

# 16

## M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual

RENESAS 16-BIT SINGLE-CHIP MICROCOMPUTER  
M16C FAMILY / M16C/60 SERIES

Hardware Manual

Before using this material, please visit our website to verify that this is the most updated document available.

Rev.2.41  
Revision Date: Jan 10, 2006

Renesas Technology  
[www.renesas.com](http://www.renesas.com)

### Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of non-flammable material or (iii) prevention against any malfunction or mishap.

### Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein. The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors. Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (<http://www.renesas.com>).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination. Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.

# How to Use This Manual

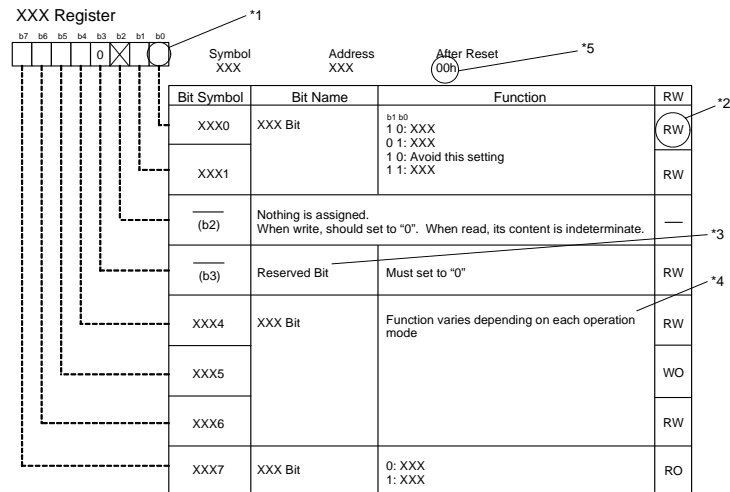
## 1. Introduction

This hardware manual provides detailed information on the M16C/62P Group (M16C/62P, M16C/62PT) of microcomputers.

Users are expected to have basic knowledge of electric circuits, logical circuits and microcomputers.

## 2. Register Diagram

The symbols, and descriptions, used for bit function in each register are shown below.



\*1

Blank: Set to "0" or "1" according to the application

0: Set to "0"

1: Set to "1"

X: Nothing is assigned

\*2

RW: Read and write

RO: Read only

WO: Write only

—: Nothing is assigned

\*3

•Reserved bit

Reserved bit. Set to specified value.

\*4

•Nothing is assigned

Nothing is assigned to the bit concerned. As the bit may be use for future functions, set to "0" when writing to this bit.

•Do not set to this value

The operation is not guaranteed when a value is set.

•Function varies depending on mode of operation

Bit function varies depending on peripheral function mode.

Refer to respective register for each mode.

### 3. M16C Family Documents

The following documents were prepared for the M16C family. <sup>(1)</sup>

Document	Contents
Short Sheet	Hardware overview
Data Sheet	Hardware overview and electrical characteristics
Hardware Manual	Hardware specifications (pin assignments, memory maps, peripheral specifications, electrical characteristics, timing charts)
Software Manual	Detailed description of assembly instructions and microcomputer performance of each instruction
Application Note	<ul style="list-style-type: none"><li>• Application examples of peripheral functions</li><li>• Sample programs</li><li>• Introduction to the basic functions in the M16C family</li><li>• Programming method with Assembly and C languages</li></ul>
RENESAS TECHNICAL UPDATE	Preliminary report about the specification of a product, a document, etc.

#### NOTES:

1. Before using this material, please visit the our website to confirm that this is the most current document available.

# Table of Contents

SFR Page Reference	B - 1
1. Overview	1
1.1 Applications .....	1
1.2 Performance Outline .....	2
1.3 Block Diagram .....	5
1.4 Product List .....	7
1.5 Pin Configuration .....	14
1.6 Pin Description .....	25
2. Central Processing Unit (CPU)	30
2.1 Data Registers (R0, R1, R2 and R3) .....	30
2.2 Address Registers (A0 and A1) .....	31
2.3 Frame Base Register (FB) .....	31
2.4 Interrupt Table Register (INTB) .....	31
2.5 Program Counter (PC) .....	31
2.6 User Stack Pointer (USP) and Interrupt Stack Pointer (ISP) .....	31
2.7 Static Base Register (SB) .....	31
2.8 Flag Register (FLG) .....	31
2.8.1 Carry Flag (C Flag) .....	31
2.8.2 Debug Flag (D Flag) .....	31
2.8.3 Zero Flag (Z Flag) .....	31
2.8.4 Sign Flag (S Flag) .....	31
2.8.5 Register Bank Select Flag (B Flag) .....	31
2.8.6 Overflow Flag (O Flag) .....	31
2.8.7 Interrupt Enable Flag (I Flag) .....	31
2.8.8 Stack Pointer Select Flag (U Flag) .....	32
2.8.9 Processor Interrupt Priority Level (IPL) .....	32
2.8.10 Reserved Area .....	32
3. Memory	33
4. Special Function Register (SFR)	34
5. Reset	40
5.1 Hardware Reset 1 .....	40
5.1.1 Reset on a Stable Supply Voltage .....	40
5.1.2 Power-on Reset .....	40

5.2	Brown-out Detection Reset (Hardware Reset 2) .....	42
5.3	Software Reset.....	43
5.4	Watchdog Timer Reset.....	43
5.5	Oscillation Stop Detection Reset.....	43
5.6	Internal Space .....	44
6.	Voltage Detection Circuit .....	45
6.1	Low Voltage Detection Interrupt .....	49
6.2	Limitations on Exiting Stop Mode .....	51
6.3	Limitations on Exiting Wait Mode .....	51
6.4	Cold Start-up / Warm Start-up Determine Function .....	52
7.	Processor Mode .....	54
7.1	Types of Processor Mode .....	54
7.2	Setting Processor Modes .....	55
8.	Bus .....	59
8.1	Bus Mode .....	59
8.1.1	Separate Bus.....	59
8.1.2	Multiplexed Bus .....	59
8.2	Bus Control.....	60
8.2.1	Address Bus .....	60
8.2.2	Data Bus.....	60
8.2.3	Chip Select Signal .....	60
8.2.4	Read and Write Signals.....	63
8.2.5	ALE Signal.....	63
8.2.6	RDY Signal .....	64
8.2.7	HOLD Signal.....	65
8.2.8	8.2.8 BCLK Output .....	65
8.2.9	External Bus Status When Internal Area Accessed .....	67
8.2.10	Software Wait .....	68
9.	Memory Space Expansion Function .....	72
9.1	1-Mbyte Mode .....	72
9.2	4-Mbyte Mode .....	72
9.2.1	9.2.1 Addresses 04000h to 3FFFFh, C0000h to FFFFFh .....	72
9.2.2	9.2.2 Addresses 40000h to BFFFFh .....	72

10. Clock Generation Circuit	82
10.1 Types of the Clock Generation Circuit.....	82
10.1.1 Main Clock.....	89
10.1.2 Sub Clock .....	90
10.1.3 On-chip Oscillator Clock .....	91
10.1.4 PLL Clock .....	91
10.2 CPU Clock and Peripheral Function Clock.....	93
10.2.1 CPU Clock and BCLK.....	93
10.2.2 Peripheral Function Clock (f1, f2, f8, f32, f1SIO, f2SIO, f8SIO, f32SIO, fAD, fC32).....	93
10.3 Clock Output Function.....	93
10.4 Power Control.....	94
10.4.1 Normal Operating Mode .....	94
10.4.2 Wait Mode .....	96
10.4.3 Stop Mode .....	98
10.5 System Clock Protection Function .....	102
10.6 Oscillation Stop and Re-oscillation Detect Function.....	103
10.6.1 Operation When CM27 bit = 0 (Oscillation Stop Detection Reset) ...	103
10.6.2 Operation When CM27 bit = 0 (Oscillation Stop and Re-oscillation Detect Interrupt).....	103
10.6.3 How to Use Oscillation Stop and Re-oscillation Detect Function .....	104
11. Protection	105
12. Interrupt	106
12.1 Type of Interrupts .....	106
12.2 Software Interrupts .....	107
12.2.1 Undefined Instruction Interrupt .....	107
12.2.2 Overflow Interrupt.....	107
12.2.3 BRK Interrupt.....	107
12.2.4 INT Instruction Interrupt.....	107
12.3 Hardware Interrupts.....	108
12.3.1 Special Interrupts.....	108
12.3.2 Peripheral Function Interrupts .....	108
12.4 Interrupts and Interrupt Vector .....	109
12.4.1 Fixed Vector Tables.....	109
12.4.2 Relocatable Vector Tables .....	110

12.5	Interrupt Control .....	111
12.5.1	I Flag.....	113
12.5.2	IR Bit.....	113
12.5.3	ILVL2 to ILVL0 Bits and IPL .....	113
12.5.4	Interrupt Sequence .....	114
12.5.5	Interrupt Response Time .....	115
12.5.6	Variation of IPL when Interrupt Request is Accepted.....	115
12.5.7	Saving Registers .....	116
12.5.8	Returning from an Interrupt Routine .....	118
12.5.9	Interrupt Priority .....	118
12.5.10	Interrupt Priority Level Select Circuit .....	119
12.6	$\overline{\text{INT}}$ Interrupt .....	120
12.7	$\overline{\text{NMI}}$ Interrupt .....	121
12.8	Key Input Interrupt.....	121
12.9	Address Match Interrupt.....	122
13.	Watchdog Timer .....	124
13.1	Count source protective mode .....	125
14.	DMAC .....	126
14.1	Transfer Cycles .....	132
14.1.1	Effect of Source and Destination Addresses .....	132
14.1.2	Effect of BYTE Pin Level .....	132
14.1.3	Effect of Software Wait.....	132
14.1.4	Effect of RDY Signal.....	132
14.2	DMA Transfer Cycles .....	134
14.3	DMA Enable .....	135
14.4	DMA Request.....	135
14.5	Channel Priority and DMA Transfer Timing.....	136
15.	Timers .....	137
15.1	Timer A.....	139
15.1.1	Timer Mode .....	144
15.1.2	Event Counter Mode.....	146
15.1.3	One-shot Timer Mode.....	151
15.1.4	Pulse Width Modulation (PWM) Mode.....	153



15.2	Timer B.....	156
15.2.1	Timer Mode .....	159
15.2.2	Event Counter Mode.....	160
15.2.3	Pulse Period and Pulse Width Measurement Mode .....	162
16.	Three-Phase Motor Control Timer Function .....	165
17.	Serial Interface .....	176
17.1	UARTi (i=0 to 2) .....	176
17.1.1	Clock Synchronous Serial I/O Mode.....	189
17.1.2	Clock Asynchronous Serial I/O (UART) Mode.....	197
17.1.3	Special Mode 1 (I <sup>2</sup> C mode) .....	205
17.1.4	Special Mode 2.....	215
17.1.5	Special Mode 3 (IE mode) .....	220
17.1.6	Special Mode 4 (SIM Mode) (UART2).....	222
17.2	SI/O3 and SI/O4.....	227
17.2.1	SI/Oi Operation Timing.....	231
17.2.2	CLK Polarity Selection.....	231
17.2.3	Functions for Setting an SOUTi Initial Value .....	232
18.	A/D Converter .....	233
18.1	Mode Description .....	238
18.1.1	One-Shot Mode .....	238
18.1.2	Repeat Mode .....	240
18.1.3	Single Sweep Mode.....	242
18.1.4	Repeat Sweep Mode 0 .....	244
18.1.5	Repeat Sweep Mode 1 .....	246
18.2	Function.....	248
18.2.1	Resolution Select Function.....	248
18.2.2	Sample and Hold .....	248
18.2.3	Extended Analog Input Pins .....	248
18.2.4	18.2.4 External Operation Amplifier (Op-Amp) Connection Mode....	248
18.2.5	18.2.5 Current Consumption Reducing Function .....	249
18.2.6	Output Impedance of Sensor under A/D Conversion .....	249

19. D/A Converter	251
20. CRC Calculation	253
21. Programmable I/O Ports	255
21.1 Port Pi Direction Register (PD <sub>i</sub> Register, i = 0 to 13)	256
21.2 Port Pi Register (Pi Register, i = 0 to 13)	256
21.3 Pull-up Control Register 0 to Pull-up Control Register 3 (PUR0 to PUR3 Registers)	256
21.4 Port Control Register (PCR Register)	256
22. Flash Memory Version	270
22.1 Memory Map	272
22.1.1 Boot Mode	273
22.2 Functions To Prevent Flash Memory from Rewriting	273
22.2.1 ROM Code Protect Function	273
22.2.2 ID Code Check Function	273
22.3 CPU Rewrite Mode	275
22.3.1 EW0 Mode	276
22.3.2 EW1 Mode	276
22.3.3 Flash memory Control Register (FIDR, FMR0 and FMR1 registers)	276
22.3.4 Precautions on CPU Rewrite Mode	284
22.3.5 Software Commands	286
22.3.6 Data Protect Function	291
22.3.7 Status Register	291
22.3.8 Full Status Check	293
22.4 Standard Serial I/O Mode	295
22.4.1 ID Code Check Function	295
22.4.2 Example of Circuit Application in the Standard Serial I/O Mode	301
22.5 Parallel I/O Mode	303
22.5.1 User ROM and Boot ROM Areas	303
22.5.2 ROM Code Protect Function	303
23. Electrical Characteristics	304
23.1 Electrical Characteristics (M16C/62P)	304
23.2 Electrical Characteristics (M16C/62PT)	346

24. Precautions	359
24.1 SFR .....	359
24.1.1 Register Settings .....	359
24.2 Reset .....	360
24.3 Bus .....	361
24.4 PLL Frequency Synthesizer .....	362
24.5 Power Control.....	363
24.6 Protect.....	365
24.7 Interrupt.....	366
24.7.1 Reading address 00000h .....	366
24.7.2 Setting the SP.....	366
24.7.3 The $\overline{\text{NMI}}$ Interrupt .....	366
24.7.4 Changing the Interrupt Generate Factor.....	367
24.7.5 $\overline{\text{INT}}$ Interrupt .....	367
24.7.6 Rewrite the Interrupt Control Register .....	368
24.7.7 Watchdog Timer Interrupt.....	368
24.8 DMAC.....	369
24.8.1 Write to DMAE Bit in DMiCON Register .....	369
24.9 Timers .....	370
24.9.1 Timer A.....	370
24.9.2 Timer B .....	372
24.10 Serial interface .....	373
24.10.1 Clock Synchronous Serial I/O.....	373
24.10.2 UART .....	374
24.10.3 SI/O3, SI/O4 .....	374
24.11 A/D Converter.....	375
24.12 Programmable I/O Ports.....	377
24.13 Electric Characteristic Differences Between Mask ROM and Flash Memory Version Microcomputers.....	378
24.14 Mask ROM .....	378
24.15 Flash Memory Version .....	379
24.15.1 Functions to Inhibit Rewriting Flash Memory Rewrite .....	379
24.15.2 Stop mode .....	379
24.15.3 Wait mode .....	379
24.15.4 Low power dissipation mode, on-chip oscillator low power dissipation mode ....	379
24.15.5 Writing command and data.....	379

24.15.6 Program Command .....	379
24.15.7 Lock Bit Program Command .....	379
24.15.8 Operation speed .....	380
24.15.9 Instructions inhibited against use .....	380
24.15.10 Interrupts .....	380
24.15.11 How to access .....	380
24.15.12 Writing in the user ROM area .....	380
24.15.13 DMA transfer .....	381
24.15.14 Regarding Programming/Erasing Endurance and Execution Time ..	381
24.16 Noise .....	382
25. Differences Depending on Manufacturing Period	383
Appendix 1. Package Dimensions	385
Appendix 2. Difference between M16C/62P and M16C/30P	387
Register Index	390

## SFR Page Reference

Address	Register	Symbol	Page
0000h			
0001h			
0002h			
0003h			
0004h	Processor Mode Register 0	PM0	56
0005h	Processor Mode Register 1	PM1	57
0006h	System Clock Control Register 0	CM0	84
0007h	System Clock Control Register 1	CM1	85
0008h	Chip Select Control Register	CSR	61
0009h	Address Match Interrupt Enable Register	AIER	123
000Ah	Protect Register	PRCR	105
000Bh	Data Bank Register	DBR	73
000Ch	Oscillation Stop Detection Register	CM2	86
000Dh			
000Eh	Watchdog Timer Start Register	WDTs	125
000Fh	Watchdog Timer Control Register	WDC	53, 125
0010h	Address Match Interrupt Register 0	RMAD0	123
0011h			
0012h			
0013h			
0014h	Address Match Interrupt Register 1	RMAD1	123
0015h			
0016h			
0017h			
0018h			
0019h	Voltage Detection Register 1	VCR1	46
001Ah	Voltage Detection Register 2	VCR2	46
001Bh	Chip Select Expansion Control Register	CSE	68
001Ch	PLL Control Register 0	PLC0	88
001Dh			
001Eh	Processor Mode Register 2	PM2	87
001Fh	Low Voltage Detection Interrupt Register	D4INT	47
0020h	DMA0 Source Pointer	SAR0	131
0021h			
0022h			
0023h			
0024h	DMA0 Destination Pointer	DAR0	131
0025h			
0026h			
0027h			
0028h	DMA0 Transfer Counter	TCR0	131
0029h			
002Ah			
002Bh			
002Ch	DMA0 Control Register	DM0CON	130
002Dh			
002Eh			
002Fh			
0030h	DMA1 Source Pointer	SAR1	131
0031h			
0032h			
0033h			
0034h	DMA1 Destination Pointer	DAR1	131
0035h			
0036h			
0037h			
0038h	DMA1 Transfer Counter	TCR1	131
0039h			
003Ah			
003Bh			
003Ch	DMA1 Control Register	DM1CON	130
003Dh			
003Eh			
003Fh			

### NOTES:

- Blank columns are all reserved space. No access is allowed.

Address	Register	Symbol	Page
0040h			
0041h			
0042h			
0043h			
0044h	INT3 Interrupt Control Register	INT3IC	112
0045h	Timer B5 Interrupt Control Register	TB5IC	111
0046h	Timer B4 Interrupt Control Register, UART1 BUS Collision Detection Interrupt Control Register	TB4IC, U1BCNIC	111
0047h	Timer B3 Interrupt Control Register, UART0 BUS Collision Detection Interrupt Control Register	TB3IC, U0BCNIC	111
0048h	SI/O4 Interrupt Control Register, INT5 Interrupt Control Register	S4IC, INT5IC	112
0049h	SI/O3 Interrupt Control Register, IINT4 Interrupt Control Register	S3IC, INT4IC	112
004Ah	UART2 Bus Collision Detection Interrupt Control Register	BCNIC	111
004Bh	DMA0 Interrupt Control Register	DM0IC	111
004Ch	DMA1 Interrupt Control Register	DM1IC	111
004Dh	Key Input Interrupt Control Register	KUPIC	111
004Eh	A/D Conversion Interrupt Control Register	ADIC	111
004Fh	UART2 Transmit Interrupt Control Register	S2TIC	111
0050h	UART2 Receive Interrupt Control Register	S2RIC	111
0051h	UART0 Transmit Interrupt Control Register	S0TIC	111
0052h	UART0 Receive Interrupt Control Register	S0RIC	111
0053h	UART1 Transmit Interrupt Control Register	S1TIC	111
0054h	UART1 Receive Interrupt Control Register	S1RIC	111
0055h	Timer A0 Interrupt Control Register	TA0IC	111
0056h	Timer A1 Interrupt Control Register	TA1IC	111
0057h	Timer A2 Interrupt Control Register	TA2IC	111
0058h	Timer A3 Interrupt Control Register	TA3IC	111
0059h	Timer A4 Interrupt Control Register	TA4IC	111
005Ah	Timer B0 Interrupt Control Register	TB0IC	111
005Bh	Timer B1 Interrupt Control Register	TB1IC	111
005Ch	Timer B2 Interrupt Control Register	TB2IC	111
005Dh	INT0 Interrupt Control Register	INT0IC	112
005Eh	INT1 Interrupt Control Register	INT1IC	112
005Fh	INT2 Interrupt Control Register	INT2IC	112
0060h			
0061h			
0062h			
0063h			
0064h			
0065h			
0066h			
0067h			
0068h			
0069h			
006Ah			
006Bh			
006Ch			
006Dh			
006Eh			
006Fh			
0070h			
0071h			
0072h			
0073h			
0074h			
0075h			
0076h			
0077h			
0078h			
0079h			
007Ah			
007Bh			
007Ch			
007Dh			
007Eh			
007Fh			

Address	Register	Symbol	Page
0080h			
0081h			
0082h			
0083h			
0084h			
0085h			
0086h			
0087h			
to			
01AFh			
01B0h			
01B2h			
01B3h			
01B4h	Flash Identification Register	FIDR	276
01B5h	Flash Memory Control Register 1	FMR1	278
01B6h			
01B7h	Flash Memory Control Register 0	FMR0	277
01B8h	Address Match Interrupt Register 2	RMAD2	123
01B9h			
01BAh			
01BBh	Address Match Interrupt Enable Register 2	AIER2	123
01BCh	Address Match Interrupt Register 3	RMAD3	123
01BDh			
01BEh			
01BFh			
01C0h			
to			
02AFh			
0250h			
0251h			
0252h			
0253h			
0254h			
0255h			
0256h			
0257h			
0258h			
0259h			
025Ah			
025Bh			
025Ch			
025Dh			
025Eh	Peripheral Clock Select Register	PCLKR	87
025Fh			
0260h			
0261h			
0262h			
0263h			
0264h			
0265h			
0266h			
0267h			
0268h			
0269h			
026Ah			
to			
0335h			
0336h			
0337h			
0338h			
0339h			
033Ah			
033Bh			
033Ch			
033Dh			
033Eh			
033Fh			

Address	Register	Symbol	Page
0340h	Timer B3, 4, 5 Count Start Flag	TBSR	158
0341h			
0342h	Timer A1-1 Register	TA11	169
0343h			
0344h	Timer A2-1 Register	TA21	169
0345h			
0346h	Timer A4-1 Register	TA41	169
0347h			
0348h	Three-Phase PWM Control Register 0	INVC0	167
0349h	Three-Phase PWM Control Register 1	INVC1	168
034Ah	Three-Phase Output Buffer Register 0	IDB0	170
034Bh	Three-Phase Output Buffer Register 1	IDB1	170
034Ch	Dead Time Timer	DTT	171
034Dh	Timer B2 Interrupt Occurrence Frequency Set Counter	ICTB2	169
034Eh			
034Fh			
0350h	Timer B3 Register	TB3	157
0351h			
0352h	Timer B4 Register	TB4	157
0353h			
0354h	Timer B5 Register	TB5	157
0355h			
0356h			
0357h			
0358h			
0359h			
035Ah			
035Bh	Timer B3 Mode Register	TB3MR	157
035Ch	Timer B4 Mode Register	TB4MR	157
035Dh	Timer B5 Mode Register	TB5MR	157
035Eh	Interrupt Factor Select Register 2	IFSR2A	120
035Fh	Interrupt Factor Select Register	IFSR	120
0360h	SI/O3 Transmit/Receive Register	S3TRR	229
0361h			
0362h	SI/O3 Control Register	S3C	228
0363h	SI/O3 Bit Rate Generator	S3BRG	229
0364h	SI/O4 Transmit/Receive Register	S4TRR	229
0365h			
0366h	SI/O4 Control Register	S4C	228
0367h	SI/O4 Bit Rate Generator	S4BRG	229
0368h			
0369h			
036Ah			
036Bh			
036Ch	UART0 Special Mode Register 4	U0SMR4	188
036Dh	UART0 Special Mode Register 3	U0SMR3	187
036Eh	UART0 Special Mode Register 2	U0SMR2	187
036Fh	UART0 Special Mode Register	U0SMR	186
0370h	UART1 Special Mode Register 4	U1SMR4	188
0371h	UART1 Special Mode Register 3	U1SMR3	187
0372h	UART1 Special Mode Register 2	U1SMR2	187
0373h	UART1 Special Mode Register	U1SMR	186
0374h	UART2 Special Mode Register 4	U2SMR4	188
0375h	UART2 Special Mode Register 3	U2SMR3	187
0376h	UART2 Special Mode Register 2	U2SMR2	187
0377h	UART2 Special Mode Register	U2SMR	186
0378h	UART2 Transmit/Receive Mode Register	U2MR	183
0379h	UART2 Bit Rate Generator	U2BRG	182
037Ah	UART2 Transmit Buffer Register	U2TB	181
037Bh			
037Ch	UART2 Transmit/Receive Control Register 0	U2C0	184
037Dh	UART2 Transmit/Receive Control Register 1	U2C1	185
037Eh	UART2 Receive Buffer Register	U2RB	181
037Fh			

NOTES:

- Blank columns are all reserved space. No access is allowed.

Address	Register	Symbol	Page
0380h	Count Start Flag	TABSR	141, 158
0381h	Clock Prescaler Reset Flag	CPSRF	143, 158
0382h	One-Shot Start Flag	ONSF	142
0383h	Trigger Select Register	TRGSR	142
0384h	Up-Down Flag	UDF	141
0385h			
0386h 0387h	Timer A0 Register	TA0	140
0388h 0389h	Timer A1 Register	TA1	140
038Ah 038Bh	Timer A2 Register	TA2	140
038Ch 038Dh	Timer A3 Register	TA3	140
038Eh 038Fh	Timer A4 Register	TA4	140
0390h 0391h	Timer B0 Register	TB0	157
0392h 0393h	Timer B1 Register	TB1	157
0394h 0395h	Timer B2 Register	TB2	157
0396h	Timer A0 Mode Register	TA0MR	140
0397h	Timer A1 Mode Register	TA1MR	140
0398h	Timer A2 Mode Register	TA2MR	140
0399h	Timer A3 Mode Register	TA3MR	140
039Ah	Timer A4 Mode Register	TA4MR	140
039Bh	Timer B0 Mode Register	TB0MR	157
039Ch	Timer B1 Mode Register	TB1MR	157
039Dh	Timer B2 Mode Register	TB2MR	157
039Eh	Timer B2 Special Mode Register	TB2SC	170
039Fh			
03A0h	UART0 Transmit/Receive Mode Register	U0MR	183
03A1h	UART0 Bit Rate Generator	U0BRG	182
03A2h	UART0 Transmit Buffer Register	U0TB	181
03A3h			
03A4h	UART0 Transmit/Receive Control Register 0	U0C0	184
03A5h	UART0 Transmit/Receive Control Register 1	U0C1	185
03A6h	UART0 Receive Buffer Register	U0RB	181
03A7h			
03A8h	UART1 Transmit/Receive Mode Register	U1MR	183
03A9h	UART1 Bit Rate Generator	U1BRG	182
03AAh	UART1 Transmit Buffer Register	U1TB	181
03ABh			
03ACh	UART1 Transmit/Receive Control Register 0	U1C0	184
03ADh	UART1 Transmit/Receive Control Register 1	U1C1	185
03AEh	UART1 Receive Buffer Register	U1RB	181
03AFh			
03B0h	UART Transmit/Receive Control Register 2	UCON	186
03B1h			
03B2h			
03B3h			
03B4h			
03B5h			
03B6h			
03B7h			
03B8h	DMA0 Request Factor Select Register	DM0SL	128
03B9h			
03BAh	DMA1 Request Factor Select Register	DM1SL	129
03BBh			
03BCh	CRC Data Register	CRCD	253
03BDh			
03BEh	CRC Input Register	CRCIN	253
03BFh			

Address	Register	Symbol	Page
03C0h 03C1h	A/D Register 0	AD0	237
03C2h 03C3h	A/D Register 1	AD1	237
03C4h 03C5h	A/D Register 2	AD2	237
03C6h 03C7h	A/D Register 3	AD3	237
03C8h 03C9h	A/D Register 4	AD4	237
03CAh 03CBh	A/D Register 5	AD5	237
03CCh 03CDh	A/D Register 6	AD6	237
03CEh 03CFh	A/D Register 7	AD7	237
03D0h			
03D1h			
03D2h			
03D3h			
03D4h	A/D Control Register 2	ADCON2	236
03D5h			
03D6h	A/D Control Register 0	ADCON0	235
03D7h	A/D Control Register 1	ADCON1	235
03D8h	D/A Register 0	DA0	252
03D9h			
03DAh	D/A Register 1	DA1	252
03DBh			
03DCh	D/A Control Register	DACON	252
03DDh			
03DEh	Port P14 Control Register	PC14	264
03DFh	Pull-Up Control Register 3	PUR3	264
03E0h	Port P0 Register	P0	263
03E1h	Port P1 Register	P1	263
03E2h	Port P0 Direction Register	PD0	262
03E3h	Port P1 Direction Register	PD1	262
03E4h	Port P2 Register	P2	263
03E5h	Port P3 Register	P3	263
03E6h	Port P2 Direction Register	PD2	262
03E7h	Port P3 Direction Register	PD3	262
03E8h	Port P4 Register	P4	263
03E9h	Port P5 Register	P5	263
03EAh	Port P4 Direction Register	PD4	262
03EBh	Port P5 Direction Register	PD5	262
03ECh	Port P6 Register	P6	263
03EDh	Port P7 Register	P7	263
03EEh	Port P6 Direction Register	PD6	262
03EFh	Port P7 Direction Register	PD7	262
03F0h	Port P8 Register	P8	263
03F1h	Port P9 Register	P9	263
03F2h	Port P8 Direction Register	PD8	262
03F3h	Port P9 Direction Register	PD9	262
03F4h	Port P10 Register	P10	263
03F5h	Port P11 Register	P11	263
03F6h	Port P10 Direction Register	PD10	262
03F7h	Port P11 Direction Register	PD11	262
03F8h	Port P12 Register	P12	263
03F9h	Port P13 Register	P13	263
03FAh	Port P12 Direction Register	PD12	262
03FBh	Port P13 Direction Register	PD13	262
03FCh	Pull-Up Control Register 0	PUR0	265
03FDh	Pull-Up Control Register 1	PUR1	265
03FEh	Pull-Up Control Register 2	PUR2	266
03FFh	Port Control Register	PCR	266

NOTES:

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

## **1. Overview**

The M16C/62P Group (M16C/62P, M16C/62PT) of single-chip microcomputers are built using the high performance silicon gate CMOS process using a M16C/60 Series CPU core and are packaged in a 80-pin, 100-pin and 128-pin plastic molded QFP. These single-chip microcomputers operate using sophisticated instructions featuring a high level of instruction efficiency. With 1M bytes of address space, they are capable of executing instructions at high speed. In addition, this microcomputer contains a multiplier and DMAC which combined with fast instruction processing capability, makes it suitable for control of various OA, communication, and industrial equipment which requires high-speed arithmetic/logic operations.

### **1.1 Applications**

Audio, cameras, television, home appliance, office/communications/portable/industrial equipment, automobile, etc.

Specifications written in this manual are believed to be accurate, but are not guaranteed to be entirely free of error. Specifications in this manual may be changed for functional or performance improvements. Please make sure your manual is the latest edition.



## 1.2 Performance Outline

Table 1.1 to 1.3 list Performance Outline of M16C/62P Group (M16C/62P, M16C/62PT)(128-pin version).

**Table 1.1 Performance Outline of M16C/62P Group (M16C/62P, M16C/62PT)(128-pin version)**

	Item	Performance
		M16C/62P
CPU	Number of Basic Instructions	91 instructions
	Minimum Instruction Execution Time	41.7ns(f(BCLK)=24MHz, VCC1=3.3 to 5.5V) 100ns(f(BCLK)=10MHz, VCC1=2.7 to 5.5V)
	Operating Mode	Single-chip, memory expansion and microprocessor mode
	Address Space	1 Mbyte (Available to 4 Mbytes by memory space expansion function)
	Memory Capacity	See Table 1.4 to 1.5 Product List
Peripheral Function	Port	Input/Output : 113 pins, Input : 1 pin
	Multifunction Timer	Timer A : 16 bits x 5 channels, Timer B : 16 bits x 6 channels, Three phase motor control circuit
	Serial Interface	3 channels Clock synchronous, UART, I <sup>2</sup> C bus <sup>(1)</sup> , IEBus <sup>(2)</sup> 2 channels Clock synchronous
	A/D Converter	10-bit A/D converter: 1 circuit, 26 channels
	D/A Converter	8 bits x 2 channels
	DMAC	2 channels
	CRC Calculation Circuit	CCITT-CRC
	Watchdog Timer	15 bits x 1 channel (with prescaler)
	Interrupt	Internal: 29 sources, External: 8 sources, Software: 4 sources, Priority level: 7 levels
	Clock Generation Circuit	4 circuits Main clock generation circuit (*), Subclock generation circuit (*), On-chip oscillator, PLL synthesizer (*)Equipped with a built-in feedback resistor.
	Oscillation Stop Detection Function	Stop detection of main clock oscillation, re-oscillation detection function
Electric Characteristics	Supply Voltage	VCC1=3.0 to 5.5 V, VCC2=2.7V to VCC1 (f(BCLK=24MHz) VCC1=2.7 to 5.5 V, VCC2=2.7V to VCC1 (f(BCLK=10MHz)
	Power Consumption	14 mA (VCC1=VCC2=5V, f(BCLK)=24MHz) 8 mA (VCC1=VCC2=3V, f(BCLK)=10MHz) 1.8μA (VCC1=VCC2=3V, f(XCIN)=32kHz, wait mode) 0.7μA (VCC1=VCC2=3V, stop mode)
Flash memory version	Program/Erase Supply Voltage	3.3±0.3 V or 5.0±0.5 V
	Program and Erase Endurance	100 times (all area) or 1,000 times (user ROM area without block A and block 1) / 10,000 times (block A, block 1) <sup>(3)</sup>
Operating Ambient Temperature		-20 to 85°C, -40 to 85°C <sup>(3)</sup>
Package		128-pin plastic mold LQFP

### NOTES:

1. I<sup>2</sup>C bus is a registered trademark of Koninklijke Philips Electronics N. V.
2. IEBus is a registered trademark of NEC Electronics Corporation.
3. See **Table 1.8 Product Code** for the program and erase endurance, and operating ambient temperature.  
In addition 1,000 times/10,000 times are under development as of Jul., 2005. Please inquire about a release schedule.
4. All options are on request basis.

**Table 1.2 Performance Outline of M16C/62P Group (M16C/62P, M16C/62PT)(100-pin version)**

	Item	Performance	
		M16C/62P	M16C/62PT <sup>(4)</sup>
CPU	Number of Basic Instructions	91 instructions	
	Minimum Instruction Execution Time	41.7ns(f(BCLK)=24MHz, VCC1=3.3 to 5.5V) 100ns(f(BCLK)=10MHz, VCC1=2.7 to 5.5V)	41.7ns(f(BCLK)=24MHz, VCC1=4.0 to 5.5V)
	Operating Mode	Single-chip, memory expansion and microprocessor mode	Single-chip
	Address Space	1 Mbyte (Available to 4 Mbytes by memory space expansion function)	1 Mbyte
	Memory Capacity	See <b>Table 1.4 to 1.7 Product List</b>	
Peripheral Function	Port	Input/Output : 87 pins, Input : 1 pin	
	Multifunction Timer	Timer A : 16 bits x 5 channels, Timer B : 16 bits x 6 channels, Three phase motor control circuit	
	Serial Interface	3 channels Clock synchronous, UART, I <sup>2</sup> C bus <sup>(1)</sup> , IEBus <sup>(2)</sup> 2 channels Clock synchronous	
	A/D Converter	10-bit A/D converter: 1 circuit, 26 channels	
	D/A Converter	8 bits x 2 channels	
	DMAC	2 channels	
	CRC Calculation Circuit	CCITT-CRC	
	Watchdog Timer	15 bits x 1 channel (with prescaler)	
	Interrupt	Internal: 29 sources, External: 8 sources, Software: 4 sources, Priority level: 7 levels	
	Clock Generation Circuit	4 circuits Main clock generation circuit (*), Subclock generation circuit (*), On-chip oscillator, PLL synthesizer (*)Equipped with a built-in feedback resistor.	
	Oscillation Stop Detection Function	Stop detection of main clock oscillation, re-oscillation detection function	
Electric Characteristics	Voltage Detection Circuit	Available (option <sup>(5)</sup> )	Absent
	Supply Voltage	VCC1=3.0 to 5.5 V, VCC2=2.7V to VCC1 (f(BCLK=24MHz) VCC1=2.7 to 5.5 V, VCC2=2.7V to VCC1 (f(BCLK=10MHz)	VCC1=VCC2=4.0 to 5.5V (f(BCLK=24MHz)
	Power Consumption	14 mA (VCC1=VCC2=5V, f(BCLK)=24MHz) 8 mA (VCC1=VCC2=3V, f(BCLK)=10MHz) 1.8μA (VCC1=VCC2=3V, f(XCIN)=32kHz, wait mode) 0.7μA (VCC1=VCC2=3V, stop mode)	14 mA (VCC1=VCC2=5V, f(BCLK)=24MHz) 2.0μA (VCC1=VCC2=5V, f(XCIN)=32kHz, wait mode) 0.8μA (VCC1=VCC2=5V, stop mode)
Flash memory version	Program/Erase Supply Voltage	3.3±0.3 V or 5.0±0.5 V	5.0±0.5 V
	Program and Erase Endurance	100 times (all area) or 1,000 times (user ROM area without block A and block 1) / 10,000 times (block A, block 1) <sup>(3)</sup>	
Operating Ambient Temperature		-20 to 85°C, -40 to 85°C <sup>(3)</sup>	T version : -40 to 85°C V version : -40 to 125°C
Package		100-pin plastic mold QFP, LQFP	

## NOTES:

1. I<sup>2</sup>C bus is a registered trademark of Koninklijke Philips Electronics N. V.
2. IEBus is a registered trademark of NEC Electronics Corporation.
3. See **Table 1.8 and 1.9 Product Code** for the program and erase endurance, and operating ambient temperature.  
In addition 1,000 times/10,000 times are under development as of Jul., 2005. Please inquire about a release schedule.
4. Use the M16C/62PT on VCC1=VCC2
5. All options are on request basis.

**Table 1.3 Performance Outline of M16C/62P Group (M16C/62P, M16C/62PT)(80-pin version)**

	Item	Performance	
		M16C/62P	M16C/62PT <sup>(4)</sup>
CPU	Number of Basic Instructions	91 instructions	
	Minimum Instruction Execution Time	41.7ns(f(BCLK)=24MHz, VCC1=3.3 to 5.5V) 100ns(f(BCLK)=10MHz, VCC1=2.7 to 5.5V)	41.7ns(f(BCLK)=24MHz, VCC1=4.0 to 5.5V)
	Operating Mode	Single-chip mode	
	Address Space	1 Mbyte	
	Memory Capacity	See <b>Table 1.4 to 1.7 Product List</b>	
Peripheral Function	Port	Input/Output : 70 pins, Input : 1 pin	
	Multifunction Timer	Timer A : 16 bits x 5 channels (Timer A1 and A2 are internal timer), Timer B : 16 bits x 6 channels (Timer B1 is internal timer)	
	Serial Interface	2 channels Clock synchronous, UART, I <sup>2</sup> C bus <sup>(1)</sup> , IEBus <sup>(2)</sup> 1 channel Clock synchronous, I <sup>2</sup> C bus <sup>(1)</sup> , IEBus <sup>(2)</sup> 2 channels Clock synchronous (1 channel is only transmission)	
	A/D Converter	10-bit A/D converter: 1 circuit, 26 channels	
	D/A Converter	8 bits x 2 channels	
	DMAC	2 channels	
	CRC Calculation Circuit	CCITT-CRC	
	Watchdog Timer	15 bits x 1 channel (with prescaler)	
	Interrupt	Internal: 29 sources, External: 5 sources, Software: 4 sources, Priority level: 7 levels	
	Clock Generation Circuit	4 circuits Main clock generation circuit (*), Subclock generation circuit (*), On-chip oscillator, PLL synthesizer (*)Equipped with a built-in feedback resistor.	
	Oscillation Stop Detection Function	Stop detection of main clock oscillation, re-oscillation detection function	
	Voltage Detection Circuit	Available (option <sup>(4)</sup> )	Absent
Electric Characteristics	Supply Voltage	VCC1=3.0 to 5.5 V, (f(BCLK)=24MHz) VCC1=2.7 to 5.5 V, (f(BCLK)=10MHz)	VCC1=4.0 to 5.5V, (f(BCLK)=24MHz)
	Power Consumption	14 mA (VCC1=5V, f(BCLK)=24MHz) 8 mA (VCC1=3V, f(BCLK)=10MHz) 1.8μA (VCC1=3V, f(XCIN)=32kHz, wait mode) 0.7μA (VCC1=3V, stop mode)	14 mA (VCC1=5V, f(BCLK)=24MHz) 2.0μA (VCC1=5V, f(XCIN)=32kHz, wait mode) 0.8μA (VCC1=5V, stop mode)
Flash memory version	Program/Erase Supply Voltage	3.3 ± 0.3V or 5.0 ± 0.5V	5.0 ± 0.5V
	Program and Erase Endurance	100 times (all area) or 1,000 times (user ROM area without block A and block 1) / 10,000 times (block A, block 1) <sup>(3)</sup>	
Operating Ambient Temperature		-20 to 85°C, -40 to 85°C <sup>(3)</sup>	T version : -40 to 85°C V version : -40 to 125°C
Package		80-pin plastic mold QFP	

## NOTES:

- I<sup>2</sup>C bus is a registered trademark of Koninklijke Philips Electronics N. V.
- IEBus is a registered trademark of NEC Electronics Corporation.
- See **Table 1.8 and 1.9 Product Code** for the program and erase endurance, and operating ambient temperature.  
In addition 1,000 times/10,000 times are under development as of Jul., 2005. Please inquire about a release schedule.
- All options are on request basis.

### 1.3 Block Diagram

Figure 1.1 is a M16C/62P Group (M16C/62P, M16C/62PT) 128-pin and 100-pin version Block Diagram,  
Figure 1.2 is a M16C/62P Group (M16C/62P, M16C/62PT) 80-pin version Block Diagram.

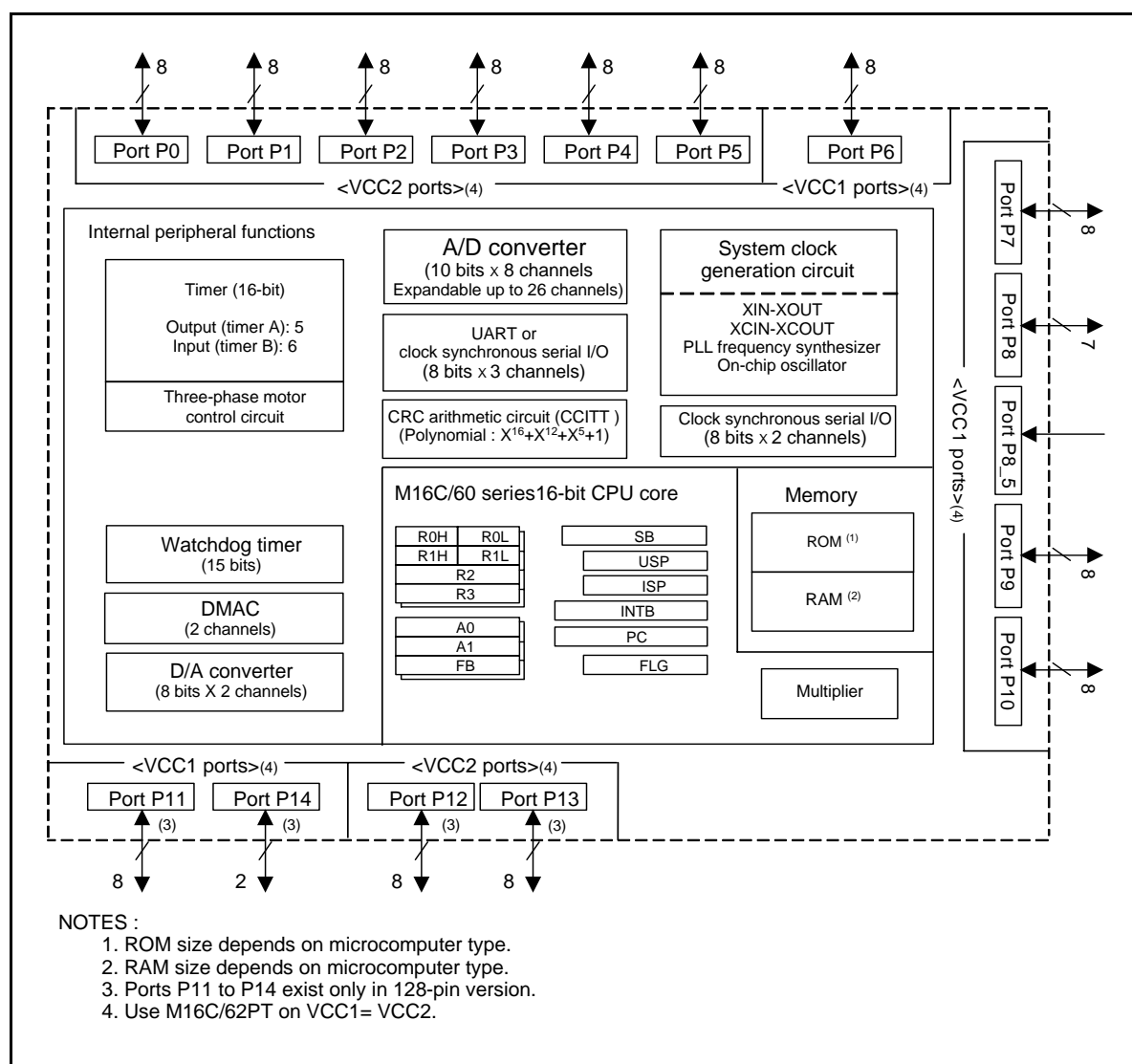


Figure 1.1 M16C/62P Group (M16C/62P, M16C/62PT) 128-pin and 100-pin version Block Diagram

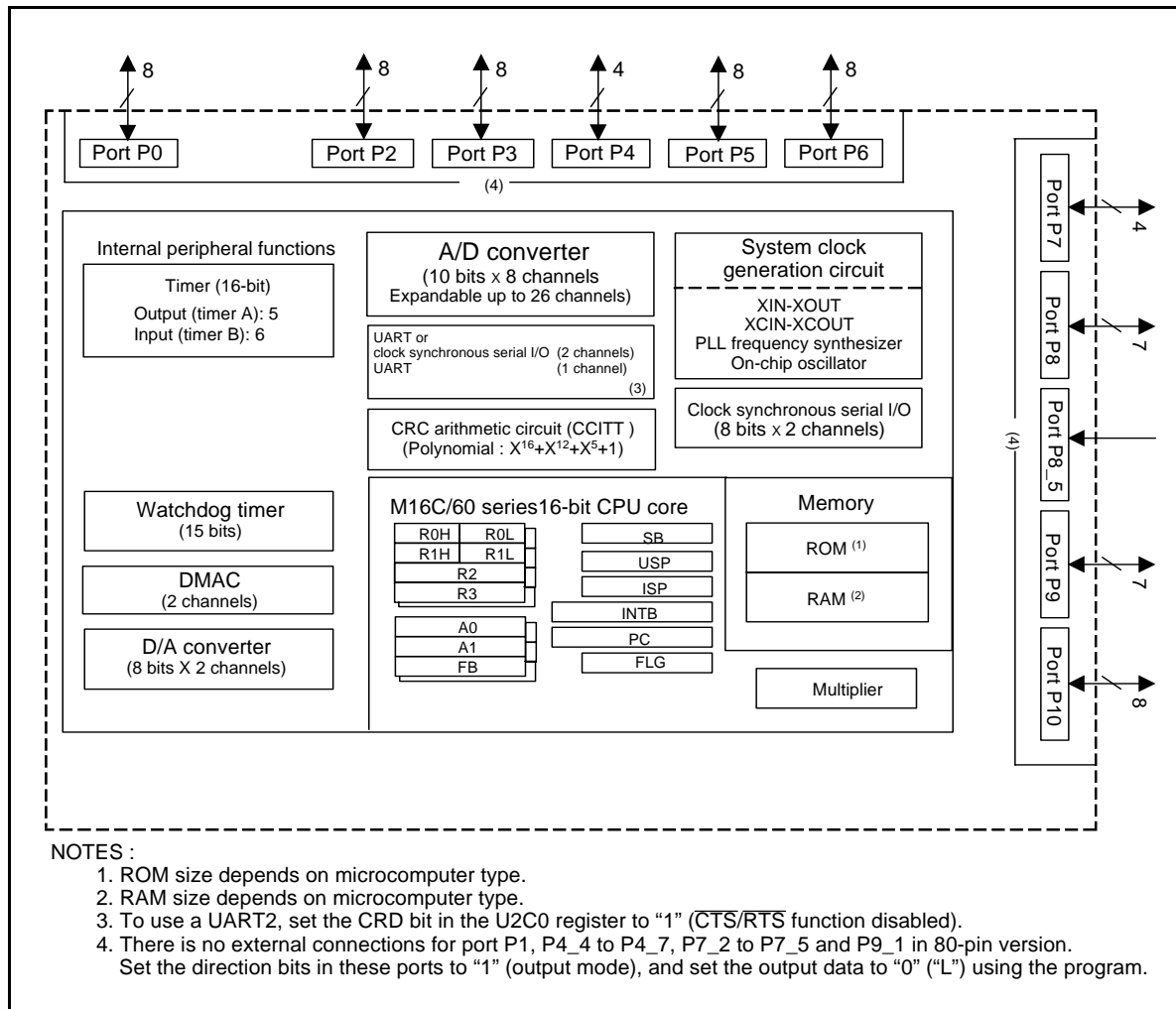


Figure 1.2 M16C/62P Group (M16C/62P, M16C/62PT) 80-pin version Block Diagram

## 1.4 Product List

Table 1.4 to 1.7 list the product list, Figure 1.3 shows the Type No., Memory Size, and Package, Table 1.8 lists the Product Code of Flash Memory version and ROMless version for M16C/62P, and Table 1.9 lists the Product Code of Flash Memory version for M16C/62PT. Figure 1.4 shows the Marking Diagram of Flash Memory version and ROM-less version for M16C/62P (Top View), and Figure 1.5 shows the Marking Diagram of Flash Memory version for M16C/62PT (Top View) at the time of ROM order.

**Table 1.4 Product List (1) (M16C/62P)**
**As of Dec. 2005**

Type No.	ROM Capacity	RAM Capacity	Package Type <sup>(1)</sup>	Remarks
M30622M6P-XXXFP	48 Kbytes	4 Kbytes	PRQP0100JB-A	Mask ROM version
M30622M6P-XXXGP			PLQP0100KB-A	
M30622M8P-XXXFP	64 Kbytes	4 Kbytes	PRQP0100JB-A	
M30622M8P-XXXGP			PLQP0100KB-A	
M30623M8P-XXXGP			PRQP0080JA-A	
M30622MAP-XXXFP	96 Kbytes	5 Kbytes	PRQP0100JB-A	
M30622MAP-XXXGP			PLQP0100KB-A	
M30623MAP-XXXGP			PRQP0080JA-A	
M30620MCP-XXXFP	128 Kbytes	10 Kbytes	PRQP0100JB-A	
M30620MCP-XXXGP			PLQP0100KB-A	
M30621MCP-XXXGP			PRQP0080JA-A	
M30622MEP-XXXFP	192 Kbytes	12 Kbytes	PRQP0100JB-A	
M30622MEP-XXXGP			PLQP0100KB-A	
M30623MEP-XXXGP			PLQP0128KB-A	
M30622MGP-XXXFP	256 Kbytes	12 Kbytes	PRQP0100JB-A	
M30622MGP-XXXGP			PLQP0100KB-A	
M30623MGP-XXXGP			PLQP0128KB-A	
M30624MGP-XXXFP		20 Kbytes	PRQP0100JB-A	
M30624MGP-XXXGP			PLQP0100KB-A	
M30625MGP-XXXGP			PLQP0128KB-A	
M30622MWP-XXXFP	320 Kbytes	16 Kbytes	PRQP0100JB-A	
M30622MWP-XXXGP			PLQP0100KB-A	
M30623MWP-XXXGP			PLQP0128KB-A	
M30624MWP-XXXFP		24 Kbytes	PRQP0100JB-A	
M30624MWP-XXXGP			PLQP0100KB-A	
M30625MWP-XXXGP			PLQP0128KB-A	
M30626MWP-XXXFP		31 Kbytes	PRQP0100JB-A	
M30626MWP-XXXGP			PLQP0100KB-A	
M30627MWP-XXXGP			PLQP0128KB-A	

(D): Under development

### NOTES:

- The old package type numbers of each package type are as follows.

PLQP0128KB-A : 128P6Q-A,  
 PRQP0100JB-A : 100P6S-A,  
 PLQP0100KB-A : 100P6Q-A,  
 PRQP0080JA-A : 80P6S-A

Table 1.5 Product List (2) (M16C/62P)

As of Dec. 2005

Type No.	ROM Capacity	RAM Capacity	Package Type (1)	Remarks
M30622MHP-XXXFP	384 Kbytes	16 Kbytes	PRQP0100JB-A	Mask ROM version
M30622MHP-XXXGP			PLQP0100KB-A	
M30623MHP-XXXGP			PLQP0128KB-A	
M30624MHP-XXXFP		24 Kbytes	PRQP0100JB-A	
M30624MHP-XXXGP			PLQP0100KB-A	
M30625MHP-XXXGP			PLQP0128KB-A	
M30626MHP-XXXFP		31 Kbytes	PRQP0100JB-A	
M30626MHP-XXXGP			PLQP0100KB-A	
M30627MHP-XXXGP			PLQP0128KB-A	
M30626MJP-XXXFP (D)	512 Kbytes	31 Kbytes	PRQP0100JB-A	
M30626MJP-XXXGP (D)			PLQP0100KB-A	
M30627MJP-XXXGP (D)			PLQP0128KB-A	
M30622F8PFP	64K+4 Kbytes	4 Kbytes	PRQP0100JB-A	Flash memory version (2)
M30622F8PGP			PLQP0100KB-A	
M30623F8PGP			PRQP0080JA-A	
M30620FCPFP	128K+4 Kbytes	10 Kbytes	PRQP0100JB-A	
M30620FCPGP			PLQP0100KB-A	
M30621FCPGP			PRQP0080JA-A	
M3062LFGPFP(3) (D)	256K+4 Kbytes	20 Kbytes	PRQP0100JB-A	
M3062LFGPGP(3) (D)			PLQP0100KB-A	
M30625FGPGP			PLQP0128KB-A	
M30626FHPFP	384K+4 Kbytes	31 Kbytes	PRQP0100JB-A	
M30626FHPPGP			PLQP0100KB-A	
M30627FHPPGP			PLQP0128KB-A	
M30626FJPFP	512K+4 Kbytes	31 Kbytes	PRQP0100JB-A	
M30626FJPGP			PLQP0100KB-A	
M30627FJPGP			PLQP0128KB-A	
M30622SPFP	—	4 Kbytes	PRQP0100JB-A	ROM-less version
M30622SPGP			PLQP0100KB-A	
M30620SPFP		10 Kbytes	PRQP0100JB-A	
M30620SPGP			PLQP0100KB-A	
M30624SPFP (D)	—	20 Kbytes	PRQP0100JB-A	
M30624SPGP (D)			PLQP0100KB-A	
M30626SPFP (D)		31 Kbytes	PRQP0100JB-A	
M30626SPGP (D)			PLQP0100KB-A	

(D): Under development

## NOTES:

- The old package type numbers of each package type are as follows.  
 PLQP0128KB-A : 128P6Q-A,  
 PRQP0100JB-A : 100P6S-A,  
 PLQP0100KB-A : 100P6Q-A,  
 PRQP0080JA-A : 80P6S-A
- In the flash memory version, there is 4K bytes area (block A).
- Please use M3062LFGPFP and M3062LFGPGP for your new system instead of M30624FGPFP and M30624FGPGP. The M16C/62P Group (M16C/62P, M16C/62PT) hardware manual is still good for M30624FGPFP and M30624FGPGP.

M30624FGPFP	256K+4 Kbytes	20 Kbytes	PRQP0100JB-A	Flash memory version
M30624FGPGP			PLQP0100KB-A	

Table 1.6 Product List (3) (T version (M16C/62PT))

As of Dec. 2005

Type No.	ROM Capacity	RAM Capacity	Package Type <sup>(1)</sup>	Remarks	
M3062CM6T-XXXFP (D)	48 Kbytes	4 Kbytes	PRQP0100JB-A	Mask ROM version	T Version (High reliability 85°C version)
M3062CM6T-XXXGP (D)			PLQP0100KB-A		
M3062EM6T-XXXGP (P)			PRQP0080JA-A		
M3062CM8T-XXXFP (D)	64 Kbytes	4 Kbytes	PRQP0100JB-A		
M3062CM8T-XXXGP (D)			PLQP0100KB-A		
M3062EM8T-XXXGP (P)			PRQP0080JA-A		
M3062CMAT-XXXFP (D)	96 Kbytes	5 Kbytes	PRQP0100JB-A		
M3062CMAT-XXXGP (D)			PLQP0100KB-A		
M3062EMAT-XXXGP (P)			PRQP0080JA-A		
M3062AMCT-XXXFP (D)	128 Kbytes	10 Kbytes	PRQP0100JB-A		
M3062AMCT-XXXGP (D)			PLQP0100KB-A		
M3062BMCT-XXXGP (P)			PRQP0080JA-A		
M3062CF8TFP (D)	64 K+4 Kbytes	4 Kbytes	PRQP0100JB-A	Flash memory version <sup>(2)</sup>	
M3062CF8TGP			PLQP0100KB-A		
M3062AFCTFP (D)	128K+4 Kbytes	10 Kbytes	PRQP0100JB-A		
M3062AFCTGP (D)			PLQP0100KB-A		
M3062BFCTGP (P)			PRQP0080JA-A		
M3062JFHTFP (D)	384K+4 Kbytes	31 Kbytes	PRQP0100JB-A		
M3062JFHTGP (D)			PLQP0100KB-A		

(D): Under development

(P): Under planning

## NOTES:

- The old package type numbers of each package type are as follows.  
PRQP0100JB-A : 100P6S-A,  
PLQP0100KB-A : 100P6Q-A,  
PRQP0080JA-A : 80P6S-A
- In the flash memory version, there is 4K bytes area (block A).



Table 1.7 Product List (4) (V version (M16C/62PT))

As of Dec. 2005

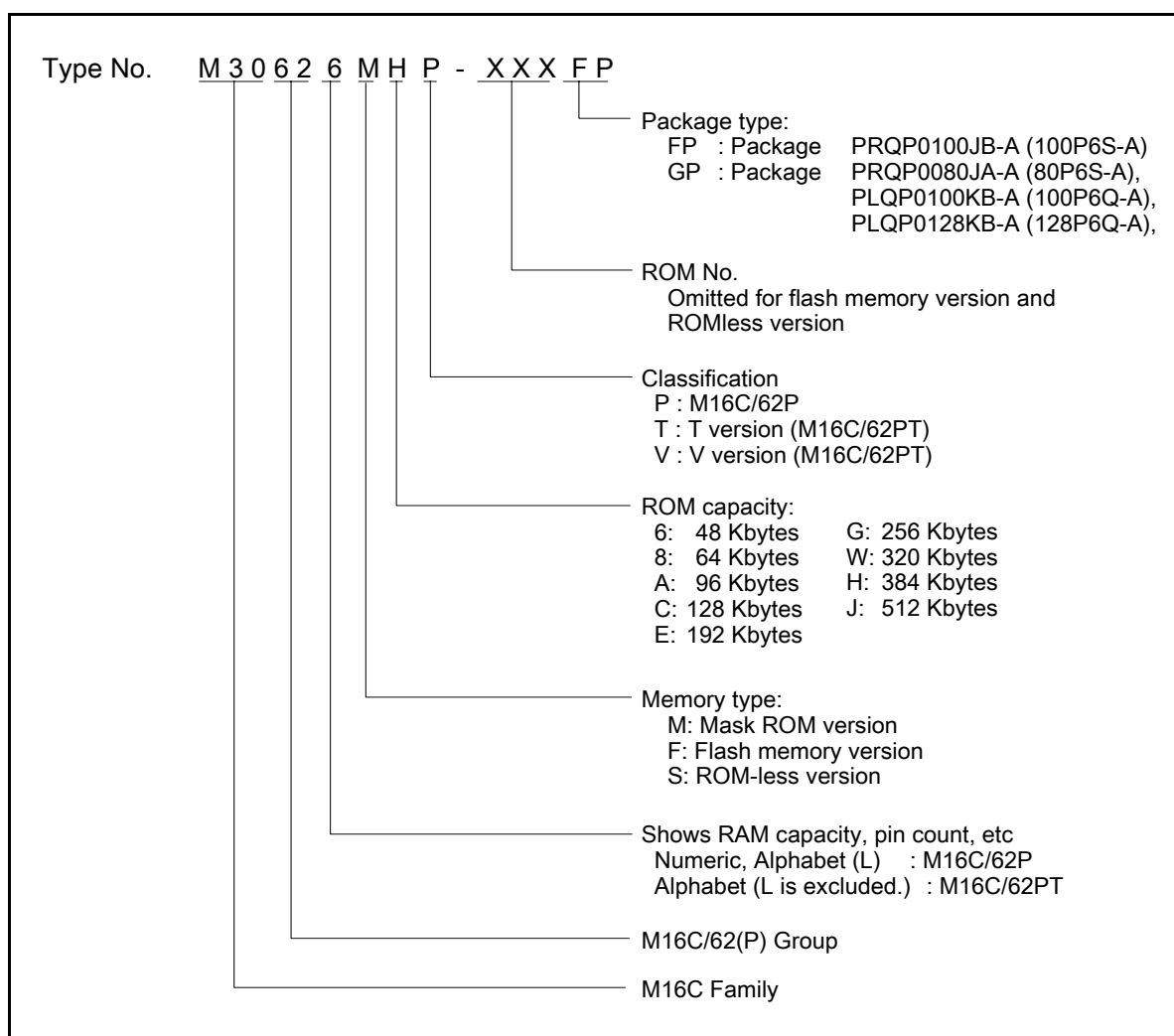
Type No.	ROM Capacity	RAM Capacity	Package Type <sup>(1)</sup>	Remarks	
M3062CM6V-XXXFP (P)	48 Kbytes	4 Kbytes	PRQP0100JB-A	Mask ROM version	V Version (High reliability 125°C version)
M3062CM6V-XXXGP (P)			PLQP0100KB-A		
M3062EM6V-XXXGP (P)			PRQP0080JA-A		
M3062CM8V-XXXFP (P)	64 Kbytes	4 Kbytes	PRQP0100JB-A		
M3062CM8V-XXXGP (P)			PLQP0100KB-A		
M3062EM8V-XXXGP (P)			PRQP0080JA-A		
M3062CMAV-XXXFP (P)	96 Kbytes	5 Kbytes	PRQP0100JB-A		
M3062CMAV-XXXGP (P)			PLQP0100KB-A		
M3062EMAV-XXXGP (P)			PRQP0080JA-A		
M3062AMCV-XXXFP (D)	128 Kbytes	10 Kbytes	PRQP0100JB-A		
M3062AMCV-XXXGP (D)			PLQP0100KB-A		
M3062BMCV-XXXGP (P)			PRQP0080JA-A		
M3062AFCVFP (D)	128K+4 Kbytes	10 Kbytes	PRQP0100JB-A	Flash memory version <sup>(2)</sup>	
M3062AFCVGP (D)			PLQP0100KB-A		
M3062BFCVGP (P)			PRQP0080JA-A		
M3062JFHVFP (P)	384K+4 Kbytes	31 Kbytes	PRQP0100JB-A		
M3062JFHVGP (P)			PLQP0100KB-A		

(D): Under development

(P): Under planning

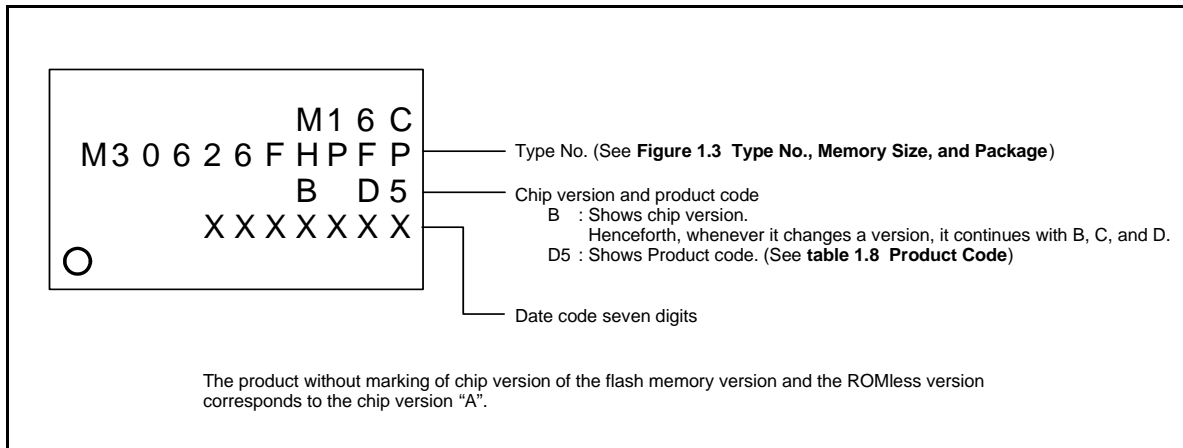
## NOTES:

- The old package type numbers of each package type are as follows.  
 PLQP0128KB-A : 128P6Q-A,  
 PRQP0100JB-A : 100P6S-A,  
 PLQP0100KB-A : 100P6Q-A,  
 PRQP0080JA-A : 80P6S-A
- In the flash memory version, there is 4K bytes area (block A).

**Figure 1.3    Type No., Memory Size, and Package**

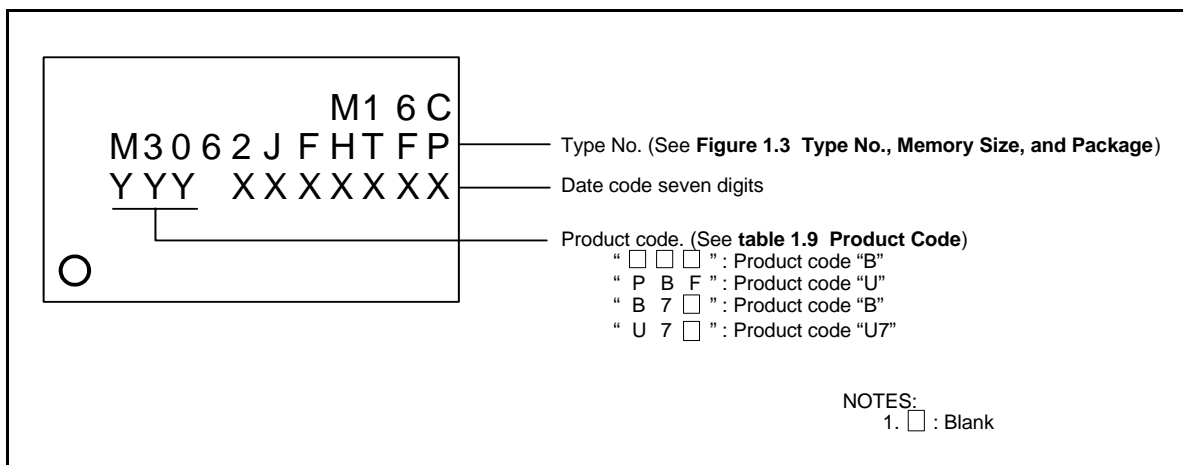
**Table 1.8 Product Code of Flash Memory version and ROMless version for M16C/62P**

	Product Code	Package	Internal ROM (User ROM Area Without Block A, Block 1)		Internal ROM (Block A, Block 1)		Operating Ambient Temperature
			Program and Erase Endurance	Temperature Range	Program and Erase Endurance	Temperature Range	
Flash memory Version	D3	Lead-included	100	0°C to 60°C	100	0°C to 60°C	-40°C to 85°C
	D5						-20°C to 85°C
	D7		1,000		10,000	-40°C to 85°C	-40°C to 85°C
	D9					-20°C to 85°C	-20°C to 85°C
	U3	Lead-free	100		100	0°C to 60°C	-40°C to 85°C
	U5						-20°C to 85°C
	U7		1,000		10,000	-40°C to 85°C	-40°C to 85°C
	U9					-20°C to 85°C	-20°C to 85°C
ROM-less version	D3	Lead-included	–	–	–	–	-40°C to 85°C
	D5						-20°C to 85°C
	U3	Lead-free	–	–	–	–	-40°C to 85°C
	U5						-20°C to 85°C

**Figure 1.4 Marking Diagram of Flash Memory version and ROM-less version for M16C/62P (Top View)**

**Table 1.9 Product Code of Flash Memory version for M16C/62PT**

		Product Code	Package	Internal ROM (User ROM Area Without Block A, Block 1)		Internal ROM (Block A, Block 1)		Operating Ambient Temperature
				Program and Erase Endurance	Temperature Range	Program and Erase Endurance	Temperature Range	
Flash memory Version	T Version	B	Lead- included	100	0°C to 60°C	100	0°C to 60°C	-40°C to 85°C
	V Version							-40°C to 125°C
	T Version	B7		1,000		10,000	-40°C to 85°C	-40°C to 85°C
	V Version						-40°C to 125°C	-40°C to 125°C
	T Version	U	Lead-free	100		100	0°C to 60°C	-40°C to 85°C
	V Version							-40°C to 125°C
	T Version	U7		1,000		10,000	-40°C to 85°C	-40°C to 85°C
	V Version						-40°C to 125°C	-40°C to 125°C

**Figure 1.5 Marking Diagram of Flash Memory version for M16C/62PT (Top View)**

## 1.5 Pin Configuration

Figures 1.6 to 1.9 show the Pin Configuration (Top View).

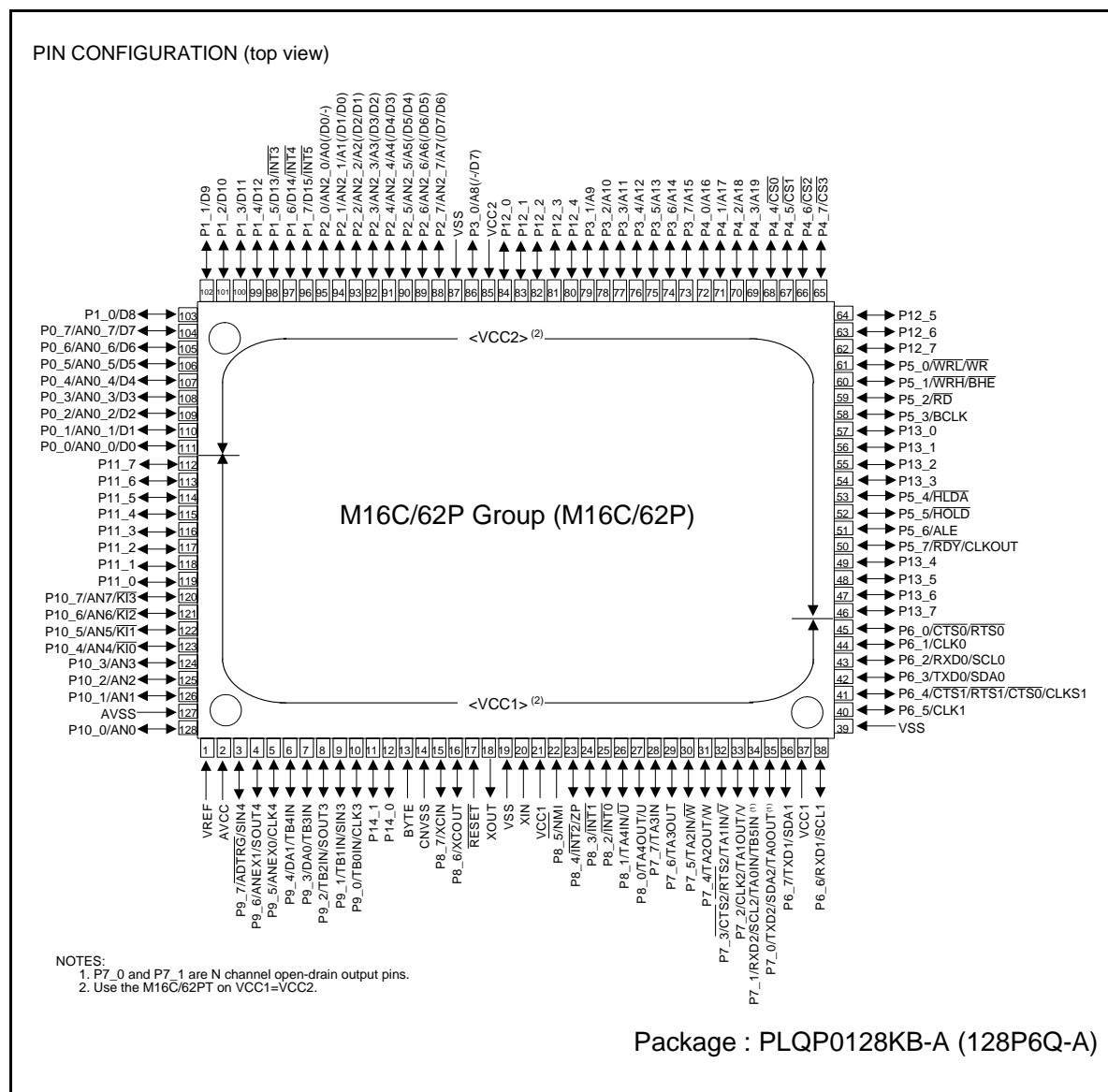


Figure 1.6 Pin Configuration (Top View)

**Table 1.10 Pin Characteristics for 128-Pin Package (1)**

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Analog Pin	Bus Control Pin
1	VREF						
2	AVCC						
3		P9_7			SIN4	ADTRG	
4		P9_6			SOUT4	ANEX1	
5		P9_5			CLK4	ANEX0	
6		P9_4		TB4IN		DA1	
7		P9_3		TB3IN		DA0	
8		P9_2		TB2IN	SOUT3		
9		P9_1		TB1IN	SIN3		
10		P9_0		TB0IN	CLK3		
11		P14_1					
12		P14_0					
13	BYTE						
14	CNVSS						
15	XCIN	P8_7					
16	XCOU	P8_6					
17	RESET						
18	XOUT						
19	VSS						
20	XIN						
21	VCC1						
22		P8_5	NMI				
23		P8_4	INT2	ZP			
24		P8_3	INT1				
25		P8_2	INT0				
26		P8_1		TA4IN/ $\bar{U}$			
27		P8_0		TA4OUT/U			
28		P7_7		TA3IN			
29		P7_6		TA3OUT			
30		P7_5		TA2IN/ $\bar{W}$			
31		P7_4		TA2OUT/W			
32		P7_3		TA1IN/ $\bar{V}$	CTS2/RTS2		
33		P7_2		TA1OUT/V	CLK2		
34		P7_1		TA0IN/TB5IN	RXD2/SCL2		
35		P7_0		TA0OUT	TXD2/SDA2		
36		P6_7			TXD1/SDA1		
37	VCC1						
38		P6_6			RXD1/SCL1		
39	VSS						
40		P6_5			CLK1		
41		P6_4			CTS1/RTS1/CTS0/CLKS1		
42		P6_3			TXD0/SDA0		
43		P6_2			RXD0/SCL0		
44		P6_1			CLK0		
45		P6_0			CTS0/RTS0		
46		P13_7					
47		P13_6					
48		P13_5					
49		P13_4					
50		P5_7					RDY/CLKOUT

**Table 1.11 Pin Characteristics for 128-Pin Package (2)**

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Analog Pin	Bus Control Pin
51		P5_6					ALE
52		P5_5					HOLD
53		P5_4					HLDA
54		P13_3					
55		P13_2					
56		P13_1					
57		P13_0					
58		P5_3					BCLK
59		P5_2					RD
60		P5_1					WRH/BHE
61		P5_0					WRL/WR
62		P12_7					
63		P12_6					
64		P12_5					
65		P4_7					CS3
66		P4_6					CS2
67		P4_5					CS1
68		P4_4					CS0
69		P4_3					A19
70		P4_2					A18
71		P4_1					A17
72		P4_0					A16
73		P3_7					A15
74		P3_6					A14
75		P3_5					A13
76		P3_4					A12
77		P3_3					A11
78		P3_2					A10
79		P3_1					A9
80		P12_4					
81		P12_3					
82		P12_2					
83		P12_1					
84		P12_0					
85	VCC2						
86		P3_0					A8(/-D7)
87	VSS						
88		P2_7				AN2_7	A7(/D7/D6)
89		P2_6				AN2_6	A6(/D6/D5)
90		P2_5				AN2_5	A5(/D5/D4)
91		P2_4				AN2_4	A4(/D4/D3)
92		P2_3				AN2_3	A3(/D3/D2)
93		P2_2				AN2_2	A2(/D2/D1)
94		P2_1				AN2_1	A1(/D1/D0)
95		P2_0				AN2_0	A0(/D0/-)
96		P1_7	INT5				D15
97		P1_6	INT4				D14
98		P1_5	INT3				D13
99		P1_4					D12
100		P1_3					D11

**Table 1.12 Pin Characteristics for 128-Pin Package (3)**

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Analog Pin	Bus Control Pin
101		P1_2					D10
102		P1_1					D9
103		P1_0					D8
104		P0_7				AN0_7	D7
105		P0_6				AN0_6	D6
106		P0_5				AN0_5	D5
107		P0_4				AN0_4	D4
108		P0_3				AN0_3	D3
109		P0_2				AN0_2	D2
110		P0_1				AN0_1	D1
111		P0_0				AN0_0	D0
112		P11_7					
113		P11_6					
114		P11_5					
115		P11_4					
116		P11_3					
117		P11_2					
118		P11_1					
119		P11_0					
120		P10_7	$\overline{\text{KI3}}$			AN7	
121		P10_6	$\overline{\text{KI2}}$			AN6	
122		P10_5	$\overline{\text{KI1}}$			AN5	
123		P10_4	$\overline{\text{KI0}}$			AN4	
124		P10_3				AN3	
125		P10_2				AN2	
126		P10_1				AN1	
127	AVSS						
128		P10_0				AN0	



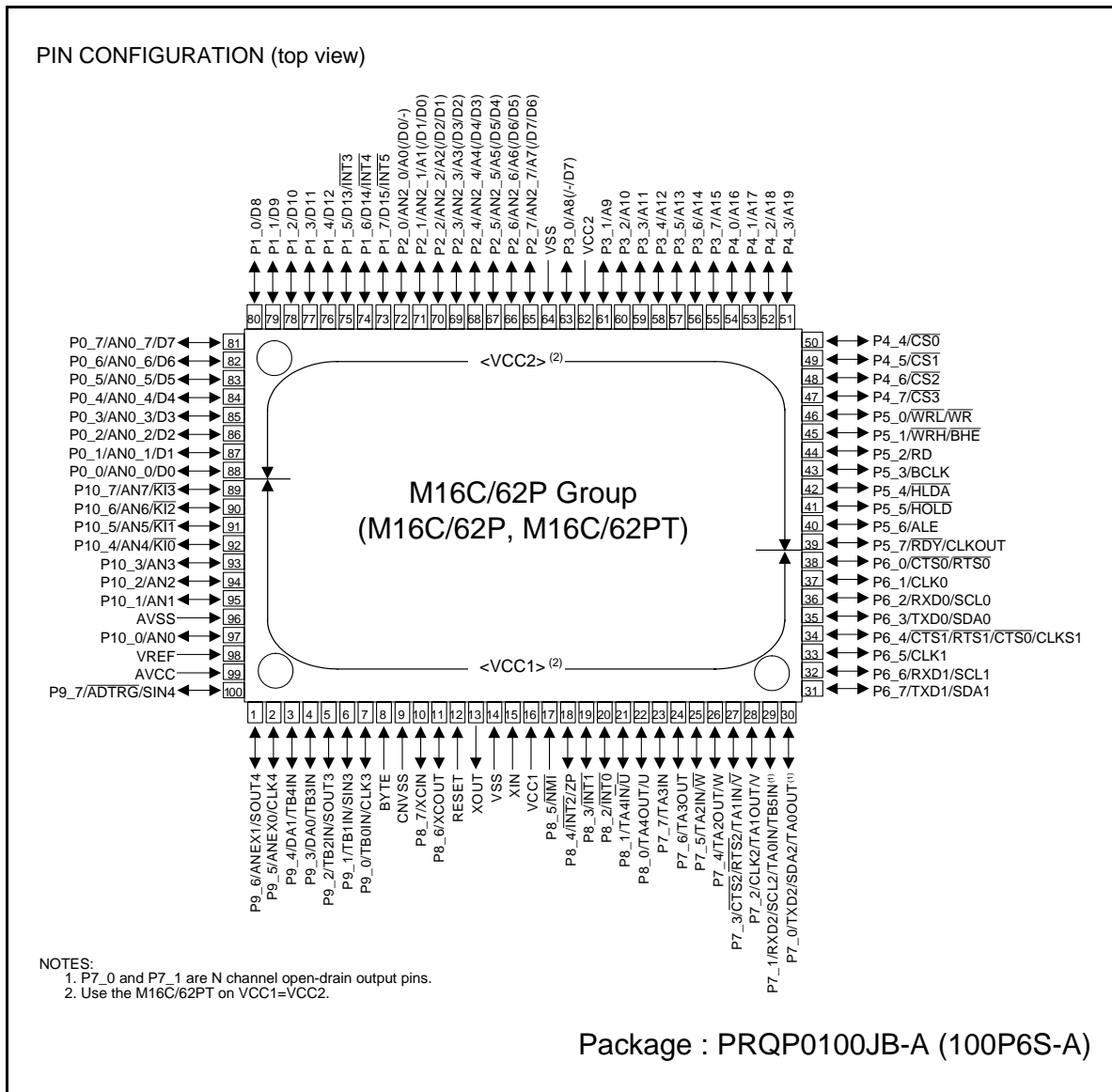
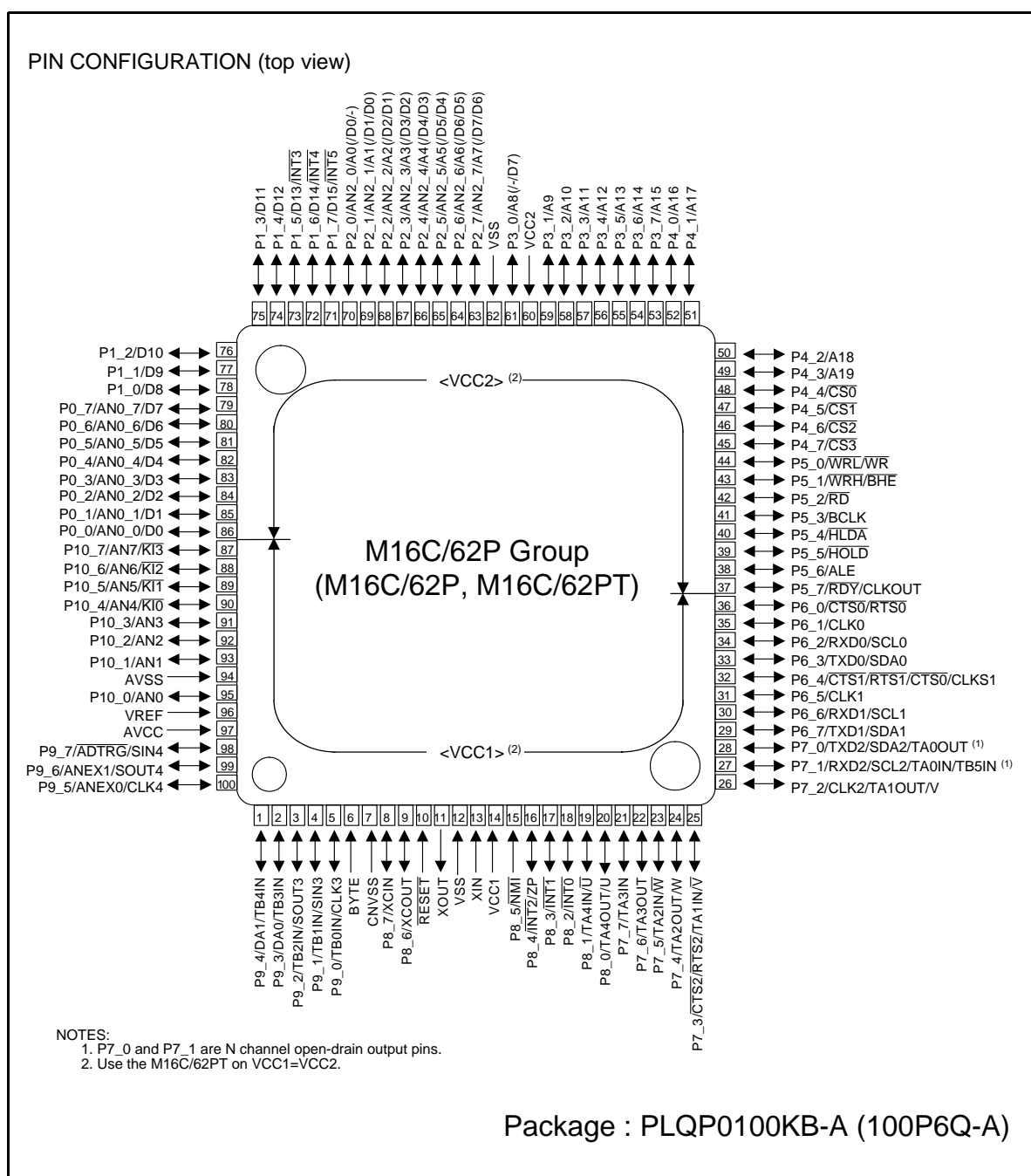


Figure 1.7 Pin Configuration (Top View)



### Figure 1.8 Pin Configuration (Top View)

**Table 1.13 Pin Characteristics for 100-Pin Package (1)**

Pin No.		Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Analog Pin	Bus Control Pin
FP	GP							
1	99		P9_6			SOUT4	ANEX1	
2	100		P9_5			CLK4	ANEX0	
3	1		P9_4		TB4IN		DA1	
4	2		P9_3		TB3IN		DA0	
5	3		P9_2		TB2IN	SOUT3		
6	4		P9_1		TB1IN	SIN3		
7	5		P9_0		TB0IN	CLK3		
8	6	BYTE						
9	7	CNVSS						
10	8	XCIN	P8_7					
11	9	XCOUT	P8_6					
12	10	RESET						
13	11	XOUT						
14	12	VSS						
15	13	XIN						
16	14	VCC1						
17	15		P8_5	NMI				
18	16		P8_4	INT2	ZP			
19	17		P8_3	INT1				
20	18		P8_2	INT0				
21	19		P8_1		TA4IN/ $\bar{U}$			
22	20		P8_0		TA4OUT/U			
23	21		P7_7		TA3IN			
24	22		P7_6		TA3OUT			
25	23		P7_5		TA2IN/ $\bar{W}$			
26	24		P7_4		TA2OUT/W			
27	25		P7_3		TA1IN/ $\bar{V}$	CTS2/RTS2		
28	26		P7_2		TA1OUT/V	CLK2		
29	27		P7_1		TA0IN/TB5IN	RXD2/SCL2		
30	28		P7_0		TA0OUT	TXD2/SDA2		
31	29		P6_7			TXD1/SDA1		
32	30		P6_6			RXD1/SCL1		
33	31		P6_5			CLK1		
34	32		P6_4			CTS1/RTS1/CTS0/CLKS1		
35	33		P6_3			TXD0/SDA0		
36	34		P6_2			RXD0/SCL0		
37	35		P6_1			CLK0		
38	36		P6_0			CTS0/RTS0		
39	37		P5_7					RDY/CLKOUT
40	38		P5_6					ALE
41	39		P5_5					HOLD
42	40		P5_4					HLAD
43	41		P5_3					BCLK
44	42		P5_2					RD
45	43		P5_1					WRH/BHE
46	44		P5_0					WRL/WR
47	45		P4_7					CS3
48	46		P4_6					CS2
49	47		P4_5					CS1
50	48		P4_4					CS0

**Table 1.14 Pin Characteristics for 100-Pin Package (2)**

Pin No.		Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Analog Pin	Bus Control Pin
FP	GP							
51	49		P4_3					A19
52	50		P4_2					A18
53	51		P4_1					A17
54	52		P4_0					A16
55	53		P3_7					A15
56	54		P3_6					A14
57	55		P3_5					A13
58	56		P3_4					A12
59	57		P3_3					A11
60	58		P3_2					A10
61	59		P3_1					A9
62	60	VCC2						
63	61		P3_0					A8(/-D7)
64	62	VSS						
65	63		P2_7				AN2_7	A7(/D7/D6)
66	64		P2_6				AN2_6	A6(/D6/D5)
67	65		P2_5				AN2_5	A5(/D5/D4)
68	66		P2_4				AN2_4	A4(/D4/D3)
69	67		P2_3				AN2_3	A3(/D3/D2)
70	68		P2_2				AN2_2	A2(/D2/D1)
71	69		P2_1				AN2_1	A1(/D1/D0)
72	70		P2_0				AN2_0	A0(/D0/-)
73	71		P1_7	INT5				D15
74	72		P1_6	INT4				D14
75	73		P1_5	INT3				D13
76	74		P1_4					D12
77	75		P1_3					D11
78	76		P1_2					D10
79	77		P1_1					D9
80	78		P1_0					D8
81	79		P0_7				AN0_7	D7
82	80		P0_6				AN0_6	D6
83	81		P0_5				AN0_5	D5
84	82		P0_4				AN0_4	D4
85	83		P0_3				AN0_3	D3
86	84		P0_2				AN0_2	D2
87	85		P0_1				AN0_1	D1
88	86		P0_0				AN0_0	D0
89	87		P10_7	KI3			AN7	
90	88		P10_6	KI2			AN6	
91	89		P10_5	KI1			AN5	
92	90		P10_4	KI0			AN4	
93	91		P10_3				AN3	
94	92		P10_2				AN2	
95	93		P10_1				AN1	
96	94	AVSS						
97	95		P10_0				AN0	
98	96	VREF						
99	97	AVCC						
100	98		P9_7			SIN4	ADTRG	

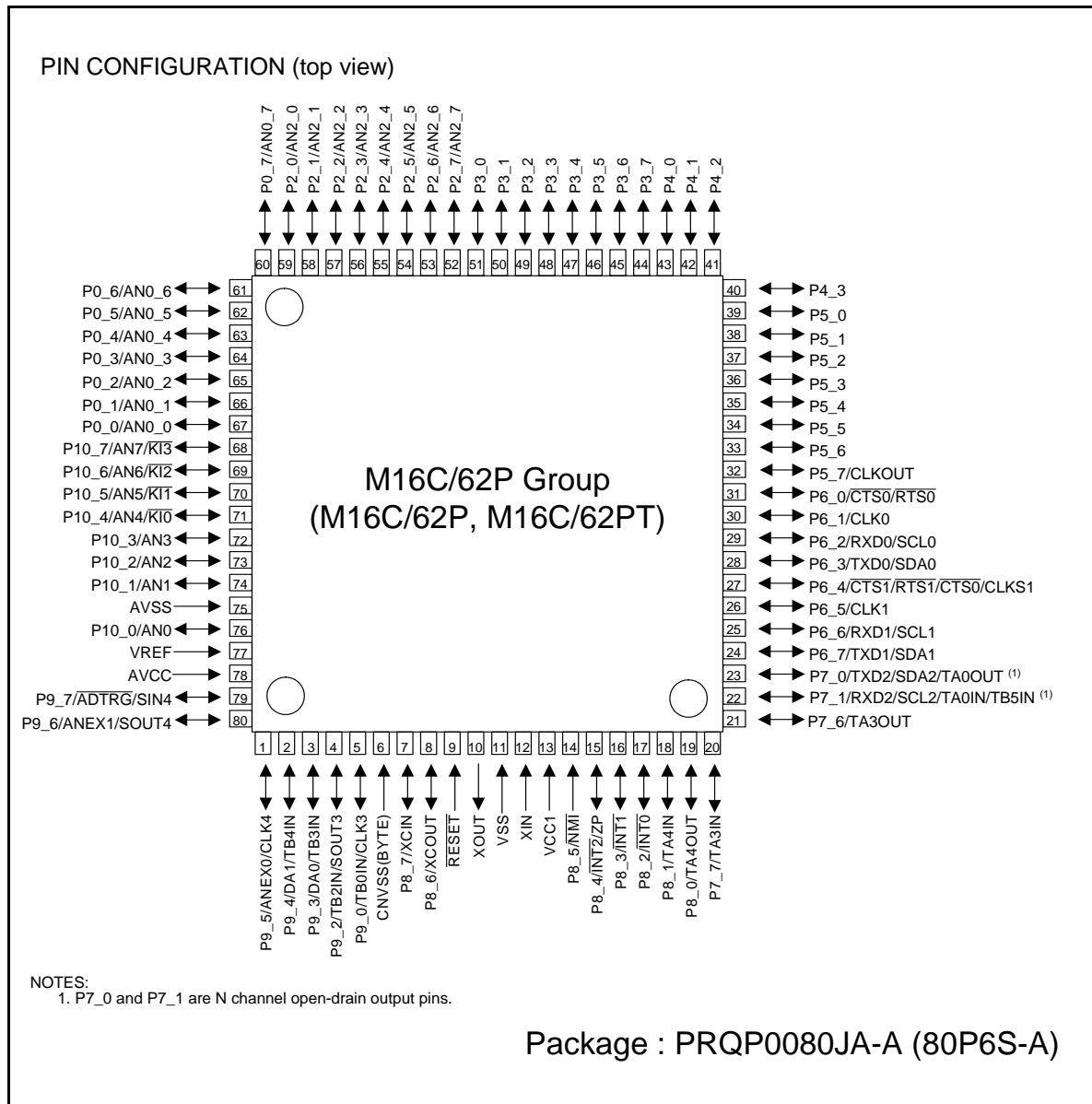


Figure 1.9 Pin Configuration (Top View)

**Table 1.15 Pin Characteristics for 80-Pin Package (1)**

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Analog Pin	Bus Control Pin
1		P9_5			CLK4	ANEX0	
2		P9_4		TB4IN		DA1	
3		P9_3		TB3IN		DA0	
4		P9_2		TB2IN	SOUT3		
5		P9_0		TB0IN	CLK3		
6	CNVSS (BYTE)						
7	XCIN	P8_7					
8	XCOU $\overline{T}$	P8_6					
9	RESE $\overline{T}$						
10	XOUT						
11	VSS						
12	XIN						
13	VCC1						
14		P8_5	NMI				
15		P8_4	INT2	ZP			
16		P8_3	INT1				
17		P8_2	INT0				
18		P8_1		TA4IN			
19		P8_0		TA4OUT			
20		P7_7		TA3IN			
21		P7_6		TA3OUT			
22		P7_1		TA0IN/TB5IN	RXD2/SCL2		
23		P7_0		TA0OUT	TXD2/SDA2		
24		P6_7			TXD1/SDA1		
25		P6_6			RXD1/SCL1		
26		P6_5			CLK1		
27		P6_4			CTS1/RTS1/CTS0/CLKS1		
28		P6_3			TXD0/SDA0		
29		P6_2			RXD0/SCL0		
30		P6_1			CLK0		
31		P6_0			CTS0/RTS0		
32		P5_7					CLKOUT
33		P5_6					
34		P5_5					
35		P5_4					
36		P5_3					
37		P5_2					
38		P5_1					
39		P5_0					
40		P4_3					
41		P4_2					
42		P4_1					
43		P4_0					
44		P3_7					
45		P3_6					
46		P3_5					
47		P3_4					
48		P3_3					
49		P3_2					
50		P3_1					

**Table 1.16 Pin Characteristics for 80-Pin Package (2)**

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART Pin	Analog Pin	Bus Control Pin
51		P3_0					
52		P2_7				AN2_7	
53		P2_6				AN2_6	
54		P2_5				AN2_5	
55		P2_4				AN2_4	
56		P2_3				AN2_3	
57		P2_2				AN2_2	
58		P2_1				AN2_1	
59		P2_0				AN2_0	
60		P0_7				AN0_7	
61		P0_6				AN0_6	
62		P0_5				AN0_5	
63		P0_4				AN0_4	
64		P0_3				AN0_3	
65		P0_2				AN0_2	
66		P0_1				AN0_1	
67		P0_0				AN0_0	
68		P10_7	$\overline{KI3}$			AN7	
69		P10_6	$\overline{KI2}$			AN6	
70		P10_5	$\overline{KI1}$			AN5	
71		P10_4	$\overline{KI0}$			AN4	
72		P10_3				AN3	
73		P10_2				AN2	
74		P10_1				AN1	
75	AVSS						
76		P10_0				AN0	
77	VREF						
78	AVCC						
79		P9_7			SIN4	$\overline{ADTRG}$	
80		P9_6			SOUT4	ANEX1	

## 1.6 Pin Description

**Table 1.17 Pin Description (100-pin and 128-pin Version) (1)**

Signal Name	Pin Name	I/O Type	Power Supply <sup>(3)</sup>	Description
Power supply input	VCC1,VCC2 VSS	I	—	Apply 2.7 to 5.5 V to the VCC1 and VCC2 pins and 0 V to the VSS pin. The VCC apply condition is that $VCC1 \geq VCC2$ . (1, 2)
Analog power supply input	AVCC AVSS	I	VCC1	Applies the power supply for the A/D converter. Connect the AVCC pin to VCC1. Connect the AVSS pin to VSS.
Reset input	RESET	I	VCC1	The microcomputer is in a reset state when applying "L" to the this pin.
CNVSS	CNVSS	I	VCC1	Switches processor mode. Connect this pin to VSS to when after a reset to start up in single-chip mode. Connect this pin to VCC1 to start up in microprocessor mode.
External data bus width select input	BYTE	I	VCC1	Switches the data bus in external memory space. The data bus is 16 bits long when the this pin is held "L" and 8 bits long when the this pin is held "H". Set it to either one. Connect this pin to VSS when an single-chip mode.
Bus control pins <sup>(4)</sup>	D0 to D7	I/O	VCC2	Inputs and outputs data (D0 to D7) when these pins are set as the separate bus.
	D8 to D15	I/O	VCC2	Inputs and outputs data (D8 to D15) when external 16-bit data bus is set as the separate bus.
	A0 to A19	O	VCC2	Output address bits (A0 to A19).
	A0/D0 to A7/D7	I/O	VCC2	Input and output data (D0 to D7) and output address bits (A0 to A7) by timesharing when external 8-bit data bus are set as the multiplexed bus.
	A1/D0 to A8/D7	I/O	VCC2	Input and output data (D0 to D7) and output address bits (A1 to A8) by timesharing when external 16-bit data bus are set as the multiplexed bus.
	CS0 to CS3	O	VCC2	Output CS0 to CS3 signals. CS0 to CS3 are chip-select signals to specify an external space.
	WRL/WR WRH/BHE RD	O	VCC2	Output WRL, WRH, (WR, BHE), RD signals. WRL and WRH or BHE and WR can be switched by program. • WRL, WRH and RD are selected The WRL signal becomes "L" by writing data to an even address in an external memory space. The WRH signal becomes "L" by writing data to an odd address in an external memory space. The RD pin signal becomes "L" by reading data in an external memory space. • WR, BHE and RD are selected The WR signal becomes "L" by writing data in an external memory space. The RD signal becomes "L" by reading data in an external memory space. The BHE signal becomes "L" by accessing an odd address. Select WR, BHE and RD for an external 8-bit data bus.
	ALE	O	VCC2	ALE is a signal to latch the address.
	HOLD	I	VCC2	While the HOLD pin is held "L", the microcomputer is placed in a hold state.
	HLDA	O	VCC2	In a hold state, HLDA outputs a "L" signal.
	RDY	I	VCC2	While applying a "L" signal to the RDY pin, the microcomputer is placed in a wait state.

I : Input O : Output I/O : Input and output

Power Supply : Power supplies which relate to the external bus pins are separated as VCC2, thus they can be interfaced using the different voltage as VCC1.

**NOTES:**

1. In this manual, hereafter, VCC refers to VCC1 unless otherwise noted.
2. In M16C/62PT, apply 4.0 to 5.5 V to the VCC1 and VCC2 pins. Also the apply condition is that  $VCC1 = VCC2$ .
3. When use  $VCC1 > VCC2$ , contacts due to some points or restrictions to be checked.
4. Bus control pins in M16C/62PT cannot be used.



**Table 1.18 Pin Description (100-pin and 128-pin Version) (2)**

Signal Name	Pin Name	I/O Type	Power Supply <sup>(1)</sup>	Description
Main clock input	XIN	I	VCC1	I/O pins for the main clock generation circuit. Connect a ceramic resonator or crystal oscillator between XIN and XOUT <sup>(3)</sup> . To use the external clock, input the clock from XIN and leave XOUT open.
Main clock output	XOUT	O	VCC1	
Sub clock input	XCIN	I	VCC1	I/O pins for a sub clock oscillation circuit. Connect a crystal oscillator between XCIN and XCOUT <sup>(3)</sup> . To use the external clock, input the clock from XCIN and leave XCOUT open.
Sub clock output	XCOUT	O	VCC1	
BCLK output <sup>(2)</sup>	BCLK	O	VCC2	Outputs the BCLK signal.
Clock output	CLKOUT	O	VCC2	The clock of the same cycle as fC, f8, or f32 is outputted.
$\overline{\text{INT}}$ interrupt input	$\overline{\text{INT0}}$ to $\overline{\text{INT2}}$	I	VCC1	Input pins for the $\overline{\text{INT}}$ interrupt.
	$\overline{\text{NT3}}$ to $\overline{\text{INT5}}$	I	VCC2	
$\overline{\text{NMI}}$ interrupt input	$\overline{\text{NMI}}$	I	VCC1	Input pin for the $\overline{\text{NMI}}$ interrupt. Pin states can be read by the P8_5 bit in the P8 register.
Key input interrupt input	$\overline{\text{KI0}}$ to $\overline{\text{KI3}}$	I	VCC1	Input pins for the key input interrupt.
Timer A	TA0OUT to TA4OUT	I/O	VCC1	These are timer A0 to timer A4 I/O pins. (however, output of TA0OUT for the N-channel open drain output.)
	TA0IN to TA4IN	I	VCC1	These are timer A0 to timer A4 input pins.
	ZP	I	VCC1	Input pin for the Z-phase.
Timer B	TB0IN to TB5IN	I	VCC1	These are timer B0 to timer B5 input pins.
Three-phase motor control output	U, $\overline{\text{U}}$ , V, $\overline{\text{V}}$ , W, $\overline{\text{W}}$	O	VCC1	These are Three-phase motor control output pins.
Serial interface	$\overline{\text{CTS0}}$ to $\overline{\text{CTS2}}$	I	VCC1	These are send control input pins.
	$\overline{\text{RTS0}}$ to $\overline{\text{RTS2}}$	O	VCC1	These are receive control output pins.
	CLK0 to CLK4	I/O	VCC1	These are transfer clock I/O pins.
	RXD0 to RXD2	I	VCC1	These are serial data input pins.
	SIN3, SIN4	I	VCC1	These are serial data input pins.
	TXD0 to TXD2	O	VCC1	These are serial data output pins. (however, output of TXD2 for the N-channel open drain output.)
	SOUT3, SOUT4	O	VCC1	These are serial data output pins.
	CLKS1	O	VCC1	This is output pin for transfer clock output from multiple pins function.
I <sup>2</sup> C mode	SDA0 to SDA2	I/O	VCC1	These are serial data I/O pins. (however, output of SDA2 for the N-channel open drain output.)
	SCL0 to SCL2	I/O	VCC1	These are transfer clock I/O pins. (however, output of SCL2 for the N-channel open drain output.)

I : Input O : Output I/O : Input and output

## NOTES:

1. When use VCC1 > VCC2, contacts due to some points or restrictions to be checked.
2. This pin function in M16C/62PT cannot be used.
3. Ask the oscillator maker the oscillation characteristic.

**Table 1.19 Pin Description (100-pin and 128-pin Version) (3)**

Signal Name	Pin Name	I/O Type	Power Supply <sup>(1)</sup>	Description
Reference voltage input	VREF	I	VCC1	Applies the reference voltage for the A/D converter and D/A converter.
A/D converter	AN0 to AN7, AN0_0 to AN0_7, AN2_0 to AN2_7	I	VCC1	Analog input pins for the A/D converter.
	ADTRG	I	VCC1	This is an A/D trigger input pin.
	ANEX0	I/O	VCC1	This is the extended analog input pin for the A/D converter, and is the output in external op-amp connection mode.
	ANEX1	I	VCC1	This is the extended analog input pin for the A/D converter.
D/A converter	DA0, DA1	O	VCC1	This is the output pin for the D/A converter.
I/O port	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7 <sup>(2)</sup> , P13_0 to P13_7 <sup>(2)</sup>	I/O	VCC2	8-bit I/O ports in CMOS, having a direction register to select an input or output. Each pin is set as an input port or output port. An input port can be set for a pull-up or for no pull-up in 4-bit unit by program.
	P6_0 to P6_7, P7_0 to P7_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7 <sup>(2)</sup>	I/O	VCC1	8-bit I/O ports having equivalent functions to P0. (however, output of P7_0 and P7_1 for the N-channel open drain output.)
	P8_0 to P8_4, P8_6, P8_7, P14_0, P14_1 <sup>(2)</sup>	I/O	VCC1	I/O ports having equivalent functions to P0.
Input port	P8_5	I	VCC1	Input pin for the $\overline{\text{NMI}}$ interrupt. Pin states can be read by the P8_5 bit in the P8 register.

I : Input O : Output I/O : Input and output

## NOTES:

1. When use VCC1 > VCC2, contacts due to some points or restrictions to be checked.
2. Ports P11 to P14 in M16C/62P (100-pin version) and M16C/62PT (100-pin version) cannot be used.

**Table 1.20 Pin Description (80-pin Version) (1) (1)**

Signal Name	Pin Name	I/O Type	Power Supply	Description
Power supply input	VCC1, VSS	I	–	Apply 2.7 to 5.5 V to the VCC1 pin and 0 V to the VSS pin. (1, 2)
Analog power supply input	AVCC AVSS	I	VCC1	Applies the power supply for the A/D converter. Connect the AVCC pin to VCC1. Connect the AVSS pin to VSS.
Reset input	RESET	I	VCC1	The microcomputer is in a reset state when applying “L” to the this pin.
CNVSS	CNVSS (BYTE)	I	VCC1	Switches processor mode. Connect this pin to VSS to when after a reset to start up in single-chip mode. Connect this pin to VCC1 to start up in microprocessor mode. As for the BYTE pin of the 80-pin versions, pull-up processing is performed within the microcomputer.
Main clock input	XIN	I	VCC1	I/O pins for the main clock generation circuit. Connect a ceramic resonator or crystal oscillator between XIN and XOUT (3). To use the external clock, input the clock from XIN and leave XOUT open.
Main clock output	XOUT	O	VCC1	
Sub clock input	XCIN	I	VCC1	I/O pins for a sub clock oscillation circuit. Connect a crystal oscillator between XCIN and XCOU (3). To use the external clock, input the clock from XCIN and leave XCOU open.
Sub clock output	XCOU	O	VCC1	
Clock output	CLKOUT	O	VCC2	The clock of the same cycle as fC, f8, or f32 is outputted.
INT interrupt input	INT0 to INT2	I	VCC1	Input pins for the INT interrupt.
NMI interrupt input	NMI	I	VCC1	Input pin for the NMI interrupt.
Key input interrupt input	KI0 to KI3	I	VCC1	Input pins for the key input interrupt.
Timer A	TA0OUT, TA3OUT, TA4OUT	I/O	VCC1	These are Timer A0, Timer A3 and Timer A4 I/O pins. (however, output of TA0OUT for the N-channel open drain output.)
	TA0IN, TA3IN, TA4IN	I	VCC1	These are Timer A0, Timer A3 and Timer A4 input pins.
	ZP	I	VCC1	Input pin for the Z-phase.
Timer B	TB0IN, TB2IN to TB5IN	I	VCC1	These are Timer B0, Timer B2 to Timer B5 input pins.
Serial interface	CTS0 to CTS1	I	VCC1	These are send control input pins.
	RTS0 to RTS1	O	VCC1	These are receive control output pins.
	CLK0, CLK1, CLK3, CLK4	I/O	VCC1	These are transfer clock I/O pins.
	RXD0 to RXD2	I	VCC1	These are serial data input pins.
	SIN4	I	VCC1	This is serial data input pin.
	TXD0 to TXD2	O	VCC1	These are serial data output pins. (however, output of TXD2 for the N-channel open drain output.)
	SOUT3, SOUT4	O	VCC1	These are serial data output pins.
	CLKS1	O	VCC1	This is output pin for transfer clock output from multiple pins function.
I <sup>2</sup> C mode	SDA0 to SDA2	I/O	VCC1	These are serial data I/O pins. (however, output of SDA2 for the N-channel open drain output.)
	SCL0 to SCL2	I/O	VCC1	These are transfer clock I/O pins. (however, output of SCL2 for the N-channel open drain output.)

I : Input O : Output I/O : Input and output

## NOTES:

1. In this manual, hereafter, VCC refers to VCC1 unless otherwise noted.
2. In M16C/62PT, apply 4.0 to 5.5 V to the VCC1 pin.
3. Ask the oscillator maker the oscillation characteristic.

**Table 1.21 Pin Description (80-pin Version) (2)**

Signal Name	Pin Name	I/O Type	Power Supply <sup>(1)</sup>	Description
Reference voltage input	VREF	I	VCC1	Applies the reference voltage for the A/D converter and D/A converter.
A/D converter	AN0 to AN7, AN0_0 to AN0_7, AN2_0 to AN2_7	I	VCC1	Analog input pins for the A/D converter.
	ADTRG	I	VCC1	This is an A/D trigger input pin.
	ANEX0	I/O	VCC1	This is the extended analog input pin for the A/D converter, and is the output in external op-amp connection mode.
	ANEX1	I	VCC1	This is the extended analog input pin for the A/D converter.
D/A converter	DA0, DA1	O	VCC1	This is the output pin for the D/A converter.
I/O port <sup>(1)</sup>	P0_0 to P0_7, P2_0 to P2_7, P3_0 to P3_7, P5_0 to P5_7, P6_0 to P6_7, P10_0 to P10_7	I/O	VCC1	8-bit I/O ports in CMOS, having a direction register to select an input or output. Each pin is set as an input port or output port. An input port can be set for a pull-up or for no pull-up in 4-bit unit by program.
	P8_0 to P8_4, P8_6, P8_7, P9_0, P9_2 to P9_7	I/O	VCC1	I/O ports having equivalent functions to P0.
	P4_0 to P4_3, P7_0, P7_1, P7_6, P7_7	I/O	VCC1	I/O ports having equivalent functions to P0. (however, output of P7_0 and P7_1 for the N-channel open drain output.)
Input port	P8_5	I	VCC1	Input pin for the $\overline{\text{NMI}}$ interrupt. Pin states can be read by the P8_5 bit in the P8 register.

I : Input O : Output I/O : Input and output

**NOTES:**

1. There is no external connections for port P1, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version. Set the direction bits in these ports to "1" (output mode), and set the output data to "0" ("L") using the program.

## 2. Central Processing Unit (CPU)

Figure 2.1 shows the CPU registers. The CPU has 13 registers. Of these, R0, R1, R2, R3, A0, A1 and FB comprise a register bank. There are two register banks.

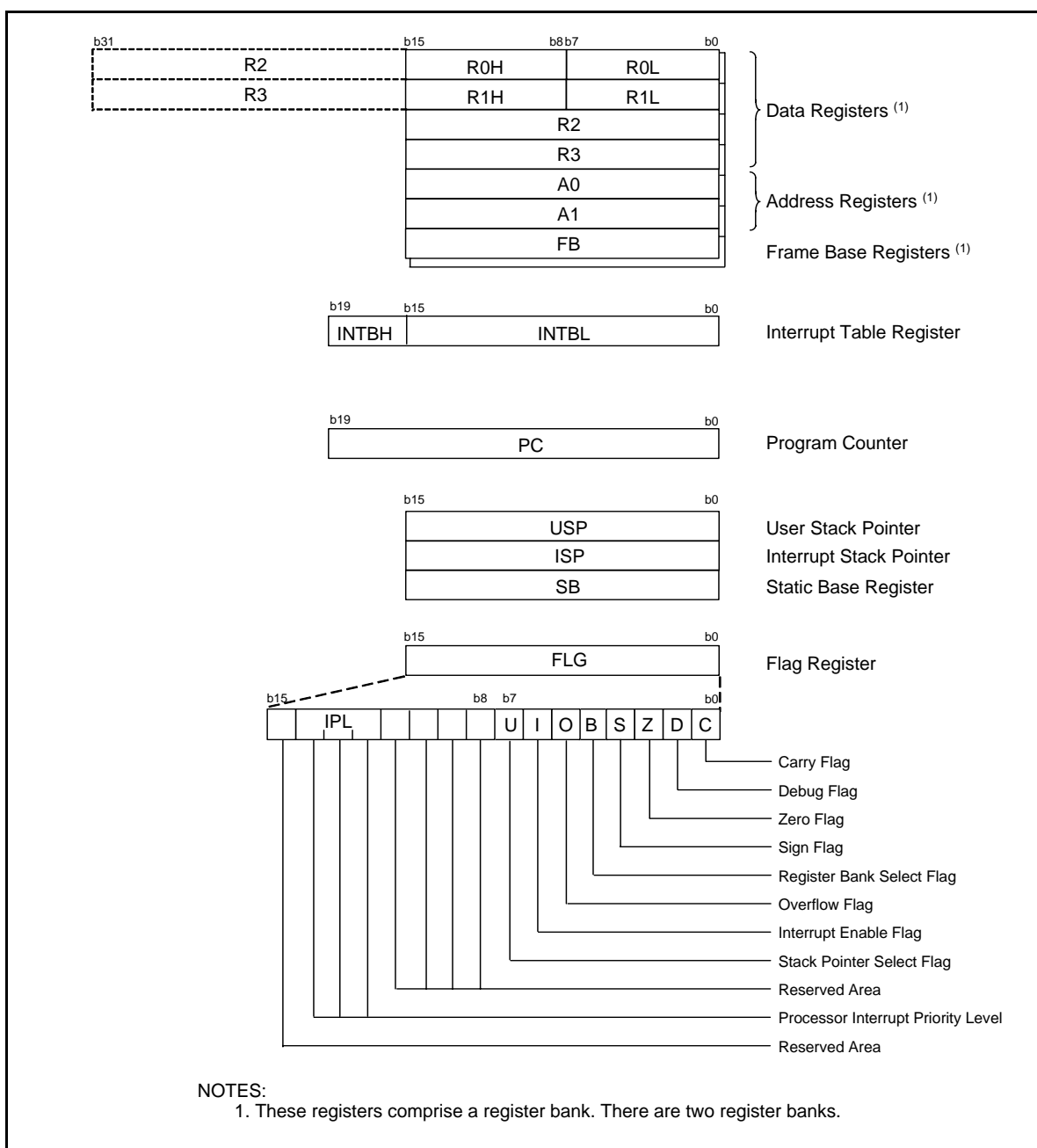


Figure 2.1 Central Processing Unit Register

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

The R0 register consists of 16 bits, and is used mainly for transfers and arithmetic/logic operations. R1 to R3 are the same as R0.

The R0 register can be separated between high (R0H) and low (R0L) for use as two 8-bit data registers.

R1H and R1L are the same as R0H and R0L. Conversely, R2 and R0 can be combined for use as a 32-bit data register (R2R0). R3R1 is the same as R2R0.

## 2.2 Address Registers (A0 and A1)

The register A0 consists of 16 bits, and is used for address register indirect addressing and address register relative addressing. They also are used for transfers and logic/logic operations. A1 is the same as A0.

In some instructions, registers A1 and A0 can be combined for use as a 32-bit address register (A1A0).

## 2.3 Frame Base Register (FB)

FB is configured with 16 bits, and is used for FB relative addressing.

## 2.4 Interrupt Table Register (INTB)

INTB is configured with 20 bits, indicating the start address of an interrupt vector table.

## 2.5 Program Counter (PC)

PC is configured with 20 bits, indicating the address of an instruction to be executed.

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

Stack pointer (SP) comes in two types: USP and ISP, each configured with 16 bits.

Your desired type of stack pointer (USP or ISP) can be selected by the U flag of FLG.

## 2.7 Static Base Register (SB)

SB is configured with 16 bits, and is used for SB relative addressing.

## 2.8 Flag Register (FLG)

FLG consists of 11 bits, indicating the CPU status.

### 2.8.1 Carry Flag (C Flag)

This flag retains a carry, borrow, or shift-out bit that has occurred in the arithmetic/logic unit.

### 2.8.2 Debug Flag (D Flag)

The D flag is used exclusively for debugging purpose. During normal use, it must be set to "0".

### 2.8.3 Zero Flag (Z Flag)

This flag is set to "1" when an arithmetic operation resulted in 0; otherwise, it is "0".

### 2.8.4 Sign Flag (S Flag)

This flag is set to "1" when an arithmetic operation resulted in a negative value; otherwise, it is "0".

### 2.8.5 Register Bank Select Flag (B Flag)

Register bank 0 is selected when this flag is "0"; register bank 1 is selected when this flag is "1".

### 2.8.6 Overflow Flag (O Flag)

This flag is set to "1" when the operation resulted in an overflow; otherwise, it is "0".

### 2.8.7 Interrupt Enable Flag (I Flag)

This flag enables a maskable interrupt.

Maskable interrupts are disabled when the I flag is "0", and are enabled when the I flag is "1". The I flag is cleared to "0" when the 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 is cleared to “0” when a hardware interrupt request is accepted or an INT instruction for software interrupt Nos. 0 to 31 is executed.

### 2.8.9 Processor Interrupt Priority Level (IPL)

IPL is configured with three bits, for specification of up to eight processor interrupt priority levels from level 0 to level 7.

If a requested interrupt has priority greater than IPL, the interrupt is enabled.

### 2.8.10 Reserved Area

When write to this bit, write “0”. When read, its content is indeterminate.

### 3. Memory

Figure 3.1 is a Memory Map of the M16C/62P group. The address space extends the 1M bytes from address 00000h to FFFFFh.

The internal ROM is allocated in a lower address direction beginning with address FFFFFh. For example, a 64-Kbyte internal ROM is allocated to the addresses from F0000h to FFFFFh.

As for the flash memory version, 4-Kbyte space (block A) exists in 0F000h to 0FFFFh. 4-Kbyte space is mainly for storing data. In addition to storing data, 4-Kbyte space also can store programs.

The fixed interrupt vector table is allocated to the addresses from FFFDCh to FFFFFh. Therefore, store the start address of each interrupt routine here.

The internal RAM is allocated in an upper address direction beginning with address 00400h. For example, a 10-Kbyte internal RAM is allocated to the addresses from 00400h to 02BFFh. In addition to storing data, the internal RAM also stores the stack used when calling subroutines and when interrupts are generated.

The SRF is allocated to the addresses from 00000h to 003FFh. Peripheral function control registers are located here. Of the SFR, any area which has no functions allocated is reserved for future use and cannot be used by users.

The special page vector table is allocated to the addresses from FFE00h to FFFDBh. This vector is used by the JMPS or JSRS instruction. For details, refer to the **M16C/60 and M16C/20 Series Software Manual**.

In memory expansion and microprocessor modes, some areas are reserved for future use and cannot be used by users. Use M16C/62P (80-pin version) and M16C/62PT in single-chip mode. The memory expansion and microprocessor modes cannot be used

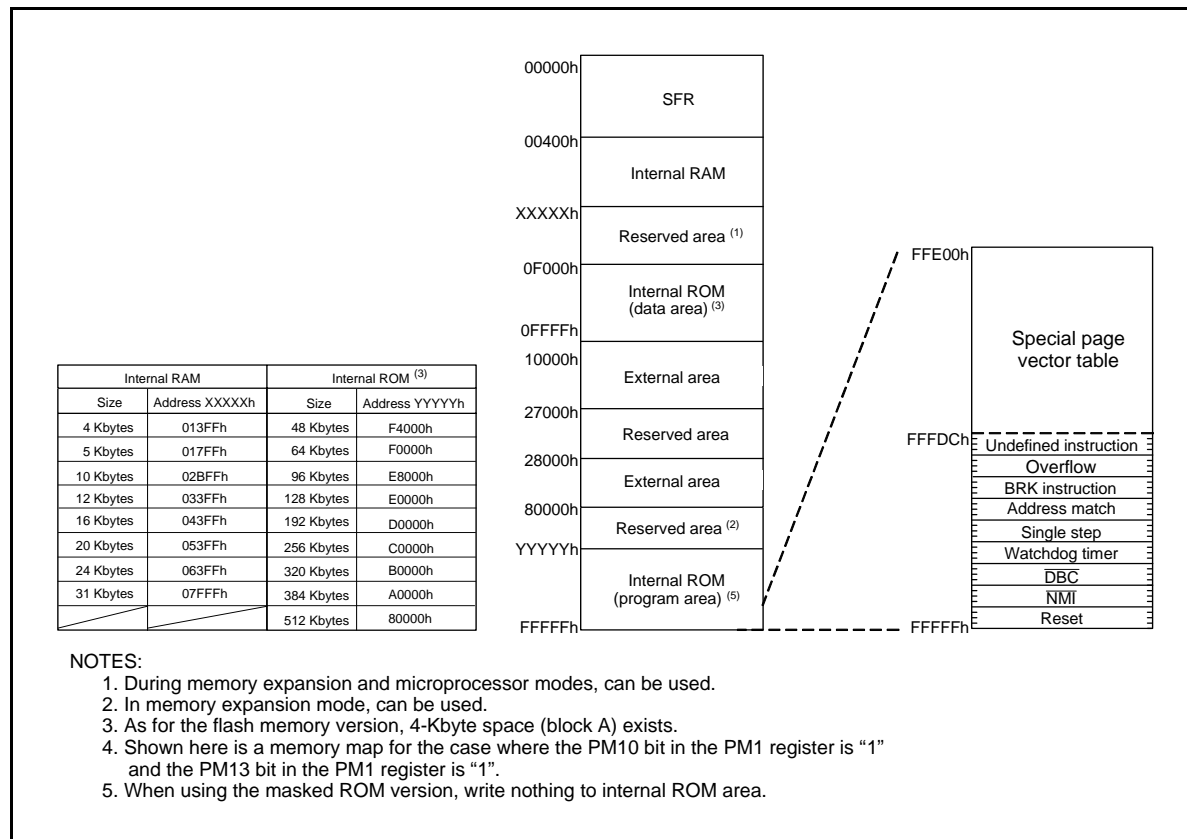


Figure 3.1 Memory Map



## 4. Special Function Register (SFR)

SFR(Special Function Register) is the control register of peripheral functions. Tables 4.1 to 4.6 list the SFR information.

**Table 4.1 SFR Information (1) (1)**

Address	Register	Symbol	After Reset
0000h			
0001h			
0002h			
0003h			
0004h	Processor Mode Register 0 <sup>(2)</sup>	PM0	00000000b(CNVSS pin is "L") 00000011b(CNVSS pin is "H")
0005h	Processor Mode Register 1	PM1	00001000b
0006h	System Clock Control Register 0	CM0	01001000b
0007h	System Clock Control Register 1	CM1	00100000b
0008h	Chip Select Control Register <sup>(6)</sup>	CSR	00000001b
0009h	Address Match Interrupt Enable Register	AIER	XXXXXXXX00b
000Ah	Protect Register	PRCR	XX000000b
000Bh	Data Bank Register <sup>(6)</sup>	DBR	00h
000Ch	Oscillation Stop Detection Register <sup>(3)</sup>	CM2	0X000000b
000Dh			
000Eh	Watchdog Timer Start Register	WDTs	XXh
000Fh	Watchdog Timer Control Register	WDC	00XXXXXXb <sup>(4)</sup>
0010h	Address Match Interrupt Register 0	RMAD0	00h
0011h			00h
0012h			X0h
0013h			
0014h	Address Match Interrupt Register 1	RMAD1	00h
0015h			00h
0016h			X0h
0017h			
0018h			
0019h	Voltage Detection Register 1 <sup>(5, 6)</sup>	VCR1	00001000b
001Ah	Voltage Detection Register 2 <sup>(5, 6)</sup>	VCR2	00h
001Bh	Chip Select Expansion Control Register <sup>(6)</sup>	CSE	00h
001Ch	PLL Control Register 0	PLC0	0001X010b
001Dh			
001Eh	Processor Mode Register 2	PM2	XXX00000b
001Fh	Low Voltage Detection Interrupt Register <sup>(6)</sup>	D4INT	00h
0020h	DMA0 Source Pointer	SAR0	XXh
0021h			XXh
0022h			XXh
0023h			
0024h	DMA0 Destination Pointer	DAR0	XXh
0025h			XXh
0026h			XXh
0027h			
0028h	DMA0 Transfer Counter	TCR0	XXh
0029h			XXh
002Ah			
002Bh			
002Ch	DMA0 Control Register	DM0CON	00000X00b
002Dh			
002Eh			
002Fh			
0030h	DMA1 Source Pointer	SAR1	XXh
0031h			XXh
0032h			XXh
0033h			
0034h	DMA1 Destination Pointer	DAR1	XXh
0035h			XXh
0036h			XXh
0037h			
0038h	DMA1 Transfer Counter	TCR1	XXh
0039h			XXh
003Ah			
003Bh			
003Ch	DMA1 Control Register	DM1CON	00000X00b
003Dh			
003Eh			
003Fh			

**NOTES:**

- The blank areas are reserved and cannot be accessed by users.
- The PM00 and PM01 bits do not change at software reset, watchdog timer reset and oscillation stop detection reset.
- The CM20, CM21, and CM27 bits do not change at oscillation stop detection reset.
- The WDC5 bit is "0" (cold start) immediately after power-on. It can only be set to "1" in a program.
- This register does not change at software reset, watchdog timer reset and oscillation stop detection reset.
- This register in M16C/62PT cannot be used.

X : Nothing is mapped to this bit

**Table 4.2 SFR Information (2) <sup>(1)</sup>**

Address	Register	Symbol	After Reset
0040h			
0041h			
0042h			
0043h			
0044h	INT3 Interrupt Control Register	INT3IC	XX00X000b
0045h	Timer B5 Interrupt Control Register	TB5IC	XXXXX000b
0046h	Timer B4 Interrupt Control Register, UART1 BUS Collision Detection Interrupt Control Register	TB4IC, U1BCNIC	XXXXX000b
0047h	Timer B3 Interrupt Control Register, UART0 BUS Collision Detection Interrupt Control Register	TB3IC, U0BCNIC	XXXXX000b
0048h	SI/O4 Interrupt Control Register, INT5 Interrupt Control Register	S4IC, INT5IC	XX00X000b
0049h	SI/O3 Interrupt Control Register, INT4 Interrupt Control Register	S3IC, INT4IC	XX00X000b
004Ah	UART2 Bus Collision Detection Interrupt Control Register	BCNIC	XXXXX000b
004Bh	DMA0 Interrupt Control Register	DM0IC	XXXXX000b
004Ch	DMA1 Interrupt Control Register	DM1IC	XXXXX000b
004Dh	Key Input Interrupt Control Register	KUPIC	XXXXX000b
004Eh	A/D Conversion Interrupt Control Register	ADIC	XXXXX000b
004Fh	UART2 Transmit Interrupt Control Register	S2TIC	XXXXX000b
0050h	UART2 Receive Interrupt Control Register	S2RIC	XXXXX000b
0051h	UART0 Transmit Interrupt Control Register	S0TIC	XXXXX000b
0052h	UART0 Receive Interrupt Control Register	S0RIC	XXXXX000b
0053h	UART1 Transmit Interrupt Control Register	S1TIC	XXXXX000b
0054h	UART1 Receive Interrupt Control Register	S1RIC	XXXXX000b
0055h	Timer A0 Interrupt Control Register	TA0IC	XXXXX000b
0056h	Timer A1 Interrupt Control Register	TA1IC	XXXXX000b
0057h	Timer A2 Interrupt Control Register	TA2IC	XXXXX000b
0058h	Timer A3 Interrupt Control Register	TA3IC	XXXXX000b
0059h	Timer A4 Interrupt Control Register	TA4IC	XXXXX000b
005Ah	Timer B0 Interrupt Control Register	TB0IC	XXXXX000b
005Bh	Timer B1 Interrupt Control Register	TB1IC	XXXXX000b
005Ch	Timer B2 Interrupt Control Register	TB2IC	XXXXX000b
005Dh	INT0 Interrupt Control Register	INT0IC	XX00X000b
005Eh	INT1 Interrupt Control Register	INT1IC	XX00X000b
005Fh	INT2 Interrupt Control Register	INT2IC	XX00X000b
0060h			
0061h			
0062h			
0063h			
0064h			
0065h			
0066h			
0067h			
0068h			
0069h			
006Ah			
006Bh			
006Ch			
006Dh			
006Eh			
006Fh			
0070h			
0071h			
0072h			
0073h			
0074h			
0075h			
0076h			
0077h			
0078h			
0079h			
007Ah			
007Bh			
007Ch			
007Dh			
007Eh			
007Fh			

**NOTES:**

1. The blank areas are reserved and cannot be accessed by users.

X : Nothing is mapped to this bit

**Table 4.3 SFR Information (3) <sup>(1)</sup>**

Address	Register	Symbol	After Reset
0080h			
0081h			
0082h			
0083h			
0084h			
0085h			
0086h			
0087h to 01AFh			
01B0h			
01B1h			
01B2h			
01B3h			
01B4h	Flash Identification Register <sup>(2)</sup>	FIDR	XXXXXX00b
01B5h	Flash Memory Control Register 1 <sup>(2)</sup>	FMR1	0X00XX0Xb
01B6h			
01B7h	Flash Memory Control Register 0 <sup>(2)</sup>	FMR0	00000001b
01B8h	Address Match Interrupt Register 2	RMAD2	00h
01B9h			00h
01BAh			XXh
01BBh	Address Match Interrupt Enable Register 2	AIER2	XXXXXX00b
01BCh	Address Match Interrupt Register 3	RMAD3	00h
01BDh			00h
01BEh			XXh
01C0h to 024Fh			
0250h			
0251h			
0252h			
0253h			
0254h			
0255h			
0256h			
0257h			
0258h			
0259h			
025Ah			
025Bh			
025Ch			
025Dh			
025Eh	Peripheral Clock Select Register	PCLKR	00000011b
025Fh			
0260h to 032Fh			
0330h			
0331h			
0332h			
0333h			
0334h			
0335h			
0336h			
0337h			
0338h			
0339h			
033Ah			
033Bh			
033Ch			
033Dh			
033Eh			
033Fh			

**NOTES:**

1. The blank areas are reserved and cannot be accessed by users.
2. This register is included in the flash memory version.

X : Nothing is mapped to this bit

**Table 4.4 SFR Information (4) <sup>(1)</sup>**

Address	Register	Symbol	After Reset
0340h	Timer B3, 4, 5 Count Start Flag	TBSR	000XXXXXb
0341h			
0342h	Timer A1-1 Register	TA11	XXh
0343h			XXh
0344h	Timer A2-1 Register	TA21	XXh
0345h			XXh
0346h	Timer A4-1 Register	TA41	XXh
0347h			XXh
0348h	Three-Phase PWM Control Register 0	INVC0	00h
0349h	Three-Phase PWM Control Register 1	INVC1	00h
034Ah	Three-Phase Output Buffer Register 0	IDB0	00h
034Bh	Three-Phase Output Buffer Register 1	IDB1	00h
034Ch	Dead Time Timer	DTT	XXh
034Dh	Timer B2 Interrupt Occurrence Frequency Set Counter	ICTB2	XXh
034Eh			
034Fh			
0350h	Timer B3 Register	TB3	XXh
0351h			XXh
0352h	Timer B4 Register	TB4	XXh
0353h			XXh
0354h	Timer B5 Register	TB5	XXh
0355h			XXh
0356h			
0357h			
0358h			
0359h			
035Ah			
035Bh	Timer B3 Mode Register	TB3MR	00XX0000b
035Ch	Timer B4 Mode Register	TB4MR	00XX0000b
035Dh	Timer B5 Mode Register	TB5MR	00XX0000b
035Eh	Interrupt Factor Select Register 2	IFSR2A	00XXXXXXb
035Fh	Interrupt Factor Select Register	IFSR	00h
0360h	SI/O3 Transmit/Receive Register	S3TRR	XXh
0361h			
0362h	SI/O3 Control Register	S3C	01000000b
0363h	SI/O3 Bit Rate Generator	S3BRG	XXh
0364h	SI/O4 Transmit/Receive Register	S4TRR	XXh
0365h			
0366h	SI/O4 Control Register	S4C	01000000b
0367h	SI/O4 Bit Rate Generator	S4BRG	XXh
0368h			
0369h			
036Ah			
036Bh			
036Ch	UART0 Special Mode Register 4	U0SMR4	00h
036Dh	UART0 Special Mode Register 3	U0SMR3	000X0X0Xb
036Eh	UART0 Special Mode Register 2	U0SMR2	X0000000b
036Fh	UART0 Special Mode Register	U0SMR	X0000000b
0370h	UART1 Special Mode Register 4	U1SMR4	00h
0371h	UART1 Special Mode Register 3	U1SMR3	000X0X0Xb
0372h	UART1 Special Mode Register 2	U1SMR2	X0000000b
0373h	UART1 Special Mode Register	U1SMR	X0000000b
0374h	UART2 Special Mode Register 4	U2SMR4	00h
0375h	UART2 Special Mode Register 3	U2SMR3	000X0X0Xb
0376h	UART2 Special Mode Register 2	U2SMR2	X0000000b
0377h	UART2 Special Mode Register	U2SMR	X0000000b
0378h	UART2 Transmit/Receive Mode Register	U2MR	00h
0379h	UART2 Bit Rate Generator	U2BRG	XXh
037Ah	UART2 Transmit Buffer Register	U2TB	XXh
037Bh			XXh
037Ch	UART2 Transmit/Receive Control Register 0	U2C0	00001000b
037Dh	UART2 Transmit/Receive Control Register 1	U2C1	00000010b
037Eh	UART2 Receive Buffer Register	U2RB	XXh
037Fh			XXh

**NOTES:**

1. The blank areas are reserved and cannot be accessed by users.

X : Nothing is mapped to this bit

**Table 4.5 SFR Information (5) <sup>(1)</sup>**

Address	Register	Symbol	After Reset
0380h	Count Start Flag	TABSR	00h
0381h	Clock Prescaler Reset Flag	CPSRF	0XXXXXXb
0382h	One-Shot Start Flag	ONSF	00h
0383h	Trigger Select Register	TRGSR	00h
0384h	Up-Down Flag	UDF	00h <sup>(2)</sup>
0385h			
0386h	Timer A0 Register	TA0	XXh
0387h			XXh
0388h	Timer A1 Register	TA1	XXh
0389h			XXh
038Ah	Timer A2 Register	TA2	XXh
038Bh			XXh
038Ch	Timer A3 Register	TA3	XXh
038Dh			XXh
038Eh	Timer A4 Register	TA4	XXh
038Fh			XXh
0390h	Timer B0 Register	TB0	XXh
0391h			XXh
0392h	Timer B1 Register	TB1	XXh
0393h			XXh
0394h	Timer B2 Register	TB2	XXh
0395h			XXh
0396h	Timer A0 Mode Register	TA0MR	00h
0397h	Timer A1 Mode Register	TA1MR	00h
0398h	Timer A2 Mode Register	TA2MR	00h
0399h	Timer A3 Mode Register	TA3MR	00h
039Ah	Timer A4 Mode Register	TA4MR	00h
039Bh	Timer B0 Mode Register	TB0MR	00XX0000b
039Ch	Timer B1 Mode Register	TB1MR	00XX0000b
039Dh	Timer B2 Mode Register	TB2MR	00XX0000b
039Eh	Timer B2 Special Mode Register	TB2SC	XXXXXX00b
039Fh			
03A0h	UART0 Transmit/Receive Mode Register	U0MR	00h
03A1h	UART0 Bit Rate Generator	U0BRG	XXh
03A2h	UART0 Transmit Buffer Register	U0TB	XXh
03A3h			XXh
03A4h	UART0 Transmit/Receive Control Register 0	U0C0	00001000b
03A5h	UART0 Transmit/Receive Control Register 1	U0C1	00XX0010b
03A6h	UART0 Receive Buffer Register	U0RB	XXh
03A7h			XXh
03A8h	UART1 Transmit/Receive Mode Register	U1MR	00h
03A9h	UART1 Bit Rate Generator	U1BRG	XXh
03AAh	UART1 Transmit Buffer Register	U1TB	XXh
03ABh			XXh
03ACh	UART1 Transmit/Receive Control Register 0	U1C0	00001000b
03ADh	UART1 Transmit/Receive Control Register 1	U1C1	00XX0010b
03AEh	UART1 Receive Buffer Register	U1RB	XXh
03AFh			XXh
03B0h	UART Transmit/Receive Control Register 2	UCON	X0000000b
03B1h			
03B2h			
03B3h			
03B4h			
03B5h			
03B6h			
03B7h			
03B8h	DMA0 Request Factor Select Register	DM0SL	00h
03B9h			
03BAh	DMA1 Request Factor Select Register	DM1SL	00h
03BBh			
03BCh	CRC Data Register	CRCD	XXh
03BDh			XXh
03BEh	CRC Input Register	CRCIN	XXh
03BFh			

## NOTES:

1. The blank areas are reserved and cannot be accessed by users.
2. Bit 5 in the Up-down flag is "0" by reset. However, The values in these bits when read are indeterminate.

X : Nothing is mapped to this bit

**Table 4.6 SFR Information (6) <sup>(1)</sup>**

Address	Register	Symbol	After Reset
03C0h 03C1h	A/D Register 0	AD0	XXh XXh
03C2h 03C3h	A/D Register 1	AD1	XXh XXh
03C4h 03C5h	A/D Register 2	AD2	XXh XXh
03C6h 03C7h	A/D Register 3	AD3	XXh XXh
03C8h 03C9h	A/D Register 4	AD4	XXh XXh
03CAh 03CBh	A/D Register 5	AD5	XXh XXh
03CCh 03CDh	A/D Register 6	AD6	XXh XXh
03CEh 03CFh	A/D Register 7	AD7	XXh XXh
03D0h			
03D1h			
03D2h			
03D3h			
03D4h 03D5h	A/D Control Register 2	ADCON2	00h
03D6h	A/D Control Register 0	ADCON0	00000XXXb
03D7h	A/D Control Register 1	ADCON1	00h
03D8h 03D9h	D/A Register 0	DA0	00h
03DAh 03DBh	D/A Register 1	DA1	00h
03DCh 03DDh	D/A Control Register	DACON	00h
03DEh	Port P14 Control Register <sup>(3)</sup>	PC14	XX00XXXXb
03DFh	Pull-Up Control Register 3 <sup>(3)</sup>	PUR3	00h
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	00X00000b
03F3h	Port P9 Direction Register	PD9	00h
03F4h	Port P10 Register	P10	XXh
03F5h	Port P11 Register <sup>(3)</sup>	P11	XXh
03F6h	Port P10 Direction Register	PD10	00h
03F7h	Port P11 Direction Register <sup>(3)</sup>	PD11	00h
03F8h	Port P12 Register <sup>(3)</sup>	P12	XXh
03F9h	Port P13 Register <sup>(3)</sup>	P13	XXh
03FAh	Port P12 Direction Register <sup>(3)</sup>	PD12	00h
03FBh	Port P13 Direction Register <sup>(3)</sup>	PD13	00h
03FCh	Pull-Up Control Register 0	PUR0	00h
03FDh	Pull-Up Control Register 1	PUR1	00000000b <sup>(2)</sup> 00000010b <sup>(2)</sup>
03FEh	Pull-Up Control Register 2	PUR2	00h
03FFh	Port Control Register	PCR	00h

## NOTES:

- The blank areas are reserved and cannot be accessed by users.
- At hardware reset 1 or hardware reset 2, the register is as follows:
  - "00000000b" where "L" is inputted to the CNVSS pin
  - "00000010b" where "H" is inputted to the CNVSS pin
 At software reset, watchdog timer reset and oscillation stop detection reset, the register is as follows:
  - "00000000b" where the PM01 to PM00 bits in the PM0 register are "00b" (single-chip mode).
  - "00000010b" where the PM01 to PM00 bits in the PM0 register are "01b" (memory expansion mode) or "11b" (microprocessor mode).
- These registers do not exist in M16C/62P (80-pin version), and M16C/62PT (80-pin version).

X : Nothing is mapped to this bit

## 5. Reset

Hardware reset 1, brown-out detection reset (hardware reset 2), software reset, watchdog timer reset and oscillation stop detection reset are available to reset the microcomputer.

### 5.1 Hardware Reset 1

The microcomputer resets pins, the CPU and SFR by setting the  $\overline{\text{RESET}}$  pin. If the supply voltage meets the recommended operating conditions, the microcomputer resets all pins when an “L” signal is applied to the  $\overline{\text{RESET}}$  pin (see **Table 5.1 Pin Status When  $\overline{\text{RESET}}$  Pin Level is “L”**). The oscillation circuit is also reset and the main clock starts oscillation. The microcomputer resets the CPU and SFR when the signal applied to the  $\overline{\text{RESET}}$  pin changes low (“L”) to high (“H”). The microcomputer executes the program in an address indicated by the reset vector. The internal RAM is not reset. When an “L” signal is applied to the  $\overline{\text{RESET}}$  pin while writing data to the internal RAM, the internal RAM is in an indeterminate state.

Figure 5.1 shows an Example Reset Circuit. Figure 5.2 shows a Reset Sequence. Table 5.1 lists Pin Status When  $\overline{\text{RESET}}$  Pin Level is “L”. Figure 5.3 shows CPU Register Status After Reset. Refer to **4. Special Function Register (SFR)** for SFR states after reset.

#### 5.1.1 Reset on a Stable Supply Voltage

- (1) Apply “L” to the  $\overline{\text{RESET}}$  pin
- (2) Apply 20 or more clock cycles to the XIN pin
- (3) Apply an “H” signal to the  $\overline{\text{RESET}}$  pin

#### 5.1.2 Power-on Reset

- (1) Apply “L” to the  $\overline{\text{RESET}}$  pin
- (2) Raise the supply voltage to the recommended operating level
- (3) Insert  $t_d(\text{P-R})$  ms as wait time for the internal voltage to stabilize
- (4) Apply 20 or more clock cycles to the XIN pin
- (5) Apply “H” to the  $\overline{\text{RESET}}$  pin

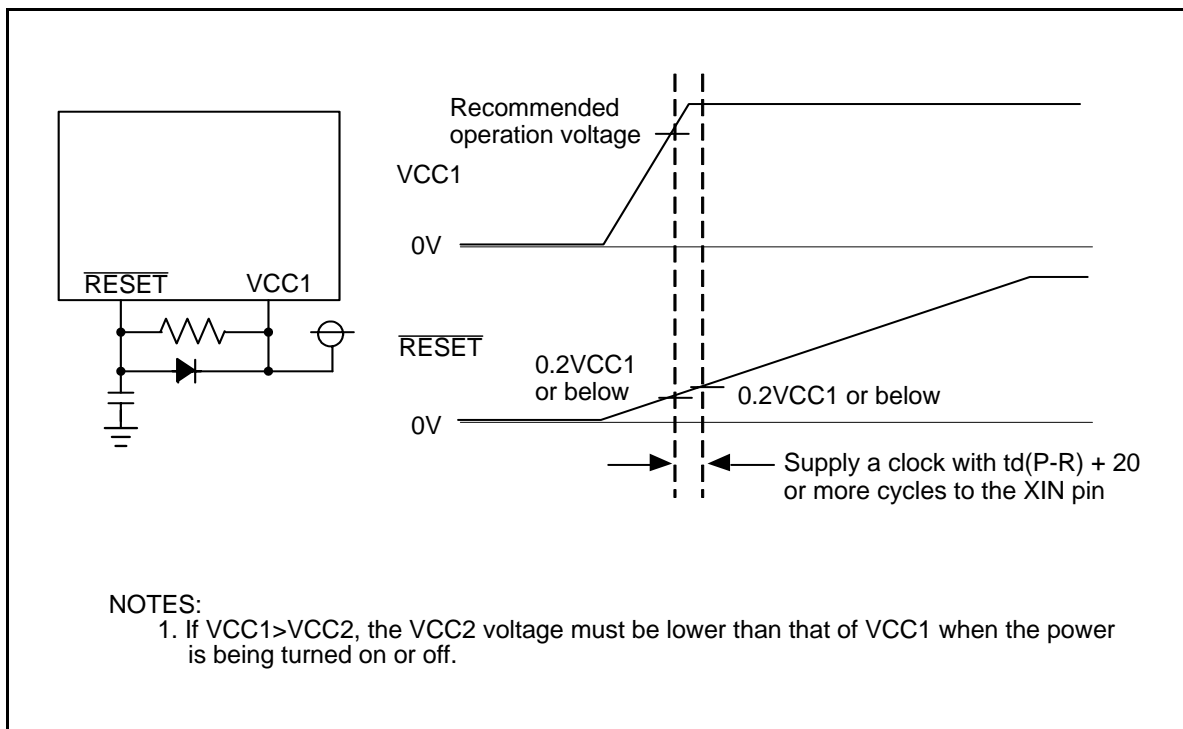


Figure 5.1 Example Reset Circuit

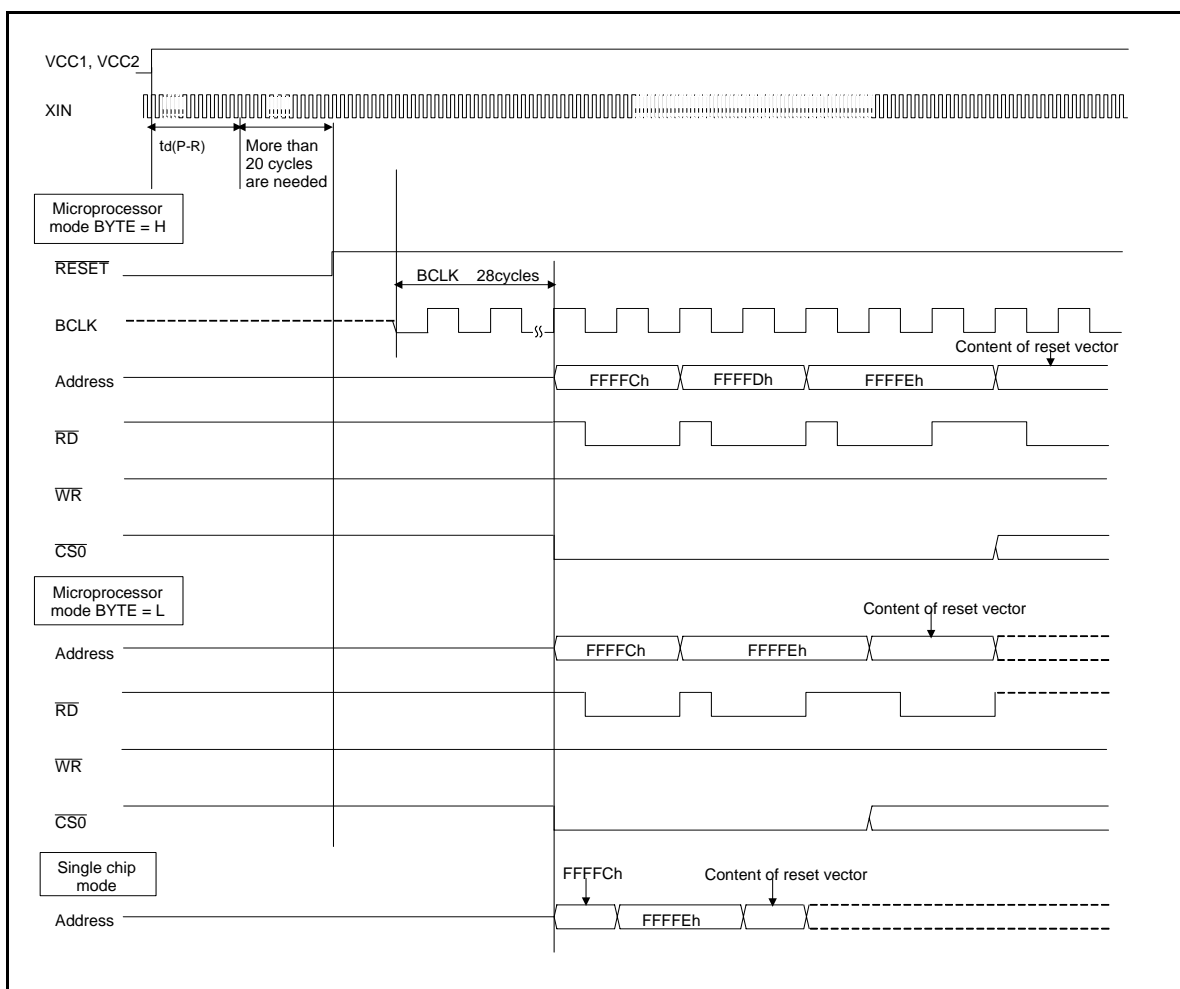


Figure 5.2 Reset Sequence



**Table 5.1 Pin Status When  $\overline{\text{RESET}}$  Pin Level is “L”**

Pin Name	Status		
	CNVSS = VSS	CNVSS = VCC1 <sup>(1)</sup>	
		BYTE = VSS	BYTE = VCC1
P0	Input port	Data input	Data input
P1	Input port	Data input	Input port
P2, P3, P4_0 to P4_3	Input port	Address output (underfined)	Address output (underfined)
P4_4	Input port	$\overline{\text{CS0}}$ output (“H” is output)	$\overline{\text{CS0}}$ output (“H” is output)
P4_5 to P4_7	Input port	Input port (Pulled high)	Input port (Pulled high)
P5_0	Input port	$\overline{\text{WR}}$ output (“H” is output)	$\overline{\text{WR}}$ output (“H” is output)
P5_1	Input port	$\overline{\text{BHE}}$ output (undefined)	$\overline{\text{BHE}}$ output (undefined)
P5_2	Input port	$\overline{\text{RD}}$ output (“H” is output)	$\overline{\text{RD}}$ output (“H” is output)
P5_3	Input port	$\overline{\text{BCLK}}$ output	$\overline{\text{BCLK}}$ output
P5_4	Input port	$\overline{\text{HLDA}}$ output (The output value depends on the input to the $\overline{\text{HOLD}}$ pin)	$\overline{\text{HLDA}}$ output (The output value depends on the input to the $\overline{\text{HOLD}}$ pin)
P5_5	Input port	$\overline{\text{HOLD}}$ input	$\overline{\text{HOLD}}$ input
P5_6	Input port	$\overline{\text{ALE}}$ output (“L” is output)	$\overline{\text{ALE}}$ output (“L” is output)
P5_7	Input port	$\overline{\text{RDY}}$ input	$\overline{\text{RDY}}$ input
P6, P7, P8_0 to P8_4, P8_6, P8_7, P9, P10	Input port	Input port	Input port
P11, P12, P13, P14_0, P14_1 <sup>(2)</sup>	Input port	Input port	Input port

**NOTES:**

1. Shown here is the valid pin state when the internal power supply voltage has stabilized after power on.  
When CNVSS = VCC1, the pin state is indeterminate until the internal power supply voltage stabilizes.
2. P11, P12, P13, P14\_0, P14\_1 pins exist in 128-pin version.

**5.2 Brown-out Detection Reset (Hardware Reset 2)**

The microcomputer resets pins, the CPU or SFR by setting the built-in voltage detect circuit. The voltage detect circuit monitors the voltage applied to the VCC1 pin.

When the VC26 bit in the VCR2 register is set to “1” (reset level detect circuit enabled), the microcomputer resets pins, the CPU and SFR as soon as the voltage that is applied to the VCC1 pin drops to Vdet3 or below.

The microcomputer resets pins and it is in a reset state when the voltage that is applied to the VCC1 pin is Vdet3 or below. The microcomputer resets pins, CPU and SFR with Vdet3r or above and it executes the program from the address determined by the reset vector. The microcomputer executes the program after detecting Vdet3r and waiting td(S-R) ms. The same pins and registers are reset by the hardware reset 1 and brown-out detection reset (hardware reset 2), and are also placed in the same reset state.

The microcomputer cannot exit stop mode by the brown-out detection reset (hardware reset 2).

### 5.3 Software Reset

The microcomputer resets pins, the CPU and SFR when the PM03 bit in the PM0 register is set to “1” (microcomputer reset). Then the microcomputer executes the program in an address determined by the reset vector. Set the PM03 bit to “1” while the main clock is selected as the CPU clock and the main clock oscillation is stable. In the software reset, the microcomputer does not reset a part of the SFR. Refer to **4. Special Function Register (SFR)** for details.

Processor mode remains unchanged since the PM01 to PM00 bits in the PM0 register are not reset.

### 5.4 Watchdog Timer Reset

The microcomputer resets pins, the CPU and SFR when the CM06 bit in the CM0 register is set to “1” (reset) and the watchdog timer underflows. Then the microcomputer executes the program in an address determined by the reset vector.

In the watchdog timer reset, the microcomputer does not reset a part of the SFR. Refer to **4. Special Function Register (SFR)** for details. Processor mode remains unchanged since the PM01 to PM00 bits in the PM0 register are not reset.

### 5.5 Oscillation Stop Detection Reset

The microcomputer resets and stops pins, the CPU and SFR when the CM27 bit in the CM2 register is 0, if it detects main clock oscillation circuit stop. Refer to **10.6 Oscillation Stop and Re-oscillation Detect Function** for details.

In the oscillation stop detection reset, the microcomputer does not reset a part of the SFR. Refer to **4. Special Function Register (SFR)** for details. Processor mode remains unchanged since the PM01 to PM00 bits in the PM0 register are not reset.

## 5.6 Internal Space

Figure 5.3 shows CPU Register Status After Reset. Refer to **4. Special Function Register (SFR)** for SFR states after reset.

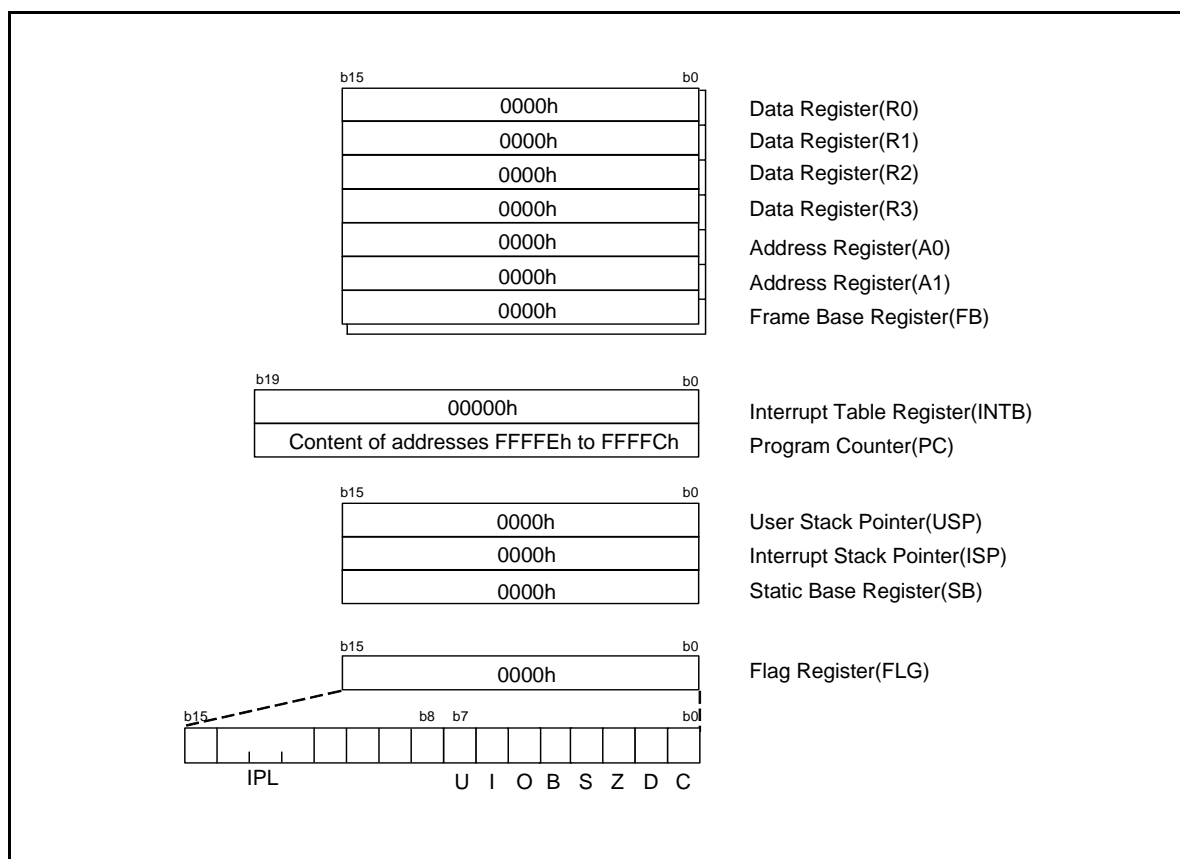


Figure 5.3 CPU Register Status After Reset

## 6. Voltage Detection Circuit

### Note

The M16C/62PT do not use the voltage detection circuit.  
However, the cold start-up/warm start-up determine function is available.

The voltage detection circuit consists of the reset level detection circuit and the low voltage detection circuit. The reset level detection circuit monitors the voltage applied to the VCC1 pin. The microcomputer is reset if the reset level detection circuit detects VCC1 is Vdet3 or below. This circuit is disabled when the microcomputer is in stop mode.

The voltage detection circuit also monitors the voltage applied to the VCC1 pin. The low voltage detection signal is generated when the low voltage detection circuit detects VCC1 is above or below Vdet4. This signal generates the low voltage detection interrupt. The VC13 bit in the VCR1 register determines whether VCC1 is above or below Vdet4. The voltage detection circuit is available when VCC1=5.0V.

Figure 6.1 shows a Voltage Detection Circuit Block.

