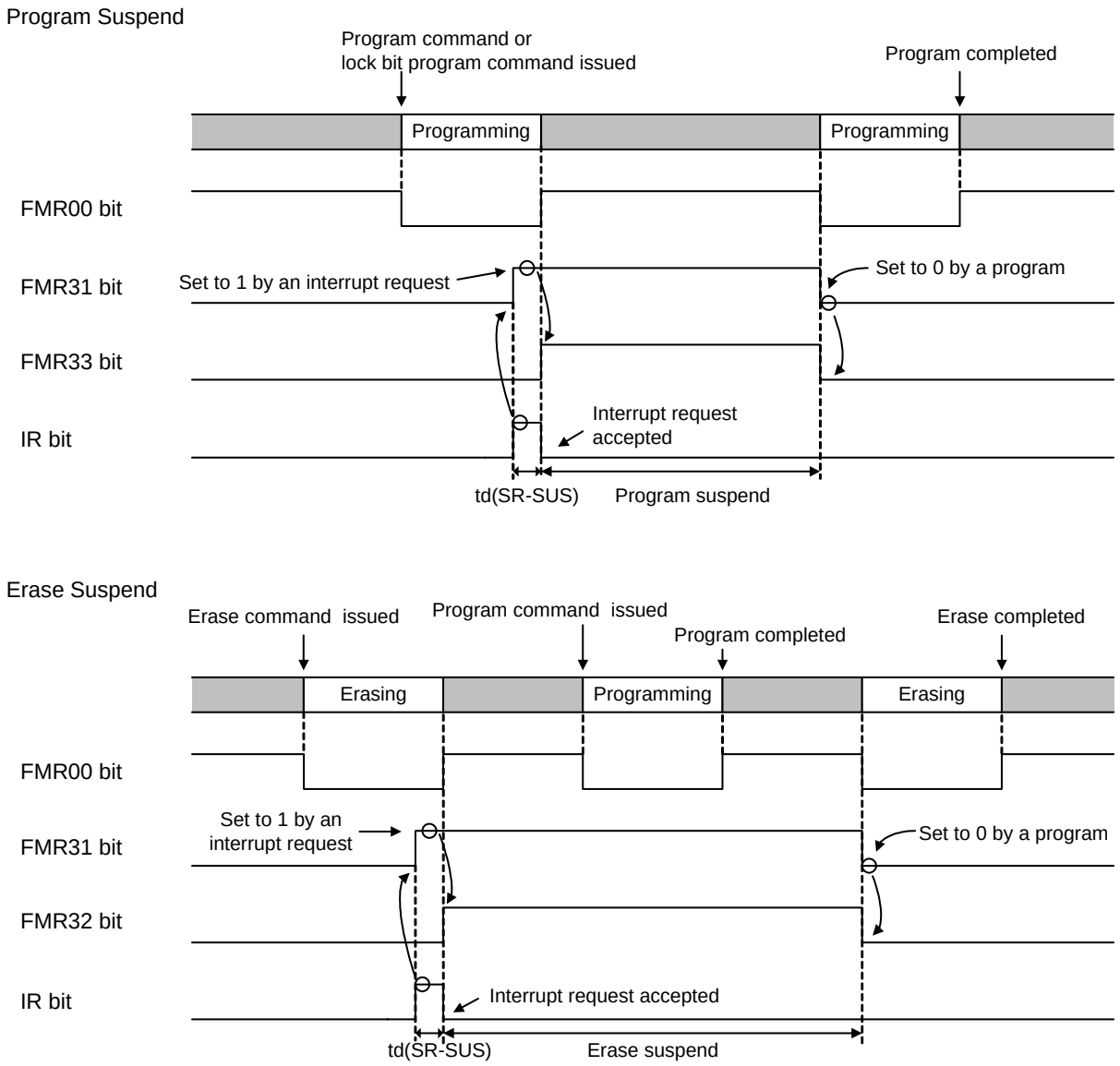
Note:
1. An interrupt request is not accepted until td(SR-SUS) elapses after the interrupt request is generated. Enable interrupts used to enter suspend mode beforehand.

Figure 24.23 Block Erase Flowchart in EW1 Mode (Suspend Function Enabled)



Note:
1. An interrupt request is not accepted until td(SR-SUS) elapses after the interrupt request is generated. Enable interrupts used to enter suspend mode beforehand.

Figure 24.24 Lock Bit Program Flowchart in EW1 Mode (Suspend Function Enabled)



FMR00: Bit in the FMR0 register
FMR33, FMR32, FMR31 : Bits in the FMR3 register
IR: Interrupt control register to enter suspend mode

The above diagram applies under the following condition.
- The FMR30 bit in the FMR3 register is 1 (suspend enabled).

Figure 24.25 Suspend Operation Example in EW1 Mode

24.9 Standard Serial I/O Mode

In standard serial I/O mode, a serial programmer supporting the M16C/6S1 Group can be used to rewrite program ROM 1, program ROM 2, and data flash while the MCU mounted on a board.
Standard serial I/O mode has following modes:

- Standard serial I/O mode 1: The MCU is connected to the serial programmer by using clock synchronous serial I/O
- Standard serial I/O mode 2: The MCU is connected to the serial programmer by using 2-wire clock asynchronous serial I/O

For more information about the serial programmers, contact the serial programmer manufacturer. Refer to the user manual included with your serial programmer for instructions.

24.9.1 ID Code Check Function

Use the ID code check function in standard serial I/O mode. This function determines whether the ID codes sent from the serial programmer match those written in the flash memory. If the ID codes do not match, commands sent from the serial programmer are not accepted. However, if the four bytes of the reset vector are FFFFFFFFh, ID codes are not compared, allowing all commands to be accepted.
The ID codes are 7-byte data stored consecutively, starting with the first byte, at addresses 0FFFDFh, 0FFFE3h, 0FFFEBh, 0FFFEFh, 0FFFF3h, 0FFFF7h, and 0FFFFBh. The flash memory must have a program with the ID codes set in these addresses. Figure 24.26 shows ID Code Storage Addresses.
The ID code of “ALeRASE” in ASCII code is used for forced erase function. The ID code of “Protect” in ASCII code is used for standard serial I/O mode disable function. Table 24.21 lists Reserved Word of ID Code. All ID code storage addresses and data must match the combinations listed in Table 24.21. When the forced erase function or standard serial I/O mode disable function is not used, use another combination of ID codes.

Table 24.21 Reserved Word of ID Code

ID Code Storage Address		Reserved word of ID Code (ASCII)	
		ALeRASE	Protect
FFFDFh	ID1	41h (upper-case A)	50h (upper-case P)
FFFE3h	ID2	4Ch (upper-case L)	72h (lower-case r)
FFFEBh	ID3	65h (lower-case e)	6Fh (lower-case o)
FFFEFh	ID4	52h (upper-case R)	74h (lower-case t)
FFFF3h	ID5	41h (upper-case A)	65h (lower-case e)
FFFF7h	ID6	53h (upper-case S)	63h (lower-case c)
FFFFBh	ID7	45h (upper-case E)	74h (lower-case t)

All ID code storage addresses and data must match the combinations listed in Table 24.21.

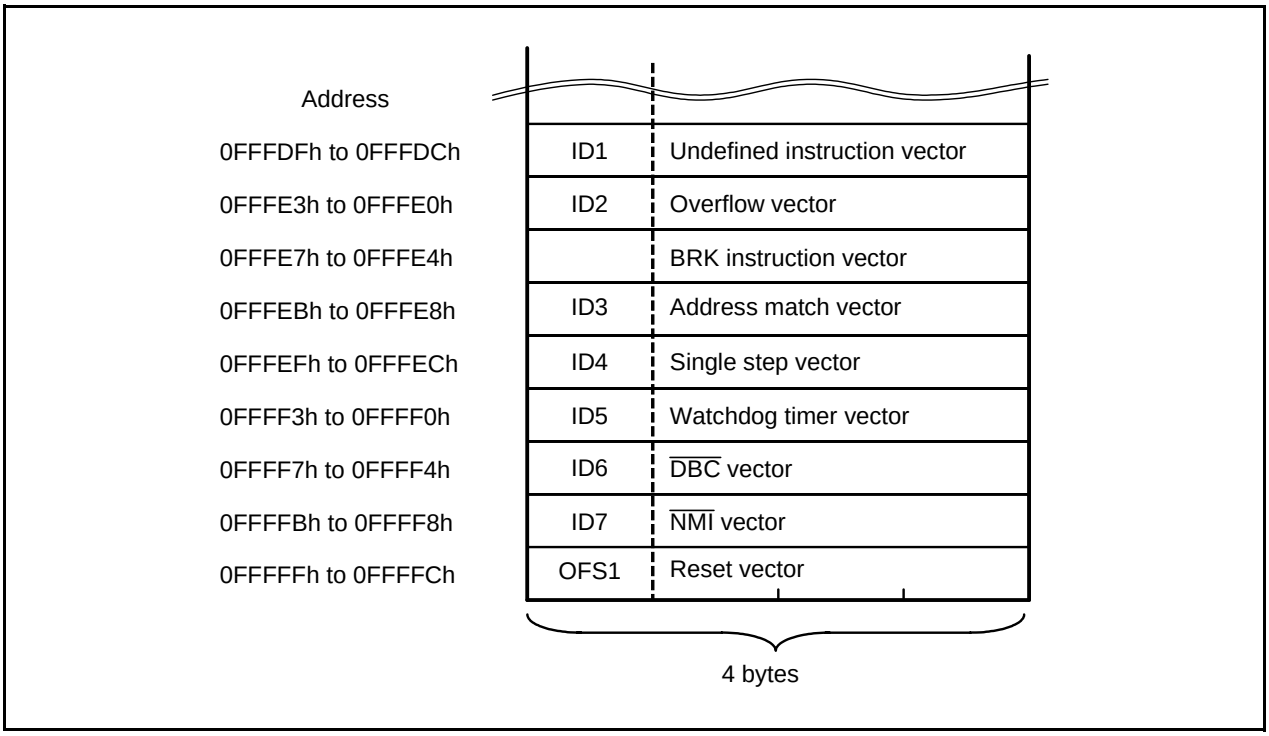


Figure 24.26 ID Code Storage Addresses

24.9.2 Forced Erase Function

Use the forced erase function in standard serial I/O mode. When the reserved word, “ALeRASE” in ASCII code, are sent from the serial programmer as ID codes, the contents of program ROM 1 and program ROM 2 will be erased at once. However, if the ID codes stored in the ID code storage addresses are set to a reserved word other than “ALeRASE” (other than the combination table listed in Table 24.21), the ROMCR bit in the OFS1 address is 1 (ROMCP1 bit enabled), and the ROMCP1 bit in the OFS1 address is 0 (ROM code protect enabled), the forced erase function is ignored and ID code check is executed by the ID code check function. Table 24.22 lists conditions and functions for forced erase function.

When both the ID codes sent from the serial programmer and the ID codes stored in the ID code storage addresses correspond to the reserved word “ALeRASE”, program ROM 1 and program ROM 2 will be erased. However, when the serial programmer sends other than “ALeRASE”, even if the ID codes stored in the ID code storage addresses are “ALeRASE”, there is no ID match and any command is ignored. The flash memory cannot be operated.

Table 24.22 Forced Erase Function

Condition			Function
ID code from serial programmer	Code in ID code storage address	ROMCP1 bit in the OFS1 address	
ALeRASE	ALeRASE	—	Program ROM 1 and program ROM 2 all erase (forced erase function)
	Other than ALeRASE (1)	1 (ROM code protect disabled)	
		0 (ROM code protect enabled)	ID code check (ID code check function)
Other than ALeRASE	ALeRASE	—	ID code check (ID code check function. No ID match)
	Other than ALeRASE (1)	—	ID code check (ID code check function)

Note:
1. For the combination of the stored addresses is “Protect”, refer to 24.9.3 “Standard Serial I/O Mode Disable Function”.

24.9.3 Standard Serial I/O Mode Disable Function

Use the standard serial I/O mode disable function in standard serial I/O mode. When the ID codes in the ID code stored addresses are set to “Protect” in ASCII code (see Table 24.21 “Reserved Word of ID Code”), the MCU does not communicate with the serial programmer. Therefore, the flash memory cannot be read, written or erased by the serial programmer. User boot mode can be selected, when the ID codes are set to “Protect”.

When the ID codes are set to “Protect”, the ROMCR bit in the OFS1 address is 1 (ROMCP1 bit enabled), and the ROMCP1 bit in the OFS1 address is set to 0 (ROM code protect enabled), ROM code protection cannot be disabled by the serial programmer. Therefore, the flash memory cannot be read, written or erased by the serial or parallel programmer.

24.9.4 Standard Serial I/O Mode 1

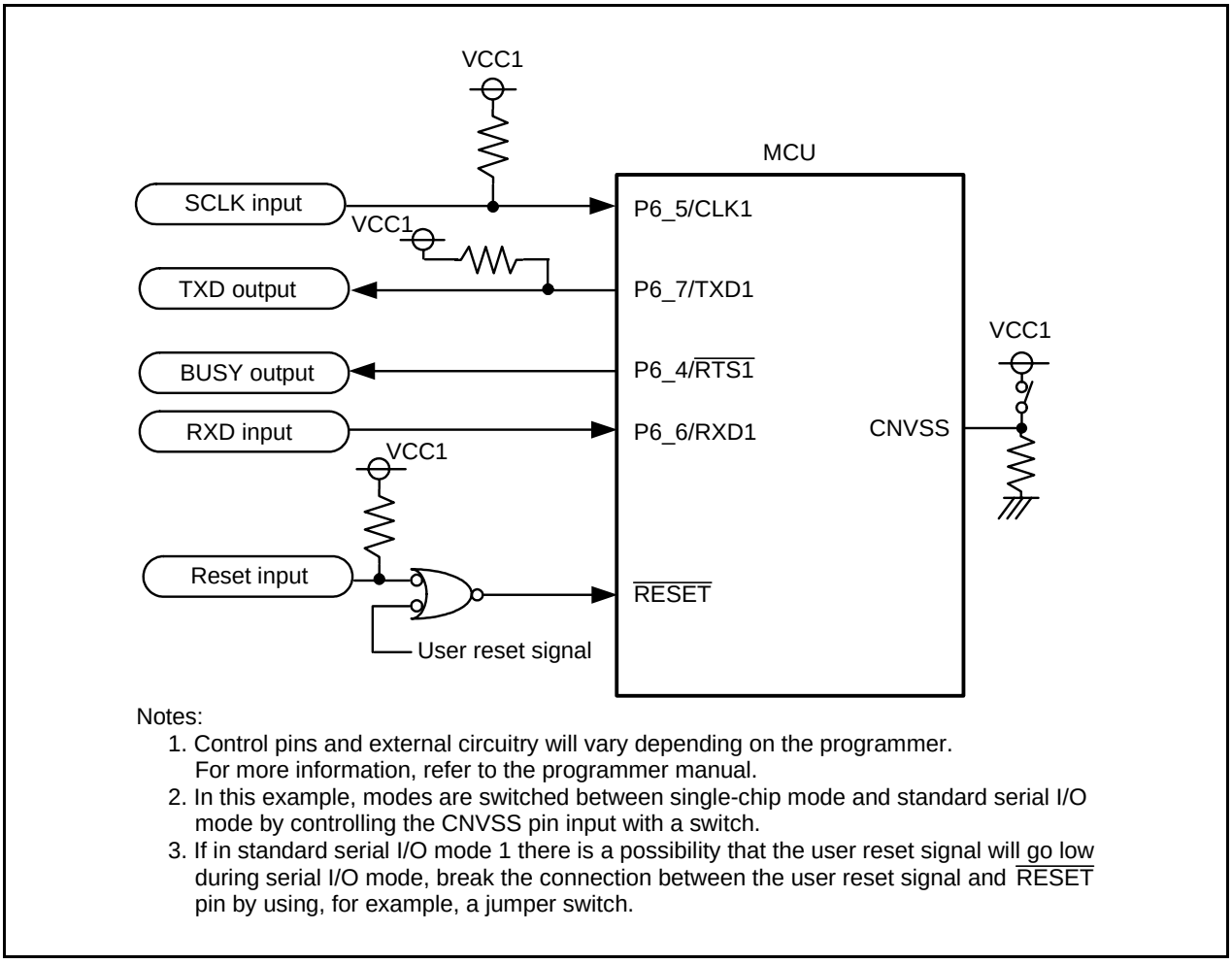
In standard serial I/O mode 1, a serial programmer is connected to the MCU by using clock synchronous serial I/O.

Table 24.23 Pin Functions (Flash Memory Standard Serial I/O Mode 1)

Pin	Name	I/O	Power Supply	Description
VCC1, VCC2, VSS	Power input		—	Apply 3.0 V to 3.6 V to VCC1 and VCC2. The VCC application condition is that VCC2 = VCC1. Apply 0 V to the VSS pin.
CNVSS	CNVSS	I	VCC1	Connect to the VCC1 pin.
$\overline{\text{RESET}}$	Reset input	I	VCC1	Reset input pin.
XIN	Clock input	I	VCC1	Input a high-level signal to the XIN pin and open the XOUT pin when a main clock is not used. Connect a ceramic resonator or crystal between pins XIN and XOUT when the main clock is used. To input an externally generated clock, input it to the XIN pin and open the XOUT pin.
XOUT	Clock output	O		
AVCC, AVSS	Analog power supply input			Connect the AVCC pin to VCC1 and the AVSS pin to VSS, respectively.
VREF	Reference voltage input	I		Reference voltage input pin for A/D converter. When using standard serial I/O mode 1, and power supply to VREF is not supplied, connect with VSS.
P0_0 to P0_7	Input port P0	I	VCC2	Input a high- or low-level signal or leave open.
P1_0 to P1_7	Input port P1	I	VCC2	Input a high- or low-level signal or leave open.
P6_0 to P6_3	Input port P6	I	VCC1	Input a high- or low-level signal or leave open.
P6_4 / $\overline{\text{RTS1}}$	BUSY output	O	VCC1	BUSY signal output pin
P6_5/CLK1	SCLK input	I	VCC1	Serial clock input pin
P6_6 / RXD1	RXD input	I	VCC1	Serial data input pin.
P6_7 / TXD1	TXD output	O	VCC1	Serial data output pin.
P7_0 to P7_7	Input port P7	I	VCC1	Input a high- or low-level signal or leave open.
P8_0 to P8_7	Input port P8	I	VCC1	Input a high- or low-level signal or leave open.
P9_0 to P9_7	Input port P9	I	VCC1	Input a high- or low-level signal or leave open.
P10_0 to P10_7	Input port P10	I	VCC1	Input a high- or low-level signal or leave open.

Table 24.24 Setting of Standard Serial I/O Mode 1

Signal	Input Level
CNVSS	VCC1
$\overline{\text{RESET}}$	VSS → VCC1
SCLK	VCC1



Notes:
1. Control pins and external circuitry will vary depending on the programmer. For more information, refer to the programmer manual.
2. In this example, modes are switched between single-chip mode and standard serial I/O mode by controlling the CNVSS pin input with a switch.
3. If in standard serial I/O mode 1 there is a possibility that the user reset signal will go low during serial I/O mode, break the connection between the user reset signal and $\overline{\text{RESET}}$ pin by using, for example, a jumper switch.

Figure 24.27 Circuit Application in Standard Serial I/O Mode 1

24.9.5 Standard Serial I/O Mode 2

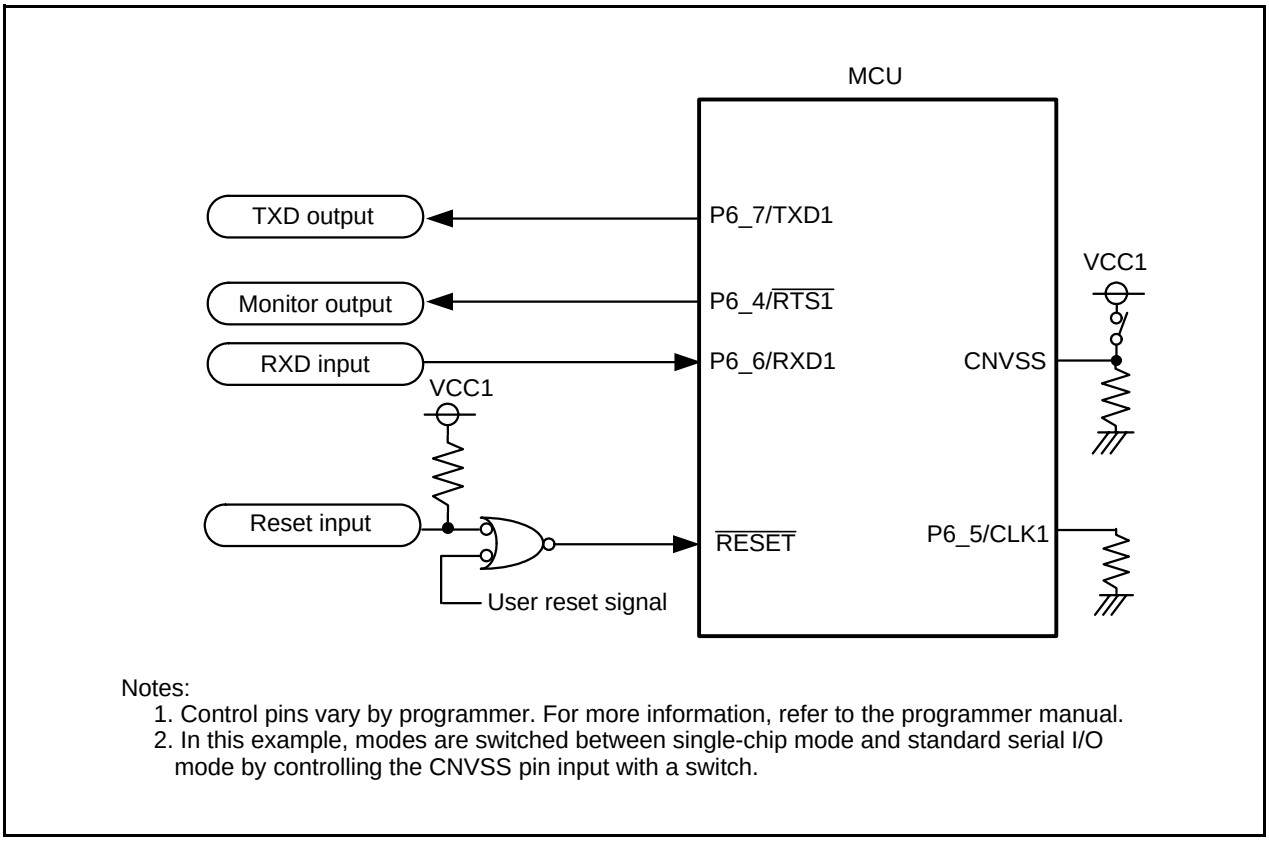
In standard serial I/O mode 2, a serial programmer is connected to the MCU by using 2-wire clock asynchronous serial I/O. The main clock is used.

Table 24.25 Pin Functions (Flash Memory Standard Serial I/O Mode 2)

Pin	Name	I/O	Power Supply	Description
VCC1, VCC2, VSS	Power input		—	Apply the flash memory program and erase voltage to the VCC1 pin, and VCC2 to the VCC2 pin. The VCC application condition is that VCC2 = VCC1. Apply 0 V to the VSS pin.
CNVSS	CNVSS	I	VCC1	Connect to the VCC1 pin.
$\overline{\text{RESET}}$	Reset input	I	VCC1	Reset input pin.
XIN	Clock input	I	VCC1	Connect a ceramic resonator or crystal between pins XIN and XOUT when the main clock is used. To input an externally generated clock, input it to the XIN pin and open the XOUT pin.
XOUT	Clock output	O	VCC1	
AVCC, AVSS	Analog power supply input			Connect the AVCC pin to VCC1 and the AVSS pin to VSS, respectively.
VREF	Reference voltage input	I		Reference voltage input pin for A/D converter. When using standard serial I/O mode 2, and power supply to VREF is not supplied, connect with VSS.
P0_0 to P0_7	Input port P0	I	VCC2	Input a high- or low-level signal or leave open.
P1_0 to P1_7	Input port P1	I	VCC2	Input a high- or low-level signal or leave open.
P6_0 to P6_3	Input port P6	I	VCC1	Input a high- or low-level signal or leave open.
P6_4 / $\overline{\text{RTS1}}$	BUSY output	O	VCC1	Monitor signal output pin for checking the boot program operation.
P6_5/CLK1	SCLK input	I	VCC1	Input a low-level signal
P6_6 / RXD1	RXD input	I	VCC1	Serial data input pin.
P6_7 / TXD1	TXD output	O	VCC1	Serial data output pin.
P7_0 to P7_7	Input port P7	I	VCC1	Input a high- or low-level signal or leave open.
P8_0 to P8_7	Input port P8	I	VCC1	Input a high- or low-level signal or leave open.
P9_0 to P9_7	Input port P9	I	VCC1	Input a high- or low-level signal or leave open.
P10_0 to P10_7	Input port P10	I	VCC1	Input a high- or low-level signal or leave open.

Table 24.26 Setting of Standard Serial I/O Mode 2

Signal	Input Level
CNVSS	VCC1
$\overline{\text{RESET}}$	VSS → VCC1
P6_5/CLK1	VSS



Notes:
1. Control pins vary by programmer. For more information, refer to the programmer manual.
2. In this example, modes are switched between single-chip mode and standard serial I/O mode by controlling the CNVSS pin input with a switch.

Figure 24.28 Circuit Application in Standard Serial I/O Mode 2

24.10 Parallel I/O Mode

In parallel I/O mode, program ROM 1, program ROM 2, and data flash can be rewritten using a parallel programmer supporting the M16C/6S1 Group. Contact the parallel programmer manufacturer for more information. Refer to the user manual included with your parallel programmer for instructions.

24.10.1 ROM Code Protect Function

The ROM code protect function inhibits the flash memory from being read or rewritten during parallel I/O mode. Refer to 24.4.1 “Optional Function Select Address 1 (OFS1)”. The OFS1 address is located in block 0 in program ROM 1.

When the ROMCR bit in the OFS1 address is 1 (ROMCP1 bit enabled) and the ROMCP1 bit is set to 0, the ROM code protect function is enabled.

To cancel ROM code protect, erase block 0 including the OFS1 address using standard serial I/O mode or CPU rewrite mode.

24.11 Notes on Flash Memory

24.11.1 OFS1 Address and ID Code Storage Address

The OFS1 address and ID code storage address are part of flash memory. When writing a program to flash memory, write an appropriate value to those addresses simultaneously.

In the OFS1 address, the MCU state after reset and the function to prevent rewrite in parallel I/O mode are selected. The OFS1 address is 0FFFFFh. This is the most significant address of block 0 in program ROM 1 and upper address of reset vector. Also, the ID code storage address is in block 0 and upper address of the interrupt vector.

The ID code check function cannot be disabled. Even if the protect using the ID code check function is unnecessary, input the appropriate ID code when using a serial programmer or debugger. Without the appropriate ID code, the serial programmer or debugger cannot be used.

ex) Set FEh to the OFS1 address

When using an address control instruction and logical addition:

```
.org 0FFFFCh
RESET:
.lword start | 0FE000000h
```

When using an address control instruction:

```
.org 0FFFFCh
RESET:
.addr start
.byte 0FEh
```

(Program format varies depending on the compiler. Refer to the compiler manual.)

24.11.2 Reading Data Flash

When 2.7 V ≤ VCC1 ≤ 3.0 V, one wait must be inserted to execute the program on the data flash and read the data. Set the PM17 in the PM1 register or FMR17 bit in the FMR1 register to insert one wait.

24.11.3 CPU Rewrite Mode

24.11.3.1 Operating Speed

Set a CPU clock frequency of 10 MHz or less by the CM06 bit in the CM0 register and bits CM17 and CM16 in the CM1 register before entering CPU rewrite mode (EW0 or EW1 mode). Also, set the PM17 bit in the PM1 register to 1 (wait state).

24.11.3.2 CPU Rewrite Mode Select

Change FMR01 bit in the FMR0 register, FMR11 bit in the FMR1 register, and FMR60 bit in the FMR6 register while in the following state:

- PM24 bit in the PM2 register is 0 ($\overline{\text{NMI}}$ interrupt disabled).
- High is input to the $\overline{\text{NMI}}$ pin.

Change the FMR60 bit while the FMR00 bit in the FMR0 register is 1 (ready).

24.11.3.3 Prohibited Instructions

Do not use the following instructions in EW0 mode:
UND instruction, INTO instruction, JMPS instruction, JSRS instruction, and BRK instruction.

24.11.3.4 Interrupts (EW0 Mode and EW1 Mode)

- Do not use an address match interrupt during command execution because the address match interrupt vector is located in ROM.
- Do not use a non-maskable interrupt during block 0 erase because fixed vector is located in block 0.

