

16/32

# M32C/80 Group

## Hardware Manual

RENESAS 16/32-BIT SINGLE-CHIP MICROCOMPUTER  
M16C FAMILY / M32C/80 SERIES

Hardware Manual

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

### 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.

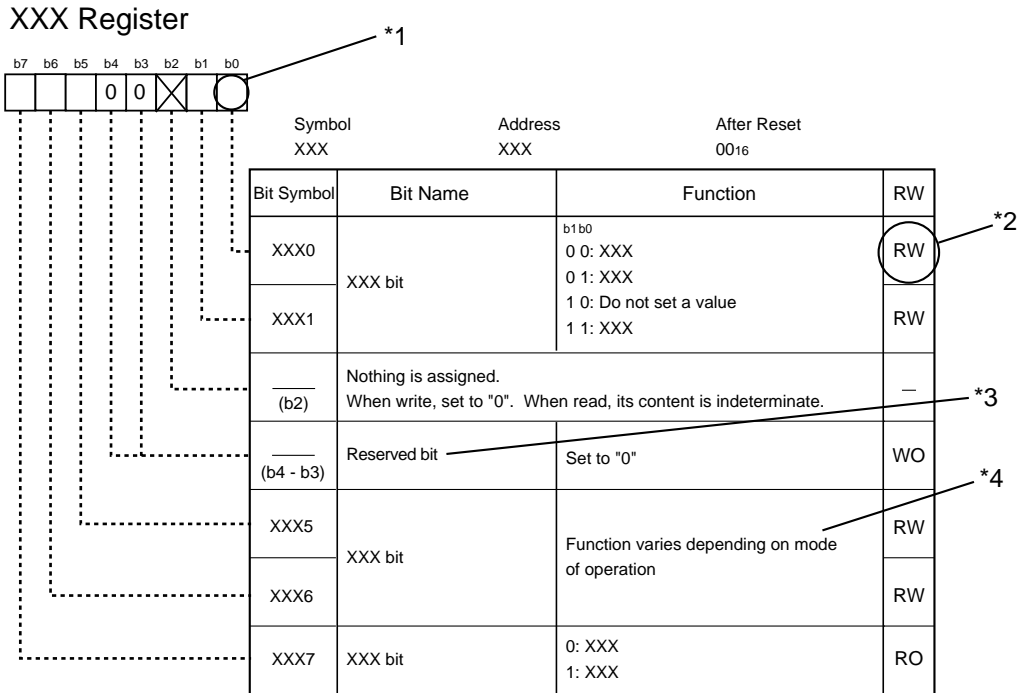
# How to Use This Manual

## 1. Introduction

This hardware manual provides detailed information on the M32C/80 Group 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 a 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 verify that this is the most current document available.

# Table of Contents

<b>Quick Reference by Address</b> .....	<b>B-1</b>
<b>1. Overview</b> .....	<b>1</b>
1.1 Applications .....	1
1.2 Performance Overview .....	2
1.3 Block Diagram .....	3
1.4 Product Information .....	4
1.5 Pin Assignment .....	5
1.6 Pin Description .....	9
<b>2. Central Processing Unit (CPU)</b> .....	<b>12</b>
2.1 General Registers .....	13
2.1.1 Data Registers (R0, R1, R2 and R3).....	13
2.1.2 Address Registers (A0 and A1) .....	13
2.1.3 Static Base Register (SB).....	13
2.1.4 Frame Base Register (FB) .....	13
2.1.5 Program Counter (PC) .....	13
2.1.6 Interrupt Table Register (INTB).....	13
2.1.7 User Stack Pointer (USP), Interrupt Stack Pointer (ISP) .....	13
2.1.8 Flag Register (FLG).....	13
2.2 High-Speed Interrupt Registers .....	14
2.3 DMAC-Associated Registers .....	14
<b>3. Memory</b> .....	<b>15</b>
<b>4. Special Function Registers (SFRs)</b> .....	<b>16</b>
<b>5. Reset</b> .....	<b>30</b>
5.1 Hardware Reset 1 .....	30
5.1.1 Reset on a Stable Supply Voltage .....	30
5.1.2 Power-on Reset .....	30
5.2 Software Reset .....	33
5.3 Watchdog Timer Reset .....	33
5.4 Internal Space .....	33
<b>6. Processor Mode</b> .....	<b>34</b>
6.1 Types of Processor Mode.....	34
6.2 Setting of Processor Mode.....	35

<b>7. Bus</b>	<b>39</b>
7.1 Bus Settings	39
7.1.1 Selecting External Address Bus	40
7.1.2 Selecting External Data Bus	40
7.1.3 Selecting Separate/Multiplexed Bus	40
7.2 Bus Control	42
7.2.1 Address Bus and Data Bus	42
7.2.2 Chip-Select Signal	42
7.2.3 Read and Write Signals	44
7.2.4 Bus Timing	45
7.2.5 ALE Signal	53
7.2.6 $\overline{\text{RDY}}$ Signal	53
7.2.8 External Bus Status when Accessing Internal Space	55
7.2.7 $\overline{\text{HOLD}}$ Signal	55
7.2.9 BCLK Output	55
<b>8. Clock Generation Circuit</b>	<b>56</b>
8.1 Types of the Clock Generation Circuit	56
8.1.1 Main Clock	65
8.1.2 Sub Clock	66
8.1.3 On-Chip Oscillator Clock	67
8.1.4 PLL Clock	69
8.2 CPU Clock and BCLK	70
8.3 Peripheral Function Clock	70
8.3.1 $f_1, f_8, f_{32}$ and $f_{2n}$	70
8.3.2 $f_{AD}$	70
8.3.3 $f_{C32}$	71
8.4 Clock Output Function	71
8.5 Power Consumption Control	72
8.5.1 Normal Operating Mode	72
8.5.2 Wait Mode	73
8.5.3 Stop Mode	75
8.6 System Clock Protect Function	80
<b>9. Protection</b>	<b>81</b>
<b>10. Interrupts</b>	<b>82</b>
10.1 Types of Interrupts	82

10.2	Software Interrupts .....	83
10.2.1	Undefined Instruction Interrupt .....	83
10.2.2	Overflow Interrupt .....	83
10.2.3	BRK Interrupt .....	83
10.2.4	BRK2 Interrupt .....	83
10.2.5	INT Instruction Interrupt .....	83
10.3	Hardware Interrupts .....	84
10.3.1	Special Interrupts .....	84
10.3.2	Peripheral Function Interrupt .....	84
10.4	High-Speed Interrupt .....	85
10.5	Interrupts and Interrupt Vectors .....	85
10.5.1	Fixed Vector Tables .....	86
10.5.2	Relocatable Vector Tables .....	86
10.6	Interrupt Request Acknowledgement .....	89
10.6.1	I Flag and IPL .....	89
10.6.2	Interrupt Control Register and RLVL Register .....	89
10.6.3	Interrupt Sequence .....	93
10.6.4	Interrupt Response Time .....	94
10.6.5	IPL Change when Interrupt Request is Acknowledged .....	95
10.6.6	Saving a Register .....	96
10.6.7	Restoration from Interrupt Routine .....	96
10.6.8	Interrupt Priority .....	97
10.6.9	Interrupt Priority Level Select Circuit .....	97
10.7	$\overline{\text{INT}}$ Interrupt .....	99
10.8	$\overline{\text{NMI}}$ Interrupt .....	100
10.9	Key Input Interrupt .....	100
10.10	Address Match Interrupt .....	101
10.11	Intelligent I/O Interrupt .....	102
<b>11.</b>	<b>Watchdog Timer .....</b>	<b>105</b>
11.1	Count Source Protection Mode .....	108
<b>12.</b>	<b>DMAC .....</b>	<b>109</b>
12.1	Transfer Cycle .....	116
12.1.1	Effect of Source and Destination Addresses .....	116
12.1.2	Effect of the DS Register .....	116
12.1.3	Effect of Software Wait State .....	116
12.1.4	Effect of $\overline{\text{RDY}}$ Signal .....	116
12.2	DMAC Transfer Cycle .....	118
12.3	Channel Priority and DMA Transfer Timing .....	118

<b>13. DMAC II</b>	<b>120</b>
13.1 DMAC II Settings	120
13.1.1 RLVL Register	120
13.1.2 DMAC II Index	122
13.1.3 Interrupt Control Register for the Peripheral Function	124
13.1.4 Relocatable Vector Table for the Peripheral Function	124
13.1.5 IRLT Bit in the IIOiE Register (i=0 to 4)	124
13.2 DMAC II Performance	124
13.3 Transfer Data	124
13.3.1 Memory-to-memory Transfer	124
13.3.2 Immediate Data Transfer	125
13.3.3 Calculation Transfer	125
13.4 Transfer Modes	125
13.4.1 Single Transfer	125
13.4.2 Burst Transfer	125
13.5 Multiple Transfer	125
13.6 Chained Transfer	126
13.7 End-of-Transfer Interrupt	126
13.8 Execution Time	127
<b>14. Timer</b>	<b>128</b>
14.1 Timer A	130
14.1.1 Timer Mode	136
14.1.2 Event Counter Mode	138
14.1.3 One-Shot Timer Mode	142
14.1.4 Pulse Width Modulation Mode	144
14.2 Timer B	147
14.2.1 Timer Mode	150
14.2.2 Event Counter Mode	151
14.2.3 Pulse Period/Pulse Width Measurement Mode	153
<b>15. Three-Phase Motor Control Timer Functions</b>	<b>156</b>
<b>16. Serial I/O</b>	<b>167</b>
16.1 Clock Synchronous Serial I/O Mode	177
16.1.1 Selecting CLK Polarity Selecting	181
16.1.2 Selecting LSB First or MSB First	181
16.1.3 Continuous Receive Mode	182
16.1.4 Serial Data Logic Inverse	182



16.2	Clock Asynchronous Serial I/O (UART) Mode .....	183
16.2.1	Bit Rate .....	187
16.2.2	Selecting LSB First or MSB First.....	188
16.2.3	Serial Data Logic Inverse .....	188
16.2.4	TxD and RxD I/O Polarity Inverse .....	189
16.3	Special Mode 1 (I <sup>2</sup> C Mode) .....	190
16.3.1	Detecting Start Condition and Stop Condition .....	196
16.3.2	Start Condition or Stop Condition Output.....	196
16.3.3	Arbitration .....	198
16.3.4	Transfer Clock .....	198
16.3.5	SDA Output.....	198
16.3.6	SDA Input.....	199
16.3.7	ACK, NACK.....	199
16.3.8	Transmit and Receive Reset .....	199
16.4	Special Mode 2 .....	200
16.4.1	$\overline{SS}_i$ Input Pin Function (i=0 to 4).....	203
16.4.2	Clock Phase Setting Function .....	204
16.5	Special Mode 3 (GCI Mode) .....	206
16.6	Special Mode 4 (IE Mode) .....	210
16.7	Special Mode 5 (SIM Mode) .....	214
16.7.1	Parity Error Signal .....	218
16.7.2	Format .....	219
<b>17.</b>	<b>A/D Converter .....</b>	<b>220</b>
17.1	Mode Description .....	227
17.1.1	One-shot Mode .....	227
17.1.2	Repeat Mode.....	228
17.1.3	Single Sweep Mode .....	229
17.1.4	Repeat Sweep Mode 0 .....	230
17.1.5	Repeat Sweep Mode 1 .....	231
17.2	Functions .....	232
17.2.1	Resolution Select Function.....	232
17.2.2	Sample and Hold Function.....	232
17.2.3	Trigger Select Function .....	232
17.2.4	DMAC Operating Mode .....	232
17.2.5	Extended Analog Input Pins .....	233
17.2.6	External Operating Amplifier (Op-Amp) Connection Mode.....	233
17.2.7	Power Consumption Reducing Function .....	234
17.2.8	Output Impedance of Sensor Equivalent Circuit under A/D Conversion ...	234

<b>18. D/A Converter</b>	<b>236</b>
<b>19. CRC Calculation</b>	<b>239</b>
<b>20. X/Y Conversion</b>	<b>241</b>
<b>21. Intelligent I/O</b>	<b>244</b>
21.1 Communication Unit 0 and 1 Communication Function	246
21.1.1 Clock Synchronous Serial I/O Mode (Communication Units 0 and 1)	256
21.1.2 HDLC Data Processing Mode (Communication Units 0 and 1)	259
<b>22. Programmable I/O Ports</b>	<b>262</b>
22.1 Port Pi Direction Register (PDi Register, i=0 to 10)	262
22.2 Port Pi Register (Pi Register, i=0 to 10)	262
22.3 Function Select Register Aj (PSj Register) (j=0 to 3)	262
22.4 Function Select Register B0 to B3 (PSL0 to PSL3 Registers)	262
22.5 Function Select Register C (PSC and PSC3 Registers)	263
22.6 Function Select Register D (PSD1 Register)	263
22.7 Pull-up Control Register 0 to 3 (PUR0 to PUR3 Registers)	263
22.8 Port Control Register (PCR Register)	263
22.9 Analog Input and Other Peripheral Function Input	263
<b>23. Electrical Characteristics</b>	<b>281</b>
<b>24. Precautions</b>	<b>307</b>
24.1 Reset	307
24.2 Bus	308
24.2.1 $\overline{\text{HOLD}}$ Signal	308
24.3 Special Function Registers (SFRs)	309
24.3.1 Register Settings	309
24.4 Clock Generation Circuit	310
24.4.1 CPU Clock	310
24.4.2 Sub Clock	310
24.4.3 PLL Frequency Synthesizer	311
24.4.4 External Clock	311
24.4.5 Clock Divide Ratio	311
24.4.6 Power Consumption Control	311
24.5 Protection	314
24.6 Interrupts	315
24.6.1 ISP Setting	315
24.6.2 $\overline{\text{NMI}}$ Interrupt	315

24.6.3	$\overline{\text{INT}}$ Interrupt .....	315
24.6.4	Watchdog Timer Interrupt .....	316
24.6.5	Changing Interrupt Control Register .....	316
24.6.6	Changing IIOiR Register (i = 0 to 4) .....	316
24.6.7	Changing RLVL Register .....	316
24.7	DMAC .....	317
24.8	Timer .....	318
24.8.1	Timers A and B .....	318
24.8.2	Timer A .....	318
24.8.3	Timer B .....	320
24.9	Serial I/O .....	321
24.9.1	Clock Synchronous Serial I/O Mode .....	321
24.9.2	UART Mode .....	322
24.9.3	Special Mode 1 (I <sup>2</sup> C Mode) .....	322
24.10	A/D Converter .....	323
24.11	Intelligent I/O .....	325
24.11.1	Register Setting .....	325
24.12	Programmable I/O Ports .....	326
24.13	Noise .....	327
<b>Package Dimensions</b> .....		<b>328</b>
<b>Register Index</b> .....		<b>329</b>

## Quick Reference by Address

Address	Register	Page
0000 <sub>16</sub>		
0001 <sub>16</sub>		
0002 <sub>16</sub>		
0003 <sub>16</sub>		
0004 <sub>16</sub>	Processor Mode Register 0 (PM0)	36
0005 <sub>16</sub>	Processor Mode Register 1 (PM1)	37
0006 <sub>16</sub>	System Clock Control Register 0 (CM0)	58
0007 <sub>16</sub>	System Clock Control Register 1 (CM1)	59
0008 <sub>16</sub>		
0009 <sub>16</sub>	Address Match Interrupt Enable Register (AIER)	101
000A <sub>16</sub>	Protect Register (PRCR)	81
000B <sub>16</sub>	External Data Bus Width Control Register (DS)	39
000C <sub>16</sub>	Main Clock Division Register (MCD)	60
000D <sub>16</sub>	Oscillation Stop Detection Register (CM2)	61
000E <sub>16</sub>	Watchdog Timer Start Register (WDTS)	106
000F <sub>16</sub>	Watchdog Timer Control Register (WDC)	
0010 <sub>16</sub>		
0011 <sub>16</sub>	Address Match Interrupt Register 0 (RMAD0)	101
0012 <sub>16</sub>		
0013 <sub>16</sub>	Processor Mode Register 2 (PM2)	64
0014 <sub>16</sub>		
0015 <sub>16</sub>	Address Match Interrupt Register 1 (RMAD1)	101
0016 <sub>16</sub>		
0017 <sub>16</sub>		
0018 <sub>16</sub>		
0019 <sub>16</sub>	Address Match Interrupt Register 2 (RMAD2)	101
001A <sub>16</sub>		
001B <sub>16</sub>		
001C <sub>16</sub>		
001D <sub>16</sub>	Address Match Interrupt Register 3 (RMAD3)	101
001E <sub>16</sub>		
001F <sub>16</sub>		
0020 <sub>16</sub>		
0021 <sub>16</sub>		
0022 <sub>16</sub>		
0023 <sub>16</sub>		
0024 <sub>16</sub>		
0025 <sub>16</sub>		
0026 <sub>16</sub>	PLL Control Register 0 (PLC0)	68
0027 <sub>16</sub>	PLL Control Register 1 (PLC1)	
0028 <sub>16</sub>		
0029 <sub>16</sub>	Address Match Interrupt Register 4 (RMAD4)	101
002A <sub>16</sub>		
002B <sub>16</sub>		
002C <sub>16</sub>		
002D <sub>16</sub>	Address Match Interrupt Register 5 (RMAD5)	101
002E <sub>16</sub>		
002F <sub>16</sub>		

Address	Register	Page
0030 <sub>16</sub>		
0031 <sub>16</sub>		
0032 <sub>16</sub>		
0033 <sub>16</sub>		
0034 <sub>16</sub>		
0035 <sub>16</sub>		
0036 <sub>16</sub>		
0037 <sub>16</sub>		
0038 <sub>16</sub>		
0039 <sub>16</sub>	Address Match Interrupt Register 6 (RMAD6)	101
003A <sub>16</sub>		
003B <sub>16</sub>		
003C <sub>16</sub>		
003D <sub>16</sub>	Address Match Interrupt Register 7 (RMAD7)	1201
003E <sub>16</sub>		
003F <sub>16</sub>		
0040 <sub>16</sub>		
0041 <sub>16</sub>		
0042 <sub>16</sub>		
0043 <sub>16</sub>		
0044 <sub>16</sub>		
0045 <sub>16</sub>		
0046 <sub>16</sub>		
0047 <sub>16</sub>		
0048 <sub>16</sub>	External Space Wait Control Register 0 (EWCR0)	45
0049 <sub>16</sub>	External Space Wait Control Register 1 (EWCR1)	
004A <sub>16</sub>	External Space Wait Control Register 2 (EWCR2)	
004B <sub>16</sub>	External Space Wait Control Register 3 (EWCR3)	
004C <sub>16</sub>		
004D <sub>16</sub>		
004E <sub>16</sub>		
004F <sub>16</sub>		
0050 <sub>16</sub>		
0051 <sub>16</sub>		
0052 <sub>16</sub>		
0053 <sub>16</sub>		
0054 <sub>16</sub>		
0055 <sub>16</sub>		
0056 <sub>16</sub>		
0057 <sub>16</sub>		
0058 <sub>16</sub>		
0059 <sub>16</sub>		
005A <sub>16</sub>		
005B <sub>16</sub>		
005C <sub>16</sub>		
005D <sub>16</sub>		
005E <sub>16</sub>		
005F <sub>16</sub>		

Blank spaces are reserved. No access is allowed.

## Quick Reference by Address

Address	Register	Page	Address	Register	Page
0060 <sub>16</sub>			0090 <sub>16</sub>	UART0 Transmit /NACK Interrupt Control Register (S0TIC)	
0061 <sub>16</sub>			0091 <sub>16</sub>	UART1 Bus Conflict Detect Interrupt Control Register (BCN1IC)/ UART4 Bus Conflict Detect Interrupt Control Register (BCN4IC)	
0062 <sub>16</sub>			0092 <sub>16</sub>	UART1 Transmit/NACK Interrupt Control Register (S1TIC)	
0063 <sub>16</sub>			0093 <sub>16</sub>	Key Input Interrupt Control Register (KUPIIC)	
0064 <sub>16</sub>			0094 <sub>16</sub>	Timer B0 Interrupt Control Register (TB0IC)	
0065 <sub>16</sub>			0095 <sub>16</sub>	Intelligent I/O Interrupt Control Register 1 (IIO1IC)/ CAN Interrupt 4 Control Register (CAN4IC)	91
0066 <sub>16</sub>			0096 <sub>16</sub>	Timer B2 Interrupt Control Register (TB2IC)	
0067 <sub>16</sub>			0097 <sub>16</sub>	Intelligent I/O Interrupt Control Register 3 (IIO3IC)	
0068 <sub>16</sub>	DMA0 Interrupt Control Register (DM0IC)		0098 <sub>16</sub>	Timer B4 Interrupt Control Register (TB4IC)	
0069 <sub>16</sub>	Timer B5 Interrupt Control Register (TB5IC)		0099 <sub>16</sub>		
006A <sub>16</sub>	DMA2 Interrupt Control Register (DM2IC)		009A <sub>16</sub>	INT4 Interrupt Control Register (INT4IC)	91
006B <sub>16</sub>	UART2 Receive /ACK Interrupt Control Register (S2RIC)		009B <sub>16</sub>		
006C <sub>16</sub>	Timer A0 Interrupt Control Register (TA0IC)		009C <sub>16</sub>	INT2 Interrupt Control Register (INT2IC)	91
006D <sub>16</sub>	UART3 Receive /ACK Interrupt Control Register (S3RIC)		009D <sub>16</sub>		
006E <sub>16</sub>	Timer A2 Interrupt Control Register (TA2IC)		009E <sub>16</sub>	INT0 Interrupt Control Register (INT0IC)	91
006F <sub>16</sub>	UART4 Receive /ACK Interrupt Control Register (S4RIC)		009F <sub>16</sub>	Exit Priority Control Register (RLVL)	92
0070 <sub>16</sub>	Timer A4 Interrupt Control Register (TA4IC)		00A0 <sub>16</sub>	Interrupt Request Register 0 (IIO0IR)	
0071 <sub>16</sub>	UART0 Bus Conflict Detect Interrupt Control Register (BCN0IC)/ UART3 Bus Conflict Detect Interrupt Control Register (BCN3IC)	90	00A1 <sub>16</sub>	Interrupt Request Register 1 (IIO1IR)	
0072 <sub>16</sub>	UART0 Receive/ACK Interrupt Control Register (S0RIC)		00A2 <sub>16</sub>	Interrupt Request Register 2 (IIO2IR)	
0073 <sub>16</sub>	A/D0 Conversion Interrupt Control Register (AD0IC)		00A3 <sub>16</sub>	Interrupt Request Register 3 (IIO3IR)	
0074 <sub>16</sub>	UART1 Receive/ACK Interrupt Control Register (S1RIC)		00A4 <sub>16</sub>	Interrupt Request Register 4 (IIO4IR)	
0075 <sub>16</sub>	Intelligent I/O Interrupt Control Register 0 (IIO0IC)/ CAN Interrupt 3 Control Register (CAN3IC)		00A5 <sub>16</sub>	Interrupt Request Register 5 (IIO5IR)	
0076 <sub>16</sub>	Timer B1 Interrupt Control Register (TB1IC)		00A6 <sub>16</sub>		
0077 <sub>16</sub>	Intelligent I/O Interrupt Control Register 2 (IIO2IC)		00A7 <sub>16</sub>		
0078 <sub>16</sub>	Timer B3 Interrupt Control Register (TB3IC)		00A8 <sub>16</sub>		
0079 <sub>16</sub>	Intelligent I/O Interrupt Control Register 4 (IIO4IC)		00A9 <sub>16</sub>		
007A <sub>16</sub>	INT5 Interrupt Control Register (INT5IC)	91	00AA <sub>16</sub>		
007B <sub>16</sub>			00AB <sub>16</sub>		
007C <sub>16</sub>	INT3 Interrupt Control Register (INT3IC)	91	00AC <sub>16</sub>		
007D <sub>16</sub>			00AD <sub>16</sub>		
007E <sub>16</sub>	INT1 Interrupt Control Register (INT1IC)	91	00AE <sub>16</sub>		
007F <sub>16</sub>			00AF <sub>16</sub>		
0080 <sub>16</sub>			00B0 <sub>16</sub>	Interrupt Enable Register 0 (IIO0IE)	
0081 <sub>16</sub>			00B1 <sub>16</sub>	Interrupt Enable Register 1 (IIO1IE)	
0082 <sub>16</sub>			00B2 <sub>16</sub>	Interrupt Enable Register 2 (IIO2IE)	
0083 <sub>16</sub>			00B3 <sub>16</sub>	Interrupt Enable Register 3 (IIO3IE)	
0084 <sub>16</sub>			00B4 <sub>16</sub>	Interrupt Enable Register 4 (IIO4IE)	
0085 <sub>16</sub>			00B5 <sub>16</sub>		
0086 <sub>16</sub>			00B6 <sub>16</sub>		
0087 <sub>16</sub>			00B7 <sub>16</sub>		
0088 <sub>16</sub>	DMA1 Interrupt Control Register (DM1IC)		00B8 <sub>16</sub>		
0089 <sub>16</sub>	UART2 Transmit /NACK Interrupt Control Register (S2TIC)		00B9 <sub>16</sub>		
008A <sub>16</sub>	DMA3 Interrupt Control Register (DM3IC)		00BA <sub>16</sub>		
008B <sub>16</sub>	UART3 Transmit /NACK Interrupt Control Register (S3TIC)		00BB <sub>16</sub>		
008C <sub>16</sub>	Timer A1 Interrupt Control Register (TA1IC)	90	00BC <sub>16</sub>		
008D <sub>16</sub>	UART4 Transmit /NACK Interrupt Control Register (S4TIC)		00BD <sub>16</sub>		
008E <sub>16</sub>	Timer A3 Interrupt Control Register (TA3IC)		00BE <sub>16</sub>		
008F <sub>16</sub>	UART2 Bus Conflict Detect Interrupt Control Register (BCN2IC)		00BF <sub>16</sub>		

Blank spaces are reserved. No access is allowed.

## Quick Reference by Address

Address	Register	Page	Address	Register	Page
00C0 <sub>16</sub>			00F0 <sub>16</sub>	Data Compare Register 00 (G0CMP0)	254
00C1 <sub>16</sub>			00F1 <sub>16</sub>	Data Compare Register 01 (G0CMP1)	
00C2 <sub>16</sub>			00F2 <sub>16</sub>	Data Compare Register 02 (G0CMP2)	
00C3 <sub>16</sub>			00F3 <sub>16</sub>	Data Compare Register 03 (G0CMP3)	
00C4 <sub>16</sub>			00F4 <sub>16</sub>	Data Mask Register 00 (G0MSK0)	
00C5 <sub>16</sub>			00F5 <sub>16</sub>	Data Mask Register 01 (G0MSK1)	255
00C6 <sub>16</sub>			00F6 <sub>16</sub>	Communication Clock Select Register (CCS)	
00C7 <sub>16</sub>			00F7 <sub>16</sub>		254
00C8 <sub>16</sub>			00F8 <sub>16</sub>	Receive CRC Code Register 0 (G0RCRC)	
00C9 <sub>16</sub>			00F9 <sub>16</sub>		
00CA <sub>16</sub>			00FA <sub>16</sub>	Transmit CRC Code Register 0 (G0TCRC)	249
00CB <sub>16</sub>			00FB <sub>16</sub>		
00CC <sub>16</sub>			00FC <sub>16</sub>	SI/O Extended Mode Register 0 (G0EMR)	251
00CD <sub>16</sub>			00FD <sub>16</sub>	SI/O Extended Receive Control Register 0 (G0ERC)	252
00CE <sub>16</sub>			00FE <sub>16</sub>	SI/O Special Communication Interrupt Detect Register 0 (G0IRF)	250
00CF <sub>16</sub>			00FF <sub>16</sub>	SI/O Extended Transmit Control Register 0 (G0ETC)	
00D0 <sub>16</sub>			0100 <sub>16</sub>		
00D1 <sub>16</sub>			0101 <sub>16</sub>		
00D2 <sub>16</sub>			0102 <sub>16</sub>		
00D3 <sub>16</sub>			0103 <sub>16</sub>		
00D4 <sub>16</sub>			0104 <sub>16</sub>		
00D5 <sub>16</sub>			0105 <sub>16</sub>		
00D6 <sub>16</sub>			0106 <sub>16</sub>		
00D7 <sub>16</sub>			0107 <sub>16</sub>		
00D8 <sub>16</sub>			0108 <sub>16</sub>		
00D9 <sub>16</sub>			0109 <sub>16</sub>		
00DA <sub>16</sub>			010A <sub>16</sub>		
00DB <sub>16</sub>			010B <sub>16</sub>		
00DC <sub>16</sub>			010C <sub>16</sub>		
00DD <sub>16</sub>			010D <sub>16</sub>		
00DE <sub>16</sub>			010E <sub>16</sub>		
00DF <sub>16</sub>			010F <sub>16</sub>		
00E0 <sub>16</sub>			0110 <sub>16</sub>		
00E1 <sub>16</sub>			0111 <sub>16</sub>		
00E2 <sub>16</sub>			0112 <sub>16</sub>		
00E3 <sub>16</sub>			0113 <sub>16</sub>		
00E4 <sub>16</sub>			0114 <sub>16</sub>		
00E5 <sub>16</sub>			0115 <sub>16</sub>		
00E6 <sub>16</sub>			0116 <sub>16</sub>		
00E7 <sub>16</sub>			0117 <sub>16</sub>		
00E8 <sub>16</sub>			0118 <sub>16</sub>		
00E9 <sub>16</sub>	SI/O Receive Buffer Register0 (G0RB)	247	0119 <sub>16</sub>		
00EA <sub>16</sub>	Transmit Buffer/Receive Data Register 0 (G0TB/G0DR)	253	011A <sub>16</sub>		
00EB <sub>16</sub>			011B <sub>16</sub>		
00EC <sub>16</sub>	Receive Input Register 0 (G0RI)	246	011C <sub>16</sub>		
00ED <sub>16</sub>	SI/O Communication Mode Register 0 (G0MR)	248	011D <sub>16</sub>		
00EE <sub>16</sub>	Transmit Output Register 0 (G0TO)	246	011E <sub>16</sub>		
00EF <sub>16</sub>	SI/O Communication Control Register 0 (G0CR)	247	011F <sub>16</sub>		

Blank spaces are reserved. No access is allowed.

## Quick Reference by Address

Address	Register	Page	Address	Register	Page
0120 <sub>16</sub>			0150 <sub>16</sub>		
0121 <sub>16</sub>			0151 <sub>16</sub>		
0122 <sub>16</sub>			0152 <sub>16</sub>		
0123 <sub>16</sub>			0153 <sub>16</sub>		
0124 <sub>16</sub>			0154 <sub>16</sub>		
0125 <sub>16</sub>			0155 <sub>16</sub>		
0126 <sub>16</sub>			0156 <sub>16</sub>		
0127 <sub>16</sub>			0157 <sub>16</sub>		
0128 <sub>16</sub>			0158 <sub>16</sub>		
0129 <sub>16</sub>	SI/O Receive Buffer Register 1 (G1RB)	247	0159 <sub>16</sub>		
012A <sub>16</sub>	Transmit Buffer/Receive Data Register 1 (G1TB/G1DR)	253	015A <sub>16</sub>		
012B <sub>16</sub>			015B <sub>16</sub>		
012C <sub>16</sub>	Receive Input Register 1 (G1RI)	246	015C <sub>16</sub>		
012D <sub>16</sub>	SI/O Communication Mode Register 1 (G1MR)	248	015D <sub>16</sub>		
012E <sub>16</sub>	Transmit Output Register 1 (G1TO)	246	015E <sub>16</sub>		
012F <sub>16</sub>	SI/O Communication Control Register 1 (G1CR)	247	015F <sub>16</sub>		
0130 <sub>16</sub>	Data Compare Register 10 (G1CMP0)		0160 <sub>16</sub>		
0131 <sub>16</sub>	Data Compare Register 11 (G1CMP1)		0161 <sub>16</sub>		
0132 <sub>16</sub>	Data Compare Register 12 (G1CMP2)	254	0162 <sub>16</sub>		
0133 <sub>16</sub>	Data Compare Register 13 (G1CMP3)		0163 <sub>16</sub>		
0134 <sub>16</sub>	Data Mask Register 10 (G1MSK0)		0164 <sub>16</sub>		
0135 <sub>16</sub>	Data Mask Register 11 (G1MSK1)		0165 <sub>16</sub>		
0136 <sub>16</sub>			0166 <sub>16</sub>		
0137 <sub>16</sub>			0167 <sub>16</sub>		
0138 <sub>16</sub>			0168 <sub>16</sub>		
0139 <sub>16</sub>	Receive CRC Code Register1 (G1RCRC)	254	0169 <sub>16</sub>		
013A <sub>16</sub>			016A <sub>16</sub>		
013B <sub>16</sub>	Transmit CRC Code Register1 (G1TCRC)		016B <sub>16</sub>		
013C <sub>16</sub>	SI/O Extended Mode Register 1 (G1EMR)	249	016C <sub>16</sub>		
013D <sub>16</sub>	SI/O Extended Receive Control Register 1 (G1ERC)	251	016D <sub>16</sub>		
013E <sub>16</sub>	SI/O Special Communication Interrupt Detect Register 1 (G1IRF)	253	016E <sub>16</sub>		
013F <sub>16</sub>	SI/O Extended Transmit Control Register 1 (G1ETC)	250	016F <sub>16</sub>		
0140 <sub>16</sub>			0170 <sub>16</sub>		
0141 <sub>16</sub>			0171 <sub>16</sub>		
0142 <sub>16</sub>			0172 <sub>16</sub>		
0143 <sub>16</sub>			0173 <sub>16</sub>		
0144 <sub>16</sub>			0174 <sub>16</sub>		
0145 <sub>16</sub>			0175 <sub>16</sub>		
0146 <sub>16</sub>			0176 <sub>16</sub>		
0147 <sub>16</sub>			0177 <sub>16</sub>		
0148 <sub>16</sub>			0178 <sub>16</sub>		
0149 <sub>16</sub>			0179 <sub>16</sub>		
014A <sub>16</sub>			017A <sub>16</sub>		
014B <sub>16</sub>			017B <sub>16</sub>		
014C <sub>16</sub>			017C <sub>16</sub>		
014D <sub>16</sub>			017D <sub>16</sub>		
014E <sub>16</sub>			to		
014F <sub>16</sub>			02AF <sub>16</sub>		

Blank spaces are reserved. No access is allowed.

## Quick Reference by Address

Address	Register	Page
02B0 <sub>16</sub>		
02B1 <sub>16</sub>		
02B2 <sub>16</sub>		
02B3 <sub>16</sub>		
02B4 <sub>16</sub>		
02B5 <sub>16</sub>		
02B6 <sub>16</sub>		
02B7 <sub>16</sub>		
02B8 <sub>16</sub>		
02B9 <sub>16</sub>		
02BA <sub>16</sub>		
02BB <sub>16</sub>		
02BC <sub>16</sub>		
02BD <sub>16</sub>		
02BE <sub>16</sub>		
02BF <sub>16</sub>		
02C0 <sub>16</sub>	X0 Register Y0 Register (X0R,Y0R)	241
02C1 <sub>16</sub>		
02C2 <sub>16</sub>	X1 Register Y1 Register (X1R,Y1R)	
02C3 <sub>16</sub>		
02C4 <sub>16</sub>	X2 Register Y2 Register (X2R,Y2R)	
02C5 <sub>16</sub>		
02C6 <sub>16</sub>	X3 Register Y3 Register (X3R,Y3R)	
02C7 <sub>16</sub>		
02C8 <sub>16</sub>	X4 Register Y4 Register (X4R,Y4R)	
02C9 <sub>16</sub>		
02CA <sub>16</sub>	X5 Register Y5 Register (X5R,Y5R)	
02CB <sub>16</sub>		
02CC <sub>16</sub>	X6 Register Y6 Register (X6R,Y6R)	
02CD <sub>16</sub>		
02CE <sub>16</sub>	X7 Register Y7 Register (X7R,Y7R)	
02CF <sub>16</sub>		
02D0 <sub>16</sub>	X8 Register Y8 Register (X8R,Y8R)	
02D1 <sub>16</sub>		
02D2 <sub>16</sub>	X9 Register Y9 Register (X9R,Y9R)	
02D3 <sub>16</sub>		
02D4 <sub>16</sub>	X10 Register Y10 Register (X10R,Y10R)	
02D5 <sub>16</sub>		
02D6 <sub>16</sub>	X11 Register Y11 Register (X11R,Y11R)	
02D7 <sub>16</sub>		
02D8 <sub>16</sub>	X12 Register Y12 Register (X12R,Y12R)	
02D9 <sub>16</sub>		
02DA <sub>16</sub>	X13 Register Y13 Register (X13R,Y13R)	
02DB <sub>16</sub>		
02DC <sub>16</sub>	X14 Register Y14 Register (X14R,Y14R)	
02DD <sub>16</sub>		
02DE <sub>16</sub>	X15 Register Y15 Register (X15R,Y15R)	
02DF <sub>16</sub>		

Address	Register	Page
02E0 <sub>16</sub>	X/Y Control Register (XYC)	241
02E1 <sub>16</sub>		
02E2 <sub>16</sub>		
02E3 <sub>16</sub>		
02E4 <sub>16</sub>	UART1 Special Mode Register 4 (U1SMR4)	175
02E5 <sub>16</sub>	UART1 Special Mode Register 3 (U1SMR3)	174
02E6 <sub>16</sub>	UART1 Special Mode Register 2 (U1SMR2)	173
02E7 <sub>16</sub>	UART1 Special Mode Register (U1SMR)	172
02E8 <sub>16</sub>	UART1 Transmit/Receive Mode Register (U1MR)	170
02E9 <sub>16</sub>	UART1 Bit Rate Register (U1BRG)	
02EA <sub>16</sub>	UART1 Transmit Buffer Register (U1TB)	169
02EB <sub>16</sub>		
02EC <sub>16</sub>	UART1 Transmit/Receive Control Register 0 (U1C0)	171
02ED <sub>16</sub>	UART1 Transmit/Receive Control Register 1 (U1C1)	172
02EE <sub>16</sub>	UART1 Receive Buffer Register (U1RB)	169
02EF <sub>16</sub>		
02F0 <sub>16</sub>		
02F1 <sub>16</sub>		
02F2 <sub>16</sub>		
02F3 <sub>16</sub>		
02F4 <sub>16</sub>	UART4 Special Mode Register 4 (U4SMR4)	175
02F5 <sub>16</sub>	UART4 Special Mode Register 3 (U4SMR3)	174
02F6 <sub>16</sub>	UART4 Special Mode Register 2 (U4SMR2)	173
02F7 <sub>16</sub>	UART4 Special Mode Register (U4SMR)	172
02F8 <sub>16</sub>	UART4 Transmit/Receive Mode Register (U4MR)	170
02F9 <sub>16</sub>	UART4 Bit Rate Register (U4BRG)	
02FA <sub>16</sub>	UART4 Transmit Buffer Register (U4TB)	169
02FB <sub>16</sub>		
02FC <sub>16</sub>	UART4 Transmit/Receive Control Register 0 (U4C0)	171
02FD <sub>16</sub>	UART4 Transmit/Receive Control Register 1 (U4C1)	172
02FE <sub>16</sub>	UART4 Receive Buffer Register (U4RB)	169
02FF <sub>16</sub>		
0300 <sub>16</sub>	Timer B3,B4,B5 Count Start Flag (TBSR)	149
0301 <sub>16</sub>		
0302 <sub>16</sub>	Timer A1-1 Register (TA11)	
0303 <sub>16</sub>		
0304 <sub>16</sub>	Timer A2-1 Register (TA21)	
0305 <sub>16</sub>		
0306 <sub>16</sub>	Timer A4-1 Register (TA41)	162
0307 <sub>16</sub>		
0308 <sub>16</sub>	Three-Phase PWM Control Register 0 (INVC0)	159
0309 <sub>16</sub>	Three-Phase PWM Control Register 1 (INVC1)	160
030A <sub>16</sub>	Three-Phase Output Buffer Register 0 (IDB0)	161
030B <sub>16</sub>	Three-Phase Output Buffer Register 1 (IDB1)	
030C <sub>16</sub>	Dead Time Timer (DTT)	161
030D <sub>16</sub>	Timer B2 Interrupt Generating Frequency Set Counter (ICTB2)	162
030E <sub>16</sub>		
030F <sub>16</sub>		

Blank spaces are reserved. No access is allowed.



## Quick Reference by Address

Address	Register	Page	Address	Register	Page
0310 <sub>16</sub>	Timer B3 Register (TB3)	147	0340 <sub>16</sub>	Count Start Flag (TABSR)	132
0311 <sub>16</sub>			0341 <sub>16</sub>	Clock Prescaler Reset Flag (CPSRF)	62
0312 <sub>16</sub>			Timer B4 Register (TB4)	0342 <sub>16</sub>	One-Shot Start Flag (ONSF)
0313 <sub>16</sub>	0343 <sub>16</sub>	Trigger Select Register (TRGSR)		134	
0314 <sub>16</sub>	Timer B5 Register (TB5)	147	0344 <sub>16</sub>	Up/Down Flag (UDF)	133
0315 <sub>16</sub>			0345 <sub>16</sub>		
0316 <sub>16</sub>		148	0346 <sub>16</sub>	Timer A0 Register (TA0)	131
0317 <sub>16</sub>			0347 <sub>16</sub>		
0318 <sub>16</sub>			0348 <sub>16</sub>	Timer A1 Register (TA1)	
0319 <sub>16</sub>			0349 <sub>16</sub>		
031A <sub>16</sub>			034A <sub>16</sub>	Timer A2 Register (TA2)	
031B <sub>16</sub>	Timer B3 Mode Register (TB3MR)		034B <sub>16</sub>		
031C <sub>16</sub>	Timer B4 Mode Register (TB4MR)	148	034C <sub>16</sub>	Timer A3 Register (TA3)	
031D <sub>16</sub>	Timer B5 Mode Register (TB5MR)		034D <sub>16</sub>		
031E <sub>16</sub>			034E <sub>16</sub>	Timer A4 Register (TA4)	
031F <sub>16</sub>	External Interrupt Request Source Select Register (IFSR)	99	034F <sub>16</sub>		
0320 <sub>16</sub>		175	0350 <sub>16</sub>	Timer B0 Register (TB0)	147
0321 <sub>16</sub>			0351 <sub>16</sub>		
0322 <sub>16</sub>			0352 <sub>16</sub>	Timer B1 Register (TB1)	
0323 <sub>16</sub>			0353 <sub>16</sub>		
0324 <sub>16</sub>	UART3 Special Mode Register 4 (U3SMR4)	175	0354 <sub>16</sub>	Timer B2 Register (TB2)	
0325 <sub>16</sub>	UART3 Special Mode Register 3 (U3SMR3)	174	0355 <sub>16</sub>		
0326 <sub>16</sub>	UART3 Special Mode Register 2 (U3SMR2)	173	0356 <sub>16</sub>	Timer A0 Mode Register (TA0MR)	132
0327 <sub>16</sub>	UART3 Special Mode Register (U3SMR)	172	0357 <sub>16</sub>	Timer A1 Mode Register (TA1MR)	
0328 <sub>16</sub>	UART3 Transmit/Receive Mode Register (U3MR)	170	0358 <sub>16</sub>	Timer A2 Mode Register (TA2MR)	
0329 <sub>16</sub>	UART3 Bit Rate Register (U3BRG)		0359 <sub>16</sub>	Timer A3 Mode Register (TA3MR)	
032A <sub>16</sub>	UART3 Transmit Buffer Register (U3TB)	169	035A <sub>16</sub>	Timer A4 Mode Register (TA4MR)	
032B <sub>16</sub>			035B <sub>16</sub>	Timer B0 Mode Register (TB0MR)	
032C <sub>16</sub>	UART3 Transmit/Receive Control Register 0 (U3C0)	171	035C <sub>16</sub>	Timer B1 Mode Register (TB1MR)	148
032D <sub>16</sub>	UART3 Transmit/Receive Control Register 1 (U3C1)	172	035D <sub>16</sub>	Timer B2 Mode Register (TB2MR)	
032E <sub>16</sub>	UART3 Receive Buffer Register (U3RB)	169	035E <sub>16</sub>	Timer B2 Special Mode Register (TB2SC)	162
032F <sub>16</sub>			035F <sub>16</sub>	Count Source Prescaler Register (TCSPR)	62
0330 <sub>16</sub>		175	0360 <sub>16</sub>		175
0331 <sub>16</sub>			0361 <sub>16</sub>		
0332 <sub>16</sub>			0362 <sub>16</sub>		
0333 <sub>16</sub>			0363 <sub>16</sub>		
0334 <sub>16</sub>	UART2 Special Mode Register 4 (U2SMR4)	175	0364 <sub>16</sub>	UART0 Special Mode Register 4 (U0SMR4)	175
0335 <sub>16</sub>	UART2 Special Mode Register 3 (U2SMR3)	174	0365 <sub>16</sub>	UART0 Special Mode Register 3 (U0SMR3)	174
0336 <sub>16</sub>	UART2 Special Mode Register 2 (U2SMR2)	173	0366 <sub>16</sub>	UART0 Special Mode Register 2 (U0SMR2)	173
0337 <sub>16</sub>	UART2 Special Mode Register (U2SMR)	172	0367 <sub>16</sub>	UART0 Special Mode Register (U0SMR)	172
0338 <sub>16</sub>	UART2 Transmit/Receive Mode Register (U2MR)	170	0368 <sub>16</sub>	UART0 Transmit/Receive Mode Register (U0MR)	170
0339 <sub>16</sub>	UART2 Bit Rate Register (U2BRG)		0369 <sub>16</sub>	UART0 Bit Rate Register (U0BRG)	
033A <sub>16</sub>	UART2 Transmit Buffer Register (U2TB)	169	036A <sub>16</sub>	UART0 Transmit Buffer Register (U0TB)	169
033B <sub>16</sub>			036B <sub>16</sub>		
033C <sub>16</sub>	UART2 Transmit/Receive Control Register 0 (U2C0)	171	036C <sub>16</sub>	UART0 Transmit/Receive Control Register 0 (U0C0)	171
033D <sub>16</sub>	UART2 Transmit/Receive Control Register 1 (U2C1)	172	036D <sub>16</sub>	UART0 Transmit/Receive Control Register 1 (U0C1)	172
033E <sub>16</sub>	UART2 Receive Buffer Register (U2RB)	169	036E <sub>16</sub>	UART0 Receive Buffer Register (U0RB)	169
033F <sub>16</sub>			036F <sub>16</sub>		

Blank spaces are reserved. No access is allowed.

## Quick Reference by Address

Address	Register	Page
0370 <sub>16</sub>		
0371 <sub>16</sub>		
0372 <sub>16</sub>		
0373 <sub>16</sub>		
0374 <sub>16</sub>		
0375 <sub>16</sub>		
0376 <sub>16</sub>		
0377 <sub>16</sub>		
0378 <sub>16</sub>	DMA0 Request Source Select Register (DM0SL)	111
0379 <sub>16</sub>	DMA1 Request Source Select Register (DM1SL)	
037A <sub>16</sub>	DMA2 Request Source Select Register (DM2SL)	
037B <sub>16</sub>	DMA3 Request Source Select Register (DM3SL)	
037C <sub>16</sub>	CRC Data Register (CRCD)	239
037D <sub>16</sub>		
037E <sub>16</sub>	CRC Input Register (CRCIN)	
037F <sub>16</sub>		
0380 <sub>16</sub>	A/D0 Register0 (AD00)	
0381 <sub>16</sub>		
0382 <sub>16</sub>	A/D0 Register1 (AD01)	
0383 <sub>16</sub>		
0384 <sub>16</sub>	A/D0 Register2 (AD02)	
0385 <sub>16</sub>		
0386 <sub>16</sub>	A/D0 Register3 (AD03)	226
0387 <sub>16</sub>		
0388 <sub>16</sub>	A/D0 Register4 (AD04)	
0389 <sub>16</sub>		
038A <sub>16</sub>	A/D0 Register5 (AD05)	
038B <sub>16</sub>		
038C <sub>16</sub>	A/D0 Register6 (AD06)	
038D <sub>16</sub>		
038E <sub>16</sub>	A/D0 Register7 (AD07)	
038F <sub>16</sub>		
0390 <sub>16</sub>		
0391 <sub>16</sub>		
0392 <sub>16</sub>		
0393 <sub>16</sub>		
0394 <sub>16</sub>	A/D0 Control Register 2 (AD0CON2)	224
0395 <sub>16</sub>	A/D0 Control Register 3 (AD0CON3)	225
0396 <sub>16</sub>	A/D0 Control Register 0 (AD0CON0)	222
0397 <sub>16</sub>	A/D0 Control Register 1 (AD0CON1)	223
0398 <sub>16</sub>	D/A Register 0 (DA0)	238
0399 <sub>16</sub>		
039A <sub>16</sub>	D/A Register 1 (DA1)	238
039B <sub>16</sub>		
039C <sub>16</sub>	D/A Control Register (DACON)	238
039D <sub>16</sub>		
039E <sub>16</sub>		
039F <sub>16</sub>		

Address	Register	Page
03A0 <sub>16</sub>		
03A1 <sub>16</sub>		
03A2 <sub>16</sub>		
03A3 <sub>16</sub>		
03A4 <sub>16</sub>		
03A5 <sub>16</sub>		
03A6 <sub>16</sub>		
03A7 <sub>16</sub>	Function Select Register D1 (PSD1)	274
03A8 <sub>16</sub>		
03A9 <sub>16</sub>		
03AA <sub>16</sub>		
03AB <sub>16</sub>		
03AC <sub>16</sub>		
03AD <sub>16</sub>	Function Select Register C3 (PSC3)	274
03AE <sub>16</sub>		
03AF <sub>16</sub>	Function Select Register C (PSC)	273
03B0 <sub>16</sub>	Function Select Register A0 (PS0)	269
03B1 <sub>16</sub>	Function Select Register A1 (PS1)	
03B2 <sub>16</sub>	Function Select Register B0 (PSL0)	271
03B3 <sub>16</sub>	Function Select Register B1 (PSL1)	
03B4 <sub>16</sub>	Function Select Register A2 (PS2)	270
03B5 <sub>16</sub>	Function Select Register A3 (PS3)	
03B6 <sub>16</sub>	Function Select Register B2 (PSL2)	272
03B7 <sub>16</sub>	Function Select Register B3 (PSL3)	
03B8 <sub>16</sub>		
03B9 <sub>16</sub>		
03BA <sub>16</sub>		
03BB <sub>16</sub>		
03BC <sub>16</sub>		
03BD <sub>16</sub>		
03BE <sub>16</sub>		
03BF <sub>16</sub>		
03C0 <sub>16</sub>	Port P6 Register (P6)	268
03C1 <sub>16</sub>	Port P7 Register (P7)	
03C2 <sub>16</sub>	Port P6 Direction Register (PD6)	267
03C3 <sub>16</sub>	Port P7 Direction Register (PD7)	
03C4 <sub>16</sub>	Port P8 Register (P8)	268
03C5 <sub>16</sub>	Port P9 Register (P9)	
03C6 <sub>16</sub>	Port P8 Direction Register (PD8)	267
03C7 <sub>16</sub>	Port P9 Direction Register (PD9)	
03C8 <sub>16</sub>	Port P10 Register (P10)	268
03C9 <sub>16</sub>		
03CA <sub>16</sub>	Port P10 Direction Register (PD10)	267
03CB <sub>16</sub>		
03CC <sub>16</sub>		
03CD <sub>16</sub>		
03CE <sub>16</sub>		
03CF <sub>16</sub>		

Blank spaces are reserved. No access is allowed.