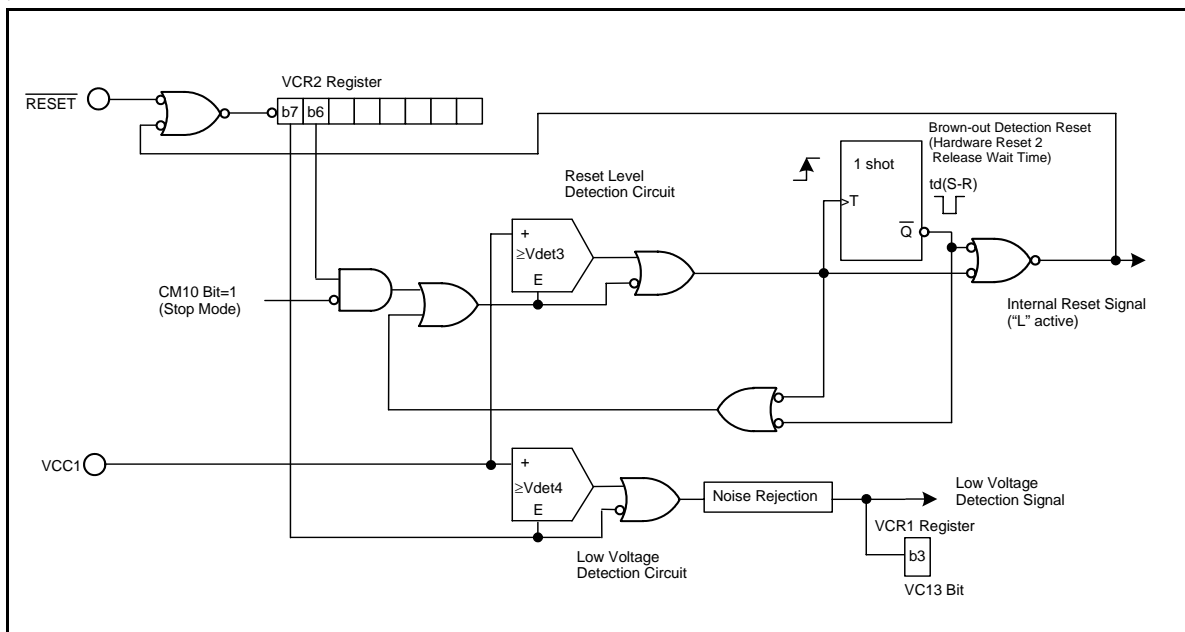
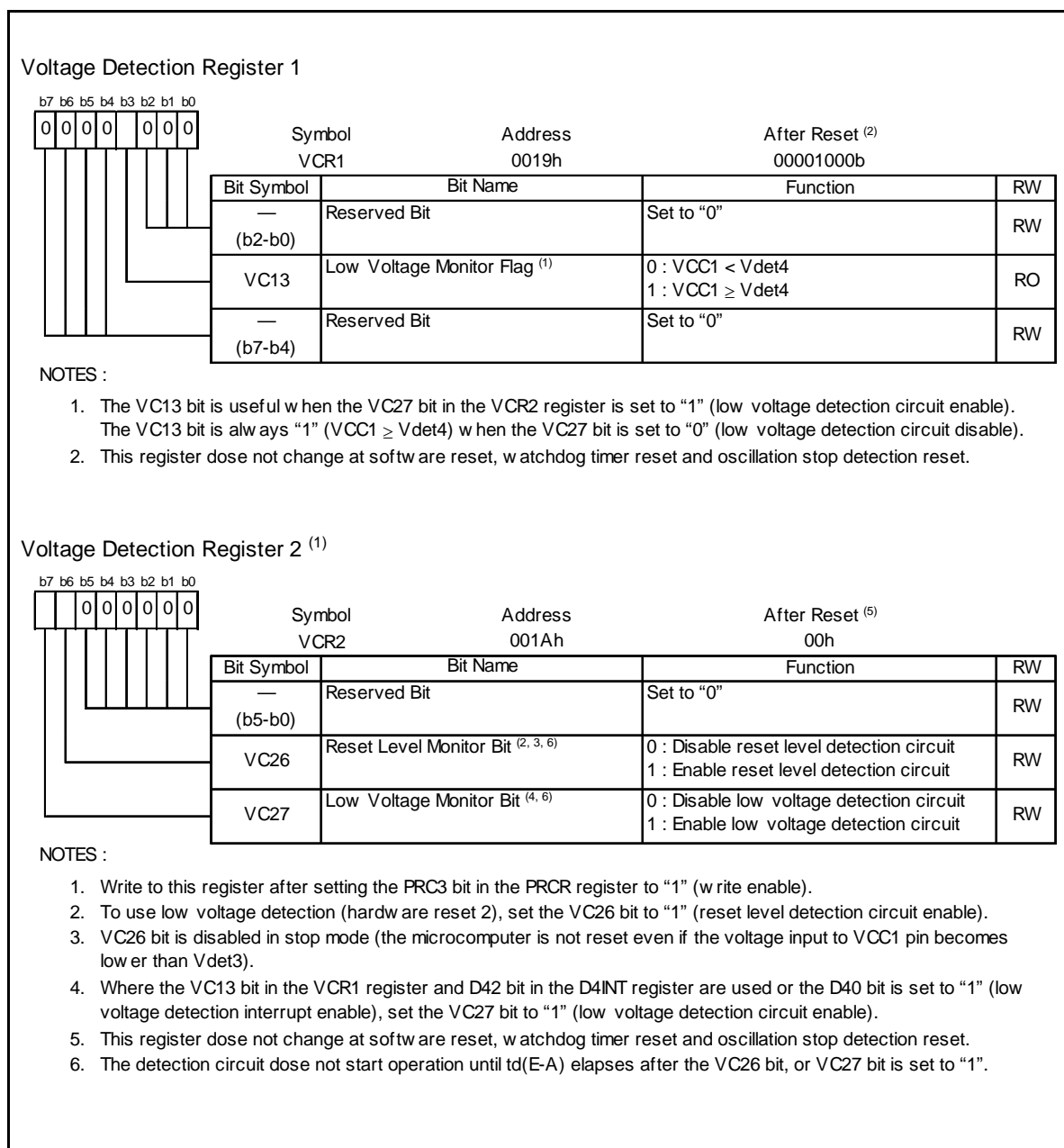


Figure 6.1 Voltage Detection Circuit Block

**Figure 6.2 VCR1 and VCR2 Registers**

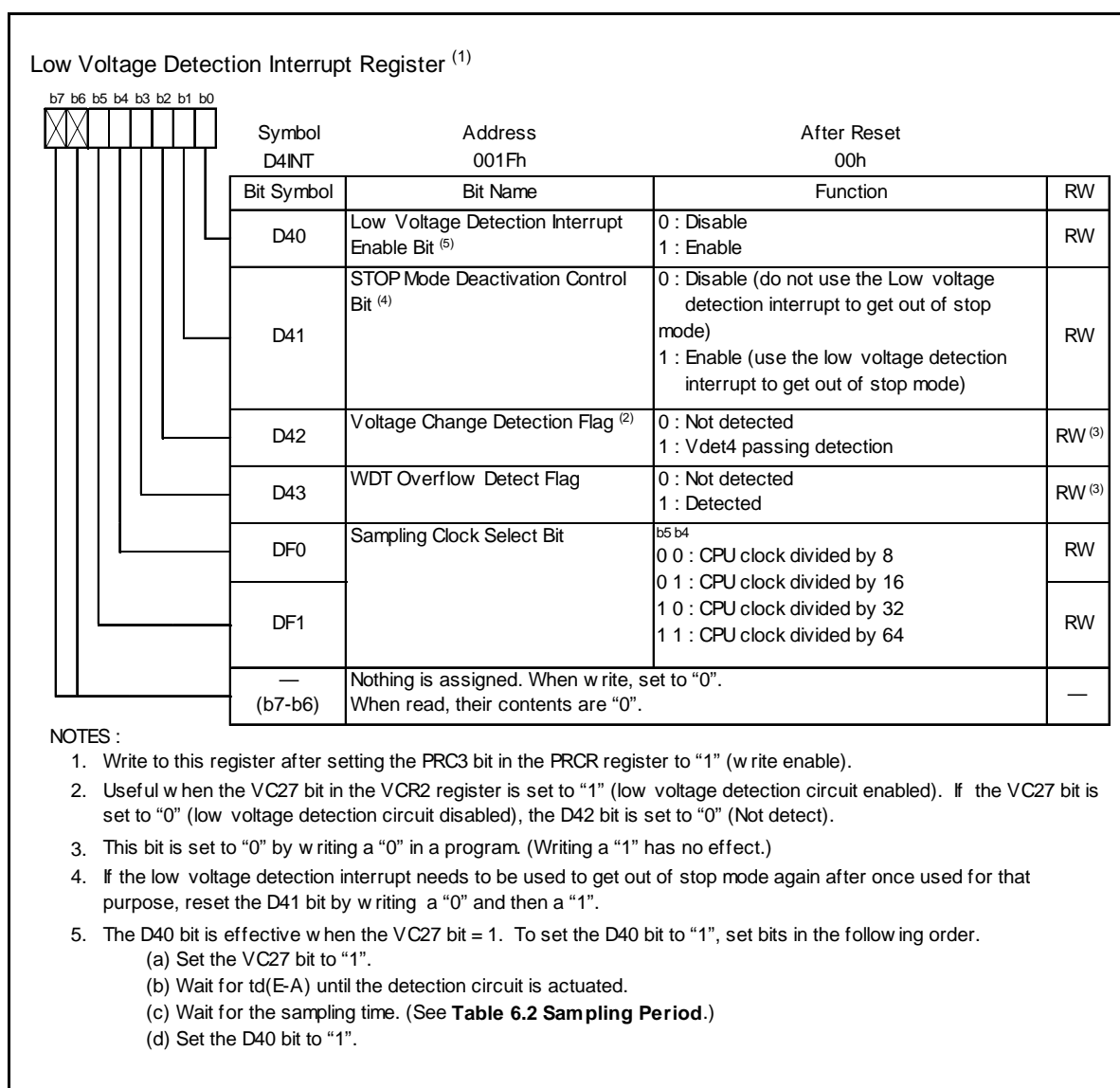
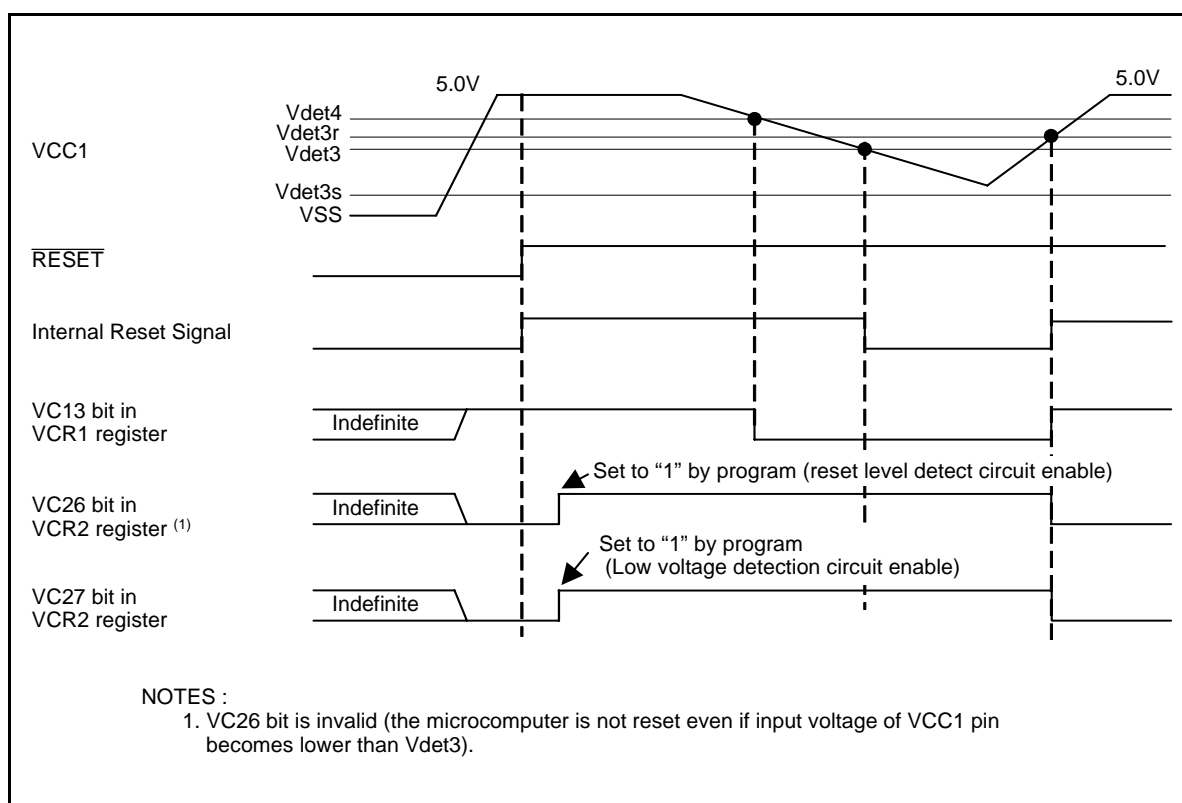


Figure 6.3 D4INT Register



**Figure 6.4 Typical Operation of Brown-out Detection Reset (Hardware Reset 2)**

## 6.1 Low Voltage Detection Interrupt

If the D40 bit in the D4INT register is set to “1” (low voltage detection interrupt enabled), the low voltage detection interrupt request is generated when the voltage applied to the VCC1 pin is above or below Vdet4.

The low voltage detection interrupt shares the same interrupt vector with the watchdog timer interrupt and oscillation stop, re-oscillation detection interrupt.

Set the D41 bit in the D4INT register to “1” (enabled) to use the low voltage detection interrupt to exit stop mode.

The D42 bit in the D4INT register is set to “1” as soon as the voltage applied to the VCC1 pin reaches Vdet4 due to the voltage rise and voltage drop. When the D42 bit changes “0” to “1”, the low voltage detection interrupt request is generated. Set the D42 bit to “0” by program. However, when the D41 bit is set to “1” and the microcomputer is in stop mode, the low voltage detection interrupt request is generated regardless of the D42 bit state if the voltage applied to the VCC1 pin is detected to be above Vdet4. The microcomputer then exits stop mode.

Table 6.1 shows Low Voltage Detection Interrupt Request Generation Conditions.

The DF1 to DF0 bits in the D4INT register determine the sampling period that detects the voltage applied to the VCC1 pin reaches Vdet4. Table 6.2 shows the Sampling Periods.

**Table 6.1 Low Voltage Detection Interrupt Request Generation Conditions**

Operating Mode	VC27 Bit	D40 Bit	D41 Bit	D42 Bit	CM02 Bit	VC13 Bit
Normal Operating Mode <sup>(1)</sup>	1	1	–	0 to 1	–	0 to 1 <sup>(3)</sup>
						1 to 0 <sup>(3)</sup>
Wait Mode <sup>(2)</sup>			–	0 to 1	0	0 to 1 <sup>(3)</sup>
						1 to 0 <sup>(3)</sup>
			1	–	1	0 to 1
Stop Mode <sup>(2)</sup>				–	0	0 to 1

– : “0” or “1”

1. The status except the wait mode and stop mode is handled as the normal mode. (Refer to **10. Clock Generation Circuit**)
2. Refer to **6.2 Limitations on Exiting Stop Mode**, **6.3 Limitations on Exiting Wait Mode**.
3. An interrupt request for voltage reduction is generated a sampling time after the value of the VC13 bit has changed.

See the **Figure 6.6 Low Voltage Detection Interrupt Generation Circuit Operation Example** for details.

**Table 6.2 Sampling Periods**

CPU Clock (D4INT clock) (MHz)	Sampling Clock (μs)			
	DF1 to DF0=00 (CPU clock divided by 8)	DF1 to DF0=01 (CPU clock divided by 16)	DF1 to DF0=10 (CPU clock divided by 32)	DF1 to DF0=11 (CPU clock divided by 64)
16	3.0	6.0	12.0	24.0



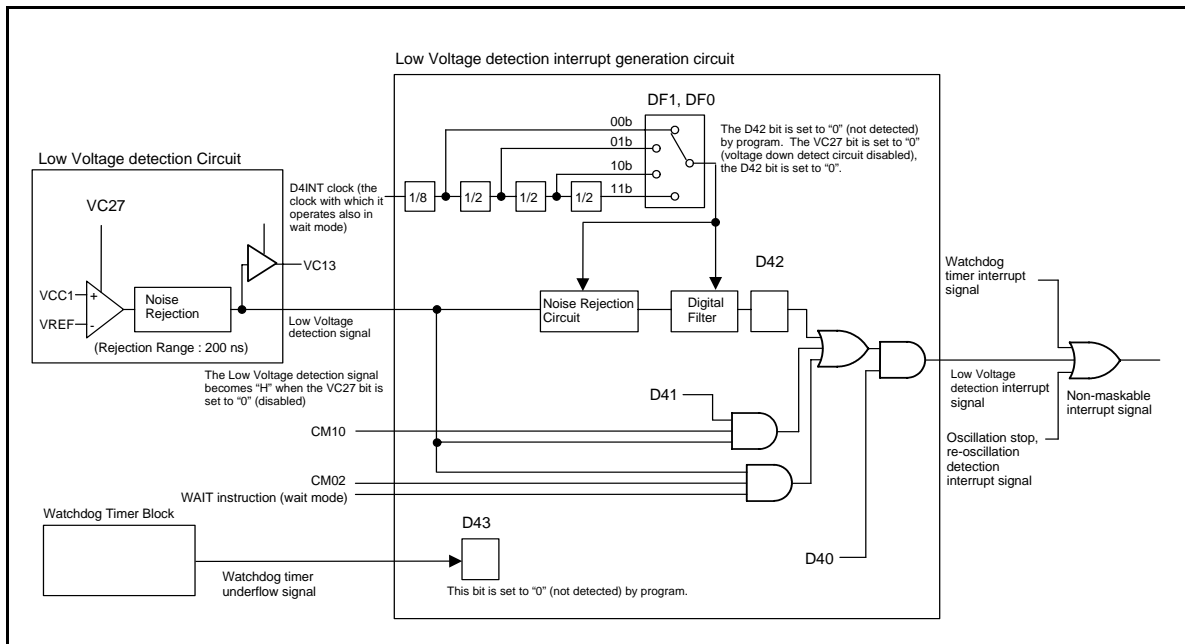


Figure 6.5 Low Voltage detection Interrupt Generation Block

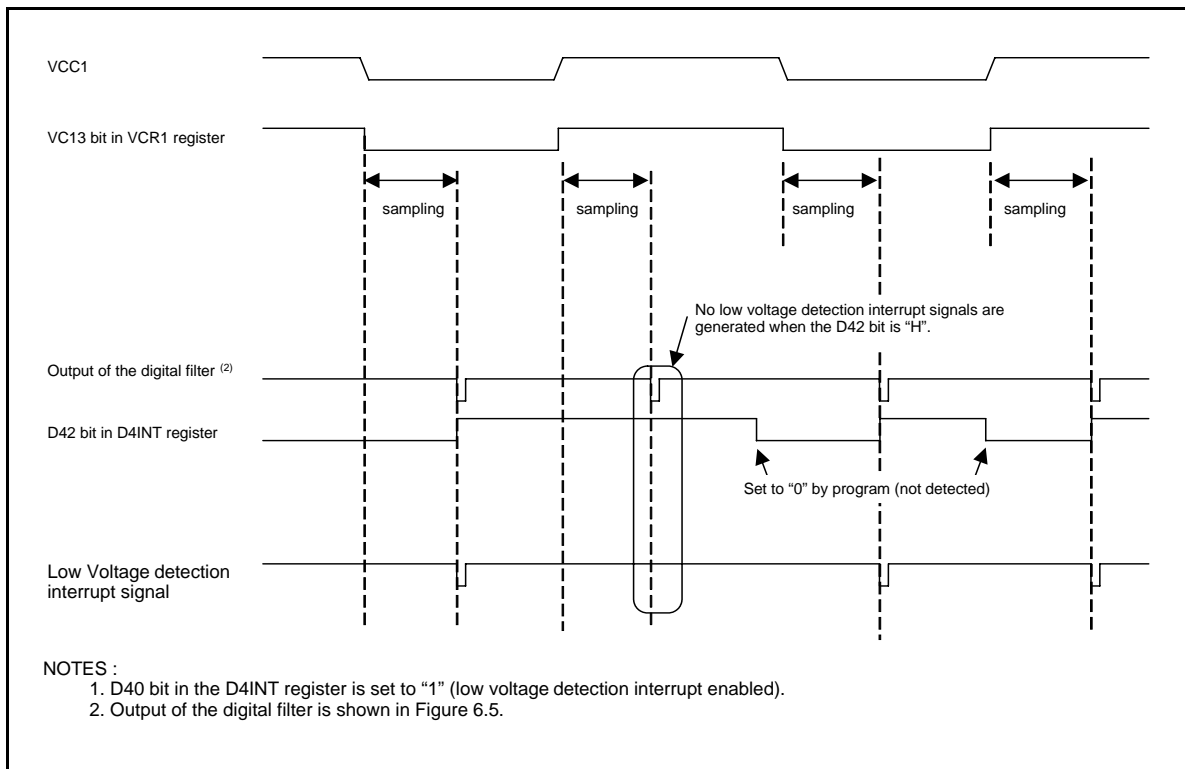


Figure 6.6 Low Voltage Detection Interrupt Generation Circuit Operation Example

## 6.2 Limitations on Exiting Stop Mode

The low voltage detection interrupt is immediately generated and the microcomputer exits stop mode if the CM10 bit in the CM1 register is set to “1” under the conditions below.

- the VC27 bit in the VCR2 register is set to “1” (low voltage detection circuit enabled),
- the D40 bit in the D4INT register is set to “1” (low voltage detection interrupt enabled),
- the D41 bit in the D4INT register is set to “1” (low voltage detection interrupt is used to exit stop mode), and
- the voltage applied to the VCC1 pin is higher than Vdet4 (the VC13 bit in the VCR1 register is “1”)

If the microcomputer is set to enter stop mode when the voltage applied to the VCC1 pin drops below Vdet4 and to exit stop mode when the voltage applied rises to Vdet4 or above, set the CM10 bit to “1” when VC13 bit is “0” ( $VCC1 < Vdet4$ ).

## 6.3 Limitations on Exiting Wait Mode

The low voltage detection interrupt is immediately generated and the microcomputer exits wait mode If WAIT instruction is executed under the conditions below.

- the CM02 bit in the CM0 register is set to “1” (stop peripheral function clock),
- the VC27 bit in the VCR2 register is set to “1” (low voltage detection circuit enabled),
- the D40 bit in the D4INT register is set to “1” (low voltage detection interrupt enabled),
- the D41 bit in the D4INT register is set to “1” (low voltage detection interrupt is used to exit wait mode), and
- the voltage applied to the VCC1 pin is higher than Vdet4 (the VC13 bit in the VCR1 register is “1”)

If the microcomputer is set to enter wait mode when the voltage applied to the VCC1 pin drops below Vdet4 and to exit wait mode when the voltage applied rises to Vdet4 or above, perform WAIT instruction when VC13 bit is “0” ( $VCC1 < Vdet4$ ).

## 6.4 Cold Start-up / Warm Start-up Determine Function

As for the cold start-up/warm start-up determine function, the WDC5 flag in the WDC register determines either cold start-up (reset process) when power-on or warm start-up (reset process) when reset signal is applied during the microcomputer running.

Default value of the WDC5 bit is "0" (cold start-up) when power-on. It is set to "1" (warm start-up) by writing desired values to the WDC register. The WDC bit is not reset, regardless of a software reset or a reset operation.

Figure 6.7 shows Cold Start-up/Warm Start-up Determine Function Block Diagram. Figure 6.8 shows the Cold Start-up/Warm Start-up Determine Function Operation Example. Figure 6.9 shows WDC Register.

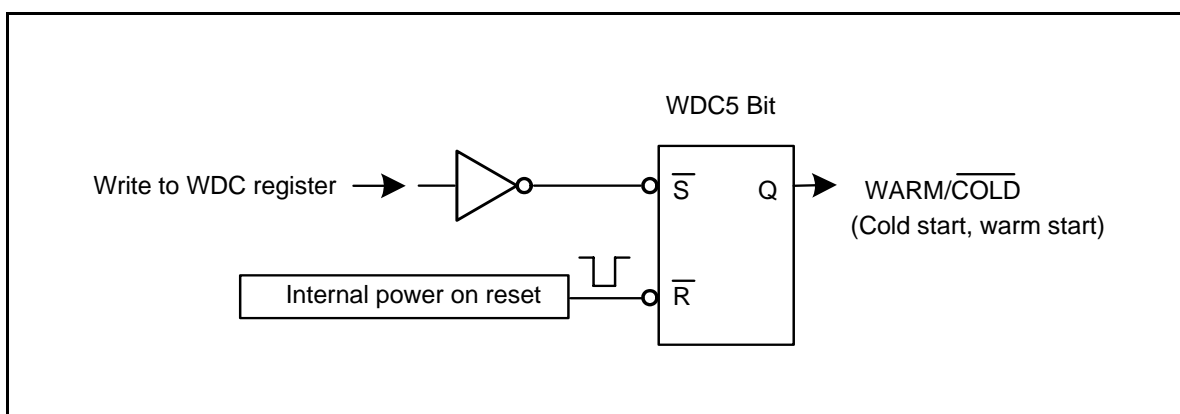


Figure 6.7 Cold Start-up/Warm Start-up Determine Function Block Diagram

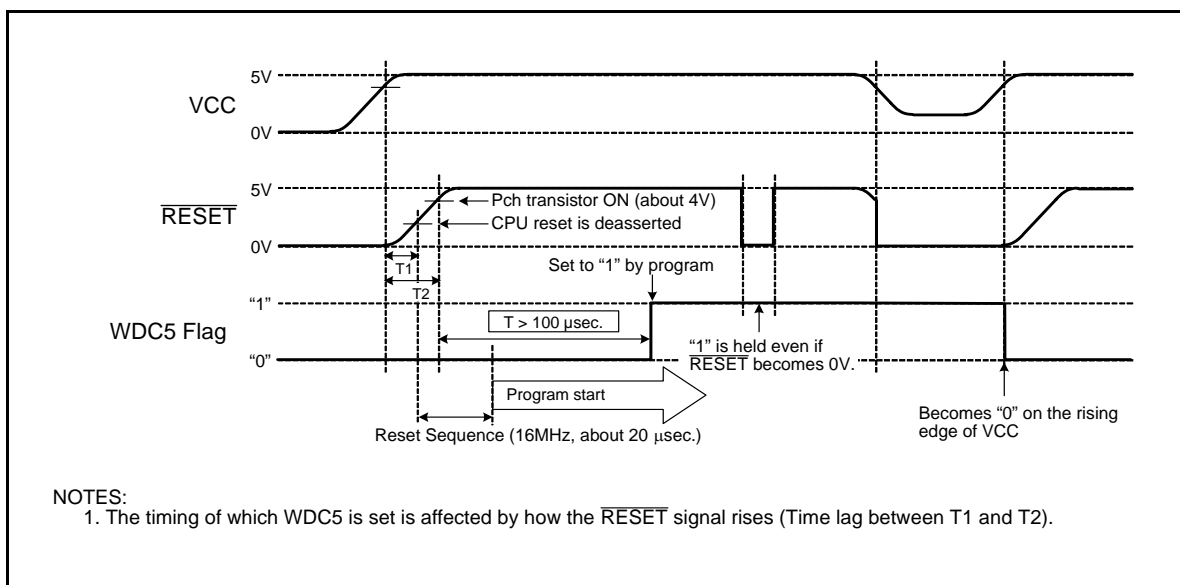
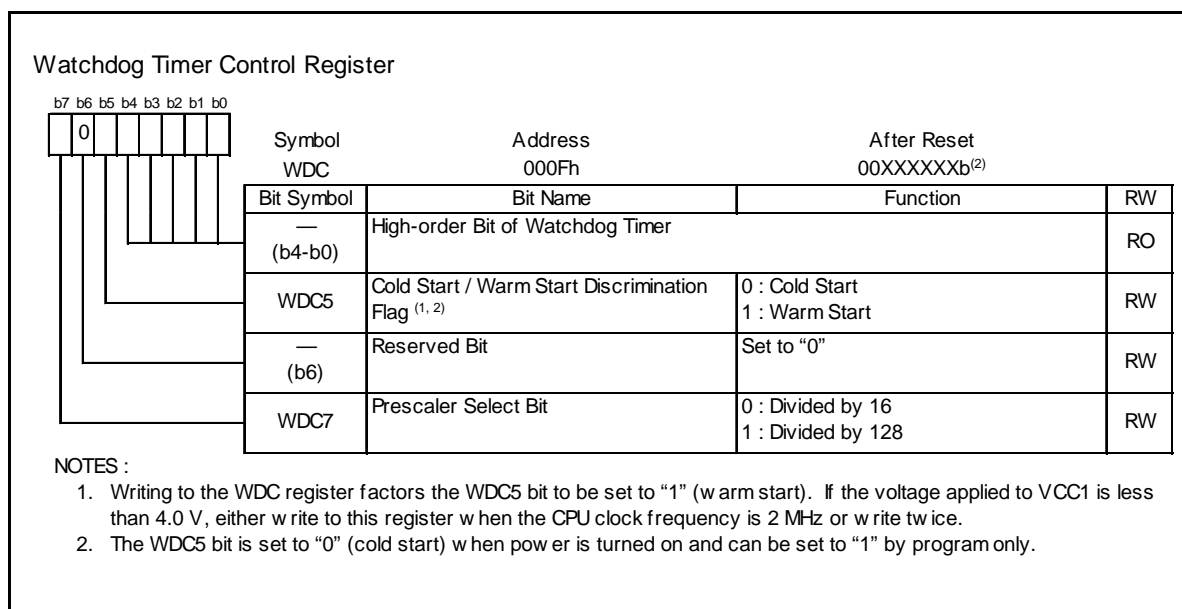


Figure 6.8 Cold Start-up/Warm Start-up Determine Function Operation Example

**Figure 6.9 WDC Register**

## 7. Processor Mode

### Note

The M16C/62P (80-pin version) and M16C/62PT do not use memory expansion mode, and microprocessor mode.

### 7.1 Types of Processor Mode

Three processor modes are available to choose from: single-chip mode, memory expansion mode, and microprocessor mode. Table 7.1 shows the Features of Processor Modes.

**Table 7.1 Features of Processor Modes**

Processor Modes	Access Space	Pins which are Assigned I/O Ports
Single-Chip Mode	SFR, Internal RAM, Internal ROM	All pins are I/O ports or peripheral function I/O pins
Memory Expansion Mode	SFR, Internal RAM, Internal ROM, External Area <sup>(1)</sup>	Some pins serve as bus control pins <sup>(1)</sup>
Microprocessor Mode	SFR, Internal RAM, External Area <sup>(1)</sup>	Some pins serve as bus control pins <sup>(1)</sup>

#### NOTES:

1. Refer to **8. Bus**.

## 7.2 Setting Processor Modes

Processor mode is set by using the CNVSS pin and the PM01 to PM00 bits in the PM0 register.

Table 7.2 shows the Processor Mode After Hardware Reset. Table 7.3 shows the PM01 to PM00 Bits Set Values and Processor Modes.

**Table 7.2 Processor Mode After Hardware Reset**

CNVSS Pin Input Level	Processor Modes
VSS	Single-Chip Mode
VCC1 (1, 2)	Microprocessor Mode

**NOTES:**

1. If the microcomputer is reset in hardware by applying VCC1 to the CNVSS pin (hardware reset 1 or brown-out detection reset (hardware reset 2)), the internal ROM cannot be accessed regardless of PM10 to PM00 bits.
2. The multiplexed bus cannot be assigned to the entire  $\overline{CS}$  space.

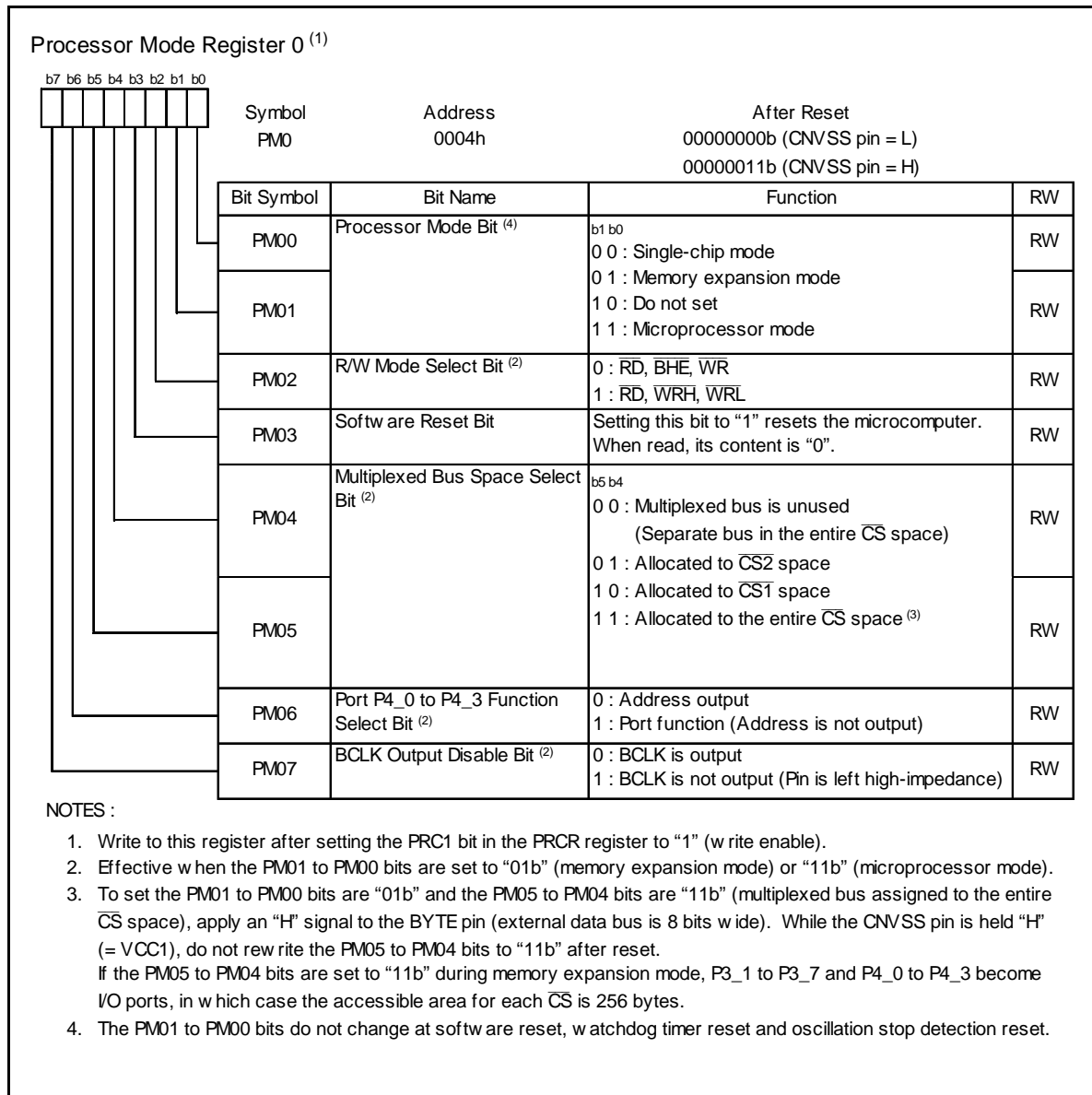
**Table 7.3 PM01 to PM00 Bits Set Values and Processor Modes**

PM01 to PM00 Bits	Processor Modes
00b	Single-Chip Mode
01b	Memory Expansion Mode
10b	Do not set
11b	Microprocessor Mode

Rewriting the PM01 to PM00 bits places the microcomputer in the corresponding processor mode regardless of whether the input level on the CNVSS pin is “H” or “L”. Note, however, that the PM01 to PM00 bits cannot be rewritten to “01b” (memory expansion mode) or “11b” (microprocessor mode) at the same time the PM07 to PM02 bits are rewritten. Note also that these bits cannot be rewritten to enter microprocessor mode in the internal ROM, nor can they be rewritten to exit microprocessor mode in areas overlapping the internal ROM.

If the microcomputer is reset in hardware by applying VCC1 to the CNVSS pin (hardware reset 1 or brown-out detection reset (hardware reset 2)), the internal ROM cannot be accessed regardless of PM01 to PM00 bits.

Figures 7.1 and 7.2 show the PM0 Register and PM1 Register. Figure 7.3 show the Memory Map in Single Chip Mode.

**Figure 7.1 PM0 Register**

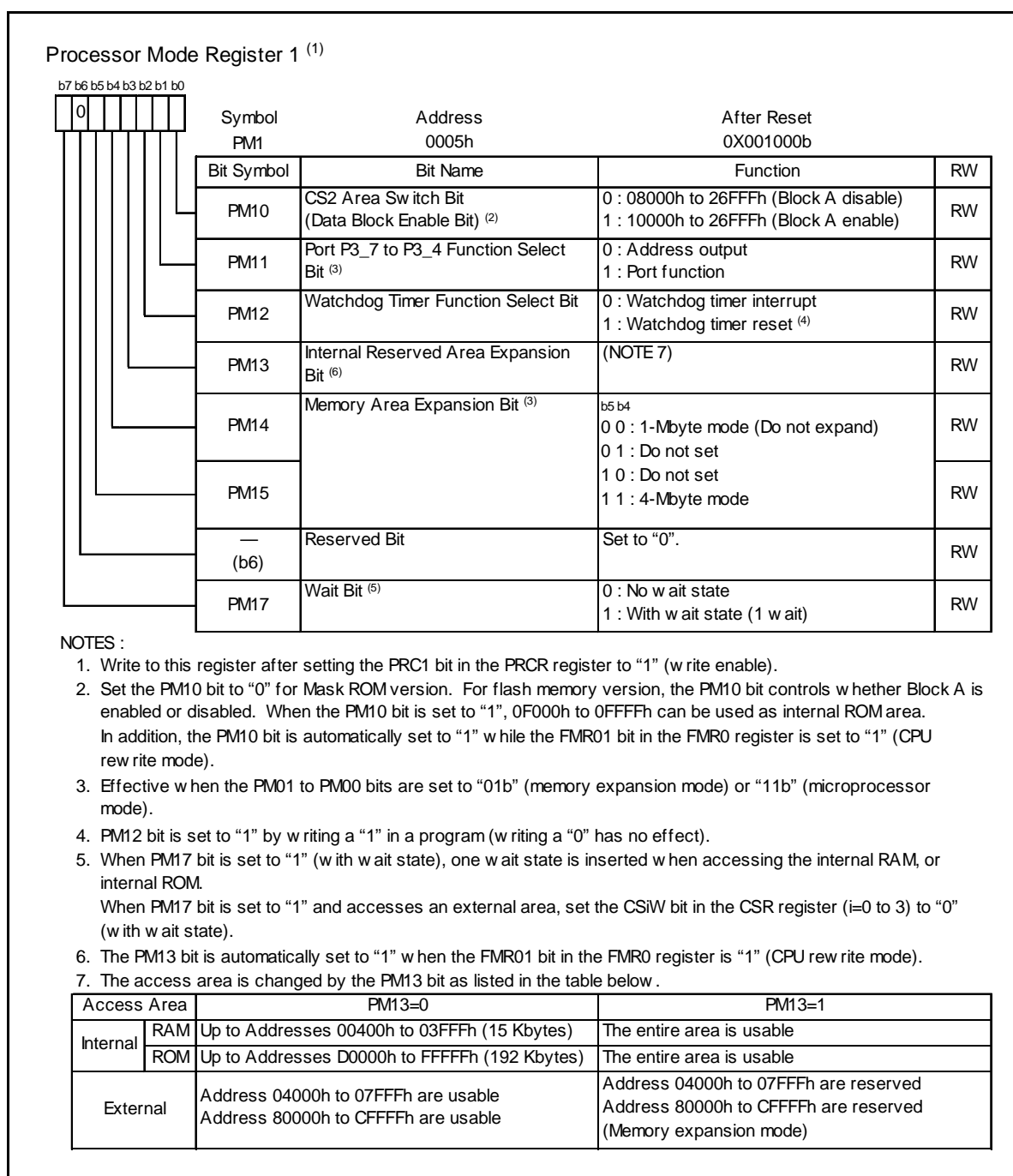
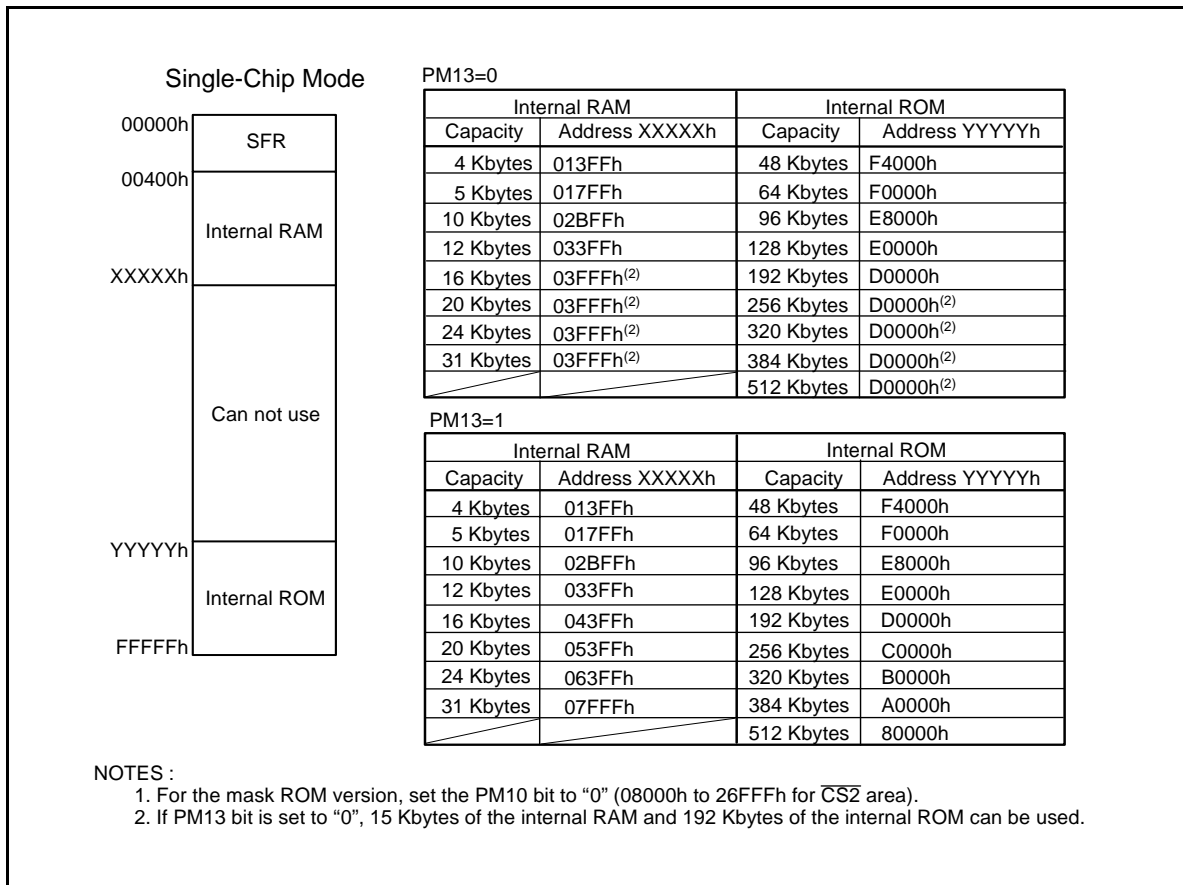


Figure 7.2 PM1 Register



**Figure 7.3 Memory Map in Single Chip Mode**

## 8. Bus

### Note

The M16C/62P (80-pin version) and M16C/62PT do not use bus control pins.

During memory expansion or microprocessor mode, some pins serve as the bus control pins to perform data input/output to and from external devices. These bus control pins include A0 to A19, D0 to D15,  $\overline{CS0}$  to  $\overline{CS3}$ ,  $\overline{RD}$ ,  $\overline{WRL}/\overline{WR}$ ,  $\overline{WRH}/\overline{BHE}$ ,  $\overline{ALE}$ ,  $\overline{RDY}$ ,  $\overline{HOLD}$ ,  $\overline{HLDA}$  and  $\overline{BCLK}$ .

### 8.1 Bus Mode

The bus mode, either multiplexed or separate, can be selected using the PM05 to PM04 bits in the PM0 register. Table 8.1 shows the Difference Between a Separate Bus and Multiplexed Bus.

#### 8.1.1 Separate Bus

In this bus mode, data and address are separate.

#### 8.1.2 Multiplexed Bus

In this bus mode, data and address are multiplexed.

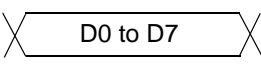
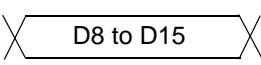
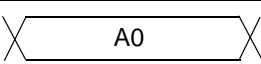

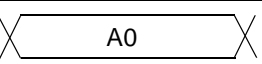
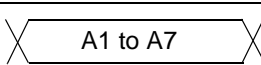
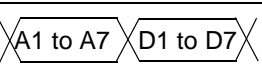
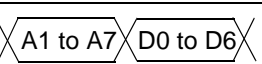
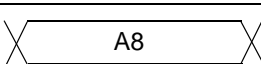
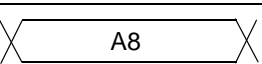

##### 8.1.2.1 When the input level on BYTE pin is high (8-bit data bus)

D0 to D7 and A0 to A7 are multiplexed.

##### 8.1.2.2 When the input level on BYTE pin is low (16-bit data bus)

D0 to D7 and A1 to A8 are multiplexed. D8 to D15 are not multiplexed. Do not use D8 to D15. External devices connecting to a multiplexed bus are allocated to only the even addresses of the microcomputer. Odd addresses cannot be accessed.

**Table 8.1 Difference Between a Separate Bus and Multiplexed Bus**

Pin Name (1)	Separate Bus	Multiplex Bus	
		BYTE = H	BYTE = L
P0_0 to P0_7/D0 to D7		(NOTE 2)	(NOTE 2)
P1_0 to P1_7/D8 to D15		I/O Port P1_0 to P1_7	(NOTE 2)
P2_0/A0 (/D0/-)			
P2_1 to P2_7/A1 to A7 (/D1 to D7/D0 to D6)			
P3_0/A8 (/D7)			

#### NOTES:

- See **Table 8.6 Pin Functions for Each Processor Mode** for bus control signals other than the above. Setting Processor Modes.
- It changes with a setup of PM05 to PM04, and area to access.  
See **Table 8.6 Pin Functions for Each Processor Mode** for details.

## 8.2 Bus Control

The following describes the signals needed for accessing external devices and the functionality of software wait.

### 8.2.1 Address Bus

The address bus consists of 20 lines, A0 to A19. The address bus width can be chosen to be 12, 16 or 20 bits by using the PM06 bit in the PM0 register and the PM11 bit in the PM1 register. Table 8.2 shows the PM06 and PM11 Bits Set Value and Address Bus Width.

**Table 8.2 PM06 and PM11 Bits Set Value and Address Bus Width**

Set Value <sup>(1)</sup>	Pin Function	Address Bus Width
PM11=1	P3_4 to P3_7	12 bits
PM06=1	P4_0 to P4_3	
PM11=0	A12 to A15	16 bits
PM06=1	P4_0 to P4_3	
PM11=0	A12 to A15	20 bits
PM06=0	A16 to A19	

**NOTES:**

1. No values other than those shown above can be set.

When processor mode is changed from single-chip mode to memory extension mode, the address bus is indeterminate until any external area is accessed.

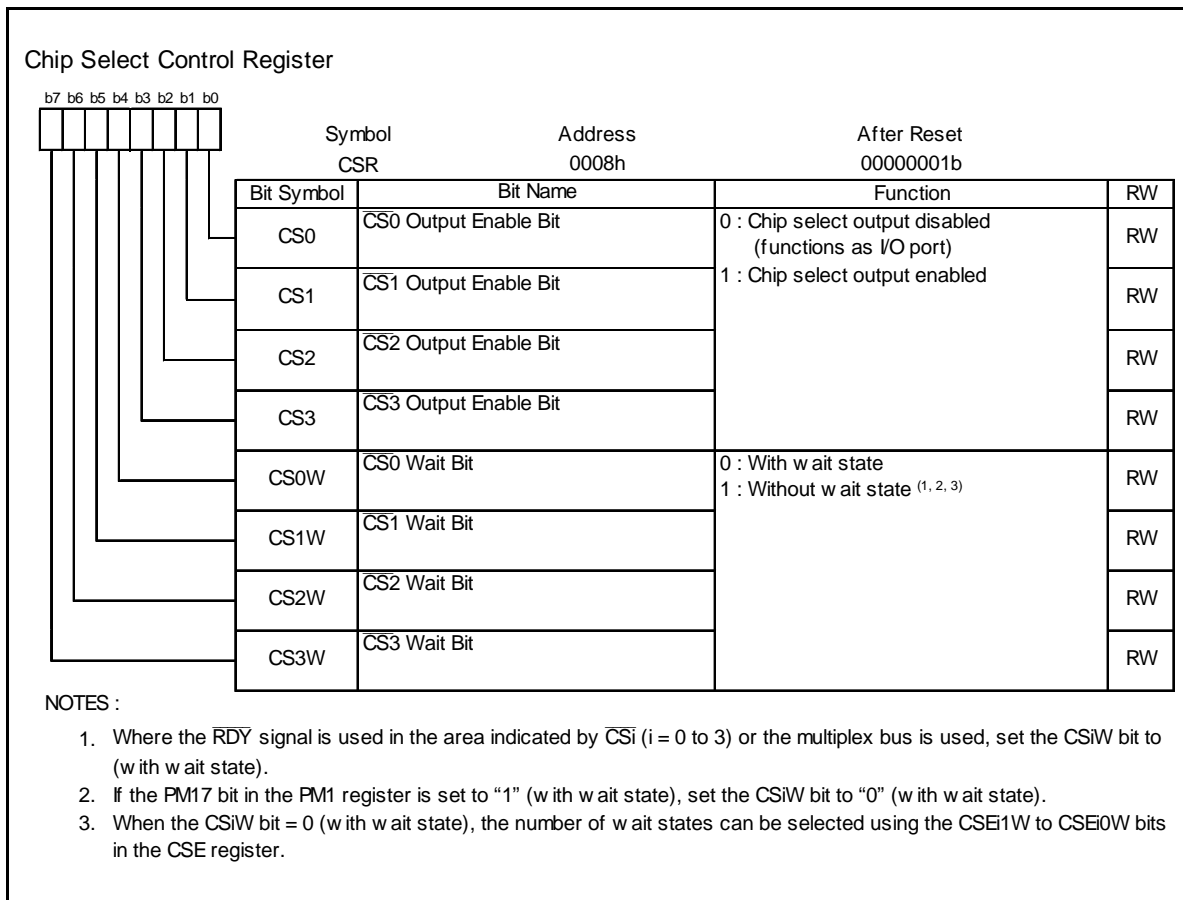
### 8.2.2 Data Bus

When input on the BYTE pin is high (data bus is 8 bits wide), 8 lines D0 to D7 comprise the data bus; when input on the BYTE pin is low (data bus is 16 bits wide), 16 lines D0 to D15 comprise the data bus. Do not change the input level on the BYTE pin while in operation.

### 8.2.3 Chip Select Signal

The chip select (hereafter referred to as the  $\overline{\text{CSi}}$ ) signals are output from the  $\overline{\text{CSi}}$  ( $i = 0$  to 3) pins. These pins can be chosen to function as I/O ports or as  $\overline{\text{CS}}$  by using the CSi bit in the CSR register. Figure 8.1 shows the CSR Register.

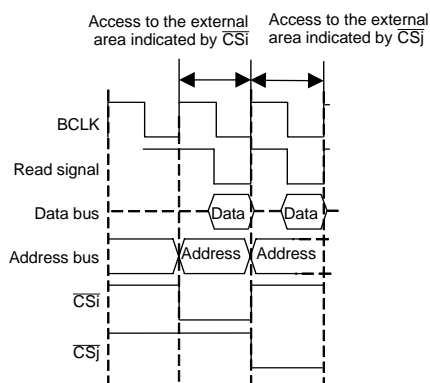
During 1-Mbyte mode, the external area can be separated into up to 4 by the  $\overline{\text{CSi}}$  signal which is output from the  $\overline{\text{CSi}}$  pin. During 4-Mbyte mode,  $\overline{\text{CSi}}$  signal or bank number is output from the  $\overline{\text{CSi}}$  pin. Refer to **9. Memory Space Expansion Function**. Figure 8.2 shows the Example of Address Bus and  $\overline{\text{CSi}}$  Signal Output in 1-Mbyte mode.

**Figure 8.1 CSR Register**

## Example 1

To access the external area indicated by  $\overline{CSi}$  in the next cycle after accessing the external area indicated by  $\overline{CSj}$

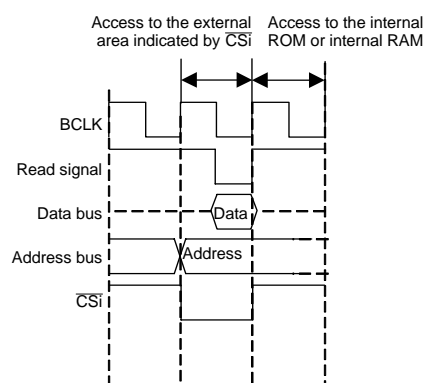
The address bus and the chip select signal both change state between these two cycles.



## Example 2

To access the internal ROM or internal RAM in the next cycle after accessing the external area indicated by  $\overline{CSi}$

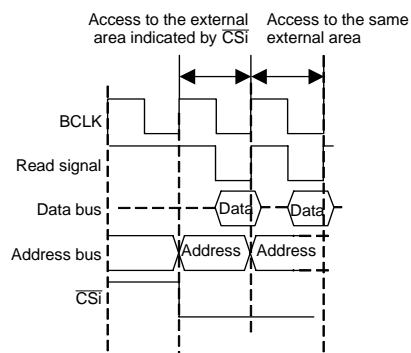
The chip select signal changes state but the address bus does not change state



## Example 3

To access the external area indicated by  $\overline{CSi}$  in the next cycle after accessing the external area indicated by the same  $\overline{CSi}$

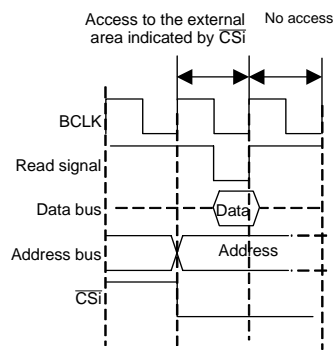
The address bus changes state but the chip select signal does not change state



## Example 4

Not to access any area (nor instruction prefetch generated) in the next cycle after accessing the external area indicated by  $\overline{CSi}$

Neither the address bus nor the chip select signal changes state between these two cycles



## NOTES :

- These examples show the address bus and chip select signal when accessing areas in two successive cycles. The chip select bus cycle may be extended more than two cycles depending on a combination of these examples.

Shown above is the case where separate bus is selected and the area is accessed for read without wait states.  $i = 0$  to  $3$ ,  $j = 0$  to  $3$  (not including  $i$ , however)

Figure 8.2 Example of Address Bus and  $\overline{CSi}$  Signal Output in 1-Mbyte mode

### 8.2.4 Read and Write Signals

When the data bus is 16 bits wide, the read and write signals can be chosen to be a combination of  $\overline{RD}$ ,  $\overline{BHE}$  and  $\overline{WR}$  or a combination of  $\overline{RD}$ ,  $\overline{WRL}$  and  $\overline{WRH}$  by using the PM02 bit in the PM0 register. When the data bus is 8 bits wide, use a combination of  $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{BHE}$ .

Table 8.3 shows the Operation of  $\overline{RD}$ ,  $\overline{WRL}$ , and  $\overline{WRH}$  Signals. Table 8.4 shows the Operation of  $\overline{RD}$ ,  $\overline{WRL}$ , and  $\overline{BHE}$  Signals.

**Table 8.3 Operation of  $\overline{RD}$ ,  $\overline{WRL}$  and  $\overline{WRH}$  Signals**

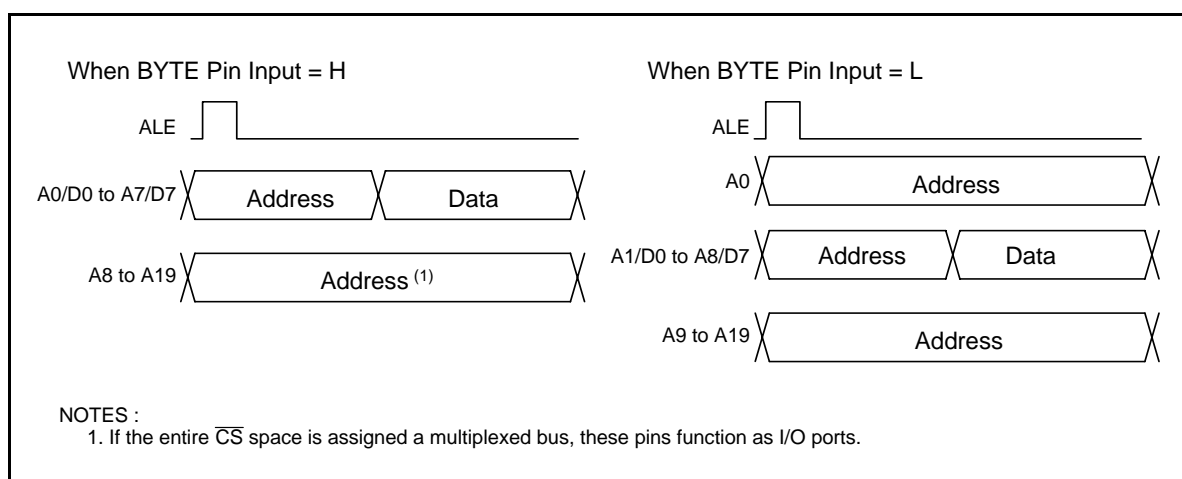
Data Bus Width	$\overline{RD}$	$\overline{WRL}$	$\overline{WRH}$	Status of External Data Bus
16-bit (BYTE pin input = L)	L	H	H	Read data
	H	L	H	Write 1 byte of data to an even address
	H	H	L	Write 1 byte of data to an odd address
	H	L	L	Write data to both even and odd addresses

**Table 8.4 Operation of  $\overline{RD}$ ,  $\overline{WRL}$  and  $\overline{BHE}$  Signals**

Data Bus Width	$\overline{RD}$	$\overline{WRL}$	$\overline{BHE}$	A0	Status of External Data Bus
16-bit (BYTE pin input = L)	H	L	L	H	Write 1 byte of data to an odd address
	L	H	L	H	Read 1 byte of data from an odd address
	H	L	H	L	Write 1 byte of data to an even address
	L	H	H	L	Read 1 byte of data from an even address
	H	L	L	L	Write data to both even and odd addresses
	L	H	L	L	Read data from both even and odd addresses
8-bit (BYTE pin input = H)	H	L	Not used	H or L	Write 1 byte of data
	L	H	Not used	H or L	Read 1 byte of data

### 8.2.5 ALE Signal

The ALE signal latches the address when accessing the multiplex bus space. Latch the address when the ALE signal falls.



**Figure 8.3 ALE Signal, Address Bus, Data Bus**

### 8.2.6 $\overline{\text{RDY}}$ Signal

This signal is provided for accessing external devices which need to be accessed at low speed. If input on the  $\overline{\text{RDY}}$  pin is asserted low at the last falling edge of BCLK of the bus cycle, one wait state is inserted in the bus cycle. While in a wait state, the following signals retain the state in which they were when the  $\overline{\text{RDY}}$  signal was acknowledged.

A0 to A19, D0 to D15,  $\overline{\text{CS0}}$  to  $\overline{\text{CS3}}$ ,  $\overline{\text{RD}}$ ,  $\overline{\text{WRL}}$ ,  $\overline{\text{WRH}}$ ,  $\overline{\text{WR}}$ ,  $\overline{\text{BHE}}$ ,  $\overline{\text{ALE}}$ ,  $\overline{\text{HLDA}}$

Then, when the input on the  $\overline{\text{RDY}}$  pin is detected high at the falling edge of BCLK, the remaining bus cycle is executed. Figure 8.4 shows Example in which the Wait State was Inserted into Read Cycle by  $\overline{\text{RDY}}$  Signal. To use the  $\overline{\text{RDY}}$  signal, set the corresponding bit ( $\text{CS3W}$  to  $\text{CS0W}$  bits) in the CSR register to "0" (with wait state). When not using the  $\overline{\text{RDY}}$  signal, the  $\overline{\text{RDY}}$  pin must be pulled-up.

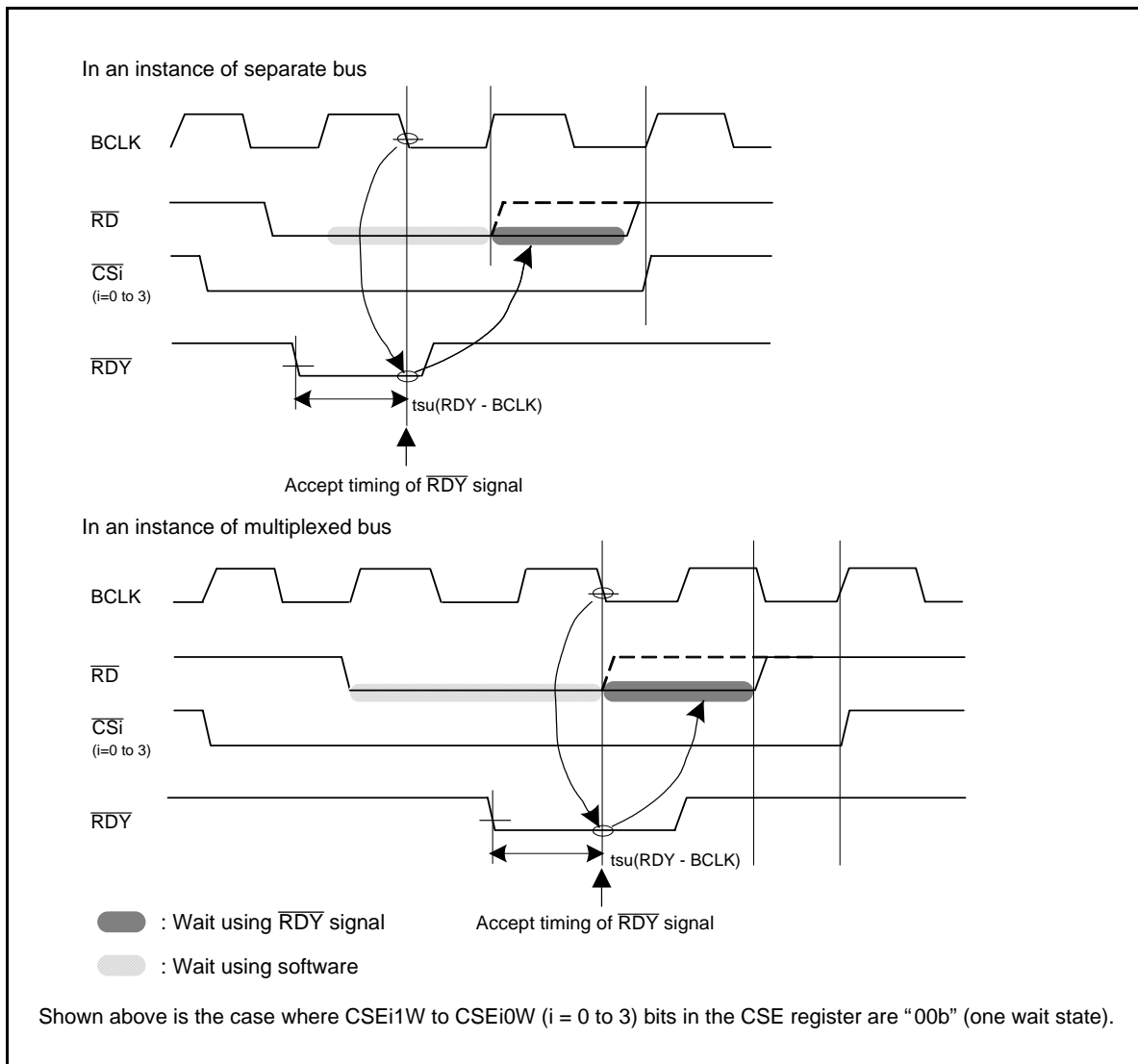


Figure 8.4 Example in which Wait State was Inserted into Read Cycle by  $\overline{\text{RDY}}$  Signal

### 8.2.7 $\overline{\text{HOLD}}$ Signal

This signal is used to transfer control of the bus from the CPU or DMAC to an external circuit. When the input on  $\overline{\text{HOLD}}$  pin is pulled low, the microcomputer is placed in a hold state after the bus access then in process finishes. The microcomputer remains in the hold state while the  $\overline{\text{HOLD}}$  pin is held low, during which time the  $\overline{\text{HLDA}}$  pin outputs a low-level signal.

Table 8.5 shows the Microcomputer Status in Hold State.

Bus-using priorities are given to  $\overline{\text{HOLD}}$ , DMAC, and CPU in order of decreasing precedence. However, if the CPU is accessing an odd address in word units, the DMAC cannot gain control of the bus during two separate accesses.

$\overline{\text{HOLD}} > \text{DMAC} > \text{CPU}$

**Figure 8.5 Bus-Using Priorities**

**Table 8.5 Microcomputer Status in Hold State**

Item		Status
BCLK		Output
A0 to A19, D0 to D15, $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$ , $\overline{\text{RD}}$ , $\overline{\text{WRL}}$ , $\overline{\text{WRH}}$ , $\overline{\text{WR}}$ , $\overline{\text{BHE}}$		High-impedance
I/O ports	P0, P1, P3, P4 <sup>(2)</sup>	High-impedance
	P6 to P14 <sup>(1)</sup>	Maintains status when $\overline{\text{HOLD}}$ signal is received
$\overline{\text{HLDA}}$		Output "L"
Internal Peripheral Circuits		ON (but watchdog timer stops) <sup>(3)</sup>
ALE Signal		Undefined

**NOTES:**

1. P11 to P14 are included in the 128-pin version.
2. When I/O port function is selected.
3. The watchdog timer does not stop when the PM22 bit in the PM2 register is set to "1" (the count source for the watchdog timer is the on-chip oscillator clock).

### 8.2.8 8.2.8 BCLK Output

If the PM07 bit in the PM0 register is set to "0" (output enable), a clock with the same frequency as that of the CPU clock is output as BCLK from the BCLK pin. Refer to **10.2 CPU Clock and Peripheral Function Clock**.



**Table 8.6 Pin Functions for Each Processor Mode**

Processor Mode		Memory Expansion Mode or Microprocessor Mode				Memory Expansion Mode
PM05 to PM04 bits		00b(separate bus) bits		01b( $\overline{CS2}$ is for multiplexed bus and others are for separate bus) 10b( $\overline{CS1}$ is for multiplexed bus and others are for separate bus)		11b (multiplexed bus for the entire space) <sup>(1)</sup>
Data Bus Width BYTE Pin		8 bits “H”	16 bits “L”	8 bits “H”	16 bits “L”	8 bits “H”
P0_0 to P0_7		D0 to D7	D0 to D7	D0 to D7 <sup>(4)</sup>	D0 to D7 <sup>(4)</sup>	I/O ports
P1_0 to P1_7		I/O ports	D8 to D15	I/O ports	D8 to D15 <sup>(4)</sup>	I/O ports
P2_0		A0	A0	A0/D0 <sup>(2)</sup>	A0	A0/D0
P2_1 to P2_7		A1 to A7	A1 to A7	A1 to A7 /D1 to D7 <sup>(2)</sup>	A1 to A7 /D0 to D6 <sup>(2)</sup>	A1 to A7 /D1 to D7
P3_0		A8	A8	A8	A8/D7 <sup>(2)</sup>	A8
P3_1 to P3_3		A9 to A11				I/O ports
P3_4 to P3_7	PM11=0	A12 to A15				I/O ports
	PM11=1	I/O ports				
P4_0 to P4_3	PM06=0	A16 to A19				I/O ports
	PM06=1	I/O ports				
P4_4	CS0=0	I/O ports				
	CS0=1	$\overline{CS0}$				
P4_5	CS1=0	I/O ports				
	CS1=1	$\overline{CS1}$				
P4_6	CS2=0	I/O ports				
	CS2=1	$\overline{CS2}$				
P4_7	CS3=0	I/O ports				
	CS3=1	$\overline{CS3}$				
P5_0	PM02=0	$\overline{WR}$				
	PM02=1	$\text{--}$ <sup>(3)</sup>	$\overline{WRL}$	$\text{--}$ <sup>(3)</sup>	$\overline{WRL}$	$\text{--}$ <sup>(3)</sup>
P5_1	PM02=0	$\overline{BHE}$				
	PM02=1	$\text{--}$ <sup>(3)</sup>	$\overline{WRH}$	$\text{--}$ <sup>(3)</sup>	$\overline{WRH}$	$\text{--}$ <sup>(3)</sup>
P5_2		$\overline{RD}$				
P5_3		BCLK				
P5_4		$\overline{HLDA}$				
P5_5		HOLD				
P5_6		ALE				
P5_7		RDY				

I/O ports : Function as I/O ports or peripheral function I/O pins.

**NOTES:**

1. To set the PM01 to PM00 bits are set to "01b" and the PM05 to PM04 bits are set to "11b" (multiplexed bus assigned to the entire  $\overline{CS}$  space), apply "H" to the BYTE pin (external data bus 8 bits wide). While the CNVSS pin is held "H" (= VCC1), do not rewrite the PM05 to PM04 bits to "11b" after reset. If the PM05 to PM04 bits are set to "11b" during memory expansion mode, P3\_1 to P3\_7 and P4\_0 to P4\_3 become I/O ports, in which case the accessible area for each  $\overline{CS}$  is 256 bytes.
2. In separate bus mode, these pins serve as the address bus.
3. If the data bus is 8 bits wide, make sure the PM02 bit is set to "0" ( $\overline{RD}$ ,  $\overline{BHE}$ ,  $\overline{WR}$ ).
4. When accessing the area that uses a multiplexed bus, these pins output an indeterminate value during a write.

### 8.2.9 External Bus Status When Internal Area Accessed

Table 8.7 shows the External Bus Status When Internal Area Accessed.

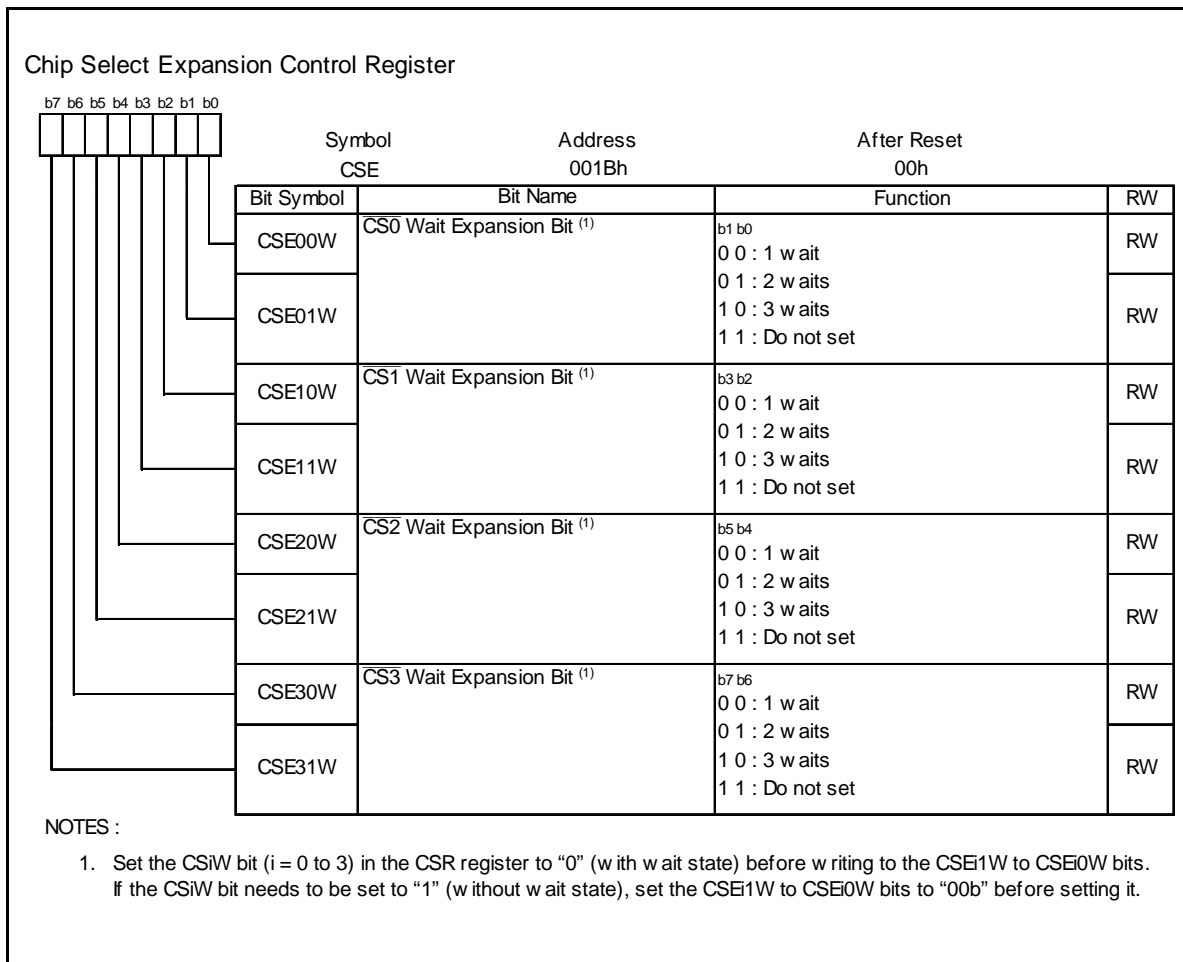
**Table 8.7 External Bus Status When Internal Area Accessed**

Item		SFR Accessed	Internal ROM, RAM Accessed
A0 to A19		Address output	Maintain status before accessed address of external area or SFR
D0 to D15	When Read	High-impedance	High-impedance
	When Write	Output data	Undefined
$\overline{RD}$ , $\overline{WR}$ , $\overline{WRL}$ , $\overline{WRH}$		$\overline{RD}$ , $\overline{WR}$ , $\overline{WRL}$ , $\overline{WRH}$ output	Output "H"
$\overline{BHE}$		$\overline{BHE}$ output	Maintain status before accessed status of external area or SFR
$\overline{CS0}$ to $\overline{CS3}$		Output "H"	Output "H"
ALE		Output "L"	Output "L"

### 8.2.10 Software Wait

Software wait states can be inserted by using the PM17 bit in the PM1 register, the CS0W to CS3W bits in the CSR register, and the CSE register. The SFR area is unaffected by these control bits. This area is always accessed in 2 BCLK or 3 BCLK cycles as determined by the PM20 bit in the PM2 register. See **Table 8.8 Bit and Bus Cycle Related to Software Wait** for details.

To use the RDY signal, set the corresponding CS3W to CS0W bit to "0" (with wait state). Figure 8.6 shows the CSE Register. Table 8.8 shows the Bit and Bus Cycle Related to Software Wait. Figure 8.7 and 8.8 show the Typical Bus Timings Using Software Wait.



**Figure 8.6 CSE Register**

**Table 8.8 Bit and Bus Cycle Related to Software Wait**

Area	Bus Mode	PM2 Register PM20 Bit	PM1 Register PM17 Bit <sup>(5)</sup>	CSR Register CS3W Bit <sup>(1)</sup> CS2W Bit <sup>(1)</sup> CS1W Bit <sup>(1)</sup> CS0W Bit <sup>(1)</sup>	CSE Register CSE31W to CSE30W Bit CSE21W to CSE20W Bit CSE11W to CSE10W Bit CSE01W to CSE00W Bit	Software Wait	Bus Cycle
SFR	–	1	–	–	–	–	2 BCLK cycles <sup>(3)</sup>
	–	0	–	–	–	–	3 BCLK cycles <sup>(3)</sup>
Internal RAM, ROM	–	–	0	–	–	No wait	1 BCLK cycle <sup>(4)</sup>
	–	–	1	–	–	1 wait	2 BCLK cycles
External Area	Separate Bus	–	0	1	00b	No wait	1 BCLK cycle (read)
		–	0	1	00b	No wait	2 BCLK cycles (write)
		–	–	0	00b	1 wait	2 BCLK cycle <sup>(4)</sup>
		–	–	0	01b	2 waits	3 BCLK cycles
		–	–	0	10b	3 waits	4 BCLK cycle
		–	1	0	00b	1 wait	2 BCLK cycle
	Multiplexed Bus <sup>(2)</sup>	–	–	0	00b	1 wait	3 BCLK cycles
		–	–	0	01b	2 waits	3 BCLK cycles
		–	–	0	10b	3 waits	4 BCLK cycles
		–	1	0	00b	1 wait	3 BCLK cycles

**NOTES:**

1. To use the  $\overline{\text{RDY}}$  signal, set this bit to “0”.
2. To access in multiplexed bus mode, set the corresponding bit of CS0W to CS3W to “0” (with wait state).
3. When the selected CPU clock source is the PLL clock, the number of wait cycles can be altered by the PM20 bit in the PM2 register. When using a 16 MHz or higher PLL clock, be sure to set the PM20 bit to “0” (2 wait cycles).
4. After reset, the PM17 bit is set to “0” (without wait state), all of the CS0W to CS3W bits are set to “0” (with wait state), and the CSE register is set to “00h” (one wait state for  $\overline{\text{CS0}}$  to  $\overline{\text{CS3}}$ ). Therefore, the internal RAM and internal ROM are accessed with no wait states, and all external areas are accessed with one wait state.
5. When PM17 bit is set to “1” and accesses an external area, set the CSiW (i=0 to 3) bits to “0” (with wait state).

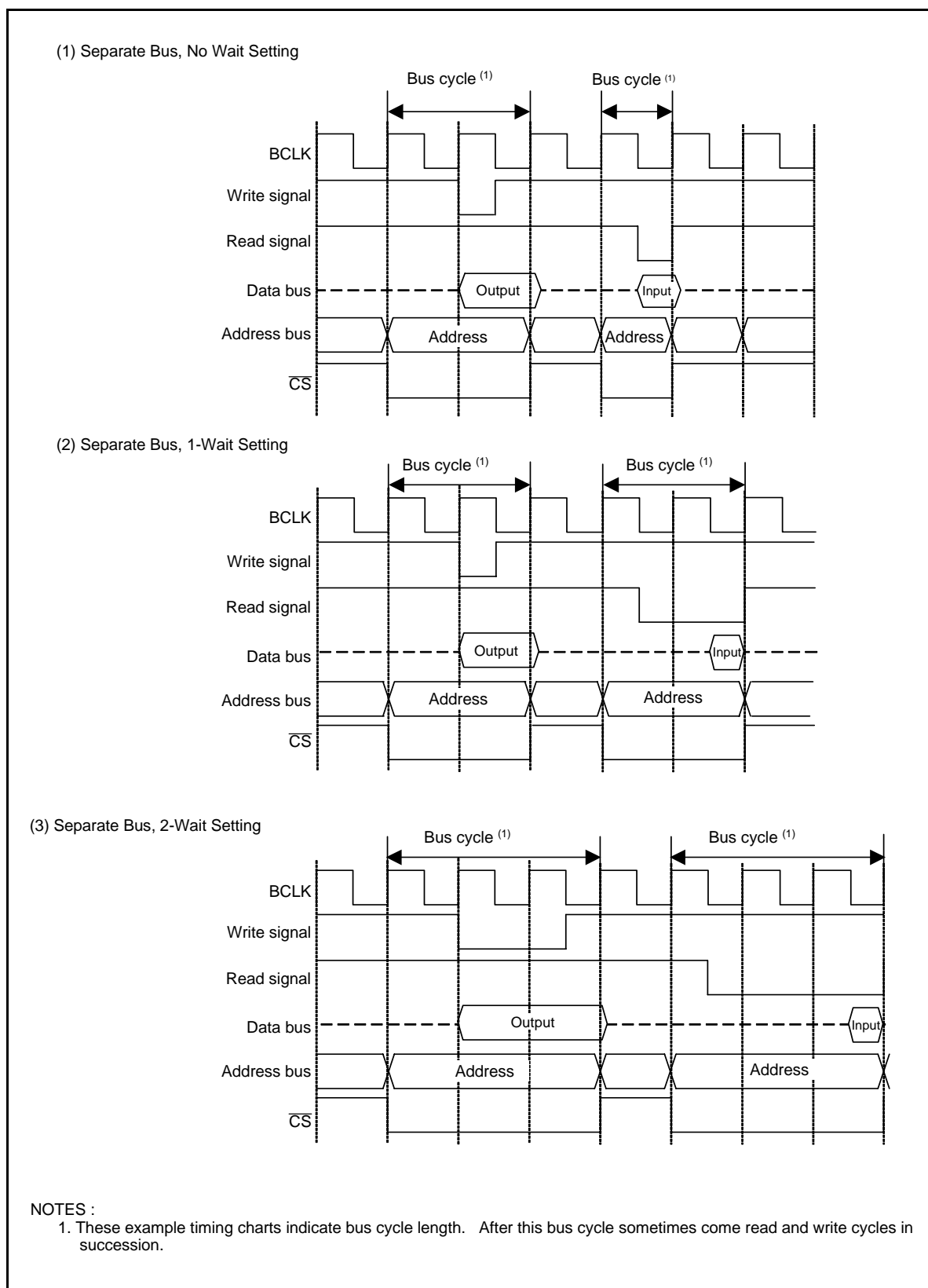


Figure 8.7 Typical Bus Timings Using Software Wait (1)

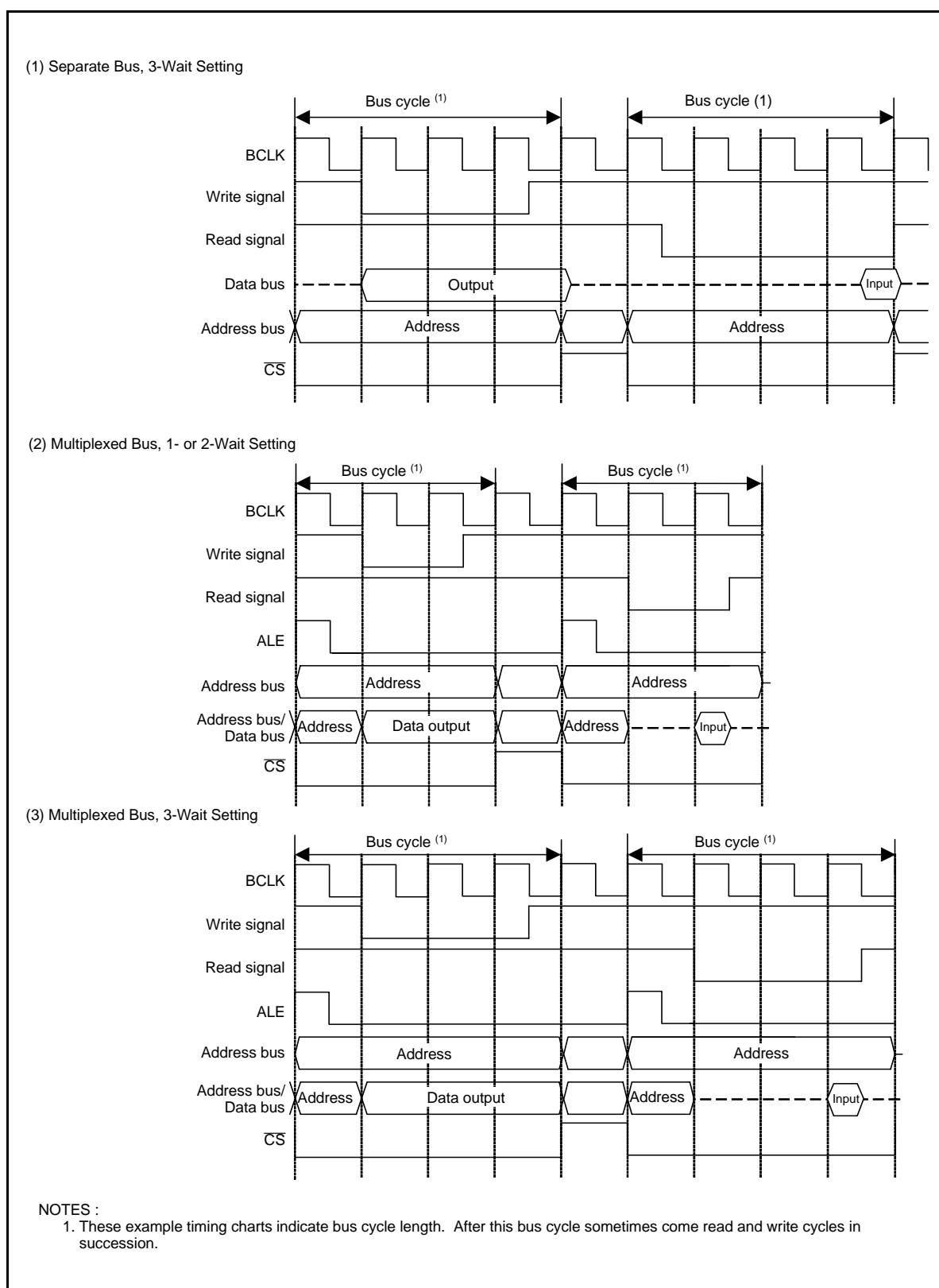


Figure 8.8 Typical Bus Timings Using Software Wait (2)

## 9. Memory Space Expansion Function

### Note

The M16C/62P (80-pin version) and M16C/62PT do not use the memory space expansion function.

The following describes a memory space extension function.

During memory expansion or microprocessor mode, the memory space expansion function allows the access space to be expanded using the appropriate register bits.

Table 9.1 shows The Way of Setting Memory Space Expansion Function, Memory Space.

**Table 9.1 The Way of Setting Memory Space Expansion Function, Memory Space**

Memory Space Expansion Function	How to Set (PM15 to PM14)	Memory Space
1-Mbyte Mode	00b	1 Mbyte (no expansion)
4-Mbyte Mode	11b	4 Mbytes

### 9.1 1-Mbyte Mode

In this mode, the memory space is 1 Mbytes. In 1-Mbyte mode, the external area to be accessed is specified using the  $\overline{CSi}$  ( $i = 0$  to 3) signals (hereafter referred to as the  $\overline{CSi}$  area). Figures 9.2 to 9.3 show the Memory Mapping and  $\overline{CS}$  Area in 1-Mbyte mode.

### 9.2 4-Mbyte Mode

In this mode, the memory space is 4 Mbytes. Figure 9.1 shows the DBR Register. The BSR2 to BSR0 bits in the DBR register select a bank number which is to be accessed to read or write data. Setting the OFS bit to "1" (with offset) allows the accessed address to be offset by 40000h.

In 4-Mbyte mode, the  $\overline{CSi}$  ( $i=0$  to 3) pin functions differently for each area to be accessed.

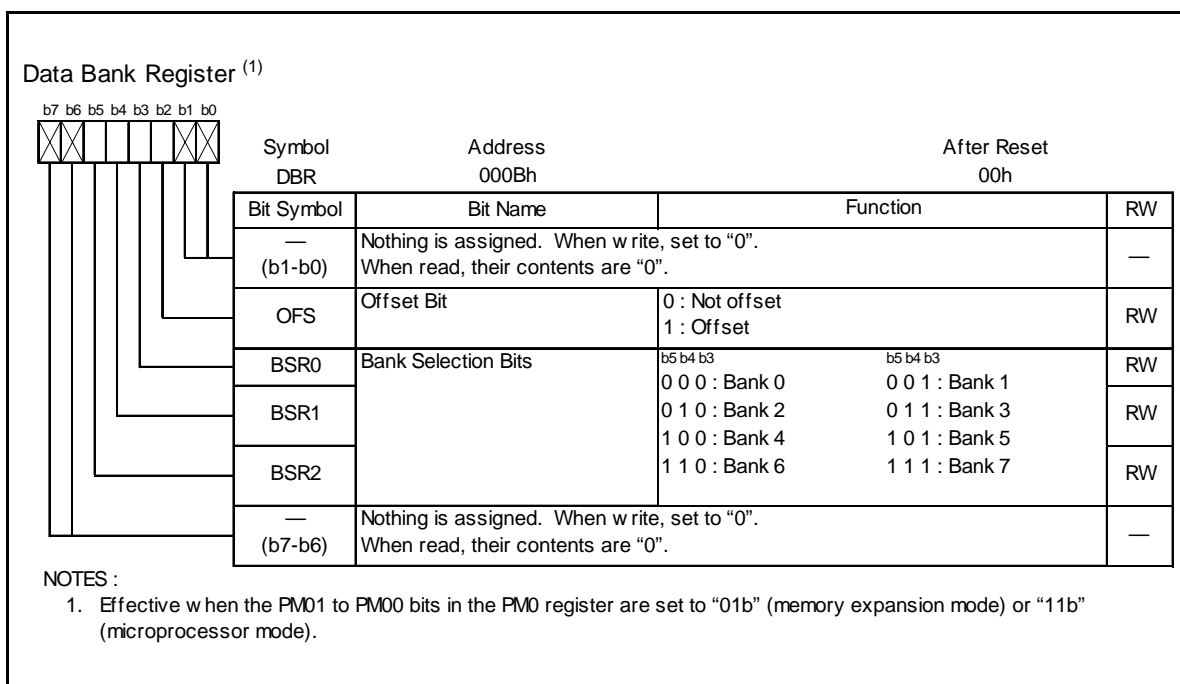
#### 9.2.1 9.2.1 Addresses 04000h to 3FFFFh, C0000h to FFFFFh

- The  $\overline{CSi}$  signal is output from the  $\overline{CSi}$  pin (same operation as 1-Mbyte mode. However, the last address of  $\overline{CSi}$  area is 3FFFFh).

#### 9.2.2 9.2.2 Addresses 40000h to BFFFFh

- The  $\overline{CS0}$  pin outputs "L"
- The  $\overline{CS1}$  to  $\overline{CS3}$  pins output the value of setting as the BSR2 to BSR0 bits (bank number)

Figures 9.4 to 9.5 show the Memory Mapping and  $\overline{CS}$  Area in 4-Mbyte mode. Note that banks 0 to 6 are data-only areas. Locate the program in bank 7 or the  $\overline{CSi}$  area.

**Figure 9.1 DBR Register**



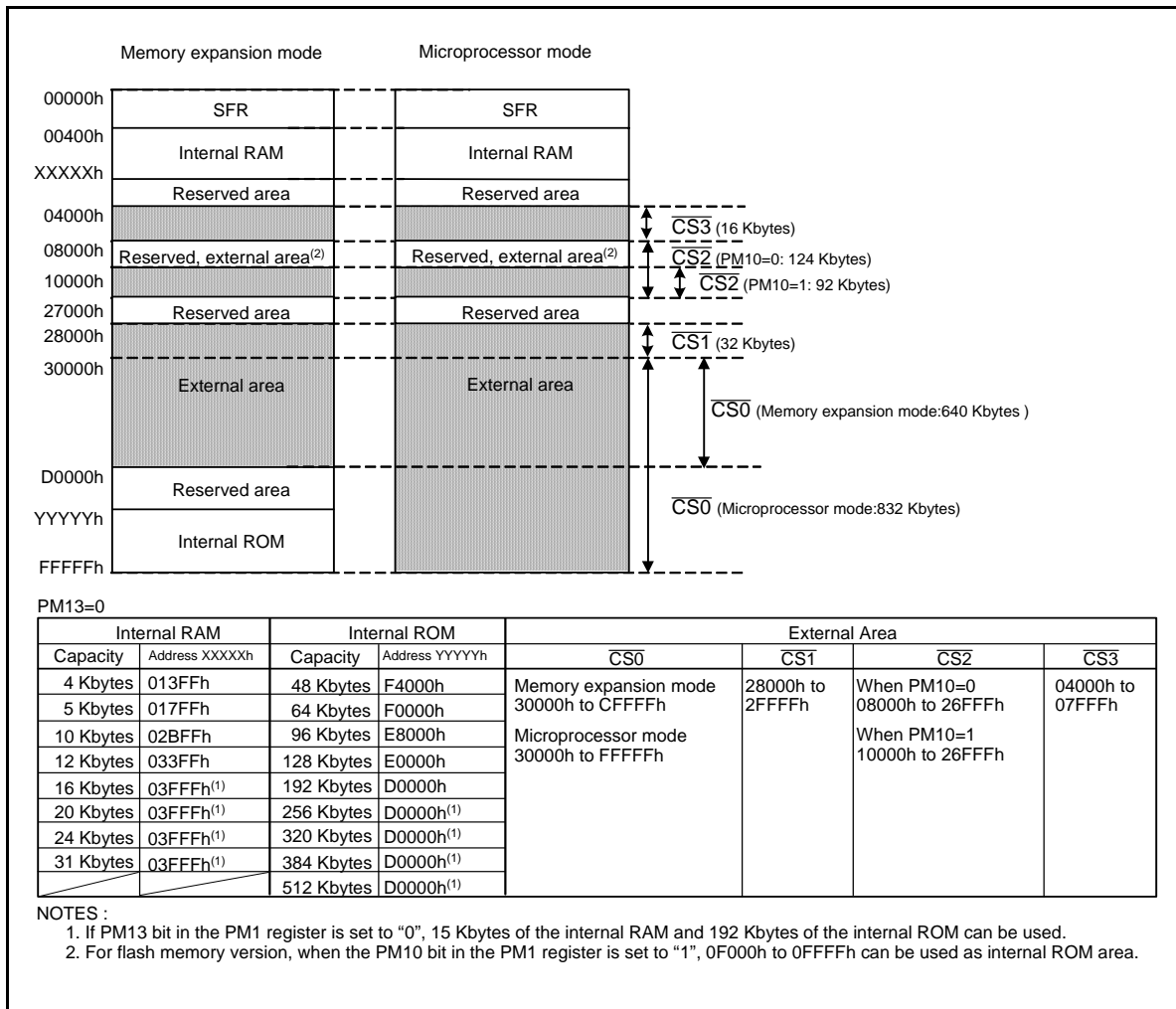


Figure 9.2 Memory Mapping and CS Area in 1-Mbyte mode (PM13=0)

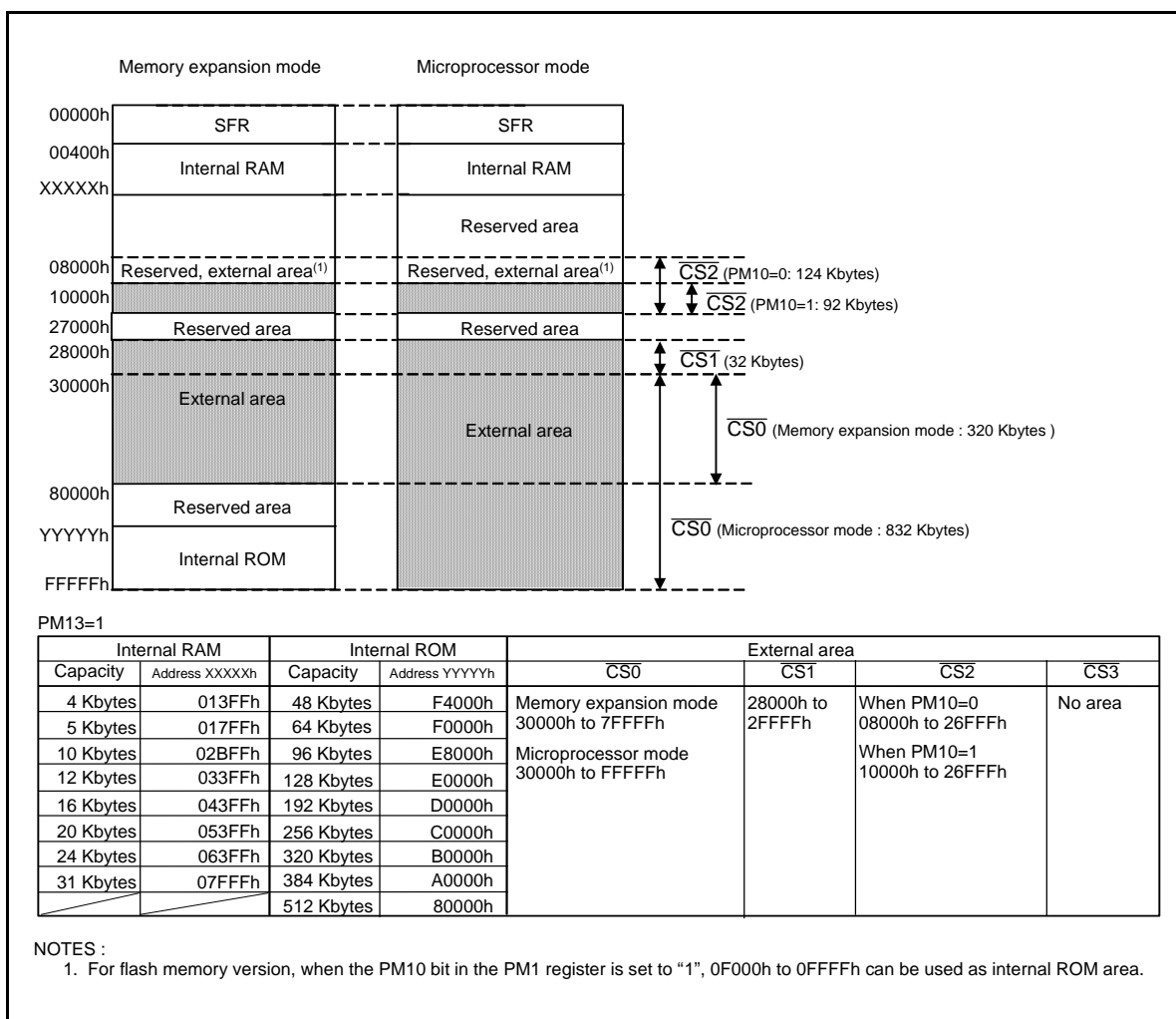


Figure 9.3 Memory Mapping and CS Area in 1-Mbyte mode (PM13=1)

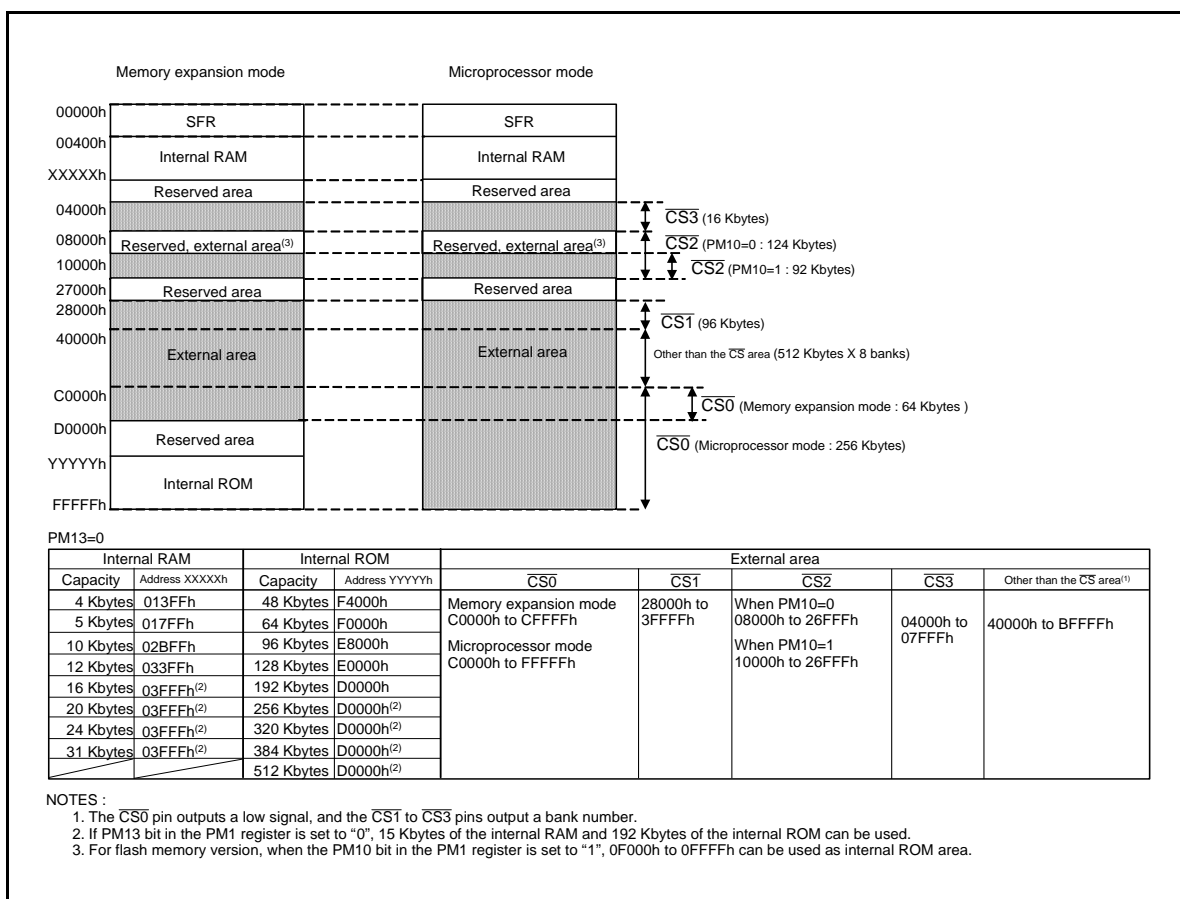


Figure 9.4 Memory Mapping and CS Area in 4-Mbyte mode (PM13=0)

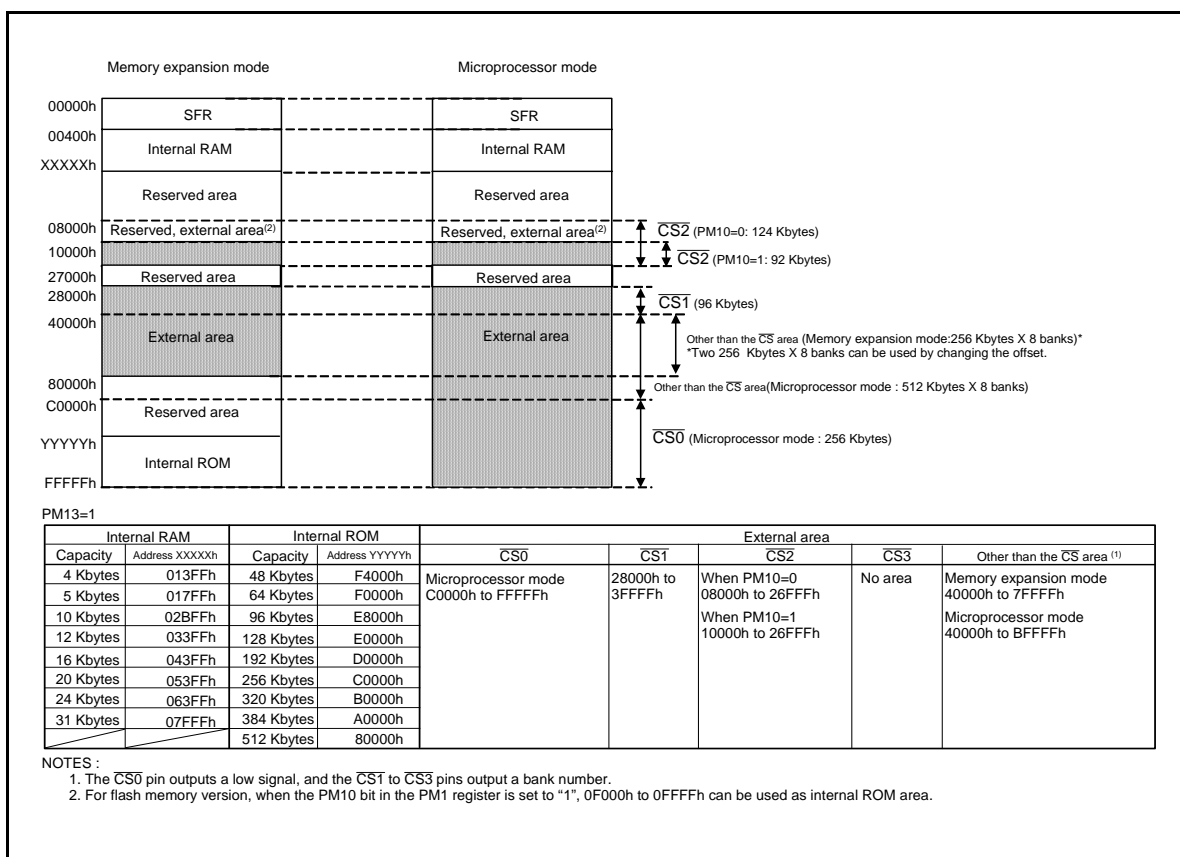


Figure 9.5 Memory Mapping and CS Area in 4-Mbyte mode (PM13=1)

Figure 9.6 shows the External Memory Connect Example in 4-Mbyte Mode.

In this example, the  $\overline{CS}$  pin of 4-Mbyte ROM is connected to the  $\overline{CS0}$  pin of microcomputer. The 4 Mbyte ROM address input AD21, AD20 and AD19 pins are connected to the  $\overline{CS3}$ ,  $\overline{CS2}$  and  $\overline{CS1}$  pins of microcomputer, respectively. The address input AD18 pin is connected to the A19 pin of microcomputer. Figures Figure 9.7 to 9.9 show the Relationship of Addresses Between the 4-Mbyte ROM and the Microcomputer for the Case of a Connection Example in Figure 9.6.

In microprocessor mode, or in memory expansion mode where the PM13 bit in the PM1 register is "0", banks are located every 512 Kbytes. Setting the OFS bit in the DBR register to "1" (offset) allows the accessed address to be offset by 40000h, so that even the data overlapping a bank boundary can be accessed in succession.

In memory expansion mode where the PM13 bit is "1," each 512-Kbyte bank can be accessed in 256 Kbyte units by switching them over with the OFS bit.

Because the SRAM can be accessed on condition that the chip select signals  $S2 = H$  and  $S1 = L$ ,  $\overline{CS0}$  and  $\overline{CS2}$  can be connected to  $S2$  and  $\overline{S1}$ , respectively. If the SRAM does not have the input pins to accept "H" active and "L" active chip select signals( $\overline{S1}$ ,  $S2$ ),  $\overline{CS0}$  and  $\overline{CS2}$  should be decoded external to the chip.

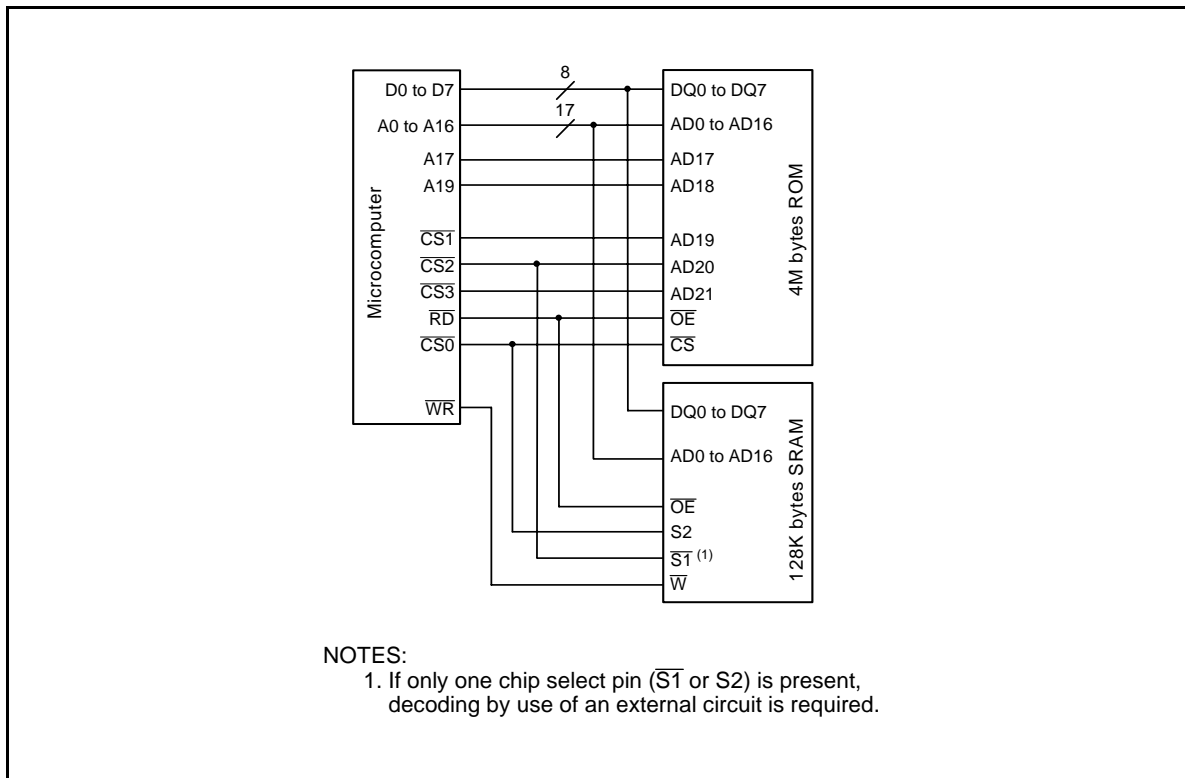
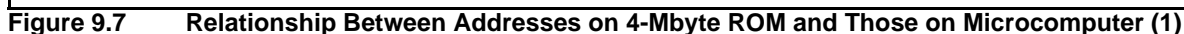


Figure 9.6 External Memory Connect Example in 4-Mbyte Mode



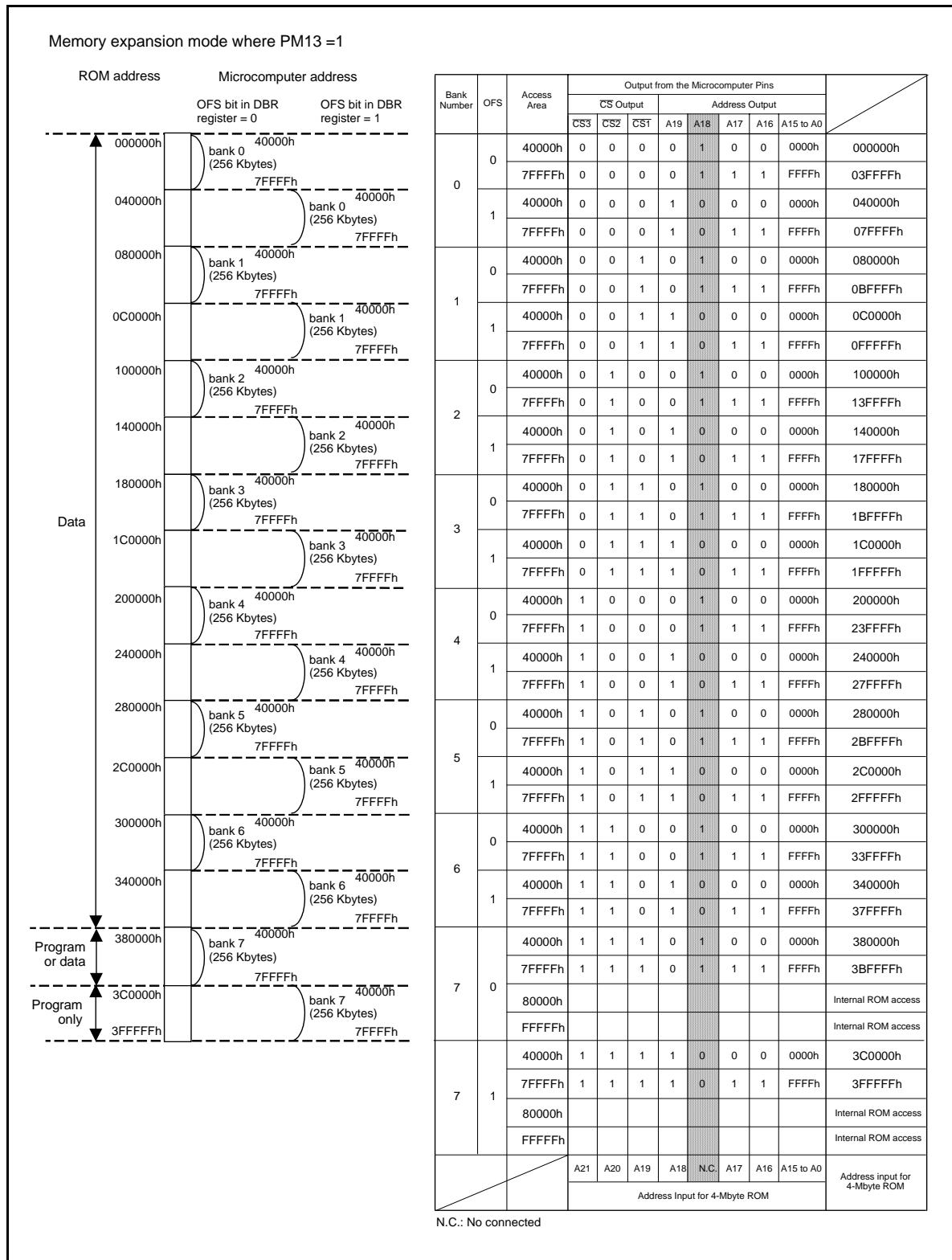


Figure 9.8 Relationship Between Addresses on 4-Mbyte ROM and Those on Microcomputer (2)

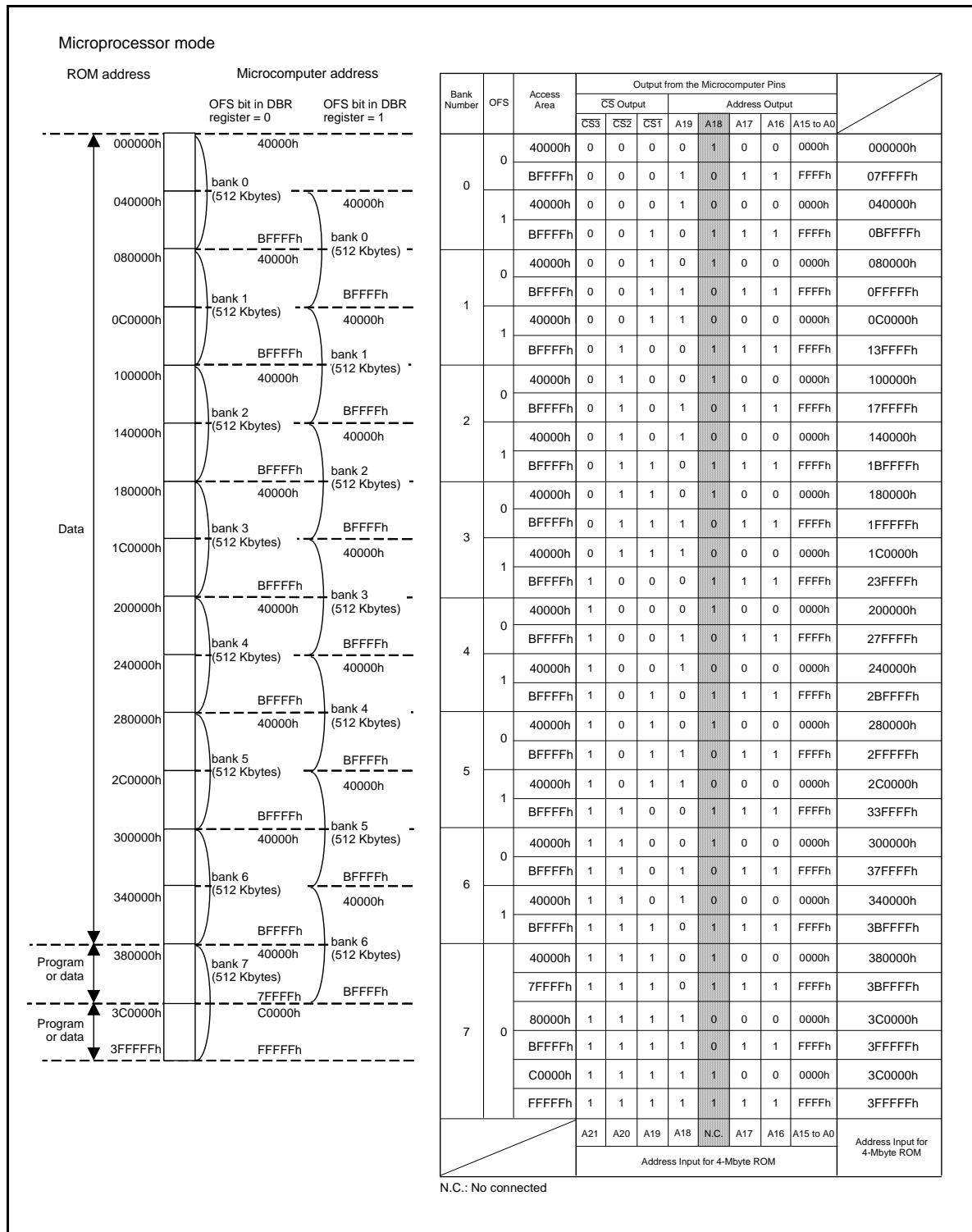


Figure 9.9 Relationship Between Addresses on 4-Mbyte ROM and Those on Microcomputer (3)



## 10. Clock Generation Circuit

### 10.1 Types of the Clock Generation Circuit

4 circuits are incorporated to generate the system clock signal :

- Main clock oscillation circuit
- Sub clock oscillation circuit
- On-chip oscillator
- PLL frequency synthesizer

Table 10.1 lists the Clock Generation Circuit Specifications. Figure 10.1 shows the Clock Generation Circuit. Figures 10.2 to 10.6 show the clock-related registers.

**Table 10.1 Clock Generation Circuit Specifications**

Item	Main Clock Oscillation Circuit	Sub Clock Oscillation Circuit	On-chip oscillator	PLL frequency synthesizer
Use of Clock	<ul style="list-style-type: none"> <li>• CPU clock source</li> <li>• Peripheral function clock source</li> </ul>	<ul style="list-style-type: none"> <li>• CPU clock source</li> <li>• Timer A, B's clock source</li> </ul>	<ul style="list-style-type: none"> <li>• CPU clock source</li> <li>• Peripheral function clock source</li> <li>• CPU and peripheral function clock sources when the main clock stops oscillating</li> </ul>	<ul style="list-style-type: none"> <li>• CPU clock source</li> <li>• Peripheral function clock source</li> </ul>
Clock Frequency	0 to 16 MHz	32.768 kHz	About 1 MHz	10 to 24MHz
Usable Oscillator	<ul style="list-style-type: none"> <li>• Ceramic oscillator</li> <li>• Crystal oscillator</li> </ul>	<ul style="list-style-type: none"> <li>• Crystal oscillator</li> </ul>	–	–
Pins to Connect Oscillator	XIN, XOUT	XCIN, XCOUT	–	–
Oscillation Stop, Restart Function	Presence	Presence	Presence	Presence
Oscillator Status After Reset	Oscillating	Stopped	Stopped	Stopped
Other	Externally derived clock can be input		–	–

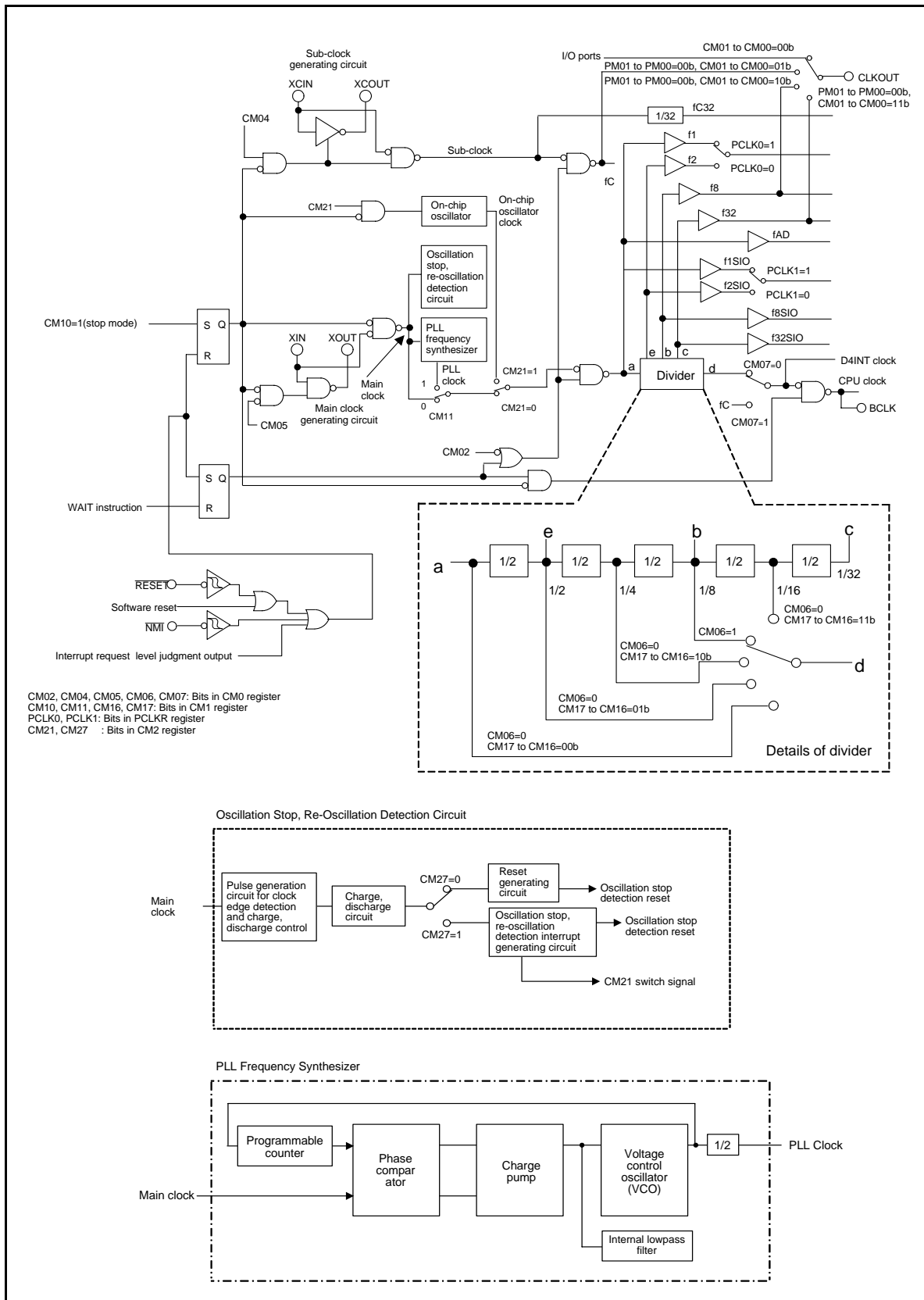
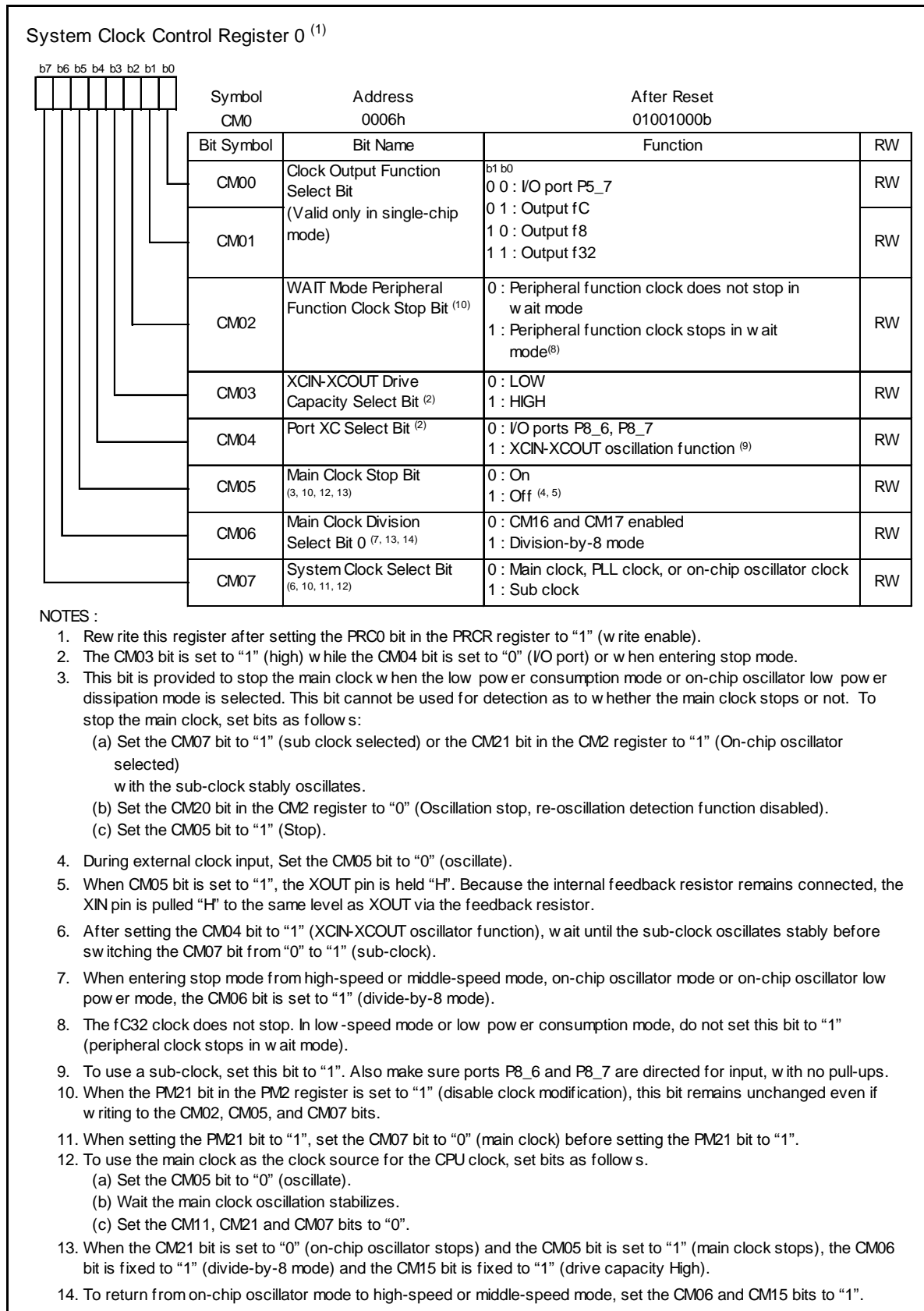
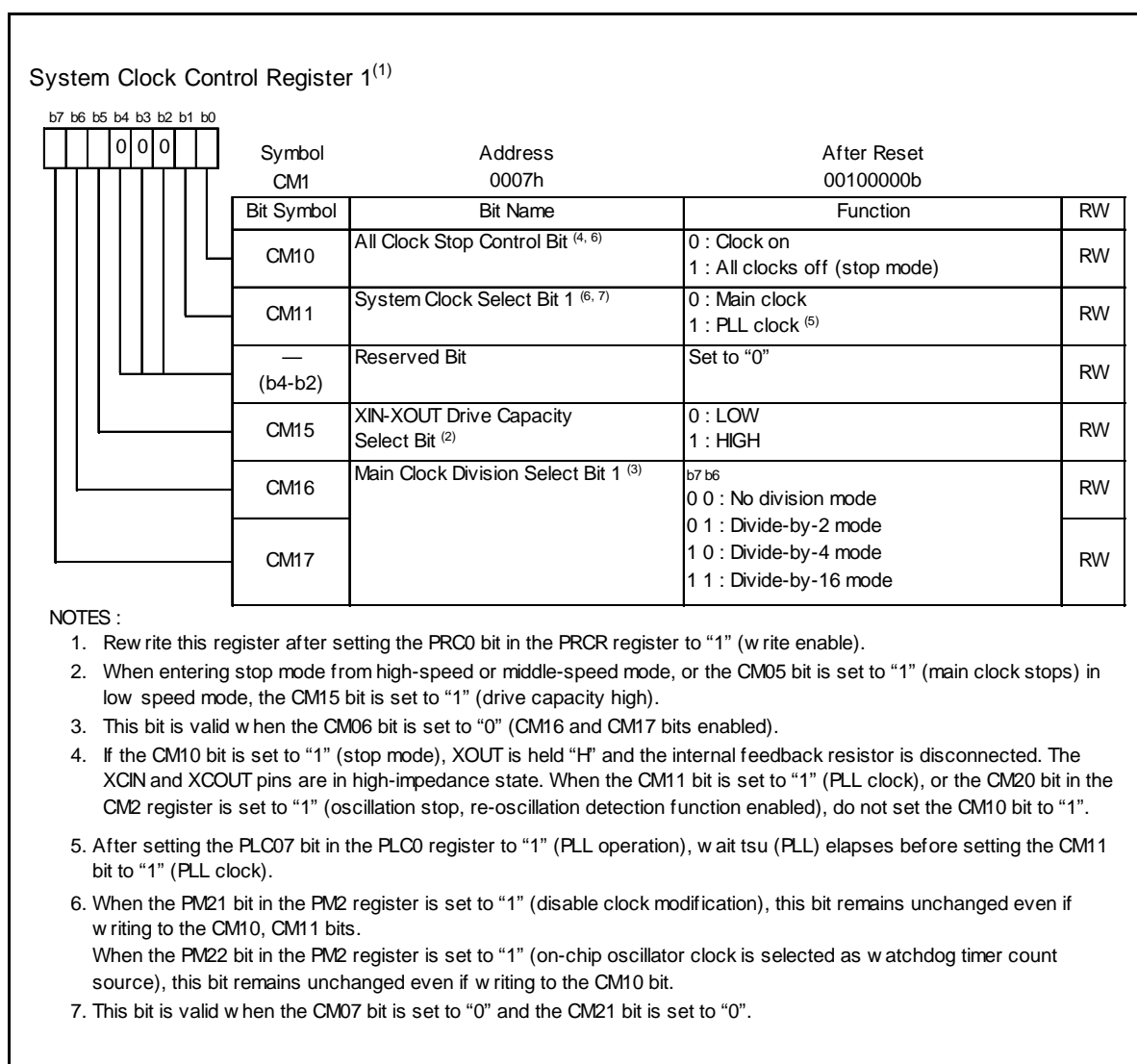
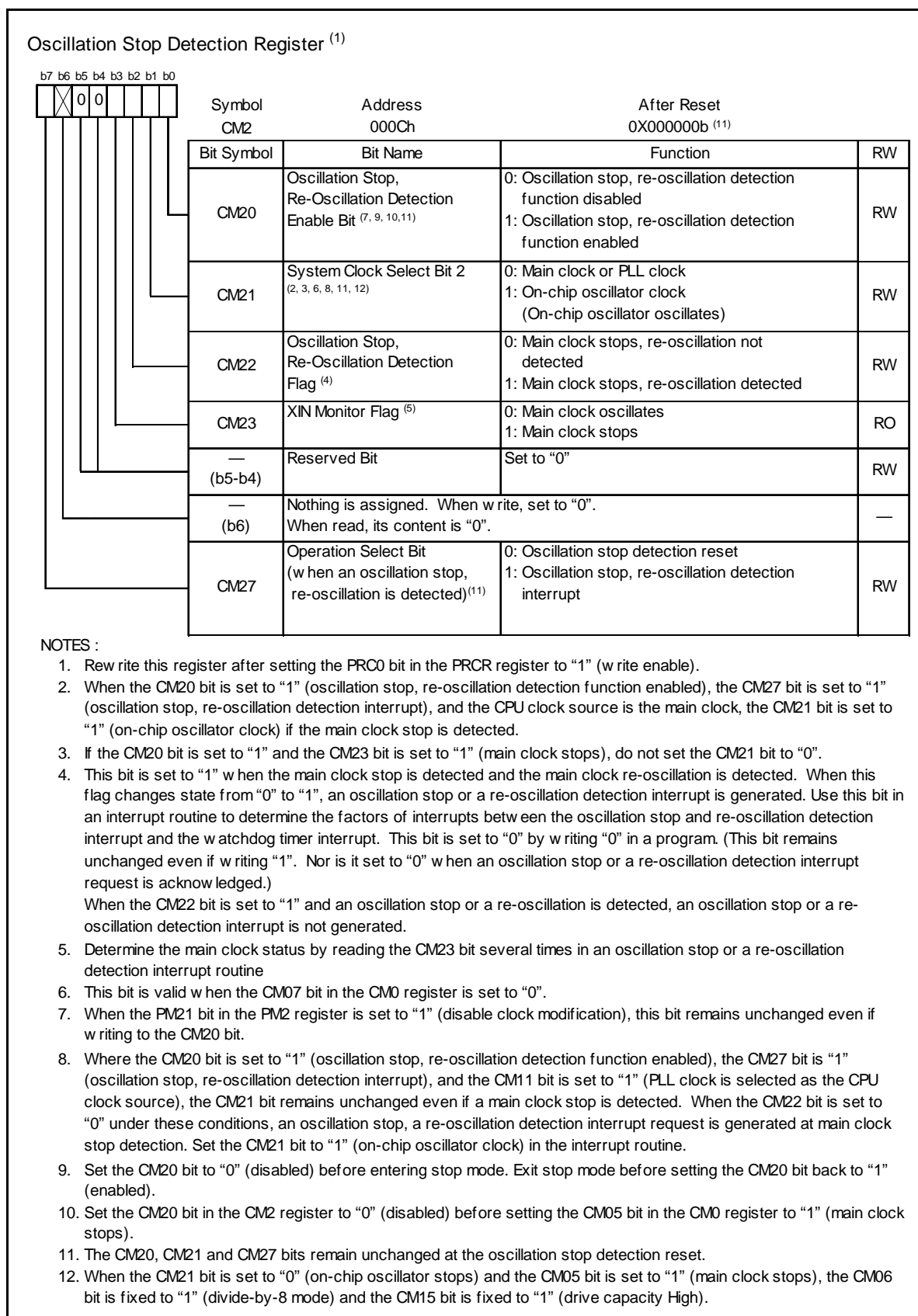


Figure 10.1 Clock Generation Circuit

**Figure 10.2 CM0 Register**

**Figure 10.3 CM1 Register**

**Figure 10.4 CM2 Register**

Peripheral Clock Select Register <sup>(1)</sup>

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset	
0	0	0	0	0	0			PCLKR	025Eh	00000011b	
								Bit Symbol	Bit Name	Function	RW
								PCLK0	Timers A, B Clock Select Bit (Clock source for Timers A , B, and the dead timer)	0 : f2 1 : f1	RW
								PCLK1	S/I/O Clock Select Bit (Clock source for UART0 to UART2, S/I/O3, and S/I/O4)	0 : f2SIO 1 : f1SIO	RW
								— (b7-b2)	Reserved bit	Set to “0”	RW

## NOTES :

- Write to this register after setting the PRC0 bit in the PRCR register to "1" (write enable).

Processor Mode Register 2 <sup>(1)</sup>

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset	
X	X	X	0	0				PM2	001Eh	XXX00000b	
								Bit Symbol	Bit Name	Function	RW
								PM20	Specifying Wait when Accessing SFR at PLL Operation <sup>(2)</sup>	0 : 2 waits 1 : 1 waits	RW
								PM21	System Clock Protective Bit <sup>(3, 4)</sup>	0 : Clock is protected by PRCR register 1 : Clock modification disabled	RW
								PM22	WDT Count Source Protective Bit <sup>(3, 5)</sup>	0 : CPU clock is used for the watchdog timer count source 1 : On-chip oscillator clock is used for the watchdog timer count source	RW
								— (b4-b3)	Reserved Bit	Set to “0”	RW
								— (b7-b5)	Nothing is assigned. When write, set to “0”. When read, their contents are indeterminate.		—

## NOTES :

- Write to this register after setting the PRC1 bit in the PRCR register to "1" (write enable).
- The PM20 bit become effective when PLC07 bit in the PLC0 register is set to "1" (PLL on). Change the PM20 bit when the PLC07 bit is set to "0" (PLL off). Set the PM20 bit to "0" (2 waits) when PLL clock > 16MHz.
- Once this bit is set to "1", it cannot be cleared to "0" in a program.
- If the PM21 bit is set to "1", writing to the following bits has no effect:
  - CM02 bit in CM0 register
  - CM05 bit in CM0 register (main clock does not stop)
  - CM07 bit in CM0 register (clock source for the CPU clock does not change)
  - CM10 bit in CM1 register (stop mode is not entered)
  - CM11 bit in CM1 register (clock source for the CPU clock does not change)
  - CM20 bit in CM2 register (oscillation stop, re-oscillation detection function settings do not change)
  - All bits in PLC0 register (PLL frequency synthesizer settings do not change)
 Be aware that the WAIT instruction cannot be executed when the PM21 bit = 1.
- Setting the PM22 bit to "1" results in the following conditions:
  - The on-chip oscillator starts oscillating, and the on-chip oscillator clock becomes the watchdog timer count source.
  - The CM10 bit is disabled against write. (Writing a "1" has no effect, nor is stop mode entered.)
  - The watchdog timer does not stop when in wait mode or hold state.

Figure 10.5 PCLKR Register and PM2 Register

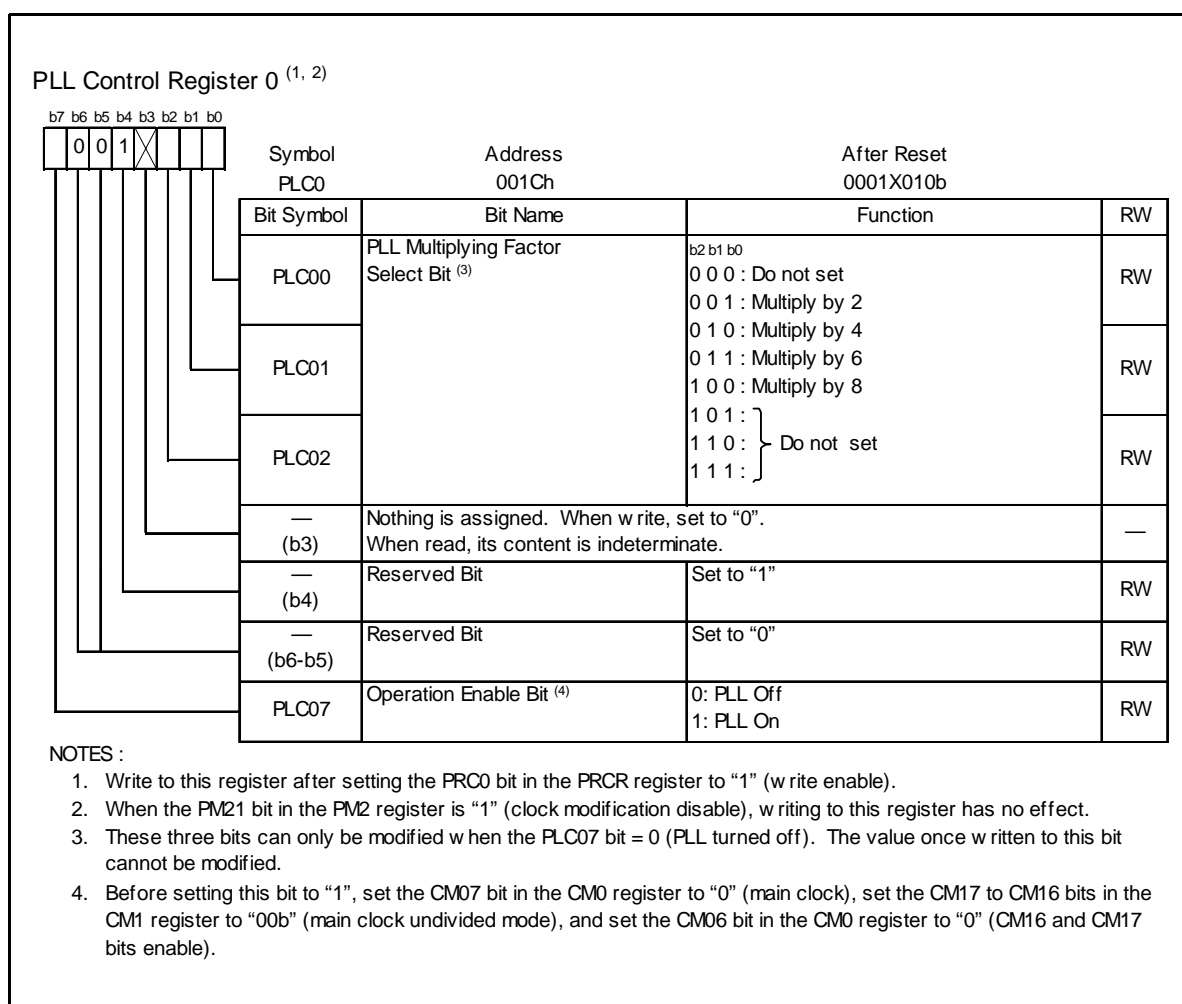


Figure 10.6 PLC0 Register

The following describes the clocks generated by the clock generation circuit.

### 10.1.1 Main Clock

This clock is used as the clock source for the CPU and peripheral function clocks. This clock is used as the clock source for the CPU and peripheral function clocks. The main clock oscillator circuit is configured by connecting a resonator between the XIN and XOUT pins. The main clock oscillator circuit contains a feedback resistor, which is disconnected from the oscillator circuit during stop mode in order to reduce the amount of power consumed in the chip. The main clock oscillator circuit may also be configured by feeding an externally generated clock to the XIN pin. Figure 10.7 shows the Examples of Main Clock Connection Circuit.

After reset, the main clock divided by 8 is selected for the CPU clock.

The power consumption in the chip can be reduced 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 a sub clock or on-chip oscillator clock. In this case, XOUT goes "H". Furthermore, because the internal feedback resistor remains on, XIN is pulled "H" to XOUT via the feedback resistor. Note that if an externally generated clock is fed into the XIN pin, the main clock cannot be turned off by setting the CM05 bit to "1," unless the sub clock is chosen as a CPU clock. If necessary, use an external circuit to turn off the clock.

During stop mode, all clocks including the main clock are turned off. Refer to **10.4 Power Control**.

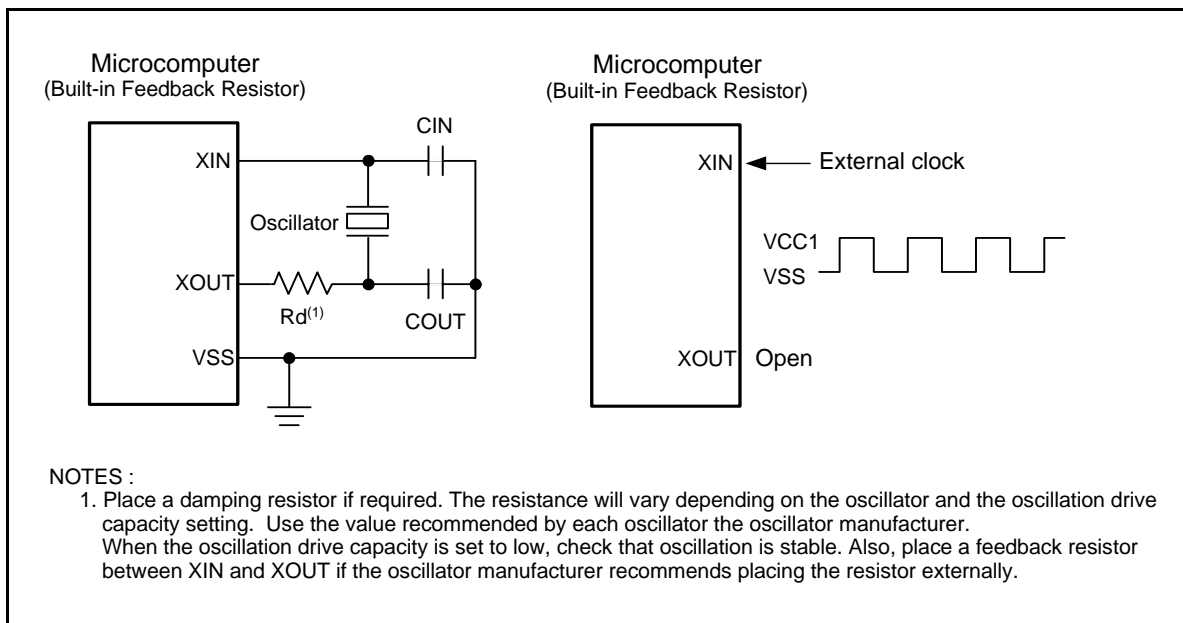


Figure 10.7 Examples of Main Clock Connection Circuit



### 10.1.2 Sub Clock

The sub clock is generated by the sub clock oscillation circuit. This clock is used as the clock source for the CPU clock, as well as the timer A and timer B count sources. In addition, an fc clock with the same frequency as that of the sub clock can be output from the CLKOUT pin.

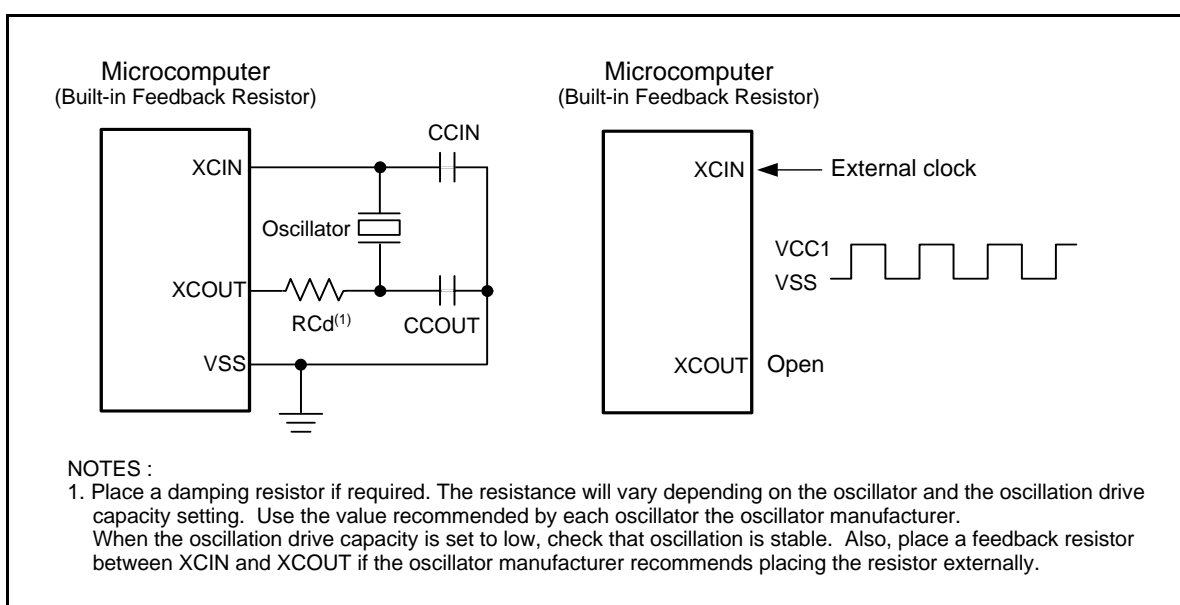
The sub clock oscillator circuit is configured by connecting a crystal resonator between the XCIN and XCOUT pins. The sub clock oscillator circuit contains a feedback resistor, which is disconnected from the oscillator circuit during stop mode in order to reduce the amount of power consumed in the chip. The sub clock oscillator circuit may also be configured by feeding an externally generated clock to the XCIN pin.

Figure 10.8 shows the Examples of Sub Clock Connection Circuit.

After reset, the sub clock is turned off. At this time, the feedback resistor is disconnected from the oscillator circuit.

To use the sub clock for the CPU clock, set the CM07 bit in the CM0 register to "1" (sub clock) after the sub clock becomes oscillating stably.

During stop mode, all clocks including the sub clock are turned off. Refer to **10.4 Power Control**.



**Figure 10.8 Examples of Sub Clock Connection Circuit**

### 10.1.3 On-chip Oscillator Clock

This clock, approximately 1MHz, is supplied by a on-chip oscillator. This clock is used as the clock source for the CPU and peripheral function clocks. In addition, if the PM22 bit in the PM2 register is “1” (on-chip oscillator clock for the watchdog timer count source), this clock is used as the count source for the watchdog timer (Refer to **13.1 Count source protective mode**).

After reset, the on-chip oscillator is turned off. It is turned on by setting the CM21 bit in the CM2 register to “1” (on-chip oscillator clock), and is used as the clock source for the CPU and peripheral function clocks, in place of the main clock. If the main clock stops oscillating when the CM20 bit in the CM2 register is “1” (oscillation stop, re-oscillation detection function enabled) and the CM27 bit is “1” (oscillation stop, re-oscillation detection interrupt), the on-chip oscillator automatically starts operating, supplying the necessary clock for the microcomputer.

### 10.1.4 PLL Clock

The PLL clock is generated PLL frequency synthesizer. This clock is used as the clock source for the CPU and peripheral function clocks. After reset, the PLL clock is turned off. The PLL frequency synthesizer is activated by setting the PLC07 bit to “1” (PLL operation). When the PLL clock is used as the clock source for the CPU clock, wait tsu(PLL) for the PLL clock to be stable, and then set the CM11 bit in the CM1 register to “1”.

Before entering wait mode or stop mode, be sure to set the CM11 bit to “0” (CPU clock source is the main clock). Furthermore, before entering stop mode, be sure to set the PLC07 bit in the PLC0 register to “0” (PLL stops). Figure 10.9 shows the Procedure to Use PLL Clock as CPU Clock Source.

The PLL clock frequency is determined by the equation below. When the PLL clock frequency is 16 MHz or more, set the PM20 bit in the PM2 register to “0” (2 waits).

$$\text{PLL clock frequency} = f(\text{XIN}) \times (\text{multiplying factor set by the PLC02 to PLC00 bits in the PLC0 register})$$

(However, 10 MHz PLL clock frequency 24 MHz)

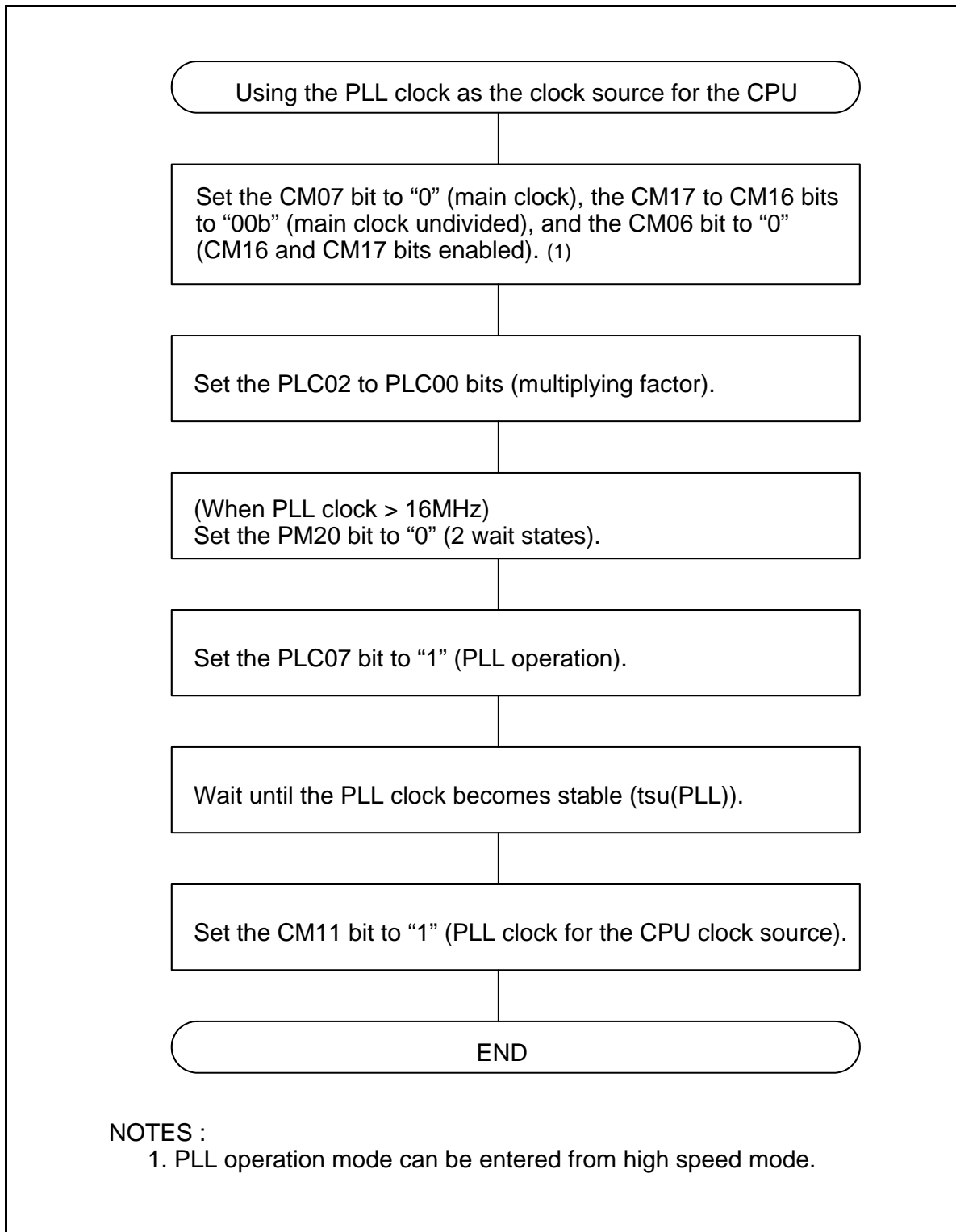
The PLC02 to PLC00 bits can be set only once after reset. Table 10.2 shows the Example for Setting PLL Clock Frequencies.

**Table 10.2 Example for Setting PLL Clock Frequencies**

XIN (MHz)	PLC02	PLC01	PLC00	Multiplying Factor	PLL Clock (MHz) <sup>(1)</sup>
10	0	0	1	2	20
5	0	1	0	4	
3.33	0	1	1	6	
2.5	1	0	0	8	
12	0	0	1	2	24
6	0	1	0	4	
4	0	1	1	6	
3	1	0	0	8	

NOTES:

1.  $10\text{MHz} \leq \text{PLL clock frequency} \leq 24\text{MHz}$ .

**Figure 10.9 Procedure to Use PLL Clock as CPU Clock Source**

## 10.2 CPU Clock and Peripheral Function Clock

Two type clocks: CPU clock to operate the CPU and peripheral function clocks to operate the peripheral functions.

### 10.2.1 CPU Clock and BCLK

These are operating clocks for the CPU and watchdog timer.

The clock source for the CPU clock can be chosen to be the main clock, sub clock, on-chip oscillator clock or the PLL clock.

If the main clock or on-chip oscillator clock is selected as the clock source for the CPU clock, the selected clock source can be divided by 1 (undivided), 2, 4, 8 or 16 to produce the CPU clock. Use the CM06 bit in CM0 register and the CM17 to CM16 bits in the CM1 register to select the divide-by-n value.

When the PLL clock is selected as the clock source for the CPU clock, the CM06 bit should be set to "0" and the CM17 to CM16 bits to "00b" (undivided).

After reset, the main clock divided by 8 provides the CPU clock.

During memory expansion or microprocessor mode, a BCLK signal with the same frequency as the CPU clock can be output from the BCLK pin by setting the PM07 bit in the PM0 register to "0" (output enabled).

Note that when entering stop mode from high or middle speed mode, on-chip oscillator mode or on-chip oscillator low power dissipation mode, or when the CM05 bit in the CM0 register is set to "1" (main clock turned off) in low-speed mode, the CM06 bit in the CM0 register is set to "1" (divide-by-8 mode).

### 10.2.2 Peripheral Function Clock (f1, f2, f8, f32, f1SIO, f2SIO, f8SIO, f32SIO, fAD, fC32)

These are operating clocks for the peripheral functions.

Of these, fi (i = 1, 2, 8, 32) and fiSIO are derived from the main clock, PLL clock or on-chip oscillator clock by dividing them by i. The clock fi is used for timers A and B, and fiSIO is used for serial I/O. The f8 and f32 clocks can be output from the CLKOUT pin.

The fAD clock is produced from the main clock, PLL clock or on-chip oscillator clock, and is used for the A/D converter.

When the WAIT instruction is executed after setting the CM02 bit in the CM0 register to "1" (peripheral function clock turned off during wait mode), or when the microcomputer is in low power dissipation mode, the fi, fiSIO and fAD clocks are turned off.

The fC32 clock is produced from the sub clock, and is used for timers A and B. This clock can be used when the sub clock is on.

## 10.3 Clock Output Function

During single-chip mode, the f8, f32 or fC clock can be output from the CLKOUT pin. Use the CM01 to CM00 bits in the CM0 register to select.

## 10.4 Power Control

Normal operating mode, wait mode and stop mode are provided as the power consumption control. All mode states, except wait mode and stop mode, are called normal operating mode in this document.

### 10.4.1 Normal Operating Mode

Normal operating mode is further classified into seven modes.

In normal operating mode, because the CPU clock and the peripheral function clocks both are on, 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 greater the processing capability. The lower the CPU clock frequency, the smaller the power consumption in the chip. If the unnecessary oscillator circuits are turned off, the power consumption is further reduced.

Before the clock sources for the CPU clock can be switched over, the new clock source to which switched must be oscillating stably. If the new clock source is the main clock, sub clock or PLL clock, allow a sufficient wait time in a program until it becomes oscillating stably.

Note that operating modes cannot be changed directly from low speed or low power dissipation mode to on-chip oscillator or on-chip oscillator low power dissipation mode. Nor can operating modes be changed directly from on-chip oscillator or on-chip oscillator low power dissipation mode to low speed or low power dissipation mode. Where the CPU clock source is changed from the on-chip oscillator to the main clock, change the operating mode to the medium speed mode (divided by 8 mode) after the clock was divided by 8 (the CM06 bit in the CM0 register was set to "1") in the on-chip oscillator mode.

#### 10.4.1.1 High-speed Mode

The main clock divided by 1 provides the CPU clock. If the sub clock is on, fC32 can be used as the count source for Timers A and B.

#### 10.4.1.2 PLL Operating Mode

The main clock multiplied by 2, 4, 6 or 8 provides the PLL clock, and this PLL clock serves as the CPU clock. If the sub clock is on, fC32 can be used as the count source for Timers A and B. PLL operating mode can be entered from high speed mode. If PLL operating mode is to be changed to wait or stop mode, first go to high speed mode before changing.

#### 10.4.1.3 Medium-Speed Mode

The main clock divided by 2, 4, 8 or 16 provides the CPU clock. If the sub clock is on, fC32 can be used as the count source for Timers A and B.

#### 10.4.1.4 Low-Speed Mode

The sub clock provides the CPU clock. The main clock is used as the clock source for the peripheral function clock when the CM21 bit in the CM2 register is set to "0" (on-chip oscillator turned off), and the on-chip oscillator clock is used when the CM21 bit is set to "1" (on-chip oscillator oscillating).

The fC32 clock can be used as the count source for Timers A and B.

#### 10.4.1.5 Low Power Dissipation Mode

In this mode, the main clock is turned off after being placed in low speed mode. The sub clock provides the CPU clock. The fC32 clock can be used as the count source for Timers A and B.

Simultaneously when this mode is selected, the CM06 bit becomes "1" (divided by 8 mode). In the low power dissipation mode, do not change the CM06 bit. Consequently, the medium speed (divided by 8) mode is to be selected when the main clock is operated next

#### 10.4.1.6 On-chip Oscillator Mode

The on-chip oscillator clock divided by 1 (undivided), 2, 4, 8 or 16 provides the CPU clock. The on-chip oscillator clock is also the clock source for the peripheral function clocks. If the sub clock is on, fC32 can be used as the count source for Timers A and B. When the operating mode is returned to the high and medium speed modes, set the CM06 bit in the CM0 register to "1" (divided by 8 mode).

### 10.4.1.7 On-chip Oscillator Low Power Dissipation Mode

The main clock is turned off after being placed in on-chip oscillator mode. The CPU clock can be selected as in the on-chip oscillator mode. The on-chip oscillator clock is the clock source for the peripheral function clocks. If the sub clock is on, fC32 can be used as the count source for Timers A and B.