24.11.3.5 Rewrite (EW0 Mode)

If the power supply voltage drops while rewriting the block where the rewrite control program is stored, the rewrite control program is not correctly rewritten. This may prevent the flash memory from being rewritten. If this error occurs, use standard serial I/O mode or parallel I/O mode for rewriting.

24.11.3.6 Rewrite (EW1 Mode)

Do not rewrite any blocks in which the rewrite control program is stored.

24.11.3.7 DMA transfer

In EW0 mode, do not use flash memory as a source of the DMA transfer.
In EW1 mode, do not generate a DMA transfer while the FMR00 bit in the FMR0 register is set to 0 (auto programming or auto erasing).

24.11.3.8 Wait Mode

To enter wait mode, set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled) before executing the WAIT instruction.

24.11.3.9 Stop Mode

To enter stop mode, set the FMR01 bit to 0 (CPU rewrite mode disabled), and then disable DMA transfer before setting the CM10 bit in the CM 1 register to 1 (stop mode).

24.11.3.10 Software Command

Observe the notes below when using the following commands.

- Program
- Block erase
- Lock bit program
- Read lock bit status
- Block blank check

(a) The FMR00 bit in the FMR0 register indicates the status while executing these commands. Do not execute other commands while the FMR00 bit is 0 (busy).
(b) Use these commands in high-speed mode and medium-speed mode. Do not change clock modes while the FMR00 bit in the FMR0 register is 0 (busy).
(c) After executing the program, block erase, or lock bit program command, perform a full status check per one command (i.e. do not perform a single full status check after multiple commands are executed).
(d) Do not execute the program, block erase, lock bit program, or block blank check command when either or both the FMR06 bit in the FMR0 register and the FMR75 bit in the FMR7 register are 1 (completed in error).
(e) Do not use these commands in slow read mode (when the FMR22 bit is 1) or low current consumption read mode (when both bits FMR22 and FMR23 are 1).

24.11.3.11 Area Where the Rewrite Control Program is Executed

The PM10 bit in the PM1 register become 1 in CPU rewrite mode. Execute the rewrite program in internal RAM.

24.11.3.12 Program and Erase Cycles and Execution Time

Execution time of the program, block erase, and lock bit program commands becomes longer as the number of programming and erasing increases.

24.11.3.13 Suspending the Auto-Erase and Auto-Program Operations

When the program, block erase, and lock bit program commands are suspended, the blocks for those commands must be erased. Execute the program and lock bit program commands again after erasing.

Those commands are suspended by the following reset or interrupts:

- Hardware, voltage monitor 0, oscillation stop detect, watchdog timer, software resets.
- NMI, watchdog timer and oscillation stop/restart detect interrupts.

24.11.4 User Boot Mode

24.11.4.1 User Boot Mode Program

Note the following when using user boot mode:

- When using user boot mode, make sure to allocate the program to be executed to program ROM 2.
- Bits VDSEL1 and LVDAS in the OFS1 address are disabled in boot mode.
- When restarting the MCU in user boot mode after starting it in user boot mode, RAM becomes undefined.
- If addresses 13FF8h to 13FFBh are all 00h, the MCU does not enter standard serial I/O mode. Therefore, the programmer or on-chip debugger cannot be connected.
- As the reset sequence differs, the time necessary for starting the program is longer than in single-chip mode.
- Functions in user boot mode cannot be debugged by the on-chip debugging emulator or full spec emulator.
- While using user boot mode, do not change the input level of the pin used for user boot entry. However, if there is a possibility that the input level may change, perform the necessary processes in user boot mode, then restart the MCU in single-chip mode before the input level changes.
- To use user boot mode after standard serial I/O mode, turn off the power when exiting standard serial I/O mode, and then turn on the power again (cold start). The MCU enters user boot mode under the right conditions.

24.11.5 EW1 Mode

(Technical update number: TN-16C-A175A/E)
Adhere to the following when using EW1 mode:

24.11.5.1 Frequency Limitation of EW1 Mode

Set the CPU clock to 1 MHz or higher when using EW1 mode.

24.11.5.2 Frequency Limitation of Block Blank Check Command

Set the CPU clock to 3 MHz or higher when using the block blank check command.

24.11.5.3 Disabling the Lock Bit

Set the FMR02 bit in the FMR0 register to 1 (lock bit disabled).
Do not execute the read lock bit status command or lock bit program command.

24.11.5.4 Entering EW1 Mode in the User Program Using Wait or Stop Mode

When using EW1 mode in the user program in which the MCU enters wait mode or stop mode, set the FMSTP bit in the FMR0 register to 1 (flash memory off) on RAM. Then, set the FMSTP bit to 0 (flash memory on) again and enter the EW1 mode on flash memory. Execute these processes while an interrupt is disabled.

25. PLC Modem Core

25.1 Introduction

This MCU integrates a power line communication (PLC) modem core and an analog front end. A Block Diagram of Inside of LSI is shown in Figure 25.1.

In the digital block of the PLC modem core, there are two built-in modem cores, IT700 and IT900.

The IT700 modem core enables transmission/reception using the DCSK technique, providing communication compatibility with the IT800 modem core built in the M16C/6S Group MCU.

The IT900 modem core enables communication using the DCSK turbo technique, allowing faster communication than DCSK mode.

The IT900 and IT700 modem cores share a single analog front end. The DCSK turbo modem coexists with the DCSK modem. The IT900 and IT700 modem cores are controlled by hardware DLL, and perform transmission/reception correctly even if the DCSK and DCSK turbo packets are mixed in the communication path.

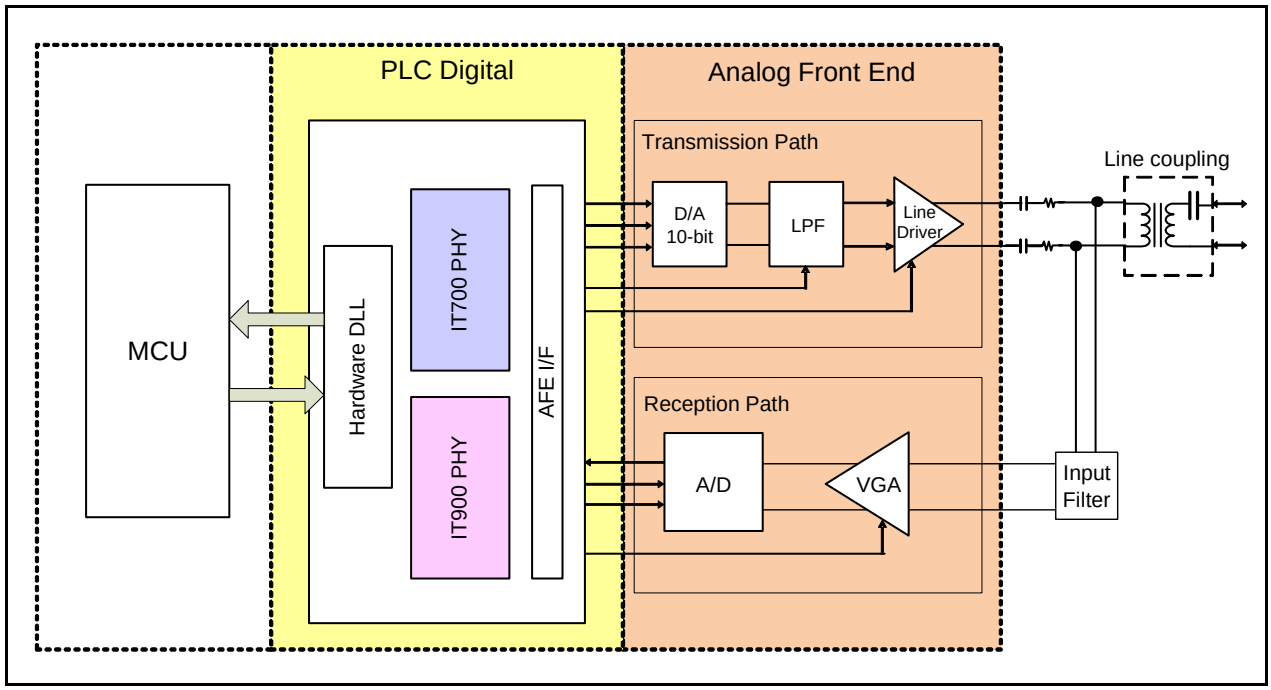


Figure 25.1 Block Diagram of Inside of LSI

25.2 Connection between PLC Modem Core and MCU

The connection between the MCU and the PLC modem core is shown in the figure below.
The PLC modem core and the MCU are connected through ports (PORT2 to 5), interrupts ($\overline{INT0}$, $\overline{INT6}$, and $\overline{INT7}$), and timer A0 (TA0IN and TA0OUT). Transmit/receive data is transferred between RAM and UART7 using DMA0, and communication is performed with the PLC modem core in clock synchronous I/O mode. At this time, $\overline{INT0}$ is used as a trigger for DMA transfer.
Since these MCU peripheral functions are used by the PLC modem core and the driver software, they cannot be used by the user.

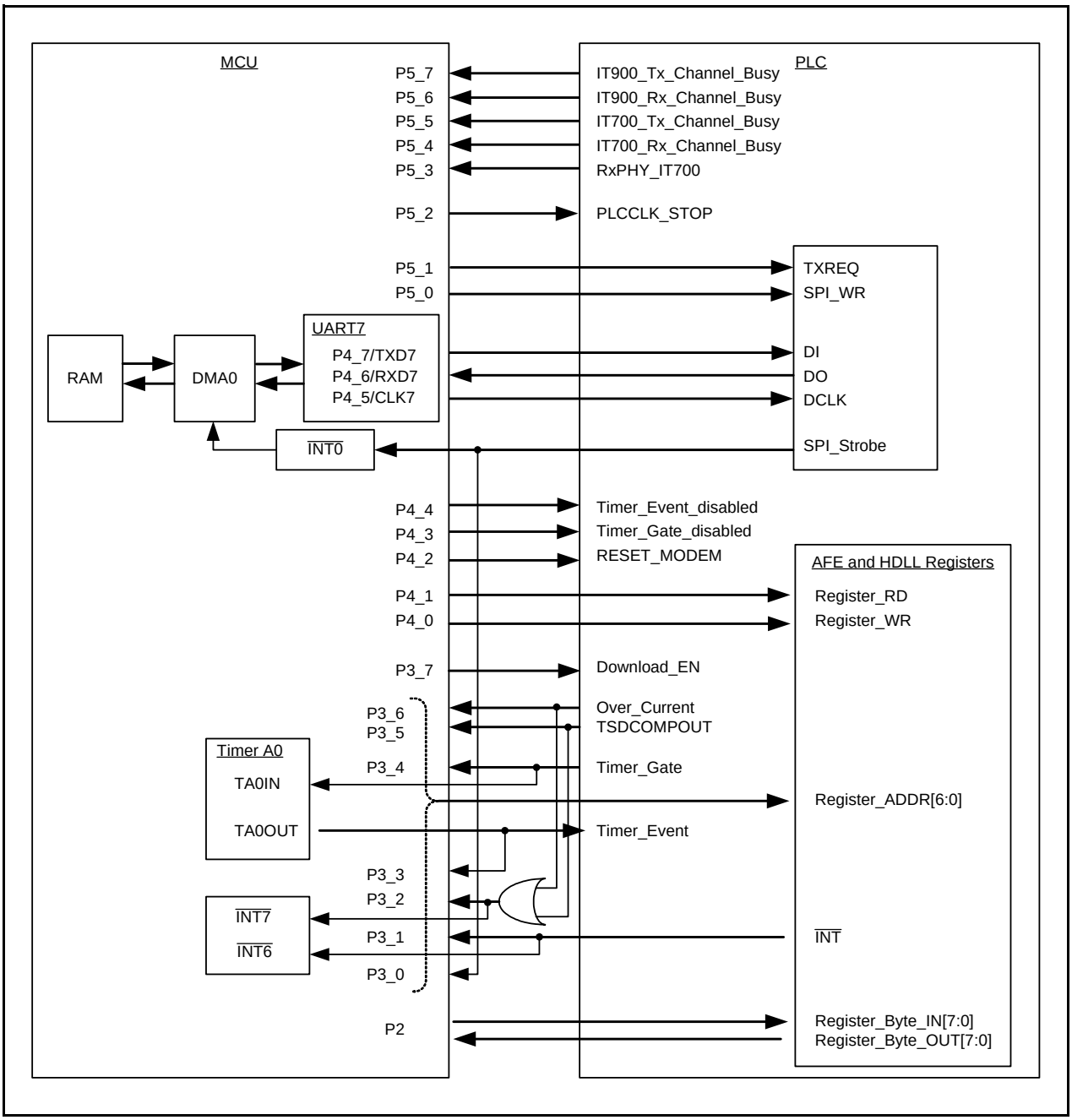


Figure 25.2 Connection between PLC modem Core and MCU

25.3 Registers in PLC Block

The AFE and hardware DLL registers can be read or written from the MCU, using PORT3, PORT2, PORT4_1, and PORT4_0.
The hardware DLL registers are allocated to addresses 0x00 to 0x6F specified by Register_ADDR.
Events generated in the PLC modem core are transmitted to the MCU using $\overline{\text{INT6}}$.
$\overline{\text{INT7}}$ is used to transmit signals from the protection circuit (for the temperature and overcurrent sensors) inside the AFE.
The AFE registers are allocated to addresses 0x70 to 0x77 specified by Register_ADDR.
These registers and interrupts are used by driver software provided from Renesas.

26. Analog Front End (AFE)

26.1 Introduction

Analog front end (AFE) is the circuit located between the PLC digital block and a power line.

There are the following two signal paths in the AFE.

- Transmission path:
  Consists of a DAC driven by the PLC digital block, a low-pass filter (LPF), and a differential line driver amplifier (LD) and a line coupling circuit which drive the power line.
- Reception path:
  Consists of a line coupling circuit, an input filter, a differential variable gain amplifier (VGA), a low-pass filter (LPF), and an ADC. The line coupling circuit is common to both transmission and reception.

The M16C/6S1 Group integrates a DAC, LPF, LD, VGA, and ADC. It is possible to connect to an external line driver without using the on-chip LD.
A block diagram of the analog front end circuit is shown in Figure 26.1.

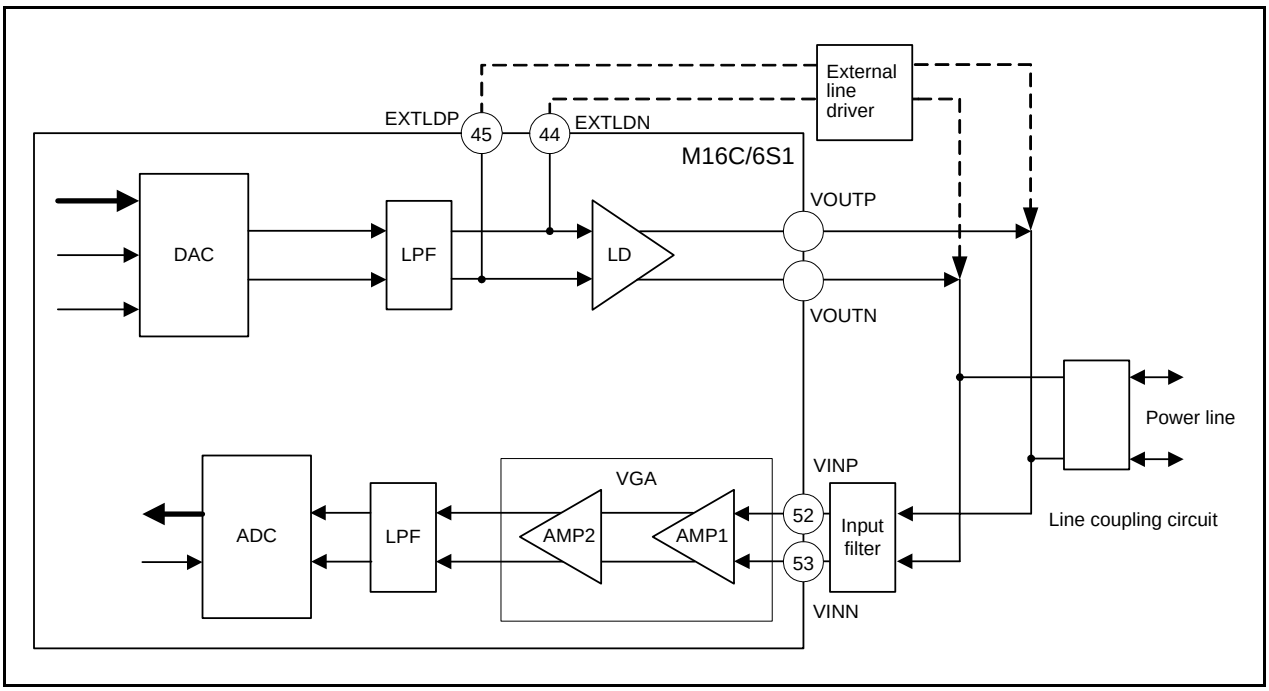


Figure 26.1 Analog Front End Circuit Block Diagram

26.2 Transmission Path

The characteristics required for the communication path are as follows:

- A suitable output signal level should be obtained on rated load.
- Frequency characteristics should be flat in signal band.
- The out-of-band signal level is within regulations.

This system assumes operation in the following three basic signal bands.

- The U.S. and Japan: 120 kHz to 400 kHz
- Europe: 95 kHz to 125 kHz (CENELEC B Band)
- Europe: 20 kHz to 80 kHz (CENELEC A Band)

Signal bands can be switched by changing the configuration and output amplitude of the PLC modem core, and adjusting the analog filter to the signal band.
Refer the following standards and for signal level and out-of-band conditions.

- U.S.: FCC standard, part 15
- Europe: CENELEC standard, EN 50065-1
- Japan: ARIB, STD-T84

Provide sufficient protection for the coupling circuit to ensure that a signal surge from the power line or a signal that exceeds the allowable range does not enter the M16C/6S1 device.

A transmitter analog circuit which is built into the M16C/6S1 Group is shown in Figure 26.2.

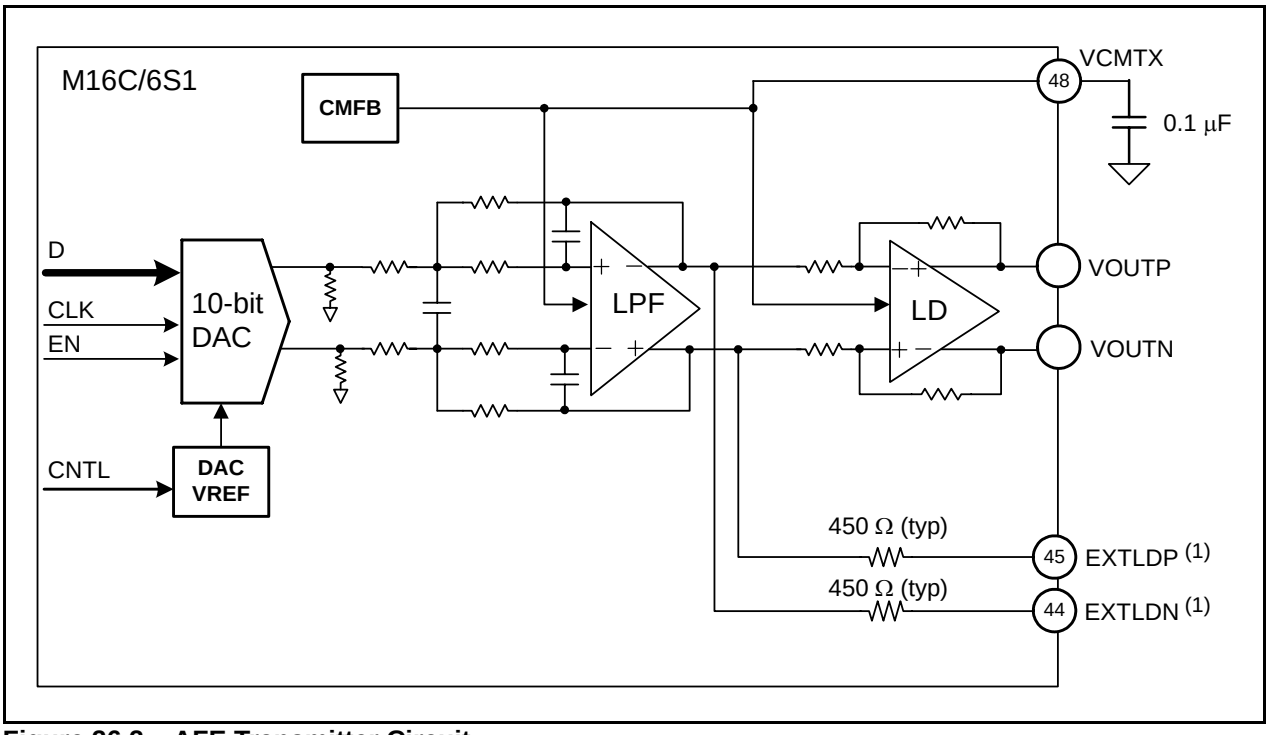
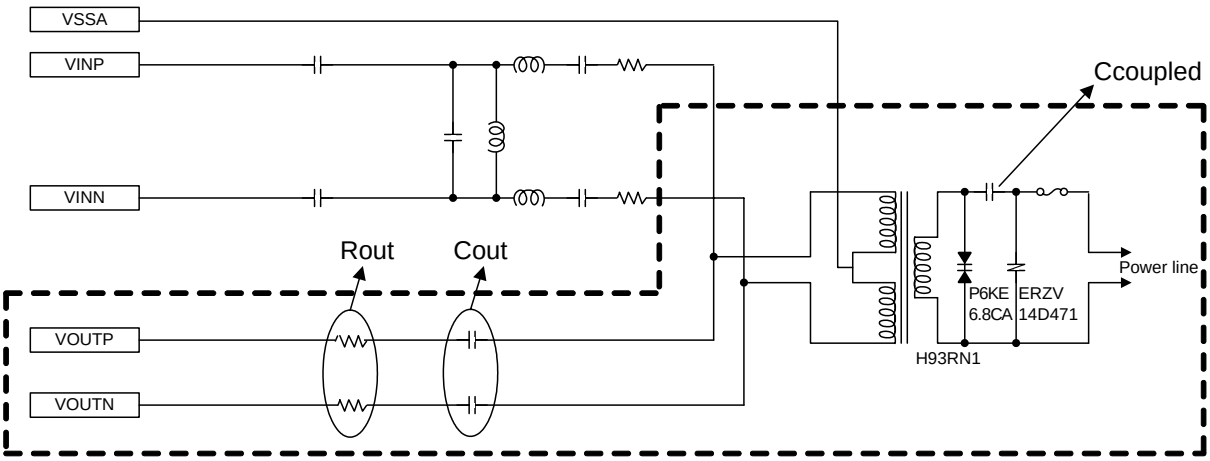


Figure 26.2 AFE Transmitter Circuit

Note:

1. When not using an external line drive amplifier, leave EXTLDP and EXTLDN open.

A recommended peripheral circuit for the transmitter is shown in Figure 26.3.



Recommended Values

Standard	Rout (1)	Cout	Ccoupled
FCC, ARIB	1.5 Ω	4.7 μF	0.22 μF
CENELEC A	1.5 Ω	10 μF	1 μF
CENELEC B	1.5 Ω	10 μF	0.47 μF

Note:
1. Ensure Rout resistor ≥ 0.5 Ω

Figure 26.3 Recommended Peripheral Circuit of Transmitter

26.2.1 Adjustment of Output Amplitude

The transmission output amplitude can be changed by changing the DACVREF register setting in the AFE registers. The typical characteristics of DACVREF code (4 bits) and amplitude (when unloaded) are shown in Figure 26.4.
The DACVREF register is set by driver software provided from Renesas.
For reference, the typical characteristics of load resistance and transmission output amplitude are shown in Figure 26.5.
The output amplitude on the actual power line changes depending on an external circuit or line impedance.

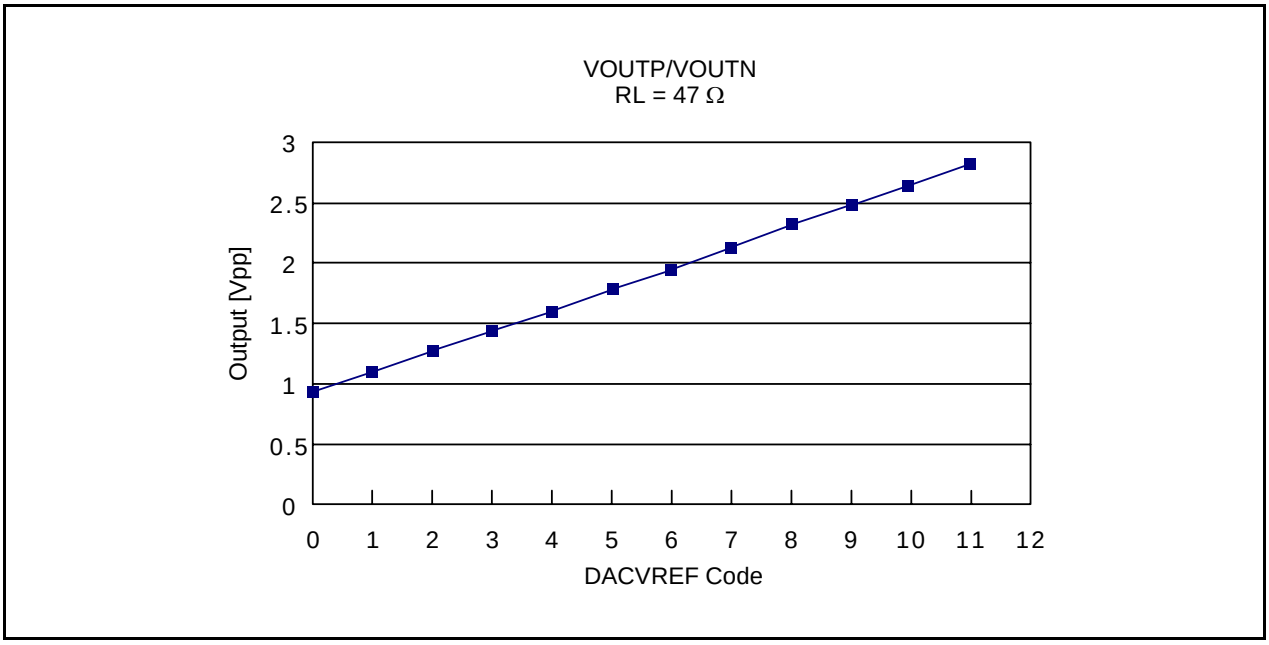


Figure 26.4 DACVREFF and Line Driver Output Amplitude Characteristics

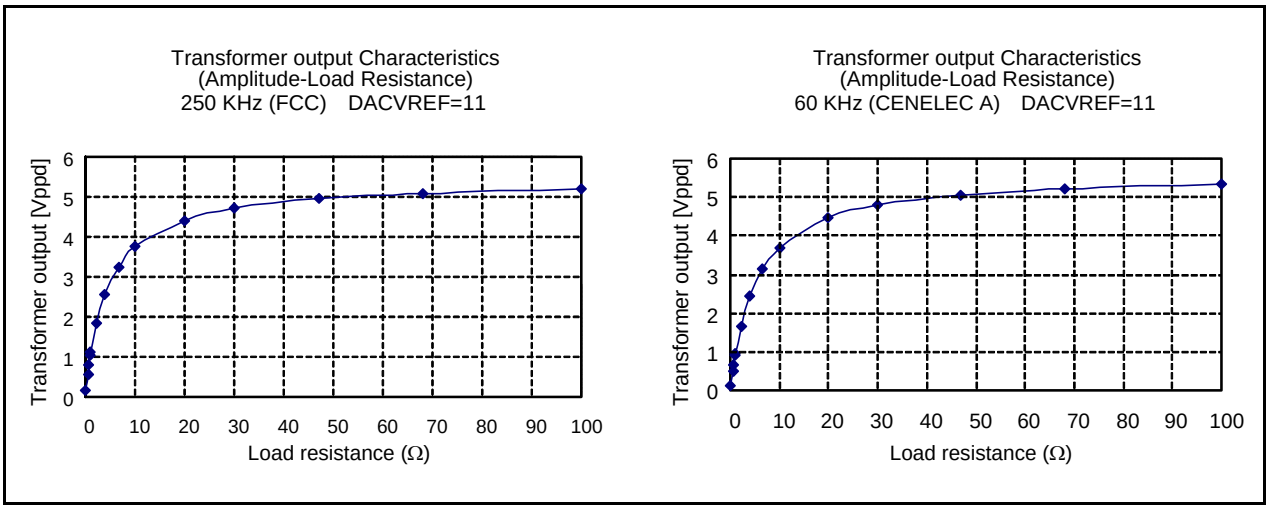


Figure 26.5 Line Driver Transfer R-V Typical Characteristics (Measuring Condition: Rout = 1 Ω as in Figure 26.3)

Note:
1. When load resistance is small, the on-chip line driver generates more heat during transmit operation. Ensure that transmit operation is performed in a range in which the package surface temperature does not exceed 125°C.