## Quick Reference by Address

Address	Register	Page
03D0 <sub>16</sub>		
03D1 <sub>16</sub>		
03D2 <sub>16</sub>		
03D3 <sub>16</sub>		
03D4 <sub>16</sub>		
03D5 <sub>16</sub>		
03D6 <sub>16</sub>		
03D7 <sub>16</sub>		
03D8 <sub>16</sub>		
03D9 <sub>16</sub>		
03DA <sub>16</sub>	Pull-Up Control Register 2 (PUR2)	275
03DB <sub>16</sub>	Pull-Up Control Register 3 (PUR3)	276
03DC <sub>16</sub>		
03DD <sub>16</sub>		
03DE <sub>16</sub>		
03DF <sub>16</sub>		
03E0 <sub>16</sub>	Port P14 Register (P0)	268
03E1 <sub>16</sub>	Port P14 Register (P1)	
03E2 <sub>16</sub>	Port P14 Direction Register (PD0)	267
03E3 <sub>16</sub>	Port P14 Direction Register (PD1)	
03E4 <sub>16</sub>	Port P14 Register (P2)	268
03E5 <sub>16</sub>	Port P14 Register (P3)	
03E6 <sub>16</sub>	Port P14 Direction Register (PD2)	267
03E7 <sub>16</sub>	Port P14 Direction Register (PD3)	
03E8 <sub>16</sub>	Port P14 Register (P4)	268
03E9 <sub>16</sub>	Port P14 Register (P5)	
03EA <sub>16</sub>	Port P14 Direction Register (PD4)	267
03EB <sub>16</sub>	Port P14 Direction Register (PD5)	
03EC <sub>16</sub>		
03ED <sub>16</sub>		
03EE <sub>16</sub>		
03EF <sub>16</sub>		
03F0 <sub>16</sub>	Pull-up Control Register 0 (PUR0)	275
03F1 <sub>16</sub>	Pull-up Control Register 1 (PUR1)	
03F2 <sub>16</sub>		
03F3 <sub>16</sub>		
03F4 <sub>16</sub>		
03F5 <sub>16</sub>		
03F6 <sub>16</sub>		
03F7 <sub>16</sub>		
03F8 <sub>16</sub>		
03F9 <sub>16</sub>		
03FA <sub>16</sub>		
03FB <sub>16</sub>		
03FC <sub>16</sub>		
03FD <sub>16</sub>		
03FE <sub>16</sub>		
03FF <sub>16</sub>	Port Control Register (PCR)	276

Blank spaces are reserved. No access is allowed.

---

# M32C/80 Group

## SINGLE-CHIP 16/32-BIT CMOS MICROCOMPUTER

---

## 1. Overview

The M32C/80 Group microcomputer is a single-chip control unit that utilizes high-performance silicon gate CMOS technology with the M32C/80 series CPU core. The M32C/80 Group is available in 100-pin plastic molded LQFP/QFP package.

With a 16-Mbyte address space, this microcomputer combines advanced instruction manipulation capabilities to process complex instructions by less bytes and execute instructions at higher speed.

It incorporates a multiplier and DMAC adequate for office automation, communication devices and industrial equipments and other high-speed processing applications.

The M32C/80 Group is ROMless device.  
Use the M32C/80 Group in microprocessor mode after reset.

### 1.1 Applications

Audio, cameras, office equipment, communications equipment, portable equipment, etc.

## 1.2 Performance Overview

Table 1.1 lists performance overview of the M32C/80 Group.

**Table 1.1 M32C/80 Group Performance**

Item		Performance
CPU	Basic Instructions	108 instructions
	Minimum Instruction Execution Time	31.3 ns ( f(BCLK)=32 MHz, Vcc1=4.2 to 5.5 V ) 41.7 ns ( f(BCLK)=24 MHz, Vcc1=3.0 to 5.5 V )
	Operating Mode	Single-chip mode, Memory expansion mode, Microprocessor mode
	Memory Space	16 Mbytes
	Memory Capacity	See Table 1.2
Peripheral function	I/O Port	47 I/O pins (when using 16-bit bus) and 1 input pin
	Multifunction Timer	Timer A: 16 bits x 5 channels, Timer B: 16 bits x 6 channels Three-phase motor control circuit
	Intelligent I/O Communication Function	2 channels
	Serial I/O	5 channels Clock synchronous serial I/O, Clock asynchronous serial I/O, IEBus <sup>(1)</sup> , I <sup>2</sup> C Bus <sup>(2)</sup>
	A/D Converter	10-bit A/D converter: 1 circuit, 10 channels
	D/A Converter	8 bits x 2 channels
	DMAC	4 channels
	DMAC II	Can be activated by all peripheral function interrupt sources Immediate transfer, operation and chain transfer function
	CRC Calculation Circuit	CRC-CCITT
	X/Y Converter	16 bits x 16 bits
	Watchdog Timer	15 bits x 1 channel (with prescaler)
	Interrupt	34 internal sources and 8 external sources, 5 software sources Interrupt priority level: 7
	Clock Generation Circuit	4 circuits Main Clock oscillation circuit (*), Sub clock oscillation circuit (*), On-chip oscillator, PLL frequency synthesizer (* )Equipped with a built-in feedback resistor
Oscillation Stop Detect Function	Main clock oscillation stop detect circuit	
Electrical Characteristics	Supply Voltage	Vcc1=4.2 to 5.5 V, Vcc2=3.0 to Vcc1 (f(BCLK)=32 MHz) Vcc1=3.0 to 5.5 V, Vcc2=3.0 to Vcc1 (f(BCLK)=24 MHz)
	Power Consumption	22 mA (Vcc1=Vcc2=5 V, f(BCLK)=32 MHz) 17 mA (Vcc1=Vcc2=3.3 V, f(BCLK)=24 MHz) 10 μA (Vcc1=Vcc2=3.3 V, f(BCLK)=32 kHz, in wait mode)
Operating Ambient Temperature		-20 to 85°C, -40 to 85°C(optional)
Package		100-pin plastic molded LQFP/QFP

NOTES:

1. IEBus is a trademark of NEC Electronics Corporation.
  2. I<sup>2</sup>C bus is a trademark of Koninklijke Philips Electronics N. V.
- All options are on a request basis.

### 1.3 Block Diagram

Figure 1.1 shows a block diagram of the M32C/80 Group microcomputer.

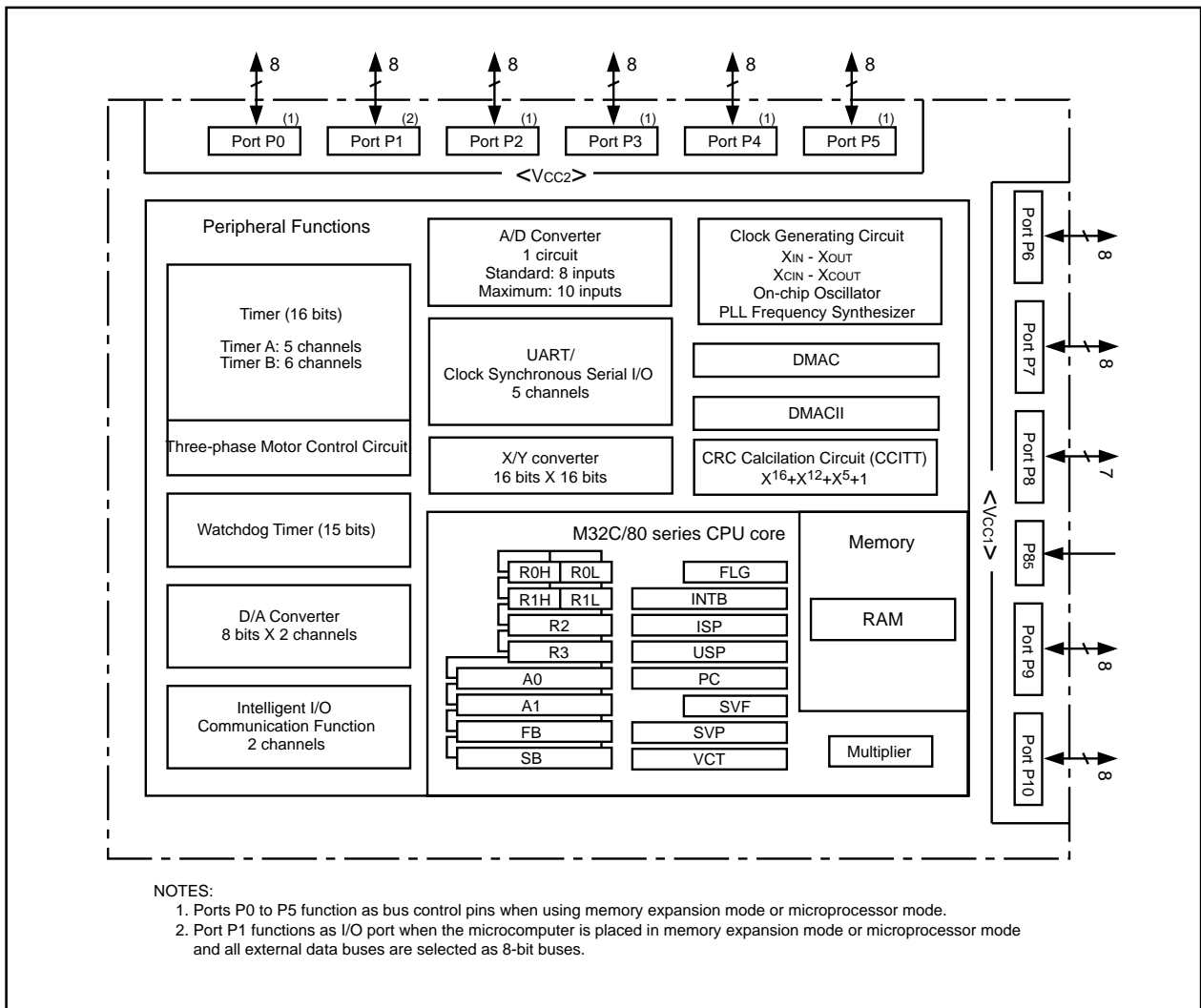


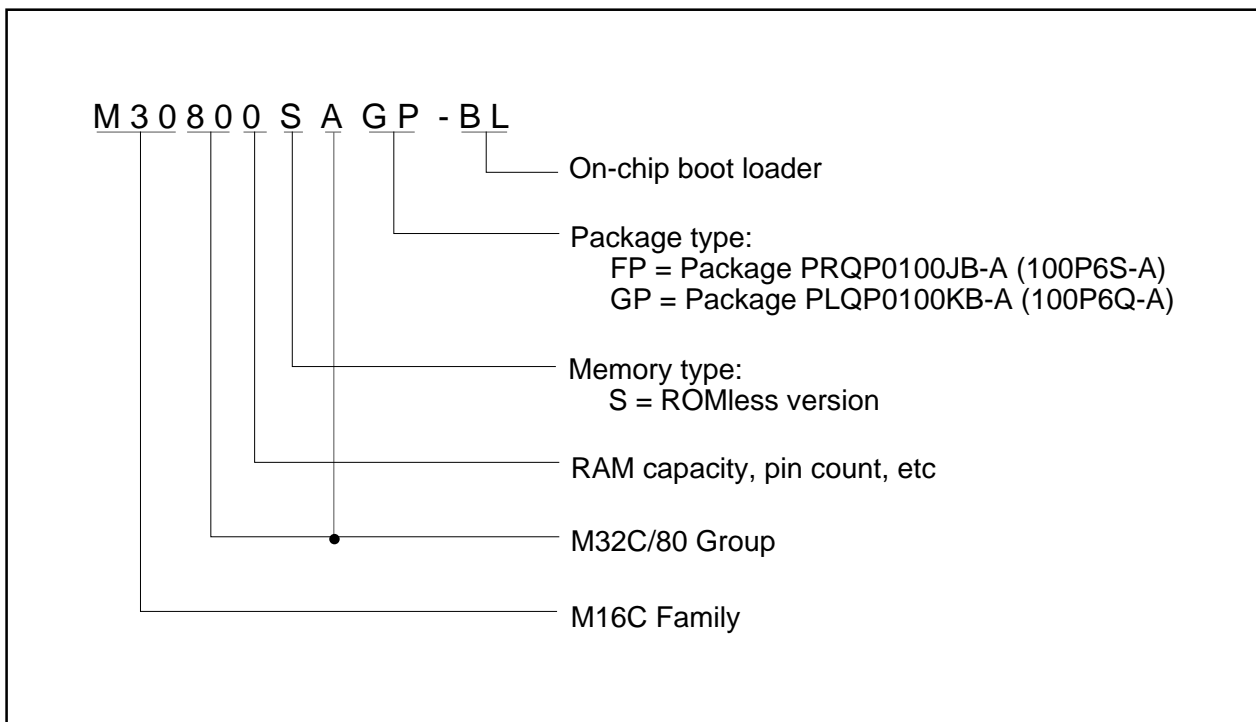
Figure 1.1 M32C/80 Group Block Diagram

### 1.4 Product Information

Table 1.2 lists the product information. Figure 1.2 shows the product numbering system.

**Table 1.2 M32C/80 Group** **As of November, 2005**

Type Number	Package Type	ROM Capacity	RAM Capacity	Remarks
M30800SAGP	PLQP0100KB-A (100P6Q-A)	—	8K	ROMless
M30800SAFP	PRQP0100JB-A (100P6S-A)			
M30800SAGP-BL	PLQP0100KB-A (100P6Q-A)			ROMless with on-chip boot loader
M30800SAFP-BL	PRQP0100JB-A (100P6S-A)			



**Figure 1.2 Product Numbering System**

### 1.5 Pin Assignment

Figures 1.3 and 1.4 show pin assignments (top view).

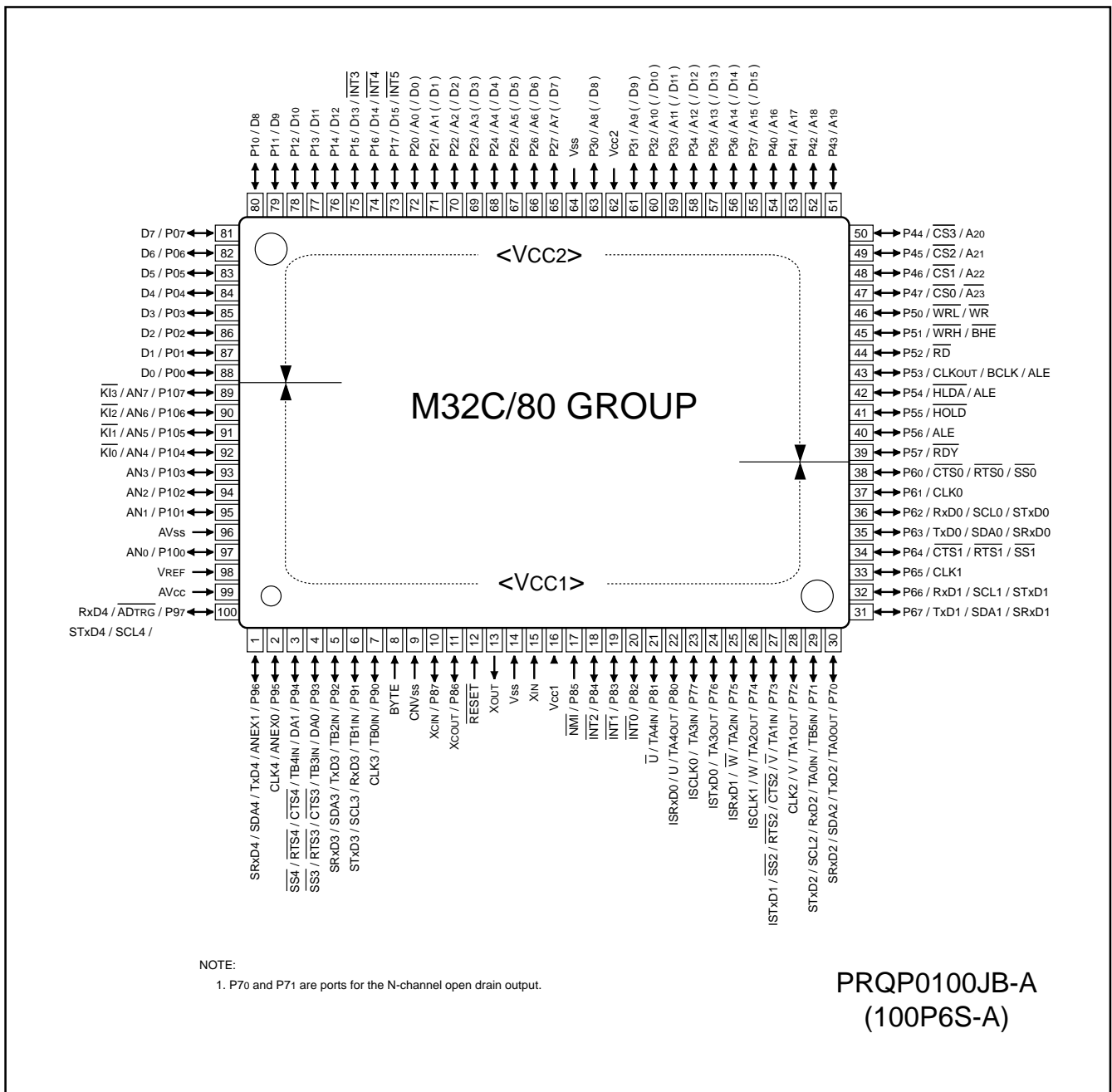


Figure 1.3 Pin Assignment



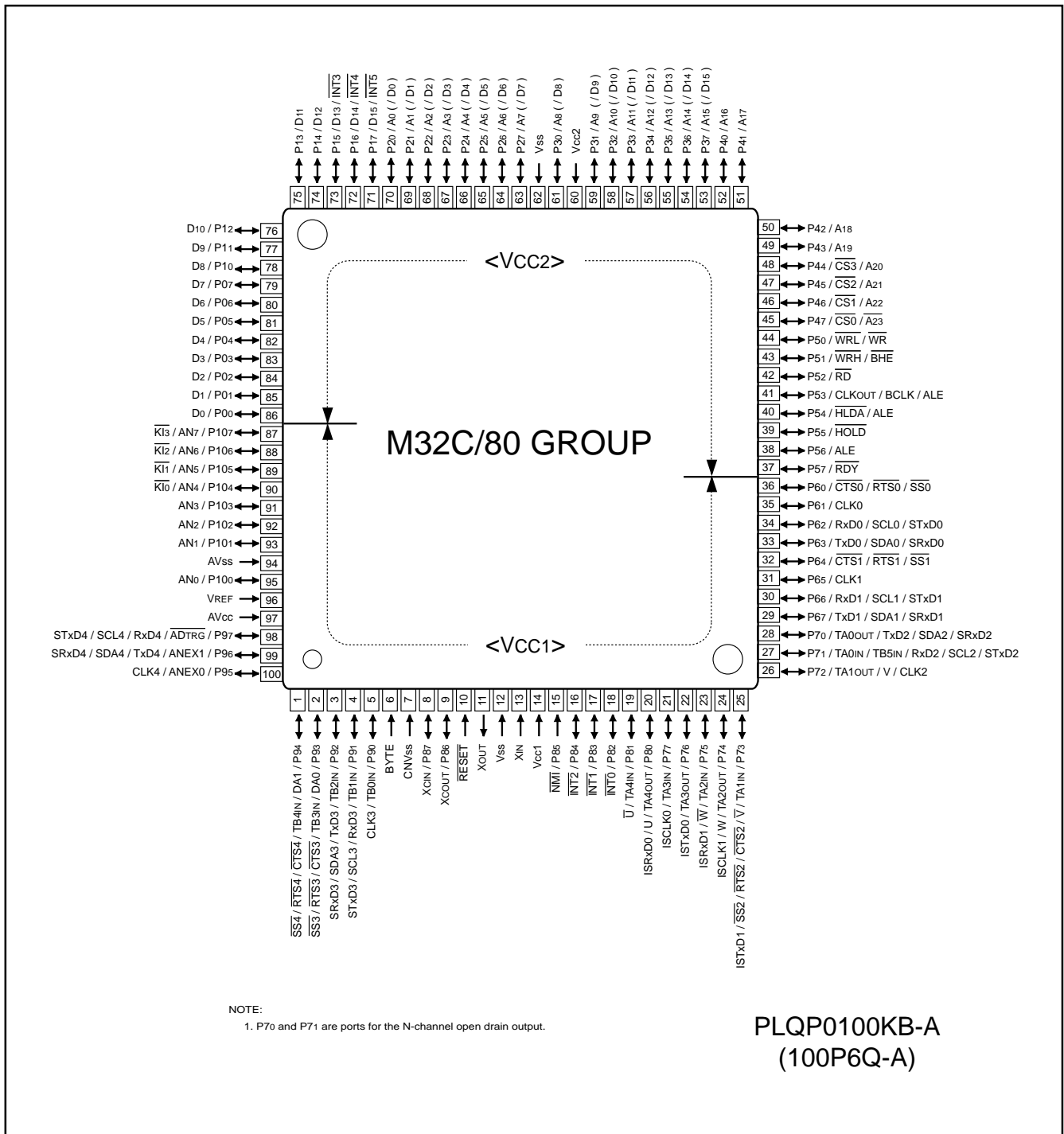


Figure 1.4 Pin Assignment

Table 1.3 Pin Characteristics

Package Pin No		Control pins	Port	Interrupt pins	Timer pins	UART pins	Analog pins	Bus control pins	Intelligent I/O pins
FP	GP								
1	99		P96			TxD4/SDA4/SRx4D4	ANEX1		
2	100		P95			CLK4	ANEX0		
3	1		P94		TB4IN	CTS4/RTS4/SS4	DA1		
4	2		P93		TB3IN	CTS3/RTS3/SS3	DA0		
5	3		P92		TB2IN	TxD3/SDA3/SRx4D3			
6	4		P91		TB1IN	RxD3/SCL3/STxD3			
7	5		P90		TB0IN	CLK3			
8	6	BYTE							
9	7	CNV <sub>SS</sub>							
10	8	XCIN	P87						
11	9	XCOUT	P86						
12	10	RESET							
13	11	XOUT							
14	12	V <sub>SS</sub>							
15	13	XIN							
16	14	V <sub>CC1</sub>							
17	15		P85	NMI					
18	16		P84	INT2					
19	17		P83	INT1					
20	18		P82	INT0					
21	19		P81		TA4IN/ $\bar{U}$				
22	20		P80		TA4OUT/U				ISRxD0
23	21		P77		TA3IN				ISCLK0
24	22		P76		TA3OUT				ISTxD0
25	23		P75		TA2IN/ $\bar{W}$				ISRxD1
26	24		P74		TA2OUT/W				ISCLK1
27	25		P73		TA1IN/ $\bar{V}$	CTS2/RTS2/SS2			ISTxD1
28	26		P72		TA1OUT/V	CLK2			
29	27		P71		TB5IN/TA0IN	RxD2/SCL2/STxD2			
30	28		P70		TA0OUT	TxD2/SDA2/SRx4D2			
31	29		P67			TxD1/SDA1/SRx4D1			
32	30		P66			RxD1/SCL1/STxD1			
33	31		P65			CLK1			
34	32		P64			CTS1/RTS1/SS1			
35	33		P63			TxD0/SDA0/SRx4D0			
36	34		P62			RxD0/SCL0/STxD0			
37	35		P61			CLK0			
38	36		P60			CTS0/RTS0/SS0			
39	37		P57					RDY	
40	38		P56					ALE	
41	39		P55					HOLD	
42	40		P54					H $\bar{L}$ DA/ALE	
43	41		P53					CLKout/BCLK/ALE	
44	42		P52					RD	
45	43		P51					WRH/BHE	
46	44		P50					WRL/WR	
47	45		P47					CS0/A23	
48	46		P46					CS1/A22	
49	47		P45					CS2/A21	
50	48		P44					CS3/A20	

Table 1.3 Pin Characteristics (Continued)

Package pin No		Control pins	Port	Interrupt pins	Timer pins	UART pins	Analog pins	Bus control pins	Intelligent I/O pins
FP	GP								
51	49		P43					A19	
52	50		P42					A18	
53	51		P41					A17	
54	52		P40					A16	
55	53		P37					A15(/D15)	
56	54		P36					A14(/D14)	
57	55		P35					A13(/D13)	
58	56		P34					A12(/D12)	
59	57		P33					A11(/D11)	
60	58		P32					A10(/D10)	
61	59		P31					A9(/D9)	
62	60	VCC2							
63	61		P30					A8(/D8)	
64	62	VSS							
65	63		P27					A7(/D7)	
66	64		P26					A6(/D6)	
67	65		P25					A5(/D5)	
68	66		P24					A4(/D4)	
69	67		P23					A3(/D3)	
70	68		P22					A2(/D2)	
71	69		P21					A1(/D1)	
72	70		P20					A0(/D0)	
73	71		P17	$\overline{\text{INT5}}$				D15	
74	72		P16	$\overline{\text{INT4}}$				D14	
75	73		P15	$\overline{\text{INT3}}$				D13	
76	74		P14					D12	
77	75		P13					D11	
78	76		P12					D10	
79	77		P11					D9	
80	78		P10					D8	
81	79		P07					D7	
82	80		P06					D6	
83	81		P05					D5	
84	82		P04					D4	
85	83		P03					D3	
86	84		P02					D2	
87	85		P01					D1	
88	86		P00					D0	
89	87		P107	$\overline{\text{KI3}}$			AN7		
90	88		P106	$\overline{\text{KI2}}$			AN6		
91	89		P105	$\overline{\text{KI1}}$			AN5		
92	90		P104	$\overline{\text{KI0}}$			AN4		
93	91		P103				AN3		
94	92		P102				AN2		
95	93		P101				AN1		
96	94	AVSS							
97	95		P100				AN0		
98	96						VREF		
99	97	AVCC							
100	98		P97			RxD4/SCL4/STxD4	$\overline{\text{ADTRG}}$		

## 1.6 Pin Description

**Table 1.4 Pin Description**

Signal name	Pin name	I/O type	Supply voltage	Description
Power supply	VCC1, VCC2 VSS	I	-	Apply 3.0 to 5.5 V to both VCC1 and VCC2 pins. Apply 0 V to the VSS pin. $VCC1 \geq VCC2^{(1)}$
Analog power supply input	AVCC AVSS	I	VCC1	Supplies power for the A/D converter. Connect the AVCC pin to VCC1 and the AVSS pin to VSS
Reset input	RESET	I	VCC1	The microcomputer is in a reset state when "L" is applied to the RESET pin
CNVss	CNVss	I	VCC1	Connect this pin to VCC1
External data bus width select input	BYTE	I	VCC1	Switches the data bus in external memory space 3. 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.
Bus control pins	D0 to D7	I/O	VCC2	Inputs and outputs data (D0 to D7) while accessing an external memory space with separate bus
	D8 to D15	I/O	VCC2	Inputs and outputs data (D8 to D15) while accessing an external memory space with 16-bit separate bus
	A0 to A22	O	VCC2	Outputs address bits (A0 to A22)
	A23	O	VCC2	Outputs inversed address bit A23
	A0/D0 to A7/D7	I/O	VCC2	Inputs and outputs data (D0 to D7) and outputs 8 low-order address bits (A0 to A7) by time-sharing while accessing an external memory space with multiplexed bus
	A8/D8 to A15/D15	I/O	VCC2	Inputs and outputs data (D8 to D15) and outputs 8 middle-order address bits (A8 to A15) by time-sharing while accessing an external memory space with multiplexed bus
	CS0 to CS3	O	VCC2	Output CS0 to CS3 that are chip-select signals specifying an external space
	WRL/WR WRH/BHE RD	O	VCC2	Outputs WRL, WRH, (WR, BHE) and RD signals. WRL and WRH can be switched with WR and BHE by program <ul style="list-style-type: none"> <li>■ WRL, WRH and RD are selected: If external data bus is 16 bits wide, data is writtenn to an even address when WRL is held "L". Data is written to an odd address when WRH is held "L". Data is read when RD is held "L".</li> <li>■ WR, BHE and RD are selected Data is written to external memory space when WR is held "L". Data is read when RD is held "L". An odd address is accessed when BHE is held "L". Select WR, BHE and RD for an external 8-bit data bus</li> </ul>
	ALE	O	VCC2	ALE is a signal latching address
	HOLD	I	VCC2	The microcomputer is placed in a hold state while the HOLD pin is held "L"
HLDA	O	VCC2	Outputs an "L" siganl while the microcomputer is placed in a hold state	
RDY	I	VCC2	Bus is placed in a wait state while the RDY pin is held "L"	

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

NOTE:

1. In this manual, hereafter, VCC refers to VCC1 unless otherwise noted.

**Table 1.4 Pin Description (Continued)**

Signal name	Pin name	I/O type	Supply voltage	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. To apply 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. To apply external clock, input the clock from XCIN and leave XCOU open
Sub clock output	XCOU	O	VCC1	
BCLK output	BCLK	O	VCC2	Outputs BCLK signal
Clock output	CLKOUT	O	VCC2	Outputs clock having the same frequency as fc, f8, or f32
INT interrupt input	INT0 to INT2	I	VCC1	Input pins for the INT interrupt
	INT3 to INT5		VCC2	
NMI interrupt input	NMI	I	VCC1	Input pin for the NMI interrupt
Key input interrupt	KI0 to KI3	I	VCC1	Input pins for the key input interrupt
Timer A	TA0OUT to TA4OUT	I/O	VCC1	I/O pins for the timer A0 to A4 (TA0OUT is a pin for the N-channel open drain output.)
	TA0IN to TA4IN		I	
Timer B	TB0IN to TB5IN	I	VCC1	Input pins for the timer B0 to B5
Three-phase motor control output	U, $\bar{U}$ , V, $\bar{V}$ , W, $\bar{W}$	O	VCC1	output pins for the three-phase motor control timer
Serial I/O	CTS0 to CTS4	I	VCC1	Input pins for data transmission control
	RTS0 to RTS4	O	VCC1	Output pins for data reception control
	CLK0 to CLK4	I/O	VCC1	Inputs and outputs the transfer clock
	RxD0 to RxD4	I	VCC1	Inputs serial data
	TxD0 to TxD4	O	VCC1	Outputs serial data (TxD2 is a pin for the N-channel open drain output.)
I <sup>2</sup> C mode	SDA0 to SDA4	I/O	VCC1	Inputs and outputs serial data (SDA2 is a pin for the N-channel open drain output.)
	SCL0 to SCL4		VCC1	Inputs and outputs the transfer clock (SCL2 is a pin for the N-channel open drain output.)
Serial I/O special function	STxD0 to STxD4	I	VCC1	Outputs serial data when slave mode is selected (SDA2 is a pin for the N-channel open drain output.)
	SRxD0 to SRxD4	I	VCC1	Inputs serial data when slave mode is selected
	SS0 to SS4	I	VCC1	Input pins to control serial I/O special function

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

**Table 1.5 Pin Description (Continued)**

Signal name	Pin name	I/O type	Supply voltage	Description			
Reference voltage input	VREF	I	-	Applies reference voltage for the A/D converter and D/A converter			
A/D converter	AN <sub>0</sub> to AN <sub>7</sub>	I	VCC1	Analog input pins for the A/D converter			
	ADTRG	I	VCC1	Input pin for an external A/D trigger			
	ANEX0	I/O	VCC1	Extended analog input pin for the A/D converter and output pin in external op-amp connection mode			
	ANEX1	I	VCC1	Extended analog input pin for the A/D converter			
D/A converter	DA0, DA1	O	VCC1	Output pin for the D/A converter			
Intelligent I/O communication function	ISCLK0	I/O	VCC1	Inputs and outputs clock for the intelligent I/O communication function			
	ISCLK1						
	ISTxD0	O	VCC1	Outputs data for the intelligent I/O communication function			
	ISTxD1						
	ISRxD0	I	VCC1	Inputs data for the intelligent I/O communication function			
	ISRxD1						
I/O port	P0 <sub>0</sub> to P0 <sub>7</sub> <sup>(1)</sup> P1 <sub>0</sub> to P1 <sub>7</sub> <sup>(2)</sup> P2 <sub>0</sub> to P2 <sub>7</sub> <sup>(1)</sup> P3 <sub>0</sub> to P3 <sub>7</sub> <sup>(1)</sup> P4 <sub>0</sub> to P4 <sub>7</sub> <sup>(1)</sup> P5 <sub>0</sub> to P5 <sub>7</sub> <sup>(1)</sup>	I/O	VCC2	I/O ports for CMOS. Each port can be programmed for input or output under the control of the direction register. An input port can be set, by program, for a pull-up resistor available or for no pull-up resistor available in 4-bit units			
	P6 <sub>0</sub> to P6 <sub>7</sub> P7 <sub>0</sub> to P7 <sub>7</sub> P9 <sub>0</sub> to P9 <sub>7</sub> P10 <sub>0</sub> to P10 <sub>7</sub>				I/O	VCC1	I/O ports having equivalent functions to P0 (P7 <sub>0</sub> and P7 <sub>1</sub> are ports for the N-channel open drain output.)
	P8 <sub>0</sub> to P8 <sub>4</sub> , P8 <sub>6</sub> , P8 <sub>7</sub>						
	P8 <sub>5</sub>				I	VCC1	Shares a pin with NMI. NMI input state can be got by reading P8 <sub>5</sub>

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

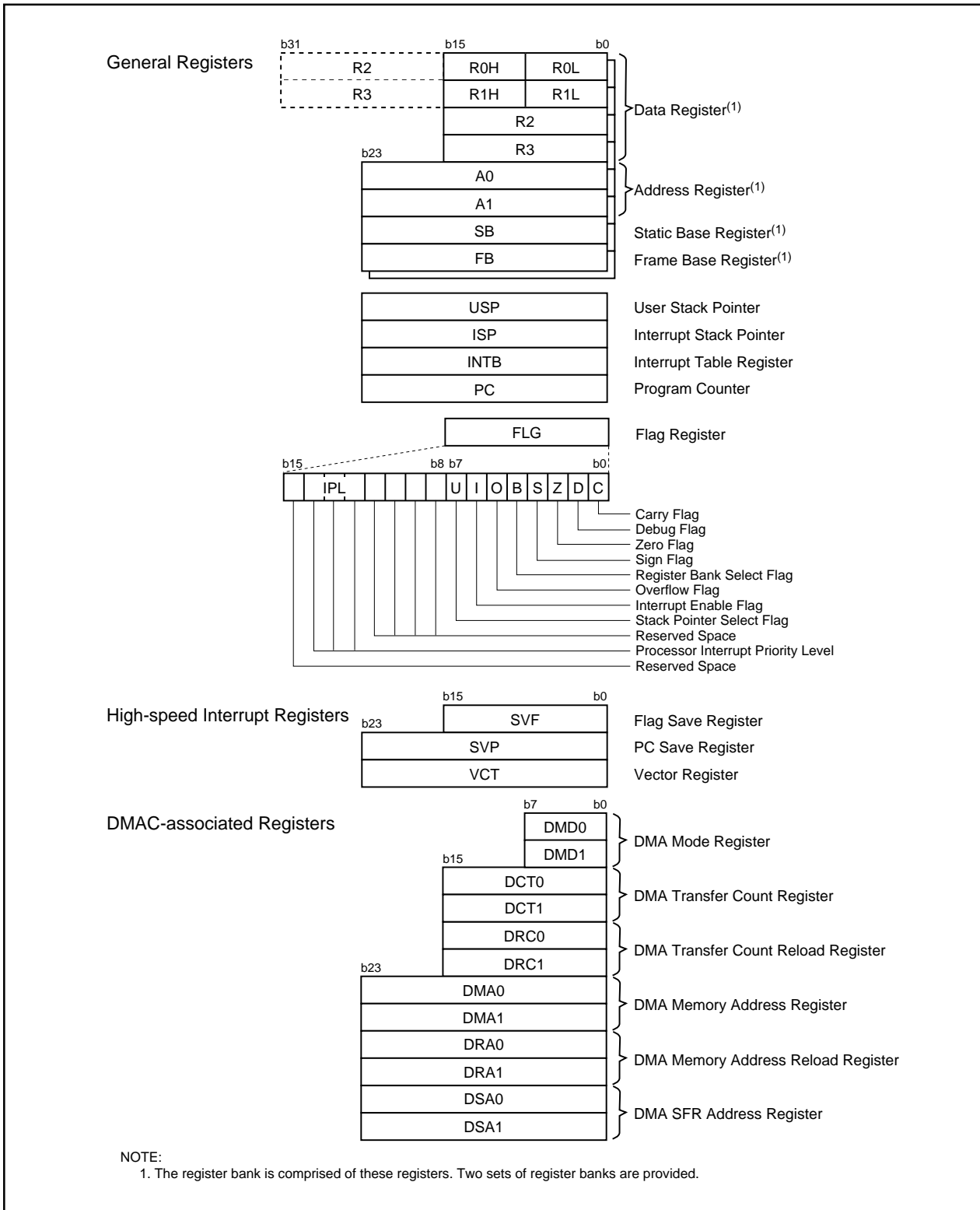
**NOTES:**

- Ports P0 to P5 function as bus control pins when using memory expansion mode or microprocessor mode. They cannot be used as I/O ports.
- Port P1 functions as I/O port when the microcomputer is placed in memory expansion mode or microprocessor mode and all external data buses are selected as 8-bit buses.

## 2. Central Processing Unit (CPU)

Figure 2.1 shows the CPU registers.

The register bank is comprised of 8 registers (R0, R1, R2, R3, A0, A1, SB and FB) out of 28 CPU registers. Two sets of register banks are provided.



**Figure 2.1 CPU Register**

## 2.1 General Registers

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

R0, R1, R2 and R3 are 16-bit registers for transfer, arithmetic and logic operations. R0 and R1 can be split into high-order bits (R0H) and low-order bits (R0L) to be used separately as 8-bit data registers. R0 can be combined with R2 to be used as a 32-bit data register (R2R0). The same applies to R1 and R3.

### 2.1.2 Address Registers (A0 and A1)

A0 and A1 are 24-bit registers for A0-/A1-indirect addressing, A0-/A1-relative addressing, transfer, arithmetic and logic operations.

### 2.1.3 Static Base Register (SB)

SB is a 24-bit register for SB-relative addressing.

### 2.1.4 Frame Base Register (FB)

FB is a 24-bit register for FB-relative addressing.

### 2.1.5 Program Counter (PC)

PC, 24 bits wide, indicates the address of an instruction to be executed.

### 2.1.6 Interrupt Table Register (INTB)

INTB is a 24-bit register indicating the starting address of an relocatable interrupt vector table.

### 2.1.7 User Stack Pointer (USP), Interrupt Stack Pointer (ISP)

The stack pointers (SP), USP and ISP, are 24 bits wide each. The U flag is used to switch between USP and ISP. Refer to **2.1.8 Flag Register (FLG)** for details on the U flag. Set USP and ISP to even addresses to execute an interrupt sequence efficiently.

### 2.1.8 Flag Register (FLG)

FLG is a 16-bit register indicating a CPU state.

#### 2.1.8.1 Carry Flag (C)

The C flag indicates whether carry or borrow has occurred after executing an instruction.

#### 2.1.8.2 Debug Flag (D)

The D flag is for debug only. Set to "0".

#### 2.1.8.3 Zero Flag (Z)

The Z flag is set to "1" when the value of zero is obtained from an arithmetic operation; otherwise "0".

#### 2.1.8.4 Sign Flag (S)

The S flag is set to "1" when a negative value is obtained from an arithmetic operation; otherwise "0".



### 2.1.8.5 Register Bank Select Flag (B)

The register bank 0 is selected when the B flag is set to "0". The register bank 1 is selected when this flag is set to "1".

### 2.1.8.6 Overflow Flag (O)

The O flag is set to "1" when the result of an arithmetic operation overflows; otherwise "0".

### 2.1.8.7 Interrupt Enable Flag (I)

The I flag enables a maskable interrupt.

Interrupt is disabled when the I flag is set to "0" and enabled when the I flag is set to "1". The I flag is set to "0" when an interrupt is acknowledged.

### 2.1.8.8 Stack Pointer Select Flag (U)

ISP is selected when the U flag is set to "0". USP is selected when this flag is set to "1".

The U flag is set to "0" when a hardware interrupt is acknowledged or the INT instruction of software interrupt numbers 0 to 31 is executed.

### 2.1.8.9 Processor Interrupt Priority Level (IPL)

IPL, 3 bits wide, assigns processor interrupt priority levels from level 0 to level 7.

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

### 2.1.8.10 Reserved Space

When writing to a reserved space, set to "0". When reading, its content is indeterminate.

## 2.2 High-Speed Interrupt Registers

Registers associated with the high-speed interrupt are as follows:

- Flag save register (SVF)
- PC save register (SVP)
- Vector register (VCT)

Refer to **10.4 High-Speed Interrupt** for details.

## 2.3 DMAC-Associated Registers

Registers associated with DMAC are as follows:

- DMA mode register (DMD0, DMD1)
- DMA transfer count register (DCT0, DCT1)
- DMA transfer count reload register (DRC0, DRC1)
- DMA memory address register (DMA0, DMA1)
- DMA SFR address register (DSA0, DSA1)
- DMA memory address reload register (DRA0, DRA1)

Refer to **12. DMAC** for details.

### 3. Memory

Figure 3.1 shows a memory map of the M32C/80 Group.

The M32C/80 Group provides 16-Mbyte address space addressed from 000000<sub>16</sub> to FFFFFFF<sub>16</sub>.

The fixed interrupt vectors are allocated from address FFFFDC<sub>16</sub> to FFFFFFF<sub>16</sub>. It stores the starting address of each interrupt routine.

The internal RAM is allocated from address 000400<sub>16</sub> to higher. For example, a 8-Kbyte internal RAM is allocated from address 000400<sub>16</sub> to 0023FF<sub>16</sub>. Besides storing data, it becomes stacks when the subroutine is called or an interrupt is acknowledged.

SFRs, consisting of control registers for peripheral functions such as I/O port, A/D converter, serial I/O, timers, is allocated from address 000000<sub>16</sub> to 0003FF<sub>16</sub>. All blank spaces within SFRs are reserved and cannot be accessed by users.

The special page vector table is addressed from FFFE00<sub>16</sub> to FFFFDB<sub>16</sub>. It is used for the JMPS instruction and JSRS instruction. Refer to the Renesas publication **M32C/80 Series Software Manual** for details. In microprocessor mode, some spaces are reserved and cannot be accessed by users.

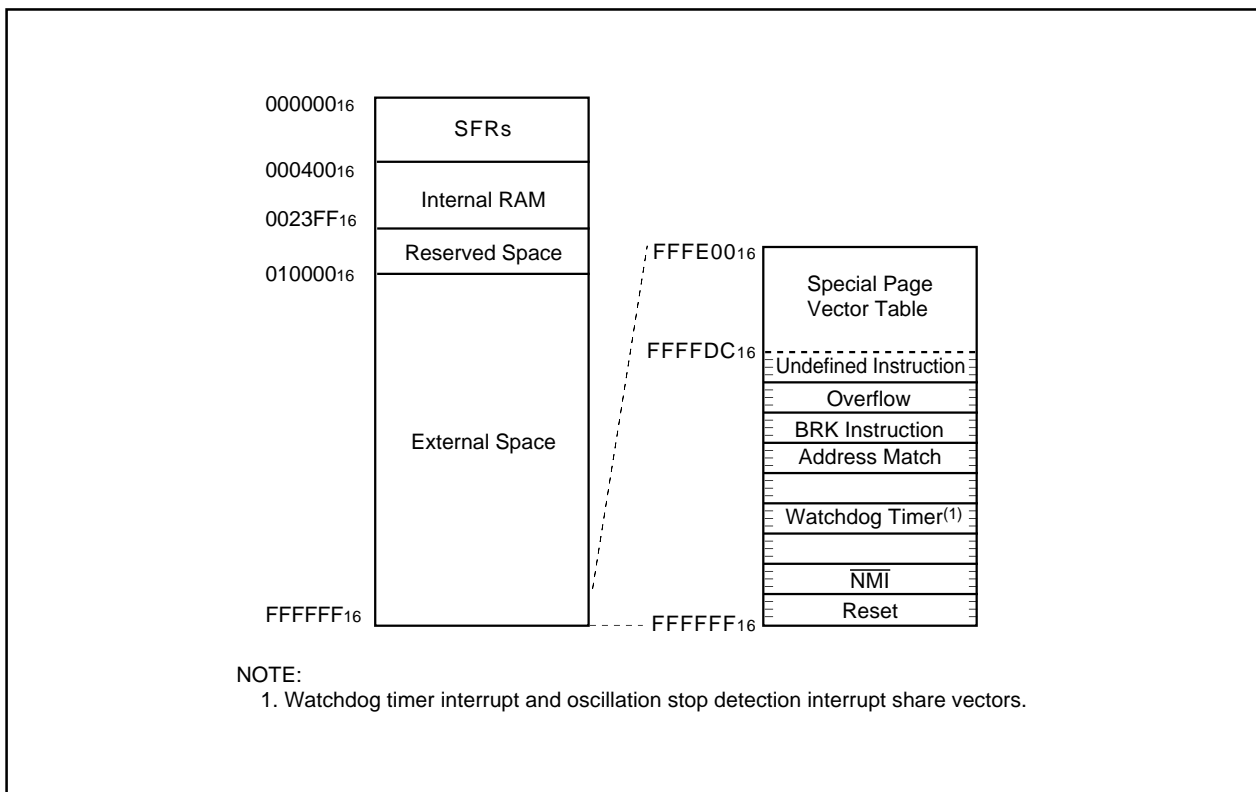


Figure 3.1 Memory Map

## 4. Special Function Registers (SFRs)

Address	Register	Symbol	Value after RESET
0000 <sub>16</sub>			
0001 <sub>16</sub>			
0002 <sub>16</sub>			
0003 <sub>16</sub>			
0004 <sub>16</sub>	Processor Mode Register <sup>(1)</sup>	PM0	0000 0011 <sub>2</sub> (CNVss pin ="H")
0005 <sub>16</sub>	Processor Mode Register 1	PM1	00 <sub>16</sub>
0006 <sub>16</sub>	System Clock Control Register 0	CM0	0000 1000 <sub>2</sub>
0007 <sub>16</sub>	System Clock Control Register 1	CM1	0010 0000 <sub>2</sub>
0008 <sub>16</sub>			
0009 <sub>16</sub>	Address Match Interrupt Enable Register	AIER	00 <sub>16</sub>
000A <sub>16</sub>	Protect Register	PRCR	XXXX 0000 <sub>2</sub>
000B <sub>16</sub>	External Data Bus Width Control Register	DS	XXXX 1000 <sub>2</sub> (BYTE pin ="L") XXXX 0000 <sub>2</sub> (BYTE pin ="H")
000C <sub>16</sub>	Main Clock Division Register	MCD	XXX0 1000 <sub>2</sub>
000D <sub>16</sub>	Oscillation Stop Detection Register	CM2	00 <sub>16</sub>
000E <sub>16</sub>	Watchdog Timer Start Register	WDTS	XX <sub>16</sub>
000F <sub>16</sub>	Watchdog Timer Control Register	WDC	000X XXXX <sub>2</sub>
0010 <sub>16</sub>			
0011 <sub>16</sub>	Address Match Interrupt Register 0	RMAD0	000000 <sub>16</sub>
0012 <sub>16</sub>			
0013 <sub>16</sub>	Processor Mode Register 2	PM2	00 <sub>16</sub>
0014 <sub>16</sub>			
0015 <sub>16</sub>	Address Match Interrupt Register 1	RMAD1	000000 <sub>16</sub>
0016 <sub>16</sub>			
0017 <sub>16</sub>			
0018 <sub>16</sub>			
0019 <sub>16</sub>	Address Match Interrupt Register 2	RMAD2	000000 <sub>16</sub>
001A <sub>16</sub>			
001B <sub>16</sub>			
001C <sub>16</sub>			
001D <sub>16</sub>	Address Match Interrupt Register 3	RMAD3	000000 <sub>16</sub>
001E <sub>16</sub>			
001F <sub>16</sub>			
0020 <sub>16</sub>			
0021 <sub>16</sub>			
0022 <sub>16</sub>			
0023 <sub>16</sub>			
0024 <sub>16</sub>			
0025 <sub>16</sub>			
0026 <sub>16</sub>	PLL Control Register 0	PLC0	0001 X010 <sub>2</sub>
0027 <sub>16</sub>	PLL Control Register 1	PLC1	000X 0000 <sub>2</sub>
0028 <sub>16</sub>			
0029 <sub>16</sub>	Address Match Interrupt Register 4	RMAD4	000000 <sub>16</sub>
002A <sub>16</sub>			
002B <sub>16</sub>			
002C <sub>16</sub>			
002D <sub>16</sub>	Address Match Interrupt Register 5	RMAD5	000000 <sub>16</sub>
002E <sub>16</sub>			
002F <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTE:

1. The PM01 and PM00 bits in the PM0 register maintain values set before reset, even after software reset or watchdog timer reset has been performed.