**Table 10.3 Setting Clock Related Bit and Modes**

Modes		CM2 Register	CM1 Register		CM0 Register			
		CM21	CM11	CM17, CM16	CM07	CM06	CM05	CM04
PLL Operating Mode		0	1	00b	0	0	0	–
High-Speed Mode		0	0	00b	0	0	0	–
Medium-Speed Mode	divided by 2	0	0	01b	0	0	0	–
	divided by 4	0	0	10b	0	0	0	–
	divided by 8	0	0	–	0	1	0	–
	divided by 16	0	0	11b	0	0	0	–
Low-Speed Mode		–	0	–	1	–	0	1
Low Power Dissipation Mode		0	0	–	1	1 <sup>(1)</sup>	1 <sup>(1)</sup>	1
On-chip Oscillator Mode	divided by 1	1	0	00b	0	0	0	–
	divided by 2	1	0	01b	0	0	0	–
	divided by 4	1	0	10b	0	0	0	–
	divided by 8	1	0	–	0	1	0	–
	divided by 16	1	0	11b	0	0	0	–
On-chip Oscillator Low Power Dissipation Mode		1	0	(NOTE 2)	0	(NOTE 2)	1	–

– : “0” or “1”

**NOTES:**

1. When the CM05 bit is set to “1” (main clock turned off) in low-speed mode, the mode goes to low power dissipation mode and CM06 bit is set to “1” (divided by 8 mode) simultaneously.
2. The divide-by-n value can be selected the same way as in on-chip oscillator mode.

### 10.4.2 Wait Mode

In wait mode, the CPU clock is turned off, so are the CPU (because operated by the CPU clock) and the watchdog timer. However, if the PM22 bit in the PM2 register is “1” (on-chip oscillator clock for the watchdog timer count source), the watchdog timer remains active. Because the main clock, sub clock and on-chip oscillator clock all are on, the peripheral functions using these clocks keep operating.

#### 10.4.2.1 Peripheral Function Clock Stop Function

If the CM02 bit in the CM0 register is “1” (peripheral function clocks turned off during wait mode), the f1, f2, f8, f32, f1SIO, f8SIO, f32SIO and fAD clocks are turned off when in wait mode, with the power consumption reduced that much. However, fC32 remains on.

#### 10.4.2.2 Entering Wait Mode

The microcomputer is placed into wait mode by executing the WAIT instruction.

When the CM11 bit = 1 (CPU clock source is the PLL clock), be sure to clear the CM11 bit in the CM1 register to “0” (CPU clock source is the main clock) before going to wait mode. The power consumption of the chip can be reduced by clearing the PLC07 bit in the PLC0 register to “0” (PLL stops).

#### 10.4.2.3 Pin Status During Wait Mode

Table 10.4 lists Pin Status During Wait Mode.

**Table 10.4 Pin Status During Wait Mode**

Pin		Memory Expansion Mode Microprocessor Mode	Single-Chip Mode
A0 to A19, D0 to D15, $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$ , $\overline{\text{BHE}}$		Retains status before wait mode	Does not become a bus control pin
$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WRL}}$ , $\overline{\text{WRH}}$		“H”	
HLDA, BCLK		“H”	
$\overline{\text{ALE}}$		“L”	
I/O ports		Retains status before wait mode	Retains status before wait mode
CLKOUT	When fC selected	Does not become a CLKOUT pin	Does not stop
	When 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 maintained.

#### 10.4.2.4 Exiting Wait Mode

The microcomputer is moved out of wait mode by a hardware reset,  $\overline{\text{NMI}}$  interrupt, low voltage detection interrupt or peripheral function interrupt.

If the microcomputer is to be moved out of exit wait mode by a hardware reset,  $\overline{\text{NMI}}$  interrupt or low voltage detection interrupt, set the peripheral function interrupt priority ILVL2 to ILVL0 bits to “000b” (interrupts disabled) before executing the WAIT instruction.

The peripheral function interrupts are affected by the CM02 bit. If CM02 bit is “0” (peripheral function clocks not turned off during wait mode), peripheral function interrupts can be used to exit wait mode. If CM02 bit is “1” (peripheral function clocks turned off during wait mode), the peripheral functions using the peripheral function clocks stop operating, so that only the peripheral functions clocked by external signals can be used to exit wait mode.

**Table 10.5 Interrupts to Exit Wait Mode and Use Conditions**

Interrupt	CM02=0	CM02=1
$\overline{\text{NMI}}$ Interrupt	Can be used	Can be used
Serial Interface Interrupt	Can be used when operating with internal or external clock	Can be used when operating with external clock
Key Input Interrupt	Can be used	Can be used
A/D Conversion Interrupt	Can be used in one-shot mode or single sweep mode	—(Do not use)
Timer A Interrupt Timer B Interrupt	Can be used in all modes	Can be used in event counter mode or when the count source is fC32
$\overline{\text{INT}}$ Interrupt	Can be used	Can be used
Low Voltage Detection Interrupt	Can be used	Can be used

Table 10.5 lists the Interrupts to Exit Wait Mode and Use Conditions.

If the microcomputer is to be moved out of wait mode by a peripheral function interrupt, set up the following before executing the WAIT instruction.

- (1) Set the ILVL2 to ILVL0 bits in the interrupt control register, for peripheral function interrupts used to exit wait mode.  
The ILVL2 to ILVL0 bits in all other interrupt control registers, for peripheral function interrupts not used to exit wait mode, are set to “000b” (interrupt disable).
- (2) Set the I flag to “1”.
- (3) Start operating the peripheral functions used to exit wait mode.  
When the peripheral function interrupt is used, an interrupt routine is performed as soon as an interrupt request is acknowledged and the CPU clock is supplied again.

When the microcomputer exits wait mode by the peripheral function interrupt, the CPU clock is the same clock as the CPU clock executing the WAIT instruction.



### 10.4.3 Stop Mode

In stop mode, all oscillator circuits are turned off, so are the CPU clock and the peripheral function clocks. Therefore, the CPU and the peripheral functions clocked by these clocks stop operating. The least amount of power is consumed in this mode. If the voltage applied to VCC1 and VCC2 pins is VRAM or more, the internal RAM is retained. When applying 2.7 or less voltage to VCC1 and VCC2 pins, make sure  $VCC1 \geq VCC2 \geq VRAM$ .

However, the peripheral functions clocked by external signals keep operating. The following interrupts can be used to exit stop mode. Table 10.6 lists Interrupts to Stop Mode and Use Conditions

**Table 10.6 Interrupts to Stop Mode and Use Conditions**

Interrupt	Condition
NMI Interrupt	Can be used
Key Input Interrupt	Can be used
INT Interrupt	Can be used
Timer A Interrupt Timer B Interrupt	Can be used (when counting external pulses in event counter mode)
Serial Interface Interrupt	Can be used (when external clock is selected)
Low Voltage Detection Interrupt	Can be used (Refer to <b>6.1 Low Voltage Detection Interrupt</b> for an Operating Condition)

#### 10.4.3.1 Entering Stop Mode

The microcomputer is placed into 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 is set to “1” (divide-by-8 mode) and the CM15 bit in the CM1 register is set to “1” (main clock oscillator circuit drive capability high).

Before entering stop mode, set the CM20 bit in the CM2 register to “0” (oscillation stop, re-oscillation detection function disable).

Also, if the CM11 bit in the CM1 register is “1” (PLL clock for the CPU clock source), set the CM11 bit to “0” (main clock for the CPU clock source) and the PLC07 bit in the PLC0 register to “0” (PLL turned off) before entering stop mode.

#### 10.4.3.2 Pin Status in Stop Mode

Table 10.7 lists Pin Status in Stop Mode.

**Table 10.7 Pin Status in Stop Mode**

Pin		Memory Expansion Mode Microprocessor Mode	Single-Chip Mode
A0 to A19, D0 to D15, $\overline{CS0}$ to $\overline{CS3}$ , BHE		Retains status before stop mode	Does not become a bus control pin
$\overline{RD}$ , $\overline{WR}$ , $\overline{WRL}$ , $\overline{WRH}$		“H”	
HLDA, BCLK		“H”	
ALE		indeterminate	
I/O ports		Retains status before stop mode	Retains status before stop mode
CLKOUT	When fC selected	Does not become a CLKOUT pin	“H”
	When f8, f32 selected		Retains status before stop mode

### 10.4.3.3 Exiting Stop Mode

Stop mode is exited by a hardware reset,  $\overline{\text{NMI}}$  interrupt, low voltage detection interrupt or peripheral function interrupt.

When the hardware reset,  $\overline{\text{NMI}}$  interrupt or low voltage detection interrupt is used to exit stop mode, set all ILVL2 to ILVL0 bits in the interrupt control registers for the peripheral function interrupt to “000b” (interrupt disabled) before setting the CM10 bit to “1”.

When the peripheral function interrupt is used to exit stop mode, set the CM10 bit to “1” after the following settings are completed.

- (1) Set the ILVL2 to ILVL0 bits in the interrupt control registers to decide the peripheral priority level of the peripheral function interrupt.

Set the interrupt priority levels of the interrupts, not being used to exit stop mode, to “0” by setting the all ILVL2 to ILVL0 bits to “000b”.

- (2) Set the I flag to “1”.

- (3) Start operation of peripheral function being used to exit wait mode.

When exiting stop mode by the peripheral function interrupt, the interrupt routine is performed when an interrupt request is generated and the CPU clock is supplied again.

When stop mode is exited by the peripheral function interrupt or NMI interrupt, the CPU clock source is as follows, in accordance with the CPU clock source setting before the microcomputer had entered stop mode.

- When the sub clock is the CPU clock before entering stop mode : Sub clock
- When the main clock is the CPU clock source before entering stop mode : Main clock divided by 8
- When the on-chip oscillator clock is the CPU clock source before entering stop mode : On-chip oscillator clock divided by 8

Figure 10.10 shows the State Transition from Normal Operating Mode to Stop Mode and Wait Mode. Figure 10.11 shows the State Transition in Normal Operating Mode.

Table 10.8 shows a state transition matrix describing Allowed Transition and Setting. The vertical line shows current state and horizontal line shows state after transition.

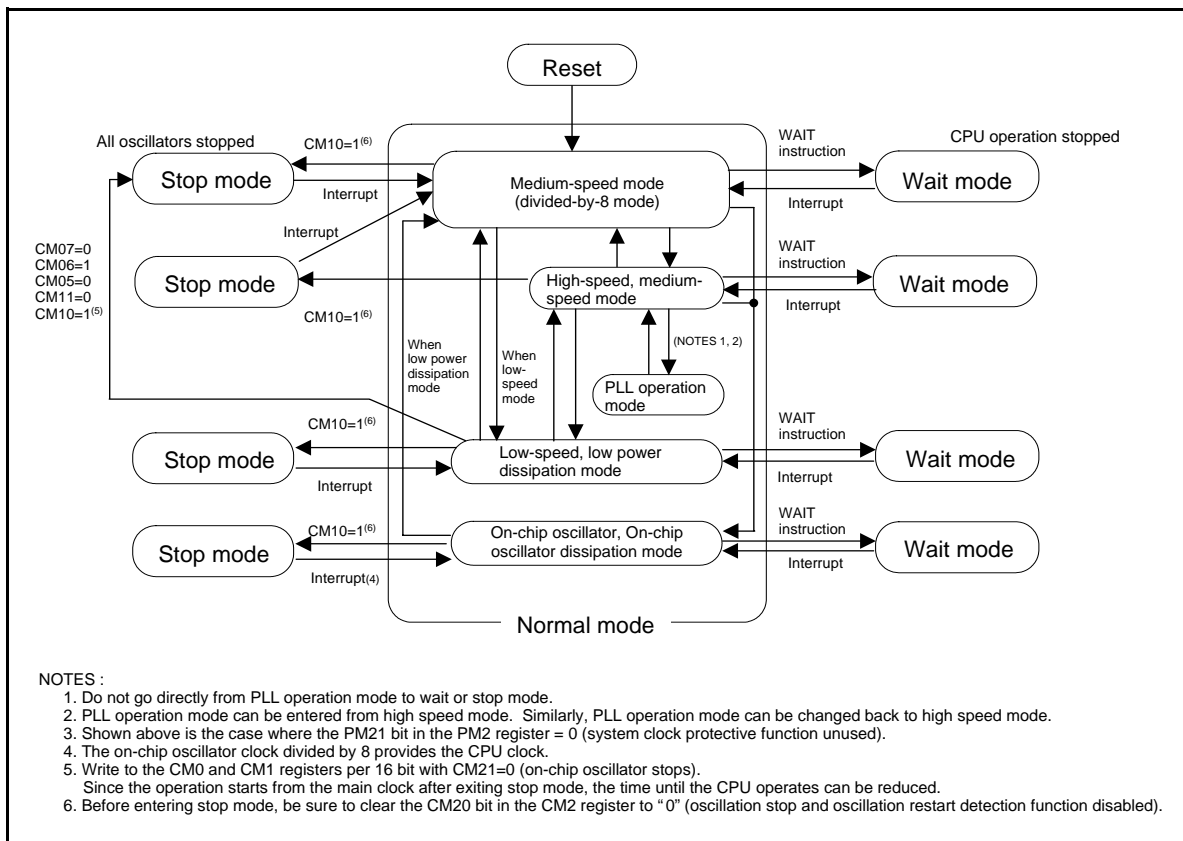


Figure 10.10 State Transition to Stop Mode and Wait Mode

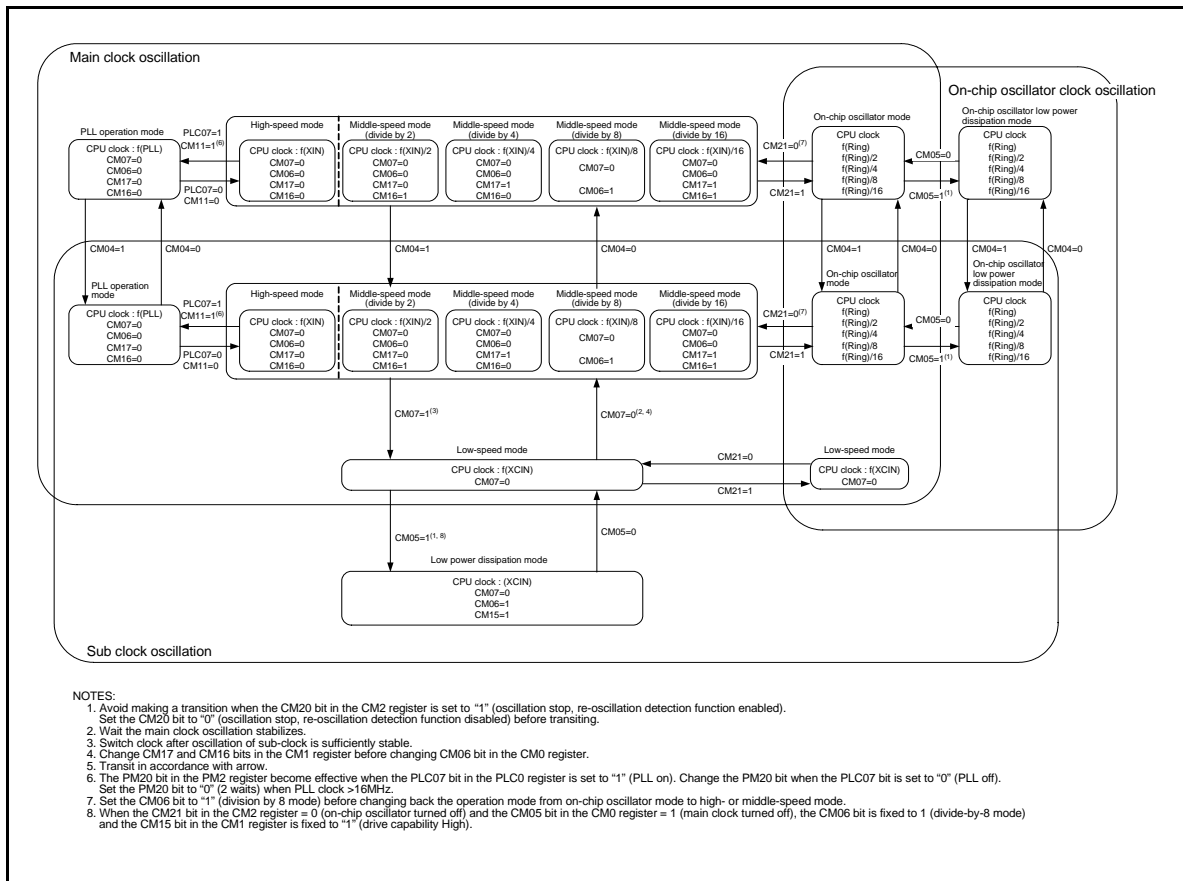


Figure 10.11 State Transition in Normal Operating Mode

**Table 10.8 Allowed Transition and Setting<sup>(9)</sup>**

		State After Transition							
		High-Speed Mode, Middle-Speed Mode	Low-Speed Mode <sup>(2)</sup>	Low Power Dissipation Mode	PLL Operating Mode <sup>(2)</sup>	On-chip Oscillator Mode	On-chip Oscillator Low Power Dissipation Mode	Stop Mode	Wait Mode
Current State	High-Speed Mode, Middle-Speed Mode	(NOTE 8)	(9)(NOTE 7)	-	(13)(NOTE 3)	(15)	-	(16)(NOTE 1)	(17)
	Low-Speed Mode <sup>(2)</sup>	(8)		(11)(NOTE 1, 6)	-	-	-	(16)(NOTE 1)	(17)
	Low Power Dissipation Mode	-	(10)		-	-	-	(16)(NOTE 1)	(17)
	PLL Operating Mode <sup>(2)</sup>	(12)(NOTE 3)	-	-		-	-	-	-
	On-chip Oscillator Mode	(14)(NOTE 4)	-	-	-	(NOTE 8)	(11)(NOTE 1)	(16)(NOTE 1)	(17)
	On-chip Oscillator Low Power Dissipation Mode	-	-	-	-	(10)	(NOTE 8)	(16)(NOTE 1)	(17)
	Stop Mode	(18)(NOTE 5)	(18)	(18)	-	(18)(NOTE 5)	(18)(NOTE 5)		-
	Wait Mode	(18)	(18)	(18)	-	(18)	(18)	-	

--: Cannot transit

## NOTES:

1. Avoid making a transition when the CM20 bit is set in to "1" (oscillation stop, re-oscillation detection function enabled). Set the CM20 bit to "0" (oscillation stop, re-oscillation detection function disabled) before transitioning.
2. On-chip oscillator clock oscillates and stops in low-speed mode. In this mode, the on-chip oscillator can be used as peripheral function clock. Sub clock oscillates and stops in PLL operating mode. In this mode, sub clock can be used as peripheral function clock.
3. PLL operating mode can only be entered from and changed to high-speed mode.
4. Set the CM06 bit to "1" (division by 8 mode) before transitioning from on-chip oscillator mode to high- or middle-speed mode.
5. When exiting stop mode, the CM05 bit is set to "1" (division by 8 mode).
6. If the CM05 bit set to "1" (main clock stop), then the CM06 bit is set to "1" (division by 8 mode).
7. A transition can be made only when sub clock is oscillating.
8. State transitions within the same mode (divide-by-n values changed or subclock oscillation turned on or off) are shown in the table below.

		Sub Clock Oscillating					Sub Clock Turned Off				
		No Division	Divided by 2	Divided by 4	Divided by 8	Divided by 16	No Division	Divided by 2	Divided by 4	Divided by 8	Divided by 16
Sub clock Oscillating	No Division		(4)	(5)	(7)	(6)	(1)	-	-	-	-
	Divided by 2	(3)		(5)	(7)	(6)	-	(1)	-	-	-
	Divided by 4	(3)	(4)		(7)	(6)	-	-	(1)	-	-
	Divided by 8	(3)	(4)	(5)		(6)	-	-	-	(1)	-
	Divided by 16	(3)	(4)	(5)	(7)		-	-	-	-	(1)
Sub clock Turned Off	No Division	(2)	-	-	-	-		(4)	(5)	(7)	(6)
	Divided by 2	-	(2)	-	-	-	(3)		(5)	(7)	(6)
	Divided by 4	-	-	(2)	-	-	(3)	(4)		(7)	(6)
	Divided by 8	-	-	-	(2)	-	(3)	(4)	(5)		(6)
	Divided by 16	-	-	-	-	(2)	(3)	(4)	(5)	(7)	

--: Cannot transit

9. ( ) : setting method. See the following table.

	Setting	Operation		Setting	Operation
(1)	CM04 = 0	Sub clock turned off	(10)	CM05 = 0	Main clock oscillating
(2)	CM04 = 1	Sub clock oscillating	(11)	CM05 = 1	Main clock turned off
(3)	CM06 = 0, CM17 = 0, CM16 = 0	CPU clock no division mode	(12)	PLC07=0, CM11=0	Main clock selected
(4)	CM06 = 0, CM17 = 0, CM16 = 1	CPU clock division by 2 mode	(13)	PLC07=1, CM11=1	PLL clock selected
(5)	CM06 = 0, CM17 = 1, CM16 = 0	CPU clock division by 4 mode	(14)	CM21=0	Main clock or PLL clock selected
(6)	CM06 = 0, CM17 = 1, CM16 = 1	CPU clock division by 16 mode	(15)	CM21=1	On-chip oscillator clock selected
(7)	CM06 = 1	CPU clock division by 8 mode	(16)	CM10=1	Transition to stop mode
(8)	CM07 = 0	Main clock, PLL clock, or on-chip oscillator clock selected	(17)	Wait Instruction	Transition to wait mode
(9)	CM07 = 1	Sub clock selected	(18)	Hardware Interrupt	Exit stop mode or wait mode

CM04, CM05, CM06, CM07 : Bits in CM0 register  
 CM10, CM11, CM16, CM17 : Bits in CM1 register  
 CM20, CM21 : Bits in CM2 register  
 PLC07 : Bits in PLC0 register

## 10.5 System Clock Protection Function

The system clock protection function prohibits the CPU clock from changing clock sources when the main clock is selected the CPU clock source. This prevents the CPU clock from stopping should the program crash. This function is available when the main clock is selected as the CPU clock source.

When the PM21 bit in the PM2 register is set to “1” (clock change disabled), the following bits cannot be written to:

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

When using the system clock protection 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 follow the procedure below.

- (1) Set the PRC1 bit in the PRCR register to “1” (write enable).
- (2) Set the PM21 bit in the PM2 register to “1” (protects the clock).
- (3) Set the PRC1 bit in the PRCR register to “0” (write disable).

When the PM21 bit is set to “1,” do not execute the WAIT instruction.

## 10.6 Oscillation Stop and Re-oscillation Detect Function

The oscillation stop and re-oscillation detect function is such that main clock oscillation circuit stop and re-oscillation are detected. At oscillation stop, re-oscillation detection, reset or oscillation stop, re-oscillation detection interrupt are generated. Which is to be generated can be selected using the CM27 bit in the CM2 register. The oscillation stop detection function can be enabled and disabled by the CM20 bit in the CM2 register. Table 10.9 lists a Specification Overview of Oscillation Stop and Re-Oscillation Detect Function.

**Table 10.9 Specification Overview of Oscillation Stop and Re-Oscillation Detect Function**

Item	Specification
Oscillation Stop Detectable Clock and Frequency Bandwidth	$f(XIN) \geq 2 \text{ MHz}$
Enabling Condition for Oscillation Stop, Re-Oscillation Detection Function	Set CM20 bit to "1" (enable)
Operation at Oscillation Stop, Re-Oscillation Detection	<ul style="list-style-type: none"> <li>Reset occurs (when CM27 bit = 0)</li> <li>Oscillation stop, re-oscillation detection interrupt generated (when CM27 bit = 1)</li> </ul>

### 10.6.1 Operation When CM27 bit = 0 (Oscillation Stop Detection Reset)

Where main clock stop is detected when the CM20 bit is "1" (oscillation stop, re-oscillation detection function enabled), the microcomputer is initialized, coming to a halt (oscillation stop reset; refer to **4. Special Function Register (SFR)**, **5. Reset**).

This status is reset with hardware reset 1 or hardware reset 2. Also, even when re-oscillation is detected, the microcomputer can be initialized and stopped; it is, however, necessary to avoid such usage (During main clock stop, do not set the CM20 bit to "1" and the CM27 bit to "0").

### 10.6.2 Operation When CM27 bit = 1 (Oscillation Stop and Re-oscillation Detect Interrupt)

Where the main clock corresponds to the CPU clock source and the CM20 bit is "1" (oscillation stop and re-oscillation detect function enabled), the system is placed in the following state if the main clock comes to a halt:

- Oscillation stop and re-oscillation detect interrupt request occurs.
- The on-chip oscillator starts oscillation, and the on-chip oscillator clock becomes the clock source for CPU clock and peripheral functions in place of the main clock.
- CM21 bit = 1 (on-chip oscillator clock for CPU clock source and clock source of peripheral function.)
- CM22 bit = 1 (main clock stop detected)
- CM23 bit = 1 (main clock stopped)

Where the PLL clock corresponds to the CPU clock source and the CM20 bit is "1," the system is placed in the following state if the main clock comes to a halt: Since the CM21 bit remains unchanged, set it to "1" (on-chip oscillator clock) inside the interrupt routine.

- Oscillation stop and re-oscillation detect interrupt request occurs.
- CM22 bit = 1 (main clock stop detected)
- CM23 bit = 1 (main clock stopped)
- CM21 bit remains unchanged

Where the CM20 bit is "1", the system is placed in the following state if the main clock re-oscillates from the stop condition:

- Oscillation stop and re-oscillation detect interrupt request occurs.
- CM22 bit = 1 (main clock re-oscillation detected)
- CM23 bit = 0 (main clock oscillation)
- CM21 bit remains unchanged

### 10.6.3 How to Use Oscillation Stop and Re-oscillation Detect Function

- The oscillation stop and re-oscillation detect interrupt shares the vector with the watchdog timer interrupt and low voltage detection interrupt. If the oscillation stop, re-oscillation detection and watchdog timer interrupts both are used, read the CM22 bit in an interrupt routine to determine which interrupt source is requesting the interrupt.
- Where the main clock re-oscillated after oscillation stop, the clock source for the CPU clock and peripheral functions must be switched to the main clock in the program. Figure 10.12 shows the Procedure to Switch Clock Source From On-chip Oscillator to Main Clock.
- Simultaneously with oscillation stop, re-oscillation detection interrupt occurrence, the CM22 bit becomes "1". When the CM22 bit is set at "1," oscillation stop, re-oscillation detection interrupt are disabled. By setting the CM22 bit to "0" in the program, oscillation stop, re-oscillation detection interrupt are enabled.
- If the main clock stops during low speed mode where the CM20 bit is "1", an oscillation stop, re-oscillation detection interrupt request is generated. At the same time, the on-chip oscillator starts oscillating. In this case, although the CPU clock is derived from the sub clock as it was before the interrupt occurred, the peripheral function clocks now are derived from the on-chip oscillator clock.
- To enter wait mode while using the oscillation stop, re-oscillation detection function, set the CM02 bit to "0" (peripheral function clocks not turned off during wait mode).
- Since the oscillation stop, re-oscillation detection function is provided in preparation for main clock stop due to external factors, set the CM20 bit to "0" (Oscillation stop, re-oscillation detection function disabled) where the main clock is stopped or oscillated in the program, that is where the stop mode is selected or the CM05 bit is altered.
- This function cannot be used if the main clock frequency is 2 MHz or less. In that case, set the CM20 bit to "0".

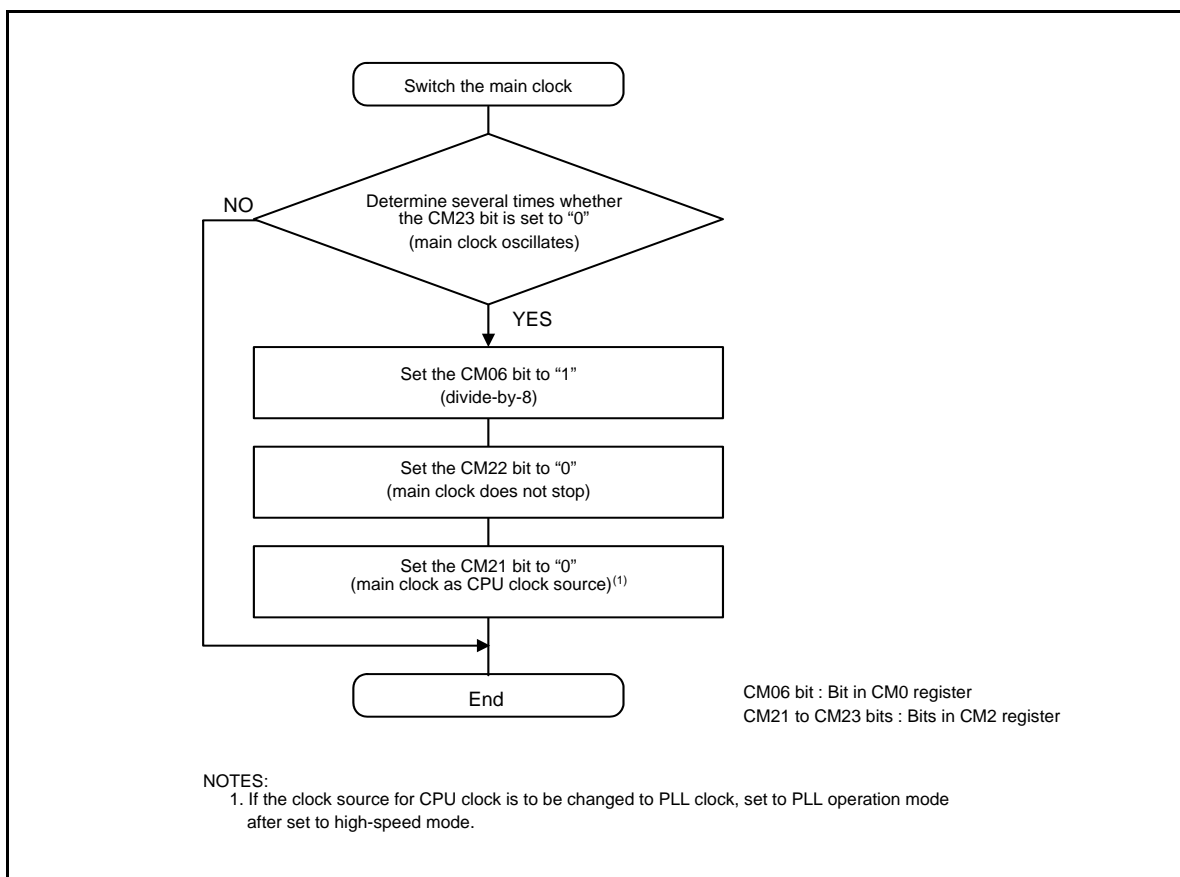


Figure 10.12 Procedure to Switch Clock Source From On-chip Oscillator to Main Clock

## 11. Protection

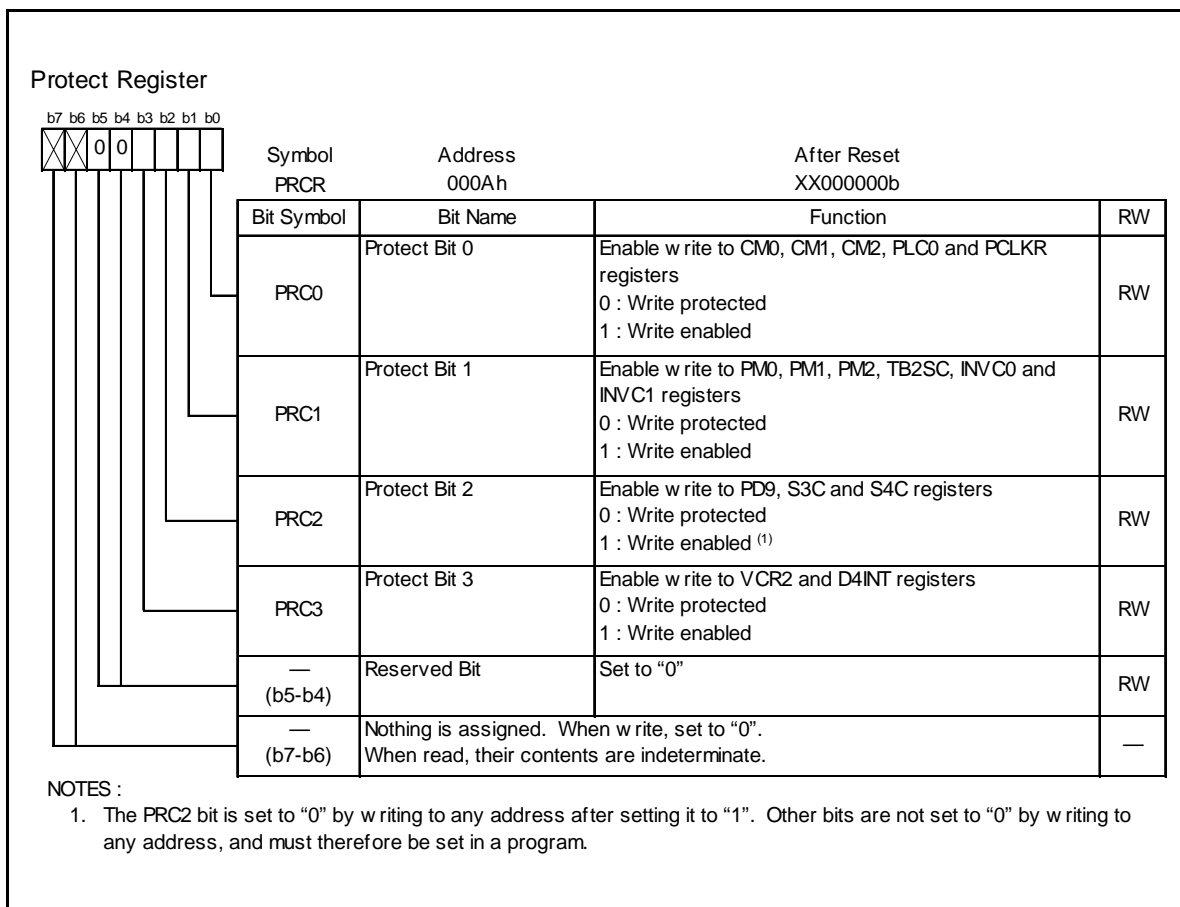
### Note

The M16C/62PT do not use the PRC3 bit in the PRCR register.

In the event that a program runs out of control, this function protects the important registers so that they will not be rewritten easily. Figure 11.1 shows the PRCR Register. The following lists the registers protected by the PRCR register.

- The PRC0 bit protects the CM0, CM1, CM2, PLC0 and PCLKR registers;
- The PRC1 bit protects the PM0, PM1, PM2, TB2SC, INVC0 and INVC1 registers;
- The PRC2 bit protects the PD9, S3C and S4C registers;
- The PRC3 bit protects the VCR2 and D4INT registers.

Set the PRC2 bit to “1” (write enabled) and then write to any address, and the PRC2 bit will be cleared to “0” (write protected). The registers protected by the PRC2 bit should be changed in the next instruction after setting the PRC2 bit to “1”. Make sure no interrupts or DMA transfers will occur between the instruction in which the PRC2 bit is set to “1” and the next instruction. The PRC0, PRC1 and PRC3 bits are not automatically cleared to “0” by writing to any address. They can only be cleared in a program.



**Figure 11.1 PRCR Register**



## 12. Interrupt

### Note

The M16C/62P (80-pin version) do not use  $\overline{\text{INT3}}$  to  $\overline{\text{INT5}}$  interrupt of peripheral function.

The M16C/62PT (100-pin version) do not use low voltage detection interrupt.

The M16C/62PT (80-pin version) do not use low voltage detection interrupt and  $\overline{\text{INT3}}$  to  $\overline{\text{INT5}}$  interrupt of peripheral function.

### 12.1 Type of Interrupts

Figure 12.1 shows Type of Interrupts.

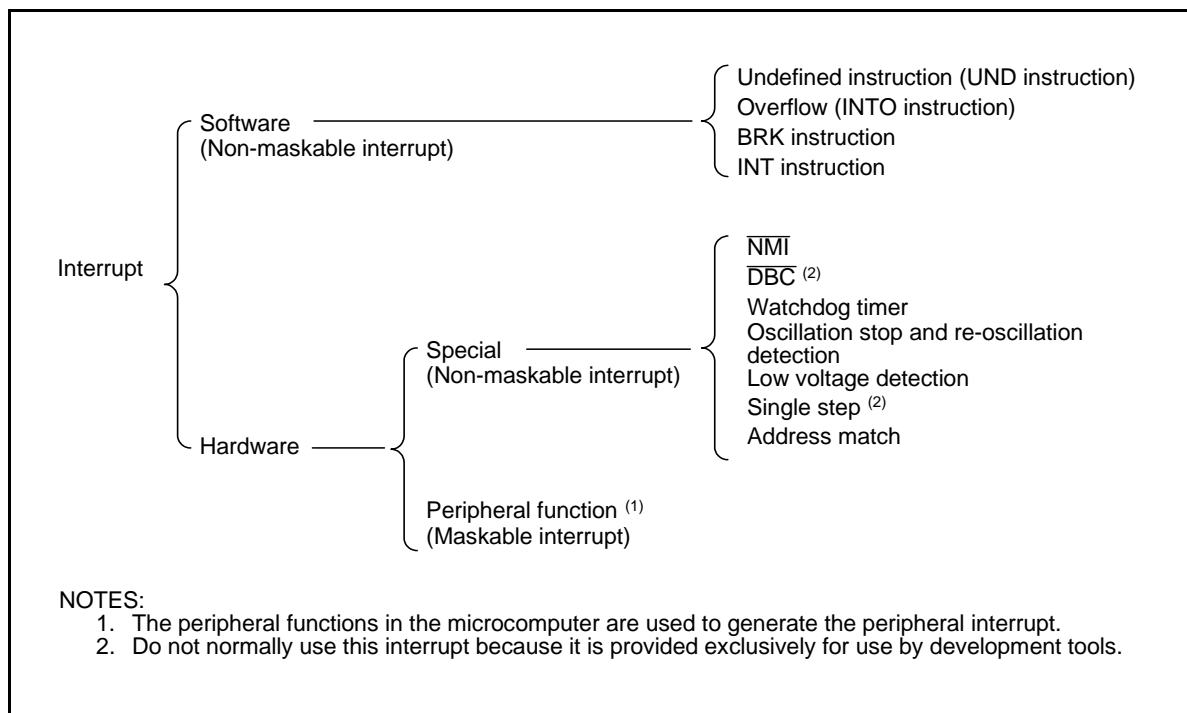


Figure 12.1 Type of Interrupts

- Maskable Interrupt : An interrupt which can be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **can be changed** by priority level.
- Non-Maskable Interrupt : An interrupt which cannot be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **cannot be changed** by priority level.

## 12.2 Software Interrupts

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

### 12.2.1 Undefined Instruction Interrupt

An undefined instruction interrupt occurs when executing the UND instruction.

### 12.2.2 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 arithmetic: ABS, ADC, ADCF, ADD, CMP, DIV, DIVU, DIVX, NEG, RMPA, SBB, SHA, SUB

### 12.2.3 BRK Interrupt

A BRK interrupt occurs when executing the BRK instruction.

### 12.2.4 INT Instruction Interrupt

An INT instruction interrupt occurs when executing the INT instruction. Software interrupt Nos. 0 to 63 can be specified for the INT instruction. Because software interrupt Nos. 4 to 31 are assigned to peripheral function interrupts, the same interrupt routine as for peripheral function interrupts can be executed by executing the INT instruction.

In software interrupt Nos. 0 to 31, the U flag is saved to 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. In software interrupt Nos. 32 to 63, the U flag does not change state during instruction execution, and the SP then selected is used.

## 12.3 Hardware Interrupts

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

### 12.3.1 Special Interrupts

Special interrupts are non-maskable interrupts.

#### 12.3.1.1 $\overline{\text{NMI}}$ 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 the **12.7  $\overline{\text{NMI}}$  Interrupt**.

#### 12.3.1.2 $\overline{\text{DBC}}$ Interrupt

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

#### 12.3.1.3 Watchdog Timer Interrupt

Generated by the watchdog timer. Once a watchdog timer interrupt is generated, be sure to initialize the watchdog timer. For details about the watchdog timer, refer to the **13. Watchdog Timer**.

#### 12.3.1.4 Oscillation Stop and Re-oscillation Detection Interrupt

Generated by the oscillation stop and re-oscillation detection function. For details about the oscillation stop and re-oscillation detection function, refer to the **10. Clock Generation Circuit**.

#### 12.3.1.5 12.3.1.5 Low Voltage Detection Interrupt

Generated by the voltage detection circuit. For details about the voltage detection circuit, refer to the **6. Voltage Detection Circuit**.

#### 12.3.1.6 Single-Step Interrupt

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

#### 12.3.1.7 Address Match Interrupt

An address match interrupt is generated immediately before executing the instruction at the address indicated by the RMAD0 to RMAD3 register that corresponds to one of the AIER0 or AIER1 bit in the AIER register or the AIER20 or AIER21 bit in the AIER2 register which is “1” (address match interrupt enabled). For details about the address match interrupt, refer to the **12.9 Address Match Interrupt**.

### 12.3.2 Peripheral Function Interrupts

The peripheral function interrupt occurs when a request from the peripheral functions in the microcomputer is acknowledged. The peripheral function interrupt is a maskable interrupt. See **Table 12.2 Relocatable Vector Tables** about how the peripheral function interrupt occurs. Refer to the descriptions of each function for details.

## 12.4 Interrupts and Interrupt Vector

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 12.2 shows the Interrupt Vector.

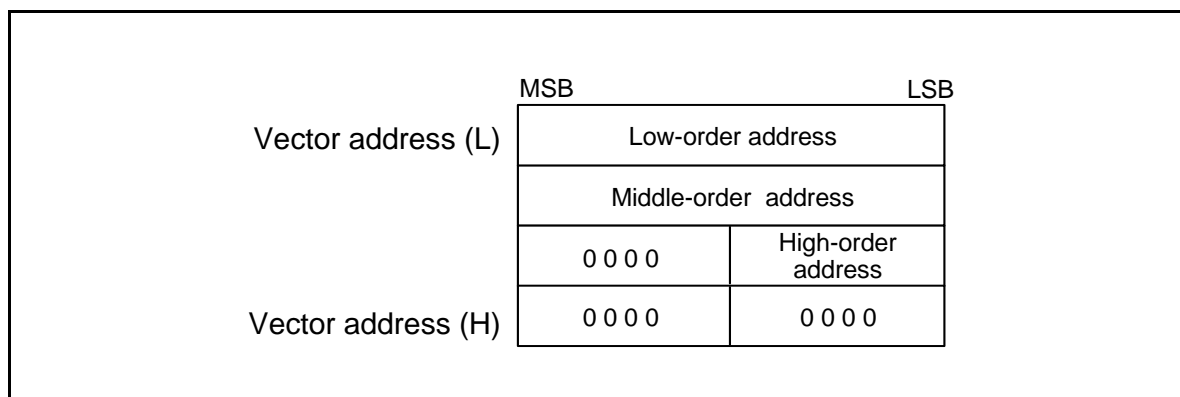


Figure 12.2 Interrupt Vector

### 12.4.1 Fixed Vector Tables

The fixed vector tables are allocated to the addresses from FFFDCh to FFFFFh. Table 12.1 lists the Fixed Vector Tables. In the flash memory version of microcomputer, the vector addresses (H) of fixed vectors are used by the ID code check function. For details, refer to the **22.2 Functions To Prevent Flash Memory from Rewriting**.

Table 12.1 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 Series software manual
Overflow (INTO instruction)	FFFE0h to FFFE3h	
BRK Instruction <sup>(2)</sup>	FFFE4h to FFFE7h	
Address Match	FFFE8h to FFEFBh	12.9 Address Match Interrupt
Single Step <sup>(1)</sup>	FFFECh to FFFEFh	
Watchdog Timer, Oscillation Stop and Re-Oscillation Detection, Low Voltage Detection	FFFF0h to FFFF3h	13. Watchdog Timer 10. Clock Generation Circuit 6. Voltage Detection Circuit
DBC <sup>(1)</sup>	FFFF4h to FFFF7h	
NMI	FFFF8h to FFFFBh	12.7 NMI interrupt
Reset	FFFFCh to FFFFFh	5. Reset

NOTES:

- Do not normally use this interrupt because it is provided exclusively for use by development tools.
- If the contents of address FFFE7h is FFh, program execution starts from the address shown by the vector in the relocatable vector table.

### 12.4.2 Relocatable Vector Tables

The 256 bytes beginning with the start address set in the INTB register comprise a relocatable vector table area. Table 12.2 lists the Relocatable Vector Tables. Setting an even address in the INTB register results in the interrupt sequence being executed faster than in the case of odd addresses.

**Table 12.2 Relocatable Vector Tables**

Interrupt Source	Vector Address <sup>(1)</sup> Address (L) to Address (H)	Software Interrupt Number	Reference
BRK Instruction <sup>(5)</sup>	+0 to +3 (0000h to 0003h)	0	M16C/60, M16C/20
–(Reserved)		1 to 3	Series software manual
INT $\overline{3}$	+16 to +19 (0010h to 0013h)	4	12.6 INT interrupt
Timer B5	+20 to +23 (0014h to 0017h)	5	15. Timers
Timer B4, UART1 Bus Collision Detect <sup>(4, 6)</sup>	+24 to +27 (0018h to 001Bh)	6	15. Timers
Timer B3, UART0 Bus Collision Detect <sup>(4, 6)</sup>	+28 to +31 (001Ch to 001Fh)	7	17. Serial Interface
SI/O4, INT $\overline{5}$ <sup>(2)</sup>	+32 to +35 (0020h to 0023h)	8	12.6 INT interrupt
SI/O3, INT $\overline{4}$ <sup>(2)</sup>	+36 to +39 (0024h to 0027h)	9	17. Serial Interface
UART 2 Bus Collision Detection <sup>(6)</sup>	+40 to +43 (0028h to 002Bh)	10	17. Serial Interface
DMA0	+44 to +47 (002Ch to 002Fh)	11	14. DMAC
DMA1	+48 to +51 (0030h to 0033h)	12	
Key Input Interrupt	+52 to +55 (0034h to 0037h)	13	12.8 Key Input Interrupt
A/D	+56 to +59 (0038h to 003Bh)	14	18. A/D Converter
UART2 Transmit, NACK2 <sup>(3)</sup>	+60 to +63 (003Ch to 003Fh)	15	17. Serial Interface
UART2 Receive, ACK2 <sup>(3)</sup>	+64 to +67 (0040h to 0043h)	16	
UART0 Transmit, NACK0 <sup>(3)</sup>	+68 to +71 (0044h to 0047h)	17	
UART0 Receive, ACK0 <sup>(3)</sup>	+72 to +75 (0048h to 004Bh)	18	
UART1 Transmit, NACK1 <sup>(3)</sup>	+76 to +79 (004Ch to 004Fh)	19	
UART1 Receive, ACK1 <sup>(3)</sup>	+80 to +83 (0050h to 0053h)	20	
Timer A0	+84 to +87 (0054h to 0057h)	21	15. Timers
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	
Timer B0	+104 to +107 (0068h to 006Bh)	26	
Timer B1	+108 to +111 (006Ch to 006Fh)	27	
Timer B2	+112 to +115 (0070h to 0073h)	28	
INT0	+116 to +119 (0074h to 0077h)	29	12.6 INT interrupt
INT1	+120 to +123 (0078h to 007Bh)	30	
INT2	+124 to +127 (007Ch to 007Fh)	31	
Software Interrupt <sup>(5)</sup>	+128 to +131 (0080h to 0083h) to +252 to +255 (00FCh to 00FFh)	32 to 63	M16C/60, M16C/20 Series software manual

**NOTES:**

1. Address relative to address in INTB.
2. Use the IFSR6 and IFSR7 bits in the IFSR register to select.
3. During I<sup>2</sup>C mode, NACK and ACK interrupts comprise the interrupt source.
4. Use the IFSR26 and IFSR27 bits in the IFSR2A register to select.
5. These interrupts cannot be disabled using the I flag.
6. Bus collision detection : During IE mode, this bus collision detection constitutes the factor of an interrupt.  
During I<sup>2</sup>C mode, however, a start condition or a stop condition detection constitutes the factor of an interrupt.

## 12.5 Interrupt Control

The following describes how to enable/disable the maskable interrupts, and how to set the priority in which order they are accepted. What is explained here does not apply to nonmaskable interrupts.

Use the I flag in the FLG register, IPL, and ILVL2 to ILVL0 bits in the each interrupt control register to enable/disable the maskable interrupts. Whether an interrupt is requested is indicated by the IR bit in the each interrupt control register.

Figure 12.3 and Figure 12.4 show the Interrupt Control Registers.

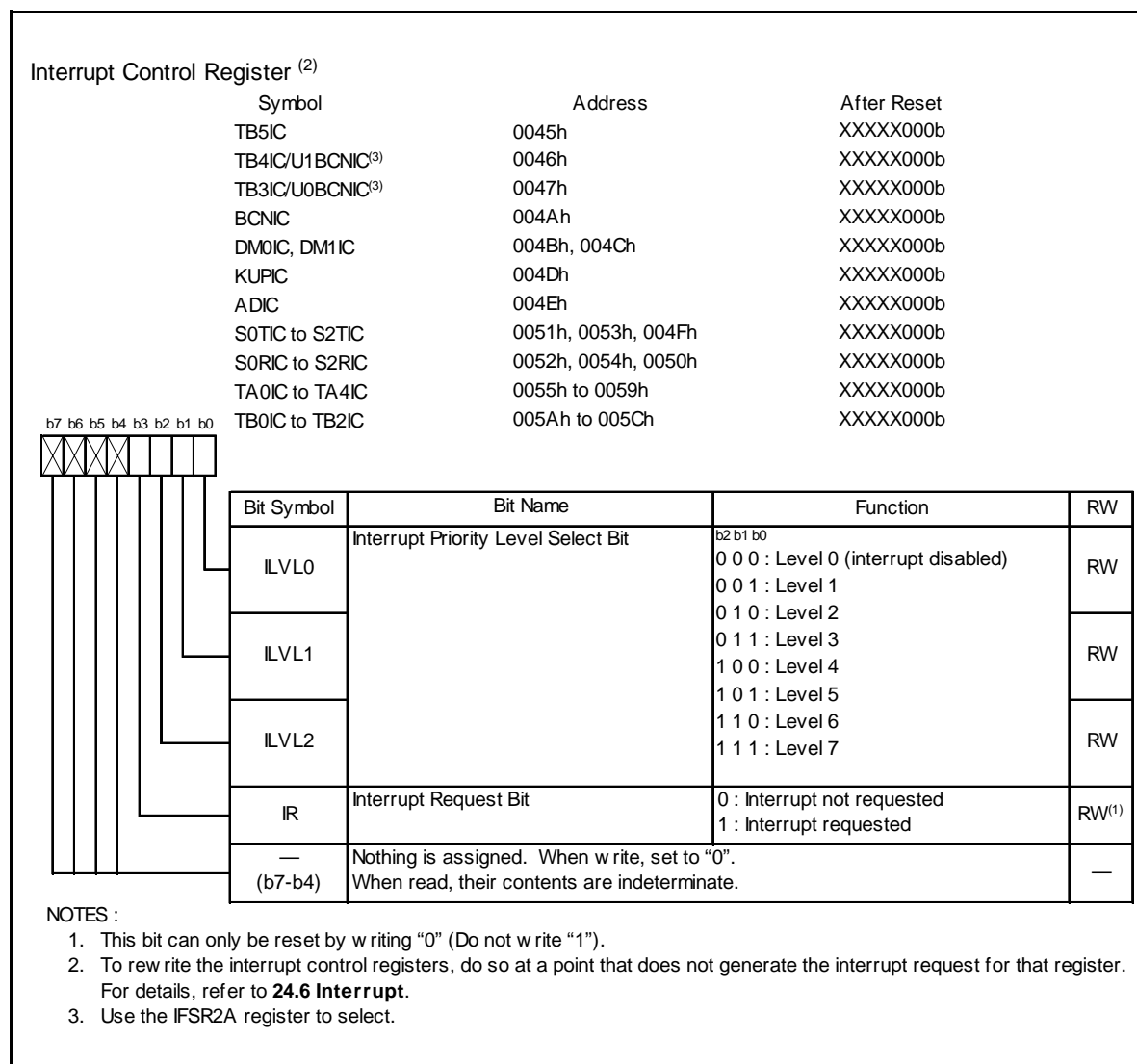
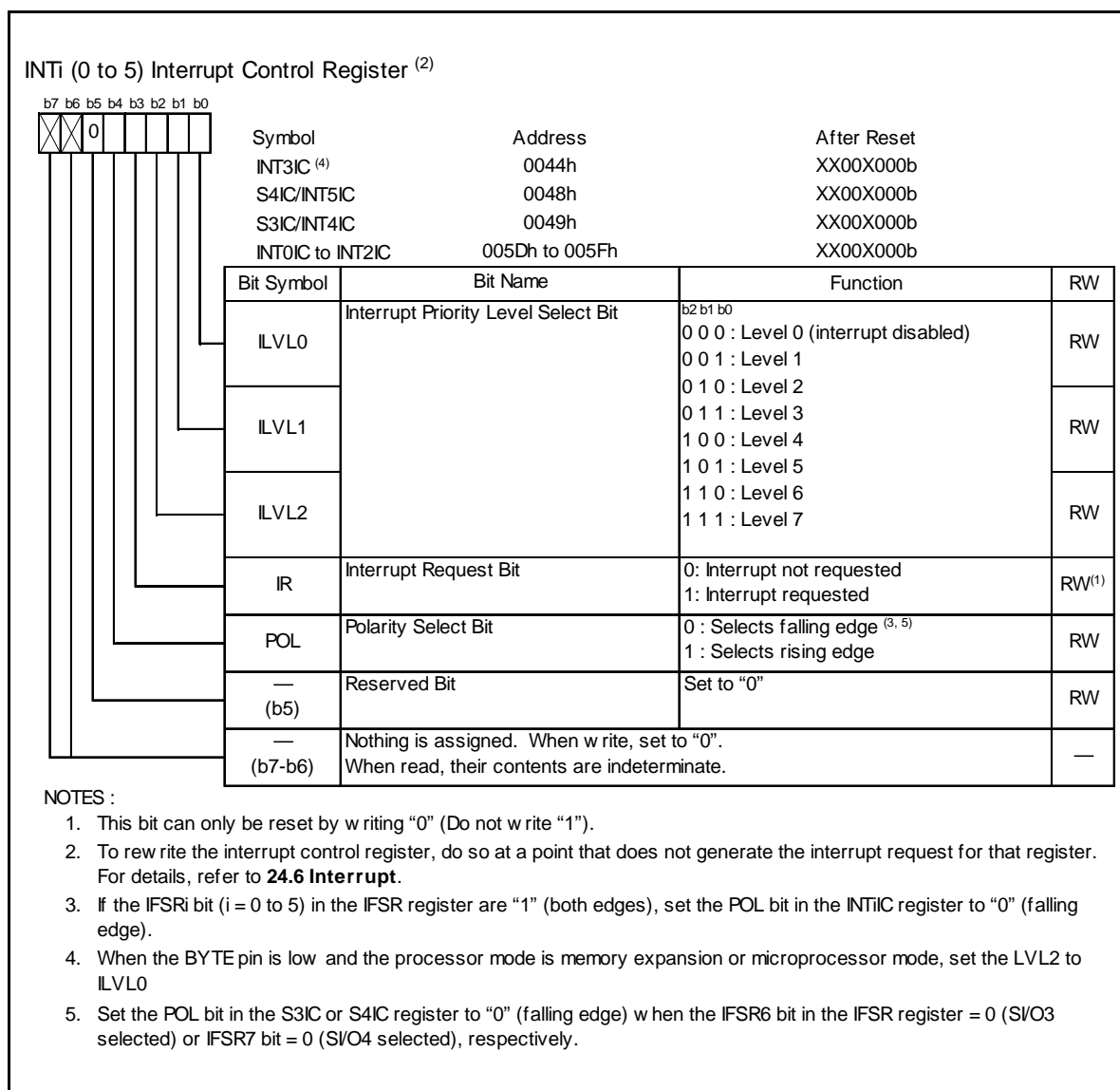


Figure 12.3 Interrupt Control Registers (1)

**Figure 12.4 Interrupt Control Registers (2)**

### 12.5.1 I Flag

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

### 12.5.2 IR Bit

The IR bit is set to “1” (= interrupt requested) when an interrupt request is generated. Then, when the interrupt request is accepted and the CPU branches to the corresponding interrupt vector, the IR bit is cleared to “0” (= interrupt not requested).

The IR bit can be cleared to “0” in a program. Note that do not write “1” to this bit.

### 12.5.3 ILVL2 to ILVL0 Bits and IPL

Interrupt priority levels can be set using the ILVL2 to ILVL0 bits.


Table 12.3 shows the Settings of Interrupt Priority Levels and Table 12.4 shows the Interrupt Priority Levels Enabled by IPL.

The following are conditions under which an interrupt is accepted:

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

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

**Table 12.3 Settings of Interrupt Priority Levels**

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

**Table 12.4 Interrupt Priority Levels Enabled by IPL**

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



### 12.5.4 Interrupt Sequence

An interrupt sequence – what are performed over a period from the instant an interrupt is accepted to the instant the interrupt routine is executed – is described here.

If an interrupt occurs during execution of an instruction, the processor determines its priority when the execution of the instruction is completed, and transfers control to the interrupt sequence from the next cycle. 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.

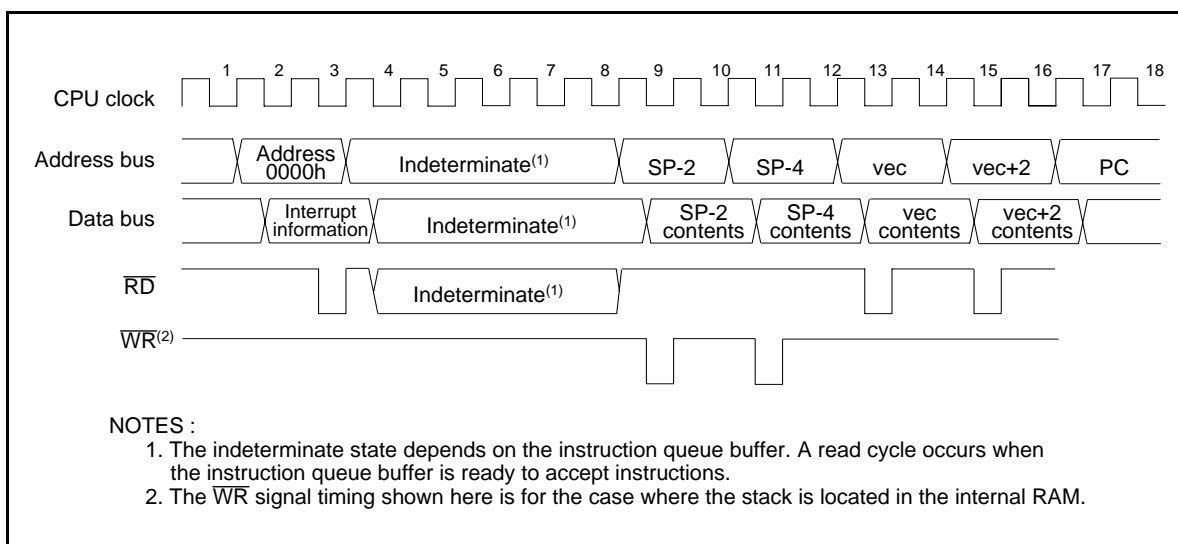
The CPU behavior during the interrupt sequence is described below. Figure 12.5 shows Time Required for Executing Interrupt Sequence.

- (1) The CPU obtains interrupt information (interrupt number and interrupt request level) by reading address 000000h. Then, the IR bit applicable to the interrupt information is set to “0” (interrupt requested).
- (2) The FLG register, prior to an interrupt sequence, is saved to a temporary register <sup>(1)</sup> within the CPU.
- (3) The I, D and U flags in the FLG register become 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)
 However, the U flag does not change state if an INT instruction for software interrupt Nos. 32 to 63 is executed.
- (4) The temporary register <sup>(1)</sup> within the CPU is saved to the stack.
- (5) The PC is saved to the stack.
- (6) The interrupt priority level of the acknowledged interrupt in IPL is set.
- (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.

#### NOTES:

1. Temporary register cannot be modified by users.



**Figure 12.5 Time Required for Executing Interrupt Sequence**

### 12.5.5 Interrupt Response Time

Figure 12.6 shows the Interrupt Response Time. The interrupt response or interrupt acknowledge time denotes a time from when an interrupt request is generated till when the first instruction in the interrupt routine is executed. Specifically, it consists of a time from when an interrupt request is generated till when the instruction then executing is completed ((a) on Figure 12.6) and a time during which the interrupt sequence is executed ((b) on Figure 12.6).

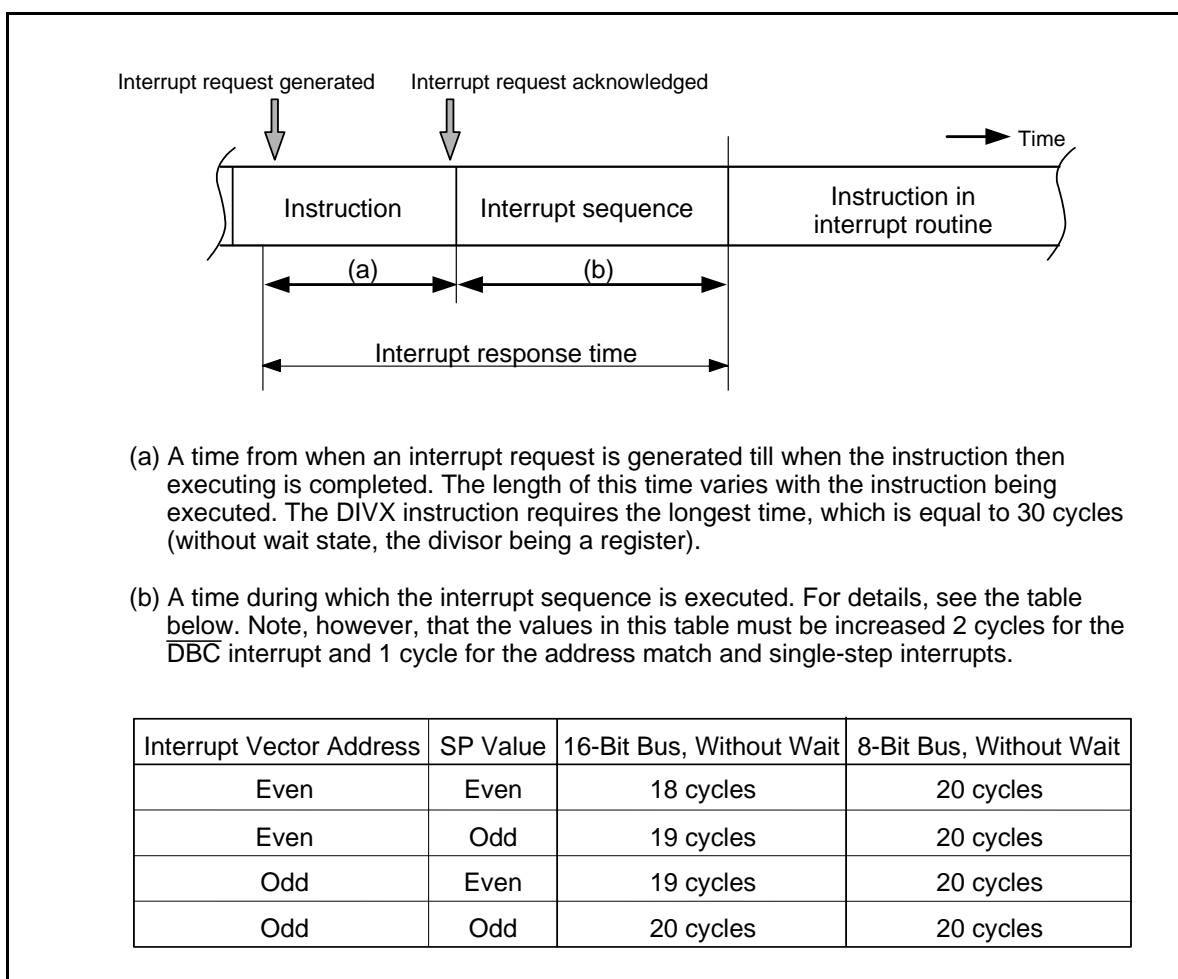


Figure 12.6 Interrupt Response Time

### 12.5.6 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 12.5 is set in the IPL. Table 12.5 lists the IPL Level That is Set to IPL When a Software or Special Interrupt is Accepted.

Table 12.5 IPL Level That is Set to IPL When a Software or Special Interrupt is Accepted

Interrupt Sources	Level that is Set to IPL
Watchdog Timer, NMI, Oscillation Stop and Re-Oscillation Detection, Low Voltage Detection	7
Software, Address Match, $\overline{\text{DBC}}$ , Single-Step	Not changed

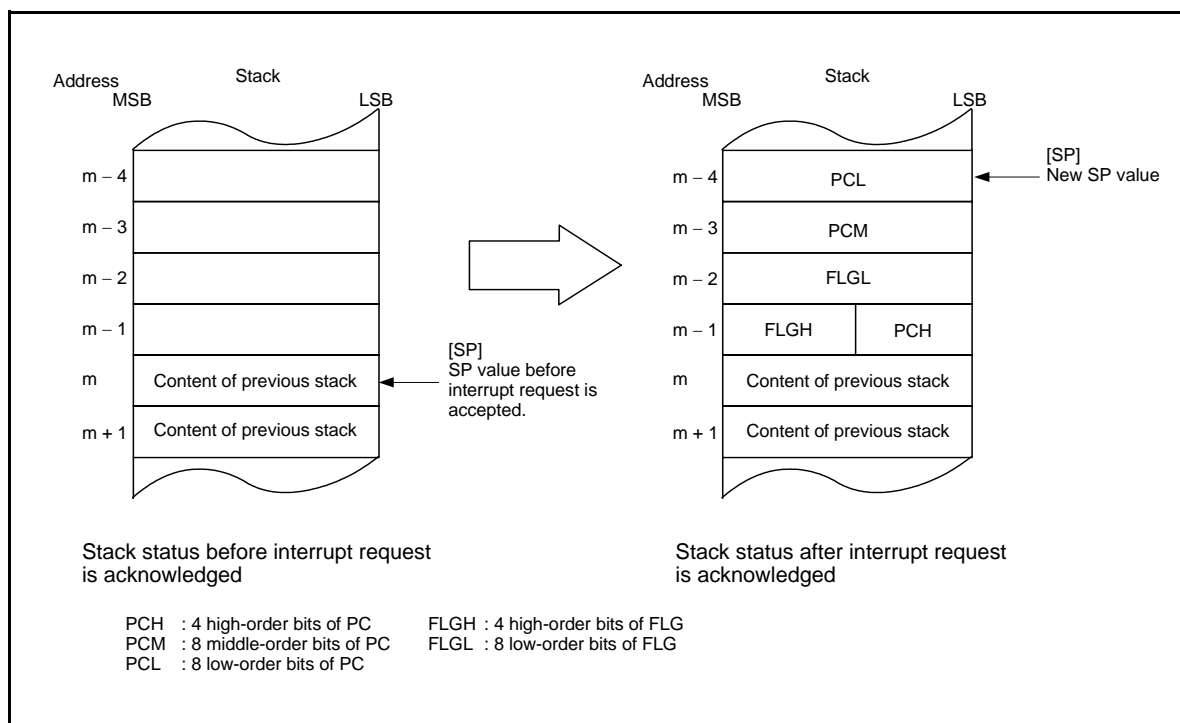
### 12.5.7 Saving Registers

In the interrupt sequence, the FLG register and PC are saved to 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 to the stack first. Next, the 16 low-order bits of the PC are saved.

Figure 12.7 shows the Stack Status Before and After Acceptance of Interrupt Request.

The other necessary registers must be saved in 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.

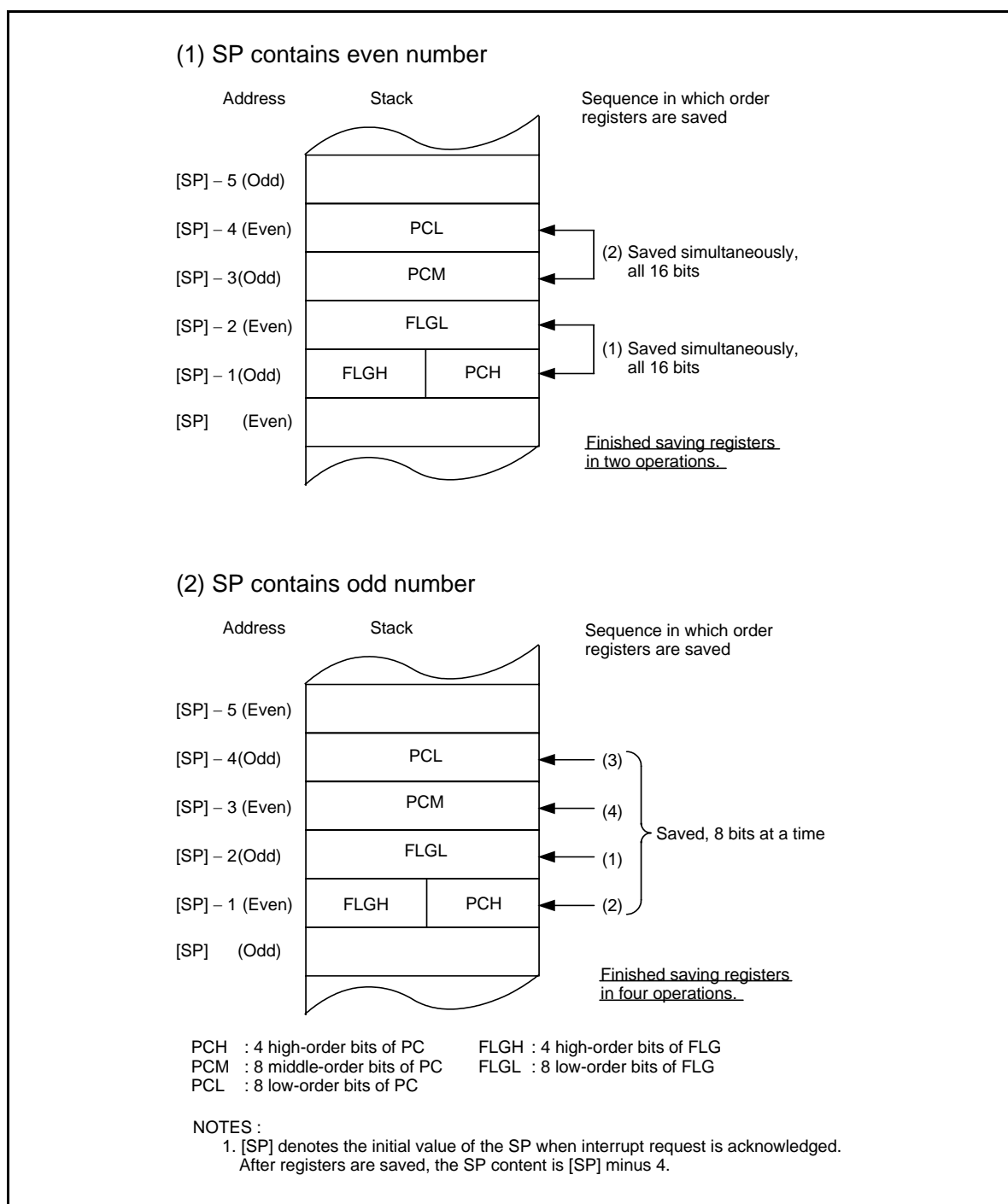


**Figure 12.7** Stack Status Before and After Acceptance of Interrupt Request

The operation of saving registers carried out in the interrupt sequence is dependent on whether the SP <sup>(1)</sup>, at the time of acceptance of an interrupt request, is even or odd. If the stack pointer <sup>(1)</sup> 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 12.8 shows the Operation of Saving Register.

NOTES:

1. When any INT instruction in software numbers 32 to 63 has been executed, this is the SP indicated by the U flag. Otherwise, it is the ISP.



**Figure 12.8 Operation of Saving Register**



### 12.5.10 Interrupt Priority Level Select Circuit

The interrupt priority level select circuit selects the highest priority interrupt when two or more interrupt requests are sampled at the same sampling point.

Figure 12.10 shows the Interrupts Priority Select Circuit.

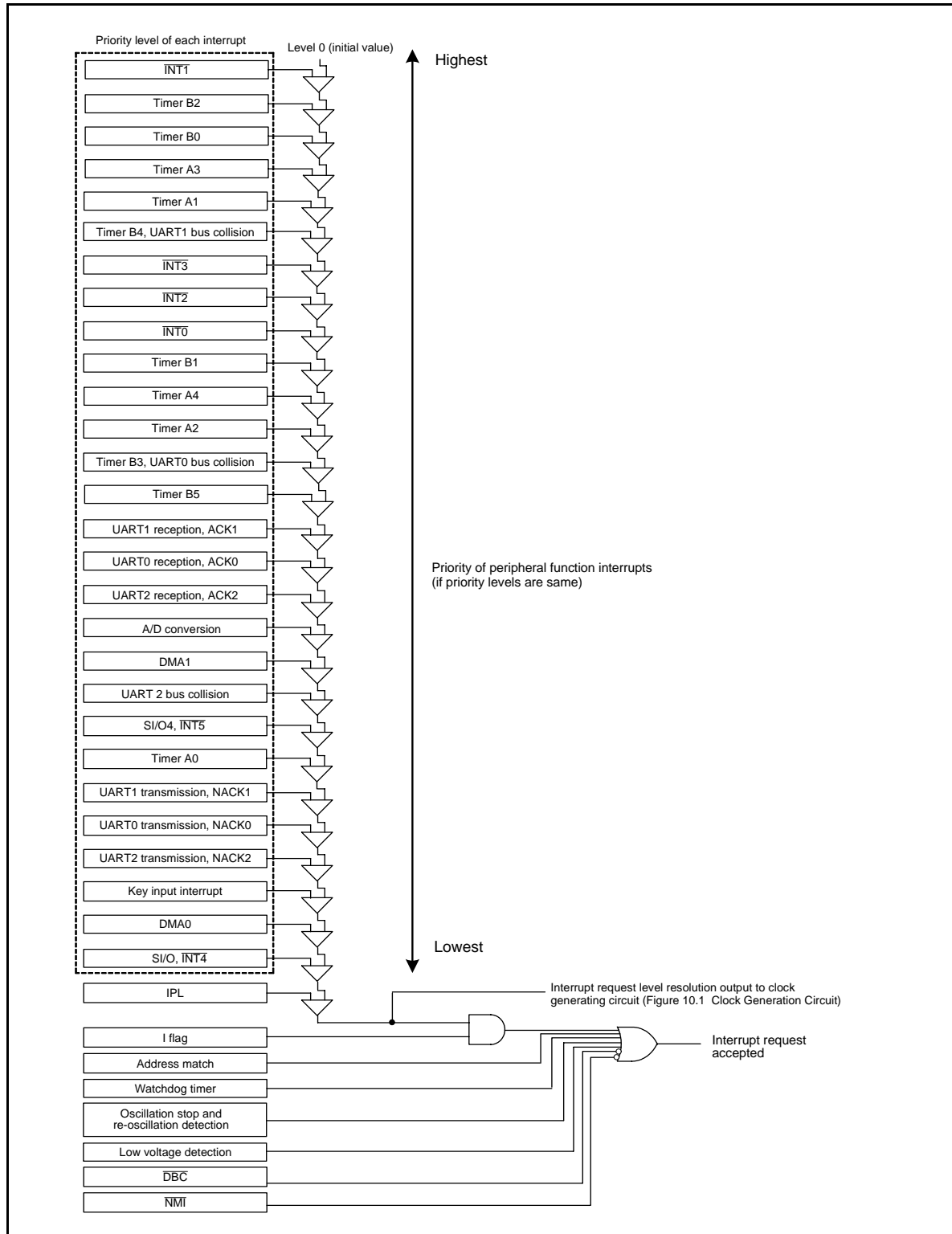


Figure 12.10 Interrupts Priority Select Circuit

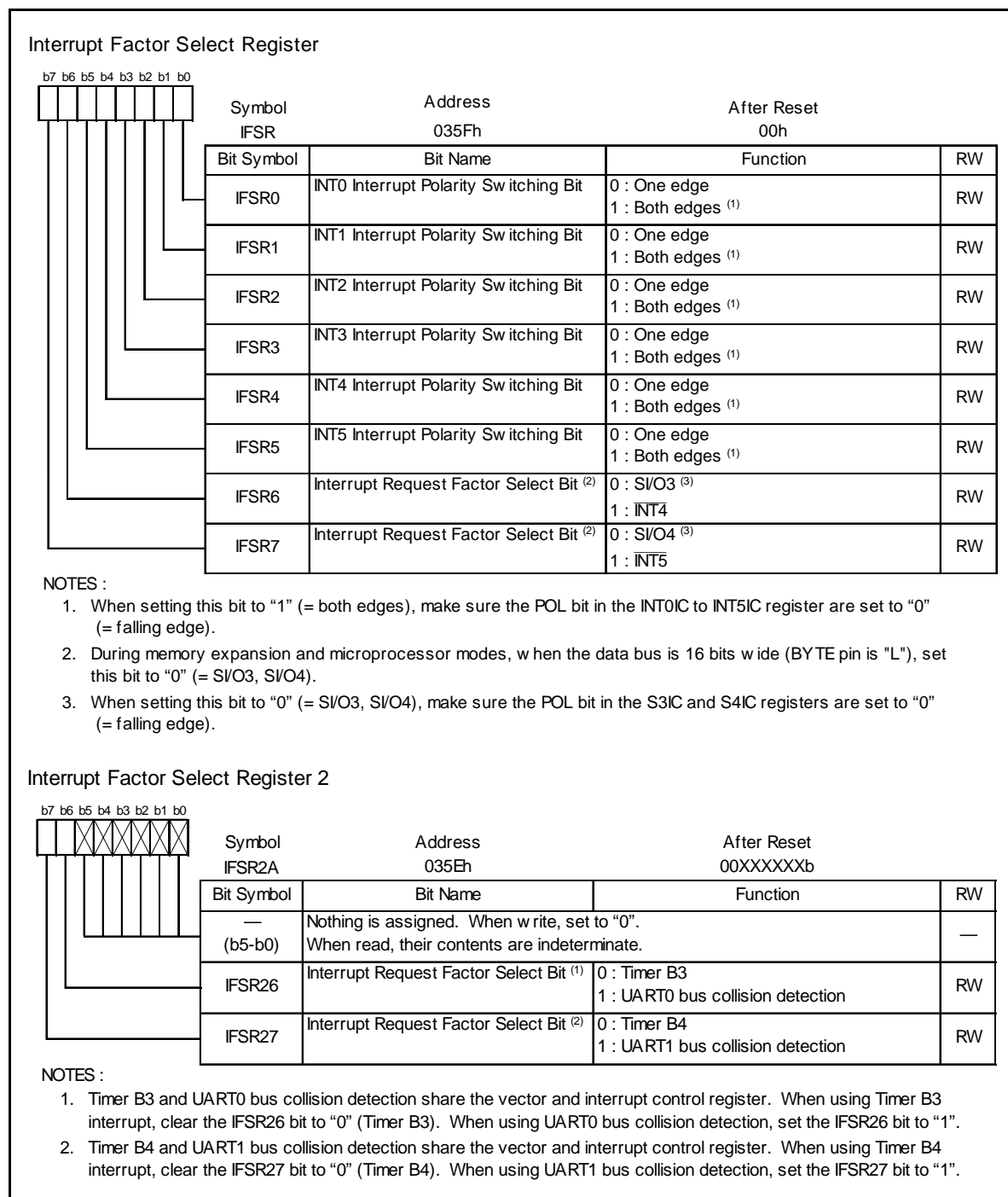
## 12.6 $\overline{\text{INT}}$ Interrupt

$\overline{\text{INT}}_i$  interrupt ( $i=0$  to 5) is triggered by the edges of external inputs. The edge polarity is selected using the IFSR $_i$  bit in the IFSR register.

$\overline{\text{INT}}_4$  and  $\overline{\text{INT}}_5$  share the interrupt vector and interrupt control register with SI/O3 and SI/O4, respectively. To use the  $\overline{\text{INT}}_4$  interrupt, set the IFSR6 bit in the IFSR register to "1" (=  $\overline{\text{INT}}_4$ ). To use the  $\overline{\text{INT}}_5$  interrupt, set the IFSR7 bit in the IFSR register to "1" (=  $\overline{\text{INT}}_5$ ).

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

Figure 12.11 shows the IFSR and IFSR2A Registers.



**Figure 12.11 IFSR and IFSR2A Registers**

## 12.7 $\overline{\text{NMI}}$ Interrupt

An  $\overline{\text{NMI}}$  interrupt is generated when input on the  $\overline{\text{NMI}}$  pin changes state from high to low. The  $\overline{\text{NMI}}$  interrupt is a non-maskable interrupt.

The input level of this  $\overline{\text{NMI}}$  interrupt input pin can be read by accessing the P8\_5 bit in the P8 register.

This pin cannot be used as an input port.

## 12.8 Key Input Interrupt

Of P10\_4 to P10\_7, a key input interrupt is generated when input on any of the P10\_4 to P10\_7 pins which has had the PD10\_4 to PD10\_7 bits in the PD10 register set to "0" (= input) goes low. Key input interrupts can be used as a key-on wake up function, the function which gets the microcomputer out of wait or stop mode. However, if you intend to use the key input interrupt, do not use P10\_4 to P10\_7 as analog input ports. Figure 12.12 shows the block diagram of the Key Input Interrupt. Note, however, that while input on any pin which has had the PD10\_4 to PD10\_7 bits set to "0" (= input mode) is pulled low, inputs on all other pins of the port are not detected as interrupts.

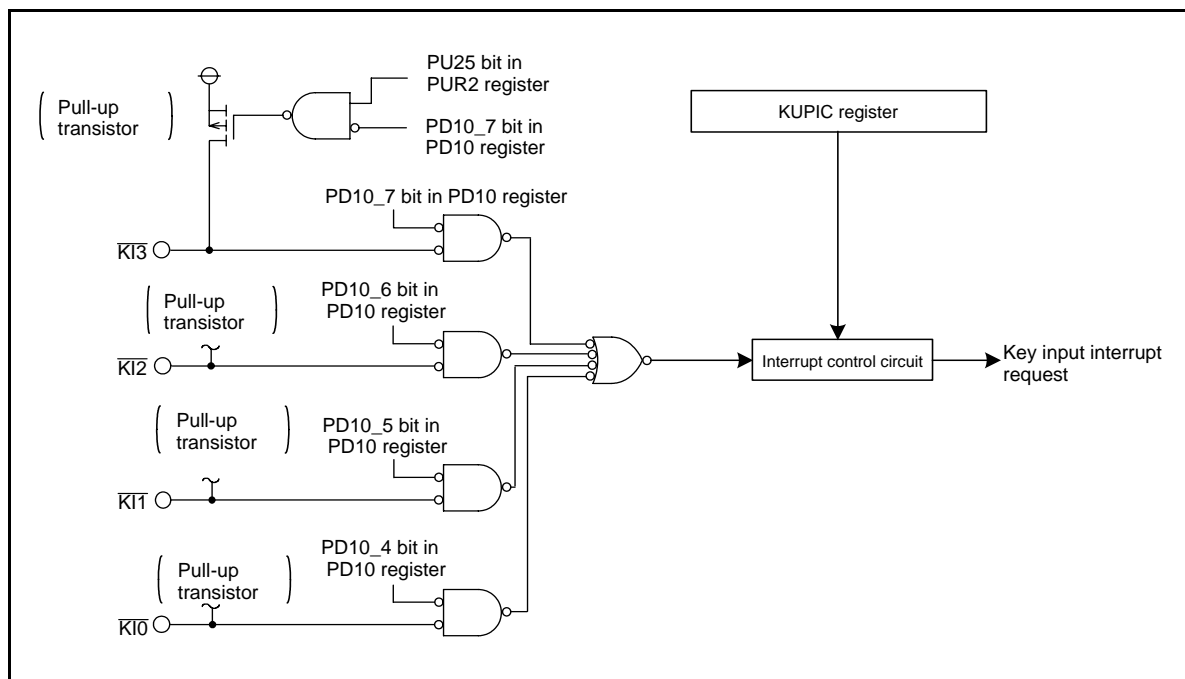


Figure 12.12 Key Input Interrupt



## 12.9 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 the AIER0 and AIER1 bits in the AIER register and AIER20 and AIER21 bits in the AIER2 register to enable or disable the interrupt. Note that the address match interrupt is unaffected by the I flag and IPL. For address match interrupts, the value of the PC that is saved to the stack area varies depending on the instruction being executed (refer to **12.5.7 Saving Registers**).

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

Table 12.6 shows the Value of the PC that is Saved to the Stack Area when an Address Match Interrupt Request is Accepted

Figure 12.13 shows the AIER, AIER2 and RMAD0 to RMAD3 Registers.

**Table 12.6 Value of the PC that is Saved to the Stack Area when an Address Match Interrupt Request is Accepted**

Instruction at the Address Indicated by the RMADi Register	Value of the PC that is saved to the stack area
<ul style="list-style-type: none"> <li>• 16-bit op-code instruction</li> <li>• Instruction shown below among 8-bit operation code instructions</li> </ul> <div> 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.B:S    #IMM8,dest  STNZ.B:S   #IMM8,dest   STZX.B:S   #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) </div>	The address indicated by the RMADi register +2
Instructions other than the above	The address indicated by the RMADi register +1

Value of the PC that is saved to the stack area : Refer to **12.5.7 Saving Registers**.

**Table 12.7 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 13	AIER21	RMAD3

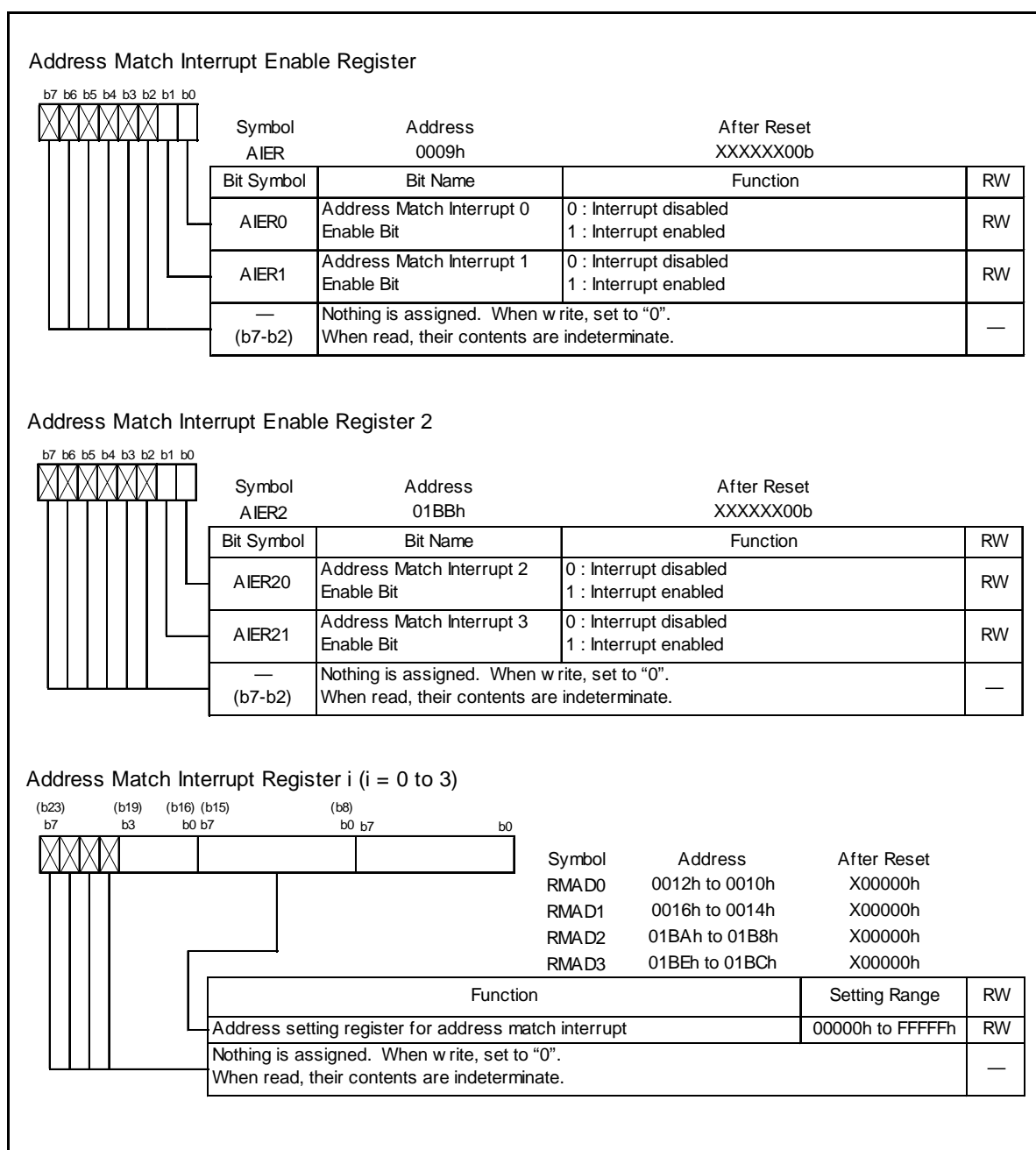


Figure 12.13 AIER, AIER2 and RMAD0 to RMAD3 Registers

## 13. Watchdog Timer

The watchdog timer is the function of detecting when the program is out of control. Therefore, we recommend using the watchdog timer to improve reliability of a system. The watchdog timer contains a 15-bit counter which counts down the clock derived by dividing the CPU clock using the prescaler. Whether to generate a watchdog timer interrupt request or apply a watchdog timer reset as an operation to be performed when the watchdog timer underflows after reaching the terminal count can be selected using the PM12 bit of PM1 register. The PM12 bit can only be set to "1" (reset). Once this bit is set to "1," it cannot be set to "0" (watchdog timer interrupt) in a program. Refer to **5.4 Watchdog Timer Reset** for the details of watchdog timer reset.

When the main clock source is selected for CPU clock, on-chip oscillator clock, PLL clock, the divide-by-N value for the prescaler can be chosen to be 16 or 128. If a sub-clock is selected for CPU clock, the divide-by-N value for the prescaler is always 2 no matter how the WDC7 bit is set. The period of watchdog timer can be calculated as given below. The period of watchdog timer is, however, subject to an error due to the prescaler.

With main clock chosen for CPU clock, on-chip oscillator clock, PLL clock

$$\text{Watchdog timer period} = \frac{\text{Prescaler dividing (16 or 128)} \times \text{Watchdog timer count (32768)}}{\text{CPU clock}}$$

With sub-clock chosen for CPU clock

$$\text{Watchdog timer period} = \frac{\text{Prescaler dividing (2)} \times \text{Watchdog timer count (32768)}}{\text{CPU clock}}$$

For example, when CPU clock = 16 MHz and the divide-by-N value for the prescaler= 16, the watchdog timer period is approx. 32.8 ms.

The watchdog timer is initialized by writing to the WDTS register. The prescaler is initialized after reset. Note that the watchdog timer and the prescaler both are inactive after reset, so that the watchdog timer is activated to start counting by writing to the WDTS register.

In stop mode, wait mode and hold state, the watchdog timer and prescaler are stopped. Counting is resumed from the held value when the modes or state are released.

Figure 13.1 shows the Watchdog Timer Block Diagram. Figure 13.2 shows the WDC and WDTS Register.

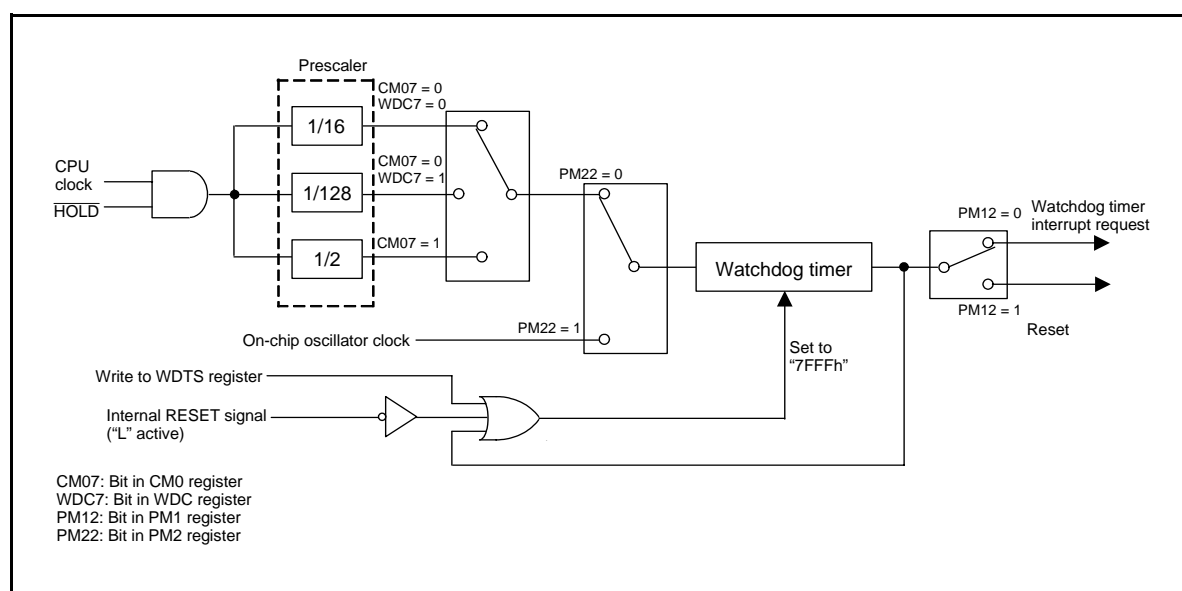


Figure 13.1 Watchdog Timer Block Diagram

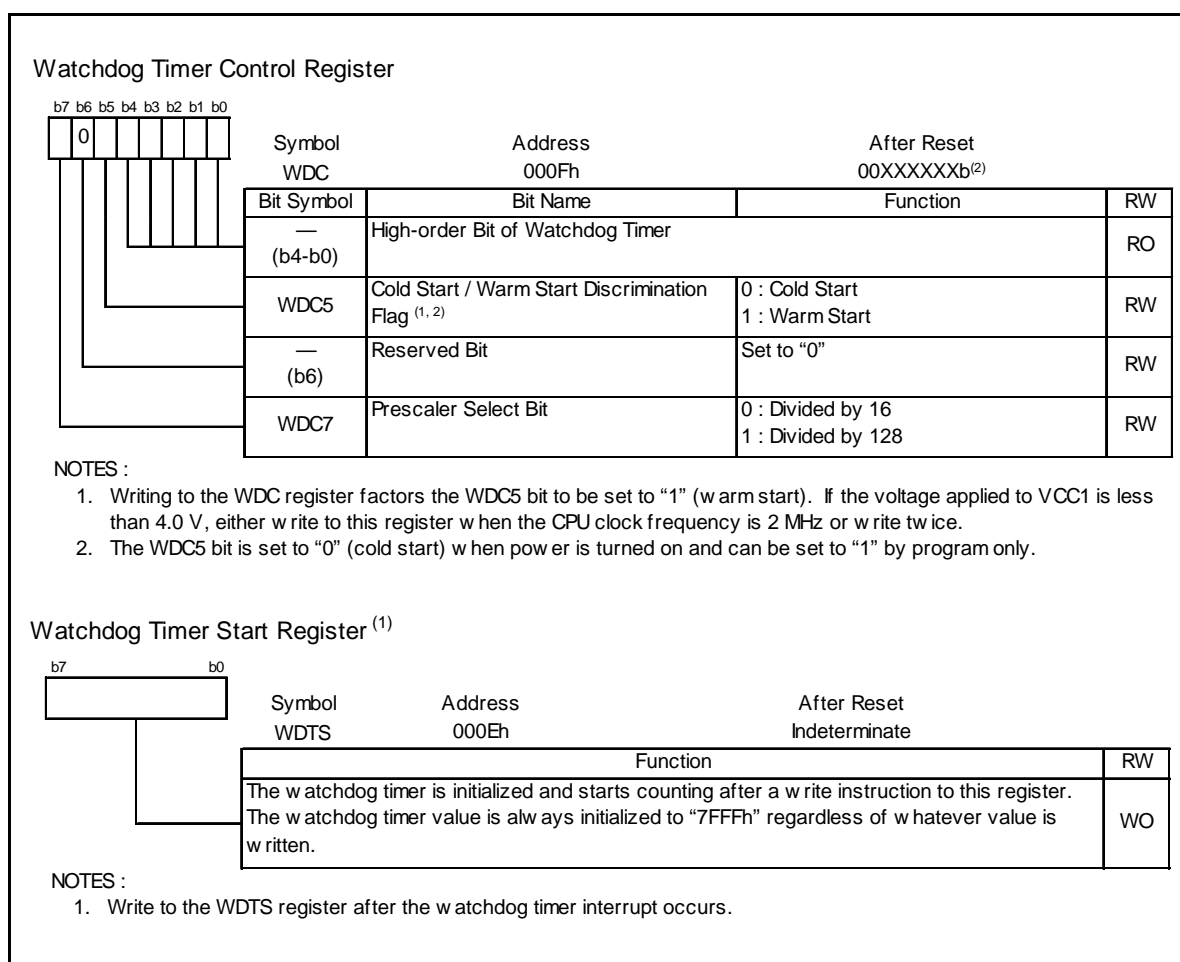


Figure 13.2 WDC and WDTS Register

### 13.1 Count source protective mode

In this mode, a on-chip oscillator clock is used for the watchdog timer count source. The watchdog timer can be kept being clocked even when CPU clock stops as a result of run-away.

Before this mode can be used, the following register settings are required:

- (1) Set the PRC1 bit in the PRCR register to "1" (enable writes to PM1 and PM2 registers).
- (2) Set the PM12 bit in the PM1 register to "1" (reset when the watchdog timer underflows).
- (3) Set the PM22 bit in the PM2 register to "1" (on-chip oscillator clock used for the watchdog timer count source).
- (4) Set the PRC1 bit in the PRCR register to "0" (disable writes to PM1 and PM2 registers).
- (5) Write to the WDTS register (watchdog timer starts counting).

Setting the PM22 bit in the PM register to "1" results in the following conditions.

- The on-chip oscillator starts oscillating, and the on-chip oscillator clock becomes the watchdog timer count source.

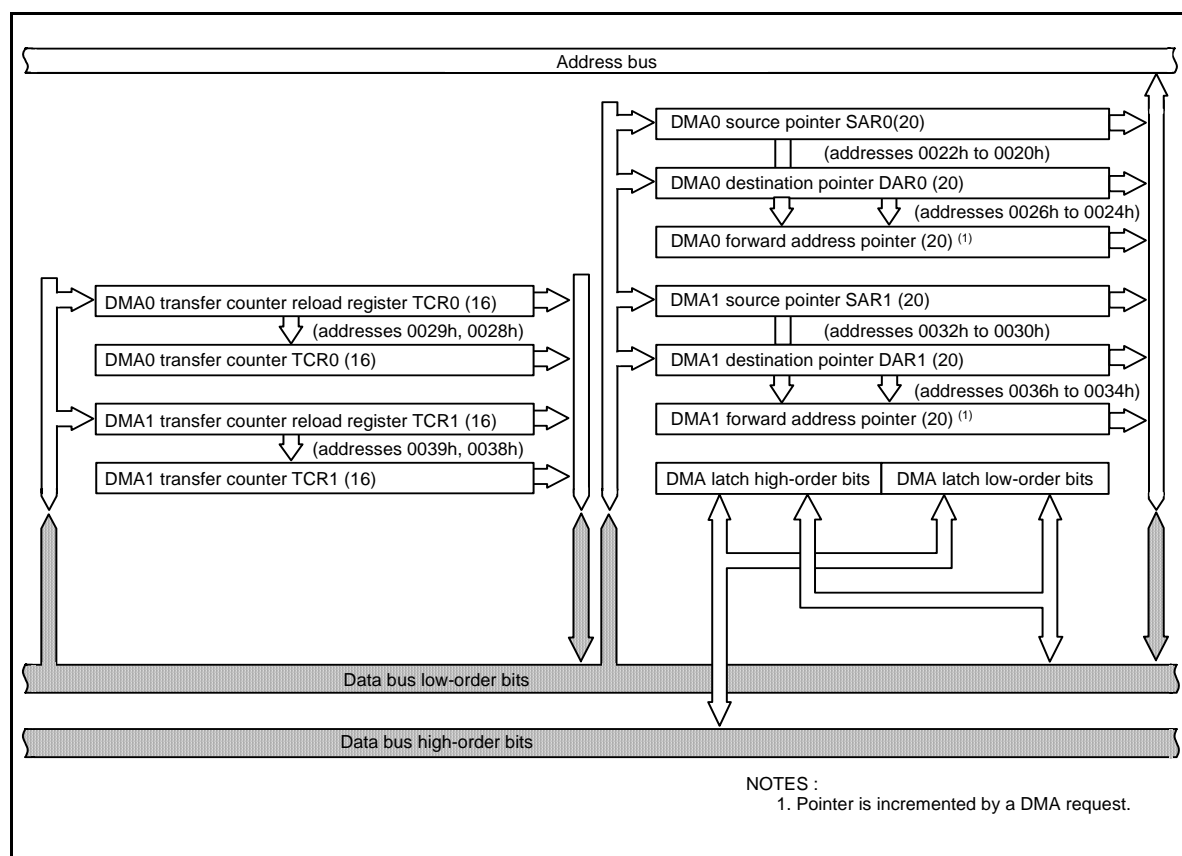
$$\text{Watchdog timer period} = \frac{\text{Watchdog timer count (32768)}}{\text{On-chip oscillator clock}}$$

- The CM10 bit in the CM1 register is disabled against write (Writing a "1" has no effect, nor is stop mode entered).
- The watchdog timer does not stop when in wait mode or hold state.

## 14. DMAC

The DMAC (Direct Memory Access Controller) allows data to be transferred without the CPU intervention.

Two DMAC channels are included. Each time a DMA request occurs, the DMAC transfers one (8 or 16-bit) data from the source address to the destination address. The DMAC uses the same data bus as used by the CPU. Because the DMAC has higher priority of 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 14.1 shows the DMAC Block Diagram. Table 14.1 lists the DMAC Specifications. Figures 14.2 to 14.4 shows the DMAC-related registers.



**Figure 14.1 DMAC Block Diagram**

A DMA request is generated by a write to the DSR bit in the DMiSL register ( $i = 0$  to  $1$ ), as well as by an interrupt request which is generated by any function specified by the DMS and DSEL3 to DSEL0 bits in the DMiSL register. However, unlike in the case of interrupt requests, DMA requests are not affected by the I flag and the interrupt control register, so that even when interrupt requests are disabled and no interrupt request can be accepted, DMA requests are always accepted. Furthermore, because the DMAC does not affect interrupts, the IR bit in the interrupt control register does not change state due to a DMA transfer.

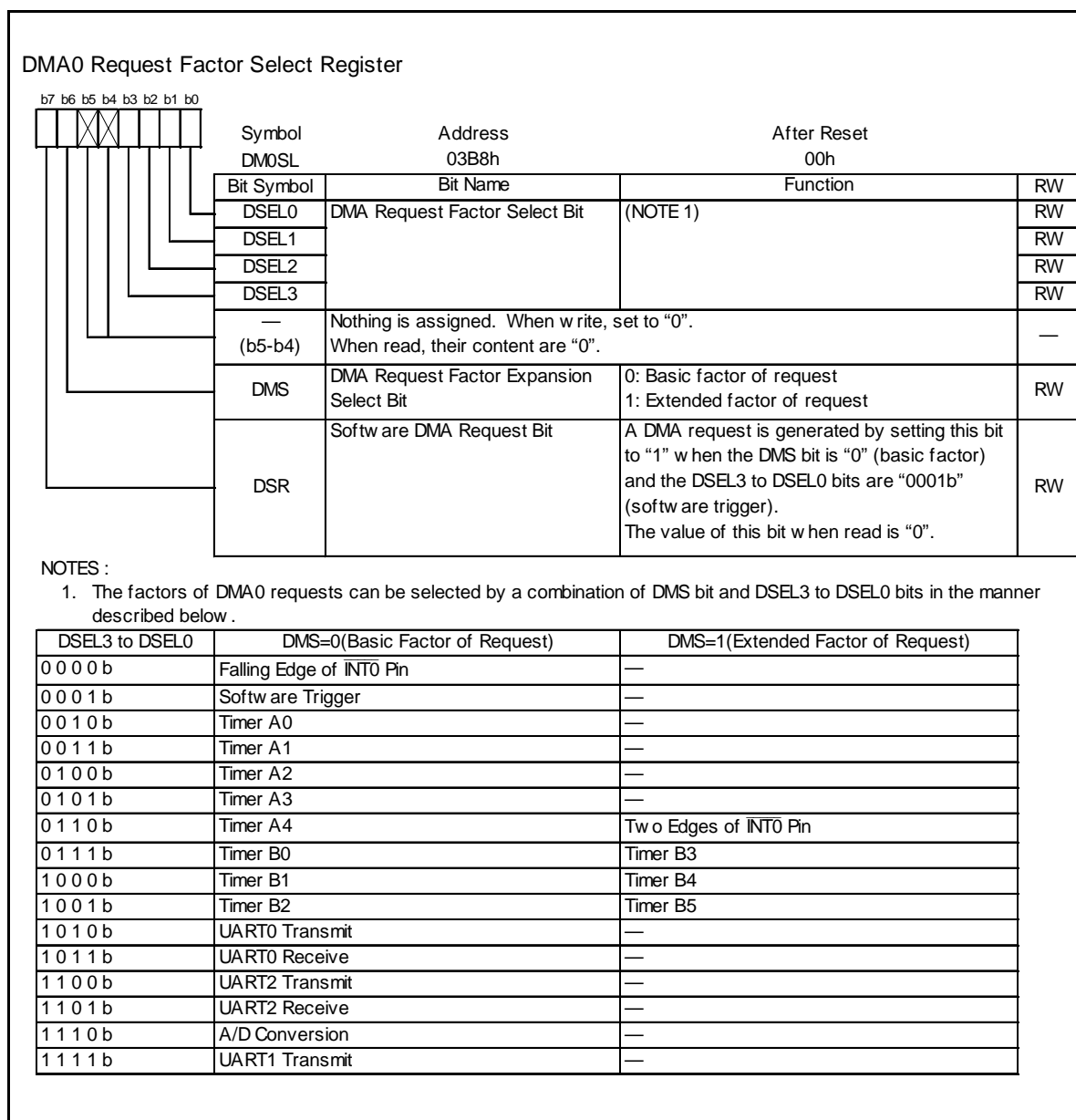
A data transfer is initiated each time a DMA request is generated when the DMAE bit in the DMiCON register = 1 (DMA enabled). However, if the cycle in which a DMA request is generated is faster than the DMA transfer cycle, the number of transfer requests generated and the number of times data is transferred may not match. Refer to **14.4 DMA Request** for details.

**Table 14.1 DMAC Specifications**

Item		Specification
No. of Channels		2 (cycle steal method)
Transfer Memory Space		<ul style="list-style-type: none"> <li>From any address in the 1-Mbyte space to a fixed address</li> <li>From a fixed address to any address in the 1-Mbyte space</li> <li>From a fixed address to a fixed address</li> </ul>
Maximum No. of Bytes Transferred		128 Kbytes (with 16-bit transfers) or 64 Kbytes (with 8-bit transfers)
DMA Request Factors <sup>(1, 2)</sup>		Falling edge of $\overline{\text{INT0}}$ or $\overline{\text{INT1}}$ Both edge of $\overline{\text{INT0}}$ or $\overline{\text{INT1}}$ Timer A0 to timer A4 interrupt requests Timer B0 to timer B5 interrupt requests UART0 transfer, UART0 reception interrupt requests UART1 transfer, UART1 reception interrupt requests UART2 transfer, UART2 reception interrupt requests SI/O3, SI/O4 interrupt requests A/D conversion interrupt requests Software triggers
Channel Priority		DMA0 > DMA1 (DMA0 takes precedence)
Transfer Unit		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 DMA <sub>i</sub> transfer counter (i = 0 to 1) underflows after reaching the terminal count.
	Repeat Transfer	When the DMA <sub>i</sub> transfer counter underflows, it is reloaded with the value of the DMA <sub>i</sub> transfer counter reload register and a DMA transfer is continued with it.
DMA Interrupt Request Generation Timing		When the DMA <sub>i</sub> transfer counter underflowed
DMA Start up		Data transfer is initiated each time a DMA request is generated when the DMAE bit in the DMA <sub>i</sub> CON register = 1 (enabled).
DMA Shutdown	Single Transfer	<ul style="list-style-type: none"> <li>When the DMAE bit is set to "0" (disabled)</li> <li>After the DMA<sub>i</sub> transfer counter underflows</li> </ul>
	Repeat Transfer	When the DMAE bit is set to "0" (disabled)
Reload Timing for Forward Address Pointer and 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 SAR <sub>i</sub> or the DAR <sub>i</sub> pointer whichever is specified to be in the forward direction and the DMA <sub>i</sub> transfer counter is reloaded with the value of the DMA <sub>i</sub> transfer counter reload register.
DMA Transfer Cycles		Minimum 3 cycles between SFR and internal RAM

**NOTES:**

1. DMA transfer is not effective to any interrupt. DMA transfer is affected neither by the I flag nor by the interrupt control register.
2. The selectable factors of DMA requests differ with each channel.
3. Make sure that no DMAC-related registers (addresses 0020h to 003Fh) are accessed by the DMAC.

**Figure 14.2 DM0SL Register**

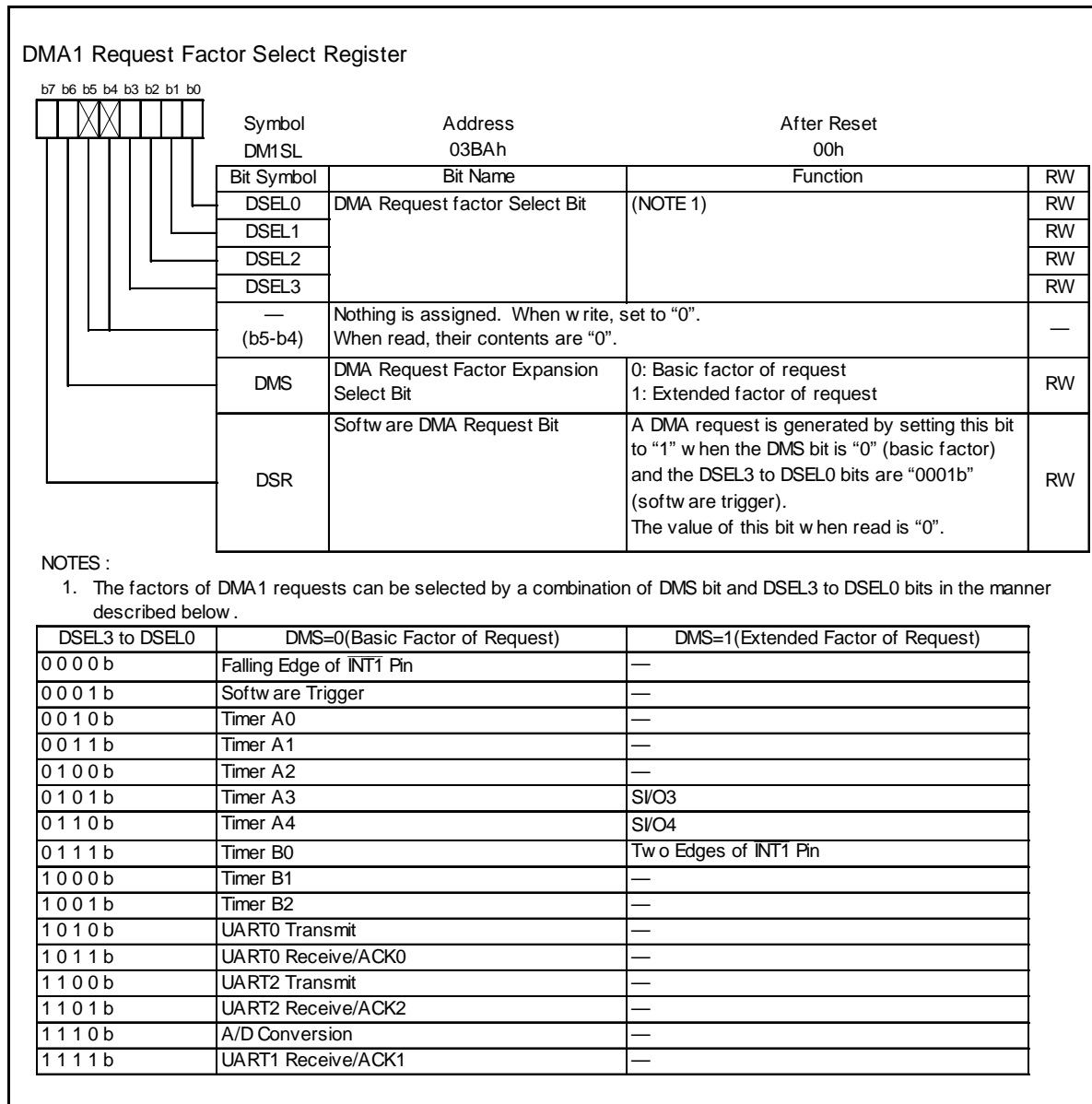
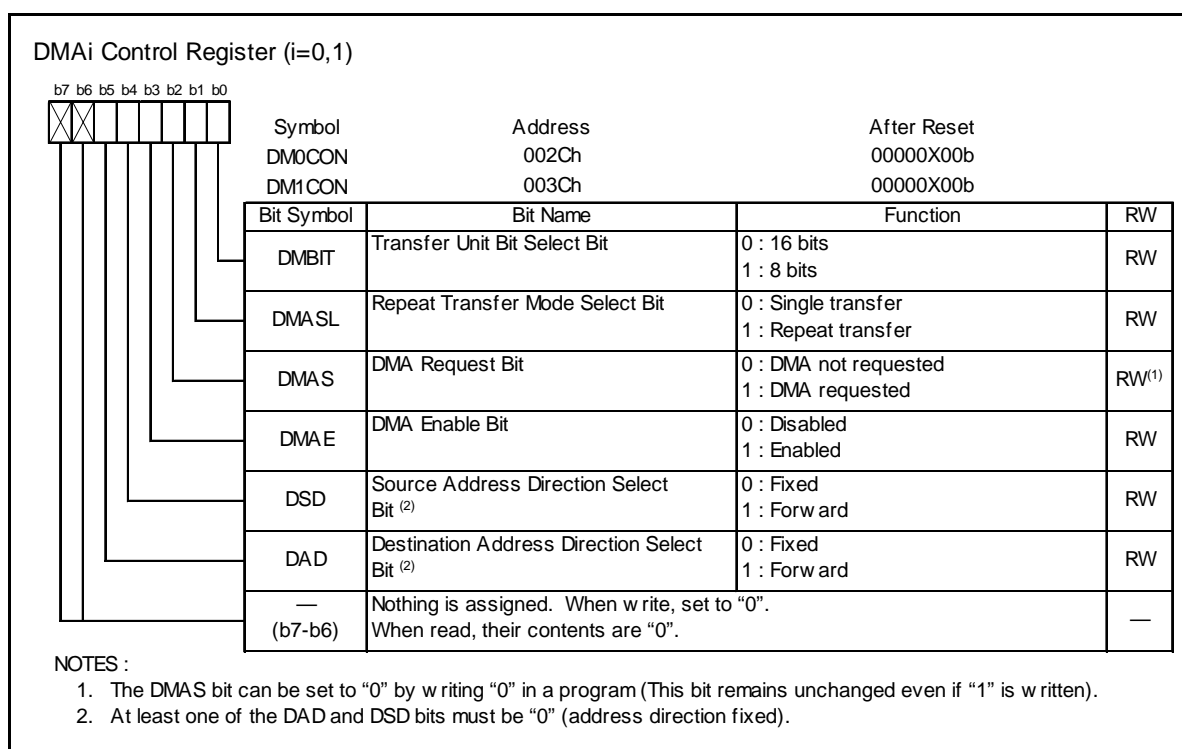


Figure 14.3 DM1SL Register



**Figure 14.4 DM0CON and DM1CON Register**

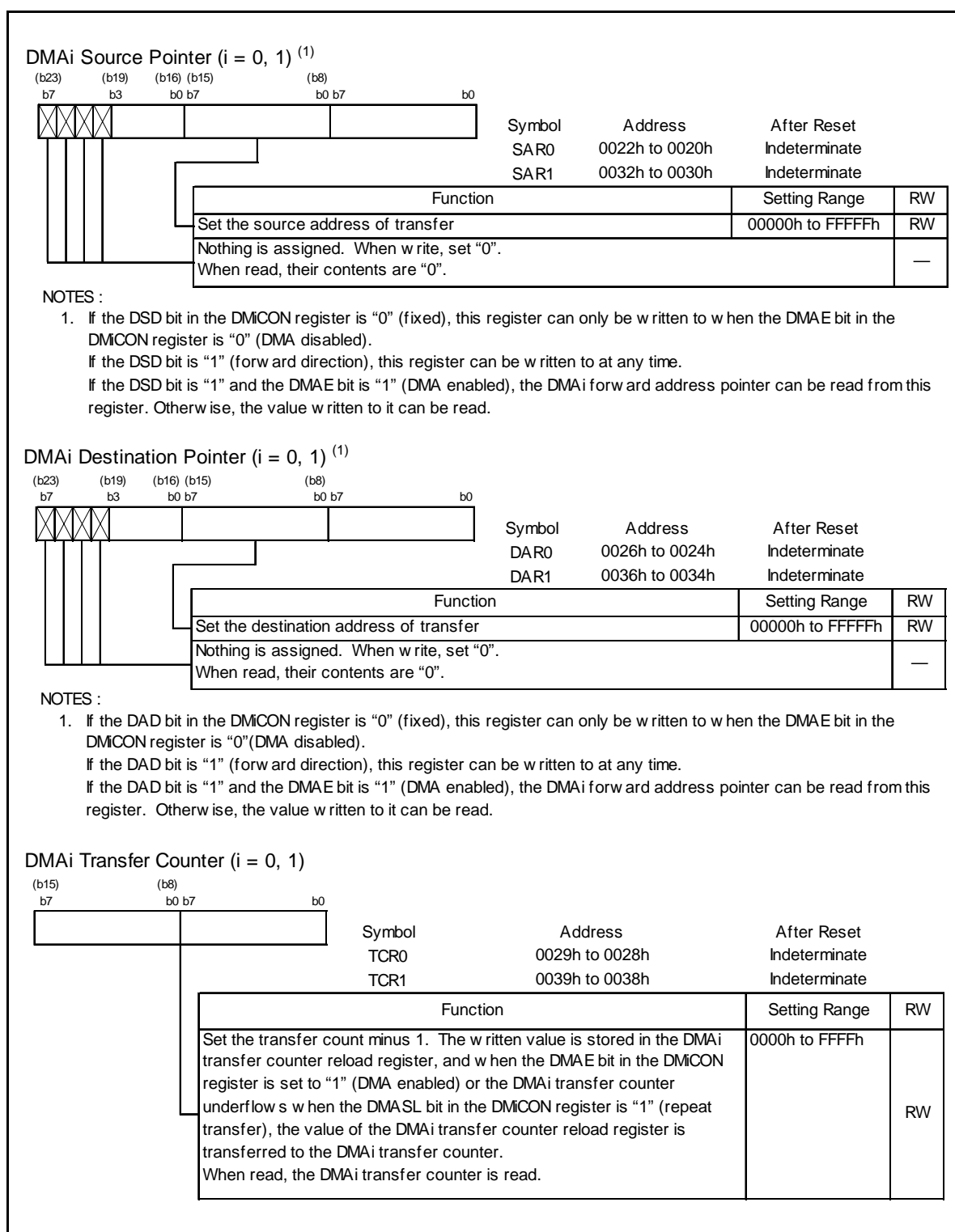


Figure 14.5 SAR0, SAR1, DAR0, DAR1, TCR0 and TCR1 Registers

## 14.1 Transfer Cycles

The transfer cycle consists of a memory or SFR read (source read) bus cycle and a write (destination write) bus cycle. The number of read and write bus cycles is affected by the source and destination addresses of transfer. During memory extension and microprocessor modes, it is also affected by the BYTE pin level. Furthermore, the bus cycle itself is extended by a software wait or  $\overline{\text{RDY}}$  signal.

### 14.1.1 Effect of Source and Destination Addresses

If the transfer unit and data bus both are 16 bits and the source address of transfer begins with an odd address, the source read cycle consists of one more bus cycle than when the source address of transfer begins with an even address.

Similarly, if the transfer unit and data bus both are 16 bits and the destination address of transfer begins with an odd address, the destination write cycle consists of one more bus cycle than when the destination address of transfer begins with an even address.

### 14.1.2 Effect of BYTE Pin Level

During memory extension and microprocessor modes, if 16 bits of data are to be transferred on an 8-bit data bus (input on the BYTE pin = high), the operation is accomplished by transferring 8 bits of data twice. Therefore, this operation requires two bus cycles to read data and two bus cycles to write data. Furthermore, if the DMAC is to access the internal area (internal ROM, internal RAM, or SFR), unlike in the case of the CPU, the DMAC does it through the data bus width selected by the BYTE pin.

### 14.1.3 Effect of Software Wait

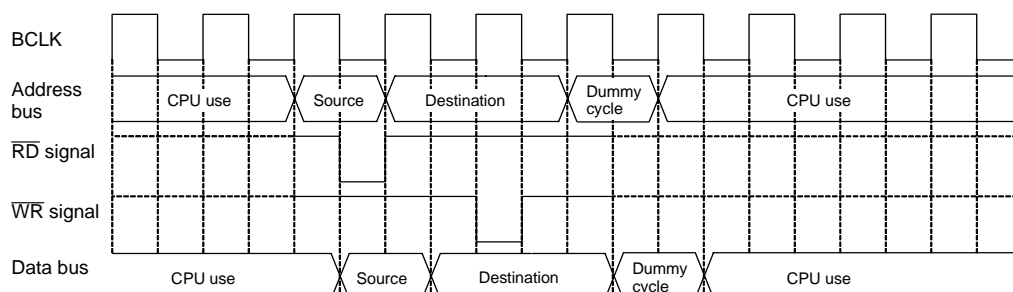
For memory or SFR accesses in which one or more software wait states are inserted, the number of bus cycles required for that access increases by an amount equal to software wait states.

### 14.1.4 Effect of $\overline{\text{RDY}}$ Signal

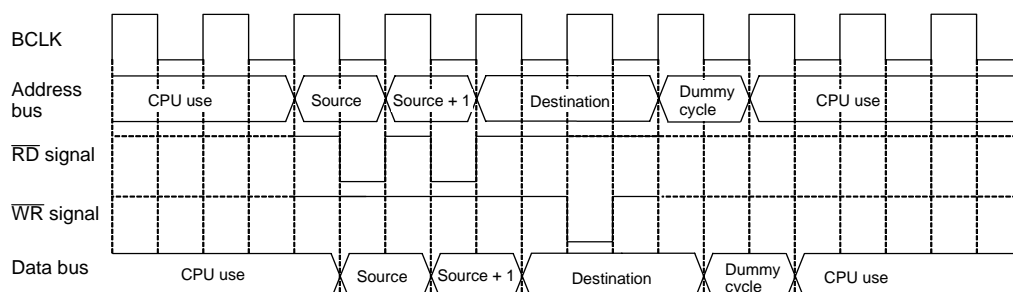
During memory extension and microprocessor modes, DMA transfers to and from an external area are affected by the  $\overline{\text{RDY}}$  signal. Refer to **8.2.6  $\overline{\text{RDY}}$  Signal**.

Figure 14.6 shows the example of the Transfer Cycles for Source Read. For convenience, the destination write cycle is shown as one 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, respectively. For example, when data is transferred in 16 bit units using an 8-bit bus ((2) on Figure 14.6), two source read bus cycles and two destination write bus cycles are required.

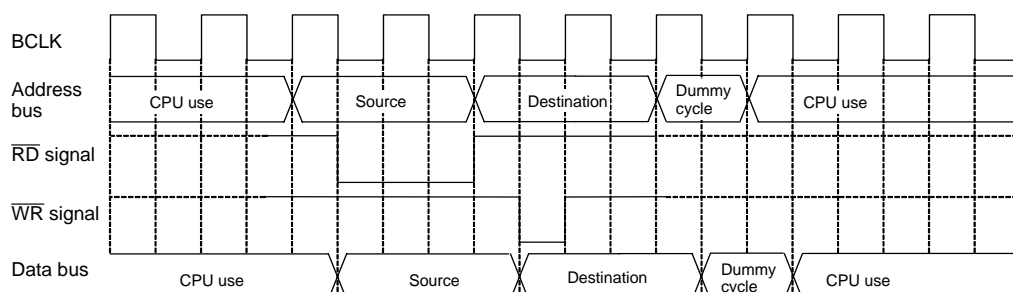
(1) When the transfer unit is 8 or 16 bits and the source of transfer is an even address



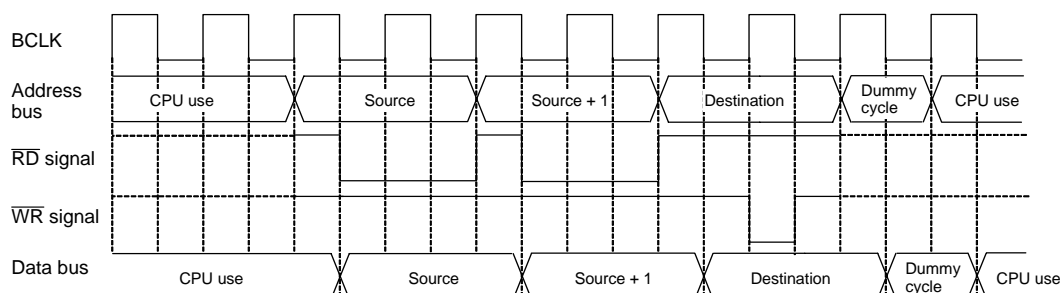
(2) When the transfer unit is 16 bits and the source address of transfer is an odd address, or when the transfer unit is 16 bits and an 8-bit bus is used



(3) When the source read cycle under condition (1) has one wait state inserted



(4) When the source read cycle under condition (2) has one wait state inserted



#### NOTES :

1. The same timing changes occur with the respective conditions at the destination as at the source.

**Figure 14.6 Transfer Cycles for Source Read**

## 14.2 DMA Transfer Cycles

Any combination of even or odd transfer read and write addresses is possible. Table 14.2 lists the DMA Transfer Cycles. Table 14.3 lists the Coefficient j, k.

The number of DMAC transfer cycles can be calculated as follows:

$$\text{No. of transfer cycles per transfer unit} = \text{No. of read cycles} \times j + \text{No. of write cycles} \times k$$

**Table 14.2 DMA Transfer Cycles**

Transfer Unit	Bus Width	Access Address	Single-Chip Mode		Memory Expansion Mode Microprocessor Mode	
			No. of Read Cycles	No. of Write Cycles	No. of Read Cycles	No. of Write Cycles
8-bit Transfers (DMBIT= 1)	16-bit (BYTE= L)	Even	1	1	1	1
		Odd	1	1	1	1
	8-bit (BYTE = H)	Even	—	—	1	1
		Odd	—	—	1	1
16-bit Transfers (DMBIT= 0)	16-bit (BYTE = L)	Even	1	1	1	1
		Odd	2	2	2	2
	8-bit (BYTE = H)	Even	—	—	2	2
		Odd	—	—	2	2

— : This condition does not exist.

**Table 14.3 Coefficient j, k**

	Internal Area				External Area						
	Internal ROM, RAM		SFR		Separate Bus				Multiplex Bus		
	No Wait	With Wait	1-Wait (2)	2-Wait (2)	No Wait	With Wait (1)			With Wait (1)		
						1-Wait	2-Wait	3-Wait	1-Wait	2-Wait	3-Wait
j	1	2	2	3	1	2	3	4	3	3	4
k	1	2	2	3	2	2	3	4	3	3	4

**NOTES:**

1. Depends on the set value of CSE register.
2. Depends on the set value of PM20 bit in the PM2 register.

### 14.3 DMA Enable

When a data transfer starts after setting the DMAE bit in the DMiCON register ( $i = 0, 1$ ) to “1” (enabled), the DMAC operates as follows:

- (1) Reload the forward address pointer with the SAR<sub>i</sub> register value when the DSD bit in the DMiCON register is “1” (forward) or the DAR<sub>i</sub> register value when the DAD bit in the DMiCON register is “1” (forward).
- (2) Reload the DMA<sub>i</sub> transfer counter with the DMA<sub>i</sub> transfer counter reload register value.

If the DMAE bit is set to “1” again while it remains set, the DMAC performs the above operation. However, if a DMA request may occur simultaneously when the DMAE bit is being written, follow the steps below.

Step 1: Write “1” to the DMAE bit and DMAS bit in the DMiCON register simultaneously.

Step 2: Make sure that the DMA<sub>i</sub> is in an initial state as described above (1) and (2) in a program.

If the DMA<sub>i</sub> is not in an initial state, the above steps should be repeated.

### 14.4 DMA Request

The DMAC can generate a DMA request as triggered by the factor of request that is selected with the DMS and DSEL3 to DSEL0 bits in the DMiSL register ( $i = 0, 1$ ) on either channel. Table 14.4 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 was set to “1” (enabled) when this occurred, the DMAS bit is set to “0” (DMA not requested) immediately before a data transfer starts. This bit cannot be set to “1” in a program (it can only be set to “0”).

The DMAS bit may be set to “1” when the DMS or the DSEL3 to DSEL0 bits change state. Therefore, always be sure to set the DMAS bit to “0” after changing the DMS or the DSEL3 to DSEL0 bits.

Because if the DMAE bit is “1”, a data transfer starts immediately after a DMA request is generated, the DMAS bit in almost all cases is “0” when read in a program. Read the DMAE bit to determine whether the DMAC is enabled.

**Table 14.4 Timing at Which the DMAS Bit Changes State**

DMA Factor	DMAS Bit of 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”	<ul style="list-style-type: none"> <li>• Immediately before a data transfer starts</li> <li>• When set by writing “0” in a program</li> </ul>
Peripheral Function	When the interrupt control register for the peripheral function that is selected by the DSEL3 to DSEL0 and DMS bits in the DMiSL register has its IR bit set to “1”	

### 14.5 Channel Priority and DMA Transfer Timing

If both DMA0 and DMA1 are enabled and DMA transfer request signals from DMA0 and DMA1 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 channel priority, DMA0 > DMA1. The following describes DMAC operation when DMA0 and DMA1 requests are detected active in the same sampling period. Figure 14.7 shows an example of DMA Transfer by External Factors.

DMA0 request having priority is received first to start a transfer when a DMA0 request and DMA1 request are generated simultaneously. After one DMA0 transfer is completed, a bus arbitration 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 arbitration is again returned to the CPU.

In addition, DMA requests cannot be counted up since each channel has one DMAS bit. Therefore, when DMA requests, as DMA1 in Figure 14.7, occurs more than one time, the DMAS bit is set to “0” as soon as getting the bus arbitration.

Refer to **8.2.7 Hold Signal** for details about bus arbitration between the CPU and DMA.

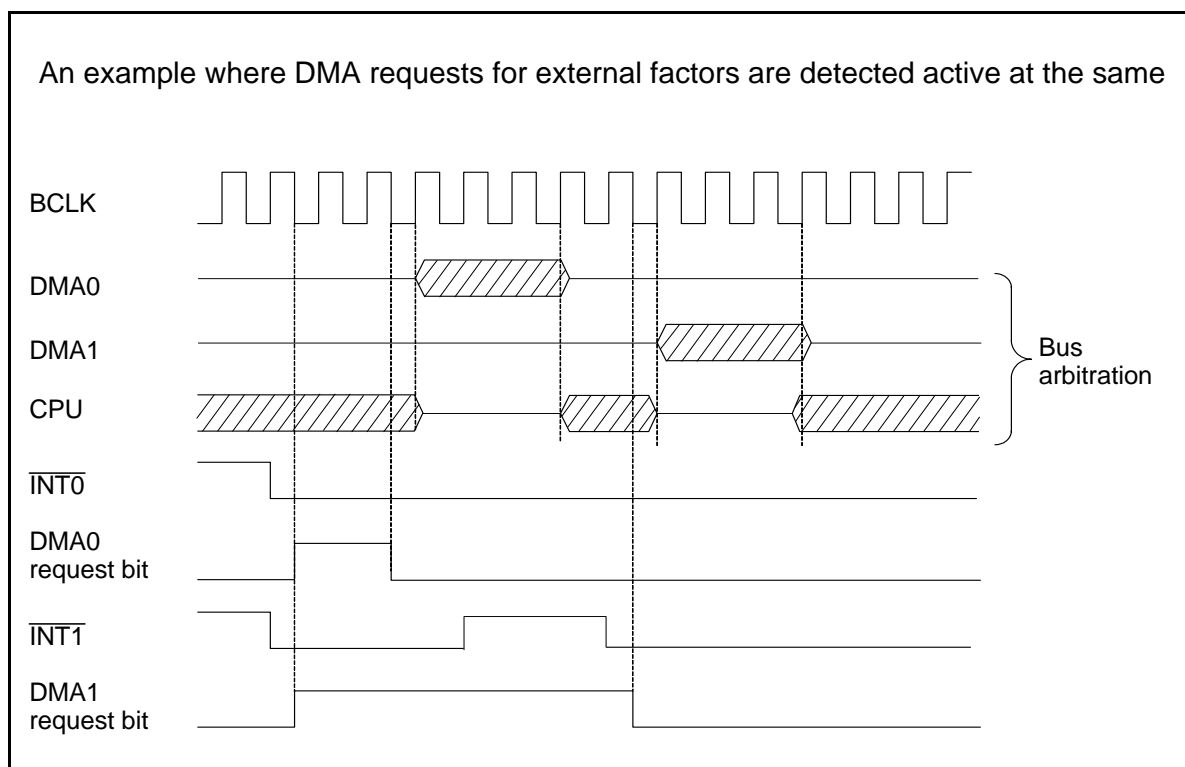


Figure 14.7 DMA Transfer by External Factors

## 15. Timers

### Note

The M16C/62P (80-pin version) and M16C/62PT (80-pin version) do not include TA1IN, TA1OUT, TA2IN, TA2OUT and TB pins. Do not use the function which needs these pins.

Eleven 16-bit timers, each capable of operating independently of the others, can be classified by function as either Timer A (five) and Timer B (six). The count source for each timer acts as a clock, to control such timer operations as counting, reloading, etc. Figures 15.1 and 15.2 show block diagrams of Timer A and Timer B configuration, respectively.

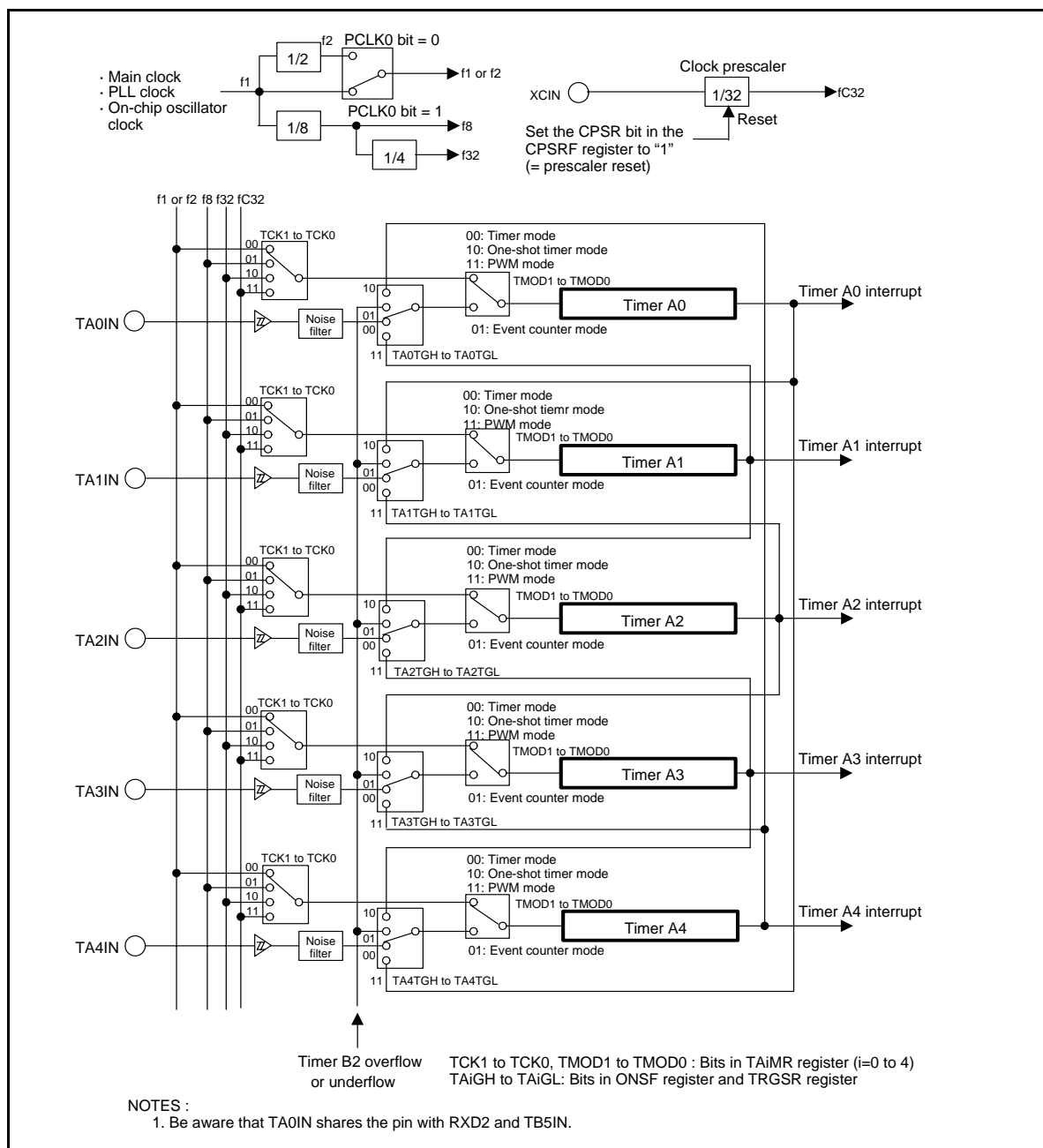


Figure 15.1 Timer A Configuration



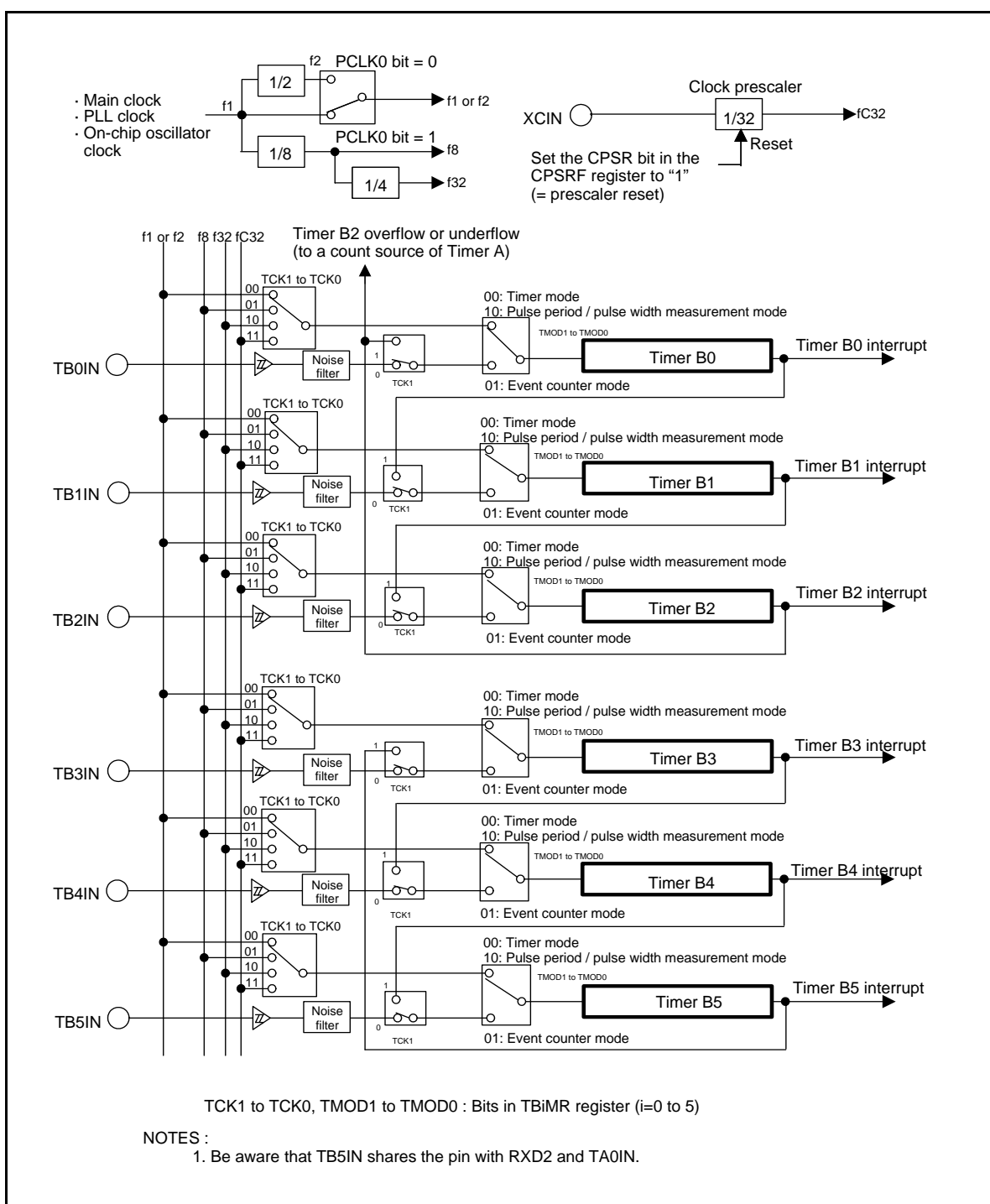


Figure 15.2 Timer B Configuration

## 15.1 Timer A

### Note

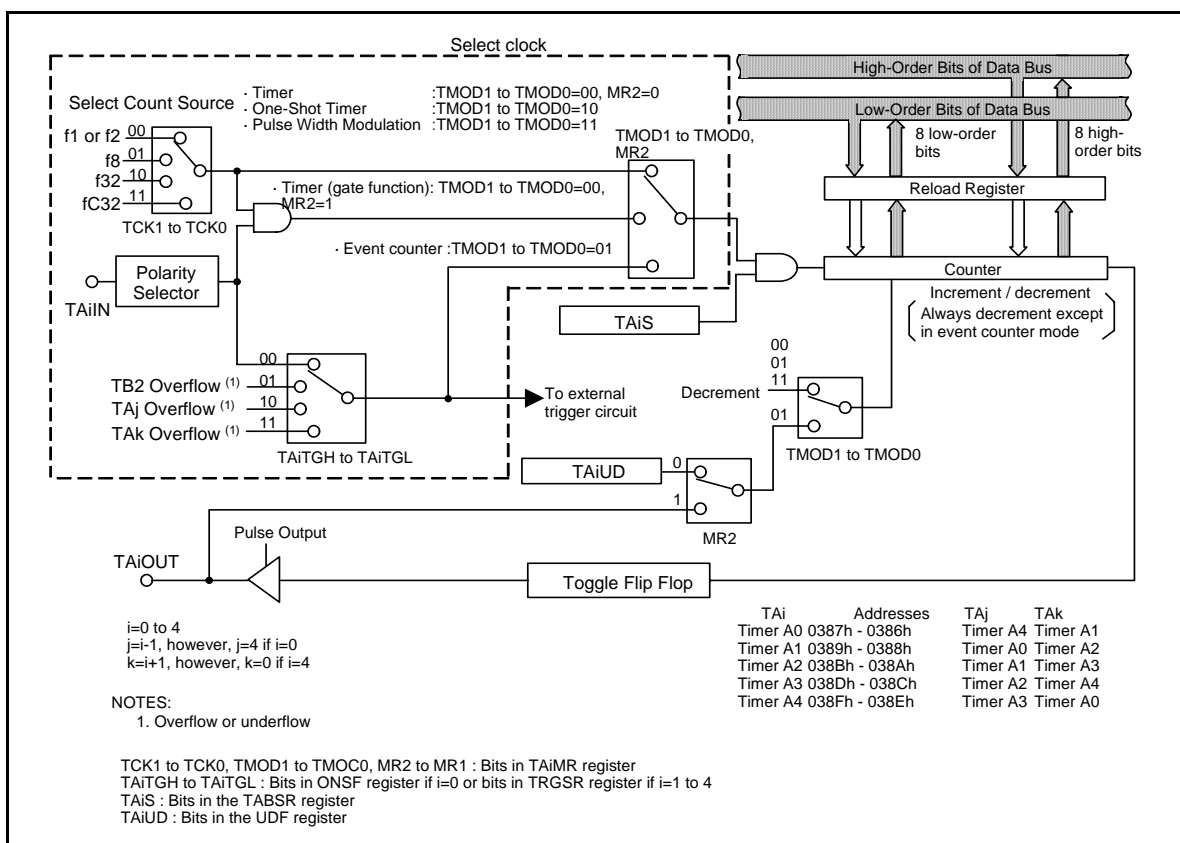
The M16C/62P (80-pin version) and M16C/62PT (80-pin version) do not include TA1IN and TA1OUT pins of Timer A1, and TA2IN and TA2OUT pins of Timer A2.

### [Precautions when using Timer A1 and Timer A2]

- |                               |   |
|-------------------------------|---|
| • Timer Mode                  | The Gate Function and the Pulse Output Function cannot be used. Set the MR2 to MR0 bits in the TA1MR and TA2MR registers to “000b” when using Timer Mode.   |
| • Event Counter Mode          | The Pulse Output Function cannot be used and external input signals cannot be counted. Two-phase Pulse Signal of Timer A2 cannot be used. Set the MR2 to MR0 bits in the TA1MR and TA2MR registers to “000b” when using the Event Counter Mode. |
| • One-shot Timer Mode         | The Pulse Output Function cannot be used and count start by the external trigger cannot be counted. Set the MR1 to MR0 bits in the TA1MR and TA2MR registers to “00b” when using the One-shot Timer Mode.                                       |
| • Pulse Width Modulation Mode | PWM pulse cannot be outputted.  |

Figure 15.3 shows a Timer A Block Diagram. Figures 15.4 to 15.7 show registers related to Timer A. Timer A supports the following four modes. Except in event counter mode, Timers A0 to A4 all have the same function. Use the TMOD1 to TMOD0 bits in the TAI<sub>i</sub>MR register (i = 0 to 4) to select the desired 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. |
| • One-shot Timer Mode:               | The timer outputs a pulse only once before it reaches the minimum count “0000h”.             |
| • Pulse Width Modulation (PWM) Mode: | The timer outputs pulses in a given width successively.                                      |



**Figure 15.3**     **Timer A Block Diagram**

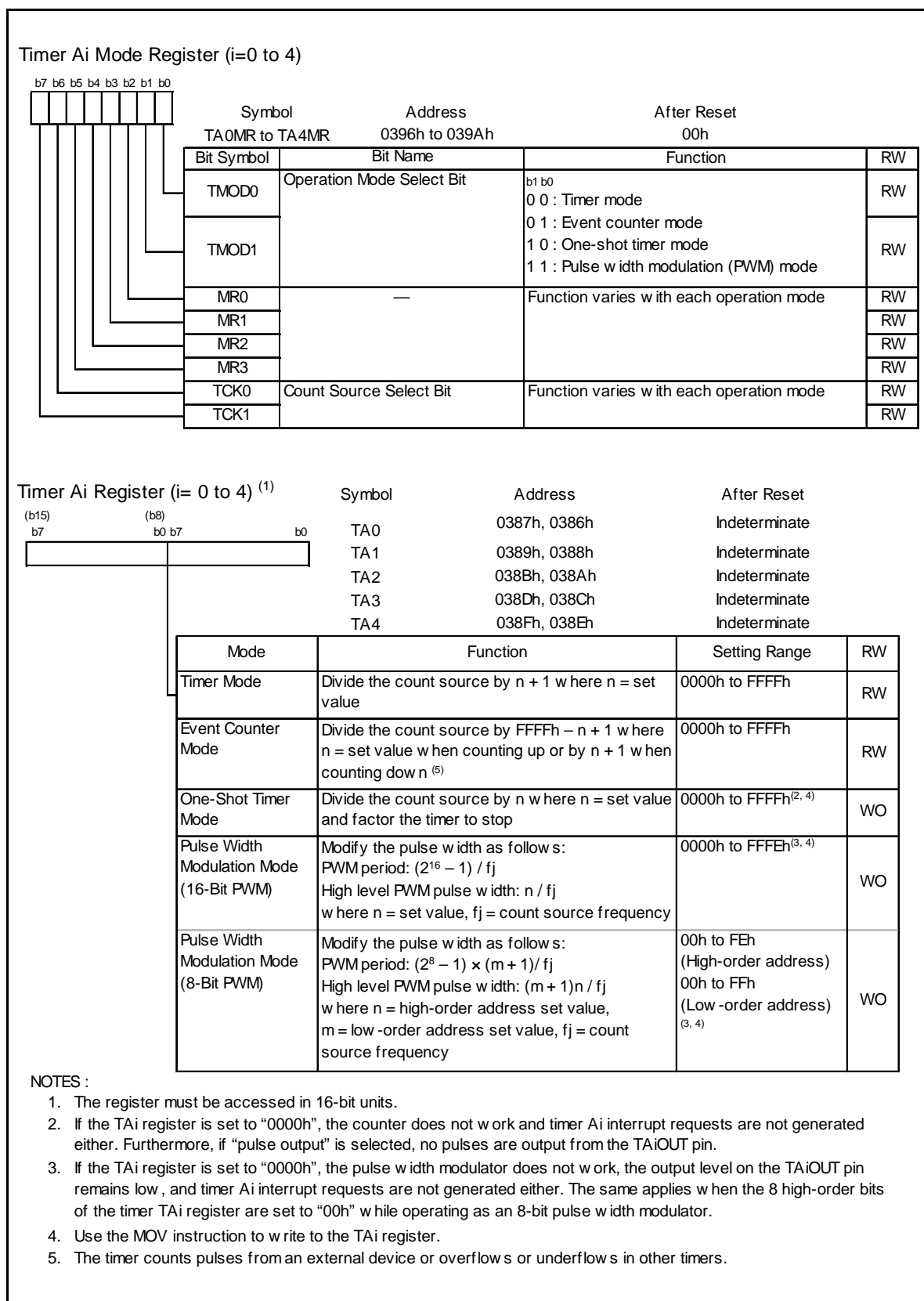


Figure 15.4 TAIiMR and TAI Registers

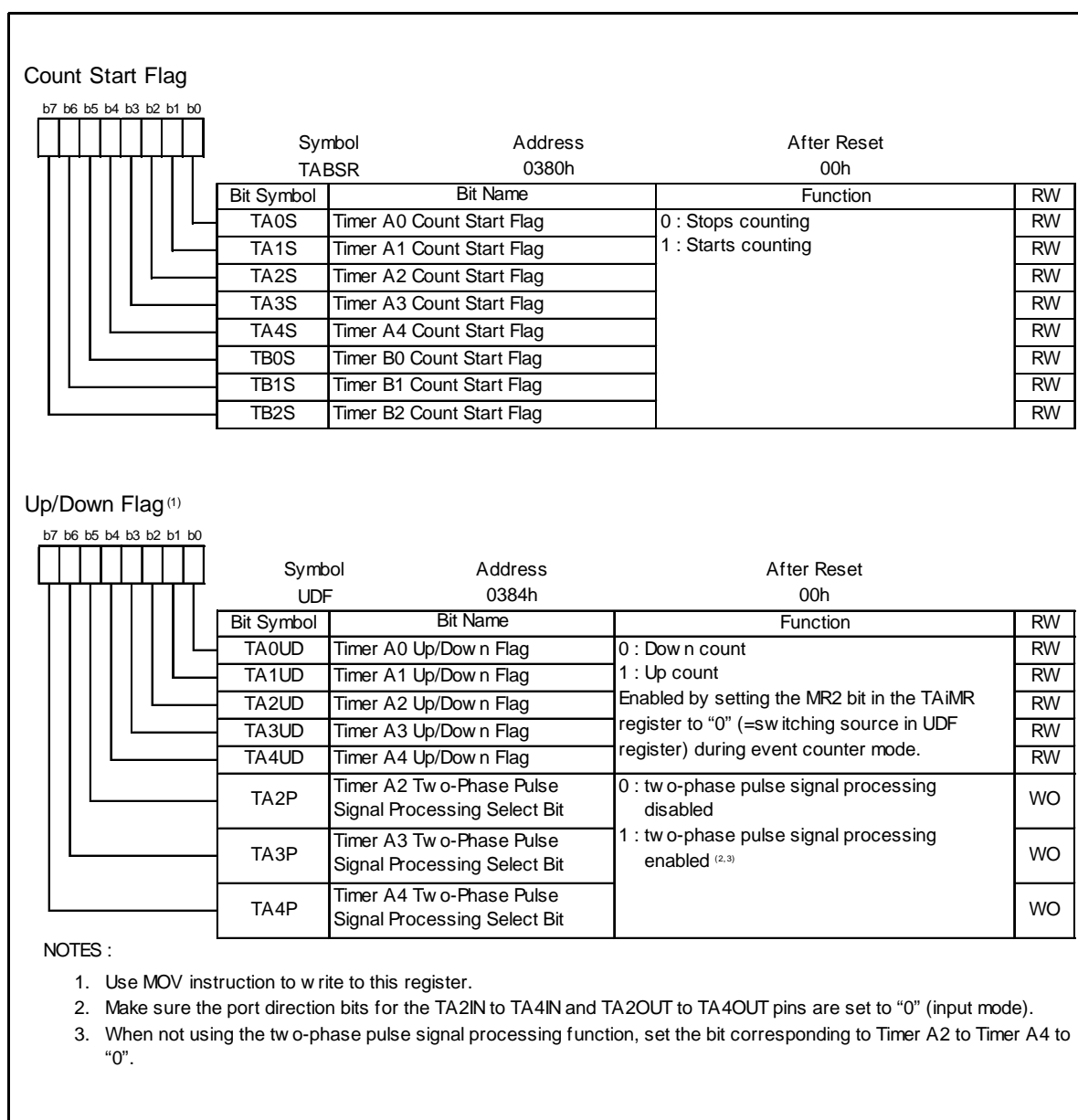


Figure 15.5 TABSR and UDF Registers

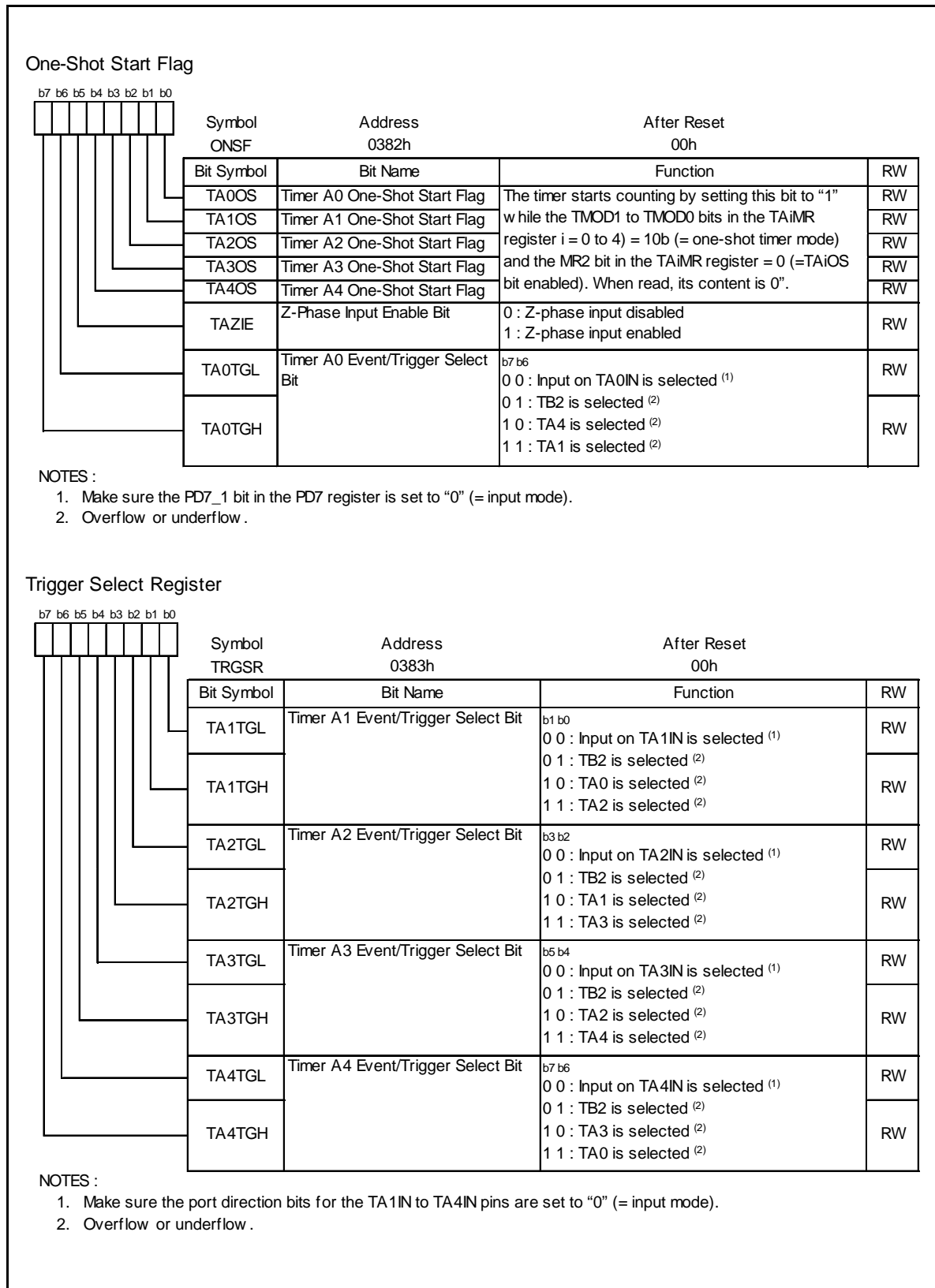


Figure 15.6 ONSF, RGSR Registers

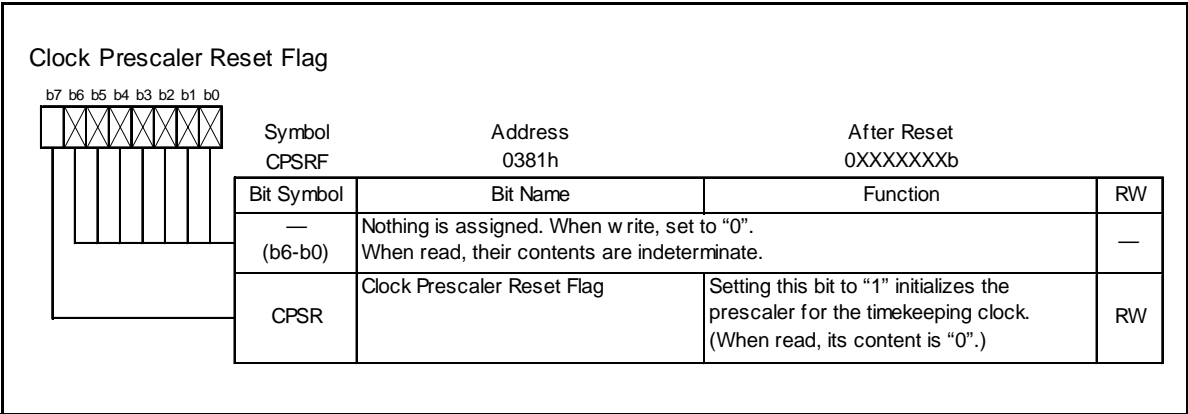


Figure 15.7 CPSRF Register

### 15.1.1 Timer Mode

In timer mode, the timer counts a count source generated internally (see Table 15.1). Figure 15.8 shows TAI<sub>MR</sub> Register in Timer Mode.