26.2.2 Short-circuit Protection (Overcurrent Protection Circuit)

When the on-chip line driver of the M16C/6S1 Group is in use, if DC overcurrent flows due to a short across LSI pins, the on-chip protection circuit stops the line drive output.
The OC_EN pin enables/disables the overcurrent protection circuit.
Set the OC_EN pin to high to enable the overcurrent protection circuit.
Once the overcurrent protection circuit operates, no transmission can be performed until the overcurrent protection circuit is reset.
The overcurrent protection circuit is reset by setting the OC_EN pin to low or by an external reset.
Also, the overcurrent protection circuit can be reset by software.

26.2.3 Peripheral Circuit When Using External Line Driver

When using a driver circuit which is located externally, connect pins EXTLDP and EXTLDN to the external driver input.
An example of external line driver connection is shown in Figure 26.6.
When using the external line driver, it is necessary to set the on-chip registers appropriately.
These settings are set by driver software provided from Renesas.
Pins EXTLDP and EXTLDN have an internal resistor for protection. Add an internal resistor value when calculating a gain external driver amplifier.

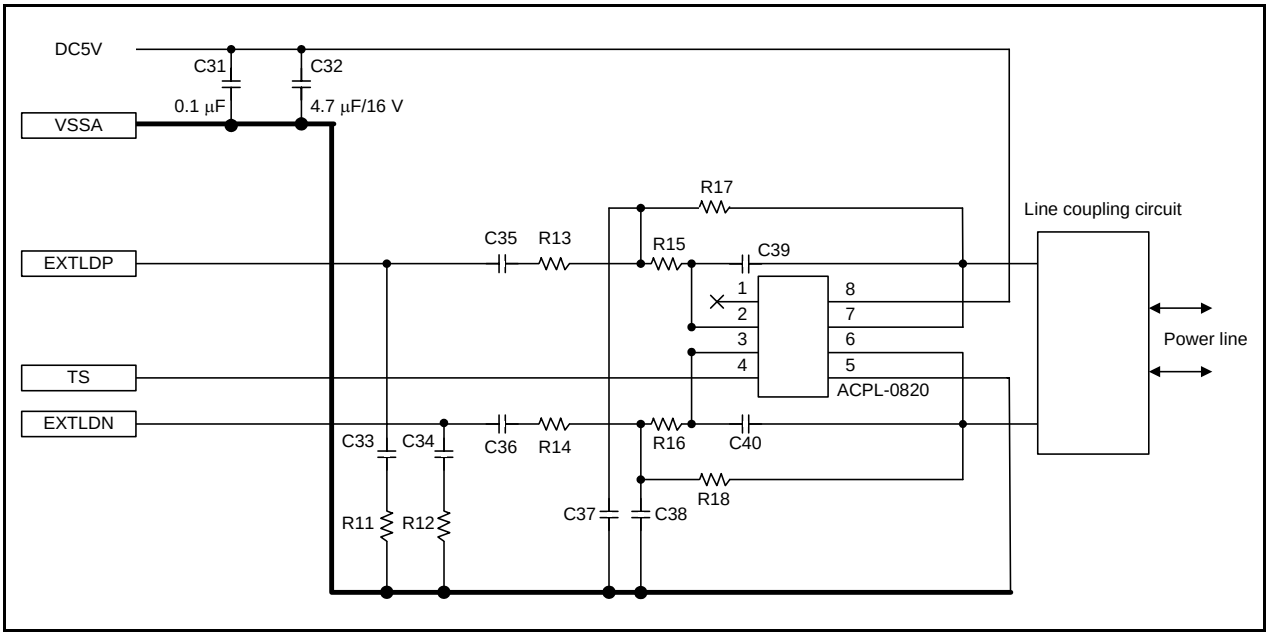


Figure 26.6 External Line Driver Connection Example (LSI in the figure is ACPL-0820 by AVAGO Technologies)

Internal resistance of EXTLDP/EXTLDN: 450 Ω (typical value)

26.3 Receive path

A receiver analog circuit which is built into the M16C/6S1 Group is shown in Figure 26.7.
Signals which are differentially input from VINP and VINN are amplitude-adjusted by the VGA, converted to digital signals by the ADC, and then input to the PLC digital block.
The VGA gain is automatically set to an appropriate gain by the PLC digital block.
The reception performance is improved by placing a suitable band-path filter to reduce out-of-band noise before inputting a signal to the on-chip VGA.
A recommended input filter is shown in Figure 26.8.

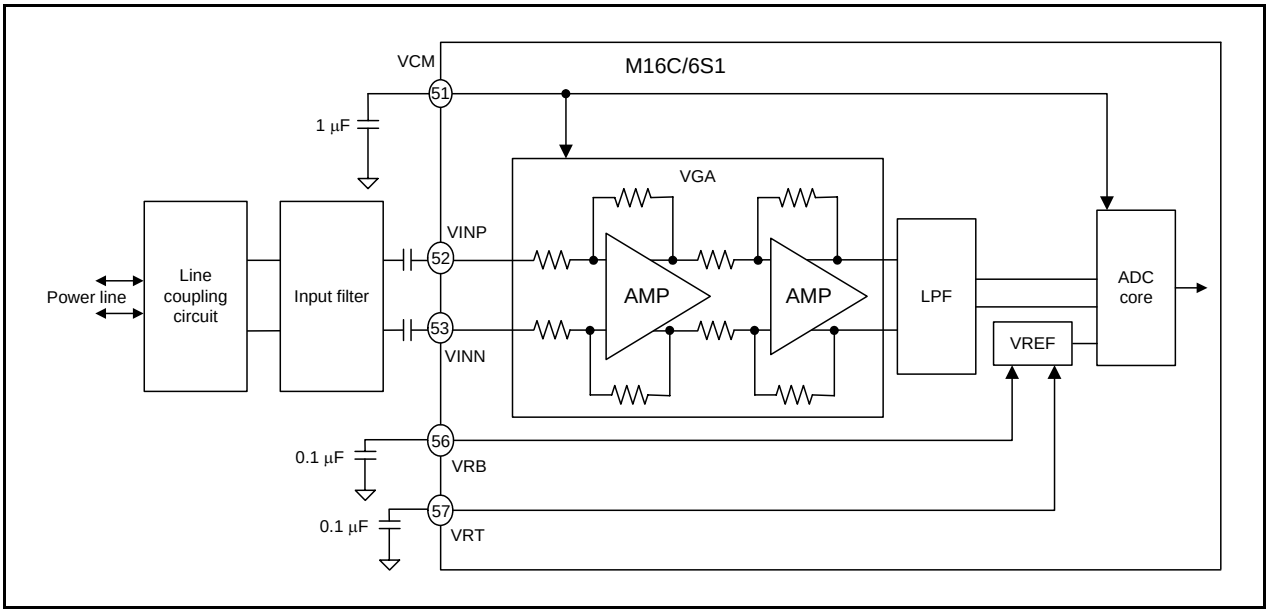


Figure 26.7 AFE Receiver Analog Circuit

	CA	CB	ARIB	FCC
C24, C25	100 nF	10 nF	18 nF	18 nF
C26	16 nF	27 nF	3.9 nF	3.9 nF
L1, L2	150 µH	220 µH	33 µH	33 µH
L3	1.2 mH	82 µH	150 µH	150 µH

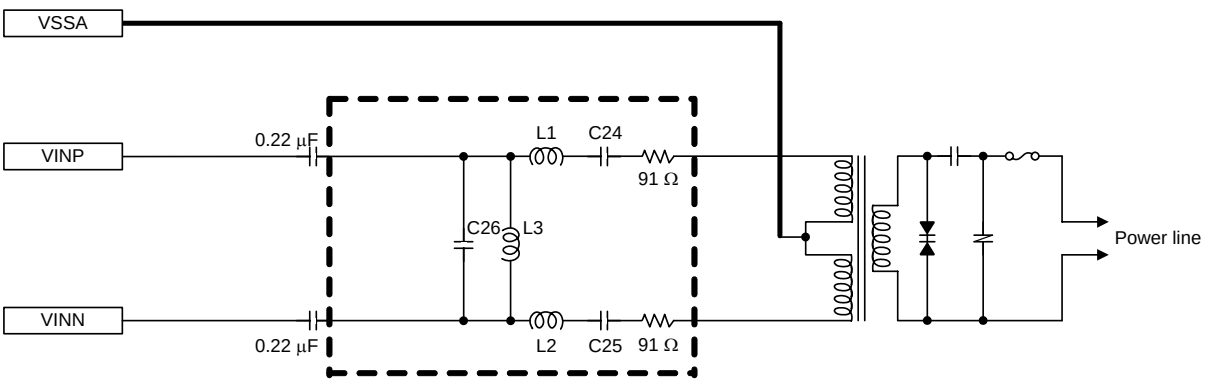
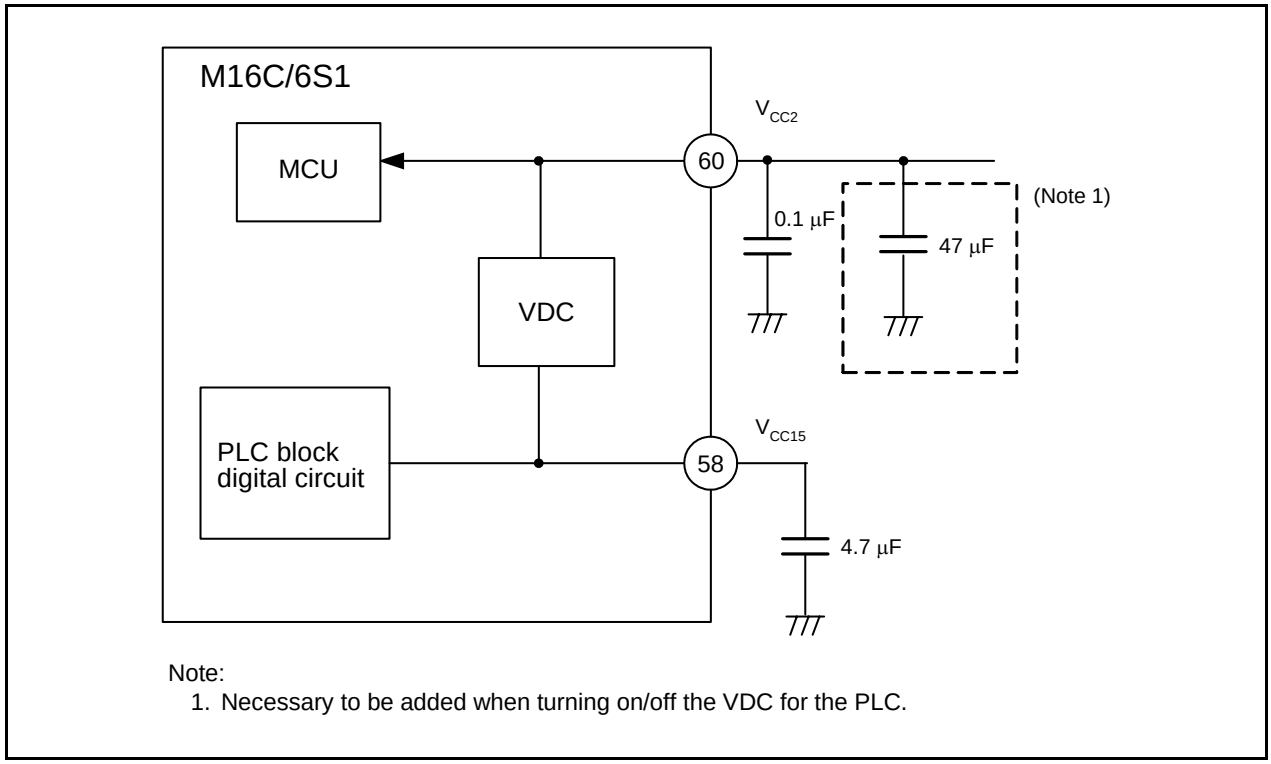


Figure 26.8 Recommended Input Filter

Note:
1. If a strong noise or signal enters from the power line, insert an appropriate protection circuit or attenuator so as not to exceed the acceptable input range for VINP/VINN.

26.4 Other Circuits

The VCC15 pin is a regulator (VDC) output pin for the digital circuit of the PLC block. Connect this pin to a bypass capacitance between the VSS. It cannot be used for the power supply for any other circuits. The power for this on-chip regulator is supplied from the VCC2.
When stopping the PLC block and also turning on/off the VDC, note that the VCC2 voltage may drop due to inrush current. A recommended peripheral circuit is shown in Figure 26.9. The PLC block is controlled by driver software provided from Renesas.



Note:
1. Necessary to be added when turning on/off the VDC for the PLC.

Figure 26.9 Recommended Peripheral Circuit for PLC Regulator

26.5 Notes on Mounting

- Since the M16C/6S1 Group has a built-in line driver circuit (power amplifier), solder the bare die pad tightly with the PCB. It is recommended that at least a four-layer PCB be used, for heat diffusion.
- Keep the enough line width of power supply for the line driver circuit on the PCB according to the specifications shown in the electric characteristics. If the line resistance is high, distortion may occur or sufficient amplitude may not be obtained.
- Be sure not to short the line driver output pins. If a short occurs between VOUTP and VOUTN or between these pins and the power supply or GND, the LSI may be damaged due to overcurrent. Be sure to connect all of the line drive output pins (32, 33, 34, 38, 39, and 40), the power supply pins (29, 30, 31, 41, 42, and 43), and the GND pins (35, 36, and 37).
- Use a transformer that has a 1:1 transfer ratio.
- Select a capacitance that has a small internal resistance (ESR) to be connected to the line driver output. If the ESR value becomes greater, the output amplitude from the transformer output becomes smaller.
- In the power supply (VCCA) for the analog circuit, separate the digital block and the power supply to reduce effects of noise from the digital circuit.
- Add a bypass capacitance of 0.1 μF near the LSI pins for each power supply. However, add a bypass capacitance of 4.7 μF or more for the line driver power supply pins (pin no: (29, 30, and 31) and (41, 42, and 43)).

27. Electrical Characteristics

27.1 Absolute Maximum Rating

Table 27.1 Absolute Maximum Ratings

Symbol	Parameter		Condition	Rated Value	Unit
V_{CC1}	Supply voltage		$V_{CC1} = AV_{CC}$	-0.3 to 4.6	V
V_{CC2}	Supply voltage		$V_{CC1} = AV_{CC}$	-0.3 to V_{CC1} + 0.1 (1)	V
V_{CCA}	Supply voltage			-0.3 to 4.6	V
AV_{CC}	Analog supply voltage		$V_{CC1} = AV_{CC}$	-0.3 to 4.6	V
V_{REF}	Analog reference voltage			-0.3 to V_{CC1} + 0.1 (1)	V
V_{CC15}	1.5 V power supply (output)			-0.3 to 2.1	V
V_I	Input voltage	$\overline{RESET}$, CNVSS, TMOD P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, XIN		-0.3 to V_{CC1} + 0.3 (1)	V
		P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, UROM_EN		-0.3 to V_{CC2} + 0.3 (1)	V
		P7_0, P7_1, P8_5		-0.3 to 4.6	V
		VINP, VINN, TESTP, TESTN, OC_EN		-0.3 to V_{CCA} + 0.3(1)	V
V_O	Output voltage	P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, XOUT		-0.3 to V_{CC1} + 0.3 (1)	V
		P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, TS		-0.3 to V_{CC2} + 0.3 (1)	V
		P7_0, P7_1, P8_5		-0.3 to 4.6	V
		VOUTP, VOUTN, EXTLDP, EXTLDN, VCM, VCMTX, VRT, VRB, TESTP, TESTN, OC		-0.3 to V_{CCA} + 0.3(1)	V
P_d	Power consumption		$-40°C \le T_{opr} \le 85°C$	1200	mW
T_{opr}	Operating temperature	When the MCU is operating	Program area	-20 to 85/-40 to 85	°C
		Flash program erase	Data area	-20 to 85/-40 to 85	
T_{stg}	Storage temperature			-65 to 150	°C

Notes:
1. Maximum value is 4.6 V.

27.2 Recommended Operating Conditions

Table 27.2 Recommended Operating Conditions (1/4)

V_{CC1} = V_{CC2} = 2.7 to 3.6 V at T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified.

Symbol	Parameter			Standard			Unit
				Min.	Typ.	Max.	
V_{CC1}	Supply voltage	PLC operation	V_{CC1} = V_{CC2} = V_{CCA}	3.0		3.6	V
		No PLC operation	V_{CC1} = V_{CC2} = V_{CCA}	2.7		3.6	V
V_{CC2}	Supply voltage		V_{CC1} = V_{CC2} = V_{CCA}		V_{CC1}		V
V_{CCA}	Supply voltage		V_{CC1} = V_{CC2} = V_{CCA}		V_{CC1}		V
AV_{CC}	Analog supply voltage				V_{CC1}		V
V_{SS}	Supply voltage				0		V
V_{SSA}	Supply voltage				0		V
AV_{SS}	Analog supply voltage				0		V
V_{IH}	High input voltage	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, UROM_EN		0.8 V_{CC2}		V_{CC2}	V
		P7_2 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, XIN, $\overline{\text{RESET}}$, CNVSS, TMOD		0.8 V_{CC1}		V_{CC1}	V
		P7_0, P7_1, P8_5		0.8 V_{CC1}		4.6	V
		OC_EN		0.8 V_{CCA}		V_{CCA}	V
V_{IL}	Low input voltage	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, UROM_EN		0		0.2 V_{CC2}	V
		P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, XIN, $\overline{\text{RESET}}$, CNVSS, TMOD		0		0.2 V_{CC1}	V
		OC_EN		0		0.2 V_{CCA}	V

Table 27.3 Recommended Operating Conditions (2/4)

V_{CC1} = V_{CC2} = 2.7 to 3.6 V at T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified.

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
$I_{OH(sum)}$	High peak sum output current	Sum of $I_{OH(peak)}$ at P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7			-40.0	mA
		Sum of $I_{OH(peak)}$ at P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7			-40.0	mA
$I_{OH(peak)}$	High peak output current	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7			-10.0	mA
$I_{OH(avg)}$	High average output current (1)	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7			-5.0	mA
$I_{OL(sum)}$	Low peak sum output current	Sum of $I_{OL(peak)}$ at P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7			80.0	mA
		Sum of $I_{OL(peak)}$ at P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7			80.0	mA
$I_{OL(peak)}$	Low peak output current	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7			10.0	mA
$I_{OL(avg)}$	Low average output current (1)	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7			5.0	mA

Note:
1. The average output current is the mean value within 100 ms.

Table 27.4 Recommended Operating Conditions (3/4)

V_{CC1} = V_{CC2} = 2.7 to 3.6 V at T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified.

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
$f_{(XIN)}$	Main clock oscillation frequency			15.36		MHz
$f_{(XCIN)}$	Sub clock oscillation frequency		30	32.768	35	kHz
$f_{(PLL)}$	PLL clock oscillation frequency	V_{CC} = 2.7 to 3.6 V	10		32	MHz
$f_{(BCLK)}$	CPU operation clock	V_{CC1} = V_{CC2} = 3.0 to 3.6 V	2		32 (1)	MHz
$f_{su(PLL)}$	PLL frequency synthesizer stabilization wait time	V_{CC1} = 3.0 V			3	ms

Note:
1. 24 MHz when V_{CC1} = V_{CC2} = 2.7 to 3.0 V.

Table 27.5 Recommended Operating Conditions (4/4)(1)

V_{CC1} = 2.7 to 3.6 V, V_{SS} = 0 V, and T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified (1).
The ripple voltage must not excess $V_{r(VCC1)}$ and/or $dV_{r(VCC1)}/dt$.

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
$V_{r(VCC1)}$	Allowable ripple voltage			0.3	Vp-p
$V_{r(VCCA)}$	VCCA allowable ripple voltage			0.025	Vp-p
$dV_{r(VCC1)}/dt$	Ripple voltage falling gradient			0.3	V/ms

Note:
1. The device is operationally guaranteed under these operating conditions.

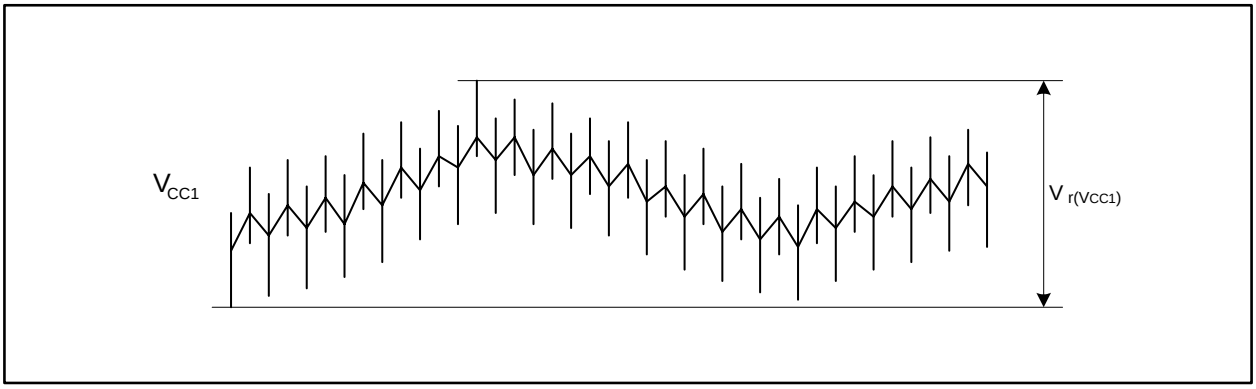


Figure 27.1 Ripple Waveform

27.3 A/D Conversion Characteristics

Table 27.6 A/D Conversion Characteristics (1/2) (1)

$AV_{CC} = V_{CC1} = V_{CC2} = 2.7$ to 3.6 V ≥ V_{REF}, $V_{SS} = AV_{SS} = 0$ V at T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified.

Symbol	Parameter		Measuring Condition	Standard			Unit
				Min.	Typ.	Max.	
—	Resolution		$AV_{CC} = V_{CC1} = V_{CC2} \geq V_{REF}$			10	Bits
I_{NL}	Integral non-linearity error	10bit	AN0 to AN7 input, AN0_0 to AN0_7 input, ANEX0, ANEX1 input (Note 2)			±3	LSB
—	Absolute accuracy	10bit	AN0 to AN7 input, AN0_0 to AN0_7 input, ANEX0, ANEX1 input (Note 2)			±3	LSB

Notes:
1. Use when $AV_{CC} = V_{CC1}$.
2. Flash memory rewrite disabled. Except for the analog input pin, set the pins to be measured as input ports and connect them to V_{SS}. See Figure 27.2 “A/D Accuracy Measure Circuit”.

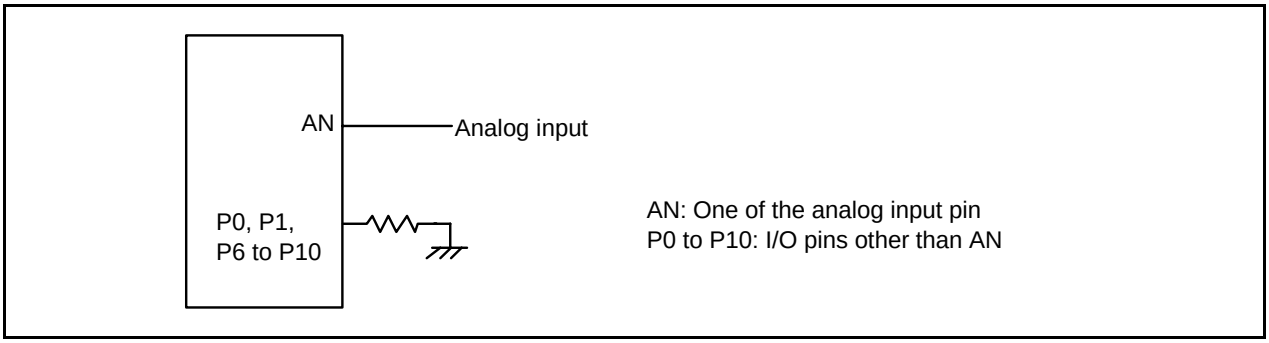


Figure 27.2 A/D Accuracy Measure Circuit

Table 27.7 A/D Conversion Characteristics (2/2) (1)

$AV_{CC} = V_{CC1} = V_{CC2} = 2.7$ to 3.6 V ≥ V_{REF}, $V_{SS} = AV_{SS} = 0$ V at T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified.

Symbol	Parameter	Measuring Condition	Standard			Unit
			Min.	Typ.	Max.	
ϕAD	A/D operating clock frequency	3.2 V ≤ V_{REF} ≤ AV_{CC} ≤ 3.6 V	2		15.36	MHz
		3.0 V ≤ V_{REF} ≤ AV_{CC} ≤ 3.6 V	2		11.52	MHz
		2.7 V ≤ V_{REF} ≤ AV_{CC} ≤ 3.6 V	2		5.72	MHz
—	Tolerance level impedance			3		kΩ
D_{NL}	Differential non-linearity error	(3)			±1	LSB
—	Offset error	(3)			±3	LSB
—	Gain error	(3)			±3	LSB
t_{CONV}	10-bit conversion time	V_{CC1} = 3.3 V, ϕAD = 15.36 MHz	2.8			µs
t_{SAMP}	Sampling time		0.98			µs
V_{REF}	Reference voltage		2.7		V_{CC1}	V
V_{IA}	Analog input voltage (2)		0		V_{REF}	V

Notes:
1. Use when $AV_{CC} = V_{CC1} = V_{CC2}$.
2. When analog input voltage is over reference voltage, the result of A/D conversion is 3FFh.
3. Flash memory rewrite disabled. Except for the analog input pin, set the pins to be measured as input ports and connect them to V_{SS}. See Figure 27.2 “A/D Accuracy Measure Circuit”.

27.4 Flash Memory Electrical Characteristics

Table 27.8 CPU Clock When Operating Flash Memory ($f_{(BCLK)}$)

V_{CC1} = 2.7 to 3.6 V, T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified.

Symbol	Parameter	Conditions	Standard			Unit
			Min.	Typ.	Max.	
—	CPU rewrite mode				10 (1)	MHz
$f_{(SLOW\_R)}$	Slow read mode				5 (3)	MHz
—	Low current consumption read mode			fC(32.768)	35	kHz
—	Data flash read	3.0 V < V_{CC1} < 3.6 V			20 (2)	MHz
		2.7 V ≤ V_{CC1} ≤ 3.0 V			16 (2)	MHz

Notes:
1. Set the PM17 bit in the PM1 register to 1 (one wait).
2. When the frequency is over this value, set the FMR17 bit in the FMR1 register to 0 (one wait) or the PM17 bit in the PM1 register to 1 (one wait)
3. Set the PM17 bit in the PM1 register to 1 (one wait). When using 125 kHz on-chip oscillator clock or sub clock as the CPU clock source, a wait is not necessary.

Table 27.9 Flash Memory (Program ROM 1, 2) Electrical Characteristics

V_{CC1} = 2.7 to 3.6 V at T_{opr} = 0 to 60°C (option: -40°C to 85°C), unless otherwise specified.

Symbol	Parameter	Conditions	Standard			Unit
			Min.	Typ.	Max.	
—	Program and erase cycles (1), (3), (4)	V_{CC1} = 3.3 V, T_{opr} = 25°C	1,000 (2)			times
—	Two words program time	V_{CC1} = 3.3 V, T_{opr} = 25°C		150	4000	µs
—	Lock bit program time	V_{CC1} = 3.3 V, T_{opr} = 25°C		70	3000	µs
—	Block erase time	V_{CC1} = 3.3 V, T_{opr} = 25°C		0.2	3.0	s
$t_{d(SR-SUS)}$	Time delay from suspend request until suspend				5 + CPU clock × 3 cycles	ms
—	Interval from erase start/restart until following suspend request		0			µs
—	Suspend interval necessary for auto-erasure to complete (7)		20			ms
—	Time from suspend until erase restart				30 + CPU clock × 1 cycle	µs
—	Program, erase voltage		2.7		3.6	V
—	Read voltage	Topr= -20 to 85°C/-40 to 85°C	2.7		3.6	V
—	Program, erase temperature		0		60	°C
t_{PS}	Flash memory circuit stabilization wait time				50	µs
—	Data hold time (6)	Ambient temperature = 55°C	20			year

Notes:
1. Definition of program and erase cycles:
   The program and erase cycles refer to the number of per-block erasures. If the program and erase cycles are n (n = 1,000), each block can be erased n times. For example, if a 64 Kbyte block is erased after writing two word data 16,384 times, each to a different address, this counts as one program and erase cycles. Data cannot be written to the same address more than once without erasing the block (rewrite prohibited).
2. Cycles to guarantee all electrical characteristics after program and erase. (1 to Min. value can be guaranteed).
3. In a system that executes multiple programming operations, the actual erasure count can be reduced by writing to sequential addresses in turn so that as much of the block as possible is used up before performing an erase operation. It is advisable to retain data on the erasure cycles of each block and limit the number of erase operations to a certain number.
4. If an error occurs during block erase, attempt to execute the clear status register command, then execute the block erase command at least three times until the erase error does not occur.
5. Customers desiring program/erase failure rate information should contact their Renesas Electronics support representative.
6. The data hold time includes time that the power supply is off or the clock is not supplied.
7. After an erase start or erase restart, if an interval of at least 20 ms is not set before the next suspend request, the erase sequence cannot be completed.