Address	Register	Symbol	Value after RESET
0030 <sub>16</sub>			
0031 <sub>16</sub>			
0032 <sub>16</sub>			
0033 <sub>16</sub>			
0034 <sub>16</sub>			
0035 <sub>16</sub>			
0036 <sub>16</sub>			
0037 <sub>16</sub>			
0038 <sub>16</sub>	Address Match Interrupt Register 6	RMAD6	000000 <sub>16</sub>
0039 <sub>16</sub>			
003A <sub>16</sub>			
003B <sub>16</sub>			
003C <sub>16</sub>	Address Match Interrupt Register 7	RMAD7	000000 <sub>16</sub>
003D <sub>16</sub>			
003E <sub>16</sub>			
003F <sub>16</sub>			
0040 <sub>16</sub>			
0041 <sub>16</sub>			
0042 <sub>16</sub>			
0043 <sub>16</sub>			
0044 <sub>16</sub>			
0045 <sub>16</sub>			
0046 <sub>16</sub>			
0047 <sub>16</sub>			
0048 <sub>16</sub>	External Space Wait Control Register 0	EWCR0	X0X0 0011 <sub>2</sub>
0049 <sub>16</sub>	External Space Wait Control Register 1	EWCR1	X0X0 0011 <sub>2</sub>
004A <sub>16</sub>	External Space Wait Control Register 2	EWCR2	X0X0 0011 <sub>2</sub>
004B <sub>16</sub>	External Space Wait Control Register 3	EWCR3	X0X0 0011 <sub>2</sub>
004C <sub>16</sub>			
004D <sub>16</sub>			
004E <sub>16</sub>			
004F <sub>16</sub>			
0050 <sub>16</sub>			
0051 <sub>16</sub>			
0052 <sub>16</sub>			
0053 <sub>16</sub>			
0054 <sub>16</sub>			
0055 <sub>16</sub>			
0056 <sub>16</sub>			
0057 <sub>16</sub>			
0058 <sub>16</sub>			
0059 <sub>16</sub>			
005A <sub>16</sub>			
005B <sub>16</sub>			
005C <sub>16</sub>			
005D <sub>16</sub>			
005E <sub>16</sub>			
005F <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0060 <sub>16</sub>			
0061 <sub>16</sub>			
0062 <sub>16</sub>			
0063 <sub>16</sub>			
0064 <sub>16</sub>			
0065 <sub>16</sub>			
0066 <sub>16</sub>			
0067 <sub>16</sub>			
0068 <sub>16</sub>	DMA0 Interrupt Control Register	DM0IC	XXXX X0002
0069 <sub>16</sub>	Timer B5 Interrupt Control Register	TB5IC	XXXX X0002
006A <sub>16</sub>	DMA2 Interrupt Control Register	DM2IC	XXXX X0002
006B <sub>16</sub>	UART2 Receive /ACK Interrupt Control Register	S2RIC	XXXX X0002
006C <sub>16</sub>	Timer A0 Interrupt Control Register	TA0IC	XXXX X0002
006D <sub>16</sub>	UART3 Receive /ACK Interrupt Control Register	S3RIC	XXXX X0002
006E <sub>16</sub>	Timer A2 Interrupt Control Register	TA2IC	XXXX X0002
006F <sub>16</sub>	UART4 Receive /ACK Interrupt Control Register	S4RIC	XXXX X0002
0070 <sub>16</sub>	Timer A4 Interrupt Control Register	TA4IC	XXXX X0002
0071 <sub>16</sub>	UART0/UART3 Bus Conflict Detect Interrupt Control Register	BCN0IC/BCN3IC	XXXX X0002
0072 <sub>16</sub>	UART0 Receive/ACK Interrupt Control Register	S0RIC	XXXX X0002
0073 <sub>16</sub>	A/D0 Conversion Interrupt Control Register	AD0IC	XXXX X0002
0074 <sub>16</sub>	UART1 Receive/ACK Interrupt Control Register	S1RIC	XXXX X0002
0075 <sub>16</sub>	Intelligent I/O Interrupt Control Register 0	IIO0IC	XXXX X0002
0076 <sub>16</sub>	Timer B1 Interrupt Control Register	TB1IC	XXXX X0002
0077 <sub>16</sub>	Intelligent I/O Interrupt Control Register 2	IIO2IC	XXXX X0002
0078 <sub>16</sub>	Timer B3 Interrupt Control Register	TB3IC	XXXX X0002
0079 <sub>16</sub>	Intelligent I/O Interrupt Control Register 4	IIO4IC	XXXX X0002
007A <sub>16</sub>	INT5 Interrupt Control Register	INT5IC	XX00 X0002
007B <sub>16</sub>			
007C <sub>16</sub>	INT3 Interrupt Control Register	INT3IC	XX00 X0002
007D <sub>16</sub>			
007E <sub>16</sub>	INT1 Interrupt Control Register	INT1IC	XX00 X0002
007F <sub>16</sub>			
0080 <sub>16</sub>			
0081 <sub>16</sub>			
0082 <sub>16</sub>			
0083 <sub>16</sub>			
0084 <sub>16</sub>			
0085 <sub>16</sub>			
0086 <sub>16</sub>			
0087 <sub>16</sub>			
0088 <sub>16</sub>	DMA1 Interrupt Control Register	DM1IC	XXXX X0002
0089 <sub>16</sub>	UART2 Transmit /NACK Interrupt Control Register	S2TIC	XXXX X0002
008A <sub>16</sub>	DMA3 Interrupt Control Register	DM3IC	XXXX X0002
008B <sub>16</sub>	UART3 Transmit /NACK Interrupt Control Register	S3TIC	XXXX X0002
008C <sub>16</sub>	Timer A1 Interrupt Control Register	TA1IC	XXXX X0002
008D <sub>16</sub>	UART4 Transmit /NACK Interrupt Control Register	S4TIC	XXXX X0002
008E <sub>16</sub>	Timer A3 Interrupt Control Register	TA3IC	XXXX X0002
008F <sub>16</sub>	UART2 Bus Conflict Detect Interrupt Control Register	BCN2IC	XXXX X0002

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0090 <sub>16</sub>	UART0 Transmit /NACK Interrupt Control Register	S0TIC	XXXX X000 <sub>2</sub>
0091 <sub>16</sub>	UART1/UART4 Bus Conflict Detect Interrupt Control Register	BCN1IC/BCN4IC	XXXX X000 <sub>2</sub>
0092 <sub>16</sub>	UART1 Transmit/NACK Interrupt Control Register	S1TIC	XXXX X000 <sub>2</sub>
0093 <sub>16</sub>	Key Input Interrupt Control Register	KUPIC	XXXX X000 <sub>2</sub>
0094 <sub>16</sub>	Timer B0 Interrupt Control Register	TB0IC	XXXX X000 <sub>2</sub>
0095 <sub>16</sub>	Intelligent I/O Interrupt Control Register 1	IIO1IC	XXXX X000 <sub>2</sub>
0096 <sub>16</sub>	Timer B2 Interrupt Control Register	TB2IC	XXXX X000 <sub>2</sub>
0097 <sub>16</sub>	Intelligent I/O Interrupt Control Register 3	IIO3IC	XXXX X000 <sub>2</sub>
0098 <sub>16</sub>	Timer B4 Interrupt Control Register	TB4IC	XXXX X000 <sub>2</sub>
0099 <sub>16</sub>			
009A <sub>16</sub>	INT4 Interrupt Control Register	INT4IC	XX00 X000 <sub>2</sub>
009B <sub>16</sub>			
009C <sub>16</sub>	INT2 Interrupt Control Register	INT2IC	XX00 X000 <sub>2</sub>
009D <sub>16</sub>			
009E <sub>16</sub>	INT0 Interrupt Control Register	INT0IC	XX00 X000 <sub>2</sub>
009F <sub>16</sub>	Exit Priority Control Register	RLVL	XXXX 0000 <sub>2</sub>
00A0 <sub>16</sub>	Interrupt Request Register 0	IIO0IR	0000 000X <sub>2</sub>
00A1 <sub>16</sub>	Interrupt Request Register 1	IIO1IR	0000 000X <sub>2</sub>
00A2 <sub>16</sub>	Interrupt Request Register 2	IIO2IR	0000 000X <sub>2</sub>
00A3 <sub>16</sub>	Interrupt Request Register 3	IIO3IR	0000 000X <sub>2</sub>
00A4 <sub>16</sub>	Interrupt Request Register 4	IIO4IR	0000 000X <sub>2</sub>
00A5 <sub>16</sub>			
00A6 <sub>16</sub>			
00A7 <sub>16</sub>			
00A8 <sub>16</sub>			
00A9 <sub>16</sub>			
00AA <sub>16</sub>			
00AB <sub>16</sub>			
00AC <sub>16</sub>			
00AD <sub>16</sub>			
00AE <sub>16</sub>			
00AF <sub>16</sub>			
00B0 <sub>16</sub>	Interrupt Enable Register 0	IIO0IE	00 <sub>16</sub>
00B1 <sub>16</sub>	Interrupt Enable Register 1	IIO1IE	00 <sub>16</sub>
00B2 <sub>16</sub>	Interrupt Enable Register 2	IIO2IE	00 <sub>16</sub>
00B3 <sub>16</sub>	Interrupt Enable Register 3	IIO3IE	00 <sub>16</sub>
00B4 <sub>16</sub>	Interrupt Enable Register 4	IIO4IE	00 <sub>16</sub>
00B5 <sub>16</sub>			
00B6 <sub>16</sub>			
00B7 <sub>16</sub>			
00B8 <sub>16</sub>			
00B9 <sub>16</sub>			
00BA <sub>16</sub>			
00BB <sub>16</sub>			
00BC <sub>16</sub>			
00BD <sub>16</sub>			
00BE <sub>16</sub>			
00BF <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
00C0 <sub>16</sub>			
00C1 <sub>16</sub>			
00C2 <sub>16</sub>			
00C3 <sub>16</sub>			
00C4 <sub>16</sub>			
00C5 <sub>16</sub>			
00C6 <sub>16</sub>			
00C7 <sub>16</sub>			
00C8 <sub>16</sub>			
00C9 <sub>16</sub>			
00CA <sub>16</sub>			
00CB <sub>16</sub>			
00CC <sub>16</sub>			
00CD <sub>16</sub>			
00CE <sub>16</sub>			
00CF <sub>16</sub>			
00D0 <sub>16</sub>			
00D1 <sub>16</sub>			
00D2 <sub>16</sub>			
00D3 <sub>16</sub>			
00D4 <sub>16</sub>			
00D5 <sub>16</sub>			
00D6 <sub>16</sub>			
00D7 <sub>16</sub>			
00D8 <sub>16</sub>			
00D9 <sub>16</sub>			
00DA <sub>16</sub>			
00DB <sub>16</sub>			
00DC <sub>16</sub>			
00DD <sub>16</sub>			
00DE <sub>16</sub>			
00DF <sub>16</sub>			
00E0 <sub>16</sub>			
00E1 <sub>16</sub>			
00E2 <sub>16</sub>			
00E3 <sub>16</sub>			
00E4 <sub>16</sub>			
00E5 <sub>16</sub>			
00E6 <sub>16</sub>			
00E7 <sub>16</sub>			
00E8 <sub>16</sub> 00E9 <sub>16</sub>	SI/O Receive Buffer Register 0	G0RB	XXXX XXXX <sub>2</sub> XXX0 XXXX <sub>2</sub>
00EA <sub>16</sub> 00EB <sub>16</sub>	Transmit Buffer/Receive Data Register 0	G0TB/G0DR	XX <sub>16</sub>
00EC <sub>16</sub>	Receive Input Register 0	G0RI	XX <sub>16</sub>
00ED <sub>16</sub>	SI/O Communication Mode Register 0	G0MR	00 <sub>16</sub>
00EE <sub>16</sub>	Transmit Output Register 0	G0TO	XX <sub>16</sub>
00EF <sub>16</sub>	SI/O Communication Control Register 0	G0CR	0000 X011 <sub>2</sub>

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
00F0 <sub>16</sub>	Data Compare Register 00	G0CMP0	XX <sub>16</sub>
00F1 <sub>16</sub>	Data Compare Register 01	G0CMP1	XX <sub>16</sub>
00F2 <sub>16</sub>	Data Compare Register 02	G0CMP2	XX <sub>16</sub>
00F3 <sub>16</sub>	Data Compare Register 03	G0CMP3	XX <sub>16</sub>
00F4 <sub>16</sub>	Data Mask Register 00	G0MSK0	XX <sub>16</sub>
00F5 <sub>16</sub>	Data Mask Register 01	G0MSK1	XX <sub>16</sub>
00F6 <sub>16</sub>	Communication Clock Select Register	CCS	XXXX 0000 <sub>2</sub>
00F7 <sub>16</sub>			
00F8 <sub>16</sub>	Receive CRC Code Register 0	G0RCRC	XX <sub>16</sub>
00F9 <sub>16</sub>			XX <sub>16</sub>
00FA <sub>16</sub>	Transmit CRC Code Register 0	G0TCRC	00 <sub>16</sub>
00FB <sub>16</sub>			00 <sub>16</sub>
00FC <sub>16</sub>	SI/O Expansion Mode Register 0	G0EMR	00 <sub>16</sub>
00FD <sub>16</sub>	SI/O Expansion Receive Control Register 0	G0ERC	00 <sub>16</sub>
00FE <sub>16</sub>	SI/O Special Communication Interrupt Detect Register 0	G0IRF	00 <sub>16</sub>
00FF <sub>16</sub>	SI/O Expansion Transmit Control Register 0	G0ETC	0000 0XXX <sub>2</sub>
0100 <sub>16</sub>			
0101 <sub>16</sub>			
0102 <sub>16</sub>			
0103 <sub>16</sub>			
0104 <sub>16</sub>			
0105 <sub>16</sub>			
0106 <sub>16</sub>			
0107 <sub>16</sub>			
0108 <sub>16</sub>			
0109 <sub>16</sub>			
010A <sub>16</sub>			
010B <sub>16</sub>			
010C <sub>16</sub>			
010D <sub>16</sub>			
010E <sub>16</sub>			
010F <sub>16</sub>			
0110 <sub>16</sub>			
0111 <sub>16</sub>			
0112 <sub>16</sub>			
0113 <sub>16</sub>			
0114 <sub>16</sub>			
0115 <sub>16</sub>			
0116 <sub>16</sub>			
0117 <sub>16</sub>			
0118 <sub>16</sub>			
0119 <sub>16</sub>			
011A <sub>16</sub>			
011B <sub>16</sub>			
011C <sub>16</sub>			
011D <sub>16</sub>			
011E <sub>16</sub>			
011F <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.



Address	Register	Symbol	Value after RESET
0120 <sub>16</sub>			
0121 <sub>16</sub>			
0122 <sub>16</sub>			
0123 <sub>16</sub>			
0124 <sub>16</sub>			
0125 <sub>16</sub>			
0126 <sub>16</sub>			
0127 <sub>16</sub>			
0128 <sub>16</sub> 0129 <sub>16</sub>	SI/O Receive Buffer Register 1	G1RB	XXXX XXXX <sub>2</sub> XXX0 XXXX <sub>2</sub>
012A <sub>16</sub> 012B <sub>16</sub>	Transmit Buffer/Receive Data Register 1	G1TB/G1DR	XX <sub>16</sub>
012C <sub>16</sub>	Receive Input Register 1	G1RI	XX <sub>16</sub>
012D <sub>16</sub>	SI/O Communication Mode Register 1	G1MR	00 <sub>16</sub>
012E <sub>16</sub>	Transmit Output Register 1	G1TO	XX <sub>16</sub>
012F <sub>16</sub>	SI/O Communication Control Register 1	G1CR	0000 X011 <sub>2</sub>
0130 <sub>16</sub>	Data Compare Register 10	G1CMP0	XX <sub>16</sub>
0131 <sub>16</sub>	Data Compare Register 11	G1CMP1	XX <sub>16</sub>
0132 <sub>16</sub>	Data Compare Register 12	G1CMP2	XX <sub>16</sub>
0133 <sub>16</sub>	Data Compare Register 13	G1CMP3	XX <sub>16</sub>
0134 <sub>16</sub>	Data Mask Register 10	G1MSK0	XX <sub>16</sub>
0135 <sub>16</sub>	Data Mask Register 11	G1MSK1	XX <sub>16</sub>
0136 <sub>16</sub>			
0137 <sub>16</sub>			
0138 <sub>16</sub> 0139 <sub>16</sub>	Receive CRC Code Register 1	G1RCRC	XX <sub>16</sub> XX <sub>16</sub>
013A <sub>16</sub> 013B <sub>16</sub>	Transmit CRC Code Register 1	G1TCRC	00 <sub>16</sub> 00 <sub>16</sub>
013C <sub>16</sub>	SI/O Expansion Mode Register 1	G1EMR	00 <sub>16</sub>
013D <sub>16</sub>	SI/O Expansion Receive Control Register 1	G1ERC	00 <sub>16</sub>
013E <sub>16</sub>	SI/O Special Communication Interrupt Detection Register 1	G1IRF	00 <sub>16</sub>
013F <sub>16</sub>	SI/O Expansion Transmit Control Register 1	G1ETC	0000 0XXX <sub>2</sub>
0140 <sub>16</sub>			
0141 <sub>16</sub>			
0142 <sub>16</sub>			
0143 <sub>16</sub>			
0144 <sub>16</sub>			
0145 <sub>16</sub>			
0146 <sub>16</sub>			
0147 <sub>16</sub>			
0148 <sub>16</sub>			
0149 <sub>16</sub>			
014A <sub>16</sub>			
014B <sub>16</sub>			
014C <sub>16</sub>			
014D <sub>16</sub> to 02AF <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
02B1 <sub>16</sub>			
02B2 <sub>16</sub>			
02B3 <sub>16</sub>			
02B4 <sub>16</sub>			
02B5 <sub>16</sub>			
02B6 <sub>16</sub>			
02B7 <sub>16</sub>			
02B8 <sub>16</sub>			
02B9 <sub>16</sub>			
02BA <sub>16</sub>			
02BB <sub>16</sub>			
02BC <sub>16</sub>			
02BD <sub>16</sub>			
02BE <sub>16</sub>			
02BF <sub>16</sub>			
02C0 <sub>16</sub> 02C1 <sub>16</sub>	X0 Register Y0 Register	X0R,Y0R	XX <sub>16</sub> XX <sub>16</sub>
02C2 <sub>16</sub> 02C3 <sub>16</sub>	X1 Register Y1 Register	X1R,Y1R	XX <sub>16</sub> XX <sub>16</sub>
02C4 <sub>16</sub> 02C5 <sub>16</sub>	X2 Register Y2 Register	X2R,Y2R	XX <sub>16</sub> XX <sub>16</sub>
02C6 <sub>16</sub> 02C7 <sub>16</sub>	X3 Register Y3 Register	X3R,Y3R	XX <sub>16</sub> XX <sub>16</sub>
02C8 <sub>16</sub> 02C9 <sub>16</sub>	X4 Register Y4 Register	X4R,Y4R	XX <sub>16</sub> XX <sub>16</sub>
02CA <sub>16</sub> 02CB <sub>16</sub>	X5 Register Y5 Register	X5R,Y5R	XX <sub>16</sub> XX <sub>16</sub>
02CC <sub>16</sub> 02CD <sub>16</sub>	X6 Register Y6 Register	X6R,Y6R	XX <sub>16</sub> XX <sub>16</sub>
02CE <sub>16</sub> 02CF <sub>16</sub>	X7 Register Y7 Register	X7R,Y7R	XX <sub>16</sub> XX <sub>16</sub>
02D0 <sub>16</sub> 02D1 <sub>16</sub>	X8 Register Y8 Register	X8R,Y8R	XX <sub>16</sub> XX <sub>16</sub>
02D2 <sub>16</sub> 02D3 <sub>16</sub>	X9 Register Y9 Register	X9R,Y9R	XX <sub>16</sub> XX <sub>16</sub>
02D4 <sub>16</sub> 02D5 <sub>16</sub>	X10 Register Y10 Register	X10R,Y10R	XX <sub>16</sub> XX <sub>16</sub>
02D6 <sub>16</sub> 02D7 <sub>16</sub>	X11 Register Y11 Register	X11R,Y11R	XX <sub>16</sub> XX <sub>16</sub>
02D8 <sub>16</sub> 02D9 <sub>16</sub>	X12 Register Y12 Register	X12R,Y12R	XX <sub>16</sub> XX <sub>16</sub>
02DA <sub>16</sub> 02DB <sub>16</sub>	X13 Register Y13 Register	X13R,Y13R	XX <sub>16</sub> XX <sub>16</sub>
02DC <sub>16</sub> 02DD <sub>16</sub>	X14 Register Y14 Register	X14R,Y14R	XX <sub>16</sub> XX <sub>16</sub>
02DE <sub>16</sub> 02DF <sub>16</sub>	X15 Register Y15 Register	X15R,Y15R	XX <sub>16</sub> XX <sub>16</sub>

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
02E0 <sub>16</sub>	X/Y Control Register	XYC	XXXX XX00 <sub>2</sub>
02E1 <sub>16</sub>			
02E2 <sub>16</sub>			
02E3 <sub>16</sub>			
02E4 <sub>16</sub>	UART1 Special Mode Register 4	U1SMR4	00 <sub>16</sub>
02E5 <sub>16</sub>	UART1 Special Mode Register 3	U1SMR3	00 <sub>16</sub>
02E6 <sub>16</sub>	UART1 Special Mode Register 2	U1SMR2	00 <sub>16</sub>
02E7 <sub>16</sub>	UART1 Special Mode Register	U1SMR	00 <sub>16</sub>
02E8 <sub>16</sub>	UART1 Transmit/Receive Mode Register	U1MR	00 <sub>16</sub>
02E9 <sub>16</sub>	UART1 Bit Rate Register	U1BRG	XX <sub>16</sub>
02EA <sub>16</sub>	UART1 Transmit Buffer Register	U1TB	XX <sub>16</sub>
02EB <sub>16</sub>			XX <sub>16</sub>
02EC <sub>16</sub>	UART1 Transmit/Receive Control Register 0	U1C0	0000 1000 <sub>2</sub>
02ED <sub>16</sub>	UART1 Transmit/Receive Control Register 1	U1C1	0000 0010 <sub>2</sub>
02EE <sub>16</sub>	UART1 Receive Buffer Register	U1RB	XX <sub>16</sub>
02EF <sub>16</sub>			XX <sub>16</sub>
02F0 <sub>16</sub>			
02F1 <sub>16</sub>			
02F2 <sub>16</sub>			
02F3 <sub>16</sub>			
02F4 <sub>16</sub>	UART4 Special Mode Register 4	U4SMR4	00 <sub>16</sub>
02F5 <sub>16</sub>	UART4 Special Mode Register 3	U4SMR3	00 <sub>16</sub>
02F6 <sub>16</sub>	UART4 Special Mode Register 2	U4SMR2	00 <sub>16</sub>
02F7 <sub>16</sub>	UART4 Special Mode Register	U4SMR	00 <sub>16</sub>
02F8 <sub>16</sub>	UART4 Transmit/Receive Mode Register	U4MR	00 <sub>16</sub>
02F9 <sub>16</sub>	UART4 Bit Rate Register	U4BRG	XX <sub>16</sub>
02FA <sub>16</sub>	UART4 Transmit Buffer Register	U4TB	XX <sub>16</sub>
02FB <sub>16</sub>			XX <sub>16</sub>
02FC <sub>16</sub>	UART4 Transmit/Receive Control Register 0	U4C0	0000 1000 <sub>2</sub>
02FD <sub>16</sub>	UART4 Transmit/Receive Control Register 1	U4C1	0000 0010 <sub>2</sub>
02FE <sub>16</sub>	UART4 Receive Buffer Register	U4RB	XX <sub>16</sub>
02FF <sub>16</sub>			XX <sub>16</sub>
0300 <sub>16</sub>	Timer B3, B4, B5 Count Start Flag	TBSR	000X XXXX <sub>2</sub>
0301 <sub>16</sub>			
0302 <sub>16</sub>	Timer A1-1 Register	TA11	XX <sub>16</sub>
0303 <sub>16</sub>			XX <sub>16</sub>
0304 <sub>16</sub>	Timer A2-1 Register	TA21	XX <sub>16</sub>
0305 <sub>16</sub>			XX <sub>16</sub>
0306 <sub>16</sub>	Timer A4-1 Register	TA41	XX <sub>16</sub>
0307 <sub>16</sub>			XX <sub>16</sub>
0308 <sub>16</sub>	Three-Phase PWM Control Register 0	INVC0	00 <sub>16</sub>
0309 <sub>16</sub>	Three-Phase PWM Control Register 1	INVC1	00 <sub>16</sub>
030A <sub>16</sub>	Three-Phase Output Buffer Register 0	IDB0	XX11 1111 <sub>2</sub>
030B <sub>16</sub>	Three-Phase Output Buffer Register 1	IDB1	XX11 1111 <sub>2</sub>
030C <sub>16</sub>	Dead Time Timer	DTT	XX <sub>16</sub>
030D <sub>16</sub>	Timer B2 Interrupt Generation Frequency Set Counter	ICTB2	XX <sub>16</sub>
030E <sub>16</sub>			
030F <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0310 <sub>16</sub> 0311 <sub>16</sub>	Timer B3 Register	TB3	XX <sub>16</sub> XX <sub>16</sub>
0312 <sub>16</sub> 0313 <sub>16</sub>	Timer B4 Register	TB4	XX <sub>16</sub> XX <sub>16</sub>
0314 <sub>16</sub> 0315 <sub>16</sub>	Timer B5 Register	TB5	XX <sub>16</sub> XX <sub>16</sub>
0316 <sub>16</sub>			
0317 <sub>16</sub>			
0318 <sub>16</sub>			
0319 <sub>16</sub>			
031A <sub>16</sub>			
031B <sub>16</sub>	Timer B3 Mode Register	TB3MR	00XX 0000 <sub>2</sub>
031C <sub>16</sub>	Timer B4 Mode Register	TB4MR	00XX 0000 <sub>2</sub>
031D <sub>16</sub>	Timer B5 Mode Register	TB5MR	00XX 0000 <sub>2</sub>
031E <sub>16</sub>			
031F <sub>16</sub>	External Interrupt Request Source Select Register	IFSR	00 <sub>16</sub>
0320 <sub>16</sub>			
0321 <sub>16</sub>			
0322 <sub>16</sub>			
0323 <sub>16</sub>			
0324 <sub>16</sub>	UART3 Special Mode Register 4	U3SMR4	00 <sub>16</sub>
0325 <sub>16</sub>	UART3 Special Mode Register 3	U3SMR3	00 <sub>16</sub>
0326 <sub>16</sub>	UART3 Special Mode Register 2	U3SMR2	00 <sub>16</sub>
0327 <sub>16</sub>	UART3 Special Mode Register	U3SMR	00 <sub>16</sub>
0328 <sub>16</sub>	UART3 Transmit/Receive Mode Register	U3MR	00 <sub>16</sub>
0329 <sub>16</sub>	UART3 Bit Rate Register	U3BRG	XX <sub>16</sub>
032A <sub>16</sub> 032B <sub>16</sub>	UART3 Transmit Buffer Register	U3TB	XX <sub>16</sub> XX <sub>16</sub>
032C <sub>16</sub>	UART3 Transmit/Receive Control Register 0	U3C0	0000 1000 <sub>2</sub>
032D <sub>16</sub>	UART3 Transmit/Receive Control Register 1	U3C1	0000 0010 <sub>2</sub>
032E <sub>16</sub> 032F <sub>16</sub>	UART3 Receive Buffer Register	U3RB	XX <sub>16</sub> XX <sub>16</sub>
0330 <sub>16</sub>			
0331 <sub>16</sub>			
0332 <sub>16</sub>			
0333 <sub>16</sub>			
0334 <sub>16</sub>	UART2 Special Mode Register 4	U2SMR4	00 <sub>16</sub>
0335 <sub>16</sub>	UART2 Special Mode Register 3	U2SMR3	00 <sub>16</sub>
0336 <sub>16</sub>	UART2 Special Mode Register 2	U2SMR2	00 <sub>16</sub>
0337 <sub>16</sub>	UART2 Special Mode Register	U2SMR	00 <sub>16</sub>
0338 <sub>16</sub>	UART2 Transmit/Receive Mode Register	U2MR	00 <sub>16</sub>
0339 <sub>16</sub>	UART2 Bit Rate Register	U2BRG	XX <sub>16</sub>
033A <sub>16</sub> 033B <sub>16</sub>	UART2 Transmit Buffer Register	U2TB	XX <sub>16</sub> XX <sub>16</sub>
033C <sub>16</sub>	UART2 Transmit/Receive Control Register 0	U2C0	0000 1000 <sub>2</sub>
033D <sub>16</sub>	UART2 Transmit/Receive Control Register 1	U2C1	0000 0010 <sub>2</sub>
033E <sub>16</sub> 033F <sub>16</sub>	UART2 Receive Buffer Register	U2RB	XX <sub>16</sub> XX <sub>16</sub>

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0340 <sub>16</sub>	Count Start Flag	TABSR	00 <sub>16</sub>
0341 <sub>16</sub>	Clock Prescaler Reset Flag	CPSRF	0XXX XXXX <sub>2</sub>
0342 <sub>16</sub>	One-Shot Start Flag	ONSF	00 <sub>16</sub>
0343 <sub>16</sub>	Trigger Select Register	TRGSR	00 <sub>16</sub>
0344 <sub>16</sub>	Up/Down Flag	UDF	00 <sub>16</sub>
0345 <sub>16</sub>			
0346 <sub>16</sub> 0347 <sub>16</sub>	Timer A0 Register	TA0	XX <sub>16</sub> XX <sub>16</sub>
0348 <sub>16</sub> 0349 <sub>16</sub>	Timer A1 Register	TA1	XX <sub>16</sub> XX <sub>16</sub>
034A <sub>16</sub> 034B <sub>16</sub>	Timer A2 Register	TA2	XX <sub>16</sub> XX <sub>16</sub>
034C <sub>16</sub> 034D <sub>16</sub>	Timer A3 Register	TA3	XX <sub>16</sub> XX <sub>16</sub>
034E <sub>16</sub> 034F <sub>16</sub>	Timer A4 Register	TA4	XX <sub>16</sub> XX <sub>16</sub>
0350 <sub>16</sub> 0351 <sub>16</sub>	Timer B0 Register	TB0	XX <sub>16</sub> XX <sub>16</sub>
0352 <sub>16</sub> 0353 <sub>16</sub>	Timer B1 Register	TB1	XX <sub>16</sub> XX <sub>16</sub>
0354 <sub>16</sub> 0355 <sub>16</sub>	Timer B2 Register	TB2	XX <sub>16</sub> XX <sub>16</sub>
0356 <sub>16</sub>	Timer A0 Mode Register	TA0MR	00 <sub>16</sub>
0357 <sub>16</sub>	Timer A1 Mode Register	TA1MR	00 <sub>16</sub>
0358 <sub>16</sub>	Timer A2 Mode Register	TA2MR	00 <sub>16</sub>
0359 <sub>16</sub>	Timer A3 Mode Register	TA3MR	00 <sub>16</sub>
035A <sub>16</sub>	Timer A4 Mode Register	TA4MR	00 <sub>16</sub>
035B <sub>16</sub>	Timer B0 Mode Register	TB0MR	00XX 0000 <sub>2</sub>
035C <sub>16</sub>	Timer B1 Mode Register	TB1MR	00XX 0000 <sub>2</sub>
035D <sub>16</sub>	Timer B2 Mode Register	TB2MR	00XX 0000 <sub>2</sub>
035E <sub>16</sub>	Timer B2 Special Mode Register	TB2SC	XXXX XXX0 <sub>2</sub>
035F <sub>16</sub>	Count Source Prescaler Register <sup>(1)</sup>	TCSPR	0XXX 0000 <sub>2</sub>
0360 <sub>16</sub>			
0361 <sub>16</sub>			
0362 <sub>16</sub>			
0363 <sub>16</sub>			
0364 <sub>16</sub>	UART0 Special Mode Register 4	U0SMR4	00 <sub>16</sub>
0365 <sub>16</sub>	UART0 Special Mode Register 3	U0SMR3	00 <sub>16</sub>
0366 <sub>16</sub>	UART0 Special Mode Register 2	U0SMR2	00 <sub>16</sub>
0367 <sub>16</sub>	UART0 Special Mode Register	U0SMR	00 <sub>16</sub>
0368 <sub>16</sub>	UART0 Transmit/Receive Mode Register	U0MR	00 <sub>16</sub>
0369 <sub>16</sub>	UART0 Bit Rate Register	U0BRG	XX <sub>16</sub>
036A <sub>16</sub> 036B <sub>16</sub>	UART0 Transmit Buffer Register	U0TB	XX <sub>16</sub> XX <sub>16</sub>
036C <sub>16</sub>	UART0 Transmit/Receive Control Register 0	U0C0	0000 1000 <sub>2</sub>
036D <sub>16</sub>	UART0 Transmit/Receive Control Register 1	U0C1	0000 0010 <sub>2</sub>
036E <sub>16</sub> 036F <sub>16</sub>	UART0 Receive Buffer Register	U0RB	XX <sub>16</sub> XX <sub>16</sub>

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTE:

1. The TCSPR register maintains values set before reset, even after software reset or watchdog timer reset has been performed.

Address	Register	Symbol	Value after RESET
0370 <sub>16</sub>			
0371 <sub>16</sub>			
0372 <sub>16</sub>			
0373 <sub>16</sub>			
0374 <sub>16</sub>			
0375 <sub>16</sub>			
0376 <sub>16</sub>			
0377 <sub>16</sub>			
0378 <sub>16</sub>	DMA0 Request Source Select Register	DM0SL	0X00 0000 <sub>2</sub>
0379 <sub>16</sub>	DMA1 Request Source Select Register	DM1SL	0X00 0000 <sub>2</sub>
037A <sub>16</sub>	DMA2 Request Source Select Register	DM2SL	0X00 0000 <sub>2</sub>
037B <sub>16</sub>	DMA3 Request Source Select Register	DM3SL	0X00 0000 <sub>2</sub>
037C <sub>16</sub>	CRC Data Register	CRCD	XX <sub>16</sub>
037D <sub>16</sub>			XX <sub>16</sub>
037E <sub>16</sub>	CRC Input Register	CRCIN	XX <sub>16</sub>
037F <sub>16</sub>			
0380 <sub>16</sub>	A/D0 Register 0	AD00	XXXX XXXX <sub>2</sub>
0381 <sub>16</sub>			0000 0000 <sub>2</sub>
0382 <sub>16</sub>	A/D0 Register 1	AD01	XX <sub>16</sub>
0383 <sub>16</sub>			XX <sub>16</sub>
0384 <sub>16</sub>	A/D0 Register 2	AD02	XX <sub>16</sub>
0385 <sub>16</sub>			XX <sub>16</sub>
0386 <sub>16</sub>	A/D0 Register 3	AD03	XX <sub>16</sub>
0387 <sub>16</sub>			XX <sub>16</sub>
0388 <sub>16</sub>	A/D0 Register 4	AD04	XX <sub>16</sub>
0389 <sub>16</sub>			XX <sub>16</sub>
038A <sub>16</sub>	A/D0 Register 5	AD05	XX <sub>16</sub>
038B <sub>16</sub>			XX <sub>16</sub>
038C <sub>16</sub>	A/D0 Register 6	AD06	XX <sub>16</sub>
038D <sub>16</sub>			XX <sub>16</sub>
038E <sub>16</sub>	A/D0 Register 7	AD07	XX <sub>16</sub>
038F <sub>16</sub>			XX <sub>16</sub>
0390 <sub>16</sub>			
0391 <sub>16</sub>			
0392 <sub>16</sub>			
0393 <sub>16</sub>			
0394 <sub>16</sub>	A/D0 Control Register 2	AD0CON2	XX0X X000 <sub>2</sub>
0395 <sub>16</sub>	A/D0 Control Register 3	AD0CON3	XXXX X000 <sub>2</sub>
0396 <sub>16</sub>	A/D0 Control Register 0	AD0CON0	00 <sub>16</sub>
0397 <sub>16</sub>	A/D0 Control Register 1	AD0CON1	00 <sub>16</sub>
0398 <sub>16</sub>	D/A Register 0	DA0	XX <sub>16</sub>
0399 <sub>16</sub>			
039A <sub>16</sub>	D/A Register 1	DA1	XX <sub>16</sub>
039B <sub>16</sub>			
039C <sub>16</sub>	D/A Control Register	DACON	XXXX XX00 <sub>2</sub>
039D <sub>16</sub>			
039E <sub>16</sub>			
039F <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
03A0 <sub>16</sub>			
03A1 <sub>16</sub>			
03A2 <sub>16</sub>			
03A3 <sub>16</sub>			
03A4 <sub>16</sub>			
03A5 <sub>16</sub>			
03A6 <sub>16</sub>			
03A7 <sub>16</sub>	Function Select Register D1	PSD1	X0XX XX00 <sub>2</sub>
03A8 <sub>16</sub>			
03A9 <sub>16</sub>			
03AA <sub>16</sub>			
03AB <sub>16</sub>			
03AC <sub>16</sub>			
03AD <sub>16</sub>	Function Select Register C3	PSC3	X0XX XXXX <sub>2</sub>
03AE <sub>16</sub>			
03AF <sub>16</sub>	Function Select Register C	PSC	00X0 0000 <sub>2</sub>
03B0 <sub>16</sub>	Function Select Register A0	PS0	00 <sub>16</sub>
03B1 <sub>16</sub>	Function Select Register A1	PS1	00 <sub>16</sub>
03B2 <sub>16</sub>	Function Select Register B0	PSL0	00 <sub>16</sub>
03B3 <sub>16</sub>	Function Select Register B1	PSL1	00 <sub>16</sub>
03B4 <sub>16</sub>	Function Select Register A2	PS2	00X0 0000 <sub>2</sub>
03B5 <sub>16</sub>	Function Select Register A3	PS3	00 <sub>16</sub>
03B6 <sub>16</sub>	Function Select Register B2	PSL2	00X0 0000 <sub>2</sub>
03B7 <sub>16</sub>	Function Select Register B3	PSL3	00 <sub>16</sub>
03B8 <sub>16</sub>			
03B9 <sub>16</sub>			
03BA <sub>16</sub>			
03BB <sub>16</sub>			
03BC <sub>16</sub>			
03BD <sub>16</sub>			
03BE <sub>16</sub>			
03BF <sub>16</sub>			
03C0 <sub>16</sub>	Port P6 Register	P6	XX <sub>16</sub>
03C1 <sub>16</sub>	Port P7 Register	P7	XX <sub>16</sub>
03C2 <sub>16</sub>	Port P6 Direction Register	PD6	00 <sub>16</sub>
03C3 <sub>16</sub>	Port P7 Direction Register	PD7	00 <sub>16</sub>
03C4 <sub>16</sub>	Port P8 Register	P8	XX <sub>16</sub>
03C5 <sub>16</sub>	Port P9 Register	P9	XX <sub>16</sub>
03C6 <sub>16</sub>	Port P8 Direction Register	PD8	00X0 0000 <sub>2</sub>
03C7 <sub>16</sub>	Port P9 Direction Register	PD9	00 <sub>16</sub>
03C8 <sub>16</sub>	Port P10 Register	P10	XX <sub>16</sub>
03C9 <sub>16</sub>			
03CA <sub>16</sub>	Port P10 Direction Register	PD10	00 <sub>16</sub>
03CB <sub>16</sub>			
03CC <sub>16</sub>			
03CD <sub>16</sub>			
03CE <sub>16</sub>			
03CF <sub>16</sub>			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
03D0 <sub>16</sub>			
03D1 <sub>16</sub>			
03D2 <sub>16</sub>			
03D3 <sub>16</sub>			
03D4 <sub>16</sub>			
03D5 <sub>16</sub>			
03D6 <sub>16</sub>			
03D7 <sub>16</sub>			
03D8 <sub>16</sub>			
03D9 <sub>16</sub>			
03DA <sub>16</sub>	Pull-Up Control Register 2	PUR2	00 <sub>16</sub>
03DB <sub>16</sub>	Pull-Up Control Register 3	PUR3	00 <sub>16</sub>
03DC <sub>16</sub>			
03DD <sub>16</sub>			
03DE <sub>16</sub>			
03DF <sub>16</sub>			
03E0 <sub>16</sub>	Port P0 Register <sup>(1)</sup>	P0	XX <sub>16</sub>
03E1 <sub>16</sub>	Port P1 Register <sup>(1)</sup>	P1	XX <sub>16</sub>
03E2 <sub>16</sub>	Port P0 Direction Register <sup>(1)</sup>	PD0	00 <sub>16</sub>
03E3 <sub>16</sub>	Port P1 Direction Register <sup>(1)</sup>	PD1	00 <sub>16</sub>
03E4 <sub>16</sub>	Port P2 Register <sup>(1)</sup>	P2	XX <sub>16</sub>
03E5 <sub>16</sub>	Port P3 Register <sup>(1)</sup>	P3	XX <sub>16</sub>
03E6 <sub>16</sub>	Port P2 Direction Register <sup>(1)</sup>	PD2	00 <sub>16</sub>
03E7 <sub>16</sub>	Port P3 Direction Register <sup>(1)</sup>	PD3	00 <sub>16</sub>
03E8 <sub>16</sub>	Port P4 Register <sup>(1)</sup>	P4	XX <sub>16</sub>
03E9 <sub>16</sub>	Port P5 Register <sup>(1)</sup>	P5	XX <sub>16</sub>
03EA <sub>16</sub>	Port P4 Direction Register <sup>(1)</sup>	PD4	00 <sub>16</sub>
03EB <sub>16</sub>	Port P5 Direction Register <sup>(1)</sup>	PD5	00 <sub>16</sub>
03EC <sub>16</sub>			
03ED <sub>16</sub>			
03EE <sub>16</sub>			
03EF <sub>16</sub>			
03F0 <sub>16</sub>	Pull-up Control Register 0	PUR0	00 <sub>16</sub>
03F1 <sub>16</sub>	Pull-up Control Register 1	PUR1	XXXX 0000 <sub>2</sub>
03F2 <sub>16</sub>			
03F3 <sub>16</sub>			
03F4 <sub>16</sub>			
03F5 <sub>16</sub>			
03F6 <sub>16</sub>			
03F7 <sub>16</sub>			
03F8 <sub>16</sub>			
03F9 <sub>16</sub>			
03FA <sub>16</sub>			
03FB <sub>16</sub>			
03FC <sub>16</sub>			
03FD <sub>16</sub>			
03FE <sub>16</sub>			
03FF <sub>16</sub>	Port Control Register	PCR	XXXX XXX0 <sub>2</sub>

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTE:

1. Pins, functioning as bus control pins, cannot be selected as I/O ports.



## 5. Reset

Hardware reset 1, software reset, and watchdog timer reset are available to reset the microcomputer.

### 5.1 Hardware Reset 1

Pins, the CPU and SFRs are reset by setting the  $\overline{\text{RESET}}$  pin. If the supply voltage meets the recommended operating conditions, all pins are reset when a low-level ("L") signal is applied to the  $\overline{\text{RESET}}$  pin (see **Table 5.1**). The oscillation circuit is also reset and the main clock starts oscillating. The CPU and SFR are reset when the signal applied to the  $\overline{\text{RESET}}$  pin changes "L" to high level ("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.

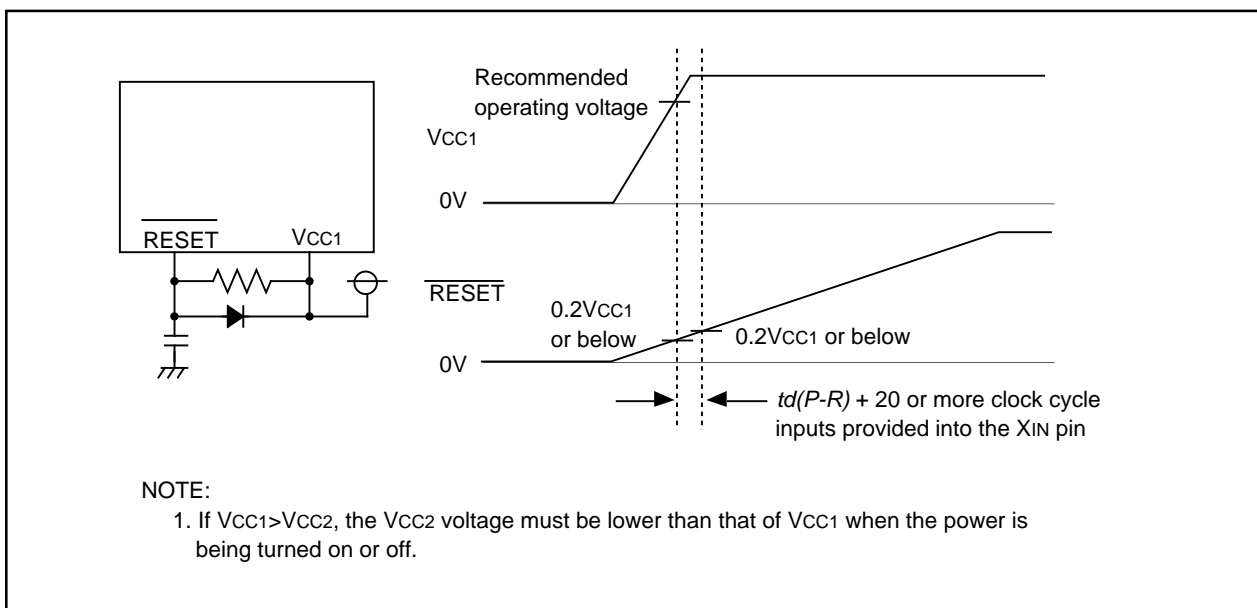
Figure 5.1 shows an example of the reset circuit. Figure 5.2 shows a reset sequence. Table 5.1 lists pin states while the  $\overline{\text{RESET}}$  pin is held "L".

#### 5.1.1 Reset on a Stable Supply Voltage

- (1) Apply an "L" signal to the  $\overline{\text{RESET}}$  pin
- (2) Provide 20 or more clock cycle inputs into the XIN pin
- (3) Apply an "H" signal to the  $\overline{\text{RESET}}$  pin

#### 5.1.2 Power-on Reset

- (1) Apply an "L" signal to the  $\overline{\text{RESET}}$  pin
- (2) Raise the supply voltage to the recommended operating level
- (3) Insert  $td(P-R)$  ms as wait time for the internal voltage to stabilize
- (4) Provide 20 or more clock cycle inputs into the XIN pin
- (5) Apply an "H" signal to the  $\overline{\text{RESET}}$  pin



**Figure 5.1** Reset Circuit

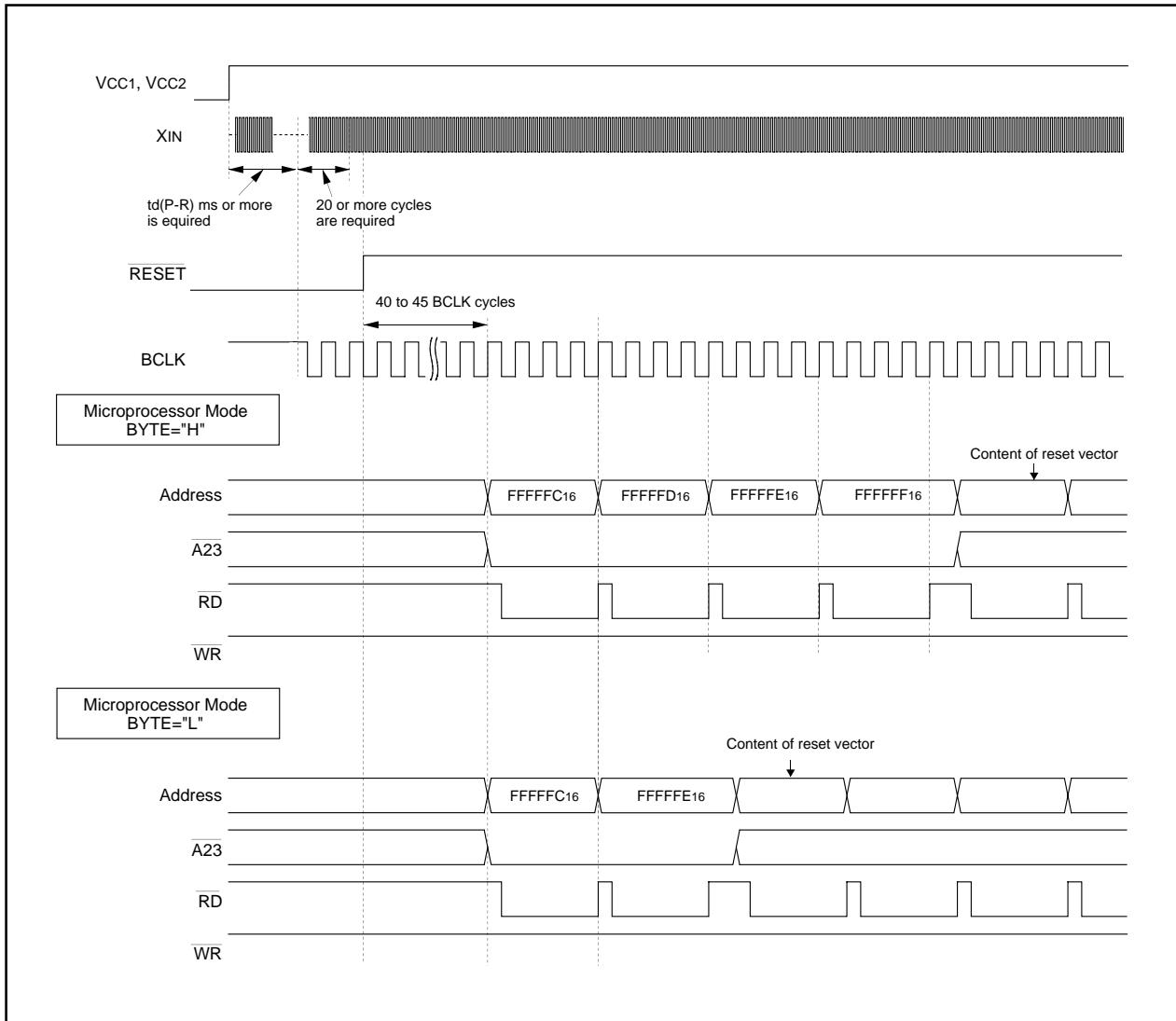


Figure 5.2 Reset Sequence

**Table 5.1 Pin States while  $\overline{\text{RESET}}$  Pin is Held "L"**

Pin Name	Pin States <sup>(1)</sup>		
	CNVss=Vss	CNVss=Vcc	
		BYTE=Vss	BYTE=Vcc
P0	Input port (high-impedance)	Inputs data (high-impedance)	
P1	Input port (high-impedance)	Inputs data (high-impedance)	Input port (high-impedance)
P2, P3, P4	Input port (high-impedance)	Output addresses (indeterminate)	
P50	Input port (high-impedance)	Outputs the $\overline{\text{WR}}$ signal ("H") <sup>(2)</sup>	
P51	Input port (high-impedance)	Outputs the $\overline{\text{BHE}}$ signal (indeterminate)	
P52	Input port (high-impedance)	Outputs the $\overline{\text{RD}}$ signal ("H") <sup>(2)</sup>	
P53	Input port (high-impedance)	Outputs the BCLK <sup>(2)</sup>	
P54	Input port (high-impedance)	Outputs the $\overline{\text{HLDA}}$ signal (Output signal depends on an input signal to the $\overline{\text{HOLD}}$ pin) <sup>(2)</sup>	
P55	Input port (high-impedance)	Inputs the HOLD signal (high-impedance)	
P56	Input port (high-impedance)	Outputs an "H" signal <sup>(2)</sup>	
P57	Input port (high-impedance)	Inputs the RDY signal (high-impedance)	
P6 to P10	Input port (high-impedance)	Input port (high-impedance)	

## NOTES:

1. The availability of pull-up resistors is indeterminate until internal supply voltage stabilizes.
2. Each port is in this state after power is on and internal supply voltage stabilizes, but in an indeterminate state until internal supply voltage stabilizes.

### 5.2 Software Reset

Pins, the CPU and SFRs are reset 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 SFR. Refer to **4. Special Function Registers (SFRs)** for details. Processor mode remains unchanged since the PM01 and PM00 bits in the PM0 register are not reset.

### 5.3 Watchdog Timer Reset

Pins, the CPU and SFRs are reset 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 Registers (SFRs)** for details. Processor mode remains unchanged since the PM01 and PM00 bits in the PM0 register are not reset.

### 5.4 Internal Space

Figure 5.3 shows CPU register states after reset. Refer to **4. Special Function Registers (SFRs)** for SFR states after reset.

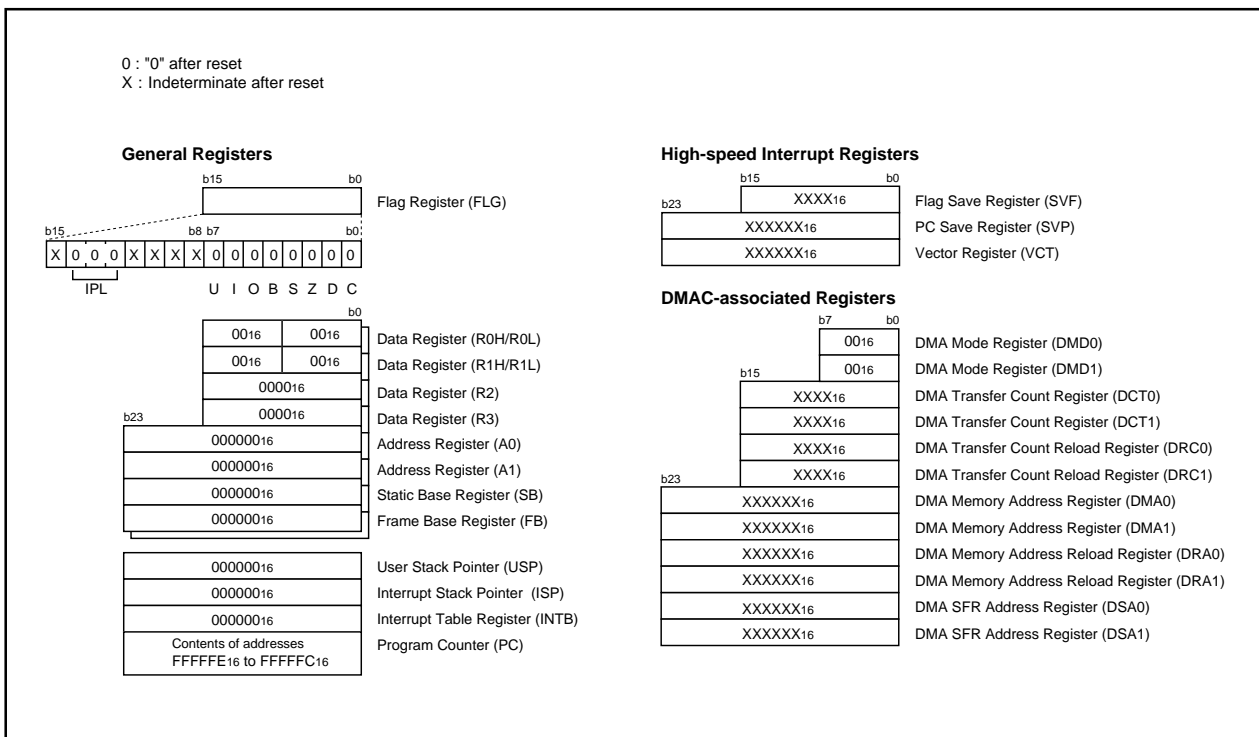


Figure 5.3 CPU Register States after Reset

## 6. Processor Mode

### NOTE

The M32C/80 Group is ROMless device. Connect the CNVss pin to VCC1.  
Use the M32C/80 Group in microprocessor mode after reset.

### 6.1 Types of Processor Mode

Single-chip mode, memory expansion mode, or microprocessor mode can be selected as a processor mode. Table 6.1 lists a feature of the processor mode.