**Table 15.1 Specifications in Timer Mode**

Item	Specification
Count source	f1, f2, f8, f32, fC32
Count Operation	<ul style="list-style-type: none"> <li>• Down-count</li> <li>• When the timer underflows, it reloads the reload register contents and continues counting</li> </ul>
Divide Ratio	1/(n+1) n: set value of TAI register (i= 0 to 4) 0000h to FFFFh
Count Start Condition	Set TAI <sub>S</sub> bit in TABSR register to "1" (= start counting)
Count Stop Condition	Set TAI <sub>S</sub> bit to "0" (= stop counting)
Interrupt Request Generation Timing	Timer underflow
TAI <sub>IN</sub> Pin Function	I/O port or gate input
TAI <sub>OUT</sub> Pin Function	I/O port or pulse output
Read from Timer	Count value can be read by reading TAI register
Write to Timer	<ul style="list-style-type: none"> <li>• When not counting and until the 1st count source is input after counting start Value written to TAI register is written to both reload register and counter</li> <li>• When counting (after 1st count source input) Value written to TAI register is written to only reload register (Transferred to counter when reloaded next)</li> </ul>
Select Function	<ul style="list-style-type: none"> <li>• Gate function Counting can be started and stopped by an input signal to TAI<sub>IN</sub> pin</li> <li>• Pulse output function Whenever the timer underflows, the output polarity of TAI<sub>OUT</sub> pin is inverted. When TAI<sub>S</sub> bit is set to "0" (stop counting), the pin outputs a low.</li> </ul>

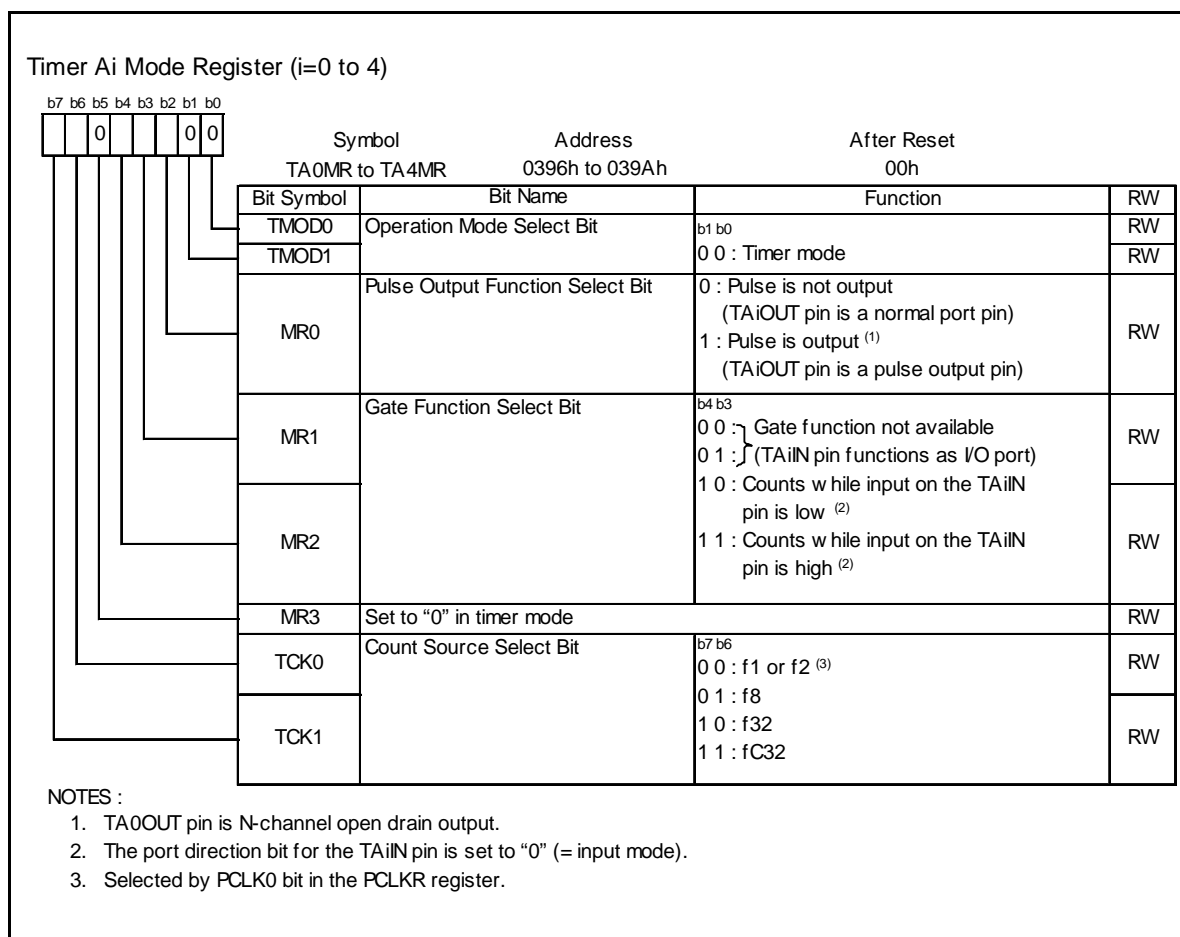


Figure 15.8 TAiMR Register in Timer Mode



### 15.1.2 Event Counter Mode

In event counter mode, the timer counts pulses from an external device or overflows and underflows of other timers. Timer A2, A3 and A4 can count two-phase external signals. Table 15.2 lists Specifications in Event Counter Mode (when not processing two-phase pulse signal). Table 15.3 lists Specifications in Event Counter Mode (when processing two-phase pulse signal with Timer A2, A3 and A4). Figure 15.9 shows TAIiMR Register in Event Counter Mode (when not using two-phase pulse signal processing). Figure 15.10 shows TA2MR to TA4MR Registers in Event Counter Mode (when using two-phase pulse signal processing with Timer A2, A3 and A4).

**Table 15.2 Specifications in Event Counter Mode (when not processing two-phase pulse signal)**

Item	Specification
Count Source	<ul style="list-style-type: none"> <li>External signals input to TAIiN pin (i=0 to 4) (effective edge can be selected in program)</li> <li>Timer B2 overflows or underflows,</li> <li>Timer Aj (j=i-1, except j=4 if i=0) overflows or underflows,</li> <li>Timer Ak (k=i+1, except k=0 if i=4) overflows or underflows</li> </ul>
Count Operation	<ul style="list-style-type: none"> <li>Up-count or down-count can be selected by external signal or program</li> <li>When the timer overflows or underflows, it reloads the reload register contents and continues counting. When operating in free-running mode, the timer continues counting without reloading.</li> </ul>
Divided Ratio	1/ (FFFFh - n + 1) for up-count 1/ (n + 1) for down-count n : set value of TAI register 0000h to FFFFh
Count Start Condition	Set TAIiS bit in the TABSR register to "1" (= start counting)
Count Stop Condition	Set TAIiS bit to "0" (= stop counting)
Interrupt Request Generation Timing	Timer overflow or underflow
TAIiN Pin Function	I/O port or count source input
TAIiOUT Pin Function	I/O port, pulse output, or up/down-count select input
Read from Timer	Count value can be read by reading TAIi register
Write to Timer	<ul style="list-style-type: none"> <li>When not counting and until the 1st count source is input after counting start Value written to TAIi register is written to both reload register and counter</li> <li>When counting (after 1st count source input) Value written to TAIi register is written to only reload register (Transferred to counter when reloaded next)</li> </ul>
Select Function	<ul style="list-style-type: none"> <li>Free-run count function Even when the timer overflows or underflows, the reload register content is not reloaded to it</li> <li>Pulse output function Whenever the timer underflows or underflows, the output polarity of TAIiOUT pin is inverted. When TAIiS bit is set to "0" (stop counting), the pin outputs a low.</li> </ul>

Timer Ai Mode Register (i=0 to 4)

(when not using two-phase pulse signal processing)

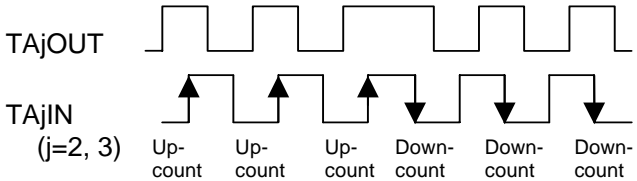
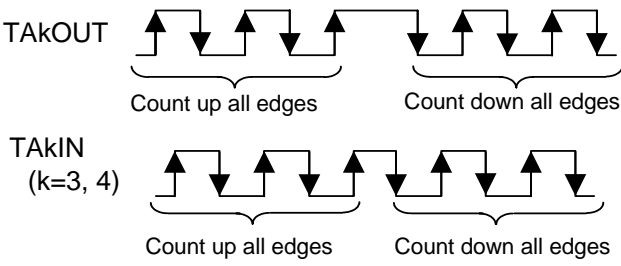
b7 b6 b5 b4 b3 b2 b1 b0								Symbol	Address	After Reset
<div><div></div><div>0</div><div></div><div></div><div></div><div>0</div><div>1</div></div>								TA0MR to TA4MR	0396h to 039Ah	00h
Bit Symbol		Bit Name		Function		RW				
TMOD0		Operation Mode Select Bit		b1 b0		RW				
TMOD1				0 1 : Event counter mode <sup>(1)</sup>		RW				
MR0		Pulse Output Function Select Bit		0 : Pulse is not output (TAiOUT pin functions as I/O port) 1 : Pulse is output <sup>(2)</sup> (TAiOUT pin functions as pulse output pin)		RW				
MR1		Count Polarity Select Bit <sup>(3)</sup>		0 : Counts falling edge of external signal 1 : Counts rising edge of external signal		RW				
MR2		Up/Down Switching Factor Select Bit		0 : UDF register 1 : Input signal to TAiOUT pin <sup>(4)</sup>		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				

## NOTES :

1. During event counter mode, the count source can be selected using the ONSF and TRGSR registers.
2. TA0OUT pin is N-channel open drain output.
3. Effective when the TAiTGh and TAiTGL bits in the ONSF or TRGSR register are "00b" (TAiIN pin input).
4. Count down when input on TAiOUT pin is low or count up when input on that pin is high. The port direction bit for TAiOUT pin is set to "0" (= input mode).

**Figure 15.9** TAiMR Register in Event Counter Mode (when not using two-phase pulse signal processing)

**Table 15.3 Specifications in Event Counter Mode (when processing two-phase pulse signal with Timer A2, A3 and A4)**

Item	Specification
Count Source	• Two-phase pulse signals input to TAI <sub>IN</sub> or TAI <sub>OUT</sub> pins (i=2 to 4)
Count Operation	• Up-count or down-count can be selected by two-phase pulse signal • When the timer overflows or underflows, it reloads the reload register contents and continues counting. When operating in free-running mode, the timer continues counting without reloading.
Divide Ratio	1/ (FFFFh - n + 1) for up-count 1/ (n + 1) for down-count    n : set value of TAI register    0000h to FFFFh
Count Start Condition	Set TAI <sub>S</sub> bit of TABSR register to "1" (= start counting)
Count Stop Condition	Set TAI <sub>S</sub> bit to "0" (= stop counting)
Interrupt Request Generation Timing	Timer overflow or underflow
TA2IN Pin Function	Two-phase pulse input
TA2OUT 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 and until the 1st count source is input after counting start Value written to TAI register is written to both reload register and counter • When counting (after 1st count source input) Value written to TAI register is written to reload register (Transferred to counter when reloaded next)
Select Function <sup>(1)</sup>	<p>• Normal processing operation (Timer A2 and Timer A3) The timer counts up rising edges or counts down falling edges on TAJ<sub>IN</sub> pin when input signals on TAJ<sub>OUT</sub> pin is "H".</p>  <p>• Multiply-by-4 processing operation (Timer A3 and Timer A4) If the phase relationship is such that TAK<sub>IN</sub>(k=3, 4) pin goes "H" when the input signal on TAK<sub>OUT</sub> pin is "H," the timer counts up rising and falling edges on TAK<sub>OUT</sub> and TAK<sub>IN</sub> pins. If the phase relationship is such that TAK<sub>IN</sub> pin goes "L" when the input signal on TAK<sub>OUT</sub> pin is "H," the timer counts down rising and falling edges on TAK<sub>OUT</sub> and TAK<sub>IN</sub> pins.</p>  <p>• Counter initialization by Z-phase input (Timer A3) The timer count value is initialized to 0 by Z-phase input.</p>

**NOTES:**

- Only Timer A3 is selectable. Timer A2 is fixed to normal processing operation, and Timer A4 is fixed to multiply-by-4 processing operation.

## Timer A2 Mode Register (i=2 to 4)

(when using two-phase pulse signal processing)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	After Reset	
		0	1	0	0	0	1	TA2MR to TA4MR	0398h to 039Ah	00h	
								Bit Symbol	Bit Name	Function	RW
								TMOD0	Operation Mode Select Bit	b1 b0 0 1 : Event counter mode	RW
								TMOD1			RW
								MR0	To use tw o-phase pulse signal processing, set this bit to "0".		RW
								MR1	To use tw o-phase pulse signal processing, set this bit to "0".		RW
								MR2	To use tw o-phase pulse signal processing, set this bit to "1".		RW
								MR3	To use tw o-phase pulse signal processing, set this bit to "0".		RW
								TCK0	Count Operation Type Select Bit	0 : Reload type 1 : Free-run type	RW
								TCK1	Tw o-phase pulse signal processing Operation Type Select Bit <sup>(1, 2)</sup>	0 : Normal processing operation 1 : Multiply-by-4 processing operation	RW

## NOTES :

1. TCK1 bit is valid for Timer A3 mode register. No matter how this bit is set, Timer A2 and A4 always operate in normal processing mode and x4 processing mode, respectively.
2. If tw o-phase pulse signal processing is desired, following register settings are required:
  - Set the TAIp bit in the UDF register to "1" (tw o-phase pulse signal processing function enabled).
  - Set the TAITGH and TAITGL bits in the TRGSR register to "00b" (TAiIN pin input).
  - Set the port direction bits for TAIiN and TAIiOUT to "0" (input mode).

**Figure 15.10 TA2MR to TA4MR Registers in Event Counter Mode (when using two-phase pulse signal processing with Timer A2, A3 and A4)**

### 15.1.2.1 Counter Initialization by Two-Phase Pulse Signal Processing

This function initializes the timer count value to “0” 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, x4 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 detecting Z-phase input edge. The active edge can be chosen to be the rising or falling edge by using the POL bit in the INT2IC register. The Z-phase pulse width applied to the INT2 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 recognizing Z-phase input. Figure 15.11 shows the Relationship Between the Two-Phase Pulse (A phase and B phase) and the Z-Phase.

If Timer A3 overflow or underflow coincides with the counter initialization by Z-phase input, a Timer A3 interrupt request is generated twice in succession. Do not use Timer A3 interrupt when using this function.

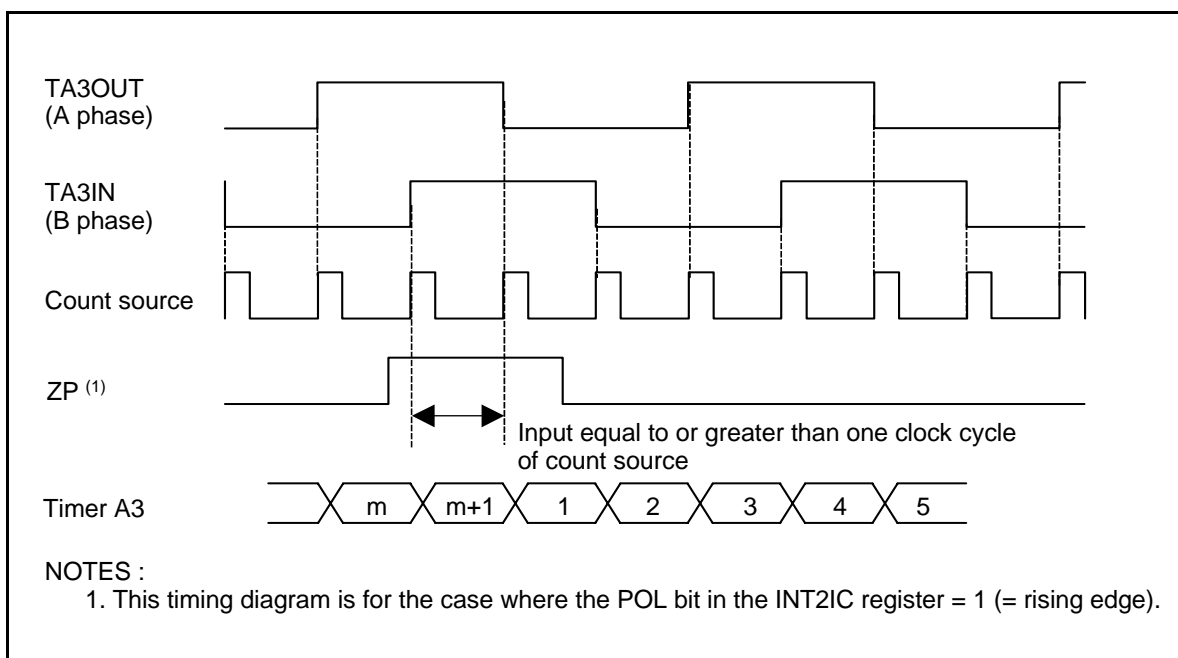


Figure 15.11 Two-Phase Pulse (A phase and B phase) and the Z-Phase

### 15.1.3 One-shot Timer Mode

In one-shot timer mode, the timer is activated only once by one trigger (see Table 15.4). When the trigger occurs, the timer starts up and continues operating for a given period. Figure 15.12 shows the TAI<sub>MR</sub> Register in One-Shot Timer Mode.

**Table 15.4 Specifications in One-shot Timer Mode**

Item	Specification
Count Source	f1, f2, f8, f32, fC32
Count Operation	<ul style="list-style-type: none"> <li>• Down-count</li> <li>• When the counter reaches “0000h,” it stops counting after reloading a new value</li> <li>• If a trigger occurs when counting, the timer reloads a new count and restarts counting</li> </ul>
Divide Ratio	1/n n : set value of TAI register (i=0 to 4) 0000h to FFFFh However, the counter does not work if the divide-by-n value is set to “0000h”.
Count start Condition	TAI <sub>S</sub> bit in the TABSR register = 1 (start counting) and one of the following triggers occurs. <ul style="list-style-type: none"> <li>• External trigger input from the TAI<sub>IN</sub> pin</li> <li>• Timer B2 overflow or underflow, Timer A<sub>j</sub> (j=i-1, except j=4 if i=0) overflow or underflow, Timer A<sub>k</sub> (k=i+1, except k=0 if i=4) overflow or underflow</li> <li>• The TAI<sub>OS</sub> bit in the ONSF register is set to “1”(= timer starts)</li> </ul>
Count Stop Condition	<ul style="list-style-type: none"> <li>• When the counter is reloaded after reaching “0000h”</li> <li>• TAI<sub>S</sub> bit is set to “0” (= stop counting)</li> </ul>
Interrupt Request Generation Timing	When the counter reaches “0000h”
TAI <sub>IN</sub> Pin Function	I/O port or trigger input
TAI <sub>OUT</sub> Pin Function	I/O port or pulse output
Read from Timer	An indeterminate value is read by reading TAI register
Write to Timer	<ul style="list-style-type: none"> <li>• When not counting and until the 1st count source is input after counting start Value written to TAI register is written to both reload register and counter</li> <li>• When counting (after 1st count source input) Value written to TAI register is written to only reload register (Transferred to counter when reloaded next)</li> </ul>
Select Function	<ul style="list-style-type: none"> <li>• Pulse output function The timer outputs a low when not counting and a high when counting.</li> </ul>

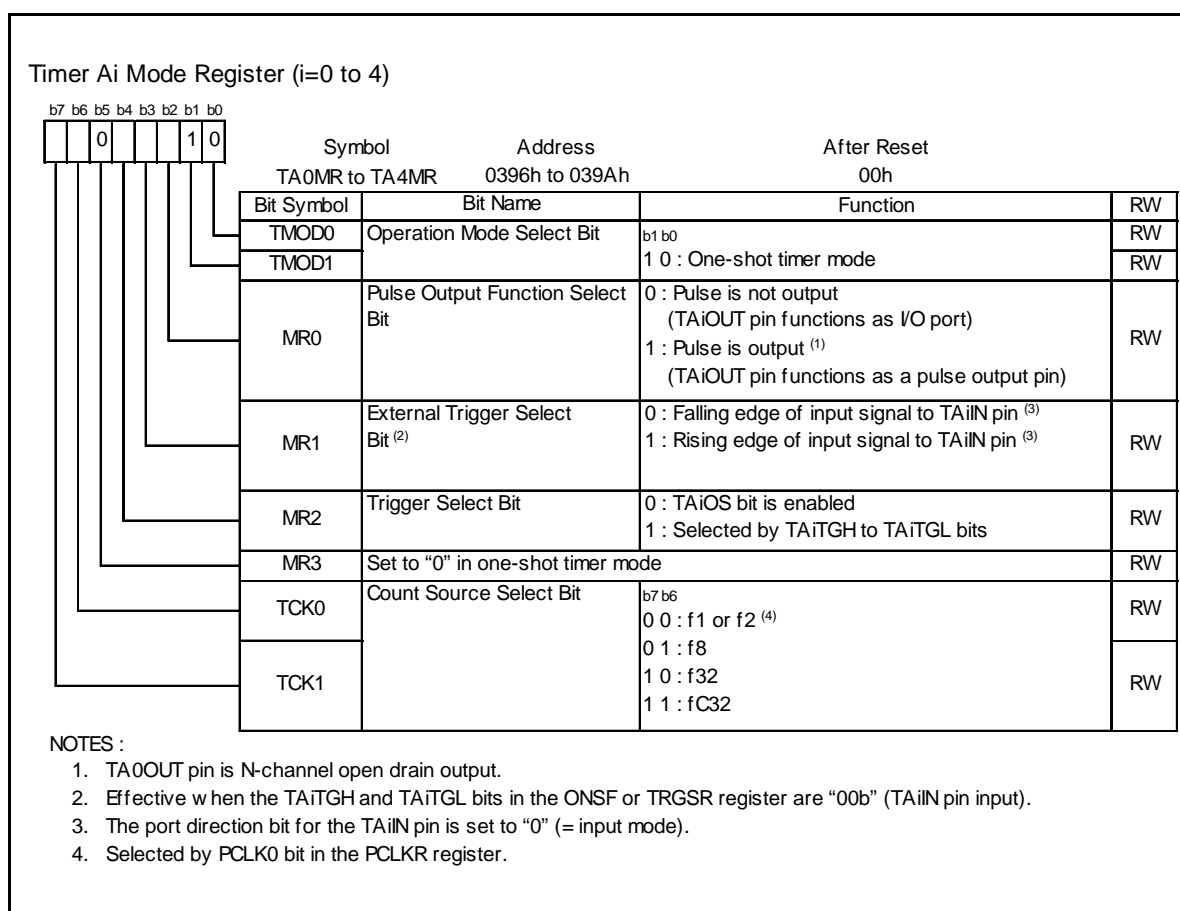


Figure 15.12 TAIiMR Register in One-Shot Timer Mode

### 15.1.4 Pulse Width Modulation (PWM) Mode

In PWM mode, the timer outputs pulses of a given width in succession (see Table 15.5). The counter functions as either 16-bit pulse width modulator or 8-bit pulse width modulator. Figure 15.13 shows TAI<sub>i</sub>MR Register in PWM Mode. Figures 15.14 and 15.15 show Example of 16-bit Pulse Width Modulator Operation and Example of 8-bit Pulse Width Modulator Operation.

**Table 15.5 Specifications in PWM Mode**

Item	Specification
Count Source	f1, f2, f8, f32, fC32
Count Operation	<ul style="list-style-type: none"> <li>• Down-count (operating as an 8-bit or a 16-bit pulse width modulator)</li> <li>• The timer reloads a new value at a rising edge of PWM pulse and continues counting</li> <li>• The timer is not affected by a trigger that occurs during counting</li> </ul>
16-bit PWM	<ul style="list-style-type: none"> <li>• High level width <math>n / f_j</math>      <math>n</math> : set value of TAI register (<math>i=0</math> to 4)</li> <li>• Cycle time <math>(2^{16}-1) / f_j</math> fixed      <math>f_j</math>: count source frequency (f1, f2, f8, f32, fC32)</li> </ul>
8-bit PWM	<ul style="list-style-type: none"> <li>• High level width <math>n \times (m+1) / f_j</math>      <math>n</math> : set value of TAI register high-order address</li> <li>• Cycle time <math>(2^8-1) \times (m+1) / f_j</math>      <math>m</math> : set value of TAI register low-order address</li> </ul>
Count Start Condition	<ul style="list-style-type: none"> <li>• TAI<sub>i</sub>S bit of TABSR register is set to "1" (= start counting)</li> <li>• The TAI<sub>i</sub>S bit = 1 and external trigger input from the TAI<sub>i</sub>N pin</li> <li>• The TAI<sub>i</sub>S bit = 1 and one of the following external triggers occurs</li> <li>• Timer B2 overflow or underflow, Timer A<sub>j</sub> (<math>j=i-1</math>, except <math>j=4</math> if <math>i=0</math>) overflow or underflow, Timer A<sub>k</sub> (<math>k=i+1</math>, except <math>k=0</math> if <math>i=4</math>) overflow or underflow</li> </ul>
Count Stop Condition	TAI <sub>i</sub> S bit is set to "0" (= stop counting)
Interrupt Request Generation Timing	On the falling edge of PWM pulse
TAI <sub>i</sub> N Pin Function	I/O port or trigger input
TAI <sub>i</sub> OUT Pin Function	Pulse output
Read from Timer	An indeterminate value is read by reading TAI register
Write to Timer	<ul style="list-style-type: none"> <li>• When not counting and until the 1st count source is input after counting start Value written to TAI register is written to both reload register and counter</li> <li>• When counting (after 1st count source input) Value written to TAI register is written to only reload register (Transferred to counter when reloaded next)</li> </ul>



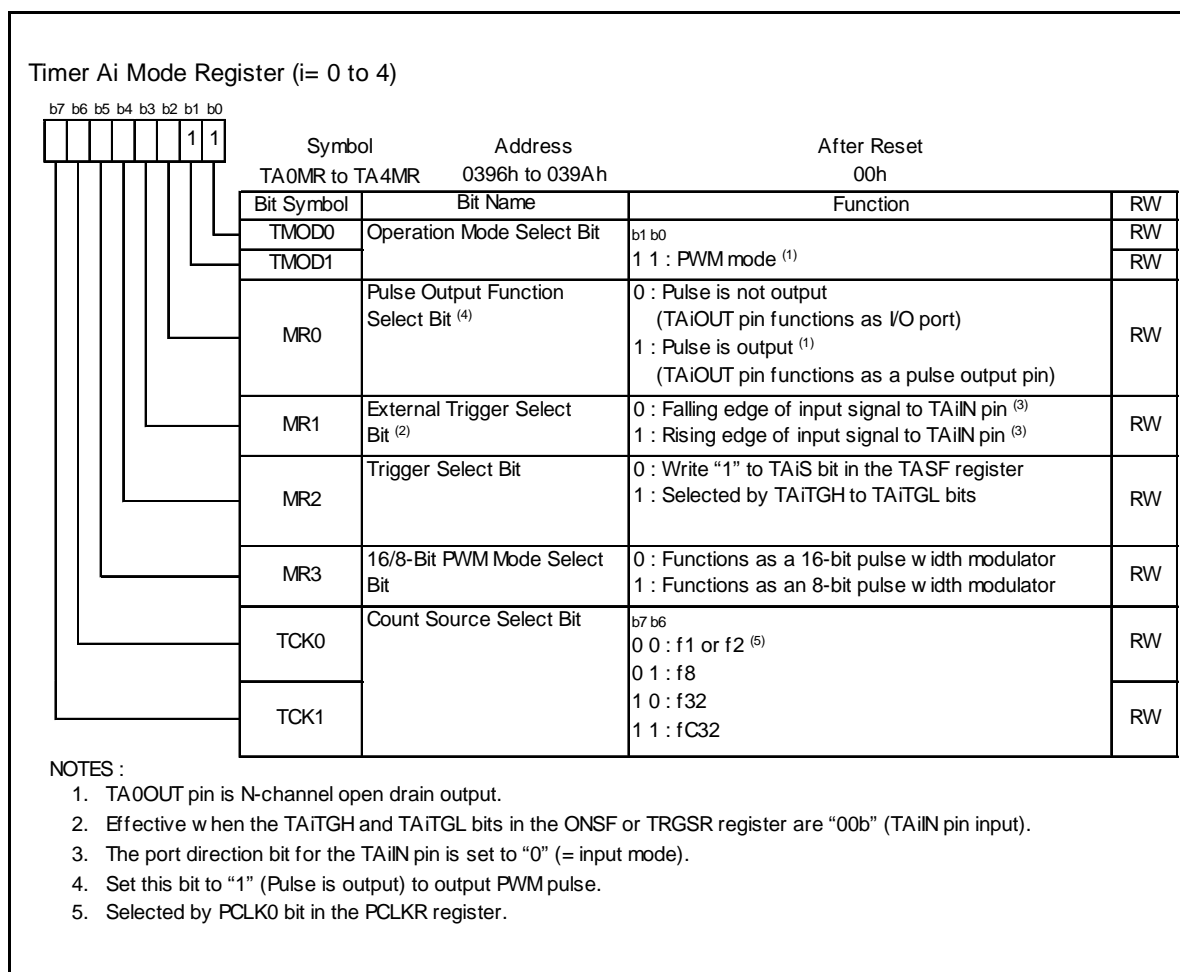


Figure 15.13 TAIiMR Register in PWM Mode

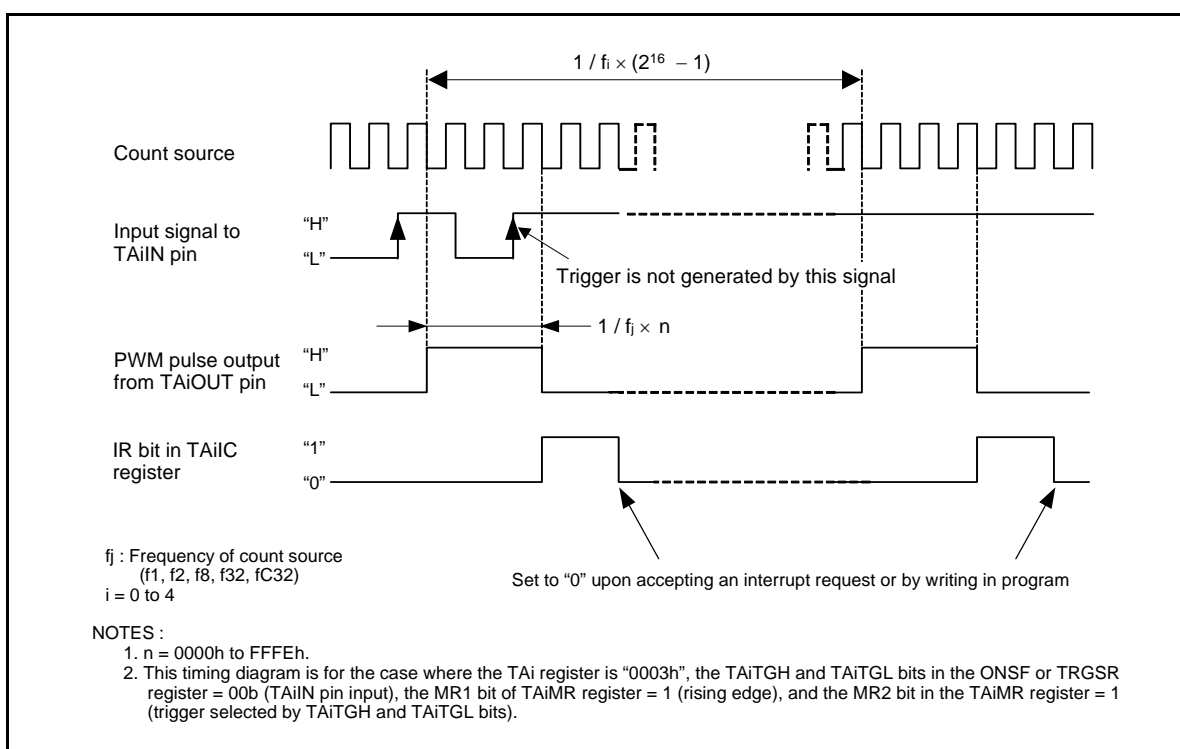


Figure 15.14 Example of 16-bit Pulse Width Modulator Operation

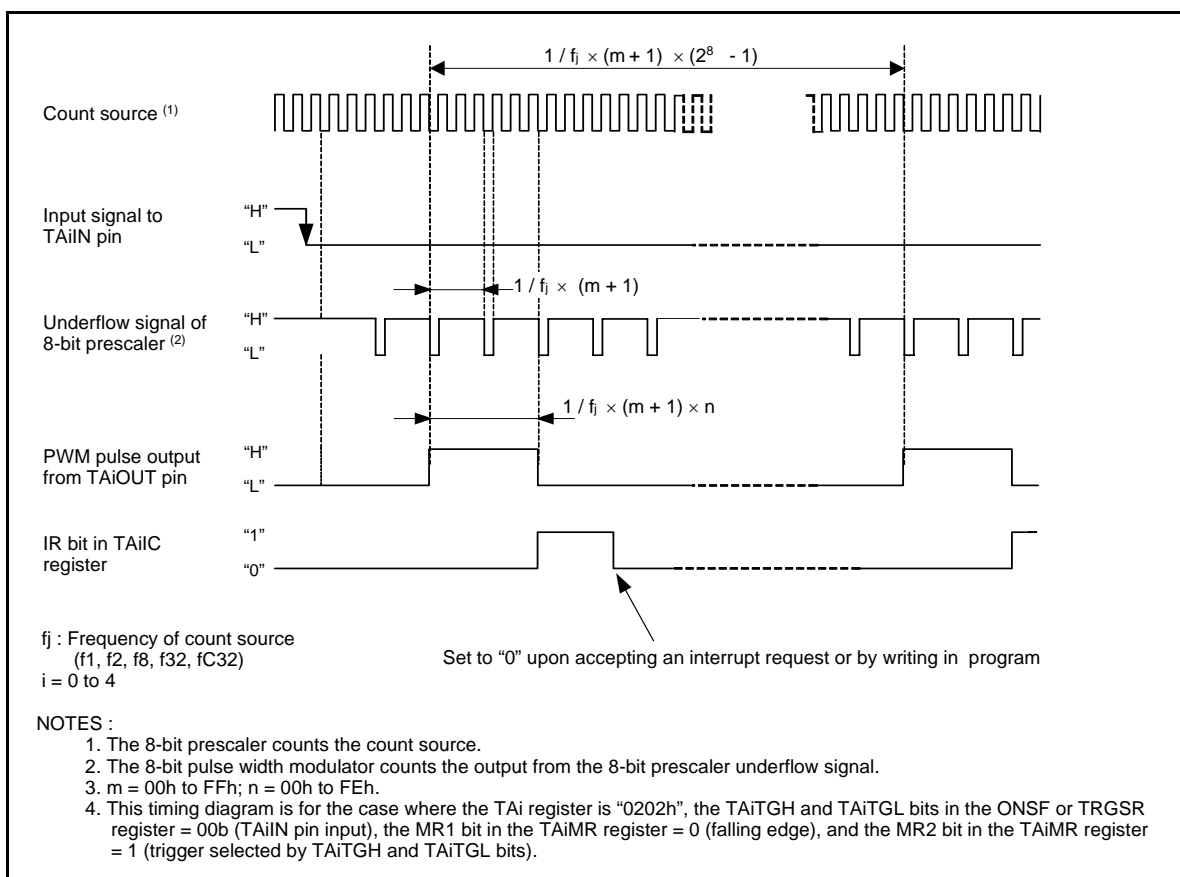


Figure 15.15 Example of 8-bit Pulse Width Modulator Operation



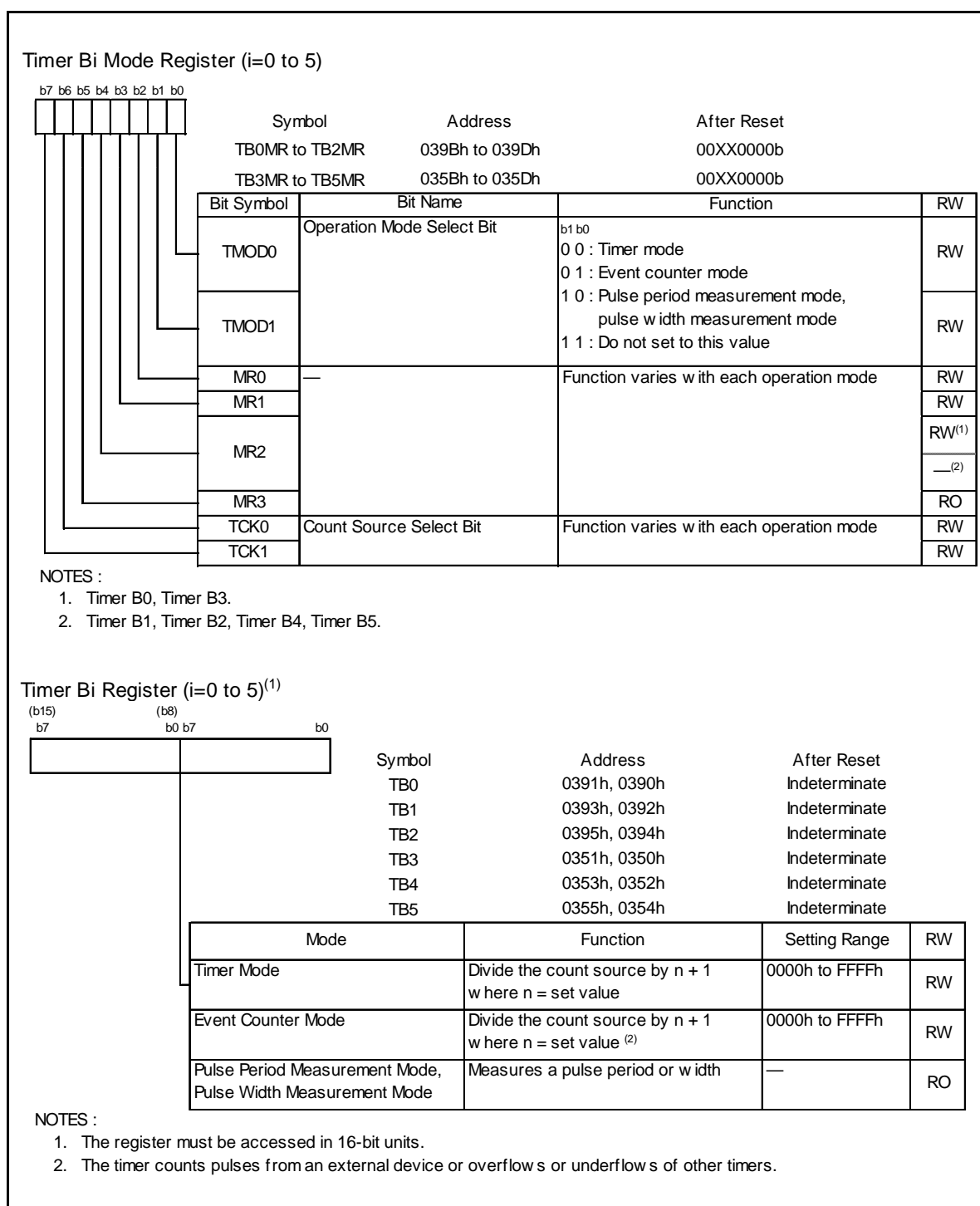


Figure 15.17 TBiMR and TBi Registers

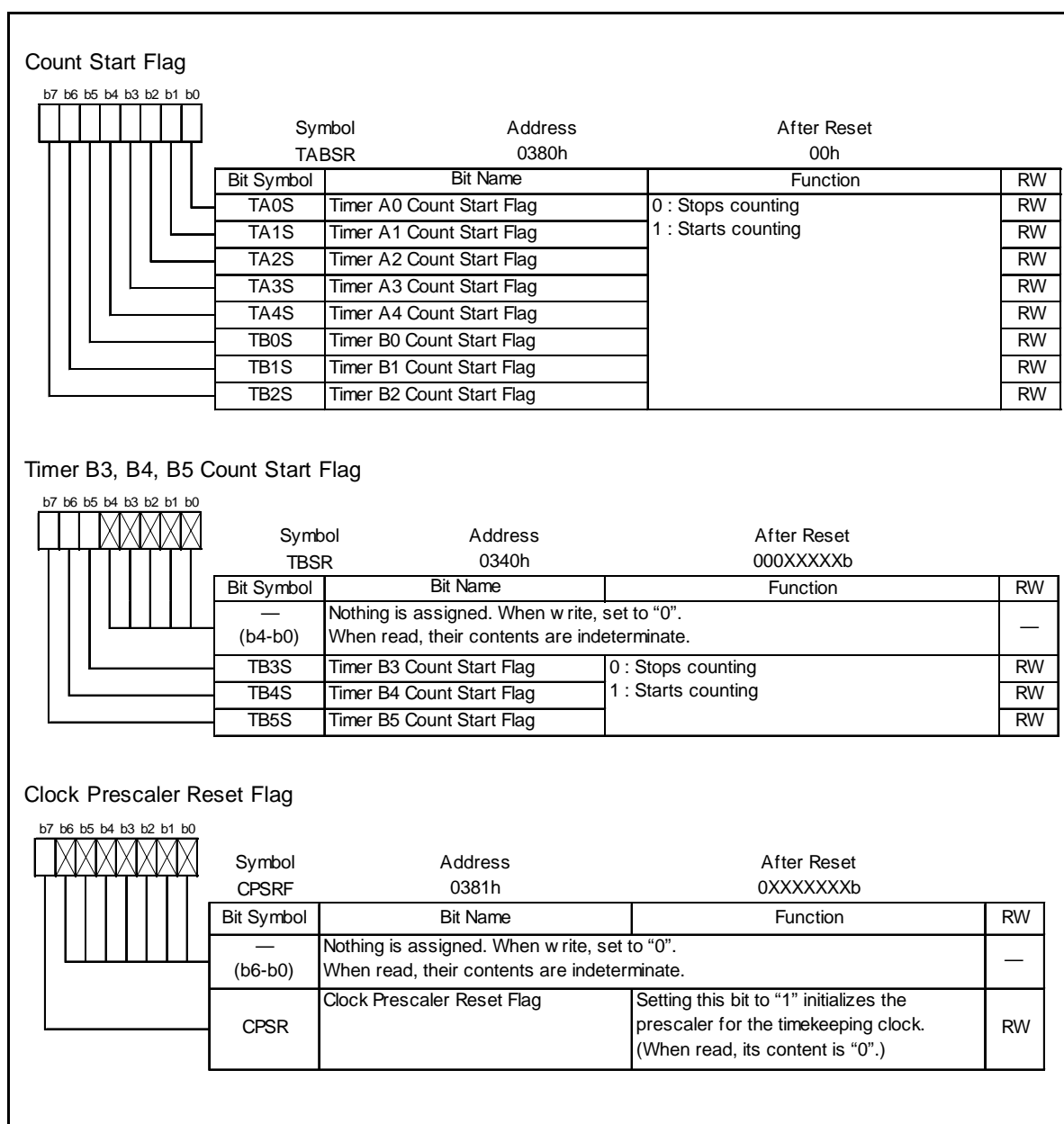


Figure 15.18 TABSR, TBSR and CPSRF Registers

### 15.2.1 Timer Mode

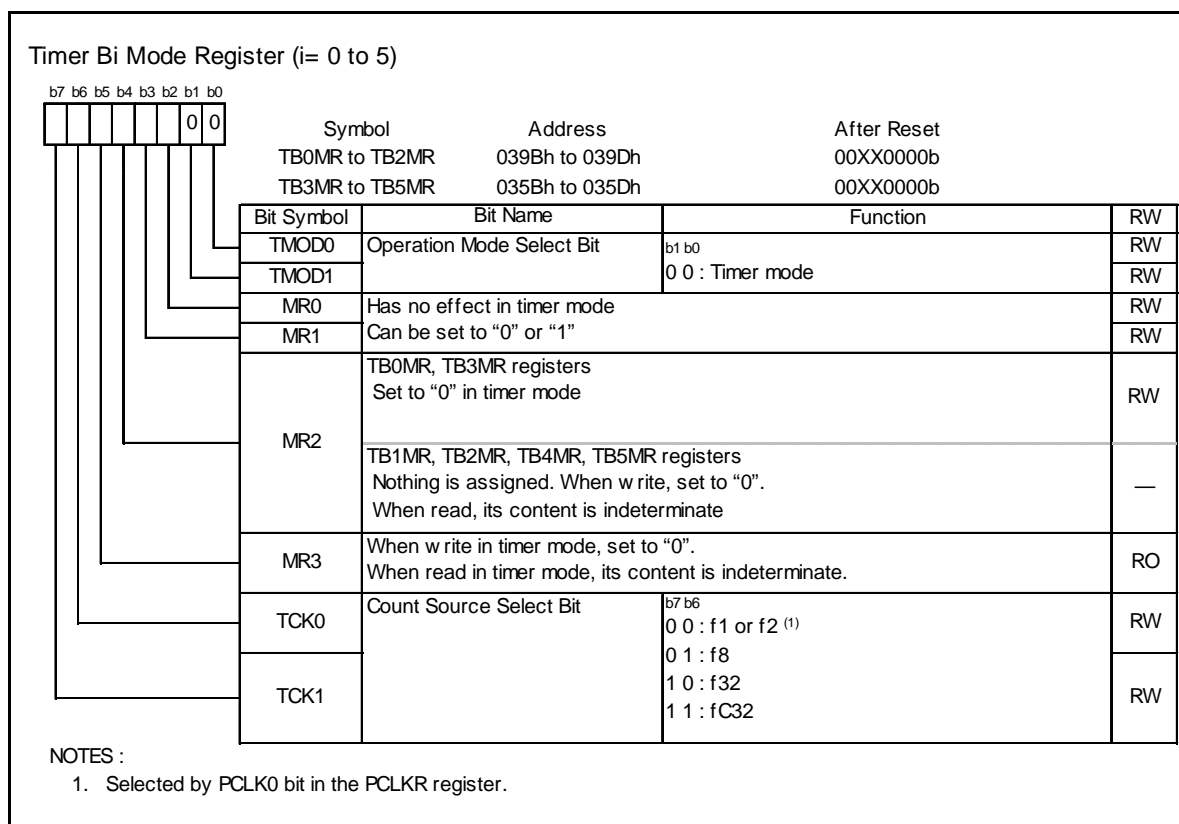
In timer mode, the timer counts a count source generated internally (see Table 15.6). Figure 15.19 shows TBiMR Register in Timer Mode.

**Table 15.6 Specifications in Timer Mode**

Item	Specification
Count Source	f1, f2, f8, f32, fC32
Count Operation	<ul style="list-style-type: none"> <li>Down-count</li> <li>When the timer underflows, it reloads the reload register contents and continues counting</li> </ul>
Divide Ratio	1/(n+1) n: set value of TBi register (i= 0 to 5) 0000h to FFFFh
Count Start Condition	Set TBiS bit <sup>(1)</sup> to "1" (= start counting)
Count Stop Condition	Set 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 TBi register
Write to Timer	<ul style="list-style-type: none"> <li>When not counting and until the 1st count source is input after counting start Value written to TBi register is written to both reload register and counter</li> <li>When counting (after 1st count source input) Value written to TBi register is written to only reload register (Transferred to counter when reloaded next)</li> </ul>

**NOTES:**

- The TB0S to TB2S bits are assigned to the bit 5 to bit 7 in the TABSR register, and the TB3S to TB5S bits are assigned to the bit 5 to bit 7 in the TBSR register.



**Figure 15.19 TBiMR Register in Timer Mode**

### 15.2.2 Event Counter Mode

In event counter mode, the timer counts pulses from an external device or overflows and underflows of other timers (see Table 15.7). Figure 15.20 shows TBiMR Register in Event Counter Mode.

**Table 15.7 Specifications in Event Counter Mode**

Item	Specification
Count Source	<ul style="list-style-type: none"> <li>External signals input to TBiIN pin (i=0 to 5) (effective edge can be selected in program)</li> <li>Timer Bj overflow or underflow (j=i-1, except j=2 if i=0, j=5 if i=3)</li> </ul>
Count Operation	<ul style="list-style-type: none"> <li>Down-count</li> <li>When the timer underflows, it reloads the reload register contents and continues counting</li> </ul>
Divide Ratio	1/(n+1) n: set value of TBi register 0000h to FFFFh
Count Start Condition	Set TBiS bit <sup>(1)</sup> to "1" (= start counting)
Count Stop Condition	Set 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 TBi register
Write to Timer	<ul style="list-style-type: none"> <li>When not counting and until the 1st count source is input after counting start Value written to TBi register is written to both reload register and counter</li> <li>When counting (after 1st count source input) Value written to TBi register is written to only reload register (Transferred to counter when reloaded next)</li> </ul>

**NOTES:**

- The TB0S to TB2S bits are assigned to the bit 5 to bit 7 in the TABSR register, and the TB3S to TB5S bits are assigned to the bit 5 to bit 7 in the TBSR register.

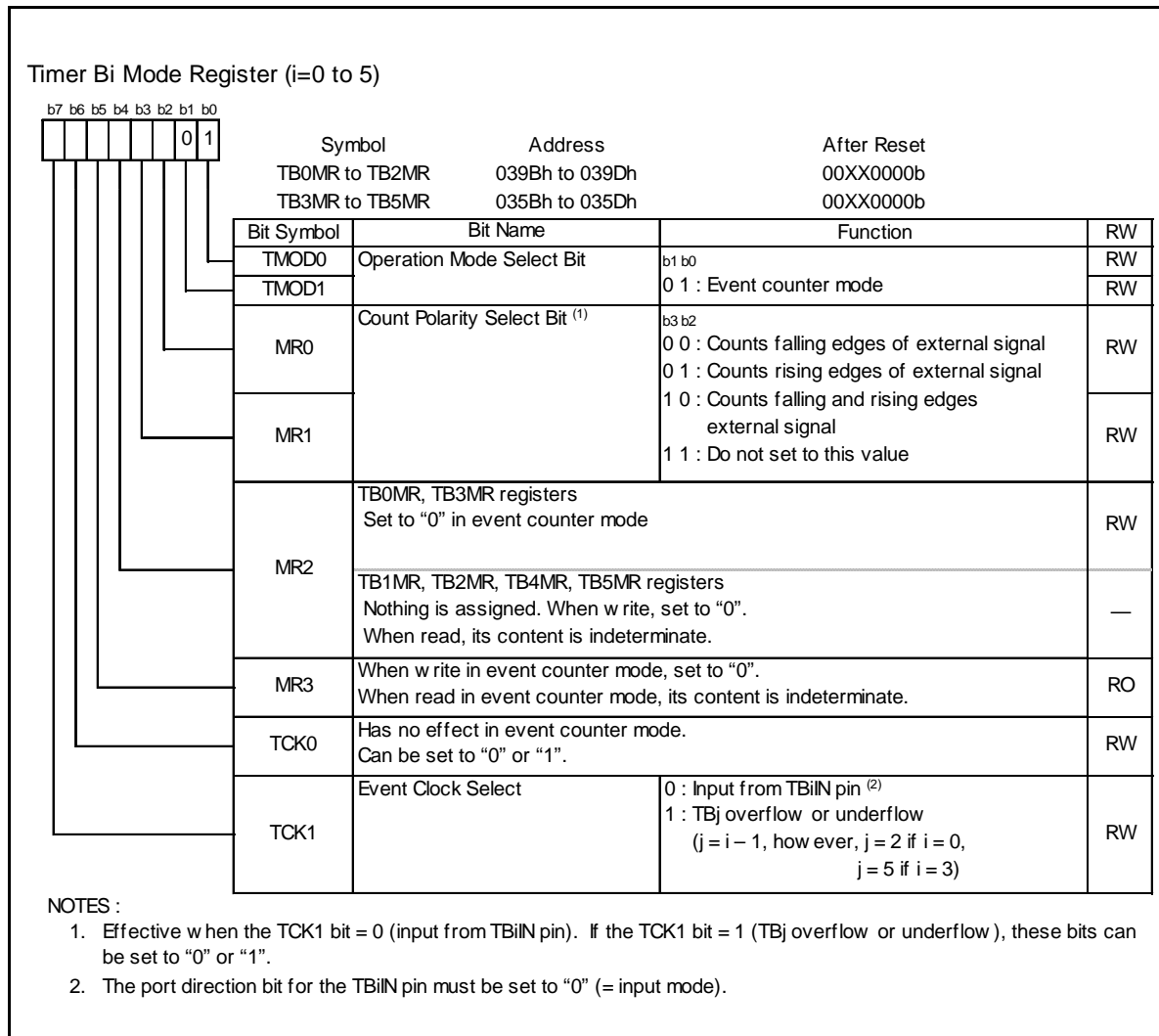


Figure 15.20 TBiMR Register in Event Counter Mode



### 15.2.3 Pulse Period and Pulse Width Measurement Mode

In pulse period and pulse width measurement mode, the timer measures pulse period or pulse width of an external signal (see Table 15.8). Figure 15.21 shows TBiMR Register in Pulse Period and Pulse Width Measurement Mode. Figure 15.22 shows the Operation Timing when Measuring a Pulse Period. Figure 15.23 shows the Operation Timing when Measuring a Pulse Width.

**Table 15.8 Specifications in Pulse Period and Pulse Width Measurement Mode**

Item	Specification
Count Source	f1, f2, f8, f32, fC32
Count Operation	<ul style="list-style-type: none"> <li>Up-count</li> <li>Counter value is transferred to reload register at an effective edge of measurement pulse. The counter value is set to "0000h" to continue counting.</li> </ul>
Count Start Condition	Set TBiS (i=0 to 5) bit <sup>(3)</sup> to "1" (= start counting)
Count Stop Condition	Set TBiS bit to "0" (= stop counting)
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>When an effective edge of measurement pulse is input <sup>(1)</sup></li> <li>Timer overflow. When an overflow occurs, MR3 bit in the TBiMR register is set to "1" (overflowed) simultaneously. MR3 bit is set to "0" (no overflow) by writing to TBiMR register at the next count timing or later after MR3 bit was set to "1". At this time, make sure TBiS bit is set to "1" (start counting).</li> </ul>
TBiIN Pin Function	Measurement pulse input
Read from Timer	Contents of the reload register (measurement result) can be read by reading TBi register <sup>(2)</sup>
Write to Timer	Value written to TBi register is written to neither reload register nor counter

**NOTES:**

1. Interrupt request is not generated when the first effective edge is input after the timer started counting.
2. Value read from TBi register is indeterminate until the second valid edge is input after the timer starts counting.
3. The TB0S to TB2S bits are assigned to the bit 5 to bit 7 in the TABSR register, and the TB3S to TB5S bits are assigned to the bit 5 to bit 7 in the TBSR register.

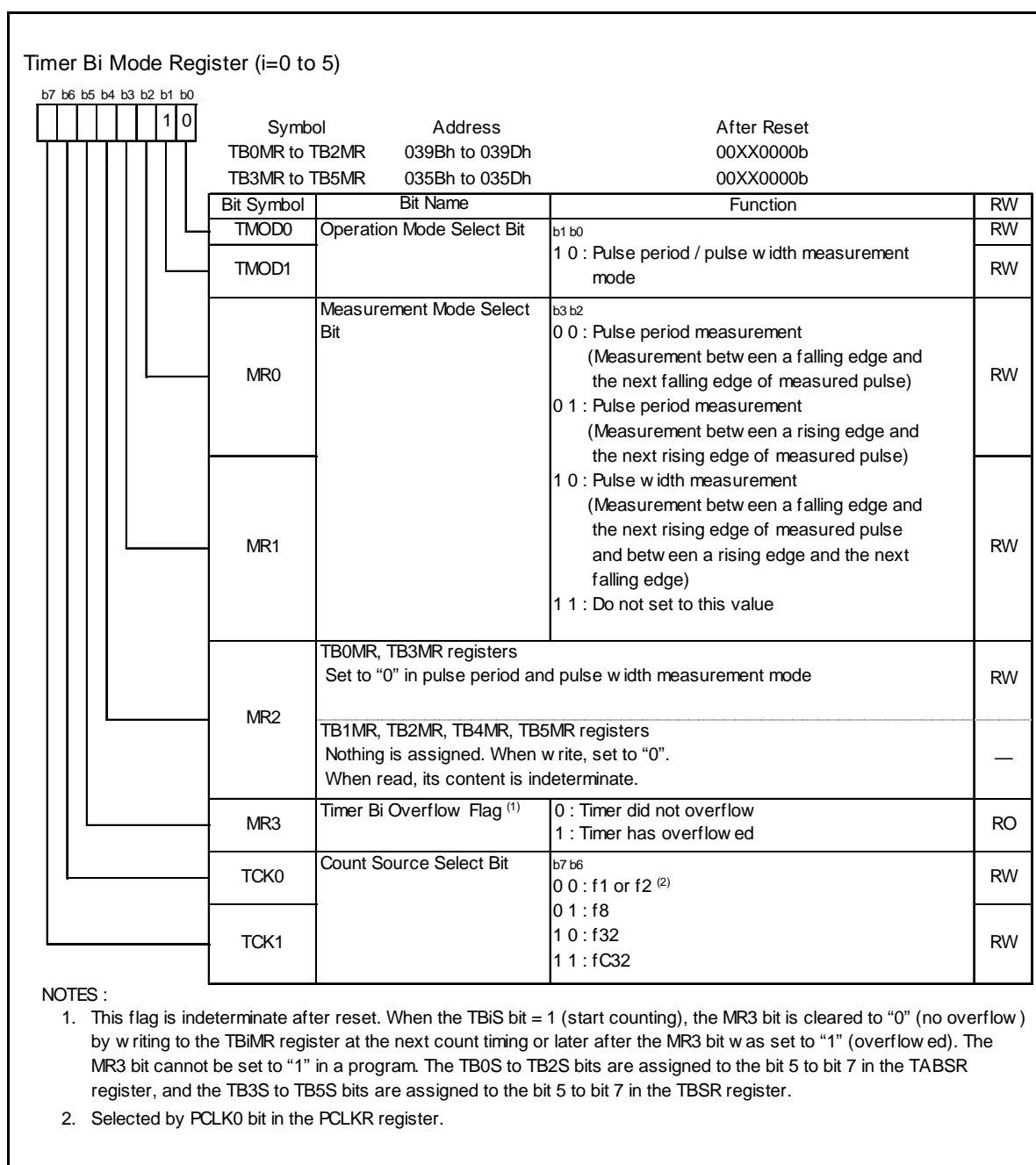


Figure 15.21 TBiMR Register in Pulse Period and Pulse Width Measurement Mode

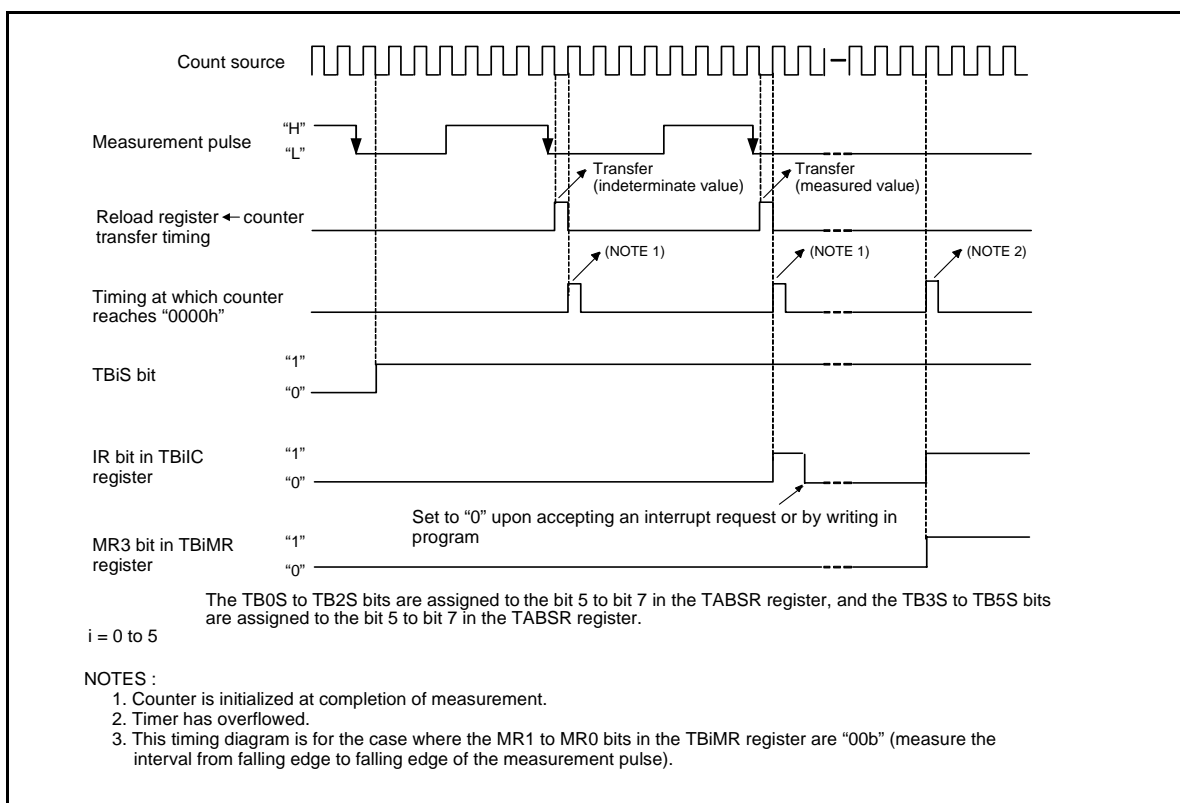


Figure 15.22 Operation Timing when Measuring a Pulse Period

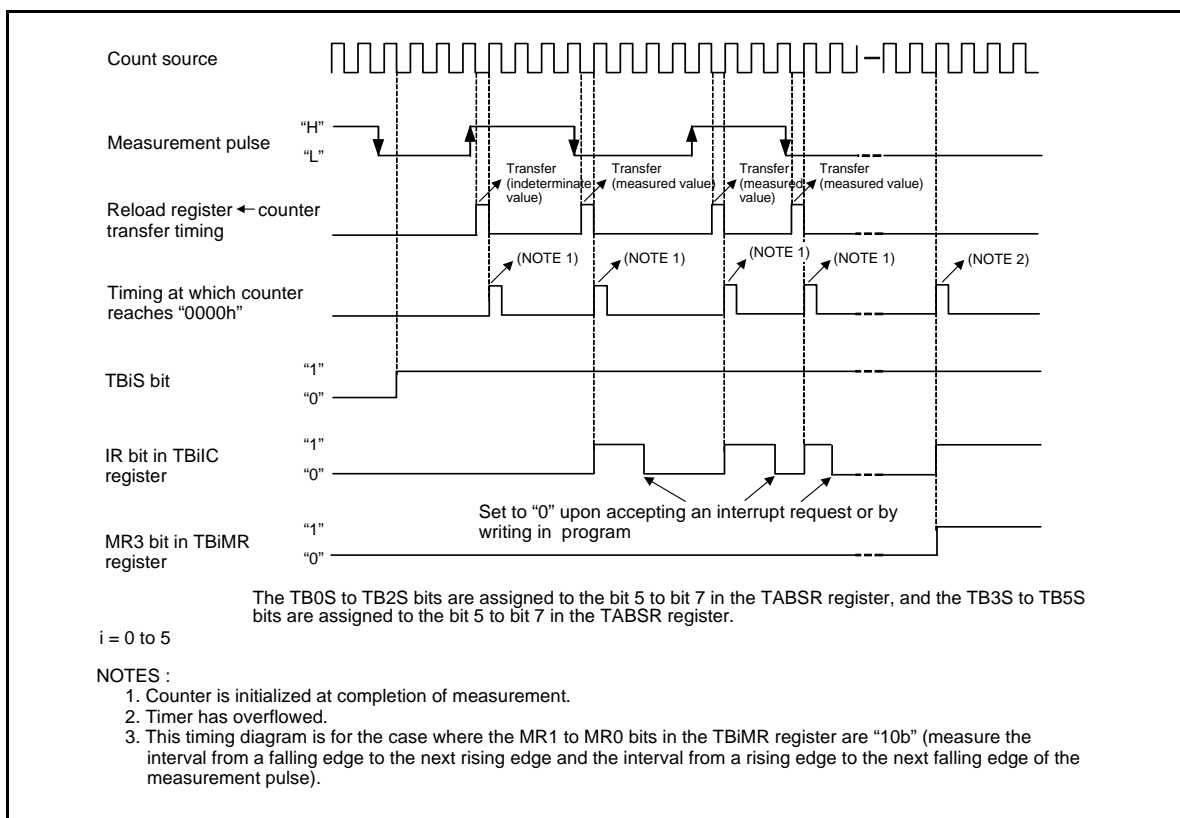


Figure 15.23 Operation Timing when Measuring a Pulse Width

## 16. Three-Phase Motor Control Timer Function

### Note

The M16C/62P (80-pin version) and M16C/62PT (80-pin version) do not use this function.

Timers A1, A2, A4 and B2 can be used to output three-phase motor drive waveforms. Table 16.1 lists the Three-phase Motor Control Timer Functions Specifications. Figure 16.1 shows the Three-phase Motor Control Timer Functions Block Diagram. Also, the related registers are shown on Figure 16.2 to Figure 16.8.

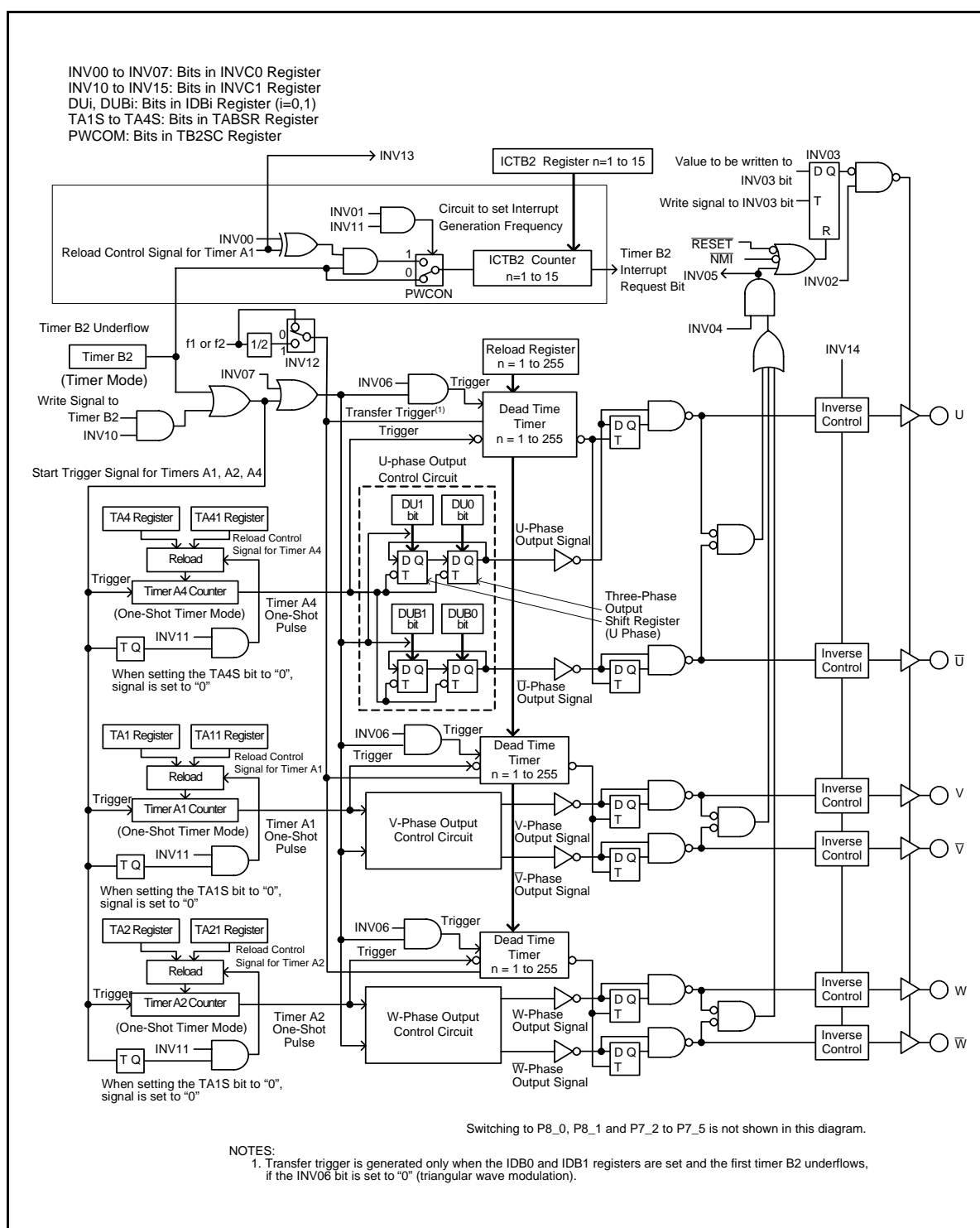
**Table 16.1 Three-phase Motor Control Timer Functions Specifications**

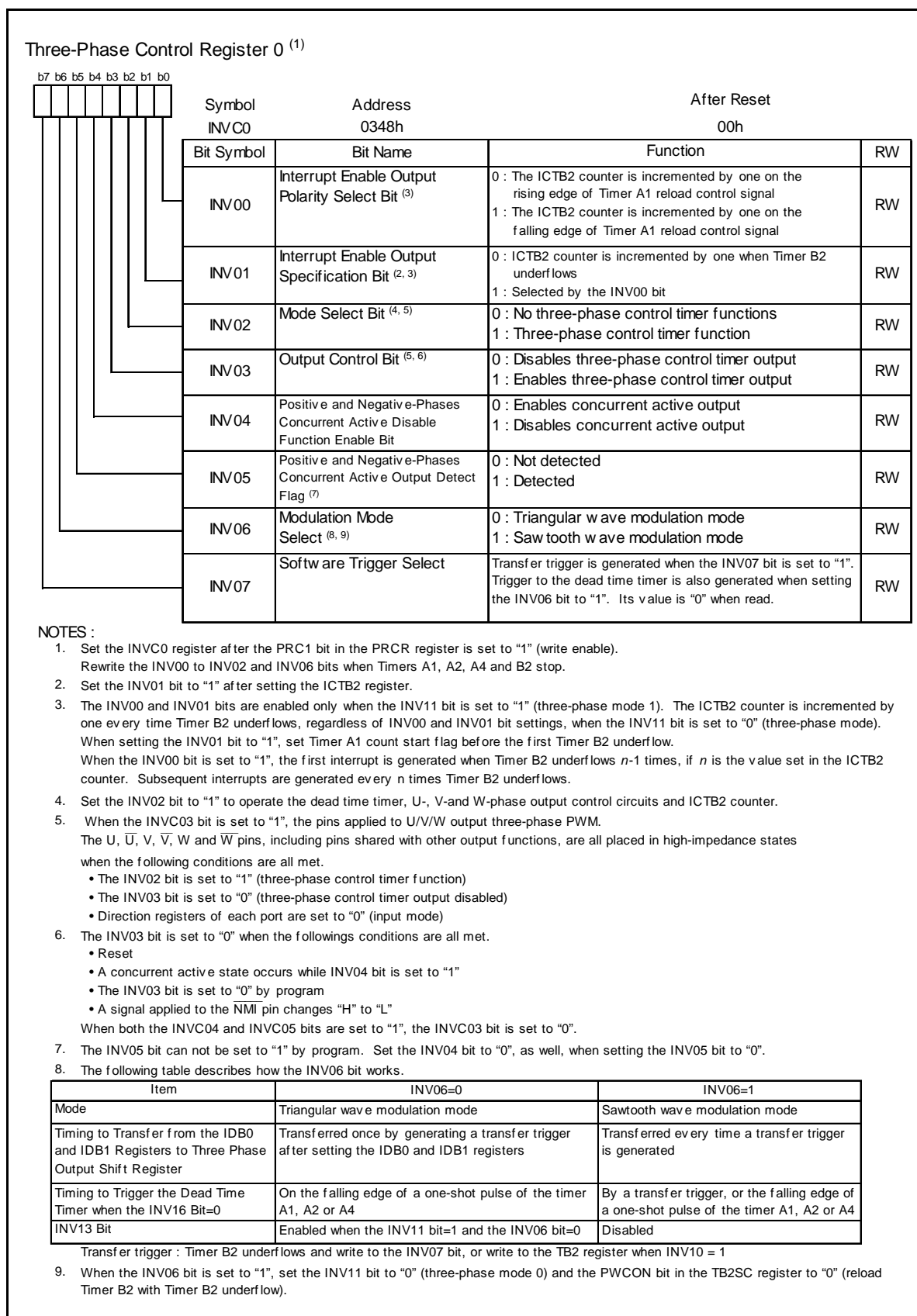
Item	Specification
Three-Phase Waveform Output Pin	Six pins (U, $\bar{U}$ , V, $\bar{V}$ , W, $\bar{W}$ )
Forced Cutoff Input <sup>(1)</sup>	Input "L" to $\bar{NMI}$ pin
Used Timers	Timer A4, A1, A2 (used in the one-shot timer mode) Timer A4: U- and $\bar{U}$ -phase waveform control Timer A1: V- and $\bar{V}$ -phase waveform control Timer A2: W- and $\bar{W}$ -phase waveform control Timer B2 (used in the timer mode) Carrier wave cycle control Dead time timer (3 eight-bit timer and shared reload register) Dead time control
Output Waveform	Triangular wave modulation, Sawtooth wave modification Enable to output "H" or "L" for one cycle Enable to set positive-phase level and negative-phase level respectively
Carrier Wave Cycle	Triangular wave modulation: count source x (m+1) x 2 Sawtooth wave modulation: count source x (m+1) m: Setting value of TB2 register, 0000h to FFFFh Count source: f1, f2, f8, f32, fC32
Three-Phase PWM Output Width	Triangular wave modulation: count source x n x 2 Sawtooth wave modulation: count source x n n: Setting value of TA4, TA1 and TA2 register (of TA4, TA41, TA1, TA11, TA2 and TA21 registers when setting the INV11 bit to "1"), 0001h to FFFFh Count source: f1, f2, f8, f32, fC32
Dead Time	Count source x p, or no dead time p: Setting value of DTT register, 01h to FFh Count source: f1, f2, f1 divided by 2, f2 divided by 2
Active Level	Enable to select "H" or "L"
Positive and Negative-Phase Concurrent Active Disable Function	Positive and negative-phases concurrent active disable function Positive and negative-phases concurrent active detect function
Interrupt Frequency	For Timer B2 interrupt, select a carrier wave cycle-to-cycle basis through 15 times carrier wave cycle-to-cycle basis

### NOTES:

1. Forced cutoff with  $\bar{NMI}$  input is effective when the IVPCR1 bit in the TB2SC register is set to "1" (three-phase output forcible cutoff by  $\bar{NMI}$  input enabled). If an "L" signal is applied to the  $\bar{NMI}$  pin when the IVPCR1 bit is "1," the related pins go to a high-impedance state regardless of which functions of those pins are being used.

Related pins    P7\_2/CLK2/TA1OUT/V, P7\_3/ $\overline{CTS2}$ / $\overline{RTS2}$ /TA1IN/ $\bar{V}$ , P7\_4/TA2OUT/W,  
P7\_5/TA2IN/ $\bar{W}$ , P8\_0/TA4OUT/U, P8\_1/TA4IN/ $\bar{U}$



**Figure 16.2 INV0 Register**

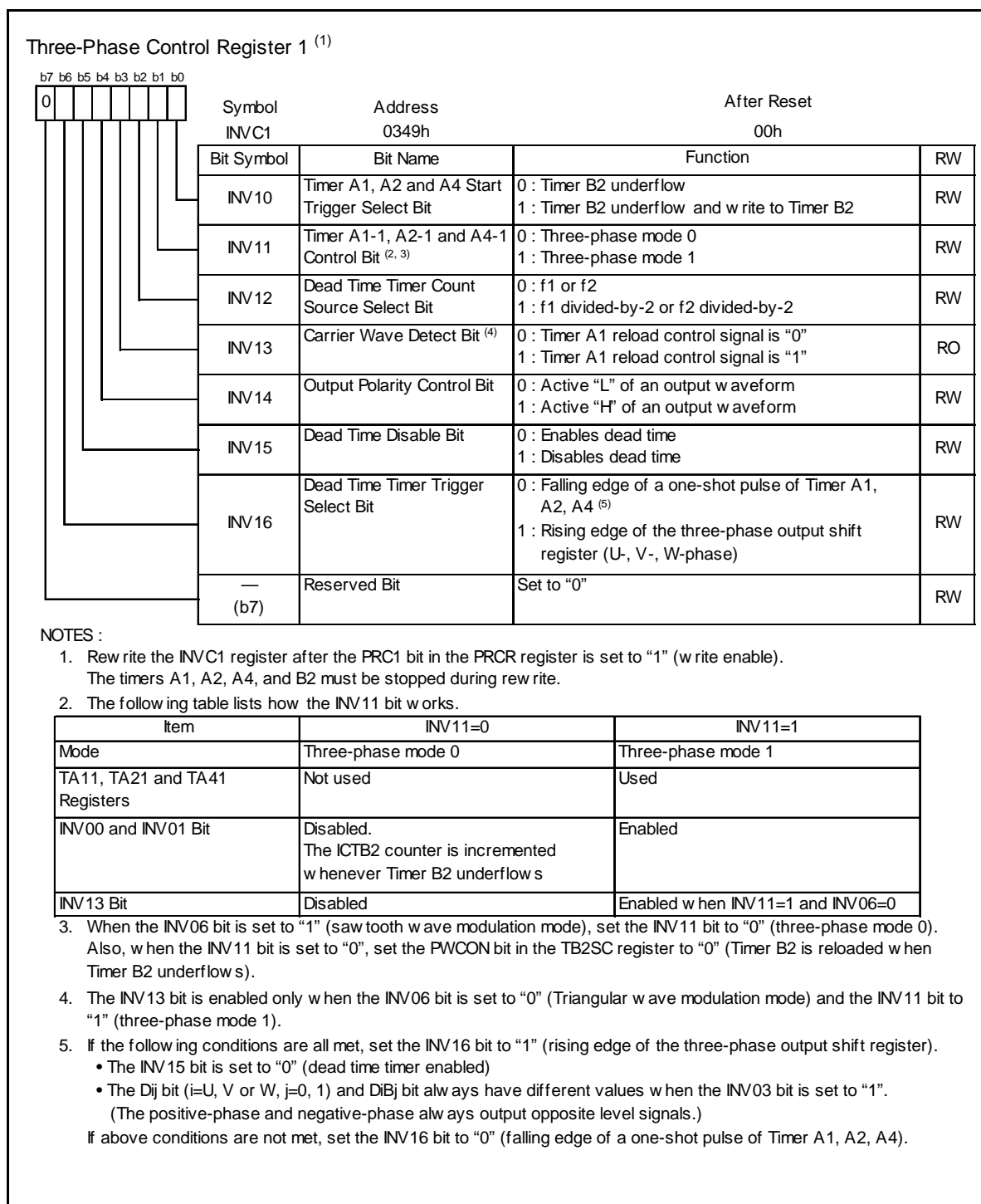


Figure 16.3 INVC1 Register

Timer B2 Interrupt Generation Frequency Set Counter<sup>(1, 2, 3)</sup>

<div><div><div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div></div><div><div></div><div></div><div></div><div></div></div></div><div><div>b7</div><div>b0</div></div></div>	<div>Symbol</div> <div>ICTB2</div> <div>Address</div> <div>034Dh</div> <div>After Reset</div> <div>Indeterminate</div>										
<div><div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div><div><div></div><div></div><div></div><div></div></div></div><div><div></div><div></div><div></div><div></div></div></div>	<table><tr><th>Function</th><th>Setting Range</th><th>RW</th></tr><tr><td>When the INV01 bit is set to “0” (the ICTB2 counter increments whenever Timer B2 underflow s) and the setting value is <math>n</math>, Timer B2 interrupt is generated every <math>n</math>th time Timer B2 underflow occurs.</td><td rowspan="2">1 to 15</td><td rowspan="2">WO</td></tr><tr><td>When the INV01 bit is set to “1” (the INV00 bit selects count timing of the ICTB2 counter) and setting value is <math>n</math>, Timer B2 interrupt is generated every <math>n</math>th time Timer B2 underflow meeting the condition selected in the INV00 bit occurs.</td></tr><tr><td>Nothing is assigned. When write, set to “0”.</td><td></td><td>—</td></tr></table>	Function	Setting Range	RW	When the INV01 bit is set to “0” (the ICTB2 counter increments whenever Timer B2 underflow s) and the setting value is $n$ , Timer B2 interrupt is generated every $n$ th time Timer B2 underflow occurs.	1 to 15	WO	When the INV01 bit is set to “1” (the INV00 bit selects count timing of the ICTB2 counter) and setting value is $n$ , Timer B2 interrupt is generated every $n$ th time Timer B2 underflow meeting the condition selected in the INV00 bit occurs.	Nothing is assigned. When write, set to “0”.		—
Function	Setting Range	RW									
When the INV01 bit is set to “0” (the ICTB2 counter increments whenever Timer B2 underflow s) and the setting value is $n$ , Timer B2 interrupt is generated every $n$ th time Timer B2 underflow occurs.	1 to 15	WO									
When the INV01 bit is set to “1” (the INV00 bit selects count timing of the ICTB2 counter) and setting value is $n$ , Timer B2 interrupt is generated every $n$ th time Timer B2 underflow meeting the condition selected in the INV00 bit occurs.											
Nothing is assigned. When write, set to “0”.		—									

## NOTES :

1. Use the MOV instruction to set the ICTB2 register.
2. If the INV01 bit is set to "1", set the ICTB2 register when the TB2S bit is set to "0" (Timer B2 counter stopped).  
If the INV01 bit is set to "0" and the TB2S bit to "1" (Timer B2 counter start), do not set the ICTB2 register when Timer B2 underflows.
3. If the INV00 bit is set to "1", the first interrupt is generated when Timer B2 underflows  $n-1$  times,  $n$  being the value set in the ICTB2 counter. Subsequent interrupts are generated every  $n$  times Timer B2 underflows.

Timer Ai, Ai-1 Register ( $i = 1, 2, 4$ )<sup>(1, 2, 3, 4, 5, 6, 7)</sup>

(b15) b7	(b8) b0 b7	b0			
			Symbol	Address	After Reset
			TA1, TA2	0389h to 0388h, 038Bh to 038Ah	Indeterminate
			TA4	038Fh to 038Eh	Indeterminate
			TA11, TA21	0343h to 0342h, 0345h to 0344h	Indeterminate
			TA41	0347h to 0346h	Indeterminate
			Function		Setting Range
			If setting value is $n$ , the timer stops when the $n$ th count source is counted after a start trigger is generated.		0000h to FFFFh
			Positive phase changes to negative phase, and vice versa, when Timers A1, A2 and A4 stop.		
					WO

## NOTES :

1. Use a 16-bit data for read and write.
2. If the TAI or TAI1 register is set to "0000h", no counters start and no Timer Ai interrupt is generated.
3. Use the MOV instruction to set the TAI and TAI1 registers.
4. When the INV15 bit in the INVC1 register is set to "0" (dead timer enabled), phase switches from an inactive level to an active level when the dead time timer stops.
5. When the INV11 bit is set to "0" (three-phase mode 0), the value of the TAI register is transferred to the reload register by a Timer Ai start trigger.  
When the INV11 bit is set to "1" (three-phase mode 1), the value of the TAI1 register is first transferred to the reload register by a Timer Ai start trigger. Then, the value of the TAI register is transferred by the next trigger. The values of the TAI1 and TAI registers are transferred alternately to the reload register with every Timer Ai start trigger.
6. Do not write to these registers when the Timer B2 underflows.
7. Follow the procedure below to set the TAI1 register.
  - (a) Write value to the TAI1 register,
  - (b) Wait one Timer Ai count source cycle, and
  - (c) Write the same value as (a) to the TAI1 register.

Figure 16.4 ICTB2, TA1, TA2, TA4, TA11, TA21 and TA41 Registers



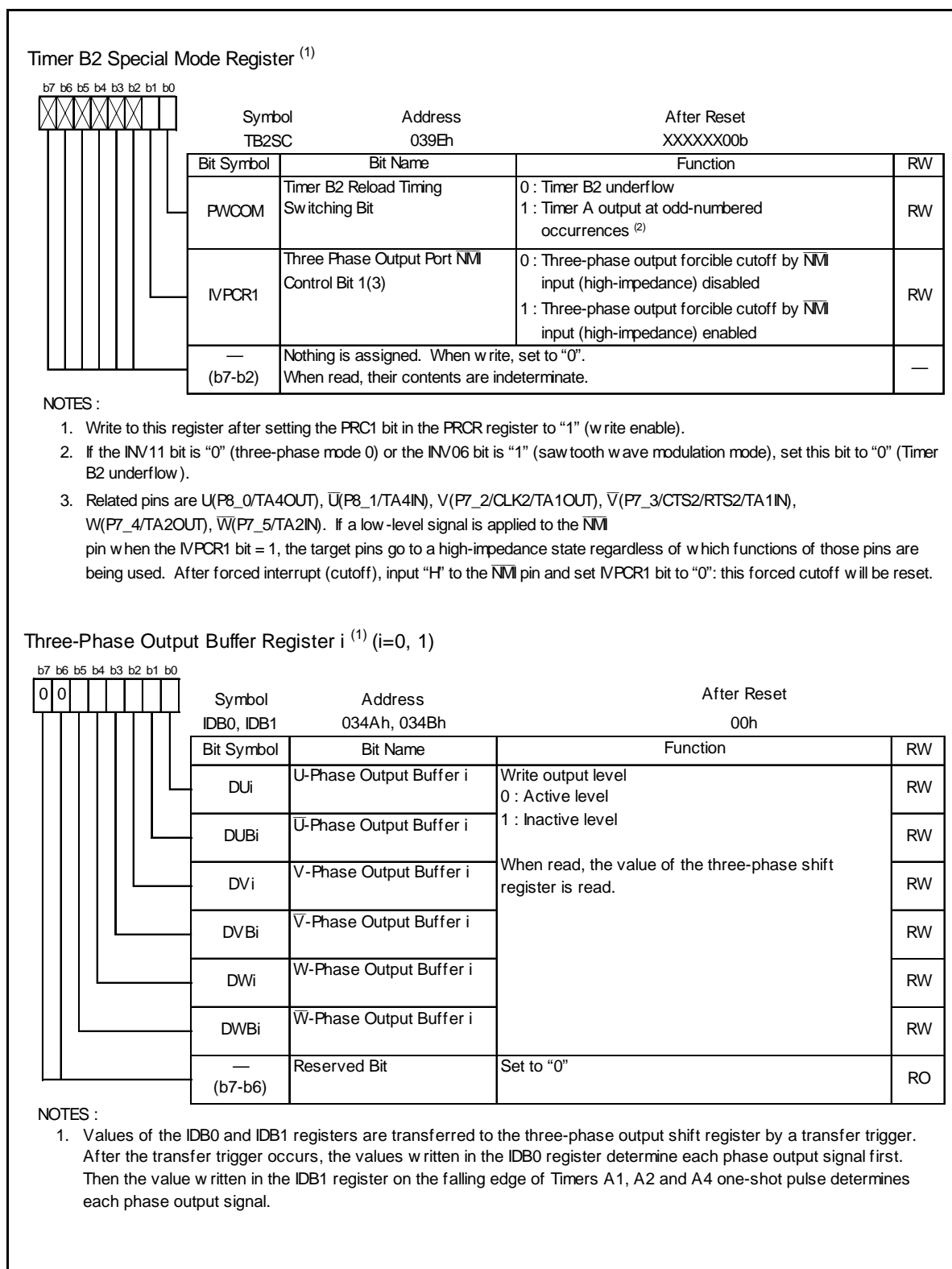


Figure 16.5 TB2SC, IDB0 and IDB1 Registers

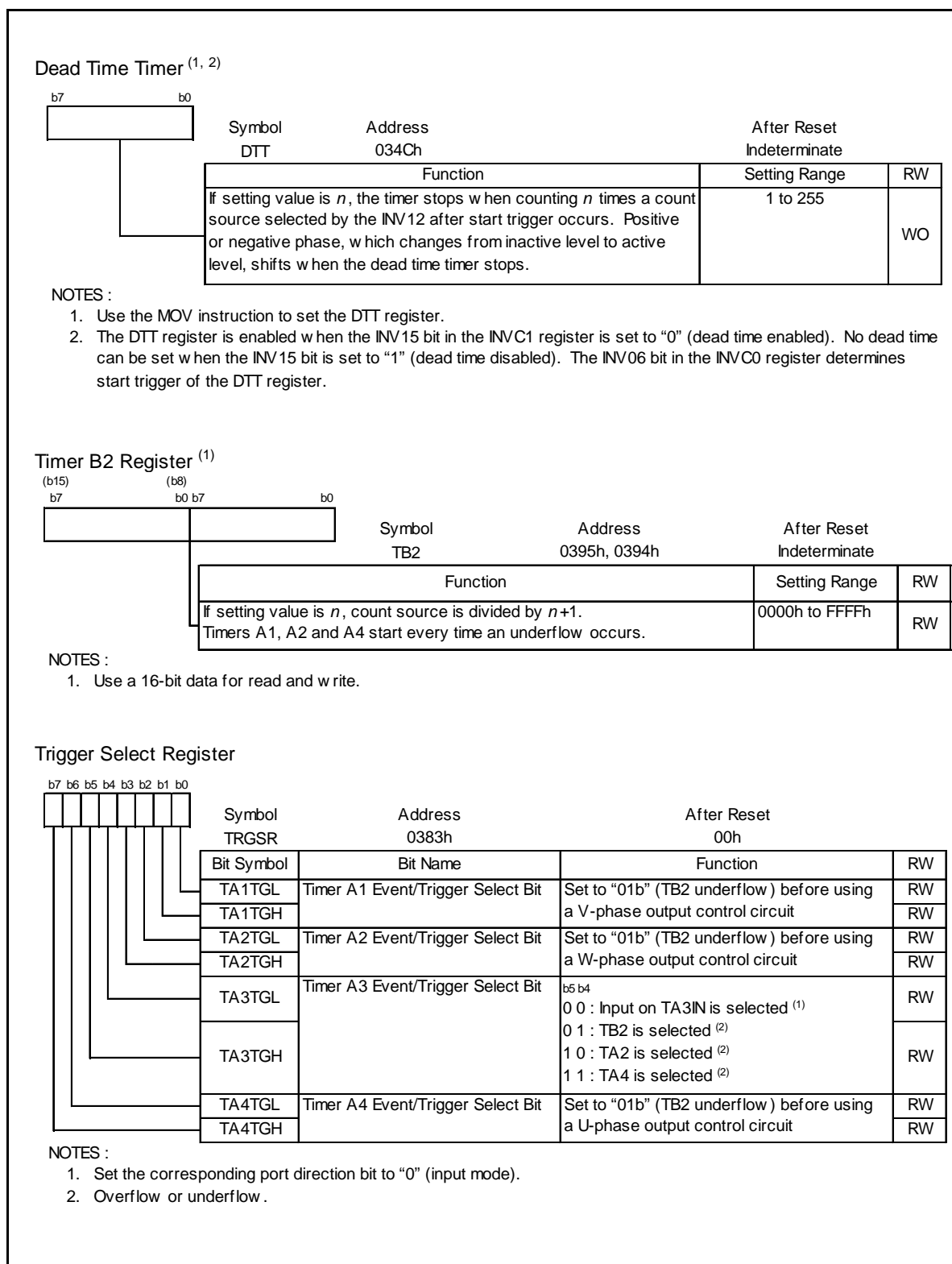


Figure 16.6 DTT, TB2 and TRGSR Registers

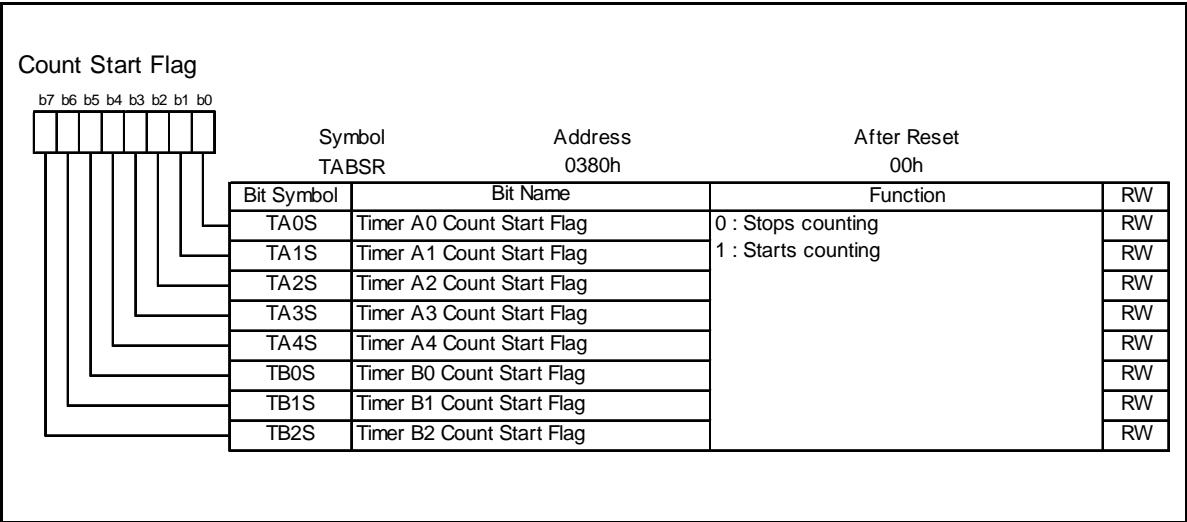


Figure 16.7 TABSR Register

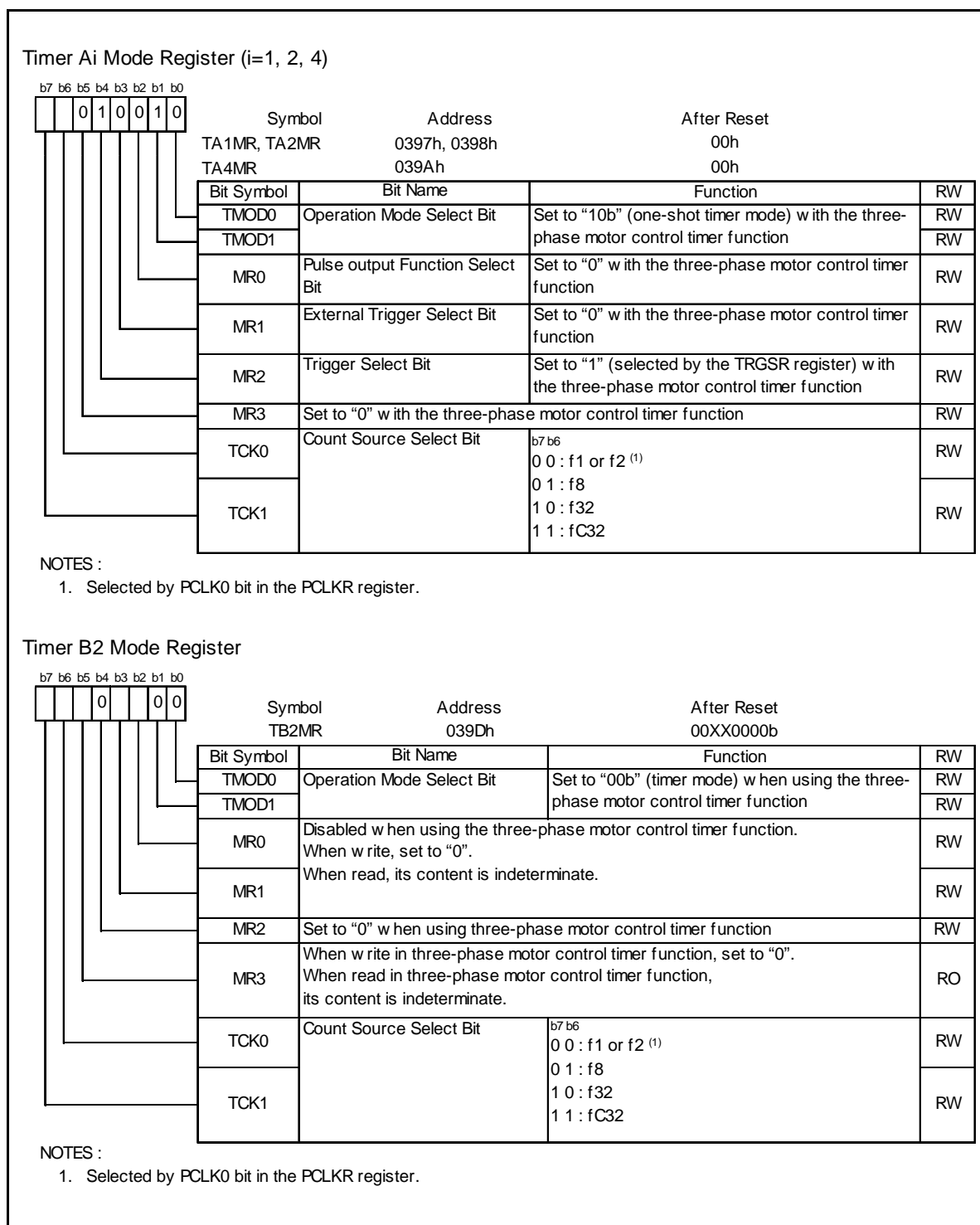


Figure 16.8 TA1MR, TA2MR, TA4MR and TB2MR Registers

The three-phase motor control timer function is enabled by setting the INV02 bit in the INVC0 register to “1”. When this function is on, timer B2 is used to control the carrier wave, and timers A4, A1 and A2 are used to control three-phase PWM outputs (U,  $\bar{U}$ , V,  $\bar{V}$ , W and  $\bar{W}$ ). The dead time is controlled by a dedicated dead time timer. Figure 16.9 shows the example of Triangular Wave Modulation Operation and Figure 16.10 shows the example of Sawtooth Wave Modulation Operation.

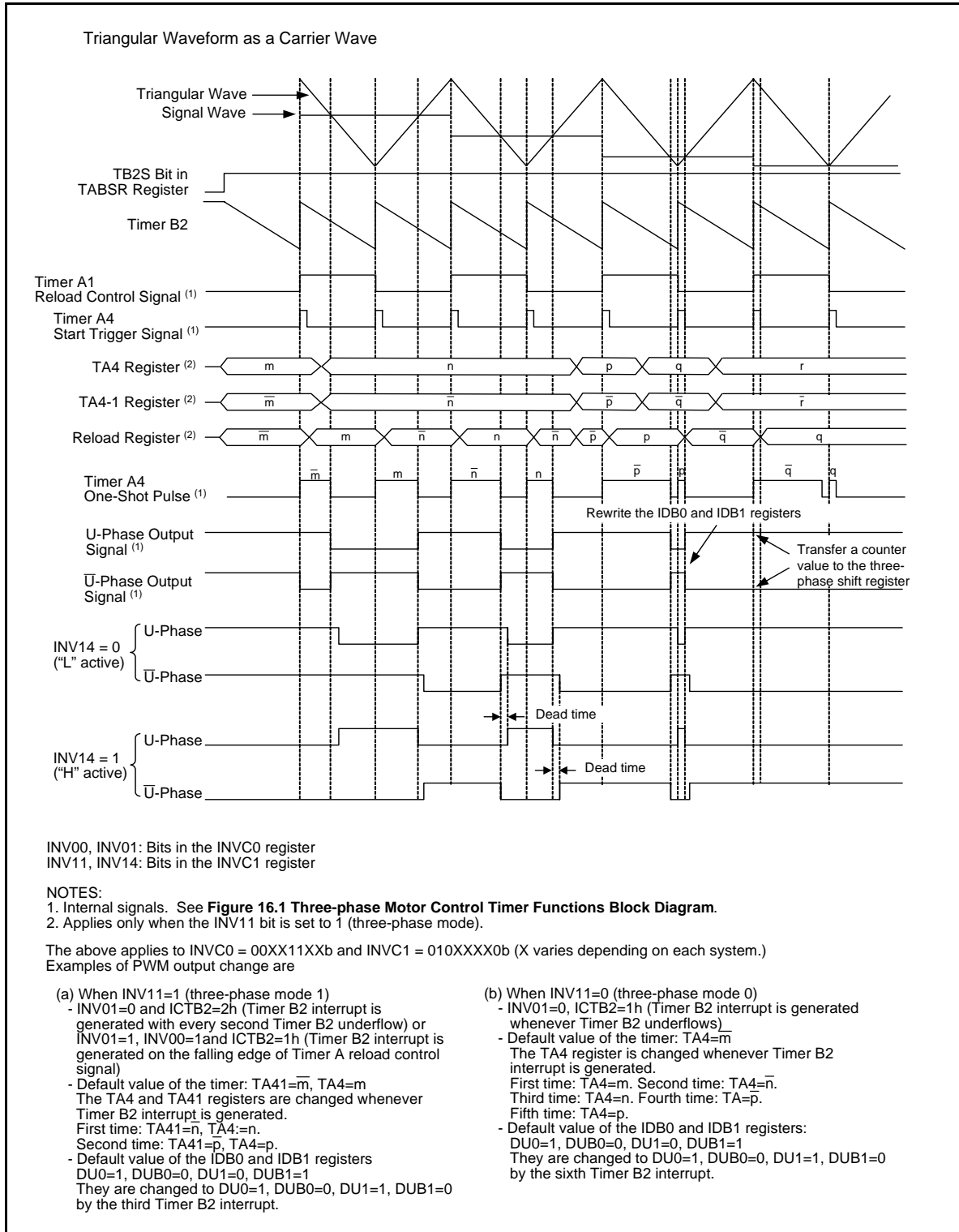
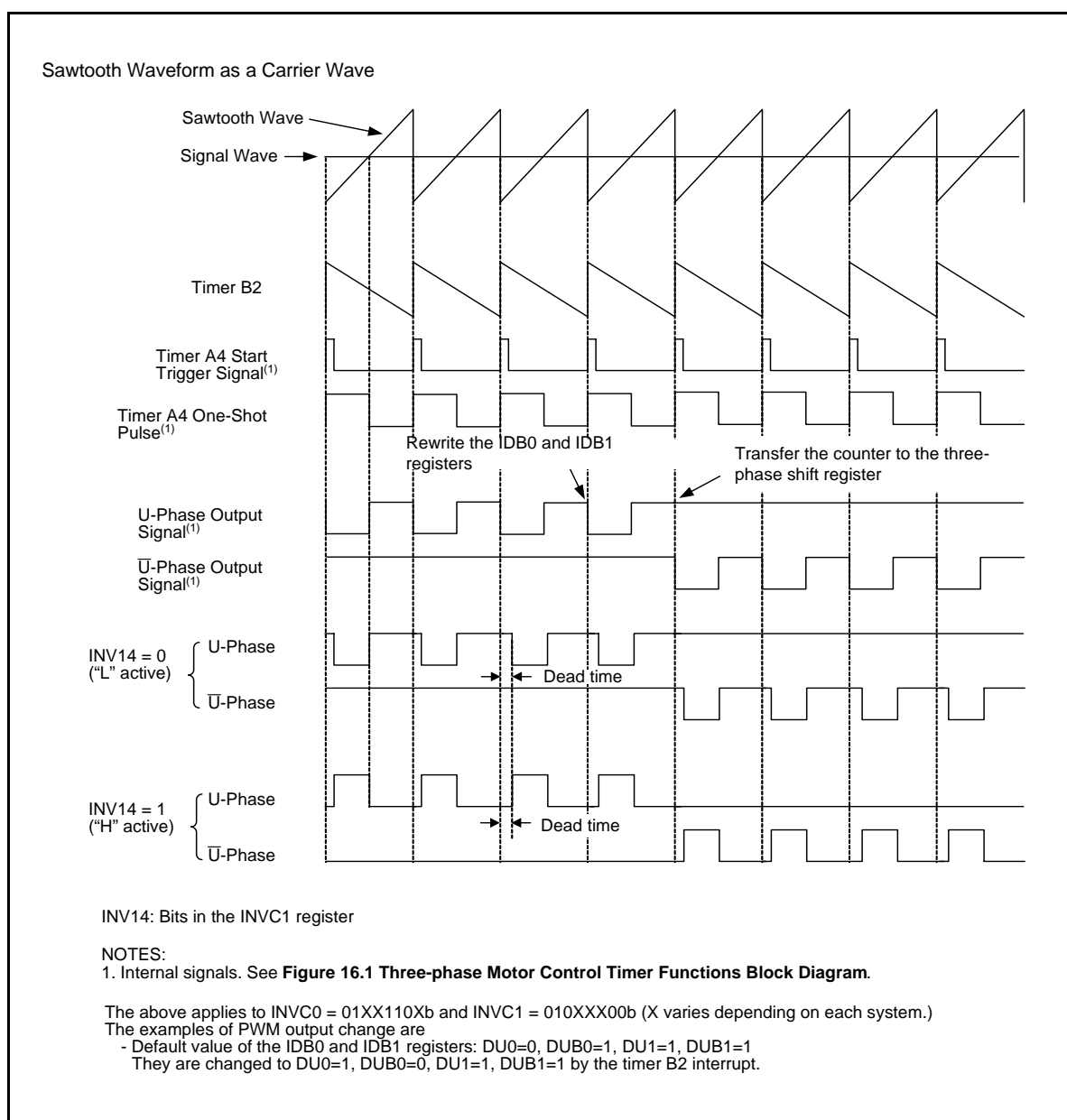


Figure 16.9 Triangular Wave Modulation Operation

**Figure 16.10 Sawtooth Wave Modulation Operation**

## 17. Serial Interface

### Note

The M16C/62P (80-pin version) and M16C/62PT (80-pin version) do not include CLK2,  $\overline{\text{CTS2}}$ / $\overline{\text{RTS2}}$  and SIN pins. Do not use the function which needs these pins.

Serial interface is configured with 5 channels: UART0 to UART2, SI/O3 and SI/O4.

### 17.1 UARTi (i=0 to 2)

#### Note

The M16C/62P (80-pin version) and M16C/62PT (80-pin version) do not include CLK2,  $\overline{\text{CTS2}}$ / $\overline{\text{RTS2}}$  pins of UART2.

#### [Precautions when using UART2]

- |  |   |
|--|---|
| • Clock synchronous serial I/O mode              | Cannot be used.   |
| • Clock asynchronous serial I/O mode (UART mode) | The $\overline{\text{CTS2}}$ / $\overline{\text{RTS2}}$ function and the external clock of transfer clock cannot be used. Set the CKDIR bit in the U2MR register to "0" and the CRD bit in the U2C0 register to "1" when using the UART mode. |
| • Special mode 2                                 | The slave mode cannot be used. Set the CKDIR bit register to "0" when using the Special mode 2.   |
| • Special mode 3                                 | The external clock of transfer clock cannot be used. Set the CKDIR bit register to "0" when using the Special mode 3.   |
| • Special mode 4 (SIM mode)                      | The external clock of transfer clock cannot be used. Set the CKDIR bit register to "0" when using the Special mode 4 (SIM mode).  |

UARTi each have an exclusive timer to generate a transfer clock, so they operate independently of each other. Figures 17.1 to 17.3 shows the block diagram of UART0 to UART2. Figure 17.4 shows the UARTi Transmit/Receive Unit.

UARTi has the following modes:

- Clock synchronous serial I/O mode
- Clock asynchronous serial I/O mode (UART mode).
- Special mode 1 (I<sup>2</sup>C mode)
- Special mode 2
- Special mode 3 (Bus collision detection function, IE mode) : UART0, UART1
- Special mode 4 (SIM mode) : UART2

Figures 17.5 to 17.12 show the UARTi-related registers. Refer to tables listing each mode for register setting.

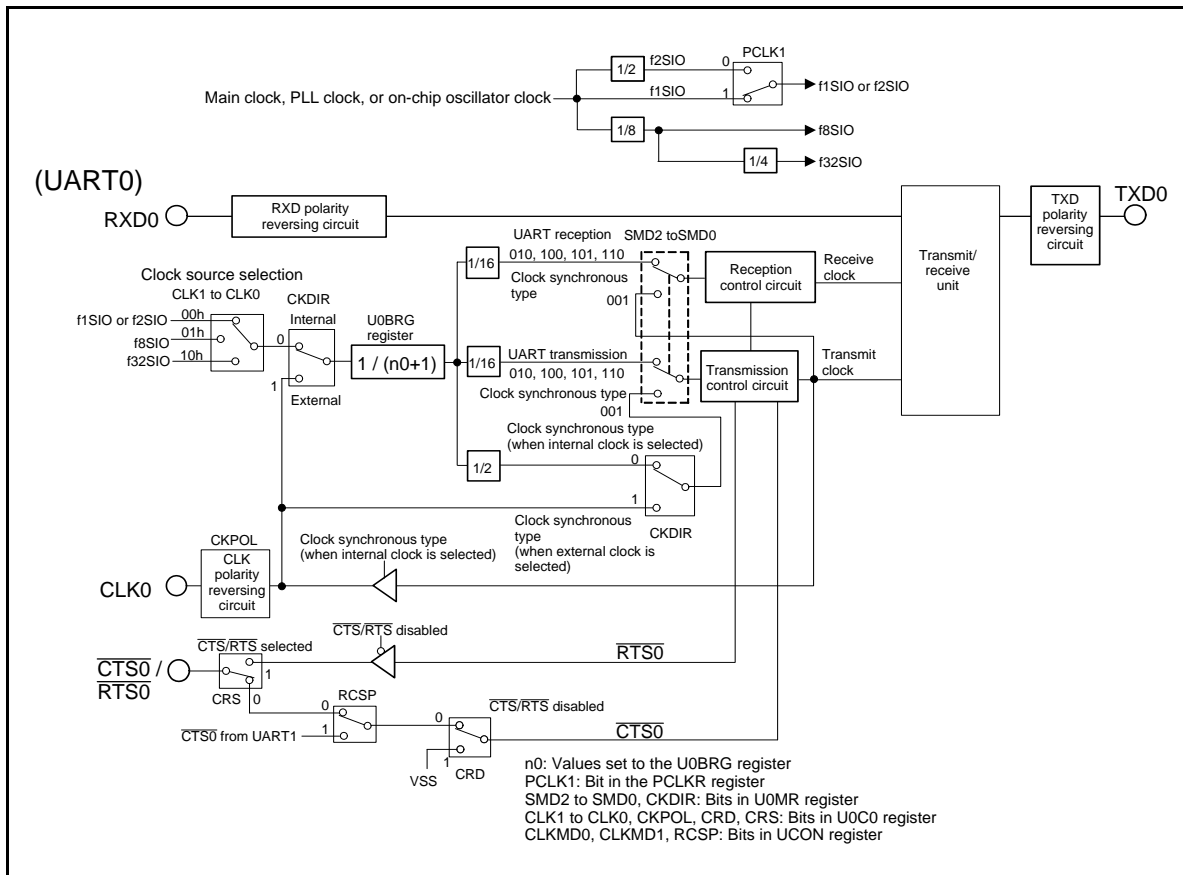


Figure 17.1 UART0 Block Diagram





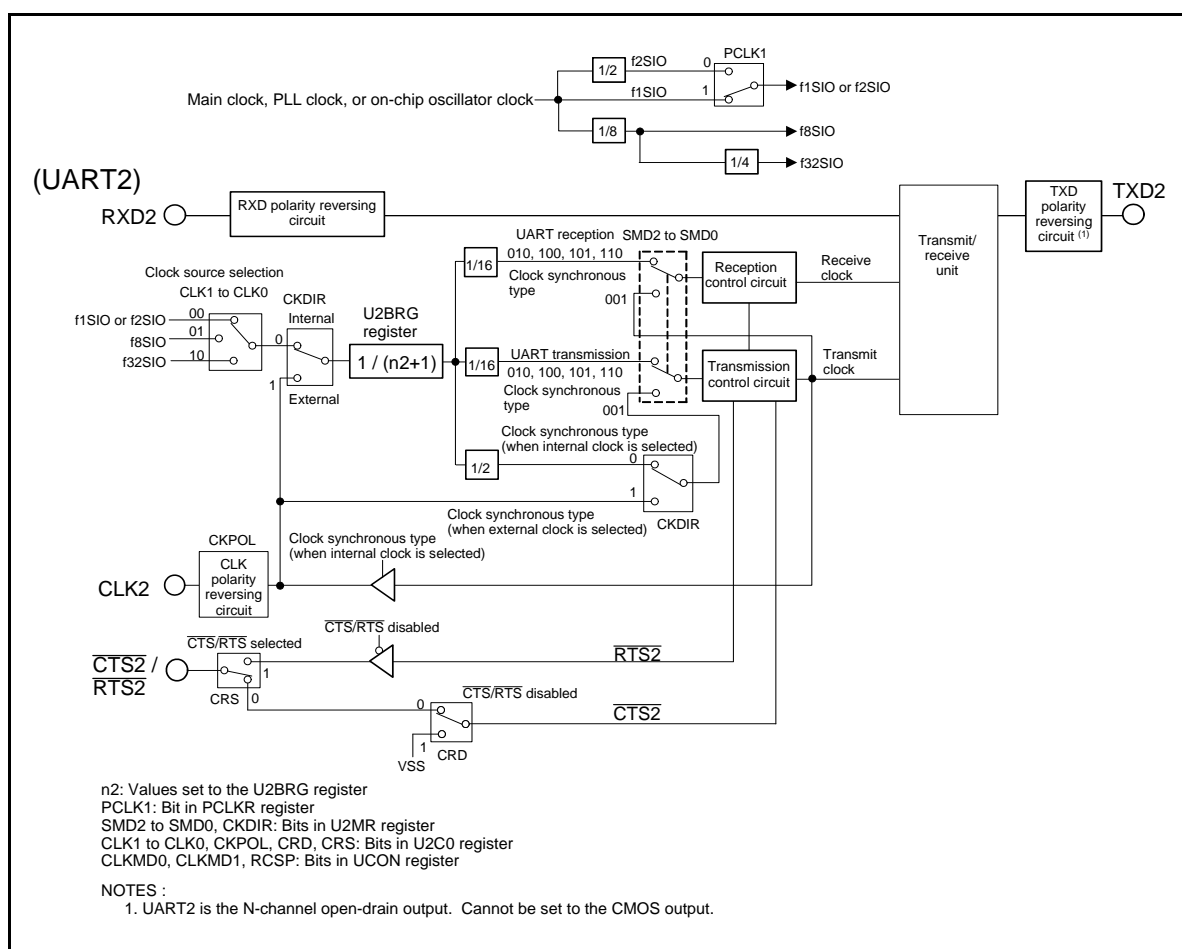


Figure 17.3 UART2 Block Diagram

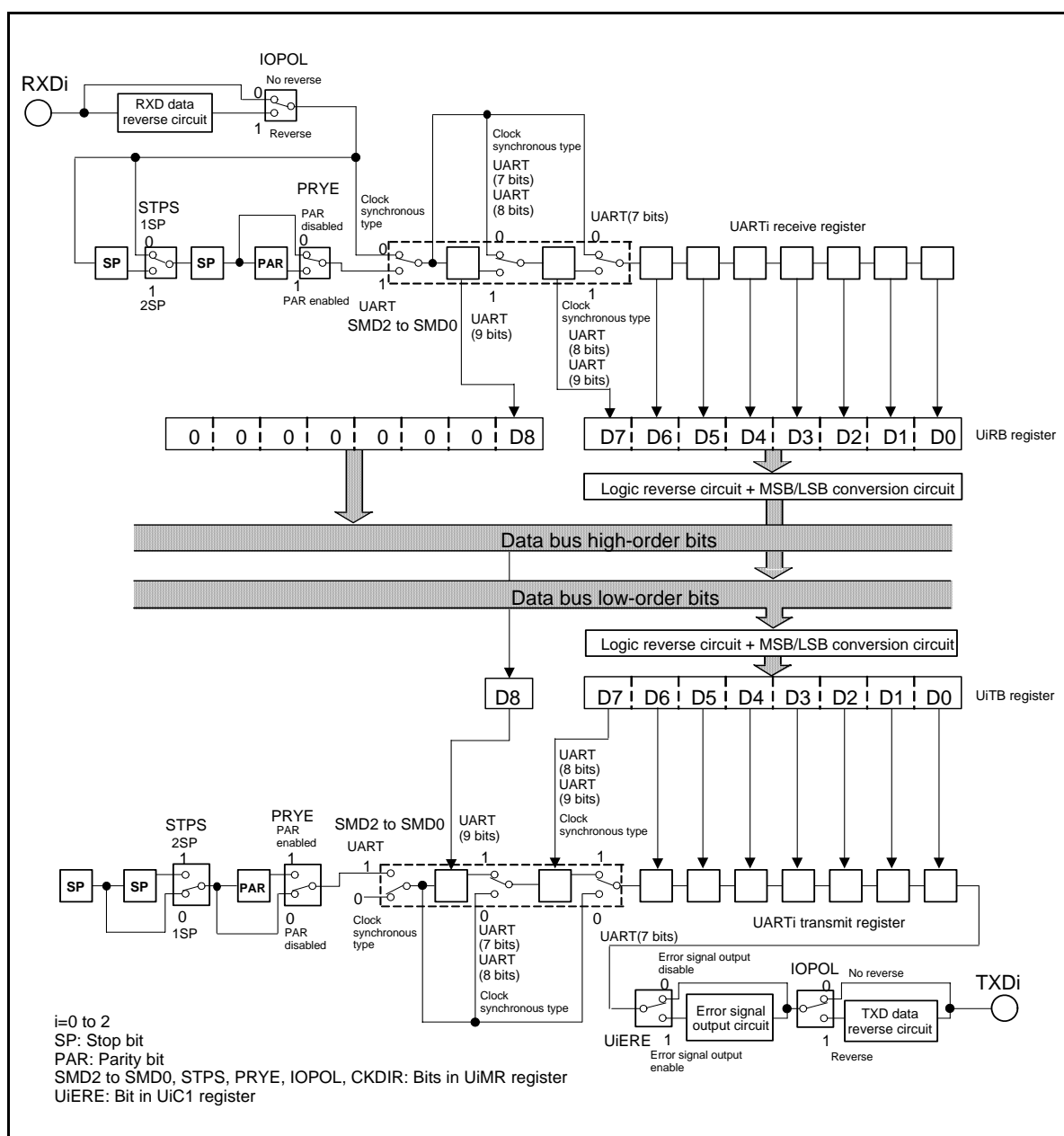


Figure 17.4 UARTi Transmit/Receive Unit

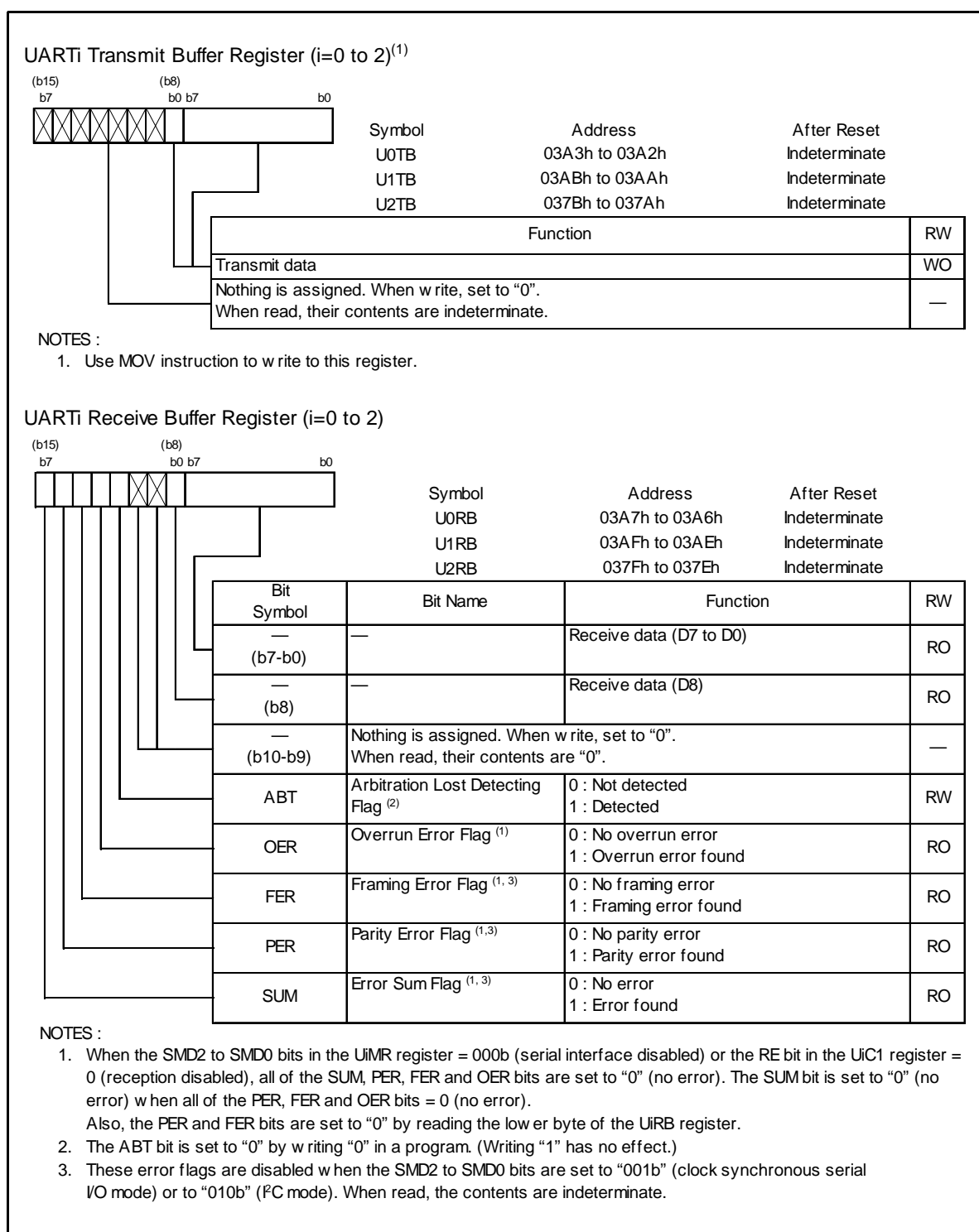


Figure 17.5 UiTB and UIRB Registers

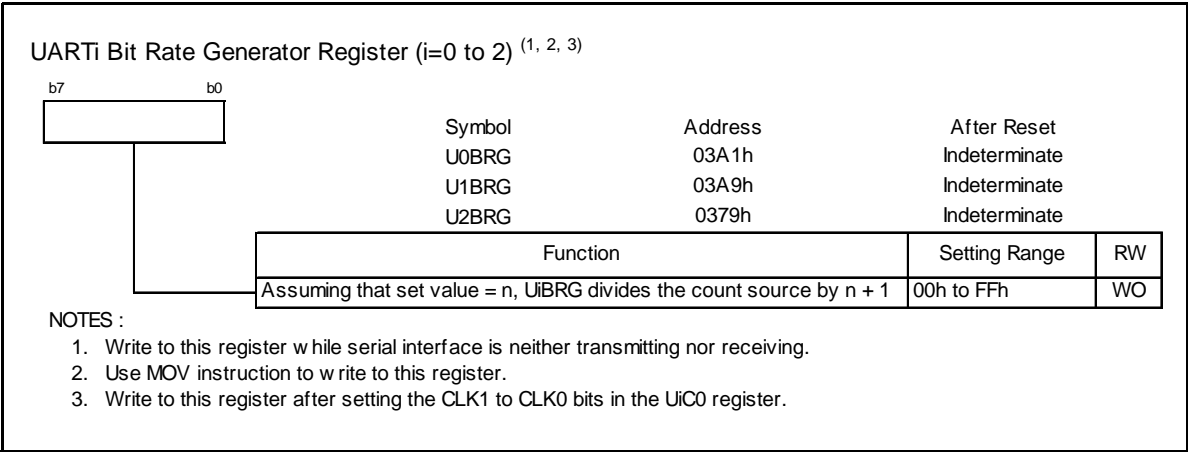


Figure 17.6 UiBRG Register

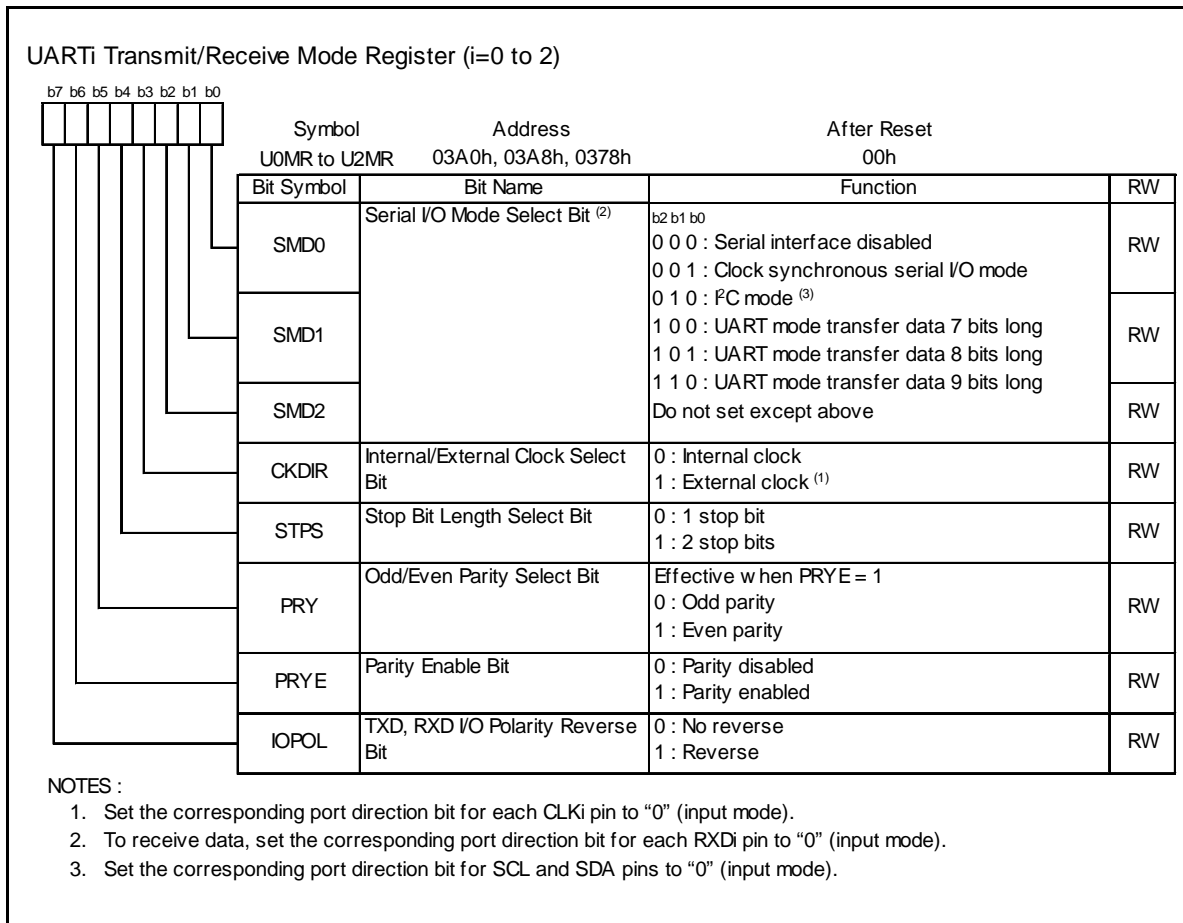


Figure 17.7 UiMR Register

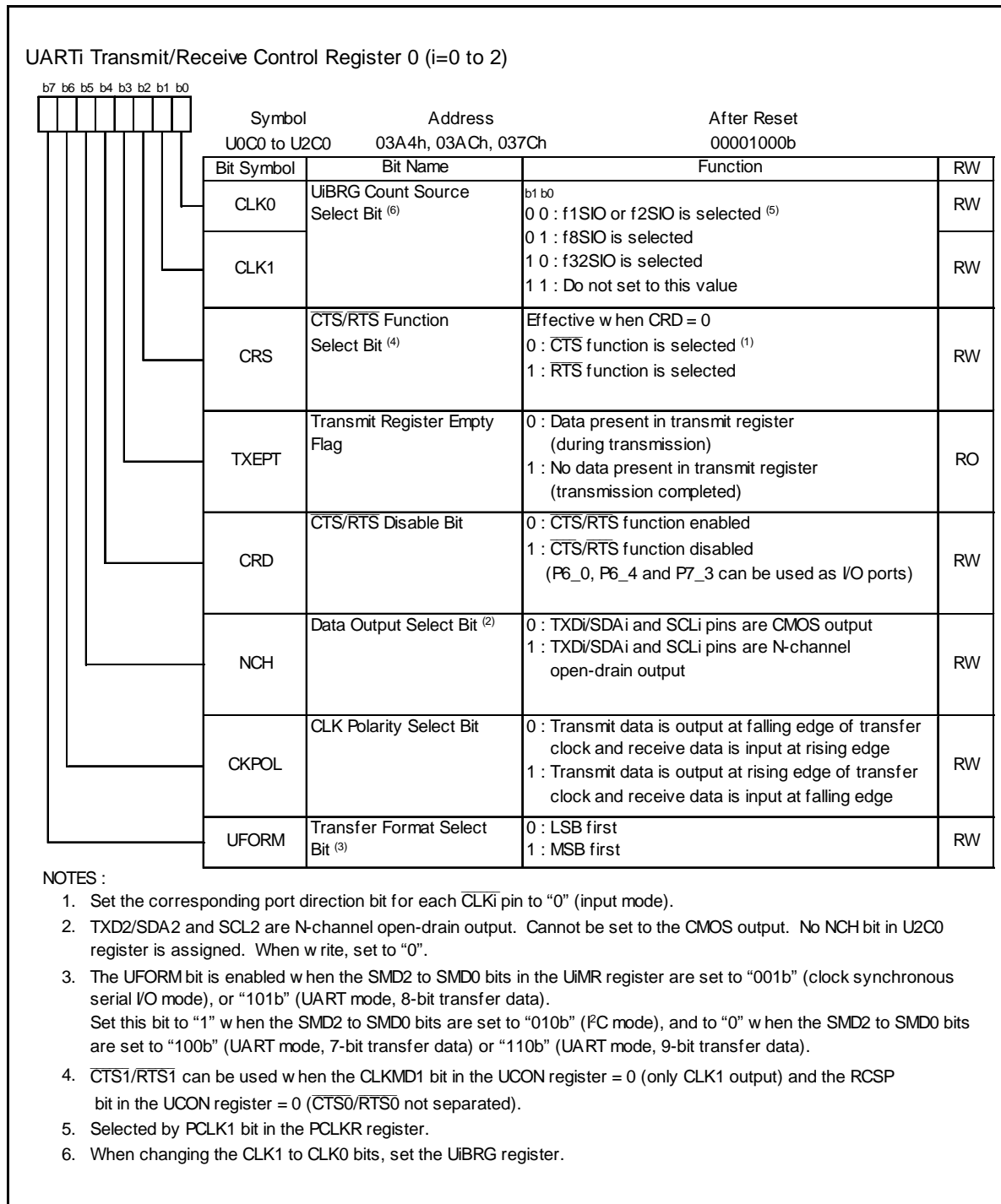


Figure 17.8 UiC0 Register

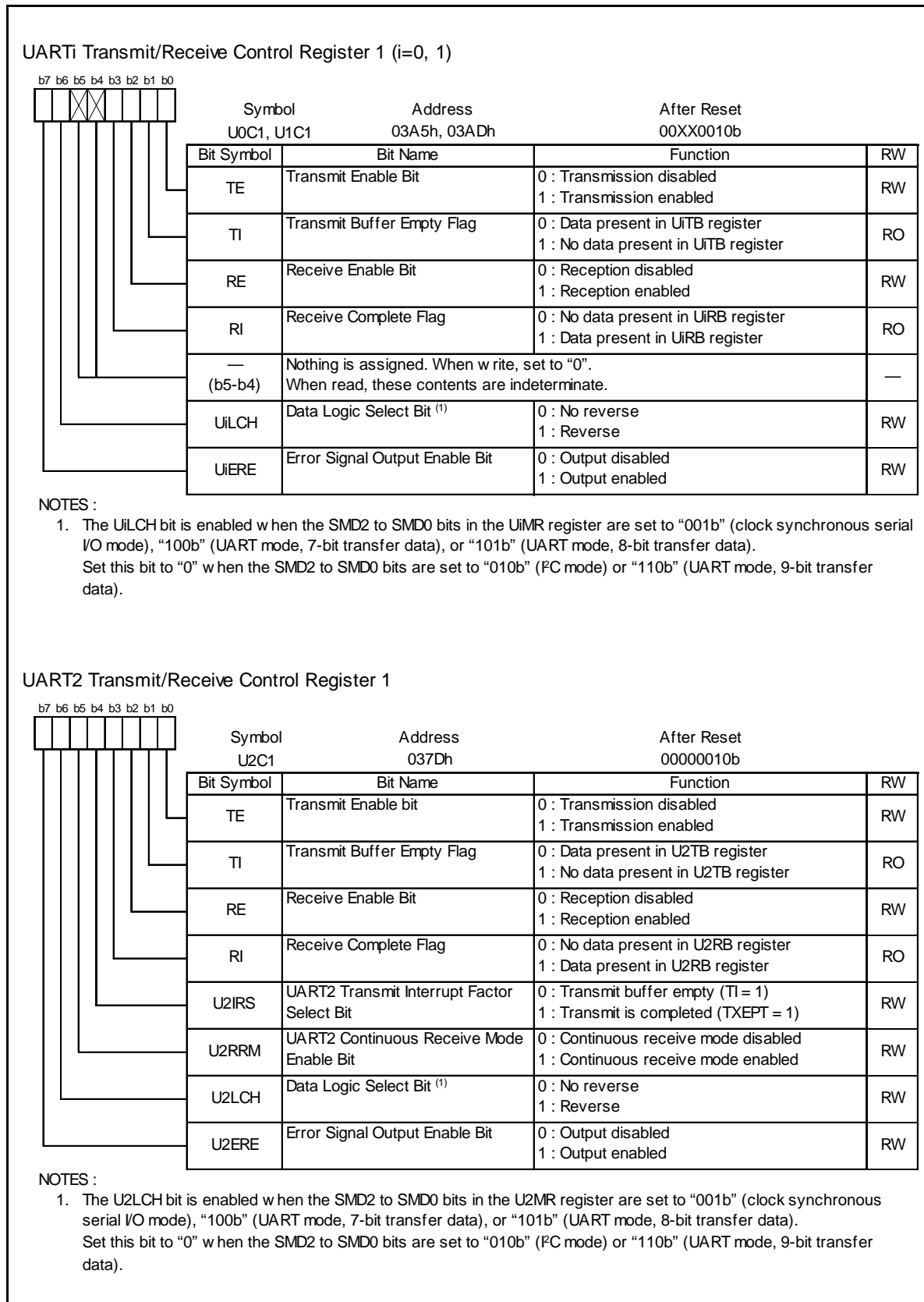


Figure 17.9 U0C1 to U2C1 Registers



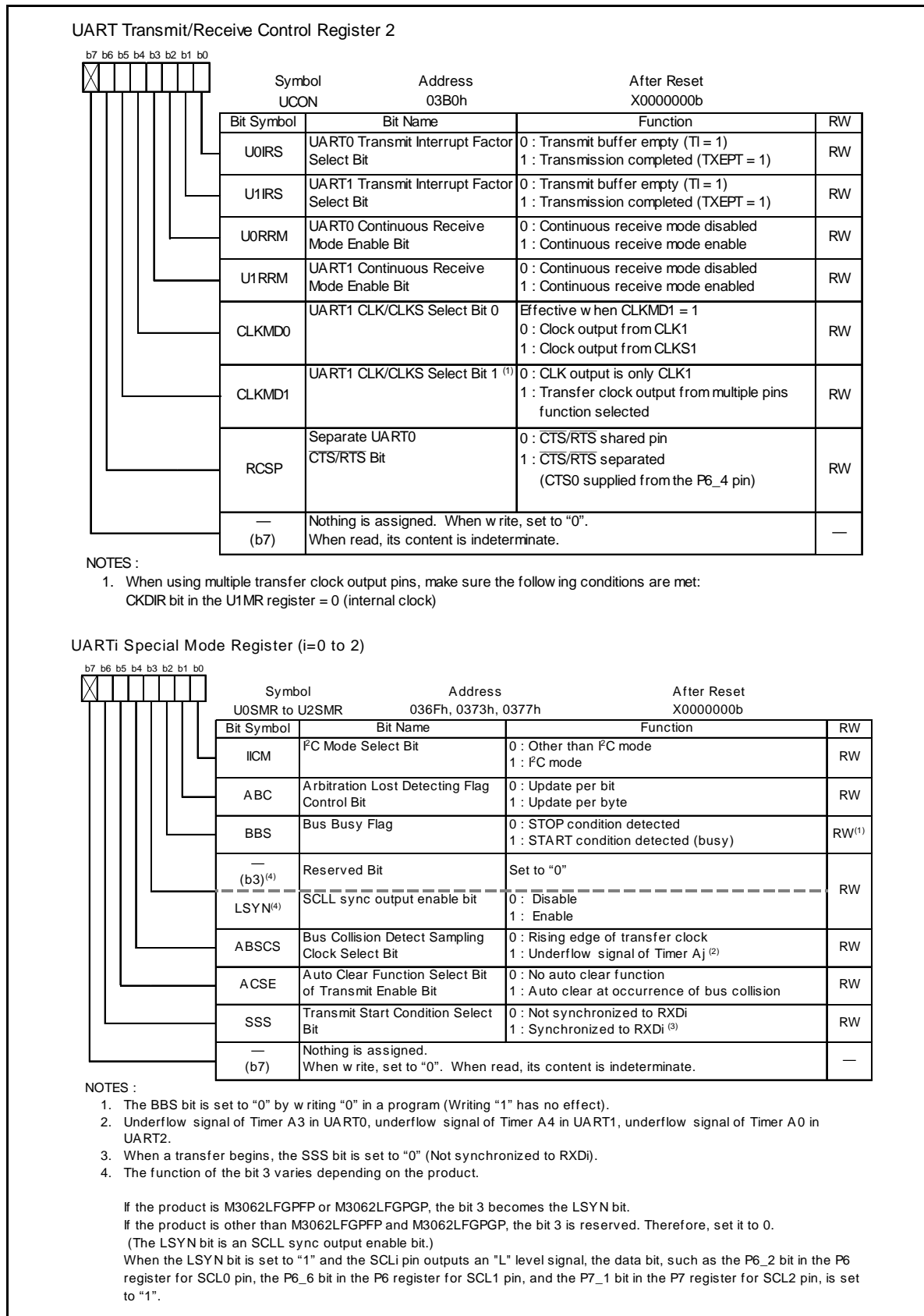
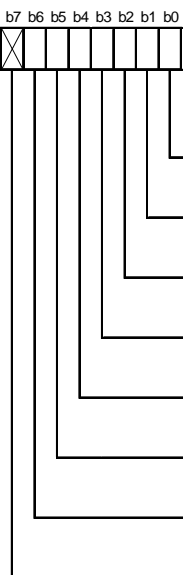


Figure 17.10 UCON and UiSMR Registers

## UARTi Special Mode Register 2 (i=0 to 2)

								Symbol U0SMR2 to U2SMR2	Address 036Eh, 0372h, 0376h	After Reset X0000000b
Bit Symbol	Bit Name	Function		RW						
IICM2	I <sup>2</sup> C Mode Select Bit 2	See <b>Table 17.13 I<sup>2</sup>C Mode Functions</b>		RW						
CSC	Clock-Synchronous Bit	0 : Disabled 1 : Enabled		RW						
SWC	SCL Wait Output Bit	0 : Disabled 1 : Enabled		RW						
ALS	SDA Output Stop Bit	0 : Disabled 1 : Enabled		RW						
STAC	UARTi Initialization Bit	0 : Disabled 1 : Enabled		RW						
SWC2	SCL Wait Output Bit 2	0: Transfer clock 1: "L" output		RW						
SDHI	SDA Output Disable Bit	0: Enabled 1: Disabled (high-impedance)		RW						
— (b7)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.			—						

## UARTi special mode register 3 (i=0 to 2)

<div><div>b7b6b5b4b3b2b1b0</div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div></div>								Symbol	Address	After Reset
								U0SMR3 to U2SMR3	036Dh, 0371h, 0375h	000X0X0Xb
Bit Symbol		Bit Name		Function				RW		
— (b0)		Nothing is assigned. When write, set to “0”. When read, its content is indeterminate.						—		
CKPH		Clock Phase Set Bit		0 : Without clock delay 1 : With clock delay				RW		
— (b2)		Nothing is assigned. When write, set to “0”. When read, its content is indeterminate.						—		
NODC		Clock Output Select Bit		0 : CLKi is CMOS output 1 : CLKi is N-channel open drain output				RW		
— (b4)		Nothing is assigned. When write, set to “0”. When read, its content is indeterminate.						—		
DL0		SDAi Digital Delay Setup Bit <sup>(1, 2)</sup>		b7 b6 b5 0 0 0 : Without delay 0 0 1 : 1 to 2 cycle(s) 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		

## NOTES :

1. The DL2 to DL0 bits are used to generate a delay in SDAi output by digital means during I<sup>2</sup>C mode. In other than I<sup>2</sup>C mode, set these bits to "000b" (no delay).
2. The amount of delay varies with the load on SCLi and SDAi pins. Also, when using an external clock, the amount of delay increases by about 100 ns.

Figure 17.11 UiSMR2 and UiSMR3 Registers

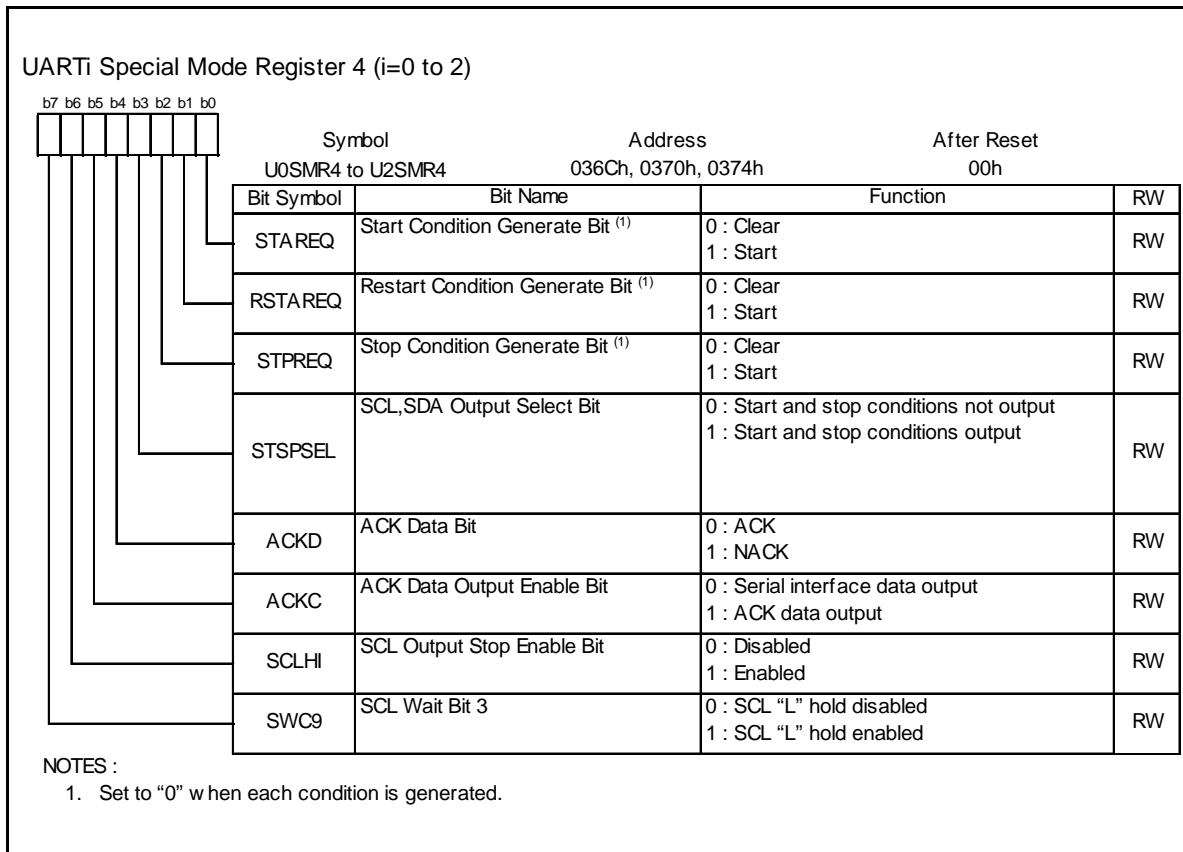


Figure 17.12 UiSMR4 Register

### 17.1.1 Clock Synchronous Serial I/O Mode

The clock synchronous serial I/O mode uses a transfer clock to transmit and receive data. Table 17.1 lists the Clock Synchronous Serial I/O Mode Specifications. Table 17.2 lists the Registers to Be Used and Settings in Clock Synchronous Serial I/O Mode.

**Table 17.1 Clock Synchronous Serial I/O Mode Specifications**

Item	Specification
Transfer Data Format	Transfer data length: 8 bits
Transfer Clock	<ul style="list-style-type: none"> <li>CKDIR bit in the UiMR(i=0 to 2) register = 0 (internal clock) : <math>f_j / (2(n+1))</math>  <math>f_j = f1SIO, f2SIO, f8SIO, f32SIO</math> n: Setting value of UiBRG register 00h to FFh</li> <li>CKDIR bit = 1 (external clock) : Input from CLKi pin</li> </ul>
Transmission, Reception Control	Selectable from $\overline{CTS}$ function, $\overline{RTS}$ function or $\overline{CTS}/\overline{RTS}$ function disable
Transmission Start Condition	Before transmission can start, meet the following requirements <sup>(1)</sup> <ul style="list-style-type: none"> <li>The TE bit in the UiC1 register = 1 (transmission enabled)</li> <li>The TI bit in the UiC1 register = 0 (data present in UiTB register)</li> <li>If <math>\overline{CTS}</math> function is selected, input on the CTSi pin = L</li> </ul>
Reception Start Condition	Before reception can start, meet the following requirements <sup>(1)</sup> <ul style="list-style-type: none"> <li>The RE bit in the UiC1 register = 1 (reception enabled)</li> <li>The TE bit in the UiC1 register = 1 (transmission enabled)</li> <li>The TI bit in the UiC1 register = 0 (data present in the UiTB register)</li> </ul>
Interrupt Request Generation Timing	For transmission, one of the following conditions can be selected <ul style="list-style-type: none"> <li>The UiIRS bit <sup>(3)</sup> = 0 (transmit buffer empty): when transferring data from the UiTB register to the UARTi transmit register (at start of transmission)</li> <li>The UiIRS bit = 1 (transfer completed): when the serial interface finished sending data from the UARTi transmit register</li> </ul> For reception <ul style="list-style-type: none"> <li>When transferring data from the UARTi receive register to the UiRB register (at completion of reception)</li> </ul>
Error Detection	Overrun error <sup>(2)</sup> This error occurs if the serial interface started receiving the next data before reading the UiRB register and received the 7th bit of the next data
Select Function	<ul style="list-style-type: none"> <li>CLK polarity selection Transfer data input/output can be chosen to occur synchronously with the rising or the falling edge of the transfer clock</li> <li>LSB first, MSB first selection Whether to start sending/receiving data beginning with bit 0 or beginning with bit 7 can be selected</li> <li>Continuous receive mode selection Reception is enabled immediately by reading the UiRB register</li> <li>Switching serial data logic This function reverses the logic value of the transmit/receive data</li> <li>Transfer clock output from multiple pins selection (UART1) The output pin can be selected in a program from two UART1 transfer clock pins that have been set</li> <li>Separate <math>\overline{CTS}/\overline{RTS}</math> pins (UART0) <math>\overline{CTS0}</math> and <math>\overline{RTS0}</math> are input/output from separate pins</li> </ul>

**NOTES:**

- When an external clock is selected, the conditions must be met while if the CKPOL bit in the UiC0 register = 0 (transmit data output at the falling edge and the receive data taken in at the rising edge of the transfer clock), the external clock is in the high state; if the CKPOL bit in the UiC0 register = 1 (transmit data output at the rising edge and the receive data taken in at the falling edge of the transfer clock), the external clock is in the low state.
- If an overrun error occurs, the receive data of UiRB register will be indeterminate. The IR bit in the SiRIC register does not change.
- The U0IRS and U1IRS bits respectively are the bits 0 and 1 in the UCON register; the U2IRS bit is the bit 4 in the U2C1 register.

**Table 17.2 Registers to Be Used and Settings in Clock Synchronous Serial I/O Mode**

Register	Bit	Function
UiTB <sup>(3)</sup>	0 to 7	Set transmission data
UiRB <sup>(3)</sup>	0 to 7	Reception data can be read
	OER	Overrun error flag
UiBRG	0 to 7	Set a bit rate
UiMR <sup>(3)</sup>	SMD2 to SMD0	Set to "001b"
	CKDIR	Select the internal clock or external clock
	IOPOL	Set to "0"
UiC0	CLK1 to CLK0	Select the count source for the UiBRG register
	CRS	Select $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ 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 <sup>(2)</sup>
	CKPOL	Select the transfer clock polarity
	UFORM	Select the LSB first or MSB first
UiC1	TE	Set this bit to "1" to enable transmission/reception
	TI	Transmit buffer empty flag
	RE	Set this bit to "1" to enable reception
	RI	Reception complete flag
	U2IRS <sup>(1)</sup>	Select the source of UART2 transmit interrupt
	U2RRM <sup>(1)</sup>	Set this bit to "1" to use continuous receive mode
	UiLCH	Set this bit 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, U1IRS	Select the source of UART0/UART1 transmit interrupt
	U0RRM, U1RRM	Set this bit to "1" to use continuous receive mode
	CLKMD0	Select the transfer clock output pin when CLKMD1 = 1
	CLKMD1	Set this bit to "1" to output UART1 transfer clock from two pins
	RCSP	Set this bit to "1" to accept as input the $\overline{\text{CTS0}}$ signal of the UART0 from the P6_4 pin
	7	Set to "0"

**NOTES:**

1. Set the bit 4 and bit 5 in the U0C1 and U1C1 register to "0". The U0IRS, U1IRS, U0RRM and U1RRM bits in the UCON register.
2. TXD2 pin is N channel open-drain output. Set the NCH bit in the U2C0 register to "0".
3. Not all register bits are described above. Set those bits to "0" when writing to the registers in clock synchronous serial I/O mode.

i=0 to 2

Table 17.3 lists the functions of the input/output pins during clock synchronous serial I/O mode. Table 17.3 shows pin functions for the case where the multiple transfer clock output pin select function is deselected.

Table 17.4 lists the P6\_4 Pin Functions during clock synchronous serial I/O mode. Note that for a period from when the UARTi operating mode is selected to when transfer starts, the TXDi pin outputs an “H” (If the N-channel open-drain output is selected, this pin is in a high-impedance state).

**Table 17.3 Pin Functions (when not select multiple transfer clock output pin function)**

Pin Name	Function	Method of Selection
TXDi (i = 0 to 2) (P6_3, P6_7, P7_0)	Serial Data Output	(Outputs dummy data when performing reception only)
RXDi (P6_2, P6_6, P7_1)	Serial Data Input	PD6_2 bit and PD6_6 bit in the PD6 register = 0, PD7_1 bit in the PD7 register = 0 (Can be used as an input port when performing transmission only)
CLKi (P6_1, P6_5, P7_2)	Transfer Clock Output	CKDIR bit in the UiMR register = 0
	Transfer Clock Input	CKDIR bit = 1 PD6_1 bit and PD6_5 bit in the PD6 register = 0, PD7_2 bit in the PD7 register = 0
CTS $\overline{\text{RTS}}$ i (P6_0, P6_4, P7_3)	CTS Input	CRD bit in the UiC0 register = 0 CRS bit in the UiC0 register = 0 PD6_0 and PD6_4 bit in the PD6 register = 0, PD7_3 bit in the PD7 register = 0
	RTS Output	CRD bit = 0 CRS bit = 1
	I/O Port	CRD bit = 1

**Table 17.4 P6\_4 Pin Functions**

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
CTS1	0	0	0	0	–	0
RTS1	0	1	0	0	–	–
CTS0 (1)	0	0	1	0	–	0
CLKS1	–	–	–	1 (2)	1	–

– : “0” or “1”

**NOTES:**

- In addition to this, set the CRD bit in the U0C0 register to “0” (CTS0/RTS0 enabled) and the CRS bit in the U0C0 register to “1” (RTS0 selected).
- When the CLKMD1 bit = 1 and the CLKMD0 bit = 0, the following logic levels are output:
  - High if the CLKPOL bit in the U1C0 register = 0
  - Low if the CLKPOL bit = 1

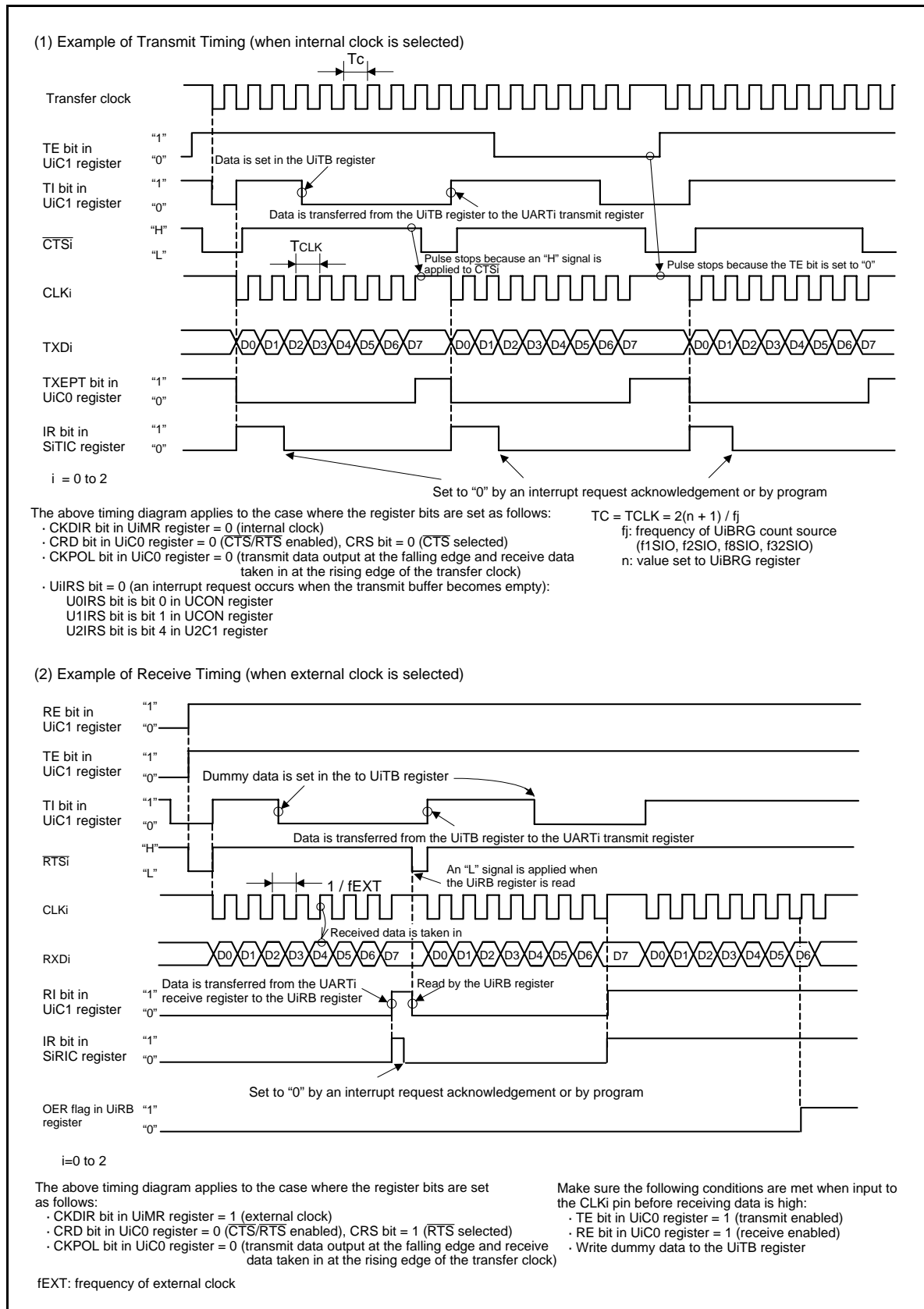


Figure 17.13 Transmit and Receive Operation

### 17.1.1.1 Counter Measure for Communication Error Occurs

If a communication error occurs while transmitting or receiving in clock synchronous serial I/O mode, follow the procedures below.

- Resetting the UiRB register (i=0 to 2)
  - (1) Set the RE bit in the UiC1 register to “0” (reception disabled)
  - (2) Set the SMD2 to SMD0 bits in the UiMR register to “000b” (Serial interface disabled)
  - (3) Set the SMD2 to SMD0 bits in the UiMR register to “001b” (Clock synchronous serial I/O mode)
  - (4) Set the RE bit in the UiC1 register to “1” (reception enabled)
- Resetting the UiTB register (i=0 to 2)
  - (1) Set the SMD2 to SMD0 bits in the UiMR register “000b” (Serial interface disabled)
  - (2) Set the SMD2 to SMD0 bits in the UiMR register “001b” (Clock synchronous serial I/O mode)
  - (3) “1” is written to RE bit in the UiC1 register (transmission enabled), regardless of the TE bit in the UiCi register

### 17.1.1.2 CLK Polarity Select Function

Use the CKPOL bit in the UiC0 register (i = 0 to 2) to select the transfer clock polarity. Figure 17.14 shows the Transfer Clock Polarity.