Table 27.10 Flash Memory (Data Flash) Electrical Characteristics

V_{CC1} = 2.7 to 3.6 V at T_{opr} = -20 to 85°C/-40 to 85°C, unless otherwise specified.

Symbol	Parameter	Conditions	Standard			Unit
			Min.	Typ.	Max.	
—	Program and erase cycles (1, 3, 4)	V_{CC1} = 3.3 V, T_{opr} = 25°C	10,000 (2)			times
—	Two words program time	V_{CC1} = 3.3 V, T_{opr} = 25°C		300	4000	µs
—	Lock bit program time	V_{CC1} = 3.3 V, T_{opr} = 25°C		140	3000	µs
—	Block erase time	V_{CC1} = 3.3 V, T_{opr} = 25°C		0.2	3.0	s
$t_{d(SR-SUS)}$	Time delay from suspend request until suspend				5 + CPU clock × 3 cycles	ms
—	Interval from erase start/restart until following suspend request		0			µs
—	Suspend interval necessary for auto-erasure to complete (7)		20			ms
—	Time from suspend until erase restart				30 + CPU clock × 1 cycle	µs
—	Program, erase voltage		2.7		3.6	V
—	Read voltage		2.7		3.6	V
—	Program, erase temperature		-20/-40		85	°C
t_{PS}	Flash Memory Circuit Stabilization Wait Time				50	µs
—	Data hold time (6)	Ambient temperature = 55 °C	20			year

Notes:

1. Definition of program and erase cycles
   The program and erase cycles refer to the number of per-block erasures.
   If the program and erase cycles are n (n = 10,000), each block can be erased n times.
   For example, if a 4 Kbyte block is erased after writing two word data 1,024 times, each to a different address, this counts as one program and erase cycles. Data cannot be written to the same address more than once without erasing the block (rewrite prohibited).
2. Cycles to guarantee all electrical characteristics after program and erase. (1 to Min. value can be guaranteed).
3. In a system that executes multiple programming operations, the actual erasure count can be reduced by writing to sequential addresses in turn so that as much of the block as possible is used up before performing an erase operation. For example, when programming groups of 16 bytes, the effective number of rewrites can be minimized by programming up to 256 groups before erasing them all in one operation. In addition, averaging the erasure cycles between blocks A and B can further reduce the actual erasure cycles. It is also advisable to retain data on the erasure cycles of each block and limit the number of erase operations to a certain number.
4. If an error occurs during block erase, attempt to execute the clear status register command, then execute the block erase command at least three times until the erase error does not occur.
5. Customers desiring program/erase failure rate information should contact their Renesas technical support representative.
6. The data hold time includes time that the power supply is off or the clock is not supplied.
7. After an erase start or erase restart, if an interval of at least 20 ms is not set before the next suspend request, the erase sequence cannot be completed.

27.5 Voltage Detector and Power Supply Circuit Electrical Characteristics

Table 27.11 Voltage Detector 0 Electrical Characteristics

The measurement condition is V_{CC1} = 2.7 to 3.6 V, T_{opr} = -20 to 85°C/-40 to 85°C, unless otherwise specified.

Symbol	Parameter	Condition	Standard			Unit
			Min.	Typ.	Max.	
V_{det0}	Voltage detection level Vdet0_0 (1)	When V_{CC1} is falling.	1.80	1.90	2.10	V
	Voltage detection level Vdet0_2 (1)	When V_{CC1} is falling.	2.70	2.85	3.00	V
—	Voltage detector 0 response time (3)	When V_{CC1} falls from 3.6 V to (Vdet0_0 - 0.1) V			200	µs
—	Voltage detector self power consumption	VC25 = 1, V_{CC1} = 3.3 V		1.5		µA
$t_{d(E-A)}$	Waiting time until voltage detector operation starts (2)				100	µs

Notes:
1. Select the voltage detection level with the VDSEL1 bit in the OFS1 address.
2. Necessary time until the voltage detector operates when setting to 1 again after setting the VC25 bit in the VCR2 register to 0.
3. Time from when passing the V_{det0} until when a voltage monitor 0 reset is generated.

Table 27.12 Power Supply Circuit Timing Characteristics

The measurement condition is V_{CC1} = 2.7 to 3.6 V and T_{opr} = 25°C, unless otherwise specified.

Symbol	Parameter	Condition	Standard			Unit
			Min.	Typ.	Max.	
$t_{d(P-R)}$	Internal power supply stability time when power is on (1)				5	ms
$t_{d(R-S)}$	STOP release time				150	µs
$t_{d(W-S)}$	Low power mode wait mode release time				150	µs

Note:
1. Waiting time until the internal power supply generator stabilizes when power is on.

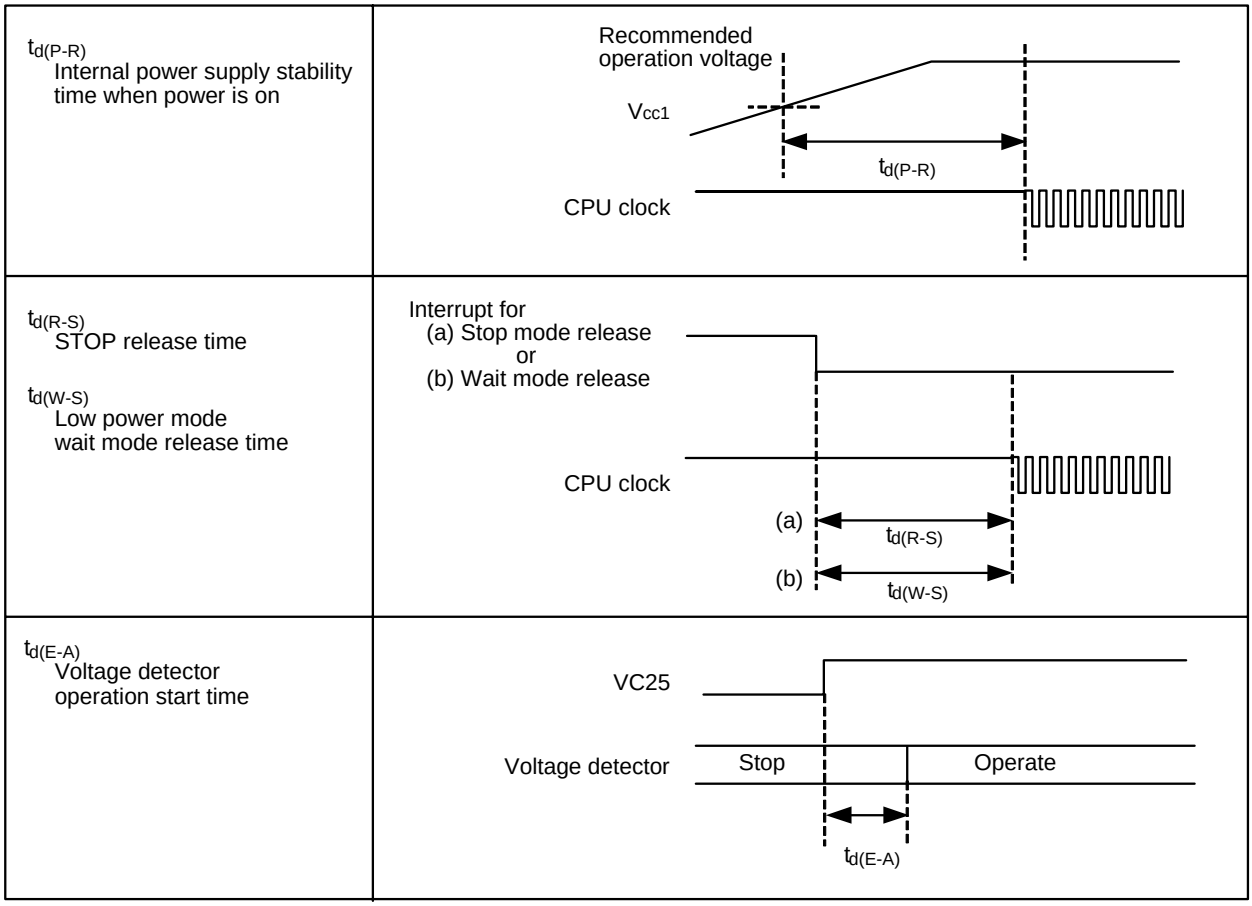


Figure 27.3 Power Supply Circuit Timing Diagram

27.6 Oscillation Circuit Electrical Characteristics

Table 27.13 125 kHz On-Chip Oscillator Circuit Electrical Characteristics

V_{CC1} = 2.7 to 3.6 V, T_{opr} = -20 to 85°C/-40 to 85°C, unless otherwise specified.

Symbol	Parameter	Condition	Standard			Unit
			Min.	Typ.	Max.	
f_{OCO-S}	125 kHz on-chip oscillator frequency	Average frequency in a 10 ms period	100	125	150	kHz
$t_{su(fOCO-S)}$	Wait time until 125 kHz on-chip oscillator stabilizes				20	µs

27.7 Analog Front End Electrical Characteristics

Table 27.14 AFE Electrical Characteristics

$V_{CC1} = A_{VCC} = V_{CCA} = 3.3$ V, $V_{SSA} = 0$ V, $T_{opr} = 25°C$ unless otherwise specified.

Parameter	Measuring Condition	Pin Name	Standard			Unit
			Min.	Typ.	Max.	
Analog supply voltage	DC	VCCA	3.0	3.3	3.6	V
PLC digital block supply voltage (internal supply)	DC	VCC15		1.5		V
DC reference voltage	DC	VCMTX		VCCA/2		V
	DC	VCM		1.5		V
	DC	VRT		2.0		V
	DC	VRB		1.0		V
Analog power supply current	Recommended peripheral circuit (Figure 26.3) DACVREF = 11 Load resistance = 1 Ω	VCCA (pin no,: 29, 30, 31, 41, 42, 43)			350	mA_{RMS}
Maximum output amplitude	VOUTP/VOUTN load resistance = 50 Ω DACVREF = 11	VOUTP VOUTN		5.0		Vppd
Output pin DC potential	During transmission, While transmission is stopped	VOUTP VOUTN		VCCA/2		
SFDR (Spurious Free Dynamic Range)	Load resistance = 50 Ω f = 100 KHz RBW = 9 kHz Recommended peripheral circuit (Figure 26.3)	VOUTP VOUTN		-60 (1)		dB_C
Output impedance	While transmission is stopped (DC)	VOUTP VOUTN		16		kΩ
Maximum output amplitude (with external line driver connected)	Load resistance DACVREF = 11	EXTLDP EXTLDN		4.2		Vppd
Input impedance	DC	VINP, VINN		1		Kohms
Input amplitude	VGA = -6 dB	VINP, VINN			3.0	Vpp
Input pin DC potential	DC	VINP, VINN		1.5		V
AGC gain step	VGA gain settings	—		6		dB

Note:

1. These values are defined at the LSI output pin. Output amplitude and SFDR vary depending on the following conditions:
   - Set value of DACVREF
   - Parameters of the peripheral circuit and the coupling circuit, and the conditions of the transmission channel

27.8 Electrical Characteristics

Table 27.15 Electrical Characteristics (1)

V_{CC1} = V_{CC2} = 2.7 to 3.6 V, V_{SS} = 0 V at T_{opr} = -20 to 85°C/-40 to 85°C, $f_{(BCLK)}$ = 30.72 MHz unless otherwise specified.

Symbol	Parameter			Measuring Condition	Standard			Unit
					Min.	Typ.	Max.	
V_{OH}	High output voltage	P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7		I_{OH} = –1 mA	V_{CC1} – 0.5		V_{CC1}	V
		P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7		I_{OH} = –1 mA	V_{CC2} – 0.5		V_{CC2}	
V_{OH}	High output voltage XOUT		High drive	I_{OH} = –0.1 mA	V_{CC1} – 0.5		V_{CC1}	V
			Low drive	I_{OH} = –50 µA	V_{CC1} – 0.5		V_{CC1}	
	High output voltage XCOUT			With no load applied		1.5		V
V_{OL}	Low output voltage	P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7		I_{OL} = 1 mA			0.5	V
		P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7		I_{OL} = 1 mA			0.5	
V_{OL}	Low output voltage XOUT		High drive	I_{OL} = 0.1 mA			0.5	V
			Low drive	I_{OL} = 50 µA			0.5	
	Low output voltage XCOUT			With no load applied		0		V
V_{T+}-V_{T-}	Hysteresis	TA1IN to TA4IN, TB0IN to TB5IN, INT1 to INT5, NMI, ADTRG, CTS0 to CTS2, CTS5 to CTS6, SCL0 to SCL2, SCL5 to SCL6, SDA0 to SDA2, SDA5 to SDA6, CLK0 to CLK6, TA1OUT to TA4OUT, KI0 to KI7, RXD0 to RXD2, RXD5 to RXD6, SIN3, SIN4, SCLMM, SDAMM, OC_EN			0.2		1.0	V
V_{T+}-V_{T-}	Hysteresis	RESET			0.2		1.8	V
I_{IH}	High input current	P0_0 to P0_7, P1_0 to P1_7,		V_I = V_{CC2}			4.0	µA
		P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7 XIN, RESET, CNVSS, UROM_EN, TMOD, OC_EN		V_I = V_{CC1}			4.0	µA
I_{IL}	Low input current	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7 XIN, RESET, CNVSS, UROM_EN, TMOD, OC_EN		V_I = 0 V			-4.0	µA
R_{PULLUP}	Pull-up resistance	P0_0 to P0_7, P1_0 to P1_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7		V_I = 0 V	30	100	500	kΩ
R_{fXIN}	Feedback resistance XIN					1		MΩ
R_{fXCIN}	Feedback resistance XCIN					8		MΩ
V_{RAM}	RAM retention voltage			In stop mode	1.8			V

Table 27.16 Electrical Characteristics (2)

$V_{CC1} = V_{CC2} = 2.7$ to 3.6 V, $V_{SS} = 0$ V at $T_{opr} = -20$ to 85°C/−40 to 85°C, $f_{(BCLK)} = 30.72$ MHz unless otherwise specified.

Symbol	Parameter	Measuring Condition				Standard			Unit
		MCU		PLC		Min.	Typ.	Max.	
I_{CC}	Power supply current The MCU output pin are open and the other MCU pins are V_{SS} PLC no operation Japanese standards (ARIB) mode FCC setting, DACVREF = 3 (2)	High-speed mode	$f_{(BCLK)}$ = 30.72 MHz XIN = 15.36 MHz, 125 kHz on-chip oscillator off, CM15 = 1 (Driving ability High) PLL operates	During reception standby	DUAL mode		75		mA
				During reception	Load resistance 10 Ω, VCCA = 3.3 V		92		mA
				During transmission			202		mA
				PLC = OFF (1)			20		mA
			$f_{(BCLK)}$ = 15.36 MHz XIN = 15.36 MHz 125 kHz on-chip oscillator off CM15 = 1 (Driving ability High) PLL stops	During reception standby	DUAL mode		66		mA
				During reception	Load resistance 10 Ω, VCCA = 3.3 V		83		mA
				During transmission			193		mA
				PLC = OFF (1)			11		mA
		Wait mode	$f_{(BCLK)}$ = 15.36 MHz XIN = 15.36 MHz 125 kHz on-chip oscillator off CM15 = 1 (Driving ability High) PLL stops	During reception standby	DUAL mode		59		mA
				PLC = OFF (1)			4		mA
		125 kHz on-chip oscillator mode	Main clock stop 125 kHz on-chip oscillator on, no division FMR22 = 1 (slow read mode)	PLC = OFF (1)			2.2		mA
		Low-power mode	$f_{(BCLK)}$ = 32 kHz Main clock stop 125 kHz on-chip oscillator off FMR 22 = FMR23 = 1 (in low-current consumption read mode) On flash memory (3)				2.1		mA
		Wait mode	$f_{(BCLK)}$ = 32 kHz Main clock stop 125 kHz on-chip oscillator off				2		mA
		Stop mode	Main clock stop Sub clock stop 125 kHz on-chip oscillator off				2		mA
		During flash memory program	$f_{(BCLK)}$ = 10 MHz, PM17 = 1 (one wait) V_{CC1} = 3.0 V				22		mA
		During flash memory erase	$f_{(BCLK)}$ = 10 MHz, PM17 = 1 (one wait) V_{CC1} = 3.0 V				32		mA

Notes:
1. The PLC setting applies in the state where the clock in the digital block is stopped, each EN = L in the AFE block, and the VDC and bias in the AFE block are operating.
2. Set by DLL software provided from Renesas.
3. This indicates the memory in which the program to be executed exists.

27.9 Timing Requirements (Peripheral Functions and Others)

(V_{CC1} = V_{CC2} = 2.7~3.6 V, V_{SS} = 0 V, at T_{opr} = -20 to 85°C/-40 to 85°C unless otherwise specified)

27.9.1 Reset Input ($\overline{\text{RESET}}$ Input)

Table 27.17 Reset Input ($\overline{\text{RESET}}$ Input)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(RSTL)}$	$\overline{\text{RESET}}$ input low pulse width	10		µs

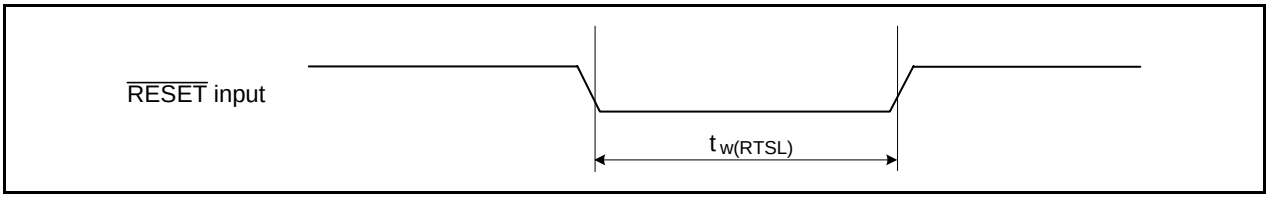


Figure 27.4 Reset Input ($\overline{\text{RESET}}$ Input)

27.9.2 Timer A Input

Table 27.18 Timer A Input (Counter Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	150		ns
$t_{w(TAH)}$	TAiIN input high pulse width	60		ns
$t_{w(TAL)}$	TAiIN input low pulse width	60		ns

Table 27.19 Timer A Input (Gating Input in Timer Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	600		ns
$t_{w(TAH)}$	TAiIN input high pulse width	300		ns
$t_{w(TAL)}$	TAiIN input low pulse width	300		ns

Table 27.20 Timer A Input (External Trigger Input in One-Shot Timer Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	300		ns
$t_{w(TAH)}$	TAiIN input high pulse width	150		ns
$t_{w(TAL)}$	TAiIN input low pulse width	150		ns

Table 27.21 Timer A Input (External Trigger Input in Pulse Width Modulation Mode and Programmable Output Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN input high pulse width	150		ns
$t_{w(TAL)}$	TAiIN input low pulse width	150		ns

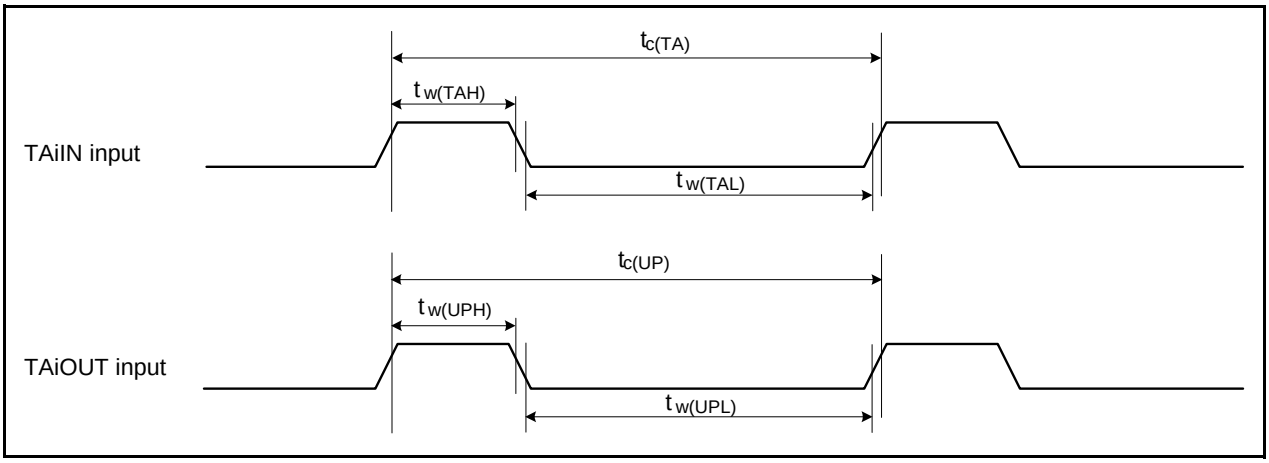


Figure 27.5 Timer A Input

Table 27.22 Timer A Input (Two-Phase Pulse Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	2		µs
$t_{su(TAIN-TAOUT)}$	TAiOUT input setup time	500		ns
$t_{su(TAOUT-TAIN)}$	TAiIN input setup time	500		ns

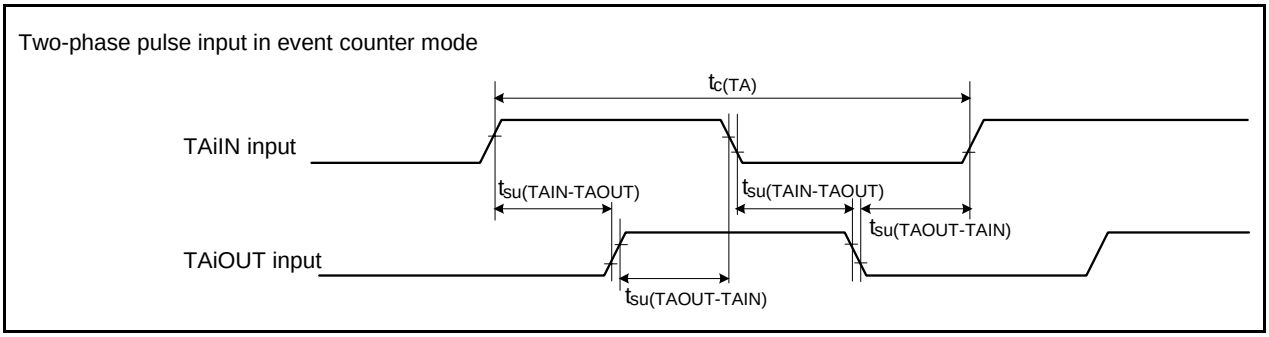


Figure 27.6 Timer A Input (Two-Phase Pulse Input in Event Counter Mode)

27.9.3 Timer B Input

Table 27.23 Timer B Input (Counter Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time (counted on one edge)	150		ns
$t_{w(TBH)}$	TBiIN input high pulse width (counted on one edge)	60		ns
$t_{w(TBL)}$	TBiIN input low pulse width (counted on one edge)	60		ns
$t_{c(TB)}$	TBiIN input cycle time (counted on both edges)	300		ns
$t_{w(TBH)}$	TBiIN input high pulse width (counted on both edges)	120		ns
$t_{w(TBL)}$	TBiIN input low pulse width (counted on both edges)	120		ns

Table 27.24 Timer B Input (Pulse Period Measurement Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input high pulse width	300		ns
$t_{w(TBL)}$	TBiIN input low pulse width	300		ns

Table 27.25 Timer B Input (Pulse Width Measurement Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input high pulse width	300		ns
$t^{w(TBL)}$	TBiIN input low pulse width	300		ns

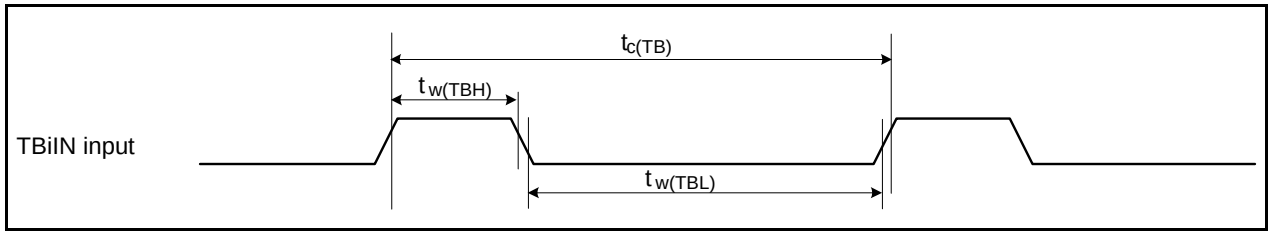


Figure 27.7 Timer B Input

27.9.4 Serial Interface

Table 27.26 Serial Interface

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi input cycle time	300		ns
$t_{w(CKH)}$	CLKi input high pulse width	150		ns
$t_{w(CKL)}$	CLKi input low pulse width	150		ns
$t_{d(C-Q)}$	TXDi output delay time		160	ns
$t_{h(C-Q)}$	TXDi hold time	0		ns
$t_{su(D-C)}$	RXDi input setup time	100		ns
$t_{h(C-D)}$	RXDi input hold time	90		ns

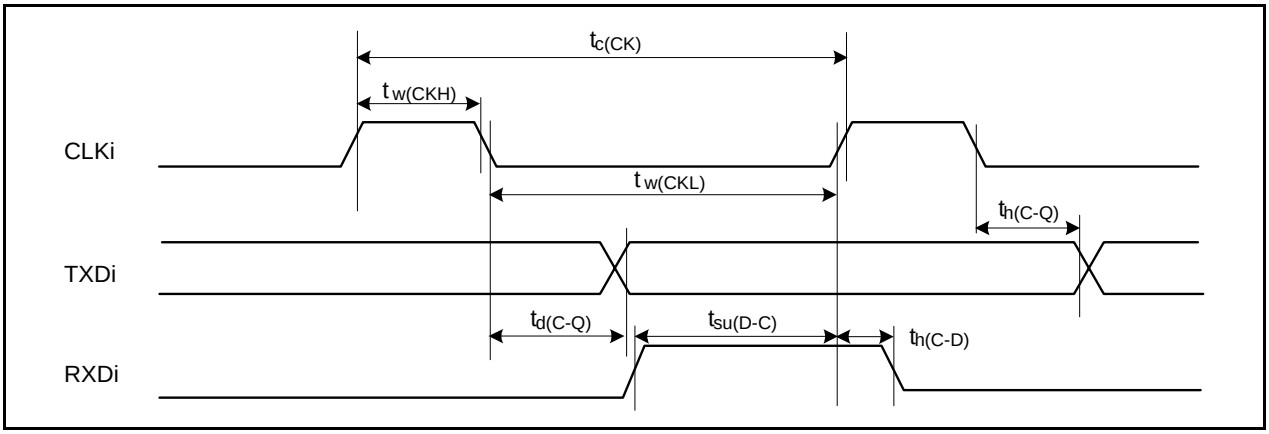


Figure 27.8 Serial Interface

27.9.5 External Interrupt $\overline{\text{INTi}}$ Input

Table 27.27 External Interrupt $\overline{\text{INTi}}$ Input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{\text{INTi}}$ input high pulse width	380		ns
$t_{w(INL)}$	$\overline{\text{INTi}}$ input low pulse width	380		ns

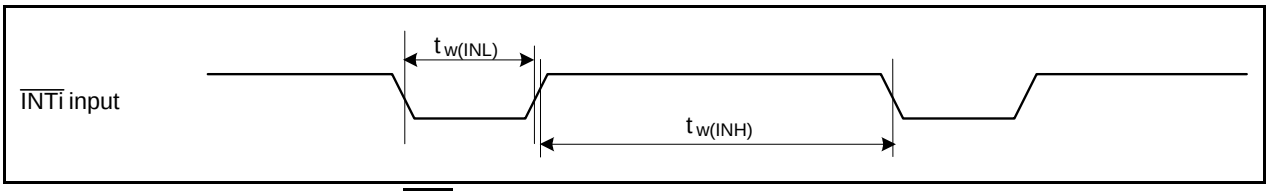


Figure 27.9 External Interrupt $\overline{\text{INTi}}$ Input

27.9.6 Multi-master I²C-bus

Table 27.28 Multi-master I²C-bus

Symbol	Parameter	Standard Clock Mode		Fast-mode Unit		Unit
		Min.	Max.	Min.	Max.	
t_{BUF}	Bus free time	4.7		1.3		µs
$t_{HD;STA}$	Hold time in start condition	4.0		0.6		µs
t_{LOW}	Hold time in SCL clock 0 status	4.7		1.3		µs
t_R	SCL, SDA signals' rising time		1000	20 + 0.1Cb	300	ns
$t_{HD;DAT}$	Data hold time	0		0	0.9	µs
t_{HIGH}	Hold time in SCL clock 1 status	4.0		0.6		µs
t_F	SCL, SDA signals' falling time		300	20 + 0.1 Cb	300	ns
$t_{su;DAT}$	Data setup time	250		100		ns
$t_{su;STA}$	Setup time in restart condition	4.7		0.6		µs
$t_{su;STO}$	Stop condition setup time	4.0		0.6		µs

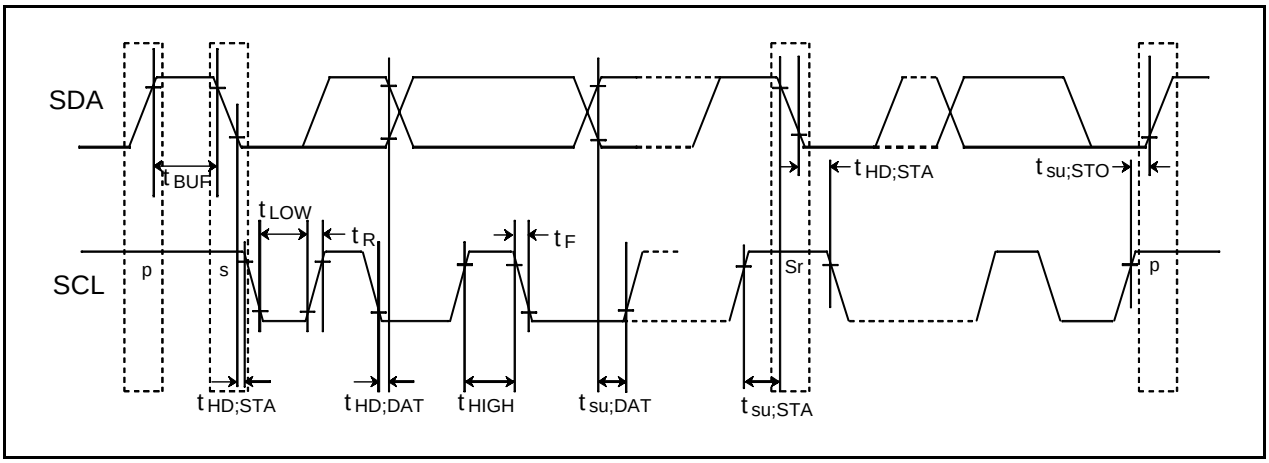


Figure 27.10 Multi-master I²C-bus

28. Usage Notes

28.1 OFS1 Address and ID Code Storage Address

The OFS1 address and ID code storage address are part of flash memory. When writing a program to flash memory, write an appropriate value to those addresses simultaneously.