**Table 6.1 Processor Mode Feature**

Processor Mode	Accessable Space	Pin Status as I/O Ports
Single-chip Mode	SFRs, Internal RAM	All pins assigned to I/O ports or to I/O pins for the peripheral functions
Memory Expansion Mode	SFRs, Internal RAM, External Space <sup>(1)</sup>	Some pins assigned to bus control pins <sup>(1)</sup>
Microprocessor Mode	SFRs, Internal RAM, External Space <sup>(1)</sup>	Some pins assigned to bus control pins <sup>(1)</sup>

#### NOTE:

1. Refer to **7. Bus** for details.

## 6.2 Setting of Processor Mode

The CNVss pin state and the PM01 and PM00 bit settings in the PM0 register determine which processor mode is selected. Table 6.2 lists processor mode after hardware reset. Table 6.3 lists processor mode selected by PM01 and PM00 bit settings.

**Table 6.2 Processor Mode after Hardware Reset**

Input Level into the CNVss pin	Processor Mode
Vcc1 <sup>(1)</sup>	Microprocessor Mode

**NOTE:**

1. Multiplex bus cannot be assigned to all  $\overline{CS}$  areas.

**Table 6.3 Processor Mode Selected by the PM01 and PM00 bit Settings**

PM01 and PM00 Bits	Processor Mode
002	Single-chip Mode
012	Memory Expansion Mode
102	Do not set to this value
112	Microprocessor Mode

If the PM01 and PM00 bits are rewritten, the PM01 and PM00 bits select a mode regardless of CNVss pin level.

Do not change the PM01 and PM00 bits to "012" (memory expansion mode) or "112" (microprocessor mode) when the PM07 to PM02 bits in the PM0 register are being rewritten.

Figures 6.1 and 6.2 show the PM0 register and PM1 register. Figure 6.3 shows a memory map in each processor mode.

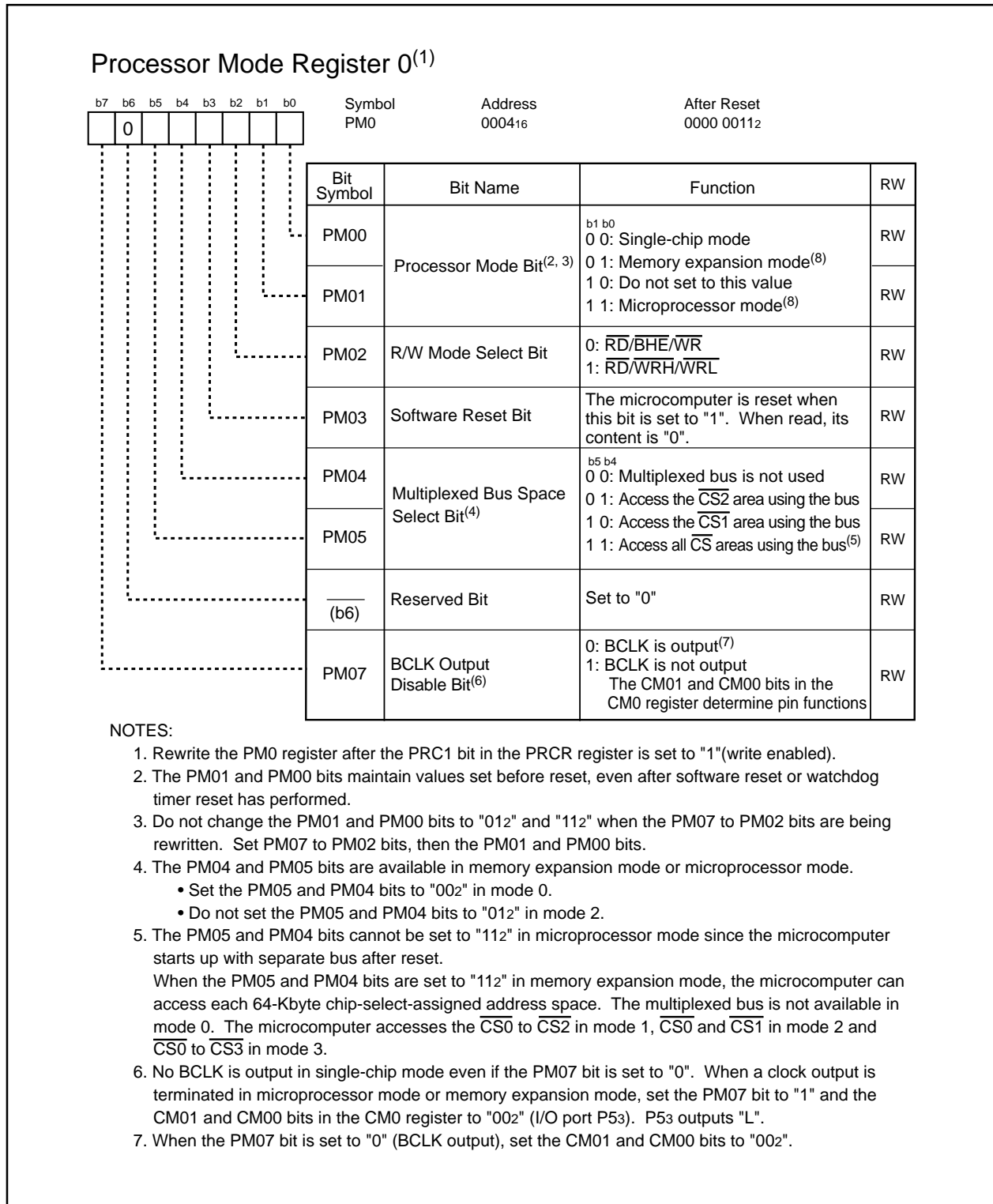


Figure 6.1 PM0 Register

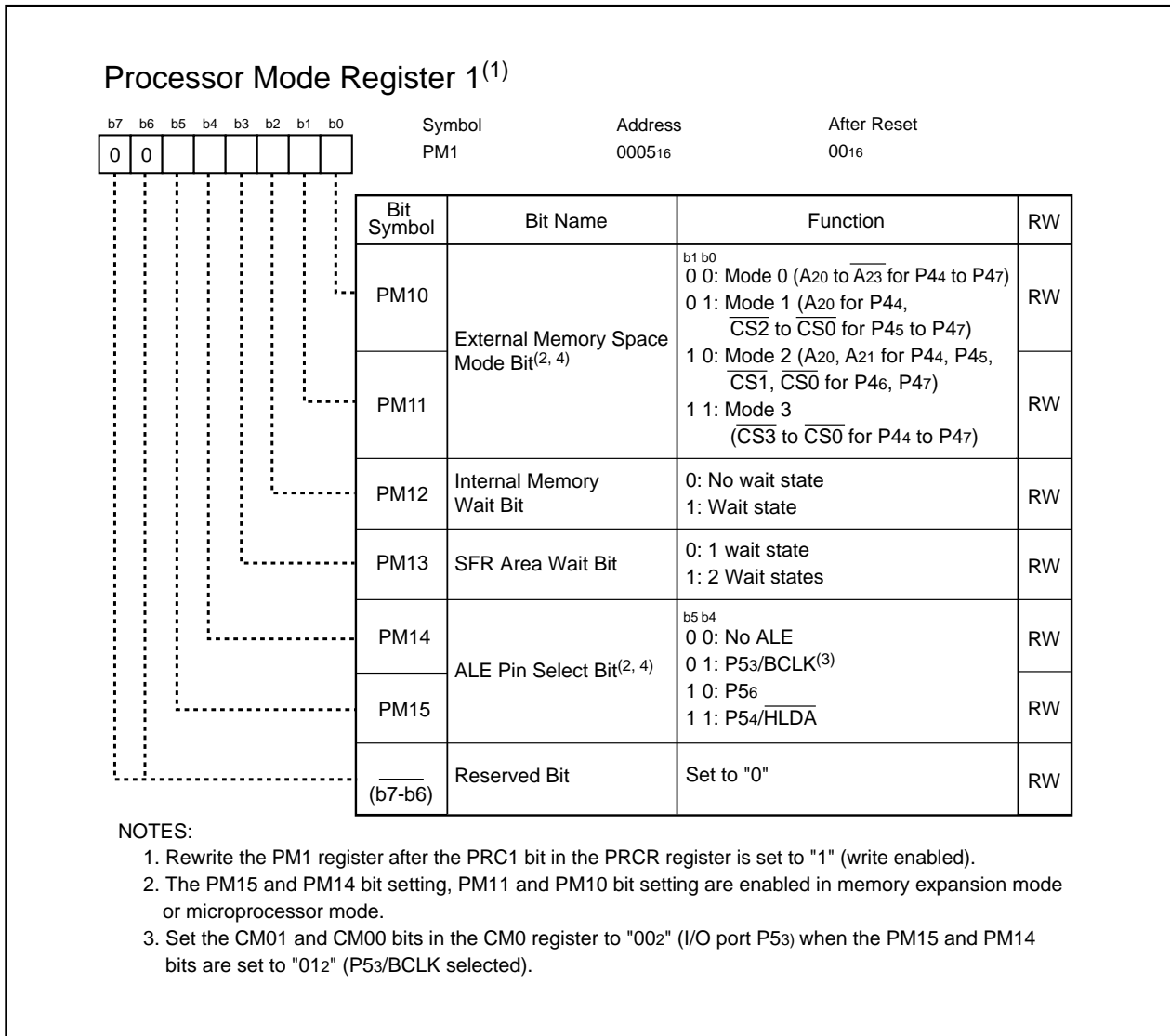


Figure 6.2 PM1 Register



Single-Chip Mode	Memory Expansion Mode				Microprocessor Mode			
	Mode 0	Mode 1	Mode 2	Mode 3	Mode 0	Mode 1	Mode 2	Mode 3
	SFRs Internal RAM	SFRs Internal RAM	SFRs Internal RAM	SFRs Internal RAM	SFRs Internal RAM	SFRs Internal RAM	SFRs Internal RAM	SFRs Internal RAM
000000 <sub>16</sub>	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space
000400 <sub>16</sub>	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space
010000 <sub>16</sub>	External Space 0	External Space 0	External Space 0	Not Used	External Space 0	External Space 0	External Space 0	Not Used
100000 <sub>16</sub>	External Space 0	CS1 2 Mbytes <sup>(1)</sup> External Space 0	CS1 4 Mbytes <sup>(2)</sup> External Space 0	CS1 1 Mbyte External Space 0	External Space 0	CS1 2 Mbytes <sup>(1)</sup> External Space 0	CS1 4 Mbytes <sup>(2)</sup> External Space 0	CS1 1 Mbyte External Space 0
200000 <sub>16</sub>	External Space 1	CS2 2 Mbytes External Space 1	External Space 0	CS2 1 Mbyte External Space 1	External Space 1	CS2 2 Mbytes External Space 1	External Space 1	CS2 1 Mbyte External Space 1
300000 <sub>16</sub>	External Space 1	External Space 1	External Space 1	External Space 1	External Space 1	External Space 1	External Space 1	External Space 1
400000 <sub>16</sub>	External Space 2	External Space 2	External Space 2	Not Used	External Space 2	External Space 2	External Space 2	Not Used
C00000 <sub>16</sub>	External Space 2	External Space 2	External Space 2	External Space 2	External Space 2	External Space 2	External Space 2	External Space 2
D00000 <sub>16</sub>	External Space 3	CS0 2 Mbytes External Space 3	CS0 3 Mbytes External Space 3	CS3 1 Mbyte External Space 2	External Space 3	Not Used	CS0 4 Mbytes External Space 3	CS3 1 Mbyte External Space 2
E00000 <sub>16</sub>	External Space 3	Not Used	External Space 3	Not Used	External Space 3	CS0 2 Mbytes External Space 3	External Space 3	Not Used
F00000 <sub>16</sub>	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	CS0 1 Mbyte External Space 3
FFFFFF <sub>16</sub>	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	Reserved Space	CS0 1 Mbyte External Space 3

NOTES:  
 1. 200000<sub>16</sub> - 010000<sub>16</sub>=1984 Kbytes. 64K bytes less than 2 Mbytes.  
 2. 400000<sub>16</sub> - 010000<sub>16</sub>=4032 Kbytes. 64K bytes less than 4 Mbytes.

Figure 6.3 Memory Map in Each Processor Mode

# 7. Bus

In memory expansion mode or microprocessor mode, some pins function as bus control pins to control the address bus and data bus. A<sub>0</sub> to A<sub>22</sub>,  $\overline{A}_{23}$ , D<sub>0</sub> to D<sub>15</sub>, CS<sub>0</sub> to CS<sub>3</sub>, WRL/WR, WRH/BHE, RD, BCLK/ALE,  $\overline{HLDA/ALE}$ ,  $\overline{HOLD}$ , ALE,  $\overline{RDY}$  are used as bus control pins.

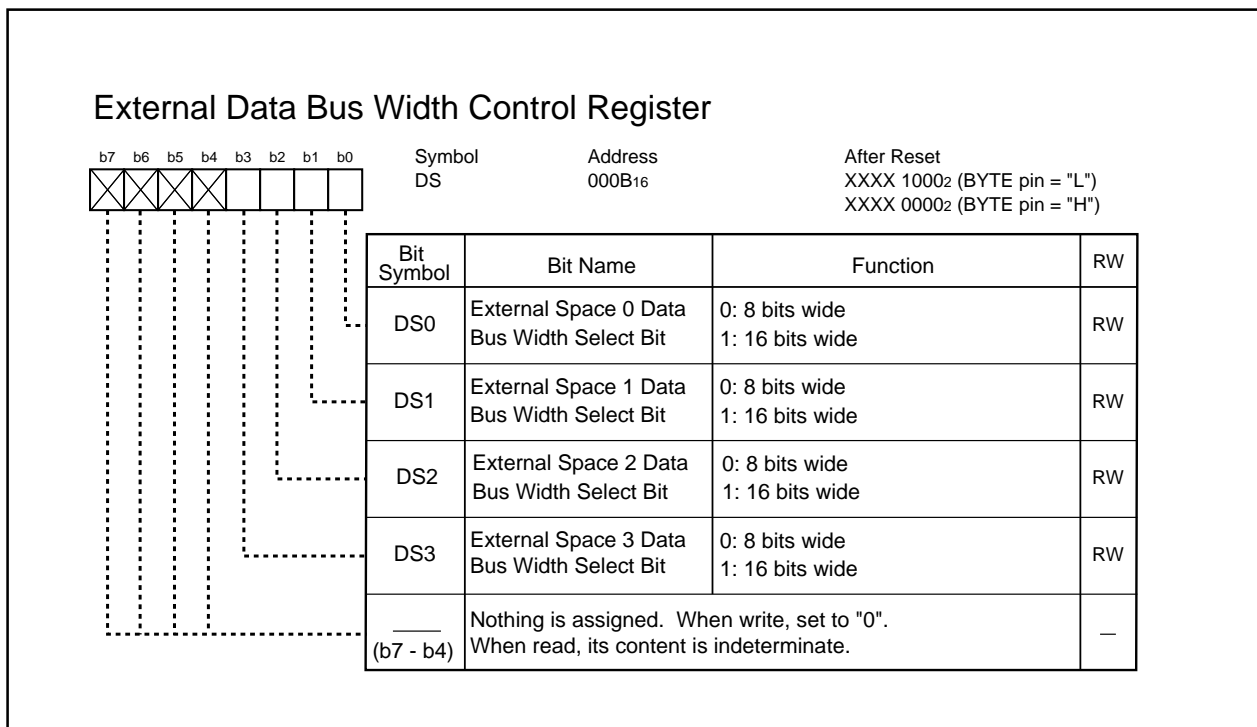
## 7.1 Bus Settings

The BYTE pin, the DS register, the PM05 and PM04 bits in the PM0 register, and the PM11 and PM10 bits in the PM1 register determine bus settings.

Table 7.1 lists how to change bus settings. Figure 7.1 shows the DS register.

**Table 7.1 Bus Settings**

Bus Setting	Changed By
Selecting External Address Bus Width	DS register
Setting Bus Width after Reset	BYTE pin (external space 3 only)
Selecting Between Separate Bus or Multiplexed Bus	PM05 and PM04 bits in PM0 register
Number of Chip-select	PM11 and PM10 bits in PM1 register



**Figure 7.1 DS Register**

### 7.1.1 Selecting External Address Bus

The number of externally-output address buses, the number of chip-select signals and chip-select-assigned address space ( $\overline{CS}$  area) vary depending on each external space mode. The PM11 and PM10 bits in the PM1 register determine the external space mode.

### 7.1.2 Selecting External Data Bus

The DS register selects either external 8-bit or 16-bit data bus per external space. The data bus in the external space 3, after reset, becomes 16 bits wide when a low-level ("L") signal is applied to the BYTE pin and 8 bits wide when a high-level ("H") signal is applied. Keep the BYTE pin input level while the microcomputer is operating. Internal bus is always 16 bits wide.

### 7.1.3 Selecting Separate/Multiplexed Bus

The PM05 and PM04 bits in the PM0 register determine either separate or multiplexed bus as bus format.

#### 7.1.3.1 Separate Bus

The separate bus is a bus format which allows the microcomputer to input and output data and address separately. The DS register selects 8-bit or 16-bit data bus as the external data bus per external space. If all DSi bits in the DS register ( $i=0$  to 3) are set to "0" (8-bit data bus), port P0 becomes the data bus and port P1, the programmable I/O port. If one of the DSi bits is set to "1" (16-bit data bus), ports P0 and P1 become the data bus. Port P1 is indeterminate when the microcomputer accesses a space where the DSi bit is set to "0".

The EWCRi register ( $i=0$  to 3) determines the number of software wait states inserted, when the microcomputer accesses space using the separate bus.

#### 7.1.3.2 Multiplexed Bus

The multiplexed bus is a bus format which allow the microcomputer to input and output data and address by timesharing. D0 to D7 are multiplexed with A0 to A7 in space accessed by the 8-bit data bus. D0 to D15 are multiplexed with A0 to A15 in space accessed by the 16-bit data bus. The DSi bit controls the data bus width. The EWCRi register ( $i=0$  to 3) controls the number of software wait states inserted, when the microcomputer accesses a space using the multiplexed bus. Refer to **7.2.4 Bus Timing** for details.

The multiplexed bus can be assigned to access the  $\overline{CS1}$  area,  $\overline{CS2}$  area or all  $\overline{CS}$  areas. However, because the microcomputer starts operation using the separate bus after reset, the multiplexed bus cannot be assigned to access all  $\overline{CS}$  areas in microprocessor mode. When the PM05 and PM04 bits in the PM0 register are set to "112" (access all  $\overline{CS}$  areas with the bus), 16 low-order bits, from A0 to A15, of an address are output. See **Table 7.2** for details.

Table 7.2 Processor Mode and Port Function

Processor Mode	Single-Chip Mode	Memory Expansion Mode/ Microprocessor Mode				Memory Expansion Mode	
PM05 to PM04 Bits in PM0 Register		"012", "102" ( Access $\overline{CS1}$ or $\overline{CS2}$ using the Multiplexed Bus Access All Other $\overline{CS}$ Areas using the Separate Bus )		"002" ( Access all $\overline{CS}$ Areas using the Separate Bus )		"112" <sup>(1)</sup> ( Access all $\overline{CS}$ Areas using the Multiplexed Bus )	
Data Bus Width		Access all external space with 8-bit data bus	Access one or more external space with 16-bit data bus	Access all external space with 8-bit data bus	Access one or more external space with 16-bit data bus	Access all external space with 8-bit data bus	Access one or more external space with 16-bit data bus
P00 to P07	I/O port	Data bus D0 to D7	Data bus D0 to D7	Data bus D0 to D7	Data bus D0 to D7	I/O port	I/O port
P10 to P17	I/O port	I/O port	Data bus D8 to D15	I/O port	Data bus D8 to D15	I/O port	I/O port
P20 to P27	I/O port	Address bus Data bus <sup>(2)</sup> A0/D0 to A7/D7	Address bus Data bus <sup>(2)</sup> A0/D0 to A7/D7	Address bus A0 to A7	Address bus A0 to A7	Address bus Data bus A0/D0 to A7/D7	Address bus Data bus A0/D0 to A7/D7
P30 to P37	I/O port	Address bus A8 to A15	Address bus/ Data bus <sup>(2)</sup> A8/D8 to A15/D15	Address bus A8 to A15	Address bus A8 to A15	Address bus A8 to A15	Address bus/ Data bus A8/D8 to A15/D15
P40 to P43	I/O port	Address bus A16 to A19	Address bus A16 to A19	Address bus A16 to A19	Address bus A16 to A19	I/O port	I/O port
P44 to P46	I/O port	$\overline{CS}$ (Chip-select signal) or Address bus (A20 to A22) (Refer to 7.2 Bus Control for details) <sup>(4)</sup>					
P47	I/O port	$\overline{CS}$ (Chip-select signal) or Address bus (A23) (Refer to 7.2 Bus Control for details) <sup>(4)</sup>					
P50 to P53	I/O port	Outputs $\overline{RD}$ , $\overline{WRL}$ , $\overline{WRH}$ and $\overline{BCLK}$ or outputs $\overline{RD}$ , $\overline{BHE}$ , $\overline{WR}$ and $\overline{BCLK}$ (Refer to 7.2 Bus Control for details) <sup>(3)</sup>					
P54	I/O port	$\overline{HDLA}$ <sup>(3)</sup>	$\overline{HDLA}$ <sup>(3)</sup>	$\overline{HDLA}$ <sup>(3)</sup>	$\overline{HDLA}$ <sup>(3)</sup>	$\overline{HDLA}$ <sup>(3)</sup>	$\overline{HDLA}$ <sup>(3)</sup>
P55	I/O port	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$
P56	I/O port	ALE <sup>(3)</sup>	ALE <sup>(3)</sup>	ALE <sup>(3)</sup>	ALE <sup>(3)</sup>	ALE <sup>(3)</sup>	ALE <sup>(3)</sup>
P57	I/O port	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$

## NOTES:

1. The PM05 and PM04 bits cannot be set to "112" (access all  $\overline{CS}$  areas using multiplexed bus) in microprocessor mode because the microcomputer starts operation using the separate bus after reset.  
When the PM05 and PM04 bits are set to "112" in memory expansion mode, the microcomputer accesses 64-Kbyte memory space per chip-select using the address bus .
2. These ports become address buses when accessing space using the separate bus.
3. The PM15 and PM14 bits in the PM1 register determines which pin outputs the ALE signal. The PM02 bit in the PM0 register selects either " $\overline{WRL}$ ,  $\overline{WRH}$ " or " $\overline{BHE}$ ,  $\overline{WR}$ " combination.  
P56 provides an indeterminate output when the PM15 and PM14 bits to "002" (no ALE). It cannot be used as an I/O port.
4. The PM11 and PM10 bits in the PM1 register determine the  $\overline{CS}$  signal and address bus.

## 7.2 Bus Control

Signals, required to access external devices, are provided and software wait states are inserted as follows. The signals are available in memory expansion mode and microprocessor mode only.

### 7.2.1 Address Bus and Data Bus

Address bus is a signal accessing 16-Mbyte space and uses 24 control pins; A0 to A22 and  $\overline{A23}$ .  $\overline{A23}$  is the inversed output signal of the highest-order address bit.

Data bus is a signal for data input and output. The DS register selects an 8-bit data bus from D0 to D7 or a 16-bit data bus from D0 to D15 for each external space. When applying a high-level ("H") signal to the BYTE pin, the data bus accessing the external memory space 3 becomes an 8-bit data bus after reset. When applying a low-level ("L") signal to the BYTE pin, the data bus accessing the external memory space 3 becomes the 16-bit data bus.

When changing single-chip mode to memory expansion mode, the address bus is in an indeterminate state until the microcomputer accesses an external memory space.

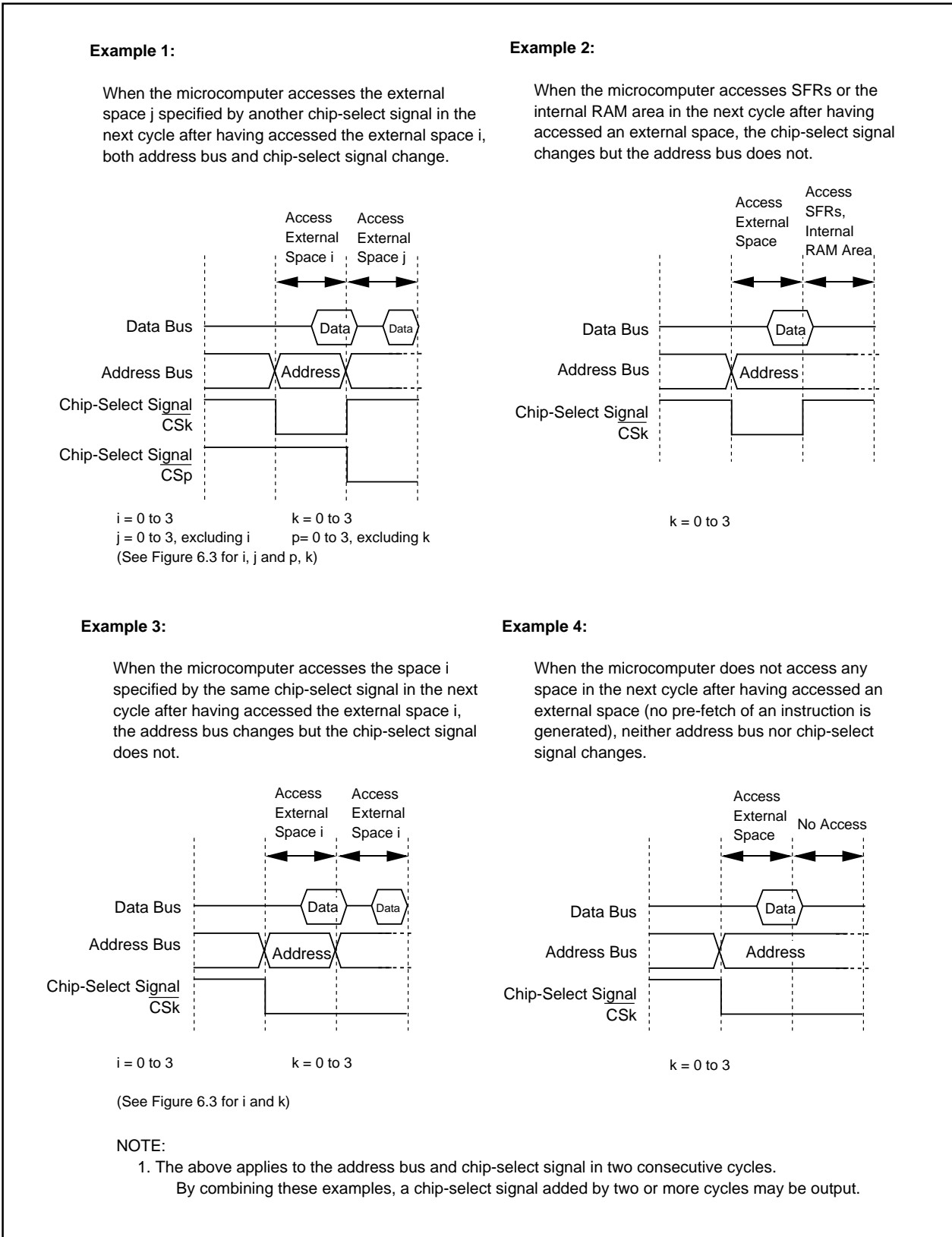
### 7.2.2 Chip-Select Signal

Chip-select signal shares pins with A20 to A22 and  $\overline{A23}$ . The PM11 and PM10 bits in the PM1 register determine which  $\overline{CS}$  area is accessed and how many chip-select signals are output. A maximum of four chip-select signals can be output.

In microprocessor mode, no chip-select signal, aside from  $\overline{A23}$  which can perform as a chip-select signal, is output after reset.

The chip-select signal becomes "L" while the microcomputer is accessing the external  $\overline{CS}_i$  area (i=0 to 3). It becomes "H" while the microcomputer is accessing other external memory space.

Figure 7.2 shows an example of the address bus and chip-select signal output.



**Figure 7.2 Address Bus and Chip-Select Signal Outputs (Separate Bus)**

### 7.2.3 Read and Write Signals

When using a 16-bit data bus, the PM02 bit in the PM0 register selects a combination of the " $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{BHE}$ " signals or the " $\overline{RD}$ ,  $\overline{WRL}$  and  $\overline{WRH}$ " signals to determine the read or write signal. When the DS3 to DS0 bits in the DS register are set to "0" (8-bit data bus), set the PM02 bit to "0" ( $\overline{RD}/\overline{WR}/\overline{BHE}$ ). When any of the DS3 to DS0 bits are set to "1" (16-bit data bus) to access an 8-bit space, the combination of " $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{BHE}$ " is automatically selected regardless of the PM02 bit setting. Tables 7.3 and 7.4 list each signal operation.

The  $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{BHE}$  signals are combined for the read or write signal after reset.

When changing the combination of " $\overline{RD}$ ,  $\overline{WRL}$  and  $\overline{WRH}$ ", set the PM02 bit first to write data to an external memory.

**Table 7.3  $\overline{RD}$ ,  $\overline{WRL}$  and  $\overline{WRH}$  Signals**

Data Bus	$\overline{RD}$	$\overline{WRL}$	$\overline{WRH}$	Status of External Data Bus
16 Bits	L	H	H	Read data
	H	L	H	Write 1-byte data to even address
	H	H	L	Write 1-byte data to odd address
	H	L	L	Write data to both even and odd addresses
8 Bits	H	L <sup>(1)</sup>	Not used	Write 1-byte data
	L	H <sup>(1)</sup>	Not used	Read 1-byte data

NOTE:

1. The  $\overline{WR}$  signal is used instead of the  $\overline{WRL}$  signal.

**Table 7.4  $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{BHE}$  Signals**

Data Bus	$\overline{RD}$	$\overline{WR}$	$\overline{BHE}$	A0	Status of External Data Bus
16 Bits	H	L	L	H	Write 1-byte data to odd address
	L	H	L	H	Read 1-byte data from odd address
	H	L	H	L	Write 1-byte data to even address
	L	H	H	L	Read 1-byte data from 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 Bits	H	L	Not used	H / L	Write 1-byte data
	L	H	Not used	H / L	Read 1-byte data

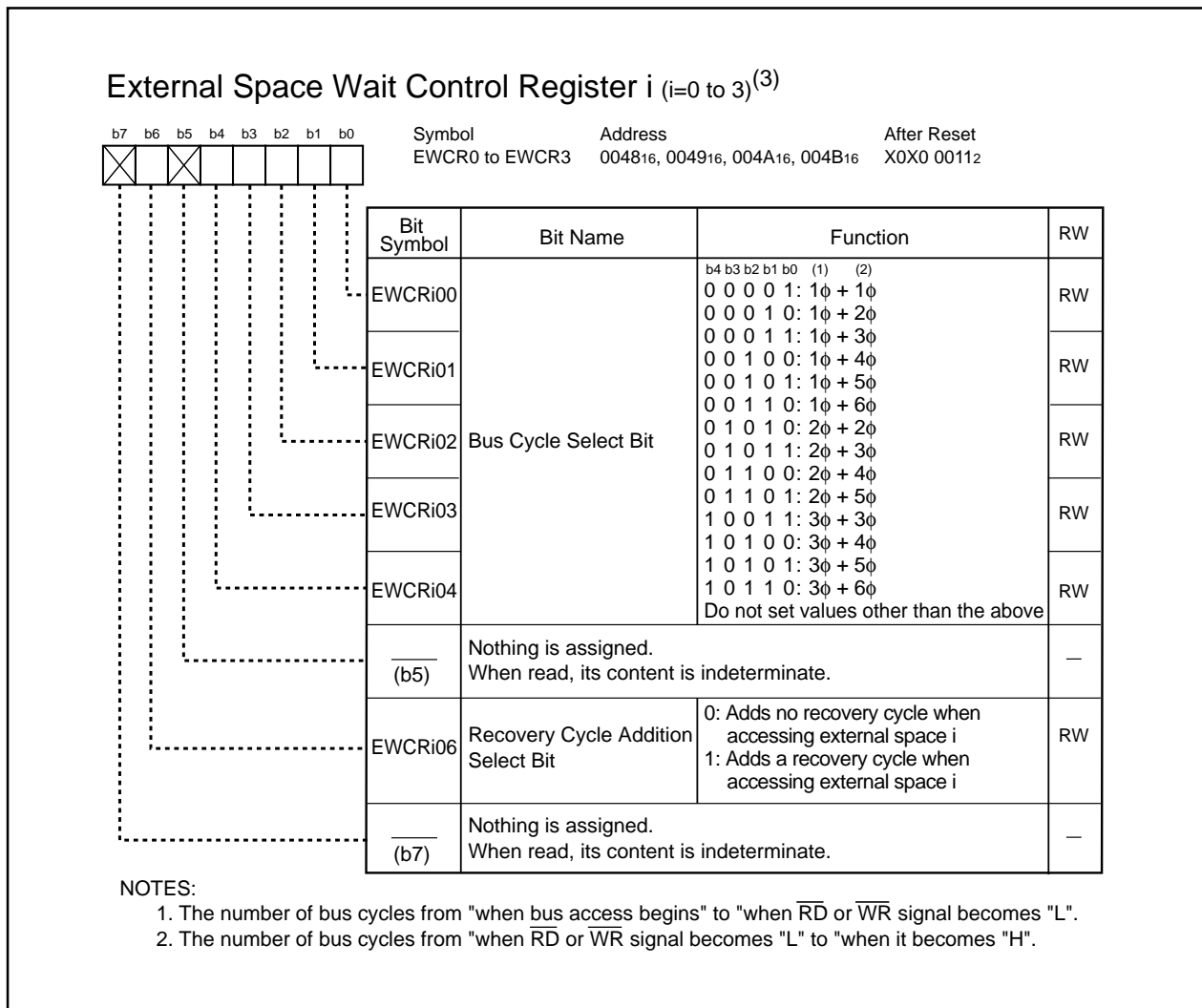
## 7.2.4 Bus Timing

Bus cycle for the internal memory is basically one BCLK cycle. When the PM12 bit in the PM1 register is set to "1" (wait state), the bus cycles are two BCLK cycles.

Bus cycles for SFRs are basically two BCLK cycles. When the PM13 bit in the PM1 register is set to "1" (2 wait states), the bus cycles are three BCLK cycles.

Basic bus cycle for an external space is  $2\phi$  ( $1\phi+1\phi$ ) to read and to write. Bus cycle is selected by the EWCRi register ( $i=0$  to 3) from 12 types of separate bus settings and 7 types of multiplexed bus settings. If the EWCRi04 to EWCRi00 bits are set to "000112" ( $1\phi+3\phi$ ), bus cycles are four BCLK cycles.

Figure 7.3 shows the EWCRi register. Figures 7.4 to 7.8 show bus timing in an external space.



**Figure 7.3 EWCR0 to EWCR3 Registers**



**Table 7.5 Software Wait State and Bus Cycle**

Space	External Bus Status	PM1 Register		EWCRi Register (i=0 to 3)	Bus Cycles	
		PM13 Bit	PM12 Bit	EWCRi04 to EWCRi00 Bits		
SFRs	---	0	---	---	2 BCLK cycles	
		1			3 BCLK cycles	
Internal RAM	---	---	0	--	1 BCLK cycles	
			1		2 BCLK cycles	
External Memory	Separate Bus	---	---	000012	2 BCLK cycles	
				000102	3 BCLK cycles	
				000112	4 BCLK cycles	
				001002	5 BCLK cycles	
				001012	6 BCLK cycles	
				001102	7 BCLK cycles	
				010102	4 BCLK cycles	
				010112	5 BCLK cycles	
				011002	6 BCLK cycles	
				100112	6 BCLK cycles	
				101002	7 BCLK cycles	
				101102	9 BCLK cycles	
	Multiplexed Bus	---	---	---	010102	4 BCLK cycles
					010112	5 BCLK cycles
					011012	7 BCLK cycles
					100112	6 BCLK cycles
					101002	7 BCLK cycles
					101012	8 BCLK cycles
				101102	9 BCLK cycles	

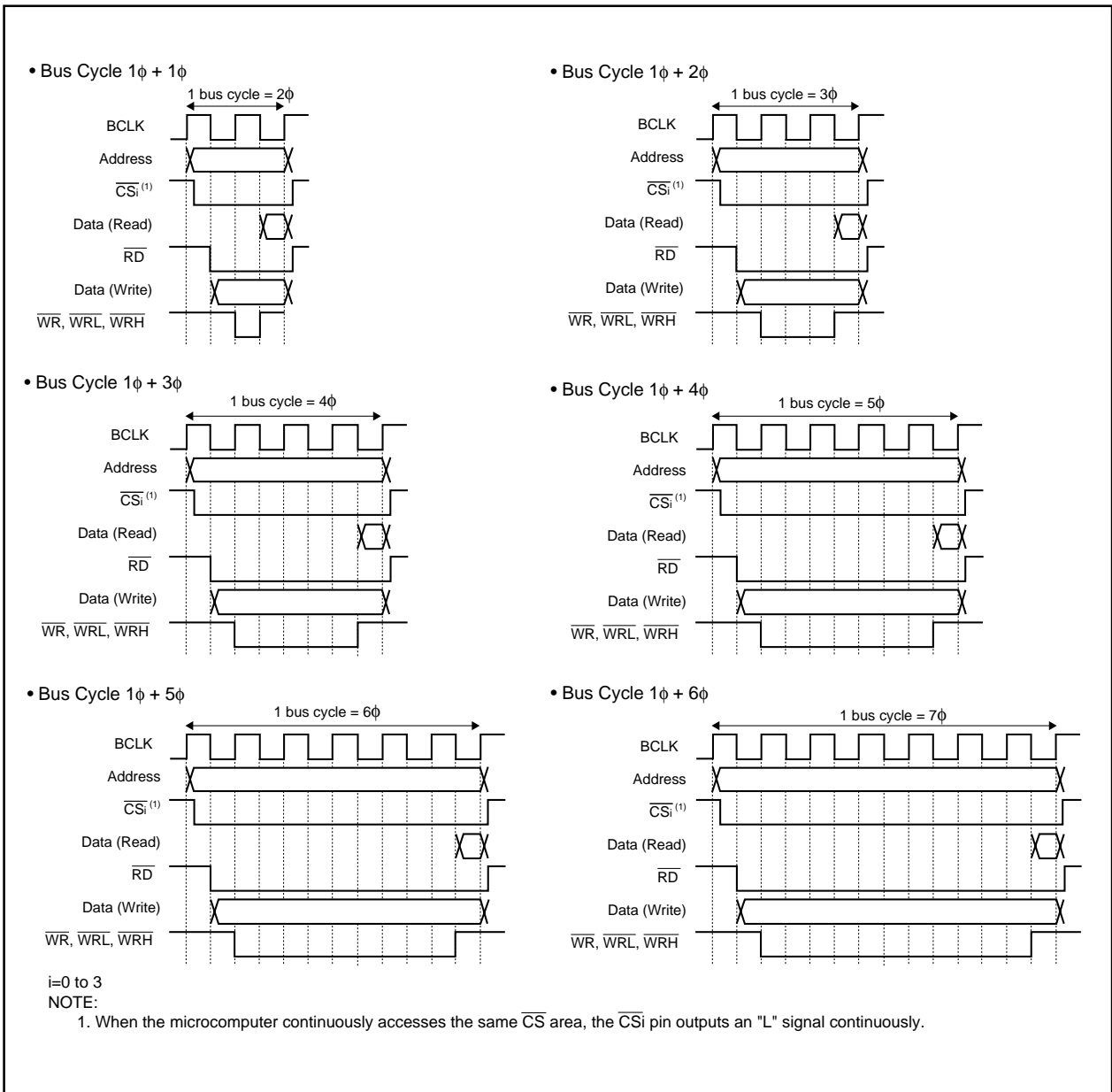


Figure 7.4 Bus Cycle with Separate Bus (1)

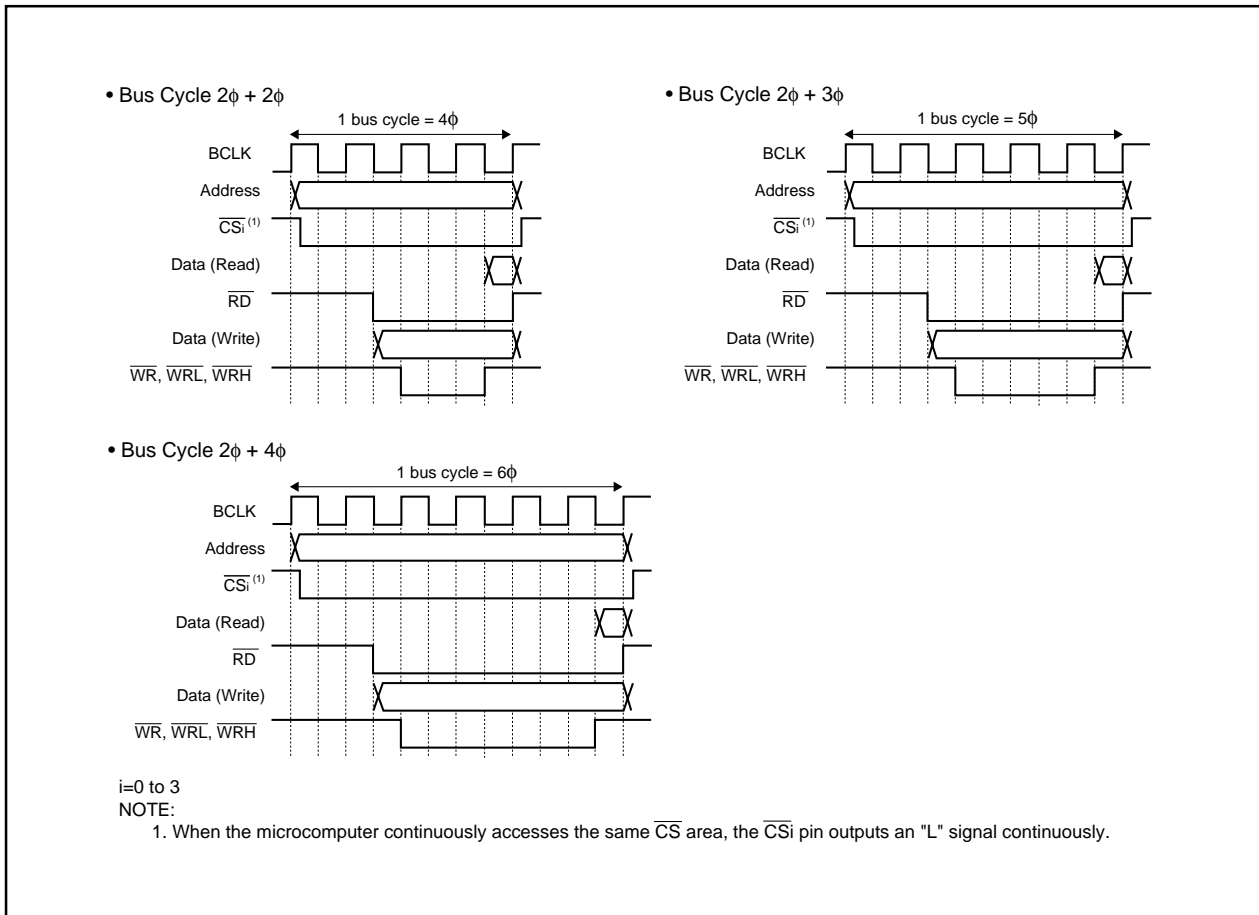


Figure 7.5 Bus Cycle with Separate Bus (2)

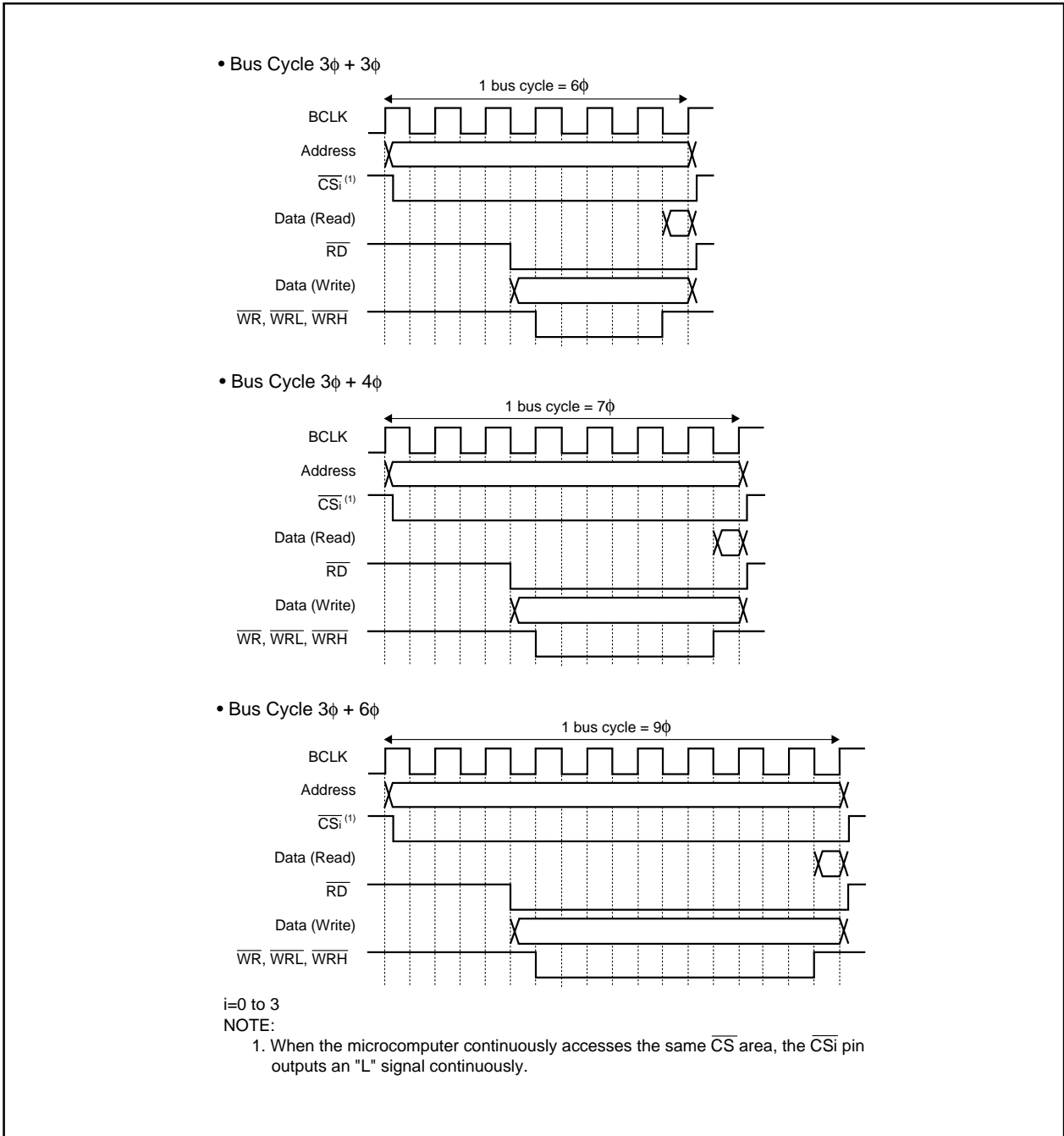


Figure 7.6 Bus Cycle with Separate Bus (3)

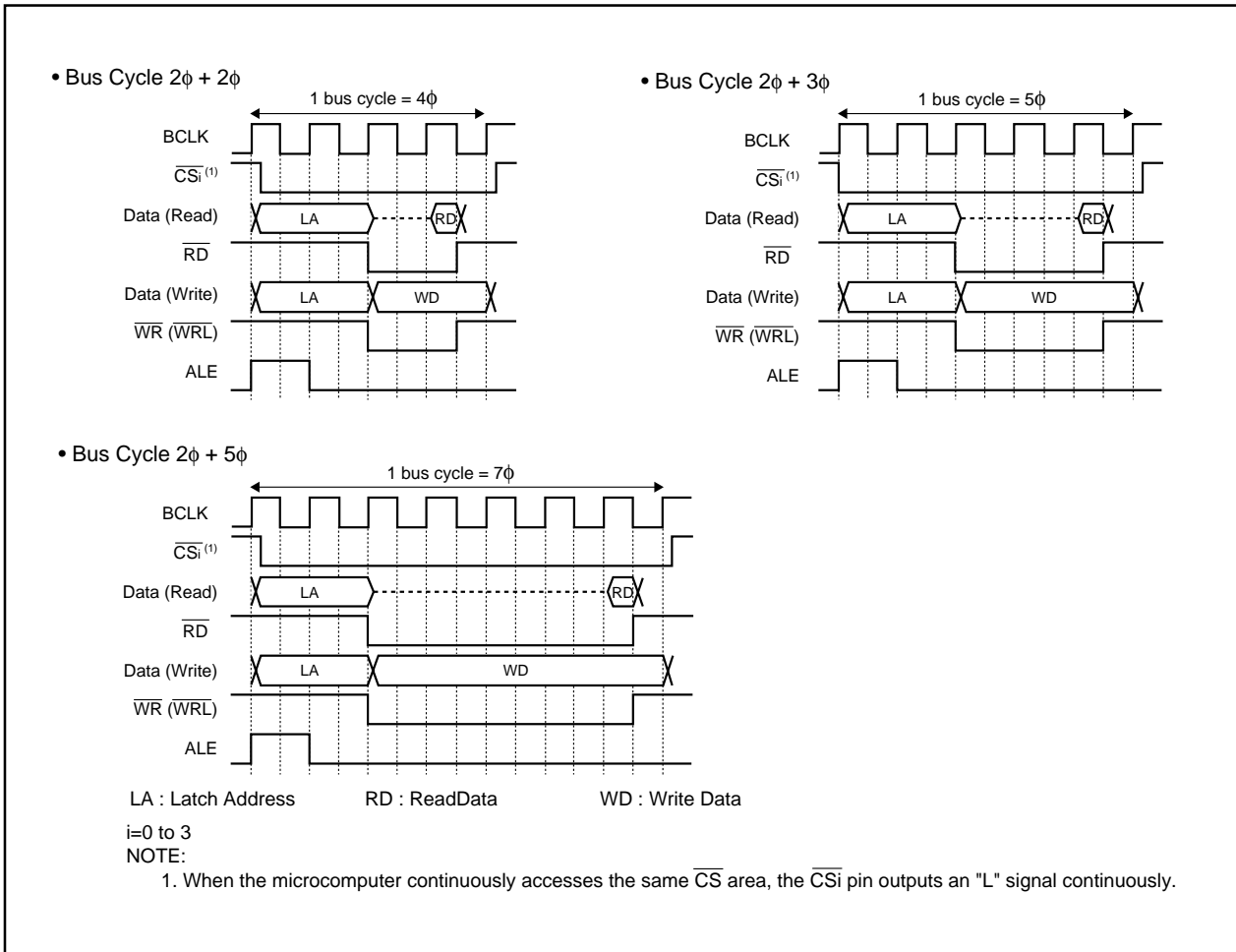


Figure 7.7 Bus Cycle with Multiplexed Bus (1)

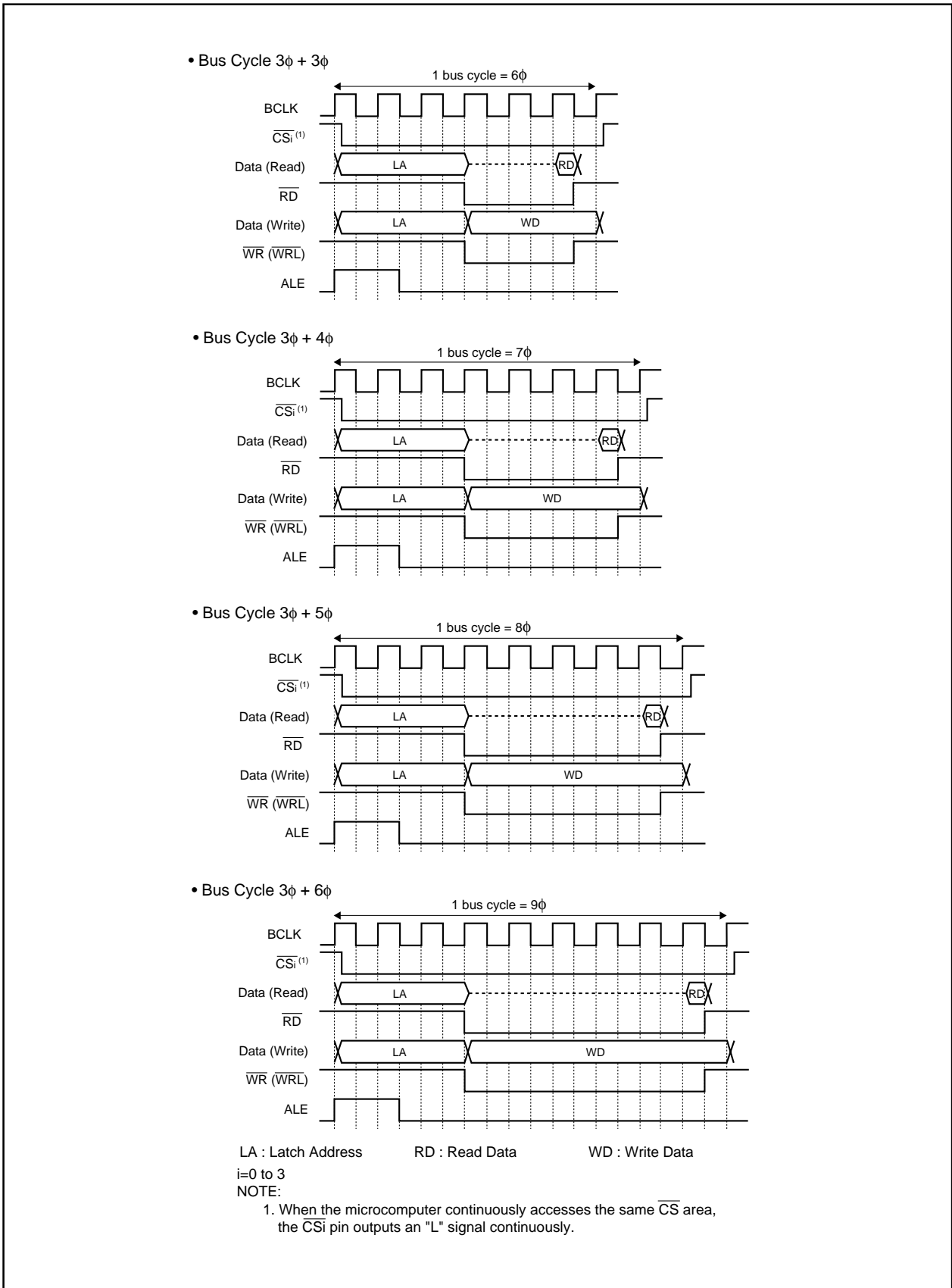
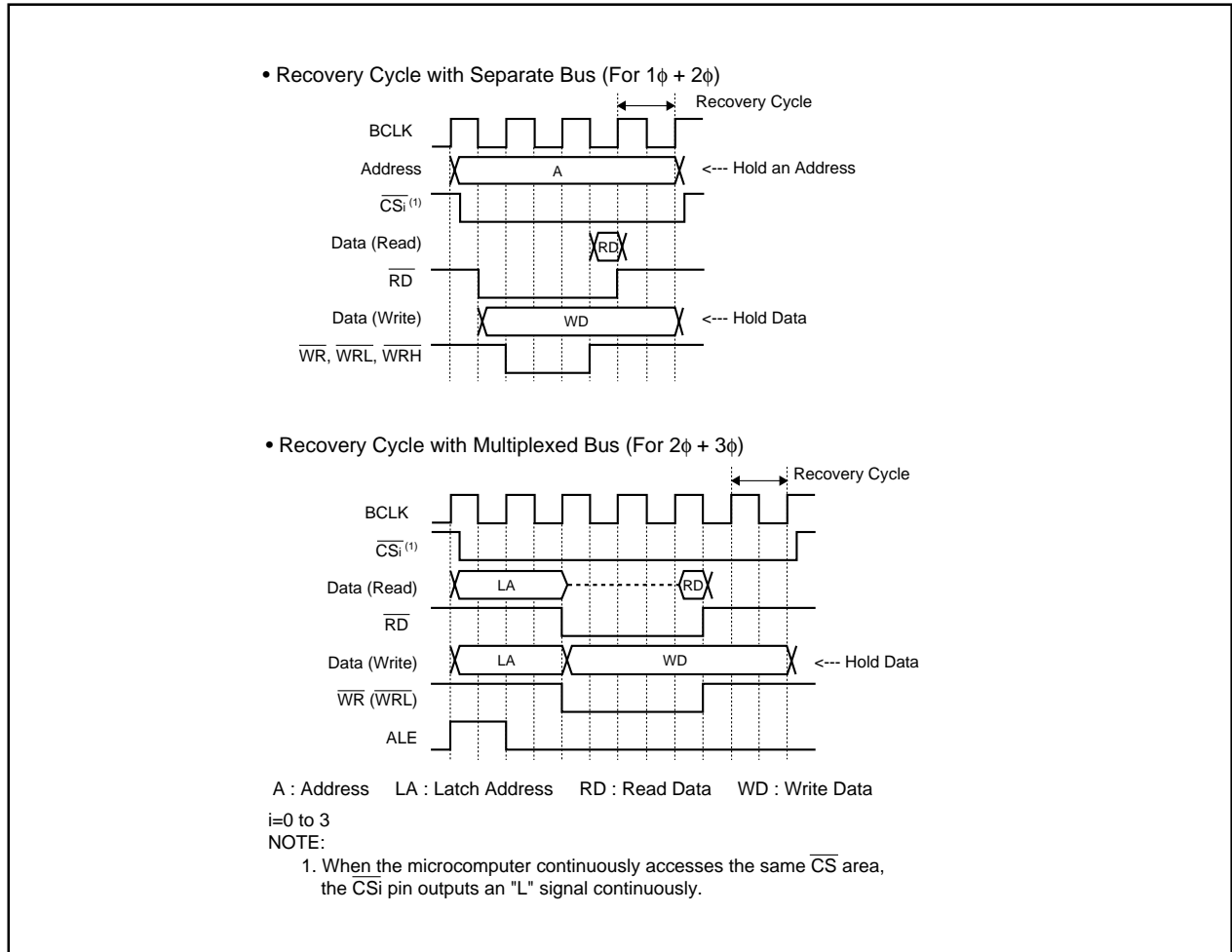


Figure 7.8 Bus Cycle with Multiplexed Bus (2)

**7.2.4.1 Bus Cycle with Recovery Cycle Added**

The EWCRi06 bit in the EWCRi register (i=0 to 3) determines whether the recovery cycle is added or not. In the recovery cycle, addresses and write data outputs are provided continuously (using the separate bus only). Devices, which take longer address hold time and data hold time to write data, are connectable.