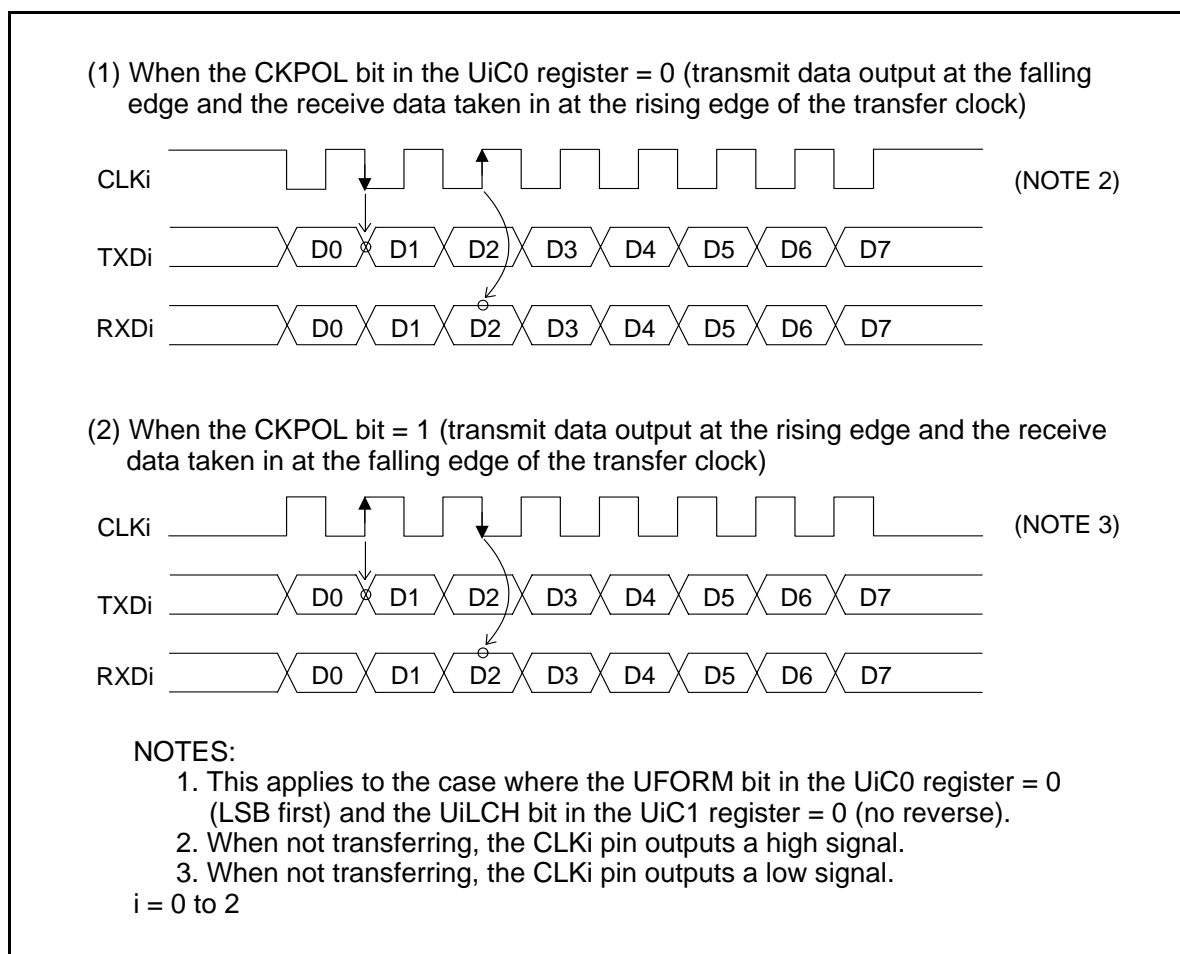


Figure 17.14 Transfer Clock Polarity



### 17.1.1.3 LSB First/MSB First Select Function

Use the UFORM bit in the UiC0 register (i = 0 to 2) to select the transfer format. Figure 17.15 shows the Transfer Format.

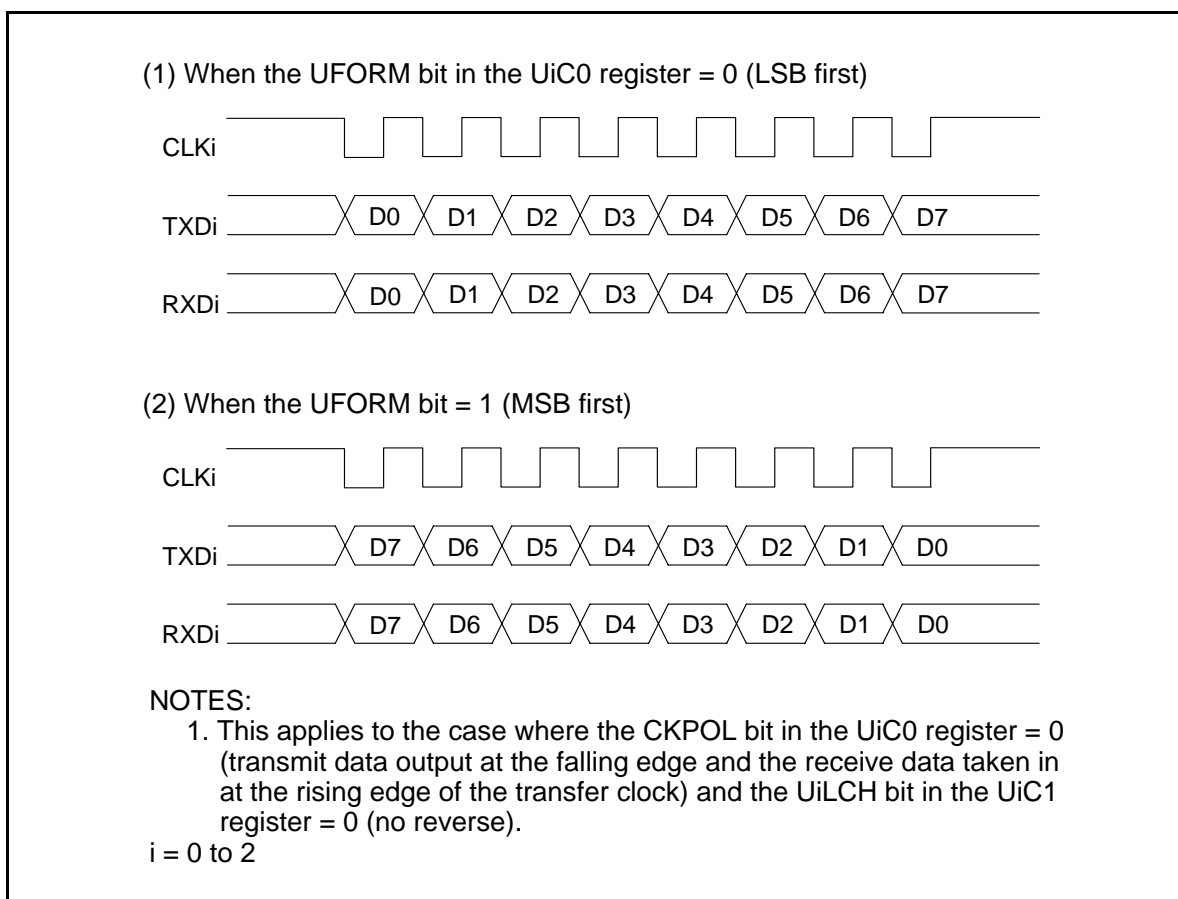


Figure 17.15 Transfer Format

### 17.1.1.4 Continuous Receive Mode

In continuous receive mode, receive operation becomes enable when the receive buffer register is read. It is not necessary to write dummy data into 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 (i = 0 to 2) = 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, i.e., UiRRM bit = 1, do not write dummy data to the UiTB register in a program. The U0RRM and U1RRM bits are the bit 2 and bit 3 in the UCON register, respectively, and the U2RRM bit is the bit 5 in the U2C1 register.

### 17.1.1.5 Serial Data Logic Switching Function

When the UiLCH bit in the UiC1 register ( $i = 0$  to  $2$ ) = 1 (reverse), the 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 17.16 shows Serial Data Logic Switching.

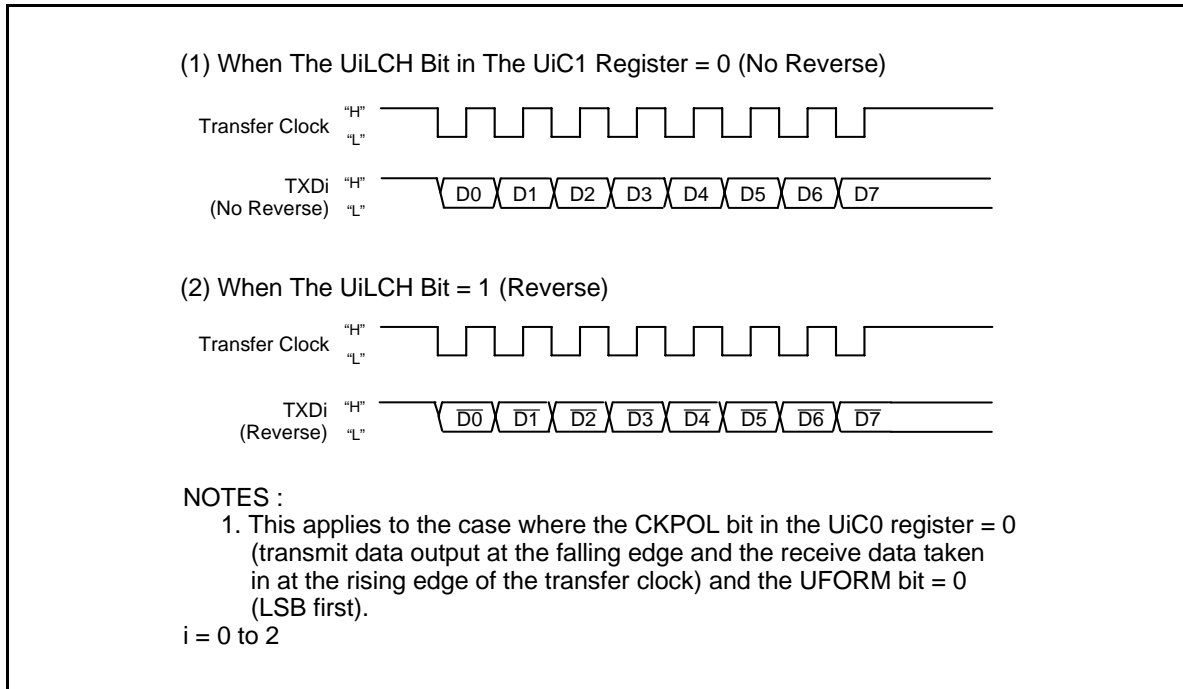


Figure 17.16 Serial Data Logic Switching

### 17.1.1.6 Transfer Clock Output From Multiple Pins (UART1)

Use the CLKMD1 to CLKMD0 bits in the UCON register to select one of the two transfer clock output pins (see Figure 17.17). This function can be used when the selected transfer clock for UART1 is an internal clock.

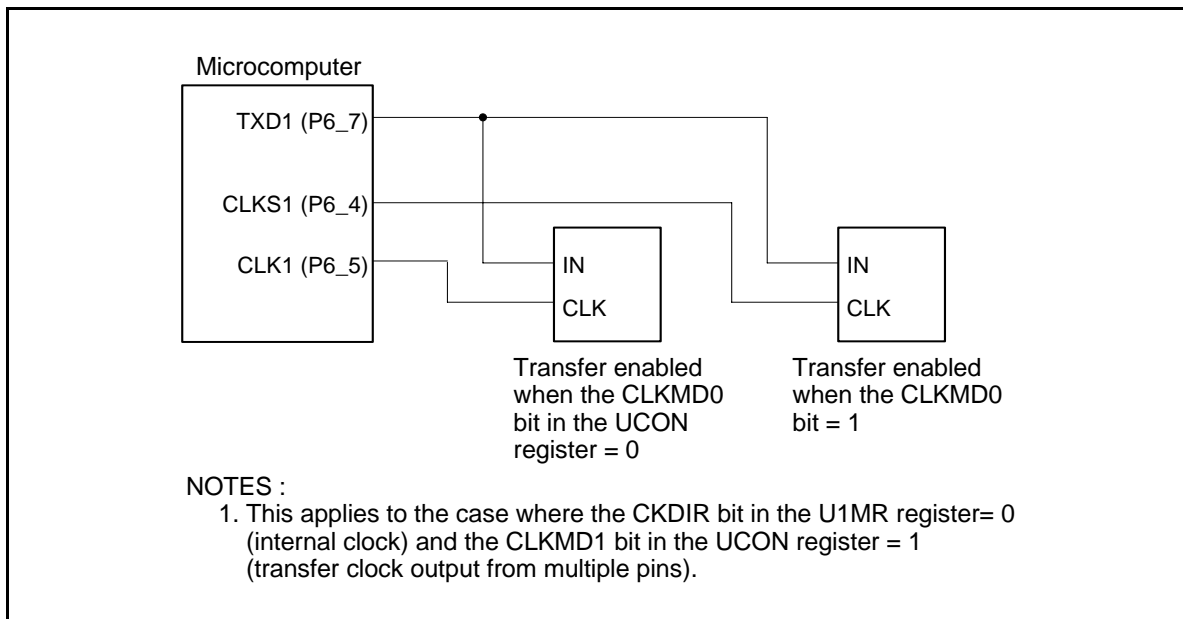


Figure 17.17 Transfer Clock Output from Multiple Pins

### 17.1.1.7 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Function

When the  $\overline{\text{CTS}}$  function is used transmit and receive operation start when “L” is applied to the  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  (i=0 to 2) pin. Transmit and receive operation begins when the  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is held “L”. If the “L” signal is switched to “H” during a transmit or receive operation, the operation stops before the next data.

When the  $\overline{\text{RTS}}$  function is used, the  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin outputs on “L” signal when the microcomputer is ready to receive. The output level becomes “H” on the first falling edge of the CLK<sub>i</sub> pin.

- CRD bit in UIC0 register = 1 (disable  $\overline{\text{CTS}}/\overline{\text{RTS}}$  of UART0)  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is programmable I/O function
- CRD bit = 0, CRS bit = 0 ( $\overline{\text{CTS}}$  function is selected)  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is  $\overline{\text{CTS}}$  function
- CRD bit = 0, CRS bit = 1 ( $\overline{\text{RTS}}$  function is selected)  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is  $\overline{\text{RTS}}$  function

### 17.1.1.8 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Separate Function (UART0)

This function separates  $\overline{\text{CTS}}_0/\overline{\text{RTS}}_0$ , outputs  $\overline{\text{RTS}}_0$  from the P6\_0 pin, and accepts as input the  $\overline{\text{CTS}}_0$  from the P6\_4 pin. To use this function, set the register bits as shown below.

- CRD bit in U0C0 register = 0 (enable  $\overline{\text{CTS}}/\overline{\text{RTS}}$  of UART0)
- CRS bit in U0C0 register = 1 (output  $\overline{\text{RTS}}$  of UART0)
- CRD bit in U1C0 register = 0 (enable  $\overline{\text{CTS}}/\overline{\text{RTS}}$  of UART1)
- CRS bit in U1C0 register = 0 (input  $\overline{\text{CTS}}$  of UART1)
- RCSP bit in UCON register = 1 (inputs  $\overline{\text{CTS}}_0$  from the P6\_4 pin)
- CLKMD1 bit in 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 separate function cannot be used.

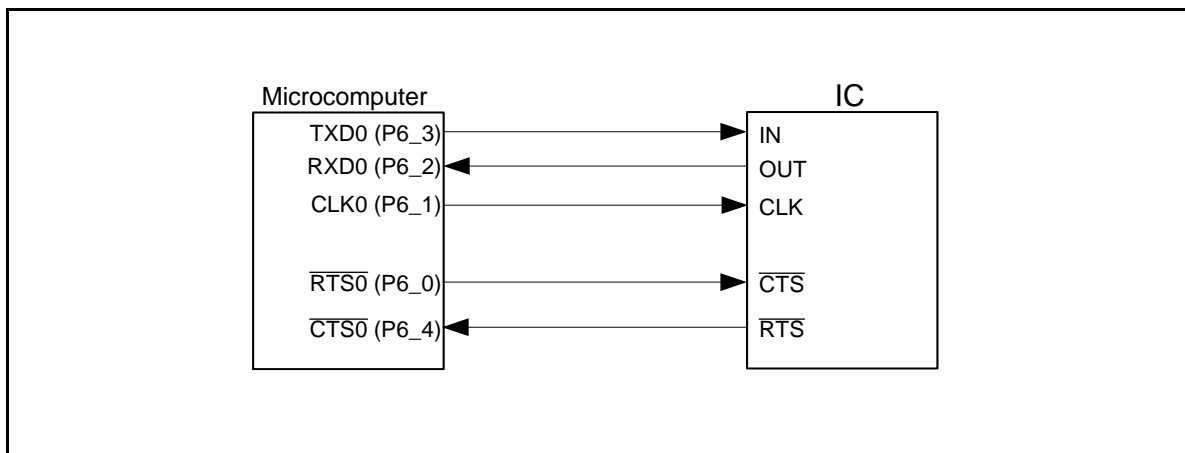


Figure 17.18  $\overline{\text{CTS}}/\overline{\text{RTS}}$  Separate Function

### 17.1.2 Clock Asynchronous Serial I/O (UART) Mode

The UART mode allows transmitting and receiving data after setting the desired bit rate and transfer data format. Table 17.5 lists the UART Mode Specifications.

**Table 17.5 UART Mode Specifications**

Item	Specification
Transfer Data Format	<ul style="list-style-type: none"> <li>Character bit (transfer data): Selectable from 7, 8 or 9 bits</li> <li>Start bit: 1 bit</li> <li>Parity bit: Selectable from odd, even, or none</li> <li>Stop bit: Selectable from 1 or 2 bits</li> </ul>
Transfer Clock	<ul style="list-style-type: none"> <li>CKDIR bit in the UiMR(i=0 to 2) register = 0 (internal clock) : <math>f_j / (16(n+1))</math>  <math>f_j = f1SIO, f2SIO, f8SIO, f32SIO</math> n: Setting value of UiBRG register 00h to FFh</li> <li>CKDIR bit = 1 (external clock) : <math>fEXT / (16(n+1))</math></li> <li>fEXT: Input from CLKi pin n: Setting value of UiBRG register 00h to FFh</li> </ul>
Transmission, Reception Control	Selectable from $\overline{CTS}$ function, $\overline{RTS}$ function or $\overline{CTS}/\overline{RTS}$ function disable
Transmission Start Condition	Before transmission can start, meet the following requirements <ul style="list-style-type: none"> <li>The TE bit in the UiC1 register = 1 (transmission enabled)</li> <li>The TI bit in the UiC1 register = 0 (data present in UiTB register)</li> <li>If <math>\overline{CTS}</math> function is selected, input on the <math>\overline{CTS}_i</math> pin = L</li> </ul>
Reception Start Condition	Before reception can start, meet the following requirements <ul style="list-style-type: none"> <li>The RE bit in the UiC1 register = 1 (reception enabled)</li> <li>Start bit detection</li> </ul>
Interrupt Request Generation Timing	For transmission, one of the following conditions can be selected <ul style="list-style-type: none"> <li>The UiIRS bit <sup>(2)</sup> = 0 (transmit buffer empty): when transferring data from the UiTB register to the UARTi transmit register (at start of transmission)</li> <li>The UiIRS bit = 1 (transfer completed): when the serial interface finished sending data from the UARTi transmit register</li> </ul> For reception <ul style="list-style-type: none"> <li>When transferring data from the UARTi receive register to the UiRB register (at completion of reception)</li> </ul>
Error Detection	<ul style="list-style-type: none"> <li>Overrun error <sup>(1)</sup> This error occurs if the serial interface started receiving the next data before reading the UiRB register and received the bit one before the last stop bit of the next data</li> <li>Framing error <sup>(3)</sup> This error occurs when the number of stop bits set is not detected</li> <li>Parity error <sup>(3)</sup> This error occurs when if parity is enabled, the number of "1" in parity and character bits does not match the number of "1" set</li> <li>Error sum flag This flag is set to "1" when any of the overrun, framing or parity errors occur</li> </ul>
Select Function	<ul style="list-style-type: none"> <li>LSB first, MSB first selection Whether to start sending/receiving data beginning with bit 0 or beginning with bit 7 can be selected</li> <li>Serial data logic switch This function reverses the logic of the transmit/receive data. The start and stop bits are not reversed.</li> <li>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 is reversed.</li> <li>Separate <math>\overline{CTS}/\overline{RTS}</math> pins (UART0) <math>\overline{CTS}_0</math> and <math>\overline{RTS}_0</math> are input/output from separate pins</li> </ul>

**NOTES:**

1. If an overrun error occurs, the receive data of UiRB register will be indeterminate. The IR bit in the SiRIC register does not change.
2. The U0IRS and U1IRS bits are bits 0 and 1 in the UCON register. The U2IRS bit is bit 4 in the U2C1 register.
3. The timing at which the framing error flag and the parity error flag are set is detected when data is transferred from the UARTi receive register to the UiRB register.

**Table 17.6 Registers to Be Used and Settings in UART Mode**

Register	Bit	Function
UiTB	0 to 8	Set transmission data <sup>(1)</sup>
UiRB	0 to 8	Reception data can be read <sup>(1)</sup>
	OER, FER, PER, SUM	Error flag
UiBRG	0 to 7	Set a bit rate
UiMR	SMD2 to SMD0	Set these bits to "100b" when transfer data is 7 bits long
		Set these bits to "101b" when transfer data is 8 bits long
		Set these bits to "110b" when transfer data is 9 bits long
	CKDIR	Select the internal clock or external clock
	STPS	Select the stop bit
	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	Select $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ 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 <sup>(3)</sup>
	CKPOL	Set to "0"
	UFORM	LSB first or MSB first can be selected when transfer data is 8 bits long. Set this bit to "0" when transfer data is 7 or 9 bits long.
UiC1	TE	Set this bit to "1" to enable transmission
	TI	Transmit buffer empty flag
	RE	Set this bit to "1" to enable reception
	RI	Reception complete flag
	U2IRS <sup>(2)</sup>	Select the source of UART2 transmit interrupt
	U2RRM <sup>(2)</sup>	Set to "0"
	UiLCH	Set this bit 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 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	Invalid because CLKMD1 = 0
	CLKMD1	Set to "0"
	RCSP	Set this bit to "1" to accept as input $\overline{\text{CTS0}}$ signal of UART0 from the P6_4 pin
	7	Set to "0"

**NOTES:**

- The bits used for transmit/receive data are as follows: Bit 0 to bit 6 when transfer data is 7 bits long; bit 0 to bit 7 when transfer data is 8 bits long; bit 0 to bit 8 when transfer data is 9 bits long.
- Set the bit 4 to bit 5 in the U0C1 and U1C1 registers to "0". The U0IRS, U1IRS, U0RRM and U1RRM bits are included in the UCON register.
- TXD2 pin is N channel open-drain output. Set the NCH bit in the U2C0 register to "0".

i=0 to 2

Table 17.7 lists the functions of the input/output pins during UART mode. Table 17.8 lists the P6\_4 Pin Functions. Note that for a period from when the UARTi operating mode is selected to when transfer starts, the TXDi pin outputs an “H” (If the N-channel open-drain output is selected, this pin is in a high-impedance state).

**Table 17.7 I/O Pin Functions**

Pin Name	Function	Method of Selection
TXDi (i = 0 to 2) (P6_3, P6_7, P7_0)	Serial Data Output	(“H” outputs when performing reception only)
RXDi (P6_2, P6_6, P7_1)	Serial Data Input	PD6_2 bit and PD6_6 bit in the PD6 register = 0, PD7_1 bit in the PD7 register = 0 (Can be used as an input port when performing transmission only)
CLKi (P6_1, P6_5, P7_2)	Input/Output Port	CKDIR bit in the UiMR register = 0
	Transfer Clock Input	CKDIR bit = 1 PD6_1 bit and PD6_5 bit in the PD6 register = 0, PD7_2 bit in the PD7 register = 0
CTS $\overline{\text{i}}$ /RTS $\overline{\text{i}}$ (P6_0, P6_4, P7_3)	CTS Input	CRD bit in the UiC0 register = 0 CRS bit in the UiC0 register = 0 PD6_0 bit and PD6_4 bit in the PD6 register = 0, PD7_3 bit in the PD7 register = 0
	RTS Output	CRD bit = 0 CRS bit = 1
	Input/Output Port	CRD bit = 1

**Table 17.8 P6\_4 Pin Functions**

Pin Function	Bit Set Value				
	U1C0 Register		U0CON Register		PD6 Register
	CRD	CRS	RCSP	CLKMD1	PD6_4
P6_4	1	–	0	0	Input: 0, Output: 1
CTS1	0	0	0	0	0
RTS1	0	1	0	0	–
CTS0 <sup>(1)</sup>	0	0	1	0	0

– : “0” or “1”

**NOTES:**

- In addition to this, set the CRD bit in the U0C0 register to “0” (CTS0/RTS0 enabled) and the CRS bit in the U0C0 register to “1” (RTS0 selected).

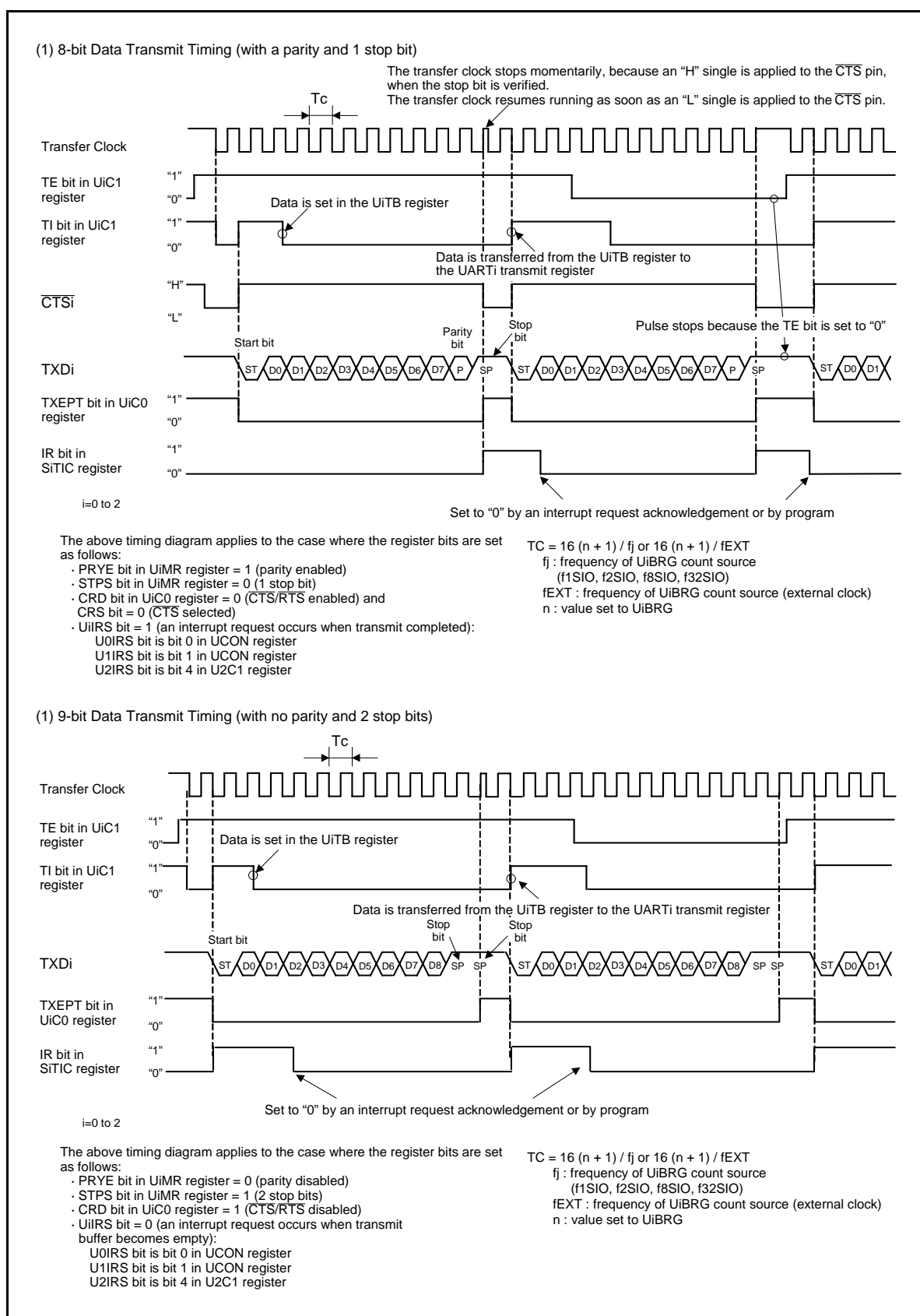


Figure 17.19 Transmit Operation

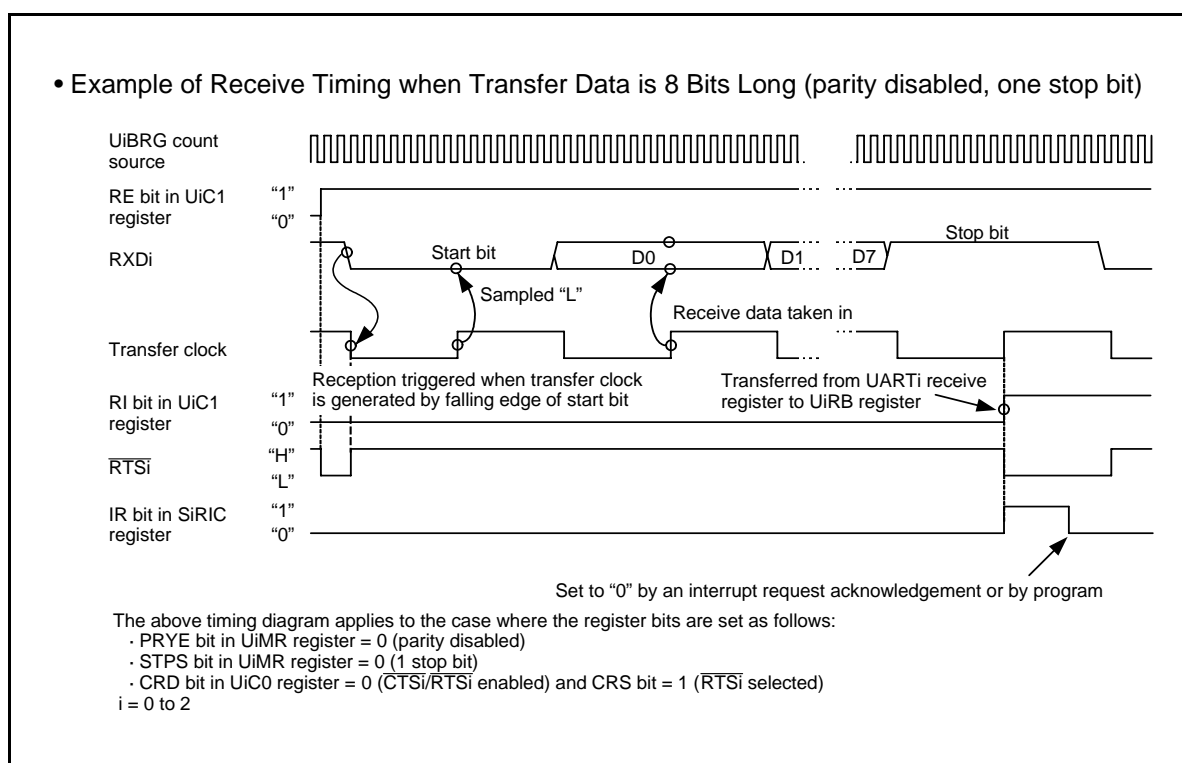


Figure 17.20 Receive Operation

### 17.1.2.1 Bit Rate

In UART mode, the frequency set by the UiBRG register ( $i=0$  to  $2$ ) divided by 16 become the bit rates. Table 17.9 lists Example of Bit Rates and Settings.

Table 17.9 Example of Bit Rates and Settings

Bit Rate (bps)	Count Source of UiBRG	Peripheral Function Clock : 16MHz		Peripheral Function Clock : 24MHz	
		Set Value of UiBRG : n	Bit Rate (bps)	Set value of UiBRG : n	Bit Rate (bps)
1200	f8	103 (67h)	1202	155 (9Bh)	1202
2400	f8	51 (33h)	2404	77 (4Dh)	2404
4800	f8	25 (19h)	4808	38 (26h)	4808
9600	f1	103 (67h)	9615	155 (9Bh)	9615
14400	f1	68 (44h)	14493	103 (67h)	14423
19200	f1	51 (33h)	19231	77 (4Dh)	19231
28800	f1	34 (22h)	28571	51 (33h)	28846
31250	f1	31 (1Fh)	31250	47 (2Fh)	31250
38400	f1	25 (19h)	38462	38 (26h)	38462
51200	f1	19 (13h)	50000	28 (1Ch)	51724



### 17.1.2.2 Counter Measure for Communication Error Occurs

If a communication error occurs while transmitting or receiving in UART mode, follow the procedures below.

- Resetting the UiRB register (i=0 to 2)
  - (1) Set the RE bit in the UiC1 register to “0” (reception disabled)
  - (2) Set the RE bit in the UiC1 register to “1” (reception enabled)
- Resetting the UiTB register (i=0 to 2)
  - (1) Set the SMD2 to SMD0 bits in the UiMR register “000b” (Serial interface disabled)
  - (2) Set the SMD2 to SMD0 bits in the UiMR register “001b”, “101b”, “110b”.
  - (3) “1” is written to RE bit in the UiC1 register (transmission enabled), regardless of the TE bit in the UiCi register

### 17.1.2.3 LSB First/MSB First Select Function

As shown in Figure 17.21, use the UFORM bit in the UiC0 register to select the transfer format. This function is valid when transfer data is 8 bits long.

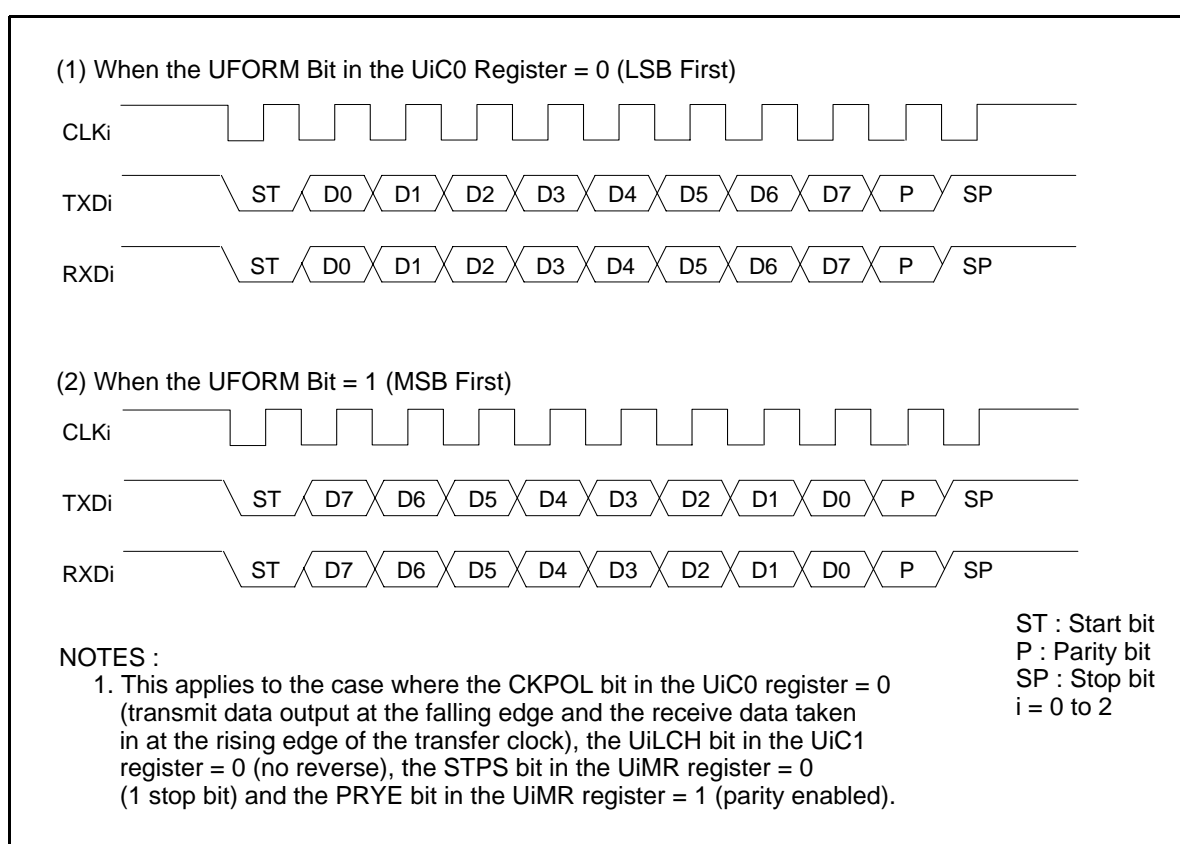


Figure 17.21 Transfer Format

### 17.1.2.4 Serial Data Logic Switching Function

The 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 17.22 shows Serial Data Logic Switching.

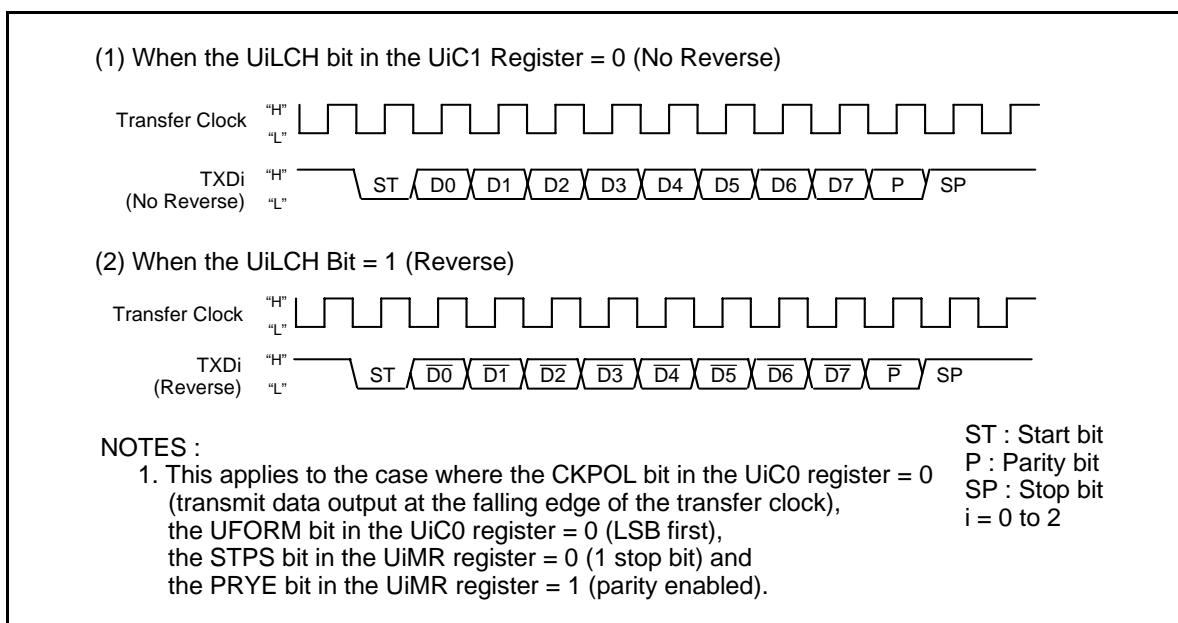


Figure 17.22 Serial Data Logic Switching

### 17.1.2.5 TXD and RXD I/O Polarity Inverse Function

This function inverses the polarities of the TXDi pin output and RXDi pin input. The logic levels of all input/output data (including the start, stop and parity bits) are inverted. Figure 17.23 shows the TXD and RXD I/O Polarity Inverse.

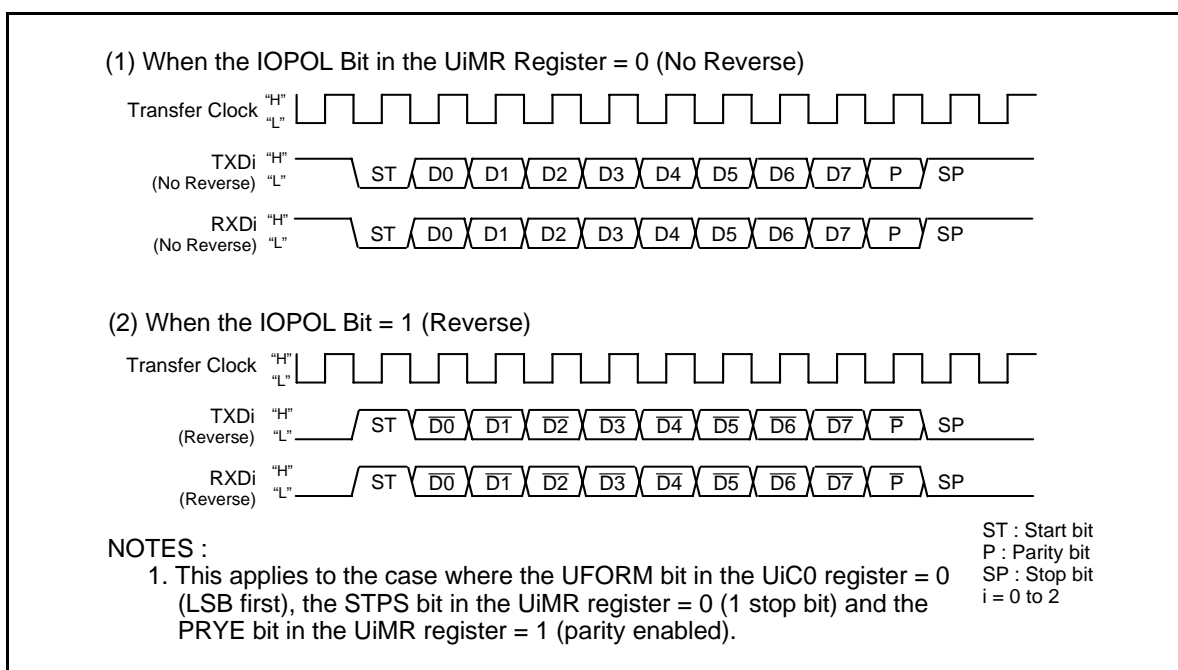


Figure 17.23 TXD and RXD I/O Polarity Inverse

### 17.1.2.6 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Function

When the  $\overline{\text{CTS}}$  function is used transmit operation start when “L” is applied to the  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  ( $i=0$  to 2) pin. Transmit operation begins when the  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is held “L”. If the “L” signal is switched to “H” during a transmit operation, the operation stops before the next data.

When the  $\overline{\text{RTS}}$  function is used, the  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin outputs on “L” signal when the microcomputer is ready to receive. The output level becomes “H” on the first falling edge of the CLK<sub>i</sub> pin.

- CRD bit in UIC0 register = 1 (disable  $\overline{\text{CTS}}/\overline{\text{RTS}}$  function of UART0)  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is programmable I/O function
- CRD bit = 0, CRS bit = 0 ( $\overline{\text{CTS}}$  function is selected)  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is  $\overline{\text{CTS}}$  function
- CRD bit = 0, CRS bit = 1 ( $\overline{\text{RTS}}$  function is selected)  $\overline{\text{CTS}}_i/\overline{\text{RTS}}_i$  pin is  $\overline{\text{RTS}}$  function

### 17.1.2.7 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Separate Function (UART0)

This function separates  $\overline{\text{CTS}}_0/\overline{\text{RTS}}_0$ , outputs  $\overline{\text{RTS}}_0$  from the P6\_0 pin, and accepts as input the  $\overline{\text{CTS}}_0$  from the P6\_4 pin. To use this function, set the register bits as shown below.

- CRD bit in U0C0 register = 0 (enable  $\overline{\text{CTS}}/\overline{\text{RTS}}$  of UART0)
- CRS bit in U0C0 register = 1 (output  $\overline{\text{RTS}}$  of UART0)
- CRD bit in U1C0 register = 0 (enable  $\overline{\text{CTS}}/\overline{\text{RTS}}$  of UART1)
- CRS bit in U1C0 register = 0 (input  $\overline{\text{CTS}}$  of UART1)
- RCSP bit in UCON register = 1 (inputs  $\overline{\text{CTS}}_0$  from the P6\_4 pin)
- CLKMD1 bit in 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 separate function cannot be used.

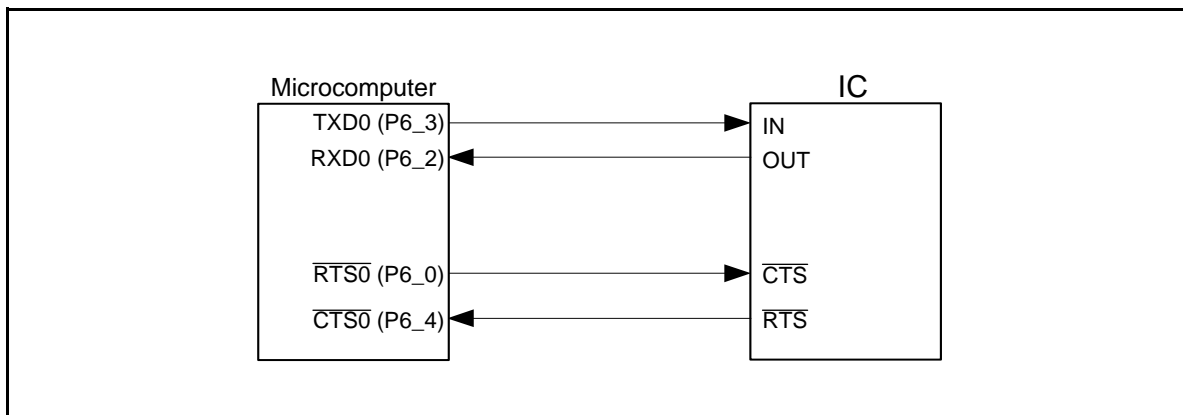


Figure 17.24  $\overline{\text{CTS}}/\overline{\text{RTS}}$  Separate Function

### 17.1.3 Special Mode 1 (I<sup>2</sup>C mode)

I<sup>2</sup>C mode is provided for use as a simplified I<sup>2</sup>C interface compatible mode. Table 17.10 lists the specifications of the I<sup>2</sup>C mode. Table 17.11 to 17.12 lists the registers used in the I<sup>2</sup>C mode and the register values set. Table 13.13 lists the I<sup>2</sup>C Mode Functions. Figure 17.25 shows the block diagram for I<sup>2</sup>C mode. Figure 17.26 shows Transfer to UiRB Register and Interrupt Timing.

As shown in Table 17.13, the microcomputer is placed in I<sup>2</sup>C mode by setting the SMD2 to SMD0 bits to “010b” and the IICM bit to “1”. Because SDAi transmit output has a delay circuit attached, SDAi output does not change state until SCLi goes low and remains stably low.

**Table 17.10 I<sup>2</sup>C Mode Specifications**

Item	Specification
Transfer Data Format	Transfer data length: 8 bits
Transfer Clock	<ul style="list-style-type: none"> <li>During master           <ul style="list-style-type: none"> <li>CKDIR bit in the UiMR (i=0 to 2) register = 0 (internal clock) : <math>f_j / (2(n+1))</math>  <math>f_j = f1SIO, f2SIO, f8SIO, f32SIO</math>    n: Setting value of UiBRG register    00h to FFh</li> </ul> </li> <li>During slave           <ul style="list-style-type: none"> <li>CKDIR bit = 1 (external clock) : Input from SCLi pin</li> </ul> </li> </ul>
Transmission Start Condition	Before transmission can start, met the following requirements <sup>(1)</sup> <ul style="list-style-type: none"> <li>The TE bit in the UiC1 register= 1 (transmission enabled)</li> <li>The TI bit in the UiC1 register = 0 (data present in UiTB register)</li> </ul>
Reception Start Condition	Before reception can start, met the following requirements <sup>(1)</sup> <ul style="list-style-type: none"> <li>The RE bit in UiC1 register= 1 (reception enabled)</li> <li>The TE bit in UiC1 register= 1 (transmission enabled)</li> <li>The TI bit in UiC1 register= 0 (data present in the UiTB register)</li> </ul>
Interrupt Request Generation Timing	When start or stop condition is detected, acknowledge undetected, and acknowledge detected
Error Detection	Overrun error <sup>(2)</sup> This error occurs if the serial interface started receiving the next data before reading the UiRB register and received the 8th bit of the next data
Select Function	<ul style="list-style-type: none"> <li>Arbitration lost Timing at which the ABT bit in the UiRB register is updated can be selected</li> <li>SDAi digital delay No digital delay or a delay of 2 to 8 UiBRG count source clock cycles selectable</li> <li>Clock phase setting With or without clock delay selectable</li> </ul>

**NOTES:**

1. When an external clock is selected, the conditions must be met while the external clock is in the high state.
2. If an overrun error occurs, the value of UiRB register will be indeterminate. The IR bit in the SiRIC register does not change.

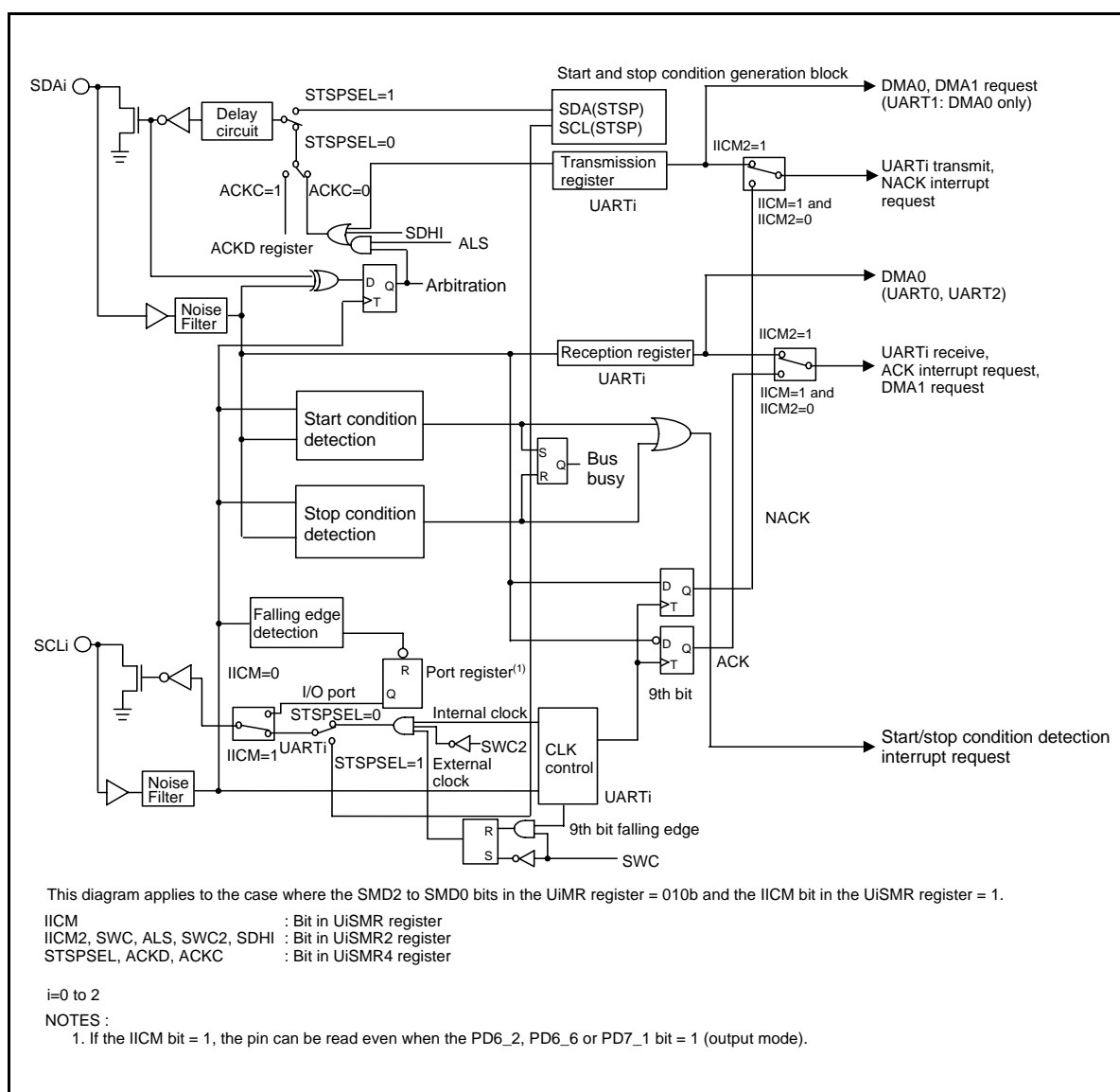


Figure 17.25 I²C Mode Block Diagram

**Table 17.11 Registers to Be Used and Settings in I<sup>2</sup>C Mode (1)**

Register	Bit	Function	
		Master	Slave
UiTB <sup>(3)</sup>	0 to 7	Set transmission data	Set transmission data
UiRB <sup>(3)</sup>	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	Invalid
	OER	Overrun error flag	Overrun error flag
UiBRG	0 to 7	Set a bit rate	Invalid
UiMR <sup>(3)</sup>	SMD2 to SMD0	Set to "010b"	Set to "010b"
	CKDIR	Set to "0"	Set to "1"
	IOPOL	Set to "0"	Set to "0"
UiC0	CLK1, CLK0	Select the count source for the UiBRG register	Invalid
	CRS	Invalid because CRD = 1	Invalid because CRD = 1
	TXEPT	Transmit buffer empty flag	Transmit buffer empty flag
	CRD <sup>(4)</sup>	Set to "1"	Set to "1"
	NCH	Set to "1" <sup>(2)</sup>	Set to "1" <sup>(2)</sup>
	CKPOL	Set to "0"	Set to "0"
	UFORM	Set to "1"	Set to "1"
UiC1	TE	Set this bit to "1" to enable transmission	Set this bit to "1" to enable transmission
	TI	Transmit buffer empty flag	Transmit buffer empty flag
	RE	Set this bit to "1" to enable reception	Set this bit to "1" to enable reception
	RI	Reception complete flag	Reception complete flag
	U2IRS <sup>(1)</sup>	Invalid	Invalid
	U2RRM <sup>(1)</sup> , UiLCH, UiERE	Set to "0"	Set to "0"
UiSMR	IICM	Set to "1"	Set to "1"
	ABC	Select the timing at which arbitration-lost is detected	Invalid
	BBS	Bus busy flag	Bus busy flag
	3 to 7	Set to "0"	Set to "0"
UiSMR2	IICM2	See <b>Table 17.13 I<sup>2</sup>C Mode Functions</b>	See <b>Table 17.13 I<sup>2</sup>C Mode Functions</b>
	CSC	Set this bit to "1" to enable clock synchronization	Set to "0"
	SWC	Set this bit to "1" to have SCLi output fixed to "L" at the falling edge of the 9th bit of clock	Set this bit to "1" to have SCLi output fixed to "L" at the falling edge of the 9th bit of clock
	ALS	Set this bit to "1" to have SDAi output stopped when arbitration-lost is detected	Set to "0"
	STAC	Set to "0"	Set this bit to "1" to initialize UAR <sub>T</sub> i at start condition detection
	SWC2	Set this bit to "1" to have SCLi output forcibly pulled low	Set this bit to "1" to have SCLi output forcibly pulled low
	SDHI	Set this bit to "1" to disable SDAi output	Set this bit to "1" to disable SDAi output
	7	Set to "0"	Set to "0"

**NOTES:**

1. Set the bit 4 and bit 5 in the U0C1 and U1C1 register to "0". The U0IRS, U1IRS, U0RRM and U1RRM bits are in the UCON register.
2. TXD2 pin is N channel open-drain output. No NCH bit in the U2C0 register is assigned. When write, set to "0".
3. Not all register bits are described above. Set those bits to "0" when writing to the registers in I<sup>2</sup>C mode.
4. When using UAR<sub>T</sub>1 in I<sup>2</sup>C mode and enabling the CTS/RTS separate function of UAR<sub>T</sub>0, set the CRD bit in the U1C0 register to "0" (CTS/RTS enable) and the CRS bit to "0" (CTS input).

i=0 to 2

**Table 17.12 Registers to Be Used and Settings in I<sup>2</sup>C Mode (2)**

Register	Bit	Function	
		Master	Slave
UiSMR3	0, 2, 4 and NODC	Set to "0"	Set to "0"
	CKPH	See <b>Table 17.13 I<sup>2</sup>C Mode Functions</b>	See <b>Table 17.13 I<sup>2</sup>C Mode Functions</b>
	DL2 to DL0	Set the amount of SDAi digital delay	Set the amount of SDAi digital delay
UiSMR4	STAREQ	Set this bit to "1" to generate start condition	Set to "0"
	RSTAREQ	Set this bit to "1" to generate restart condition	Set to "0"
	STPREQ	Set this bit to "1" to generate stop condition	Set to "0"
	STSPSEL	Set this bit to "1" to output each condition	Set to "0"
	ACKD	Select ACK or NACK	Select ACK or NACK
	ACKC	Set this bit to "1" to output ACK data	Set this bit to "1" to output ACK data
	SCLHI	Set this bit to "1" to have SCLi output stopped when stop condition is detected	Set to "0"
	SWC9	Set to "0"	Set this bit to "1" to set the SCLi to "L" hold at the falling edge of the 9th bit of clock
IFSR2A	IFSR26, ISFR27	Set to "1"	Set to "1"
UCON	U0IRS, U1IRS	Invalid	Invalid
	2 to 7	Set to "0"	Set to "0"

i=0 to 2

**Table 17.13 I<sup>2</sup>C Mode Functions**

Function	Clock Synchronous Serial I/O Mode (SMD2 to SMD0 = 001b, IICM = 0)	I <sup>2</sup> C Mode (SMD2 to SMD0 = 010b, IICM = 1)			
		IICM2 = 0 (NACK/ACK interrupt)		IICM2 = 1 (UART transmit/receive interrupt)	
		CKPH = 0 (No clock delay)	CKPH = 1 (Clock delay)	CKPH = 0 (No clock delay)	CKPH = 1 (Clock delay)
Factor of Interrupt Number 6, 7 and 10 (1, 5, 7)	–	Start condition detection or stop condition detection (See <b>Table 17.14 STSPSEL Bit Functions</b> )			
Factor of Interrupt Number 15, 17 and 19 (1, 6)	UARTi transmission Transmission started or completed (selected by UiIRS)	No acknowledgment detection (NACK) Rising edge of SCLi 9th bit		UARTi transmission Rising edge of SCLi 9th bit	UARTi transmission Falling edge of SCLi next to the 9th bit
Factor of Interrupt Number 16, 18 and 20 (1, 6)	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 the UART Reception Shift Register to the UiRB Register	CKPOL = 0 (rising edge) CKPOL = 1 (falling edge)	Rising edge of SCLi 9th bit		Falling edge of SCLi 9th bit	Falling and rising edges of SCLi 9th bit
UARTi Transmission Output Delay	Not delayed	Delayed			
Functions of P6_3, P6_7 and P7_0 Pins	TXDi output	SDAi input/output			
Functions of P6_2, P6_6 and P7_1 Pins	RXDi input	SCLi input/output			
Functions of P6_1, P6_5 and P7_2 Pins	CLKi input or output selected	– (Cannot be used in I <sup>2</sup> C mode)			
Noise Filter Width	15ns	200ns			
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 (H) CKPOL = 1 (L)	The value set in the port register before setting I <sup>2</sup> C mode (2)			
Initial and End Values of SCLi	–	H	L	H	L
DMA1 Factor (6)	UARTi reception	Acknowledgment detection (ACK)		UARTi reception Falling edge of SCLi 9th bit	
Store Received Data	1st to 8th bits of the received data are stored into bits 7 to 0 in the UiRB register	1st to 8th bits of the received data are stored into bits 7 to 0 in the UiRB register		1st to 7th bits of the received data are stored into bits 6 to 0 in the UiRB register. 8th bit is stored into bit 8 in the UiRB register.	
					1st to 8th bits are stored into bits 7 to 0 in the UiRB register (3)
Read Received Data	The UiRB register status is read				Bits 6 to 0 in the UiRB register (4) are read as bits 7 to 1. Bit 8 in the UiRB register is read as bit 0.

**NOTES:**

1. If the source or factor of any interrupt is changed, the IR bit in the interrupt control register for the changed interrupt may inadvertently be set to "1" (interrupt requested). (Refer to **24.7 Interrupt**)  
If one of the bits shown below is changed, the interrupt source, the interrupt timing, etc. change. Therefore, always be sure to clear the IR bit to "0" (interrupt not requested) after changing those bits.  
SMD2 to SMD0 bits in the UiMR register, IICM bit in the UiSMR register, IICM2 bit in the UiSMR2 register, CKPH bit in the UiSMR3 register
2. Set the initial value of SDAi output while the SMD2 to SMD0 bits in the UiMR register = 000b (serial interface disabled).
3. Second data transfer to UiRB register (Rising edge of SCLi 9th bit)
4. First data transfer to UiRB register (Falling edge of SCLi 9th bit)
5. See **Figure 17.28 STSPSEL Bit Functions**.
6. See **Figure 17.26 Transfer to UiRB Register and Interrupt Timing**.
7. When using UART0, be sure to set the IFSR26 bit in the IFSR2A register to "1" (factor of interrupt: UART0 bus collision).  
When using UART1, be sure to set the IFSR27 bit to "1" (factor of interrupt: UART1 bus collision).

i = 0 to 2



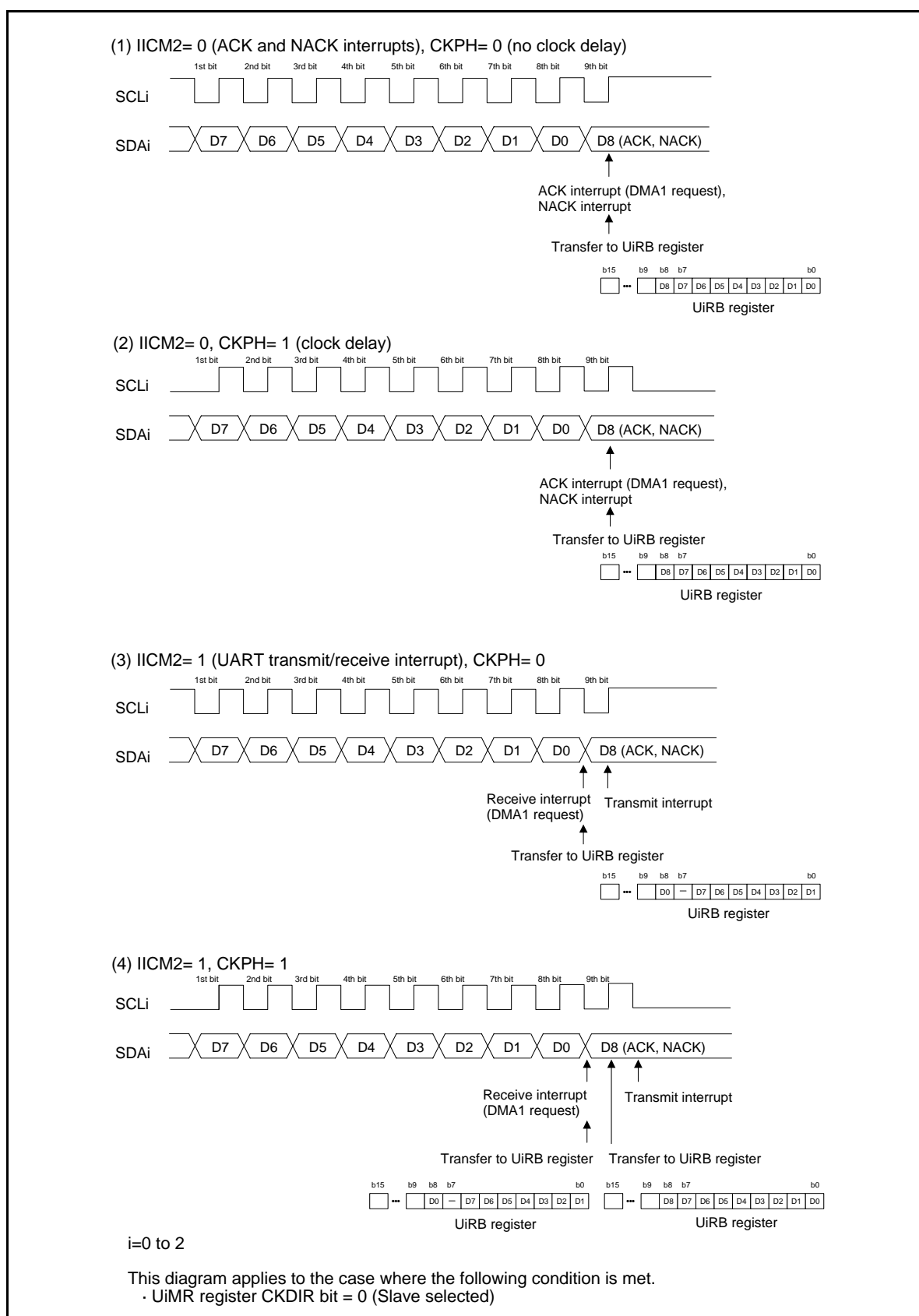


Figure 17.26 Transfer to UiRB Register and Interrupt Timing

### 17.1.3.1 Detection of Start and Stop Condition

Whether a start or a stop condition has been detected is determined.

A start condition-detected interrupt request is generated when the SDA<sub>i</sub> pin changes state from high to low while the SCL<sub>i</sub> pin is in the high state. A stop condition-detected interrupt request is generated when the SDA<sub>i</sub> pin changes state from low to high while the SCL<sub>i</sub> pin is in the high state.

Because the start and stop condition-detected interrupts share the interrupt control register and vector, check the BBS bit in the UiSMR register to determine which interrupt source is requesting the interrupt.

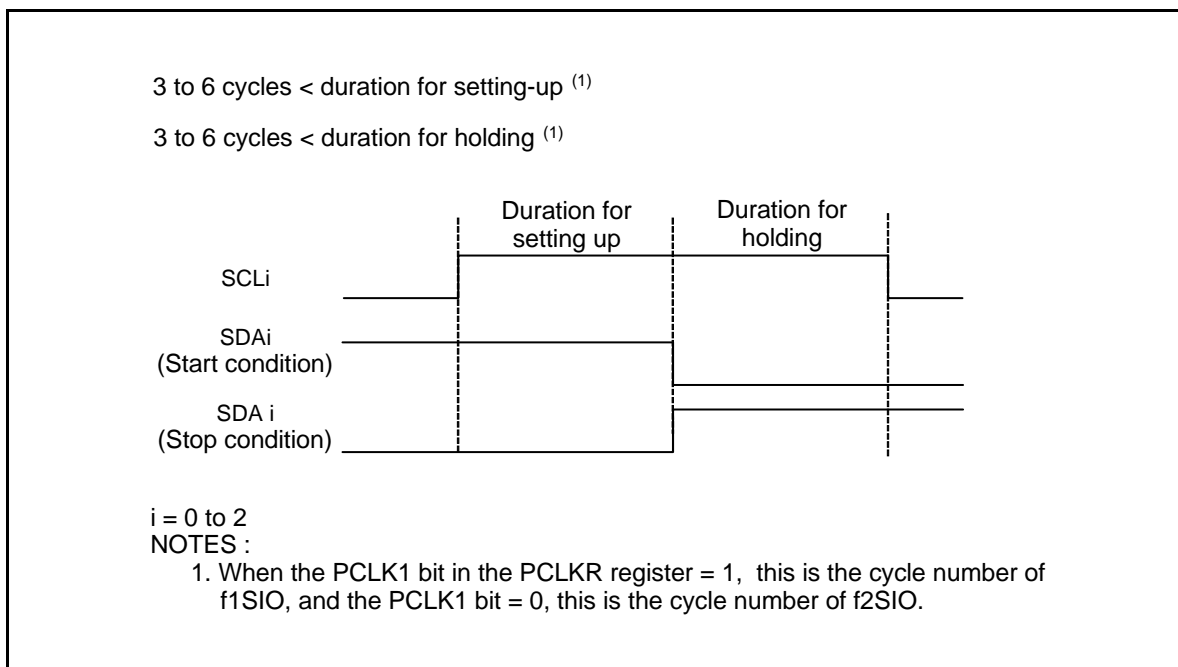


Figure 17.27 Detection of Start and Stop Condition

### 17.1.3.2 Output of Start and Stop Condition

A start condition is generated by setting the STAREQ bit in the UiSMR4 register ( $i = 0 \text{ to } 2$ ) 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).

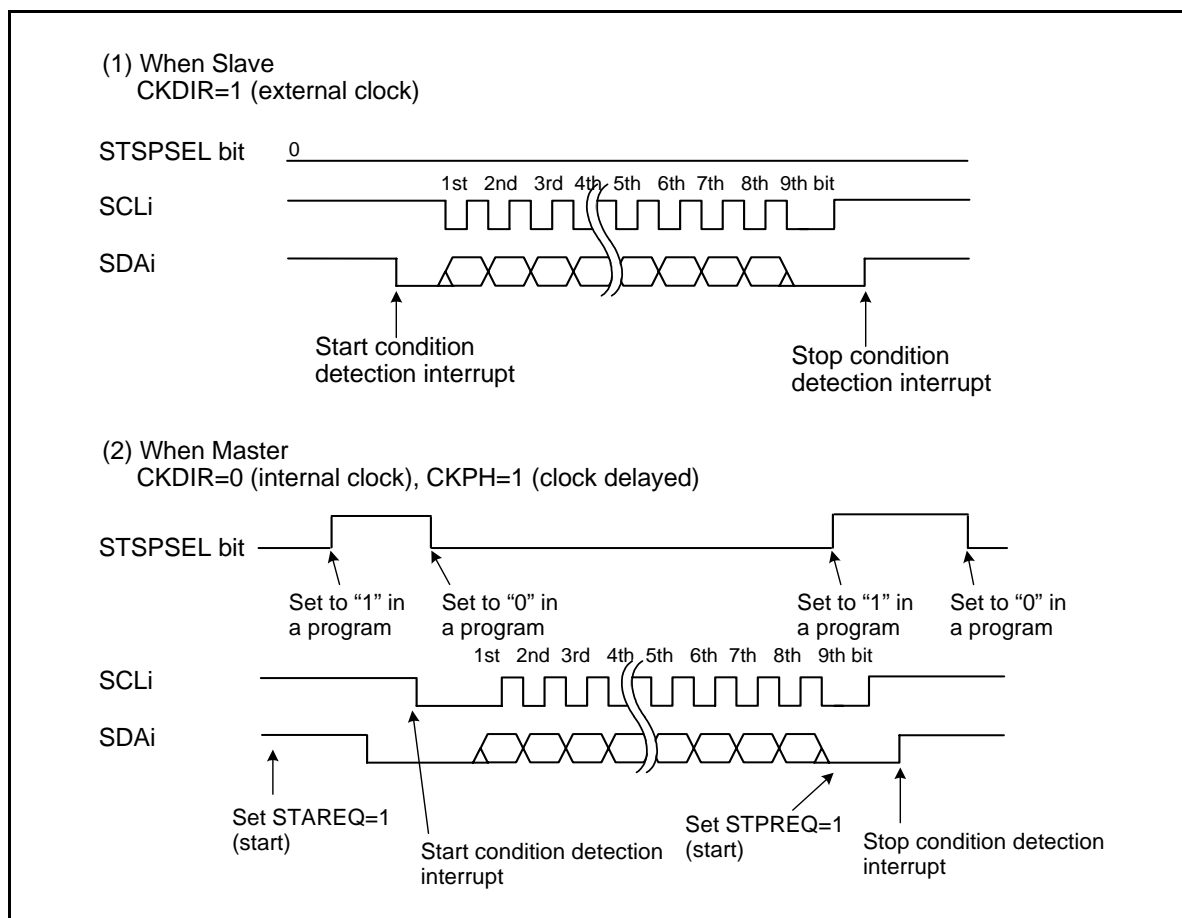
The output procedure is described below.

- (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 function of the STSPSEL bit is shown in Table 17.14 and Figure 17.28.

**Table 17.14 STSPSEL Bit Functions**

Function	STSPSEL = 0	STSPSEL = 1
Output of SCLi and SDAi Pins	Output of transfer 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 the STAREQ, RSTAREQ and STPREQ bit
Start/Stop Condition Interrupt Request Generation Timing	Start/stop condition detection	Finish generating start/stop condition

**Figure 17.28 STSPSEL Bit Functions**

### 17.1.3.3 Arbitration

Unmatching of the transmit data and SDAi pin input data is checked synchronously with the rising edge of SCLi. Use the ABC bit in the UiSMR register to select the timing at which the ABT bit in the UiRB register is updated. If the ABC bit = 0 (updated bitwise), the ABT bit is set to "1" at the same time unmatching is detected during check, and is cleared to "0" when not detected. In cases when the ABC bit is set to "1", if unmatching is detected even once during check, the ABT bit is set to "1" (unmatching detected) at the falling edge of the clock pulse of 9th bit. If the ABT bit needs to be updated byte-wise, clear the ABT bit to "0" (undetected) after detecting acknowledge in the first byte, before transferring the next byte.

Setting the ALS bit in the UiSMR2 register to "1" (SDA output stop enabled) factors arbitration-lost to occur, in which case the SDAi pin is placed in the high-impedance state at the same time the ABT bit is set to "1" (unmatching detected).

### 17.1.3.4 Transfer Clock

Data is transmitted/received using a transfer clock like the one shown in Figure 17.26 Transfer to UiRB Register and Interrupt Timing.

The CSC bit in the UiSMR2 register is used to synchronize the internally generated clock (internal SCLi) and an external clock supplied to the SCLi pin. In cases when the CSC bit is set to “1” (clock synchronization enabled), if a falling edge on the SCLi pin is detected while the internal SCLi is high, the internal SCLi goes low, at which time the value of the UiBRG register is reloaded with and starts counting in the low-level interval. If the internal SCLi changes state from low to high while the SCLi pin is low, counting stops, and when the SCLi pin goes high, counting restarts.

In this way, the UARTi transfer clock is comprised of the logical product of the internal SCLi and SCLi pin signal. The transfer clock works from a half period before the falling edge of the internal SCLi 1st bit to the rising edge of the 9th bit. To use this function, select an internal clock for the transfer clock.

The SWC bit in the UiSMR2 register allows to select whether the SCLi pin should be fixed to or freed from low-level output at the falling edge of the 9th clock pulse.

If the SCLHI bit in the UiSMR4 register is set to “1” (enabled), SCLi output is turned off (placed in the high-impedance state) when a stop condition is detected.

Setting the SWC2 bit in the UiSMR2 register = 1 (0 output) makes it possible to forcibly output a low-level signal from the SCLi pin even while sending or receiving data. Clearing the SWC2 bit to “0” (transfer clock) allows the transfer clock to be output from or supplied to the SCLi pin, instead of outputting a low-level signal.

If the SWC9 bit in the UiSMR4 register is set to “1” (SCL hold low enabled) when the CKPH bit in the UiSMR3 register = 1, the SCLi pin is fixed to low-level output at the falling edge of the clock pulse next to the 9th. Setting the SWC9 bit = 0 (SCL hold low disabled) frees the SCLi pin from low-level output.

### 17.1.3.5 SDA Output

The data written to the UiTB register bit 7 to bit 0 (D7 to D0) is sequentially output beginning with D7.

The 9th bit (D8) is ACK or NACK.

The initial value of SDAi transmit output can only be set when IICM = 1 (I<sup>2</sup>C mode) and the SMD2 to SMD0 bits in the UiMR register = 000b (Serial interface disabled).

The DL2 to DL0 bits in the UiSMR3 register allow to add no delays or a delay of 2 to 8 UiBRG count source clock cycles to SDAi output.

Setting the SDHI bit in the UiSMR2 register = 1 (SDA output disabled) forcibly places the SDAi pin in the high-impedance state. Do not write to the SDHI bit synchronously with the rising edge of the UARTi transfer clock. This is because the ABT bit may inadvertently be set to “1” (detected).

### 17.1.3.6 SDA Input

When the IICM2 bit = 0, the 1st to 8th bits (D7 to D0) of received data are stored in the UiRB register bit 7 to bit 0. The 9th bit (D8) is ACK or NACK.

When the IICM2 bit = 1, the 1st to 7th bits (D7 to D1) of received data are stored in the UiRB register bit 6 to bit 0 and the 8th bit (D0) is stored in the UiRB register bit 8. Even when the IICM2 bit = 1, providing the CKPH bit = 1, the same data as when the IICM2 bit = 0 can be read out by reading the UiRB register after the rising edge of the corresponding clock pulse of 9th bit.

### 17.1.3.7 ACK and NACK

If the STSPSEL bit in the UiSMR4 register is set to “0” (start and stop conditions not generated) and the ACKC bit in the UiSMR4 register is set to “1” (ACK data output), the value of the ACKD bit in the UiSMR4 register is output from the SDAi pin.

If the IICM2 bit = 0, a NACK interrupt request is generated if the SDAi pin remains high at the rising edge of the 9th bit of transmit clock pulse. An ACK interrupt request is generated if the SDAi pin is low at the rising edge of the 9th bit of transmit clock pulse.

If ACKi is selected for the factor of DMA1 request, a DMA transfer can be activated by detection of an acknowledge.

### 17.1.3.8 Initialization of Transmission/Reception

If a start condition is detected while the STAC bit = 1 (UARTi initialization enabled), the serial interface operates as described below.

- The transmit shift register is initialized, and the content of the UiTB register is transferred to the transmit shift register. In this way, the serial interface starts sending data synchronously with the next clock pulse 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 synchronously with the input clock.
- The receive shift register is initialized, and the serial interface starts receiving data synchronously with the next clock pulse applied.
- The SWC bit is set to “1” (SCL wait output enabled). Consequently, the SCLi pin is pulled low at the falling edge of the 9th clock pulse.

Note that when UARTi transmission/reception is started using this function, the TI does not change state. Note also that when using this function, the selected transfer clock should be an external clock.

### 17.1.4 Special Mode 2

Multiple slaves can be serially communicated from one master. Transfer clock polarity and phase are selectable. Table 17.15 lists the Special Mode 2 Specifications. Table 17.16 lists the Registers to Be Used and Settings in Special Mode 2. Figure 17.29 shows Serial Bus Communication Control Example (UART2).

**Table 17.15 Special Mode 2 Specifications**

Item	Specification
Transfer Data Format	Transfer data length: 8 bits
Transfer Clock	<ul style="list-style-type: none"> <li>Master mode CKDIR bit in UiMR(i=0 to 2) register = 0 (internal clock) : <math>f_j / (2(n+1))</math>  <math>f_j = f1SIO, f2SIO, f8SIO, f32SIO</math> n: Setting value of UiBRG register 00h to FFh</li> <li>Slave mode CKDIR bit = 1 (external clock selected) : Input from CLKi pin</li> </ul>
Transmit/Receive Control	Controlled by input/output ports
Transmission Start Condition	Before transmission can start, meet the following requirements <sup>(1)</sup> <ul style="list-style-type: none"> <li>The TE bit in UiC1 register = 1 (transmission enabled)</li> <li>The TI bit in UiC1 register = 0 (data present in UiTB register)</li> </ul>
Reception Start Condition	Before reception can start, meet the following requirements <sup>(1)</sup> <ul style="list-style-type: none"> <li>The RE bit in UiC1 register = 1 (reception enabled)</li> <li>The TE bit in UiC1 register = 1 (transmission enabled)</li> <li>The TI bit in UiC1 register = 0 (data present in the UiTB register)</li> </ul>
Interrupt Request Generation Timing	For transmission, one of the following conditions can be selected <ul style="list-style-type: none"> <li>The UiIRS bit in UiC1 register = 0 (transmit buffer empty): when transferring data from the UiTB register to the UARTi transmit register (at start of transmission)</li> <li>The UiIRS bit = 1 (transfer completed): when the serial interface finished sending data from the UARTi transmit register</li> </ul> For reception <ul style="list-style-type: none"> <li>When transferring data from the UARTi receive register to the UiRB register (at completion of reception)</li> </ul>
Error Detection	Overrun error <sup>(2)</sup> This error occurs if the serial interface started receiving the next data before reading the UiRB register and received the 7th bit of the next data
Select Function	Clock phase setting Selectable from four combinations of transfer clock polarities and phases

**NOTES:**

- When an external clock is selected, the conditions must be met while if the CKPOL bit in the UiC0 register = 0 (transmit data output at the falling edge and the receive data taken in at the rising edge of the transfer clock), the external clock is in the high state; if the CKPOL bit = 1 (transmit data output at the rising edge and the receive data taken in at the falling edge of the transfer clock), the external clock is in the low state.
- If an overrun error occurs, the value of UiRB register will be indeterminate. The IR bit in the SiRIC register does not change.

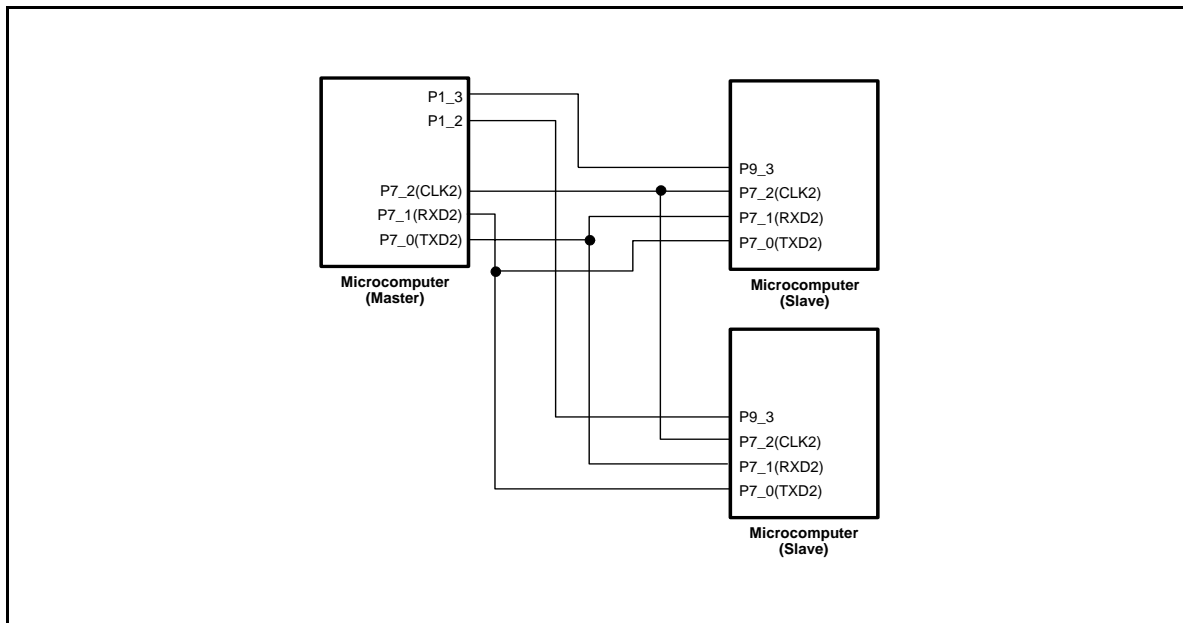


Figure 17.29 Serial Bus Communication Control Example (UART2)

**Table 17.16 Registers to Be Used and Settings in Special Mode 2**

Register	Bit	Function
UiTB <sup>(3)</sup>	0 to 7	Set transmission data
UiRB <sup>(3)</sup>	0 to 7	Reception data can be read
	OER	Overrun error flag
UiBRG	0 to 7	Set a bit rate
UiMR <sup>(3)</sup>	SMD2 to SMD0	Set to "001b"
	CKDIR	Set this bit to "0" for master mode or "1" for slave mode
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Select the count source for the UiBRG register
	CRS	Invalid because CRD = 1
	TXEPT	Transmit register empty flag
	CRD	Set to "1"
	NCH	Select TXDi pin output format <sup>(2)</sup>
	CKPOL	Clock phases can be set in combination with the CKPH bit in the UiSMR3 register
	UFORM	Set to "0"
UiC1	TE	Set this bit to "1" to enable transmission
	TI	Transmit buffer empty flag
	RE	Set this bit to "1" to enable reception
	RI	Reception complete flag
	U2IRS <sup>(1)</sup>	Select UART2 transmit interrupt factor
	U2RRM <sup>(1)</sup> , UiLCH, 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, U1IRS	Select UART0 and UART1 transmit interrupt factor
	U0RRM, U1RRM	Set to "0"
	CLKMD0	Invalid because CLKMD1 = 0
	CLKMD1, RCSP, 7	Set to "0"

**NOTES:**

1. Set the bit 4 and bit 5 in the U0C0 and U1C1 register to "0". The U0IRS, U1IRS, U0RRM and U1RRM bits are in the UCON register.
2. TXD2 pin is N channel open-drain output. No NCH bit in the U2C0 register is assigned. When write, set to "0".
3. Not all register bits are described above. Set those bits to "0" when writing to the registers in Special Mode 2.

i = 0 to 2



### 17.1.4.1 Clock Phase Setting Function

One of four combinations of transfer 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 transfer clock polarity and phase are the same for the master and slaves to be communicated.

Figure 17.30 shows the Transmission and Reception Timing in Master Mode (Internal Clock).

Figure 17.31 shows the Transmission and Reception Timing (CKPH=0) in Slave Mode (External Clock) while

Figure 17.32 shows the Transmission and Reception Timing (CKPH=1) in Slave Mode (External Clock).

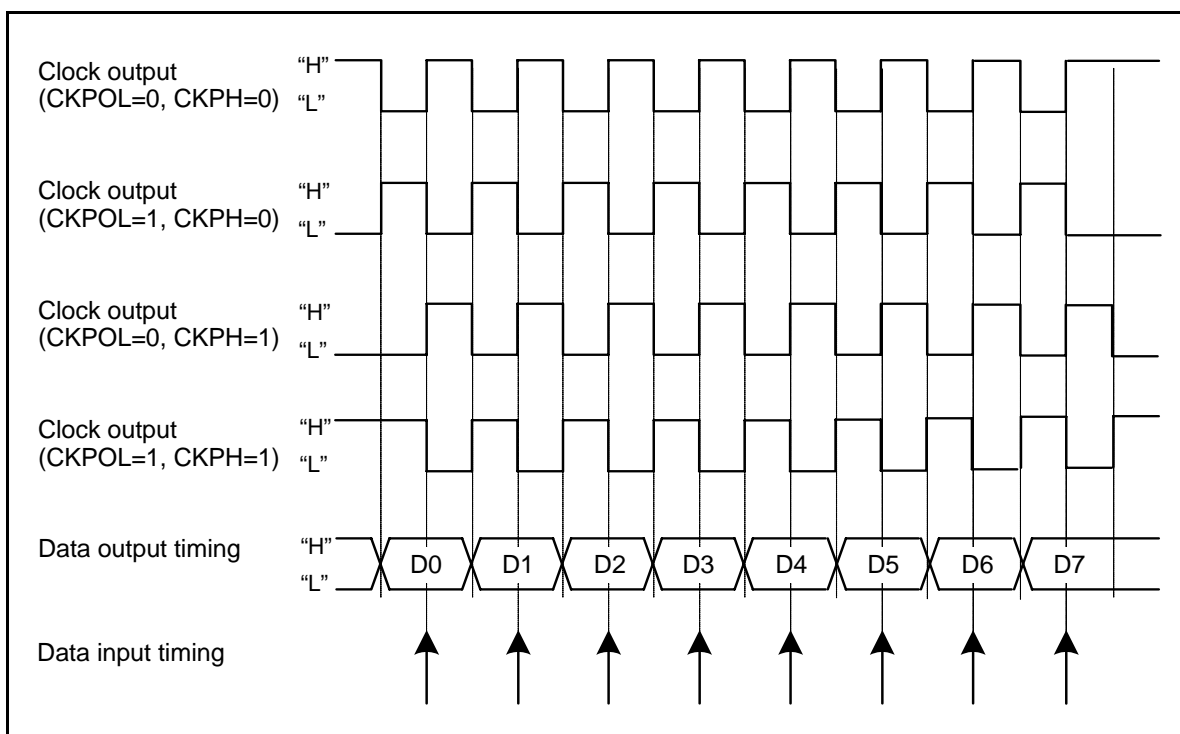


Figure 17.30 Transmission and Reception Timing in Master Mode (Internal Clock)

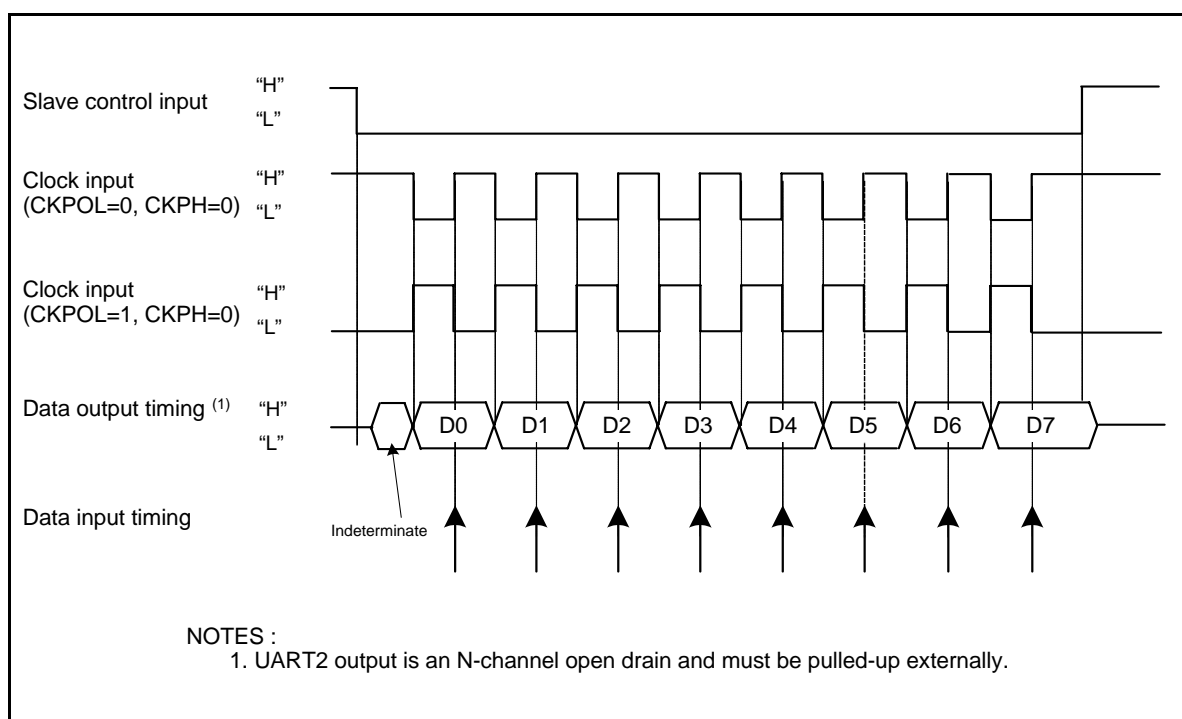


Figure 17.31 Transmission and Reception Timing (CKPH=0) in Slave Mode (External Clock)

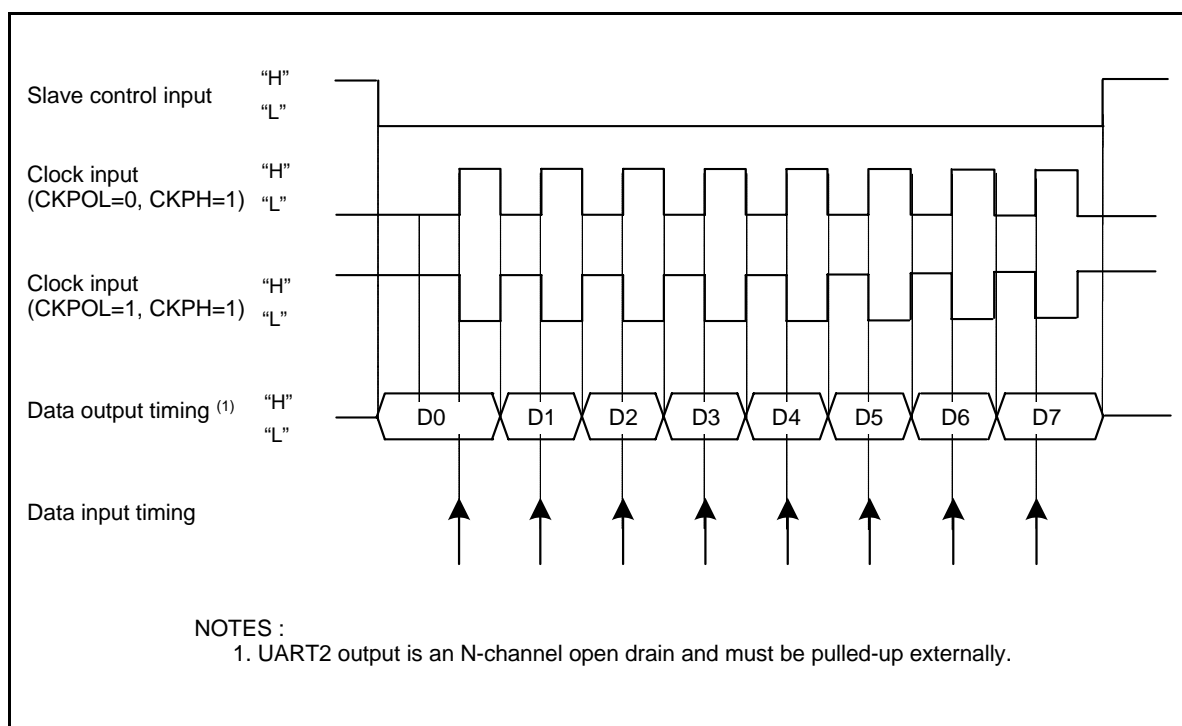


Figure 17.32 Transmission and Reception Timing (CKPH=1) in Slave Mode (External Clock)

### 17.1.5 Special Mode 3 (IE mode)

In this mode, one bit of IEBus is approximated with one byte of UART mode waveform.

Table 17.17 lists the Registers to Be Used and Settings in IE Mode. Figure 17.33 shows the Bus Collision Detect Function-Related Bits.

If the TXDi pin (i = 0 to 2) output level and RXDi pin input level do not match, a UARTi bus collision detect interrupt request is generated.

Use the IFSR26 and IFSR27 bits in the IFSR2A register to enable the UART0/UART1 bus collision detect function.

**Table 17.17 Registers to Be Used and Settings in IE Mode**

Register	Bit	Function
UiTB	0 to 8	Set transmission data
UiRB <sup>(3)</sup>	0 to 8	Reception data can be read
	OER, FER, PER, SUM	Error flag
UiBRG	0 to 7	Set a bit rate
UiMR	SMD2 to SMD0	Set to "110b"
	CKDIR	Select the internal clock or external clock
	STPS	Set to "0"
	PRY	Invalid because PRYE=0
	PRYE	Set to "0"
	IOPOL	Select the TXD/RXD input/output polarity
UiC0	CLK1, CLK0	Select the count source for the UiBRG register
	CRS	Invalid because CRD=1
	TXEPT	Transmit register empty flag
	CRD	Set to "1"
	NCH	Select TXDi pin output mode <sup>(2)</sup>
	CKPOL	Set to "0"
	UFORM	Set to "0"
UiC1	TE	Set this bit to "1" to enable transmission
	TI	Transmit buffer empty flag
	RE	Set this bit to "1" to enable reception
	RI	Reception complete flag
	U2IRS <sup>(1)</sup>	Select the source of UART2 transmit interrupt
	U2RRM <sup>(1)</sup> , UiLCH, UiERE	Set to "0"
UiSMR	0 to 3, 7	Set to "0"
	ABSCS	Select the sampling timing at which to detect a bus collision
	ACSE	Set this bit 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"
IFSR2A	IFSR26, IFSR27	Set to "1"
UCON	U0IRS, U1IRS	Select the source of UART0/UART1 transmit interrupt
	U0RRM, U1RRM	Set to "0"
	CLKMD0	Invalid because CLKMD1 = 0
	CLKMD1, RCSP, 7	Set to "0"

**NOTES:**

1. Set the bit 4 and bit 5 in the U0C0 and U1C1 registers to "0". The U0IRS, U1IRS, U0RRM and U1RRM bits are in the UCON register.
2. TXD2 pin is N channel open-drain output. No NCH bit in the U2C0 register is assigned. When write, set to "0".
3. Not all register bits are described above. Set those bits to "0" when writing to the registers in IE mode.

i= 0 to 2

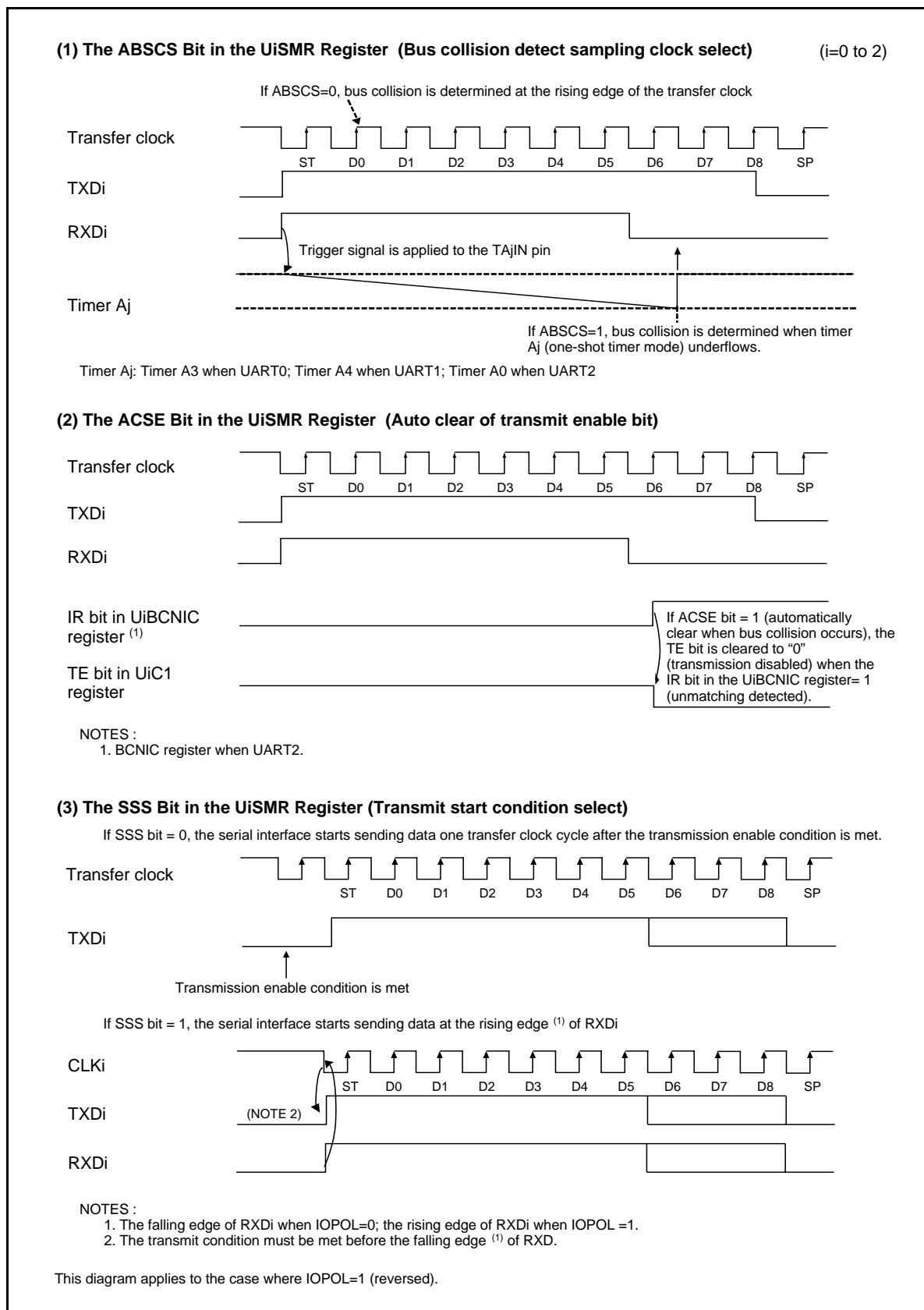


Figure 17.33 Bus Collision Detect Function-Related Bits

### 17.1.6 Special Mode 4 (SIM Mode) (UART2)

Based on UART mode, this is an SIM interface compatible mode. Direct and inverse formats can be implemented, and this mode allows to output a low from the TXD2 pin when a parity error is detected.

Table 17.18 lists the SIM Mode Specifications. Table 17.19 lists the Registers to Be Used and Settings in SIM Mode.

**Table 17.18 SIM Mode Specifications**

Item	Specification
Transfer Data Format	<ul style="list-style-type: none"> <li>• Direct format</li> <li>• Inverse format</li> </ul>
Transfer Clock	<ul style="list-style-type: none"> <li>• CKDIR bit in U2MR register = 0 (internal clock) : <math>f_i / (16(n+1))</math></li> <li>• <math>f_i = f1SIO, f2SIO, f8SIO, f32SIO</math> n: Setting value of U2BRG register 00h to FFh</li> <li>• CKDIR bit = 1 (external clock) : <math>fEXT / (16(n+1))</math></li> <li>• fEXT: Input from CLK2 pin n: Setting value of U2BRG register 00h to FFh</li> </ul>
Transmission Start Condition	<p>Before transmission can start, meet the following requirements</p> <ul style="list-style-type: none"> <li>• The TE bit in the U2C1 register = 1 (transmission enabled)</li> <li>• The TI bit in the U2C1 register = 0 (data present in U2TB register)</li> </ul>
Reception Start Condition	<p>Before reception can start, meet the following requirements</p> <ul style="list-style-type: none"> <li>• The RE bit in the U2C1 register = 1 (reception enabled)</li> <li>• Start bit detection</li> </ul>
Interrupt Request Generation Timing (2)	<ul style="list-style-type: none"> <li>• For transmission When the serial interface finished sending data from the U2TB transfer register (U2IRS bit =1)</li> <li>• For reception When transferring data from the UART2 receive register to the U2RB register (at completion of reception)</li> </ul>
Error Detection	<ul style="list-style-type: none"> <li>• Overrun error (1) This error occurs if the serial interface started receiving the next data before reading the U2RB register and received the bit one before the last stop bit of the next data</li> <li>• Framing error (3) This error occurs when the number of stop bits set is not detected</li> <li>• Parity error (3) During reception, if a parity error is detected, 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</li> <li>• Error sum flag This flag is set (= 1) when any of the overrun, framing, and parity errors is encountered</li> </ul>

**NOTES:**

1. If an overrun error occurs, the value of U2RB register will be indeterminate. The IR bit in the S2RIC register does not change.
2. A transmit interrupt request is generated by setting the U2IRS bit to "1" (transmission complete) and U2ERE bit to "1" (error signal output) in the U2C1 register after reset is deserted. Therefore, when using SIM mode, set the IR bit to "0" (no interrupt request) after setting these bits.
3. The timing at which the framing error flag and the parity error flag are set is detected when data is transferred from the UARTi receive register to the UiRB register.

**Table 17.19 Registers to Be Used and Settings in SIM Mode**

Register	Bit	Function
U2TB <sup>(1)</sup>	0 to 7	Set transmission data
U2RB <sup>(1)</sup>	0 to 7	Reception data can be read
	OER,FER,PER,SUM	Error flag
U2BRG	0 to 7	Set a bit rate
U2MR	SMD2 to SMD0	Set to "101b"
	CKDIR	Select the internal clock or external clock
	STPS	Set to "0"
	PRY	Set this bit to "1" for direct format or "0" for inverse format
	PRYE	Set to "1"
	IOPOL	Set to "0"
U2C0	CLK1, CLK0	Select the count source for the U2BRG register
	CRS	Invalid because CRD = 1
	TXEPT	Transmit register empty flag
	CRD	Set to "1"
	NCH	Set to "0"
	CKPOL	Set to "0"
	UFORM	Set this bit to "0" for direct format or "1" for inverse format
U2C1	TE	Set this bit to "1" to enable transmission
	TI	Transmit buffer empty flag
	RE	Set this bit to "1" to enable reception
	RI	Reception complete flag
	U2IRS	Set to "1"
	U2RRM	Set to "0"
	U2LCH	Set this bit to "0" for direct format or "1" for inverse format
	U2ERE	Set to "1"
U2SMR <sup>(1)</sup>	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. Not all register bits are described above. Set those bits to "0" when writing to the registers in SIM mode.

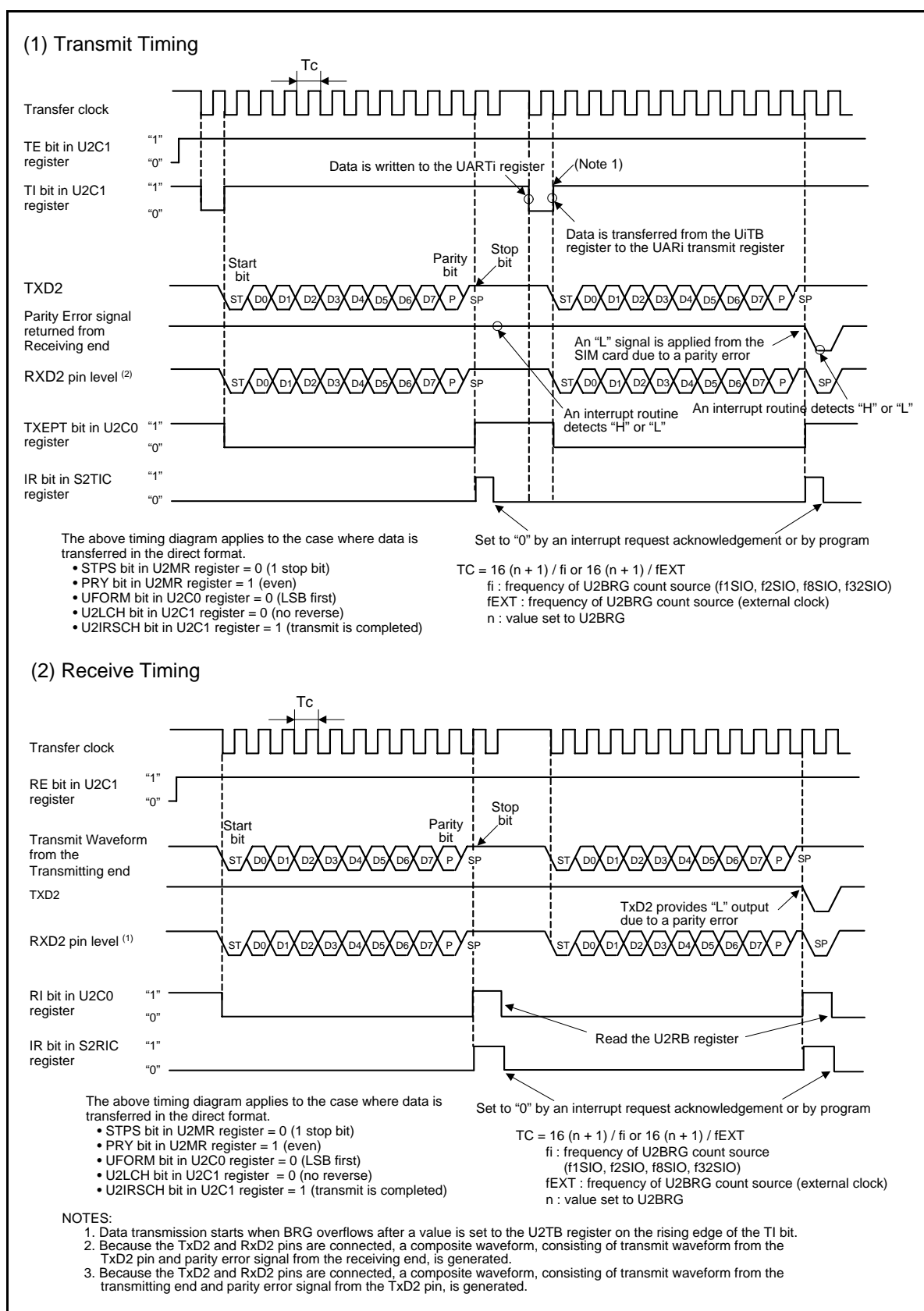


Figure 17.34 Transmit and Receive Timing in SIM Mode

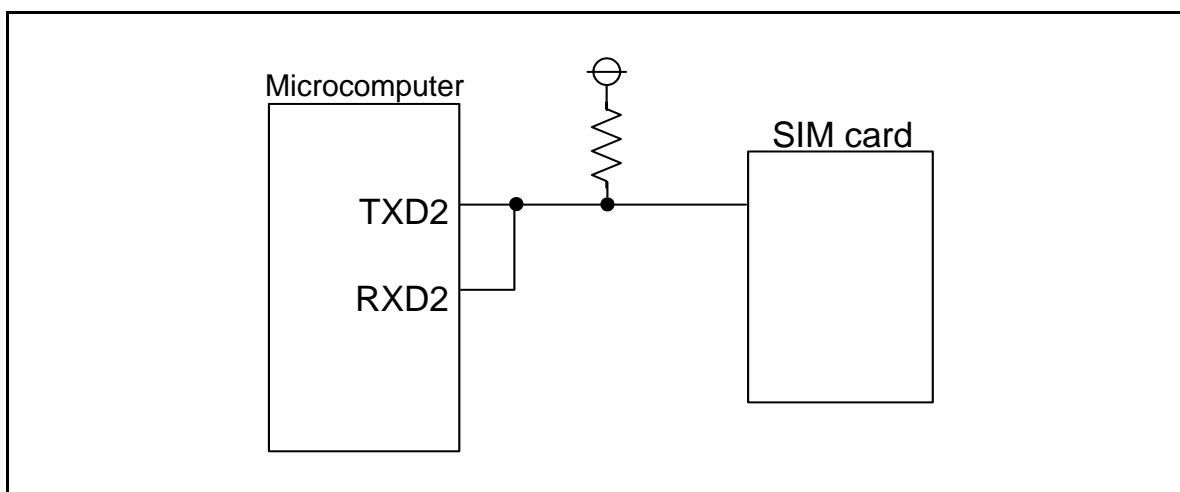


Figure 17.35 SIM Interface Connection

### 17.1.6.1 Parity Error Signal Output

The parity error signal is enabled by setting the U2ERE bit in the U2C1 register to "1".

The parity error signal is output when a parity error is detected while receiving data. This is achieved by pulling the TXD2 output low with the timing shown in Figure 17.36. If the R2RB register is read while outputting a parity error signal, the PER bit is cleared to "0" and at the same time the TXD2 output is returned high.

When transmitting, a transmission-finished interrupt request is generated at the falling edge of the transfer clock pulse that immediately follows the stop bit. Therefore, whether a parity signal has been returned can be determined by reading the port that shares the RXD2 pin in a transmission-finished interrupt routine.

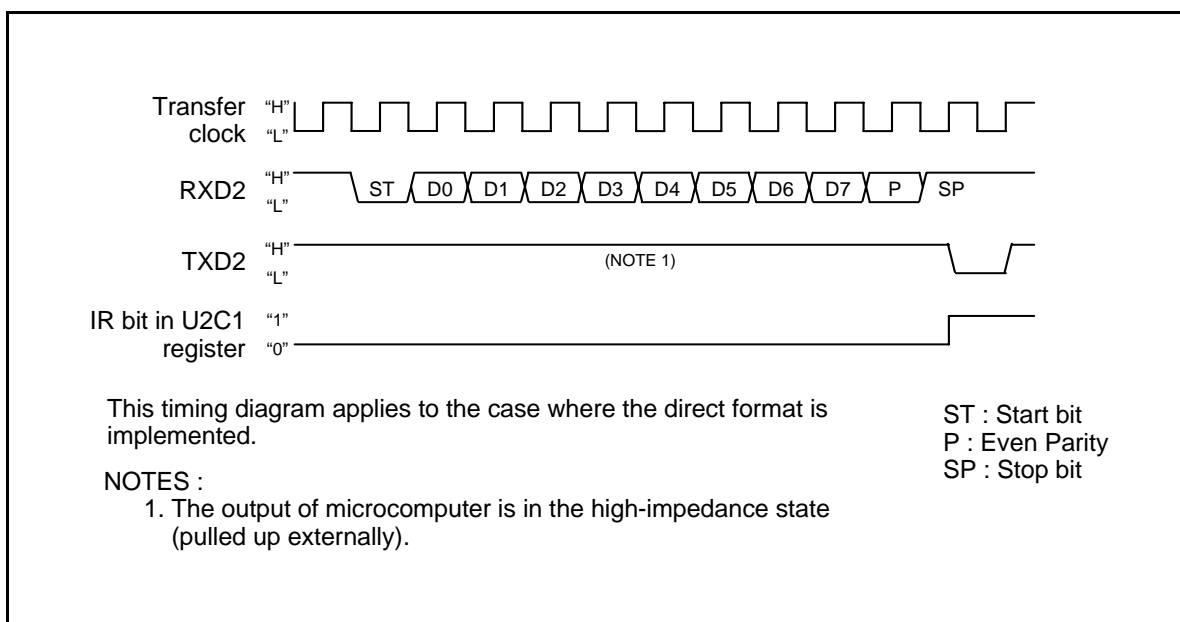


Figure 17.36 Parity Error Signal Output Timing



### 17.1.6.2 Format

When direct format, set the PRYE bit in the U2MR register to “1”, the PRY bit to “1”, the UFORM bit in the U2C0 register to “0” and the U2LCH bit in the U2C1 register to “0”. When data are transmitted, data set in the U2TB register are transmitted with the even-numbered parity, starting from D0. When data are received, received data are stored in the U2RB register, starting from D0. The even-numbered parity determines whether a parity error occurs.

When inverse format, set the PRYE bit to “1”, the PRY bit to “0”, the UFORM bit to “1” and the U2LCH bit to “1”. When data are transmitted, values set in the U2TB register are logically inverted and are transmitted with the odd-numbered parity, starting from D7. When data are received, received data are logically inverted to be stored in the U2RB register, starting from D7. The odd-numbered parity determines whether a parity error occurs.

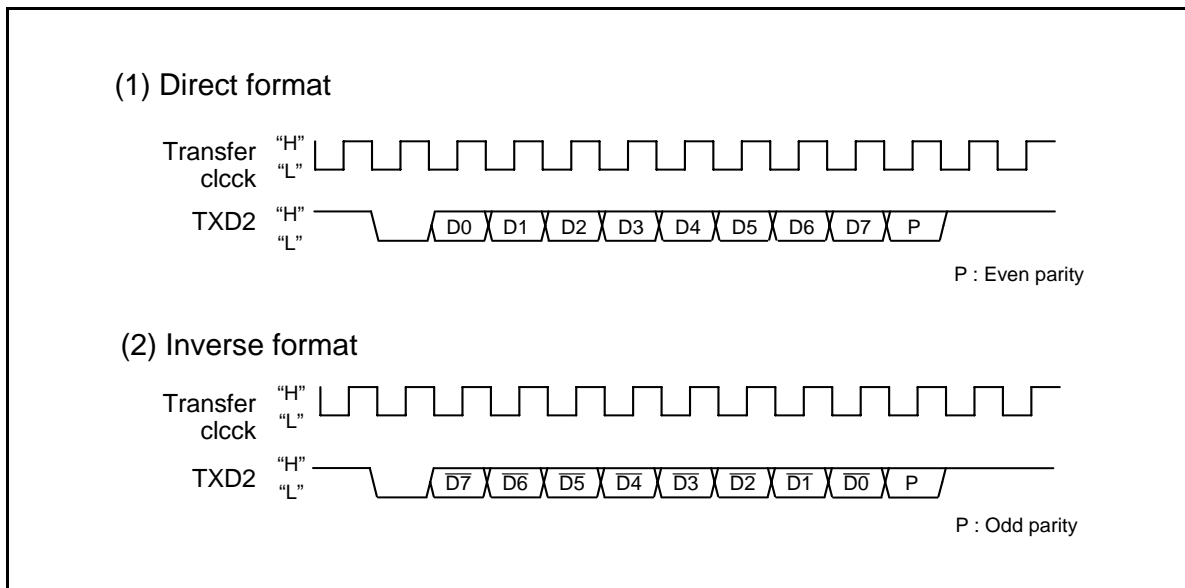


Figure 17.37 SIM Interface Format

## 17.2 SI/O3 and SI/O4

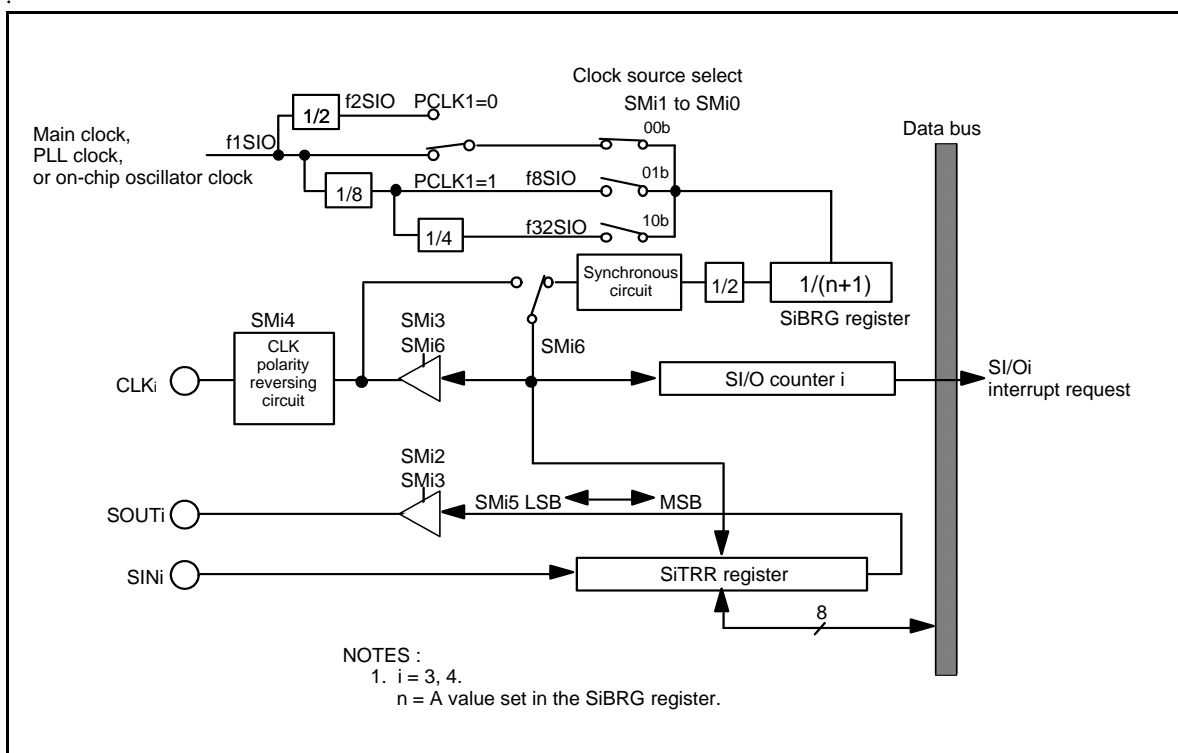
### Note

The M16C/62P (80-pin version) and M16C/62PT (80-pin version) do not include SIN3 pin of SI/O3. SI/O3 is only for transmission. Reception is impossible.

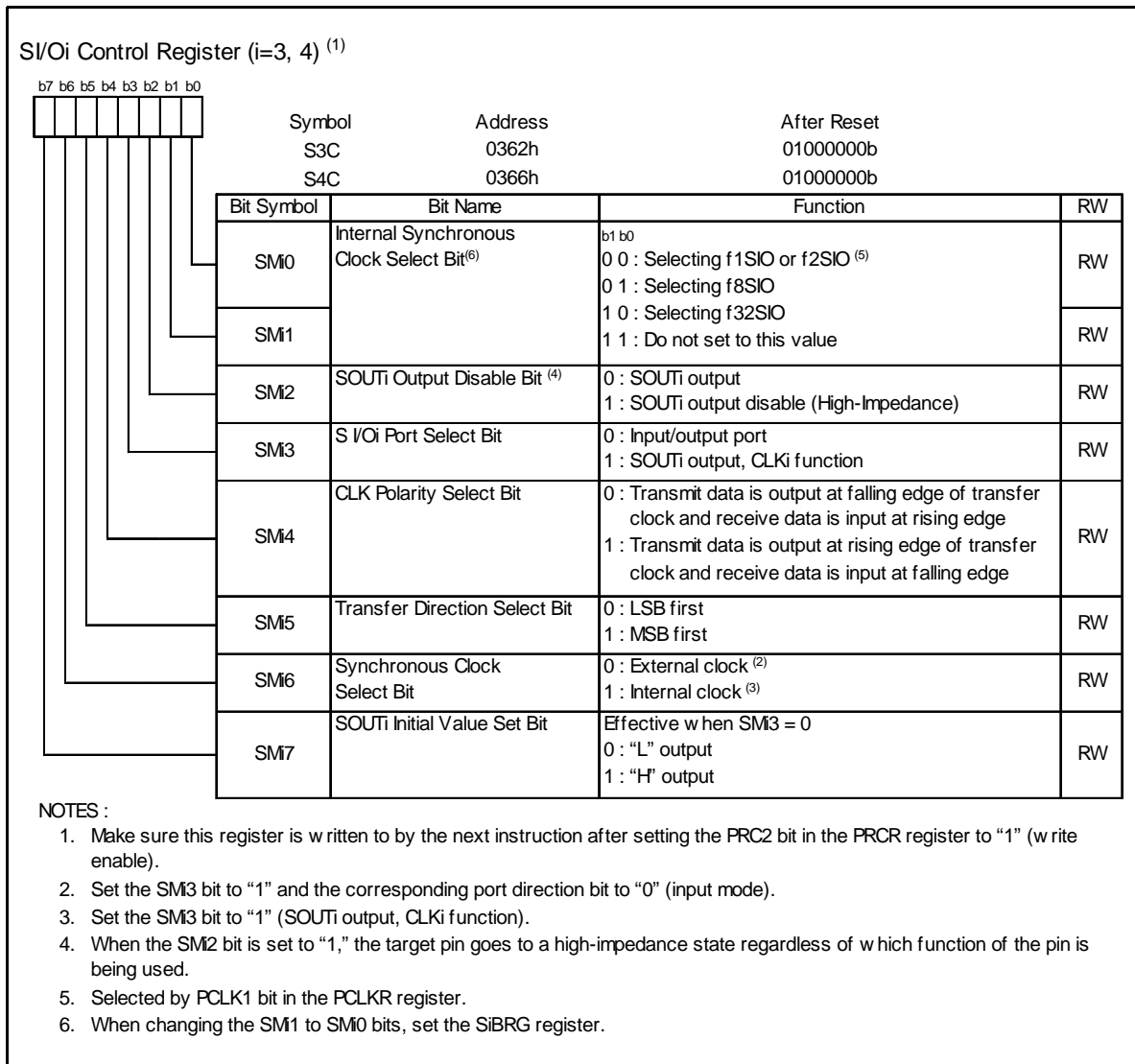
SI/O3 and SI/O4 are exclusive clock-synchronous serial I/Os.

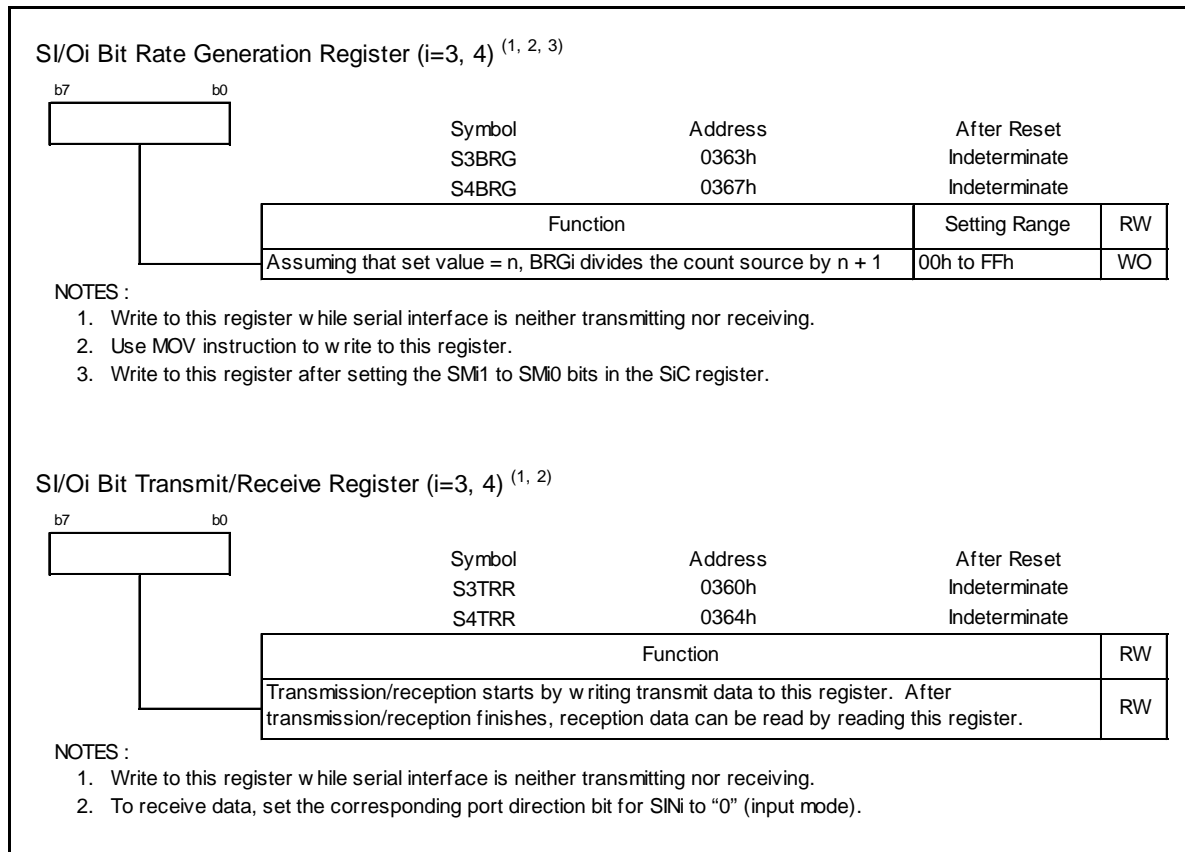
Figure 17.38 shows the SI/O3 and SI/O4 Block Diagram, and Figure 17.39 to Figure 17.40 show the SI/O3 and SI/O4-related registers.

Table 17.20 shows the SI/O3 and SI/O4 Specifications.



**Figure 17.38 SI/O3 and SI/O4 Block Diagram**

**Figure 17.39 SiC Register**

**Figure 17.40 SiBRG and SiTRR Registers**

**Table 17.20 SI/O3 and SI/O4 Specifications**

Item	Specification
Transfer Data Format	• Transfer data length: 8 bits
Transfer Clock	<ul style="list-style-type: none"> <li>• SMI6 bit in SiC (i=3, 4) register = 1 (internal clock) : <math>f_j / (2(n+1))</math>  <math>f_j = f1SIO, f8SIO, f32SIO</math>. n = Setting value of SiBRG register 00h to FFh.</li> <li>• SMI6 bit = 0 (external clock) : Input from CLKi pin <sup>(1)</sup></li> </ul>
Transmission/Reception Start Condition	<ul style="list-style-type: none"> <li>• Before transmission/reception can start, meet the following requirements  Write transmit data to the SiTRR register <sup>(2, 3)</sup></li> </ul>
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• When SMI4 bit in SiC register = 0  The rising edge of the last transfer clock pulse <sup>(4)</sup>  When SMI4 = 1  The falling edge of the last transfer clock pulse <sup>(4)</sup></li> </ul>
CLKi Pin Function	I/O port, transfer clock input, transfer clock output
SOUTi Pin Function	I/O port, transmit data output, high-impedance
SINi Pin Function	I/O port, receive data input
Select Function	<ul style="list-style-type: none"> <li>• LSB first or MSB first selection  Whether to start sending/receiving data beginning with bit 0 or beginning with bit 7 can be selected</li> <li>• Function for setting an SOUTi initial value set function  When the SMI6 bit in the SiC register = 0 (external clock), the SOUTi pin output level while not transmitting can be selected.</li> <li>• CLK polarity selection  Whether transmit data is output/input timing at the rising edge or falling edge of transfer clock can be selected.</li> </ul>

**NOTES:**

- To set SMI6 bit to "0" (external clock), follow the procedure described below.
  - If the SMI4 bit = 0, write transmit data to the SiTRR register while input on the CLKi pin is high. The same applies when rewriting the SMI7 bit in the SiC register.
  - If the SMI4 bit = 1, write transmit data to the SiTRR register while input on the CLKi pin is low. The same applies when rewriting the SMI7 bit.
  - Because shift operation continues as long as the transfer clock is supplied to the SI/Oi circuit, stop the transfer clock after supplying eight pulses. If the SMI6 bit = 1 (internal clock), the transfer clock automatically stops.
- Unlike UART0 to UART2, SI/Oi (i = 3 to 4) is not separated between the transfer register and buffer. Therefore, do not write the next transmit data to the SiTRR register during transmission.
- When SMI6 bit = 1 (internal clock), SOUTi retains the last data for a 1/2 transfer clock period after completion of transfer and, thereafter, goes to a high-impedance state. However, if transmit data is written to the SiTRR register during this period, SOUTi immediately goes to a high-impedance state, with the data hold time thereby reduced.
- When the SMI6 bit = 1 (internal clock), the transfer clock stops in the high state if the SMI4 bit = 0, or stops in the low state if the SMI4 bit = 1.

### 17.2.1 SI/Oi Operation Timing

Figure 17.41 shows the SI/Oi Operation Timing.

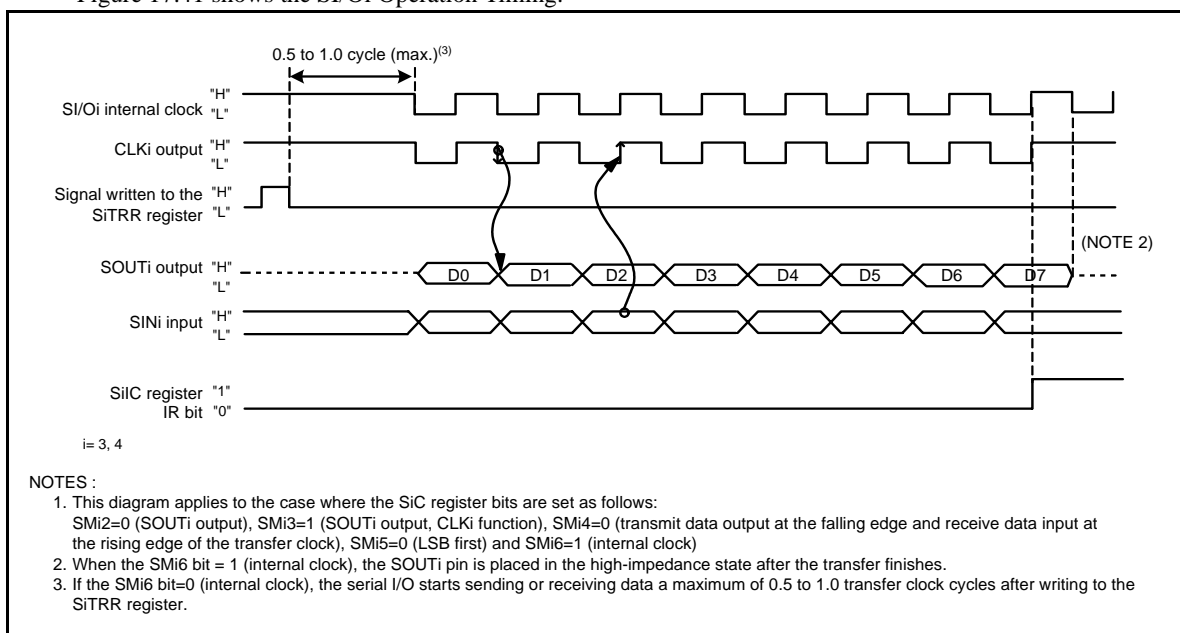


Figure 17.41 SI/Oi Operation Timing

### 17.2.2 CLK Polarity Selection

The SMi4 bit in the SiC register allows selection of the polarity of the transfer clock. Figure 17.42 shows the Polarity of Transfer Clock.

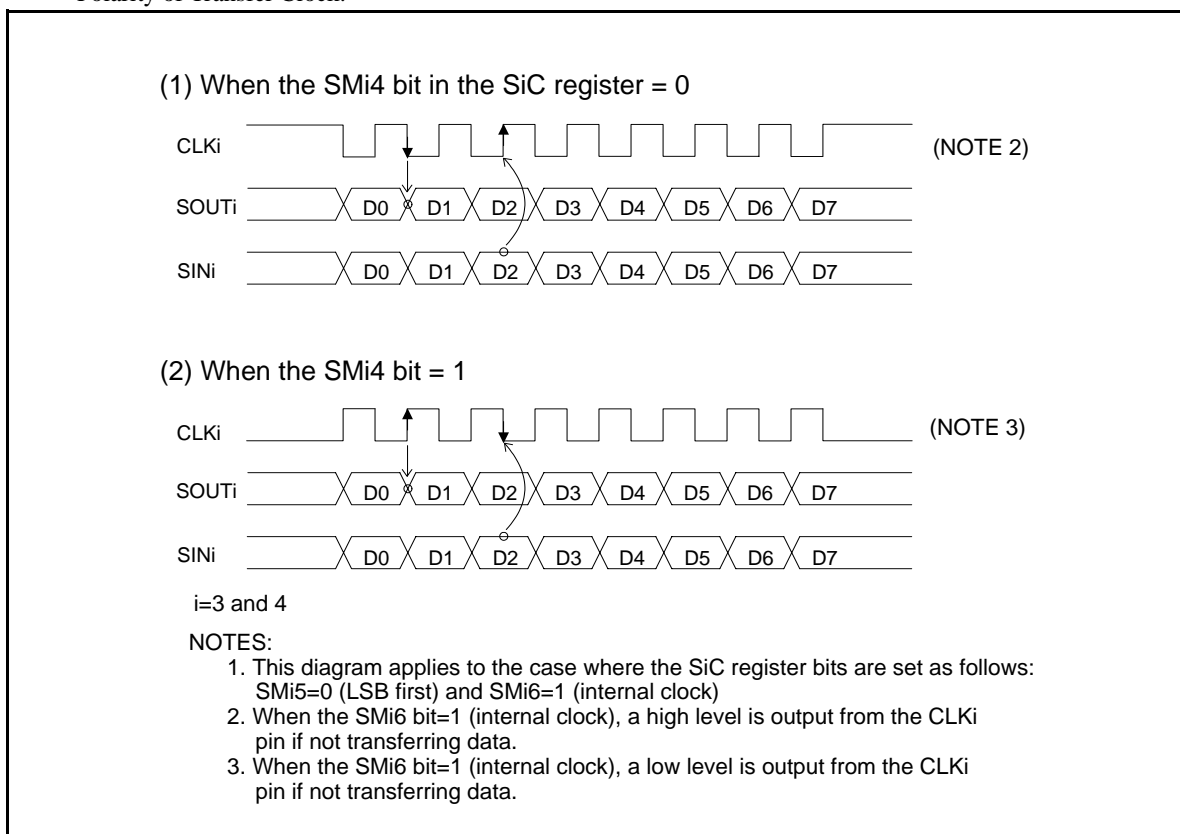


Figure 17.42 Polarity of Transfer Clock

### 17.2.3 Functions for Setting an SOUTi Initial Value

If the SMi6 bit in the SiC register = 0 (external clock), the SOUTi pin output can be fixed high or low when not transferring. However, the last bit value of the former data is retained between data and data when transmitting the continuous data. Figure 17.43 shows the timing chart for setting an SOUTi initial value and how to set it.

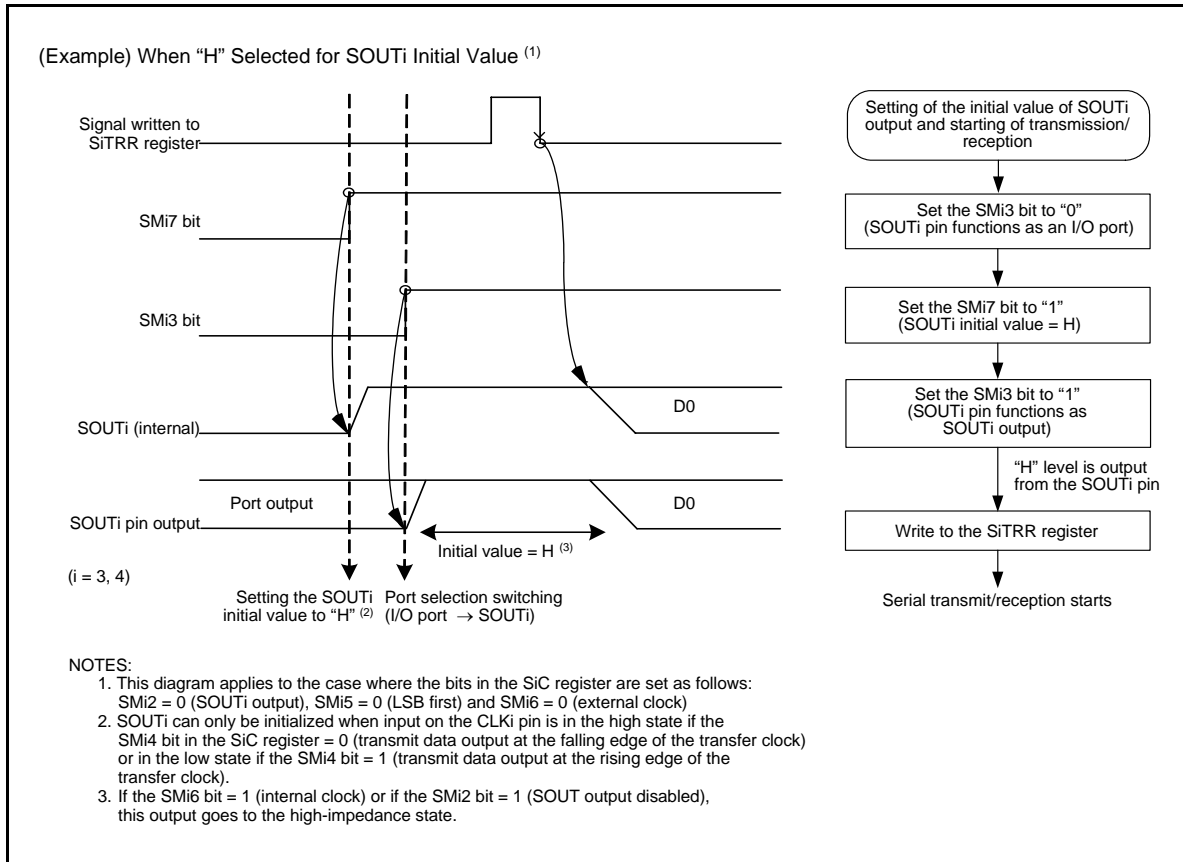


Figure 17.43 SOUTi's Initial Value Setting

## 18. A/D Converter

The microcomputer contains one A/D converter circuit based on 10-bit successive approximation method configured with a capacitive-coupling amplifier. The analog inputs share the pins with P10\_0 to P10\_7, P9\_5, P9\_6, and P0\_0 to P0\_7, and P2\_0 to P2\_7. Similarly, ADTRG input shares the pin with P9\_7. Therefore, when using these inputs, make sure the corresponding port direction bits are set to “0” (= input mode).

When not using the A/D converter, set the VCUT bit to “0” (= Vref unconnected), so that no current will flow from the VREF pin into the resistor ladder, helping to reduce the power consumption of the chip.

The A/D conversion result is stored in the ADi register bits for ANi, AN0\_i, and AN2\_i pins (i = 0 to 7).

Table 18.1 shows the Performance of A/D Converter. Figure 18.1 shows the A/D Converter Block Diagram, and Figures 18.2 and 18.3 show the A/D converter-related registers.

**Table 18.1 Performance of A/D Converter**

Item	Performance
Method of A/D Conversion	Successive approximation (capacitive coupling amplifier)
Analog input Voltage <sup>(1)</sup>	0V to AVCC (VCC1)
Operating clock $\phi$ AD <sup>(2)</sup>	fAD/divide-by-2 of fAD/divide-by-3 of fAD/divide-by-4 of fAD/divide-by-6 of fAD/divide-by-12 of fAD
Resolution	8-bit or 10-bit (selectable)
Integral Nonlinearity Error	When AVCC = VREF = 5V <ul style="list-style-type: none"> <li>• With 8-bit resolution: <math>\pm 2</math>LSB</li> <li>• With 10-bit resolution               <ul style="list-style-type: none"> <li>AN0 to AN7 input, AN0_0 to AN0_7 input and AN2_0 to AN2_7 input : 3LSB</li> <li>ANEX0 and ANEX1 input (including mode in which external Op-Amp is connected) : <math>\pm 7</math>LSB</li> </ul> </li> </ul> When AVCC = VREF = 3.3V <ul style="list-style-type: none"> <li>• With 8-bit resolution: <math>\pm 2</math>LSB</li> <li>• With 10-bit resolution               <ul style="list-style-type: none"> <li>AN0 to AN7 input, AN0_0 to AN0_7 input and AN2_0 to AN2_7 input : <math>\pm 5</math>LSB</li> <li>ANEX0 and ANEX1 input (including mode in which external Op-Amp is connected) : <math>\pm 7</math>LSB</li> </ul> </li> </ul>
Operating Modes	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, and repeat sweep mode 1
Analog Input Pins <sup>(3)</sup>	8 pins (AN0 to AN7) + 2 pins (ANEX0 and ANEX1) + 8 pins (AN0_0 to AN0_7) + 8 pins (AN2_0 to AN2_7)
A/D Conversion Start Condition	<ul style="list-style-type: none"> <li>• Software trigger The ADST bit in the ADCON0 register is set to “1” (A/D conversion starts)</li> <li>• External trigger (retriggerable) Input on the ADTRG pin changes state from high to low after the ADST bit is set to “1” (A/D conversion starts)</li> </ul>
Conversion Speed	<ul style="list-style-type: none"> <li>• Without sample and hold function 8-bit resolution: 49 <math>\phi</math>AD cycles, 10-bit resolution: 59 <math>\phi</math>AD cycles</li> <li>• With sample and hold function 8-bit resolution: 28 <math>\phi</math>AD cycles, 10-bit resolution: 33 <math>\phi</math>AD cycles</li> </ul>

**NOTES:**

1. Does not depend on use of sample and hold function.
2.  $\phi$ AD frequency must be 12 MHz or less. And divide the fAD if VCC1 is less than 4.0V, and  $\phi$ AD frequency into 10 MHz or less.  
When sample & hold is disabled,  $\phi$ AD frequency must be 250kHz or more.  
When sample & hold is enabled,  $\phi$ AD frequency must be 1MHz or more.
3. If VCC2 < VCC1, do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.



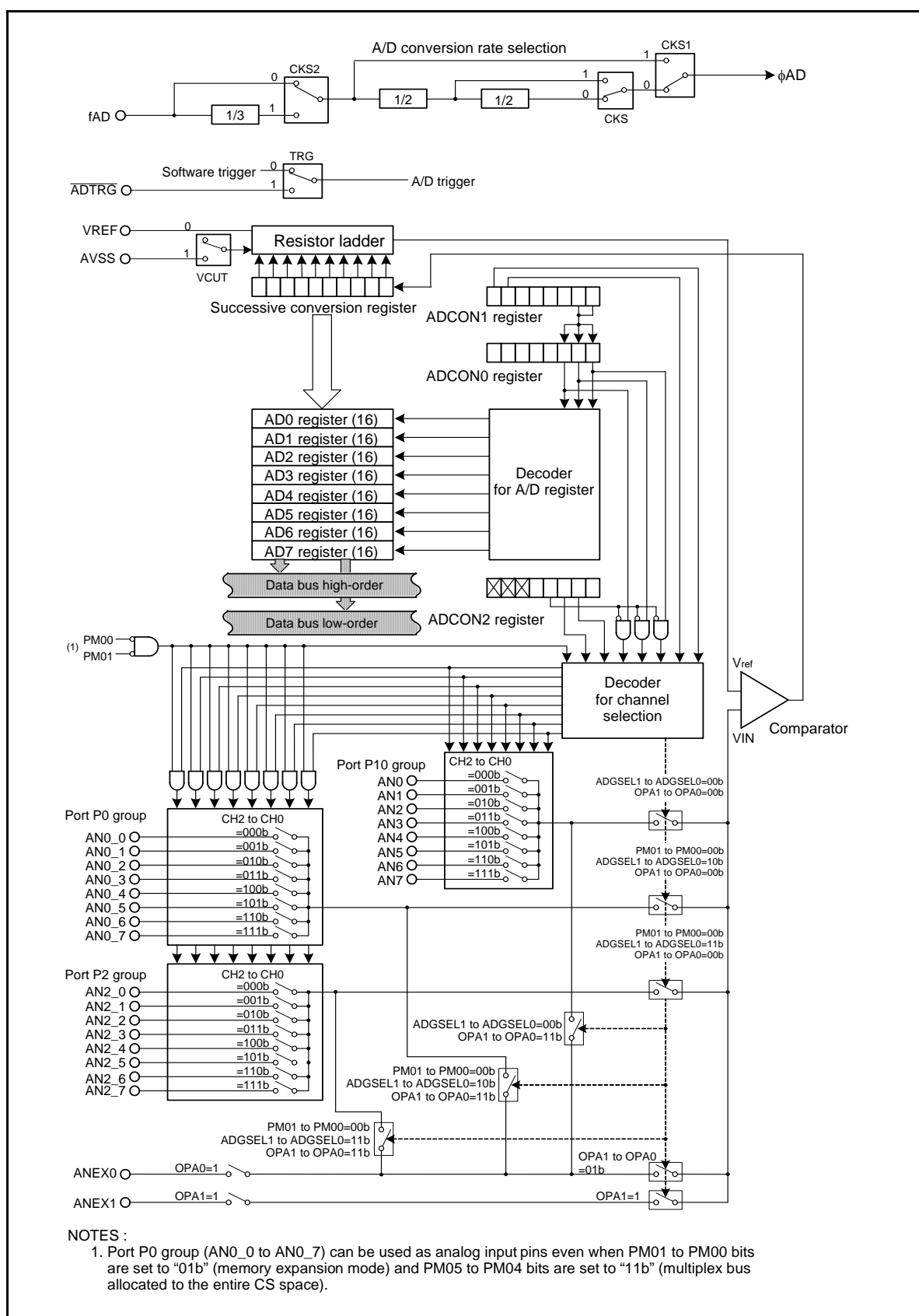


Figure 18.1 A/D Converter Block Diagram

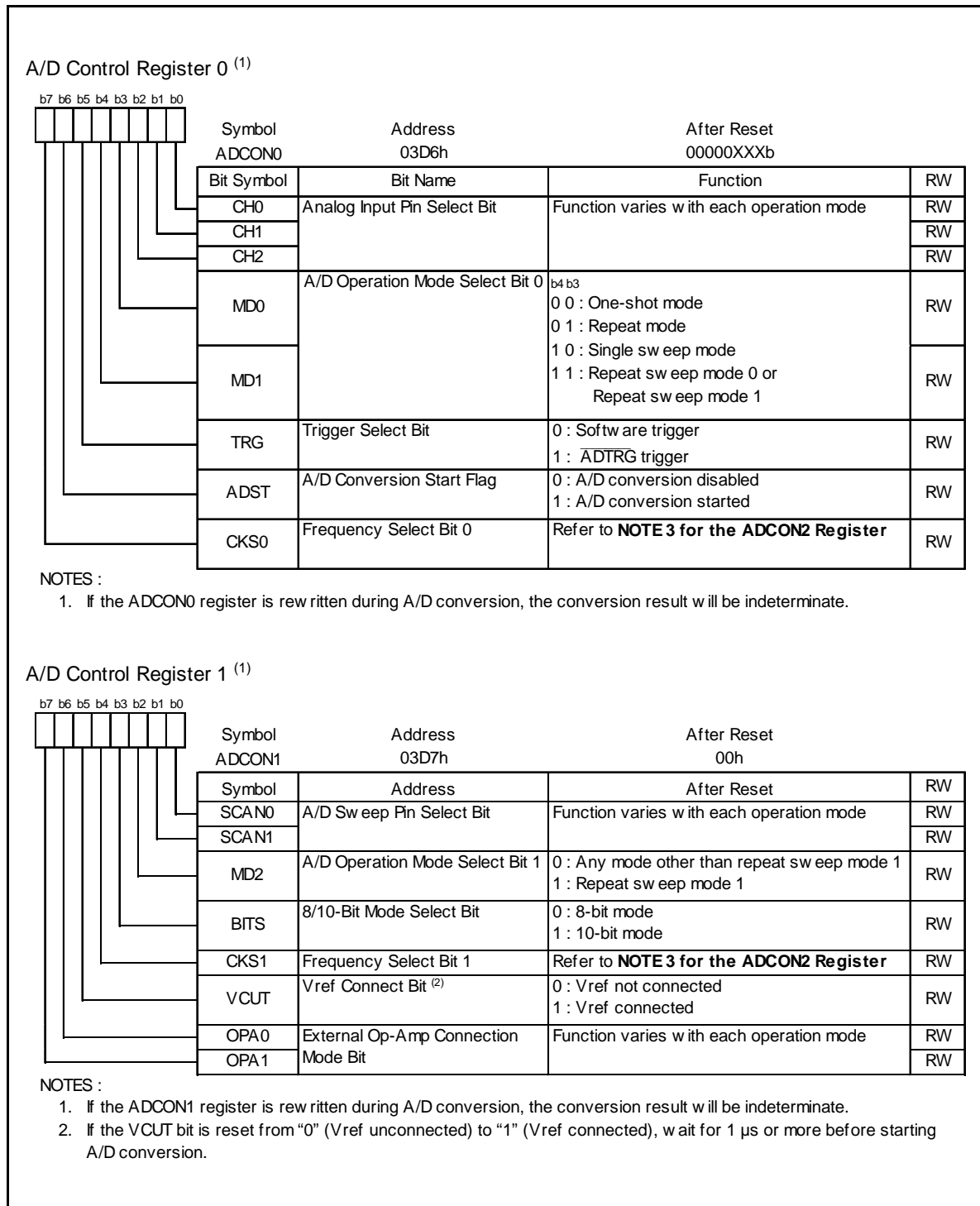
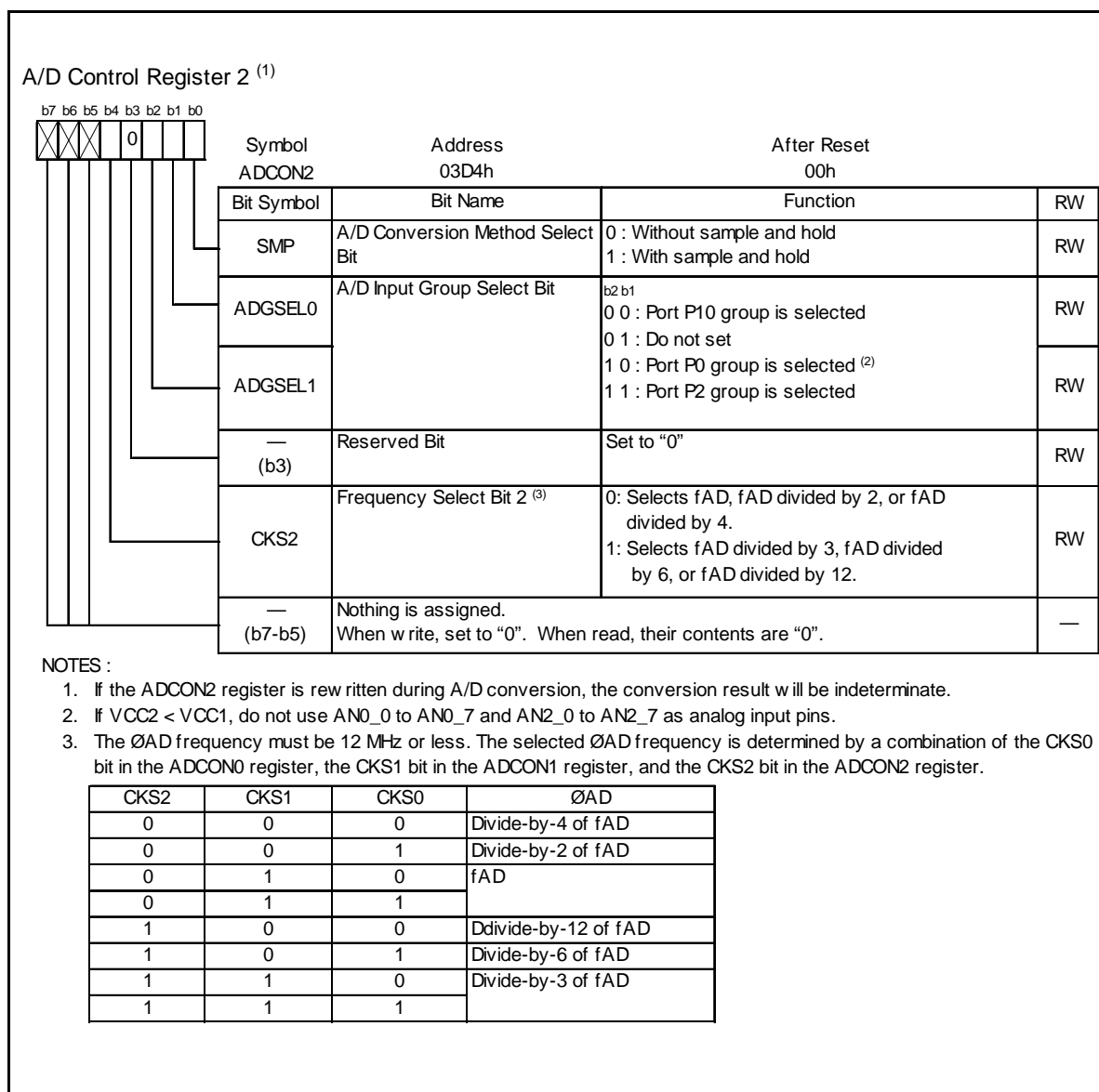


Figure 18.2 ADCON0 to ADCON1 Registers

**Figure 18.3 ADCON2 Register**

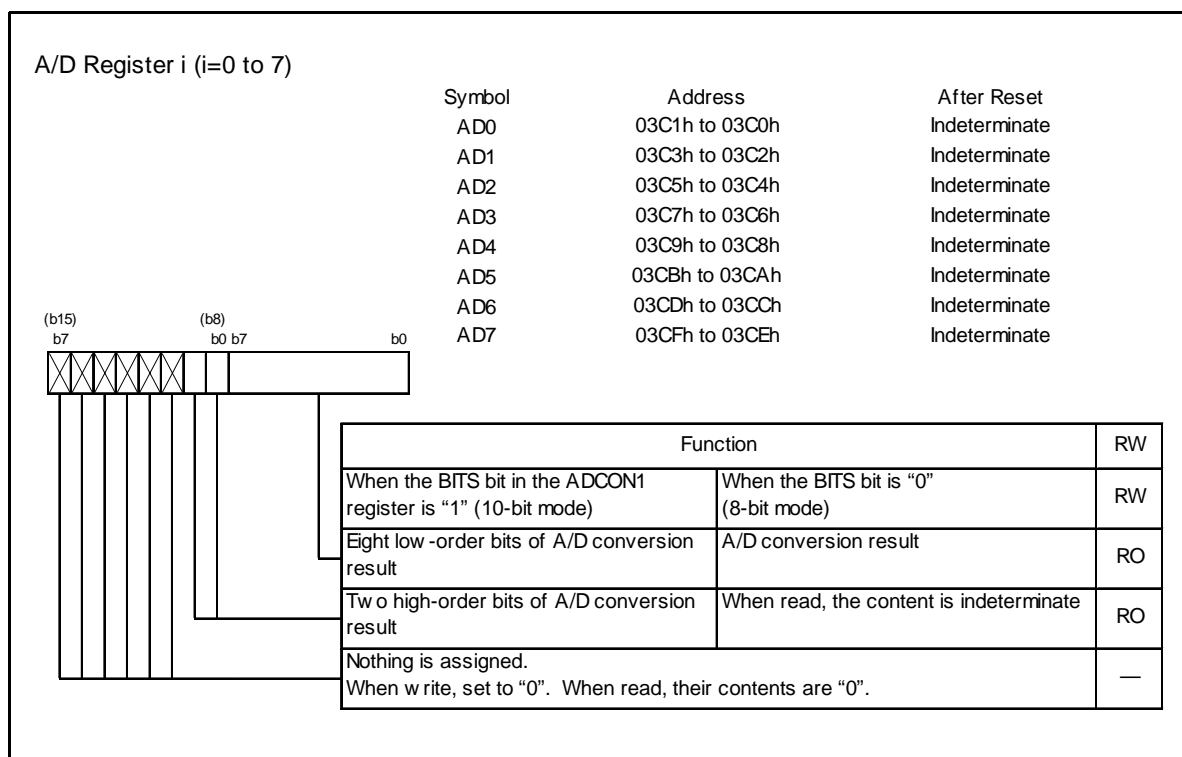


Figure 18.4 AD0 to AD7 Registers

## 18.1 Mode Description

### 18.1.1 One-Shot Mode

In one-shot mode, analog voltage applied to a selected pin is converted to a digital code once. Table 18.2 shows the One-Shot Mode Specifications. Figure 18.5 shows the ADCON0 and ADCON1 Registers (One-shot Mode).