In the OFS1 address, the MCU state after reset and the function to prevent rewrite in parallel I/O mode are selected. The OFS1 address is 0FFFFFh. This is the most significant address of block 0 in program ROM 1 and upper address of reset vector. Also, the ID code storage address is in block 0 and upper address of the interrupt vector.

The ID code check function cannot be disabled. Even if the protect using the ID code check function is unnecessary, input the appropriate ID code when using a serial programmer or debugger. Without the appropriate ID code, the serial programmer or debugger cannot be used.

ex) Set FEh to the OFS1 addres

When using an address control instruction and logical addition:

```
.org 0FFFFCh
 RESET:
.lword start | 0FE000000h
```

When using an address control instruction:

```
.org 0FFFFCh
 RESET:
.addr start
.byte 0FEh
```

(Program format varies depending on the compiler. Refer to the compiler manual.)

28.2 Notes on Noise

Connect a bypass capacitor (approximately 0.1 μF) across pins VCC1 and VSS, and pins VCC2 and VSS using the shortest and thickest possible wiring. Figure 28.1 shows the Bypass Capacitor Connection.

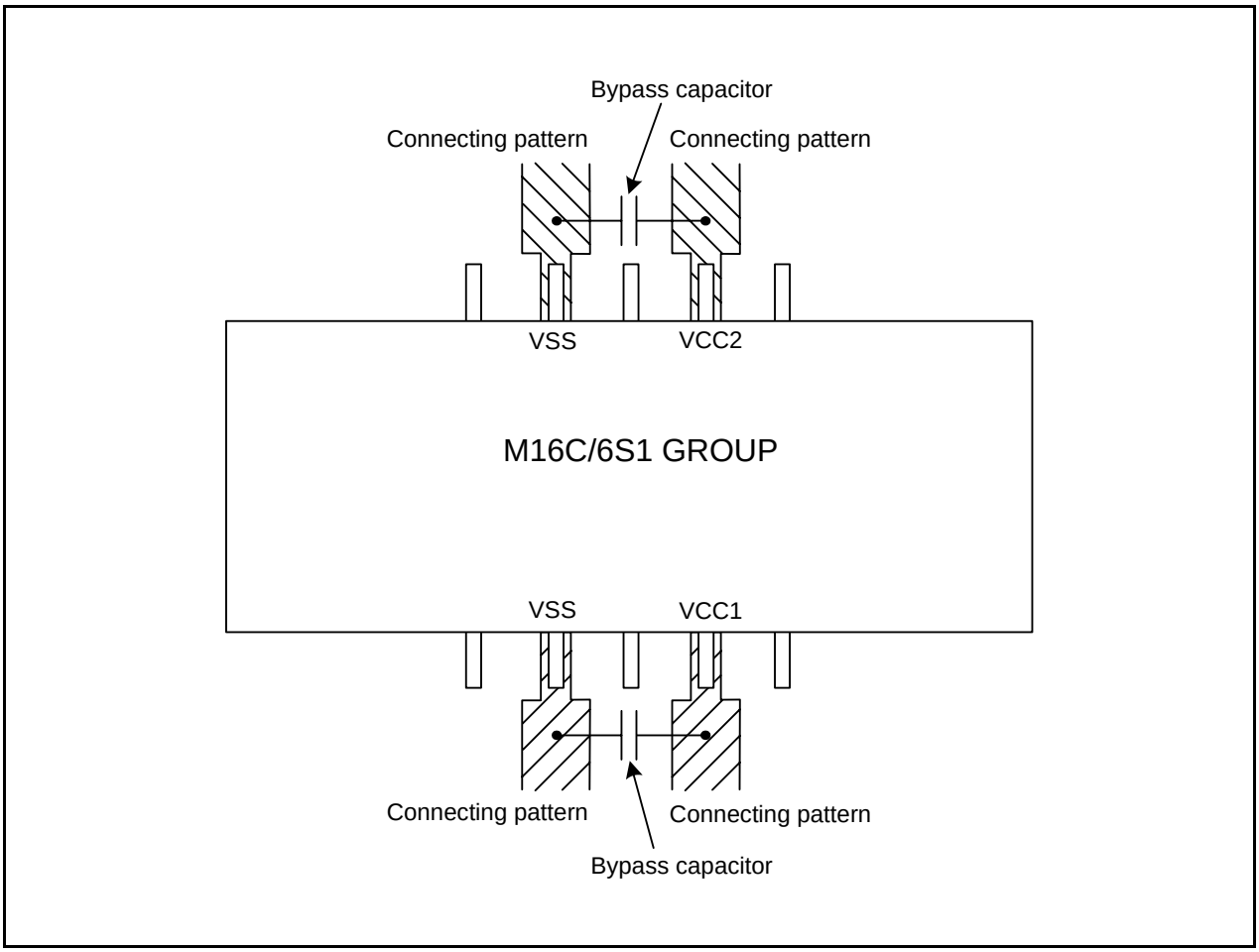


Figure 28.1 Bypass Capacitor Connection

28.3 Notes on SFRs

28.3.1 Register Settings

Table 28.1 lists Registers with Write-Only Bits and registers whose function differs between reading and writing. Set these registers with immediate values. Do not use read-modify-write instructions. When establishing the next value by altering the existing value, write the existing value to the RAM as well as to the register. Transfer the next value to the register after making changes in the RAM.

Table 28.1 Registers with Write-Only Bits

Register	Symbol	Address
Watchdog Timer Refresh Register	WDTR	037Dh
Watchdog Timer Start Register	WDTS	037Eh
Timer A0 Register	TA0	0327h to 0326h
Timer A1 Register	TA1	0329h to 0328h
Timer A2 Register	TA2	032Bh to 032Ah
Timer A3 Register	TA3	032Dh to 032Ch
Timer A4 Register	TA4	032Fh to 032Eh
Timer A1-1 Register	TA11	0303h to 0302h
Timer A2-1 Register	TA21	0305h to 0304h
Timer A4-1 Register	TA41	0307h to 0306h
UART0 Bit Rate Register	U0BRG	0249h
UART1 Bit Rate Register	U1BRG	0259h
UART2 Bit Rate Register	U2BRG	0269h
UART5 Bit Rate Register	U5BRG	0289h
UART6 Bit Rate Register	U6BRG	0299h
UART7 Bit Rate Register	U7BRG	02A9h
UART0 Transmit Buffer Register	U0TB	024Bh to 024Ah
UART1 Transmit Buffer Register	U1TB	025Bh to 025Ah
UART2 Transmit Buffer Register	U2TB	026Bh to 026Ah
UART5 Transmit Buffer Register	U5TB	028Bh to 028Ah
UART6 Transmit Buffer Register	U6TB	029Bh to 029Ah
UART7 Transmit Buffer Register	U7TB	02ABh to 02AAh
SI/O3 Bit Rate Register	S3BRG	0273h
SI/O4 Bit Rate Register	S4BRG	0277h
I2C0 Control Register 1	S3D0	02B6h
I2C0 Status Register 0	S10	02B8h

Table 28.2 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, DEC, DIV, DIVU, DIVX, EXTS, INC, MUL, MULU, NEG, SBB, and SUB
Decimal operation	DADC, DADD, DSBB, and DSUB
Logical operation	AND, NOT, OR, and XOR
Jump	ADJNZ, SBJNZ

28.4 Notes on Protection

After setting the PRC2 bit to 1 (write enabled), by writing to a given SFR, the PRC2 bit becomes 0 (write disabled). Change the registers protected by the PRC2 bit in the next instruction after setting the PRC2 bit to 1. Make sure there are no interrupts or DMA transfers between the instruction that sets the PRC2 bit to 1 and the next instruction.

28.5 Notes on Resets

28.5.1 Power Supply Rising Gradient

When supplying power to the MCU, make sure that the power supply voltage applied to the VCC1 pin meets the SVCC conditions.

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
SVCC	Power supply VCC1 rising gradient (Voltage range: 0 V to 2.0 V)	0.05			V/ms

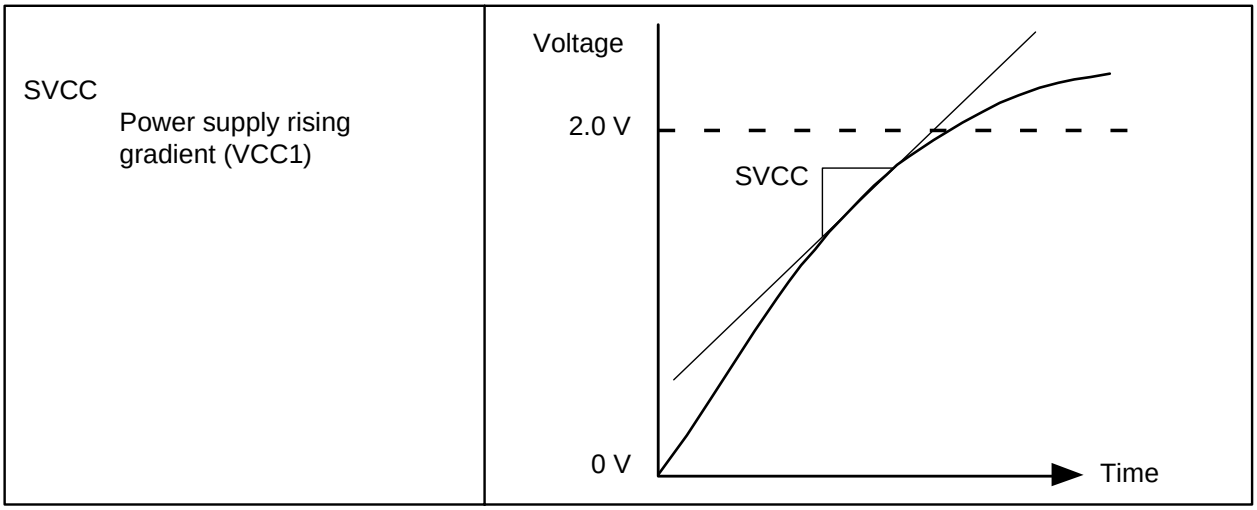


Figure 28.2 SVCC Timing

28.5.2 OSDR Bit (Oscillation Stop Detect Reset Detection Flag)

When an oscillation stop detect reset is generated, the MCU is reset and then stopped. This state is canceled by hardware reset or voltage monitor 0 reset.

Note that the OSDR bit in the RSTFR register value is not affected by a hardware reset, but becomes 0 (not detected) from a voltage monitor 0 reset.

28.5.3 Hardware Reset When VCC1 < Vdet0

If a hardware reset is executed when the LVDAS bit in the OFS1 address is 0 (voltage monitor 0 reset enabled after hardware reset) and VCC1 < Vdet0, the MCU executes the program at the address indicated by the reset vector when changing the signal applied to the RESET pin from low to high. A voltage monitor 0 reset is not generated.

28.5.4 Starting PLL Clock Oscillation

28.5.4.1 When Using Voltage Detector 0

Do not change the PLC07 bit in the PLC0 register from 0 to 1 when the VC25 bit in the VCR2 register is 1.

To change the PLC07 bit from 0 to 1 while using a voltage detector, use the following procedure:

(1) Set the VC25 bit to 0 (voltage detector off).
(2) Change the PLC07 bit from 0 to 1.
(3) Wait for 1 ms.
(4) Set the VC25 bit to 1 (voltage detector on).

28.5.4.2 When Using 125 kHz On-chip Oscillator Mode or 125 kHz On-chip Oscillator Low Power Mode

Change the PLC07 bit in the PLC0 register from 0 to 1 while dividing the clock by 8 or 16 (selectable by setting the CM06 bit in the CM0 register and bits CM17 to CM16 in the CM1 register).

28.5.4.3 Count Source for Timer A and Timer B

When using the PLL clock, do not use fOCO-S as the count source for timer A and timer B.

28.5.4.4 When Using fOCO-S as the Count Source for the Watchdog Timer

Change the PLC07 bit in the PLC0 register from 0 to 1 using the following procedure:

(1) Write 00h to the WDTR register, then write FFh (watchdog timer refresh).
(2) Change the PLC07 bit from 0 to 1.
(3) Wait for 1 ms.
(4) Write 00h to the WDTR register, then write FFh (watchdog timer refresh).

28.6 Notes on Clock Generator

28.6.1 Oscillation Circuit Using an Oscillator

The following items should be observed when connecting an oscillator:

- The oscillation characteristics are tied closely to the user's board design. Perform a careful evaluation of the board before connecting an oscillator.
- Oscillation circuit structure depends on the oscillator. The M16C/6S1 Group MCU contains a feedback resistor, but an additional external feedback resistor may be required. Contact the oscillator manufacturer regarding circuit constants, as they are dependent on the oscillator or stray capacitance of the mounted circuit.
- Check output from the CLKOUT pin to confirm that the clock generated by the oscillation circuit is properly transmitted to the MCU.

The procedure for outputting a clock from the CLKOUT pin is listed below. Set the clock output from the CLKOUT pin to 25 MHz or lower.

Outputting the main clock

(1) Set the PRC0 bit in the PRCR register to 1 (write enabled).
(2) Set the CM11 bit in the CM1 register, the CM07 bit in the CM0 register, and the CM21 bit in the CM2 register all to 0 (main clock selected).
(3) Select the clock output from the CLKOUT pin (see the table below).
(4) Set the PRC0 bit in the PRCR register to 0 (write disabled).

Table 28.3 Output from CLKOUT Pin When Selecting Main Clock

Bit Setting		Output from the CLKOUT Pin
PCLKR register	CM0 register	
PCLK5 bit	Bits CM01 to CM00	
1	00b	Clock with the same frequency as the main clock
0	10b	Main clock divided by 8
0	11b	Main clock divided by 32

Outputting the sub clock

(1) Set the PRC0 bit in the PRCR register to 1 (write enabled).
(2) Set the CM07 bit in the CM0 register to 1 (sub clock selected).
(3) Set the PCLK5 bit in the PCLKR register to 0, and bits CM01 to CM00 in the CM0 register to 01b (fC output from CLKOUT pin).
(4) Set the PRC0 bit in the PRCR register to 0 (write disabled).

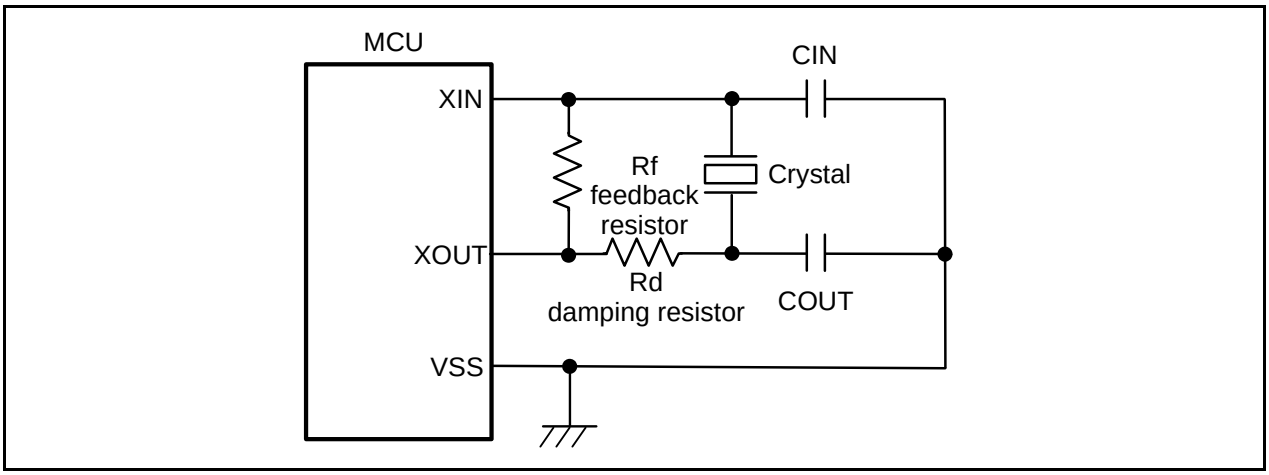


Figure 28.3 Oscillation Circuit Example

28.6.2 Noise Countermeasure

28.6.2.1 Clock I/O Pin Wiring

- Connect the shortest possible wiring to the clock I/O pin.
- Connect (a) the capacitor's ground lead connected to the oscillator, and (b) the MCU's VSS pin, with the shortest possible wiring (maximum 20 mm).

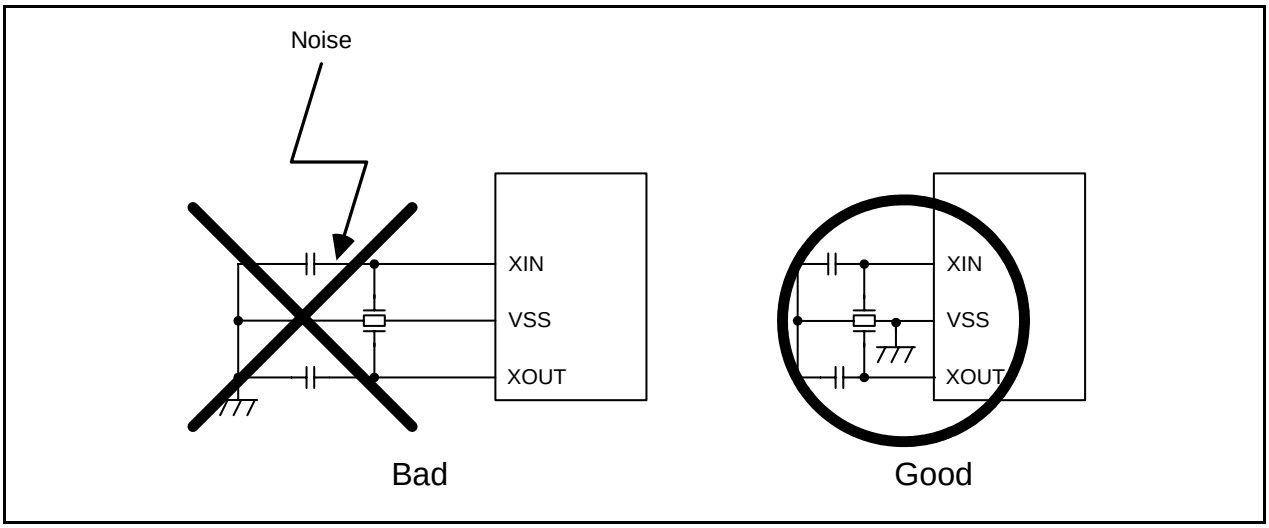


Figure 28.4 Clock I/O Pin Wiring

Reason:
If noise enters the clock I/O pin, the clock waveform becomes unstable, which causes an error in operation or a program runaway. Also, if a potential difference attributed to the noise occurs between the VSS level of the MCU and the VSS level of the oscillator, an accurate clock is not input to the MCU.

28.6.2.2 Large Current Signal Line

For large currents that exceed the MCU's current range, wire the signal lines as far away from the MCU as possible (especially the oscillator).

Reason:
In the system using the MCU, there are signal lines for controlling motors, LEDs, and thermal heads. When a large current flows through these signal lines, noise is generated due to mutual inductance.

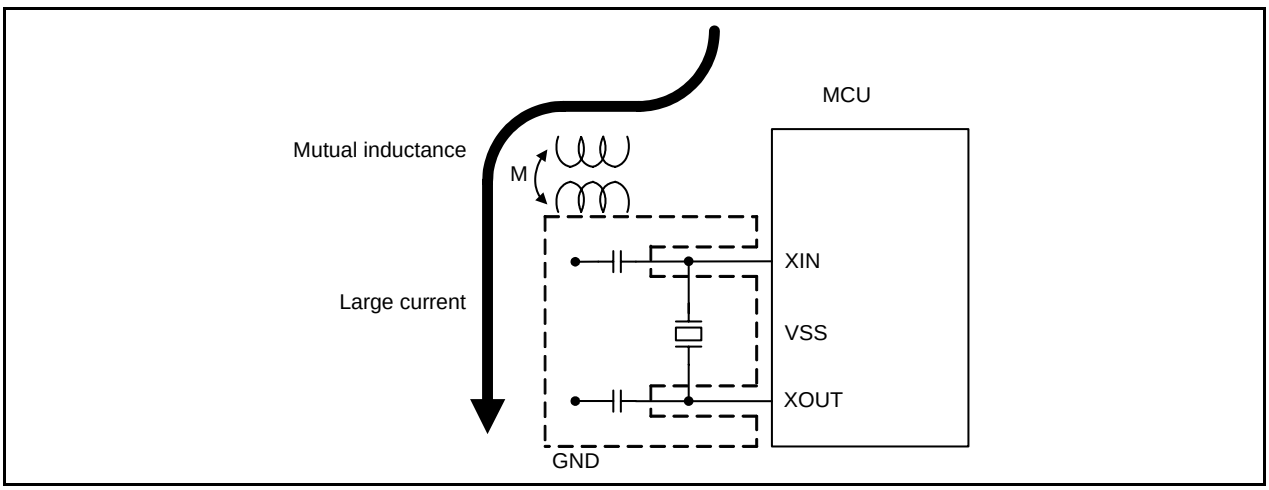


Figure 28.5 Large Current Signal Line Wiring

28.6.2.3 Signal Line Whose Level Changes at a High-Speed

For a signal line whose level changes at a high-speed, wire it as far away from the oscillator and the oscillator wiring pattern as possible. Do not wire it across or extend it parallel to a clock-related signal line or other signal lines which are sensitive to noise.

Reason:
A signal whose level changes at a high-speed (such as the signal from the TAiOUT pin) affects other signal lines due to the level change at rising or falling edges. Specifically, when the signal line crosses the clock-related signal line, the clock waveform becomes unstable, which causes an error in operation or a program runaway.

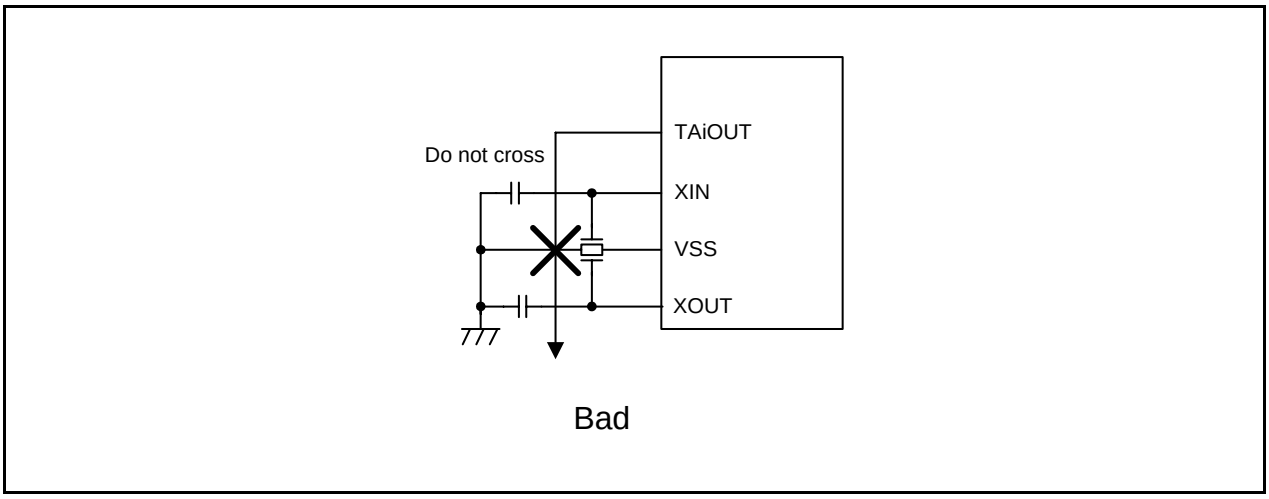


Figure 28.6 Wiring of Signal Line Whose Level Changes at High-Speed

28.6.3 Oscillation Stop/Restart Detect Function

- In the following cases, set the CM20 bit to 0 (oscillation stop/restart detect function disabled), and then change the setting of each bit.
  - When the CM05 bit is set to 1 (main clock stopped)
  - When the CM10 bit is set to 1 (stop mode)
- To enter wait mode while using the oscillation stop/restart detect function, set the CM02 bit to 0 (peripheral function clock f1 not turned off during wait mode).
- This function cannot be used if the main clock frequency is 2 MHz or lower. In that case, set the CM20 bit to 0 (oscillation stop/restart detect function disabled).

28.6.4 PLL Frequency Synthesizer

To use the PLL frequency synthesizer, stabilize the supply voltage within to the acceptable range of power supply ripple.

Table 28.4 Acceptable Range of Power Supply Ripple

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
$f_{(ripple)}$	Power supply ripple allowable frequency (VCC1)			10	kHz
$V_{P-P(ripple)}$	Power supply ripple allowable amplitude voltage			0.3	V
$V_{CC(\lvert\Delta V/\Delta T\rvert)}$	Power supply ripple rising/falling gradient			0.3	V/ms

f(ripple)
Power supply ripple allowable frequency (VCC1)

Vp-p(ripple)
Power supply ripple allowable amplitude voltage

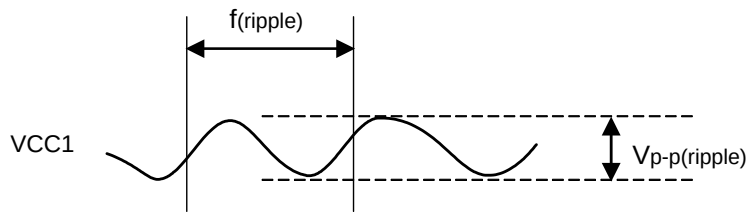


Figure 28.7 Voltage Fluctuation Timing

28.7 Notes on Power Control

28.7.1 CPU Clock

When switching the CPU clock source, wait until oscillation of the switched clock source is stable. After exiting stop mode, wait until oscillation stabilizes before changing the division.