**Figure 7.9 Recovery Cycle**

## 7.2.5 ALE Signal

The ALE signal latches an address of the multiplexed bus. Latch an address on the falling edge of the ALE signal. The PM15 and PM14 bits in the PM1 register determine the output pin for the ALE signal. The ALE signal is output to internal space and external space.

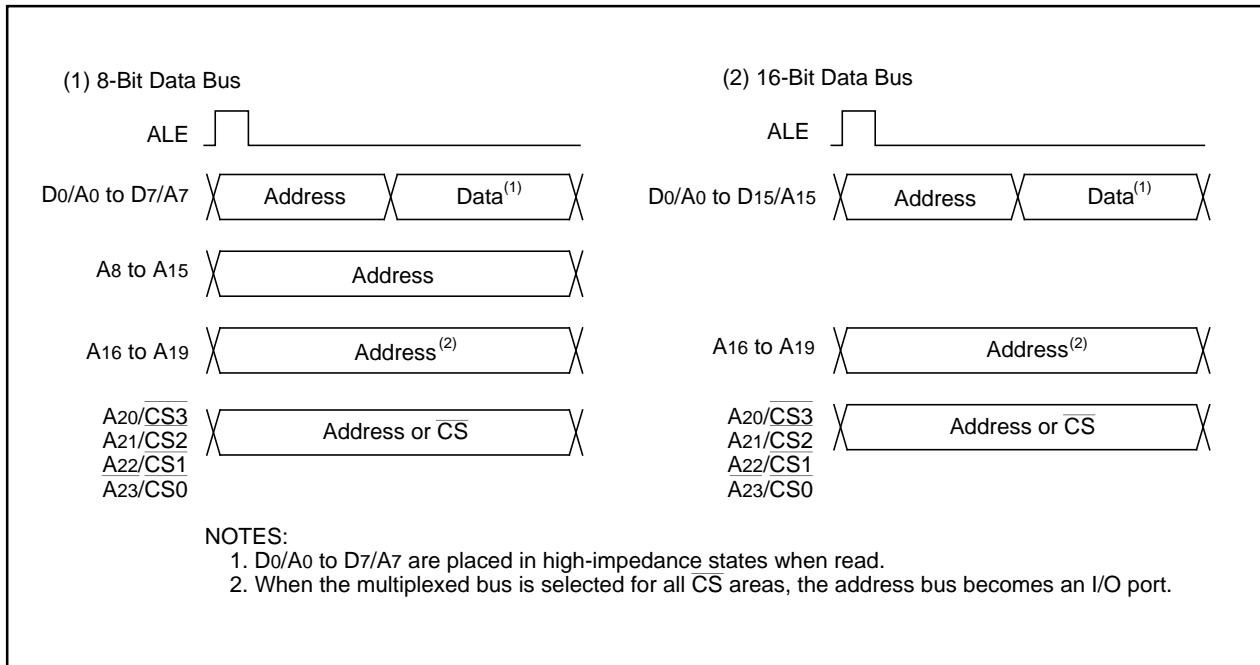


Figure 7.10 ALE Signal and Address/Data Bus

## 7.2.6 RDY Signal

The  $\overline{\text{RDY}}$  signal facilitates access to external devices requiring longer access time. When a low-level ("L") signal is applied to the  $\overline{\text{RDY}}$  pin on the falling edge of the last BCLK of the bus cycle, wait states are inserted into the bus cycle. When a high-level ("H") signal is applied to the  $\overline{\text{RDY}}$  pin on the falling edge of BCLK, the bus cycle starts running again.

Table 7.6 lists microcomputer states when the  $\overline{\text{RDY}}$  signal inserts wait states into the bus cycle. Figure 7.11 shows an example of the  $\overline{\text{RD}}$  signal that is extended by the  $\overline{\text{RDY}}$  signal.

Table 7.6 Microcomputer States in Wait State<sup>(1)</sup>

Item	State
Oscillation	On
$\overline{\text{RD}}$ Signal, $\overline{\text{WR}}$ Signal, Address Bus, Data Bus, $\overline{\text{CS}}$ , ALE Signal, $\overline{\text{HLDA}}$ , Programmable I/O Ports	Maintains the same state as when $\overline{\text{RDY}}$ signal was received
Internal Peripheral Circuits	On

NOTE:

1. The  $\overline{\text{RDY}}$  signal cannot be accepted immediately before software wait states are inserted.



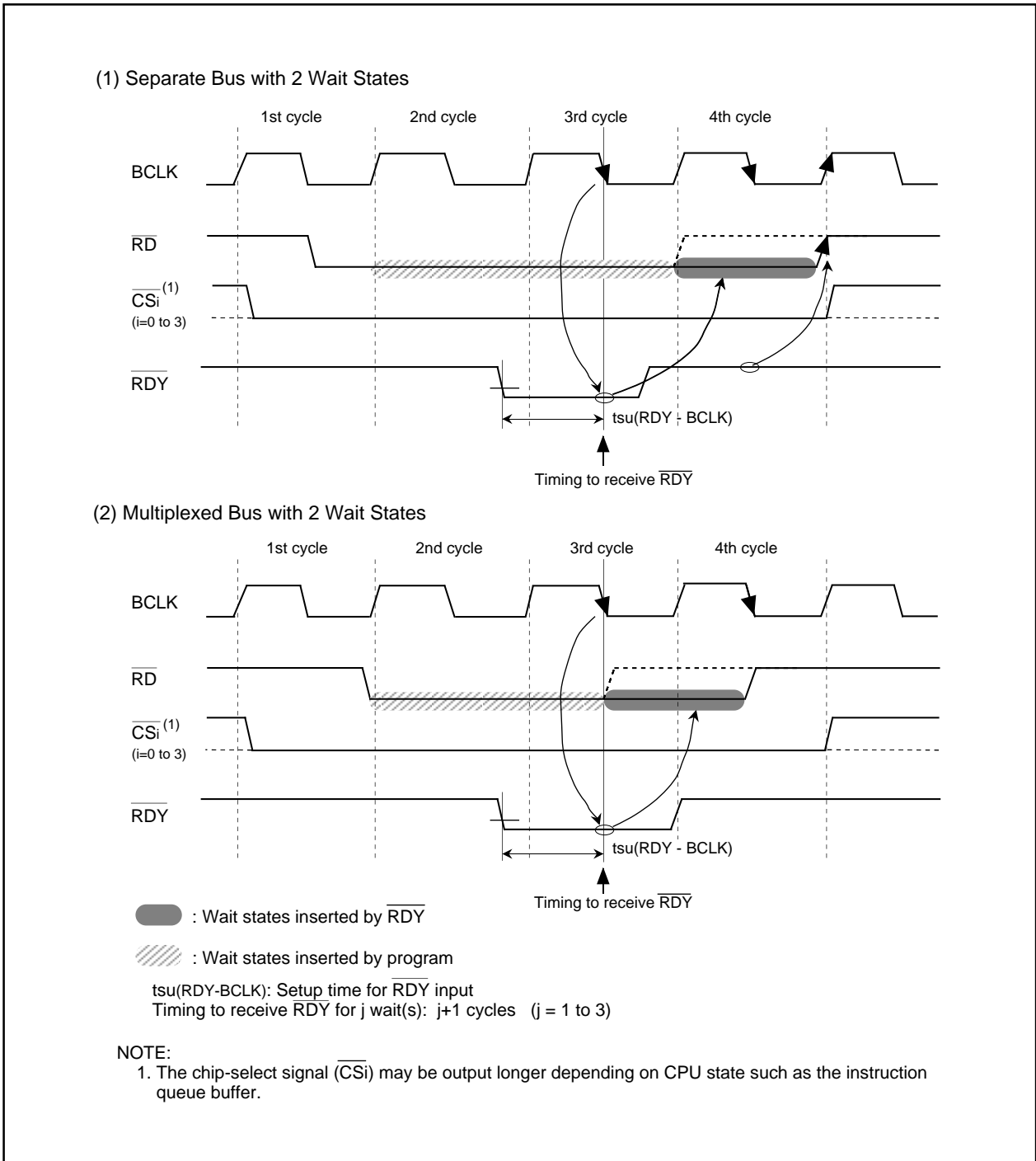


Figure 7.11  $\overline{RD}$  Signal Output Extended by  $\overline{RDY}$  Signal

### 7.2.7 HOLD Signal

The  $\overline{\text{HOLD}}$  signal transfers bus privileges from the CPU to external circuits. When a low-level ("L") signal is applied to the  $\overline{\text{HOLD}}$  pin, the microcomputer enters a hold state after bus access is completed. While the  $\overline{\text{HOLD}}$  pin is held "L", the microcomputer is in a hold state and the  $\overline{\text{HLDA}}$  pin outputs an "L" signal.

Table 7.7 shows the microcomputer status in a hold state.

Bus is used in the following priority order:  $\overline{\text{HOLD}}$ , DMAC, CPU.



**Figure 7.12 Bus Priority Order**

**Table 7.7 Microcomputer Status in Hold State**

Item	Status
Oscillation	On
$\overline{\text{RD}}$ Signal, $\overline{\text{WR}}$ Signal, Address Bus, Data Bus, CS, BHE	High-impedance
Programmable I/O Ports	Maintains the same state as when HOLD signal was received
$\overline{\text{HLDA}}$	Outputs "L"
Internal Peripheral Circuits	On (excluding the watchdog timer)
ALE Signal	Outputs "L"

### 7.2.8 External Bus Status when Accessing Internal Space

Table 7.8 shows external bus states when an internal space is accessed.

**Table 7.8 External Bus States when Accessing Internal Space**

Item	State when Accessing SFRs, Internal ROM, and Internal RAM	
Address Bus	Holds address of external space last accessed	
Data Bus	When Reading	High-impedance
	When Writing	High-impedance
$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WRL}}$ , $\overline{\text{WRH}}$	Outputs "H"	
$\overline{\text{BHE}}$	Holds state of external space last accessed	
$\overline{\text{CS}}$	Outputs "H"	
ALE	Outputs ALE	

### 7.2.9 BCLK Output

The CPU clock operates the CPU. P53 outputs the CPU clock signal as BCLK when the PM07 bit in the PM0 register is set to "0" (BCLK) and the CM01 and CM00 bits in the CM0 register are set to "002" (I/O port P53).

No BCLK is output in single-chip mode. Refer to **8. Clock Generation Circuit** for details.

## 8. Clock Generation Circuit

### 8.1 Types of the Clock Generation Circuit

Four circuits are included to generate the system clock signal:

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

Table 8.1 lists specifications of the clock generation circuit. Figure 8.1 shows a block diagram of the clock generation circuit. Figures 8.2 to 8.8 show registers controlling the clock.

**Table 8.1 Clock Generation Circuit Specifications**

Item	Main Clock Oscillation Circuit	Sub Clock Oscillation Circuit	On-chip Oscillator	PLL Frequency Synthesizer
Use	CPU clock source, Peripheral function clock source	CPU clock source, Timer A and B clock source	CPU clock source, Peripheral function clock source	CPU clock source, Peripheral function clock source
Clock Frequency	Up to 32 MHz	32.768 kHz	Approx. 1 MHz	Up to 32 MHz (See Table 8.3)
Connectable Oscillator or Additional Circuit	Ceramic resonator Crystal oscillator	Crystal oscillator	---	---
Pins for Oscillator or for Additional Circuit	XIN, XOUT	XCIN, XCOU	---	---
Oscillation Stop / Restart Function	Available	Available	Available	Available
Oscillator State after Reset	Oscillating	Stopped	Stopped	Stopped
Other	Externally generated clock can be applied.	Externally generated clock can be applied.	When the main clock stops oscillating, the on-chip oscillator starts oscillating automatically and becomes clock source for the CPU and peripheral function.	---

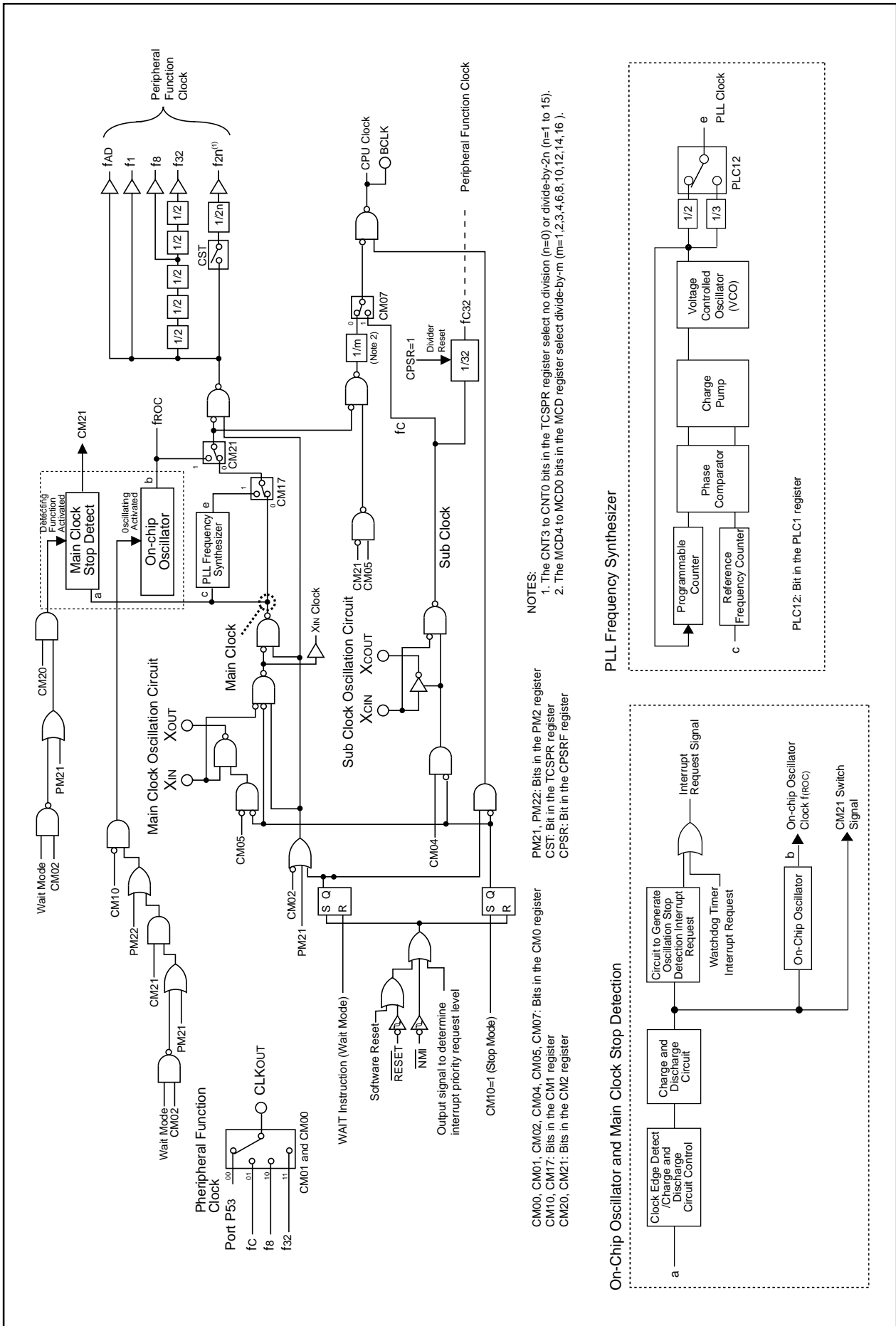


Figure 8.1 Clock Generation Circuit

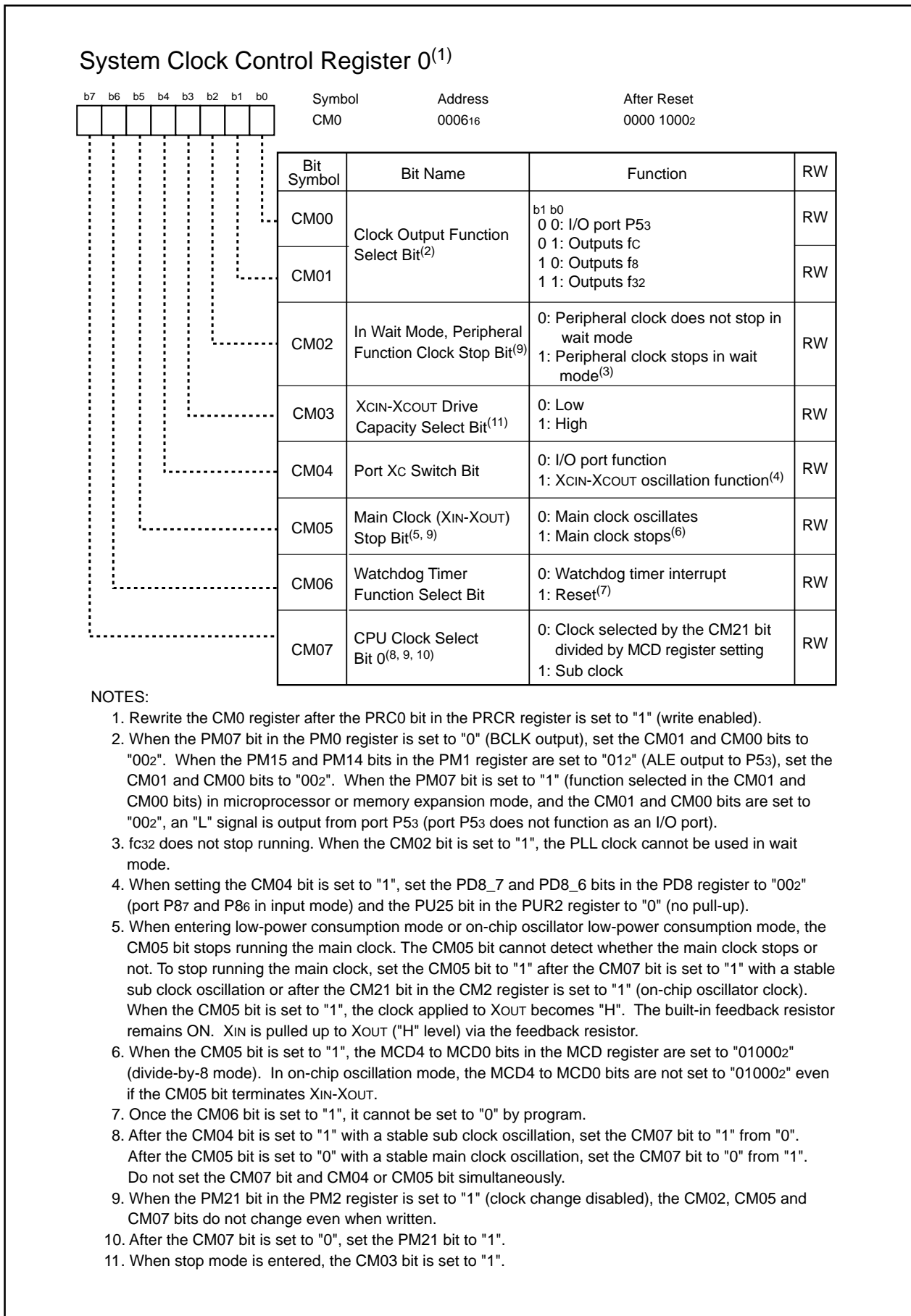


Figure 8.2 CM0 Register

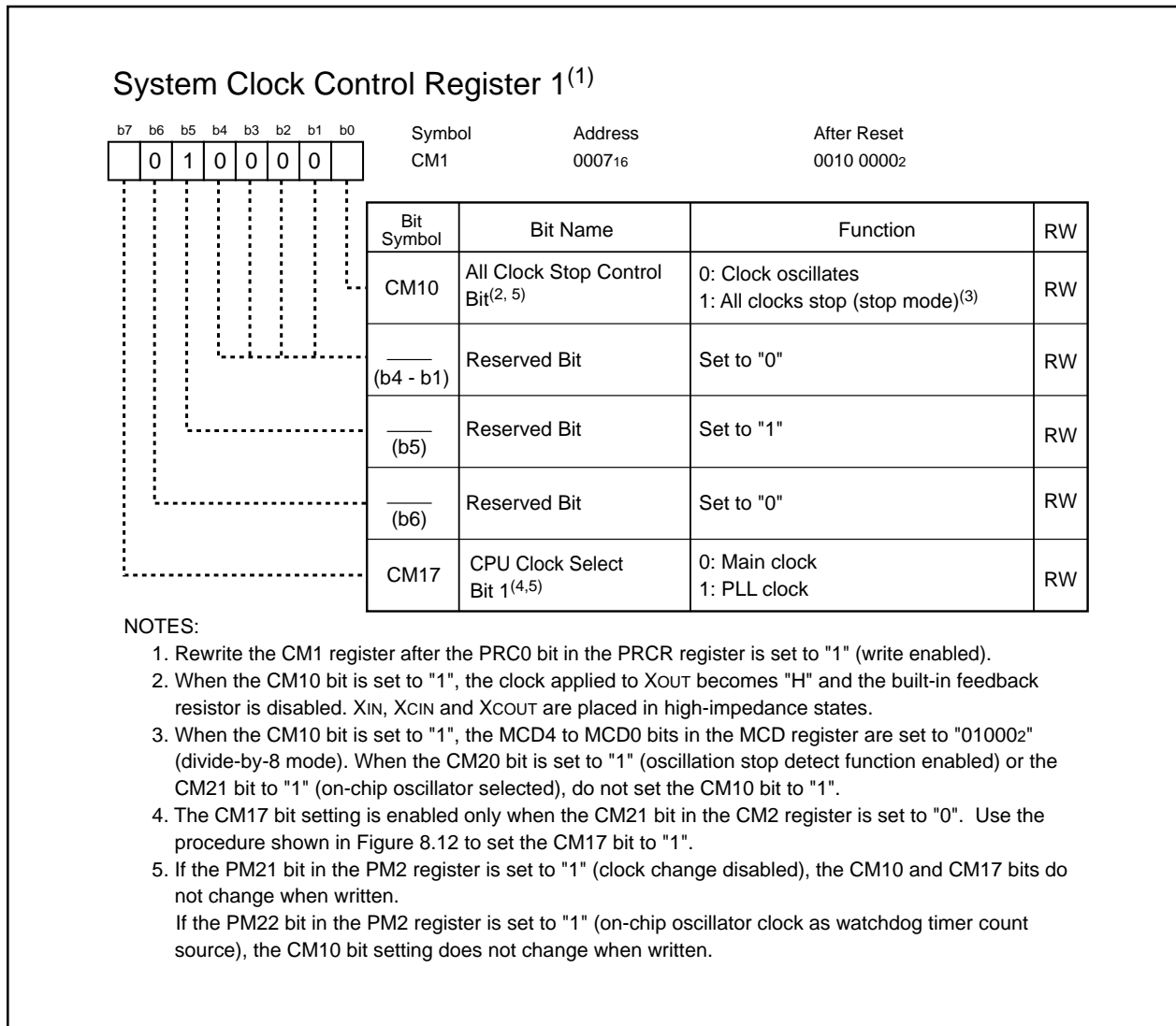
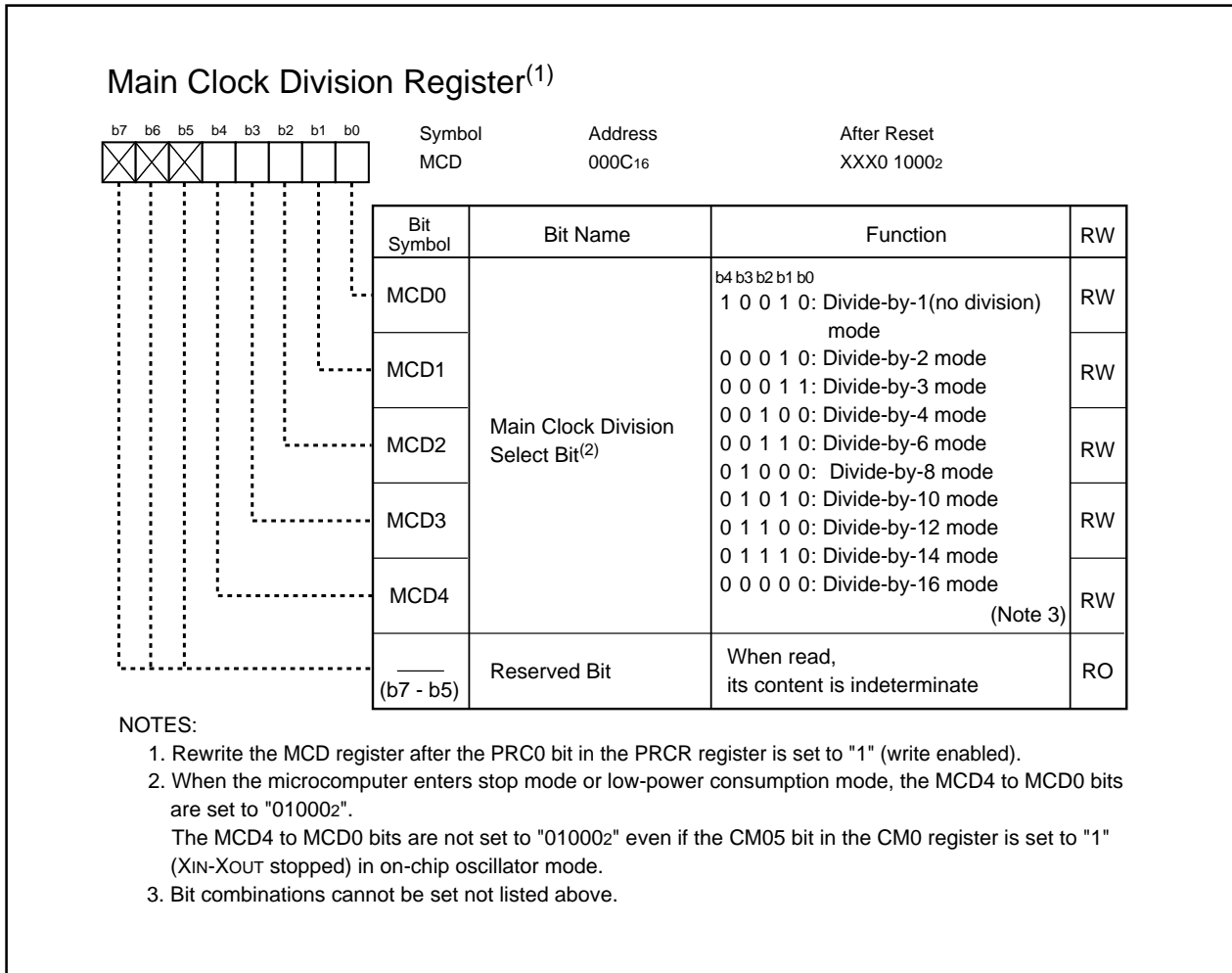


Figure 8.3 CM1 Register



**Figure 8.4 MCD Register**

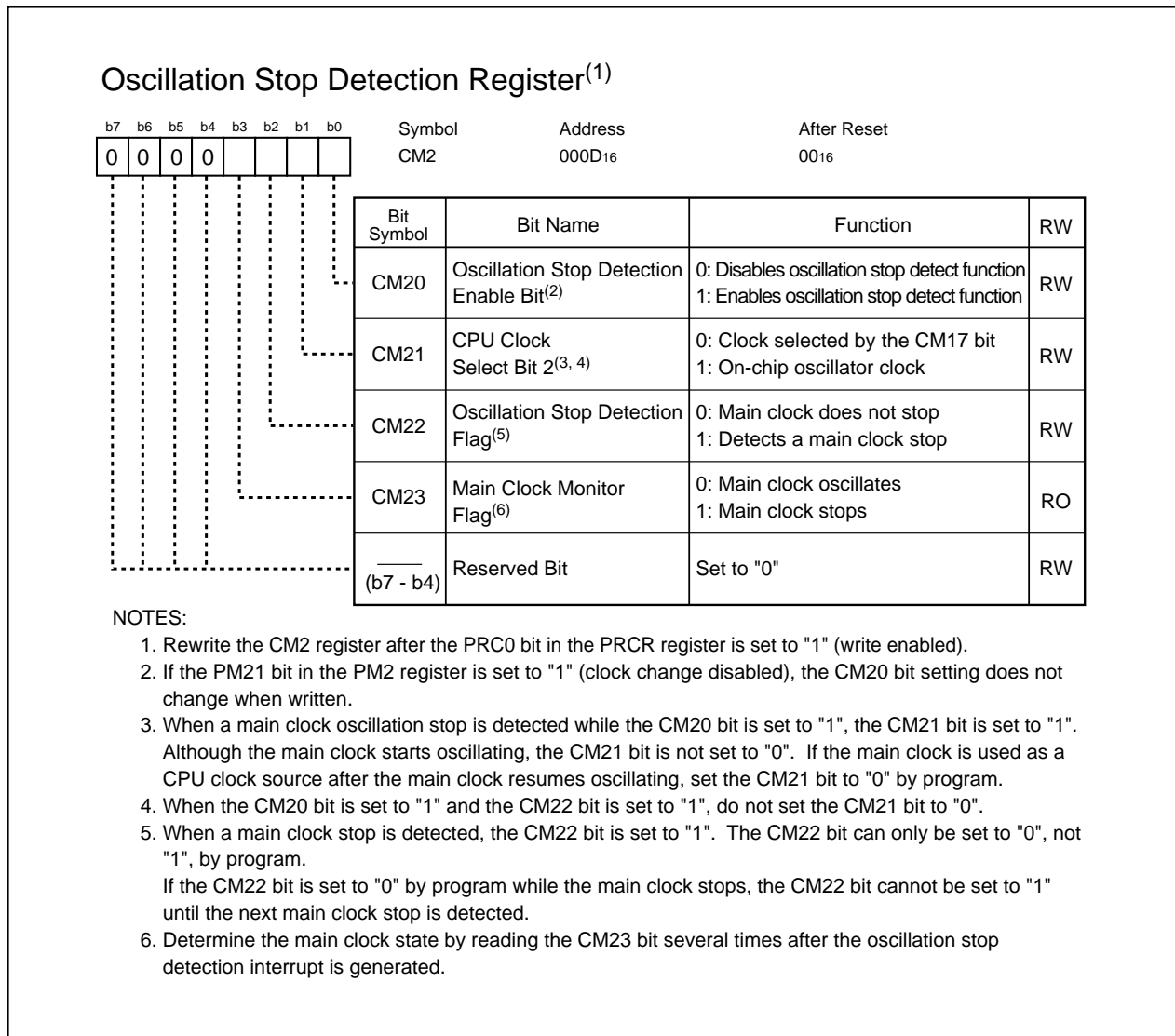


Figure 8.5 CM2 Register



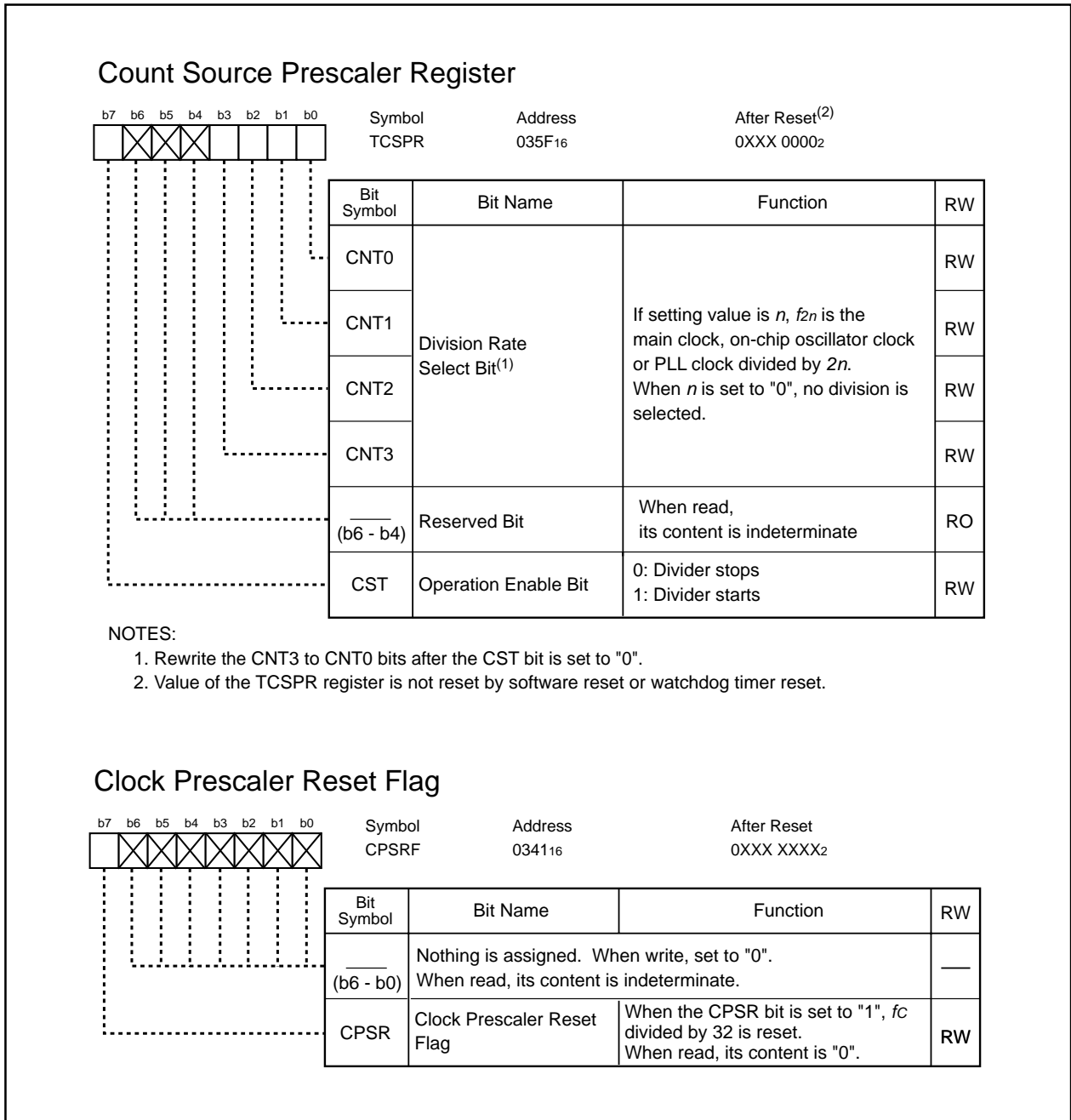


Figure 8.6 TCSR and CPSRF Registers

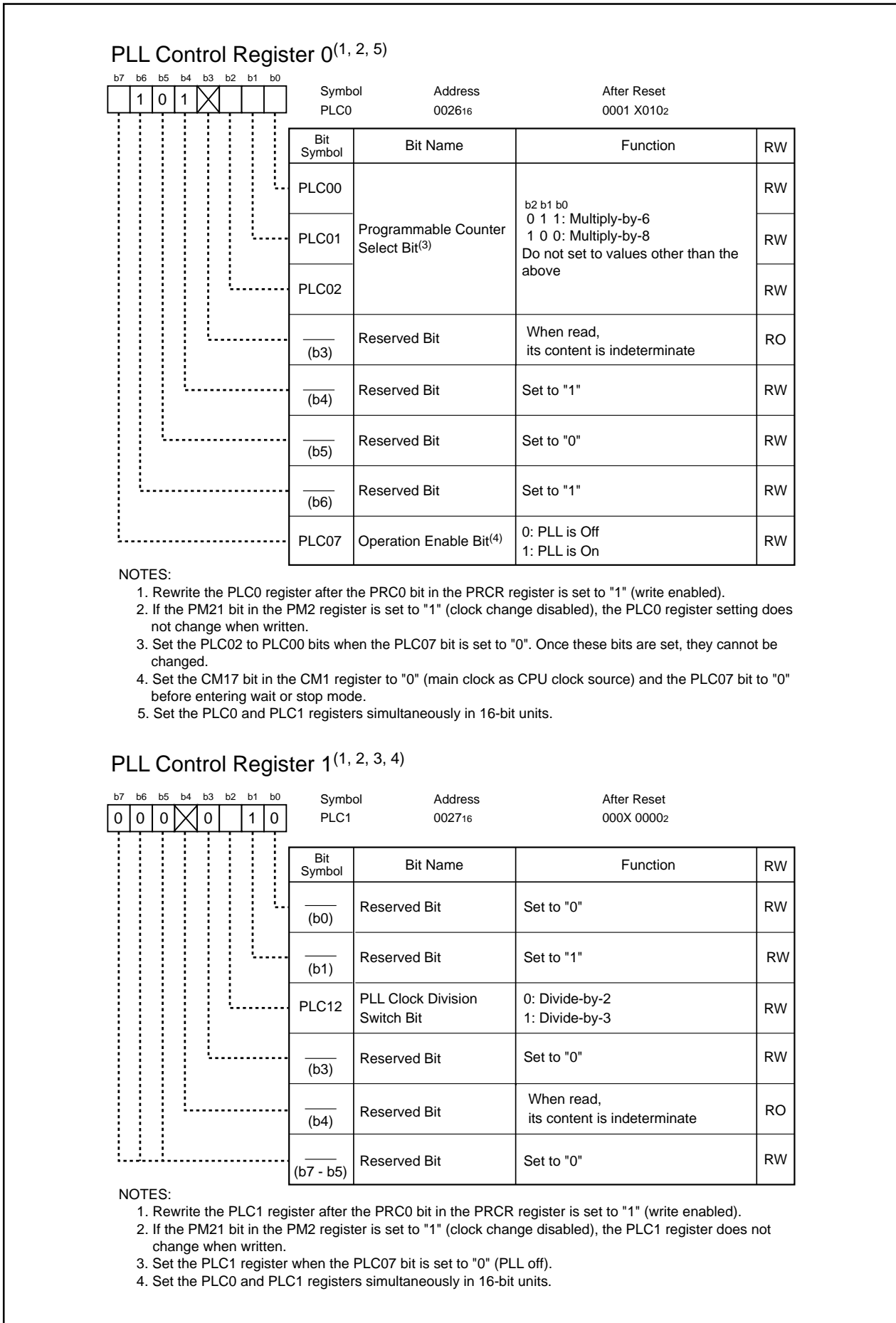
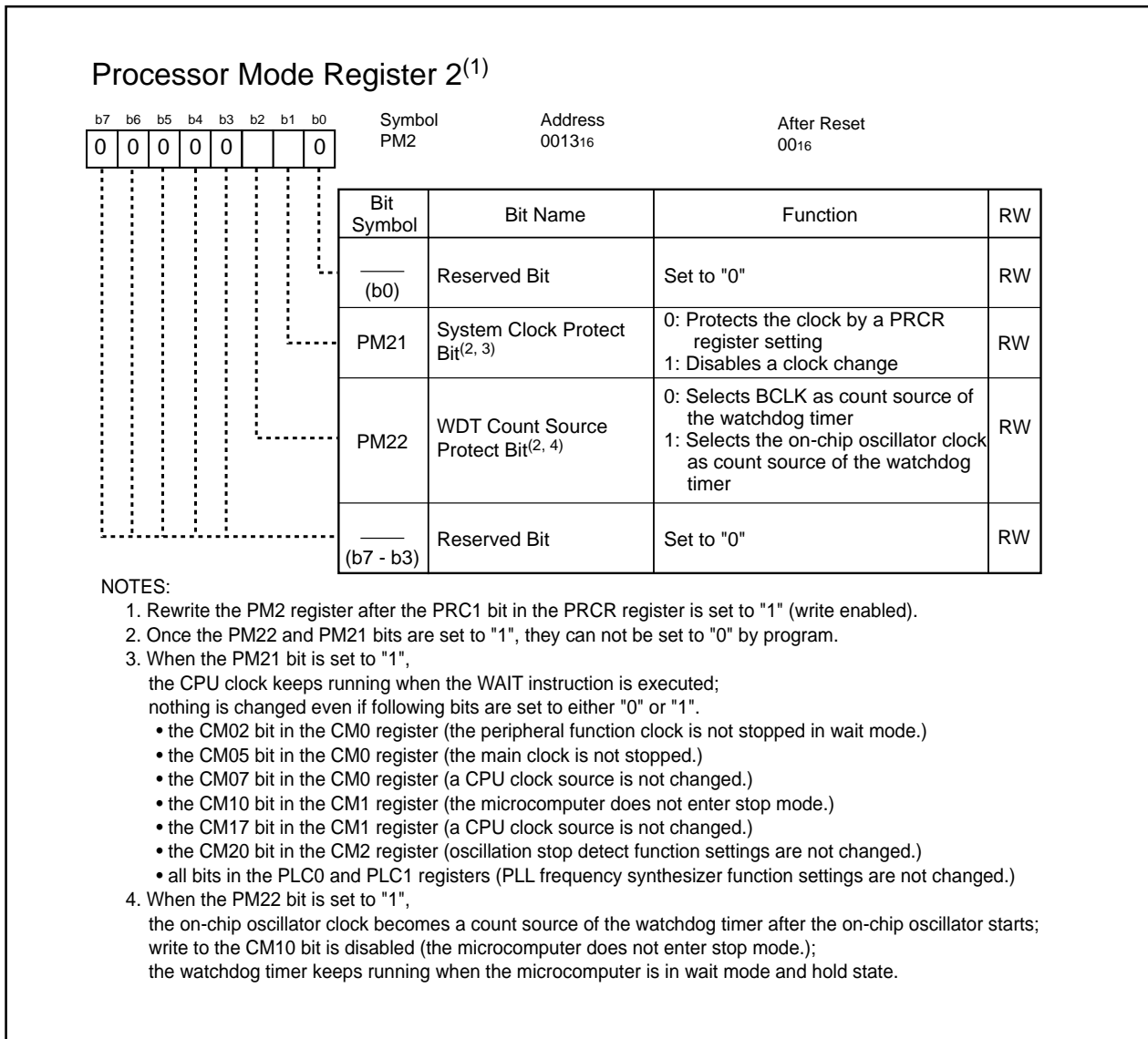


Figure 8.7 PLC0 and PLC1 Registers



**Figure 8.8 PM2 Register**

### 8.1.1 Main Clock

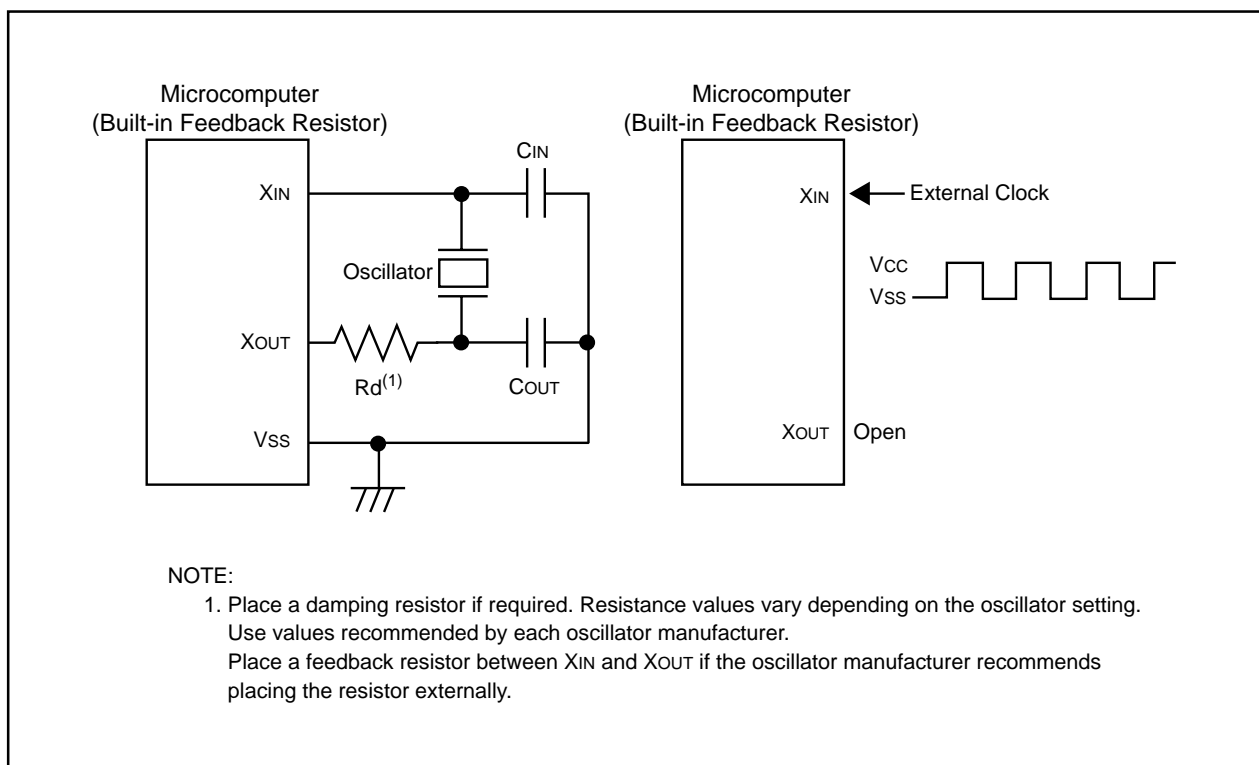
Main clock oscillation circuit generates the main clock. The main clock becomes clock source of the CPU clock and peripheral function clock.

The main clock oscillation circuit is configured by connecting an oscillator or resonator between the XIN and XOUT pins. The circuit has a built-in feedback resistor. The feedback resistor is separated from the oscillation circuit in stop mode to reduce power consumption. An external clock can be applied to the XIN pin in the main clock oscillation circuit. Figure 8.9 shows an example of a main clock circuit connection. Circuit constants vary depending on each oscillator. Use the circuit constant recommended by each oscillator manufacturer.

The main clock divided-by-eight becomes a CPU clock source after reset.

To reduce power consumption, set the CM05 bit in the CM0 register to "1" (main clock stopped) after switching the CPU clock source to the sub clock or on-chip oscillator clock. In this case, the clock applied to XOUT becomes high ("H"). XIN is pulled up by XOUT via the feedback resistor which remains on. When an external clock is applied to the XIN pin, do not set the CM05 bit to "1".

All clocks, including the main clock, stop in stop mode. Refer to **8.5 Power Consumption Control** for details.