**Table 18.2 One-Shot Mode Specifications**

Item	Specification
Function	The CH2 to CH0 bits in the ADCON0 register, the ADGSEL1 to ADGSEL0 bits in the ADCON2 register and the OPA1 to OPA0 bits in the ADCON1 register select a pin. Analog voltage applied to the pin is converted to a digital code once.
A/D Conversion Start Condition	<ul style="list-style-type: none"> <li>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)</li> <li>When the TRG bit is "1" (ADTRG trigger) Input on the ADTRG pin changes state from high to low after the ADST bit is set to "1" (A/D conversion starts)</li> </ul>
A/D Conversion Stop Condition	<ul style="list-style-type: none"> <li>Completion of A/D conversion (If a software trigger is selected, the ADST bit is cleared to "0" (A/D conversion halted))</li> <li>Set the ADST bit to "0"</li> </ul>
Interrupt Request Generation Timing	Completion of A/D conversion
Analog Input Pin <sup>(1)</sup>	Select one pin from AN0 to AN7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0 to ANEX1
Reading of Result of A/D Converter	Read one of the AD0 to AD7 registers that corresponds to the selected pin

**NOTES:**

1. If  $VCC2 < VCC1$ , do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.

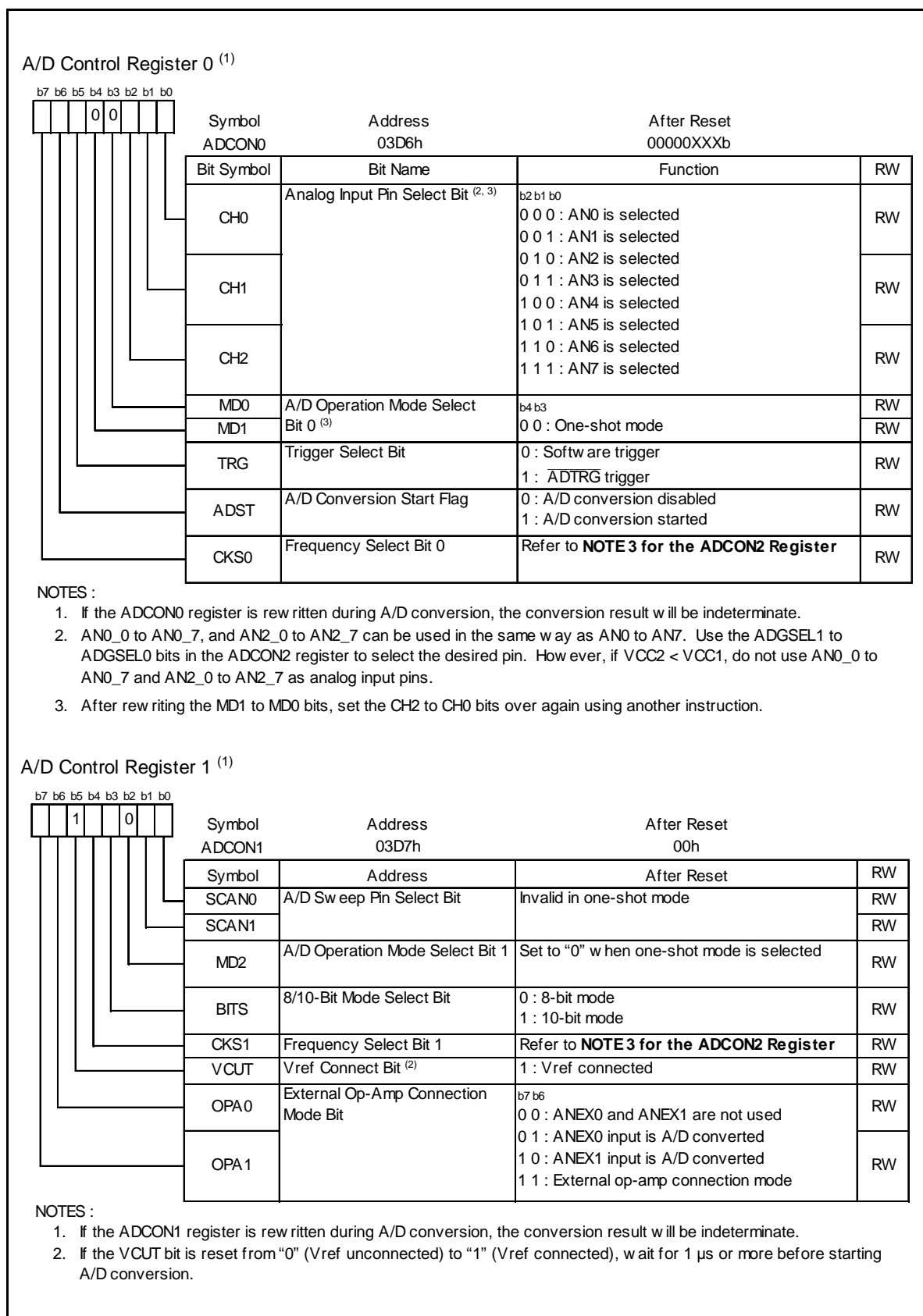


Figure 18.5 ADCON0 and ADCON1 Registers (One-shot Mode)

### 18.1.2 Repeat Mode

In repeat mode, analog voltage applied to a selected pin is repeatedly converted to a digital code. Table 18.3 shows the Repeat Mode Specifications. Figure 18.6 shows the ADCON0 and ADCON1 Registers (Repeat Mode).

**Table 18.3 Repeat Mode Specifications**

Item	Specification
Function	The CH2 to CH0 bits in the ADCON0 register, the ADGSEL1 to ADGSEL0 bits in the ADCON2 register and the OPA1 to OPA0 bits in the ADCON1 register select a pin. Analog voltage applied to this pin is repeatedly converted to a digital code.
A/D Conversion Start Condition	<ul style="list-style-type: none"> <li>• 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)</li> <li>• When the TRG bit is "1" (<math>\overline{\text{ADTRG}}</math> trigger) Input on the <math>\overline{\text{ADTRG}}</math> pin changes state from high to low after the ADST bit is set to "1" (A/D conversion starts)</li> </ul>
A/D Conversion Stop Condition	Set the ADST bit to "0" (A/D conversion halted)
Interrupt Request Generation timing	None generated
Analog Input Pin <sup>(1)</sup>	Select one pin from AN0 to AN7, AN0_0 to AN0_7, AN2_0 to AN2_7, ANEX0 to ANEX1
Reading of Result of A/D Converter	Read one of the AD0 to AD7 registers that corresponds to the selected pin

**NOTES:**

1. If  $VCC2 < VCC1$ , do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.

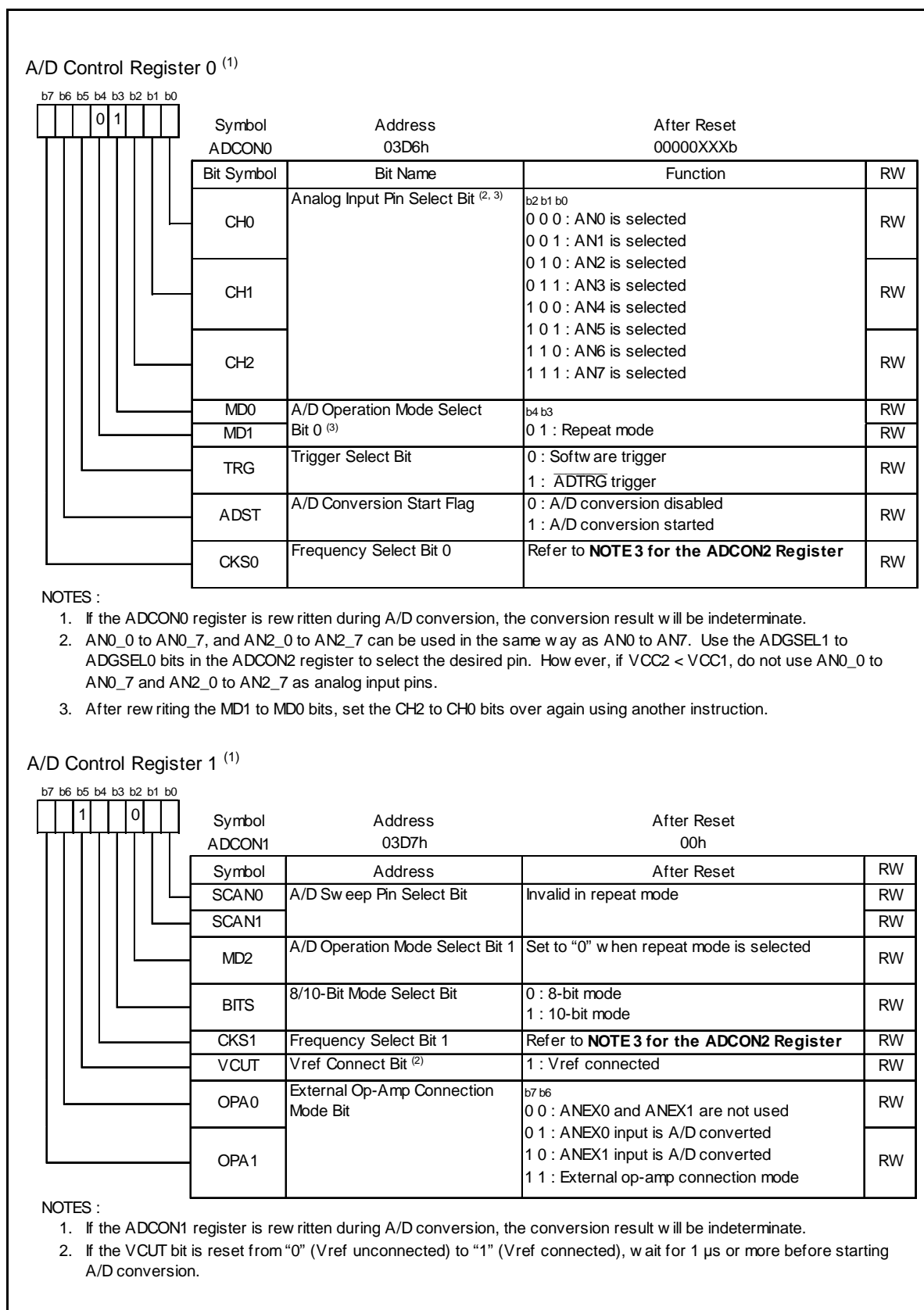


Figure 18.6 ADCON0 and ADCON1 Registers (Repeat Mode)



### 18.1.3 Single Sweep Mode

In single sweep mode, analog voltage that is applied to selected pins is converted one-by-one to a digital code. Table 18.4 shows the Single Sweep Mode Specifications. Figure 18.7 shows the ADCON0 Register and ADCON1 Register (Single Sweep Mode).

**Table 18.4 Single Sweep Mode Specifications**

Item	Specification
Function	The SCAN1 to SCAN0 bits in the ADCON1 register and the ADGSEL1 to ADGSEL0 bits in the ADCON2 register select pins. Analog voltage applied to this pins is converted one-by-one to a digital code.
A/D Conversion Start Condition	<ul style="list-style-type: none"> <li>• 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)</li> <li>• When the TRG bit is "1" (ADTRG trigger) Input on the ADTRG pin changes state from high to low after the ADST bit is set to "1" (A/D conversion starts)</li> </ul>
A/D Conversion Stop Condition	<ul style="list-style-type: none"> <li>• Completion of A/D conversion (If a software trigger is selected, the ADST bit is cleared to "0" (A/D conversion halted))</li> <li>• Set the ADST bit to "0"</li> </ul>
Interrupt Request Generation timing	Completion of A/D conversion
Analog Input Pin	Select from AN0 to AN1 (2 pins), AN0 to AN3 (4 pins), AN0 to AN5 (6 pins), AN0 to AN7 (8 pin) <sup>(1)</sup>
Reading of Result of A/D Converter	Read one of the AD0 to AD7 registers that corresponds to the selected pin

**NOTES:**

1. AN0\_0 to AN0\_7, and AN2\_0 to AN2\_7 can be used in the same way as AN0 to AN7. However, if  $VCC2 < VCC1$ , do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.

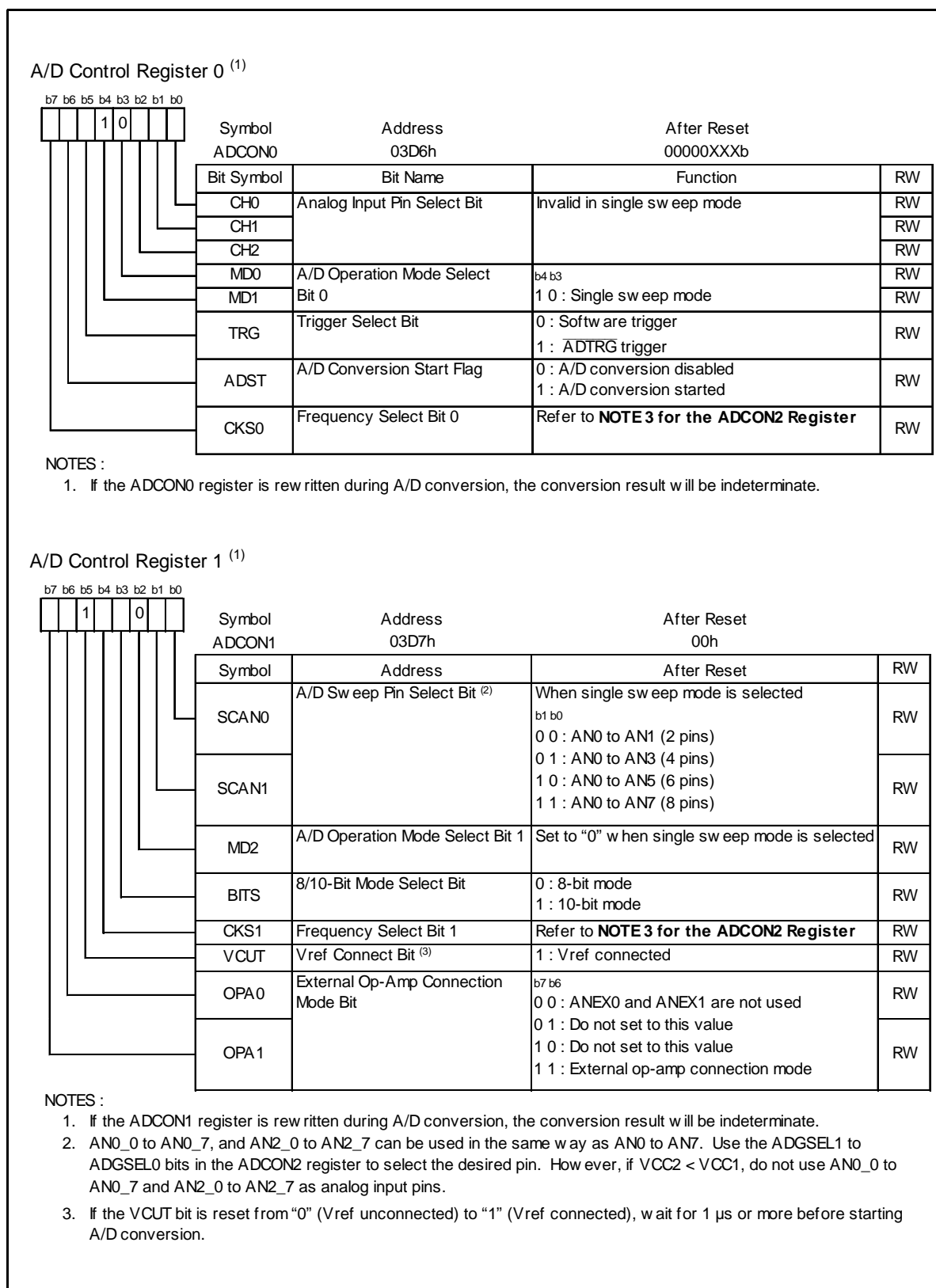


Figure 18.7 ADCON0 Register and ADCON1 Register (Single Sweep Mode)

### 18.1.4 Repeat Sweep Mode 0

In repeat sweep mode 0, analog voltage applied to selected pins is repeatedly converted to a digital code.

Table 18.5 shows the Repeat Sweep Mode 0 Specifications. Figure 18.8 shows the ADCON0 Register and ADCON1 Registers (Repeat Sweep Mode 0).

**Table 18.5 Repeat Sweep Mode 0 Specifications**

Item	Specification
Function	The SCAN1 to SCAN0 bits in the ADCON1 register and the ADGSEL1 to ADGSEL0 bits in the ADCON2 register select pins. Analog voltage applied to the pins is repeatedly converted to a digital code.
A/D Conversion Start Condition	<ul style="list-style-type: none"> <li>• 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)</li> <li>• When the TRG bit is "1" (ADTRG trigger) Input on the ADTRG pin changes state from high to low after the ADST bit is set to "1" (A/D conversion starts)</li> </ul>
A/D Conversion Stop Condition	Set the ADST bit to "0" (A/D conversion halted)
Interrupt Request Generation timing	None generated
Analog Input Pin	Select from AN0 to AN1 (2 pins), AN0 to AN3 (4 pins), AN0 to AN5 (6 pins), AN0 to AN7 (8 pin) <sup>(1)</sup>
Reading of Result of A/D Converter	Read one of the AD0 to AD7 registers that corresponds to the selected pin

**NOTES:**

1. AN0\_0 to AN0\_7, and AN2\_0 to AN2\_7 can be used in the same way as AN0 to AN7. However, if  $VCC2 < VCC1$ , do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.

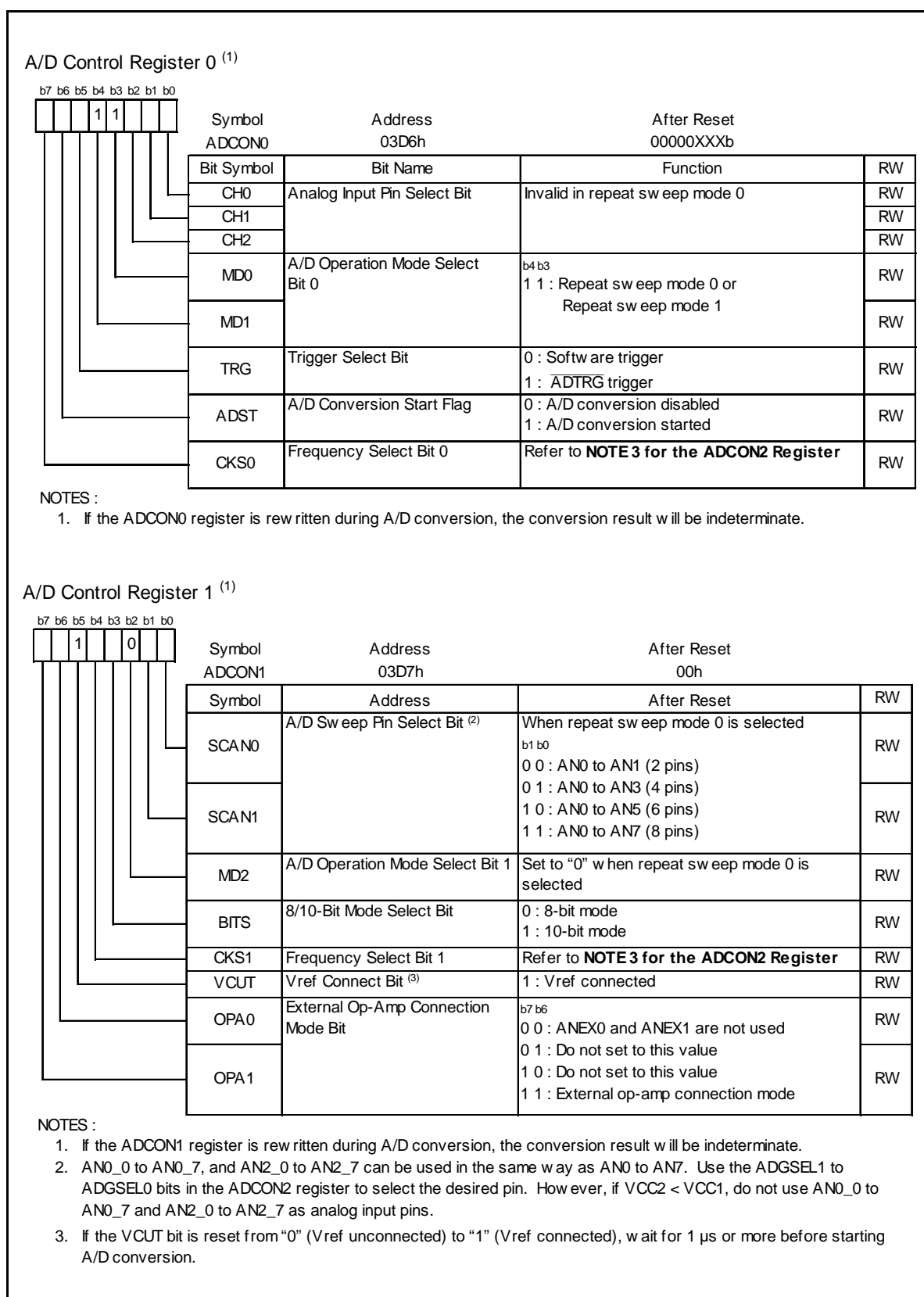


Figure 18.8 ADCON0 Register and ADCON1 Registers (Repeat Sweep Mode 0)

### 18.1.5 Repeat Sweep Mode 1

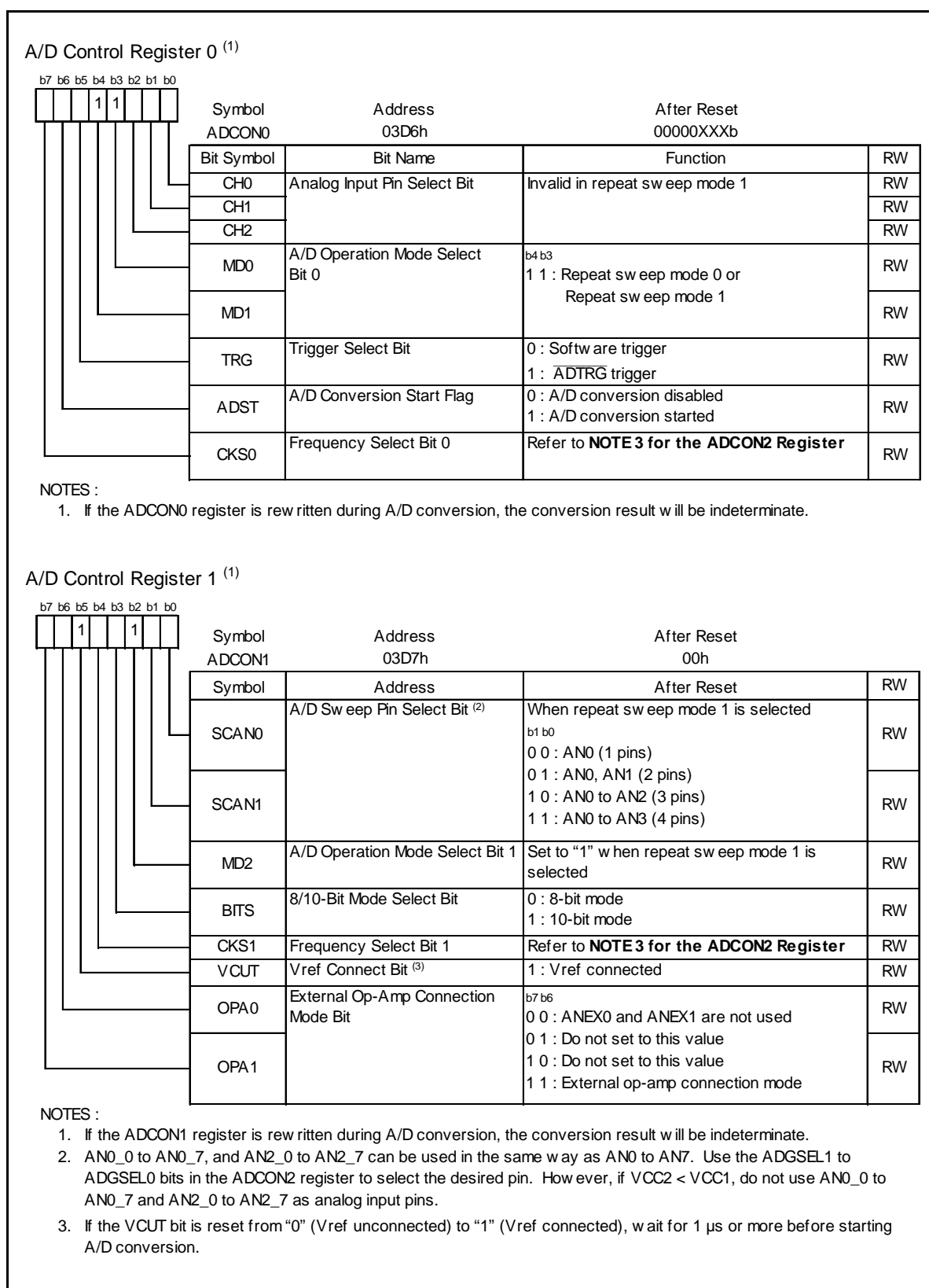
In repeat sweep mode 1, analog voltage selectively applied to all pins is repeatedly converted to a digital code. Table 18.6 shows the Repeat Sweep Mode 1 Specifications. Figure 18.9 shows the ADCON0 Register and ADCON1 Register (Repeat Sweep Mode 1).

**Table 18.6 Repeat Sweep Mode 1 Specifications**

Item	Specification
Function	The input voltages on all pins selected by the ADGSEL1 to ADGSEL0 bits in the ADCON2 register are A/D converted repeatedly, with priority given to pins selected by the SCAN1 to SCAN0 bits in the ADCON1 register and ADGSEL1 to ADGSEL0 bits. Example : If AN0 selected, input voltages are A/D converted in order of AN0→AN1→AN0→AN2→AN0→AN3, and so on.
A/D Conversion Start Condition	<ul style="list-style-type: none"> <li>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)</li> <li>When the TRG bit is "1" (ADTRG trigger) Input on the ADTRG pin changes state from high to low after the ADST bit is set to "1" (A/D conversion starts)</li> </ul>
A/D Conversion Stop Condition	Set the ADST bit to "0" (A/D conversion halted)
Interrupt Request Generation timing	None generated
Analog Input Pins to be Given Priority when A/D Converted	Select from AN0 (1 pin), AN0 to AN1 (2 pins), AN0 to AN2 (3 pins), AN0 to AN3 (4 pins) <sup>(1)</sup>
Reading of Result of A/D Converter	Read one of the AD0 to AD7 registers that corresponds to the selected pin

**NOTES:**

1. AN0\_0 to AN0\_7, and AN2\_0 to AN2\_7 can be used in the same way as AN0 to AN7. However, if VCC2 < VCC1, do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.

**Figure 18.9 ADCON0 Register and ADCON1 Register (Repeat Sweep Mode 1)**

## 18.2 Function

### 18.2.1 Resolution Select Function

The desired resolution can be selected using the BITS bit in the ADCON1 register. If the BITS bit is set to “1” (10-bit conversion accuracy), the A/D conversion result is stored in the bit 0 to bit 9 in the ADi register (i = 0 to 7). If the BITS bit is set to “0” (8-bit conversion accuracy), the A/D conversion result is stored in the bit 0 to bit 7 in the ADi register.

### 18.2.2 Sample and Hold

If the SMP bit in the ADCON2 register is set to “1” (with sample-and-hold), the conversion speed per pin is increased to 28  $\phi_{AD}$  cycles for 8-bit resolution or 33  $\phi_{AD}$  cycles for 10-bit resolution. Sample and Hold is effective in all operating modes. Select whether or not to use the Sample and Hold function before starting A/D conversion.

### 18.2.3 Extended Analog Input Pins

In one-shot and repeat modes, the ANEX0 and ANEX1 pins can be used as analog input pins. Use the OPA1 to OPA0 bits in the ADCON1 register to select whether or not use ANEX0 and ANEX1.

The A/D conversion results of ANEX0 and ANEX1 inputs are stored in the AD0 and AD1 registers, respectively.

### 18.2.4 External Operation Amplifier (Op-Amp) Connection Mode

Multiple analog inputs can be amplified using a single external op-amp via the ANEX0 and ANEX1 pins. Set the OPA1 to OPA0 bits in the ADCON1 register to “11b” (external op-amp connection mode). The inputs from ANi (i = 0 to 7) (1) are output from the ANEX0 pin. Amplify this output with an external op-amp before sending it back to the ANEX1 pin. The A/D conversion result is stored in the corresponding ADi register. The A/D conversion speed depends on the response characteristics of the external op-amp. Figure 18.10 shows an example of How to Connect the Pins in External Op-Amp.

1. AN0\_i and AN2\_i can be used the same as ANi. However, if  $VCC2 < VCC1$ , do not use AN0\_i and AN2\_i as analog input pins.

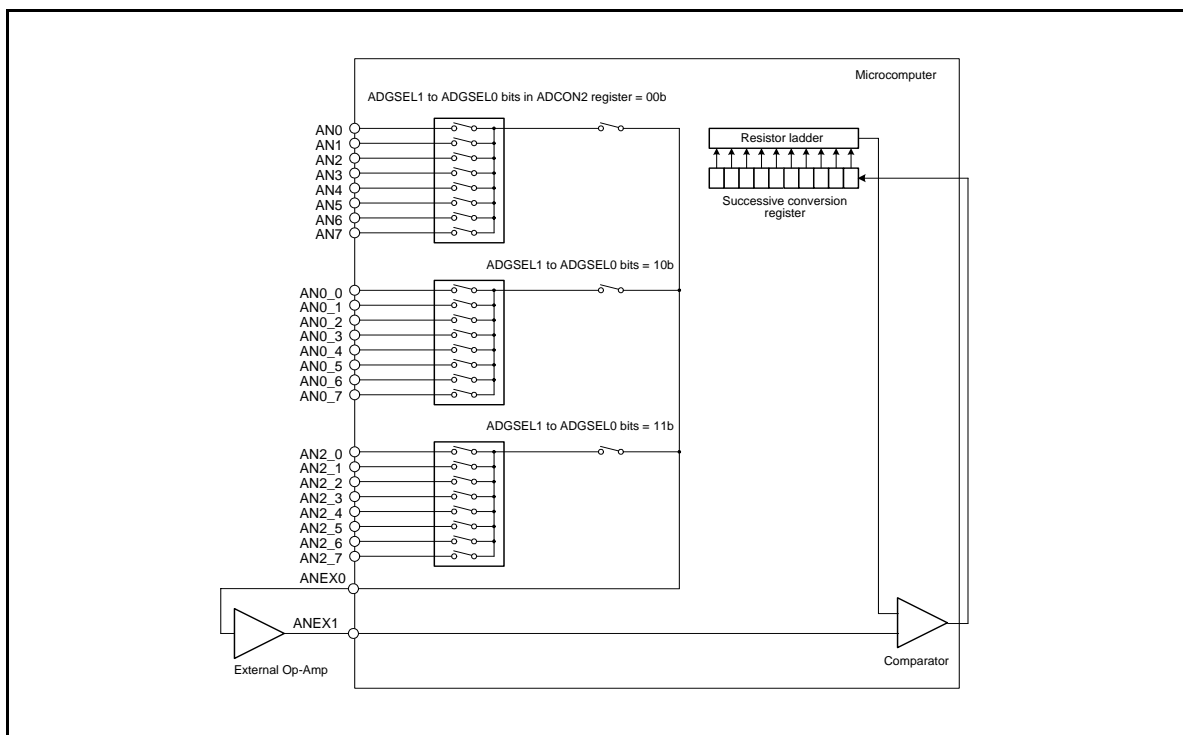


Figure 18.10 External Op-Amp Connection

### 18.2.5 18.2.5 Current Consumption Reducing Function

When not using the A/D converter, its resistor ladder and reference voltage input pin (VREF) can be separated using the VCUT bit in the ADCON1 register. When separated, no current will flow from the VREF pin into the resistor ladder, helping to reduce the power consumption of the chip.

To use the A/D converter, set the VCUT bit to "1" (Vref connected) and then set the ADST bit in the ADCON0 register to "1" (A/D conversion start). The VCUT and ADST bits cannot be set to "1" at the same time.

Nor can the VCUT bit be set to "0" (Vref unconnected) during A/D conversion.

Note that this does not affect VREF for the D/A converter (irrelevant).

### 18.2.6 Output Impedance of Sensor under A/D Conversion

To carry out A/D conversion properly, charging the internal capacitor C shown in Figure 18.11 has to be completed within a specified period of time. T (sampling time) as the specified time. Let output impedance of sensor equivalent circuit be R0, internal resistance of microcomputer be R, precision (error) of the A/D converter be X, and the resolution of A/D converter be Y (Y is 1024 in the 10-bit mode, and 256 in the 8-bit mode).

$$\text{VC is generally } VC = VIN \left\{ 1 - e^{-\frac{1}{C(R0 + R)} t} \right\}$$

$$\text{And when } t = T, \quad VC = VIN - \frac{X}{Y} VIN = VIN \left( 1 - \frac{X}{Y} \right)$$

$$e^{-\frac{1}{C(R0 + R)} T} = \frac{X}{Y}$$

$$-\frac{1}{C(R0 + R)} T = \ln \frac{X}{Y}$$

$$\text{Hence, } R0 = -\frac{T}{C \cdot \ln \frac{X}{Y}} - R$$

Figure 18.11 shows Analog Input Pin and External Sensor Equivalent Circuit. When the difference between VIN and VC becomes 0.1LSB, we find impedance R0 when voltage between pins VC changes from 0 to VIN- (0.1/1024) VIN in time T. (0.1/1024) means that A/D precision drop due to insufficient capacitor charge is held to 0.1LSB at time of A/D conversion in the 10-bit mode. Actual error however is the value of absolute precision added to 0.1LSB.

When f(φAD) = 10 MHz, T = 0.3 μs in the A/D conversion mode with sample & hold. Output impedance R0 for sufficiently charging capacitor C within time T is determined as follows.

T = 0.3 μs, R = 7.8 kΩ, C = 1.5 pF, X = 0.1, and Y = 1024. Hence,

$$R0 = -\frac{0.3 \times 10^{-6}}{1.5 \times 10^{-12} \cdot \ln \frac{0.1}{1024}} - 7.8 \times 10^3 = 13.9 \times 10^3$$

Thus, the allowable output impedance of the sensor equivalent circuit, making the precision (error) 0.1LSB or less, is approximately 13.9 kΩ maximum.



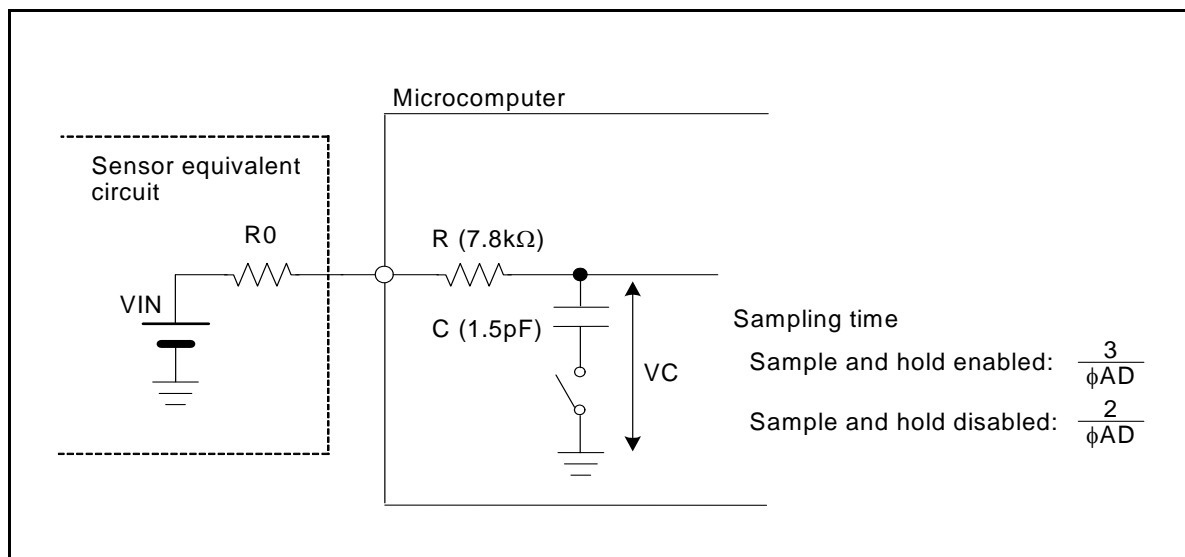


Figure 18.11 Analog Input Pin and External Sensor Equivalent Circuit

## 19. D/A Converter

This is an 8-bit, R-2R type D/A converter. These are two independent D/A converters.

D/A conversion is performed by writing to the DA<sub>i</sub> register (i = 0 to 1). To output the result of conversion, set the DA<sub>i</sub>E bit in the DACON register to “1” (output enabled). Before D/A conversion can be used, the corresponding port direction bit must be cleared to “0” (input mode). Setting the DA<sub>i</sub>E bit to “1” removes a pull-up from the corresponding port.

Output analog voltage (V) is determined by a set value (n : decimal) in the DA<sub>i</sub> register.

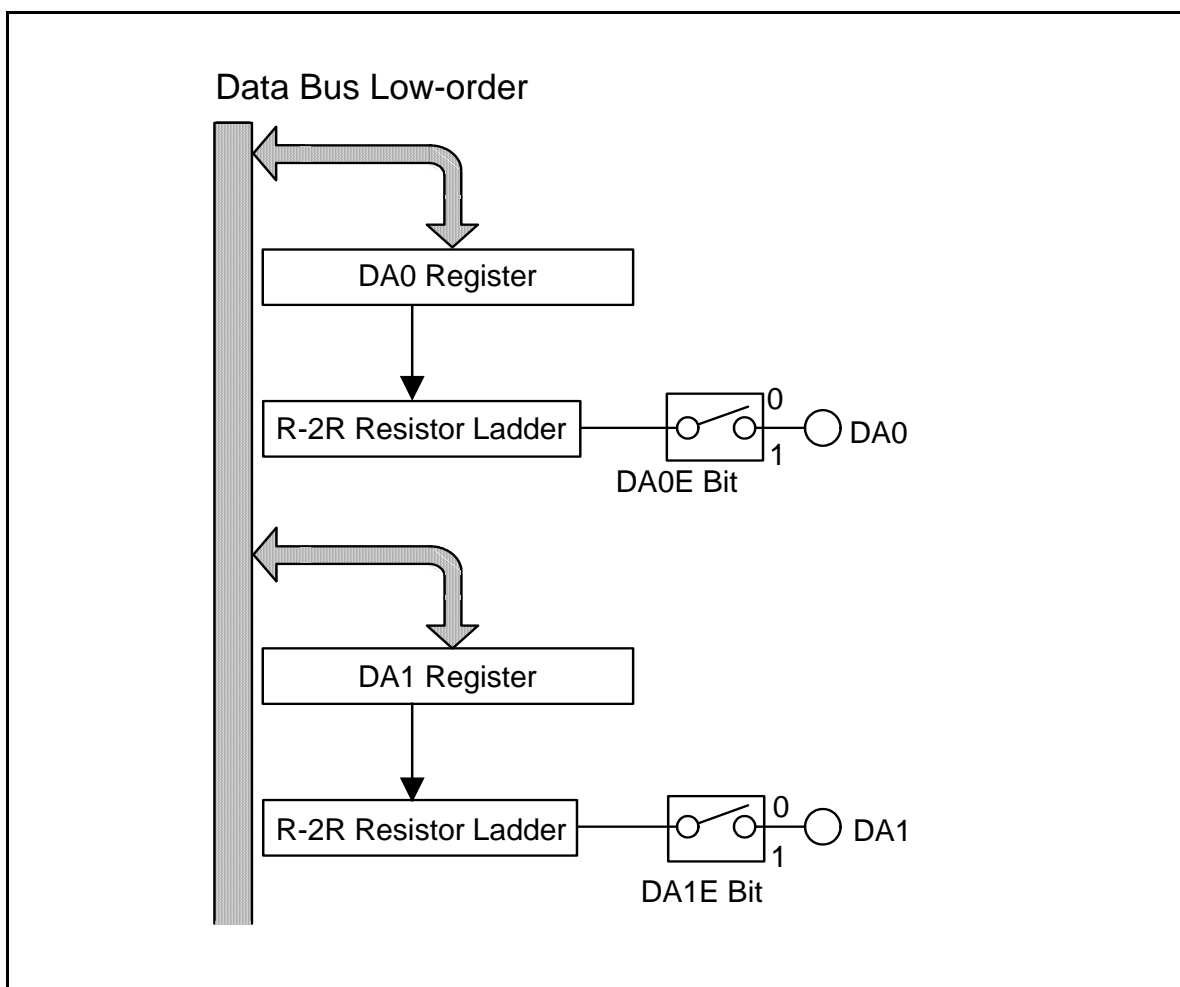
$$V = V_{REF} \times n / 256 \quad (n = 0 \text{ to } 255)$$

V<sub>REF</sub> : reference voltage

Table 19.1 lists the D/A Converter Performance. Figure 19.1 shows the D/A Converter Block Diagram. Figure 19.2 shows the D/A converter related registers. Figure 19.3 shows the D/A Converter Equivalent Circuit.

**Table 19.1 D/A Converter Performance**

Item	Performance
D/A Conversion Method	R-2R method
Resolution	8 bits
Analog Output Pin	2 channels (DA0 and DA1)



**Figure 19.1 D/A Converter Block Diagram**

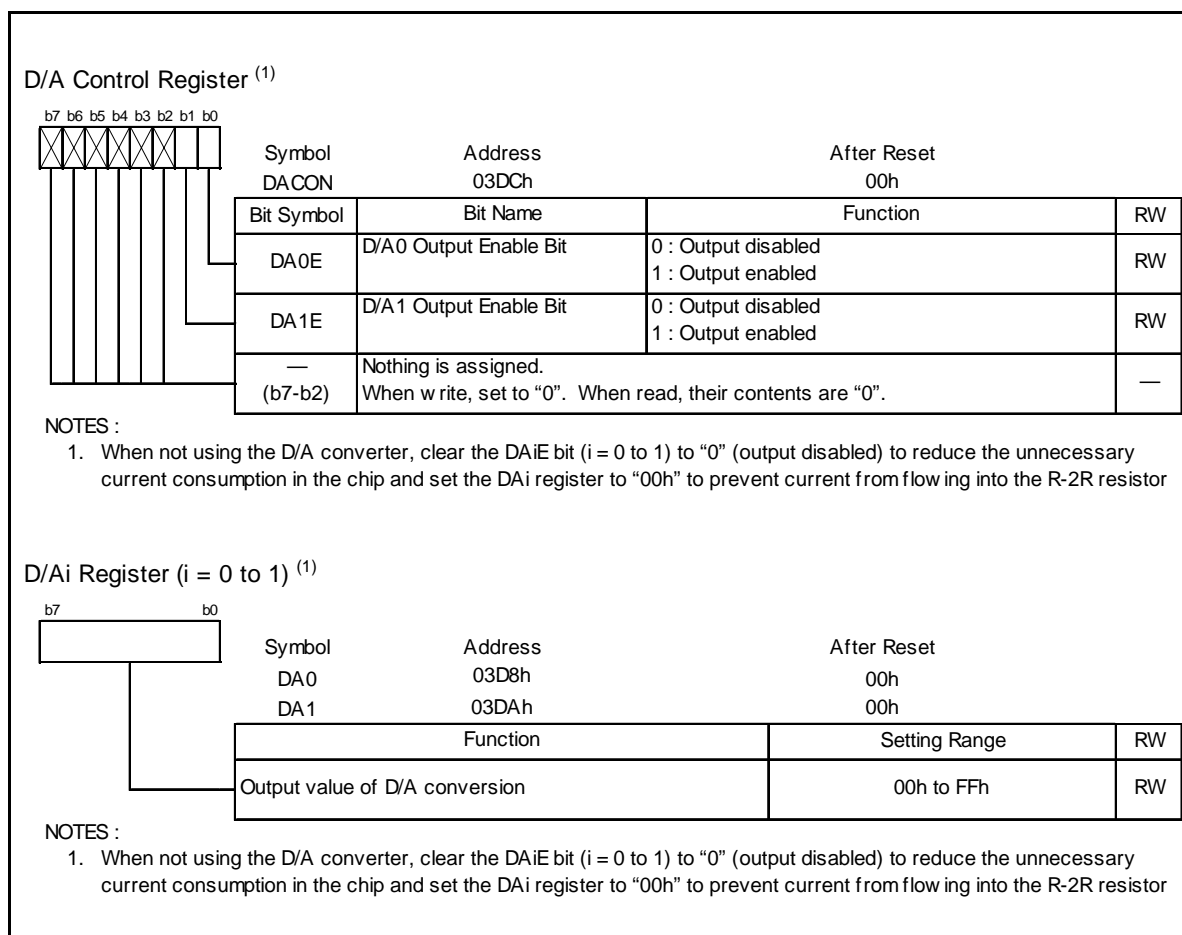


Figure 19.2 DACON, DA0 and DA1 Registers

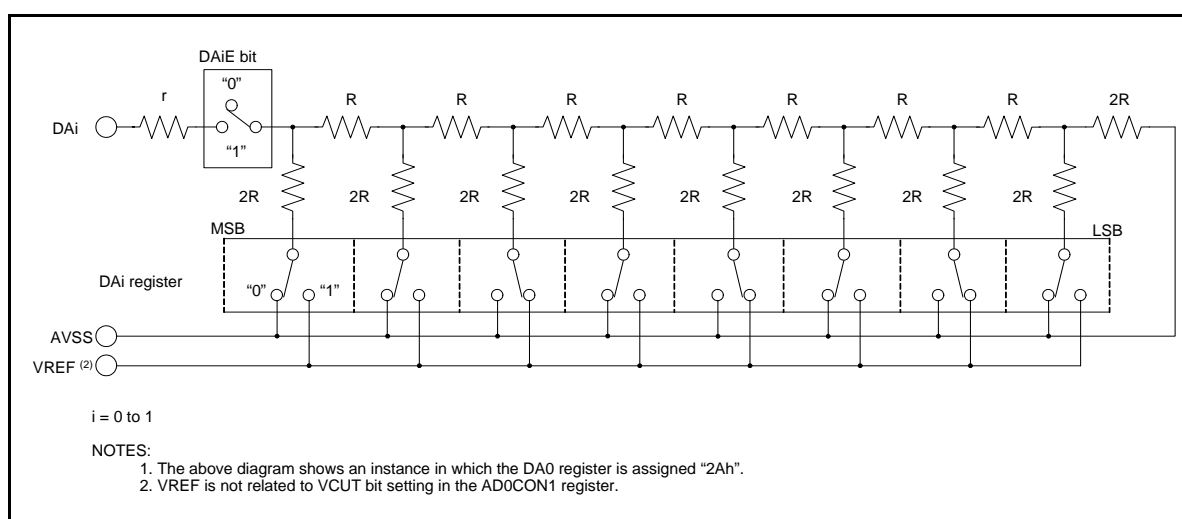


Figure 19.3 D/A Converter Equivalent Circuit

20. CRC Calculation

The Cyclic Redundancy Check (CRC) operation detects an error in data blocks. The microcomputer uses a generator polynomial of CRC\_CCITT ( $X^{16} + X^{12} + X^5 + 1$ ) to generate CRC code.

The CRC code consists of 16 bits which are generated for each data block in given length, separated in 8 bit units.

After the initial value is set in the CRCD register, the CRC code is set in that register each time one byte of data is written to the CRCIN register. CRC code generation for one-byte data is finished in two cycles.

Figure 20.1 shows the CRC Circuit Block Diagram. Figure 20.2 shows the CRC-related Registers.

Figure 20.3 shows the Calculation Example using the CRC Operation.

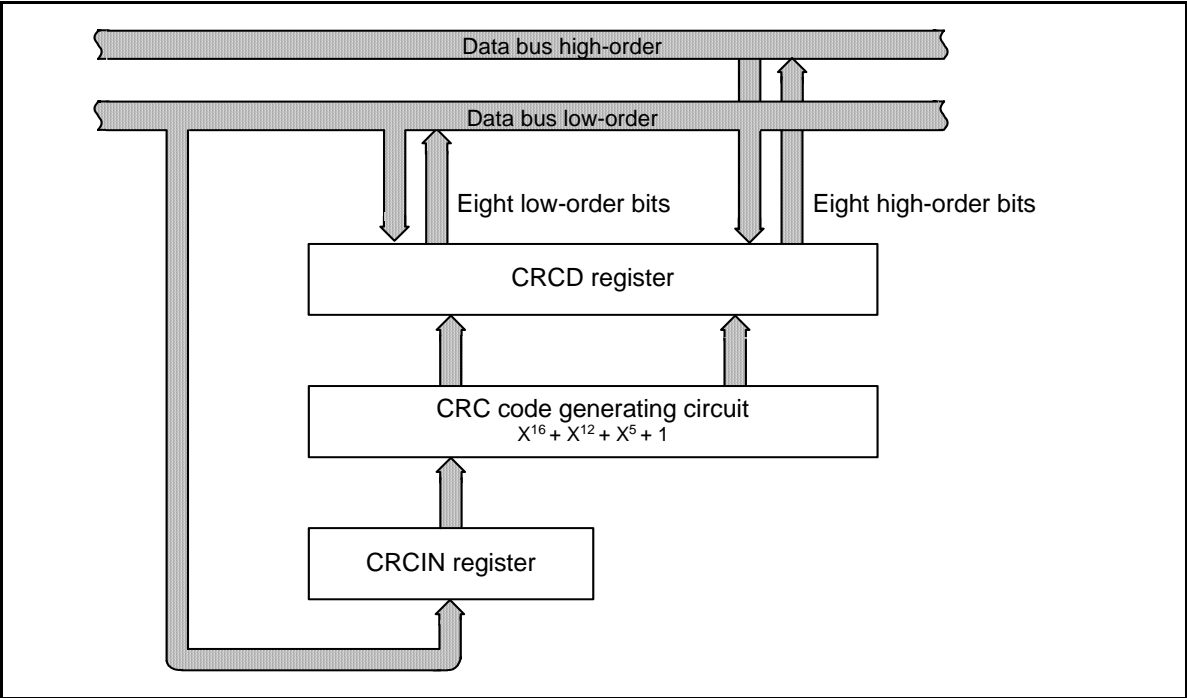


Figure 20.1 CRC Circuit Block Diagram

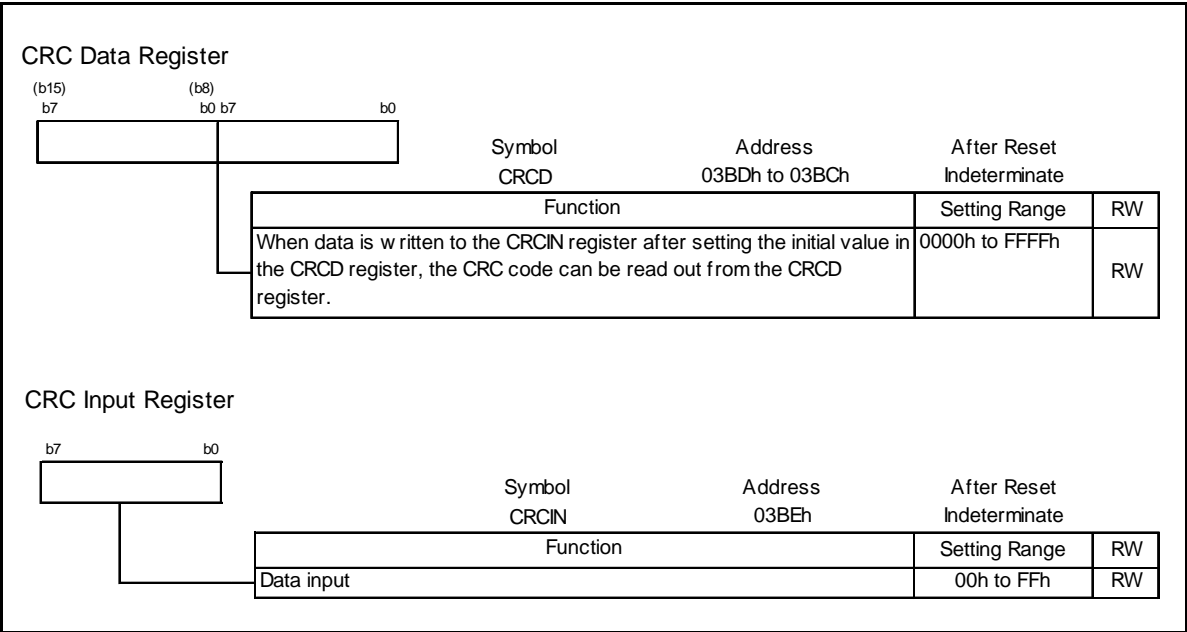


Figure 20.2 CRCD and CRCIN Registers

### Setup procedure and CRC operation when generating CRC code "80C4h"

- CRC operation performed by the M16C

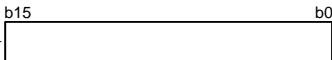
CRC code: Remainder of a division in which the value written to the CRCIN register with its bit positions reversed is divided by the generator polynomial

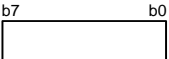
Generator polynomial:  $X^{16} + X^{12} + X^5 + 1$  (1 0001 0000 0010 0001b)

- Setting procedure

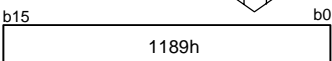
(1) Reverse the bit positions of the value "80C4h" by program in 1-byte units.

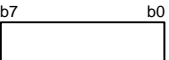
"80h" → "01h", "C4h" → "23h"

(2) Write 0000h (initial value) →  CRCD register

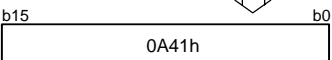
(3) Write 01h →  CRCIN register

Two cycles later, the CRC code for "80h", i.e., 9188h, has its bit positions reversed to become "1189h" which is stored in the CRCD register.

 CRCD register

(4) Write 23h →  CRCIN register

Two cycles later, the CRC code for "80C4h", i.e., 8250h, has its bit positions reversed to become "0A41h" which is stored in the CRCD register.

 CRCD register

- Details of CRC operation

As shown in (3) above, bit position of "01h" (00000001b) written to the CRCIN register is inversed and becomes "10000000b".

Add "1000 0000 0000 0000 0000 0000b", as "10000000b" plus 16 digits, to "0000 0000 0000 0000 0000 0000b", as "0000 0000 0000 0000b" plus 8 digits as the default value of the CRCD register to perform the modulo-2 division.

$$\begin{array}{r}
 \text{Generator polynomial} \quad 1 \ 0001 \ 0000 \ 0010 \ 0001 \quad | \quad \begin{array}{r} 1000 \ 1000 \\ 1000 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000 \\ 1000 \ 1000 \ 0001 \ 0000 \ 1 \\ 1000 \ 0001 \ 0000 \ 1000 \ 0 \\ 1000 \ 1000 \ 0001 \ 0000 \ 1 \\ 1001 \ 0001 \ 1000 \ 1000 \end{array} \\
 \hline
 \text{CRC code} \quad 1001 \ 0001 \ 1000 \ 1000
 \end{array}$$

Data

Modulo-2 operation is operation that complies with the law given below.

$$\begin{array}{l}
 0 + 0 = 0 \\
 0 + 1 = 1 \\
 1 + 0 = 1 \\
 1 + 1 = 0 \\
 -1 = 1
 \end{array}$$

"0001 0001 1000 1001b (1189h)", the remainder "1001 0001 1000 1000b (9188h)" with inversed bit position, can be read from the CRCD register.

When going on to (4) above, "23h (00100011b)" written in the CRCIN register is inversed and becomes "11000100b".

Add "1100 0100 0000 0000 0000 0000b", as "11000100b" plus 16 digits, to "1001 0001 1000 1000 0000 0000b", as "1001 0001 1000 1000b" plus 8 digits as a remainder of (3) left in the CRCD register to perform the modulo-2 division. "0000 1010 0100 0001b (0A41h)", the remainder with inversed bit position, can be read from CRCD register.

Figure 20.3 CRC Calculation

## 21. Programmable I/O Ports

### Note

There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in the M16C/62P (80-pin version) and the M16C/62PT (80-pin version). Set the direction bits in these ports to “1” (output mode), and set the output data to “0” (“L”) using the program.

Moreover, P11\_0 to P11\_7, P12\_0 to P12\_7, P13\_0 to P13\_7, P14\_0 and P14\_1 pins do not exist. Therefore, P11 to P13, PC14 and PUR13 register do not exist.

The programmable input/output ports (hereafter referred to simply as I/O ports) consist of 113 lines P0 to P14 for the 128-pin version, 87 lines P0 to P10 for the 100-pin version, or 70 lines P0 to P10 for the 80-pin version. Each port can be set for input or output every line by using a direction register, and can also be chosen to be or not be pulled high every 4 lines. P8\_5 is an input-only port and does not have a pull-up resistor. Port P8\_5 shares the pin with  $\overline{\text{NMI}}$ , so that the  $\overline{\text{NMI}}$  input level can be read from the P8 register P8\_5 bit.

Table 21.1 lists the Number of Pins of the Programmable I/O Ports of Each Package. Figure 21.1 to Figure 21.5 show the I/O ports. Figure 21.6 shows the I/O Pins.

Each pin functions as an I/O port, a peripheral function input/output, or a bus control pin.

For details on how to set peripheral functions, refer to each functional description in this manual. If any pin is used as a peripheral function input or D/A converter output pin, set the direction bit for that pin to “0” (input mode). Any pin used as an output pin for peripheral functions other than the D/A converter is directed for output no matter how the corresponding direction bit is set.

When using any pin as a bus control pin, refer to **8.2 Bus Control**.

P0 to P5, P12, and P13 are capable of VCC2-level input/output; P6 to P11 and P14 are capable of VCC1- level input/output.

**Table 21.1 Number of Pins of the Programmable I/O Ports of Each Package**

	128-pin Version	100-pin Version	80-pin version <sup>(1)</sup>
Programmable I/O Ports	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7 (P8_5 is an input port), P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7 (P8_5 is an input port), P9_0 to P9_7, P10_0 to P10_7,	P0_0 to P0_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_3, P5_0 to P5_7, P6_0 to P6_7, P7_0, P7_1, P7_6, P7_7, P8_0 to P8_4, P8_6, P8_7 (P8_5 is an input port), P9_0, P9_2 to P9_7, P10_0 to P10_7,
Total	113 pins	87 pins	70 pins

### NOTES:

1. There is no connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.

### 21.1 Port Pi Direction Register (PDi Register, i = 0 to 13)

Figure 21.7 shows the PDi Registers.

This register selects whether the I/O port is to be used for input or output. The bits in this register correspond one for one to each port.

During memory extension and microprocessor modes, the PDi registers for the pins functioning as bus control pins (A0 to A19, D0 to D15,  $\overline{CS0}$  to  $\overline{CS3}$ ,  $\overline{RD}$ ,  $\overline{WRL}/\overline{WR}$ ,  $\overline{WRH}/\overline{BHE}$ , ALE,  $\overline{RDY}$ ,  $\overline{HOLD}$ ,  $\overline{HLDA}$ , and BCLK) cannot be modified.

No direction register bit for P8\_5 is available.

### 21.2 Port Pi Register (Pi Register, i = 0 to 13)

Figure 21.8 shows the Pi Registers.

Data input/output to and from external devices are accomplished by reading and writing to the Pi register.

The Pi register consists of a port latch to hold the input/output data and a circuit to read the pin status. For ports set for 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 for 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. The bits in the Pi register correspond one for one to each port.

During memory extension and microprocessor modes, the PDi registers for the pins functioning as bus control pins (A0 to A19, D0 to D15,  $\overline{CS0}$  to  $\overline{CS3}$ ,  $\overline{RD}$ ,  $\overline{WRL}/\overline{WR}$ ,  $\overline{WRH}/\overline{BHE}$ , ALE,  $\overline{RDY}$ ,  $\overline{HOLD}$ ,  $\overline{HLDA}$ , and BCLK) cannot be modified.

### 21.3 Pull-up Control Register 0 to Pull-up Control Register 3 (PUR0 to PUR3 Registers)

Figure 21.9 and Figure 21.11 shows the PURi Registers.

The PUR0 to PUR2 register bits can be used to select whether or not to pull the corresponding port high in 4 bit units. The port chosen to be pulled high has a pull-up resistor connected to it when the direction bit is set for input mode. To use ports P11 to P14, set the PU37 bit in the PUR3 register to "1".

However, the pull-up control register has no effect on P0 to P3, P4\_0 to P4\_3, and P5 during memory extension and microprocessor modes. Although the register contents can be modified, no pull-up resistors are connected.

### 21.4 Port Control Register (PCR Register)

Figure 21.12 shows the PCR Register.

When the P1 register is read after setting the PCR0 bit in the PCR register to "1", the corresponding port latch can be read no matter how the PD1 register is set.

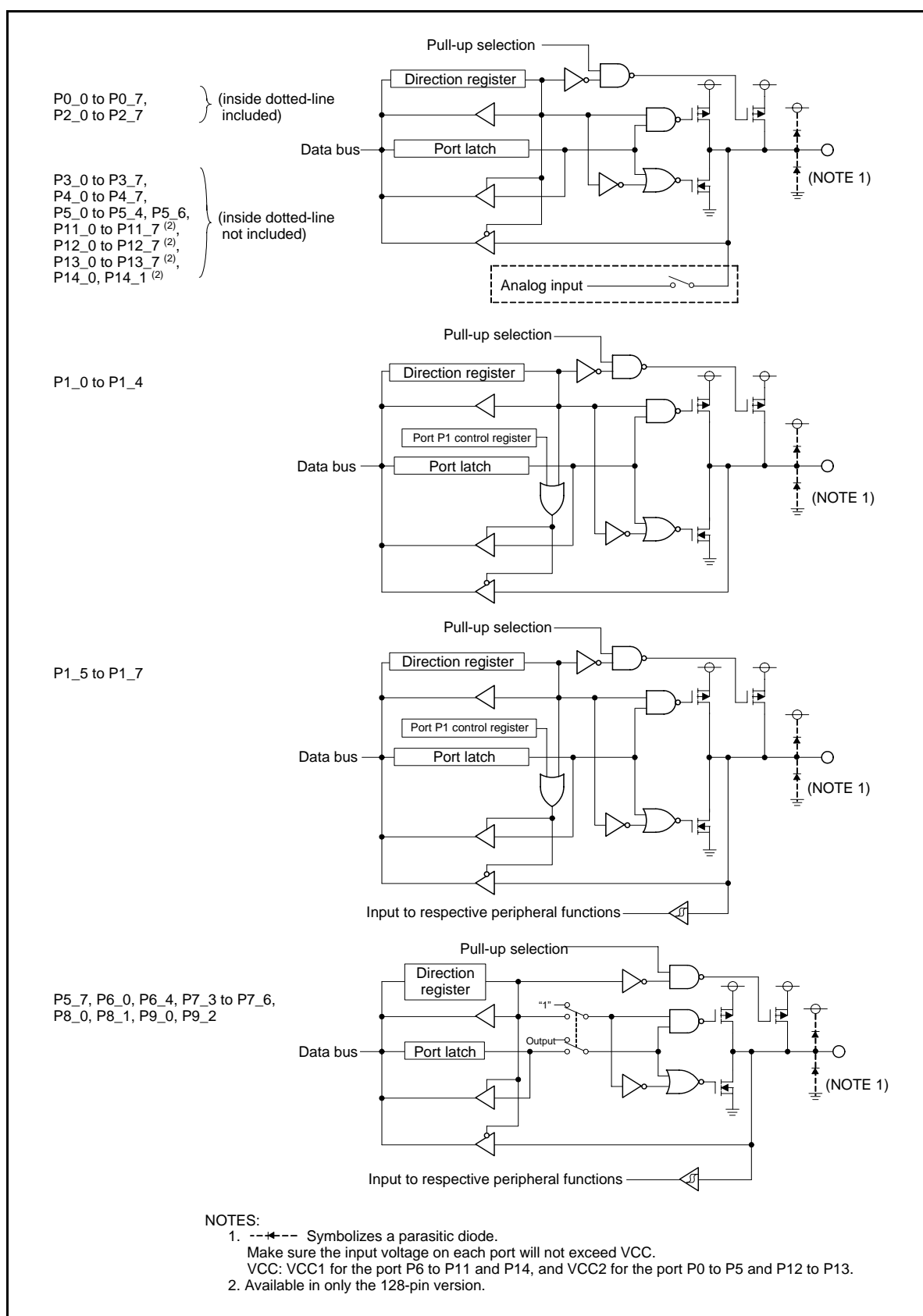


Figure 21.1 I/O Ports (1)



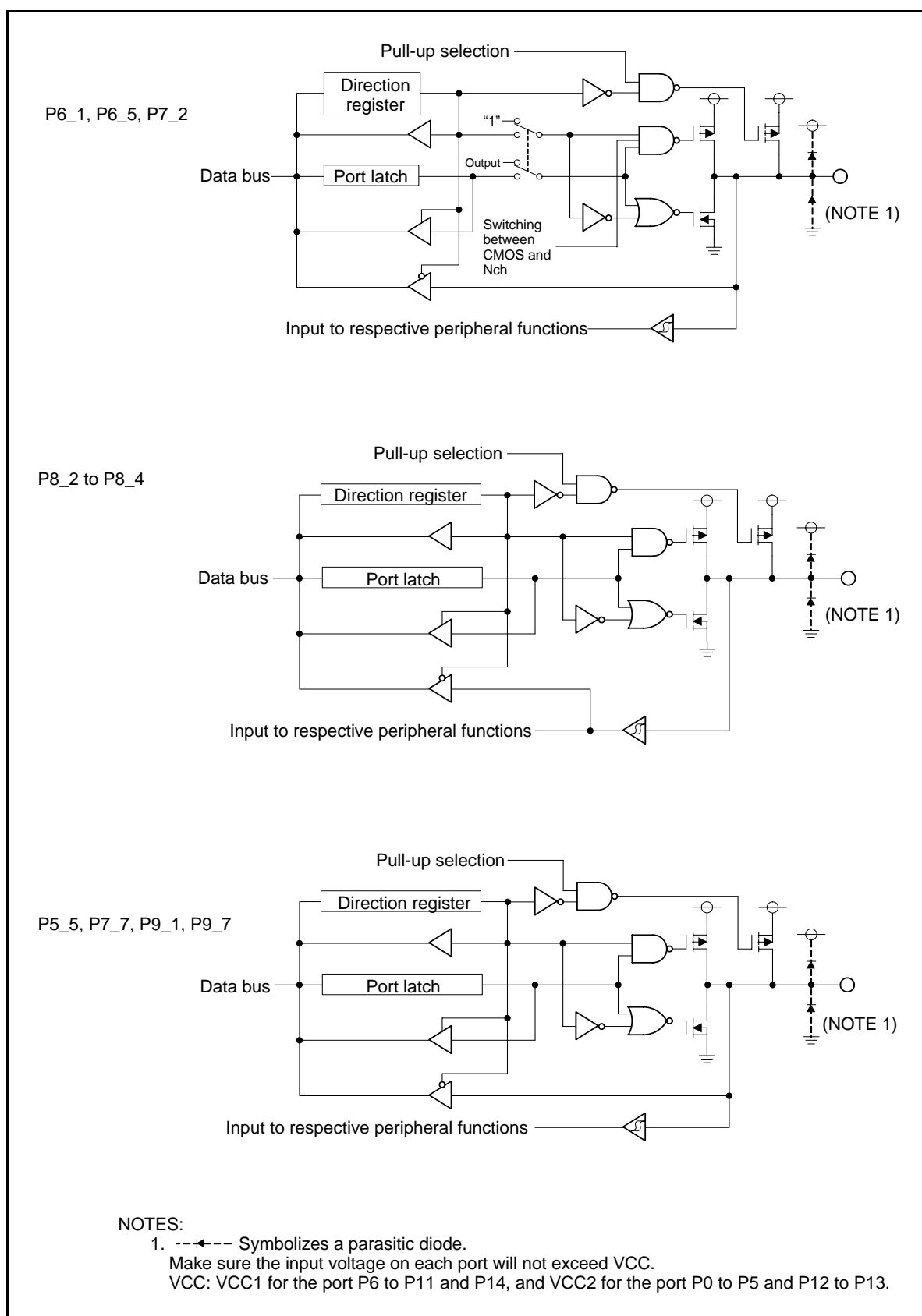


Figure 21.2 I/O Ports (2)

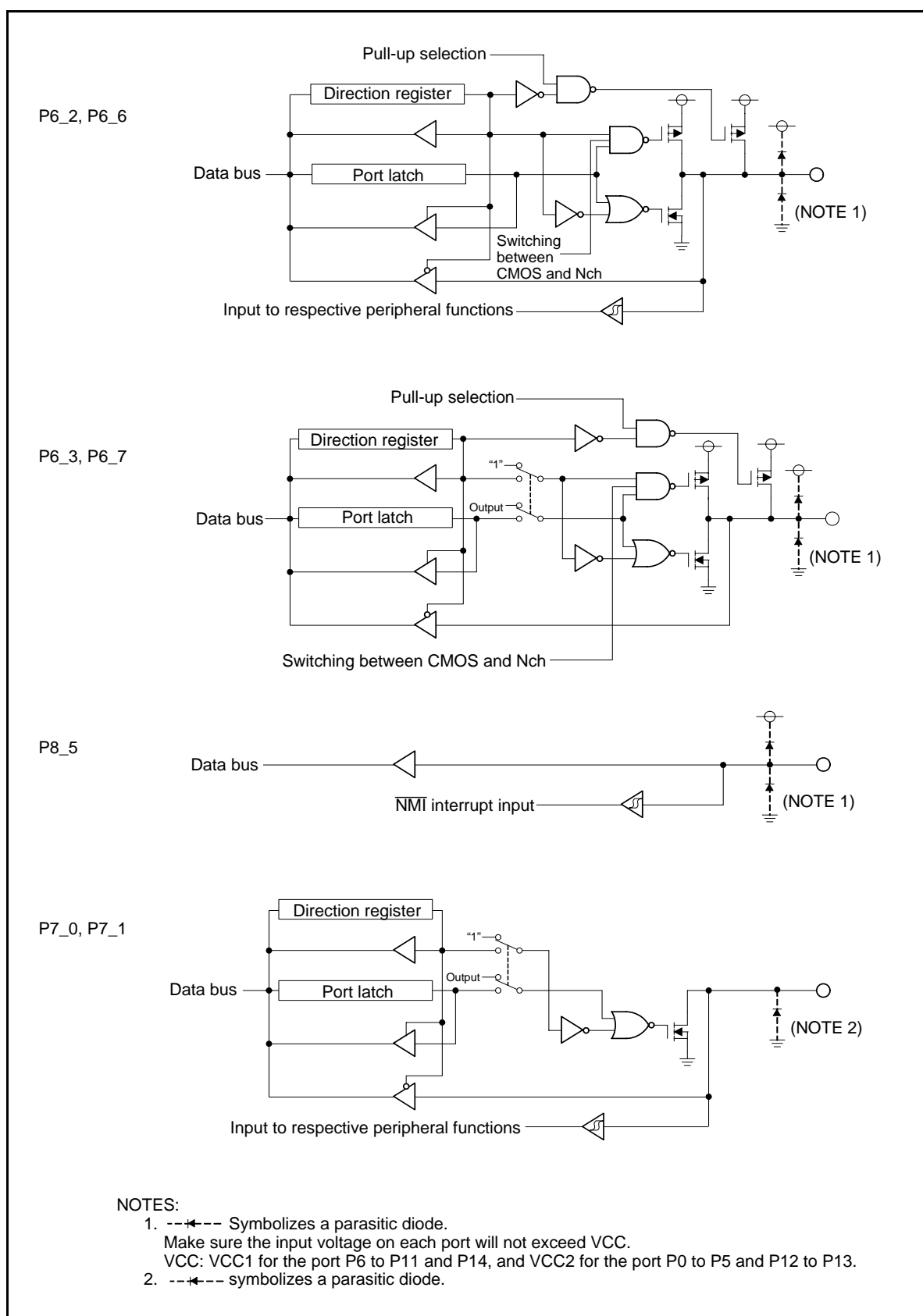


Figure 21.3 I/O Ports (3)

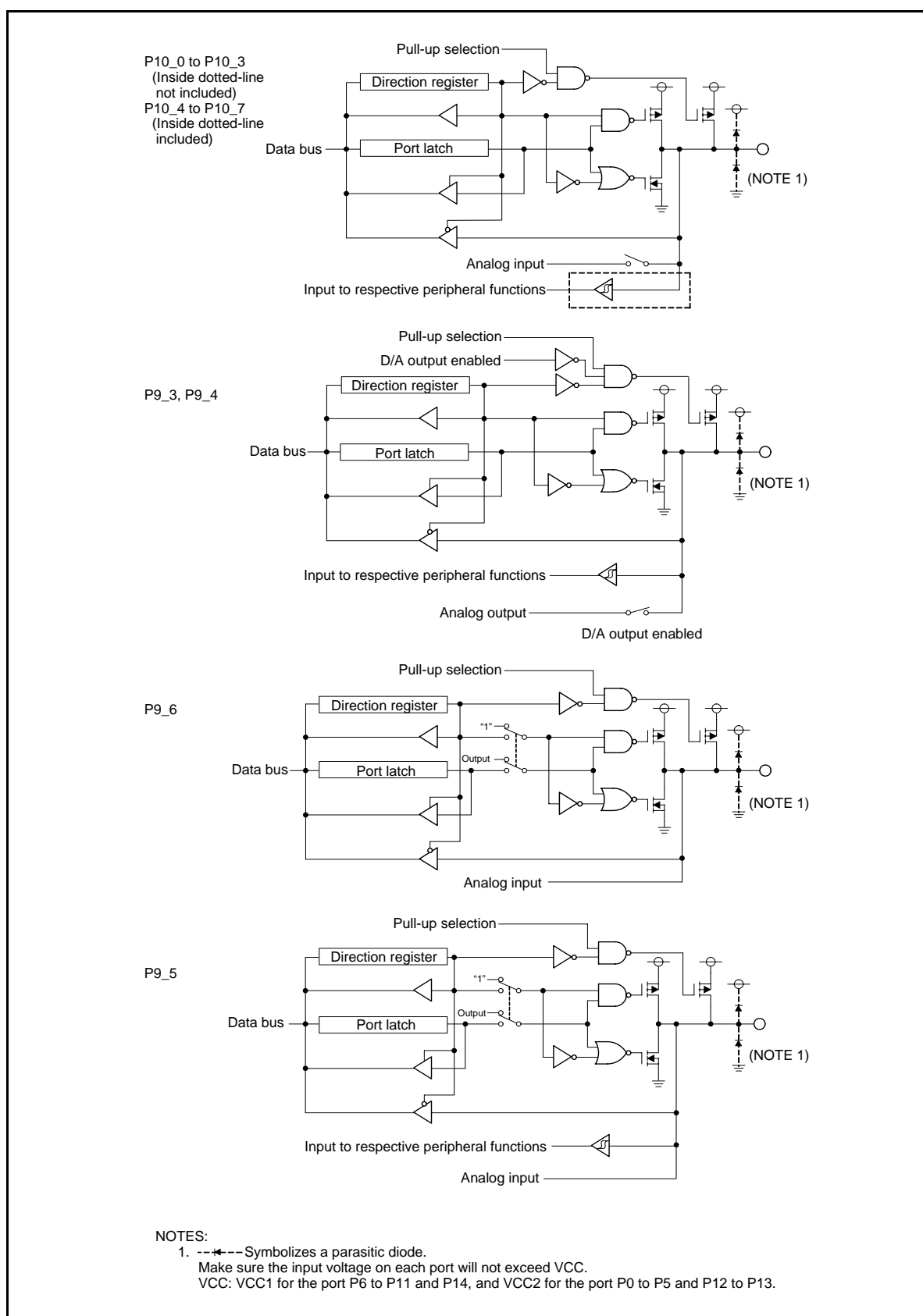
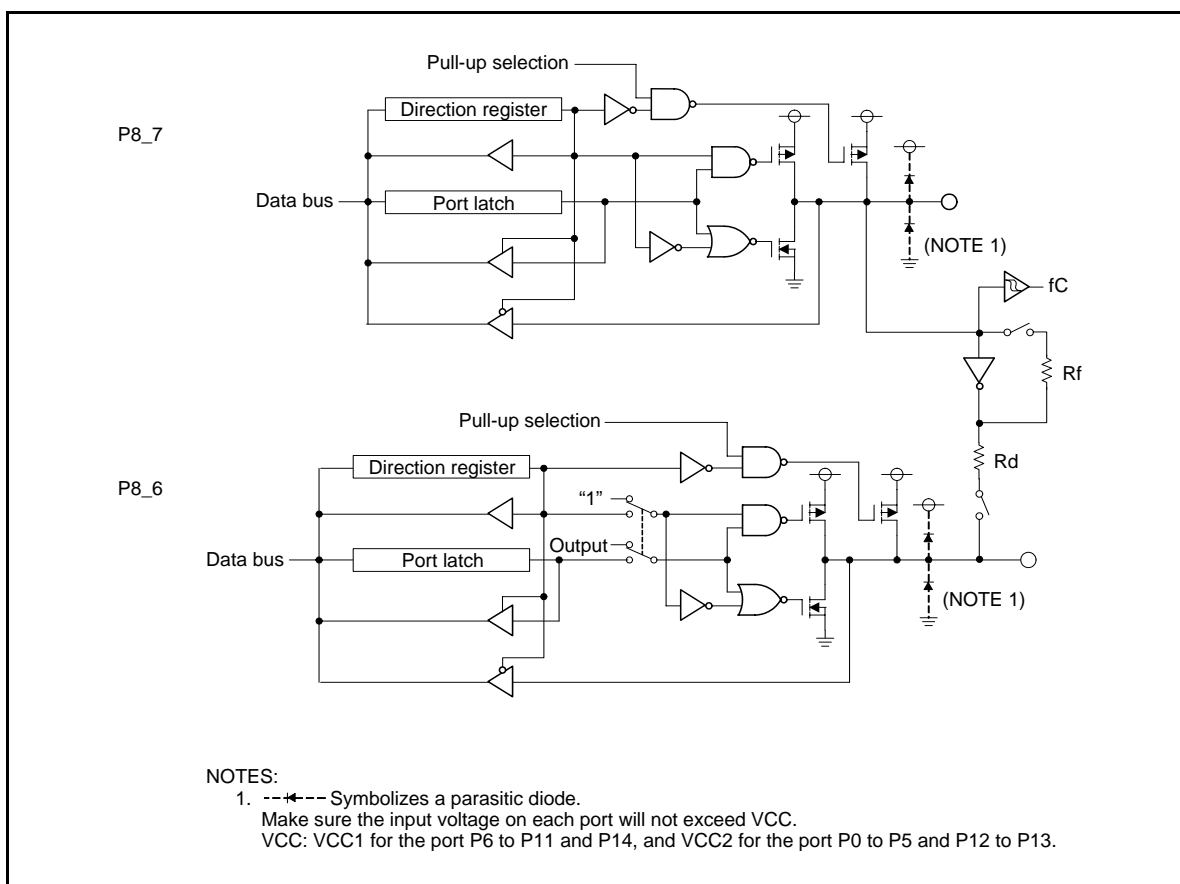
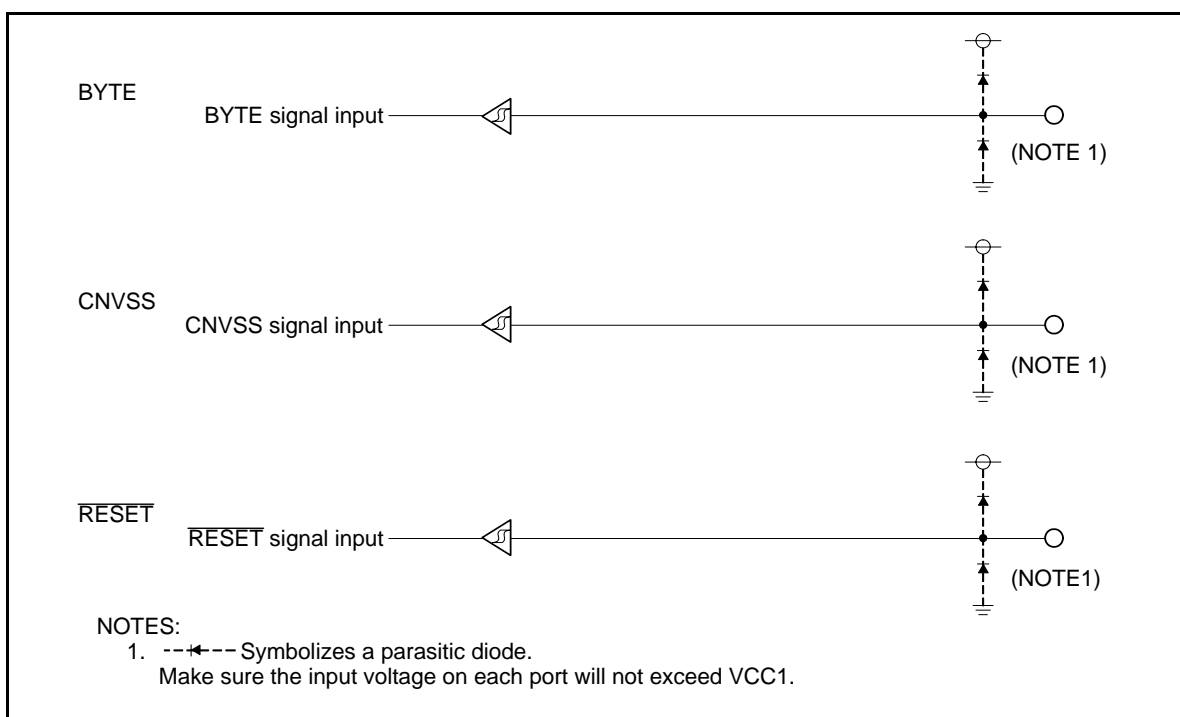


Figure 21.4 I/O Ports (4)



**Figure 21.5 I/O Ports (5)**



**Figure 21.6 I/O Pins**

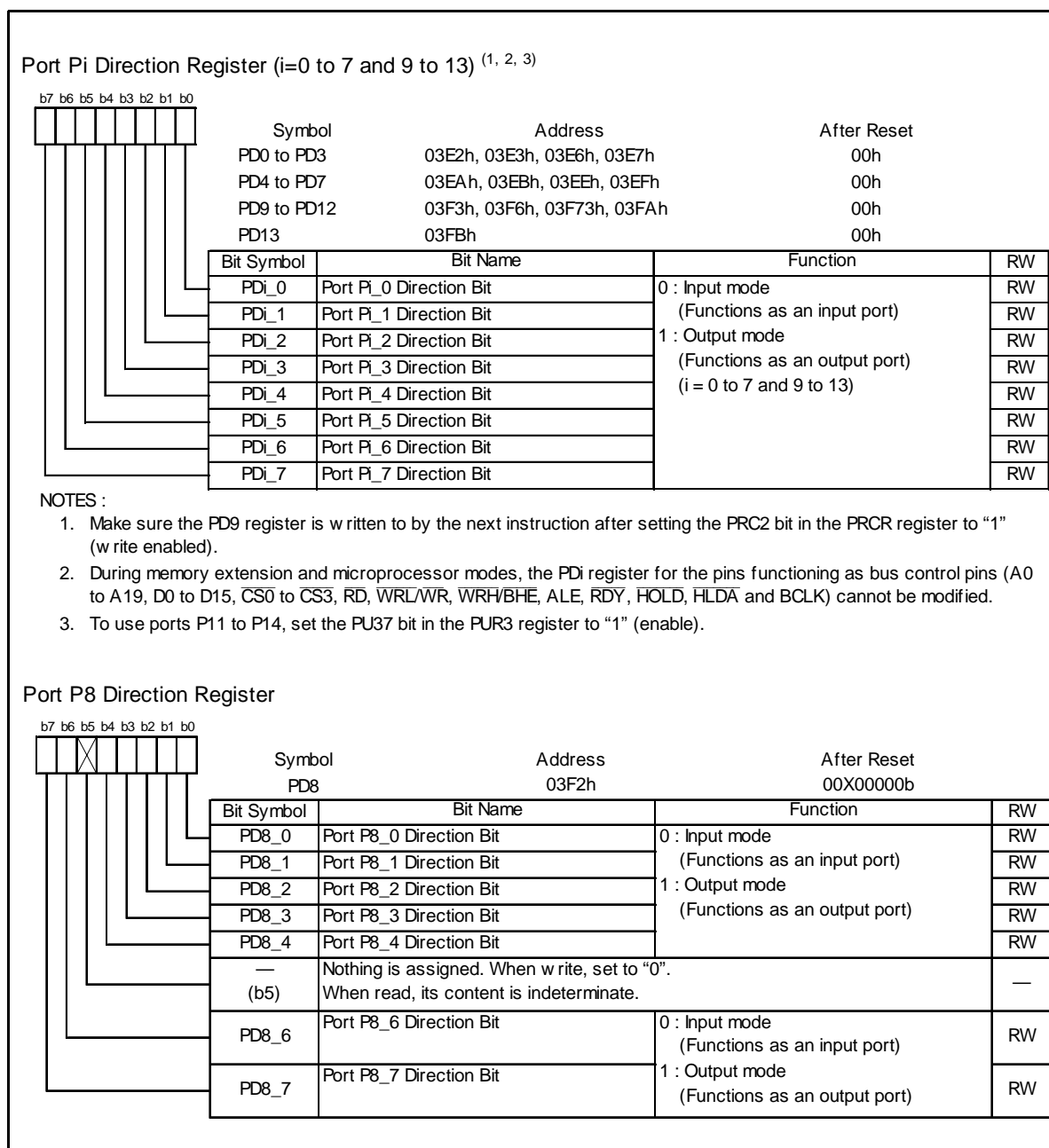


Figure 21.7 PDi Registers

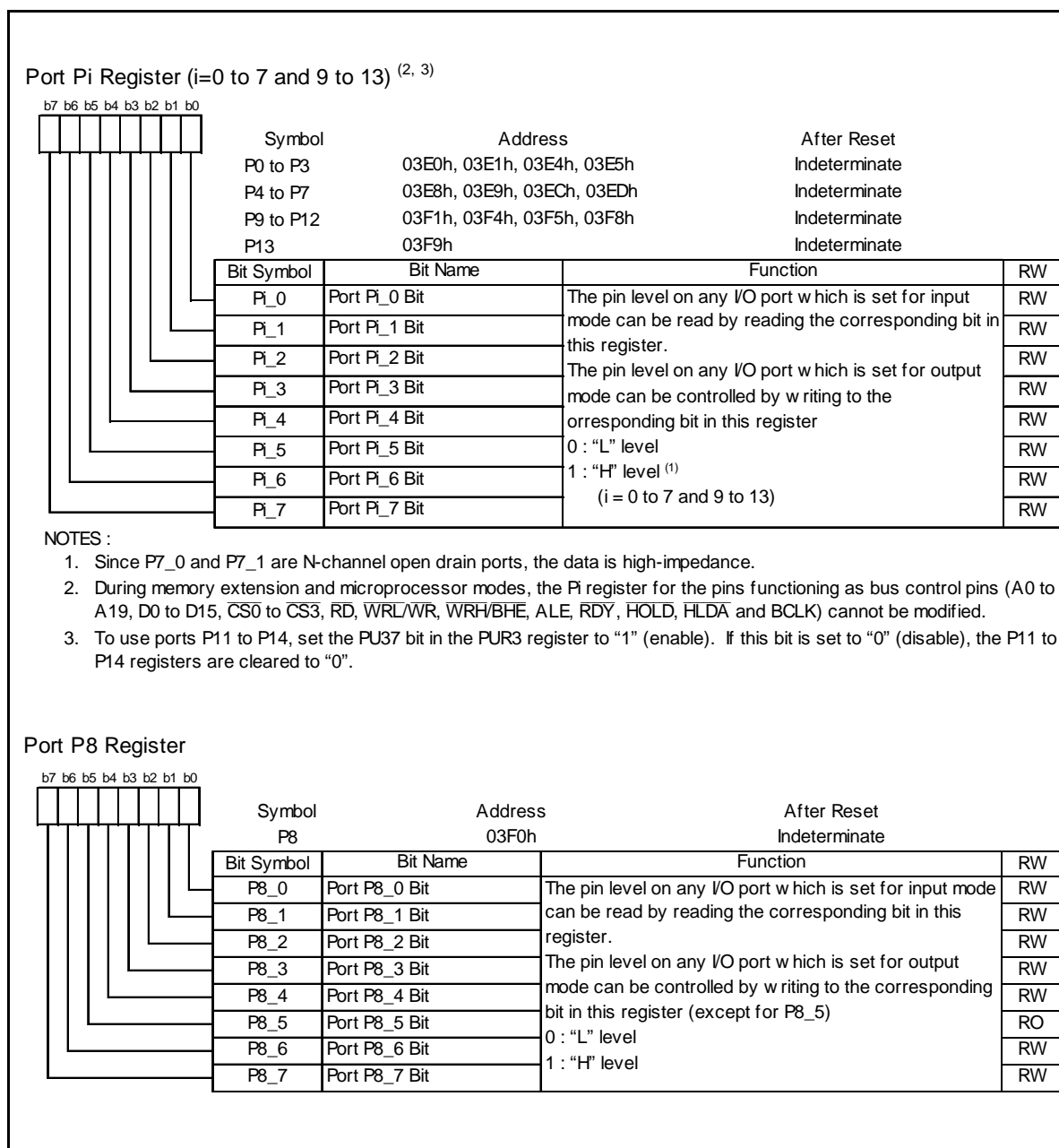


Figure 21.8 Pi Registers

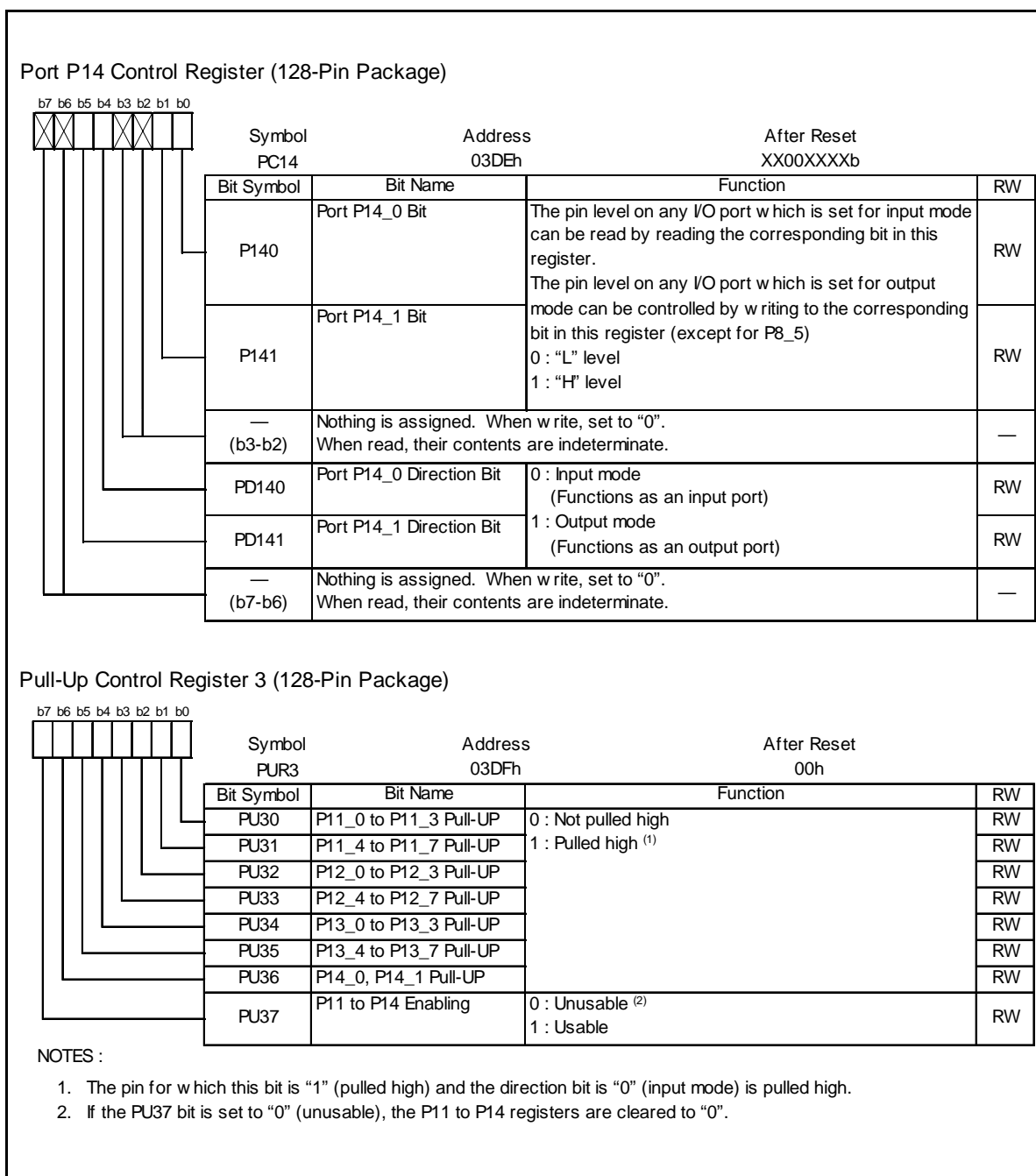


Figure 21.9 PC14 and PUR3 Registers

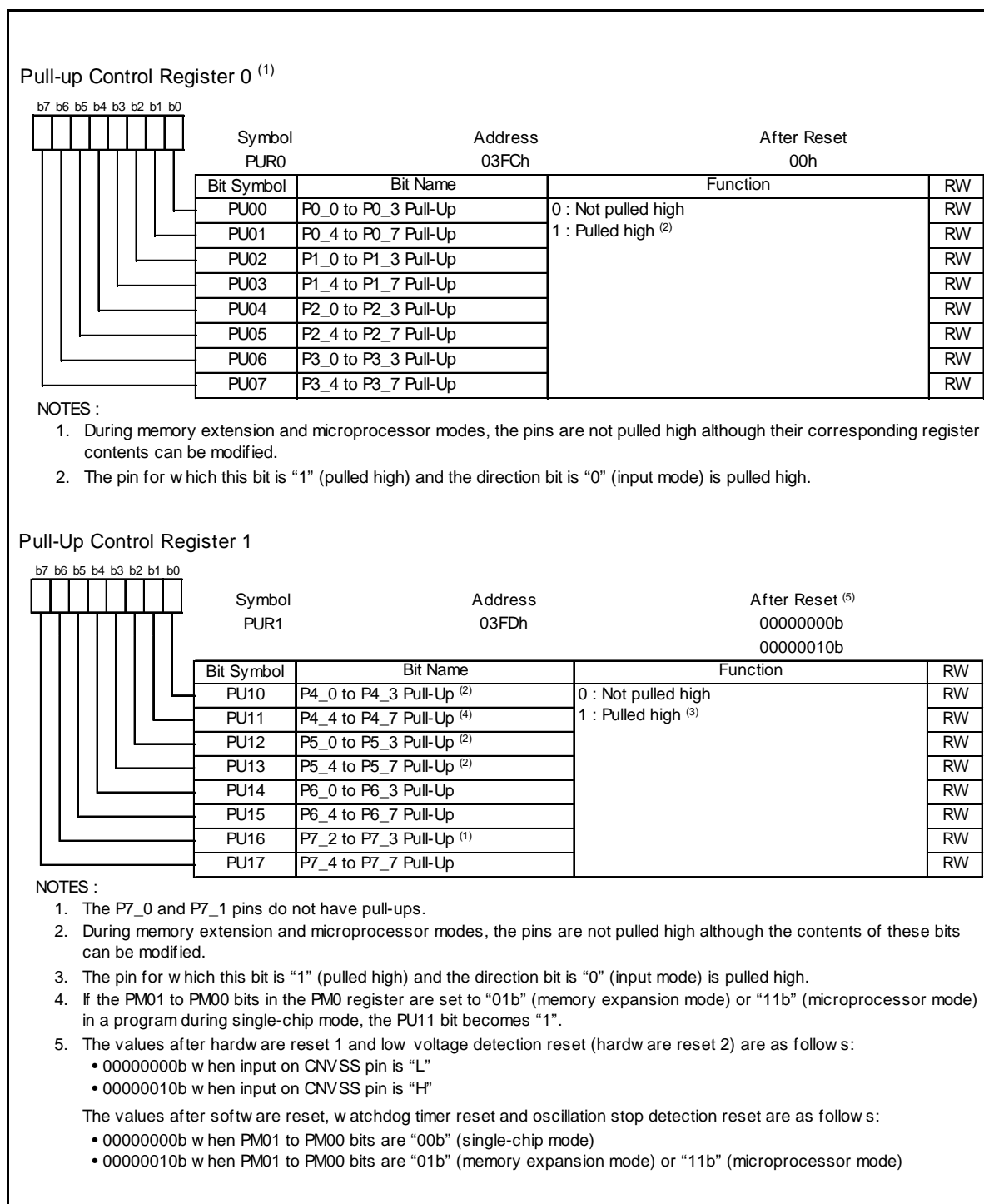


Figure 21.10 PUR0 and PUR1 Registers



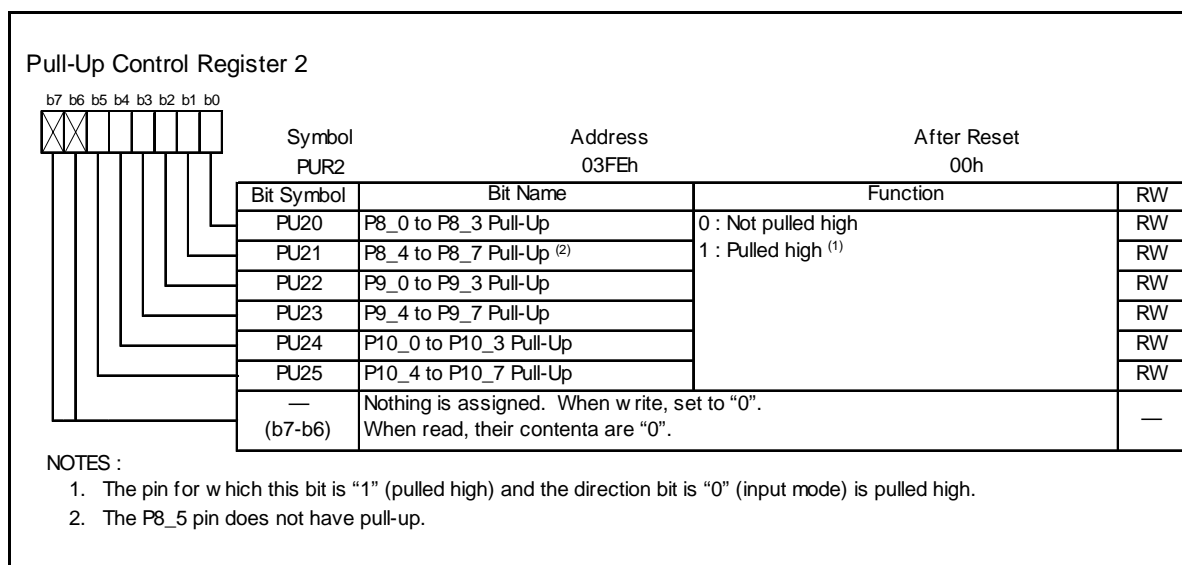


Figure 21.11 PUR2 Register

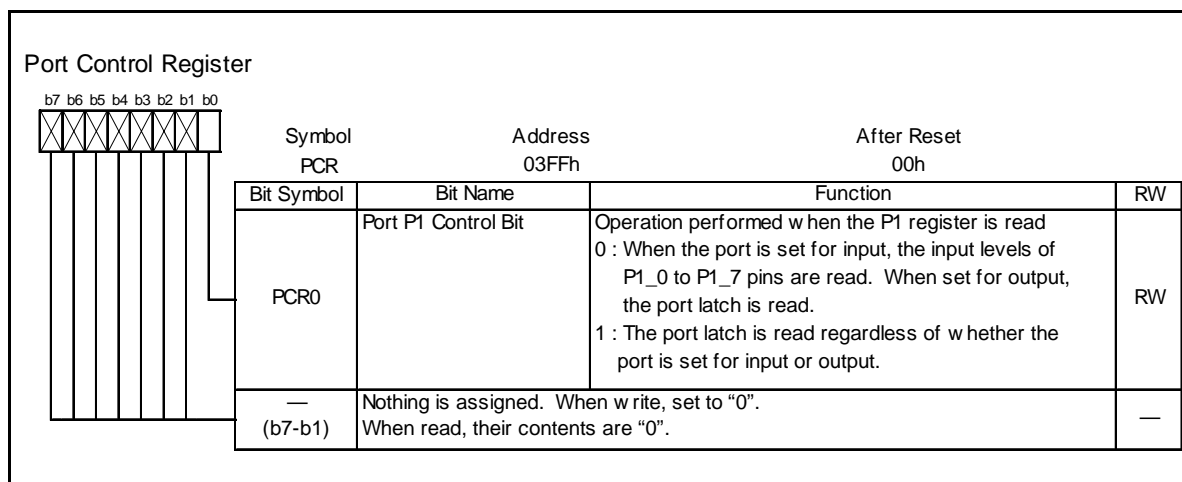


Figure 21.12 PCR Register

**Table 21.2 Unassigned Pin Handling in Single-chip Mode**

Pin Name	Connection
Ports P0 to P7, P8_0 to P8_4, P8_6 to P8_7, P9 to P14	After setting for input mode, connect every pin to VSS via a resistor (pull-down); or after setting for output mode, leave these pins open. (1, 2, 3, 5)
XOUT (4)	Open
$\overline{\text{NMI}}$	Connect via resistor to VCC1 (pull-up)
AVCC	Connect to VCC1
AVSS, VREF, BYTE	Connect to VSS

**NOTES:**

- When setting the port for output mode and leave it open, be aware that the port remains in input mode until it is switched to output mode in a program after reset. For this reason, the voltage level on the pin becomes indeterminate, causing the power supply current to increase while the port remains in input mode.

Furthermore, by considering a possibility that the contents of the direction registers could be changed by noise or noise-induced runaway, it is recommended that the contents of the direction registers be periodically reset in software, for the increased reliability of the program.

- Make sure the unused pins are processed with the shortest possible wiring from the microcomputer pins (within 2 cm).
- When the ports P7\_0 and P7\_1 are set for output mode, make sure a low-level signal is output from the pins.

The ports P7\_0 and P7\_1 are N-channel open-drain outputs.

- With external clock input to XIN pin.
- Process the port without a pin in the 80-pin version and the 100-pin version as follows.

## 80-pin version

- Set the direction bits in these ports to "1" (output mode), and set the output data to "0" ("L") using the program.
- Ports P11 to P14 do not exist.

## 100-pin version

- After reset, PU37 bit is "0" (P11 to P14 do not used).  
Do not write "1" to PU37 bit. When read, value of PU37 bit is indeterminate.
- The port direction bit in the P11 to P14 can be set "0" or "1".

**Table 21.3 Unassigned Pin Handling in Memory Expansion Mode and Microprocessor Mode**

Pin Name	Connection
Ports P0 to P7, P8_0 to P8_4, P8_6 to P8_7, P9 to P14	After setting for input mode, connect every pin to VSS via a resistor (pull-down); or after setting for output mode, leave these pins open. (1, 2, 3, 4, 7)
P4_5/CS1 to P4_7/CS3	Connect to VCC2 via a resistor (pulled high) by setting the corresponding direction bit in the PD4 register for CSi (i=1 to 3) to "0" (input mode) and the CSi bit in the CSR register to "0" (chip select disabled).
BHE, ALE, HLDA, XOUT <sup>(5)</sup> , BCLK <sup>(6)</sup>	Open
HOLD, RDY	Connect via resistor to VCC2 (pull-up)
NMI (P8_5)	Connect via resistor to VCC1 (pull-up)
AVCC	Connect to VCC1
AVSS, VREF	Connect to VSS

**NOTES:**

- When setting the port for output mode and leave it open, be aware that the port remains in input mode until it is switched to output mode in a program after reset. For this reason, the voltage level on the pin becomes indeterminate, causing the power supply current to increase while the port remains in input mode.  
Furthermore, by considering a possibility that the contents of the direction registers could be changed by noise or noise-induced runaway, it is recommended that the contents of the direction registers be periodically reset in software, for the increased reliability of the program.
- Make sure the unused pins are processed with the shortest possible wiring from the microcomputer pins (within 2 cm).
- If the CNVSS pin has the VSS level applied to it, these pins are set for input ports until the processor mode is switched over in a program after reset. For this reason, the voltage levels on these pins become indeterminate, causing the power supply current to increase while they remain set for input ports.
- When the ports P7\_0 and P7\_1 are set for output mode, make sure a low-level signal is output from the pins.  
The ports P7\_0 and P7\_1 are N-channel open-drain outputs.
- With external clock input to XIN pin.
- If the PM07 bit in the PM0 register is set to "1" (BCLK not output), connect this pin to VCC2 via a resistor (pulled high).
- Process the port without a pin in the 100-pin version as follows.
  - After reset, PU37 bit is "0" (P11 to P14 do not used).  
Do not write "1" to PU37 bit. When read, value of PU37 bit is indeterminate.
  - The port direction bit in the P11 to P14 can be set "0" or "1".

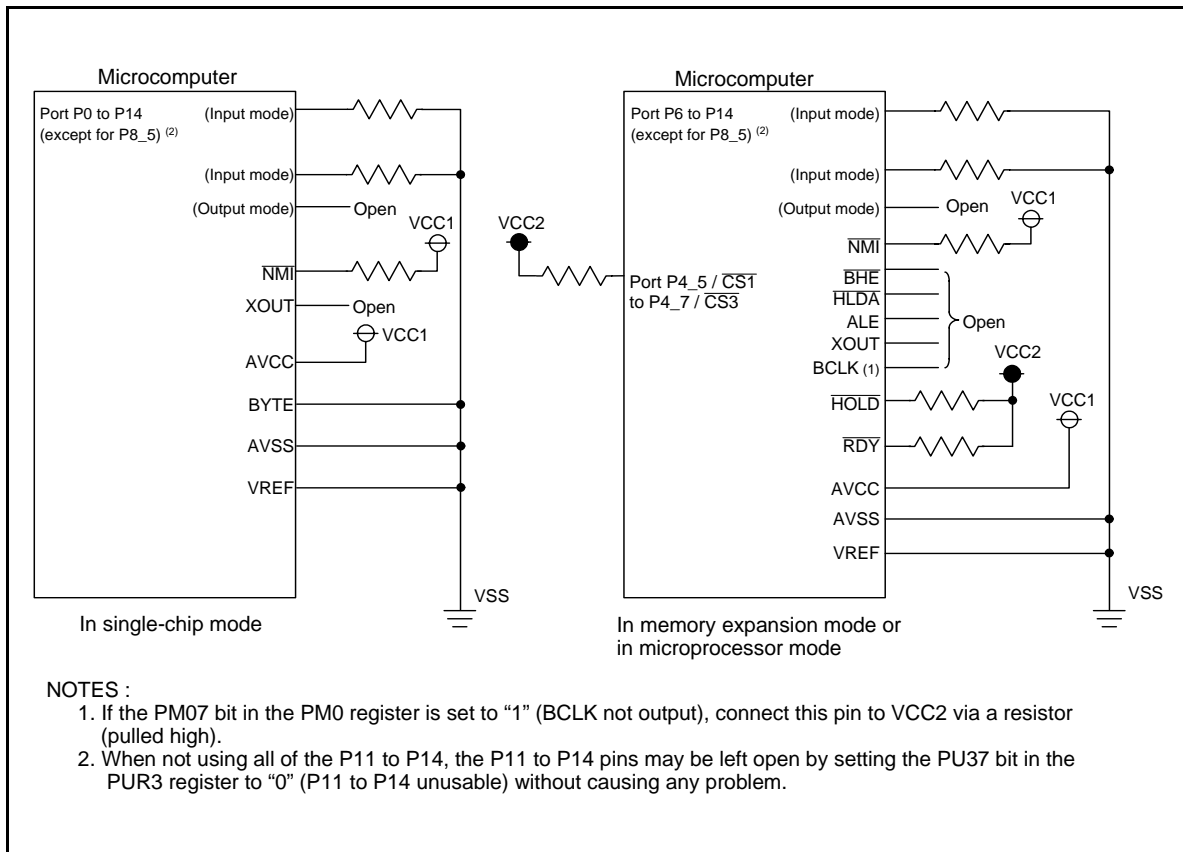


Figure 21.13 Unassigned Pins Handling

## 22. Flash Memory Version

Aside from the built-in flash memory, the flash memory version microcomputer has the same functions as the masked ROM version.

In the flash memory version, the flash memory can perform in three rewrite modes: CPU rewrite mode, standard serial I/O mode and parallel I/O mode.

Table 22.1 lists specifications of the flash memory version. See **Table 1.1 to Table 1.3 Performance outline of M16C/62PT group** for the items not listed in Table 22.1.

**Table 22.1 Flash Memory Version Specifications**

Item		Specification
Flash Memory Rewrite Mode		3 modes (CPU rewrite, standard serial I/O, parallel I/O)
Erase Block	User ROM Area	See <b>Figure 22.1 Flash Memory Block Diagram</b>
	Boot ROM Area	1 block (4 Kbytes) <sup>(1)</sup>
Program Method		In units of word, in units of byte <sup>(2)</sup>
Erase Method		Collective erase, block erase
Program and Erase Control Method		Program and erase controlled by software command
Protect Method		The lock bit protects each block
Number of Commands		8 commands
Program and Erase Endurance		100 times, 1,000 times/10,000 times (option) <sup>(3, 4)</sup>
Data Retention		10 years
ROM Code Protection		Parallel I/O and standard serial I/O modes are supported

**NOTES:**

1. The boot ROM area contains a standard serial I/O mode rewrite control program which is stored in it when shipped from the factory. This area can only be rewritten in parallel input/output mode.
2. Can be programmed in byte units in only parallel input/output mode.
3. Block 1 and block A are 10,000 times of programming and erasure. All other blocks are 1,000 times of programming and erasure.
4. Definition of program and erase endurance  
 The programming and erasure times are defined to be per-block erasure times. For example, assume a case where a 4-Kbyte block A is programmed in 2,048 operations by writing one word at a time and erased thereafter.  
 In this case, the block is reckoned as having been programmed and erased once.  
 If a product is 100 times of programming and erasure, each block in it can be erased up to 100 times. When 10,000 times of programming and erasure, block 1 and block A can each be erased up to 10,000 times. All other blocks can each be erased up to 1,000 times.

**Table 22.2 Flash Memory Rewrite Modes Overview**

Flash Memory Rewrite Mode	CPU rewrite Mode <sup>(1)</sup>	Standard Serial I/O Mode	Parallel I/O Mode
Function	The User ROM area is rewritten when the CPU executes software commands. EW0 mode: Rewrite in areas other than flash memory <sup>(2)</sup> EW1 mode: Can be rewritten in the flash memory	The user ROM area is rewritten using a dedicated serial programmer. Standard serial I/O mode 1: Clock synchronous serial I/O Standard serial I/O mode 2: UART	The boot ROM area and user ROM area is rewritten using a dedicated parallel programmer.
Areas which can be Rewritten	User ROM area	User ROM area	User ROM area Boot ROM area
Operating Mode	Single-chip mode Memory expansion mode (EW0 mode) Boot mode (EW0 mode)	Boot mode	Parallel I/O mode
ROM Programmer	None	Serial programmer	Parallel programmer

**NOTES:**

1. The PM13 bit remains set to "1" while the FMR01 bit in the FMR0 register = 1 (CPU rewrite mode enabled). The PM13 bit is reverted to its original value by clearing the FMR01 bit to "0" (CPU rewrite mode disabled). However, if the PM13 bit is changed during CPU rewrite mode, its changed value is not reflected until after the FMR01 bit is cleared to "0".
2. When in CPU rewrite mode, the PM10 and PM13 bits in the PM1 register are set to "1". The rewrite control program can only be executed in the internal RAM or in an external area that is enabled for use when the PM13 bit = 1. When the PM13 bit = 0 and the flash memory is used in 4-Mbyte mode, the extended accessible area (40000h to BFFFFh) cannot be used.

## 22.1 Memory Map

The flash memory contains the user ROM area and the boot ROM area. The user ROM area has space to store the microcomputer operating program in single-chip mode or memory expansion mode and a separate 4-Kbyte space as the block A. Figure 22.1 shows a Flash Memory Block Diagram.

The user ROM area is divided into several blocks, each of which can be protected (locked) from program or erase. The user ROM area can be rewritten in CPU rewrite, standard serial I/O and parallel I/O modes.

Block A is enabled for use by setting the PM10 bit in the PM1 register to “1” (block A enabled, CS2 area at addresses 10000h to 26FFFh).

The boot ROM area is located at the same addresses as the user ROM area. It can only be rewritten in parallel I/O mode (refer to **22.1.1 Boot Mode**). A program in the boot ROM area is executed after a hardware reset occurs while an “H” signal is applied to the CNVSS and P5\_0 pins and an “L” signal is applied to the P5\_5 pin (refer to **22.1.1 Boot Mode**). A program in the user ROM area is executed after a hardware reset occurs while an “L” signal is applied to the CNVSS pin. However, the boot ROM area cannot be read.

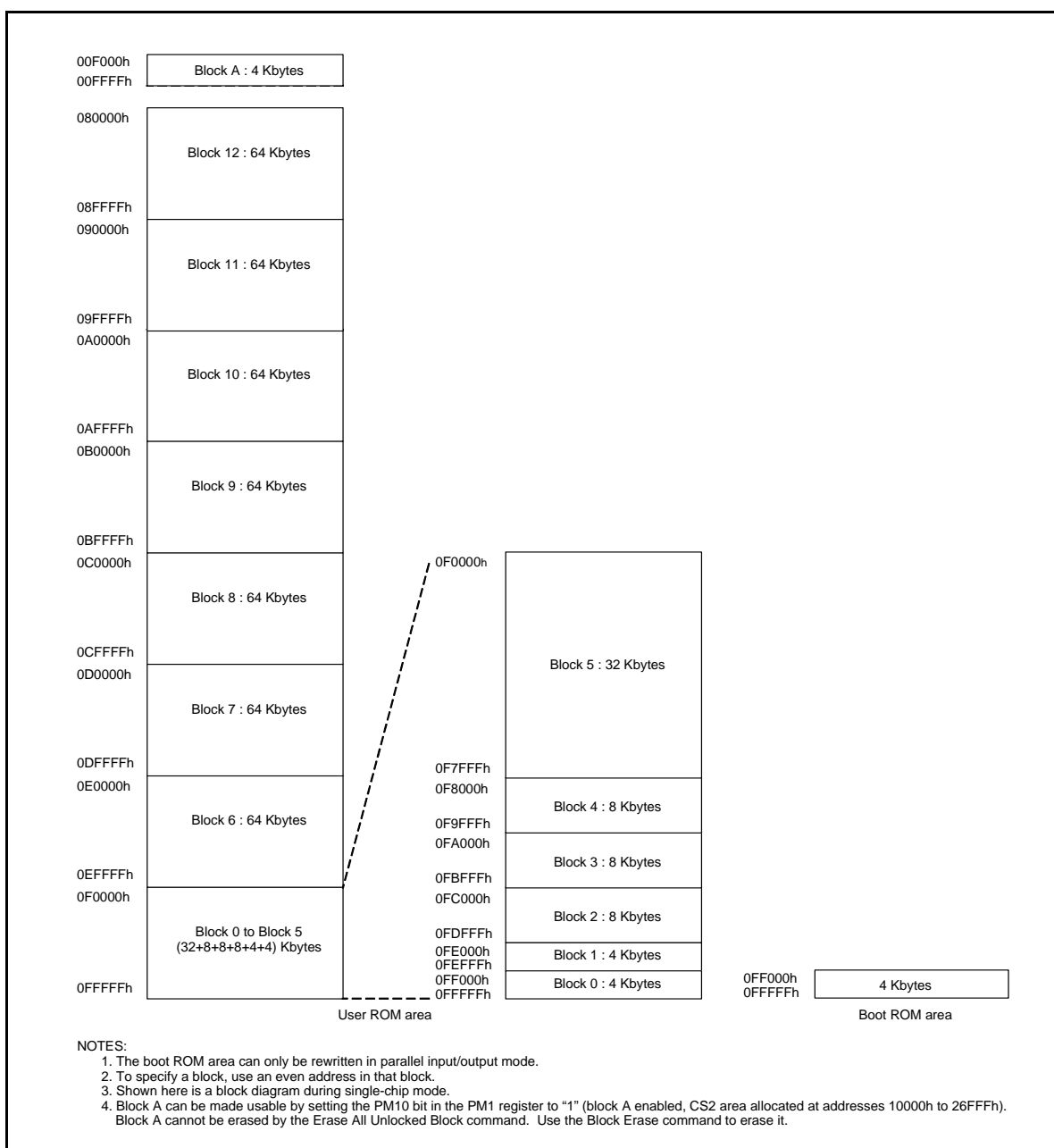


Figure 22.1 Flash Memory Block Diagram

### 22.1.1 Boot Mode

The microcomputer enters boot mode when a hardware reset occurs while an “H” signal is applied to the CNVSS and P5\_0 pins and an “L” signal is applied to the P5\_5 pin. A program in the boot ROM area is executed.

In boot mode, the FMR05 bit in the FMR0 register selects access to the boot ROM area or the user ROM area.

The rewrite control program for standard serial I/O mode is stored in the boot ROM area before shipment.

The boot ROM area can be rewritten in parallel I/O mode only. If any rewrite control program using erase-write mode (EW0 mode) is written in the boot ROM area, the flash memory can be rewritten according to the system implemented.

## 22.2 Functions To Prevent Flash Memory from Rewriting

The flash memory has a built-in ROM code protect function for parallel I/O mode and a built-in ID code check function for standard I/O mode to prevent the flash memory from reading or rewriting.

### 22.2.1 ROM Code Protect Function

The ROM code protect function inhibits the flash memory from being read or rewritten during parallel input/output mode. Figure 22.2 shows the ROMCP Register. The ROMCP register is located in the user ROM area.

The ROM code protect function is enabled when the ROMCR bits are set to other than “11b”. In this case, set the bit 5 to bit 0 to “111111b”.

When exiting ROM code protect, erase the block including the ROMCP1 register by the CPU rewrite mode or the standard serial I/O mode.

### 22.2.2 ID Code Check Function

Use the ID code check function in standard serial I/O mode. The ID code sent from the serial programmer is compared with the ID code written in the flash memory for a match. 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 “FFFFFFFh”, 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, into addresses 0FFFDf, 0FFFE3h, 0FFFEb, 0FFFEFh, 0FFFF3h, 0FFFF7h, and 0FFFFBh. The flash memory must have a program with the ID codes set in these addresses.



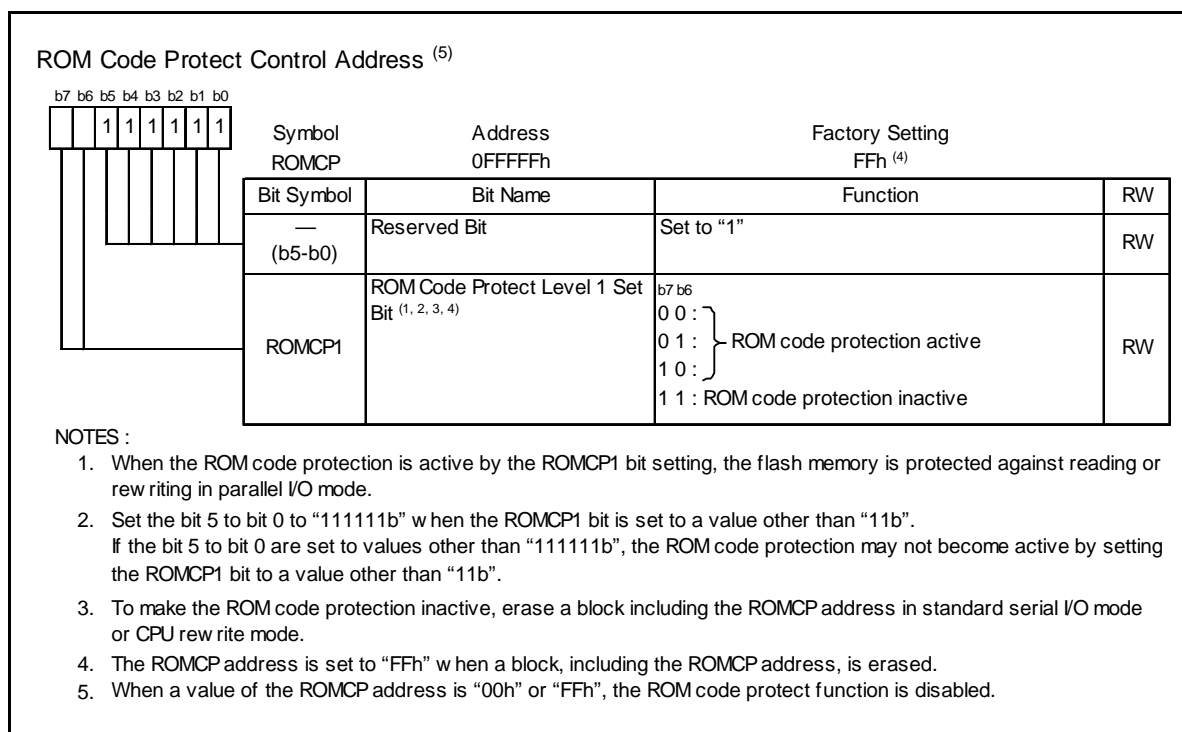


Figure 22.2 ROMCP Register

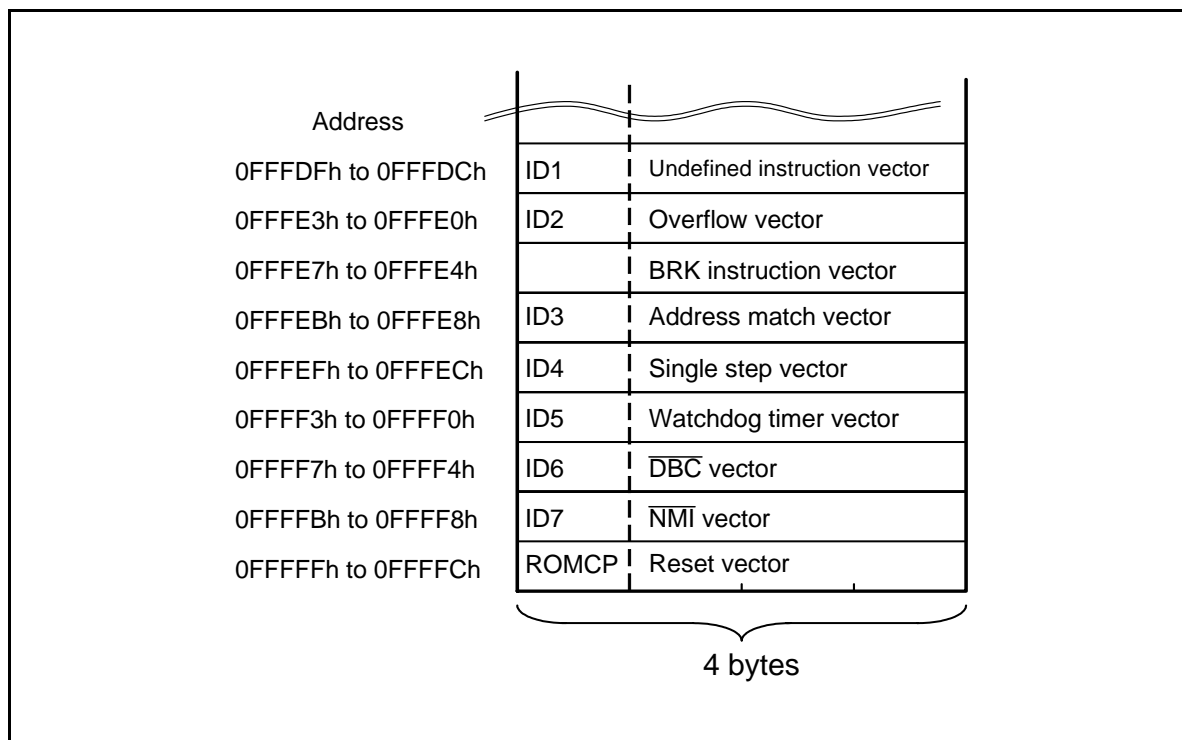


Figure 22.3 Address for ID Code Stored

## 22.3 CPU Rewrite Mode

In CPU rewrite mode, the user ROM area can be rewritten when the CPU executes software commands.

The user ROM area can be rewritten with the microcomputer mounted on a board without using a parallel or serial programmer.

In CPU rewrite mode, only the user ROM area shown in Figure 22.1 can be rewritten. The boot ROM area cannot be rewritten. Program and the block erase command are executed only in the user ROM area.

Erase-write 0 (EW0) mode and erase-write 1 (EW1) mode are provided as CPU rewrite mode. Table 22.3 lists differences between erase-write 0 (EW0) and erase-write 1 (EW1) modes.

**Table 22.3 EW0 Mode and EW1 Mode**

Item	EW0 Mode	EW1 Mode
Operating Mode	<ul style="list-style-type: none"> <li>• Single-chip mod</li> <li>• Memory expansion mode</li> <li>• Boot mode</li> </ul>	<ul style="list-style-type: none"> <li>• Single-chip mode</li> </ul>
Space where the rewrite control program can be placed	<ul style="list-style-type: none"> <li>• User ROM area</li> <li>• Boot ROM area</li> </ul>	<ul style="list-style-type: none"> <li>• User ROM area</li> </ul>
Space where the rewrite control program can be executed	The rewrite control program must be transferred to any space other than the flash memory (e.g., RAM) before being executed <sup>(2)</sup>	The rewrite control program can be executed in the user ROM area
Space which can be rewritten	User ROM area	User ROM area However, this excludes blocks with the rewrite control program
Software Command Restriction	None	<ul style="list-style-type: none"> <li>• Program and block erase commands cannot be executed in a block having the rewrite control program.</li> <li>• Erase all unlocked block command cannot be executed when the lock bit in a block having the rewrite control program is set to "1" (unlocked) or when the FMR02 bit in the FMR0 register is set to "1" (lock bit disabled).</li> <li>• Read status register command cannot be used.</li> </ul>
Mode after Program or Erasing	Read status register mode	Read array mode
CPU State during Auto Write and Auto Erase	Operating	Maintains hold state (I/O ports maintains the state before the command was executed) <sup>(1)</sup>
Flash Memory State Detection	<ul style="list-style-type: none"> <li>• Read the FMR00, FMR06 and FMR07 bits in the FMR0 register by program</li> <li>• Execute the read status register command to read the SR7, SR5 and SR4 bits in the status register.</li> </ul>	Read the FMR00, FMR06 and FMR07 bits in the FMR0 register by program

### NOTES:

1. Do not generate an interrupt (except  $\overline{\text{NMI}}$  interrupt) or DMA transfer.
2. When in CPU rewrite mode, the PM10 and PM13 bits in the PM1 register are set to "1". The rewrite control program can only be executed in the internal RAM or in an external area that is enabled for use when the PM13 bit = 1. When the PM13 bit = 0 and the flash memory is used in 4M-byte mode, the extended accessible area (40000h to BFFFFh) cannot be used.

### 22.3.1 EW0 Mode

The microcomputer enters CPU rewrite mode by setting the FMR01 bit in the FMR0 register to “1” (CPU rewrite mode enabled) and is ready to accept commands. EW0 mode is selected by setting the FMR11 bit in the FMR1 register to “0”. To set the FMR01 bit to “1”, set to “1” after first writing “0”.

The software commands control programming and erasing. The FMR0 register or the status register indicates whether a program or erase operation is completed as expected or not.

### 22.3.2 EW1 Mode

EW1 mode is selected by setting the FMR11 bit to “1” after the FMR01 bit is set to “1”. (Both bits must be set to “0” first before setting to “1”.)

The FMR0 register indicates whether or not a program or erase operation has been completed as expected. The status register cannot be read in EW1 mode.

When an erase/program operation is initiated the CPU halts all program execution until the operation is completed or erase-suspend is requested.

### 22.3.3 Flash memory Control Register (FIDR, FMR0 and FMR1 registers)

Figure 22.4 to Figure 22.6 show the FIDR, FMR0 and FMR1 Registers.

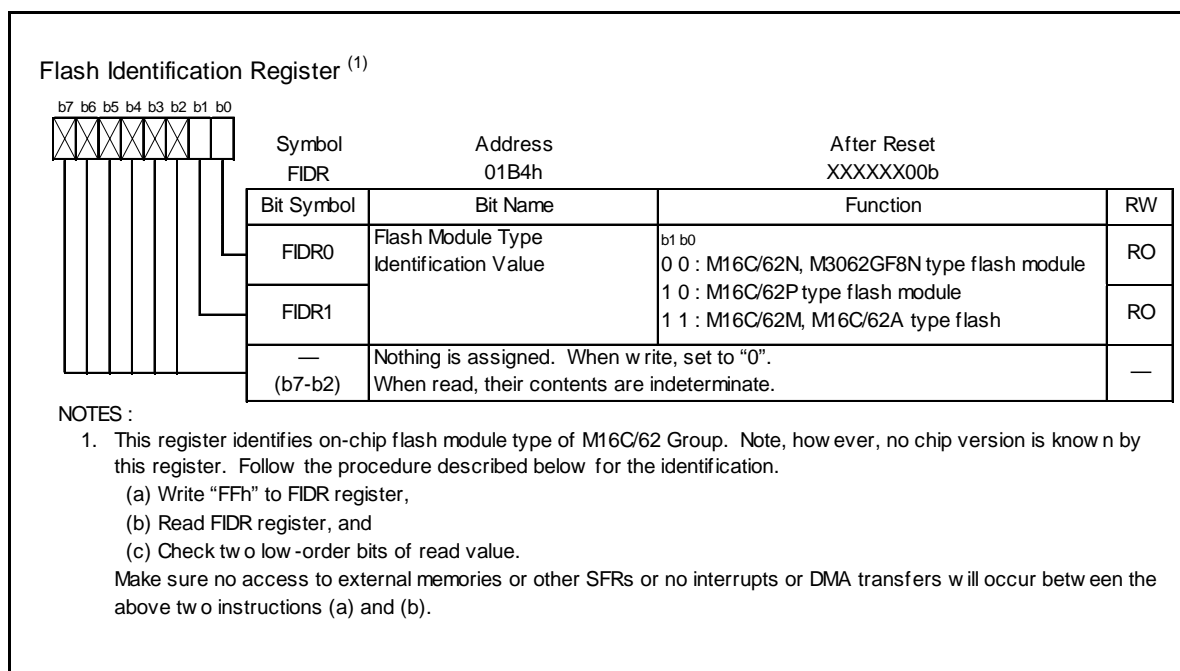
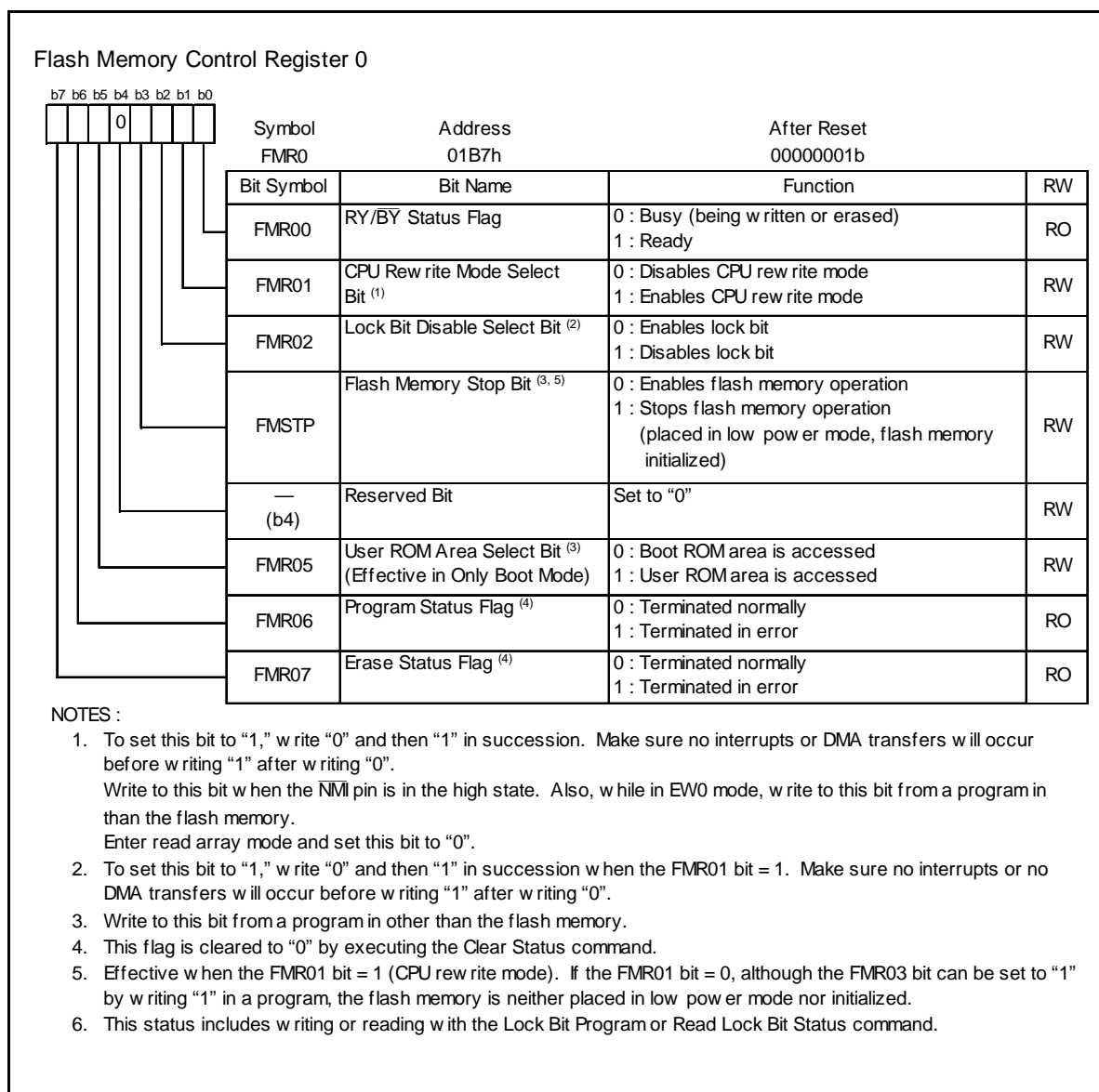


Figure 22.4 FIDR Register

**Figure 22.5 FMR0 Register**

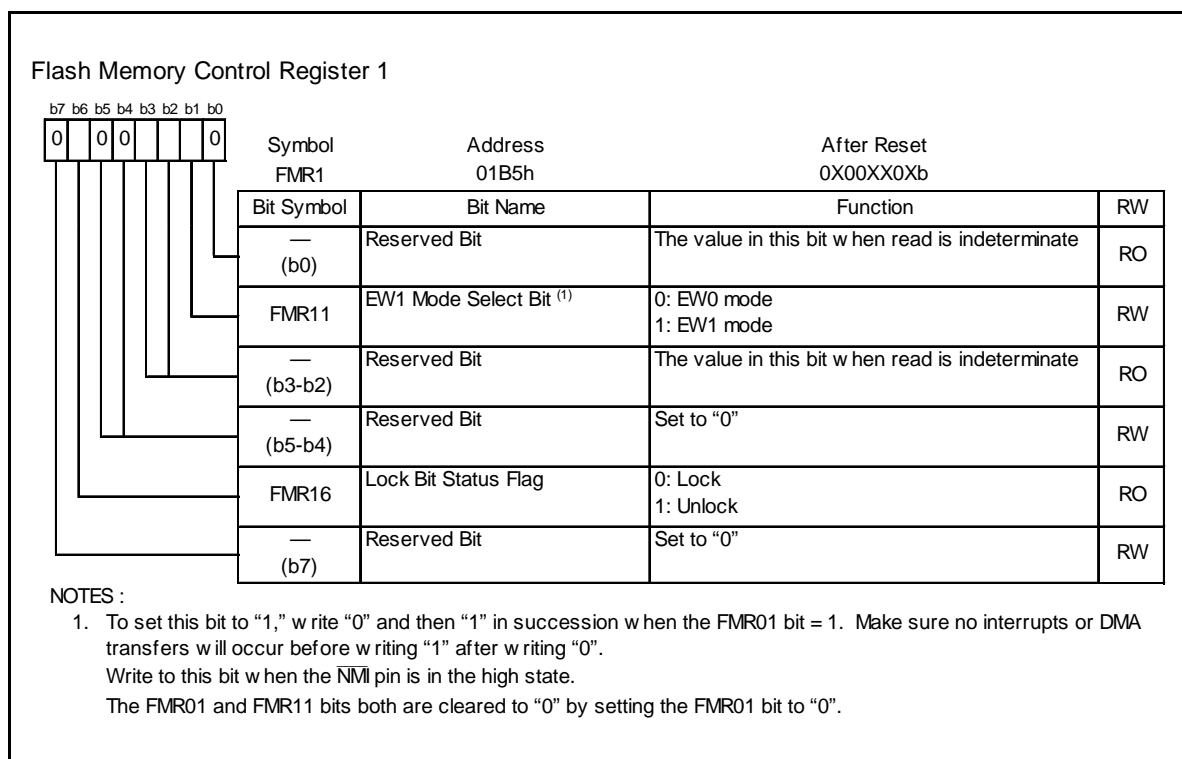


Figure 22.6 FMR1 Register

### 22.3.3.1 FMR00 Bit

This bit indicates the flash memory operating state. It is set to “0” while the program, block erase, erase all unlocked block, lock bit program, or read lock bit status command is being executed; otherwise, it is set to “1”.

### 22.3.3.2 FMR01 Bit

The microcomputer can accept commands when the FMR01 bit is set to “1” (CPU rewrite mode). Set the FMR05 bit to “1” (user ROM area access) as well if in boot mode.

### 22.3.3.3 FMR02 Bit

The lock bit is disabled by setting the FMR02 bit to “1” (lock bit disabled). (Refer to **22.3.6 Data Protect Function**.) The lock bit is enabled by setting the FMR02 bit to “0” (lock bit enabled).

The FMR02 bit does not change the lock bit status but disables the lock bit function. If the block erase or erase all unlocked block command is executed when the FMR02 bit is set to “1”, the lock bit status changes “0” (locked) to “1” (unlocked) after command execution is completed.

### 22.3.3.4 FMSTP Bit

The FMSTP bit resets the flash memory control circuits and minimizes power consumption in the flash memory. Access to the flash memory is disabled when the FMSTP bit is set to “1”. Set the FMSTP bit by program in a space other than the flash memory.

- Set the FMSTP bit to “1” if one of the followings occurs: A flash memory access error occurs while erasing or programming in EW0 mode (FMR00 bit does not switch back to “1” (ready)).
- Low-power consumption mode or on-chip oscillator low-power consumption mode is entered

Use the following the procedure to change the FMSTP bit setting.

- (1) Set the FMSTP bit to “1”
- (2) Set tps (the wait time to stabilize flash memory circuit)
- (3) Set the FMSTP bit to “0”
- (4) Set tps (the wait time to stabilize flash memory circuit)

Figure 22.9 shows a Flow Chart Illustrating How To Start and Stop the Flash Memory Processing Before and After Low Power Dissipation Mode or On-chip Oscillator Low-Power Consumption Mode. Follow the procedure on this flow chart.

When entering stop or wait mode, the flash memory is automatically turned off. When exiting stop or wait mode, the flash memory is turned back on. The FMR0 register does not need to be set.

### 22.3.3.5 FMR05 Bit

This bit selects the boot ROM or user ROM area in boot mode. Set to “0” to access (read) the boot ROM area or to “1” (user ROM access) to access (read, write or erase) the user ROM area.

#### 22.3.3.6 FMR06 Bit

This is a read-only bit indicating an auto program operation state. The FMR06 bit is set to “1” when a program error occurs; otherwise, it is set to “0”. Refer to **22.3.8 Full Status Check**.

#### 22.3.3.7 FMR07 Bit

This is a read-only bit indicating the auto erase operation status. The FMR07 bit is set to “1” when an erase error occurs; otherwise, it is set to “0”. For details, refer to **22.3.8 Full Status Check**.

Figure 22.7 shows Setting and Resetting of EW0 Mode. Figure 22.8 show Setting and Resetting of EW1 Mode.

#### 22.3.3.8 FMR11 Bit

EW0 mode is entered by setting the FMR11 bit to “0” (EW0 mode).

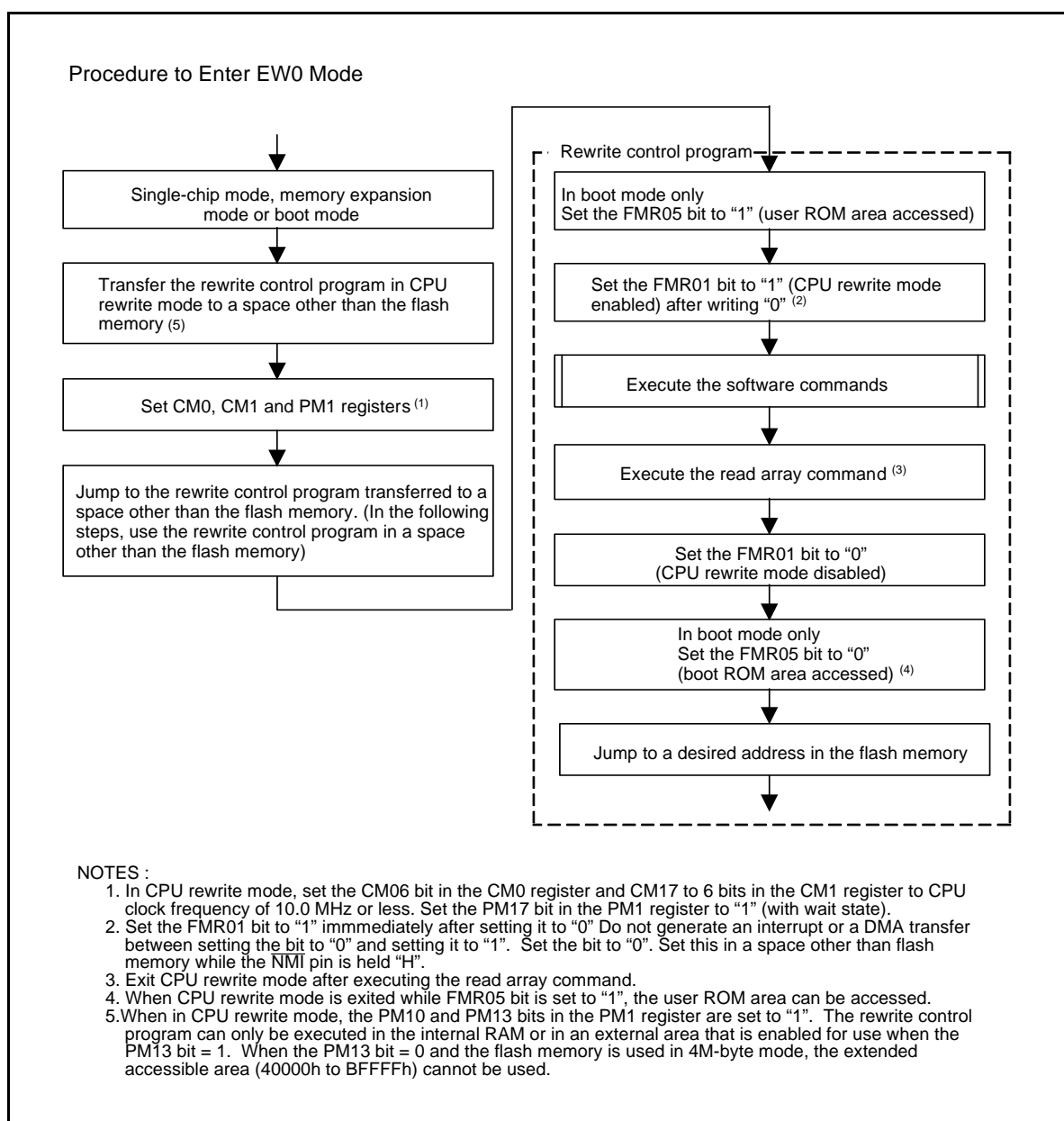
EW1 mode is entered by setting the FMR11 bit to “1” (EW1 mode).

#### 22.3.3.9 FMR16 Bit

This is a read-only bit indicating the execution result of the read lock bit status command.

When the block, where the read lock bit status command is executed, is locked, the FMR16 bit is set to “0”.

When the block, where the read lock bit status command is executed, is unlocked, the FMR16 bit is set to “1”.

**Figure 22.7 Setting and Resetting of EW0 Mode**



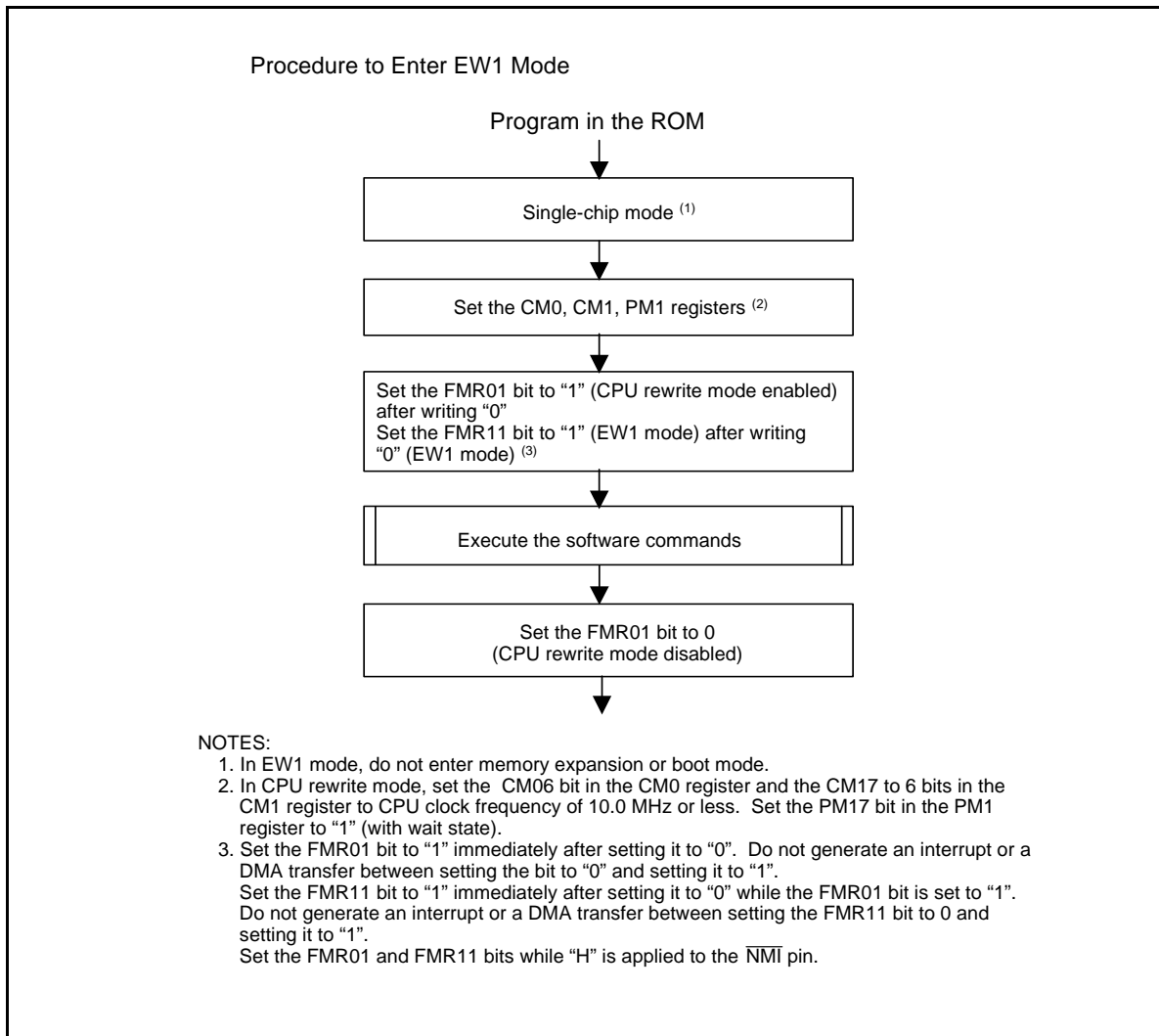
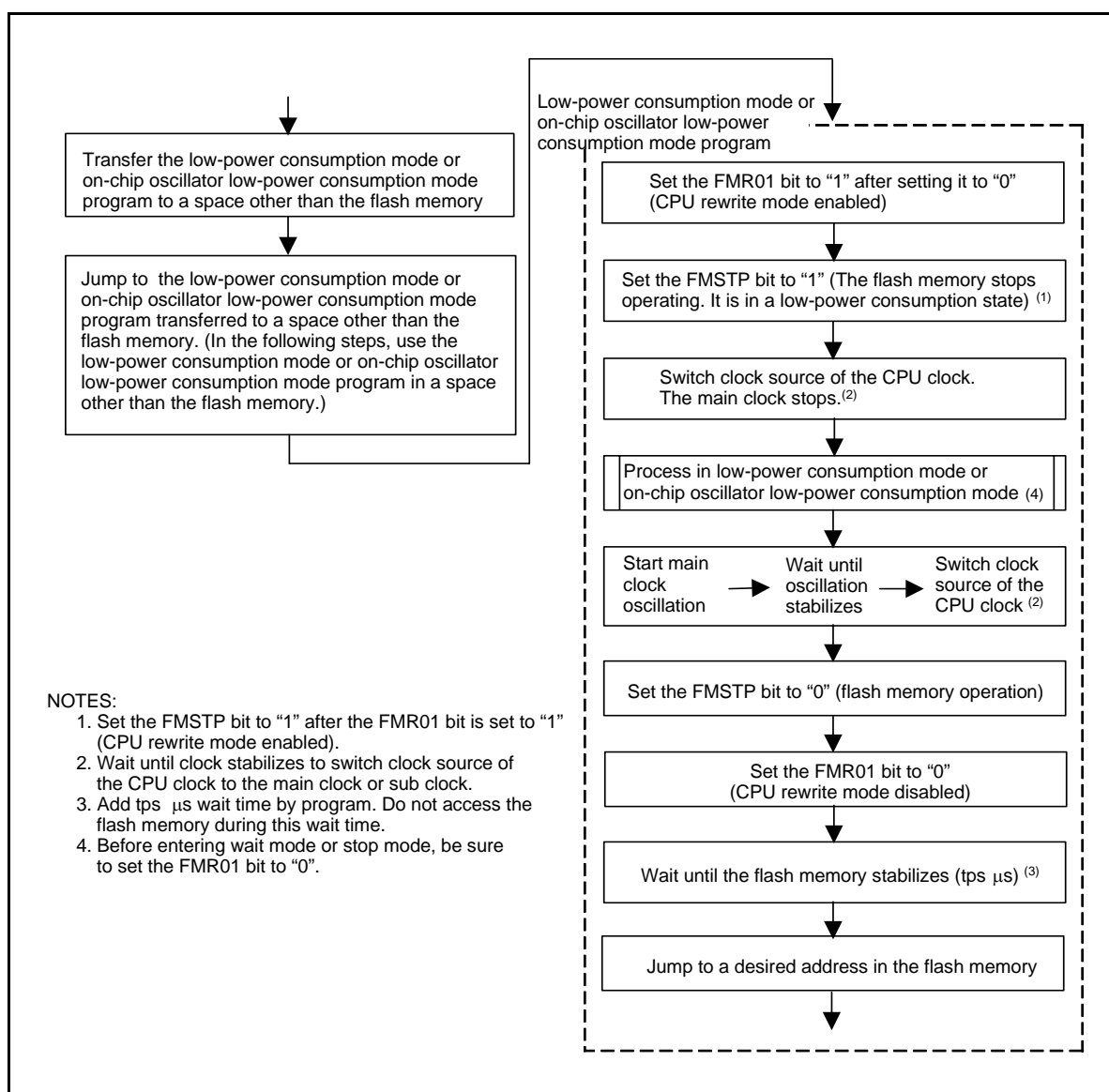


Figure 22.8 Setting and Resetting of EW1 Mode



**Figure 22.9 Processing Before and After Low Power Dissipation Mode or On-chip Oscillator Low-Power Consumption Mode**

## 22.3.4 Precautions on CPU Rewrite Mode

### 22.3.4.1 Operating Speed

Set the CM06 bit in the CM0 register and the CM17 to CM16 bits in the CM1 register to a CPU clock frequency of 10 MHz or less before entering CPU rewrite mode (EW0 or EW1 mode). Also, set the PM17 bit in the PM1 register to “1” (wait state).

### 22.3.4.2 Prohibited Instructions

The following instructions cannot be used in EW0 mode because the CPU tries to read data in the flash memory: the UND instruction, INTO instruction, JMPS instruction, JSRS instruction, and BRK instruction.

### 22.3.4.3 Interrupts (EW0 mode)

- To use interrupts having vectors in a relocatable vector table, the vectors must be relocated to the RAM area.
- The  $\overline{\text{NMI}}$  and watchdog timer interrupts are available since the FMR0 and FMR1 registers are forcibly reset when either interrupt occurs. Allocate the jump addresses for each interrupt service routines to the fixed vector table. Flash memory rewrite operation is suspended when the  $\overline{\text{NMI}}$  or watchdog timer interrupt occurs. Execute the rewrite program again after exiting the interrupt routine.
- The address match interrupt is not available since the CPU tries to read data in the flash memory.

### 22.3.4.4 Interrupts (EW1 mode)

- Do not acknowledge any interrupts with vectors in the relocatable vector table or address match interrupt during the auto program or auto erase period.
- Do not use the watchdog timer interrupt.
- The  $\overline{\text{NMI}}$  interrupt is available since the FMR0 and FMR1 registers are forcibly reset when the interrupt occurs. Allocate the jump address for the interrupt service routine to the fixed vector table. Flash memory rewrite operation is suspended when the  $\overline{\text{NMI}}$  interrupt occurs. Execute the rewrite program again after exiting the interrupt service routine.

### 22.3.4.5 22.3.4.5 How to Access

To set the FMR01, FMR02 or FMR11 bit to “1”, write “1” after first setting the bit to “0”. Do not generate an interrupt or a DMA transfer between the instruction to set the bit to “0” and the instruction to set the bit to “1”. Set the bit while an “H” signal is applied to the  $\overline{\text{NMI}}$  pin.

### 22.3.4.6 22.3.4.6 Rewriting in the User ROM Area (EW0 mode)

If the supply voltage drops while rewriting the block where the rewrite control program is stored, the flash memory cannot be rewritten because the rewrite control program is not correctly rewritten. If this error occurs, rewrite the user ROM area while in standard serial I/O mode or parallel I/O mode.

### 22.3.4.7 22.3.4.7 Rewriting in the User ROM Area (EW1 mode)

Avoid rewriting any block in which the rewrite control program is stored.

### 22.3.4.8 22.3.4.8 DMA Transfer

In EW1 mode, do not perform a DMA transfer while the FMR00 bit in the FMR0 register is set to “0” (auto programming or auto erasing).

#### **22.3.4.9 Writing Command and Data**

Write commands and data to even addresses in the user ROM area.

#### **22.3.4.10 Wait Mode**

When entering wait mode, set the FMR01 bit to “0” (CPU rewrite mode disabled) before executing the WAIT instruction.

#### **22.3.4.11 Stop Mode**

When entering stop mode, the following settings are required:

- Set the FMR01 bit to “0” (CPU rewrite mode disabled). Disable DMA transfer before setting the CM10 bit to “1” (stop mode).

#### **22.3.4.12 Low-Power Consumption Mode and On-chip Oscillator Low-power Consumption Mode**

If the CM05 bit is set to “1” (main clock stopped), do not execute the following commands:

- Program
- Block erase
- Erase all unlocked blocks
- Lock bit program
- Read lock bit status

### 22.3.5 Software Commands

Software commands are described below. The command code and data must be read and written in 16-bit units, to and from even addresses in the user ROM area. When writing command code, the 8 high-order bits (D15 to D8) are ignored.

**Table 22.4 Software Commands**

Command	First Bus Cycle			Second Bus Cycle		
	Mode	Address	Data (D0 to D7)	Mode	Address	Data (D0 to D7)
Read Array	Write	X	xxFFh			
Read Status Register	Write	X	xx70h	Read	X	SRD
Clear Status Register	Write	X	xx50h			
Program	Write	WA	xx40h	Write	WA	WD
Block Erase	Write	X	xx20h	Write	BA	xxD0h
Erase All Unlocked Block	Write	X	xxA7h	Write	X	xxD0h
Lock Bit Program	Write	BA	xx77h	Write	BA	xxD0h
Read Lock Bit Status	Write	X	xx71h	Write	BA	xxD0h

**NOTES:**

- Blocks 0 to 12 can be erased by the erase all unlocked block command.  
Block A cannot be erased. The block erase command must be used to erase the block A.  
SRD: Data in the SRD register (D7 to D0)  
WA: Address to be written (The address specified in the first bus cycle is the same even address as the address specified in the second bus cycle.)  
WD: 16-bit write data  
BA: Highest-order block address (must be an even address)  
X: Any even address in the user ROM space  
xx: 8 high-order bits of command code (ignored)

#### 22.3.5.1 Read Array Command (FFh)

The read array command reads the flash memory.

By writing command code “xxFFh” in the first bus cycle, read array mode is entered. Content of a specified address can be read in 16-bit units after the next bus cycle.

The microcomputer remains in read array mode until another command is written. Therefore, contents from multiple addresses can be read consecutively.

#### 22.3.5.2 Read Status Register Command (70h)

The read status register command reads the status register (refer to **22.3.7 Status Register** for detail).

By writing command code “xx70h” in the first bus cycle, the status register can be read in the second bus cycle. Read an even address in the user ROM area.

Do not execute this command in EW1 mode.

#### 22.3.5.3 Clear Status Register Command (50h)

The clear status register command clears the status register. By writing “xx50h” in the first bus cycle, the FMR07 to FMR06 bits in the FMR0 register are set to “00b” and the SR5 to SR4 bits in the status register are set to “00b”.

#### 22.3.5.4 Program Command (40h)

The program command writes 2-byte data to the flash memory. By writing “xx40h” in the first bus cycle and data to the write address in the second bus cycle, an auto program operation (data program and verify) will start. The address value specified in the first bus cycle must be the same even address as the write address specified in the second bus cycle.

The FMR00 bit in the FMR0 register indicates whether an auto program operation has been completed. The FMR00 bit is set to “0” (busy) during auto program and to “1” (ready) when an auto program operation is completed.

After the completion of an auto program operation, the FMR06 bit in the FMR0 register indicates whether or not the auto program operation has been completed as expected. (Refer to **22.3.8 Full Status Check**.)

An address that is already written cannot be altered or rewritten.

Figure 22.10 shows a Flow Chart of the Program Command Programming.

The lock bit protects each block from being programmed inadvertently. (Refer to **22.3.6 Data Protect Function**.)

In EW1 mode, do not execute this command on the block where the rewrite control program is allocated.

In EW0 mode, the microcomputer enters read status register mode as soon as an auto program operation starts. The status register can be read. The SR7 bit in the status register is set to “0” at the same time an auto program operation starts. It is set to “1” when auto program operation is completed. The microcomputer remains in read status register mode until the read array command is written. After completion of an auto program operation, the status register indicates whether or not the auto program operation has been completed as expected.