28.7.2 Wait Mode

- Insert four or more NOP instructions following the WAIT instruction. When entering wait mode, because the instruction queue prefetches instructions that follow the WAIT instruction, prefetched instructions are sometimes executed prior to the interrupt routine used to exit wait mode. As shown below, when the instruction to set the I flag to 1 is allocated just before the WAIT instruction, interrupt requests are not accepted before the WAIT instruction is executed.

  The following is an example program for entering wait mode:

```
Program Example:  FSET   I          ;
                  WAIT              ; Enter wait mode
                  NOP               ; Insert at least four NOP instructions
                  NOP
                  NOP
                  NOP
```

- Do not enter wait mode from PLL operating mode. To enter wait mode from PLL operating mode, first enter medium-speed mode, then set the PLC07 bit to 0 (PLL off).
- Do not enter wait mode from low current consumption read mode. To enter wait mode from low current consumption read mode, set the FMR23 bit in the FMR2 register to 0 (low current consumption read mode disabled).
- Do not enter wait mode from CPU rewrite mode. To enter wait mode from CPU rewrite mode, first set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled), then disable the DMA transfer.
- Set the PLC07 bit in the PLC0 register to 0 (PLL off). When the PLC07 bit is 1 (PLL on), current consumption cannot be reduced even in wait mode.

28.7.3 Stop Mode

- When exiting stop mode by a hardware reset, drive the $\overline{\text{RESET}}$ pin low for 20 fOCO-S cycles or more.
- Set the MR0 bit in the TAiMR register (i = 0 to 4) to 0 (pulse not output) when using timer A to exit stop mode.
- When entering stop mode, insert a JMP.B instruction immediately after executing an instruction that sets the CM10 bit in the CM1 register to 1 (stop mode), and then insert at least four NOP instructions. When entering stop mode, the instruction queue reads ahead the instructions following the instruction which sets the CM10 bit to 1. Thus, some of the instructions may be executed before the MCU enters stop mode or before the interrupt routine for returning from stop mode. As shown below, when the instruction to set the I flag to 1 is allocated just before the instruction to set the CM10 bit to 1, interrupt requests are not accepted before entering stop mode.

  The following is an example program for entering stop mode:

```
Program Example:  FSET    I
                  BSET    0, CM1   ; Enter stop mode
                  JMP.B   L2       ; Insert a JMP.B instruction
          L2:
                  NOP              ; At least four NOP instructions
                  NOP
                  NOP
                  NOP
```

- Do not enter stop mode from PLL operating mode. To enter stop mode from PLL operating mode, first enter medium-speed mode, then set the PLC07 bit to 0 (PLL off).
- Do not enter stop mode from low current consumption read mode. To enter stop mode from low current consumption read mode, set the FMR23 bit in the FMR2 register to 0 (low current consumption read mode disabled).
- Do not enter stop mode from CPU rewrite mode. To enter stop mode from CPU rewrite mode, first set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled), then disable the DMA transfer.
- Do not enter stop mode when the oscillator stop/restart detect function is enabled. To enter stop mode, set the CM20 bit in the CM2 register to 0 (oscillator stop/restart detect function disabled).
- Entering stop mode is disabled when the FMR01 bit is 1 (CPU rewrite mode enabled). Therefore, do not enter stop mode when the flash memory is stopped (bits FMR01 and FMSTP are 1).

28.7.4 Low Current Consumption Read Mode

- Enter low current consumption read mode through slow read mode (see Figure 9.5 “Setting and Canceling Low Current Consumption Read Mode”).
- When the FMR23 bit in the FMR2 register is 1 (low current consumption read mode enabled), do not set the FMSTP bit to 1 (flash memory stopped). Also, when the FMSTP bit is 1, do not set the FMR23 bit to 1.
- When the FMR01 bit in the FMR0 register to 1 (CPU rewrite mode enabled), do not set the FMR23 bit in the FMR2 register to 1 (low current consumption read mode enable).

28.7.5 Slow Read Mode

When the FMR01 bit in the FMR0 register to 1 (CPU rewrite mode enabled), do not set the FMR22 bit in the FMR2 register to 1 (slow read mode enabled).

28.8 Notes on Bus

28.8.1 Reading Data Flash

When 2.7 V ≤ VCC1 ≤ 3.0 V, one wait must be inserted to read the data flash. Use the PM17 bit or the FMR17 bit to insert one wait.

28.9 Notes on Programmable I/O Ports

Note

P2 to P5 have no external connections. These ports are connected to PLC modem internally.

28.9.1 Influence of SI/O3 and SI/O4

Setting the SM32 bit in the S3C register to 1 causes the P9_2 pin to become high-impedance. Similarly, setting the SM42 bit in the S4C register to 1 causes the P9_6 pin to become high-impedance.

28.10 Notes on Interrupts

Note

No external pin is provided for $\overline{\text{INT0}}$, $\overline{\text{INT6}}$ and $\overline{\text{INT7}}$ because it is internally connected to the PLC modem.

28.10.1 Reading Address 00000h

Do not read address 00000h by a program. When a maskable interrupt request is accepted, the CPU reads interrupt information (interrupt number and interrupt request priority level) from address 00000h during the interrupt sequence. At this time, the IR bit of the accepted interrupt is cleared to 0 (interrupt not requested).

If address 00000h is read by a program, the IR bit for the interrupt which has the highest priority among the enabled interrupts becomes 0. Thus, some problems may be caused: interrupts may be canceled, or an unexpected interrupt request may be generated.

28.10.2 SP Setting

Set a value in the SP (USP, ISP) before accepting an interrupt. The SP (USP, ISP) is set to 0000h after reset. Therefore, if an interrupt is accepted before setting a value in the SP (USP, ISP), the program may go out of control.

Set a value in the ISP at the beginning of the program. For the first instruction after reset only, all interrupts including the $\overline{\text{NMI}}$ interrupt are disabled.

28.10.3 $\overline{\text{NMI}}$ Interrupt

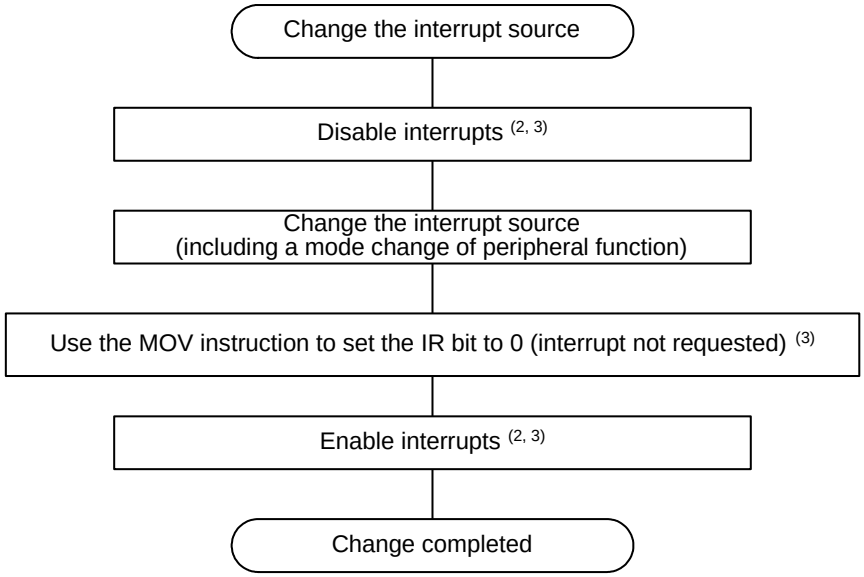
- When not using the $\overline{\text{NMI}}$ interrupt, set the PM24 bit in the PM2 register to 0 ($\overline{\text{NMI}}$ interrupt disabled).
- The $\overline{\text{NMI}}$ interrupt is disabled after reset. The $\overline{\text{NMI}}$ interrupt is enabled by setting the PM24 bit in the PM2 register to 1. Set the PM24 bit to 1 when a high-level signal is applied to the $\overline{\text{NMI}}$ pin. When the PM24 bit is set to 1 while a low-level signal is applied, an $\overline{\text{NMI}}$ interrupt is generated. Once the $\overline{\text{NMI}}$ interrupt is enabled, it cannot be disabled until the MCU is reset.
- Stop mode cannot be entered while the PM24 bit is 1 ($\overline{\text{NMI}}$ interrupt enabled) and input on the $\overline{\text{NMI}}$ pin is low. When input on the $\overline{\text{NMI}}$ pin is low, the CM10 bit in the CM1 register is fixed to 0.
- Do not enter wait mode while the PM24 bit is 1 ($\overline{\text{NMI}}$ interrupt enabled) and input on the $\overline{\text{NMI}}$ pin is low because the CPU clock remains active even though the CPU stops, and therefore, current consumption of the chip does not drop. In this case, normal condition is restored by the next interrupt generated.
- Set the low- and high-level durations of the input signal to the $\overline{\text{NMI}}$ pin to 2 CPU clock cycles + 300 ns or more.

28.10.4 Changing an Interrupt Source

When the interrupt source is changed, the IR bit in the interrupt control register may inadvertently become 1 (interrupt requested). To use an interrupt, change the interrupt source, and then set the IR bit to 0 (interrupt not requested).

In this section, the changing of an interrupt source refers to all elements (e.g. changing the mode of a peripheral function) used in changing the interrupt source, polarity, and timing assigned to each software interrupt number. When using an element to change the interrupt source, polarity, or timing, make the change before setting the IR bit to 0 (interrupt not requested). Refer to the descriptions of the individual peripheral functions for details of the peripheral function interrupts.

Figure 28.8 shows the Procedure for Changing the Interrupt Generate Factor.



IR bit: A bit in the interrupt control register

Notes:
1. Execute the above settings individually. Do not use one instruction to execute two or more settings simultaneously.
2. Use the I flag for the $\overline{\text{INTi}}$ interrupt (i = 0 to 7).
   For the interrupts from peripheral functions other than the $\overline{\text{INTi}}$ interrupt, disable the peripheral function that is the source of the interrupt so as not to generate an interrupt request before changing the interrupt source. In this case, if the maskable interrupts can all be disabled without causing a problem, use the I flag. Otherwise, use corresponding bits ILVL2 to ILVL0 in the interrupt control register whose interrupt source is to be changed.
3. Refer to 13.13.5 “Rewriting the Interrupt Control Register” below for details about the instructions to be used and the notes to be taken for instruction execution.

Figure 28.8 Procedure for Changing the Interrupt Generate Factor

28.10.5 Rewriting the Interrupt Control Register

To modify the interrupt control register, follow either of the procedures below:

- Modify in places where no requests for the interrupt control register may occur.
- If an interrupt request can be generated, disable that interrupt and then rewrite the interrupt control register.

When using the I flag to disable an interrupt, set the I flag as shown in the sample program code below. (Refer to 28.10.6 “Instruction to Rewrite the Interrupt Control Register” for rewriting the interrupt control registers using the sample program code.)

Examples 1 through 3 show how to prevent the I flag from becoming 1 (interrupt enabled) before the contents of the interrupt control register is rewritten, owing to the effects of the internal bus and the instruction queue buffer.

Example 1: Using the NOP instruction to pause the program until the interrupt control register is modified

```
INT_SWITCH1:
    FCLR    I               ; Disable interrupts.
    AND.B   #00H, 0055H     ; Set the TA0IC register to 00h.
    NOP                     ;
    NOP
    FSET    I               ; Enable interrupts.
```

Example 2: Using a dummy read to delay the FSET instruction

```
INT_SWITCH2:
    FCLR    I               ; Disable interrupts.
    AND.B   #00H, 0055H     ; Set the TA0IC register to 00h.
    MOV.W   MEM, R0         ; Dummy read.
    FSET    I               ; Enable interrupts.
```

Example 3: Using the POPC instruction to change the I flag

```
INT_SWITCH3:
    PUSHC   FLG
    FCLR    I               ; Disable interrupts.
    AND.B   #00H, 0055H     ; Set the TA0IC register to 00h.
    POPC    FLG             ; Enable interrupts.
```

28.10.6 Instruction to Rewrite the Interrupt Control Register

- Do not use the BTSTC and BTSTS instructions to rewrite the interrupt control registers.
- Use the AND, OR, BCLR, BSET, or MOV instruction to rewrite interrupt control registers.
  When an interrupt request is generated for the register being rewritten while executing an AND, OR, BCLR, BSET, or MOV instruction, the IR bit becomes 1 (interrupt requested) and remains 1.

28.10.7 $\overline{\text{INT}}$ Interrupt

- Either a low level of at least tw(INL) width or a high level of at least tw(INH) width is necessary for the signal input to pins $\overline{\text{INT1}}$ through $\overline{\text{INT5}}$ regardless of the CPU operation clock.
- If the POL bit in registers INT0IC to INT7IC, bits IFSR7 to IFSR0 in the IFSR register, or bits IFSR31 to IFSR30 in the IFSR3A register are changed, the IR bit may inadvertently become 1 (interrupt requested). Be sure to set the IR bit to 0 (interrupt not requested) after changing any of these register bits.

28.11 Notes on Watchdog Timer

After a watchdog timer interrupt is generated, use the WDTR register to refresh the watchdog timer counter.

28.12 Notes on DMAC

28.12.1 Write to the DMAE Bit in the DMiCON Register (i = 0 to 3)

When both of following conditions are met, follow steps (1) and (2) below.

- Write a 1 (DMAi is in active state) to the DMAE bit when it is 1.
- A DMA request may be generated at the same time the DMAE bit is being written.

Steps

(1) Set bits DMAE and DMAS in the DMiCON register to 1 simultaneously (1).

(2) Make sure that the DMAi circuit is in an initialized state (2) in a program.
If the DMAi is not in an initialized state, repeat these two steps.

Notes:

1. The DMAS bit remains unchanged even if set to 1. However, it becomes 0 when set to 0 (DMA not requested). To prevent the DMAS bit from being modified to 0, 1 should be written to the DMAS bit when 1 is written to the DMAE bit. This setting allows the DMAS bit to retain its value previous to being rewritten.
Similarly, when writing to the DMAE bit with a read-modify-write instruction, set the DMAS bit to 1 to retain the DMA request that was generated while executing the instruction.
2. Read the TCRi register to verify whether the DMAi is in an initialized state.
If the read value is equal to a value that was written to the TCRi register before DMA transfer starts, the DMAi is in an initialized state. (When a DMA request is generated after writing to the DMAE bit, the read value is a value written to the TCRi register minus 1.) If the read value is a value in the middle of a transfer, the DMAi is not in an initialized state.

28.12.2 Changing the DMA Request Source

When the DMS bit or bits DSEL4 to DSEL0 in the DMiSL register are changed, the DMAS bit in the DMiCON sometimes becomes 1 (DMA requested). Set the DMAS bit to 0 (DMA not requested) after the DMS bit or bits DSEL4 to DSEL0 in the DMiSL register are changed.

28.13 Notes on Timer A

Note

Pins TA0IN and TA0OUT are not provided for timer A0 because it is internally connected to the PLC modem.

28.13.1 Common Notes on Multiple Modes

28.13.1.1 Register Setting

The timer stops after reset. Set the mode, count source, counter value, etc., using registers TAiMR, TAi, TAi1, UDF, TRGSR, PWMFS, TACS0 to TACS2, TAPOFS, PCLKR, and bits TAZIE, TA0TGL, and TA0TGH in the ONSF register before setting the TAiS bit in the TABSR register to 1 (count started) (i = 0 to 4).

Always make sure registers TAiMR, UDF, TRGSR, PWMFS, TACS0 to TACS2, TAPOFS, PCLKR and bits TAZIE, TA0TGL, TA0TGH in the ONSF register are modified while the TAiS bit is 0 (count stopped), regardless of whether after reset or not.

28.13.1.2 Event or Trigger

When bits TAiTGH to TAiTGL in the registers ONSF or TRGSR are 01b, 10b, or 11b, an event or a trigger occurs when an interrupt request of the selected timer is generated. An event or trigger occurs while an interrupt is disabled because an interrupt request signal is generated regardless of the I flag, IPL, or interrupt control registers.

For some modes of the timers selected using bits TAiTGH to TAiTGL, an interrupt request is generated by a source other than overflow or underflow.

For example, when using pulse-period measurement mode or pulse-width measurement mode in timer B2, an interrupt request is generated at an active edge of the measurement pulse. For details, refer to the “Interrupt request generation timing” in each mode’s specification table.

28.13.1.3 Influence of $\overline{SD}$

When a low-level signal is applied to the $\overline{SD}$ pin while the IVPCR1 bit in the TB2SC register is 1 (three-phase output forcible cutoff by input on $\overline{SD}$ pin enabled), the following pins become high-impedance:

P7_2/CLK2/TA1OUT/V, P7_3/$\overline{CTS2}$/$\overline{RTS2}$/TA1IN/V, P7_4/TA2OUT/W,
P7_5/TA2IN/W, P8_0/TA4OUT/RXD5/SCL5/U, P8_1/TA4IN/$\overline{CTS5}$/$\overline{RTS5}$/U

28.13.2 Timer A (Timer Mode)

28.13.2.1 Read from Timer

While counting, the counter value can be read at any time by reading the TAi register. However, if the counter is read at the same time as it is reloaded, the read value is FFFFh. Also, if the counter is read before it starts counting, or after a value is set in the TAi register while not counting, the set value is read.

28.13.3 Timer A (Event Counter Mode)

28.13.3.1 Read from Timer

While counting, the counter value can be read at any time by reading the TAi register. However, while reloading, FFFFh can be read in underflow, and 0000h in overflow. When the counter is read before it starts counting and after a value is set in the TAi register while not counting, the set value is read.

28.13.4 Timer A (One-Shot Timer Mode)

28.13.4.1 Stop While Counting

When setting the TAiS bit to 0 (count stopped), the following occurs:

- The counter stops counting and the contents of the reload register are reloaded.
- The TAiOUT pin outputs a low-level signal when the POFSi bit in the TAPOFS register is 0, and outputs a high-level signal when it is 1.
- After one cycle of the CPU clock, the IR bit in the TAiIC register becomes 1 (interrupt requested).

28.13.4.2 Delay between the Trigger Input and Timer Output

As the one-shot timer output is synchronized with an internally generated count source, when an external trigger is selected, a maximum 1.5 cycle delay of the count source occurs between the trigger input to the TAiIN pin and timer output.

28.13.4.3 Changing Operating Modes

The IR bit becomes 1 when the timer operating mode is set with any of the following:

- Selecting one-shot timer mode after reset
- Changing the operating mode from timer mode to one-shot timer mode
- Changing the operating mode from event counter mode to one-shot timer mode

To use the timer Ai interrupt (IR bit), set the IR bit to 0 after the changes listed above are made.

28.13.4.4 Retrigger

When a trigger occurs while counting, the counter reloads the reload register to continue counting after generating a retrigger and decrementing once. To generate a trigger while counting, generate a retrigger after more than one cycle of the timer count source has elapsed following the previous trigger.

When an external trigger is generated, do not generate a retrigger for 300 ns before the count value becomes 0000h. The one-shot timer may stop counting.

28.13.5 Timer A (Pulse Width Modulation Mode)

28.13.5.1 Changing Operating Modes

The IR bit becomes 1 when setting a timer operating mode with any of the following:
- Selecting PWM mode or programmable output mode after reset
- Changing the operating mode from timer mode to PWM mode or programmable output mode
- Changing the operating mode from event counter mode to PWM mode or programmable output mode

To use the timer Ai interrupt (IR bit), set the IR bit to 0 by a program after the changes listed above are made.

28.13.5.2 Stop While Counting

When setting the TAiS bit to 0 (count stopped) during PWM pulse output, the following occur:
When the POFSi bit in the TAPOFS register is 0:
- Counting stops
- When the TAiOUT pin is high, the output level goes low and the IR bit becomes 1.
- When the TAiOUT pin is low, both the output level and the IR bit remain unchanged.

When the POFSi bit in the TAPOFS register is 1:
- Stop counting.
- If the TAiOUT pin output is low, the output level goes high and the IR bit is set to 1.
- If the TAiOUT pin output is high, both the output level and the IR bit remain unchanged.

28.13.6 Timer A (Programmable Output Mode)

28.13.6.1 Changing the Operating Mode

The IR bit becomes 1 when setting a timer operating mode with any of the following:
- Selecting PWM mode or programmable output mode after reset
- Changing the operating mode from timer mode to PWM mode or programmable output mode
- Changing the operating mode from event counter mode to PWM mode or programmable output mode

To use the timer Ai interrupt (IR bit), set the IR bit to 0 by a program after the changes listed above are made.

28.13.6.2 Stop While Counting

When setting the TAiS bit to 0 (count stopped) during pulse output, the following occur:
When the POFSi bit in the TAPOFS register is 0:
- Counting stops.
- When the TAiOUT pin is high, the output level goes low.
- When the TAiOUT pin is low, the output level remains unchanged.
- The IR bit remains unchanged.

When the POFSi bit in the TAPOFS register is 1:
- Counting stops
- When the TAiOUT pin output is low, the output level goes high.
- When the TAiOUT pin output is high, the output level remains unchanged.
- The IR bit remains unchanged.

28.14 Notes on Timer B

28.14.1 Common Notes on Multiple Modes

28.14.1.1 Register Setting

The timer is stopped after reset. Set the mode, count source, etc., using registers TBiMR, TBCS0 to TBCS3, TBi, PCLKR, PPWFS1, and PPWFS2 before setting the TBiS bit in the TABSR or TBSR register to 1 (count started) (i = 0 to 5).
Always make sure registers TBiMR, TBCS0 to TBCS3, PCLKR, PPWFS1, and PPWFS2 are modified while the TBiS bit is 0 (count stopped), regardless of whether after reset or not.

28.14.2 Timer B (Timer Mode)

28.14.2.1 Read from Timer

While counting, the counter value can be read at any time by reading the TBi register. However, FFFFh is read while reloading. When the counter is read before it starts counting and after a value is set in the TBi register while not counting, the set value is read.

28.14.3 Timer B (Event Counter Mode)

28.14.3.1 Read from Timer

While counting, the counter value can be read at any time by reading the TBi register. However, FFFFh is read while reloading. When the counter is read before it starts counting and after a value is set in the TBi register while not counting, the set value is read.

28.14.3.2 Event

When the TCK1 bit in the TBiMR register is 1, an event occurs when an interrupt request of the selected timer is generated. An event or trigger occurs while an interrupt is disabled because an interrupt request signal is generated regardless of the I flag, IPL, or interrupt control registers.
When the timer selected by the TCK1 bit uses pulse-period measurement mode or pulse-width measurement mode, an interrupt request is generated at an active edge of the measurement pulse.

28.14.4 Timer B (Pulse Period/Pulse Width Measurement Modes)

28.14.4.1 The MR3 Bit in the TBiMR Register

To clear the MR3 bit to 0 by writing to the TBiMR register while the TBiS bit is 1 (count started), be sure to set the same value as previously set to bits TMOD0, TMOD1, MR0, MR1, TCK0, and TCK1, and set bit 4 to 0.

28.14.4.2 Interrupts

The IR bit in the TBiIC register becomes 1 (interrupt requested) when an active edge of a measurement pulse is input or timer Bi overflows (i = 0 to 5). The source of an interrupt request can be determined using the MR3 bit in the TBiMR register within the interrupt routine.
Use the IR bit in the TBiIC register to detect overflows only. Use the MR3 bit only to determine the interrupt source.

28.14.4.3 Event or Trigger

When timer Bi in pulse-period measurement mode or pulse-width measurement mode is used as an event or trigger for timer A or timer B other than timer Bi, an event or trigger occurs at both the overflow and active edge of the measurement pulse.

28.14.4.4 Operations between Count Start and the First Measurement

When a count is started and the first active edge is input, an undefined value is transferred to the reload register. At this time, a timer Bi interrupt request is not generated.

The value of the counter is undefined after reset. If a count is started in this state, the MR3 bit may become 1 and a timer Bi interrupt request may be generated after the count starts before an active edge is input. When a value is set in the TBi register while the TBiS bit is 0 (count stopped), the same value is written to the counter.

28.14.4.5 Pulse Period Measurement Mode

When active edge and overflow are generated simultaneously, input is not recognized at the active edge because an interrupt request is generated only once. Use this mode so an overflow is not generated, or use pulse width measurement.

28.14.4.6 Pulse Width Measurement Mode

In pulse width measurement, pulse widths are measured successively. Check whether the measurement result is a high-level width or a low-level width in the user program.
When an interrupt request is generated, read the TBiIN pin level inside the interrupt routine, and check whether it is the edge of an input pulse or overflow. The TBiIN pin level can be read from bits in the register of ports sharing a pin.

28.15 Notes on Real-Time Clock

28.15.1 Starting and Stopping the Count

The real-time clock uses the TSTART bit for instructing the count to start or stop, and the TCSTF bit which indicates count started or stopped. Bits TSTART and TCSTF are in the RTCCR1 register.

The real-time clock starts counting and the TCSTF bit becomes 1 (count started) when the TSTART bit is set to 1 (count started). It takes up to two cycles of the count source until the TCSTF bit becomes 1 after setting the TSTART bit to 1. During this time, do not access registers associated with the real-time clock (1) other than the TCSTF bit.

Similarly, when setting the TSTART bit to 0 (count stopped), the real-time clock stops counting and the TCSTF bit becomes 0 (count stopped). It takes up to three cycles of the count source until the TCSTF bit becomes 0 after setting the TSTART bit to 0. During this time, do not access registers associated with the real-time clock other than the TCSTF bit.

Note:

1. Registers associated with the real-time clock: RTCSEC, RTCMIN, RTCHR, RTCWK, RTCCR1, RTCCR2, RTCCSR, RTCCSEC, RTCCMIN, and RTCCHR.

28.15.2 Register Setting (Time Data, etc.)

Write to the following registers/bits when the RUN bit in the TRHCR register is 0 (count stopped):

- Registers TRHSEC, TRHMIN, TRHHR, TRHWK, TRHDY, TRHMON, TRHYR, and TRHIER
- Bits TRHOE, HR 24, and PM in the TRHCR register
- Bits OS2 to OS1 in the TRHCSR register

Set the TRHIER register after setting other registers and bits mentioned above (immediately before the real-time clock count starts).

28.15.3 Register Setting (Alarm Data)

Write to the following registers when the BSY bit in the TRHSEC register is 0 (not while data is updated).

- Registers TRHAMN, TRHAHR, and TRHAWK

28.15.4 Time Reading Procedure of Real-Time Clock Mode

In real-time clock mode, read time data bits (1) when the BSY bit in the TRHSEC register is 0 (not while data is updated).
When reading multiple registers, if data is rewritten between reading registers, an errant time will be read. To prevent this, use the procedure below when reading:

- Using an interrupt
  Read necessary contents of time data bits in the real-time clock periodic interrupt routine.
- Monitoring by a program 1
  Monitor the IR bit in the RTCTIC register by a program and read necessary contents of time data bits after the IR bit becomes 1 (periodic interrupt requested).
- Monitoring by a program 2
  Read the time data according to Figure 28.9 “Time Data Reading”.

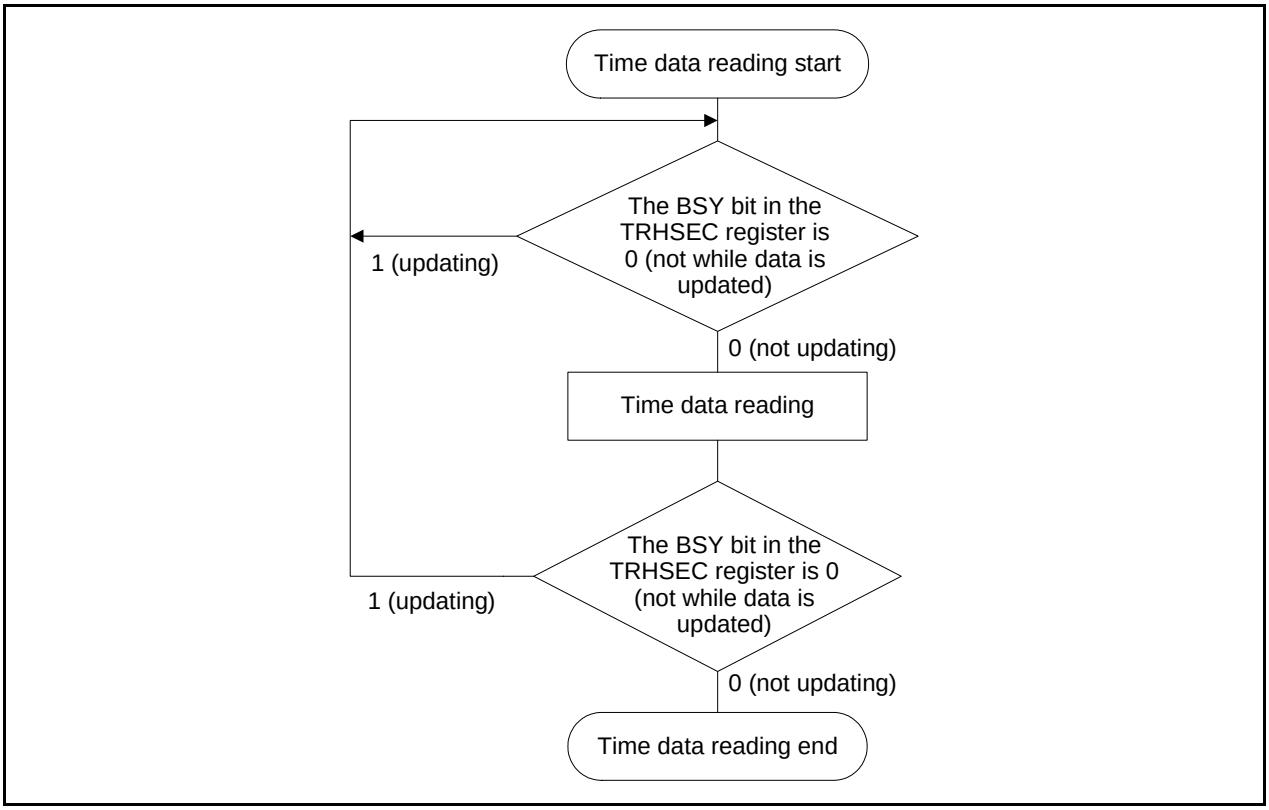


Figure 28.9 Time Data Reading

Also, when reading multiple registers, read them as continuously as possible.