**Figure 8.9 Main Clock Circuit Connection**

### 8.1.2 Sub Clock

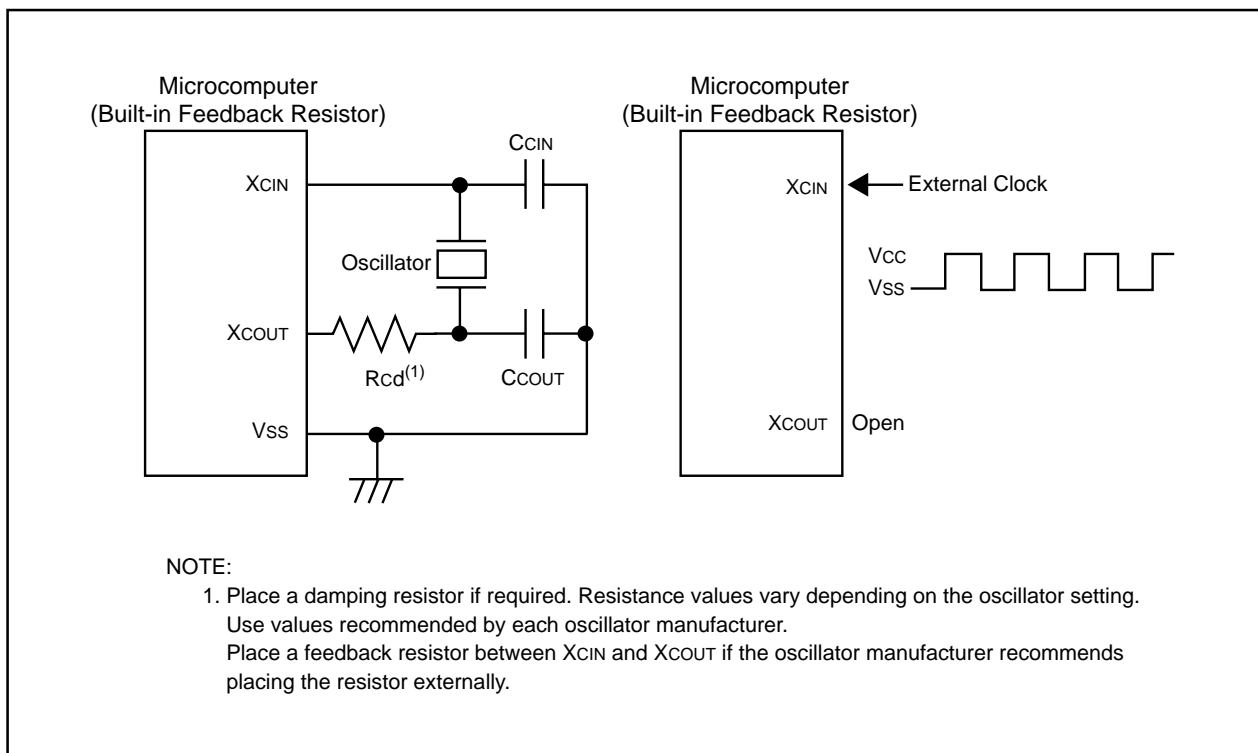
Sub clock oscillation circuit generates the sub clock. The sub clock becomes clock source of the CPU clock and for the timers A and B. The same frequency,  $f_c$ , as the sub clock can be output from the CLKOUT pin.

The sub clock oscillation circuit is configured by connecting a crystal oscillator between the XCIN and XCOUT pins. The circuit has a built-in feedback resistor. The feedback resistor is separated from the oscillation circuit in stop mode to reduce power consumption. An external clock can be applied to the XCIN pin. Figure 8.10 shows an example of a sub clock circuit connection. Circuit constants vary depending on each oscillator. Use the circuit constant recommended by each oscillator manufacturer.

The sub clock stops after reset. The feedback resistor is separated from the oscillation circuit. When the PD8\_6 and PD8\_7 bits in the PD8 register are set to "0" (input mode) and the PU25 bit in the PUR2 register is set to "0" (no pull-up), set the CM04 bit in the CM0 register to "1" (XCIN-XCOUT oscillation function). The sub clock oscillation circuit starts oscillating. To apply an external clock to the XCIN pin, set the CM04 bit to "1" when the PD8\_7 bit is set to "0" and the PU25 bit to "0". The clock applied to the XCIN pin becomes a clock source of the sub clock.

When the CM07 bit in the CM0 register is set to "1" (sub clock) after the sub clock oscillation has stabilized, the sub clock becomes a CPU clock source.

All clocks, including the sub clock, stop in stop mode. Refer to **8.5 Power Consumption Control** for details.



**Figure 8.10 Sub Clock Circuit Connection**

### 8.1.3 On-Chip Oscillator Clock

On-chip oscillator generates the on-chip oscillator clock. The 1-MHz on-chip oscillator clock becomes a clock source of the CPU clock and peripheral function clock.

The on-chip oscillator clock stops after reset. When the CM21 bit in the CM2 register is set to "1" (on-chip oscillator clock), the on-chip oscillator starts oscillating. Instead of the main clock, the on-chip oscillator clock becomes clock source of the CPU clock and peripheral function clock.

Table 8.2 shows bit settings for on-chip oscillator start condition.

**Table 8.2 Bit Settings for On-Chip Oscillator Start Condition**

CM2 Register	PM2 Register	Used as
CM21 Bit	PM22 Bit	
1	0	CPU clock source or peripheral function clock source
0	1	Watchdog timer operating clock source (The clock keeps running when entering stop mode.)

#### 8.1.3.1 Oscillation Stop Detect Function

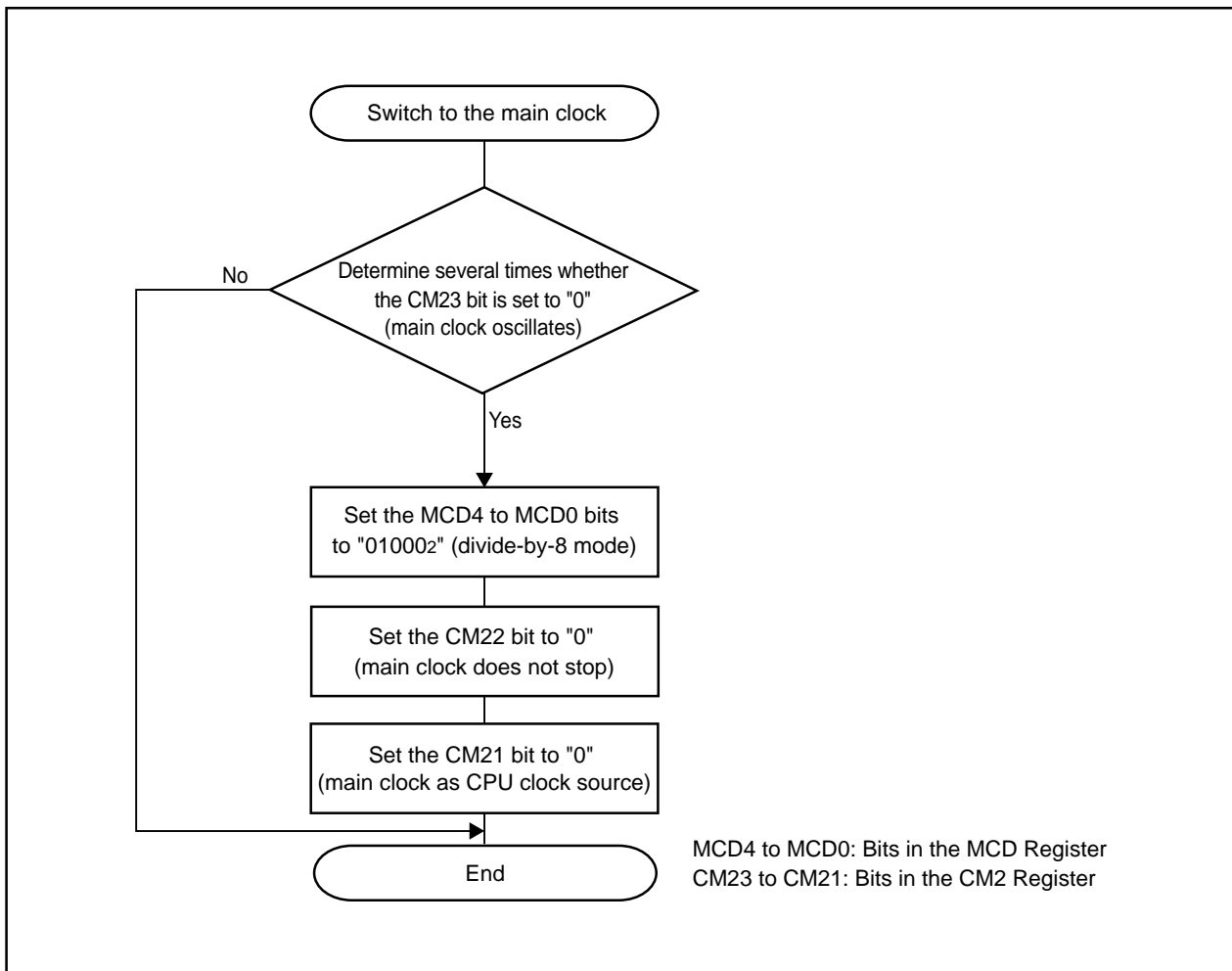
When the main clock is terminated by external source, the on-chip oscillator automatically starts oscillating to generate another clock.

When the CM 20 bit in the CM2 register is set to "1" (oscillation stop detect function enabled), an oscillation stop detection interrupt request is generated as soon as the main clock stops. Simultaneously, the on-chip oscillator starts oscillating. Instead of the main clock, the on-chip oscillator clock becomes clock source for the CPU clock and peripheral function clock. Associated bits are set as follows:

- The CM21 bit is set to "1" (on-chip oscillator clock becomes a clock source of the CPU clock.)
- The CM22 bit is set to "1" (main clock stop is detected.)
- The CM23 bit is set to "1" (main clock stops.) (See **Figure 8.14**)

#### 8.1.3.2 How to Use Oscillation Stop Detect Function

- The oscillation stop detection interrupt shares vectors with the watchdog timer interrupt and the low voltage detection interrupt. When these interrupts are used simultaneously, read the CM22 bit with an interrupt routine to determine if an oscillation stop detection interrupt request has been generated.
- When the main clock resumes running after an oscillation stop is detected, set the main clock as clock source of the CPU clock and peripheral function clock. Figure 8.11 shows the procedure to switch the on-chip oscillator clock to the main clock.
- In low-speed mode, when the main clock is stopped by setting the CM20 bit to "1", the oscillation stop detection interrupt request is generated. Simultaneously, the on-chip oscillator starts oscillating. The sub clock remains the CPU clock source. The on-chip oscillator clock becomes a clock source for the peripheral function clock.
- When the peripheral function clock stops running, the oscillation stop detect function is also disabled. To enter wait mode while the oscillation stop detect function is in use, set the CM02 bit in the CM0 register to "0" (peripheral clock does not stop in wait mode).
- The oscillation stop detect function is provided to handle main clock stop caused by external source. Set the CM20 bit to "0" (oscillation stop detect function disabled) when the main clock is terminated by program, i.e., entering stop mode or setting the CM05 bit to "1" (main clock oscillation stop).
- When the main clock frequency is 2 MHz or less, the oscillation stop detect function is not available. Set the CM20 bit to "0".



**Figure 8.11 Switching Procedure from On-chip Oscillator Clock to Main Clock**

### 8.1.4 PLL Clock

The PLL frequency synthesizer generates the PLL clock based on the main clock. The PLL clock can be used as clock source for the CPU clock and peripheral function clock.

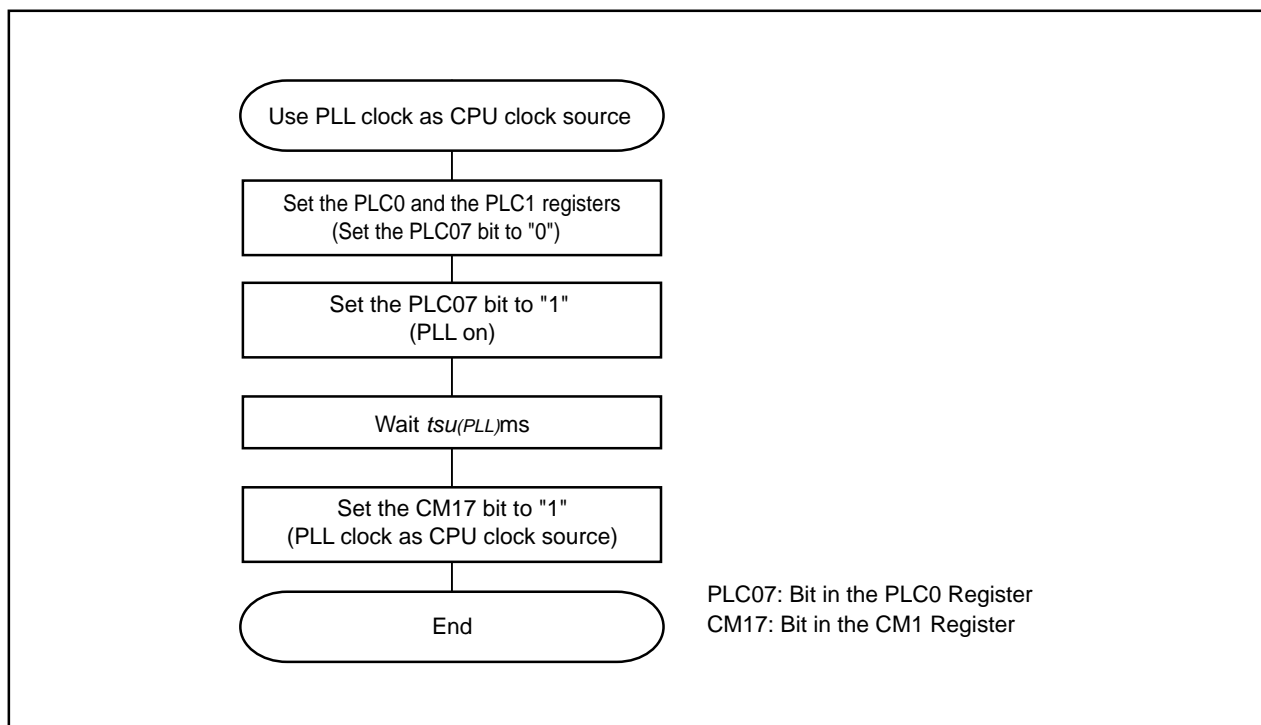
The PLL frequency synthesizer stops after reset. When the PLC07 bit is set to "1" (PLL on), the PLL frequency synthesizer starts operating. Wait  $t_{su(PLL)}$  ms for the PLL clock to stabilize.

The PLL clock can either be the clock output from the voltage controlled oscillator (VCO) divided-by-2 or divided-by-3. When the PLL clock is used as a clock source for the CPU clock or peripheral function clock, set each bit as is shown in Table 8.3. Figure 8.12 shows the procedure to use the PLL clock as the CPU clock source.

To enter wait or stop mode, set the CM17 bit to "0" (main clock as CPU clock source), set the PLC07 bit in the PLC0 register to "0" (PLL off) and then enter wait or stop mode.

**Table 8.3 Bit Settings to Use PLL Clock as CPU Clock Source**

f(X <sub>IN</sub> )	PLC0 Register			PLC1 Register	PLL Clock
	PLC02 Bit	PLC01 Bit	PLC00 Bit	PLC12 Bit	
10 MHz	0	1	1	0	30 MHz
				1	20 MHz
8 MHz	1	0	0	0	32 MHz
				1	21.3 MHz



**Figure 8.12 Procedure to Use PLL Clock as CPU Clock Source**



## 8.2 CPU Clock and BCLK

The CPU operating clock is referred to as the CPU clock. The CPU clock is also a count source for the watchdog timer. After reset, the CPU clock is the main clock divided-by-8. In memory expansion or micro-processor mode, the clock having the same frequency as the CPU clock can be output from the BCLK pin as BCLK. Refer to **8.4 Clock Output Function** for details.

The main clock, sub clock, on-chip oscillator clock or PLL clock can be selected as a clock source for the CPU clock. Table 8.4 shows CPU clock source and bit settings.

When the main clock, on-chip oscillator clock or PLL clock is selected as a clock source of the CPU clock, the selected clock divided-by-1 (no division), -2, -3, -4, -6, -8, -10, -12, -14 or -16 becomes the CPU clock. The MCD4 to MCD0 bits in the MCD register select the clock division.

When the microcomputer enters stop mode or low-power consumption mode (except when the on-chip oscillator clock is the CPU clock), the MCD4 to MCD0 bits are set to "010002" (divide-by-8 mode). Therefore, when the main clock starts running, the CPU clock enters medium-speed mode (divide-by-8).

**Table 8.4 CPU Clock Source and Bit Settings**

CPU Clock Source	CM0 Register	CM1 Register	CM2 Register
	CM07 Bit	CM17 Bit	CM21 Bit
Main Clock	0	0	0
Sub Clock	1	0	0
On-Chip Oscillator Clock	0	0	1
PLL Clock	0	1	0

## 8.3 Peripheral Function Clock

The peripheral function clock becomes an operating clock or count source for peripheral functions excluding the watchdog timer.

### 8.3.1 f<sub>1</sub>, f<sub>8</sub>, f<sub>32</sub> and f<sub>2n</sub>

f<sub>1</sub>, f<sub>8</sub> and f<sub>32</sub> are the peripheral function clock, selected by the CM21 bit, divided-by-1, -8, or -32. The PM27 and PM26 bits in the PM2 register selects a f<sub>2n</sub> count source from the peripheral clock, X<sub>IN</sub> clock, and the on-chip oscillator clock. The CNT3 to CNT0 bits in the TCSPR register selects a f<sub>2n</sub> division. (n=0 to 15. No division when n=0.)

f<sub>1</sub>, f<sub>8</sub>, f<sub>32</sub> and f<sub>2n</sub> stop when the CM02 bit in the CM0 register to "1" (peripheral function stops in wait mode) to enter wait mode or when in low-power consumption mode.

f<sub>1</sub>, f<sub>8</sub> and f<sub>2n</sub> are used as an operating clock of the serial I/O and count source of the timers A and B. f<sub>1</sub> is also used as an operating clock for the intelligent I/O.

The CLKOUT pin outputs f<sub>8</sub> and f<sub>32</sub>. Refer to **8.4 Clock Output Function** for details.

### 8.3.2 f<sub>AD</sub>

f<sub>AD</sub> is an operating clock for the A/D converter and has the same frequency as either the main clock<sup>(1)</sup> or the on-chip oscillator clock. The CM21 bit determines which clock is selected.

If the CM02 bit is set to "1" (peripheral function stop in wait mode) to enter wait mode, f<sub>AD</sub> stops. f<sub>AD</sub> also stops in low-power consumption mode.

NOTE:

1. The PLL clock, instead of the main clock, when the CM17 bit is set to "1" (PLL clock).

### 8.3.3 fc32

fc32 is the sub clock divided by 32. fc32 is used as a count source for the timers A and B. fc32 is available when the sub clock is running.

## 8.4 Clock Output Function

The CLKOUT pin outputs fc, f8 or f32.

In memory expansion mode or microprocessor mode, a clock having the same frequency as the CPU clock can be output from the BCLK pin as BCLK.

Table 8.5 lists CLKOUT pin function in single-chip mode. Table 8.6 lists CLKOUT pin function in memory expansion mode and microprocessor mode.

**Table 8.5 CLKOUT Pin in Single-Chip Mode**

PM0 Register <sup>(1)</sup>		CM0 Register <sup>(2)</sup>		CLKOUT Pin Function
PM07 Bit		CM01 Bit	CM00 Bit	
—		0	0	P53 I/O port
1		0	1	Outputs fc
1		1	0	Outputs f8
1		1	1	Outputs f32

- : Can be set to either "0" or "1"

NOTES:

1. Rewrite the PM0 register after the PRC1 bit in the PRCR register is set to "1" (write enabled).
2. Rewrite the CM0 register after the PRC0 bit in the PRCR register is set to "1" (write enabled).

**Table 8.6 CLKOUT Pin in Memory Expansion Mode and Microprocessor Mode**

PM1 Register <sup>(1)</sup>		PM0 Register <sup>(1)</sup>		CM0 Register <sup>(2)</sup>		CLKOUT Pin Function
PM15 Bit	PM14 Bit	PM07 Bit		CM01 Bit	CM00 Bit	
002, 102, 112,		0		0 <sup>(3)</sup>	0 <sup>(3)</sup>	Outputs BCLK
		1		0	0	Outputs "L" (not P53)
		1		0	1	Outputs fc
		1		1	0	Outputs f8
		1		1	1	Outputs f32
0	1	—		0 <sup>(3)</sup>	0 <sup>(3)</sup>	Outputs ALE

- : Can be set to either "0" or "1"

NOTES:

1. Rewrite the PM1 and PM0 registers after the PRC1 bit in the PRCR register is set to "1" (write enabled).
2. Rewrite the CM0 register after the PRC0 bit in the PRCR register is set to "1" (write enabled).
3. When the PM07 bit is set to "0" (selected in the CM01 and CM00 bits) or the PM15 and PM14 bits are set to "012" (P53/BCLK), set the CM01 and CM00 bits to "002" (I/O port P53).

## 8.5 Power Consumption 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 section. Figure 8.13 shows a block diagram of status transition in wait mode and stop mode. Figure 8.14 shows a block diagram of status transition in all modes.

### 8.5.1 Normal Operating Mode

The normal operating mode is further separated into six modes.

In normal operating mode, the CPU clock and peripheral function clock are supplied to operate the CPU and peripheral function. The power consumption control is enabled by controlling a CPU clock frequency. The higher the CPU clock frequency is, the more processing power increases. The lower the CPU clock frequency is, the more power consumption decreases. When unnecessary oscillation circuit stops, power consumption is further reduced.

#### 8.5.1.1 High-Speed Mode

The main clock<sup>(1)</sup> becomes the CPU clock and a clock source of the peripheral function clock. When the sub clock runs, fc32 can be used as a count source for the timers A and B.

#### 8.5.1.2 Medium-Speed Mode

The main clock<sup>(1)</sup> divided-by-2, -3, -4, -6, -8, -10, -12, -14, or -16 becomes the CPU clock. The main clock<sup>(1)</sup> is a clock source for the peripheral function clock. When the sub clock runs, fc32 can be used as a count source for the timers A and B.

#### 8.5.1.3 Low-Speed Mode

The sub clock becomes the CPU clock. The main clock<sup>(1)</sup> is a clock source for the peripheral function clock. fc32 can be used as a count source for the timers A and B.

#### 8.5.1.4 Low-Power Consumption Mode

The microcomputer enters low-power consumption mode when the main clock stops in low-speed mode. The sub clock becomes the CPU clock. Only fc32 can be used as a count source for the timers A and B and the peripheral function clock. In low-power consumption mode, the MCD4 to MCD0 bits in the MCD register are set to "010002" (divide-by-8 mode). Therefore, when the main clock resumes running, the microcomputer is in medium-speed mode (divide-by-8 mode).

#### 8.5.1.5 On-Chip Oscillator Mode

The on-chip oscillator clock divided-by-1 (no division), -2, -3, -4, -6, -8, -10, -12, -14, or -16 becomes the CPU clock. The on-chip oscillator clock is a clock source for the peripheral function clock. When the sub clock runs, fc32 can be used as a count source for the timers A and B.

#### 8.5.1.6 On-Chip Oscillator Low-Power Consumption Mode

The microcomputer enters on-chip oscillator low-power consumption mode when the main clock stops in on-chip oscillator mode. The on-chip oscillator clock divided-by-1 (no division), -2, -3, -4, -6, -8, -10, -12, -14, or -16 becomes the CPU clock. The on-chip oscillator clock is a clock source for the peripheral function clock. When the sub clock runs, fc32 can be used as a count source for the timers A and B.

NOTE:

1. The PLL clock, instead of the main clock, when the CM17 bit is set to "1" (PLL clock).

Switch the CPU clock after the clock to be switched to stabilize. Sub clock oscillation will take longer<sup>(2)</sup> to stabilize. Wait, by program, until the clock stabilizes directly after turning the microcomputer on or exiting stop mode.

To switch the on-chip oscillator clock to the main clock, enter medium-speed mode (divide-by-8) after the main clock is divided by eight in on-chip oscillator mode (the MCD4 to MCD0 bits in the MCD register are set to "010002").

Do not enter on-chip oscillator mode or on-chip oscillator low-power consumption mode from low-speed mode or low-power consumption mode and vice versa.

**NOTE:**

2. Contact your oscillator manufacturer for oscillation stabilization time.

## 8.5.2 Wait Mode

In wait mode, the CPU clock stops running. The CPU and watchdog timer, operated by the CPU clock, also stop. When the PM22 bit in the PM2 register is set to "1" (on-chip oscillator clock as watchdog timer count source), the watchdog timer continues operating. Because the main clock, sub clock and on-chip oscillator clock continue running, peripheral functions using these clocks also continue operating.

### 8.5.2.1 Peripheral Function Clock Stop Function

If the CM02 bit in the CM0 register is set to "1" (peripheral function clock stops in wait mode), f<sub>1</sub>, f<sub>8</sub>, f<sub>32</sub>, f<sub>2n</sub> (when peripheral clock is selected as a count source), and f<sub>AD</sub> stop in wait mode. Power consumption can be reduced. f<sub>2n</sub>, when X<sub>IN</sub> clock or on-chip oscillator clock is selected as a count source, and f<sub>c32</sub> do not stop running.

### 8.5.2.2 Entering Wait Mode

If wait mode is entered after setting the CM02 bit to "1", set the MCD4 to MCD0 bits in the MCD register to be the 10-MHz or less CPU clock frequency after dividing the main clock.

Enter wait mode after setting the followings.

- Initial Setting

Set each interrupt priority level after setting the exit priority level required to exit wait mode, controlled by the RLVL2 to RLVL0 bits in the RLVL register, to "7".

- Before Entering Wait Mode

(1) Set the I flag to "0"

(2) Set the interrupt priority level of the interrupt being used to exit wait mode

(3) Set the interrupt priority levels of the interrupts, not being used to exit wait mode, to "0"

(4) Set IPL in the FLG register. Then set the exit priority level to the same level as IPL

Interrupt priority level of the interrupt used to exit wait mode > IPL = the exit priority level

(5) Set the PRC0 bit in the PRCR register to "1"

(6) If the CPU clock source is the PLL clock, set the CM17 bit in the CM1 register to "0" (main clock) and PLC07 bit in the PLC0 register to "0" (PLL off)

(7) Set the I flag to "1"

(8) Execute the WAIT instruction

- After Exiting Wait Mode

Set the exit priority level to "7" as soon as exiting wait mode.

### 8.5.2.3 Pin Status in Wait Mode

Table 8.7 lists pin states in wait mode.

**Table 8.7 Pin States in Wait Mode**

Pin		Memory Expansion Mode Microprocessor Mode	Single-Chip Mode
Address Bus, Data Bus, $\overline{CS0}$ to $\overline{CS3}$ , BHE		Maintains state immediately before entering wait mode	/
RD, WR, $\overline{WRL}$ , $\overline{WRH}$		"H"	
HLDA, BCLK		"H"	
ALE		"L"	
Ports		Maintains state immediately before entering wait mode	
CLKOUT	When fc is selected	Outputs clock	
	When f8, f32 are selected	Outputs the clock when the CM02 bit in the CM0 register is set to "0" (peripheral function clock does not stop in wait mode). Maintains state immediately before entering wait mode when the CM02 bit is set to "1" (peripheral function clock stops in wait mode).	

### 8.5.2.4 Exiting Wait Mode

Wait mode is exited by the hardware reset,  $\overline{NMI}$  interrupt or peripheral function interrupts.

When the hardware reset or  $\overline{NMI}$  interrupt, but not the peripheral function interrupts, is used to exit wait mode, set the ILVL2 to ILVL0 bits for the peripheral function interrupts to "0002" (interrupt disabled) before executing the WAIT instruction.

CM02 bit setting affects the peripheral function interrupts. When the CM02 bit in the CM0 register is set to "0" (peripheral function clock does not stop in wait mode), all peripheral function interrupts can be used to exit wait mode. When the CM02 bit is set to "1" (peripheral function clock stops in wait mode), peripheral functions using the peripheral function clock stop. Therefore, the peripheral function interrupts cannot be used to exit wait mode. However, the peripheral function interrupts caused by an external clock, fc32, or f2n whose count source is the XIN clock or on-chip oscillator clock, can be used to exit wait mode.

The CPU clock used when exiting wait mode by the peripheral function interrupts or  $\overline{NMI}$  interrupt is the same CPU clock used when the WAIT instruction is executed.

Table 8.8 shows interrupts to be used to exit wait mode and usage conditions.

**Table 8.8 Interrupts to Exit Wait Mode**

Interrupt	When CM02=0	When CM02=1
NMI Interrupt	Available	Available
Serial I/O Interrupt	Available when the internal and external clocks are used	Available when the external clock or $f_{2n}$ (when XIN clock or on-chip oscillator is selected) is used
Key Input Interrupt	Available	Available
A/D Conversion Interrupt	Available in single or single-sweep mode	Do not use
Timer A Interrupt Timer B Interrupt	Available in all modes	Available in event counter mode or when count source is $fc_{32}$ or $f_{2n}$ (when XIN clock or on-chip oscillator is selected)
$\overline{INT}$ Interrupt	Available	Available
Intelligent I/O Interrupt	Available	Do not use

### 8.5.3 Stop Mode

In stop mode, all oscillators and resonators stop. The CPU clock and peripheral function clock, as well as the CPU and peripheral functions operated by these clocks, also stop. The least power required to operate the microcomputer is in stop mode. The internal RAM holds its data when the voltage applied to the VCC1 and VCC2 pins is  $V_{RAM}$  or more. If the voltage applied to the VCC1 and VCC2 pins is 2.7 V or less, the voltage must be  $V_{cc1} \geq V_{cc2} \geq V_{RAM}$ .

The following interrupts can be used to exit stop mode:

- $\overline{NMI}$  interrupt
- Key Input Interrupt
- $\overline{INT}$  interrupt
- Timer A and B interrupt (Available when the timer counts external pulse, having its 100 Hz or less frequency, in event counter mode)

### 8.5.3.1 Entering Stop Mode

Stop mode is entered when setting the CM10 bit in the CM1 register to "1" (all clocks stops). The MCD4 to MCD0 bits in the MCD register become set to "010002" (divide-by-8 mode).

Enter stop mode after setting the followings.

- Initial Setting

Set each interrupt priority level after setting the exit priority level required to exit stop mode, controlled by the RLVL2 to RLVL0 bits in the RLVL register, to "7".

- Before Entering stop mode

- (1) Set the I flag to "0"

- (2) Set the interrupt priority level of the interrupt being used to exit stop mode

- (3) Set the interrupt priority levels of the interrupts, not being used to exit stop mode, to "0"

- (4) Set IPL in the FLG register. Then set the exit priority level to the same level as IPL

Interrupt priority level of the interrupt used to exit stop mode > IPL = the exit priority level

- (5) Set the PRC0 bit in the PRCR register to "1" (write enabled)

- (6) Select the main clock as the CPU clock

- When the CPU clock source is the sub clock,

- (a) set the CM05 bit in the CM0 register to "0" (main clock oscillates)

- (b) set the CM07 bit in the CM0 register to "0" (clock selected by the CM21 bit divided by MCD register setting)

- When the CPU clock source is the PLL clock,

- (a) set the CM17 bit in the CM1 register to "0" (main clock)

- (b) set the PLC07 bit in the PLC0 register to "0" (PLL off)

- When the CPU clock source is the on-chip oscillator clock,

- (a) set MCD4 to MCD0 bits to "010002" (divide-by-8 mode)

- (b) set the CM05 bit to "0" (main clock oscillates)

- (c) set the CM21 bit in the CM2 register to "0" (clock selected by the CM17 bit)

- (7) The oscillation stop detect function is used, set the CM20 bit in the CM2 register to "0" (oscillation stop detect function disabled)

- (8) Set the I flag to "1"

- (9) Set the CM10 bit to "1" (all clocks stops)

- After Exiting Stop Mode

Set the exit priority level to "7" as soon as exiting stop mode.

### 8.5.3.2 Exiting Stop Mode

Stop mode is exited by the hardware reset,  $\overline{\text{NMI}}$  interrupt or peripheral function interrupts (key input interrupt and  $\overline{\text{INT}}$  interrupt).

When the hardware reset or  $\overline{\text{NMI}}$  interrupt, but not the peripheral function interrupts, is used to exit wait mode, set all ILVL2 to ILVL0 bits in the interrupt control registers for the peripheral function interrupt to "0002" (interrupt disabled) before setting the CM10 bit to "1" (all clocks stops).

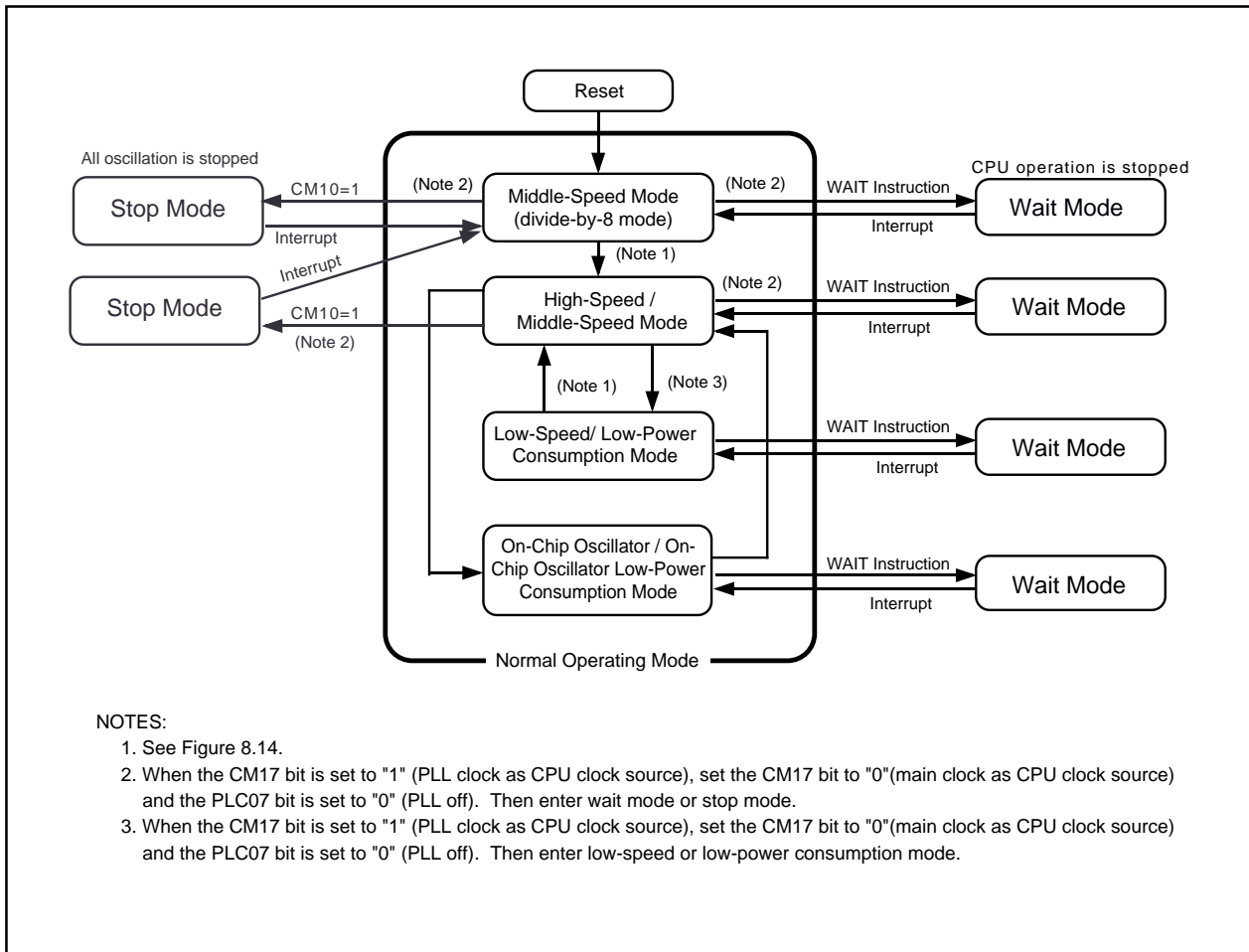
### 8.5.3.3 Pin Status in Stop Mode

Table 8.9 lists pin status in stop mode.

**Table 8.9 Pin Status in Stop Mode**

Pin		Memory Expansion Mode Microprocessor Mode	Single-Chip Mode
Address Bus, Data Bus, $\overline{CS0}$ to $\overline{CS3}$ , $\overline{BHE}$		Maintains state immediately before entering stop mode	/
$\overline{RD}$ , $\overline{WR}$ , $\overline{WRL}$ , $\overline{WRH}$		"H"	
$\overline{HLDA}$ , $\overline{BCLK}$		"H"	
ALE		"H"	
Ports		Maintains state immediately before entering stop mode	
CLKOUT	When f <sub>c</sub> selected	"H"	
	When f <sub>8</sub> , f <sub>32</sub> selected	Maintains state immediately before entering stop mode	
XIN		Placed in a high-impedance state	
XOUT		"H"	
XCIN, XCOUT		Placed in a high-impedance state	





**Figure 8.13 Status Transition in Wait Mode and Stop Mode**

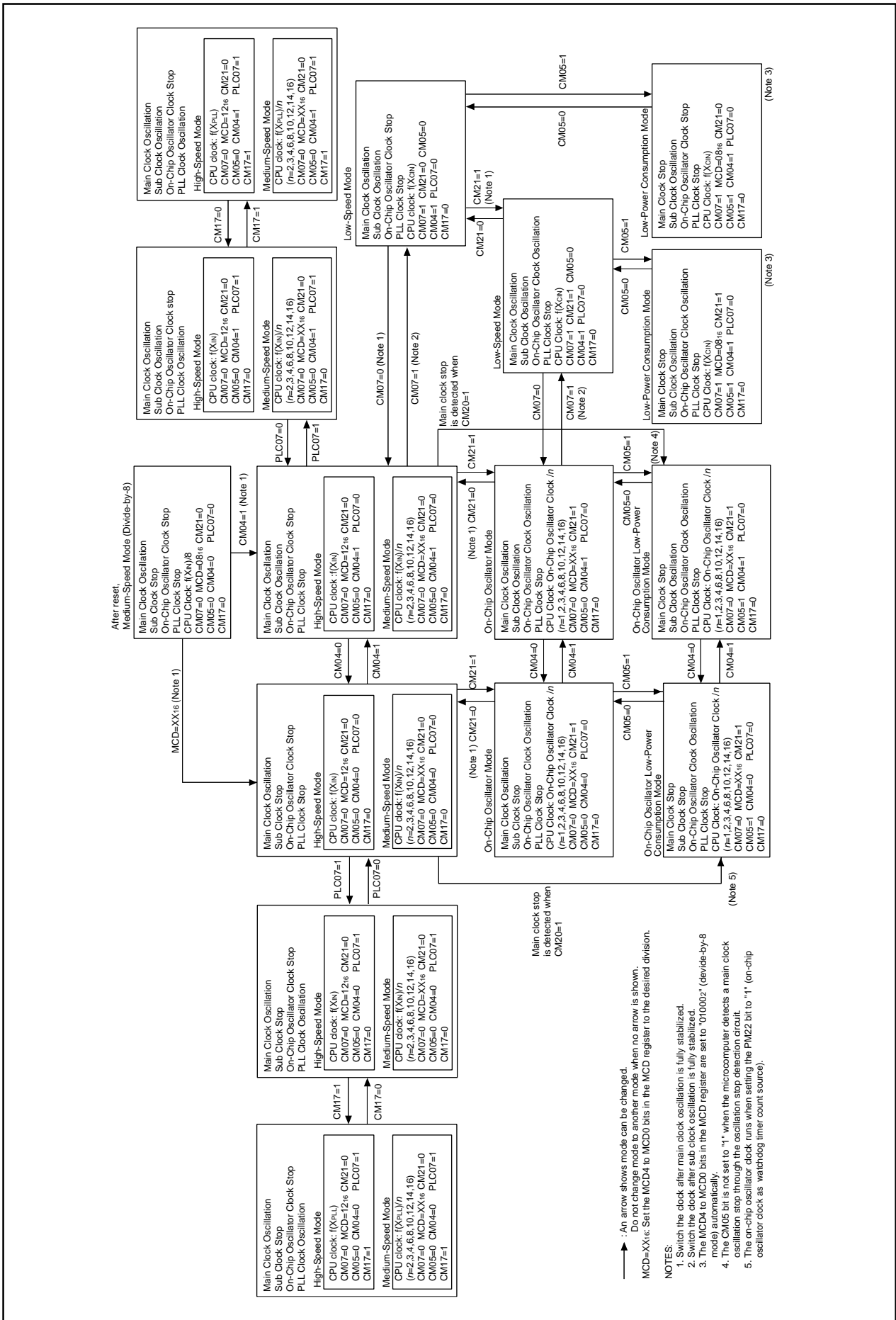


Figure 8.14 Status Transition

## 8.6 System Clock Protect Function

The system clock protect function prohibits the CPU clock from changing clock sources when the main clock is selected as the CPU clock source. This prevents the CPU clock from stopping the program crash. 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 CM17 bit in the CM1 register
- The CM20 bit in the CM2 register
- All bits in the PLC0 and PLC1 registers

The CPU clock continues running when the WAIT instruction is executed.

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

- (1) Set the PRC1 bit in the PRCR register to "1" (write enabled).
- (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 disabled).

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

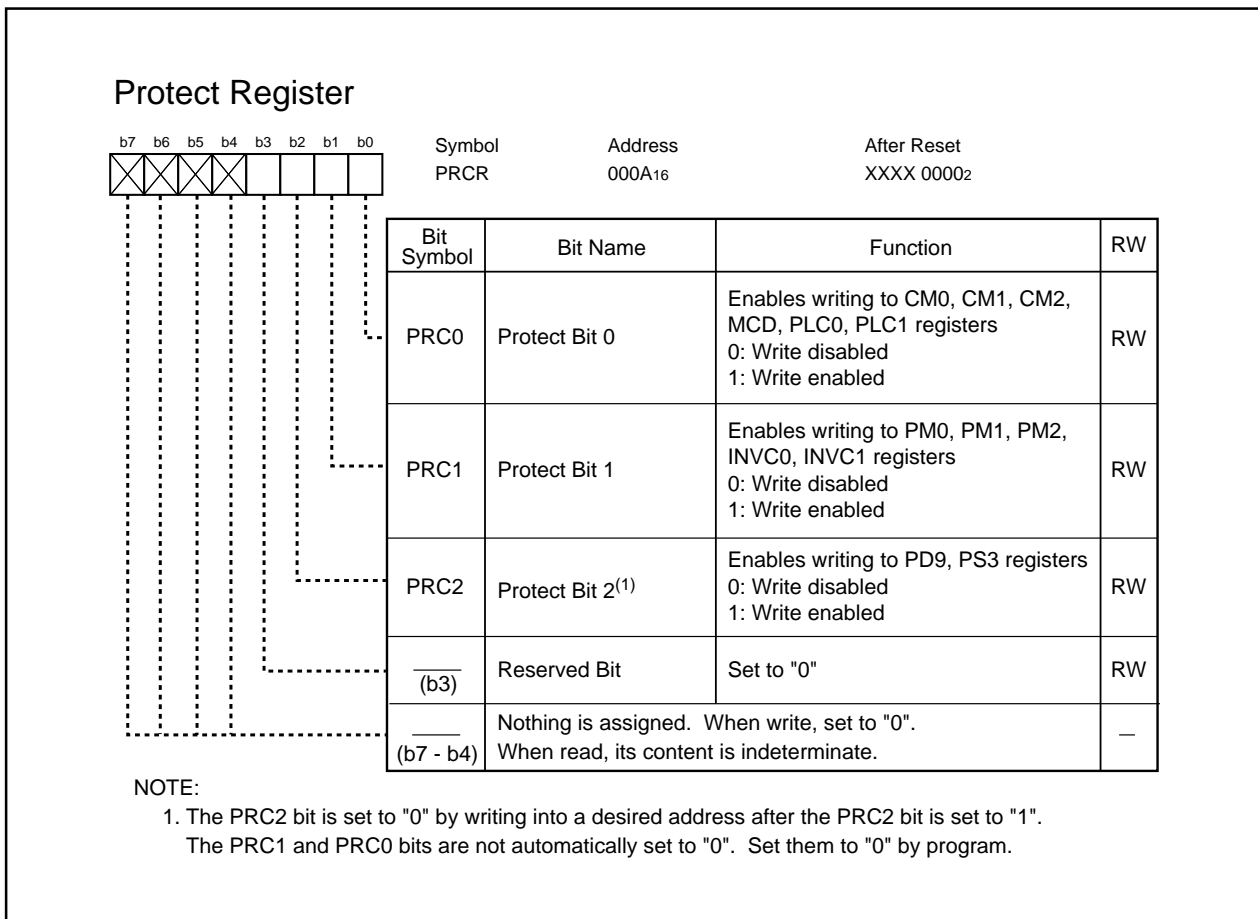
## 9. Protection

The protection function protects important registers from being easily overwritten when a program runs out of control.

Figure 9.1 shows the PRCR register. Each bit in the PRCR register protects the following registers:

- The PRC0 bit protects the CM0, CM1, CM2, MCD, PLC0 and PLC1 registers;
- The PRC1 bit protects the PM0, PM1, PM2, INVC0 and INVC1 registers;
- The PRC2 bit protects the PD9 and PS3 registers;

The PRC2 bit is set to "0" (write disabled) when data is written to a desired address after setting the PRC2 bit to "1" (write enabled). Set the PD9 and PS3 registers immediately after setting the PRC2 bit in the PRCR register to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the following instruction. The PRC1 and PRC0 bits are not set to "0" even if data is written to desired addresses. Set the PRC1 and PRC0 bits to "0" by program.

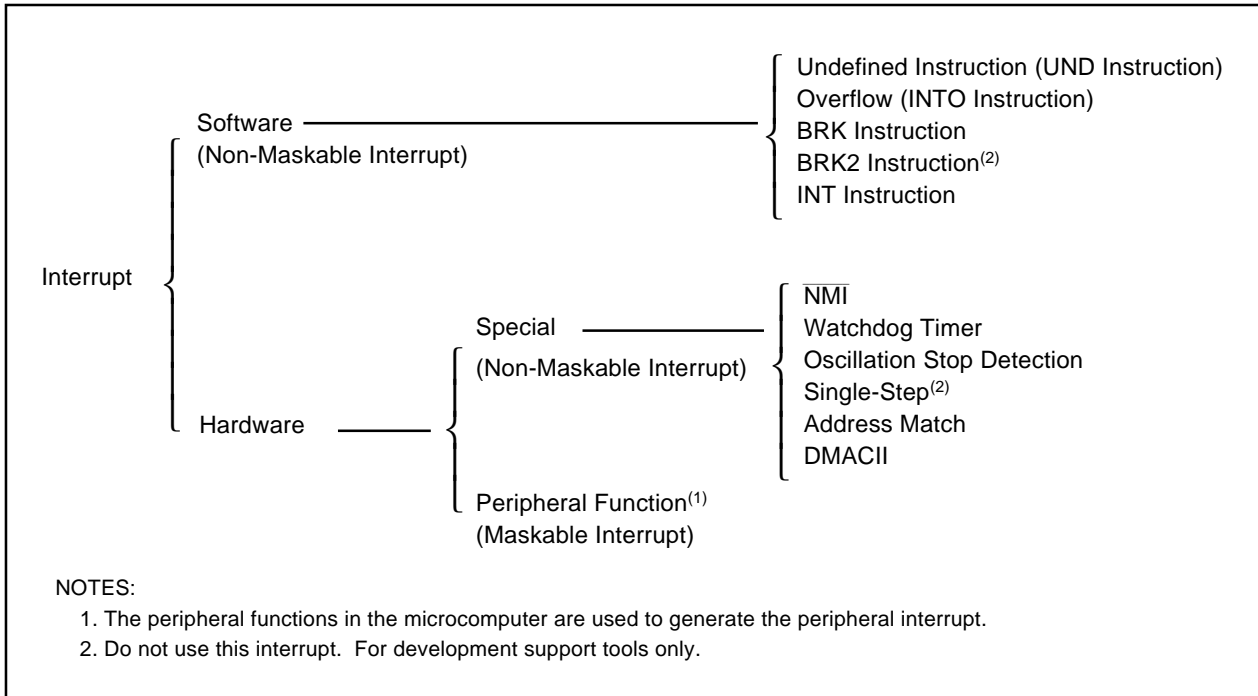


**Figure 9.1 PRCR Register**

# 10. Interrupts

## 10.1 Types of Interrupts

Figure 10.1 shows types of interrupts.



**Figure 10.1 Interrupts**

- Maskable Interrupt

The I flag enables or disables an interrupt.

The interrupt priority order based on interrupt priority level **can be changed**.

- Non-Maskable Interrupt

The I flag does not enable nor disable an interrupt .

The interrupt priority order based on interrupt priority level **cannot be changed**.

## 10.2 Software Interrupts

Software interrupt occurs when an instruction is executed. The software interrupts are non-maskable interrupts.

### 10.2.1 Undefined Instruction Interrupt

The undefined instruction interrupt occurs when the UND instruction is executed.

### 10.2.2 Overflow Interrupt

The overflow interrupt occurs when the O flag in the FLG register is set to "1" (overflow of arithmetic operation) and the INTO instruction is executed.

Instructions to set the O flag are :

ABS, ADC, ADCF, ADD, ADDX, CMP, CMPX, DIV, DIVU, DIVX, NEG, RMPA, SBB, SCMPU, SHA, SUB, SUBX

### 10.2.3 BRK Interrupt

The BRK interrupt occurs when the BRK instruction is executed.

### 10.2.4 BRK2 Interrupt

The BRK2 interrupt occurs when the BRK2 instruction is executed.

Do not use this interrupt. For development support tools only.

### 10.2.5 INT Instruction Interrupt

The INT instruction interrupt occurs when the INT instruction is executed. The INT instruction can select software interrupt numbers 0 to 63. Software interrupt numbers 8 to 48 are assigned to the vector table used for the peripheral function interrupt. Therefore, the microcomputer executes the same interrupt routine when the INT instruction is executed as when a peripheral function interrupt occurs.

When the INT instruction is executed, the FLG register and PC are saved to the stack. PC also stores the relocatable vector of specified software interrupt numbers. Where the stack is saved varies depending on a software interrupt number. ISP is selected as the stack for software interrupt numbers 0 to 31 (setting the U flag to "0"). SP, which is set before the INT instruction is executed, is selected as the stack for software interrupt numbers 32 to 63 (the U flag is not changed).

With the peripheral function interrupt, the FLG register is saved and the U flag is set to "0" (ISP select) when an interrupt request is acknowledged. With software interrupt numbers 32 to 48, SP to be used varies depending on whether the interrupt is generated by the peripheral function interrupt request or by the INT instruction.

## 10.3 Hardware Interrupts

Special interrupts and peripheral function interrupts are available as hardware interrupts.

### 10.3.1 Special Interrupts

Special interrupts are non-maskable interrupts.

#### 10.3.1.1 $\overline{\text{NMI}}$ Interrupt

The  $\overline{\text{NMI}}$  interrupt occurs when a signal applied to the  $\overline{\text{NMI}}$  pin changes from a high-level ("H") signal to a low-level ("L") signal. Refer to **10.8 NMI Interrupt** for details.

#### 10.3.1.2 Watchdog Timer Interrupt

The watchdog timer interrupt occurs when a count source of the watchdog timer underflows. Refer to **11. Watchdog Timer** for details.

#### 10.3.1.3 Oscillation Stop Detection Interrupt

The oscillation stop detection interrupt occurs when the microcomputer detects a main clock oscillation stop. Refer to **8. Clock Generation Circuit** for details.

#### 10.3.1.4 Single-Step Interrupt

Do not use the single-step interrupt. For development support tool only.

#### 10.3.1.5 Address Match Interrupt

The address match interrupt occurs immediately before executing an instruction that is stored into an address indicated by the RMADi register (i=0 to 7) when the AIERi bit in the AIER register is set to "1" (address match interrupt enabled). Set the starting address of the instruction in the RMADi register. The address match interrupt does not occur when a table data or addresses of the instruction other than the starting address, if the instruction has multiple addresses, is set. Refer to **10.10 Address Match Interrupt** for details.