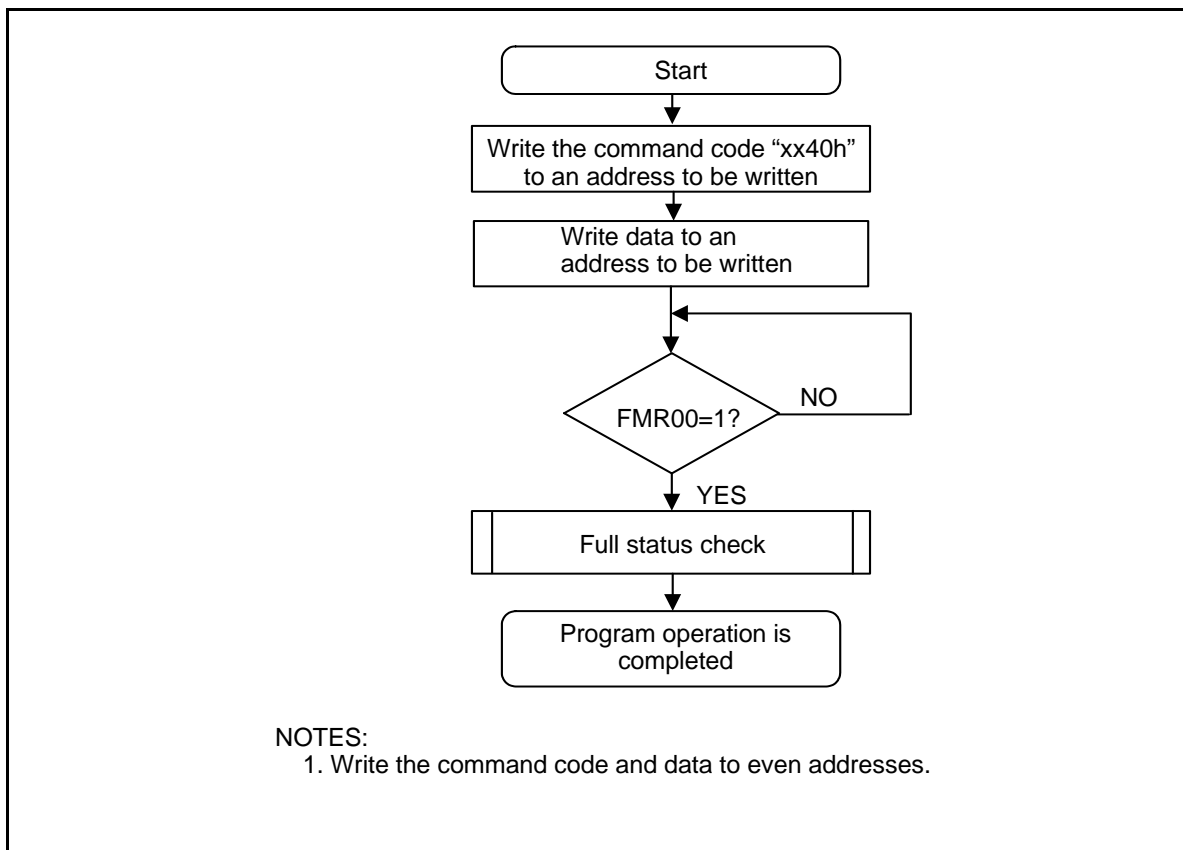


Figure 22.10 Program Command

### 22.3.5.5 Block Erase Command

The block erase command erases each block.

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) will start in the specified block.

The FMR00 bit in the FMR0 register indicates whether an auto erase operation has been completed.

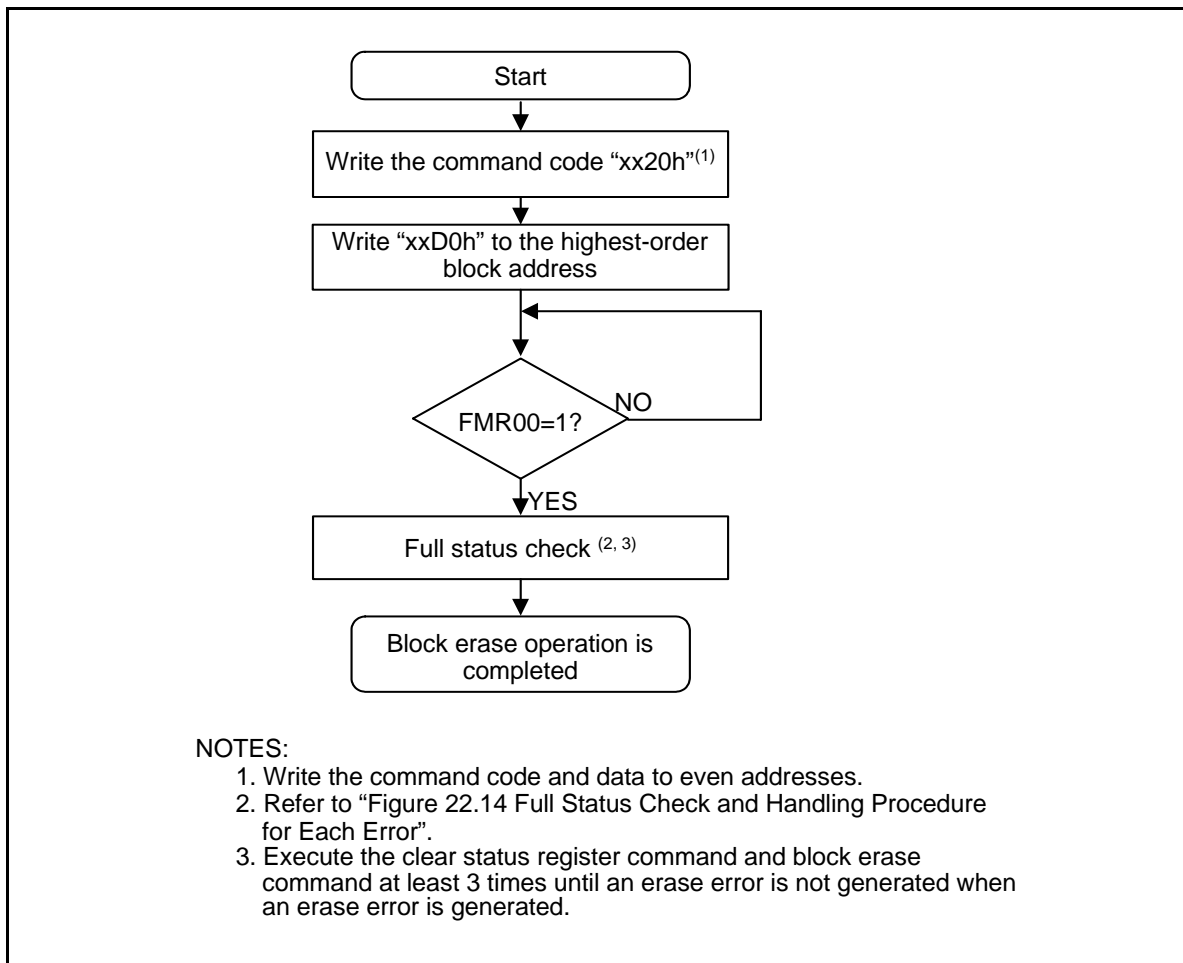
The FMR00 bit is set to “0” (busy) during auto erase and to “1” (ready) when the auto erase operation is completed.

After the completion of an auto erase operation, the FMR07 bit in the FMR0 register indicates whether or not the auto erase operation has been completed as expected. (Refer to **22.3.8 Full Status Check.**)

Figure 22.11 shows a Flow Chart of the Block Erase Command Programming.

The lock bit protects each block from being programmed inadvertently. (Refer to **22.3.6 Data Protect Function.**)

In EW1 mode, do not execute this command on the block where the rewrite control program is allocated. In EW0 mode, the microcomputer enters read status register mode as soon as an auto erase operation starts. The status register can be read. The SR7 bit in the status register is set to “0” at the same time an auto erase operation starts. It is set to “1” when an auto erase operation is completed. The microcomputer remains in read status register mode until the read array command or read lock bit status command is written. Also execute the clear status register command and block erase command at least 3 times until an erase error is not generated when an erase error is generated.



**Figure 22.11 Block Erase Command**

### 22.3.5.6 Erase All Unlocked Block

The erase all unlocked block command erases all blocks except the block A.

By writing “xxA7h” in the first bus cycle and “xxD0h” in the second bus cycle, an auto erase (erase and verify) operation will run continuously in all blocks except the block A.

The FMR00 bit in the FMR0 register indicates whether an auto erase operation has been completed.

After the completion of an auto erase operation, the FMR07 bit in the FMR0 register indicates whether or not the auto erase operation has been completed as expected.

The lock bit can protect each block from being programmed inadvertently. (Refer to **22.3.6 Data Protect Function**.)

In EW1 mode, do not execute this command when the lock bit for any block storing the rewrite control program is set to “1” (unlocked) or when the FMR02 bit in the FMR0 register is set to “1” (lock bit disabled).

In EW0 mode, the microcomputer enters read status register mode as soon as an auto erase operation starts. The status register can be read. The SR7 bit in the status register is set to “0” (busy) at the same time an auto erase operation starts. It is set to “1” (ready) when an auto erase operation is completed. The microcomputer remains in read status register mode until the read array command or read lock bit status command is written.

Only blocks 0 to 12 can be erased by the erase all unlocked block command. The block A cannot be erased. Use the block erase command to erase the block A.

### 22.3.5.7 Lock Bit Program Command

The lock bit program command sets 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 even address of a block specified in the second bus cycle.

Figure 22.12 shows a Flow Chart of the Lock Bit Program Command Programming. Execute read lock bit status command to read lock bit state (lock bit data).

The FMR00 bit in the FMR0 register indicates whether a lock bit program operation is completed.

Refer to **22.3.6 Data Protect Function** for details on lock bit functions and how to set it to “1” (unlocked).

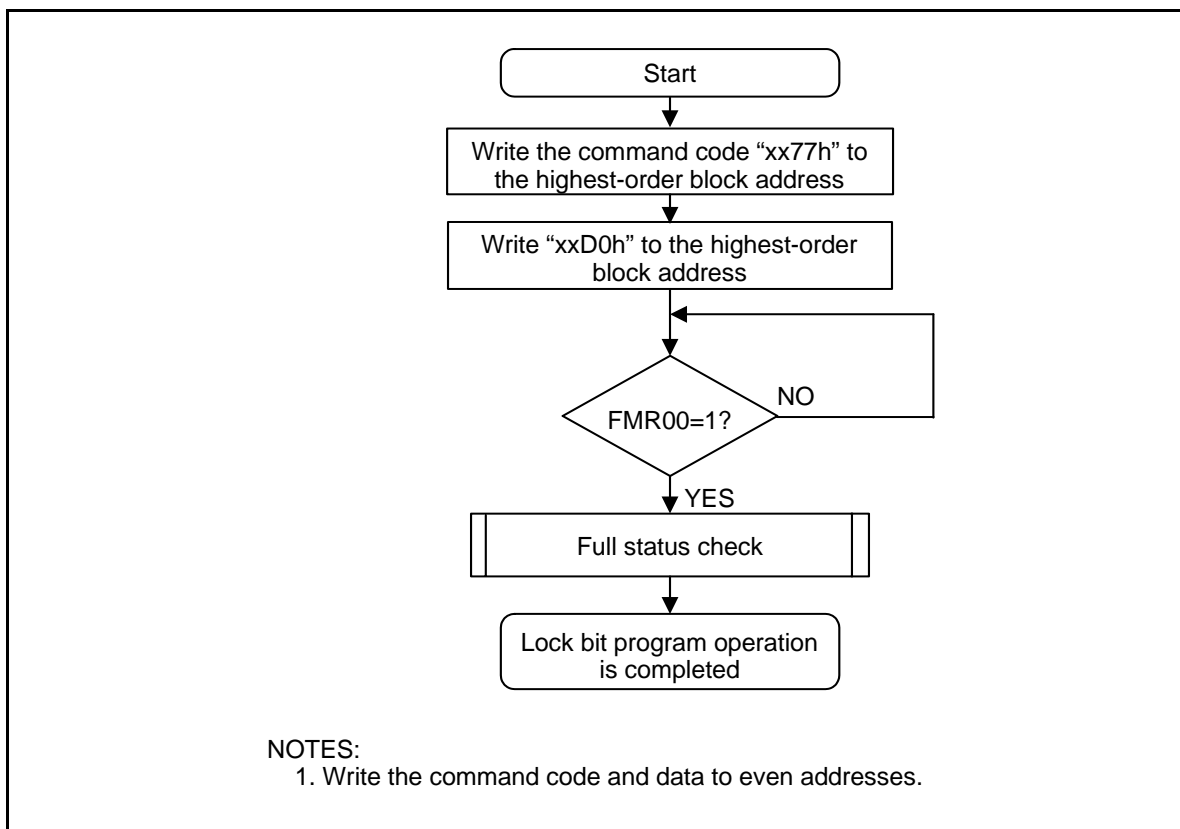


Figure 22.12 Lock Bit Program Command



### 22.3.5.8 Read Lock Bit Status Command (71h)

The read lock bit status command reads 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 whether or not the lock bit of a specified block is locked. Read the FMR16 bit after the FMR00 bit in the FMR0 register is set to “1” (ready).

Figure 22.13 shows a Flow Chart of the Read Lock Bit Status Command Programming.

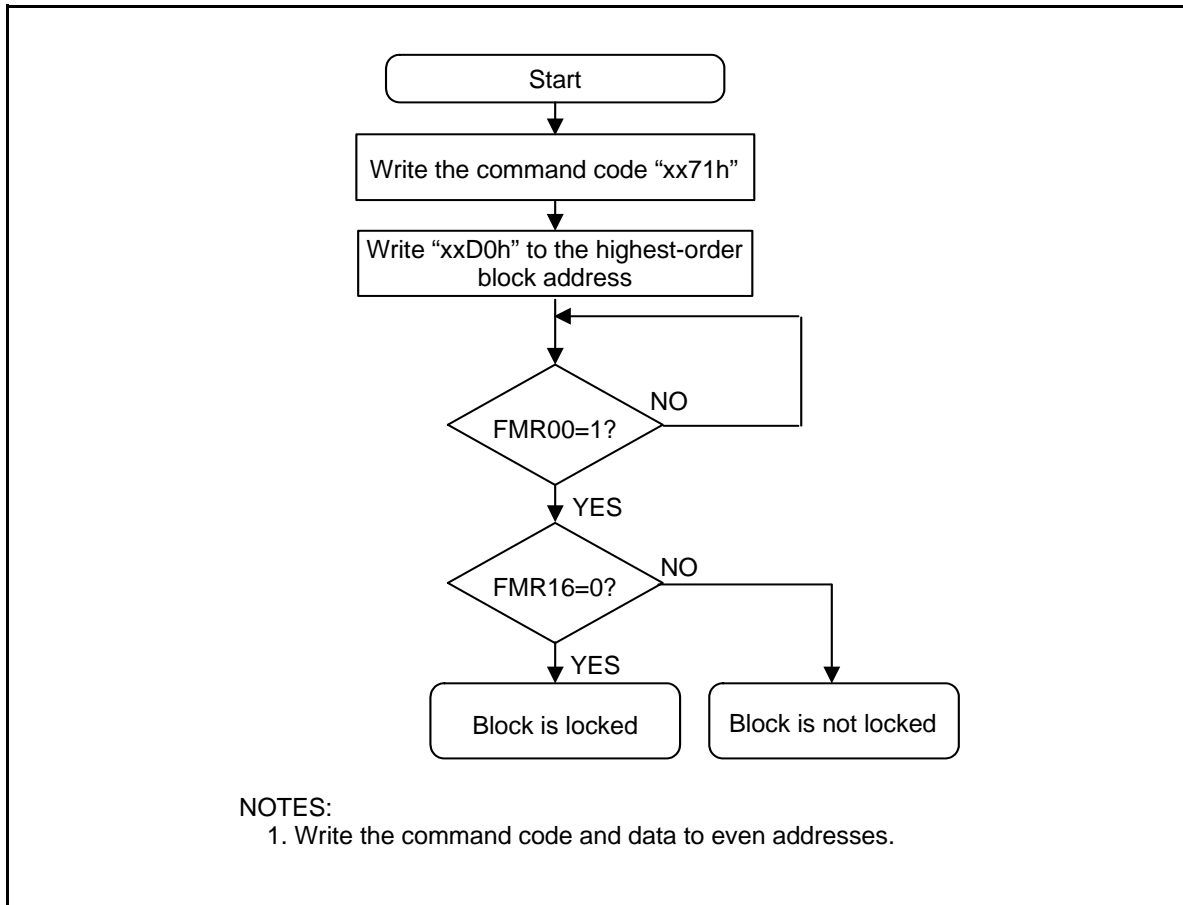


Figure 22.13 Read Lock Bit Status Command

### 22.3.6 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 each block to be individually protected (locked) against program and erase. This helps prevent data from being inadvertently written to or erased from the flash memory.

- When the lock bit status is set to “0”, the block is locked (block is protected against program and erase).
- When the lock bit status is set to “1”, the block is not locked (block can be programmed or erased).

The lock bit status is set to “0” (locked) by executing the lock bit program command and to “1” (unlocked) by erasing the block. The lock bit status cannot be set to “1” by any commands.

The lock bit status can be read by the read lock bit status command.

The lock bit function is disabled by setting the FMR02 bit to “1”. All blocks are unlocked. However, individual lock bit status remains unchanged. The lock bit function is enabled by setting the FMR02 bit to “0”. Lock bit status is retained.

If the block erase or erase all unlocked block command is executed while the FMR02 bit is set to “1”, the target block or all blocks are erased regardless of lock bit status. The lock bit status of each block are set to “1” after an erase operation is completed.

Refer to **22.3.5 Software Commands** for details on each command.

### 22.3.7 Status Register

The status register indicates the flash memory operation state and whether or not an erase or program operation is completed as expected. The FMR00, FMR06 and FMR07 bits in the FMR0 register indicate status register states.

Table 22.5 shows the Status Register.

In EW0 mode, the status register can be read when the followings occur.

- Any even address in the user ROM area is read after writing the read status register command.
- Any even address in the user ROM area is read from when the program, block erase, erase all unlocked block, or lock bit program command is executed until when the read array command is executed.

#### 22.3.7.1 Sequence Status (SR7 and FMR00 Bits)

The sequence status indicates the flash memory operation state. It is set to “0” while the program, block erase, erase all unlocked block, lock bit program, or read lock bit status command is being executed; otherwise, it is set to “1”.

#### 22.3.7.2 Erase Status (SR5 and FMR07 Bits)

Refer to **22.3.8 Full Status Check**.

#### 22.3.7.3 Program Status (SR4 and FMR06 Bits)

Refer to **22.3.8 Full Status Check**.

**Table 22.5 Status Register**

Bits in Status Register	Bit in FMR0 Register	Status name	Definition		Value after Reset
			"0"	"1"	
SR0 (D0)	—	Reserved	—	—	—
SR1 (D1)	—	Reserved	—	—	—
SR2 (D2)	—	Reserved	—	—	—
SR3 (D3)	—	Reserved	—	—	—
SR4 (D4)	FMR06	Program status	Terminated normally	Terminated in error	0
SR5 (D5)	FMR07	Erase status	Terminated normally	Terminated in error	0
SR6 (D6)	—	Reserved	—	—	—
SR7 (D7)	FMR00	Sequencer status	Busy	Ready	1

- D0 to D7: These data buses are read when the read status register command is executed.
- The FMR07 bit (SR5) and FMR06 bit (SR4) are set to "0" by executing the clear status register command.
- When the FMR07 bit (SR5) or FMR06 bit (SR4) is set to "1," the program, block erase, erase all unlocked block and lock bit program commands are not accepted.

### 22.3.8 Full Status Check

If an error occurs when a program or erase operation is completed, the FMR06 to FMR07 bits in the FMR0 register are set to “1”, indicating a specific error. Therefore, execution results can be confirmed by checking these bits (full status check).

Table 22.6 lists Errors and FMR0 Register State. Figure 22.14 shows a flow chart of the Full Status Check and Handling Procedure for Each Error.

**Table 22.6 Errors and FMR0 Register State**

FMR00 Register (Status Register) State		Error	Error Occurrence Conditions
FMR07 bit (SR5 bit)	FMR06 bit (SR4 bit)		
1	1	Command Sequence error	<ul style="list-style-type: none"> <li>• Command is written incorrectly</li> <li>• A value other than “xxD0h” or “xxFFh” is written in the second bus cycle of the lock bit program, block erase or erase all unlocked block command <sup>(1)</sup></li> </ul>
1	0	Erase error	<ul style="list-style-type: none"> <li>• The block erase command is executed on a locked block</li> <li>• The block erase or erase all unlocked block command is executed on an unlock block and auto erase operation is not completed as expected <sup>(2)</sup></li> </ul>
0	1	Program error	<ul style="list-style-type: none"> <li>• The program command is executed on locked blocks</li> <li>• The program command is executed on unlocked blocks but program operation is not completed as expected</li> <li>• The lock bit program command is executed but program operation is not completed as expected <sup>(2)</sup></li> </ul>

**NOTES:**

1. The flash memory enters read array mode by writing command code “xxFFh” in the second bus cycle of these commands. 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.

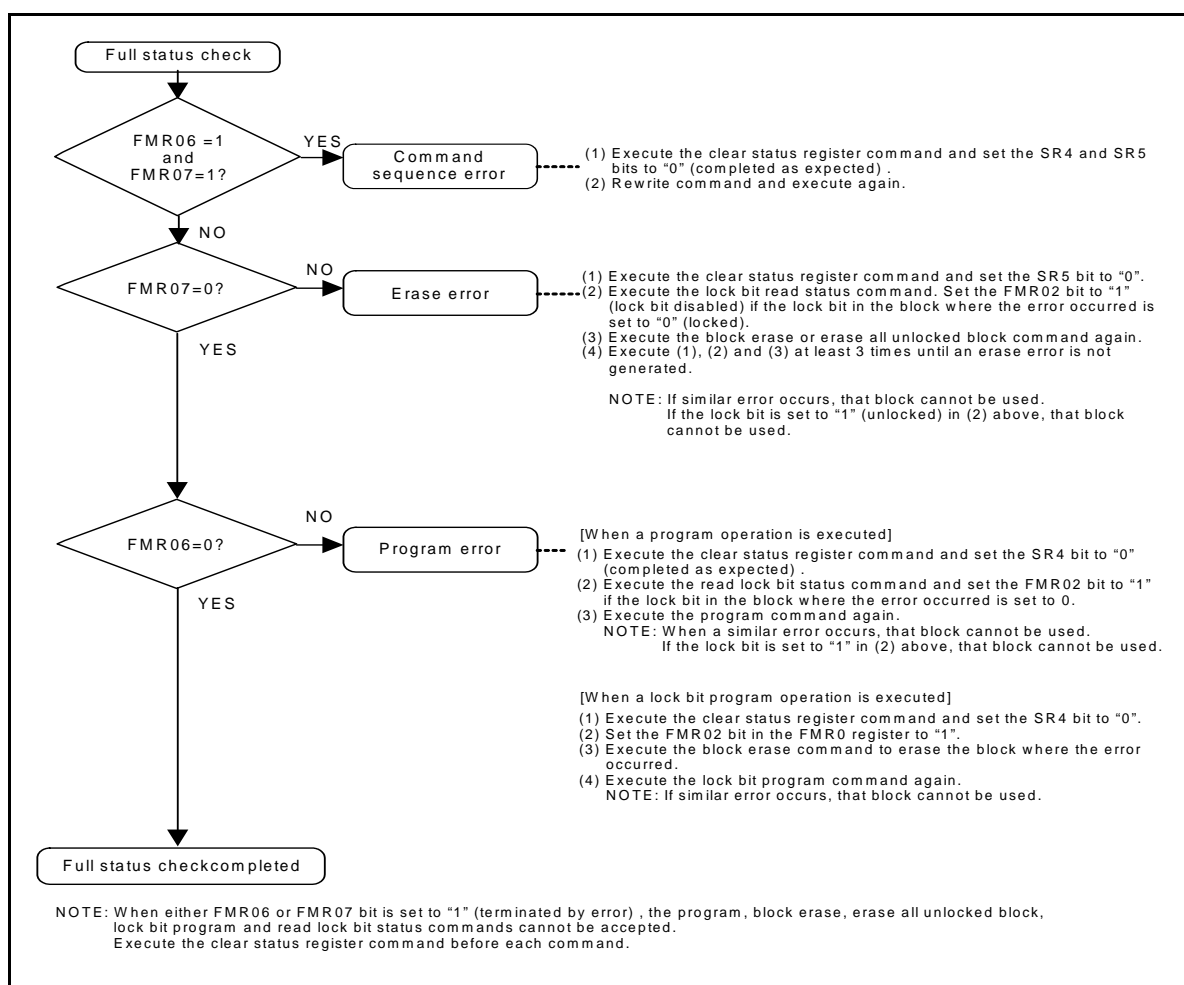


Figure 22.14 Full Status Check and Handling Procedure for Each Error

## 22.4 Standard Serial I/O Mode

In standard serial I/O mode, the serial programmer supporting the M16C/62P Group (M16C/62P, M16C/62PT) can be used to rewrite the flash memory user ROM area in the microcomputer mounted on a board.

For more information about the serial programmer, contact your serial programmer manufacturer. Refer to the user's manual included with your serial programmer for instructions.

Table 22.7 lists Pin Functions (Flash Memory Standard Serial I/O Mode). Figure 22.15 to Figure 22.18 show Pin Connections in Serial I/O Mode.

### 22.4.1 ID Code Check Function

The ID code check function determines whether the ID codes sent from the serial programmer matches those written in the flash memory. (Refer to **22.2 Functions To Prevent Flash Memory from Rewriting.**)

**Table 22.7 Pin Functions (Flash Memory Standard Serial I/O Mode)**

Pin	Name	I/O	Power Supply	Description
VCC1, VCC2, VSS	Power Input		—	Apply the Flash Program, Erase Voltage to VCC1 pin and VCC2 to the VCC2 pin. The VCC apply condition is that $VCC2 \leq VCC1$ . Apply 0 V to VSS pin.
CNVSS	CNVSS	I	VCC1	Connect to VCC1 pin.
RESET	Reset Input	I	VCC1	Reset input pin. While RESET pin is "L" level, input a 20 cycle or longer clock to XIN pin.
XIN	Clock Input	I	VCC1	Connect a ceramic resonator or crystal oscillator between XIN and XOUT pins. To input an externally generated clock, input it to XIN pin and open XOUT pin.
XOUT	Clock Output	O	VCC1	
BYTE	BYTE	I	VCC1	Connect this pin to VCC1 or VSS.
AVCC, AVSS	Analog Power Supply Input			Connect AVSS to VSS and AVCC to VCC1, respectively.
VREF	Reference Voltage Input	I		Enter the reference voltage for A/D from this pin.
P0_0 to P0_7	Input Port P0	I	VCC2	Input "H" or "L" level signal or open.
P1_0 to P1_7	Input Port P1	I	VCC2	Input "H" or "L" level signal or open.
P2_0 to P2_7	Input Port P2	I	VCC2	Input "H" or "L" level signal or open.
P3_0 to P3_7	Input Port P3	I	VCC2	Input "H" or "L" level signal or open.
P4_0 to P4_7	Input Port P4	I	VCC2	Input "H" or "L" level signal or open.
P5_1 to P5_4, P5_6, P5_7	Input Port P5	I	VCC2	Input "H" or "L" level signal or open.
P5_0	CE Input	I	VCC2	Input "H" level signal.
P5_5	EPM Input	I	VCC2	Input "L" level signal.
P6_0 to P6_3	Input Port P6	I	VCC1	Input "H" or "L" level signal or open.
P6_4/RTS1	BUSY Output	O	VCC1	Standard serial I/O mode 1: BUSY signal output pin Standard serial I/O mode 2: Monitors the boot program operation check signal output pin.
P6_5/CLK1	SCLK Input	I	VCC1	Standard serial I/O mode 1: Serial clock input pin Standard serial I/O mode 2: Input "L".
P6_6/RXD1	RXD Input	I	VCC1	Serial data input pin.
P6_7/TXD1	TXD Input	O	VCC1	Serial data output pin. (2)
P7_0 to P7_7	Input Port P7	I	VCC1	Input "H" or "L" level signal or open.
P8_0 to P8_3, P8_6, P8_7	Input Port P8	I	VCC1	Input "H" or "L" level signal or open.
P8_4	P8_4 input	I	VCC1	Input "L" level signal. (3)
P8_5/NMI	NMI Input	I	VCC1	Connect this pin to VCC1.
P9_0 to P9_7	Input Port P9	I	VCC1	Input "H" or "L" level signal or open.
P10_0 to P10_7	Input Port P10	I	VCC1	Input "H" or "L" level signal or open.
P11_0 to P11_7	Input Port P11	I	VCC1	Input "H" or "L" level signal or open. (1)
P12_0 to P12_7	Input Port P12	I	VCC2	Input "H" or "L" level signal or open. (1)
P13_0 to P13_7	Input Port P13	I	VCC2	Input "H" or "L" level signal or open. (1)
P14_0, P14_1	Input Port P14	I	VCC1	Input "H" or "L" level signal or open. (1)

**NOTES:**

1. Available in only the 128-pin version.
2. When using the standard serial I/O mode, the internal pull-up is enabled for the TXD1 (P6\_7) pin while the RESET pin is "L".
3. When using the standard serial I/O mode, the P0\_0 to P0\_7, P1\_0 to P1\_7 pins may become indeterminate while the P8\_4 pin is "H" and the RESET pin is "L". If this causes a program, apply "L" to the P8\_4 pin.

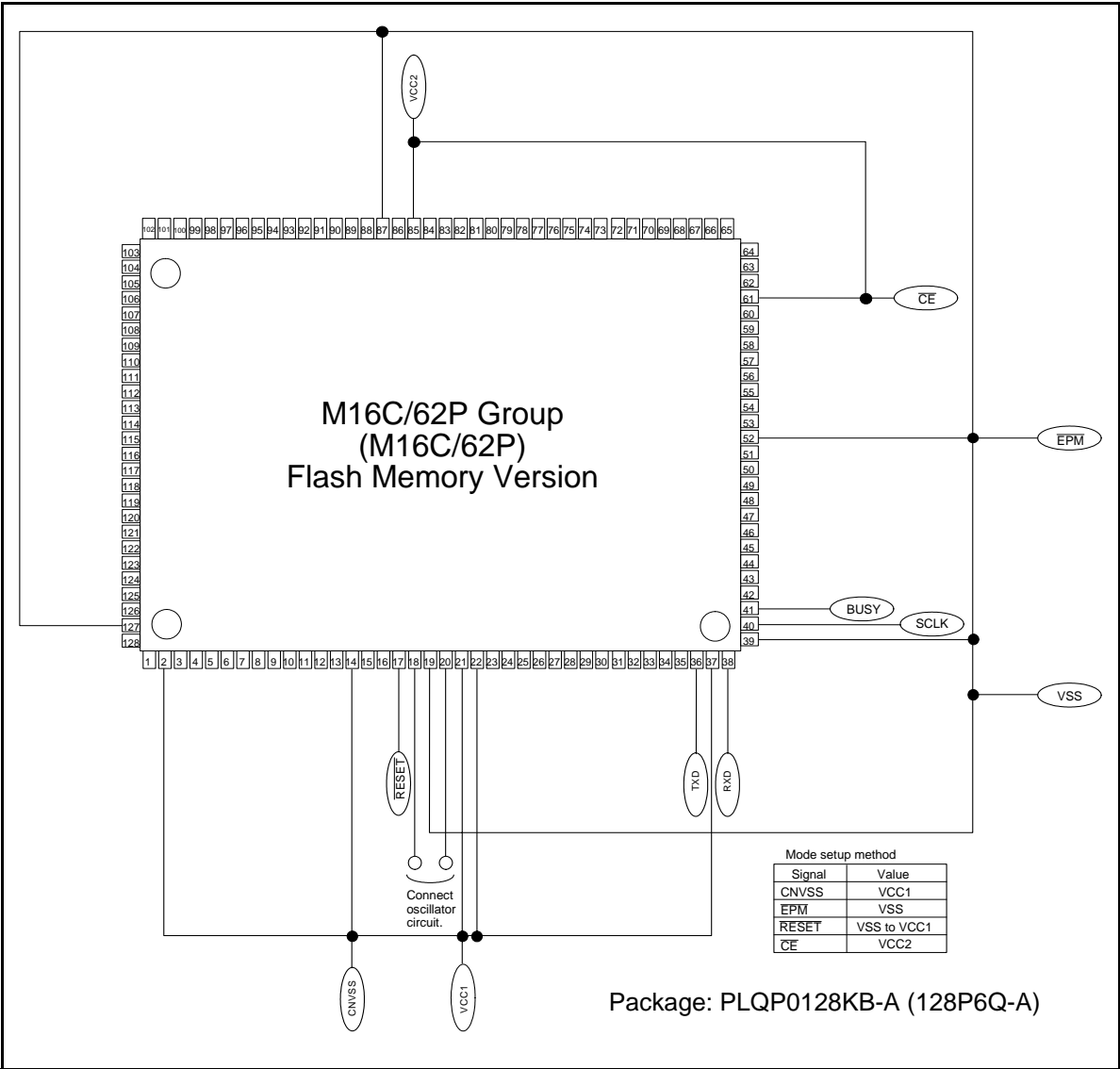


Figure 22.15 Pin Connections for Serial I/O Mode (1)



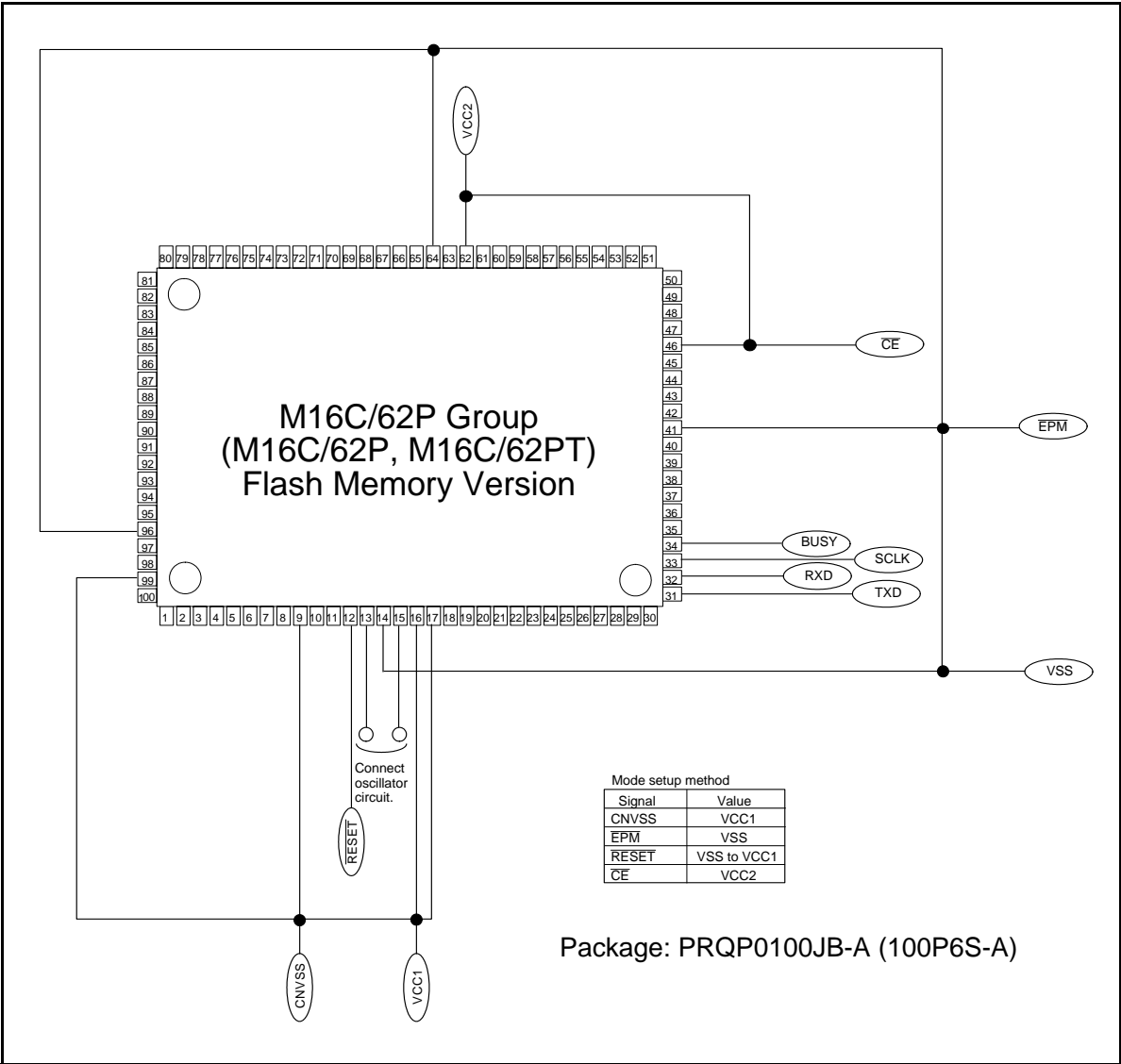


Figure 22.16 Pin Connections for Serial I/O Mode (2)

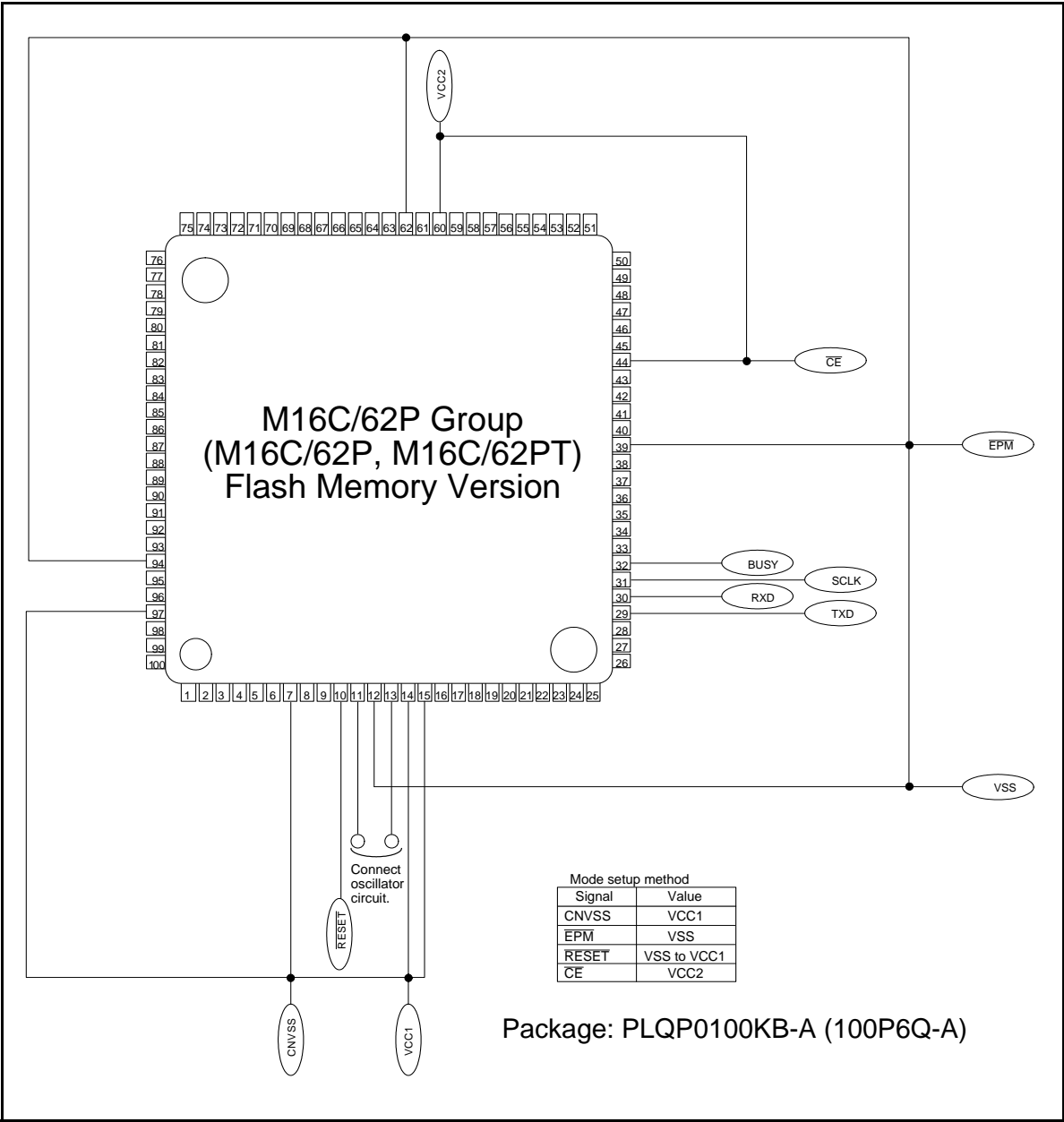


Figure 22.17 Pin Connections for Serial I/O Mode (3)

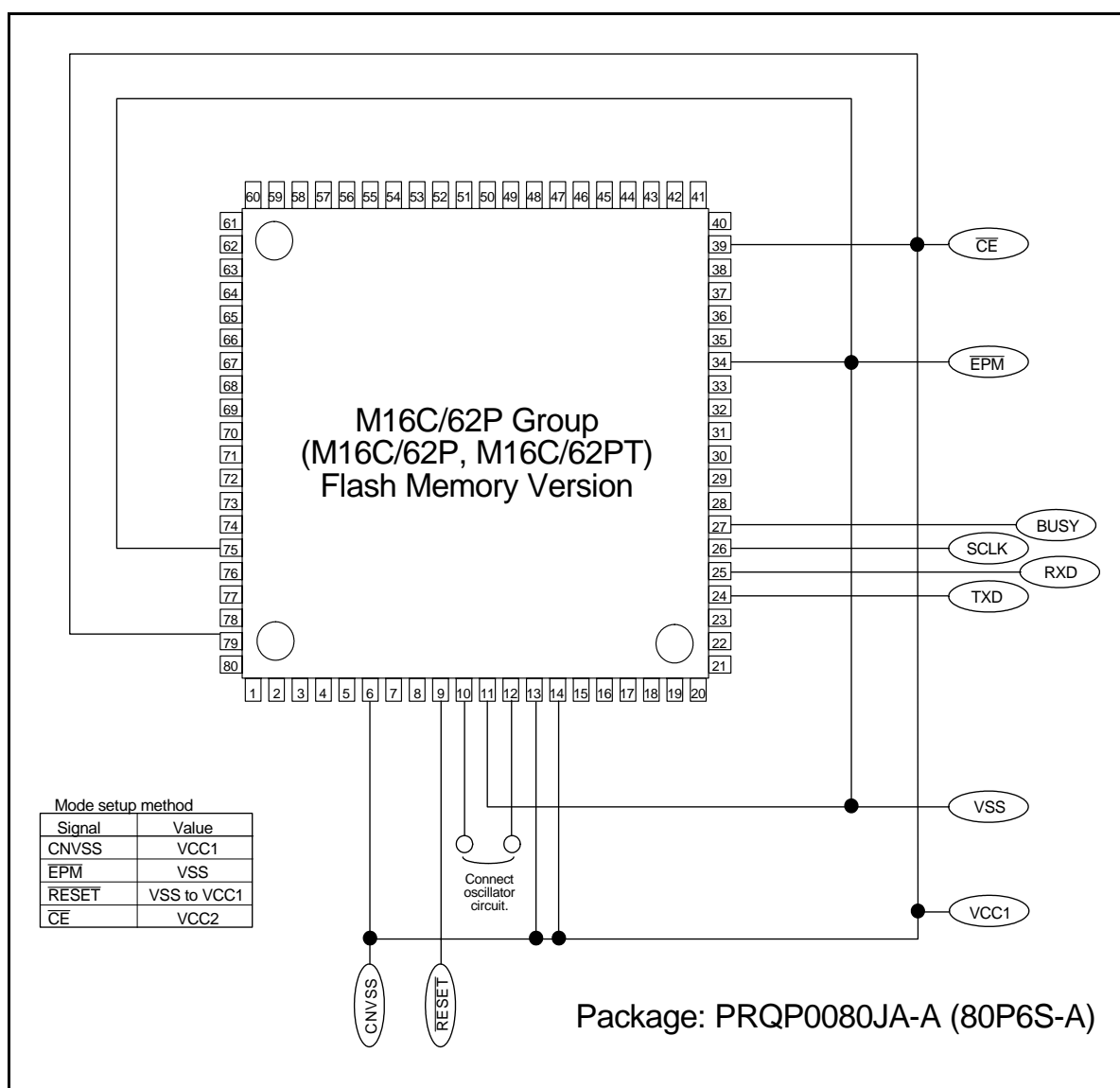


Figure 22.18 Pin Connections for Serial I/O Mode (4)

### 22.4.2 Example of Circuit Application in the Standard Serial I/O Mode

Figure 22.19 and Figure 22.20 show example of Circuit Application in Standard Serial I/O Mode 1 and Mode 2, respectively. Refer to the user's manual of your serial programmer to handle pins controlled by the serial programmer.

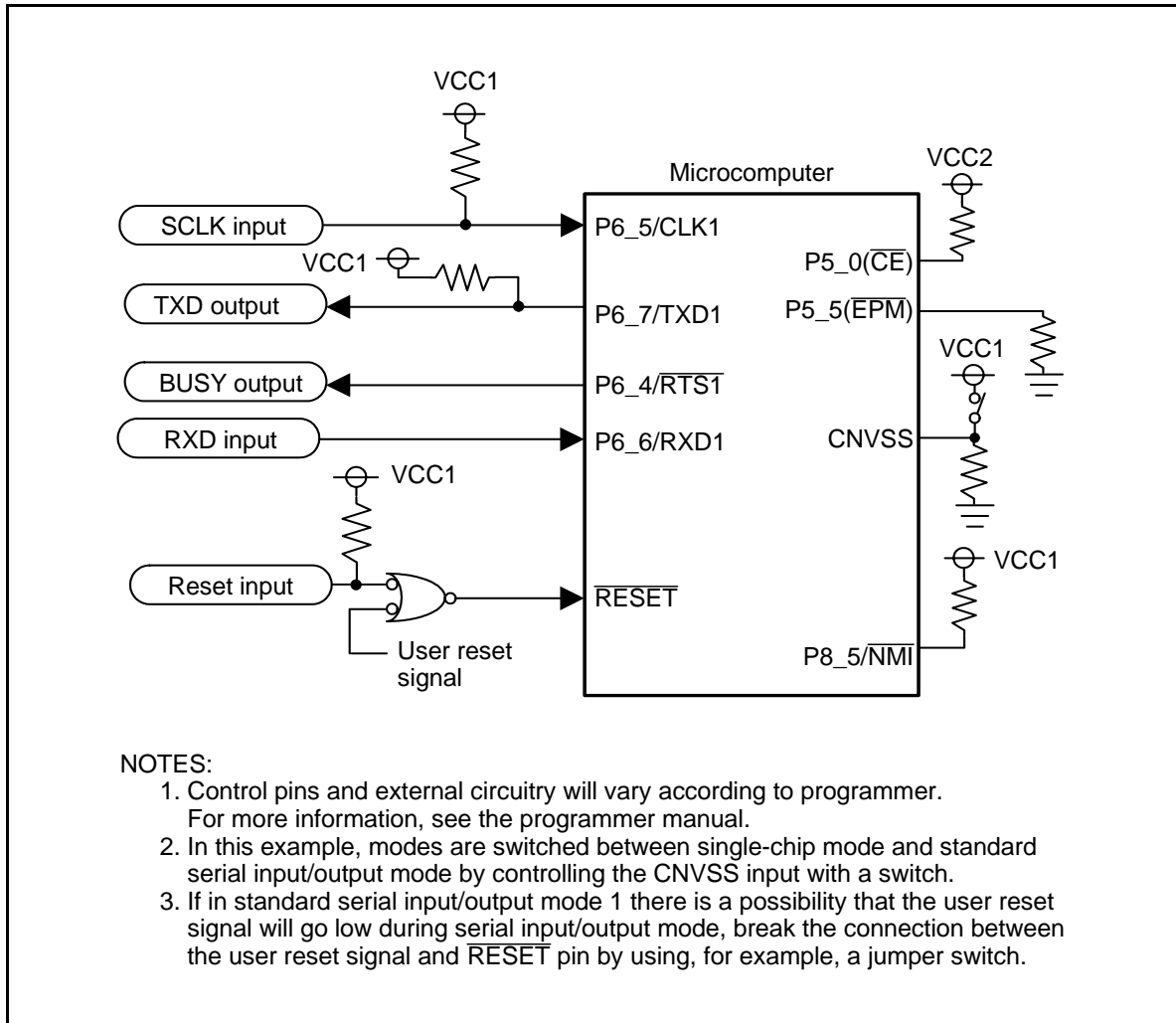


Figure 22.19 Circuit Application in Standard Serial I/O Mode 1

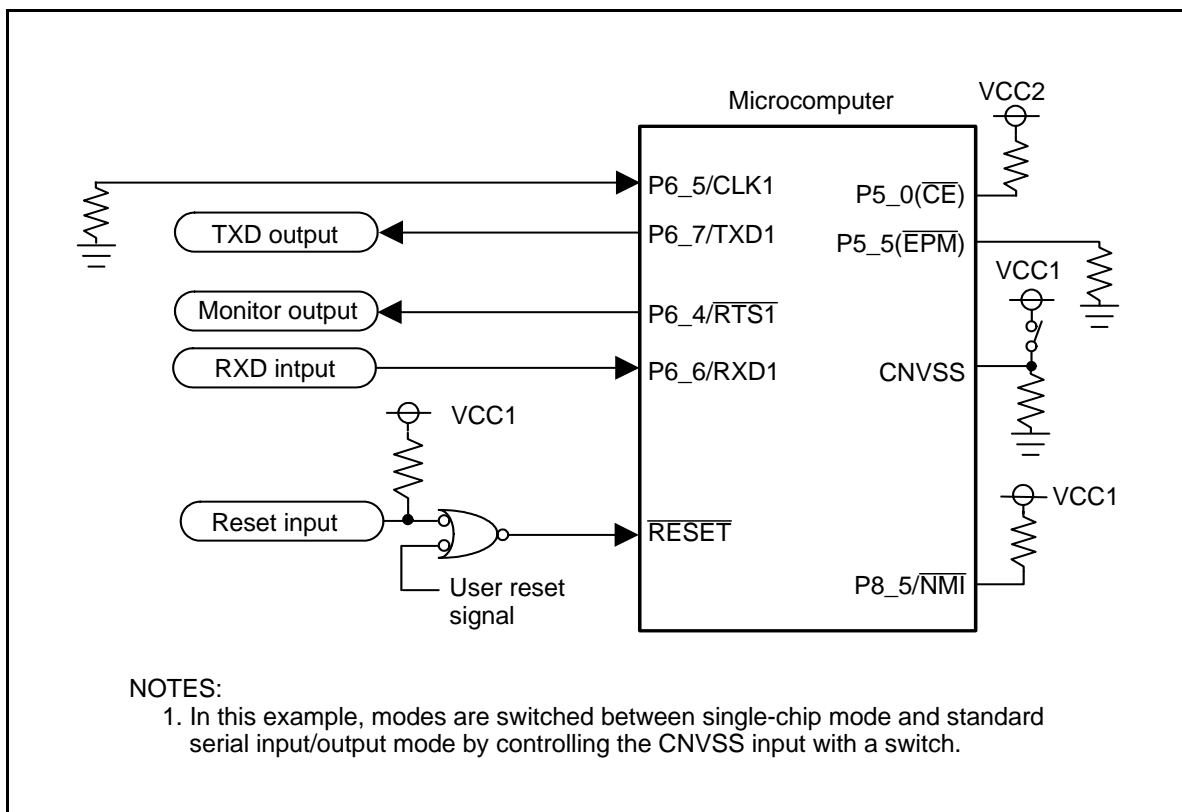


Figure 22.20 Circuit Application in Standard Serial I/o Mode 2

## 22.5 Parallel I/O Mode

In parallel I/O mode, the user ROM area and the boot ROM area can be rewritten by a parallel programmer supporting the M16C/62P Group (M16C/62P, M16C/62PT). Contact your parallel programmer manufacturer for more information on the parallel programmer. Refer to the user's manual included with your parallel programmer for instructions.

### 22.5.1 User ROM and Boot ROM Areas

An erase block operation in the boot ROM area is applied to only one 4 Kbyte block. The rewrite control program in standard serial I/O mode is written in the boot ROM area before shipment. Do not rewrite the boot ROM area if using the serial programmer.

In parallel I/O mode, the boot ROM area is located in addresses 0FF000h to 0FFFFFFh. Rewrite this address range only if rewriting the boot ROM area. (Do not access addresses other than addresses 0FF000h to 0FFFFFFh.)

### 22.5.2 ROM Code Protect Function

The ROM code protect function prevents the flash memory from being read and rewritten in parallel I/O mode. (Refer to **22.2 Functions To Prevent Flash Memory from Rewriting**.)

## 23. Electrical Characteristics

### 23.1 Electrical Characteristics (M16C/62P)

**Table 23.1 Absolute Maximum Ratings**

Symbol	Parameter		Condition	Rated Value	Unit
Vcc1, Vcc2	Supply Voltage		Vcc1=AVcc	–0.3 to 6.5	V
Vcc2	Supply Voltage		Vcc2	–0.3 to Vcc1+0.1	V
AVcc	Analog Supply Voltage		Vcc1=AVcc	–0.3 to 6.5	V
Vi	Input Voltage	RESET, CNVSS, BYTE, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1, VREF, XIN		–0.3 to Vcc1+0.3 <sup>(1)</sup>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7		–0.3 to Vcc2+0.3 <sup>(1)</sup>	V
		P7_0, P7_1		–0.3 to 6.5	V
Vo	Output Voltage	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, P11_0 to P11_7, P14_0, P14_1, XOUT		–0.3 to Vcc1+0.3 <sup>(1)</sup>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7		–0.3 to Vcc2+0.3 <sup>(1)</sup>	V
		P7_0, P7_1		–0.3 to 6.5	V
Pd	Power Dissipation		–40°C<T <sub>opr</sub> ≤85°C	300	mW
T <sub>opr</sub>	Operating Ambient Temperature	When the Microcomputer is Operating		–20 to 85 / –40 to 85	°C
		Flash Program Erase		0 to 60	
T <sub>stg</sub>	Storage Temperature			–65 to 150	°C

**NOTES:**

1. There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.

**Table 23.2 Recommended Operating Conditions (1) (1)**

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
V <sub>CC1</sub> , V <sub>CC2</sub>	Supply Voltage (V <sub>CC1</sub> ≥ V <sub>CC2</sub> )		2.7	5.0	5.5	V
AV <sub>CC</sub>	Analog Supply Voltage			V <sub>CC1</sub>		V
V <sub>SS</sub>	Supply Voltage			0		V
AV <sub>SS</sub>	Analog Supply Voltage			0		V
V <sub>IH</sub>	HIGH Input Voltage	P3_1 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	0.8V <sub>CC2</sub>		V <sub>CC2</sub>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (during single-chip mode)	0.8V <sub>CC2</sub>		V <sub>CC2</sub>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (data input during memory expansion and microprocessor mode)	0.5V <sub>CC2</sub>		V <sub>CC2</sub>	V
		P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	0.8V <sub>CC1</sub>		V <sub>CC1</sub>	V
		P7_0, P7_1	0.8V <sub>CC1</sub>		6.5	V
V <sub>IL</sub>	LOW Input Voltage	P3_1 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	0		0.2V <sub>CC2</sub>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (during single-chip mode)	0		0.2V <sub>CC2</sub>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (data input during memory expansion and microprocessor mode)	0		0.16V <sub>CC2</sub>	V
		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, P11_0 to P11_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	0		0.2V <sub>CC</sub>	V
I <sub>OH(peak)</sub>	HIGH Peak Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			−10.0	mA
I <sub>OH(avg)</sub>	HIGH Average Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			−5.0	mA
I <sub>OL(peak)</sub>	LOW Peak Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			10.0	mA
I <sub>OL(avg)</sub>	LOW Average Output Current	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			5.0	mA

## NOTES:

1. Referenced to V<sub>CC1</sub> = V<sub>CC2</sub> = 2.7 to 5.5V at T<sub>opr</sub> = −20 to 85°C / −40 to 85°C unless otherwise specified.
2. The Average Output Current is the mean value within 100ms.
3. The total I<sub>OL(peak)</sub> for ports P0, P1, P2, P8\_6, P8\_7, P9, P10, P11, P14\_0, and P14\_1 must be 80mA max. The total I<sub>OL(peak)</sub> for ports P3, P4, P5, P6, P7, P8\_0 to P8\_4, P12, and P13 must be 80mA max. The total I<sub>OH(peak)</sub> for ports P0, P1, and P2 must be −40mA max. The total I<sub>OH(peak)</sub> for ports P3, P4, P5, P12, and P13 must be −40mA max. The total I<sub>OH(peak)</sub> for ports P6, P7, and P8\_0 to P8\_4 must be −40mA max. The total I<sub>OH(peak)</sub> for ports P8\_6, P8\_7, P9, P10, P14\_0, and P14\_1 must be −40mA max. Set Average Output Current to 1/2 of peak. The total I<sub>OH(peak)</sub> for ports P8\_6, P8\_7, P9, P10, P11, P14\_0, and P14\_1 must be −40mA max.  
As for 80-pin version, the total I<sub>OL(peak)</sub> for all ports and I<sub>OH(peak)</sub> must be 80mA. max. due to one V<sub>CC</sub> and one V<sub>SS</sub>.
4. There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.

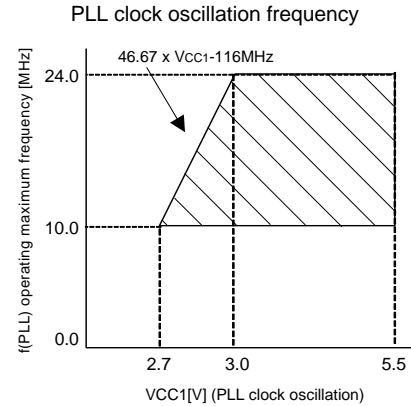
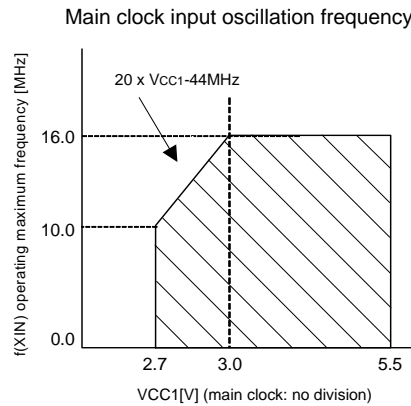


**Table 23.3 Recommended Operating Conditions (2) <sup>(1)</sup>**

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
f(XIN)	Main Clock Input Oscillation Frequency <sup>(2)</sup>				
	VCC1=3.0V to 5.5V	0		16	MHz
	VCC1=2.7V to 3.0V	0		20×VCC1 –44	MHz
f(XCIN)	Sub-Clock Oscillation Frequency		32.768	50	kHz
f(Ring)	On-chip Oscillation Frequency	0.5	1	2	MHz
f(PLL)	PLL Clock Oscillation Frequency <sup>(2)</sup>				
	VCC1=3.0V to 5.5V	10		24	MHz
	VCC1=2.7V to 3.0V	10		46.67×VCC1 –116	MHz
f(BCLK)	CPU Operation Clock	0		24	MHz
tsu(PLL)	PLL Frequency Synthesizer Stabilization Wait Time				
	VCC1=5.5V			20	ms
	VCC1=3.0V			50	ms

## NOTES:

1. Referenced to VCC1 = VCC2 = 2.7 to 5.5V at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified.
2. Relationship between main clock oscillation frequency, and supply voltage.



**Table 23.4 A/D Conversion Characteristics (1)**

Symbol	Parameter		Measuring Condition		Standard			Unit
					Min.	Typ.	Max.	
–	Resolution		VREF=VCC1				10	Bits
INL	Integral Non-Linearity Error	10bit	VREF=VCC1=5V	AN0 to AN7 input, AN0_0 to AN0_7 input, AN2_0 to AN2_7 input, ANEX0, ANEX1 input			±3	LSB
				External operation amp connection mode			±7	LSB
			VREF=VCC1=3.3V	AN0 to AN7 input, AN0_0 to AN0_7 input, AN2_0 to AN2_7 input, ANEX0, ANEX1 input			±5	LSB
				External operation amp connection mode			±7	LSB
		8bit	VREF=VCC1=5V, 3.3V				±2	LSB
–	Absolute Accuracy	10bit	VREF=VCC1=5V	AN0 to AN7 input, AN0_0 to AN0_7 input, AN2_0 to AN2_7 input, ANEX0, ANEX1 input			±3	LSB
				External operation amp connection mode			±7	LSB
			VREF=VCC1=3.3V	AN0 to AN7 input, AN0_0 to AN0_7 input, AN2_0 to AN2_7 input, ANEX0, ANEX1 input			±5	LSB
				External operation amp connection mode			±7	LSB
		8bit	VREF=VCC1=5V, 3.3V				±2	LSB
–	Tolerance Level Impedance					3		kΩ
DNL	Differential Non-Linearity Error						±1	LSB
–	Offset Error						±3	LSB
–	Gain Error						±3	LSB
RLADDER	Ladder Resistance		VREF=VCC1		10		40	kΩ
tCONV	10-bit Conversion Time, Sample & Hold Available		VREF=VCC1=5V, ϕAD=12MHz		2.75			μs
tCONV	8-bit Conversion Time, Sample & Hold Available		VREF=VCC1=5V, ϕAD=12MHz		2.33			μs
tsAMP	Sampling Time				0.25			μs
VREF	Reference Voltage				2.0		VCC1	V
VIA	Analog Input Voltage				0		VREF	V

**NOTES:**

1. Referenced to VCC1=AVCC=VREF=3.3 to 5.5V, VSS=AVSS=0V at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified.
2. If VCC1 > VCC2, do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.
3. φAD frequency must be 12 MHz or less. And divide the fAD if VCC1 is less than 4.0V, and φAD frequency into 10 MHz or less.
4. When sample & hold is disabled, φAD frequency must be 250 kHz or more, in addition to the limitation in Note 3.  
When sample & hold is enabled, φAD frequency must be 1MHz or more, in addition to the limitation in Note 3.

**Table 23.5 D/A Conversion Characteristics** <sup>(1)</sup>

Symbol	Parameter	Measuring Condition	Standard			Unit
			Min.	Typ.	Max.	
–	Resolution				8	Bits
–	Absolute Accuracy				1.0	%
t <sub>su</sub>	Setup Time				3	μs
R <sub>o</sub>	Output Resistance		4	10	20	kΩ
I <sub>VREF</sub>	Reference Power Supply Input Current	(NOTE 2)			1.5	mA

## NOTES:

1. Referenced to V<sub>CC1</sub>=V<sub>REF</sub>=3.3 to 5.5V, V<sub>SS</sub>=A<sub>VSS</sub>=0V at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified.
2. This applies when using one D/A converter, with the D/A register for the unused D/A converter set to “00h”. The resistor ladder of the A/D converter is not included. Also, when D/A register contents are not “00h”, the I<sub>VREF</sub> will flow even if V<sub>ref</sub> is disconnected by the A/D control register.

**Table 23.6 Flash Memory Version Electrical Characteristics <sup>(1)</sup> for 100 cycle products (D3, D5, U3, U5)**

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
–	Program and Erase Endurance <sup>(3)</sup>	100			cycle
–	Word Program Time (V <sub>CC1</sub> =5.0V)		25	200	μs
–	Lock Bit Program Time		25	200	μs
–	Block Erase Time (V <sub>CC1</sub> =5.0V)		0.3	4	s
–			0.3	4	s
–			0.5	4	s
–			0.8	4	s
–	Erase All Unlocked Blocks Time <sup>(2)</sup>			4xn	s
tps	Flash Memory Circuit Stabilization Wait Time			15	μs
–	Data Hold Time <sup>(5)</sup>	10			year

**Table 23.7 Flash Memory Version Electrical Characteristics <sup>(6)</sup> for 10,000 cycle products (D7, D9, U7, U9) (Block A and Block 1 <sup>(7)</sup>)**

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
–	Program and Erase Endurance <sup>(3, 8, 9)</sup>	10,000 <sup>(4)</sup>			cycle
–	Word Program Time (V <sub>CC1</sub> =5.0V)		25		μs
–	Lock Bit Program Time		25		μs
–	Block Erase Time (V <sub>CC1</sub> =5.0V)	4-Kbyte block	0.3		s
tps	Flash Memory Circuit Stabilization Wait Time			15	μs
–	Data Hold Time <sup>(5)</sup>	10			year

## NOTES:

1. Referenced to V<sub>CC1</sub>=4.5 to 5.5V, 3.0 to 3.6V at T<sub>opr</sub> = 0 to 60 °C (D3, D5, U3, U5) unless otherwise specified.
2. n denotes the number of block erases.
3. Program and Erase Endurance refers to the number of times a block erase can be performed.  
If the program and erase endurance is n (n=100, 1,000, or 10,000), each block can be erased n times.  
For example, if a 4 Kbytes block A is erased after writing 1 word data 2,048 times, each to a different address, this counts as one program and erase endurance. Data cannot be written to the same address more than once without erasing the block.  
(Rewrite prohibited)
4. Maximum number of E/W cycles for which operation is guaranteed.
5. T<sub>opr</sub> = -40 to 85 °C (D3, D7, U3, U7) / -20 to 85 °C (D5, D9, U5, U9).
6. Referenced to V<sub>CC1</sub> = 4.5 to 5.5V, 3.0 to 3.6V at T<sub>opr</sub> = -40 to 85 °C (D7, U7) / -20 to 85 °C (D9, U9) unless otherwise specified.
7. Table 23.7 applies for block A or block 1 program and erase endurance > 1,000. Otherwise, use Table 23.6.
8. To reduce the number of program and erase endurance when working with systems requiring numerous rewrites, write to unused word addresses within the block instead of rewrite. Erase block only after all possible addresses are used. For example, an 8-word program can be written 256 times maximum before erase becomes necessary.  
Maintaining an equal number of erasure between block A and block 1 will also improve efficiency. It is important to track the total number of times erasure is used.
9. Should erase error occur during block erase, attempt to execute clear status register command, then block erase command at least three times until erase error disappears.
10. Set the PM17 bit in the PM1 register to "1" (wait state) when executing more than 100 times rewrites (D7, D9, U7 and U9).
11. Customers desiring E/W failure rate information should contact their Renesas technical support representative.

**Table 23.8 Flash Memory Version Program / Erase Voltage and Read Operation Voltage Characteristics (at T<sub>opr</sub> = 0 to 60 °C(D3, D5, U3, U5), T<sub>opr</sub> = -40 to 85 °C(D7, U7) / T<sub>opr</sub> = -20 to 85 °C(D9, U9))**

Flash Program, Erase Voltage	Flash Read Operation Voltage
V <sub>CC1</sub> = 3.3 V ± 0.3 V or 5.0 V ± 0.5 V	V <sub>CC1</sub> =2.7 to 5.5 V

**Table 23.9 Low Voltage Detection Circuit Electrical Characteristics**

Symbol	Parameter	Measuring Condition	Standard			Unit
			Min.	Typ.	Max.	
V <sub>det4</sub>	Low Voltage Detection Voltage <sup>(1)</sup>	V <sub>CC1</sub> =0.8V to 5.5V	3.3	3.8	4.4	V
V <sub>det3</sub>	Reset Level Detection Voltage <sup>(1, 2)</sup>		2.2	2.8	3.6	V
V <sub>det4</sub> -V <sub>det3</sub>	Electric potential difference of Low Voltage Detection and Reset Level Detection		0.3			V
V <sub>det3s</sub>	Low Voltage Reset Retention Voltage				0.8	V
V <sub>det3r</sub>	Low Voltage Reset Release Voltage <sup>(3)</sup>		2.2	2.9	4.0	V

## NOTES:

1. V<sub>det4</sub> > V<sub>det3</sub>.
2. Where reset level detection voltage is less than 2.7 V, if the supply power voltage is greater than the reset level detection voltage, the microcomputer operates with f(BCLK) ≤ 10MHz.
3. V<sub>det3r</sub> > V<sub>det3</sub> is not guaranteed.
4. The voltage detection circuit is designed to use when V<sub>CC1</sub> is set to 5V.

**Table 23.10 Power Supply Circuit Timing Characteristics**

Symbol	Parameter	Measuring Condition	Standard			Unit
			Min.	Typ.	Max.	
t <sub>d</sub> (P-R)	Time for Internal Power Supply Stabilization During Powering-On	V <sub>CC1</sub> =2.7V to 5.5V			2	ms
t <sub>d</sub> (R-S)	STOP Release Time				150	μs
t <sub>d</sub> (W-S)	Low Power Dissipation Mode Wait Mode Release Time				150	μs
t <sub>d</sub> (S-R)	Brown-out Detection Reset (Hardware Reset 2) Release Wait Time	V <sub>CC1</sub> =V <sub>det3r</sub> to 5.5V		6 <sup>(1)</sup>	20	ms
t <sub>d</sub> (E-A)	Low Voltage Detection Circuit Operation Start Time	V <sub>CC1</sub> =2.7V to 5.5V			20	μs

## NOTES:

1. When V<sub>CC1</sub> = 5V.

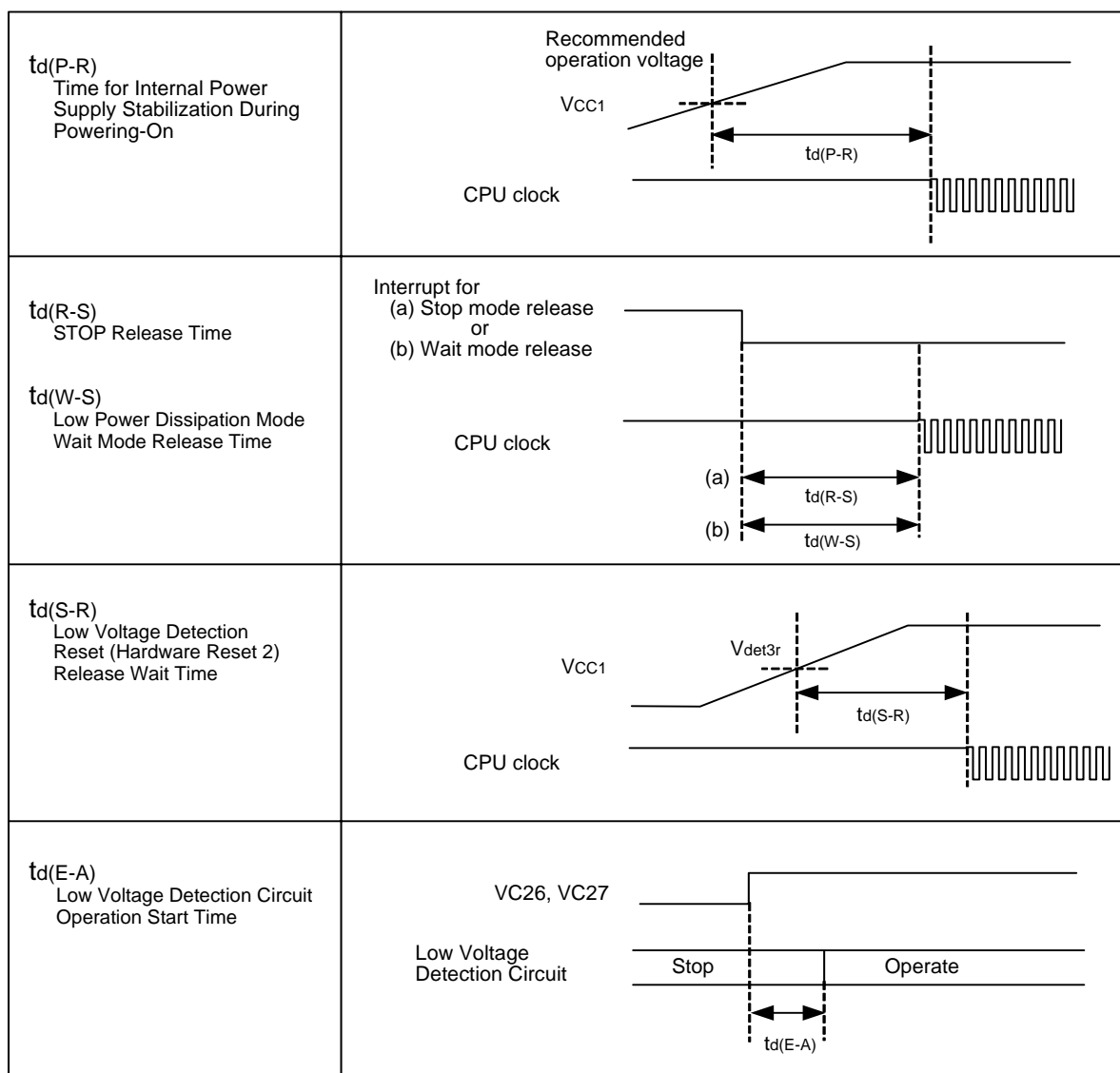


Figure 23.1 Power Supply Circuit Timing Diagram

$$V_{CC1}=V_{CC2}=5V$$

Table 23.11 Electrical Characteristics (1) <sup>(1)</sup>

Symbol	Parameter		Measuring Condition	Standard			Unit
				Min.	Typ.	Max.	
VOH	HIGH Output Voltage <sup>(3)</sup>	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, P11_0 to P11_7, P14_0, P14_1	IOH=−5mA	VCC1−2.0		VCC1	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOH=−5mA <sup>(2)</sup>	VCC2−2.0		VCC2	
VOH	HIGH Output Voltage <sup>(3)</sup>	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, P11_0 to P11_7, P14_0, P14_1	OH=−200μA	VCC1−0.3		VCC1	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOH=−200μA <sup>(2)</sup>	VCC2−0.3		VCC2	
VOH	HIGH Output Voltage XOUT	HIGHPOWER	IOH=−1mA	VCC1−2.0		VCC1	V
		LOWPOWER	IOH=−0.5mA	VCC1−2.0		VCC1	
	HIGH Output Voltage XCOUT	HIGHPOWER	With no load applied		2.5		V
		LOWPOWER	With no load applied		1.6		
VOL	LOW Output Voltage <sup>(3)</sup>	P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1	IOL=5mA			2.0	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOL=5mA <sup>(2)</sup>			2.0	
VOL	LOW Output Voltage <sup>(3)</sup>	P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1	IOL=200μA			0.45	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOL=200μA <sup>(2)</sup>			0.45	
VOL	LOW Output Voltage XOUT	HIGHPOWER	IOL=1mA			2.0	V
		LOWPOWER	IOL=0.5mA			2.0	
	LOW Output Voltage XCOUT	HIGHPOWER	With no load applied		0		V
		LOWPOWER	With no load applied		0		
VT+−VT−	Hysteresis	HOLD, RDY, TA0IN to TA4IN, TB0IN to TB5IN, INT0 to INT5, NMI, ADTRG, CTS0 to CTS2, CLK0 to CLK4, TA0OUT to TA4OUT, K10 to K13, RXD0 to RXD2, SCL0 to SCL2, SDA0 to SDA2, SIN3, SIN4		0.2		1.0	V
VT+−VT−	Hysteresis	RESET		0.2		2.5	V
IiH	HIGH Input Current <sup>(3)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	Vi=5V			5.0	μA
IiL	LOW Input Current <sup>(3)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	Vi=0V			−5.0	μA
RPULLUP	Pull-Up Resistance <sup>(3)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1	Vi=0V	30	50	170	kΩ
RiXIN	Feedback Resistance XIN				1.5		MΩ
RiXCIN	Feedback Resistance XCIN				15		MΩ
VRAM	RAM Retention Voltage		At stop mode	2.0			V

## NOTES:

1. Referenced to VCC1=VCC2=4.2 to 5.5V, VSS = 0V at T<sub>opr</sub> = −20 to 85°C / −40 to 85°C, f(BCLK)=24MHz unless otherwise specified.
2. Where the product is used at VCC1 = 5 V and VCC2 = 3 V, refer to the 3 V version value for the pin specified value on VCC2 port side.
3. There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.

**Table 23.12 Electrical Characteristics (2) <sup>(1)</sup>**

Symbol	Parameter		Measuring Condition		Standard			Unit
					Min.	Typ.	Max.	
Icc	Power Supply Current (VCC1=VCC2=4.0V to 5.5V)	In single-chip mode, the output pins are open and other pins are Vss	Mask ROM	f(BCLK)=24MHz No division, PLL operation		14	20	mA
				No division, On-chip oscillation		1		mA
			Flash Memory	f(BCLK)=24MHz, No division, PLL operation		18	27	mA
				No division, On-chip oscillation		1.8		mA
			Flash Memory Program	f(BCLK)=10MHz, VCC1=5.0V		15		mA
			Flash Memory Erase	f(BCLK)=10MHz, VCC1=5.0V		25		mA
			Mask ROM	f(XCIN)=32kHz Low power dissipation mode, ROM <sup>(3)</sup>		25		μA
			Flash Memory	f(BCLK)=32kHz Low power dissipation mode, RAM <sup>(3)</sup>		25		μA
				f(BCLK)=32kHz Low power dissipation mode, Flash Memory <sup>(3)</sup>		420		μA
				On-chip oscillation, Wait mode		50		μA
			Mask ROM Flash Memory	f(BCLK)=32kHz Wait mode <sup>(2)</sup> , Oscillation capability High		7.5		μA
				f(BCLK)=32kHz Wait mode <sup>(2)</sup> , Oscillation capability Low		2.0		μA
				Stop mode Topr =25°C		0.8	3.0	μA
Idet4	Low Voltage Detection Dissipation Current <sup>(4)</sup>					0.7	4	μA
Idet3	Reset Area Detection Dissipation Current <sup>(4)</sup>					1.2	8	μA

## NOTES:

1. Referenced to V<sub>CC1</sub>=V<sub>CC2</sub>=4.2 to 5.5V, V<sub>SS</sub> = 0V at T<sub>opr</sub> = -20 to 85°C / -40 to 85°C, f(BCLK)=24MHz unless otherwise specified.
2. With one timer operated using fC32.
3. This indicates the memory in which the program to be executed exists.
4. I<sub>det</sub> is dissipation current when the following bit is set to "1" (detection circuit enabled).  
I<sub>det4</sub>: VC27 bit in the VCR2 register  
I<sub>det3</sub>: VC26 bit in the VCR2 register



$$V_{CC1}=V_{CC2}=5V$$

**Timing Requirements**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.13 External Clock Input (XIN input) <sup>(1)</sup>**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub>	External Clock Input Cycle Time	62.5		ns
t <sub>w(H)</sub>	External Clock Input HIGH Pulse Width	25		ns
t <sub>w(L)</sub>	External Clock Input LOW Pulse Width	25		ns
t <sub>r</sub>	External Clock Rise Time		15	ns
t <sub>f</sub>	External Clock Fall Time		15	ns

**NOTES:**

1. The condition is V<sub>CC1</sub>=V<sub>CC2</sub>=3.0 to 5.0V.

**Table 23.14 Memory Expansion Mode and Microprocessor Mode**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>ac1</sub> (RD-DB)	Data Input Access Time (for setting with no wait)		(NOTE 1)	ns
t <sub>ac2</sub> (RD-DB)	Data Input Access Time (for setting with wait)		(NOTE 2)	ns
t <sub>ac3</sub> (RD-DB)	Data Input Access Time (when accessing multiplex bus area)		(NOTE 3)	ns
t <sub>su</sub> (DB-RD)	Data Input Setup Time	40		ns
t <sub>su</sub> (RDY-BCLK)	RDY Input Setup Time	30		ns
t <sub>su</sub> (HOLD-BCLK)	HOLD Input Setup Time	40		ns
t <sub>h</sub> (RD-DB)	Data Input Hold Time	0		ns
t <sub>h</sub> (BCLK-RDY)	RDY Input Hold Time	0		ns
t <sub>h</sub> (BCLK-HOLD)	HOLD Input Hold Time	0		ns

**NOTES:**

1. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 45[\text{ns}]$$

2. Calculated according to the BCLK frequency as follows:

$$\frac{(n - 0.5) \times 10^9}{f(\text{BCLK})} - 45[\text{ns}] \quad n \text{ is "2" for 1-wait setting, "3" for 2-wait setting and "4" for 3-wait setting.}$$

3. Calculated according to the BCLK frequency as follows:

$$\frac{(n - 0.5) \times 10^9}{f(\text{BCLK})} - 45[\text{ns}] \quad n \text{ is "2" for 2-wait setting, "3" for 3-wait setting.}$$

$$V_{CC1}=V_{CC2}=5V$$

**Timing Requirements**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.15 Timer A Input (Counter Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	100		ns
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	40		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	40		ns

**Table 23.16 Timer A Input (Gating Input in Timer Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	400		ns
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	200		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	200		ns

**Table 23.17 Timer A Input (External Trigger Input in One-shot Timer Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	200		ns
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	100		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	100		ns

**Table 23.18 Timer A Input (External Trigger Input in Pulse Width Modulation Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	100		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	100		ns

**Table 23.19 Timer A Input (Counter Increment/Decrement Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (UP)	TAiOUT Input Cycle Time	2000		ns
t <sub>w</sub> (UPH)	TAiOUT Input HIGH Pulse Width	1000		ns
t <sub>w</sub> (UPL)	TAiOUT Input LOW Pulse Width	1000		ns
t <sub>su</sub> (UP-TIN)	TAiOUT Input Setup Time	400		ns
t <sub>h</sub> (TIN-UP)	TAiOUT Input Hold Time	400		ns

**Table 23.20 Timer A Input (Two-phase Pulse Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	800		ns
t <sub>su</sub> (TAIN-TAOUT)	TAiOUT Input Setup Time	200		ns
t <sub>su</sub> (TAOUT-TAIN)	TAiIN Input Setup Time	200		ns

$$V_{CC1}=V_{CC2}=5V$$

**Timing Requirements**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.21 Timer B Input (Counter Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TB)	TBiIN Input Cycle Time (counted on one edge)	100		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width (counted on one edge)	40		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width (counted on one edge)	40		ns
t <sub>c</sub> (TB)	TBiIN Input Cycle Time (counted on both edges)	200		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width (counted on both edges)	80		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width (counted on both edges)	80		ns

**Table 23.22 Timer B Input (Pulse Period Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TB)	TBiIN Input Cycle Time	400		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width	200		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width	200		ns

**Table 23.23 Timer B Input (Pulse Width Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TB)	TBiIN Input Cycle Time	400		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width	200		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width	200		ns

**Table 23.24 A/D Trigger Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (AD)	ADTRG Input Cycle Time	1000		ns
t <sub>w</sub> (ADL)	ADTRG input LOW Pulse Width	125		ns

**Table 23.25 Serial Interface**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (CK)	CLKi Input Cycle Time	200		ns
t <sub>w</sub> (CKH)	CLKi Input HIGH Pulse Width	100		ns
t <sub>w</sub> (CKL)	CLKi Input LOW Pulse Width	100		ns
t <sub>d</sub> (C-Q)	TXDi Output Delay Time		80	ns
t <sub>h</sub> (C-Q)	TXDi Hold Time	0		ns
t <sub>su</sub> (D-C)	RXDi Input Setup Time	70		ns
t <sub>h</sub> (C-D)	RXDi Input Hold Time	90		ns

**Table 23.26 External Interrupt INTi Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>w</sub> (INH)	INTi Input HIGH Pulse Width	250		ns
t <sub>w</sub> (INL)	INTi Input LOW Pulse Width	250		ns

$$V_{CC1}=V_{CC2}=5V$$

**Switching Characteristics**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.27 Memory Expansion and Microprocessor Modes (for setting with no wait)**

Symbol	Parameter		Standard		Unit
			Min.	Max.	
t <sub>d</sub> (BCLK-AD)	Address Output Delay Time	See Figure 23.2		25	ns
t <sub>h</sub> (BCLK-AD)	Address Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-AD)	Address Output Hold Time (in relation to RD)		0		ns
t <sub>h</sub> (WR-AD)	Address Output Hold Time (in relation to WR)		(NOTE 2)		ns
t <sub>d</sub> (BCLK-CS)	Chip Select Output Delay Time			25	ns
t <sub>h</sub> (BCLK-CS)	Chip Select Output Hold Time (in relation to BCLK)		4		ns
t <sub>d</sub> (BCLK-ALE)	ALE Signal Output Delay Time			15	ns
t <sub>h</sub> (BCLK-ALE)	ALE Signal Output Hold Time		–4		ns
t <sub>d</sub> (BCLK-RD)	RD Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-RD)	RD Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-WR)	WR Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-WR)	WR Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-DB)	Data Output Delay Time (in relation to BCLK)			40	ns
t <sub>h</sub> (BCLK-DB)	Data Output Hold Time (in relation to BCLK) <sup>(3)</sup>		4		ns
t <sub>d</sub> (DB-WR)	Data Output Delay Time (in relation to WR)		(NOTE 1)		ns
t <sub>h</sub> (WR-DB)	Data Output Hold Time (in relation to WR) <sup>(3)</sup>		(NOTE 2)		ns
t <sub>d</sub> (BCLK-HLDA)	HLDA Output Delay Time			40	ns

**NOTES:**

1. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 40[\text{ns}] \quad f(\text{BCLK}) \text{ is } 12.5\text{MHz or less.}$$

2. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 10[\text{ns}]$$

3. This standard value shows the timing when the output is off, and does not show hold time of data bus.

Hold time of data bus varies with capacitor volume and pull-up (pull-down) resistance value.

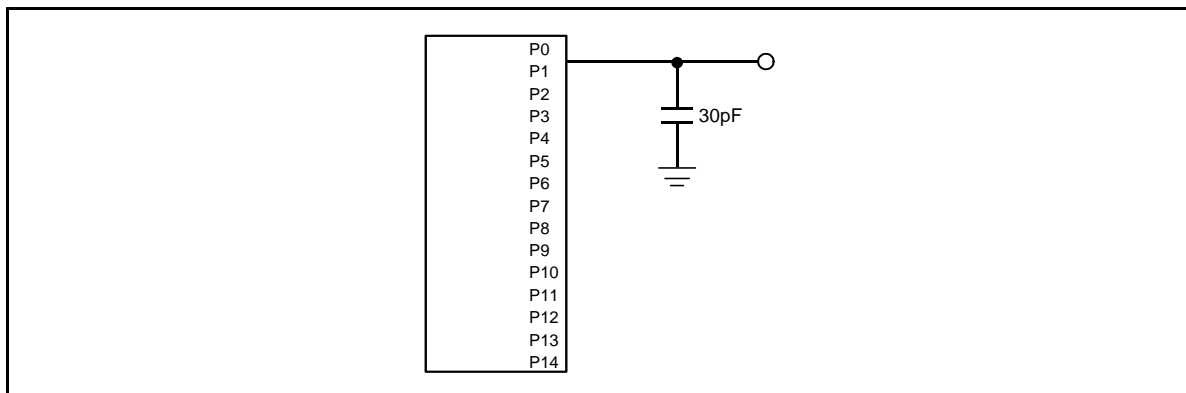
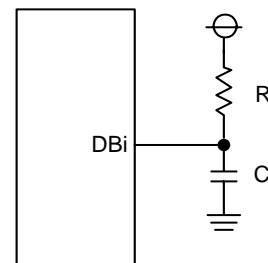
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC2})$$

by a circuit of the right figure.

For example, when V<sub>OL</sub> = 0.2V<sub>CC2</sub>, C = 30pF, R = 1kΩ, hold time of output "L" level is

$$t = -30\text{pF} \times 1\text{k}\Omega \times \ln(1 - 0.2V_{CC2} / V_{CC2}) \\ = 6.7\text{ns.}$$

**Figure 23.2 Ports P0 to P14 Measurement Circuit**

$$V_{CC1}=V_{CC2}=5V$$

**Switching Characteristics**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.28 Memory Expansion and Microprocessor Modes (for 1- to 3-wait setting and external area access)**

Symbol	Parameter		Standard		Unit
			Min.	Max.	
t <sub>d</sub> (BCLK-AD)	Address Output Delay Time	See Figure 23.2		25	ns
t <sub>h</sub> (BCLK-AD)	Address Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-AD)	Address Output Hold Time (in relation to RD)		0		ns
t <sub>h</sub> (WR-AD)	Address Output Hold Time (in relation to WR)		(NOTE 2)		ns
t <sub>d</sub> (BCLK-CS)	Chip Select Output Delay Time			25	ns
t <sub>h</sub> (BCLK-CS)	Chip Select Output Hold Time (in relation to BCLK)		4		ns
t <sub>d</sub> (BCLK-ALE)	ALE Signal Output Delay Time			15	ns
t <sub>h</sub> (BCLK-ALE)	ALE Signal Output Hold Time		-4		ns
t <sub>d</sub> (BCLK-RD)	RD Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-RD)	RD Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-WR)	WR Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-WR)	WR Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-DB)	Data Output Delay Time (in relation to BCLK)			40	ns
t <sub>h</sub> (BCLK-DB)	Data Output Hold Time (in relation to BCLK) <sup>(3)</sup>		4		ns
t <sub>d</sub> (DB-WR)	Data Output Delay Time (in relation to WR)		(NOTE 1)		ns
t <sub>h</sub> (WR-DB)	Data Output Hold Time (in relation to WR) <sup>(3)</sup>		(NOTE 2)		ns
t <sub>d</sub> (BCLK-HLDA)	HLDA Output Delay Time			40	ns

**NOTES:**

1. Calculated according to the BCLK frequency as follows:

$$\frac{(n-0.5) \times 10^9}{f(\text{BCLK})} - 40[\text{ns}]$$

n is "1" for 1-wait setting, "2" for 2-wait setting and "3" for 3-wait setting.  
(BCLK) is 12.5MHz or less.

2. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 10[\text{ns}]$$

3. This standard value shows the timing when the output is off, and does not show hold time of data bus.  
Hold time of data bus varies with capacitor volume and pull-up (pull-down) resistance value.

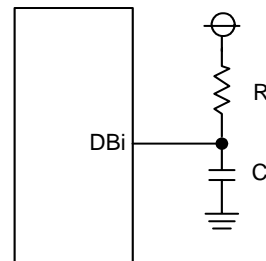
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC2})$$

by a circuit of the right figure.

For example, when V<sub>OL</sub> = 0.2V<sub>CC2</sub>, C = 30pF, R = 1kΩ, hold time of output "L" level is

$$t = -30\text{pF} \times 1\text{k}\Omega \times \ln(1 - 0.2V_{CC2} / V_{CC2}) = 6.7\text{ns}.$$



$$V_{CC1}=V_{CC2}=5V$$

**Switching Characteristics**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.29 Memory Expansion and Microprocessor Modes (for 2- to 3-wait setting, external area access and multiplex bus selection)**

Symbol	Parameter		Standard		Unit
			Min.	Max.	
t <sub>d</sub> (BCLK-AD)	Address Output Delay Time	See Figure 23.2		25	ns
t <sub>h</sub> (BCLK-AD)	Address Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-AD)	Address Output Hold Time (in relation to RD)		(NOTE 1)		ns
t <sub>h</sub> (WR-AD)	Address Output Hold Time (in relation to WR)		(NOTE 1)		ns
t <sub>d</sub> (BCLK-CS)	Chip Select Output Delay Time			25	ns
t <sub>h</sub> (BCLK-CS)	Chip Select Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-CS)	Chip Select Output Hold Time (in relation to RD)		(NOTE 1)		ns
t <sub>h</sub> (WR-CS)	Chip Select Output Hold Time (in relation to WR)		(NOTE 1)		ns
t <sub>d</sub> (BCLK-RD)	RD Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-RD)	RD Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-WR)	WR Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-WR)	WR Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-DB)	Data Output Delay Time (in relation to BCLK)			40	ns
t <sub>h</sub> (BCLK-DB)	Data Output Hold Time (in relation to BCLK)		4		ns
t <sub>d</sub> (DB-WR)	Data Output Delay Time (in relation to WR)		(NOTE 2)		ns
t <sub>h</sub> (WR-DB)	Data Output Hold Time (in relation to WR)		(NOTE 1)		ns
t <sub>d</sub> (BCLK-HLDA)	HLDA Output Delay Time			40	ns
t <sub>d</sub> (BCLK-ALE)	ALE Signal Output Delay Time (in relation to BCLK)			15	ns
t <sub>h</sub> (BCLK-ALE)	ALE Signal Output Hold Time (in relation to BCLK)		–4		ns
t <sub>d</sub> (AD-ALE)	ALE Signal Output Delay Time (in relation to Address)		(NOTE 3)		ns
t <sub>h</sub> (AD-ALE)	ALE Signal Output Hold Time (in relation to Address)		(NOTE 4)		ns
t <sub>d</sub> (AD-RD)	RD Signal Output Delay From the End of Address		0		ns
t <sub>d</sub> (AD-WR)	WR Signal Output Delay From the End of Address		0		ns
t <sub>dz</sub> (RD-AD)	Address Output Floating Start Time			8	ns

## NOTES:

1. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 10[\text{ns}]$$

2. Calculated according to the BCLK frequency as follows:

$$\frac{(n - 0.5) \times 10^9}{f(\text{BCLK})} - 40[\text{ns}] \quad n \text{ is "2" for 2-wait setting, "3" for 3-wait setting.}$$

3. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 25[\text{ns}]$$

4. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 15[\text{ns}]$$

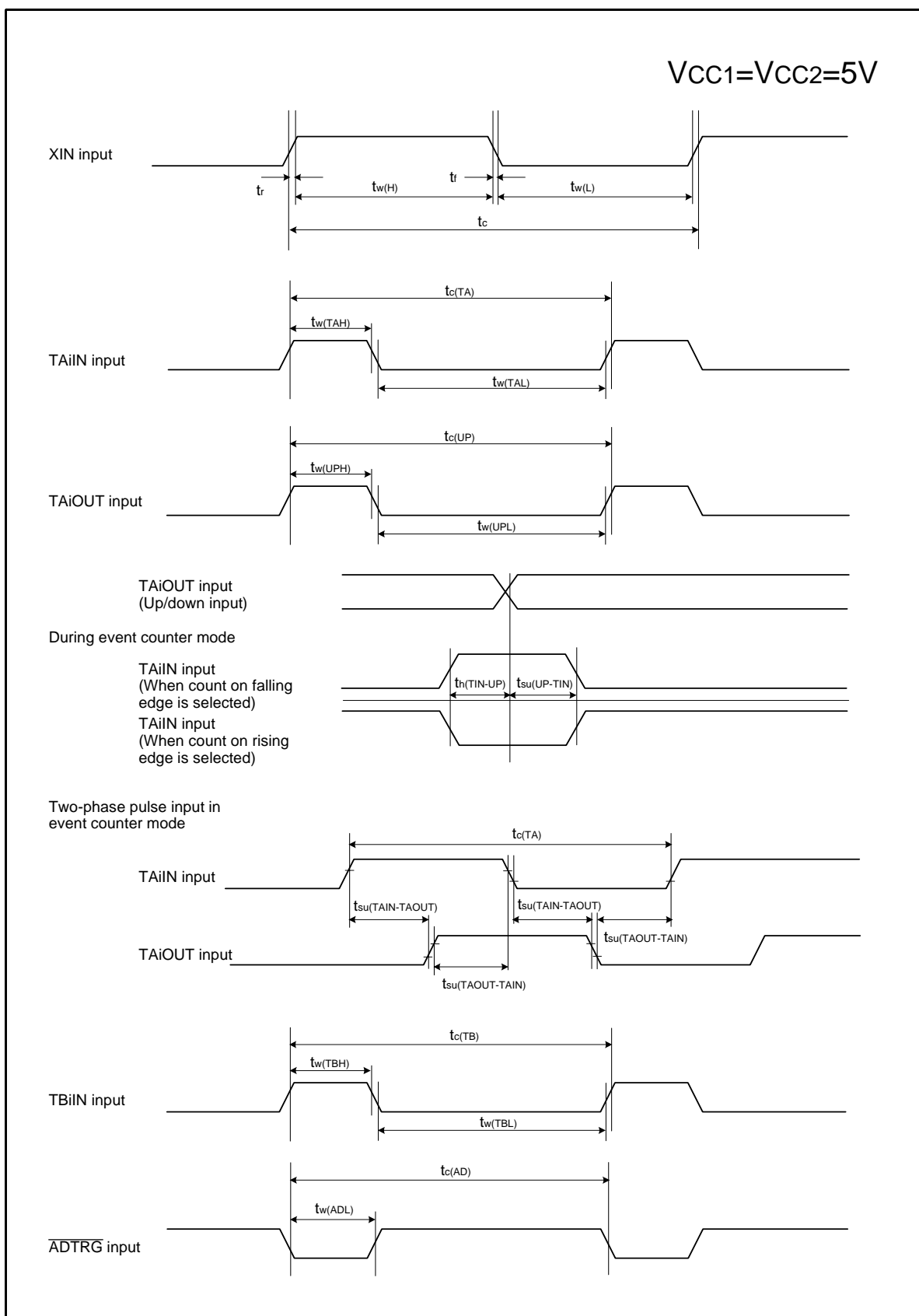


Figure 23.3 Timing Diagram (1)

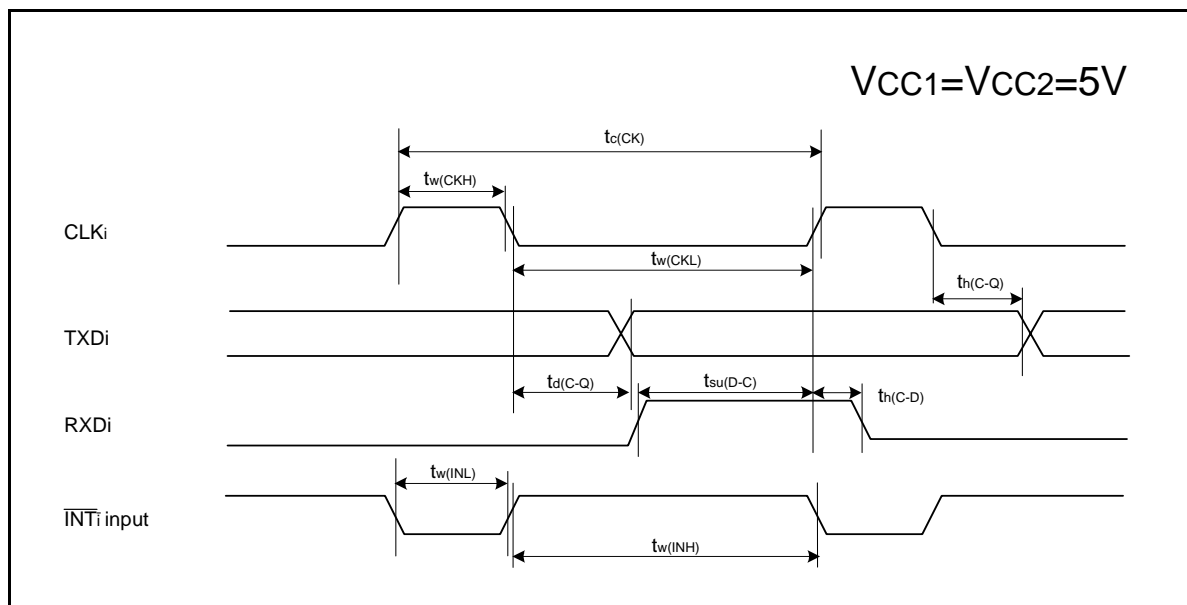


Figure 23.4 Timing Diagram (2)



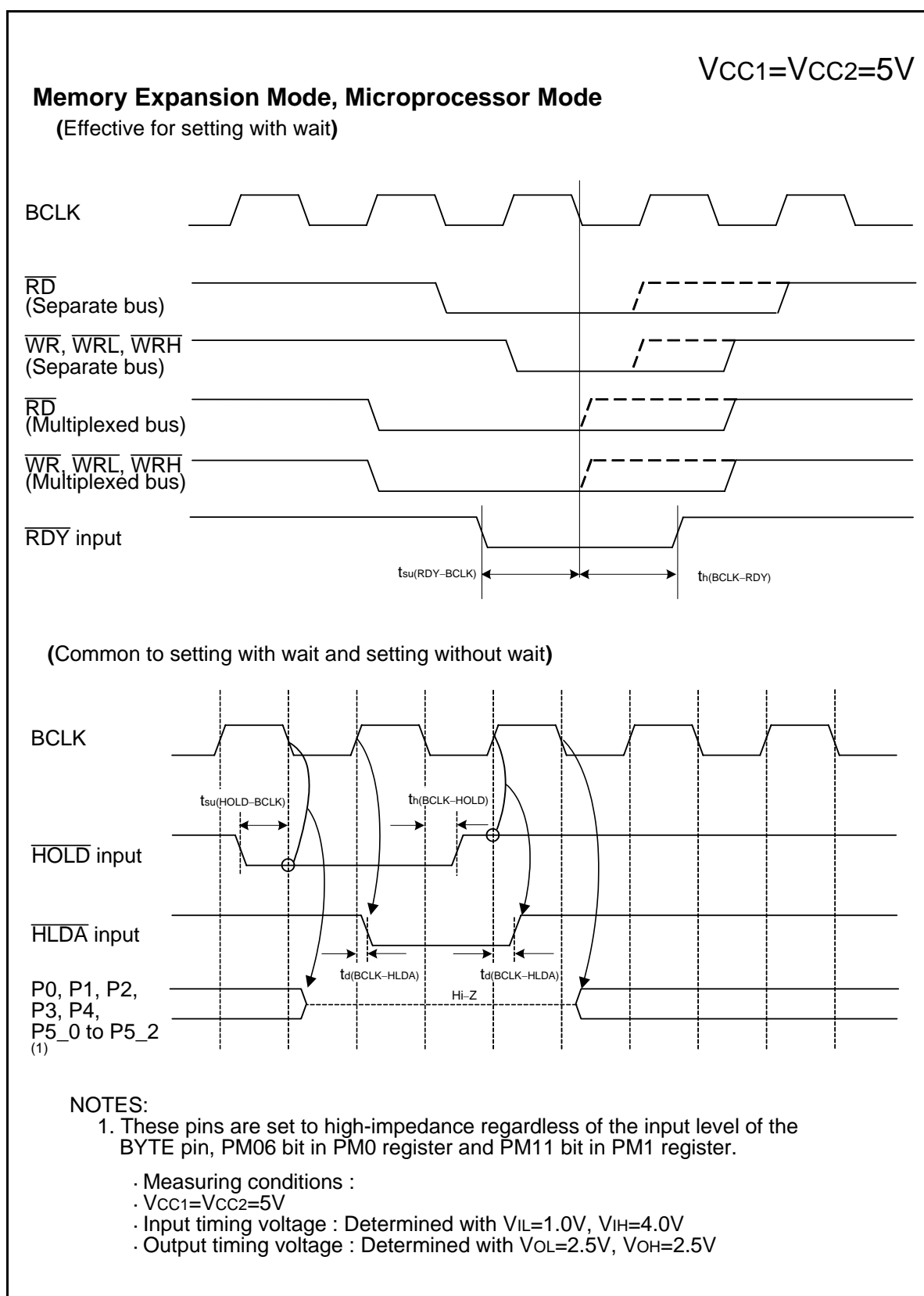


Figure 23.5 Timing Diagram (3)

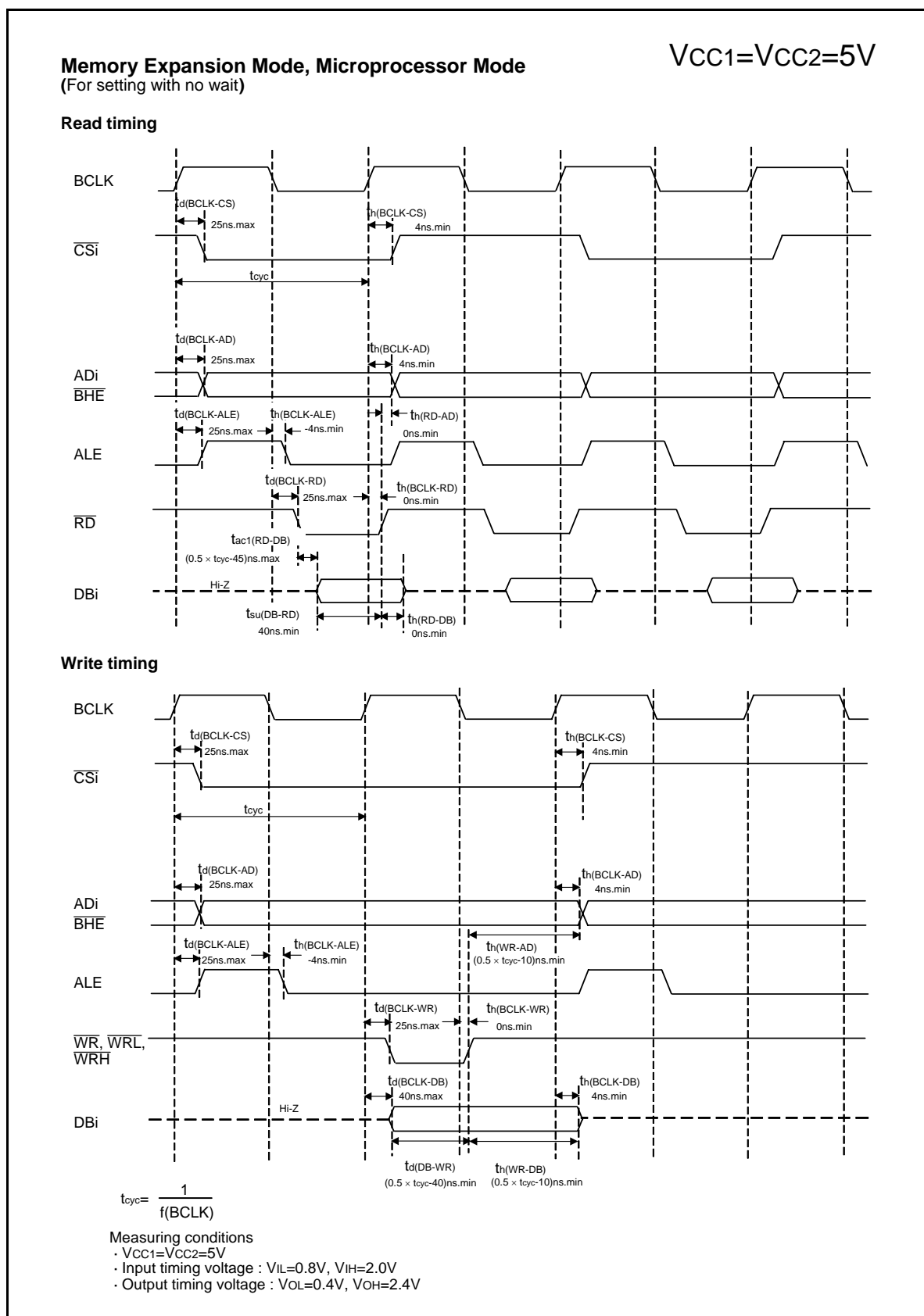


Figure 23.6 Timing Diagram (4)

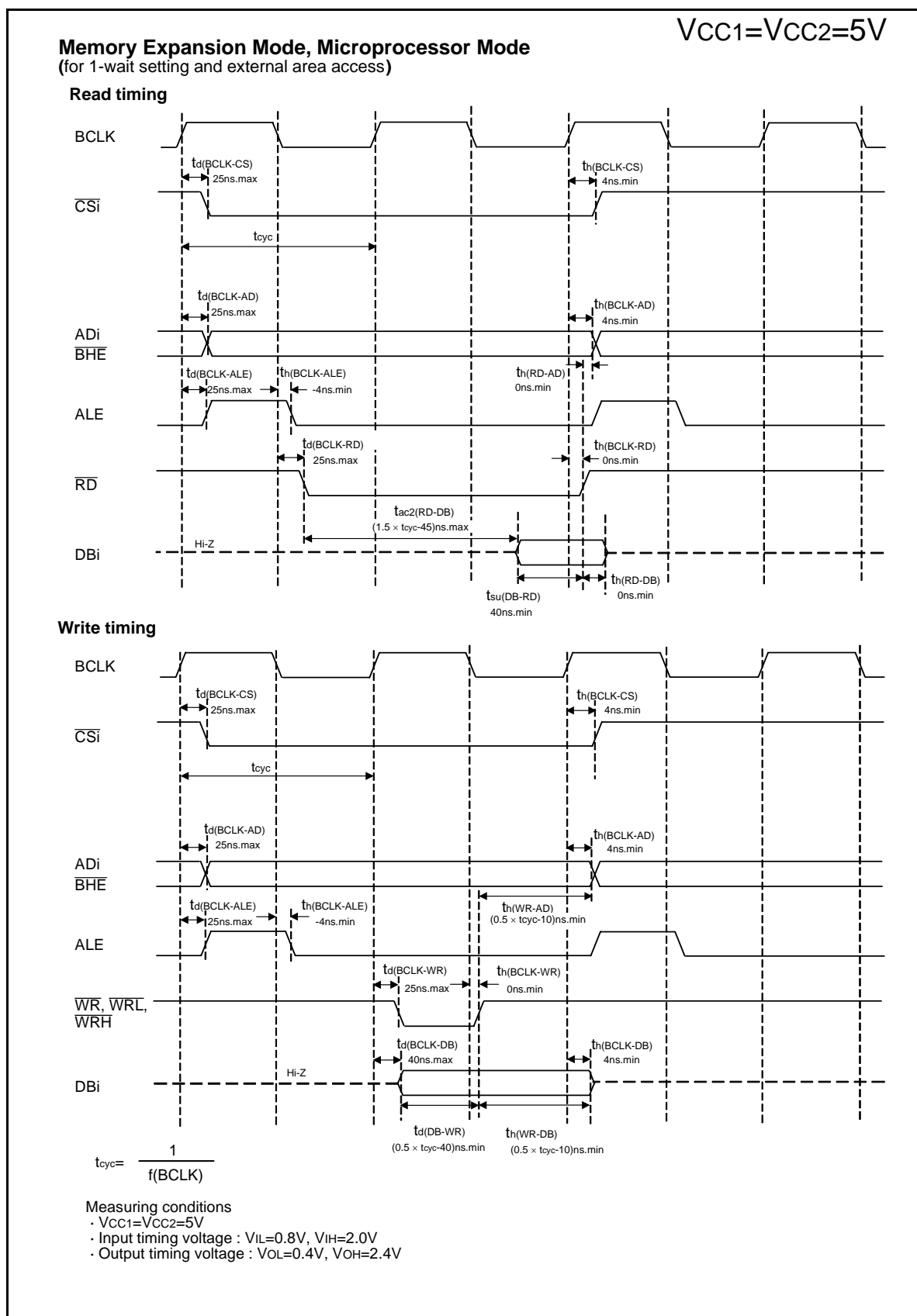


Figure 23.7 Timing Diagram (5)

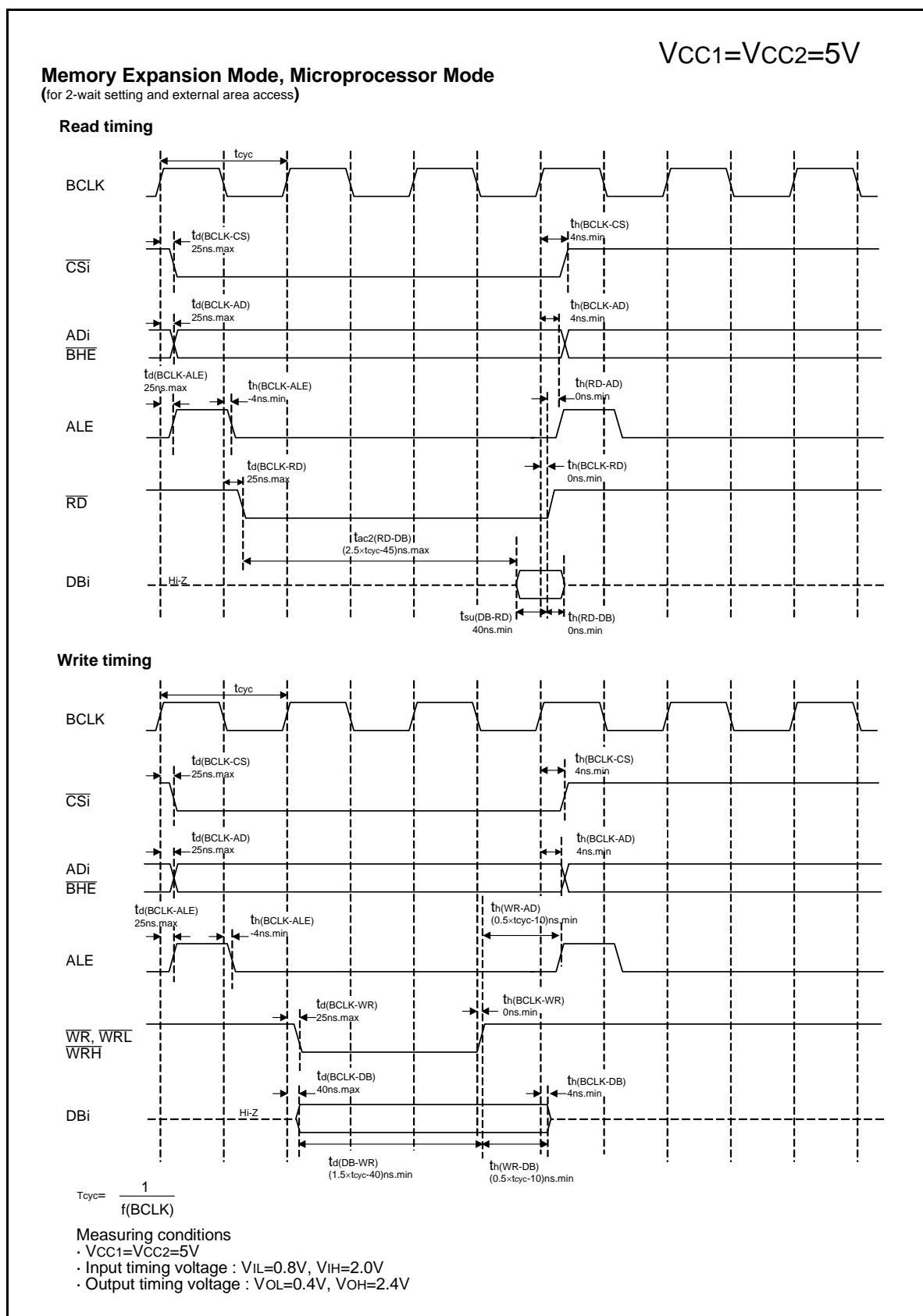


Figure 23.8 Timing Diagram (6)

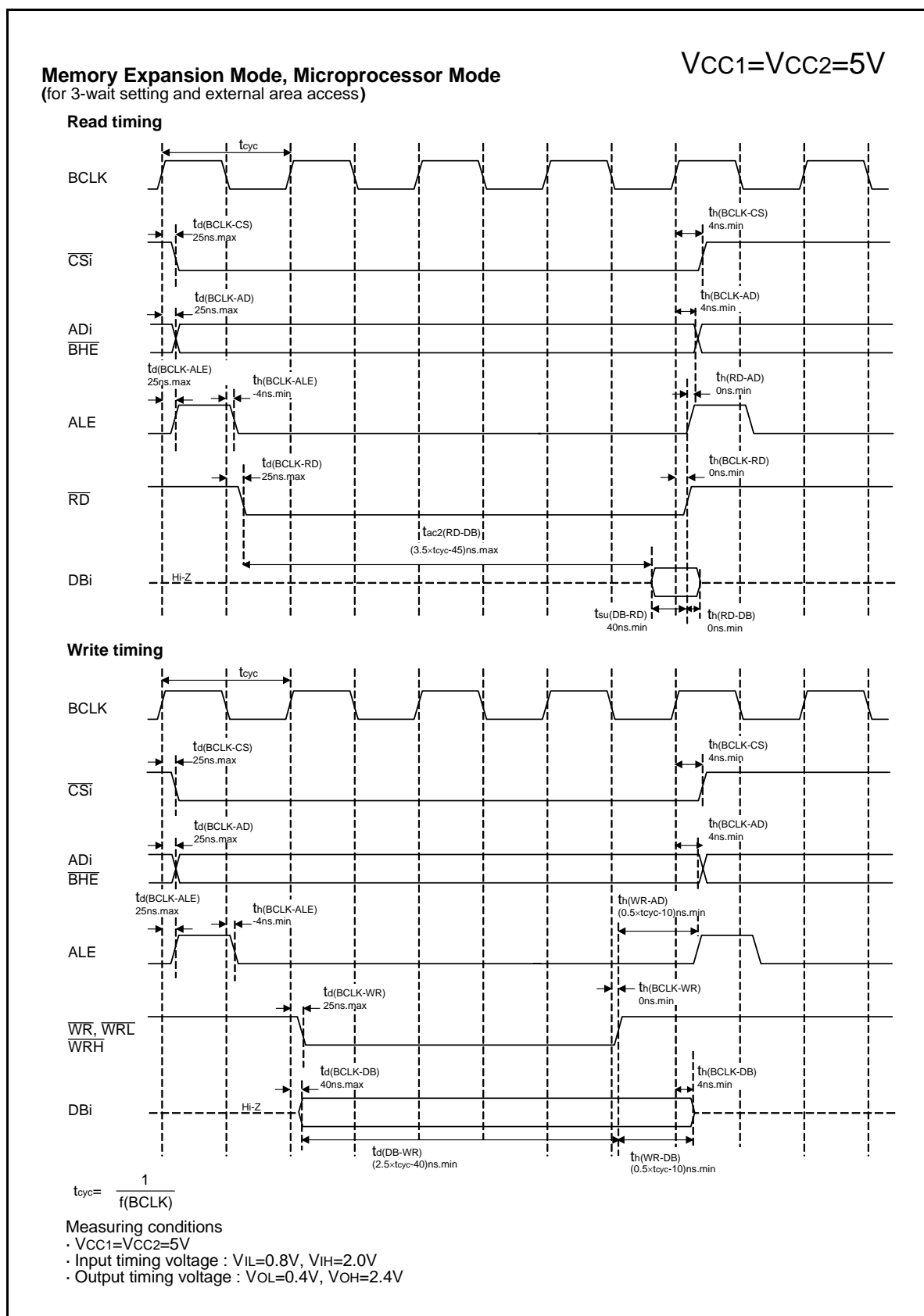


Figure 23.9 Timing Diagram (7)

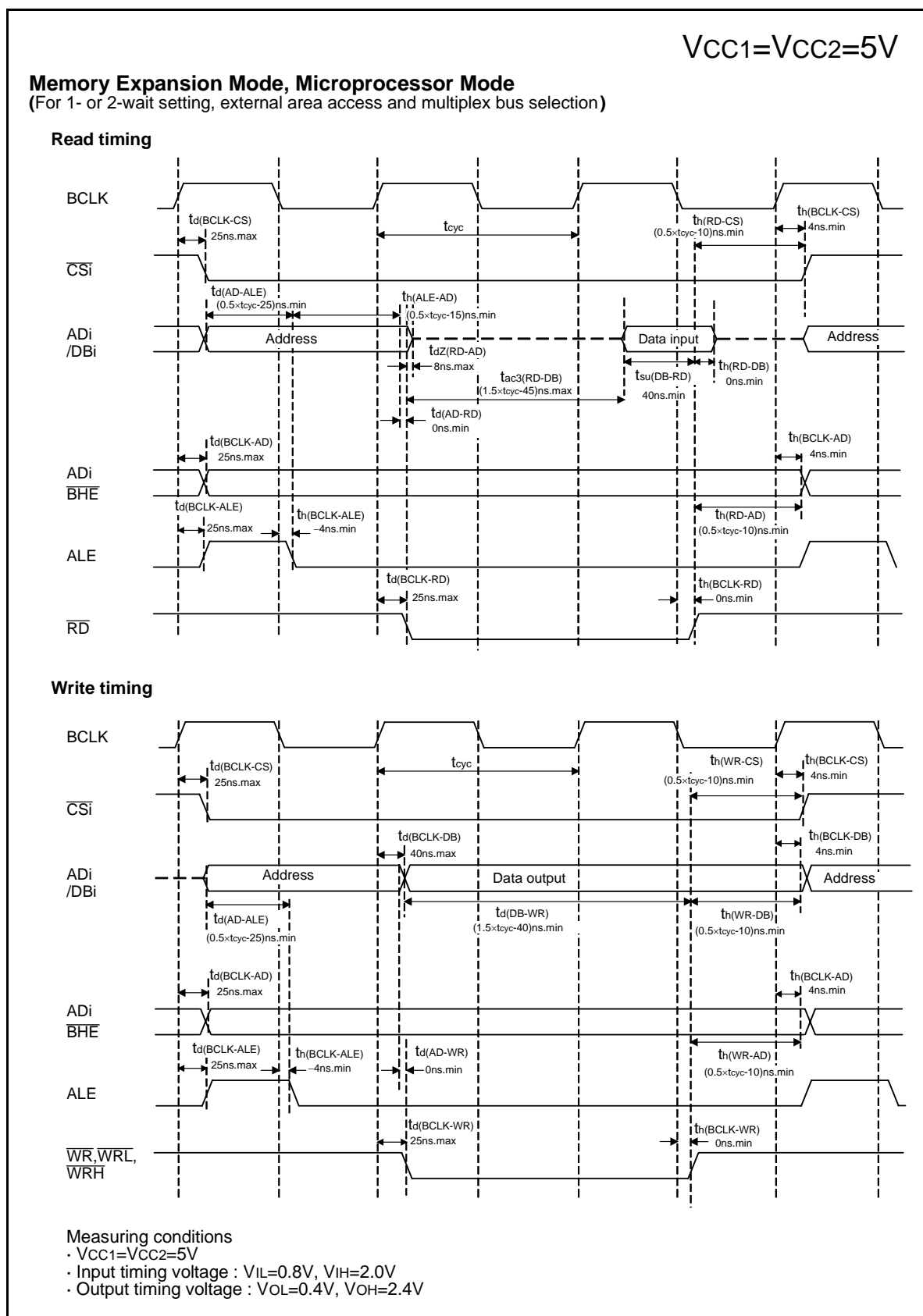


Figure 23.10 Timing Diagram (8)

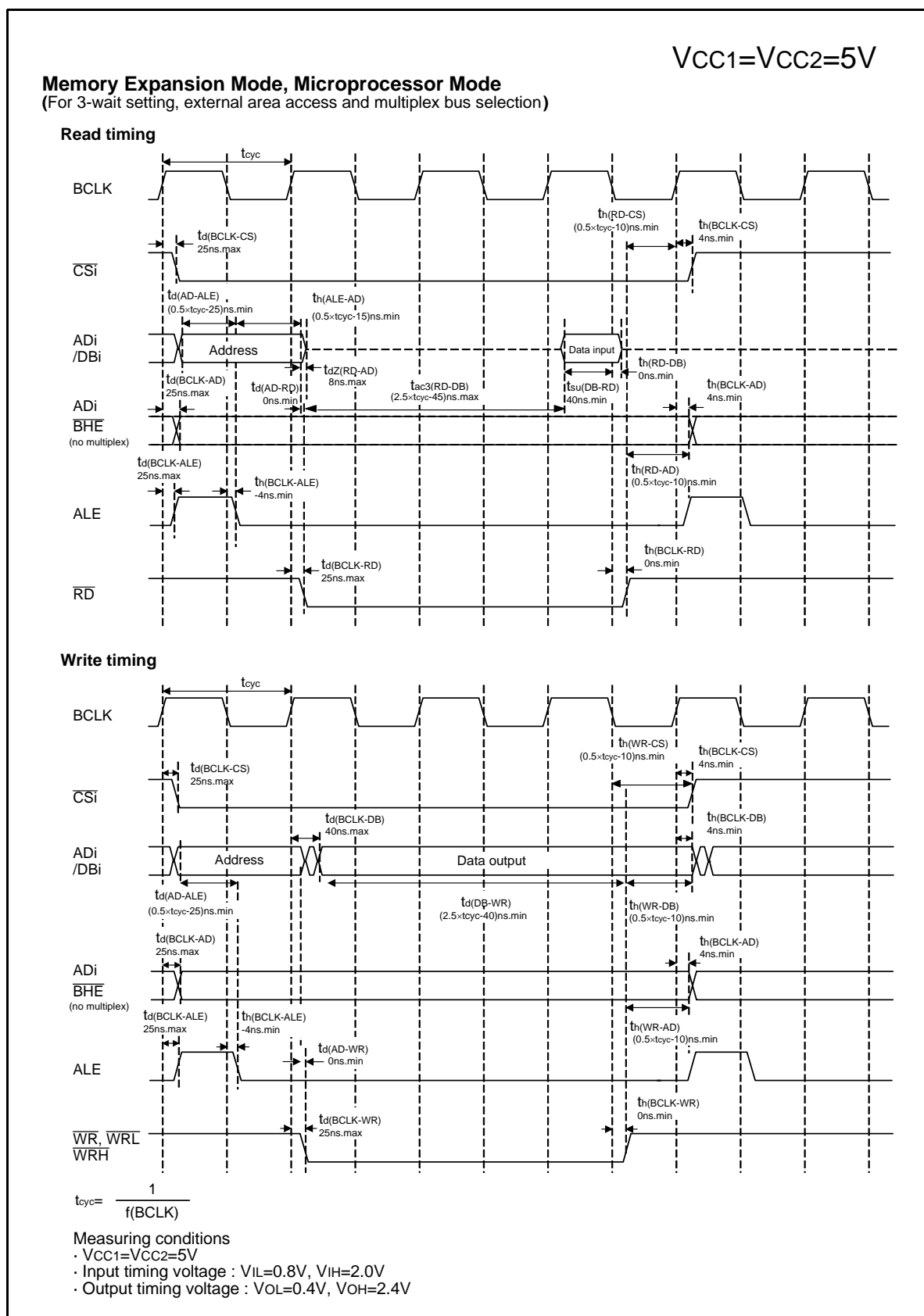


Figure 23.11 Timing Diagram (9)

$$V_{CC1}=V_{CC2}=3V$$

Table 23.30 Electrical Characteristics (1) <sup>(1)</sup>

Symbol	Parameter		Measuring Condition	Standard			Unit
				Min.	Typ.	Max.	
VOH	HIGH Output Voltage <sup>(3)</sup>	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, P11_0 to P11_7, P14_0, P14_1	IOH=−1mA	VCC1−0.5		VCC1	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOH=−1mA <sup>(2)</sup>	VCC2−0.5		VCC2	
VOH	HIGH Output Voltage XOUT	HIGHPOWER	IOH=−0.1mA	VCC1−0.5		VCC1	V
		LOWPOWER	IOH=−50μA	VCC1−0.5		VCC1	
	HIGH Output Voltage XCOUT	HIGHPOWER	With no load applied		2.5		V
		LOWPOWER	With no load applied		1.6		
VOL	LOW Output Voltage <sup>(3)</sup>	P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1	IOL=1mA			0.5	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOL=1mA <sup>(2)</sup>			0.5	
VOL	LOW Output Voltage XOUT	HIGHPOWER	IOL=0.1mA			0.5	V
		LOWPOWER	IOL=50μA			0.5	
	LOW Output Voltage XCOUT	HIGHPOWER	With no load applied		0		V
		LOWPOWER	With no load applied		0		
VT+·VT−	Hysteresis	HOLD, RDY, TA0IN to TA4IN, TB0IN to TB5IN, INT0 to INT5, NMI, ADTRG, CTS0 to CTS2, CLK0 to CLK4, TA0OUT to TA4OUT, KI0 to KI3, RXD0 to RXD2, SCL0 to SCL2, SDA0 to SDA2, SIN3, SIN4		0.2		0.8	V
VT+·VT−	Hysteresis	RESET		0.2	(0.7)	1.8	V
IIH	HIGH Input Current <sup>(3)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	VI=3V			4.0	μA
IIL	LOW Input Current <sup>(3)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	VI=0V			−4.0	μA
RPULLUP	Pull-Up Resistance <sup>(3)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1	VI=0V	50	100	500	kΩ
RfXIN	Feedback Resistance XIN				3.0		MΩ
RfXCIN	Feedback Resistance XCIN				25		MΩ
VRAM	RAM Retention Voltage		At stop mode	2.0			V

## NOTES:

1. Referenced to VCC1 = VCC2 = 2.7 to 3.3V, VSS = 0V at Topr = −20 to 85°C / −40 to 85°C, f(XIN)=10MHz no wait unless otherwise specified.
2. VCC1 for the port P6 to P11 and P14, and VCC2 for the port P0 to P5 and P12 to P13
3. There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.



**Table 23.31 Electrical Characteristics (2) <sup>(1)</sup>**

Symbol	Parameter		Measuring Condition		Standard			Unit
					Min.	Typ.	Max.	
Icc	Power Supply Current (Vcc1=Vcc2=2.7V to 3.6V)	In single-chip mode, the output pins are open and other pins are Vss	Mask ROM	f(BCLK)=10MHz No division		8	11	mA
				No division, On-chip oscillation		1		mA
			Flash Memory	f(BCLK)=10MHz, No division		8	13	mA
				No division, On-chip oscillation		1.8		mA
			Flash Memory Program	f(BCLK)=10MHz, VCC1=3.0V		12		mA
			Flash Memory Erase	f(BCLK)=10MHz, VCC1=3.0V		22		mA
			Mask ROM	f(XCIN)=32kHz Low power dissipation mode, ROM <sup>(3)</sup>		25		μA
			Flash Memory	f(BCLK)=32kHz Low power dissipation mode, RAM <sup>(3)</sup>		25		μA
				f(BCLK)=32kHz Low power dissipation mode, Flash Memory <sup>(3)</sup>		420		μA
				On-chip oscillation, Wait mode		45		μA
			Mask ROM Flash Memory	f(BCLK)=32kHz Wait mode <sup>(2)</sup> , Oscillation capability High		6.0		μA
				f(BCLK)=32kHz Wait mode <sup>(2)</sup> , Oscillation capability Low		1.8		μA
				Stop mode Topr =25°C		0.7	3.0	μA
Idet4	Low Voltage Detection Dissipation Current <sup>(4)</sup>					0.6	4	μA
Idet3	Reset Area Detection Dissipation Current <sup>(4)</sup>					0.4	2	μA

## NOTES:

1. Referenced to V<sub>CC1</sub>=V<sub>CC2</sub>=2.7 to 3.3V, V<sub>SS</sub> = 0V at T<sub>opr</sub> = -20 to 85°C / -40 to 85°C, f(BCLK)=10MHz unless otherwise specified.
2. With one timer operated using fC32.
3. This indicates the memory in which the program to be executed exists.
4. I<sub>det</sub> is dissipation current when the following bit is set to "1" (detection circuit enabled).  
I<sub>det4</sub>: VC27 bit in the VCR2 register  
I<sub>det3</sub>: VC26 bit in the VCR2 register

$$V_{CC1}=V_{CC2}=3V$$

**Timing Requirements**(V<sub>CC1</sub> = V<sub>CC2</sub> = 3V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.32 External Clock Input (XIN input)<sup>(1)</sup>**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub>	External Clock Input Cycle Time	(NOTE 2)		ns
t <sub>w(H)</sub>	External Clock Input HIGH Pulse Width	(NOTE 3)		ns
t <sub>w(L)</sub>	External Clock Input LOW Pulse Width	(NOTE 3)		ns
t <sub>r</sub>	External Clock Rise Time		(NOTE 4)	ns
t <sub>f</sub>	External Clock Fall Time		(NOTE 4)	ns

**NOTES:**

1. The condition is V<sub>CC1</sub>=V<sub>CC2</sub>=2.7 to 3.0V.
2. Calculated according to the V<sub>CC1</sub> voltage as follows:

$$\frac{10^{-6}}{20 \times V_{CC2} - 44} \text{ [ns]}$$

3. Calculated according to the V<sub>CC1</sub> voltage as follows:

$$\frac{10^{-6}}{20 \times V_{CC1} - 44} \times 0.4 \text{ [ns]}$$

4. Calculated according to the V<sub>CC1</sub> voltage as follows:

$$-10 \times V_{CC1} + 45 \text{ [ns]}$$

**Table 23.33 Memory Expansion Mode and Microprocessor Mode**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>ac1</sub> (RD-DB)	Data Input Access Time (for setting with no wait)		(NOTE 1)	ns
t <sub>ac2</sub> (RD-DB)	Data Input Access Time (for setting with wait)		(NOTE 2)	ns
t <sub>ac3</sub> (RD-DB)	Data Input Access Time (when accessing multiplex bus area)		(NOTE 3)	ns
t <sub>su</sub> (DB-RD)	Data Input Setup Time	50		ns
t <sub>su</sub> (RDY-BCLK)	RDY Input Setup Time	40		ns
t <sub>su</sub> (HOLD-BCLK)	HOLD Input Setup Time	50		ns
t <sub>h</sub> (RD-DB)	Data Input Hold Time	0		ns
t <sub>h</sub> (BCLK-RDY)	RDY Input Hold Time	0		ns
t <sub>h</sub> (BCLK-HOLD)	HOLD Input Hold Time	0		ns

**NOTES:**

1. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 60 \text{ [ns]}$$

2. Calculated according to the BCLK frequency as follows:

$$\frac{(n-0.5) \times 10^9}{f(\text{BCLK})} - 60 \text{ [ns]} \quad n \text{ is "2" for 1-wait setting, "3" for 2-wait setting and "4" for 3-wait setting.}$$

3. Calculated according to the BCLK frequency as follows:

$$\frac{(n-0.5) \times 10^9}{f(\text{BCLK})} - 60 \text{ [ns]} \quad n \text{ is "2" for 2-wait setting, "3" for 3-wait setting.}$$

$$V_{CC1}=V_{CC2}=3V$$

**Timing Requirements**(V<sub>CC1</sub> = V<sub>CC2</sub> = 3V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.34 Timer A Input (Counter Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	150		ns
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	60		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	60		ns

**Table 23.35 Timer A Input (Gating Input in Timer Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	600		ns
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	300		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	300		ns

**Table 23.36 Timer A Input (External Trigger Input in One-shot Timer Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	300		ns
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	150		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	150		ns

**Table 23.37 Timer A Input (External Trigger Input in Pulse Width Modulation Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>w</sub> (TAH)	TAiIN Input HIGH Pulse Width	150		ns
t <sub>w</sub> (TAL)	TAiIN Input LOW Pulse Width	150		ns

**Table 23.38 Timer A Input (Counter Increment/Decrement Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (UP)	TAiOUT Input Cycle Time	3000		ns
t <sub>w</sub> (UPH)	TAiOUT Input HIGH Pulse Width	1500		ns
t <sub>w</sub> (UPL)	TAiOUT Input LOW Pulse Width	1500		ns
t <sub>su</sub> (UP-TIN)	TAiOUT Input Setup Time	600		ns
t <sub>h</sub> (TIN-UP)	TAiOUT Input Hold Time	600		ns

**Table 23.39 Timer A Input (Two-phase Pulse Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TA)	TAiIN Input Cycle Time	2		μs
t <sub>su</sub> (TAIN-TAOUT)	TAiOUT Input Setup Time	500		ns
t <sub>su</sub> (TAOUT-TAIN)	TAiIN Input Setup Time	500		ns

$$V_{CC1}=V_{CC2}=3V$$

**Timing Requirements**(V<sub>CC1</sub> = V<sub>CC2</sub> = 3V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.40 Timer B Input (Counter Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TB)	TBiIN Input Cycle Time (counted on one edge)	150		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width (counted on one edge)	60		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width (counted on one edge)	60		ns
t <sub>c</sub> (TB)	TBiIN Input Cycle Time (counted on both edges)	300		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width (counted on both edges)	120		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width (counted on both edges)	120		ns

**Table 23.41 Timer B Input (Pulse Period Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TB)	TBiIN Input Cycle Time	600		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width	300		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width	300		ns

**Table 23.42 Timer B Input (Pulse Width Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (TB)	TBiIN Input Cycle Time	600		ns
t <sub>w</sub> (TBH)	TBiIN Input HIGH Pulse Width	300		ns
t <sub>w</sub> (TBL)	TBiIN Input LOW Pulse Width	300		ns

**Table 23.43 A/D Trigger Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (AD)	ADTRG Input Cycle Time	1500		ns
t <sub>w</sub> (ADL)	ADTRG Input LOW Pulse Width	200		ns

**Table 23.44 Serial Interface**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c</sub> (CK)	CLKi Input Cycle Time	300		ns
t <sub>w</sub> (CKH)	CLKi Input HIGH Pulse Width	150		ns
t <sub>w</sub> (CKL)	CLKi Input LOW Pulse Width	150		ns
t <sub>d</sub> (C-Q)	TXDi Output Delay Time		160	ns
t <sub>h</sub> (C-Q)	TXDi Hold Time	0		ns
t <sub>su</sub> (D-C)	RXDi Input Setup Time	100		ns
t <sub>h</sub> (C-D)	RXDi Input Hold Time	90		ns

**Table 23.45 External Interrupt INTi Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>w</sub> (INH)	INTi Input HIGH Pulse Width	380		ns
t <sub>w</sub> (INL)	INTi Input LOW Pulse Width	380		ns

$$V_{CC1}=V_{CC2}=3V$$

**Switching Characteristics**(V<sub>CC1</sub> = V<sub>CC2</sub> = 3V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.46 Memory Expansion and Microprocessor Modes (for setting with no wait)**

Symbol	Parameter		Standard		Unit
			Min.	Max.	
t <sub>d</sub> (BCLK-AD)	Address Output Delay Time	See Figure 23.12		30	ns
t <sub>h</sub> (BCLK-AD)	Address Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-AD)	Address Output Hold Time (in relation to RD)		0		ns
t <sub>h</sub> (WR-AD)	Address Output Hold Time (in relation to WR)		(NOTE 2)		ns
t <sub>d</sub> (BCLK-CS)	Chip Select Output Delay Time			30	ns
t <sub>h</sub> (BCLK-CS)	Chip Select Output Hold Time (in relation to BCLK)		4		ns
t <sub>d</sub> (BCLK-ALE)	ALE Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-ALE)	ALE Signal Output Hold Time		–4		ns
t <sub>d</sub> (BCLK-RD)	RD Signal Output Delay Time			30	ns
t <sub>h</sub> (BCLK-RD)	RD Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-WR)	WR Signal Output Delay Time			30	ns
t <sub>h</sub> (BCLK-WR)	WR Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-DB)	Data Output Delay Time (in relation to BCLK)			40	ns
t <sub>h</sub> (BCLK-DB)	Data Output Hold Time (in relation to BCLK) <sup>(3)</sup>		4		ns
t <sub>d</sub> (DB-WR)	Data Output Delay Time (in relation to WR)		(NOTE 1)		ns
t <sub>h</sub> (WR-DB)	Data Output Hold Time (in relation to WR) <sup>(3)</sup>		(NOTE 2)		ns
t <sub>d</sub> (BCLK-HLDA)	HLDA Output Delay Time			40	ns

**NOTES:**

1. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 40[\text{ns}] \quad f(\text{BCLK}) \text{ is } 12.5\text{MHz or less.}$$

2. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 10[\text{ns}]$$

3. This standard value shows the timing when the output is off, and does not show hold time of data bus.

Hold time of data bus varies with capacitor volume and pull-up (pull-down) resistance value.

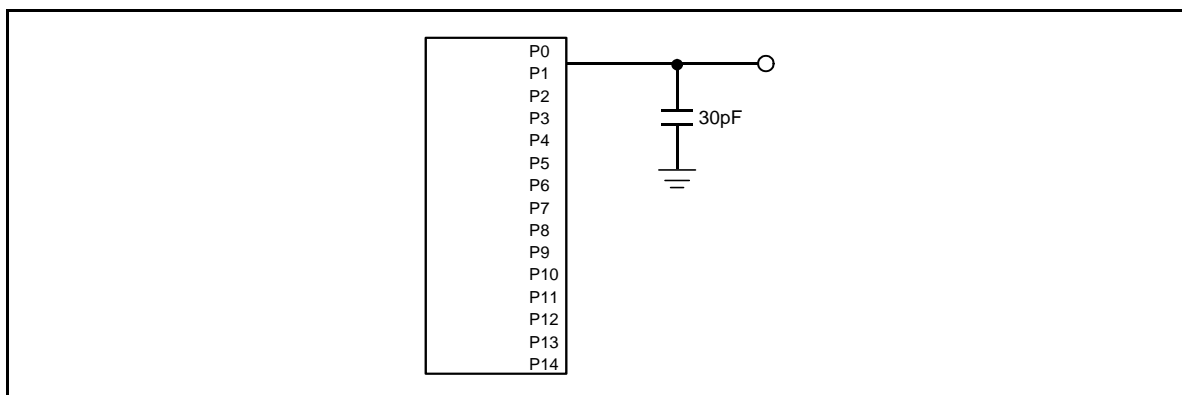
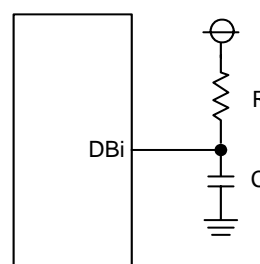
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC2})$$

by a circuit of the right figure.

For example, when V<sub>OL</sub> = 0.2V<sub>CC2</sub>, C = 30pF, R = 1kΩ, hold time of output "L" level is

$$t = -30\text{pF} \times 1\text{k}\Omega \times \ln(1 - 0.2V_{CC2} / V_{CC2}) \\ = 6.7\text{ns.}$$

**Figure 23.12 Ports P0 to P14 Measurement Circuit**

$$V_{CC1}=V_{CC2}=3V$$

**Switching Characteristics**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = -20 to 85°C / -40 to 85°C unless otherwise specified)**Table 23.47 Memory Expansion and Microprocessor Modes (for 1- to 3-wait setting and external area access)**

Symbol	Parameter		Standard		Unit
			Min.	Max.	
t <sub>d</sub> (BCLK-AD)	Address Output Delay Time	See Figure 23.12		30	ns
t <sub>h</sub> (BCLK-AD)	Address Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-AD)	Address Output Hold Time (in relation to RD)		0		ns
t <sub>h</sub> (WR-AD)	Address Output Hold Time (in relation to WR)		(NOTE 2)		ns
t <sub>d</sub> (BCLK-CS)	Chip Select Output Delay Time			30	ns
t <sub>h</sub> (BCLK-CS)	Chip Select Output Hold Time (in relation to BCLK)		4		ns
t <sub>d</sub> (BCLK-ALE)	ALE Signal Output Delay Time			25	ns
t <sub>h</sub> (BCLK-ALE)	ALE Signal Output Hold Time		-4		ns
t <sub>d</sub> (BCLK-RD)	RD Signal Output Delay Time			30	ns
t <sub>h</sub> (BCLK-RD)	RD Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-WR)	WR Signal Output Delay Time			30	ns
t <sub>h</sub> (BCLK-WR)	WR Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-DB)	Data Output Delay Time (in relation to BCLK)			40	ns
t <sub>h</sub> (BCLK-DB)	Data Output Hold Time (in relation to BCLK) <sup>(3)</sup>		4		ns
t <sub>d</sub> (DB-WR)	Data Output Delay Time (in relation to WR)		(NOTE 1)		ns
t <sub>h</sub> (WR-DB)	Data Output Hold Time (in relation to WR) <sup>(3)</sup>		(NOTE 2)		ns
t <sub>d</sub> (BCLK-HLDA)	HLDA Output Delay Time			40	ns

**NOTES:**

1. Calculated according to the BCLK frequency as follows:

$$\frac{(n-0.5) \times 10^9}{f(\text{BCLK})} - 40[\text{ns}]$$

n is "1" for 1-wait setting, "2" for 2-wait setting and "3" for 3-wait setting.  
(BCLK) is 12.5MHz or less.

2. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 10[\text{ns}]$$

3. This standard value shows the timing when the output is off, and does not show hold time of data bus.

Hold time of data bus varies with capacitor volume and pull-up (pull-down) resistance value.

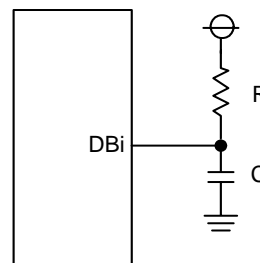
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC2})$$

by a circuit of the right figure.