Note:
1. Time data bits are as follows:
   Bits SC12 to SC10 and SC03 to SC00 in the TRHSEC register
   Bits MN12 to MN10 and MN03 to MN00 in the TRHMIN register
   Bits HR11 to HR10 and HR03 to HR00 in the TRHHR register
   Bits WK2 to WK0 in the TRHWK register
   The PM bit in the TRHCR register
   Bits DY11 to DY10 and DY03 to DY00 in the TRHDY register
   Bits MO10 and MO03 to MO00 in the TRHMON register
   Bits YR13 to YR10 and YR03 to YR00 in the TRHYR register

28.16 Notes on Serial Interface UARTi (i = 0 to 2, 5 to 7)

Note

No external pin is provided for UART7 because it is internally connected to the PLC modem.

28.16.1 Common Notes on Multiple Modes

28.16.1.1 CLKi Output

(Technical update number: TN-M16C-A178A/E)

When using the N-channel open drain output as an output mode of the CLKi pin, use following procedure to change the pin function:

When changing the pin function from the port to CLKi.
(1) Set bits SMD2 to SMD0 in the UiMR register to a value other than 000b to select serial interface mode.
(2) Set the NODC bit in the UiSMR3 register to 1.

When changing the pin function from CLKi to the port.
(1) Set the NODC bit to 0.
(2) Set bits SMD2 to SMD0 to 000b to disable the serial interface.

28.16.2 Clock Synchronous Serial I/O Mode

28.16.2.1 Transmission/Reception

When the $\overline{\text{RTS}}$ function is used with an external clock, the $\overline{\text{RTSi}}$ pin (i = 0 to 2, 5, 6) outputs a low-level signal, which informs the transmitting side that the MCU is ready for a receive operation. The $\overline{\text{RTSi}}$ pin outputs a high-level signal when a receive operation starts. Therefore, transmit timing and receive timing can be synchronized by connecting the $\overline{\text{RTSi}}$ pin to the $\overline{\text{CTSi}}$ pin on the transmitting side. The $\overline{\text{RTS}}$ function is disabled when an internal clock is selected.

28.16.2.2 Transmission

If the transmission is started while an external clock is selected and the TXEPT bit in the UiC0 register (i = 0 to 4) is 1 (no data present in transmit register), meet the last requirement at either of the following timings:

External clock level:
- The CKPOL bit in the UiC0 register is 0 (transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge) and the external clock is high.
- The CKPOL bit is 1 (transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge) and the external clock is low.

Requirements to start transmission (in no particular order):
- The TE bit in the UiC1 register is 1 (transmission enabled).
- The TI bit in the UiC1 register is 0 (data present in the UiTB register).
- When the CTS function is selected, input on the CTSi pin is low.

28.16.2.3 Reception

In clock synchronous serial I/O mode, a shift clock is generated by activating a transmitter. Set the UARTi-associated registers for a transmit operation even if the MCU is used for a receive operations only. Dummy data is output from the TXDi pin (i = 0 to 2, 5, 6) while receiving.

When an internal clock is selected, a shift clock is generated by setting the TE bit in the UiC1 register to 1 (transmission enabled) and placing dummy data in the UiTB register. When an external clock is selected, set the TE bit to 1 (transmission enabled), set dummy data in the UiTB register, and input an external clock to the CLKi pin to generate a shift clock.

If data is received consecutively, an overrun error occurs when the RI bit in the UiC1 register is 1 (data present in the UiRB register) and the next receive data is received in the UARTi receive register. Then, the OER bit in the UiRB register becomes 1 (overrun error occurred). At this time, the UiRB register is undefined. When an overrun error occurs, program the transmitting and receiving sides to retransmit the previous data. If an overrun error occurs again, the IR bit in the SiRIC register remains unchanged.

To receive data consecutively, set dummy data in the low-order byte in the UiTB register for each receive operation.

External clock level:
- The CKPOL bit in the UiC0 register is 0 (transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge) and the external clock is high.
- The CKPOL bit is 1 (transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge) and the external clock is low.

Requirements to start reception (in no particular order):
- The RE bit in the UiC1 register is 1 (reception enabled).
- The TE bit in the UiC1 register is 1 (transmission enabled).
- The TI bit in the UiC1 register is 0 (data present in the UiTB register).

28.16.3 Special Mode (I^2C Mode)

28.16.3.1 Generating Start and Stop Conditions

When generating start, stop, and restart conditions, set the STSPSEL bit in the UiSMR4 register (i = 0 to 2, 5, 6) to 0 and wait for more than a half cycle of the transmit/receive clock. Then set each condition generation bit (STAREQ, RSTAREQ, and STPREQ) from 0 to 1.

28.16.3.2 IR Bit

Set the following bits first, and then set the IR bit in the UARTi interrupt control registers to 0 (interrupt not requested).

Bits SMD2 to SMD0 in the UiMR register, the IICM bit in the UiSMR register,
the IICM2 bit in the UiSMR2 register, the CKPH bit in the UiSMR3 register

28.16.3.3 Low/High-level Input Voltage and Low-level Output Voltage

The low-level input voltage, high-level input voltage, and low-level output voltage differ from the I^2C-bus specification.
Refer to the recommended operating conditions for I/O ports which share the pins with SCL and SDA.

I^2C-bus specification
High level input voltage (V_{IH}) = min. 0.7 V_{CC}
Low level input voltage (V_{IL}) = max. 0.3 V_{CC}

28.16.3.4 Setup and Hold Times When Generating a Start/Stop Condition

When generating a start condition, the hold time (t_{HD}:STA) is a half cycle of the SCL clock.
When generating a stop condition, the setup time (t_{SU}:STO) is a half cycle of the SCL clock.
When the SDA digital delay function is enabled, take delay time into consideration (see 19.3.3.7 “SDA Digital Delay”).
The following shows a calculation example of hold and setup times when generating a start/stop condition.

Calculation example when setting 100 kbps
- U2iBRG count source: f1 = 20 MHz
- U2iBRG register setting value: n = 100 - 1
- SDA digital delay setting value: DL2 to DL0 are 101b (5 or 6 cycles of U2iBRG count source)

f_{SCL} (theoretical value) = f1 / (2(n+1)) = 20 MHz / (2 × (99 + 1)) = 100 kbps
t_{DL} = delay cycle count / f1 = 6 / 20 MHz = 0.3 μs
$t_{HD:STA}$ (theoretical value) = 1 / (2f_{SCL} (theoretical value)) = 1 / (2 × 100 kbps) = 5 μs
$t_{SU:STO}$ (theoretical value) = 1 / (2f_{SCL} (theoretical value)) = 1 / (2 × 100 kbps) = 5 μs
$f_{HD:STA}$ (actual value) = $t_{HD:STA}$ (theoretical value) - t_{DL} = 5 μs - 0.3 μs = 4.7 μs
$f_{SU:STO}$ (actual value) = $t_{SU:STO}$ (theoretical value) + t_{DL} = 5 μs + 0.3 μs = 5.3 μs

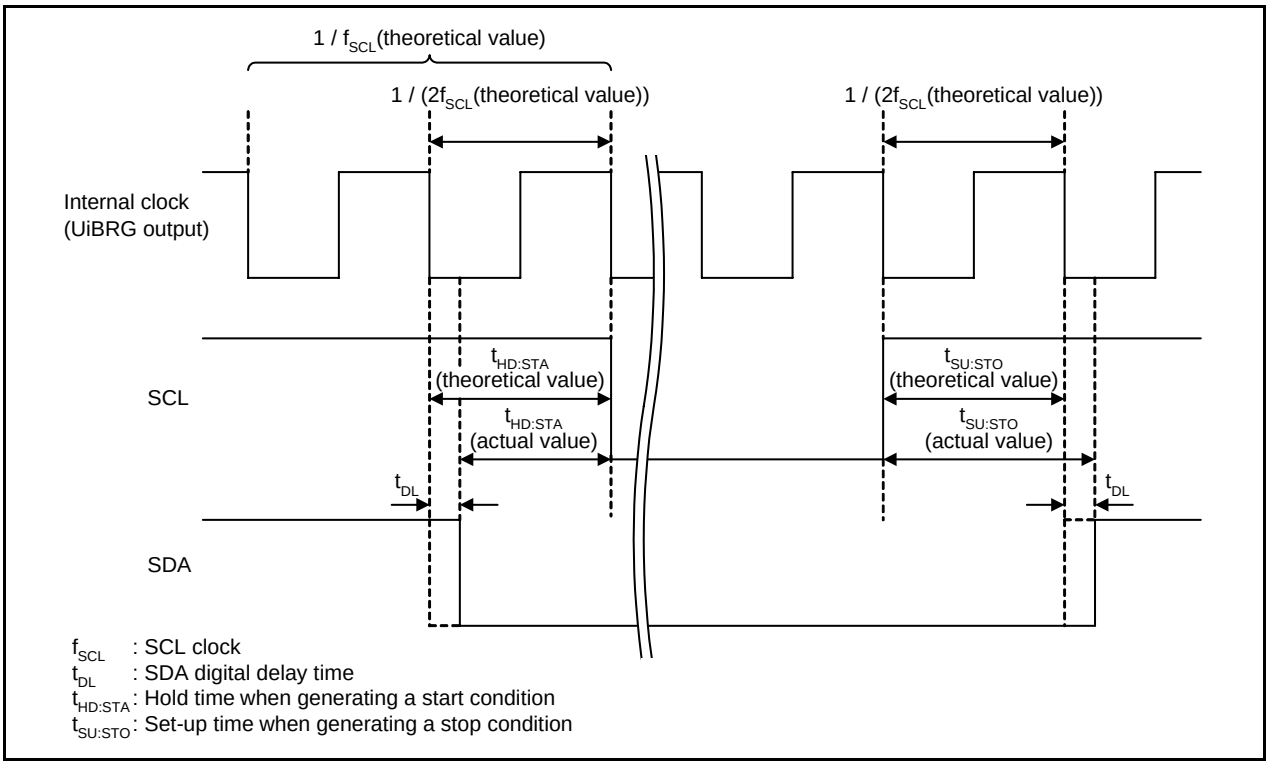


Figure 28.10 Setup and Hold Times When Generating Start and Stop Conditions

28.16.3.5 Restrictions on the Bit Rate When Using the Ui2BRG Count Source

In I^2C mode, set the Ui2BRG register to a value of 03h or greater.

A maximum of three Ui2BRG count source cycles are necessary until the internal circuit acknowledges the SCL clock level. The connectable I^2C-bus bit rate is one-third or less than the Ui2BRG count source speed. If a value between 00h to 02h is set to the Ui2BRG register, bit slippage may occur.

28.16.3.6 Restart Condition in Slave Mode

When a restart condition is detected in slave mode, the successive processes may not be executed correctly. In slave mode, do not use a restart condition.

28.16.3.7 Requirements to Start Transmission/Reception in Slave Mode

When transmission/reception is started in slave mode and the TXEPT bit in the UiC0 register is 1 (no data present in transmit register), meet the last requirement when the external clock is high.

Requirements to start transmission (in no particular order):

- The TE bit in the Ui2C1 register is 1 (transmission enabled).
- The TI bit in the Ui2C1 register is 0 (data present in the UiTB register).

Requirements to start reception (in no particular order):

- The RE bit in the Ui2C1 register is 1 (reception enabled).
- The TE bit in the Ui2C1 register is 1 (transmission enabled).
- The TI bit in the Ui2C1 register is 0 (data present in the UiTB register).

28.16.4 Special Mode 4 (SIM Mode)

After reset, a transmit interrupt request is generated by setting bits U2IRS and U2ERE in the U2C1 register to 1 (transmission completed) and 1 (error signal output), respectively. Therefore, when using SIM mode, make sure to set the IR bit to 0 (interrupt not requested) after setting these bits.

28.17 Notes on Serial Interface SI/O3 and SI/O4

28.17.1 SOUTi Pin Level When SOUTi Output Is Disabled

When the SMi2 bit in the SiC register is set to 1 (SOUTi output disabled), the target pin becomes high-impedance regardless of which pin function being used.

28.17.2 External Clock Control

The data written to the SiTRR register shifts each time the external clock is input. When completing data transmission/reception of the eighth bit, read or write to the SiTRR register before inputting the clock for the next data transmission/reception.

28.17.3 Register Access

Set the SM22 bit in the S34C2 register before setting other registers associated with SI/O3 and SI/O4. After changing the SM22 bit, set other registers associated with SI/O3 and SI/O4 again.

28.17.4 Register Access When Using the External Clock

When the SMi6 bit in the SiC register is 0 (external clock), write to the SMi7 bit in the SiC register and SiTRR register under the following conditions:

- When the SMi4 bit in the SiC register is 0 (transmit data is output at the falling edge of transmit/receive clock and receive data is input at the rising edge): CLKi input is high level.
- When the SMi4 bit in the SiC register is 1 (transmit data is output at the rising edge of transmit/receive clock and receive data is input at the falling edge): CLKi input is low level.

28.17.5 SiTRR Register Access

Write transmit data to the SiTRR register while transmission/reception is stopped. Read receive data from the SiTRR register while transmission/reception is stopped.
The IR bit in the SiIC register becomes 1 (interrupt requested) during output of the eighth bit.
When the SM26 bit (SOUT3) or SM27 bit (SOUT4) in the S34C2 register is 0 (high-impedance after transmission), the SOUTi pin becomes high-impedance when the transmit data is written to the SiTRR register immediately after an interrupt request is generated, and the hold time of the transmit data becomes shorter.

28.17.6 Pin Function Switch When Using the Internal Clock

If the SMi3 bit in the SiC register (i = 3, 4) changes from 0 (I/O port) to 1 (SOUTi output, CLKi function) when setting the SMi2 bit to 0 (SOUTi output) and the SMi6 bit to 1 (internal clock), the SOUTi initial value set to the SOUTi pin by the SMi7 bit may be output for about 10 ns. Then, the SOUTi pin becomes high-impedance.
If the output level from the SOUTi pin when the SMi3 bit changes from 0 to 1 becomes a problem, set the SOUTi initial value by the SMi7 bit.

28.17.7 Operation after Reset When Selecting the External Clock

When the SMi6 bit in the SiC register is 0 (external clock) after reset, the IR bit in the SiIC register becomes 1 (interrupt requested) by inputting the external clock for 8 bits to the CLKi pin. This will also occur even when the SMi3 bit in the SiC register is 0 (serial interface disabled) or before a value is written to the SiTRR register.

28.18 Notes on Multi-Master I2C-bus Interface

28.18.1 Limitation on CPU Clock

When the CM07 bit in the CM0 register is 1 (CPU clock is a sub clock), do not access the registers listed in Table 21.4 “Register Configuration”. Set the CM07 bit to 0 (main clock, PLL clock, or on-chip oscillator clock) to access these registers.

28.18.2 Register Access

Refer to the notes below when accessing the I2C interface control registers. The period from the rising edge of the first clock of the slave address or 1-byte data transmission/reception to the falling edge of an ACK clock is considered to be the transmission/reception period. When the ACKCLK bit is 0 (no ACK clock), the transmission/reception period is from the rising edge of the first clock of the slave address or 1-byte data transmission/reception to the falling edge of the eighth clock.

28.18.2.1 S00 Register

Do not write to the S00 register during transmission/reception.

28.18.2.2 S1D0 Register

Do not change bits other than the IHR bit in the S1D0 register during transmission/reception.

28.18.2.3 S20 Register

Do not change bits other than the ACKBIT bit in the S20 register during transmission/reception.

28.18.2.4 S3D0 Register

- Do not use the bit managing instruction (read-modify-write instruction) to access the S3D0 register. Use the MOV instruction to write to this register.
- Rewrite bits ICK1 and ICK0 when the ES0 bit in the S1D0 register is 0 (I2C interface disabled).

28.18.2.5 S4D0 Register

Rewrite bits ICK4 to ICK2 when the ES0 bit in the S1D0 register is 0 (I2C interface disabled).

28.18.2.6 S10 Register

- Do not use the bit managing instruction (read-modify-write instruction) to access the S10 register. Use the MOV instruction to write to this register.
- Do not write to the S10 register when bits MST and TRX change their values.

Refer to operation examples in 21.3 “Operations” for bits MST and TRX change.

28.18.3 Generating Stop Condition

Technical update No.: TN-16C-A176A/E

In the multi-master I2C-bus interface, when the slave device and/or other master devices drive the SCLMM line low, no normal stop condition is generated. This is because the SDAMM line is released while the SCLMM line is still driven low.

28.18.4 Low/High-level Input Voltage and Low-level Output Voltage

The low-level input voltage, high-level input voltage, and low-level output voltage differ from the I^2C-bus specification.
Refer to the recommended operating conditions for I/O ports which share the pins with SCL and SDA.

I^2C-bus specification
High level input voltage (V_{IH}) = min. 0.7 V_{CC}
Low level input voltage (V_{IL}) = max. 0.3 V_{CC}

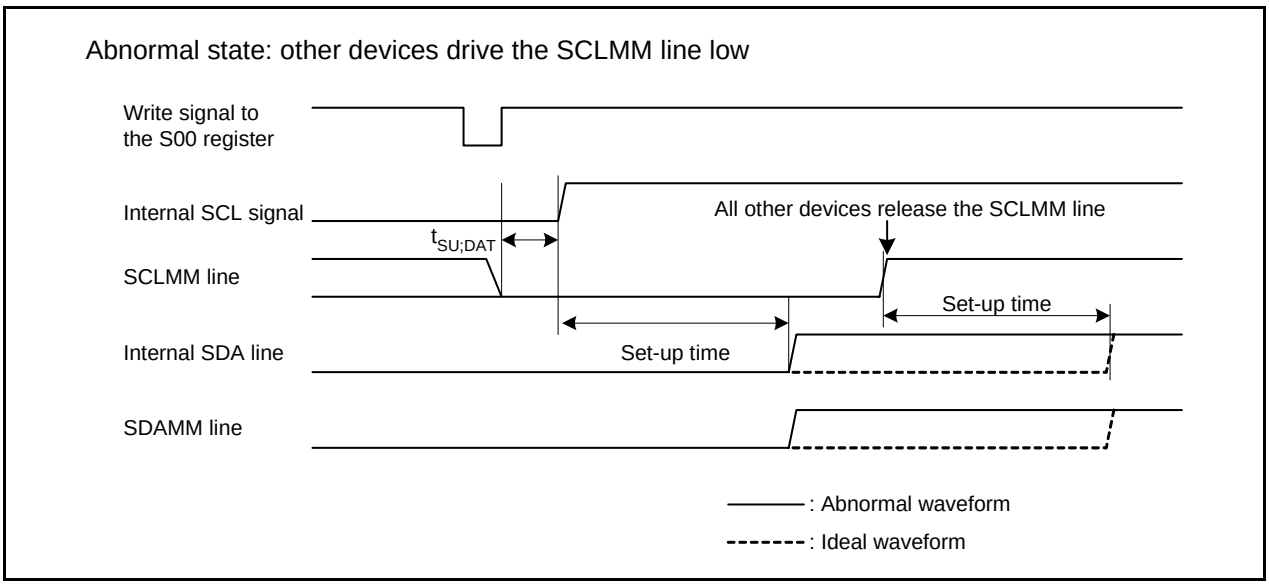


Figure 28.11 Abnormal Waveform

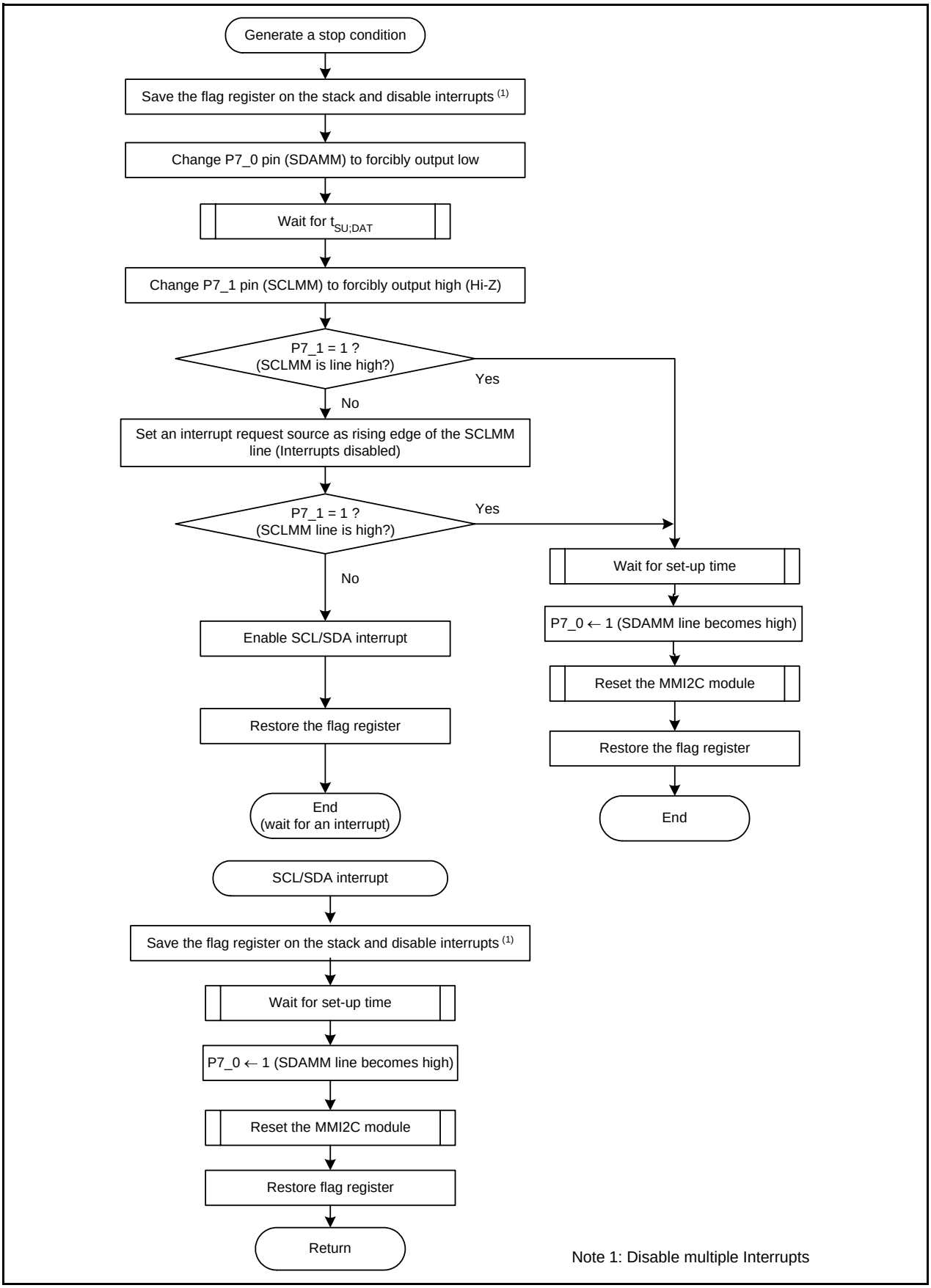


Figure 28.12 Generating a Stop Condition

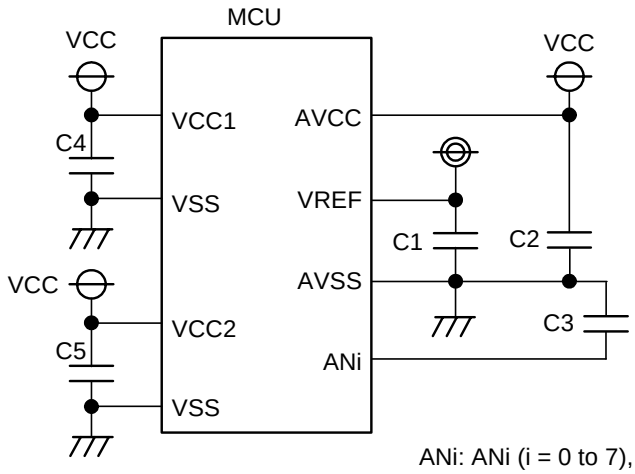
28.19 Notes on A/D Converter

28.19.1 Analog Input Pin

Do not use any of pins AN4 to AN7 as analog input pins if any one of pins $\overline{KI0}$ to $\overline{KI3}$ is used as a key input interrupt. Also, do not use any of four pins AN0 to AN3 as analog input pins if any one of pins $\overline{KI4}$ to $\overline{KI7}$ is used as a key input interrupt.

28.19.2 Pin Configuration

To prevent operation errors due to noise or latchup, and to reduce conversion errors, place capacitors between the AVSS pin and the AVCC pin, the VREF pin, and analog inputs (ANi (i = 0 to 7), ANEXi, AN0_i). Also, place a capacitor between the VCC1 pin and VSS pin.



Notes:
1. C1 ≥ 0.1 μF, C2 ≥ 0.1 μF, C3 ≥ 100 pF, C4 ≥ 0.1 μF, C5 ≥ 0.1 μF (reference values).
2. The traces for the capacitor and MCU should be short and wide as much as physically possible.

Figure 28.13 Example of Pin Configuration

28.19.3 Register Access

Set registers ADCON0 (excluding the ADST bit), ADCON1, and ADCON2 when A/D conversion stops (before a trigger is generated).
After A/D conversion stops, rewrite the ADSTBY bit in the ADCON1 register from 1 to 0.

28.19.4 A/D Conversion Start

When rewriting the ADSTBY bit in the ADCON1 register from 0 (A/D operation stopped) to 1 (A/D operation enabled), wait for one ϕAD cycle or more before starting A/D conversion.

28.19.5 A/D Operation Mode Change

When A/D operation mode has been changed, reselect analog input pins by using bits CH2 to CH0 in the ADCON0 register or bits SCAN1 to SCAN0 in the ADCON1 register.

28.19.6 State When Forcibly Terminated

If A/D conversion in progress is halted by setting the ADST bit in the ADCON0 register to 0, the conversion result is undefined. In addition, the non-converted ADi register may also become undefined. Do not use the ADi register when setting the ADST bit to 0 by a program during A/D conversion.

28.19.7 A/D Open-Circuit Detection Assist Function

The conversion result in open-circuit depends on the external circuit. Use this function only after careful evaluation of the system. When A/D conversion starts after changing the AINRST register, follow these procedures:

(1) Change bits AINRST1 to AINRST0 in the AINRST register.
(2) Wait for one cycle of ϕAD.
(3) Set the ADST bit in the ADCON0 register to 1 (A/D conversion started).

28.19.8 Detecting Completion of A/D Conversion

In one-shot mode and single sweep mode, use the IR bit in the ADIC register to detect completion of A/D conversion. When not using an interrupt, set the IR bit to 0 by a program after the detection. When 1 is written to the ADST bit in the ADCON0 register, the ADST bit becomes 1 (A/D conversion started) after start processing time elapses. (See Table 22.7 “Cycles of A/D Conversion Item”.) When reading the ADST bit shortly after writing 1, 0 (A/D conversion stop) may be read.