### 10.3.2 Peripheral Function Interrupt

The peripheral function interrupt occurs when a request from the peripheral functions in the microcomputer is acknowledged. The peripheral function interrupts and software interrupt numbers 8 to 48 for the INT instruction use the same interrupt vector table. The peripheral function interrupt is a maskable interrupt.

See **Table 10.2** about how the peripheral function interrupt occurs. Refer to the descriptions of each function for details.

## 10.4 High-Speed Interrupt

The high-speed interrupt executes an interrupt sequence in five cycles and returns from the interrupt in three cycles.

When the FSIT bit in the RLVL register is set to "1" (interrupt priority level 7 available for the high-speed interrupt), the ILVL2 to ILVL0 bits in the interrupt control registers can be set to "1112" (level 7) to use the high-speed interrupt.

Only one interrupt can be set as the high-speed interrupt. When using the high-speed interrupt, do not set multiple interrupts to interrupt priority level 7. Set the DMAII bit in the RLVL register to "0" (interrupt priority level 7 available for interrupts).

Set the starting address of the high-speed interrupt routine in the VCT register.

When the high-speed interrupt is acknowledged, the FLG register is saved into the SVF register and PC is saved into the SVP register. The program is executed from an address indicated by the VCT register.

Execute the FREIT instruction to return from the high-speed interrupt routine.

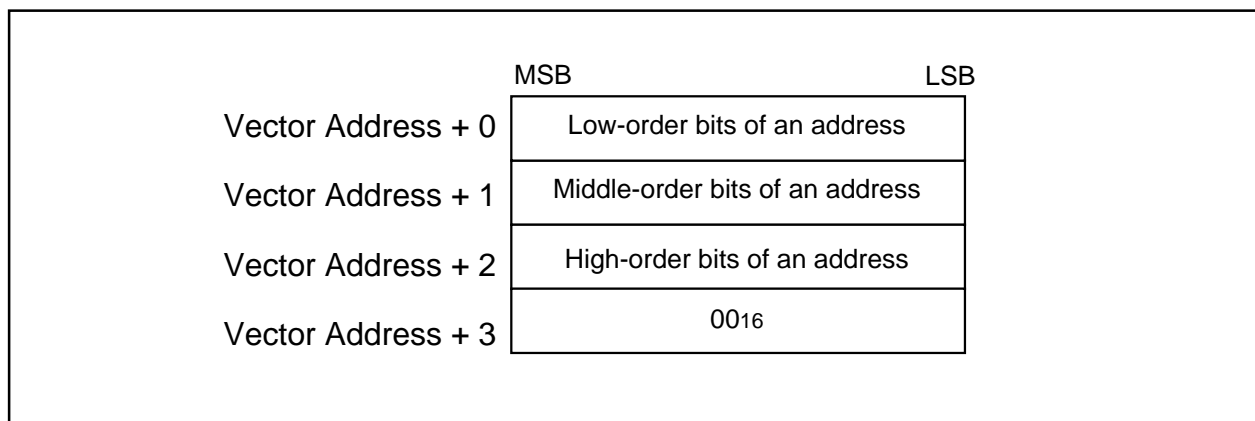
The values saved into the SVF and SVP registers are restored to the FLG register and PC by executing the FREIT instruction.

The high-speed interrupt and the DMA2 and DMA3 use the same register. When using the high-speed interrupt, neither DMA2 nor DMA3 is available. DMA0 and DMA1 can be used.

## 10.5 Interrupts and Interrupt Vectors

There are four bytes in one vector. Set the starting address of interrupt routine in each vector table. When an interrupt request is acknowledged, the interrupt routine is executed from the address set in the interrupt vectors.

Figure 10.2 shows the interrupt vector.



**Figure 10.2** Interrupt Vector



### 10.5.1 Fixed Vector Tables

The fixed vector tables are allocated addresses FFFFDC<sub>16</sub> to FFFFFFF<sub>16</sub>. Table 10.1 lists the fixed vector tables.

**Table 10.1 Fixed Vector Table**

Interrupt Generated by	Vector Addresses Low address to High address	Remarks	Reference
Undefined Instruction	FFFFDC <sub>16</sub> to FFFFDF <sub>16</sub>		M32C/80 Series Software Manual
Overflow	FFFFE0 <sub>16</sub> to FFFFFE3 <sub>16</sub>		
BRK Instruction	FFFFE4 <sub>16</sub> to FFFFFE7 <sub>16</sub>	If the content of address FFFFFE7 <sub>16</sub> is FF <sub>16</sub> , a program is executed from the address stored into software interrupt number 0 in the relocatable vector table	
Address Match	FFFFE8 <sub>16</sub> to FFFFFEB <sub>16</sub>		
-	FFFFEC <sub>16</sub> to FFFFFEF <sub>16</sub>	Reserved space	
Watchdog Timer	FFFFF0 <sub>16</sub> to FFFFFF3 <sub>16</sub>	These addresses are used for the watchdog timer interrupt and oscillation stop detection interrupt	Reset, Clock Generation Circuit, Watchdog Timer
-	FFFFF4 <sub>16</sub> to FFFFFF7 <sub>16</sub>	Reserved space	
NMI	FFFFF8 <sub>16</sub> to FFFFFFB <sub>16</sub>		
Reset	FFFFFC <sub>16</sub> to FFFFFFF <sub>16</sub>		Reset

### 10.5.2 Relocatable Vector Tables

The relocatable vector tables occupy 256 bytes from the starting address set in the INTB register. Table 10.2 lists the relocatable vector tables.

Set an even address as the starting address of the vector table set in the INTB register to increase interrupt sequence execution rate.

**Table 10.2 Relocatable Vector Tables**

Interrupt Generated by	Vector Table Address Address(L) to Address(H) <sup>(1)</sup>	Software Interrupt Number	Reference
BRK Instruction <sup>(2)</sup>	+0 to +3 (0000 <sub>16</sub> to 0003 <sub>16</sub> )	0	M32C/80 Series
Reserved Space	+4 to +31 (0004 <sub>16</sub> to 001F <sub>16</sub> )	1 to 7	Software Manual
DMA0	+32 to +35 (0020 <sub>16</sub> to 0023 <sub>16</sub> )	8	DMAC
DMA1	+36 to +39 (0024 <sub>16</sub> to 0027 <sub>16</sub> )	9	
DMA2	+40 to +43 (0028 <sub>16</sub> to 002B <sub>16</sub> )	10	
DMA3	+44 to +47 (002C <sub>16</sub> to 002F <sub>16</sub> )	11	
Timer A0	+48 to +51 (0030 <sub>16</sub> to 0033 <sub>16</sub> )	12	Timer A
Timer A1	+52 to +55 (0034 <sub>16</sub> to 0037 <sub>16</sub> )	13	
Timer A2	+56 to +59 (0038 <sub>16</sub> to 003B <sub>16</sub> )	14	
Timer A3	+60 to +63 (003C <sub>16</sub> to 003F <sub>16</sub> )	15	
Timer A4	+64 to +67 (0040 <sub>16</sub> to 0043 <sub>16</sub> )	16	
UART0 Transmission, NACK <sup>(3)</sup>	+68 to +71 (0044 <sub>16</sub> to 0047 <sub>16</sub> )	17	Serial I/O
UART0 Reception, ACK <sup>(3)</sup>	+72 to +75 (0048 <sub>16</sub> to 004B <sub>16</sub> )	18	
UART1 Transmission, NACK <sup>(3)</sup>	+76 to +79 (004C <sub>16</sub> to 004F <sub>16</sub> )	19	
UART1 Reception, ACK <sup>(3)</sup>	+80 to +83 (0050 <sub>16</sub> to 0053 <sub>16</sub> )	20	
Timer B0	+84 to +87 (0054 <sub>16</sub> to 0057 <sub>16</sub> )	21	Timer B
Timer B1	+88 to +91 (0058 <sub>16</sub> to 005B <sub>16</sub> )	22	
Timer B2	+92 to +95 (005C <sub>16</sub> to 005F <sub>16</sub> )	23	
Timer B3	+96 to +99 (0060 <sub>16</sub> to 0063 <sub>16</sub> )	24	
Timer B4	+100 to +103 (0064 <sub>16</sub> to 0067 <sub>16</sub> )	25	
INT5	+104 to +107 (0068 <sub>16</sub> to 006B <sub>16</sub> )	26	Interrupt
INT4	+108 to +111 (006C <sub>16</sub> to 006F <sub>16</sub> )	27	
INT3	+112 to +115 (0070 <sub>16</sub> to 0073 <sub>16</sub> )	28	
INT2	+116 to +119 (0074 <sub>16</sub> to 0077 <sub>16</sub> )	29	
INT1	+120 to +123 (0078 <sub>16</sub> to 007B <sub>16</sub> )	30	
INT0	+124 to +127 (007C <sub>16</sub> to 007F <sub>16</sub> )	31	
Timer B5	+128 to +131 (0080 <sub>16</sub> to 0083 <sub>16</sub> )	32	Timer B
UART2 Transmission, NACK <sup>(3)</sup>	+132 to +135 (0084 <sub>16</sub> to 0087 <sub>16</sub> )	33	Serial I/O
UART2 Reception, ACK <sup>(3)</sup>	+136 to +139 (0088 <sub>16</sub> to 008B <sub>16</sub> )	34	
UART3 Transmission, NACK <sup>(3)</sup>	+140 to +143 (008C <sub>16</sub> to 008F <sub>16</sub> )	35	
UART3 Reception, ACK <sup>(3)</sup>	+144 to +147 (0090 <sub>16</sub> to 0093 <sub>16</sub> )	36	
UART4 Transmission, NACK <sup>(3)</sup>	+148 to +151 (0094 <sub>16</sub> to 0097 <sub>16</sub> )	37	
UART4 Reception, ACK <sup>(3)</sup>	+152 to +155 (0098 <sub>16</sub> to 009B <sub>16</sub> )	38	

**Table 10.2 Relocatable Vector Tables (Continued)**

Interrupt Generated by	Vector Table Address Address(L) to Address(H) <sup>(1)</sup>	Software Interrupt Number	Reference
Bus Conflict Detect, Start Condition Detect, Stop Condition Detect (UART2) <sup>(3)</sup> ,	+156 to +159 (009C <sub>16</sub> to 009F <sub>16</sub> )	39	Serial I/O
Bus Conflict Detect, Start Condition Detect, Stop Condition Detect (UART3/UART0) <sup>(4)</sup>	+160 to +163 (00A0 <sub>16</sub> to 00A3 <sub>16</sub> )	40	
Bus Conflict Detect, Start Condition Detect, Stop Condition Detect (UART4/UART1) <sup>(4)</sup>	+164 to +167 (00A4 <sub>16</sub> to 00A7 <sub>16</sub> )	41	
A/D0	+168 to +171 (00A8 <sub>16</sub> to 00AB <sub>16</sub> )	42	A/D Converter
Key Input	+172 to +175 (00AC <sub>16</sub> to 00AF <sub>16</sub> )	43	Interrupts
Intelligent I/O Interrupt 0	+176 to +179 (00B0 <sub>16</sub> to 00B3 <sub>16</sub> )	44	Intelligent I/O
Intelligent I/O Interrupt 1	+180 to +183 (00B4 <sub>16</sub> to 00B7 <sub>16</sub> )	45	
Intelligent I/O Interrupt 2	+184 to +187 (00B8 <sub>16</sub> to 00BB <sub>16</sub> )	46	
Intelligent I/O Interrupt 3	+188 to +191 (00BC <sub>16</sub> to 00BF <sub>16</sub> )	47	
Intelligent I/O Interrupt 4	+192 to +195 (00C0 <sub>16</sub> to 00C3 <sub>16</sub> )	48	
INT Instruction <sup>(2)</sup>	+0 to +3 (0000 <sub>16</sub> to 0003 <sub>16</sub> ) to +252 to +255 (00FC <sub>16</sub> to 00FF <sub>16</sub> )	0 to 63	Interrupts

## NOTES:

1. These addresses are relative to those in the INTB register.
2. The I flag does not disable interrupts.
3. In I<sup>2</sup>C mode, NACK, ACK or start/stop condition detection causes interrupts to be generated.
4. The IFSR6 bit in the IFSR register determines whether these addresses are used for an interrupt in UART0 or in UART3.  
The IFSR7 bit in the IFSR register determines whether these addresses are used for an interrupt in UART1 or in UART4.

## 10.6 Interrupt Request Acknowledgement

Software interrupts and special interrupts occur when conditions to generate an interrupt are met.

The peripheral function interrupts are acknowledged when all conditions below are met.

- I flag = "1"
- IR bit = "1"
- ILVL2 to ILVL0 bits > IPL

The I flag, IPL, IR bit and ILVL2 to ILVL0 bits are independent of each other. The I flag and IPL are in the FLG register. The IR bit and ILVL2 to ILVL0 bits are in the interrupt control register.

### 10.6.1 I Flag and IPL

The I flag enables or disables maskable interrupts. When the I flag is set to "1" (enable), all maskable interrupts are enabled; when the I flag is set to "0" (disable), they are disabled. The I flag is automatically set to "0" after reset.

IPL, consisting of three bits, indicates the interrupt priority level from level 0 to level 7.

If a requested interrupt has higher priority level than indicated by IPL, the interrupt is acknowledged.

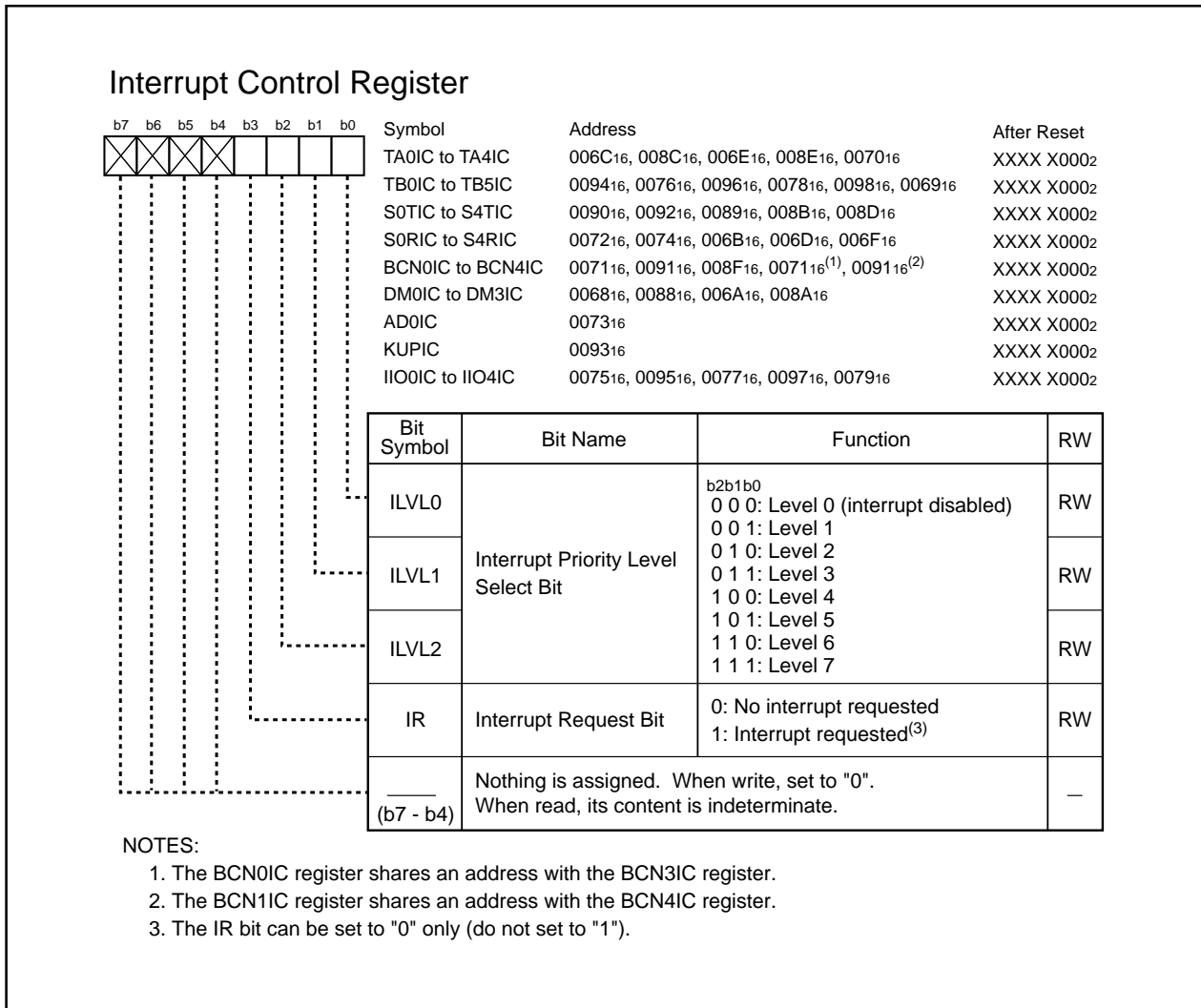
Table 10.3 lists interrupt priority levels associated with IPL.

**Table 10.3 Interrupt Priority Levels**

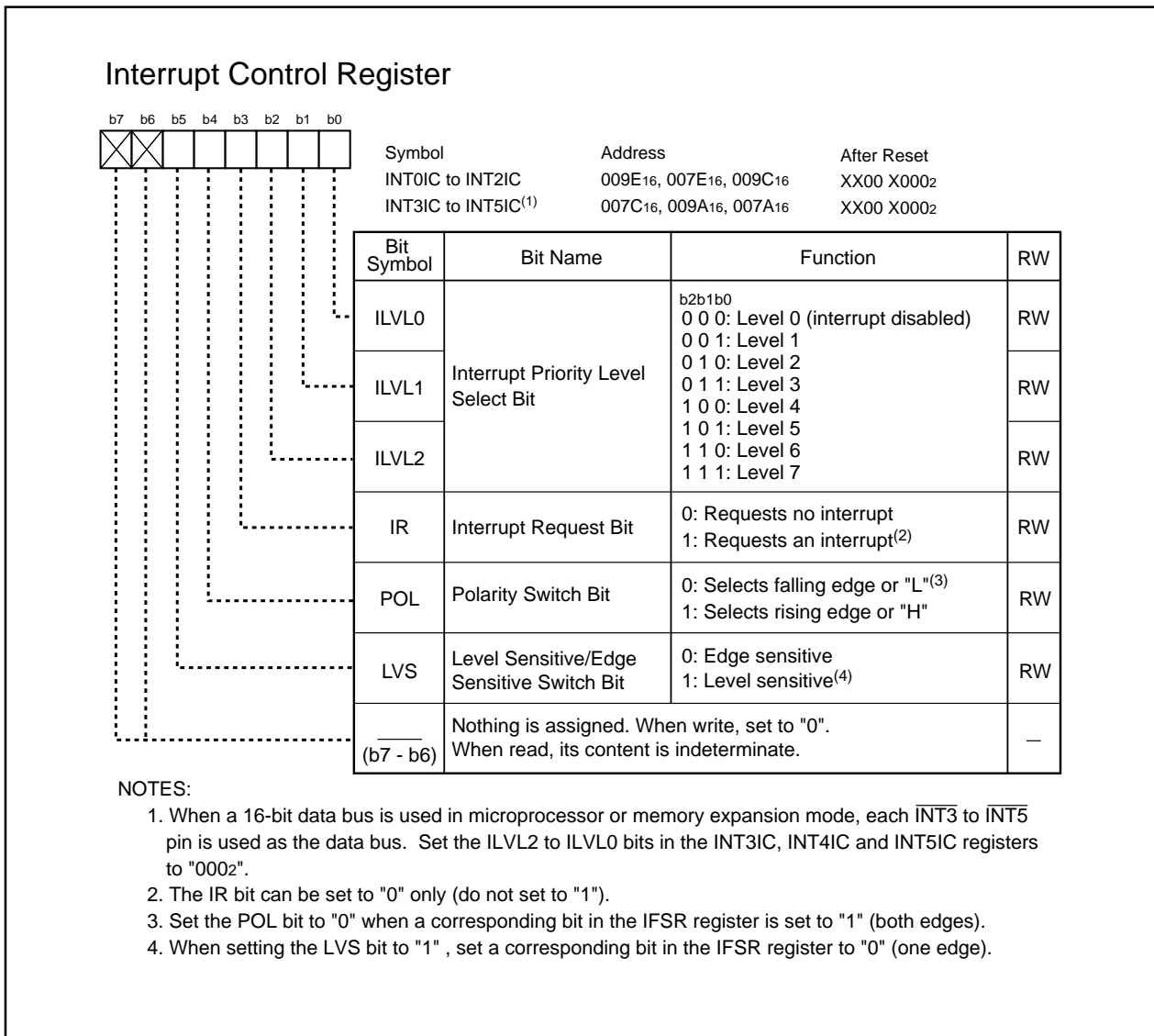
IPL2	IPL1	IPL0	Interrupt Priority Levels
0	0	0	Level 1 and above
0	0	1	Level 2 and above
0	1	0	Level 3 and above
0	1	1	Level 4 and above
1	0	0	Level 5 and above
1	0	1	Level 6 and above
1	1	0	Level 7 and above
1	1	1	All maskable interrupts are disabled

### 10.6.2 Interrupt Control Register and RLVL Register

The peripheral function interrupts use interrupt control registers to control each interrupt. Figures 10.3 and 10.4 show the interrupt control register. Figure 10.5 shows the RLVL register.



**Figure 10.3 Interrupt Control Register (1)**



**Figure 10.4 Interrupt Control Register (2)**

### 10.6.2.1 ILVL2 to ILVL0 Bits

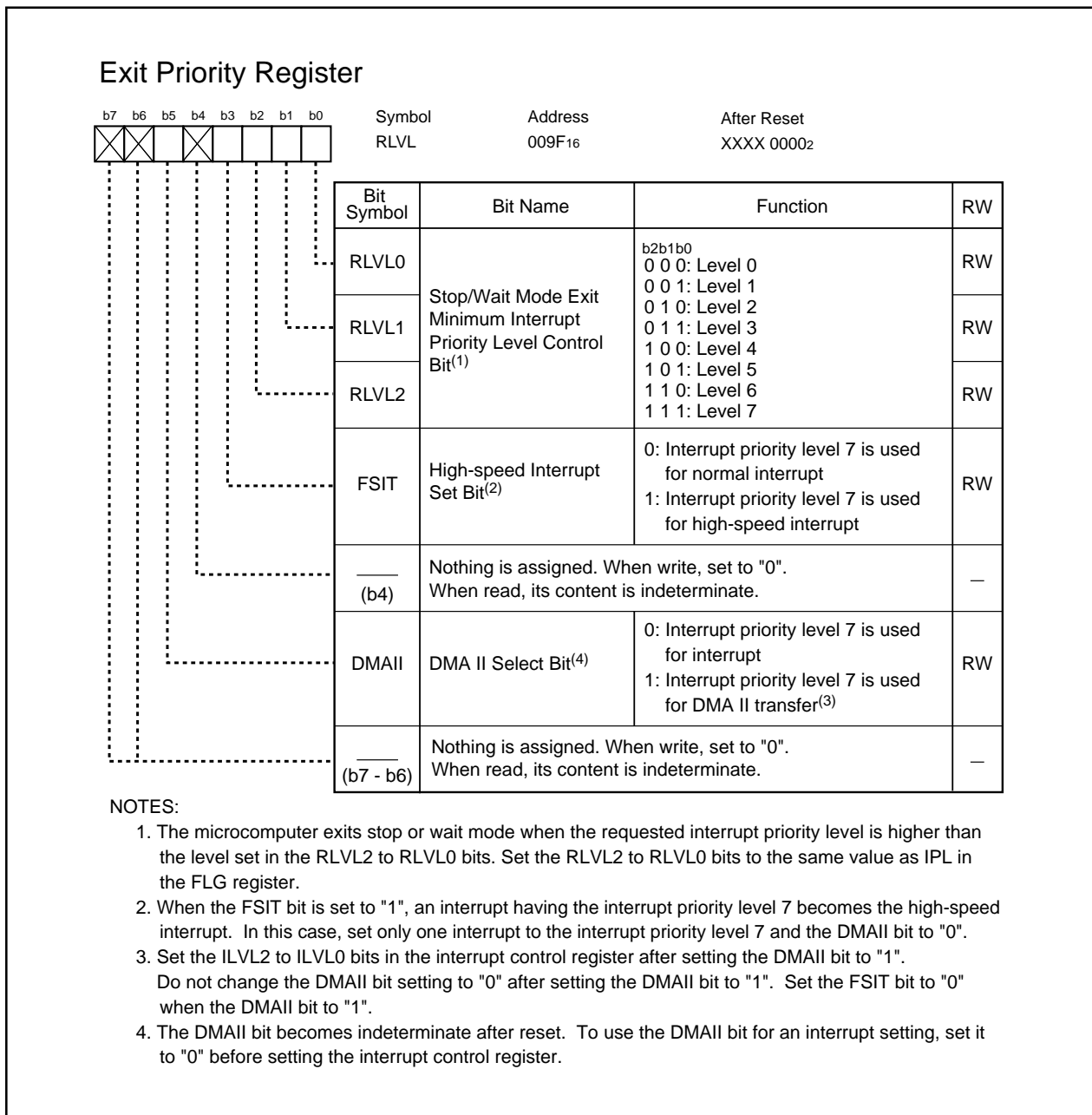
The ILVL2 to ILVL0 bits determines an interrupt priority level. The higher the interrupt priority level is, the higher interrupt priority is.

When an interrupt request is generated, its interrupt priority level is compared to IPL. This interrupt is acknowledged only when its interrupt priority level is higher than IPL. When the ILVL2 to ILVL0 bits are set to "0002" (level 0), its interrupt is ignored.

### 10.6.2.2 IR Bit

The IR bit is automatically set to "1" (interrupt requested) when an interrupt request is generated. The IR bit is automatically set to "0" (no interrupt requested) after an interrupt request is acknowledged and an interrupt routine in the corresponding interrupt vector is executed.

The IR bit can be set to "0" by program. Do not set to "1".



**Figure 10.5 RLVL Register**

### 10.6.2.3 RLVL2 to RLVL0 Bits

When using an interrupt to exit stop or wait mode, refer to **8.5.2 Wait Mode** and **8.5.3 Stop Mode** for details.

### 10.6.3 Interrupt Sequence

The interrupt sequence is performed between an interrupt request acknowledgment and interrupt routine execution.

When an interrupt request is generated while an instruction is executed, the CPU determines its interrupt priority level after the instruction is completed. The CPU starts the interrupt sequence from the following cycle. However, in regards to the SCMPU, SIN, SMOVB, SMOVF, SMOVU, SSTR, SOUT or RMPA instruction, if an interrupt request is generated while executing the instruction, the microcomputer suspends the instruction to start the interrupt sequence.

The interrupt sequence is performed as follows:

- (1) The CPU obtains interrupt information (interrupt number and interrupt request level) by reading address 000000<sub>16</sub> (address 000002<sub>16</sub> for the high-speed interrupt). 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) Each bit in the FLG register is set as follows:
  - The I flag is set to "0" (interrupt disabled)
  - The D flag is set to "0" (single-step disabled)
  - The U flag is set to "0" (ISP selected)
- (4) A temporary register within the CPU is saved to the stack; or to the SVF register for the high-speed interrupt.
- (5) PC is saved to the stack; or to the SVP register for the high-speed interrupt.
- (6) The interrupt priority level of the acknowledged interrupt is set in IPL .
- (7) A relocatable vector corresponding to the acknowledged interrupt is stored into PC.

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

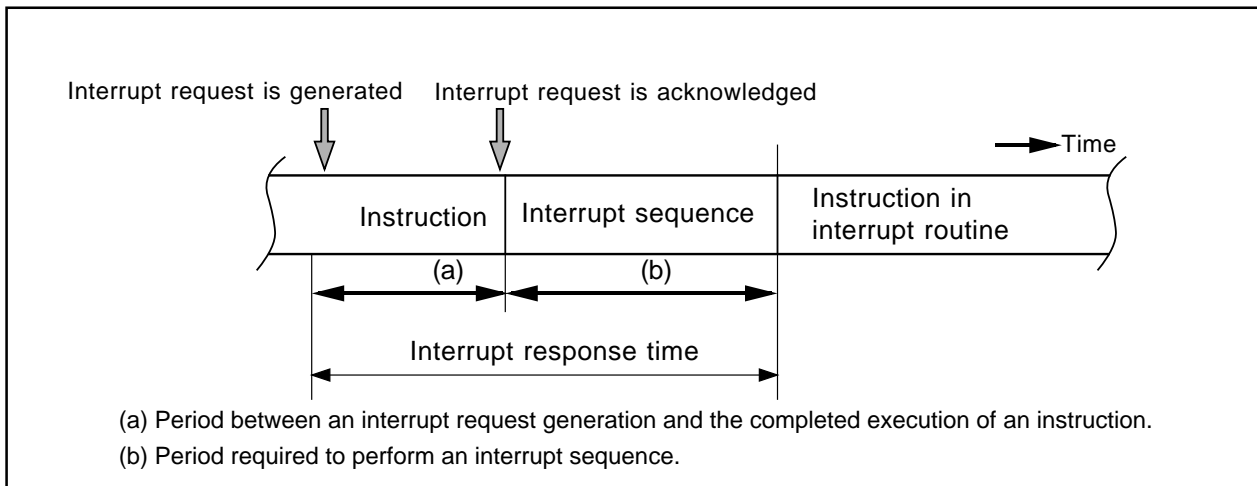
**NOTE:**

1. Temporary register cannot be modified by users.



### 10.6.4 Interrupt Response Time

Figure 10.6 shows an interrupt response time. Interrupt response time is the period between an interrupt generation and the execution of the first instruction in an interrupt routine. Interrupt response time includes the period between an interrupt request generation and the completed execution of an instruction ((a) on Figure 10.6) and the period required to perform an interrupt sequence ((b) on Figure 10.6).



**Figure 10.6 Interrupt Response Time**

Time (a) varies depending on an instruction being executed. The DIV, DIVX and DIVU instructions require the longest time (a); 42 cycles when an immediate value or register is set as the divisor. When the divisor is a value in the memory, the following value is added.

- Normal addressing :  $2 + X$
- Index addressing :  $3 + X$
- Indirect addressing :  $5 + X + 2Y$
- Indirect index addressing :  $6 + X + 2Y$

X is the number of wait states for a divisor space. Y is the number of wait states for the space that stores indirect addresses. If X and Y are in an odd address or in 8-bit bus space, the X and Y value must be doubled.

Table 10.4 lists time (b), shown Figure 10.6.

**Table 10.4 Interrupt Sequence Execution Time**

Interrupt	Interrupt Vector Address	16-Bit Bus	8-Bit Bus
Peripheral Function	Even address	14 cycles	16 cycles
	Odd address <sup>(1)</sup>	16 cycles	16 cycles
INT Instruction	Even address	12 cycles	14 cycles
	Odd address <sup>(1)</sup>	14 cycles	14 cycles
NMI Watchdog Timer Undefined Instruction Address Match	Even address <sup>(2)</sup>	13 cycles	15 cycles
Overflow	Even address <sup>(2)</sup>	14 cycles	16 cycles
BRK Instruction (relocatable vector table)	Even address	17 cycles	19 cycles
	Odd address <sup>(1)</sup>	19 cycles	19 cycles
BRK Instruction (fixed vector table)	Even address <sup>(2)</sup>	19 cycles	21 cycles
High-speed Interrupt	Vector table is internal register	5 cycles	

## NOTES:

1. Allocate interrupt vectors in even addresses.
2. Vectors are fixed to even addresses.

**10.6.5 IPL Change when Interrupt Request is Acknowledged**

When a peripheral function interrupt request is acknowledged, IPL sets the priority level for the acknowledged interrupt.

Software interrupts and special interrupts have no interrupt priority level. If an interrupt request that has no interrupt priority level is acknowledged, the value shown in Table 10.5 is set in IPL as the interrupt priority level.

**Table 10.5 Interrupts without Interrupt Priority Levels and IPL**

Interrupt Source	Level Set to IPL
Watchdog Timer, NMI, Oscillation Stop Detection	7
Reset	0
Software, Address Match	Not changed

### 10.6.6 Saving a Register

In the interrupt sequence, the FLG register and PC are saved to the stack.

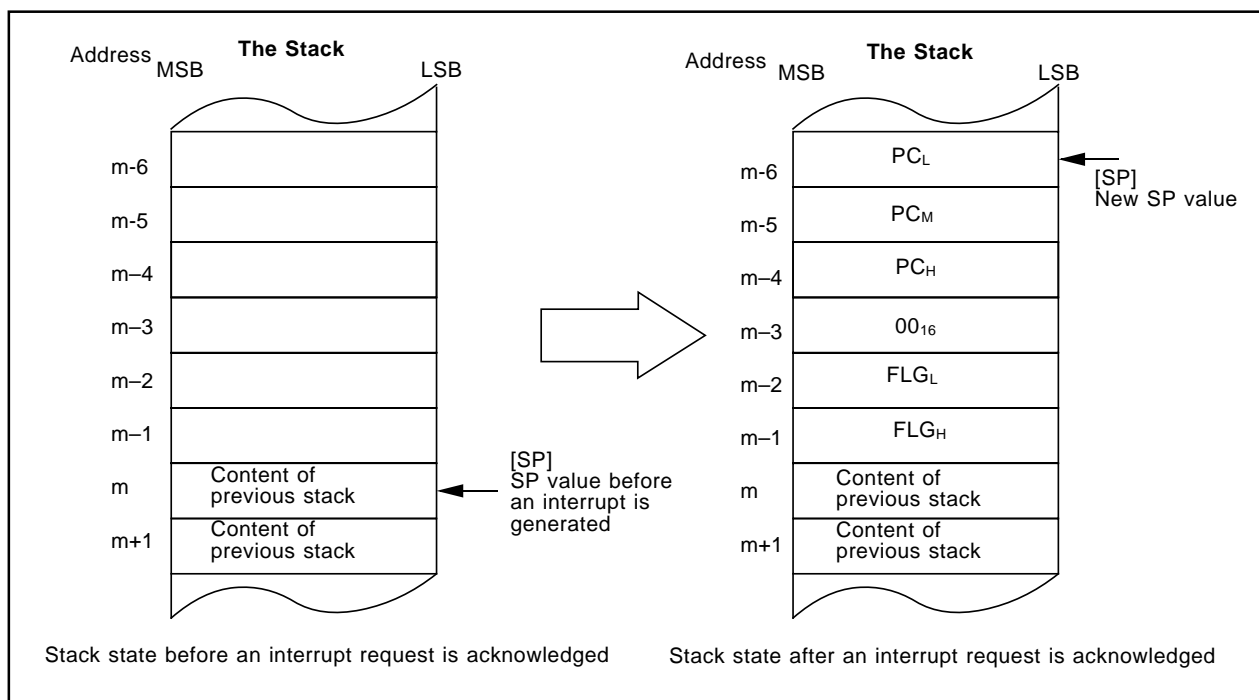
After the FLG register is saved to the stack, 16 high-order bits and 16 low-order bits of PC, extended to 32 bits, are saved to the stack. Figure 10.7 shows stack states before and after an interrupt request is acknowledged.

Other important registers are saved by program at the beginning of an interrupt routine. The PUSHM instruction can save several registers<sup>(1)</sup> in the register bank used.

Refer to **10.4 High-Speed Interrupt** for the high-speed interrupt.

#### NOTE:

1. Can be selected from the R0, R1, R2, R3, A0, A1, SB and FB registers.



**Figure 10.7 Stack States**

### 10.6.7 Restoration from Interrupt Routine

When the REIT instruction is executed at the end of an interrupt routine, the FLG register and PC before the interrupt sequence is performed, which have been saved to the stack, are automatically restored. The program, executed before an interrupt request was acknowledged, starts running again. Refer to **10.4 High-Speed Interrupt** for the high-speed interrupt.

Restore registers saved by program in an interrupt routine by the POPM instruction or others before the REIT and FREIT instructions. Register bank is switched back to the bank used prior to the interrupt sequence by the REIT or FREIT instruction.

### 10.6.8 Interrupt Priority

If two or more interrupt requests are sampled at the same sampling points (a timing to detect whether an interrupt request is generated or not), the interrupt with the highest priority is acknowledged.

Set the ILVL2 to ILVL0 bits to select the desired priority level for maskable interrupts (peripheral function interrupt).

Priority levels of special interrupts such as reset (reset has the highest priority) and watchdog timer are set by hardware. Figure 10.8 shows priority levels of hardware interrupts.

The interrupt priority does not affect software interrupts. Executing instruction causes the microcomputer to execute an interrupt routine.

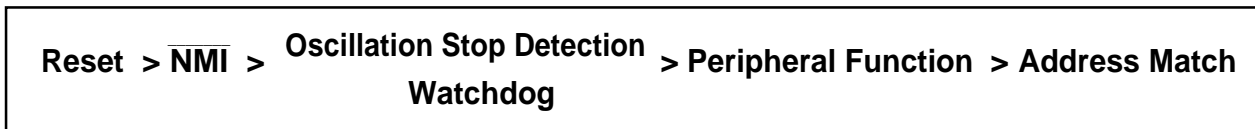


Figure 10.8 Interrupt Priority

### 10.6.9 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 10.9 shows the interrupt priority level select circuit.

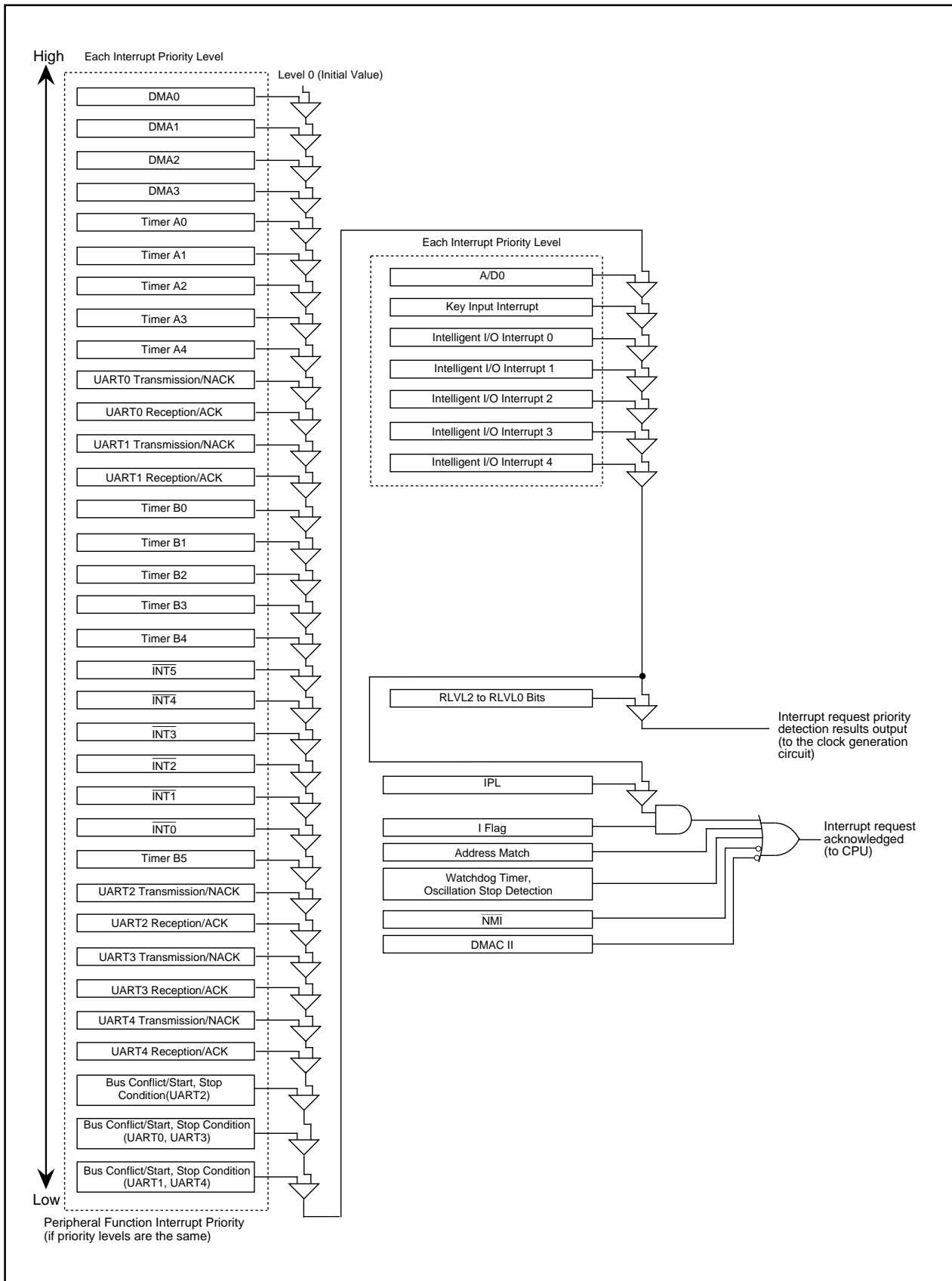


Figure 10.9 Interrupt Priority Level Select Circuit

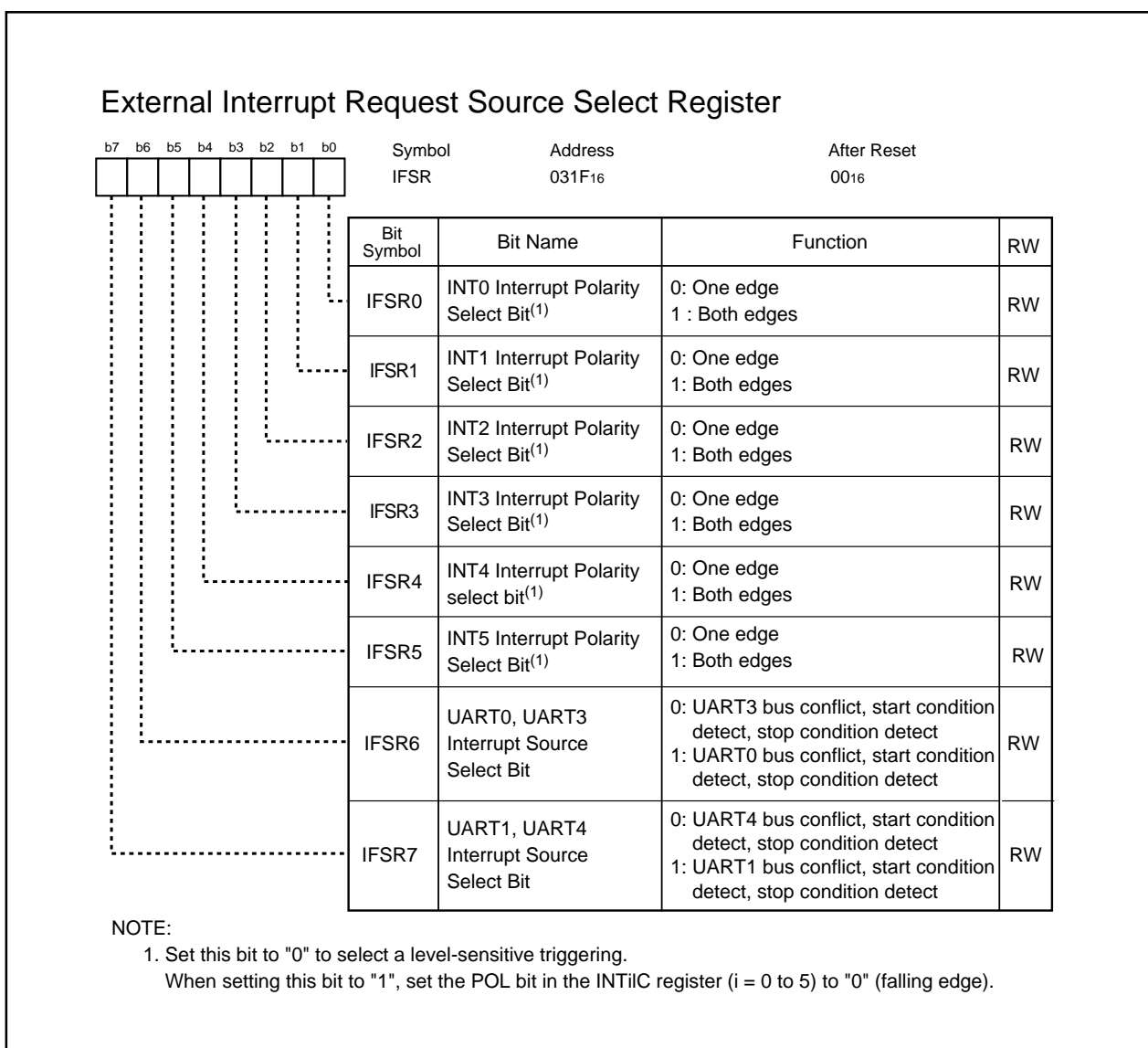
## 10.7 $\overline{\text{INT}}$ Interrupt

External input generates the  $\overline{\text{INT}}_i$  interrupt ( $i = 0$  to  $5$ ). The LVS bit in the  $\text{INTiIC}$  register selects either edge sensitive triggering to generate an interrupt on any edge or level sensitive triggering to generate an interrupt at an applied signal level. The POL bit in the  $\text{INTiIC}$  register determines the polarity.

For edge sensitive, when the  $\text{IFSR}_i$  bit in the IFSR register is set to "1", an interrupt occurs on both rising and falling edges of the external input. If the  $\text{IFSR}_i$  bit is set to "0", set the POL bit in the corresponding register to "0" (falling edge).

For level sensitive, set the  $\text{IFSR}_i$  bit to "0" (single edge). When the  $\overline{\text{INT}}_i$  pin input level reaches the level set in the POL bit, the IR bit in the  $\text{INTiIC}$  register is set to "1". The IR bit remains unchanged even if the  $\overline{\text{INT}}_i$  pin level is changed. The IR bit is set to "0" when the  $\overline{\text{INT}}_i$  interrupt is acknowledged or when the IR bit is written to "0" by program.

Figure 10.10 shows the IFSR register.



**Figure 10.10 IFSR Register**

## 10.8 $\overline{\text{NMI}}$ Interrupt

The  $\overline{\text{NMI}}$  interrupt<sup>(1)</sup> occurs when a signal applied to the  $\overline{\text{NMI}}$  pin changes from a high-level ("H") signal to a low-level ("L") signal. The  $\overline{\text{NMI}}$  interrupt is a non-maskable interrupt. Although the P85/ $\overline{\text{NMI}}$  pin is used as the  $\overline{\text{NMI}}$  interrupt input pin, the P8\_5 bit in the P8 register indicates the input level for this pin.

### NOTE:

1. When the  $\overline{\text{NMI}}$  interrupt is not used, connect the  $\overline{\text{NMI}}$  pin to VCC1 via a resistor. Because the  $\overline{\text{NMI}}$  interrupt cannot be ignored, the pin must be connected.

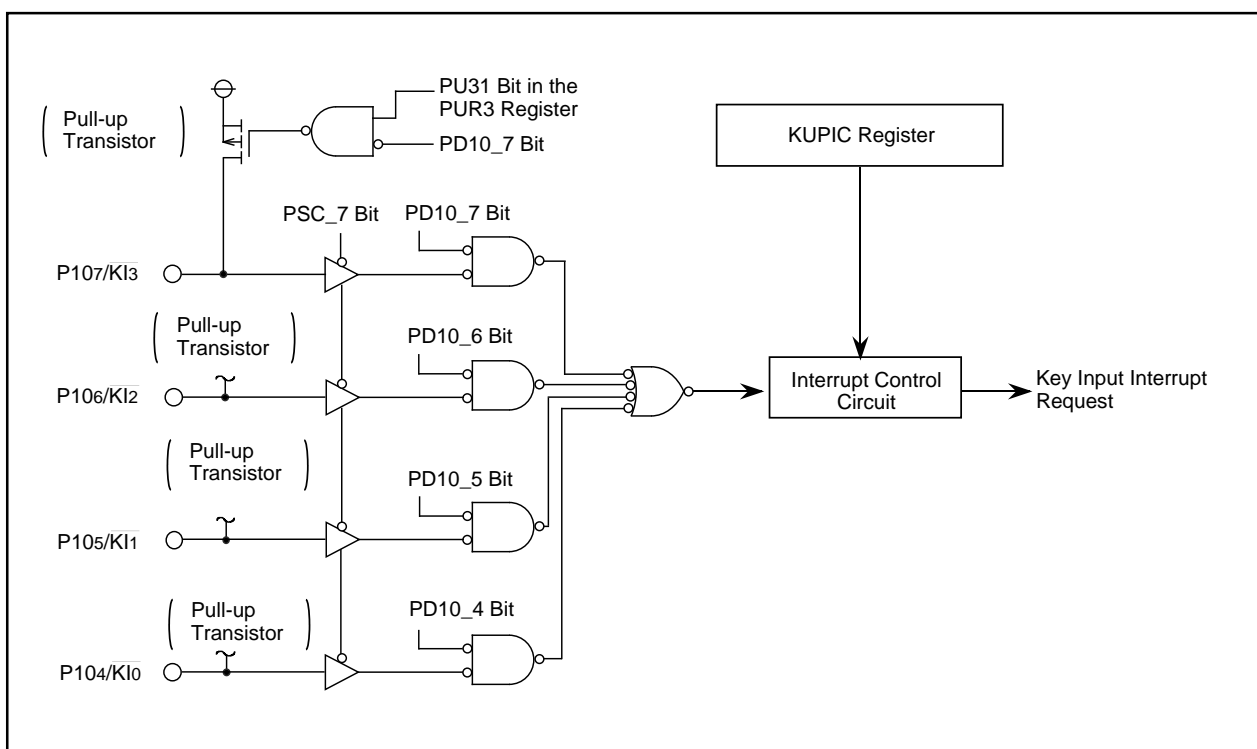
## 10.9 Key Input Interrupt

Key input interrupt request is generated when one of the signals applied to the P104 to P107 pins in input mode is on the falling edge. The key input interrupt can be also used as key-on wake-up function to exit wait or stop mode. To use the key input interrupt, do not use P104 to P107 as A/D input ports. Figure 10.11 shows a block diagram of the key input interrupt. When an "L" signal is applied to any pins in input mode, signals applied to other pins are not detected as an interrupt request signal.

When the PSC\_7 bit in the PSC register<sup>(2)</sup> is set to "1" (key input interrupt disabled), no key input interrupt occurs regardless of interrupt control register settings. When the PSC\_7 bit is set to "0", no input from a port pin is available even when in input mode.

### NOTE:

2. Refer to **22. Programmable I/O Ports** about the PSC register.



**Figure 10.11 Key Input Interrupt**

To use the intelligent I/O interrupt as a source to activate DMAC II, set the IRLT bit in the IIOiE register to "0" (interrupt request is used for DMAC, DMAC II) and enable an interrupt request source for the IIOiE register.

### 10.10 Address Match Interrupt

The address match interrupt occurs immediately before executing an instruction that is stored into an address indicated by the RMADi register (i=0 to 7). The address match interrupt can be set in eight addresses. The AIERi bit in the AIER register determines whether the interrupt is enabled or disabled. The I flag and IPL do not affect the address match interrupt.

Figure 10.12 shows registers associated with the address match interrupt.

The starting address of an instruction must be set in the RMADi register. The address match interrupt does not occur when a table data or addresses other than the starting address of the instruction is set.