For example, when V<sub>OL</sub> = 0.2V<sub>CC2</sub>, C = 30pF, R = 1kΩ, hold time of output "L" level is

$$t = -30\text{pF} \times 1\text{k}\Omega \times \ln(1 - 0.2V_{CC2} / V_{CC2}) = 6.7\text{ns}.$$



$$V_{CC1}=V_{CC2}=3V$$

**Switching Characteristics**(V<sub>CC1</sub> = V<sub>CC2</sub> = 5V, V<sub>SS</sub> = 0V, at T<sub>opr</sub> = –20 to 85°C / –40 to 85°C unless otherwise specified)**Table 23.48 Memory Expansion and Microprocessor Modes (for 2- to 3-wait setting, external area access and multiplex bus selection)**

Symbol	Parameter		Standard		Unit
			Min.	Max.	
t <sub>d</sub> (BCLK-AD)	Address Output Delay Time	See Figure 23.12		50	ns
t <sub>h</sub> (BCLK-AD)	Address Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-AD)	Address Output Hold Time (in relation to RD)		(NOTE 1)		ns
t <sub>h</sub> (WR-AD)	Address Output Hold Time (in relation to WR)		(NOTE 1)		ns
t <sub>d</sub> (BCLK-CS)	Chip Select Output Delay Time			50	ns
t <sub>h</sub> (BCLK-CS)	Chip Select Output Hold Time (in relation to BCLK)		4		ns
t <sub>h</sub> (RD-CS)	Chip Select Output Hold Time (in relation to RD)		(NOTE 1)		ns
t <sub>h</sub> (WR-CS)	Chip Select Output Hold Time (in relation to WR)		(NOTE 1)		ns
t <sub>d</sub> (BCLK-RD)	RD Signal Output Delay Time			40	ns
t <sub>h</sub> (BCLK-RD)	RD Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-WR)	WR Signal Output Delay Time			40	ns
t <sub>h</sub> (BCLK-WR)	WR Signal Output Hold Time		0		ns
t <sub>d</sub> (BCLK-DB)	Data Output Delay Time (in relation to BCLK)			50	ns
t <sub>h</sub> (BCLK-DB)	Data Output Hold Time (in relation to BCLK)		4		ns
t <sub>d</sub> (DB-WR)	Data Output Delay Time (in relation to WR)		(NOTE 2)		ns
t <sub>h</sub> (WR-DB)	Data Output Hold Time (in relation to WR)		(NOTE 1)		ns
t <sub>d</sub> (BCLK-HLDA)	HLDA Output Delay Time			40	ns
t <sub>d</sub> (BCLK-ALE)	ALE Signal Output Delay Time (in relation to BCLK)			25	ns
t <sub>h</sub> (BCLK-ALE)	ALE Signal Output Hold Time (in relation to BCLK)		–4		ns
t <sub>d</sub> (AD-ALE)	ALE Signal Output Delay Time (in relation to Address)		(NOTE 3)		ns
t <sub>h</sub> (AD-ALE)	ALE Signal Output Hold Time (in relation to Address)		(NOTE 4)		ns
t <sub>d</sub> (AD-RD)	RD Signal Output Delay From the End of Address		0		ns
t <sub>d</sub> (AD-WR)	WR Signal Output Delay From the End of Address		0		ns
t <sub>dz</sub> (RD-AD)	Address Output Floating Start Time			8	ns

## NOTES:

1. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 10[\text{ns}]$$

2. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 50[\text{ns}] \quad n \text{ is "2" for 2-wait setting, "3" for 3-wait setting.}$$

3. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 40[\text{ns}]$$

4. Calculated according to the BCLK frequency as follows:

$$\frac{0.5 \times 10^9}{f(\text{BCLK})} - 15[\text{ns}]$$

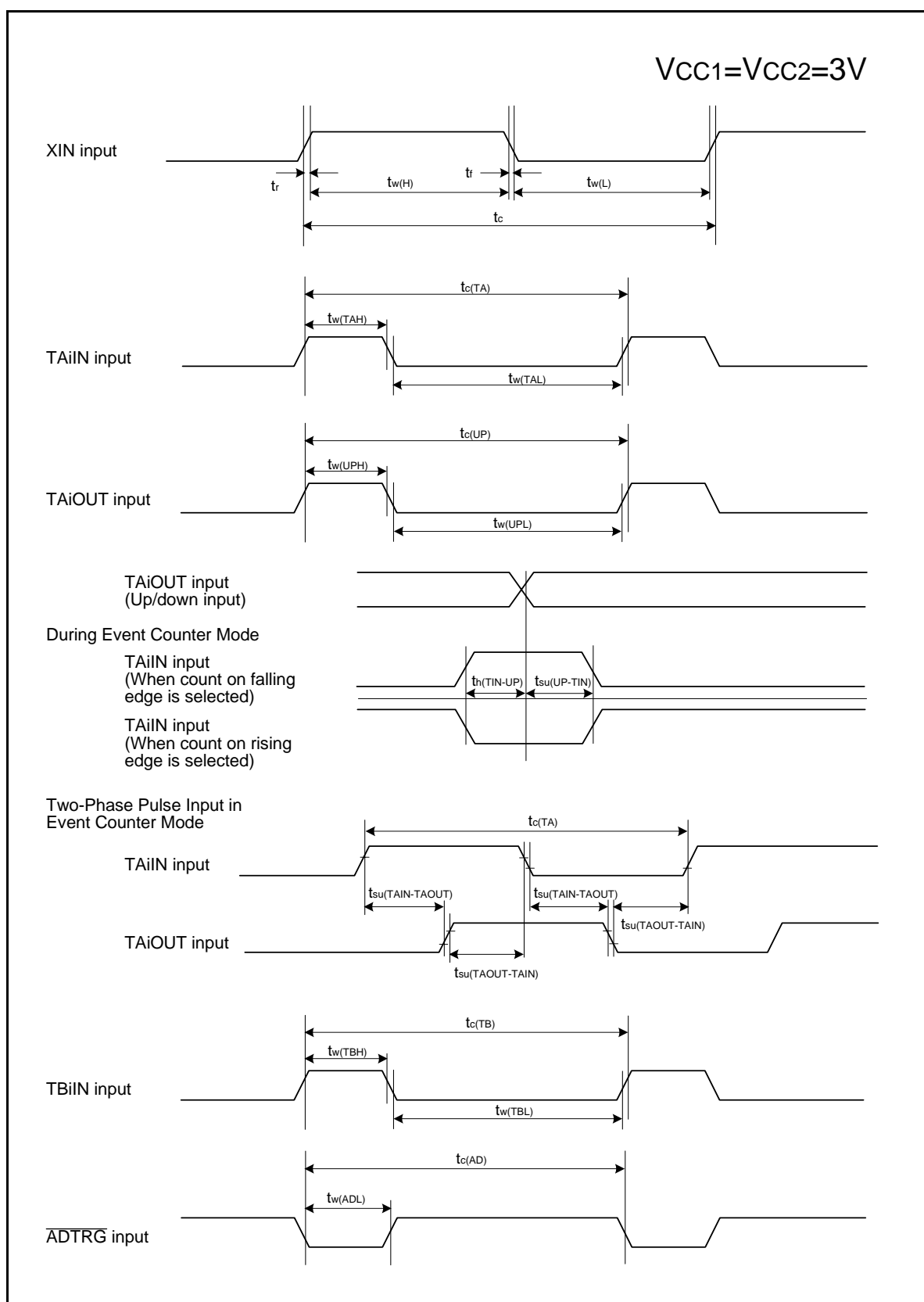


Figure 23.13 Timing Diagram (1)



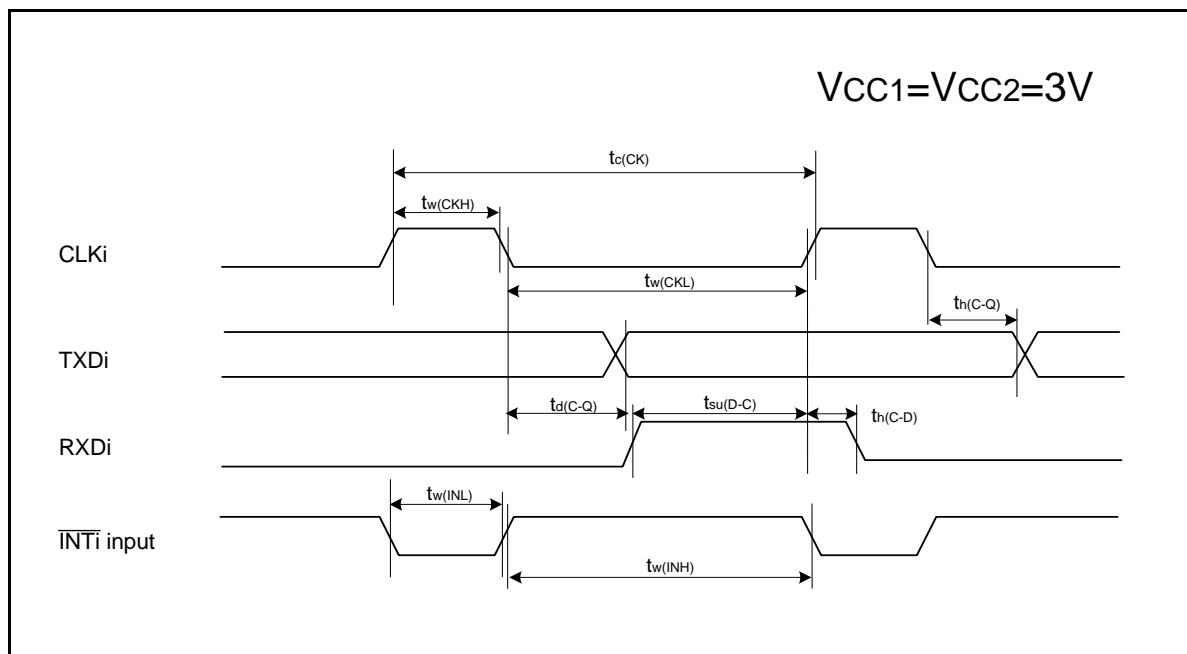


Figure 23.14 Timing Diagram (2)

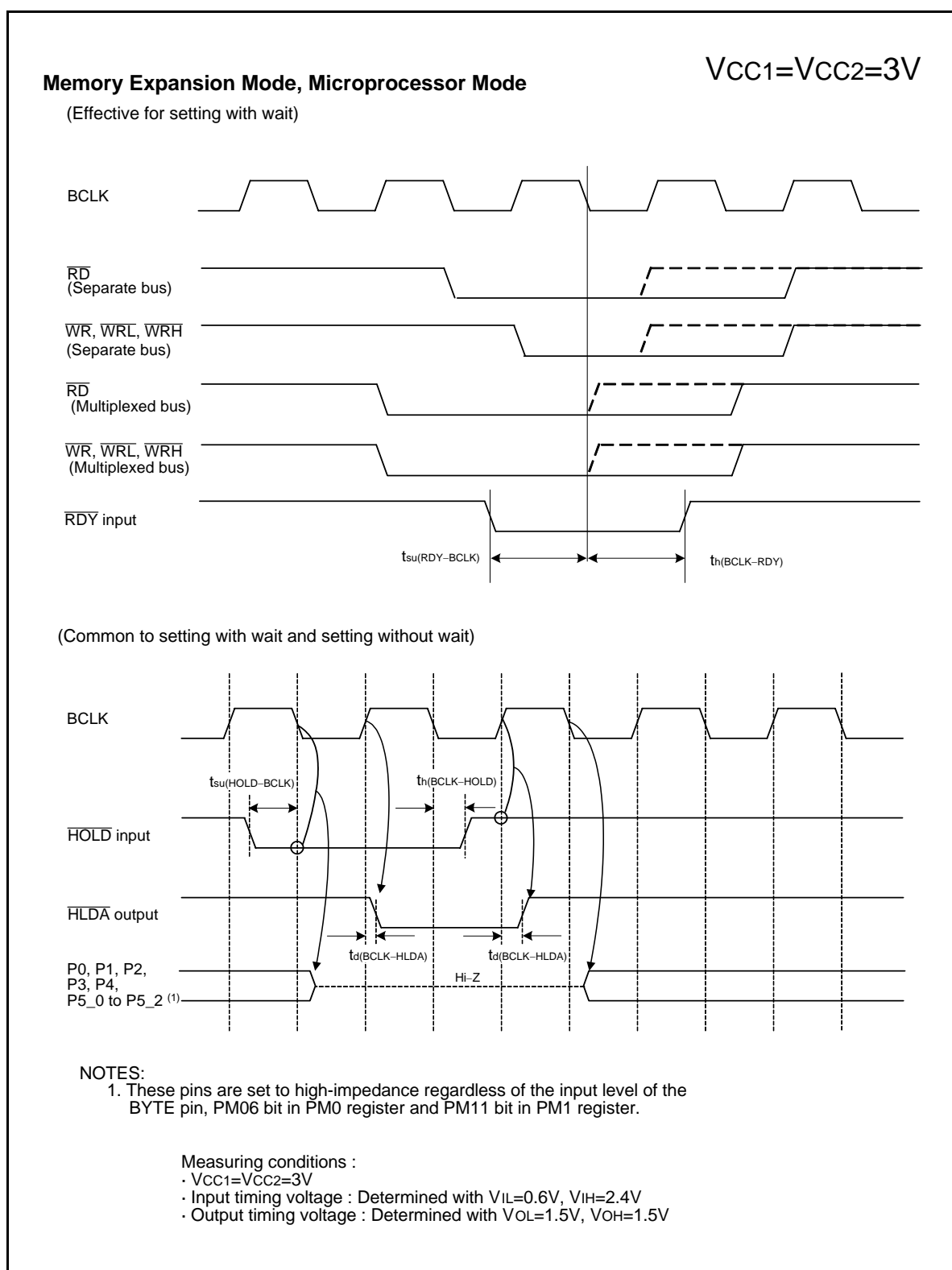


Figure 23.15 Timing Diagram (3)

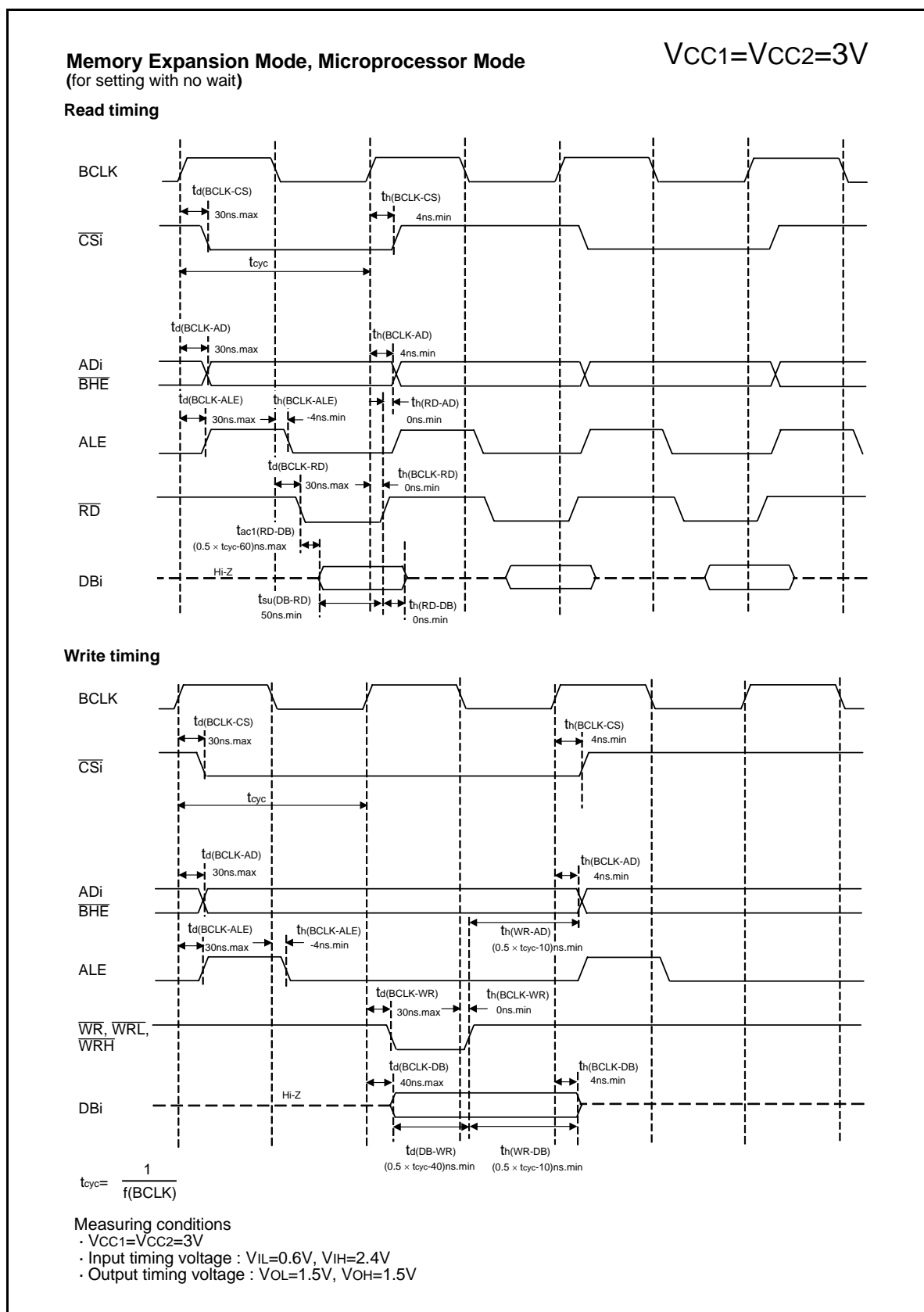


Figure 23.16 Timing Diagram (4)

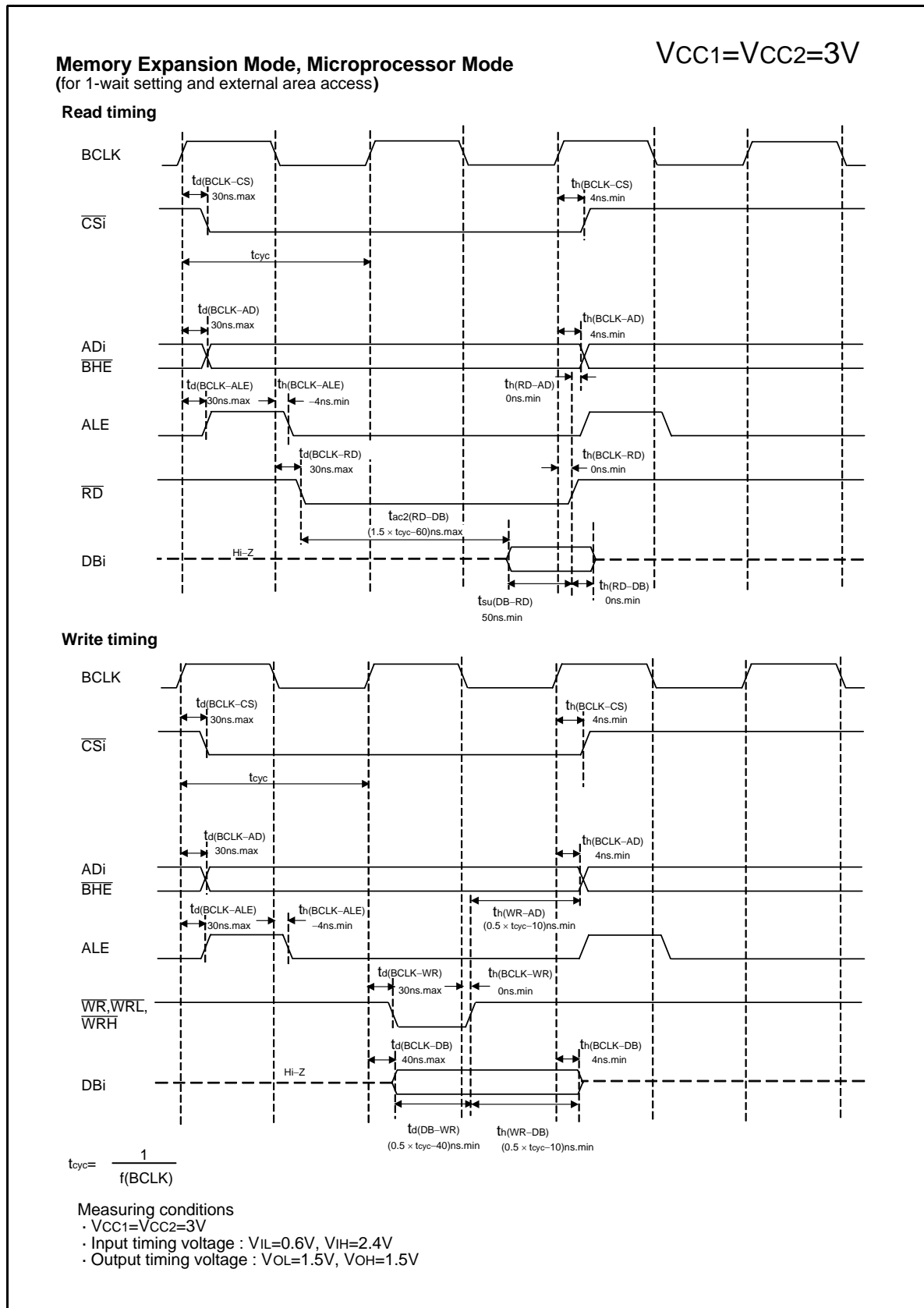


Figure 23.17 Timing Diagram (5)

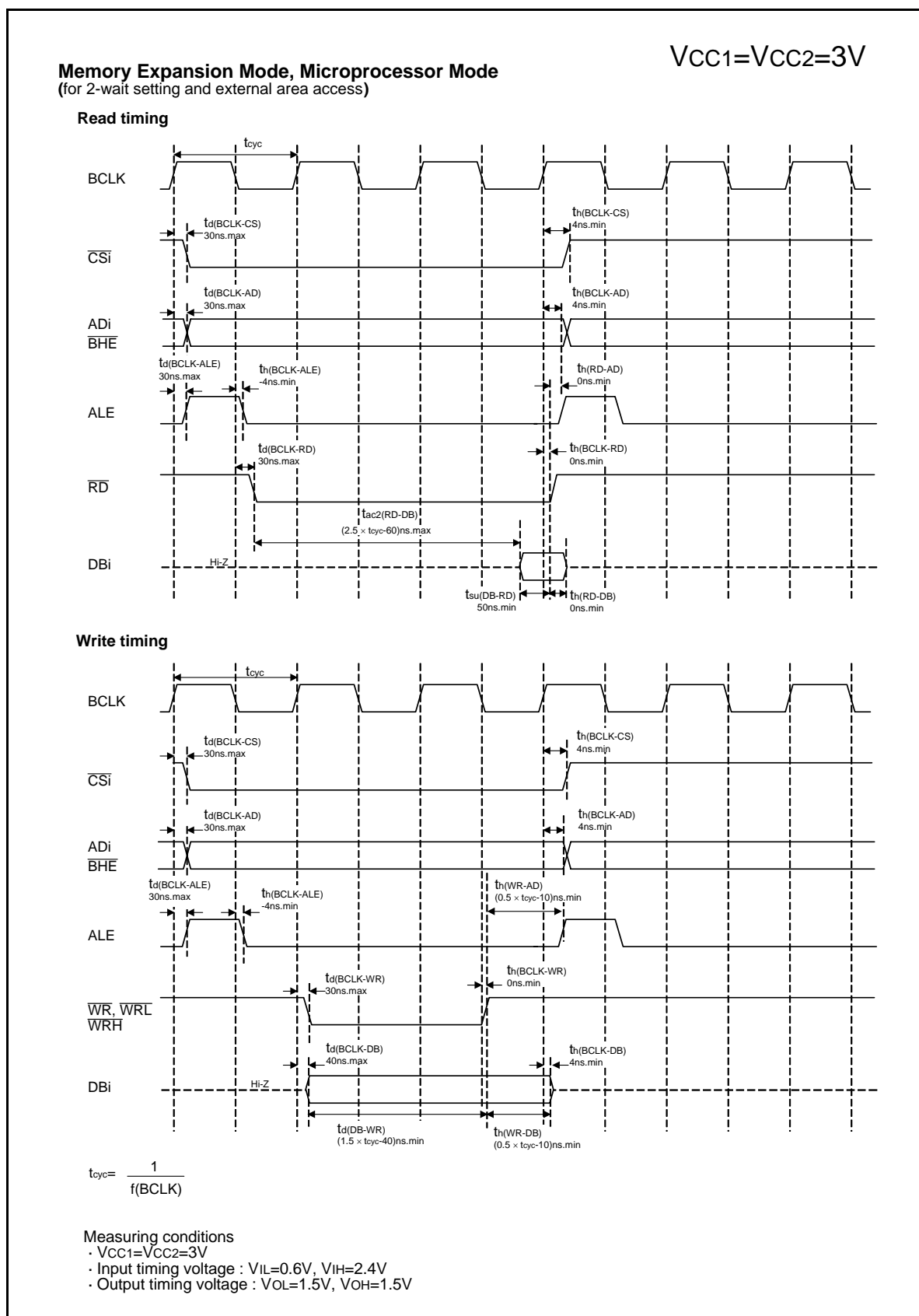


Figure 23.18 Timing Diagram (6)

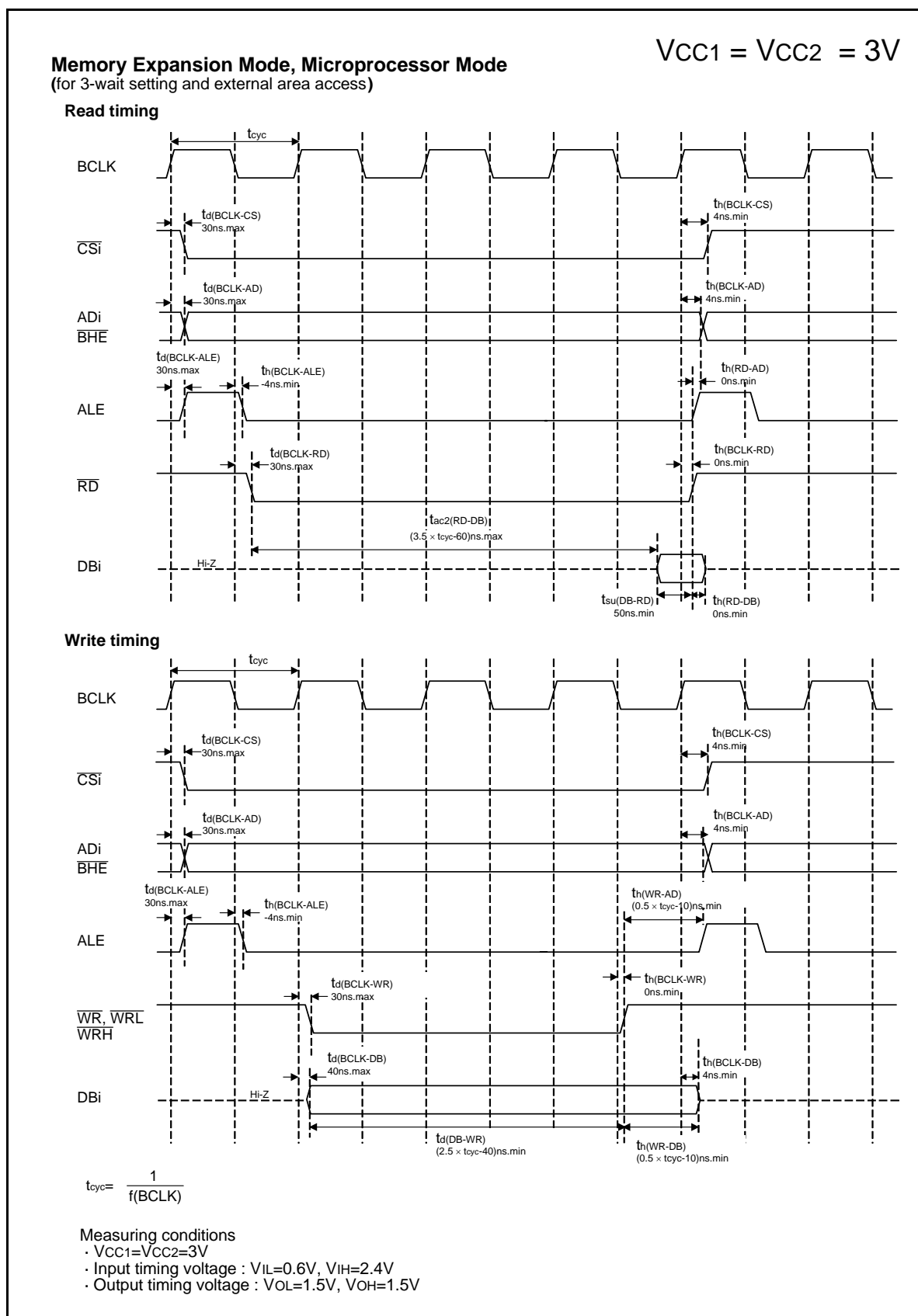


Figure 23.19 Timing Diagram (7)

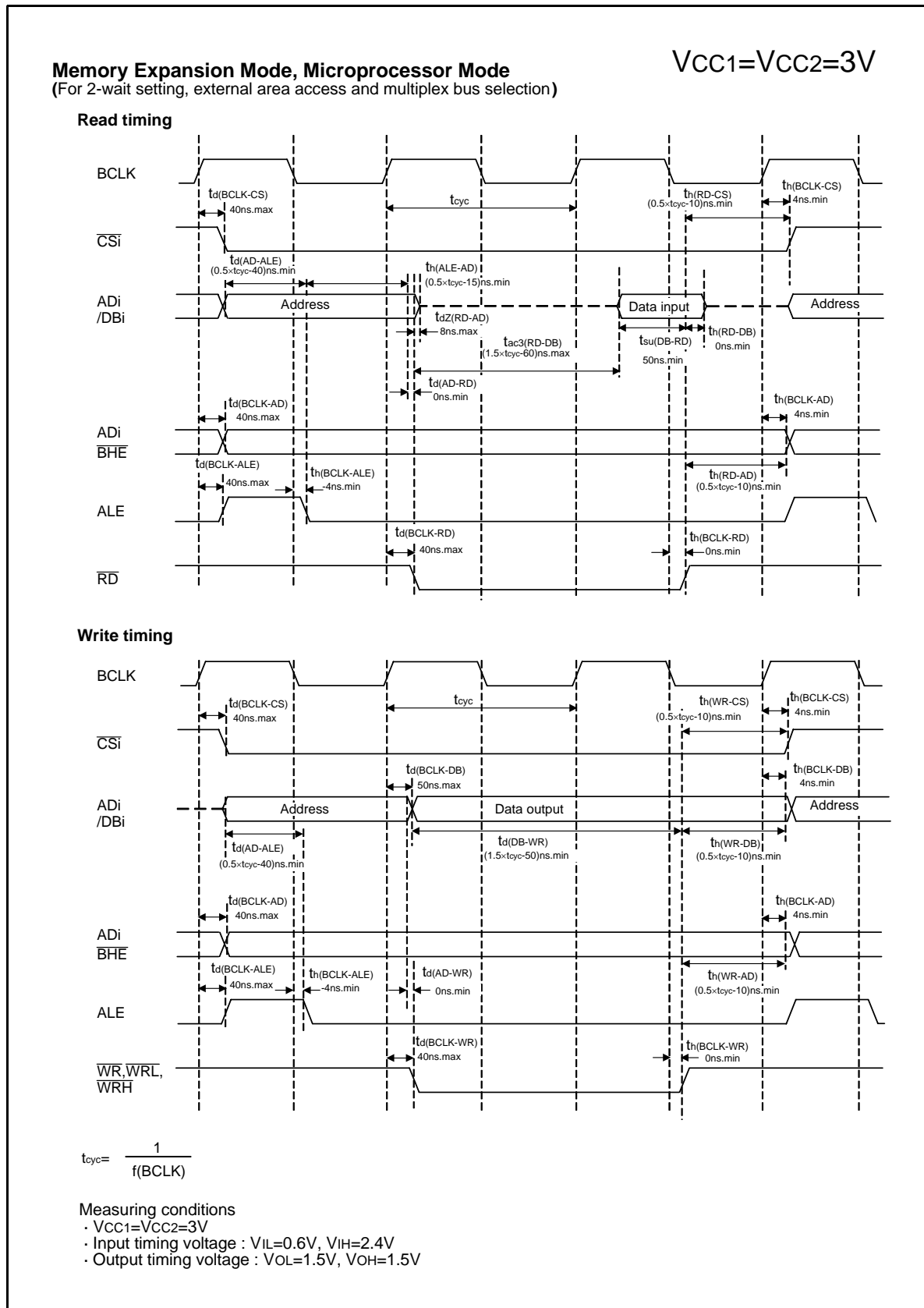
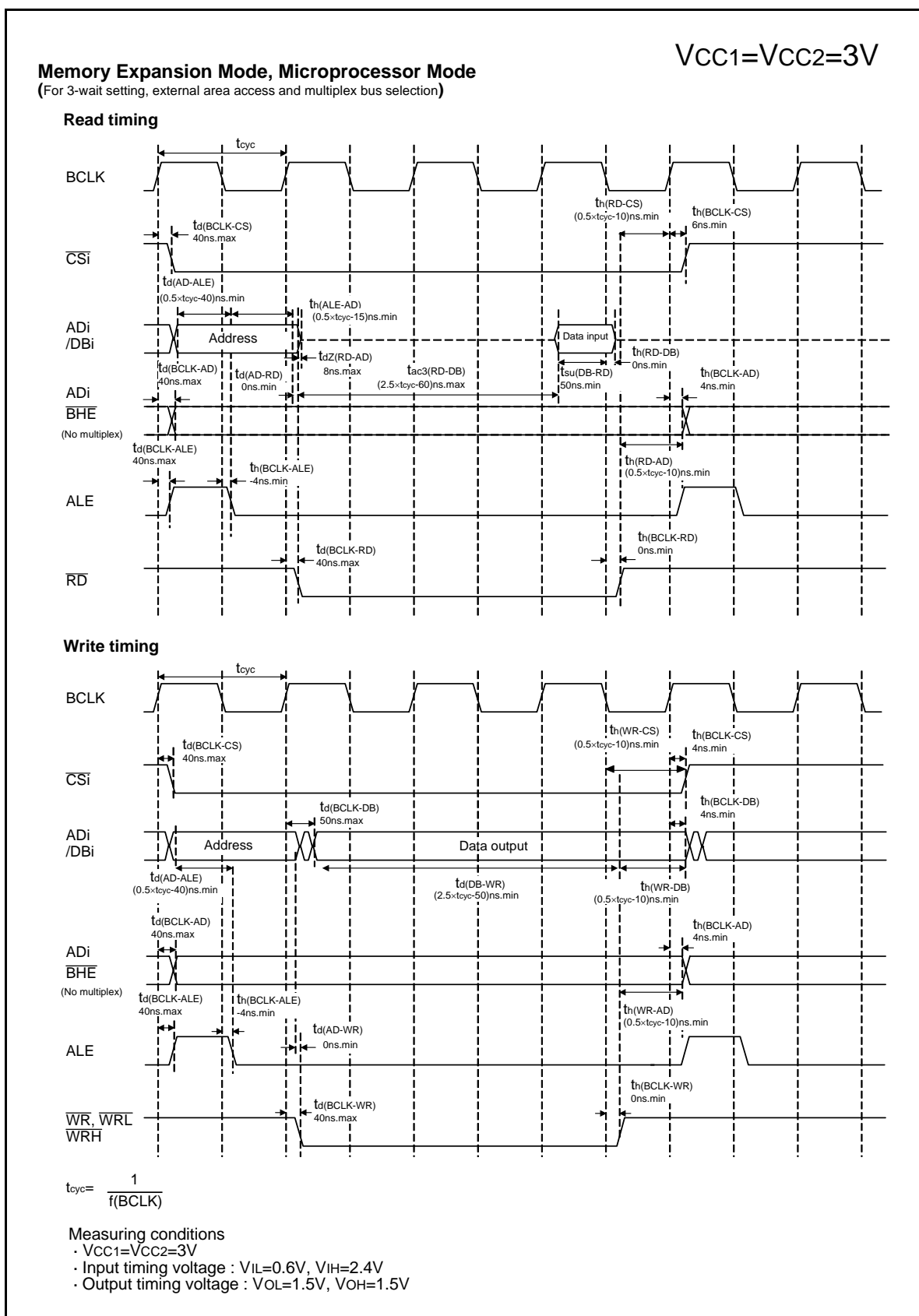


Figure 23.20 Timing Diagram (8)



**Figure 23.21 Timing Diagram (9)**



## 23.2 Electrical Characteristics (M16C/62PT)

**Table 23.49 Absolute Maximum Ratings**

Symbol	Parameter		Condition	Rated Value	Unit
V <sub>CC1</sub> , V <sub>CC2</sub>	Supply Voltage		V <sub>CC1</sub> =V <sub>CC2</sub> =AV <sub>CC</sub>	–0.3 to 6.5	V
AV <sub>CC</sub>	Analog Supply Voltage		V <sub>CC1</sub> =V <sub>CC2</sub> =AV <sub>CC</sub>	–0.3 to 6.5	V
V <sub>I</sub>	Input Voltage	RESET, CNVSS, BYTE, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1, VREF, XIN		–0.3 to V <sub>CC1</sub> +0.3 <sup>(1)</sup>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7		–0.3 to V <sub>CC2</sub> +0.3 <sup>(1)</sup>	V
		P7_0, P7_1		–0.3 to 6.5	V
V <sub>O</sub>	Output Voltage	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, P11_0 to P11_7, P14_0, P14_1, XOUT		–0.3 to V <sub>CC1</sub> +0.3 <sup>(1)</sup>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7		–0.3 to V <sub>CC2</sub> +0.3 <sup>(1)</sup>	V
		P7_0, P7_1		–0.3 to 6.5	V
P <sub>d</sub>	Power Dissipation		–40°C<T <sub>opr</sub> ≤85°C	300	mW
			85°C<T <sub>opr</sub> ≤125°C	200	
T <sub>opr</sub>	Operating Ambient Temperature	When the Microcomputer is Operating		–40 to 85 / –40 to 125 <sup>(2)</sup>	°C
		Flash Program Erase		0 to 60	
T <sub>stg</sub>	Storage Temperature			–65 to 150	°C

**NOTES:**

- There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.
- T version = –40 to 85 °C, V version= –40 to 125 °C.

**Table 23.50 Recommended Operating Conditions (1) (1)**

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
V <sub>CC1</sub> , V <sub>CC2</sub>	Supply Voltage (V <sub>CC1</sub> = V <sub>CC2</sub> )		4.0	5.0	5.5	V
AV <sub>CC</sub>	Analog Supply Voltage			V <sub>CC1</sub>		V
V <sub>SS</sub>	Supply Voltage			0		V
AV <sub>SS</sub>	Analog Supply Voltage			0		V
V <sub>IH</sub>	HIGH Input Voltage (4)	P3_1 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	0.8V <sub>CC2</sub>		V <sub>CC2</sub>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (during single-chip mode)	0.8V <sub>CC2</sub>		V <sub>CC2</sub>	V
		P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	0.8V <sub>CC1</sub>		V <sub>CC1</sub>	V
		P7_0, P7_1	0.8V <sub>CC1</sub>		6.5	V
V <sub>IL</sub>	LOW Input Voltage (4)	P3_1 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	0		0.2V <sub>CC2</sub>	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 (during single-chip mode)	0		0.2V <sub>CC2</sub>	V
		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, P11_0 to P11_7, P14_0, P14_1, XIN, RESET, CNVSS, BYTE	0		0.2V <sub>CC</sub>	V
I <sub>OH(peak)</sub>	HIGH Peak Output Current (4)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			-10.0	mA
I <sub>OH(avg)</sub>	HIGH Average Output Current (4)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			-5.0	mA
I <sub>OL(peak)</sub>	LOW Peak Output Current (4)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			10.0	mA
I <sub>OL(avg)</sub>	LOW Average Output Current (4)	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1			5.0	mA
f(XIN)	Main Clock Input Oscillation Frequency	V <sub>CC1</sub> =4.0V to 5.5V	0		16	MHz
f(XCIN)	Sub-Clock Oscillation Frequency			32.768	50	kHz
f(Ring)	On-chip Oscillation Frequency		0.5	1	2	MHz
f(PLL)	PLL Clock Oscillation Frequency	V <sub>CC1</sub> =4.0V to 5.5V	10		24	MHz
f(BCLK)	CPU Operation Clock		0		24	MHz
tsu(PLL)	PLL Frequency Synthesizer Stabilization Wait Time	V <sub>CC1</sub> =5.5V			20	ms

## NOTES:

1. Referenced to V<sub>CC1</sub> = V<sub>CC2</sub> = 4.7 to 5.5V at T<sub>opr</sub> = -40 to 85°C / -40 to 125°C unless otherwise specified.  
T version = -40 to 85 °C, V version = -40 to 125 °C.
2. The Average Output Current is the mean value within 100ms.
3. The total I<sub>OL(peak)</sub> for ports P0, P1, P2, P8\_6, P8\_7, P9, P10, P14\_0 and P14\_1 must be 80mA max. The total I<sub>OL(peak)</sub> for ports P3, P4, P5, P6, P7, P8\_0 to P8\_4, P12, and P13 must be 80mA max. The total I<sub>OH(peak)</sub> for ports P0, P1, and P2 must be -40mA max. The total I<sub>OH(peak)</sub> for ports P3, P4, P5, P12, and P13 must be -40mA max. The total I<sub>OH(peak)</sub> for ports P6, P7, and P8\_0 to P8\_4 must be -40mA max. The total I<sub>OH(peak)</sub> for ports P8\_6, P8\_7, P9, P10, P11, P14\_0, and P14\_1 must be -40mA max.  
As for 80-pin version, the total I<sub>OL(peak)</sub> for all ports and I<sub>OH(peak)</sub> must be 80mA. max. due to one V<sub>CC</sub> and one V<sub>SS</sub>.
4. There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.

**Table 23.51 A/D Conversion Characteristics (1)**

Symbol	Parameter		Measuring Condition	Standard			Unit
				Min.	Typ.	Max.	
–	Resolution		VREF=VCC1			10	Bits
INL	Integral Non-Linearity Error	10bit	VREF=VCC1=5V AN0 to AN7 input, AN0_0 to AN0_7 input, AN2_0 to AN2_7 input, ANEX0, ANEX1 input			±3	LSB
			External operation amp connection mode			±7	LSB
		8bit	VREF=VCC1=5V			±2	LSB
–	Absolute Accuracy	10bit	VREF=VCC1=5V AN0 to AN7 input, AN0_0 to AN0_7 input, AN2_0 to AN2_7 input, ANEX0, ANEX1 input			±3	LSB
			External operation amp connection mode			±7	LSB
		8bit	VREF=VCC1=5V			±2	LSB
–	Tolerance Level Impedance				3		kΩ
DNL	Differential Non-Linearity Error					±1	LSB
–	Offset Error					±3	LSB
–	Gain Error					±3	LSB
RLADDER	Ladder Resistance		VREF=VCC1	10		40	kΩ
tCONV	10-bit Conversion Time, Sample & Hold Function Available		VREF=VCC1=5V, φAD=12MHz	2.75			μs
tCONV	8-bit Conversion Time, Sample & Hold Function Available		VREF=VCC1=5V, φAD=12MHz	2.33			μs
tsAMP	Sampling Time			0.25			μs
VREF	Reference Voltage			2.0		VCC1	V
VIA	Analog Input Voltage			0		VREF	V

## NOTES:

1. Referenced to VCC1=AVCC=VREF=4.0 to 5.5V, VSS=AVSS=0V at T<sub>opr</sub> = –40 to 85°C / –40 to 125°C unless otherwise specified.  
T version = –40 to 85°C, V version = –40 to 125°C
2. φAD frequency must be 12 MHz or less.
3. When sample & hold is disabled, φAD frequency must be 250 kHz or more, in addition to the limitation in Note 2.  
When sample & hold is enabled, φAD frequency must be 1MHz or more, in addition to the limitation in Note 2.

**Table 23.52 D/A Conversion Characteristics (1)**

Symbol	Parameter	Measuring Condition	Standard			Unit
			Min.	Typ.	Max.	
–	Resolution				8	Bits
–	Absolute Accuracy				1.0	%
tsu	Setup Time				3	μs
Ro	Output Resistance		4	10	20	kΩ
IvREF	Reference Power Supply Input Current	(NOTE 2)			1.5	mA

## NOTES:

1. Referenced to VCC1=VREF=4.0 to 5.5V, VSS=AVSS=0V at T<sub>opr</sub> = –40 to 85°C / –40 to 125°C unless otherwise specified. T version = –40 to 85°C, V version = –40 to 125°C
2. This applies when using one D/A converter, with the D/A register for the unused D/A converter set to “00h”. The resistor ladder of the A/D converter is not included. Also, when D/A register contents are not “00h”, the IvREF will flow even if Vref is disconnected by the A/D control register.

**Table 23.53 Flash Memory Version Electrical Characteristics <sup>(1)</sup> for 100 cycle products (B, U)**

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
–	Program and Erase Endurance <sup>(3)</sup>	100			cycle
–	Word Program Time (V <sub>CC1</sub> =5.0V)		25	200	μs
–	Lock Bit Program Time		25	200	μs
–	Block Erase Time (V <sub>CC1</sub> =5.0V)	4	0.3	4	s
–			0.3	4	s
–			0.5	4	s
–			0.8	4	s
–	Erase All Unlocked Blocks Time <sup>(2)</sup>			4×n	s
tps	Flash Memory Circuit Stabilization Wait Time			15	μs
–	Data Hold Time <sup>(5)</sup>	20			year

**Table 23.54 Flash Memory Version Electrical Characteristics <sup>(6)</sup> for 10,000 cycle products (B7, U7) (Block A and Block 1 <sup>(7)</sup>)**

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
–	Program and Erase Endurance <sup>(3, 8, 9)</sup>	10,000 <sup>(4)</sup>			cycle
–	Word Program Time (V <sub>CC1</sub> =5.0V)		25		μs
–	Lock Bit Program Time		25		μs
–	Block Erase Time (V <sub>CC1</sub> =5.0V)	4	0.3		s
tps	Flash Memory Circuit Stabilization Wait Time			15	μs
–	Data Hold Time <sup>(5)</sup>	20			year

## NOTES:

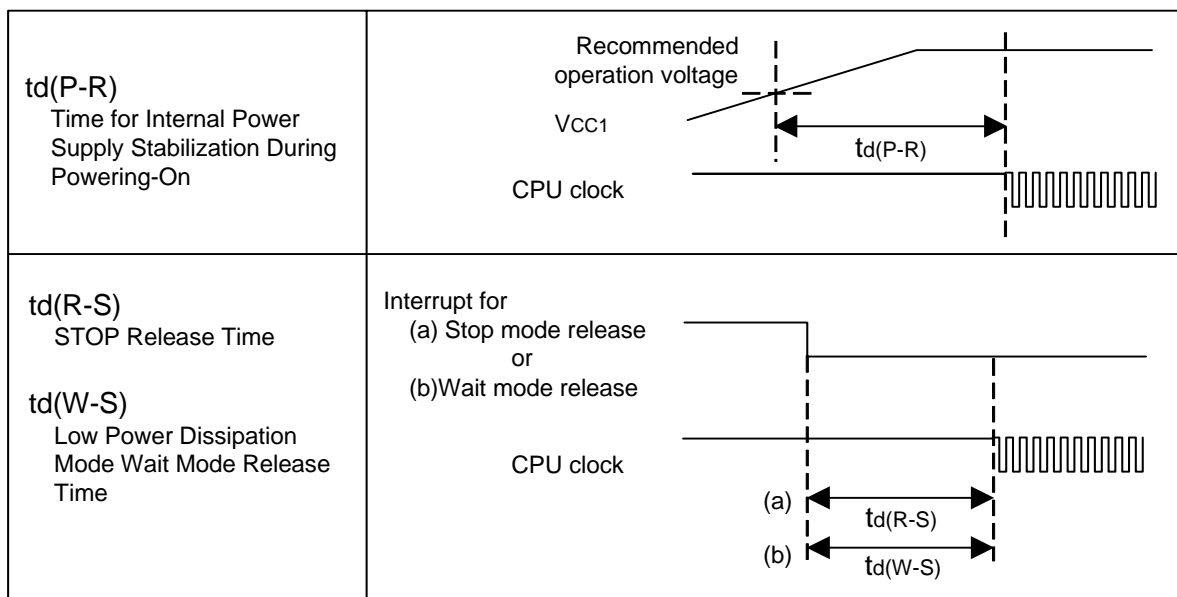
- Referenced to V<sub>CC1</sub>=4.5 to 5.5V at T<sub>opr</sub> = 0 to 60 °C unless otherwise specified.
- n denotes the number of block erases.
- Program and Erase Endurance refers to the number of times a block erase can be performed.  
If the program and erase endurance is n (n=100, 1,000, or 10,000), each block can be erased n times.  
For example, if a 4 Kbytes block A is erased after writing 1 word data 2,048 times, each to a different address, this counts as one program and erase endurance. Data cannot be written to the same address more than once without erasing the block.  
(Rewrite prohibited)
- Maximum number of E/W cycles for which operation is guaranteed.
- Ta (ambient temperature)=55 °C. As to the data hold time except Ta=55 °C, please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor.
- Referenced to V<sub>CC1</sub> = 4.5 to 5.5V at T<sub>opr</sub> = –40 to 85 °C (B7, U7 (T version)) / –40 to 125 °C (B7, U7 (V version)) unless otherwise specified.
- Table 23.54 applies for block A or block 1 program and erase endurance > 1,000. Otherwise, use Table 23.53.
- To reduce the number of program and erase endurance when working with systems requiring numerous rewrites, write to unused word addresses within the block instead of rewrite. Erase block only after all possible addresses are used. For example, an 8-word program can be written 256 times maximum before erase becomes necessary.  
Maintaining an equal number of erasure between block A and block 1 will also improve efficiency. It is important to track the total number of times erasure is used.
- Should erase error occur during block erase, attempt to execute clear status register command, then block erase command at least three times until erase error disappears.
- Set the PM17 bit in the PM1 register to “1” (wait state) when executing more than 100 times rewrites (B7 and U7).
- Customers desiring E/W failure rate information should contact their Renesas technical support representative.

**Table 23.55 Flash Memory Version Program/Erase Voltage and Read Operation Voltage Characteristics (at T<sub>opr</sub> = 0 to 60 °C(B, U), T<sub>opr</sub> = –40 to 85 °C (B7, U7 (T version)) / –40 to 125 °C (B7, U7 (V version))**

Flash Program, Erase Voltage	Flash Read Operation Voltage
V <sub>CC1</sub> = 5.0 V ± 0.5 V	V <sub>CC1</sub> =4.0 to 5.5 V

**Table 23.56 Power Supply Circuit Timing Characteristics**

Symbol	Parameter	Measuring Condition	Standard			Unit
			Min.	Typ.	Max.	
$t_d(P-R)$	Time for Internal Power Supply Stabilization During Powering-On	$V_{CC1}=4.0V$ to $5.5V$			2	ms
$t_d(R-S)$	STOP Release Time				150	$\mu s$
$t_d(W-S)$	Low Power Dissipation Mode Wait Mode Release Time				150	$\mu s$

**Figure 23.22 Power Supply Circuit Timing Diagram**

$$V_{CC1}=V_{CC2}=5V$$

Table 23.57 Electrical Characteristics (1) <sup>(1)</sup>

Symbol	Parameter		Measuring Condition	Standard			Unit
				Min.	Typ.	Max.	
VOH	HIGH Output Voltage <sup>(2)</sup>	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, P11_0 to P11_7, P14_0, P14_1	IOH=−5mA	VCC1−2.0		VCC1	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOH=−5mA	VCC2−2.0		VCC2	
VOH	HIGH Output Voltage <sup>(2)</sup>	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, P11_0 to P11_7, P14_0, P14_1	OH=−200μA	VCC1−0.3		VCC1	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOH=−200μA	VCC2−0.3		VCC2	
VOH	HIGH Output Voltage XOUT	HIGHPOWER	IOH=−1mA	VCC1−2.0		VCC1	V
		LOWPOWER	IOH=−0.5mA	VCC1−2.0		VCC1	
	HIGH Output Voltage XCOUT	HIGHPOWER	With no load applied		2.5		V
		LOWPOWER	With no load applied		1.6		
VOL	LOW Output Voltage <sup>(2)</sup>	P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1	IOL=5mA			2.0	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOL=5mA			2.0	
VOL	LOW Output Voltage <sup>(2)</sup>	P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P14_0, P14_1	IOL=200μA			0.45	V
		P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P12_0 to P12_7, P13_0 to P13_7	IOL=200μA			0.45	
VOL	LOW Output Voltage XOUT	HIGHPOWER	IOL=1mA			2.0	V
		LOWPOWER	IOL=0.5mA			2.0	
	LOW Output Voltage XCOUT	HIGHPOWER	With no load applied		0		V
		LOWPOWER	With no load applied		0		
VT+−VT−	Hysteresis	$\overline{HOLD}$ , $\overline{RDY}$ , $\overline{TA0IN}$ to $\overline{TA4IN}$ , $\overline{TB0IN}$ to $\overline{TB5IN}$ , $\overline{INT0}$ to $\overline{INT5}$ , $\overline{NMI}$ , $\overline{ADTRG}$ , $\overline{CTS0}$ to $\overline{CTS2}$ , $\overline{CLK0}$ to $\overline{CLK4}$ , $\overline{TA0OUT}$ to $\overline{TA4OUT}$ , $\overline{KIO}$ to $\overline{K13}$ , $\overline{RXD0}$ to $\overline{RXD2}$ , $\overline{SCL0}$ to $\overline{SCL2}$ , $\overline{SDA0}$ to $\overline{SDA2}$ , $\overline{SIN3}$ , $\overline{SIN4}$		0.2		1.0	V
VT+−VT−	Hysteresis	$\overline{RESET}$		0.2		2.5	V
IIH	HIGH Input Current <sup>(2)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1, XIN, $\overline{RESET}$ , CNVSS, BYTE	VI=5V			5.0	μA
IIL	LOW Input Current <sup>(2)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_0 to P7_7, P8_0 to P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1, XIN, $\overline{RESET}$ , CNVSS, BYTE	VI=0V			−5.0	μA
RPULLUP	Pull-Up Resistance <sup>(2)</sup>	P0_0 to P0_7, P1_0 to P1_7, P2_0 to P2_7, P3_0 to P3_7, P4_0 to P4_7, P5_0 to P5_7, P6_0 to P6_7, P7_2 to P7_7, P8_0 to P8_4, P8_6, P8_7, P9_0 to P9_7, P10_0 to P10_7, P11_0 to P11_7, P12_0 to P12_7, P13_0 to P13_7, P14_0, P14_1	VI=0V	30	50	170	kΩ
RIXIN	Feedback Resistance XIN				1.5		MΩ
RXCIN	Feedback Resistance XCIN				15		MΩ
VRAM	RAM Retention Voltage		At stop mode	2.0			V

## NOTES:

1. Referenced to VCC1=VCC2=4.0 to 5.5V, VSS = 0V at T<sub>opr</sub> = −40 to 85°C / −40 to 125°C, f(BCLK)=24MHz unless otherwise specified. T version = −40 to 85°C, V version = −40 to 125°C.
2. There is no external connections for port P1\_0 to P1\_7, P4\_4 to P4\_7, P7\_2 to P7\_5 and P9\_1 in 80-pin version.

**Table 23.58 Electrical Characteristics (2) <sup>(1)</sup>**

Symbol	Parameter		Measuring Condition		Standard			Unit
					Min.	Typ.	Max.	
I <sub>CC</sub>	Power Supply Current (V <sub>CC1</sub> =V <sub>CC2</sub> =4.0V to 5.5V)	In single-chip mode, the output pins are open and other pins are V <sub>SS</sub>	Mask ROM	f(BCLK)=24MHz No division, PLL operation		14	20	mA
				No division, On-chip oscillation		1		mA
			Flash Memory	f(BCLK)=24MHz, No division, PLL operation		18	27	mA
				No division, On-chip oscillation		1.8		mA
			Flash Memory Program	f(BCLK)=10MHz, V <sub>CC1</sub> =5.0V		15		mA
			Flash Memory Erase	f(BCLK)=10MHz, V <sub>CC1</sub> =5.0V		25		mA
			Mask ROM	f(XCIN)=32kHz Low power dissipation mode, ROM <sup>(3)</sup>		25		μA
			Flash Memory	f(BCLK)=32kHz Low power dissipation mode, RAM <sup>(3)</sup>		25		μA
				f(BCLK)=32kHz Low power dissipation mode, Flash Memory <sup>(3)</sup>		420		μA
				On-chip oscillation, Wait mode		50		μA
			Mask ROM Flash Memory	f(BCLK)=32kHz Wait mode <sup>(2)</sup> , Oscillation capability High		7.5		μA
				f(BCLK)=32kHz Wait mode <sup>(2)</sup> , Oscillation capability Low		2.0		μA
				Stop mode T <sub>opr</sub> =25°C		2.0	6.0	μA
				Stop mode T <sub>opr</sub> =85°C			20	μA
				Stop mode T <sub>opr</sub> =125°C			TBD	μA

## NOTES:

1. Referenced to V<sub>CC1</sub>=V<sub>CC2</sub>=4.0 to 5.5V, V<sub>SS</sub> = 0V at T<sub>opr</sub> = -40 to 85°C / -40 to 125°C, f(BCLK)=24MHz unless otherwise specified. T version = -40 to 85°C, V version = -40 to 125°C.
2. With one timer operated using fC32.
3. This indicates the memory in which the program to be executed exists.

$$V_{CC1}=V_{CC2}=5V$$

**Timing Requirements**

( $V_{CC1} = V_{CC2} = 5V$ ,  $V_{SS} = 0V$ , at  $T_{opr} = -40$  to  $85^{\circ}C$  (T version) /  $-40$  to  $125^{\circ}C$  (V version) unless otherwise specified)

**Table 23.59 External Clock Input (XIN input)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_c$	External Clock Input Cycle Time	62.5		ns
$t_{w(H)}$	External Clock Input HIGH Pulse Width	25		ns
$t_{w(L)}$	External Clock Input LOW Pulse Width	25		ns
$t_r$	External Clock Rise Time		15	ns
$t_f$	External Clock Fall Time		15	ns



$$V_{CC1}=V_{CC2}=5V$$

**Timing Requirements**

( $V_{CC1} = V_{CC2} = 5V$ ,  $V_{SS} = 0V$ , at  $T_{opr} = -40$  to  $85^{\circ}C$  (T version) /  $-40$  to  $125^{\circ}C$  (V version) unless otherwise specified)

**Table 23.60 Timer A Input (Counter Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN Input Cycle Time	100		ns
$t_{w(TAH)}$	TAiIN Input HIGH Pulse Width	40		ns
$t_{w(TAL)}$	TAiIN Input LOW Pulse Width	40		ns

**Table 23.61 Timer A Input (Gating Input in Timer Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN Input Cycle Time	400		ns
$t_{w(TAH)}$	TAiIN Input HIGH Pulse Width	200		ns
$t_{w(TAL)}$	TAiIN Input LOW Pulse Width	200		ns

**Table 23.62 Timer A Input (External Trigger Input in One-shot Timer Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN Input Cycle Time	200		ns
$t_{w(TAH)}$	TAiIN Input HIGH Pulse Width	100		ns
$t_{w(TAL)}$	TAiIN Input LOW Pulse Width	100		ns

**Table 23.63 Timer A Input (External Trigger Input in Pulse Width Modulation Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN Input HIGH Pulse Width	100		ns
$t_{w(TAL)}$	TAiIN Input LOW Pulse Width	100		ns

**Table 23.64 Timer A Input (Counter Increment/Decrement Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(UP)}$	TAiOUT Input Cycle Time	2000		ns
$t_{w(UPH)}$	TAiOUT Input HIGH Pulse Width	1000		ns
$t_{w(UPL)}$	TAiOUT Input LOW Pulse Width	1000		ns
$t_{su(UP-TIN)}$	TAiOUT Input Setup Time	400		ns
$t_h(TIN-UP)$	TAiOUT Input Hold Time	400		ns

**Table 23.65 Timer A Input (Two-phase Pulse Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN Input Cycle Time	800		ns
$t_{su(TAIN-TAOUT)}$	TAiOUT Input Setup Time	200		ns
$t_{su(TAOUT-TAIN)}$	TAiIN Input Setup Time	200		ns

$$V_{CC1}=V_{CC2}=5V$$

**Timing Requirements**

( $V_{CC1} = V_{CC2} = 5V$ ,  $V_{SS} = 0V$ , at  $T_{opr} = -40$  to  $85^{\circ}C$  (T version) /  $-40$  to  $125^{\circ}C$  (V version) unless otherwise specified)

**Table 23.66 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)	100		ns
$t_{w(TBH)}$	TBiIN Input HIGH Pulse Width (counted on one edge)	40		ns
$t_{w(TBL)}$	TBiIN Input LOW Pulse Width (counted on one edge)	40		ns
$t_{c(TB)}$	TBiIN Input Cycle Time (counted on both edges)	200		ns
$t_{w(TBH)}$	TBiIN Input HIGH Pulse Width (counted on both edges)	80		ns
$t_{w(TBL)}$	TBiIN Input LOW Pulse Width (counted on both edges)	80		ns

**Table 23.67 Timer B Input (Pulse Period Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN Input Cycle Time	400		ns
$t_{w(TBH)}$	TBiIN Input HIGH Pulse Width	200		ns
$t_{w(TBL)}$	TBiIN Input LOW Pulse Width	200		ns

**Table 23.68 Timer B Input (Pulse Width Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN Input Cycle Time	400		ns
$t_{w(TBH)}$	TBiIN Input HIGH Pulse Width	200		ns
$t_{w(TBL)}$	TBiIN Input LOW Pulse Width	200		ns

**Table 23.69 A/D Trigger Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	ADTRG Input Cycle Time	1000		ns
$t_{w(ADL)}$	ADTRG input LOW Pulse Width	125		ns

**Table 23.70 Serial Interface**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi Input Cycle Time	200		ns
$t_{w(CKH)}$	CLKi Input HIGH Pulse Width	100		ns
$t_{w(CKL)}$	CLKi Input LOW Pulse Width	100		ns
$t_{d(C-Q)}$	TXDi Output Delay Time		80	ns
$t_{h(C-Q)}$	TXDi Hold Time	0		ns
$t_{su(D-C)}$	RXDi Input Setup Time	70		ns
$t_{h(C-D)}$	RXDi Input Hold Time	90		ns

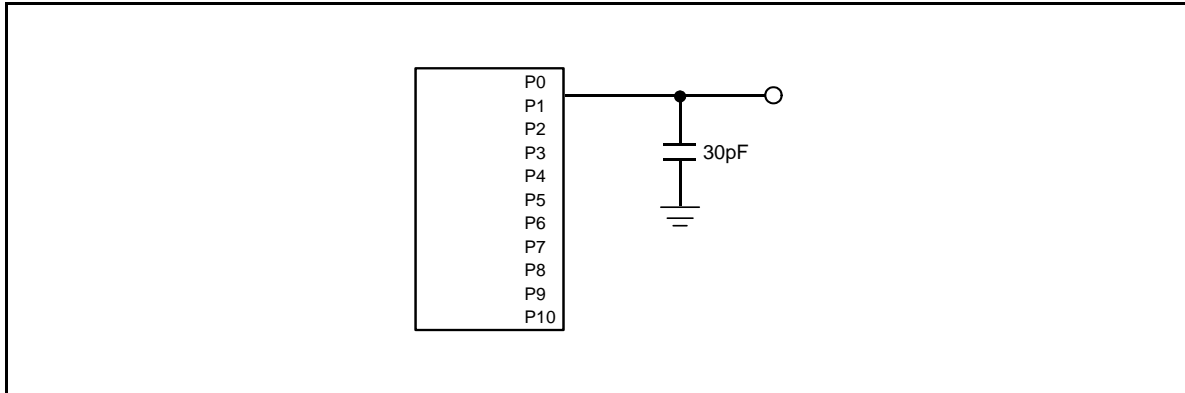
**Table 23.71 External Interrupt INTi Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{INTi}$ Input HIGH Pulse Width	250		ns
$t_{w(INL)}$	$\overline{INTi}$ Input LOW Pulse Width	250		ns

$$V_{CC1}=V_{CC2}=5V$$

**Switching Characteristics**

( $V_{CC1} = V_{CC2} = 5V$ ,  $V_{SS} = 0V$ , at  $T_{opr} = -40$  to  $85^{\circ}C$  (T version) /  $-40$  to  $125^{\circ}C$  (V version) unless otherwise specified)



**Figure 23.23** Ports P0 to P10 Measurement Circuit

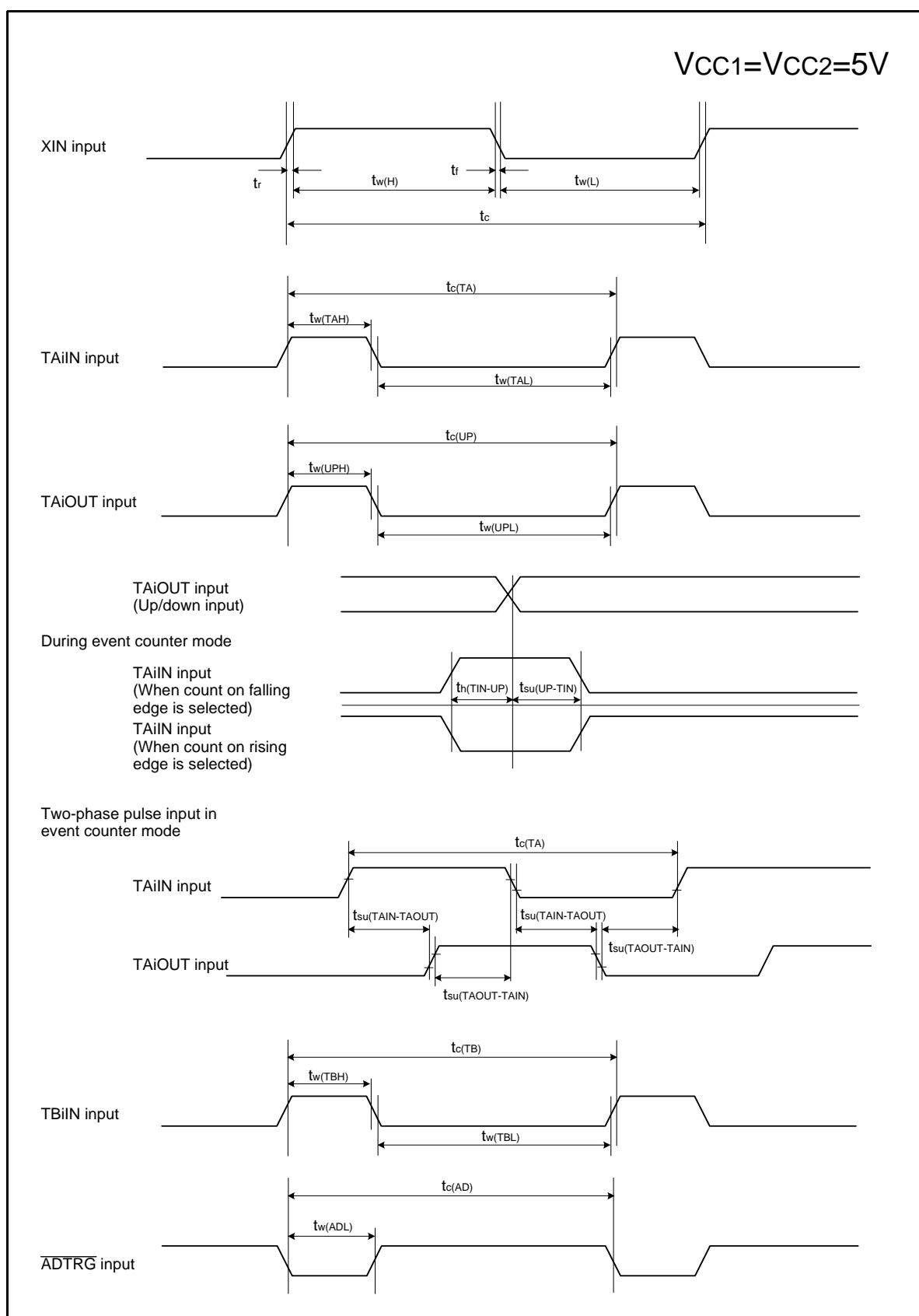


Figure 23.24 Timing Diagram (1)

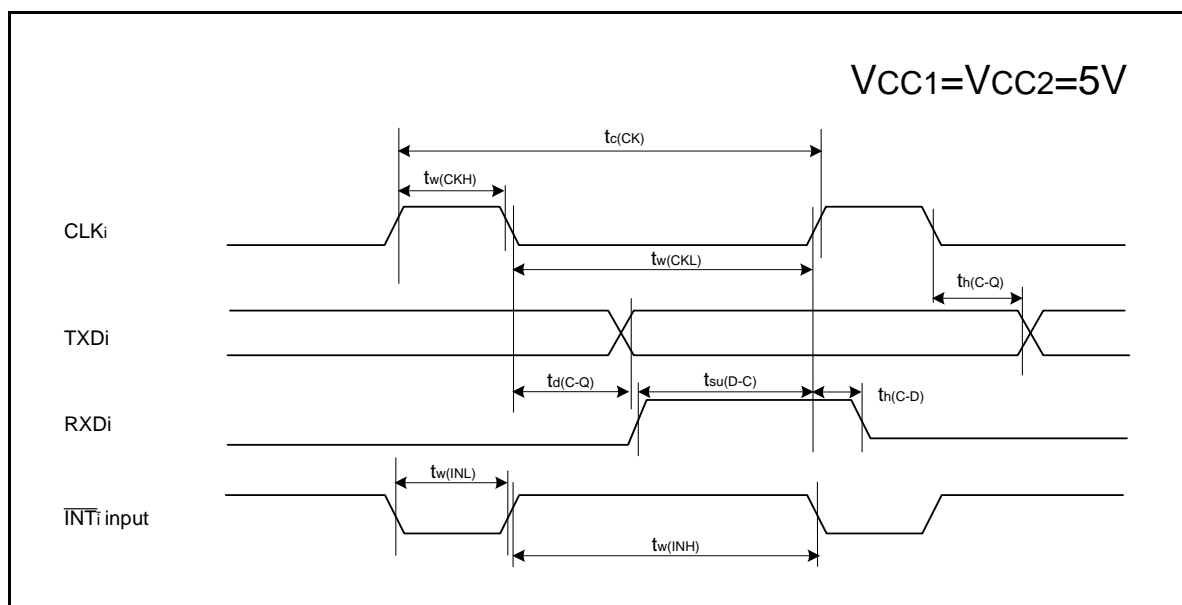


Figure 23.25 Timing Diagram (2)

## 24. Precautions

### 24.1 SFR

#### 24.1.1 Register Settings

Table 24.1 Registers with Write-only Bits which can only be written to. Set these registers with immediate values. When establishing the next value by altering the present value, write the present value to the RAM as well as to the register. Transfer the next value to the register after making changes in the RAM.

**Table 24.1 Registers with Write-only Bits**

Register	Symbol	Address
Watchdog timer start register	WDC	000E
Timer A1-1 register	TA11	0343 to 0342
Timer A2-1 register	TA21	0345 to 0344
Timer A4-1 register	TA41	0347 to 0346
Short-circuit preventionTimer	DTT	034C
Timer B2 Interrupt Generating Frequency Set Counter	ICTB2	034D
SI/03 bit rate register	S3BRG	0363
SI/04 bit rate register	S4BRG	0367
UART0 bit rate register	U0BRG	03A1
UART1 bit rate register	U1BRG	03A9
UART2 bit rate register	U2BRG	0379
UART0 Transmit buffer register	U0TB	03A3 to 03A2
UART1 Transmit buffer register	U1TB	03AB to 03AA
UART2 Transmit buffer register	U2TB	037B to 037A
Ups and downs flag	UDF	0384
Timer 0 register	TA0	0387 to 0386
Timer 1 register	TA1	0389 to 0388
Timer 2 register	TA2	038B to 038A
Timer 3 register	TA3	038D to 038C
Timer 4 register	TA4	038F to 038E

24.2 Reset

When supplying power to the microcomputer, the power supply voltage applied to the VCC1 pin must meet the conditions of SVCC.

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

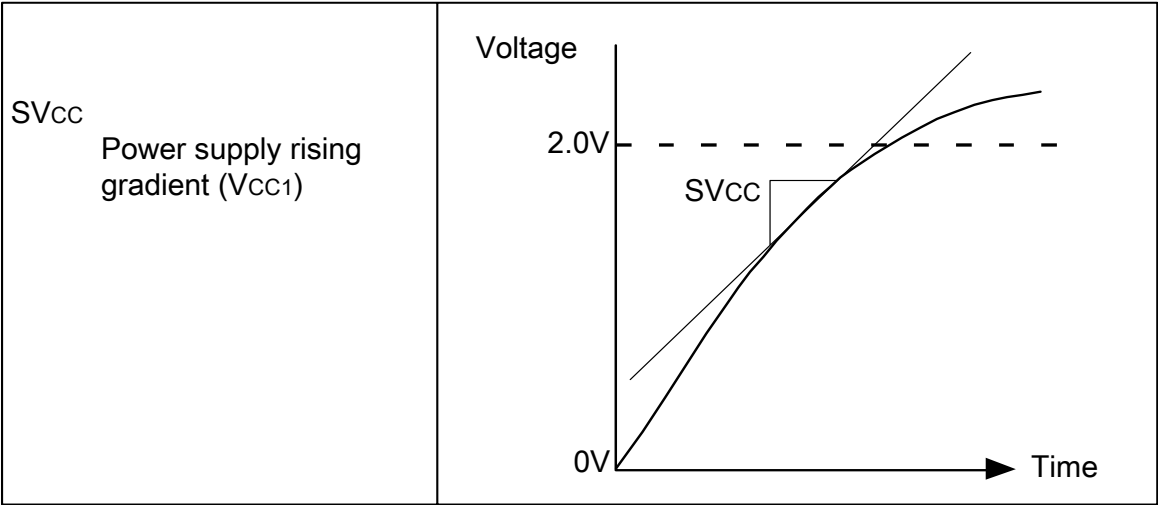


Figure 24.1 Timing of SVCC

### 24.3 Bus

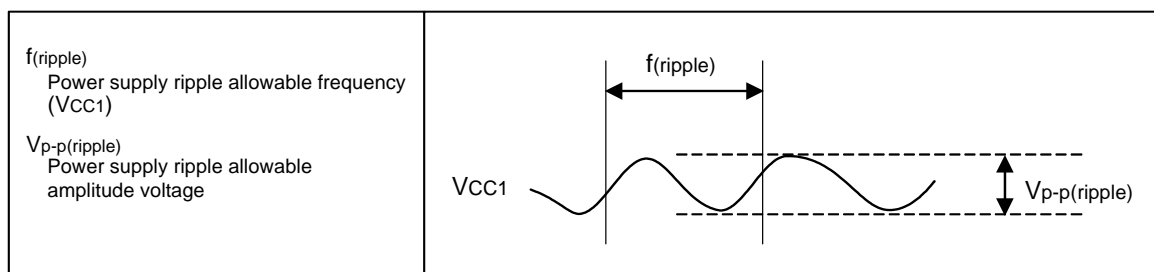
- The ROMless version can operate only in the microprocessor mode, connect the CNVSS pin to VCC1.
- When resetting CNVSS pin with “H” input, contents of internal ROM cannot be read out.



## 24.4 PLL Frequency Synthesizer

Stabilize supply voltage so that the standard of the power supply ripple is met.

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
$f(\text{ripple})$	Power supply ripple allowable frequency (VCC1)			10	kHz
$V_{P-P}(\text{ripple})$	Power supply ripple allowable amplitude voltage	(VCC1=5V)		0.5	V
		(VCC1=3V)		0.3	V
$VCC( \Delta V / \Delta T )$	Power supply ripple rising / falling gradient	(VCC1=5V)		0.3	V/ms
		(VCC1=3V)		0.3	V/ms



**Figure 24.2** Timing of Voltage Fluctuation

## 24.5 Power Control

- When exiting stop mode by hardware reset, set  $\overline{\text{RESET}}$  pin to “L” until a main clock oscillation is stabilized.
- Set the MR0 bit in the TAIMR register (i=0 to 4) to “0” (pulse is not output) to use the timer A to exit stop mode.
- When entering wait mode, insert a JMP.B instruction before a WAIT instruction. Do not execute any instructions which can generate a write to RAM between the JMP.B and WAIT instructions. Disable the DMA transfers, if a DMA transfer may occur between the JMP.B and WAIT instructions. After the WAIT instruction, insert at least 4 NOP instructions. When entering wait mode, the instruction queue reads ahead the instructions following WAIT, and depending on timing, some of these may execute before the microcomputer enters wait mode.

Program example when entering wait mode

```

Program Example:      JMP.B      L1          ; Insert JMP.B instruction before WAIT instruction
                      L1:
                      FSET       I           ;
                      WAIT       ; Enter wait mode
                      NOP        ; More than 4 NOP instructions
                      NOP
                      NOP
                      NOP

```

- When entering stop mode, insert a JMP.B instruction immediately after executing an instruction which sets the CM10 bit in the CM1 register to “1”, and then insert at least 4 NOP instructions. When entering stop mode, the instruction queue reads ahead the instructions following the instruction which sets the CM10 bit to “1” (all clock stops), and, some of these may execute before the microcomputer enters stop mode or before the interrupt routine for returning from stop mode.

Program example when entering stop mode

```

Program Example:      FSET       I           ;
                      BSET       CM10       ; Enter stop mode
                      JMP.B      L2         ; Insert JMP.B instruction
                      L2:
                      NOP        ; More than 4 NOP instructions
                      NOP
                      NOP
                      NOP

```

- Wait until the main clock oscillation stabilizes, before switching the clock source for CPU clock to the main clock.  
Similarly, wait until the sub clock oscillates stably before switching the clock source for CPU clock to the sub clock.

- Suggestions to reduce power consumption

**Ports**

The processor retains the state of each I/O port even when it goes to wait mode or to stop mode. A current flows in active I/O ports. A pass current flows in input ports that high-impedance state. When entering wait mode or stop mode, set non-used ports to input and stabilize the potential.

**A/D converter**

When A/D conversion is not performed, set the VCUT bit of ADiCON1 register to “0” (no VREF connection).

When A/D conversion is performed, start the A/D conversion at least 1  $\mu$ s or longer after setting the VCUT bit to “1” (VREF connection).

**D/A converter**

When not performing D/A conversion, set the DAiE bit (i=0, 1) of DACON register to “0” (input inhibited) and DAi register to “00h”.

**Stopping peripheral functions**

Use the CM0 register CM02 bit to stop the unnecessary peripheral functions during wait mode.

However, because the peripheral function clock (fC32) generated from the sub-clock does not stop, this measure is not conducive to reducing the power consumption of the chip. If low speed mode or low power dissipation mode is to be changed to wait mode, set the CM02 bit to “0” (do not peripheral function clock stopped when in wait mode), before changing wait mode.

**Switching the oscillation-driving capacity**

Set the driving capacity to “LOW” when oscillation is stable.

## 24.6 Protect

Set the PRC2 bit to “1” (write enabled) and then write to any address, and the PRC2 bit will be cleared to “0” (write protected). The registers protected by the PRC2 bit should be changed in the next instruction after setting the PRC2 bit to “1”. Make sure no interrupts or DMA transfers will occur between the instruction in which the PRC2 bit is set to “1” and the next instruction.

## 24.7 Interrupt

### 24.7.1 Reading address 00000h

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

If the address 00000h is read in a program, the IR bit for the interrupt which has the highest priority among the enabled interrupts is cleared to "0". This factors a problem that the interrupt is canceled, or an unexpected interrupt request is generated.

### 24.7.2 Setting the SP

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

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

### 24.7.3 The $\overline{\text{NMI}}$ Interrupt

- The  $\overline{\text{NMI}}$  interrupt cannot be disabled. If this interrupt is unused, connect the  $\overline{\text{NMI}}$  pin to VCC1 via a resistor (pull-up).
- The input level of the  $\overline{\text{NMI}}$  pin can be read by accessing the P8\_5 bit in the P8 register. Note that the P8\_5 bit can only be read when determining the pin level in  $\overline{\text{NMI}}$  interrupt routine.
- Stop mode cannot be entered into while input on the  $\overline{\text{NMI}}$  pin is low. This is because while input on the  $\overline{\text{NMI}}$  pin is low the CM10 bit in the CM1 register is fixed to "0".
- Do not go to wait mode while input on the  $\overline{\text{NMI}}$  pin is low. This is because when input on the  $\overline{\text{NMI}}$  pin goes low, the CPU stops but CPU clock remains active; therefore, the current consumption in the chip does not drop. In this case, normal condition is restored by an interrupt generated thereafter.
- The low and high level durations of the input signal to the  $\overline{\text{NMI}}$  pin must each be 2 CPU clock cycles + 300 ns or more.

### 24.7.4 Changing the Interrupt Generate Factor

If the interrupt generate factor is changed, the IR bit in the interrupt control register for the changed interrupt may inadvertently be set to “1” (interrupt requested). If you changed the interrupt generate factor for an interrupt that needs to be used, be sure to clear the IR bit for that interrupt to “0” (interrupt not requested). Changing the interrupt generate factor referred to here means any act of changing the source, polarity or timing of the interrupt assigned to each software interrupt number. Therefore, if a mode change of any peripheral function involves changing the generate factor, polarity or timing of an interrupt, be sure to clear the IR bit for that interrupt to “0” (interrupt not requested) after making such changes. Refer to the description of each peripheral function for details about the interrupts from peripheral functions. Figure 24.3 shows the Procedure for Changing the Interrupt Generate Factor.

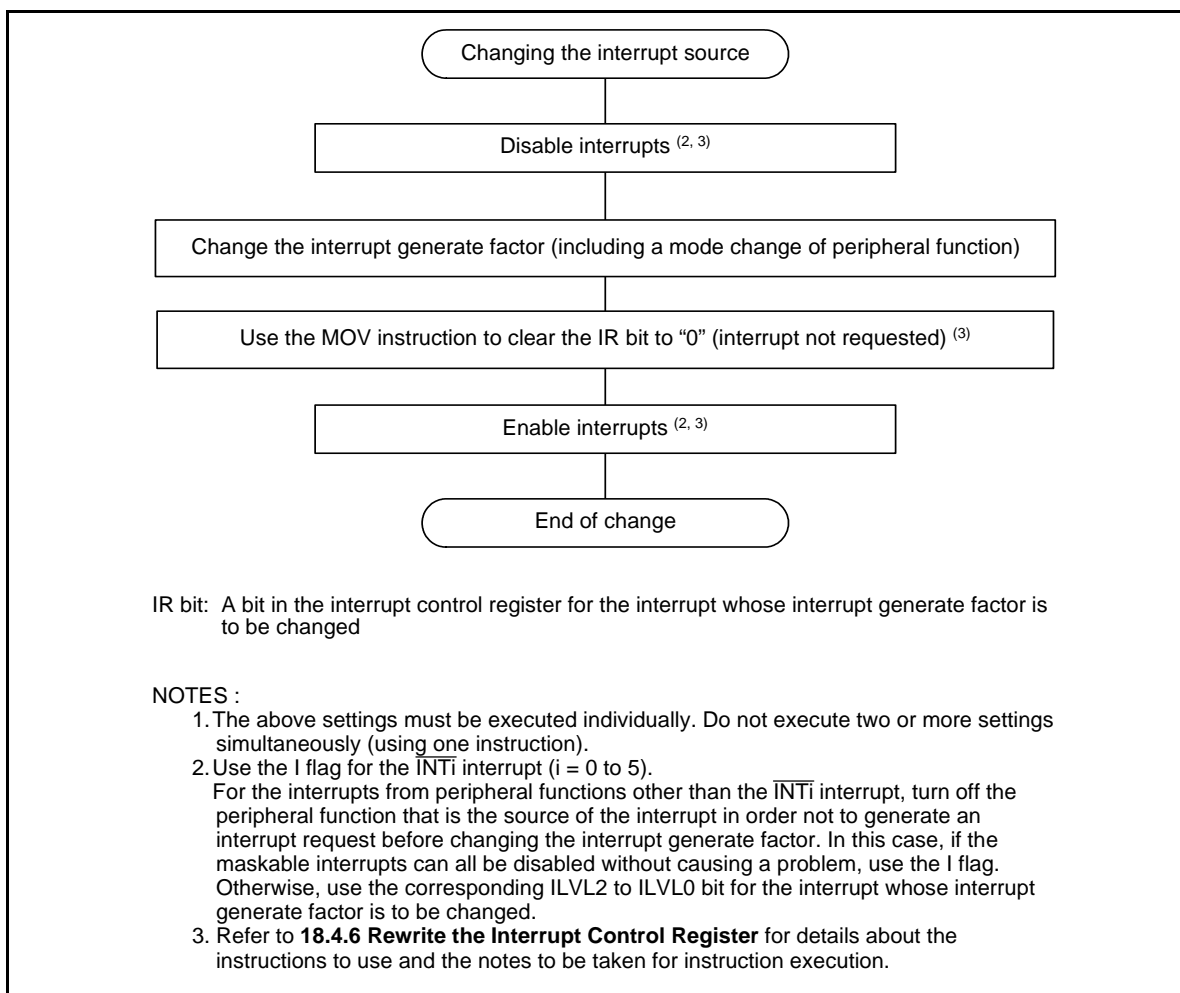


Figure 24.3 Procedure for Changing the Interrupt Generate Factor

### 24.7.5 $\overline{\text{INT}}$ Interrupt

- Either an “L” level of at least  $t_W(\text{INH})$  or an “H” level of at least  $t_W(\text{INL})$  width is necessary for the signal input to pins  $\overline{\text{INT}}_0$  through  $\overline{\text{INT}}_5$  regardless of the CPU operation clock.
- If the POL bit in the INT0IC to INT5IC registers or the IFSR7 to IFSR0 bits in the IFSR register are changed, the IR bit may inadvertently set to “1” (interrupt requested). Be sure to clear the IR bit to “0” (interrupt not requested) after changing any of those register bits.

### 24.7.6 Rewrite the Interrupt Control Register

- (a) The interrupt control register for any interrupt should be modified in places where no requests for that interrupt may occur. Otherwise, disable the interrupt before rewriting the interrupt control register.
- (b) To rewrite the interrupt control register for any interrupt after disabling that interrupt, be careful with the instruction to be used.

- Changing any bit other than the IR bit

- Changing the IR bit

Depending on the instruction used, the IR bit may not always be cleared to “0” (interrupt not requested). Therefore, be sure to use the MOV instruction to clear the IR bit.

- (c) When using the I flag to disable an interrupt, refer to the sample program fragments shown below as you set the I flag. (Refer to (b) for details about rewrite the interrupt control registers in the sample program fragments.)

Examples 1 through 3 show how to prevent the I flag from being set to “1” (interrupts enabled) before 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 keep the program waiting 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.
```

The number of NOP instruction is as follows.

PM20=1(1 wait) : 2, PM20=0(2 wait) : 3, when using HOLD function : 4.

#### Example 2: Using the dummy read to keep the FSET instruction waiting

```
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 changing 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.
```

### 24.7.7 Watchdog Timer Interrupt

Initialize the watchdog timer after the watchdog timer interrupt occurs.

## 24.8 DMAC

### 24.8.1 Write to DMAE Bit in DMiCON Register

When both of the conditions below are met, follow the steps below.

#### Conditions

- The DMAE bit is set to “1” again while it remains set (DMAi is in an active state).
- A DMA request may occur simultaneously when the DMAE bit is being written.

#### Steps

- (1) Write “1” to the DMAE bit and DMAS bit in the DMiCON register simultaneously<sup>(1)</sup>.
- (2) Make sure that the DMAi is in an initial state<sup>(2)</sup> in a program.

If the DMAi is not in an initial state, the above steps should be repeated.

#### NOTES:

1. The DMAS bit remains unchanged even if “1” is written. However, if “0” is written to this bit, it is set to “0” (DMA not requested). In order 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. In this way the state of the DMAS bit immediately before being written can be maintained.  
Similarly, when writing to the DMAE bit with a read-modify-write instruction, “1” should be written to the DMAS bit in order to maintain a DMA request which is generated during execution.
2. Read the TCRi register to verify whether the DMAi is in an initial state. If the read value is equal to a value which was written to the TCRi register before DMA transfer start, the DMAi is in an initial state. (If a DMA request occurs after writing to the DMAE bit, the value written to the TCRi register is “1”.) If the read value is a value in the middle of transfer, the DMAi is not in an initial state.



## 24.9 Timers

### 24.9.1 Timer A

#### 24.9.1.1 Timer A (Timer Mode)

The timer remains idle after reset. Set the mode, count source, counter value, etc. using the TAI<sub>MR</sub> (i = 0 to 4) register and the TAI register before setting the TAI<sub>S</sub> bit in the TABSR register to “1” (count starts). Always make sure the TAI<sub>MR</sub> register is modified while the TAI<sub>S</sub> bit remains “0” (count stops) regardless whether after reset or not.

While counting is in progress, the counter value can be read out at any time by reading the TAI register. However, if the counter is read at the same time it is reloaded, the value “FFFFh” is read.

Also, if the counter is read before it starts counting after a value is set in the TAI register while not counting, the set value is read.

If a low-level signal is applied to the  $\overline{\text{NMI}}$  pin when the IVPCR1 bit in the TB2SC register = 1 (three-phase output forcible cutoff by input on  $\overline{\text{NMI}}$  pin enabled), the TA1OUT, TA2OUT and TA4OUT pins go to a high-impedance state.

#### 24.9.1.2 Timer A (Event Counter Mode)

The timer remains idle after reset. Set the mode, count source, counter value, etc. using the TAI<sub>MR</sub> (i = 0 to 4) register, the TAI register, the UDF register, the ONSF register TAZIE, TA0TGL and TA0TGH bits and the TRGSR register before setting the TAI<sub>S</sub> bit in the TABSR register to “1” (count starts).

Always make sure the TAI<sub>MR</sub> register, the UDF register, the TAZIE, TA0TGL and TA0TGH bits in the ONSF register and the TRGSR register are modified while the TAI<sub>S</sub> bit remains “0” (count stops) regardless whether after reset or not.

While counting is in progress, the counter value can be read out at any time by reading the TAI register. However, “FFFFh” can be read in underflow, while reloading, and “0000h” in overflow. When setting TAI register to a value during a counter stop, the setting value can be read before a counter starts counting. Also, if the counter is read before it starts counting after a value is set in the TAI register while not counting, the set value is read.

If a low-level signal is applied to the  $\overline{\text{NMI}}$  pin when the IVPCR1 bit in the TB2SC register = 1 (three-phase output forcible cutoff by input on  $\overline{\text{NMI}}$  pin enabled), the TA1OUT, TA2OUT and TA4OUT pins go to a high-impedance state.

### 24.9.1.3 Timer A (One-shot Timer Mode)

The timer remains idle after reset. Set the mode, count source, counter value, etc. using the TAI<sub>MR</sub> (i = 0 to 4) register, the TAI register, the TA0TGL and TA0TGH bits in the ONSF register and the TRGSR register before setting the TAI<sub>S</sub> bit in the TABSR register to “1” (count starts).

Always make sure the TAI<sub>MR</sub> register, the TA0TGL and TA0TGH bits and the TRGSR register are modified while the TAI<sub>S</sub> bit remains “0” (count stops) regardless whether after reset or not.

When setting TAI<sub>S</sub> bit to “0” (count stop), the followings occur:

- A counter stops counting and a content of reload register is reloaded.
- TAI<sub>OUT</sub> pin outputs “L”.
- After one cycle of the CPU clock, the IR bit in the TAI<sub>IC</sub> register is set to “1” (interrupt request).

Output in one-shot timer mode synchronizes with a count source internally generated. When an external trigger has been selected, one-cycle delay of a count source as maximum occurs between a trigger input to TAI<sub>IN</sub> pin and output in one-shot timer mode.

The IR bit is set to “1” when timer operating mode is set with any of the following procedures:

- Select one-shot timer mode after reset.
- Change an operating mode from timer mode to one-shot timer mode.
- Change an operating mode from event counter mode to one-shot timer mode.

To use the Timer Ai interrupt (the IR bit), set the IR bit to “0” after the changes listed above have been made.

When a trigger occurs, while counting, a counter reloads the reload register to continue counting after generating a re-trigger and counting down once. To generate a trigger while counting, generate a second trigger between occurring the previous trigger and operating longer than one cycle of a timer count source.

If a low-level signal is applied to the  $\overline{\text{NMI}}$  pin when the IVPCR1 bit in the TB2SC register = 1 (three-phase output forcible cutoff by input on  $\overline{\text{NMI}}$  pin enabled), the TA1OUT, TA2OUT and TA4OUT pins go to a high-impedance state.

### 24.9.1.4 Timer A (Pulse Width Modulation Mode)

The timer remains idle after reset. Set the mode, count source, counter value, etc. using the TAI<sub>MR</sub> (i = 0 to 4) register, the TAI register, the TA0TGL and TA0TGH bits in the ONSF register and the TRGSR register before setting the TAI<sub>S</sub> bit in the TABSR register to “1” (count starts).

Always make sure the TAI<sub>MR</sub> register, TA0TGL and TA0TGH bits and the TRGSR register are modified while the TAI<sub>S</sub> bit remains “0” (count stops) regardless whether after reset or not.

The IR bit is set to “1” when setting a timer operating mode with any of the following procedures:

- Select the PWM mode after reset.
- Change an operating mode from timer mode to PWM mode.
- Change an operating mode from event counter mode to PWM mode.

To use the Timer Ai interrupt (interrupt request bit), set the IR bit to “0” by program after the above listed changes have been made.

When setting TAI<sub>S</sub> register to “0” (count stop) during PWM pulse output, the following action occurs:

- Stop counting.
- When TAI<sub>OUT</sub> pin is output “H”, output level is set to “L” and the IR bit is set to “1”.
- When TAI<sub>OUT</sub> pin is output “L”, both output level and the IR bit remains unchanged.

If a low-level signal is applied to the  $\overline{\text{NMI}}$  pin when the IVPCR1 bit in the TB2SC register = 1 (three-phase output forcible cutoff by input on  $\overline{\text{NMI}}$  pin enabled), the TA1OUT, TA2OUT and TA4OUT pins go to a high-impedance state.

## 24.9.2 Timer B

### 24.9.2.1 Timer B (Timer Mode)

The timer remains idle after reset. Set the mode, count source, counter value, etc. using the TBiMR (i = 0 to 5) register and TBi register before setting the TBiS bit in the TABSR or the TBSR register to “1” (count starts). Always make sure the TBiMR register is modified while the TBiS bit remains “0” (count stops) regardless whether after reset or not.

A value of a counter, while counting, can be read in TBi register at any time. “FFFFh” is read while reloading. Setting value is read between setting values in TBi register at count stop and starting a counter.

### 24.9.2.2 Timer B (Event Counter Mode)

The timer remains idle after reset. Set the mode, count source, counter value, etc. using the TBiMR (i = 0 to 5) register and TBi register before setting the TBiS bit in the TABSR or the TBSR register to “1” (count starts). Always make sure the TBiMR register is modified while the TBiS bit remains “0” (count stops) regardless whether after reset or not.

The counter value can be read out on-the-fly at any time by reading the TBi register. However, if this register is read at the same time the counter is reloaded, the read value is always “FFFFh”. If the TBi register is read after setting a value in it while not counting but before the counter starts counting, the read value is the one that has been set in the register.

### 24.9.2.3 Timer B (Pulse Period/pulse Width Measurement Mode)

The timer remains idle after reset. Set the mode, count source, etc. using the TBiMR (i = 0 to 5) register before setting the TBiS bit in the TABSR or the TBSR register to “1” (count starts). Always make sure the TBiMR register is modified while the TBiS bit remains “0” (count stops) regardless whether after reset or not. To clear the MR3 bit to “0” by writing to the TBiMR register while the TBiS bit = 1 (count starts), be sure to write the same value as previously written to the TM0D0, TM0D1, MR0, MR1, TCK0 and TCK1 bits and a 0 to the MR2 bit.

The IR bit in the TBiIC register (i=0 to 5) goes to “1” (interrupt request), when an effective edge of a measurement pulse is input or Timer Bi is overflowed. The factor of interrupt request can be determined by use of the MR3 bit in the TBiMR register within the interrupt routine.

If the source of interrupt cannot be identified by the MR3 bit such as when the measurement pulse input and a timer overflow occur at the same time, use another timer to count the number of times Timer B has overflowed.

To set the MR3 bit to “0” (no overflow), set TBiMR register with setting the TBiS bit to “1” and counting the next count source after setting the MR3 bit to “1” (overflow).

Use the IR bit to detect only overflows. Use the MR3 bit only to determine the interrupt factor.

When a count is started and the first effective edge is input, an indeterminate value is transferred to the reload register. At this time, Timer Bi interrupt request is not generated.

A value of the counter is indeterminate at the beginning of a count. MR3 may be set to “1” and Timer Bi interrupt request may be generated between a count start and an effective edge input.

For pulse width measurement, pulse widths are successively measured. Use program to check whether the measurement result is an “H” level width or an “L” level width.

## 24.10 Serial interface

### 24.10.1 Clock Synchronous Serial I/O

#### 24.10.1.1 Transmission/reception

With an external clock selected, and choosing the  $\overline{\text{RTS}}$  function, the output level of the  $\overline{\text{RTSi}}$  pin goes to “L” when the data-receivable status becomes ready, which informs the transmission side that the reception has become ready. The output level of the  $\overline{\text{RTSi}}$  pin goes to “H” when reception starts. So if the  $\overline{\text{RTSi}}$  pin is connected to the  $\overline{\text{CTS}}$  pin on the transmission side, the circuit can transmission and reception data with consistent timing. With the internal clock, the  $\overline{\text{RTS}}$  function has no effect.

If a low-level signal is applied to the  $\overline{\text{NMI}}$  pin when the IVPCR1 bit in the TB2SC register = 1 (three-phase output forcible cutoff by input on  $\overline{\text{NMI}}$  pin enabled), the  $\overline{\text{RTS2}}$  and CLK2 pins go to a high-impedance state.

#### 24.10.1.2 Transmission

When an external clock is selected, the conditions must be met while if the CKPOL bit in the UiC0 register = 0 (transmit data output at the falling edge and the receive data taken in at the rising edge of the transfer clock), the external clock is in the high state; if the CKPOL bit in the UiC0 register = 1 (transmit data output at the rising edge and the receive data taken in at the falling edge of the transfer clock), the external clock is in the low state.

- The TE bit in the UiC1 register = 1 (transmission enabled)
- The TI bit in the UiC1 register = 0 (data present in UiTB register)
- If  $\overline{\text{CTS}}$  function is selected, input on the  $\overline{\text{CTSi}}$  pin = L

#### 24.10.1.3 Reception

In operating the clock-synchronous serial I/O, operating a transmitter generates a shift clock. Fix settings for transmission even when using the device only for reception. Dummy data is output to the outside from the TXDi pin when receiving data.

When an internal clock is selected, set the TE bit in the UiC1 register (i = 0 to 2) to 1 (transmission enabled) and write dummy data to the UiTB register, and the shift clock will thereby be generated.

When an external clock is selected, set the TE bit to 1 and write dummy data to the UiTB register, and the shift clock will be generated when the external clock is fed to the CLKi input pin.

When successively receiving data, if all bits of the next receive data are prepared in the UARTi receive register while the RE bit in the UiC1 register (i = 0 to 2) = 1 (data present in the UiRB register), an overrun error occurs and the OER bit in the UiRB register is set to “1” (overrun error occurred). In this case, because the content of the UiRB register is indeterminate, a corrective measure must be taken by programs on the transmit and receive sides so that the valid data before the overrun error occurred will be retransmitted. Note that when an overrun error occurred, the IR bit in the SiRIC register does not change state.

To receive data in succession, set dummy data in the lower-order byte of the UiTB register every time reception is made.

When an external clock is selected, the conditions must be met while if the CKPOL bit = 0, the external clock is in the high state; if the CKPOL bit = 1, the external clock is in the low state.

- 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)

## 24.10.2 UART

### 24.10.2.1 Special Mode 1(I<sup>2</sup>C Mode)

When generating start, stop and restart conditions, set the STSPSEL bit in the UiSMR4 register to “0” and wait for more than half cycle of the transfer clock before setting each condition generate bit (STAREQ, RSTAREQ and STPREQ) from “0” to “1”.

### 24.10.2.2 Special Mode 2

If a low-level signal is applied to the  $\overline{\text{NMI}}$  pin when the  $\overline{\text{IVPCR1}}$  bit in the TB2SC register = 1 (three-phase output forcible cutoff by input on  $\overline{\text{NMI}}$  pin enabled), the  $\overline{\text{RTS2}}$  and CLK2 pins go to a high-impedance state.

### 24.10.2.3 Special Mode 4 (SIM Mode)

A transmit interrupt request is generated by setting the U2C1 register U2IRS bit to “1” (transmission complete) and U2ERE bit to “1” (error signal output) after reset is deasserted. Therefore, when using SIM mode, be sure to clear the IR bit to “0” (no interrupt request) after setting these bits.

## 24.10.3 SI/O3, SI/O4

The SOUTi default value which is set to the SOUTi pin by the SMi7 bit approximately 10ns may be output when changing the SMi3 bit from “0” (I/O port) to “1” (SOUTi output and CLK function) while the SMi2 bit in the SiC (i=3 and 4) to “0” (SOUTi output) and the SMi6 bit is set to “1” (internal clock). And then the SOUTi pin is held high-impedance.

If the level which is output from the SOUTi pin is a problem when changing the SMi3 bit from “0” to “1”, set the default value of the SOUTi pin by the SMi7 bit.

### 24.11 A/D Converter

Set ADCON0 (except bit 6), ADCON1 and ADCON2 registers when A/D conversion is stopped (before a trigger occurs).

When the VCUT bit in the ADCON1 register is changed from “0” (Vref not connected) to “1” (Vref connected), start A/D conversion after passing 1  $\mu$ s or longer.

To prevent noise-induced device malfunction or latchup, as well as to reduce conversion errors, insert capacitors between the AVCC, VREF, and analog input pins (ANi(i=0 to 7), AN0\_i, AN2\_i) each and the AVSS pin. Similarly, insert a capacitor between the VCC1 pin and the VSS pin. Figure 24.4 is an example connection of each pin.

Make sure the port direction bits for those pins that are used as analog inputs are set to “0” (input mode). Also, if the TGR bit in the ADCON0 register = 1 (external trigger), make sure the port direction bit for the  $\overline{\text{ADTRG}}$  pin is set to “0” (input mode).

When using key input interrupts, do not use any of the four AN4 to AN7 pins as analog inputs. (A key input interrupt request is generated when the A/D input voltage goes low.)

The  $\phi_{\text{AD}}$  frequency must be 12MHz or less. Without sample-and-hold function, limit the  $\phi_{\text{AD}}$  frequency to 250kHz or more. With the sample and hold function, limit the  $\phi_{\text{AD}}$  frequency to 1MHz or more.

When changing an A/D operating mode, select analog input pin again in the CH2 to CH0 bits in the ADCON0 register and the SCAN1 to SCAN0 bits in the ADCON1 register.

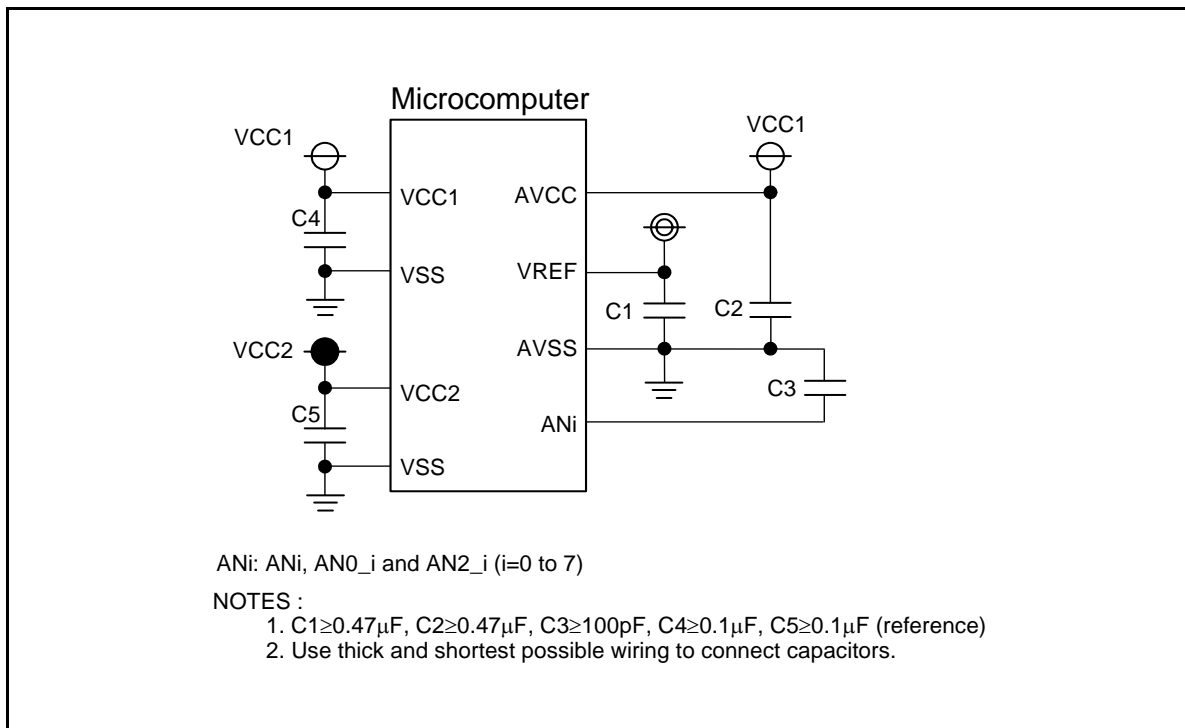


Figure 24.4 Use of Capacitors to Reduce Noise

If  $VCC2 < VCC1$ , do not use AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7 as analog input pins.

If the CPU reads the ADi register ( $i = 0$  to 7) at the same time the conversion result is stored in the ADi register after completion of A/D conversion, an incorrect value may be stored in the ADi register. This problem occurs when a divide-by-n clock derived from the main clock or a subclock is selected for CPU clock.

- When operating in one-shot or single-sweep mode  
Check to see that A/D conversion is completed before reading the target ADi register. (Check the IR bit in the ADIC register to see if A/D conversion is completed.)
- When operating in repeat mode or repeat sweep mode 0 or 1  
Use the main clock for CPU clock directly without dividing it.

If A/D conversion is forcibly terminated while in progress by setting the ADST bit in the ADCON0 register to “0” (A/D conversion halted), the conversion result of the A/D converter is indeterminate. The contents of ADi registers irrelevant to A/D conversion may also become indeterminate. If while A/D conversion is underway the ADST bit is cleared to “0” in a program, ignore the values of all ADi registers.

When setting the ADST bit in the ADCON0 register to “0” in single-sweep mode during A/D conversion and suspending A/D conversion, disable the interrupt before setting the ADST bit to “0”.

The applied intermediate potential may cause more increase in power consumption than other analog input pins (AN0 to AN3, AN0\_0 to AN0\_7 and AN2\_0 to AN2\_7), since the AN4 to AN7 are used with the KI0 to KI3.

## 24.12 Programmable I/O Ports

If a low-level signal is applied to the  $\overline{\text{NMI}}$  pin when the IVPCR1 bit in the TB2SC register = 1 (three-phase output forcible cutoff by input on  $\overline{\text{NMI}}$  pin enabled), the P7\_2 to P7\_5, P8\_0 and P8\_1 pins go to a high-impedance state.

Setting the SM32 bit in the S3C register to “1” causes the P9\_2 pin to go to a high-impedance state.  
Similarly, setting the SM42 bit in the S4C register to “1” causes the P9\_6 pin to go to a high-impedance state.

The input threshold voltage of pins differs between programmable input/output ports and peripheral functions. Therefore, if any pin is shared by a programmable input/output port and a peripheral function and the input level at this pin is outside the range of recommended operating conditions VIH and VIL (neither “high” nor “low”), the input level may be determined differently depending on which side-the programmable input/output port or the peripheral function-is currently selected.

When changing the PD14\_i bit (i=0 to 1) in the PC14 register from “0” (input port) to “1” (output port), follow the procedures below.

### Setting Procedure

- |   |   |
|---|---|
| (1) Set P14_i bit                               | :MOV.B #00000001b, PC14 ; P14_i bit setting     |
| (2) Change PD14_i bit to “1” by MOV instruction | :MOV.B #00110001b, PC14 ; Change to output port |

Indeterminate values are read from the P3\_7 to P3\_4, PD3\_7 to PD3\_4 bits by reading the P3 and PD3 registers when the PM01 to PM00 bits in the PM0 register are set to “01b” (memory expansion mode) or “11b”(microprocessor mode) and setting the PM11 bit to “1”.

Use the MOV instruction when rewriting the P3 and PD3 registers (including the case that the size specifier is “.W” and the P2 and PD2 registers are rewritten).

When the PM01 to PM00 bits are rewritten, “L” is output from the P3\_7 to P3\_4 pins during 0.5 cycles of the BCLK by setting the PM01 to PM00 bits in the PM0 register to “01b” (memory expansion mode) or “11b” (microprocessor mode) from “00b” (single-chip mode) after setting the PM11 bit to “1”.



### **24.13 Electric Characteristic Differences Between Mask ROM and Flash Memory Version Microcomputers**

Flash memory version and mask ROM version may have different characteristics, operating margin, noise tolerated dose, noise width dose in electrical characteristics due to internal ROM, different layout pattern, etc. When switching to the mask ROM version, conduct equivalent tests as system evaluation tests conducted in the flash memory version.

### **24.14 Mask ROM**

When using the masked ROM version, write nothing to internal ROM area.

## 24.15 Flash Memory Version

### 24.15.1 Functions to Inhibit Rewriting Flash Memory Rewrite

ID codes are stored in addresses 0FFFDh, 0FFFE3h, 0FFFEb, 0FFFEFh, 0FFFF3h, 0FFFF7h, and 0FFFFBh. If wrong data are written to these addresses, the flash memory cannot be read or written in standard serial I/O mode.

The ROMCP register is mapped in address 0FFFFFh. If wrong data is written to this address, the flash memory cannot be read or written in parallel I/O mode.

In the flash memory version of microcomputer, these addresses are allocated to the vector addresses (H) of fixed vectors.

### 24.15.2 Stop mode

When the microcomputer enters stop mode, execute the instruction which sets the CM10 bit to "1" (stop mode) after setting the FMR01 bit to "0" (CPU rewrite mode disabled) and disabling the DMA transfer.

### 24.15.3 Wait mode

When shifting to wait mode, set the FMR01 bit to "0" (CPU rewrite mode disabled) before executing the WAIT instruction.

### 24.15.4 Low power dissipation mode, on-chip oscillator low power dissipation mode

If the CM05 bit is set to "1" (main clock stop), the following commands must not be executed.

- Program
- Block erase
- Erase all unlocked blocks
- Lock bit program

### 24.15.5 Writing command and data

Write the command code and data at even addresses.

### 24.15.6 Program Command

Write "xx40h" in the first bus cycle and write data to the write address in the second bus cycle, and an auto program operation (data program and verify) will start. Make sure the address value specified in the first bus cycle is the same even address as the write address specified in the second bus cycle.

### 24.15.7 Lock Bit Program Command

Write "77h" in the first bus cycle and write "xxD0h" to the uppermost address of a block (even address, however) in the second bus cycle, and the lock bit for the specified block is cleared to "0". Make sure then address value specified in the first bus cycle is the same uppermost block address that is specified in the second bus cycle.

### 24.15.8 Operation speed

Before entering CPU rewrite mode (EW0 or EW1 mode), set the CM11 bit in the CM1 register to “0” (main clock), select 10 MHz or less for CPU clock using the CM06 bit in the CM0 register and CM17 to CM16 bits in the CM1 register. Also, set the PM17 bit in the PM1 register to “1” (with wait state).

### 24.15.9 Instructions inhibited against use

The following instructions cannot be used in EW0 mode because the flash memory’s internal data is referenced: UND instruction, INTO instruction, JMPS instruction, JSRS instruction, and BRK instruction.

### 24.15.10 Interrupts

#### EW0 Mode

- Any interrupt which has a vector in the relocatable vector table can be used providing that its vector is transferred into the RAM area.
- The  $\overline{\text{NMI}}$  and watchdog timer interrupts can be used because the FMR0 register and FMR1 register are initialized when one of those interrupts occurs. The jump addresses for those interrupt service routines should be set in the fixed vector table.  
Because the rewrite operation is halted when a  $\overline{\text{NMI}}$  or watchdog timer interrupt occurs, the rewrite program must be executed again after exiting the interrupt service routine.
- The address match interrupt cannot be used because the flash memory’s internal data is referenced.

#### EW1 Mode

- Make sure that any interrupt which has a vector in the variable vector table or address match interrupt will not be accepted during the auto program or auto erase period.
- Avoid using watchdog timer interrupts.
- The  $\overline{\text{NMI}}$  interrupt can be used because the FMR0 register and FMR1 register are initialized when this interrupt occurs. The jump address for the interrupt service routine should be set in the fixed vector table.  
Because the rewrite operation is halted when a  $\overline{\text{NMI}}$  interrupt occurs, the rewrite program must be executed again after exiting the interrupt service routine.

### 24.15.11 How to access

To set the FMR01, FMR02, or FMR11 bit to “1”, write “0” and then “1” in succession. This is necessary to ensure that no interrupts or DMA transfers will occur before writing “1” after writing “0”. Also only when  $\overline{\text{NMI}}$  pin is “H” level.

### 24.15.12 Writing in the user ROM area

#### EW0 Mode

- If the power supply voltage drops while rewriting any block in which the rewrite control program is stored, a problem may occur that the rewrite control program is not correctly rewritten and, consequently, the flash memory becomes unable to be rewritten thereafter. In this case, standard serial I/O or parallel I/O mode should be used.

#### EW1 Mode

- Avoid rewriting any block in which the rewrite control program is stored.

**24.15.13 DMA transfer**

In EW1 mode, make sure that no DMA transfers will occur while the FMR00 bit in the FMR0 register = 0 (during the auto program or auto erase period).

**24.15.14 Regarding Programming/Erasing Endurance and Execution Time**

As the number of programming/erasure times increases, so does the execution time for software commands (Program, Block Erase, Erase All Unlock Blocks, and Lock Bit Program). Especially when the number of programming/erasure times exceeds 1,000, the software command execution time is noticeably extended. Therefore, the software command wait time that is set must be greater than the maximum rated value of electrical characteristics.

The software commands are suspended by hardware reset 1, hardware reset 2,  $\overline{\text{NMI}}$  interrupt, and watchdog timer interrupt. If a software command is suspended by such reset or interrupt, the block that was in process must be erased before reexecuting the suspended command.

### 24.16 Noise

Connect a bypass capacitor (approximately 0.1  $\mu\text{F}$ ) across the VCC1 and XSS pins, and VCC2 and VSS pins using the shortest and thicker possible wiring. Figure 24.5 shows the Bypass Capacitor Connection.

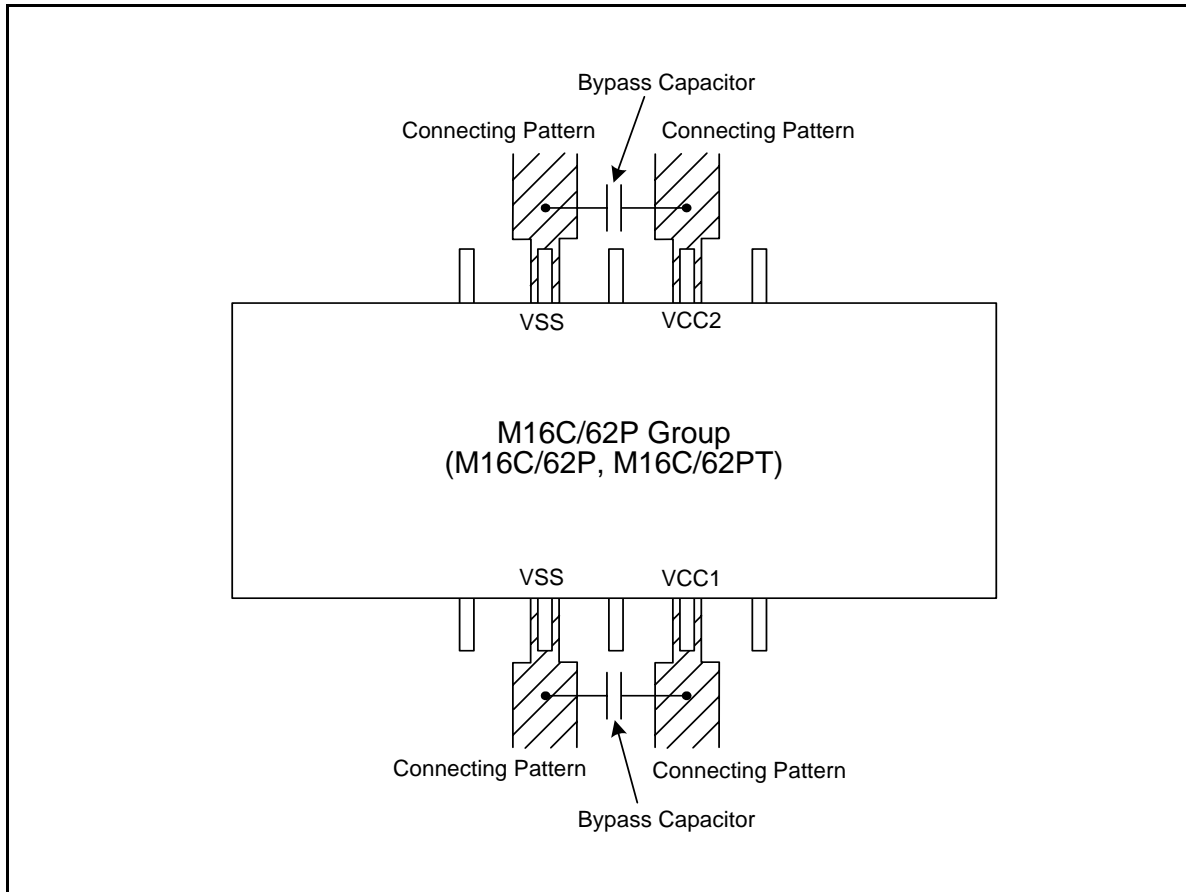


Figure 24.5 Bypass Capacitor Connection

## 25. Differences Depending on Manufacturing Period

Table 25.1 and Table 25.2 list the precautions are applicable or not applicable every chip version of M16C/62P flash and ROM external versions. Contact separately about the mask ROM version.

**Table 25.1 Technical Update Applicable Table of M16C/62P Flash and ROM External Versions (1)**

Precaution	Chip Version			TECHNICAL UPDATE
	A	B	C	
Ensure that RESET must hold valid-low state during power-on. When using a reset IC, use a CMOS type IC. When using an open-drain type reset IC, insert a capacitor between the reset input and VSS. Adjust the R-C time constant between the capacitor and pull-up resistor at least 10 times longer than the VCC rising time.	√	–	–	
If UART0 or UART1 are used as a slave in the I <sup>2</sup> C mode, P6_1 or P6_5 are placed in a high-impedance state. P6_1 or P6_5 cannot be used as an output port even if the PD6_1 or PD6_5 bits in the PD6 register are set to “1” (output mode). Therefore, set the PD6_1 or PD6_5 bits to “0” (Input mode).	√	–	–	TN-M16C-100-0309
Do not enter wait mode when the main clock or on-chip oscillator clock is selected as the CPU clock of which division is set by the CM06 bit in the CM0 register, and the CM16 and CM17 bits in the CM1 register.	√	–	–	TN-M16C-108-0309 Precaution 1.1
The CM05 bit in the CM0 register is set to “0” (main clock oscillation) and the CM02 bit is set to “1” (peripheral function clock stops in wait mode).	√	–	–	TN-M16C-108-0309 Precaution 1.2
Do not generate an NMI interrupt after entering mode.	√	–	–	TN-M16C-108-0309 Precaution 1.3
Do not generate a voltage detection interrupt after entering mode.	√	–	–	TN-M16C-108-0309 Precaution 1.4
I/O ports (P0 to P5) will be indeterminate until internal power supply is stable, such as when the power is turned on, if “H” is applied to the CNVSS pin and “L” to the RESET pin while internal power supply is unstable.	√	√	√	TN-M16C-114-0310 Precaution 1.1
I/O ports (P6 to P14) will be indeterminate until internal power supply is stable, such as when the power is turned on, if “H” is applied to the CNVSS pin and “L” to the RESET pin while internal power supply is unstable.	√	–	–	TN-M16C-114-0310 Precaution 1.1
When the RESET pin is “L” in boot mode (apply “H” to the CNVSS pin and P5_0 (CE), and “L” to the P5_5 (EPM)), internal pull-up is enabled for P10_0 to P10_3, P11_0 to P11_7, P12_5 to P12_7, P13_0 to P13_7, P14_0 and P14_1 and so become “H” level.	√	–	–	TN-M16C-114-0310 Precaution 1.2
P0_0 to P0_7 and P1_0 to P1_7 may become indeterminate when P8_4 is “H” and the RESET pin is “L” in boot mode (apply “H” to the CNVSS pin and P5_0 (CE), and “L” to P5_5 (EPM)). P0_0 to P0_7 and P1_0 to P1_7 are in a high impedance state when the RESET pin and P8_4 are “L”.	√	√	√	TN-M16C-114-0310 Precaution 1.3

√ : Applies

– : Does not apply

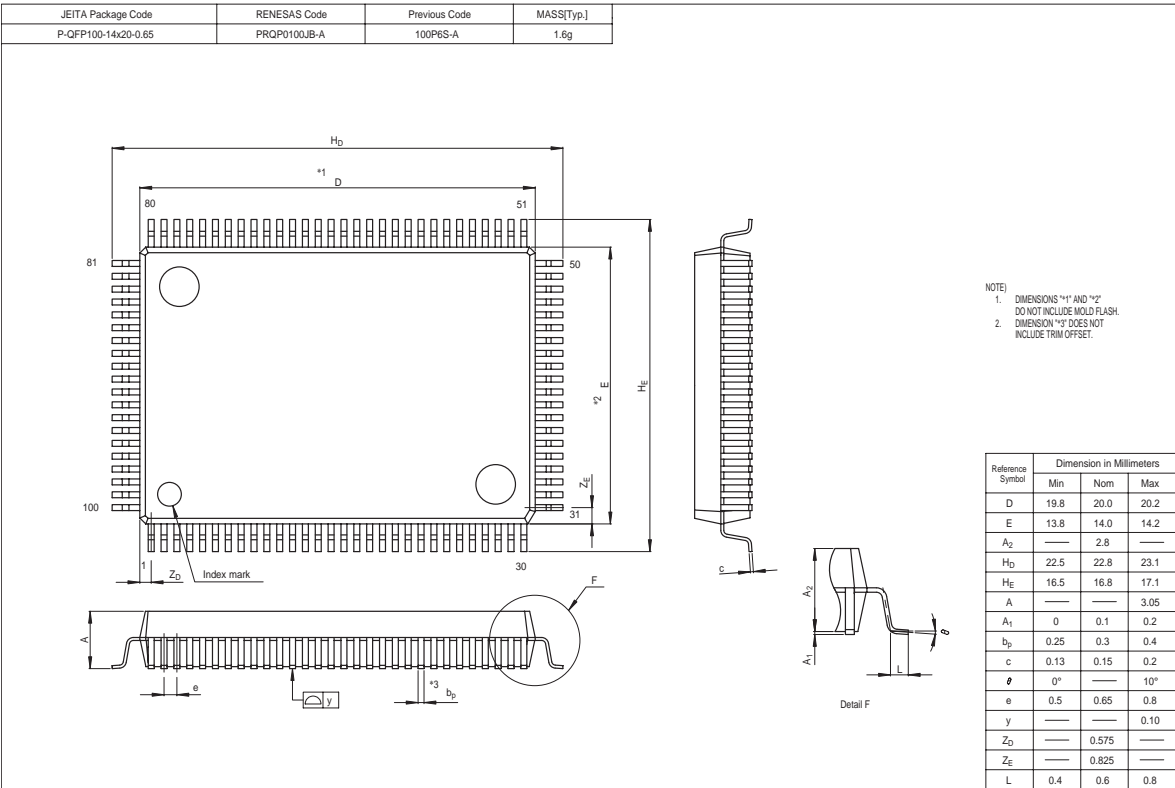
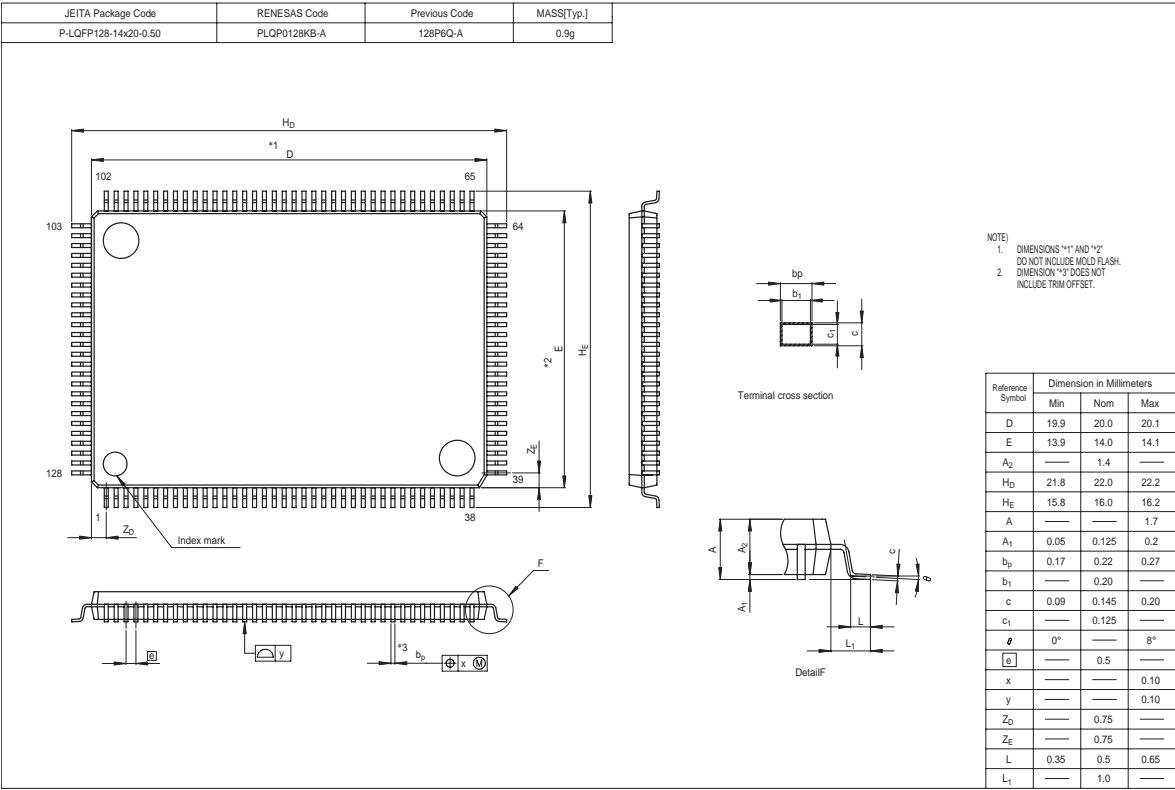
**Table 25.2 Technical Update Applicable Table of M16C/62P Flash and ROM External Versions (2)**

Precaution	Chip Version			TECHNICAL UPDATE
	A	B	C	
When supplying power to the microcomputer, the power supply voltage applied to the VCC1 pin must meet the conditions of SVCC.	÷	-	-	TN-M16C-116-0311
Do not set the CM10 bit in the CM1 register to 1 (stop mode) with setting the VC13 bit in the VCR1 register to 1 ( $VCC1 \geq V_{det} 4$ ) when a low voltage detection interrupt in the voltage detection circuit is used under the following settings: <ul style="list-style-type: none"> <li>the VC27 bit in the VCR2 register to 1 (low voltage detection circuit enabled)</li> <li>the D40 bit in the D4INT register to 1 (low voltage detection interrupt enabled)</li> <li>the D41 bit to 1 (use low voltage detection interrupt to exit stop mode)</li> </ul>	÷	-	-	TN-M16C-107-0309 Precaution 1.1
Do not generate the NMI interrupt after setting the CM10 bit in the CM1 register to "1" (stop mode) and entering stop mode.	÷	-	-	TN-M16C-107-0309 Precaution 1.2
Do not set the CM10 bit in the CM1 register to "1" (stop mode) when the microcomputer is in low-speed mode under the following settings: <ul style="list-style-type: none"> <li>the CM04 bit in the CM0 register is set to "1" (sub clock oscillation)</li> <li>the CM07 bit in the CM0 register is set to "1" (sub clock)</li> </ul>	÷	-	-	TN-M16C-107-0309 Precaution 1.3
When using the sub clock (XCIN-XCOUT) as the CPU clock (BCLK) or as the timer count source, DO NOT leave the CM03 bit set to "1" (XCINXCOUT drive capacity "HIGH").	÷	÷	-	TN-M16C-119A/EA

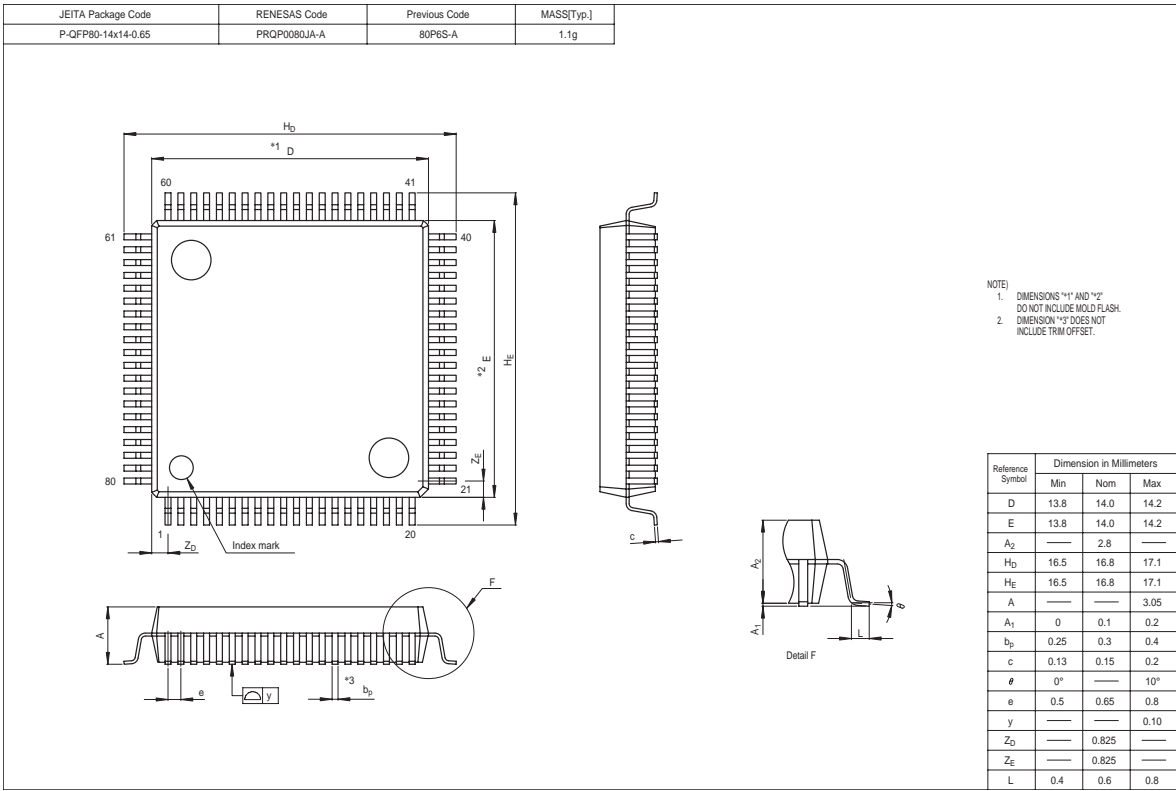
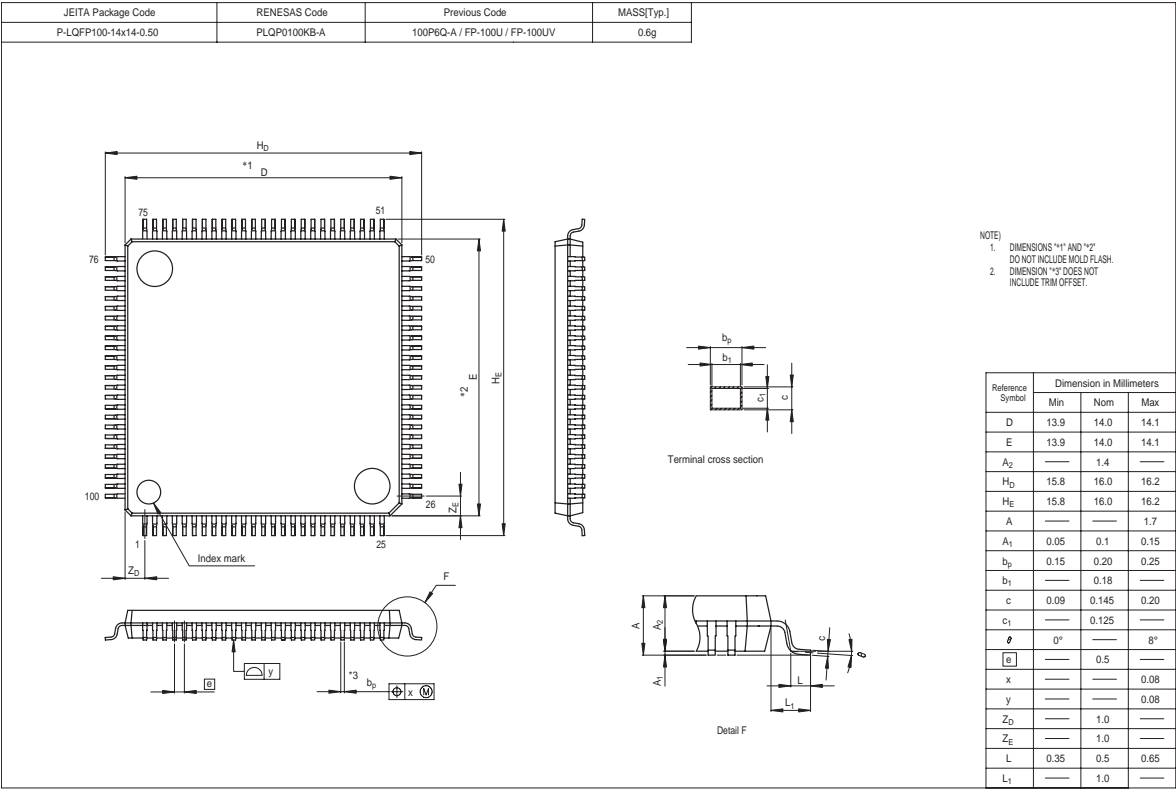
√ : Applies

- : Does not apply

Appendix 1.Package Dimensions







## Appendix 2. Difference between M16C/62P and M16C/30P

Appendix Table 2.1 Function Difference (1)<sup>(1)</sup>

Item	M16C/62P	M16C/62A
Shortest instruction execution time	41.7ns (f(BCLK)=24MHz, VCC1=3.0 to 5.5V) 100ns (f(BCLK)=10MHz, VCC1=2.7 to 5.5V)	62.5ns (f(XIN)=16MHz, VCC=4.2V to 5.5V) 100ns (f(XIN)=10MHz, VCC=2.7V to 5.5V with software one-wait)
Supply voltage	VCC1=3.0 to 5.5V, VCC2=3.0V to VCC1 (f(BCLK)=24MHz) VCC1=VCC2=2.7 to 5.5V (f(BCLK)=10MHz)	VCC=4.2V to 5.5V (f(XIN)=16MHz, without software wait) VCC=2.7V to 5.5V (f(XIN)=10MHz, with software one-wait)
I/O power supply	Double (VCC1, VCC2)	Single (VCC)
Package	80-pin, 100-pin, 128-pin plastic mold QFP	80-pin, 100-pin plastic mold QFP
Voltage detection circuit	Built-in Vdet3, Vdet4 detect Voltage down detect interrupt Hardware reset 2	None
Clock Generating Circuit	PLL, XIN, XCIN, on-chip oscillator When placed in low power mode, a divide-by-8 value is used for these clocks. The XIN drive capability is set to HIGH.	XIN, XCIN When placed in low power mode, the divide-by-n value for the main clock does not change. Nor does the XIN drive capability change.
System clock protective function	Built-in	None (protected by protect register)
Oscillation Stop, Re-oscillation Detection Function	Built-in	None
Low power consumption	18mA (VCC1=VCC2=5V, f(BCLK)=24MHz) 8mA (VCC1=VCC2=3V, f(BCLK)=10MHz) 1.8μA (VCC1=VCC2=3V, f(XCIN)=32kHz, when wait mode)	32.5mA (VCC=5V, f(XIN)=16MHz) 8.5mA (VCC=3V, f(XCIN)=10MHz with software one-wait) 0.9μA (VCC=3V, f(XCIN)=32kHz, when wait mode)
Memory Area	Memory area expandable (4 Mbytes)	1 Mbytes fixed
External Device Connect Area	04000h to 07FFFh (PM13=0) 08000h to 0FFFFh (PM10=0) 10000h to 26FFFh 28000h to 7FFFFh 80000h to CFFFFh (PM13=0) D0000h to FFFFFh (Microprocessor mode)	04000h to 05FFFh (PM13=0) 06000h to CFFFFh D0000h to FFFFFh (Microprocessor mode)
Upper address in memory expansion mode and microprocessor mode	P4_0 to P4_3 (A16 to A19), P3_4 to P3_7 (A12 to A15) : Switchable between address bus and I/O port	P4_0 to P4_3 (A16 to A19) : Switchable between address bus and I/O port A12 to A15 : No switchable
Access to SFR	Variable (1 to 2 waits)	1 wait fixed
Software wait to external area	Variable (0 to 3 waits)	Variable (0 to 1 wait)
Protect	Can be set for PM0, PM1, PM2, CM0, CM1, CM2, PLC0, INVC0, INVC1, PD9, S3C, S4C, TB2SC, PCLKR, VCR2, D4INT registers	Can be set for PM0, PM1, CM0, CM1, PD9, S3C, S4C registers
Watchdog timer	Watchdog timer interrupt or watchdog timer reset is selected Count source protective mode is available	Watchdog timer interrupt No count source protective mode
Address match interrupt	4	2

## NOTES:

- About the details and the electric characteristics, refer to hardware manual.

**Appendix Table 2.2 Function Difference (1)<sup>(1)</sup>**

Item	M16C/62P	M16C/62A
Timers A, B count source	Selectable: f1, f2, f8, f32, fC32	Selectable: f1, f8, f32, fC32
Timer A two-phase pulse signal processing	Function Z-phase (counter reset) input	No function Z-phase (counter reset) input
Timer functions for three-phase motor control	Function protect by protect register Count source is selected: f1, f2, f8, f32, fC32 Dead time timer count source is selected: f1, f1 divided by 2, f2, f2 divided by 2 Three-phase output forcible shutoff function based on NMI input is available, output polarity change, carrier wave phase detection.	Function protect by protect register Count source is selected: f1, f8, f32, fC32 Dead time timer count source is fixed at f1 divided by 2
Serial I/O (UART0 to UART2)	(UART, clock synchronous, I <sup>2</sup> C bus, IEBus) x 3	(UART, clock synchronous) x 2 (UART, clock synchronous, I <sup>2</sup> C bus, IEBus) x 1
UART0 to UART2, SI/O3, SI/O4 count source	Select from f1SIO, f2SIO, f8SIO, f32SIO	Select from f1, f8, f32
Serial I/O $\overline{\text{RTS}}$ timing	Assert low when receive buffer is read	Assert low when reception is completed
UART0 to UART2 Overrun Error Generation Timing	This error occurs if the serial I/O started receiving the next data before reading the UiRB register (i=0 to 2) and received the 7 <sup>th</sup> bit of the next data (clock synchronous) This error occurs if the serial I/O started receiving the next data before reading the UiRB register and received the bit one before the last stop bit of the next data (UART)	This error occurs when the next data is ready before contents of UARTi receive buffer register are read out
$\overline{\text{CTS}}/\overline{\text{RTS}}$ separate function	Have	None
UART2 data transmit timing	After data was written, transfer starts at the 2nd BRG overflow timing (same as UART0 and UART1)	After data was written, transfer starts at the 1st BRG overflow timing (Output starts one cycle of BRG overflow earlier than UART0 and UART1)
Serial I/O sleep function	None	Have
Serial I/O I <sup>2</sup> C mode	Start condition, stop condition: Auto-generationable	Start condition, stop condition: Not auto-generationable
Serial I/O I <sup>2</sup> C mode SDA delay	Only digital delay is selected as SDA delay SDA digital delay count source: BRG	Analog or digital delay is selected as SDA delay SDA digital delay count source: 1/f(XIN)
SI/O3, SI/O4 clock polarity	Selectable	Fixed
A/D Converter	10 bits X 8 channels Expandable up to 26 channels	10 bits X 8 channels Expandable up to 10 channels
A/D converter operation clock	Selectable: fAD, fAD divided by 2, 3, 4, 6, 12	Selectable: fAD, fAD/2, fAD/4
A/D Converter Input Pin	Select from ports P0, P2, P10	Fixed at port P10

**NOTES:**

1. About the details and the electric characteristics, refer to hardware manual.

**Appendix Table 2.3 Function Difference (1)<sup>(1)</sup>**

Item	M16C/62P	M16C/62A
User ROM blocks	14 blocks: 4 Kbytes x 3, 8 Kbytes x 3, 32 Kbytes x 1, 64 Kbytes x 7 (Flash memory: max. 512 Kbytes)	7 blocks: 8 Kbytes x 2, 16 Kbytes x 1, 32 Kbytes x 1, 64 Kbytes x 3 (Flash memory: max. 256 Kbytes)
Program manner	Word	Page
Program command (software command)	Page program command: none Program command: have (program method: in units of word, in units of byte)	Page program command: have Program command: none (program method: in units of page)
Block status after program function	None	Have
CPU rewrite mode	EW1 mode is available	No EW1 mode

**NOTES:**

1. About the details and the electric characteristics, refer to hardware manual.

## Register Index

<b>A</b>		INT0IC to INT5IC ..... 112	TA2 ..... 169
AD0 to AD7 ..... 237		INVC0 ..... 167	TA21 ..... 169
ADCON0 ..... 235		INVC1 ..... 168	TA2MR ..... 173
ADCON1 ..... 235		<b>K</b>	
ADCON2 ..... 236		KUPIC ..... 111	
ADIC ..... 111		<b>O</b>	
AIER ..... 123		ONSF ..... 142	
AIER2 ..... 123		<b>P</b>	
<b>B</b>		P0 to P13 ..... 263	TA4 ..... 169
BCNIC ..... 111		PC14 ..... 264	TA41 ..... 169
<b>C</b>		PCLKR ..... 87	TA4MR ..... 173
CM0 ..... 84		PCR ..... 266	TABSR ..... 141, 158, 172
CM1 ..... 85		PD0 to PD13 ..... 262	TAiMR ..... 140
CM2 ..... 86		PLC0 ..... 88	TB0 to TB5 ..... 157
CPSRF ..... 143, 158		PM0 ..... 56	TB0IC to TB5IC ..... 111
CRCD ..... 253		PM1 ..... 57	TB0MR to TB5MR ..... 157
CRCIN ..... 253		PM2 ..... 87	TB2 ..... 171
CSE ..... 68		PRCR ..... 105	TB2MR ..... 173
CSR ..... 61		PUR0 to PUR1 ..... 265	TB2SC ..... 170
<b>D</b>		PUR2 ..... 266	TBSR ..... 158
D4INT ..... 47		PUR3 ..... 264	TCR0 ..... 131
DA0 to DA1 ..... 252		<b>R</b>	
DACON ..... 252		RMAD0 to RMAD3 ..... 123	
DAR0 ..... 131		ROMCP ..... 274	
DAR1 ..... 131		<b>S</b>	
DBR ..... 73		S0RIC to S2RIC ..... 111	
DM0CON ..... 130		S0TIC to S2TIC ..... 111	
DM0IC to DM1IC ..... 111		S3BRG to S4BRG ..... 229	
DM0SL ..... 128		S3C to S4C ..... 228	
DM1CON ..... 130		S3IC to S4IC ..... 112	
DM1SL ..... 129		S3TRR to S4TRR ..... 229	
DTT ..... 171		SAR0 ..... 131	
<b>F</b>		SAR1 ..... 131	
FIDR ..... 276		<b>T</b>	
FMR0 ..... 277		TA0 to TA4 ..... 140	
FMR1 ..... 278		TA0IC to TA4IC ..... 111	
<b>I</b>		TA0MR to TA4MR ..... 140	
ICTB2 ..... 169		TA1 ..... 169	
IDB0 ..... 170		TA11 ..... 169	
IDB1 ..... 170		TA1MR ..... 173	
IFSR ..... 120		<b>U</b>	
IFSR2A ..... 120		U0BCNIC to U1BCNIC ..... 111	
		U0BRG to U2BRG ..... 182	
		U0C0 to U2C0 ..... 184	
		U0C1 to U2C1 ..... 185	
		U0MR to U2MR ..... 183	
		U0RB to U2RB ..... 181	
		U0SMR to U2SMR ..... 186	
		U0SMR2 to U2SMR2 ..... 187	
		U0SMR3 to U2SMR3 ..... 187	
		U0SMR4 to U2SMR4 ..... 188	
		U0TB to U2TB ..... 181	
		U0CON ..... 186	
		UDF ..... 141	
		<b>V</b>	
		VCR1 ..... 46	
		VCR2 ..... 46	
		<b>W</b>	
		WDC ..... 53, 125	
		WDTS ..... 125	

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
1.0	Jan 31, 2003	1	Applications are partly revised.
		2	Table 1.1.1 is partly revised.
		5	Table 1.1.3 is partly revised.
			Figure 1.1.2 is partly revised.
		11	Explanation of "Memory" is partly revised.
		20	Explanation of "Hardware Reset 1" is partly revised.
		21	Figure 1.5.1 is partly revised.
		22	Figure 1.5.2 is partly revised.
		24	Figure 1.5.4 is partly revised.
		25	VCR2 Register in Figure 1.5.6 is partly revised.
		26	Figure 1.5.6 is partly revised.
		27	Explanation of "Power Supply Down Detection Interrupt" is partly revised.
		30	Figure 1.6.1 is partly revised.
		31	Figure 1.6.2 is partly revised.
		39	Table 1.7.5 is partly revised.
		41	Table 1.7.7 is partly revised.
		43	Figure 1.7.8 is partly revised.
		44	Explanation of "4 Mbyte Mode" is partly revised.
		53	Notes 12 and 13 in Figure 1.9.2 is partly revised.
		54	Notes 2 and 5 in Figure 1.9.3 is partly revised.
		55	Figure 1.9.4 is partly revised.
		57	Note 4 in Figure 1.9.6 is partly revised.
		60	Explanation of "PLL Clock" is partly revised.
		61	Figure 1.9.9 is partly revised.
		62	Explanation of "CPU Clock and BCLK" is partly revised.
		63	Explanation of "Low-speed Mode" is partly revised.
			Explanation of "Low Power Dissipation Mode" is partly revised.
		64	Explanation of "Low Power Dissipation Mode" is partly revised.
			Explanation of "On-chip Oscillator Low Power Dissipation Mode" is partly revised.
			Table 1.9.3 is partly revised.
		65	Table 1.9.5 is partly revised.
		68	Figure 1.9.10 is partly revised.
		69	Figure 1.9.11 is partly revised.
		70	Table 1.9.7 is added.
		71	Explanation of "System Clock Protective Function" is partly revised.
		77	Explanation of "Power Supply Down Detection Interrupt" is partly revised.
		78	Table 1.11.1 is partly revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		88	Figure 1.11.9 is partly revised.
		96	WDTs Register in Figure 1.12.2 is partly revised.
		99	Figure 1.13.2 is partly revised.
		100	Figure 1.13.3 is partly revised.
		103	Figure 1.13.5 is partly revised.
		104	Table 1.13.3 is partly revised.
		105	Explanation of “DMA Enable” is partly revised.
		109	Figure 1.14.3 is partly revised.
		115	Table 1.14.5 is partly revised.
		117	Explanation of “Counter Initialization by Two-Phase Pulse Signal Processing” is partly revised.
		117	Figure 1.14.10 is partly revised.
		122	Figure 1.14.14 is partly revised.
			Figure 1.14.15 is partly revised.
		124	Figure 1.15.3 is partly revised.
		128	Figure 1.15.7 is partly revised.
		128	Figure 1.15.8 is partly revised.
		130	Figure 1.16.1 is partly revised.
		132	Figure 1.16.3 is partly revised.
		134	Note 7 is added to TAI, TAI1 Register in Figure 1.16.5.
		137	Figure 1.16.8 is partly revised.
		146	UiSMR2 Register in Figure 1.17.7 is partly revised.
		163	Figure 1.20.1 is partly revised.
		164, 165	Table 1.20.2 and Table 1.20.3 are partly revised.
		169	Figure 1.20.4 is partly revised.
			Explanation of “Arbitration” is partly revised.
		170	Explanation of “Transfer Clock” is partly revised.
		171	Explanation of “ACK and NACK” is partly revised.
		179	Explanation of “Special Mode 4 (SIM Mode)” is partly revised.
			Table 1.20.9 is partly revised.
		184	Figure 1.21.1 is partly revised.
		187	Figure 1.21.4 is partly revised.
		203	Explanation of “External Operation Amp Connection Mode” is partly revised.
		205	Explanation of “Caution of Using A/D Converter” is partly revised.
			Figure 1.22.11 is partly revised
		206	Table 1.23.1 is partly revised.
		207	Figure 1.23.3 is partly revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		218	Figure 1.25.9 is partly revised.
		223	Table 1.26.1 is partly revised.
		224	Table 1.26.2 is partly revised.
		225	Note 1 of Table 1.26.3 is partly revised.
			Note 1 of Table 1.26.4 is partly revised.
			Table 1.26.6 is partly revised.
		227	Note 1 of Table 1.26.9 is partly revised.
		228	Note 1 of Table 1.26.10 is partly revised.
		229	Measurement conditions of timing requirements are partly revised.
			Table 1.26.11 is partly revised.
		230	Measurement conditions of timing requirements are partly revised.
		230	Table 1.26.18 is added.
		231	Measurement conditions of timing requirements are partly revised.
		232	Measurement conditions of switching characteristics are partly revised.
		233	Measurement conditions of switching characteristics are partly revised.
		234	Measurement conditions of switching characteristics are partly revised.
		235	Figure 1.26.2 is partly revised.
		242	Figure 1.26.9 is partly revised.
		244	Note of Table 1.26.28 is partly revised.
		245	Figure 1.26.29 is partly revised.
		246	Measurement conditions of timing requirements are partly revised.
			Table 1.26.30 is partly revised.
		247	Measurement conditions of timing requirements are partly revised.
		247	Table 1.26.37 is added.
		248	Measurement conditions of timing requirements are partly revised.
		249	Measurement conditions of switching characteristics are partly revised.
		250	Measurement conditions of switching characteristics are partly revised.
		251	Measurement conditions of switching characteristics are partly revised.
		252	Figure 1.26.12 is partly revised.
		255	Figure 1.26.15 is partly revised.
		256	Figure 1.26.16 is partly revised.
		257	Figure 1.26.17 is partly revised.
		258	Figure 1.26.18 is partly revised.
		259	Figure 1.26.19 is partly revised.
		260	Figure 1.26.20 is partly revised.
		262	Explanation of "Memory Map" is partly revised.
		263	Explanation of "Boot Mode" is partly revised.
		264	Figure 1.27.3 is partly revised.
		268	Note of FIDR Register in Figure 1.27.4 is partly revised.



REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		271	Figure 1.27.7 is partly revised.
		272	Explanation of "Interrupts" is partly revised.
			Explanation of "Writing in the User ROM Space" is partly revised.
		274	Table 1.27.4 is partly revised.
			Explanation of "Read Array Command" is partly revised.
		278	Explanation of "Program Command" is partly revised.
		287	Figure 1.27.15 is partly revised.
		293	Partly revised.
1.10	May 28, 2003	2	Table 1.1.1 is partly revised.
		4-5	Table 1.1.2 and 1.1.3 is partly revised.
		14-19	SFR is partly revised.
			Note 1 is partly revised.
		20	Explanation of "Hardware Reset 1" is partly revised.
		23	Note 1 is added.
		24	Figure 1.5.4 is partly revised.
			Note 1 of Figure 1.5.5 is partly revised.
		26	Figure 1.5.7 is partly revised.
		27	Table 1.5.2 is partly revised.
			Table 1.5.3 is partly revised.
			Explanation of "1. Limitations on Stop Mode" is partly revised.
		28	Explanation of "1. Limitations on WAIT instruction" is partly revised.
			Figure 1.5.8 is partly revised.
		31	Note is added.
		33	Explanation of "Multiplexed Bus" is revised.
		34	Explanation of "(2) Data Bus" is revised.
		38	Explanation of "(7) Hold Signal" is revised.
			Note 3 of Table 1.7.4 is added.
		39	Note 4 of Table 1.7.5 is added.
		40	Explanation of "(10) Software Wait" is revised.
		41	Table 1.7.7 is revised.
		46	Table of Figure 1.8.5 is revised.
		47	Explanation is revised.
		48-50	Figures 1.8.7 to 1.8.9 is partly revised.
		51	Explanation of "Clock Generation Circuit" is revised.
		52	Figure 1.9.1 is revised.
		53	Note of Figure 1.9.2 is revised.
		55	Note 12 is added.
		58	Explanation of "(1) Main clock" is partly revised.
		60	Explanation of "(4) PLL Clock" is partly revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		63	Explanation of "Low power Dissipation Mode" is partly revised.
		64	Explanation of "Entering Wait mode" is partly revised.
		66	Explanation of "(3) Stop Mode" is partly revised.
		69	Note 9 is added.
		70	Table 1.9.7 is revised.
		75	Figure 1.11.1 is revised.
		79	Note 6 is added.
		83	Note 2 is added to Figure 1.11.4.
		84	Table 1.11.5 is partly revised.
		85	Figure 1.11.6 is partly revised.
		86	Figure 1.11.8 is partly revised.
		89	Notes 1 to 2 is added to IFSR register of Figure 1.11.4.
		91	Explanation of "Address Match Interrupt" is partly revised. Figure 1. 11.12 is changed into Table 1.11.6.
		93-94	Notes are deleted. (All notes are indicated in "M16C/62 GROUP (M16C/62P) USAGE NOTES").
		93	Explanation of "Watchdog Timer" is partly revised.
		94	A formula is added.
		104	Explanation of "Channel Priority Transfer Timing" is partly revised.
		109	TRGSR register of Figure 1.14.6 is partly revised.
		116	Table 1.14.4 is partly revised.
		117	Figure 1.14.12 is partly revised.
		129	Figure 1.16.2 is partly revised.
		130	Figure 1.16.3 is partly revised.
		143	U0SMR to U2SMR of Figure 1.17.6 is partly revised.
		144	U0SMR2 to U2SMR2 of Figure 1.17.7 is partly revised.
		154,162, 175	"-" of UiBRG of Tables 1.19.2, 1.20.2 and 1.20.8 is changed into "0 to 7".
		161	Figure 1.20.1 is partly revised.
		164	Table 1.20.4 is partly revised. Notes 5 to 7 is added.
		166	Explanation of "Output of Start and Stop Condition" is partly revised.
		177	Note 2 is added to Table 1.20.9.
		178	"-" of U2BRG of Table 1.20.10 is changed into "0 to 7".
		179	Figure 1.20.10 is revised.
		183	Note of SiC register of Figure 1.21.2 is partly revised.
		187	Note 2 of Table 1.22.1 is revised.
		188	Figure 1.22.1 is partly revised.
		190	Table of ADCON2 register of Figure 1.22.3 is partly revised.
		202	The value of a capacitor of Figure 1.22.10 is changed.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		202	Notes are deleted. (All notes are indicated in "M16C/62 GROUP (M16C/62P) USAGE NOTES").
		208-212	Note 1 of Figures 1.25.1 to 1.25.5 is partly revised.
		218	Table 1.25.1 and 1.25.2 is revised.
		219	Figure 1.25.12 is partly revised.
		222	Table 1.26.3 is partly revised.
		223	Table 1.26.5 is partly revised. Table 1.26.6 is added.
		224	Table 1.26.9 is partly revised.
		230	Notes 1 and 2 in Table 1.26.26 is partly revised.
		231	Notes 1 in Table 1.26.27 is partly revised.
		230-231	Note 3 is added to "Data output hold time (refers to BCLK)" in Table 1.26.26 and 1.26.27.
		232	Note 4 is added to "th(ALE-AD)" in Table 1.26.28.
		230-232	Switching Characteristics is partly revised.
		236-239	th(WR-AD) and th(WR-DB) in Figure 1.26.5 to 1.5.8 is partly revised.
		240-241	th(ALE-AD), th(WR-CS), th(WR-DB) and th(WR-AD) in Figure 1.26.9 to 1.5.10 is partly revised.
		242	Note 2 is added to Table 1.26.29.
		247	Notes 1 and 2 in Table 1.26.45 is partly revised.
		248	Notes 1 in Table 1.26.46 is partly revised.
		247-248	Note 3 is added to "Data output hold time(refers to BCLK)" in Table 1.26.45 and 1.26.46
		249	Note 4 is added to "th(ALE-AD)" in Table 1.26.47.
		247-249	Switching Characteristics is partly revised.
		253-256	th(WR-AD) and th(WR-DB) in Figure 1.26.15 to 1.5.18 is partly revised.
		257-258	th(ALE-AD), th(WR-CS), th(WR-DB) and th(WR-AD) in Figure 1.26.19 to 1.5.20 is partly revised.
		259	Table 1.27.1 is partly revised. Notes 3 and 4 is added.
		260	Notes 1 and 2 is added to Table 1.27.2.
		264	Note 2 is added to Table 1.27.3.
		267	Notes 1 and 3 of FMR0 register of Table 1.27.4 is partly revised.
		268	Figure 1.27.5 is partly revised. Note 2 is added.
		270	Figure 1.27.7 is partly revised.
		277	Figure 1.27.11 is partly revised.
		281	Figure 1.27.12 is partly revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		283	Table 1.27.7 is partly revised.
		284-286	Figures 1.27.13 to 1.27.15 is partly revised.
		287-288	Figures 1.27.16 and 1.27.17 is partly revised.
		292-293	Difference in Mask ROM Version and Flash Memory Version is revised.
		294	Difference in Flash Memory Version is revised.
1.11	Jun 20, 2003	259	Number of program and erasure in Table 1.26.27 is partly revised.
1.20	Sep 11, 2003	94	Figure 1.12.2 is revised.
2.30	Sep 01, 2004	–	Since high reliability version is added, a group name is revised. M16C/62P Group (M16C/62P) M16C/62P Group (M16C/62PT)
		2-4	Table 1.1 to 1.3 are revised. Note 3 is partly revised.
		6	Figure 1.2 Note5 is deleted.
		7-9	Table 1.4 to 1.7 Product List is partly revised.
		11	Table 1.8 and Figure 1.4 are added.
		12	Table 1.9 and Figure 1.5 are added.
		13-16	Figure 1.6 to 1.9 ZP is added.
		17	Table 1.10 and 1.13 ZP is added to timer A.
		18, 20	Table 1.11 to 1.13 are revised.
		19, 21	Table 1.12 to 1.14 are revised.
		24	Figure 3.1 is partly revised. Note 3 is added.
		25	Note 6 is added.
		30	After Reset of D/A register 0, 1 are revised.
		31	5.2 Voltage Down Detection Reset (Hardware Reset 2) is revised.
		32	Figure 5.1 is partly revised.
		35	Figure 6.1 is partly revised.
		36	Figure 6.2 is revised.
		37	Figure 6.3 is revised.
		39	Figure 6.4 is revised.
		40	6.2 Limitations on Exiting Stop Mode and 6.3 Limitations on Exiting Wait Mode are revised.
		41	Note in 7. Processor is added.
		44	Figure 7.2 is partly revised.
		46	Note in 8. Bus is added. 8.1.2.2 When the input level on BYTE pin is low (16-bit data bus) is revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		46	Table 8.1 is added.
		47	Note 2 in Figure 8.1 is revised.
		50	Figure 8.4 is revised.
		54	Table 8.8 is partly revised.
			Note 5 is added.
		57	Note in 9 Memory Space Expansion Function is added.
		62-64	Figure 9.7 to 9.9 are revised.
		79	Table 10.4 is partly revised.
		81	Table 10.6 is partly revised.
		82	Figure 10.10 is partly revised.
		83	Note 6 in figure 10.11 is added.
		88	Note in 11. Protect is added.
		89	Note in 12. Interrupt is added.
		108	Note 1 and 2 in figure 13.2 is revised.
		109	13.2 Cold start / Warm start is added.
		120	Note in 15. Timer is added.
		121	Note in 15.1 Timer A is added.
		126	Table 15.1 is partly revised.
		127	Table 15.2 is partly revised.
		131	15.1.2.1 Counter Initialization by Two-Phase Pulse Signal Processing is partly revised.
		137	Note in 15.2 Timer B is added.
		140	Table 15.6 is partly revised.
		144	Note in 16. Three-Phase Motor Control Timer Function is added.
		146-153	Figure 16.2 to 16.9 is revised.
		154	Note in 17. Serial I/O is added.
			Note in 17.1 UART1 is added.
		155-156	Figure 17.1 to 17.3 are revised.
		160	Figure 17.7 is partly revised.
		168	17.1.1.1 Counter Measure for Communication Error Occurs is added.
		169	17.1.1.4 Continuous Receive Mode is revised.
		171	17.1.1.7 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Function is added.
		172	Note 3 in Table 17.5 is added.
		176	17.1.2.1 Bit Rates is added.
		177	17.1.2.2 Counter Measure for Communication Error Occurs is added.
		179	17.1.2.6 $\overline{\text{CTS}}/\overline{\text{RTS}}$ Function is added.
		182	Note 2 in Table 17.11 is revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		192	Note 2 in Table 17.16 is revised.
		195	Note 2 in Table 17.17 is revised.
		197	Note 3 in Table 17.18 is added.
		202	Note in 17.2 SI/O3, SIO4 is added.
		207	Table 18.1 is revised.
		210	Figure 18.3 is partly revised.
		222	18.2.6 Output Impedance of Sensor under A/D Conversion is added.
		228	Note in 21. Programmable I/O Ports is added.
			Table 21.1 is added.
		229	21.3 Pull-up Control Register 0 to Pull-up Control Register 3 (PUR0 to PUR3 Registers) is partly revised.
		235	Note 3 in Figure 21.7 is partly revised.
		236	Note 3 in Figure 21.8 is partly revised.
		237	Note 2 in Figure 21.9 is partly revised.
		240	Note 5 in Table 21.2 is added.
			Note 7 in Table 21.3 is revised.
		242-273	Almost all pages are revised (22. Flash Memory Version).
		274	Table 23.1 is revised.
		275	Table 23.2 is revised.
		276	Table 23.3 is revised.
			Note 2 in Table 23.4 is added.
		277	Table 23.5 to 23.6 is partly revised.
		278	Table 23.8 is revised.
			Table 23.9 is revised.
		279	Table 23.10 is revised.
		280	Table 23.11 is revised.
		281	Table 23.13 is partly revised.
		283	Table 23.24 is partly revised.
		284	Figure 23.2 is partly revised.
			Table 23.26 is partly revised.
		285	Table 23.27 is partly revised.
		286	Table 23.28 is partly revised.
		287	Figure 23.3 is partly revised.
		290-291	Figure 23.6 to 23.7 is partly revised.
		292-293	Figure 23.8 to 23.9 is partly revised.
		295	Figure 23.11 is Figure 23.6 to 23.7 is partly revised.
		296	Table 23.29 is revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		297	Table 23.30 is revised.
		298	Table 23.32 is partly revised.
		300	Table 23.43 is partly revised.
		301	Figure 23.12 is partly revised.
			Table 23.45 is partly revised.
		302	Table 23.46 is partly revised.
		303	Table 23.47 is partly revised.
		304	Figure 23.13 is partly revised.
		307-308	Figure 23.16 to 23.17 is partly revised.
		309-310	Figure 23.18 to 23.19 is partly revised.
		313-339	23.2 Electrical Characteristics (M16C/62PT) is added.
		340	24.1 Reset is added.
		341	24.2 External Bus is partly revised.
		342	Figure 24.2 is added.
		343	24.4 Power Control is partly revised.
		346	24.9.2.1 Special Mode (I2C mode) is added.
		347	24.9.3 SI/O3, SI/O4 is added.
		348-349	24.10 A/D Converter is partly revised.
		352	24.13 Mask ROM Version is added.
		356	24.15 Noise is added.
		357	25. Differences Depending on Manufacturing Period is a
2.40	Dec 15, 2005	-	voltage down detection reset -> brown-out detection Reset
		2-4	Tables 1.1 to 1.3 Performance outline of M16C/62P group are partly revised.
		7	Table 1.4 Product List (1) is partly revised. Note 1 is added.
		8	Table 1.5 Product List (2) is partly revised. Note 1 and 2 are added.
		9	Table 1.6 Product List (3) is partly revised. Note 1 and 2 are added.
		10	Table 1.7 Product List (4) is partly revised. Note 1 and 2 are added.
		11	Figure 1.3 Type No., Memory Size, and Package is partly revised
		12	Table 1.8 Product Code of Flash Memory version and ROMless version for M16C/62P is partly revised.
		13	Table 1.9 Product Code of Flash Memory version for M16C/62P is partly revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		14	Figure 1.6 Pin Configuration (Top View) is partly revised.
		15-17	Tables 1.10 to 1.12 Pin Characteristics for 128-Pin Package are added.
		18-19	Figure 1.7 and 1.8 Pin Configuration (Top View) are partly revised.
		20-21	Tables 1.13 to 1.14 Pin Characteristics for 100-Pin Package are added.
		22	Figure 1.9 Pin Configuration (Top View) is partly revised.
		23-24	Tables 1.15 to 1.16 Pin Characteristics for 80-Pin Package are added.
		25-29	Tables 1.17 to 1.21 are partly revised.
		34	Note 4 of Table 4.1 SFR Information is partly revised.
		40-44	Change sections in Chapter 5.
		42	5.2 Brown-out Detection Reset (Hardware Reset 2) is partly revised.
		45	6. Voltage Detection Circuit is partly revised. Figure 6.1 Voltage Detection Circuit Block is partly revised.
		48	Figure 6.4 Typical Operation of Brown-out Detection Reset (Hardware Reset 2) is partly revised.
		49	Table 6.2 Sampling Periods is partly revised.
		52-53	6.4 Cold Start-up / Warm Start-up Determine Function is added.
		57	Note 7 of Figure 7.2 PM1 Register is partly revised.
		64	8.2.6 $\overline{\text{RDY}}$ Signal is partly revised.
		69	Table 8.8 Bit and Bus Cycle Related to Software
		80	Figure 9.8 Relationship Between Address on 4-Mbyte ROM and Those on Microcomputer (2) is partly revised.
		89	Figure 10.7 Examples of Main Clock Connection Circuit is partly revised.
		90	Figure 10.8 Examples of Sub Clock Connection Circuit is partly revised.
		91	10.1.4 PLL Clock is partly revised.
		94	10.4.1.6 On-chip Oscillator Mode is partly revised.
		95	10.4.1.7 On-chip Oscillator Low Power Dissipation Mode is partly revised.
		96	Table 10.4 Pin Status During Wait Mode is partly revised.
		97	10.4.2.4 Exiting Wait Mode is partly revised.
		98	10.4.3 Stop Mode is partly revised. Table 10.6 Interrupts to Stop Mode and Use Conditions is added.
		99	10.4.3.3 Exiting Stop Mode is partly revised.
		100	Figure 10.11 State Transition in Normal Operating Mode is partly revised.
		104	10.6.3 How to Use Oscillation Stop and Re-oscillation Detect Function is partly revised.
		107	12.2.2 Overflow Interrupt is partly revised.
		118	12.5.8 Returning from an Interrupt Routine is partly revised. 12.5.9 Interrupt Priority is partly revised.
		119	12.5.10 Interrupt Priority Level Select Circuit is partly revised.
		120	Figure 12.11 IFSR and IFSR2A Registers (Interrupt Factor Select Register) is partly revised.



REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		125	13.1 Cold Start / Warm Start moved to 5. Reset.
		127	Table 14.1 DMAC Specifications is partly revised.
		132	14.1.3 Effect of Software Wait is partly revised.
		165	Table 16.1 Three-phase Motor Control Timer Functions Specifications is partly revised.
		167	Notes 5 and 7 of Figure 16.2 INVC0 Register are partly revised.
		177-179	Figures 17.1 to 17.3 UART Block Diagram are partly revised.
		181	Note 3 of Figure 17.5 UiRB Register is added.
		186	Figure 17.10 UCON and UiSMR Registers Notes 3 is revised.
		188	Table 17.1 Clock Synchronous Serial I/O Mode Specifications is partly revised. Note 2 is partly revised.
		192	Figure 17.13 Transmit and Receive Operation is revised.
		193	17.1.1.1 Counter Measure for Communication Error Occurs is partly revised.
		197	Table 17.5 UART Mode Specifications is partly revised. Note 1 is partly revised
		200	Figure 17.19 Transmit Operation is revised.
		201	17.1.2.1 Bit Rate is partly revised. Table 17.9 Example of Bit Rates and Settings is partly revised.
		202	17.1.2.2 Counter Measure for Communication Error Occurs is partly revised.
		205	Table 17.10 I <sup>2</sup> C Mode Specifications is partly revised.
		207	Note 4 of Table 17.11 Registers to Be Used and Settings in I <sup>2</sup> C Mode (1) is added.
		215	Table 17.15 Special Mode Specifications is partly revised.
		222	Table 17.18 SIM Mode Specifications is partly revised.
		224	Figure 17.34 Transmit and Receive Timing in SIM Mode is partly revised.
		226	17.1.6.2 Format is partly revised.
		230	Table 17.20 SI/O3 and SI/O4 Specifications is partly revised.
		231	Figure 17.41 SI/Oi Operation Timing is partly revised. Figure 17.42 Polarity of Transfer Clock is partly revised.
		232	17.2.3 Functions for Settings an SOUTi Internal Value is partly revised.
		249	18.2.6 Output Impedance of Sensor under A/D Conversion is partly revised.
		250	Figure 18.11 Analog Input Pin and External Sensor Equivalent Circuit is partly revised.
		251	Table 19.1 D/A Converter Performance is partly revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		252	Figure 19.2 DA0 and DA1 Register is partly revised. Note 2 of Figure 19.3 D/A Converter is added.
		254	Figure 20.3 CRC Calculation is partly revised.
		261	Note 2 of Figure 21.6 I/O Pin is deleted.
		270	Table 22.1 Flash Memory Version Specifications is partly revised.
		274	Figure 22.2 ROMCP Register is partly revised.
		275	Note 1 of Table 22.3 EM0 Mode and EW1 Mode is partly revised.
		276	22.3.2 EW1 Mode is partly revised.
		279	22.3.3.4 FMSTP Bit is partly revised.
		283	Figure 22.9 Processing Before and After Low Power Dissipation mode is partly revised.
		285	22.3.4.12 Low-Power Consumption Mode and On-chip Oscillator Low-power Consumption Mode is partly revised.
		288	22.3.5.5 Block Erase Command is partly revised. Figure 22.11 Block Erase Command is partly revised.
		292	Table 22.5 Status Register is partly revised.
		294	Figure 22.14 Full Status Check and Handling Procedure for Each Error is partly revised.
		297	Table 22.7 Pin Functions (Flash Memory Standard Serial I/O Mode) is partly revised. Note 2 is partly revised. Note 3 is added.
		297-300	Figures 22.15 to 22.18 are partly revised.
		301	Figure 22.19 Circuit Application in Standard I/O Mode 1 is partly revised.
		302	Figure 22.20 Circuit Application in Standard I/O Mode 2 is partly revised.
		307	Table 23.4 A/D Conversion Characteristics is partly revised.
		309	Table 23.6 Flash Memory Version Electrical Characteristics for 100 cycle products is partly revised. Table 23.7 Flash Memory Version Electrical Characteristics for 10,000 cycle products is partly revised. Table 23.8 Flash Memory Version Program / Erase Voltage and Read Operation Voltage Characteristics is partly revised.
		310	Table 23.9 Low Voltage Detection Circuit Electrical Characteristics is partly revised.
		311	Figure 23.1 Power Supply Circuit Timing Diagram is partly revised.
		313	Table 23.12 Electrical Characteristics (2) is partly revised.
		314	Note 1 of Table 23.13 External Clock Input (XIN input) is added.
		331	Notes 1 to 4 of Table 23.32 External Clock Input (XIN input) are added.
		349	Table 23.53 Flash Memory Version Electrical Characteristics for 100 cycle products is partly revised. Standard (Min.) is partly revised.

REVISION HISTORY	M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual
------------------	--

Rev.	Date	Description	
		Page	Summary
		349	Table 23.54 Flash Memory Version Electrical Characteristics for 10,000 cycle products is partly revised. Standard (Min.) is partly revised. Note 5 is revised.
			Table 23.55 Flash Memory Version Program / Erase Voltage and Read Operation Voltage Characteristics is partly revised.
		352	Table 23.58 Electrical Characteristics is partly revised.
		359	24.1 SFR and 24.1.1 Register Settings, Table 24.1 Registers with Write-only Bits are added.
		360	Figure 24.1 Timing of SVCC is revised.
		363	24.5 Power Control is revised.
		375	Figure 24.4 Use of Capacitors to Reduce Noise is partly revised.
		375	24.11 A/D Converter is partly revised.
		377	24.12 Programmable I/O Ports is partly revised.
		379	24.15.2 Stop mode is partly revised.
		380	24.15.8 Operation speed is partly revised.
		381	24.15.14 Regarding Programming/Erasing Endurance and Execution Time is partly revised. (Title change.)
		383	Table 25.1 Technical Update Applicable of M16C/62P Flash and ROM External Versions is partly revised.
2.41	Jan 10, 2006	-	voltage down detection -> low voltage detection
		99	Figure 10.10 State Transition to Stop Mode and Wait Mode is partly revised.
		100	Figure 10.11 State Transition in Normal Operating Mode is partly revised.
		132	14.1.3 Effect of Software Wait is partly revised.
		186	Figure 17.10 UCON and UiSMR Registers Notes 4 is added. Bit3(LSYN) is added.
		252	Figure 19.3 D/A Converter Equivalent Circuit is partly revised.

---

M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual

Publication Date : Rev.2.41 Jan 10, 2006

Published by : Sales Strategic Planning Div.  
Renesas Technology Corp.

---

© 2006. Renesas Technology Corp., All rights reserved. Printed in Japan

# M16C/62P Group (M16C/62P, M16C/62PT) Hardware Manual



Renesas Technology Corp.  
2-6-2, Ote-machi, Chiyoda-ku, Tokyo, 100-0004, Japan