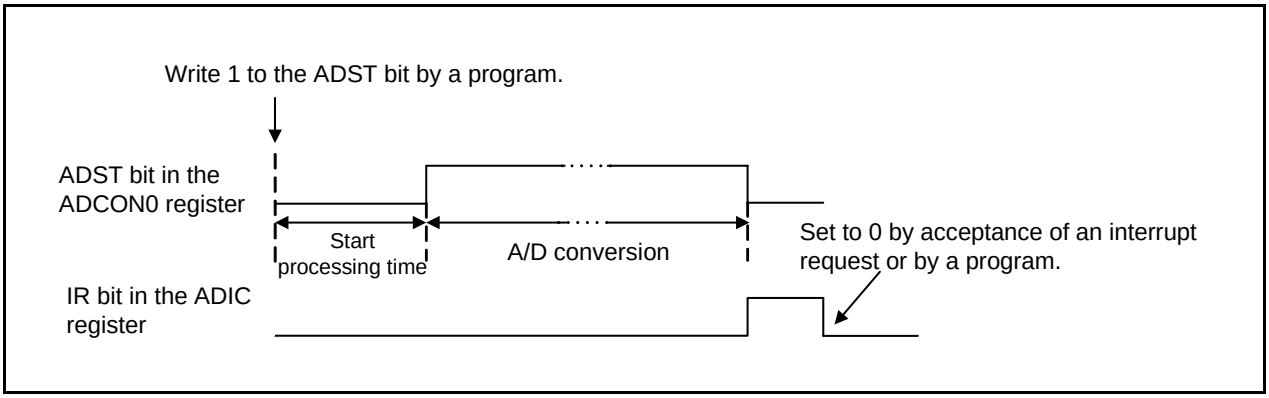


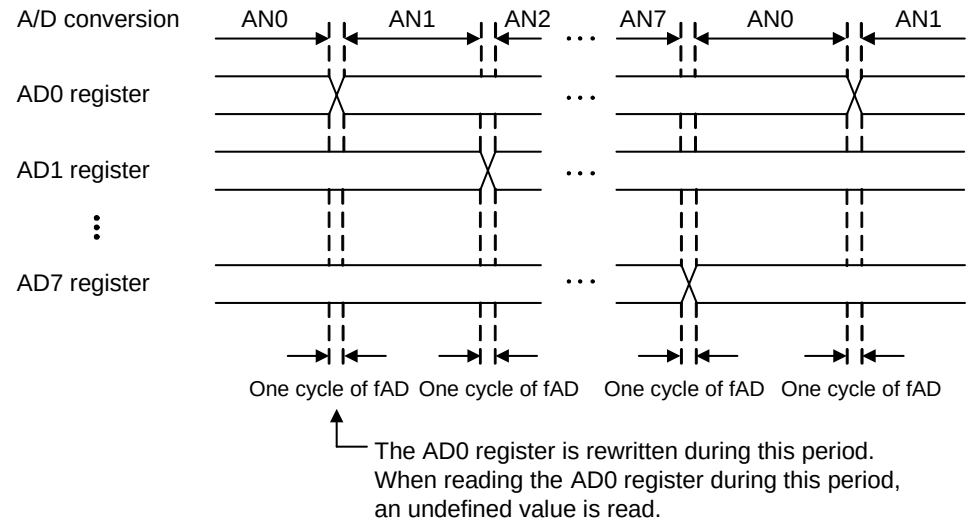
Figure 28.14 ADST Bit Operation

28.19.9 ϕAD

Divide fAD so ϕAD conforms the standard frequency.

28.19.10 Repeat Mode, and Repeat Sweep Mode 0, and Repeat Sweep Mode 1

In repeat mode, and repeat sweep mode 0, and repeat sweep mode 1, when reading the ADi register (i = 0 to 7) during the period when the ADi register value is rewritten, an undefined value may be read. Read the ADi register several times to determine whether the read value is valid. The period for reading an undefined value is one cycle of fAD.



The above diagram assumes the following:

- Bits MD1 to MD0 in the ADCON0 register are 11b, and the MD2 bit in the ADCON1 register is 0 (repeat sweep mode 0).
- Bits ADGSEL1 to ADGSEL0 in the ADCON2 register are 00b (AN0 to AN7 groups selected).
- Bits SCAN1 to SCAN0 in the ADCON1 register are 11b (AN0 to AN7 (8 pins) selected).

Figure 28.15 Period When the ADi Register Value is Rewritten

28.20 Notes on Flash Memory

28.20.1 OFS1 Address and ID Code Storage Address

The OFS1 address and ID code storage address are part of flash memory. When writing a program to flash memory, write an appropriate value to those addresses simultaneously.

In the OFS1 address, the MCU state after reset and the function to prevent rewrite in parallel I/O mode are selected. The OFS1 address is 0FFFFFh. This is the most significant address of block 0 in program ROM 1 and upper address of reset vector. Also, the ID code storage address is in block 0 and upper address of the interrupt vector.

The ID code check function cannot be disabled. Even if the protect using the ID code check function is unnecessary, input the appropriate ID code when using a serial programmer or debugger. Without the appropriate ID code, the serial programmer or debugger cannot be used.

ex) Set FEh to the OFS1 address

When using an address control instruction and logical addition:

```
.org 0FFFFCh
RESET:
.lword start | 0FE000000h
```

When using an address control instruction:

```
.org 0FFFFCh
RESET:
.addr start
.byte 0FEh
```

(Program format varies depending on the compiler. Refer to the compiler manual.)

28.20.2 Reading Data Flash

When 2.7 V ≤ VCC1 ≤ 3.0 V, one wait must be inserted to execute the program on the data flash and read the data. Set the PM17 in the PM1 register or FMR17 bit in the FMR1 register to insert one wait.

28.20.3 CPU Rewrite Mode

28.20.3.1 Operating Speed

Set a CPU clock frequency of 10 MHz or less by the CM06 bit in the CM0 register and bits CM17 and CM16 in the CM1 register before entering CPU rewrite mode (EW0 or EW1 mode). Also, set the PM17 bit in the PM1 register to 1 (wait state).

28.20.3.2 CPU Rewrite Mode Select

Change FMR01 bit in the FMR0 register, FMR11 bit in the FMR1 register, and FMR60 bit in the FMR6 register while in the following state:

- PM24 bit in the PM2 register is 0 ($\overline{\text{NMI}}$ interrupt disabled).
- High is input to the $\overline{\text{NMI}}$ pin.

Change the FMR60 bit while the FMR00 bit in the FMR0 register is 1 (ready).

28.20.3.3 Prohibited Instructions

Do not use the following instructions in EW0 mode:
UND instruction, INTO instruction, JMPS instruction, JSRS instruction, and BRK instruction.

28.20.3.4 Interrupts (EW0 Mode and EW1 Mode)

- Do not use an address match interrupt during command execution because the address match interrupt vector is located in ROM.
- Do not use a non-maskable interrupt during block 0 erase because fixed vector is located in block 0.

28.20.3.5 Rewrite (EW0 Mode)

If the power supply voltage drops while rewriting the block where the rewrite control program is stored, the rewrite control program is not correctly rewritten. This may prevent the flash memory from being rewritten. If this error occurs, use standard serial I/O mode or parallel I/O mode for rewriting.

28.20.3.6 Rewrite (EW1 Mode)

Do not rewrite any blocks in which the rewrite control program is stored.

28.20.3.7 DMA transfer

In EW0 mode, do not use flash memory as a source of the DMA transfer.
In EW1 mode, do not generate a DMA transfer while the FMR00 bit in the FMR0 register is set to 0 (auto programming or auto erasing).

28.20.3.8 Wait Mode

To enter wait mode, set the FMR01 bit in the FMR0 register to 0 (CPU rewrite mode disabled) before executing the WAIT instruction.

28.20.3.9 Stop Mode

To enter stop mode, set the FMR01 bit to 0 (CPU rewrite mode disabled), and then disable DMA transfer before setting the CM10 bit in the CM 1 register to 1 (stop mode).

28.20.3.10 Software Command

Observe the notes below when using the following commands.

- Program
- Block erase
- Lock bit program
- Read lock bit status
- Block blank check

(a) The FMR00 bit in the FMR0 register indicates the status while executing these commands. Do not execute other commands while the FMR00 bit is 0 (busy).
(b) Use these commands in high-speed mode and medium-speed mode. Do not change clock modes while the FMR00 bit in the FMR0 register is 0 (busy).
(c) After executing the program, block erase, or lock bit program command, perform a full status check per one command (i.e. do not perform a single full status check after multiple commands are executed).
(d) Do not execute the program, block erase, lock bit program, or block blank check command when either or both the FMR06 bit in the FMR0 register and the FMR75 bit in the FMR7 register are 1 (completed in error).
(e) Do not use these commands in slow read mode (when the FMR22 bit is 1) or low current consumption read mode (when both bits FMR22 and FMR23 are 1).

28.20.3.11 Area Where the Rewrite Control Program is Executed

The PM10 bit in the PM1 register become 1 in CPU rewrite mode. Execute the rewrite program in internal RAM.

28.20.3.12 Program and Erase Cycles and Execution Time

Execution time of the program, block erase, and lock bit program commands becomes longer as the number of programming and erasing increases.

28.20.3.13 Suspending the Auto-Erase and Auto-Program Operations

When the program, block erase, and lock bit program commands are suspended, the blocks for those commands must be erased. Execute the program and lock bit program commands again after erasing.

Those commands are suspended by the following reset or interrupts:

- Hardware, voltage monitor 0, oscillation stop detect, watchdog timer, software resets.
- NMI, watchdog timer and oscillation stop/restart detect interrupts.

28.20.4 User Boot Mode

28.20.4.1 User Boot Mode Program

Note the following when using user boot mode:

- When using user boot mode, make sure to allocate the program to be executed to program ROM 2.
- Bits VDSEL1 and LVDAS in the OFS1 address are disabled in boot mode.
- When restarting the MCU in user boot mode after starting it in user boot mode, RAM becomes undefined.
- If addresses 13FF8h to 13FFBh are all 00h, the MCU does not enter standard serial I/O mode. Therefore, the programmer or on-chip debugger cannot be connected.
- As the reset sequence differs, the time necessary for starting the program is longer than in single-chip mode.
- Functions in user boot mode cannot be debugged by the on-chip debugging emulator or full spec emulator.
- While using user boot mode, do not change the input level of the pin used for user boot entry. However, if there is a possibility that the input level may change, perform the necessary processes in user boot mode, then restart the MCU in single-chip mode before the input level changes.
- To use user boot mode after standard serial I/O mode, turn off the power when exiting standard serial I/O mode, and then turn on the power again (cold start). The MCU enters user boot mode under the right conditions.

28.20.5 EW1 Mode

(Technical update number: TN-16C-A175A/E)

Adhere to the following when using EW1 mode:

28.20.5.1 Frequency Limitation of EW1 Mode

Set the CPU clock to 1 MHz or higher when using EW1 mode.

28.20.5.2 Frequency Limitation of Block Blank Check Command

Set the CPU clock to 3 MHz or higher when using the block blank check command.

28.20.5.3 Disabling the Lock Bit

Set the FMR02 bit in the FMR0 register to 1 (lock bit disabled).

Do not execute the read lock bit status command or lock bit program command.

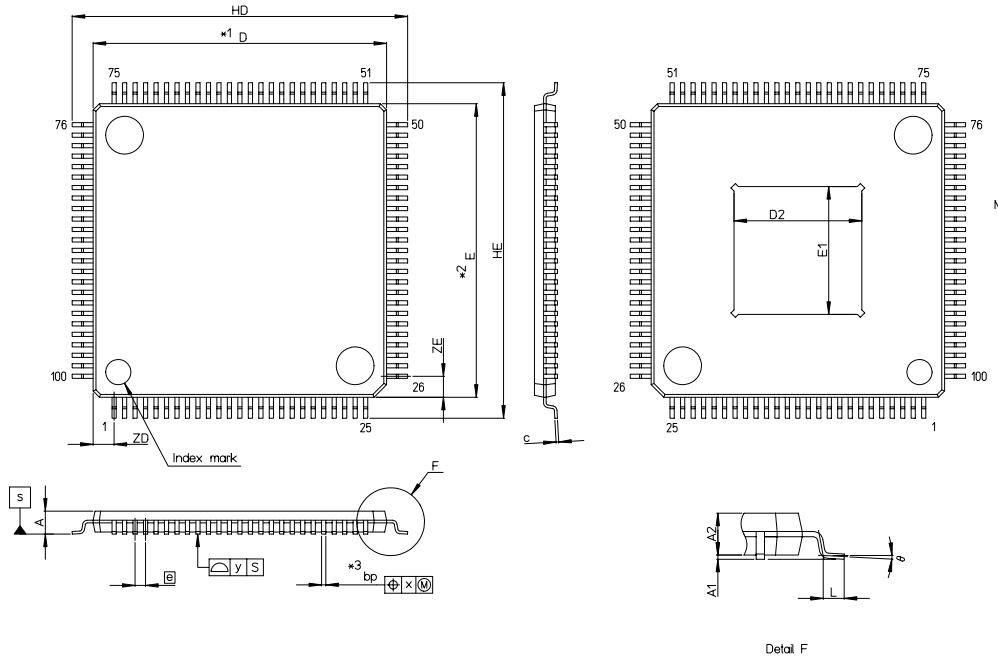
28.20.5.4 Entering EW1 Mode in the User Program Using Wait or Stop Mode

When using EW1 mode in the user program in which the MCU enters wait mode or stop mode, set the FMSTP bit in the FMR0 register to 1 (flash memory off) on RAM. Then, set the FMSTP bit to 0 (flash memory on) again and enter the EW1 mode on flash memory. Execute these processes while an interrupt is disabled.

Appendix 1. Package Dimensions

The information on the latest package dimensions or packaging may be obtained from “Packages” on the Renesas Electronics Website.

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-HTQFP100-14x14-0.50	PTQP0100KE-E	100PFW-E	0.5g



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
A2	—	1.0	—
HD	15.8	16.0	16.2
HE	15.8	16.0	16.2
A	—	—	1.2
A1	0.05	0.1	0.15
bp	0.17	0.22	0.27
c	0.09	0.145	0.20
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.08
ZD	—	1.0	—
ZE	—	1.0	—
L	0.4	0.5	0.6
D2	—	6.1	—
E1	—	6.1	—

Index

REVISION HISTORY	M16C/6S1 Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
0.10	Nov 12, 2010	—	First Edition issued
0.20	Dec 01, 2010	94	Figure 9.2 revised
		107	Table 10.2 revised
		114	Note revised Table 12.2 revised, Note 1 added
		123	Table 12.3 Note 1 deleted
		133	Figure 12.12 revised
		135	Note revised
		169	14.2.1 revised
		196	Table 16.2 Note 1 added
		304	Table 19.2 Note 1 added
		522	25.1, 25.2 revised Figure 25.1 revised
		523	Figure 25.2, Note 2 revised 25.3.1 revised
		524	Figure 25.4 revised 25.3.2 revised
		525	25.3.3 revised Figure 25.5 revised
		256	Figure 25.6 revised
		257	Figure 25.7, Figure 25.8 revised
		528	25.5 revised
		529	Table 26.1 revised
		538	Table 26.15, Note 2 revised
		539	Table 26.16 revised
		540	Table 26.17 revised
0.50	Jun 09, 2011	1	1.1, 1.1.1 revised
		2	Table 1.1 revised
		3	Table 1.3, Table 1.4 revised
		11	Table 1.10 Note 1, Table 1.11 revised
		12	Table 1.12 revised
		18	Table 4.1, Note 1, Note 6 revised
		19	Table 4.2 Note 4 revised
		24	Table 4.7 revised
		34	4.2.1 revised, Table 4.18 added
		38	6.1, Table 6.1, Figure 6.1 revised
		39	Table 6.2 revised
		40	Table 6.4, 6.2.2 revised
		42	6.3.1 VDSEL1 (Vdet0 select bit 1) (b5), LVDAS (Voltage detector 0 start bit) (b6) revised
		43	Table 6.6 revised
		45	"6.4.3 Power-On Reset Function", "Figure 6.5 Power-On Reset Circuit and Operation Example" deleted

REVISION HISTORY	M16C/6S1 Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
0.50	Jun 09, 2011	47	6.4.7 revised
		48	"6.5.2 Power-On Reset" deleted, 6.5.3 added
		49	6.5.4 to 6.5.4.4 added
		50	Table 7.1 revised
		51	Table 7.2 Note 2, 7.2.1 Note 2 revised
		52	7.2.2, Note 2 revised
		53	7.3.1 revised
		54	7.4.1 Digital Filter deleted, Figure 7.2 revised
		55	7.4.1.1, Table 7.3, Figure 7.3 revised
		56	Table 8.1 revised
		58	Table 8.3 revised
		68	8.2.6 PLC06 (PLLFCK generation enable bit) (b6), Table 8.5 revised
		69	8.2.7 revised
		71	8.2.10 revised
		73	Table 8.6, Table 8.7 revised
		77	Figure 8.5 revised
		87	9.2.1 revised
		88	9.2.2 revised
		90	9.3.1.6 revised
		105	9.6.1, 9.6.2 revised
		106	9.6.3 revised
		132	12.4.4 revised
		133	Figure 12.12 revised
		169	14.2.1 revised
		302	18.5.1 added
		304	Table 19.2, Note 1 revised
		368	19.5 "UARTi (i = 0 to 2, 5 to 7)" → "UARTi (i = 0 to 2, 5, 6)", 19.5.2.2 revised
		369	19.5.2.3 "TXDi pin (i = 0 to 2, 5 to 7)" → "TXDi pin (i = 0 to 2, 5, 6)", 19.5.3.2 "UiC0 register (i = 0 to 2, 5 to 7)" → "UiC0 register (i = 0 to 2, 5, 6)" revised
		370, 371	19.5.4, 19.5.4.1 revised, 19.5.4.3 to 19.5.4.7, Figure 19.31 added
		386	20.5.3 added
		437	21.5.3 added
		438	Figure 21.22 added
		439	Figure 21.23 added
		472	22.7.10, Figure 22.19 added
		479	Table 24.2 revised
		480	24.2 revised
		481	Table 24.4, 24.3.1 revised
		482	24.3.1 FMR02 (Lock bit disable select bit) (b2), FMSTP (Flash memory stop bit) (b3) revised
		483	FMR07 (Erase status flag) (b7) deleted
		484	24.3.2 revised

REVISION HISTORY	M16C/6S1 Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
0.50	Jun 09, 2011	488	24.3.6 added
		491	24.6 revised
		492	24.7 added
		494	Figure 24.4 added
		495	Table 29.10 revised
		498	24.8.4, Table 29.13 revised
		499	24.8.5 added
		500	24.8.5.3 revised
		501	24.8.5.4 revised
		502	24.8.5.5, Figure 24.10 added
		503	24.8.5.6 revised
		505	Figure 24.13 revised
		506	24.8.5.9, Figure 24.14 revised
		507	24.8.6, Table 24.15, Table 24.16 revised
		508	24.8.6.1, Table 24.17, Figure 24.15 revised
		509	24.8.6.2 revised
		510	24.8.7 revised
		512	Figure 24.17 revised
		513	Figure 24.18 revised
		514	Figure 24.19 revised
		517	24.8.8 revised, Table 24.19 added
		518	Figure 24.22 revised
		519	Figure 24.23 revised
		529	24.11.1, 24.11.3.2 revised
		530	24.11.3.7, 24.11.3.10 revised
		531	24.11.3.13 revised
		532	24.11.4.1 revised, "24.11.4.2 Entering User Boot Mode After Standard Serial I/O Mode" deleted, 24.11.5 to 24.11.5.4 added
		538	Figure 25.8 revised
		542	Table 26.4 revised
		543	Figure 26.2 revised
		546	"Table 26.12 Power-On Reset Circuit", "Figure 26.3 Power-On Reset Circuit Electrical Characteristics" deleted
		549	Table 26.14 revised
		550	Table 26.15 revised
		551	Table 26.16, Note 1 revised, Note 2, Note 3 added
		556	26.9.6, Table 26.28, Figure 26.10 added
		557	27.1 revised
		559	27.3.1 revised, Table 27.2 added
		561	27.5.2 Power On Reset deleted, 27.5.3 added
		562	27.5.4 to 27.5.4.4 added
		567	27.7.1, 27.7.2 revised

REVISION HISTORY	M16C/6S1 Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
0.50	Jun 09, 2011	568	27.7.3 revised
		580	27.15.1 added
		582	27.16.2.2 revised
		584, 585	27.16.4 revised, 27.16.4.3 to 27.16.4.7, Figure 27.10 added
		586	27.17.3 added
		587	27.18.3 added
		588	Figure 27.11 added
		589	Figure 27.12 added
		592	27.19.10, Figure 27.15 added
		593	27.20.1, 27.20.3.2 revised
		594	27.20.3.7, 27.20.3.10 revised
		595	27.20.3.13 revised
1.00	Mar 30, 2012	All pages	"Preliminary" and "Under development" deleted, "001Ah" and "002Ah" revised
		1	1.1.1 revised
		3	Table 1.2 "Current Consumption" and "Package" revised
		4, 6	Table 1.5, Figures 1.1, and 1.3 package type revised
		12	Table 1.12 title revised
		17	Figure 3.1 Note 3 added
		18	Table 4.1 Notes 4 revised, Notes 2 and 6 deleted
		19	Table 4.2 Note 2 revised, Note 4 deleted
		28	Table 4.11 "033Dh" revised
		38	Table 6.1 and Figure 6.1 revised
		39	Table 6.2 added
		40	Table 6.4 Note 1 revised
		41	6.2.2 CWR and OSDR description revised
		44	Table 6.7 Note 1 revised
		45	Figure 6.3 revised
		48	6.4.7 description and Figure 6.5 revised
		51	Table 7.1" Voltage to detect" added, Figure 7.1 revised
		52	7.2 description added, Table 7.2 Note 1 deleted, 7.2.1 revised
		58	Figure 8.1 revised
		63	8.2.3 CM10 description revised
		72	8.2.10 PM21 description revised
		74	8.3.2 description revised
		76	8.3.4 (3) and (4) revised
		77	8.4.1 description revised
		92	Table 9.2 Note 2 revised, Notes 3 to 6 deleted
		97	9.3.3 and 9.3.3.2 description revised
		98	9.3.3.4 description revised
		99	9.3.4.1 description revised
		100	9.3.4.3 description revised
		101	Figure 9.3, Note 4 deleted

REVISION HISTORY	M16C/6S1 Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.00	Mar 30, 2012	102	Figure 9.4 revised
		103	Figure 9.5 revised
		106, 576	9.6.4 and 28.7.4 revised, 9.6.5 and 28.7.5 added
		126	12.3.3 PU21 description revised
		128	12.3.5 "After Reset" revised
		139	13.2.2 revised
		140	13.2.3 revised
		165, 167, 578, 580	13.13.2 description, 13.13.5 "Example 1", 13.13.6 description, 28.10.2 description, 28.10.5 "Example 1", 28.10.6 description revised
		170	14.2.2 revised
		171	14.2.3 and 14.2.4 "After Reset" revised
		173	14.3.1 description revised
		182	15.2.3 "After Reset" revised
		188	15.3.3 description revised
		198	Figure 16.2 revised
		199	Figure 16.3 revised
		207, 208	16.2.8 and 16.2.9 "After Reset" revised
		221, 262	Tables 16.9 and 17.8 "PCLKR" and "TACS0 to TACS2" revised
		231, 235	16.3.5 MR1 description revised
		243, 583	16.5.1.2, 16.5.1.3, 28.3.1.2 and 28.13.1.3 added
		252, 253	17.2.4 and 17.2.5 revised
		263	17.3.3 TCK1 description added
		265	Table 17.9 revised, Note 3 added
		271, 586	17.5.3.2 and 28.14.3.2 added
		272, 587	17.5.4.3 and 28.14.4.3 added
		277	18.2.2 SC03 to SC00 and SC12 to SC10 description revised
		285	18.2.9 RUN description revised
		287	18.2.11MINUS and PLUS description revised
		288	18.2.12 ADJ30S added, RSTADJ revised
		296	Figure 18.4 revised
		298	18.3.3 description revised
		299	18.3.4.1 description revised
		300	18.3.4.2 description revised
		313	19.2.6 → 19.2.3, SMD2 to SMD0 description added
		314	19.2.5 → 19.2.4, "Setting Range" revised, 19.2.3 → 19.2.5, description revised
		315, 317, 318	19.2.7 → 19.2.6, 19.2.8 → 19.2.7, 19.2.4 → 19.2.8
		321, 322	19.2.13 → 19.2.10, title, b6 and b7, STAREQ, RSTAREQ, STPREQ description revised, STSPSEL, ACKD, ACKC, SWC9, and SCLHI added
		323	19.2.12 → 19.2.11
		324	19.2.11 → 19.2.12, and b0 to b6 "Function" revised
		325	19.2.10 → 19.2.13

Rev.	Date	Description	
		Page	Summary
1.00	Mar 30, 2012	326	Table 19.5 Note 1 revised
		330	"19.3.1.1 Transmit/Receive Circuit Initialization" deleted
		333	19.3.1.8 added
		338	Figure 19.13 revised
		340	"19.3.2.2 Transmit/Receive Circuit Initialization" deleted
		344	Table 19.14 and Note 1 revised
		345	Figure 19.18 revised, Figure 19.19 added
		346	Table 19.16 "UiTB" revised
		347	Table 19.17 "SWC" and "CKPH" revised
		348	Table 19.18 revised, Note 3 added
		349	Figure 19.20 revised
		350	19.3.3.1 description, Figure 19.21, and 19.3.3.2 title revised
		351	Figure 19.22 revised
		352	Figure 19.23 added
		353 to 355	19.3.3.3 description revised, 19.3.3.4 added
		356, 357	Figure 19.26, 19.3.3.6, 19.3.3.7 added
		357, 358	19.3.3.8 description revised, Figures 19.30 and 19.31 added
		358	19.3.3.9 and 19.3.3.10 description revised
		359	Table 19.20 Note 1 deleted
		360	Table 19.21 revised
		361	Table 19.22 "CKDIR" revised
		362	Figures 19.24 and 19.25 deleted
		365	Table 19.24 Note 2 revised
		367	Figure 19.35 (1) revised
		371	19.4.1 description revised
		374	19.5.2.3 description revised, "19.5.3 UART (Clock Asynchronous Serial I/O) Mode" deleted
		391, 402, 403, 417, 418, 421, 422, 428, 429	"High-speed clock mode" → "Fast-mode"
		392	Table 21.2 "Arbitration lost" revised
		399	21.2.5 BC2 to BC0 description revised
		402	21.2.6 CCR4 to CCR0 description revised
		403	Table 21.5 revised
		410	Table 21.9 "Rewrite this bit when the TOE bit is 0." added
		412 to 414	21.2.10 LRB, AAS, PIN description revised
		416	21.2.11 AAS0 to AAS2 description revised
		418	Table 21.11 Notes 1 and 2 added
		427	21.3.6 description revised
		444	Figure 22.1 revised
		448	22.2.3 b4 and b5 Function revised

Rev.	Date	Description	
		Page	Summary
1.00	Mar 30, 2012	457	22.3.6 description revised
		458, 459	Figures 22.6 to 22.9 revised
		461, 463, 465, 467, 469, 470	Figures 22.10 to 22.15 revised
		490	24.3.5 FMR60 description revised
		492	Figure 24.2 revised
		494	24.7.1 description revised
		496	Tables 24.9 and 24.10 revised
		497	Figure 24.4 revised
		498	24.5 description, Table 24.11 revised
		501, 502	Tables 24.14 and 24.15 Note 1 added
		514	24.8.7 description revised
		520	24.8.8 description revised
		528	24.9.2 and 24.9.3 description revised
		529	Table 24.23 "VREF" revised
		531	24.9.5 description, Table 24.25 "VREF" revised
		532	Figure 24.28, 24.10.1 description revised
		537 to 539	Chapter "25. PLC Modem Core" added
		540	Figure 26.1 revised
		544	Figure 26.6 revised
		548	Table 27.1 revised
		550	Table 27.5 revised
		552	Table 27.9 revised
		557	Table 27.14 and Note 1 added
		605	Package revised
1.10	Oct 31, 2013	3	Tables 1.2, 1.3, and 1.4 revised
		39	Table 6.2 "SFR(B)" register and bit deleted
		57	Table 8.1 Note 2 added
		76	(1) revised
		83	Figure 8.7 revised
		313	19.2.3 Bit description revised
		314	19.2.4, 19.2.5 After Reset revised
		318	19.2.8 After Reset revised
		345	Table 19.15 Note 1 added
		441	21.5.4 added
		478	23.2.3 description added
		480	Figure 23.2 revised
		481	Figure 23.3 revised
		485	24.3.1 Bit description revised
		532	Figure 24.28 revised
		542	Figure 26.3 revised

REVISION HISTORY	M16C/6S1 Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.10	Oct 31, 2013	544	Figure 26.6 revised
		545	Figures 26.7 and 26.8 revised
		550	Table 27.4 and Note 1 revised
		571	Figure 28.3 revised

M16C/6S1 Group User's Manual: Hardware

Publication Date: Rev.0.20 Dec 1, 2010
Rev.1.10 Oct 31, 2013

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: +44-1628-651-700, Fax: +44-1628-651-804

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.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 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 1.3

M16C/6S1 Group

Renesas Electronics Corporation

R01UH0105EJ0110