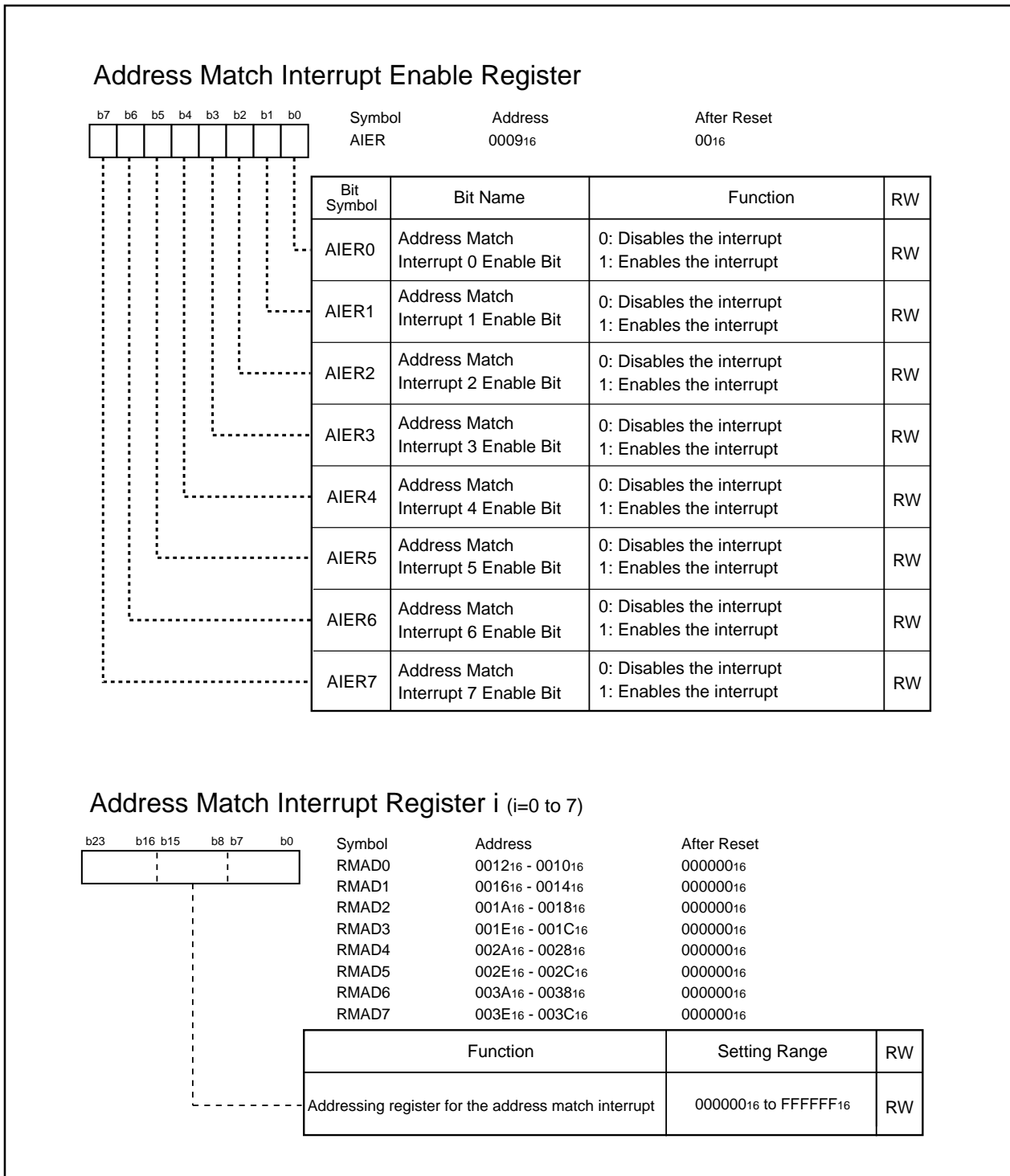


Figure 10.12 AIER Register and RMAD0 to RMAD7 Registers



### 10.11 Intelligent I/O Interrupt

The intelligent I/O interrupt is assigned to software interrupt numbers 44 to 48.

When using the intelligent I/O interrupt, set the IRLT bit in the IIOiE register ( $i = 0$  to 4) to "1" (interrupt request for interrupt used).

Various interrupt requests cause the intelligent I/O interrupt to occur. When an interrupt request is generated with each intelligent I/O or CAN functions, the corresponding bit in the IIOiR register is set to "1" (interrupt requested). When the corresponding bit in the IIOiE register is set to "1" (interrupt enabled), the IR bit in the corresponding IIOiC register is set to "1" (interrupt requested).

After the IR bit setting changes "0" to "1", the IR bit remains set to "1" when a bit in the IIOiR register is set to "1" by another interrupt request and the corresponding bit in the IIOiE register is set to "1".

Bits in the IIOiR register are not set to "0" automatically, even if an interrupt is acknowledged. Set each bit to "0" by program. If these bit settings are left "1", all generated interrupt requests are ignored.

Figure 10.13 shows a block diagram of the intelligent I/O interrupt. Figure 10.14 shows the IIOiR register. Figure 10.15 shows the IIOiE register.

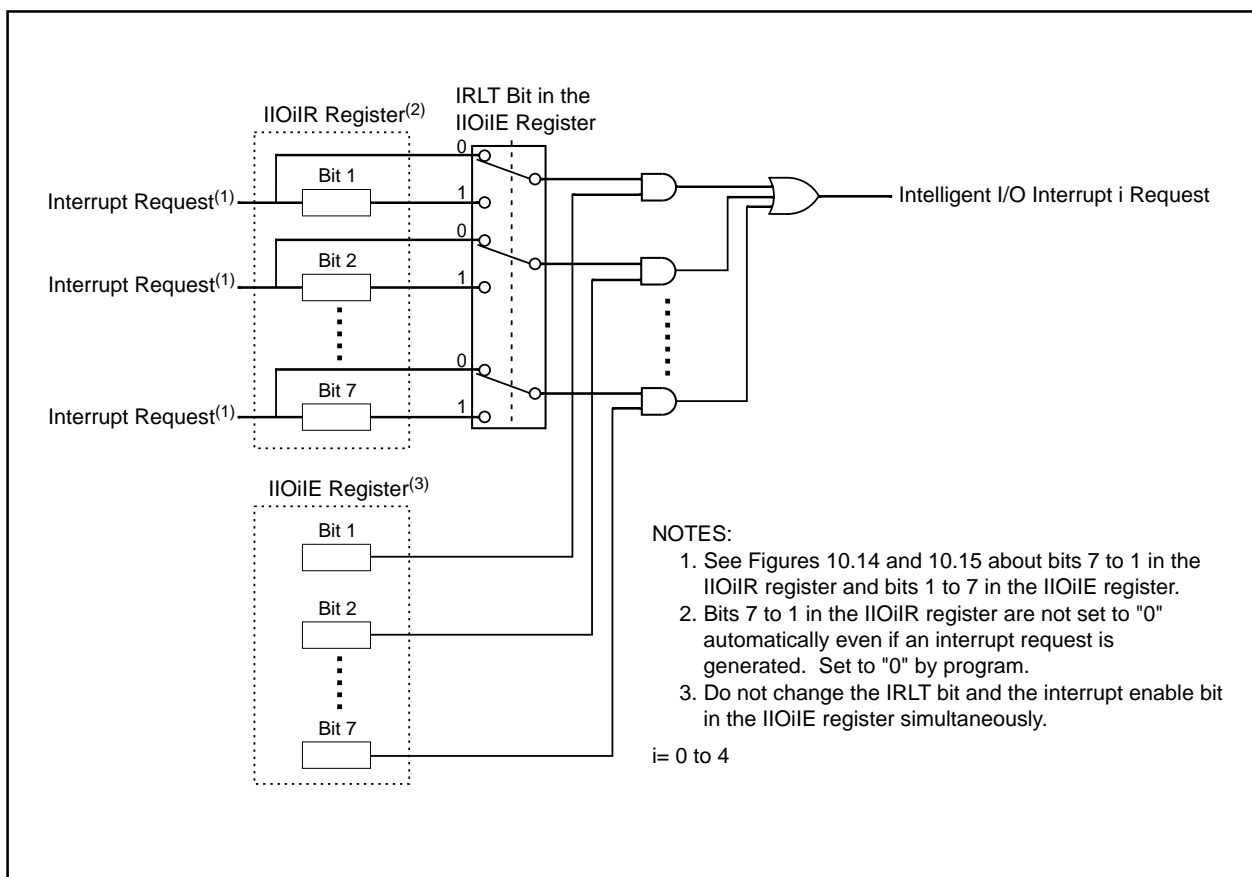
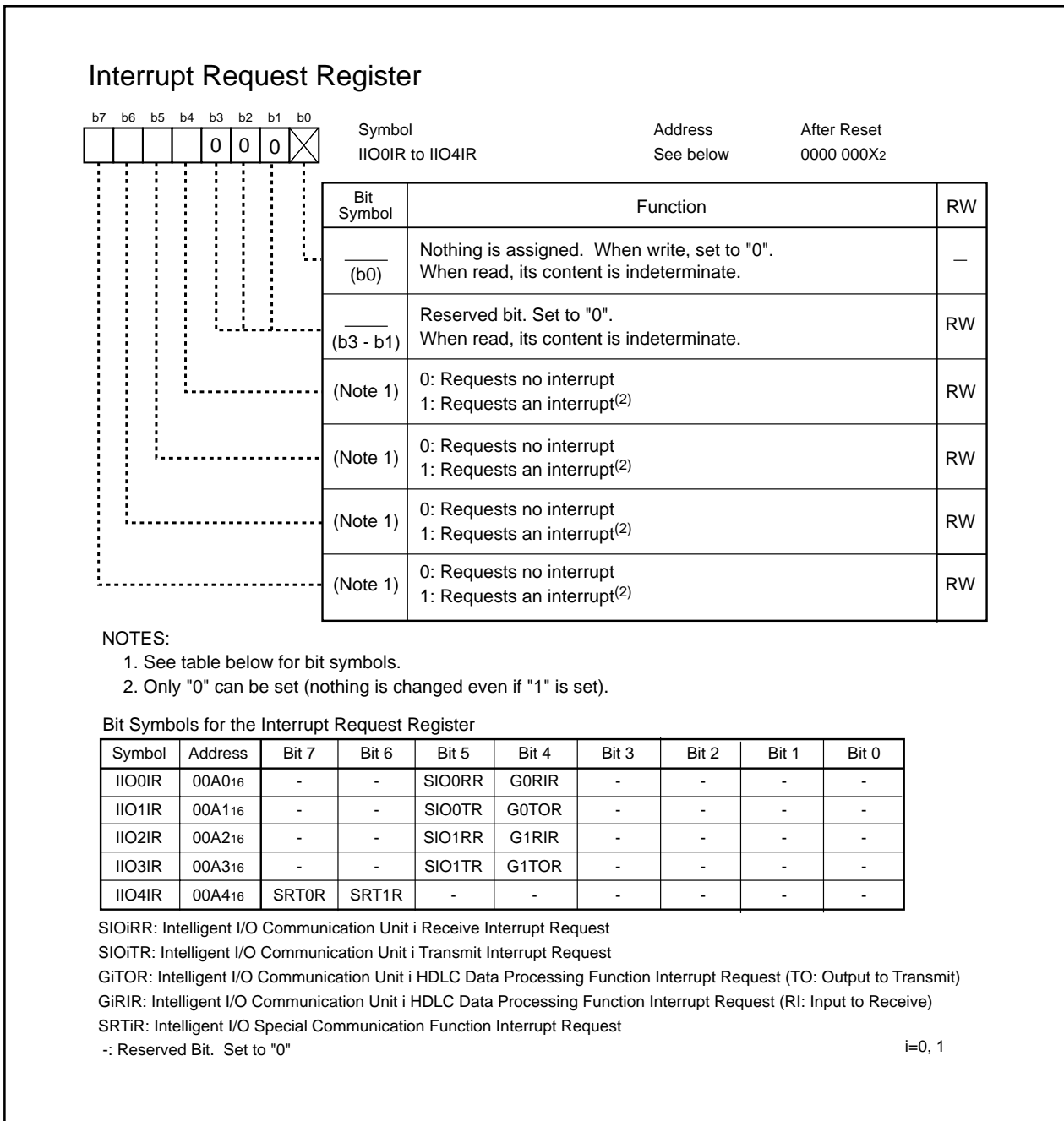
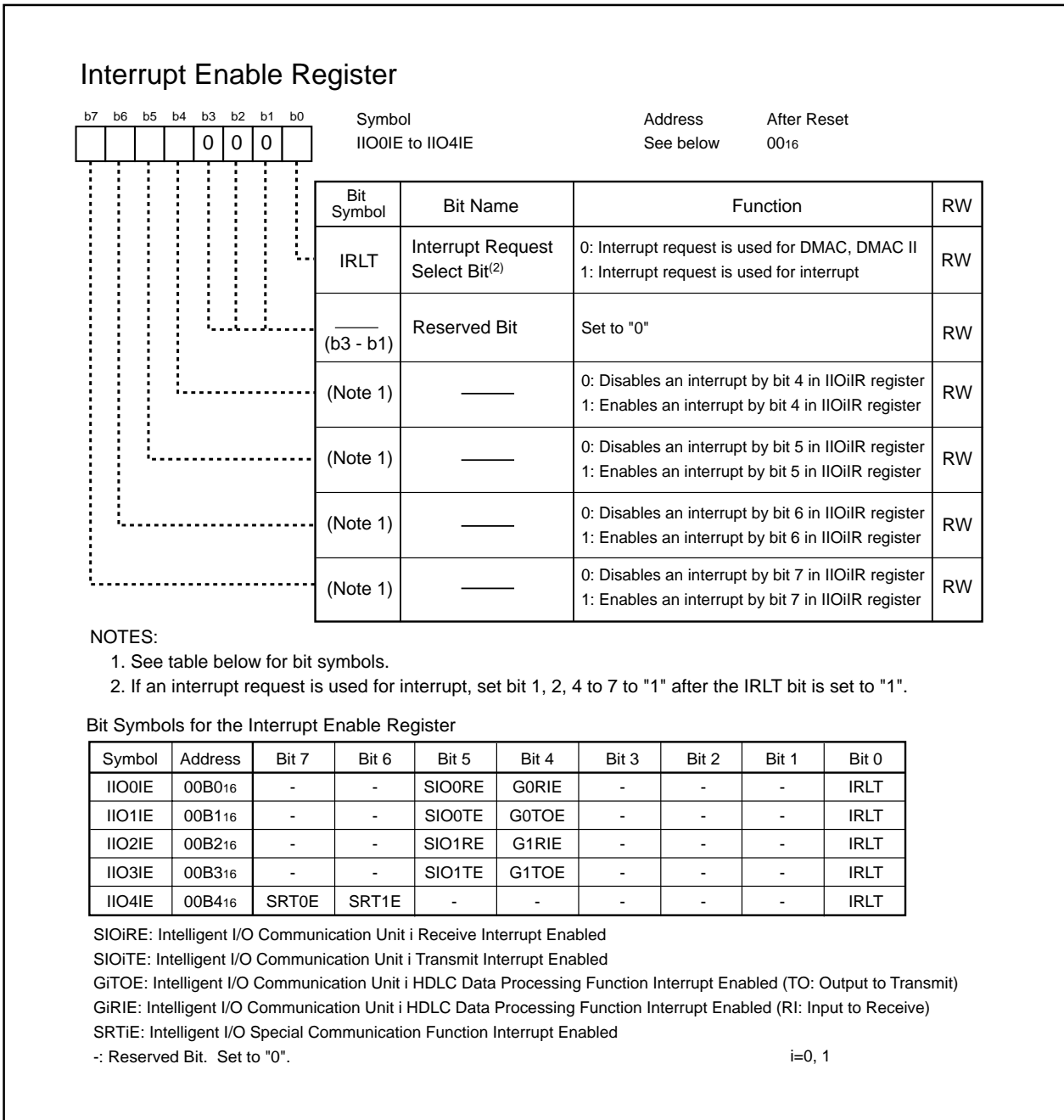


Figure 10.13 Intelligent I/O Interrupt



**Figure 10.14 IIO0iR to IIO4iR Registers**



**Figure 10.15 IIO0iE to IIO4iE Registers**

# 11. Watchdog Timer

The watchdog timer monitors the program executions and detects defective program. It allows the microcomputer to trigger a reset or to generate an interrupt if the program error occurs. The watchdog timer contains a 15-bit counter, which is decremented by the CPU clock that the prescaler divides. The CM06 bit in the CM0 register determines whether a watchdog timer interrupt request or reset is generated if the watchdog timer underflows. Once the CM06 bit is set to "1", it cannot be changed to "0" ( watchdog timer interrupt) by program. The CM06 bit is set to "0" only after reset.

When the main clock, on-chip oscillator clock, or PLL clock runs as the CPU clock, the WDC7 bit in the WDC register determine whether the prescaler divides the clock by 16 or by 128. When the sub clock runs as the CPU clock, the prescaler divides the clock by 2 regardless of the WDC7 bit setting. Watchdog timer cycle is calculated as follows. Marginal errors, due to the prescaler, may occur in watchdog timer cycle.

When the main clock, on-chip oscillator clock, or PLL clock is selected as the CPU clock,

$$\text{Watchdog timer cycle} = \frac{\text{Divide-by-16 or -128 prescaler} \times \text{counter value of watchdog timer (32768)}}{\text{CPU clock}}$$

When the sub clock is selected as the CPU clock,

$$\text{Watchdog timer cycle} = \frac{\text{Divide-by-2 prescaler} \times \text{counter value of watchdog timer (32768)}}{\text{CPU clock}}$$

For example, if the CPU clock frequency is 30MHz and the prescaler divides it by 16, the watchdog timer cycle is approximately 17.5 ms.

The watchdog timer is reset when the WDTS register is set and when a watchdog timer interrupt request is generated. The prescaler is reset only when the microcomputer is reset. Both watchdog timer and prescaler stop after reset. They begin counting when the WDTS register is set.

The watchdog timer and prescaler stop in stop mode, wait mode and hold state. They resume counting from the value held when the mode or state is exited.

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

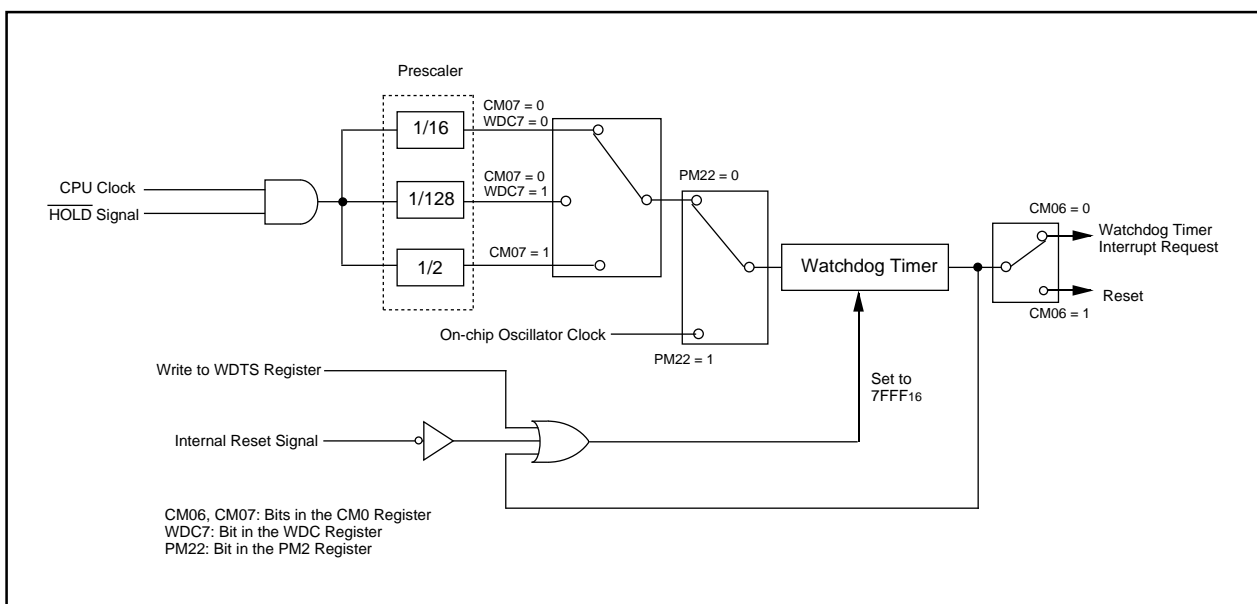


Figure 11.1 Watchdog Timer Block Diagram

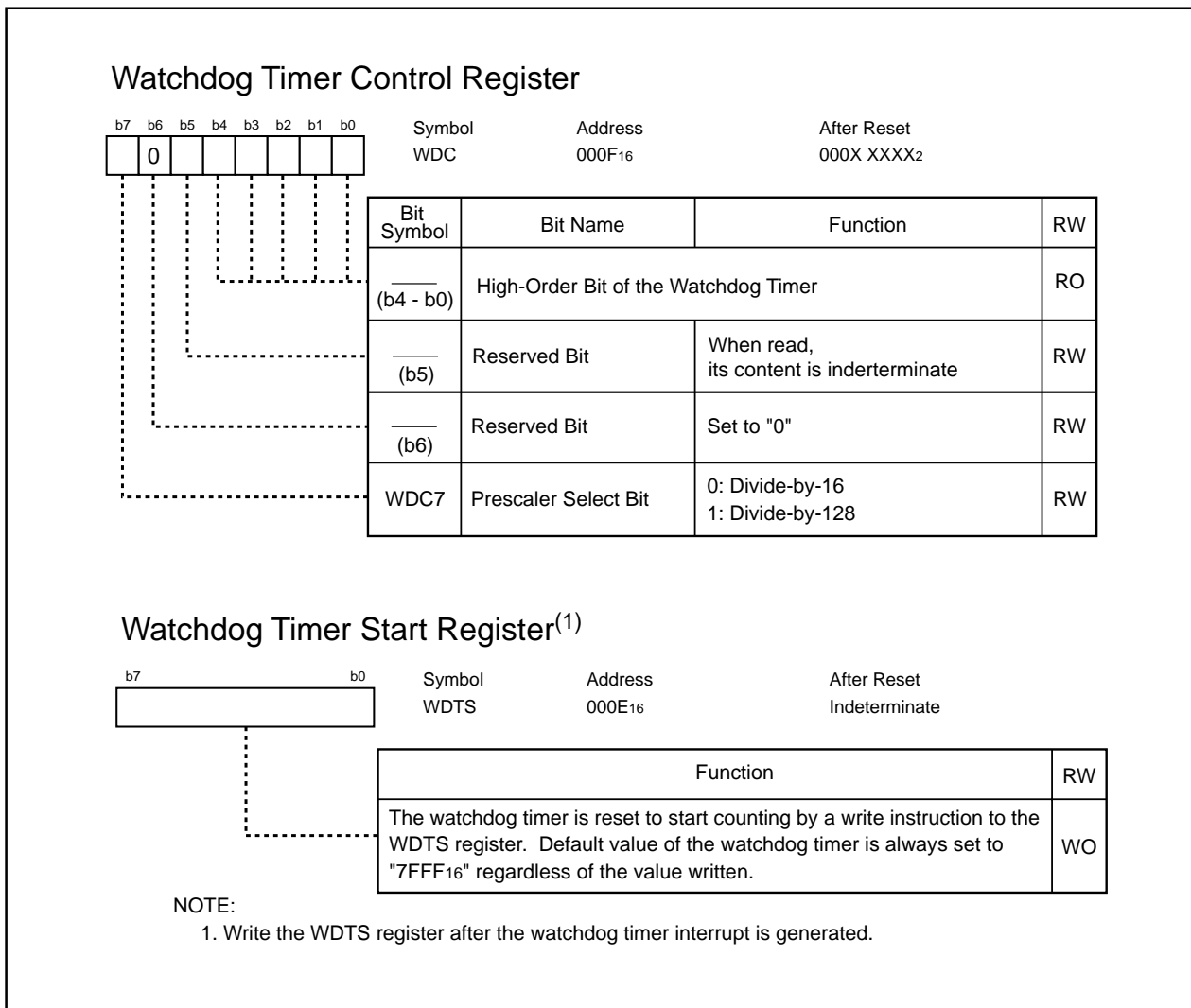


Figure 11.2 WDC Register and WDTS Register

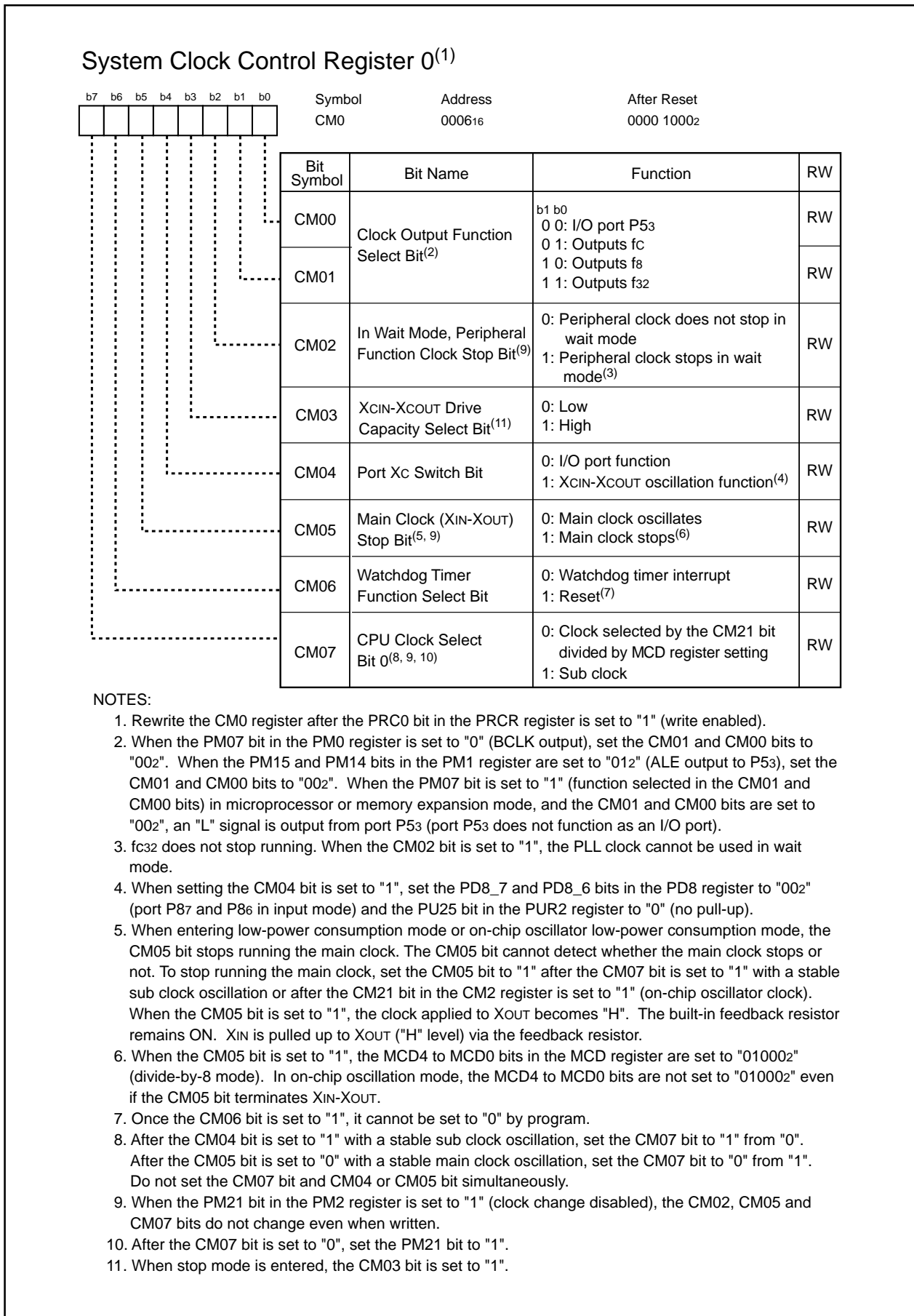


Figure 11.3 CM0 Register

## 11.1 Count Source Protection Mode

In count source protection mode, the on-chip oscillator clock is used as a count source for the watchdog timer. The count source protection mode allows the on-chip oscillator clock to run continuously, maintaining watchdog timer operation even if the program error occurs and the CPU clock stops running.

Follow the procedures below when using this mode.

- (1) Set the PRC0 bit in the PRCR register to "1" (write to CM0 register enabled)
- (2) Set the PRC1 bit in the PRCR register to "1" (write to PM2 register enabled)
- (3) Set the CM06 bit in the CM0 register to "1" (reset when the watchdog timer overflows)
- (4) Set the PM22 bit in the PM2 register to "1" (the on-chip oscillator clock as a count source of the watchdog timer)
- (5) Set the PRC0 bit to "0" (write to CM0 register disabled)
- (6) Set the PRC1 bit to "0" (write to PM2 register disabled)
- (7) Write to the WDTS register (the watchdog timer starts counting)

The followings will occur when the PM22 bit is set to "1".

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

$$\text{Watchdog timer cycle} = \frac{\text{Counter value of watchdog timer (32768)}}{\text{On-chip oscillator clock}}$$

- Write to the CM10 bit in the CM1 register is disabled. (The bit setting remains unchanged even if set it to "1". The microcomputer does not enter stop mode.)
- In wait mode or hold state, the watchdog timer continues running. However, the watchdog timer interrupt cannot be used to exit wait mode.

## 12. DMAC

This microcomputer contains four DMAC (direct memory access controller) channels that allow data to be sent to memory without using the CPU. DMAC transmits a 8- or 16-bit data from a source address to a destination address whenever a transmit request occurs. DMA0 and DMA1 must be prioritized if using DMAC. DMA2 and DMA3 share registers required for high-speed interrupts. High-speed interrupts cannot be used when using three or more DMAC channels.

The CPU and DMAC use the same data bus, but DMAC has a higher bus access privilege than the CPU. The cycle-steal method employed on DMAC enables high-speed operation between a transfer request and the complete transmission of 16-bit (word) or 8-bit (byte) data. Figure 12.1 shows a mapping of registers to be used for DMAC. Table 12.1 lists specifications of DMAC. Figures 12.2 to 12.5 show registers associated with DMAC.

Because the registers shown in Figure 12.1 are allocated in the CPU, use the LDC instruction to write to the registers. To set the DCT2, DCT3, DRC2, DRC3, DMA2 and DMA3 registers, set the B flag to "1" (register bank 1) and set the R0 to R3, A0, A1 registers with the MOV instruction.

To set the DSA2 and DSA3 registers, set the B flag to "1" and set the SB and FB registers with the LDC instruction. To set the DRA2 and DRA3 registers, set the SVP and VCT registers with the LDC instruction.

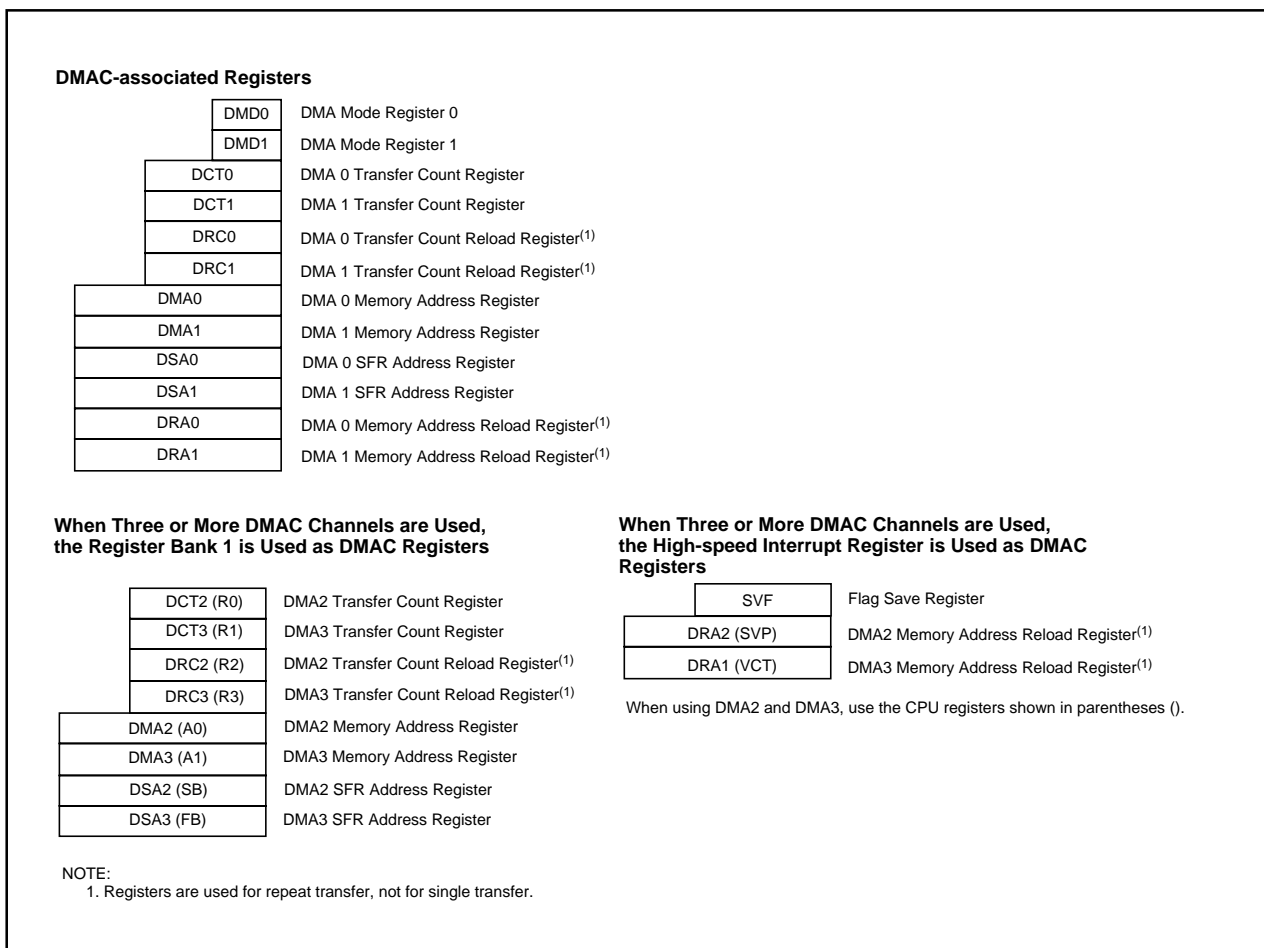


Figure 12.1 Register Mapping for DMAC



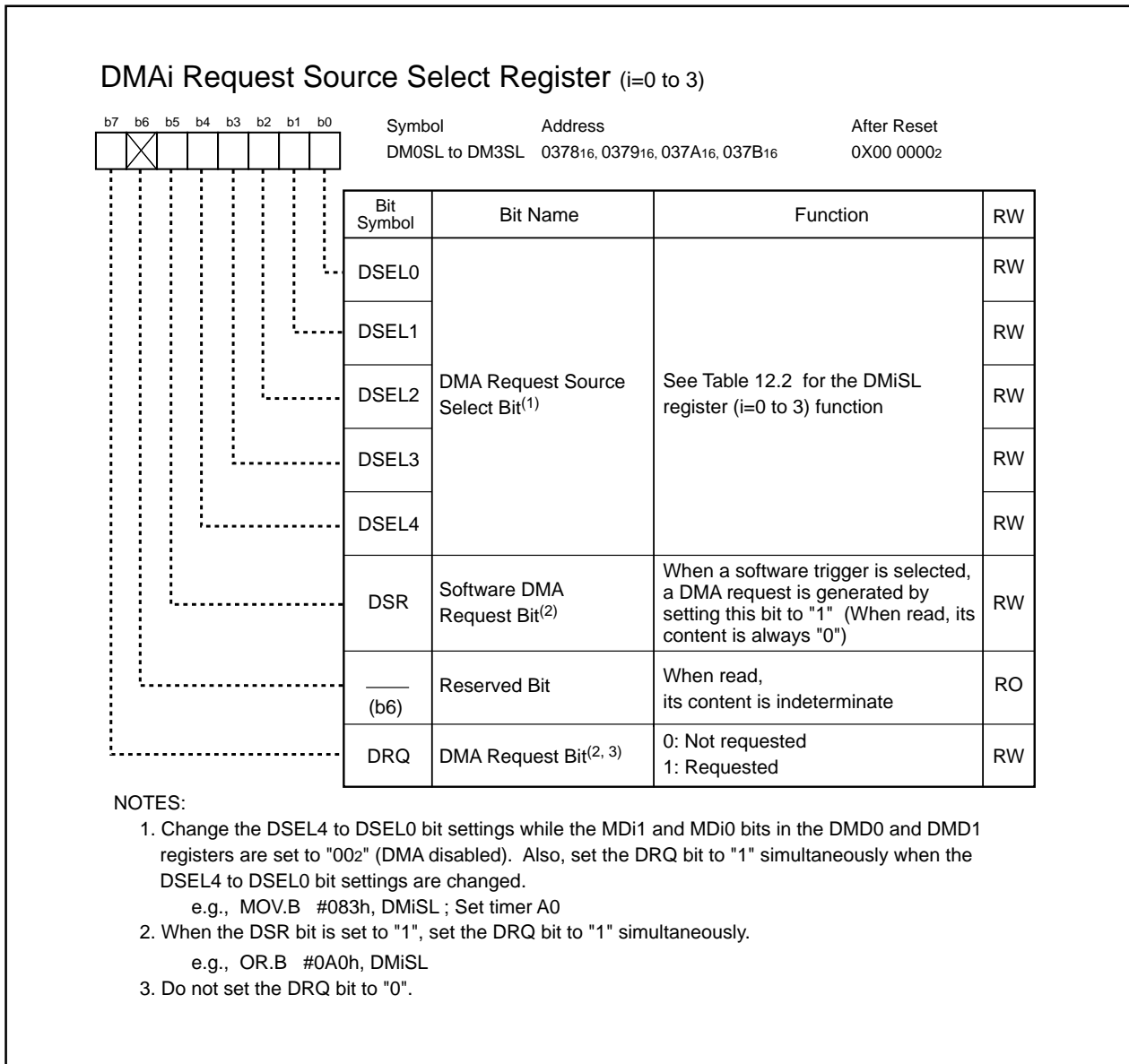
DMAC starts a data transfer by setting the DSR bit in the DMiSL register ( $i=0$  to 3) or by using an interrupt request, generated by the functions determined by the DSEL 4 to DSEL0 bits in the DMiSL register, as a DMA request. Unlike interrupt requests, the I flag and interrupt control register do not affect DMA. Therefore, a DMA request can be acknowledged even if an interrupt is disabled and cannot be acknowledged. In addition, the IR bit in the interrupt control register does not change when a DMA request is acknowledged.

**Table 12.1 DMAC Specifications**

Item	Specification	
Channels	4 channels (cycle-steal method)	
Transfer Memory Space	<ul style="list-style-type: none"> <li>From a desired address in a 16-Mbyte space to a fixed address in a 16-Mbyte space</li> <li>From a fixed address in a 16-Mbyte space to a desired address in a 16-Mbyte space</li> </ul>	
Maximum Bytes Transferred	128 Kbytes (when a 16-bit data is transferred) or 64 Kbytes (with an 8-bit data is transferred)	
DMA Request Source <sup>(1)</sup>	Falling edge or both edges of signals applied to the INT0 to INT3 pins Timers A0 to A4 interrupt requests Timers B0 to B5 interrupt requests UART0 to UART4 transmit and receive interrupt requests A/D0 conversion interrupt request Intelligent I/O interrupt request Software trigger	
Channel Priority	DMA0 > DMA1 > DMA2 > DMA3 (DMA0 has highest priority)	
Transfer Unit	8 bits, 16 bits	
Destination Address	Forward/fixed (forward and fixed directions cannot be specified when specifying source and destination addresses simultaneously)	
Transfer Mode	Single Transfer	Transfer is completed when the DCTi register ( $i = 0$ to 3) is set to "0000 <sub>16</sub> "
	Repeat Transfer	When the DCTi register is set to "0000 <sub>16</sub> ", the value of the DRCi register is reloaded into the DCTi register and the DMA transfer is continued
DMA Interrupt Request Generation Timing	When the DCTi register changes "0001 <sub>16</sub> " to "0000 <sub>16</sub> "	
DMA Startup	Single Transfer	DMA starts when a DMA request is generated after the DCTi register is set to "0001 <sub>16</sub> " or more and the MDi1 and MDi0 bits in the DMDj register ( $j = 0, 1$ ) are set to "012" (single transfer)
	Repeat Transfer	DMA starts when a DMA request is generated after the DCTi register is set to "0001 <sub>16</sub> " or more and the MDi1 and MDi0 bits are set to "112" (repeat transfer)
DMA Stop	Single Transfer	DMA stops when the MDi1 and MDi0 bits are set to "002" (DMA disabled) and the DCTi register is set to "0000 <sub>16</sub> " (0 DMA transfer) by DMA transfer or write
	Repeat Transfer	DMA stops when the MDi1 and MDi0 bits are set to "002" and the DCTi register is set to "0000 <sub>16</sub> " and the DRCi register set to "0000 <sub>16</sub> "
Reload Timing to the DCTi or DMAi Register	When the DCTi register is set to "0000 <sub>16</sub> " from "0001 <sub>16</sub> " in repeat transfer mode	
DMA Transfer Cycles	Minimum 3 cycles between SFRs and internal RAM	

**NOTE:**

1. The IR bit in the interrupt control register does not change when a DMA request is acknowledged.



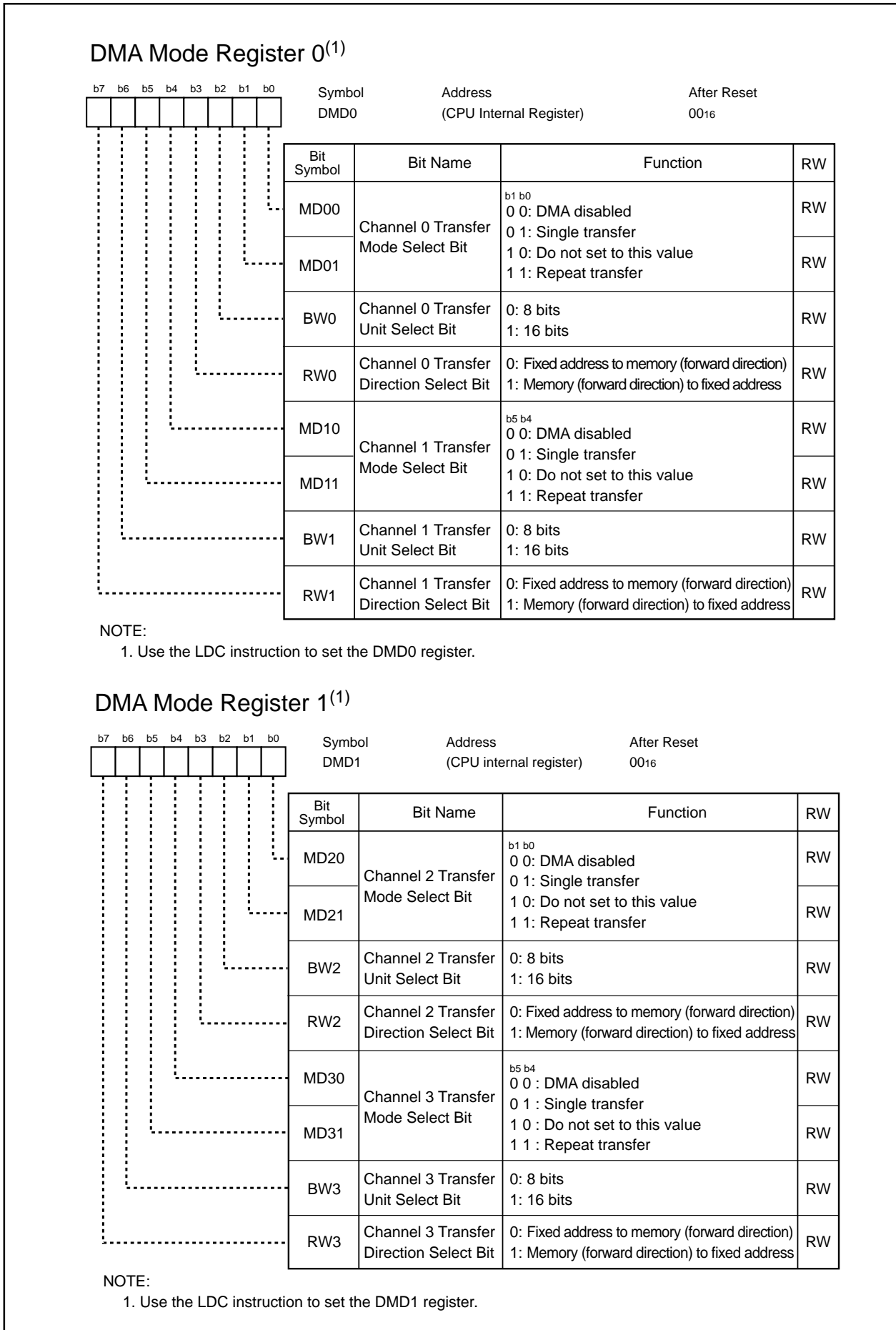
**Figure 12.2 DM0SL to DM3SL Registers**

**Table 12.2 DMiSL Register (i=0 to 3) Function**

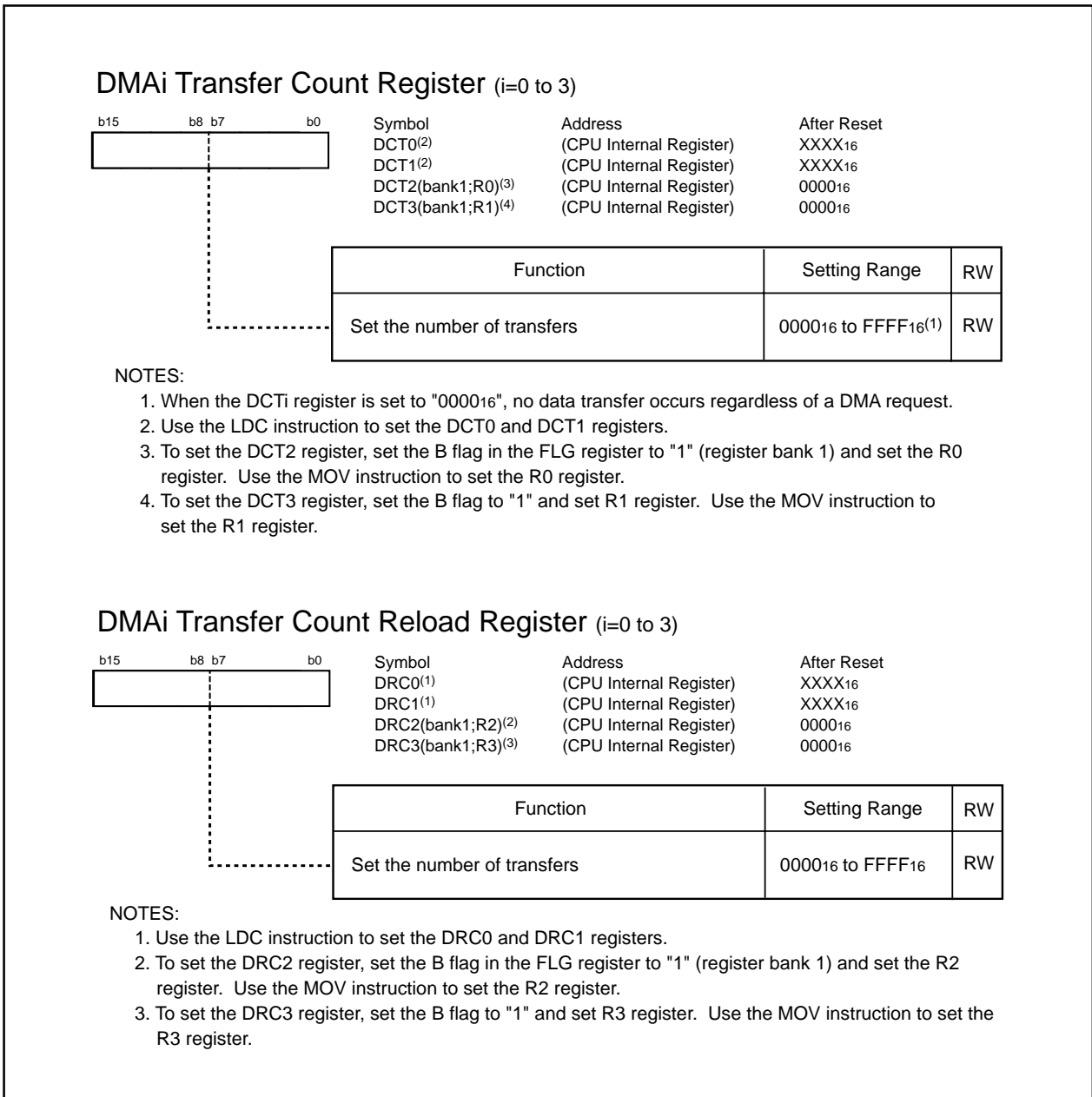
Setting Value b4 b3 b2 b1 b0	DMA Request Source			
	DMA0	DMA1	DMA2	DMA3
0 0 0 0 0	Software trigger			
0 0 0 0 1	Falling Edge of $\overline{INT0}$	Falling Edge of $\overline{INT1}$	Falling Edge of $\overline{INT2}$	Falling Edge of $\overline{INT3}^{(1)}$ (Note 2)
0 0 0 1 0	Both Edges of $\overline{INT0}$	Both Edges of $\overline{INT1}$	Both Edges of $\overline{INT2}$	Both Edges of $\overline{INT3}^{(1)}$ (Note 2)
0 0 0 1 1	Timer A0 Interrupt Request			
0 0 1 0 0	Timer A1 Interrupt Request			
0 0 1 0 1	Timer A2 Interrupt Request			
0 0 1 1 0	Timer A3 Interrupt Request			
0 0 1 1 1	Timer A4 Interrupt Request			
0 1 0 0 0	Timer B0 Interrupt Request			
0 1 0 0 1	Timer B1 Interrupt Request			
0 1 0 1 0	Timer B2 Interrupt Request			
0 1 0 1 1	Timer B3 Interrupt Request			
0 1 1 0 0	Timer B4 Interrupt Request			
0 1 1 0 1	Timer B5 Interrupt Request			
0 1 1 1 0	UART0 Transmit Interrupt Request			
0 1 1 1 1	UART0 Receive or ACK Interrupt Request <sup>(3)</sup>			
1 0 0 0 0	UART1 Transmit Interrupt Request			
1 0 0 0 1	UART1 Receive or ACK Interrupt Request <sup>(3)</sup>			
1 0 0 1 0	UART2 Transmit Interrupt Request			
1 0 0 1 1	UART2 Receive or ACK Interrupt Request <sup>(3)</sup>			
1 0 1 0 0	UART3 Transmit Interrupt Request			
1 0 1 0 1	UART3 Receive or ACK Interrupt Request <sup>(3)</sup>			
1 0 1 1 0	UART4 Transmit Interrupt Request			
1 0 1 1 1	UART4 Receive or ACK Interrupt Request <sup>(3)</sup>			
1 1 0 0 0	A/D0 Interrupt Request			
1 1 0 0 1	Intelligent I/O Interrupt 0 Request	_____	Intelligent I/O Interrupt 2 Request	_____
1 1 0 1 0	Intelligent I/O Interrupt 1 Request	_____	Intelligent I/O Interrupt 3 Request	_____
1 1 0 1 1	Intelligent I/O Interrupt 2 Request	_____	Intelligent I/O Interrupt 4 Request	_____
1 1 1 0 0	Intelligent I/O Interrupt 3 Request	_____	_____	Intelligent I/O Interrupt 0 Request
1 1 1 0 1	Intelligent I/O Interrupt 4 Request	_____	_____	Intelligent I/O Interrupt 1 Request
1 1 1 1 0	_____	Intelligent I/O Interrupt 0 Request	_____	Intelligent I/O Interrupt 2 Request
1 1 1 1 1	_____	Intelligent I/O Interrupt 1 Request	_____	Intelligent I/O Interrupt 3 Request

**NOTES:**

1. If the  $\overline{INT3}$  pin is used for data bus in memory expansion mode or microprocessor mode, a DMA3 interrupt request cannot be generated by a signal applied to the  $\overline{INT3}$  pin.
2. The falling edge and both edges of signals applied to the  $\overline{INTj}$  pin (j=0 to 3) cause a DMA request generation. The  $\overline{INT}$  interrupt (the POL bit in the INTjIC register, the LVS bit, the IFSR register) is not affected and vice versa.
3. Use the UkSMR register and UkSMR2 register (k=0 to 4) to switch between the UARTk receive and ACK interrupt as a DMA request source.  
To use the ACK interrupt for a DMA request, set the IICM bit in the UkSMR register to "1" and the IICM2 bit in the UkSMR2 register to "0".



**Figure 12.3 DMD0 and DMD1 Registers**



**Figure 12.4 DCT0 to DCT3 Registers and DRC0 to DRC3 Registers**

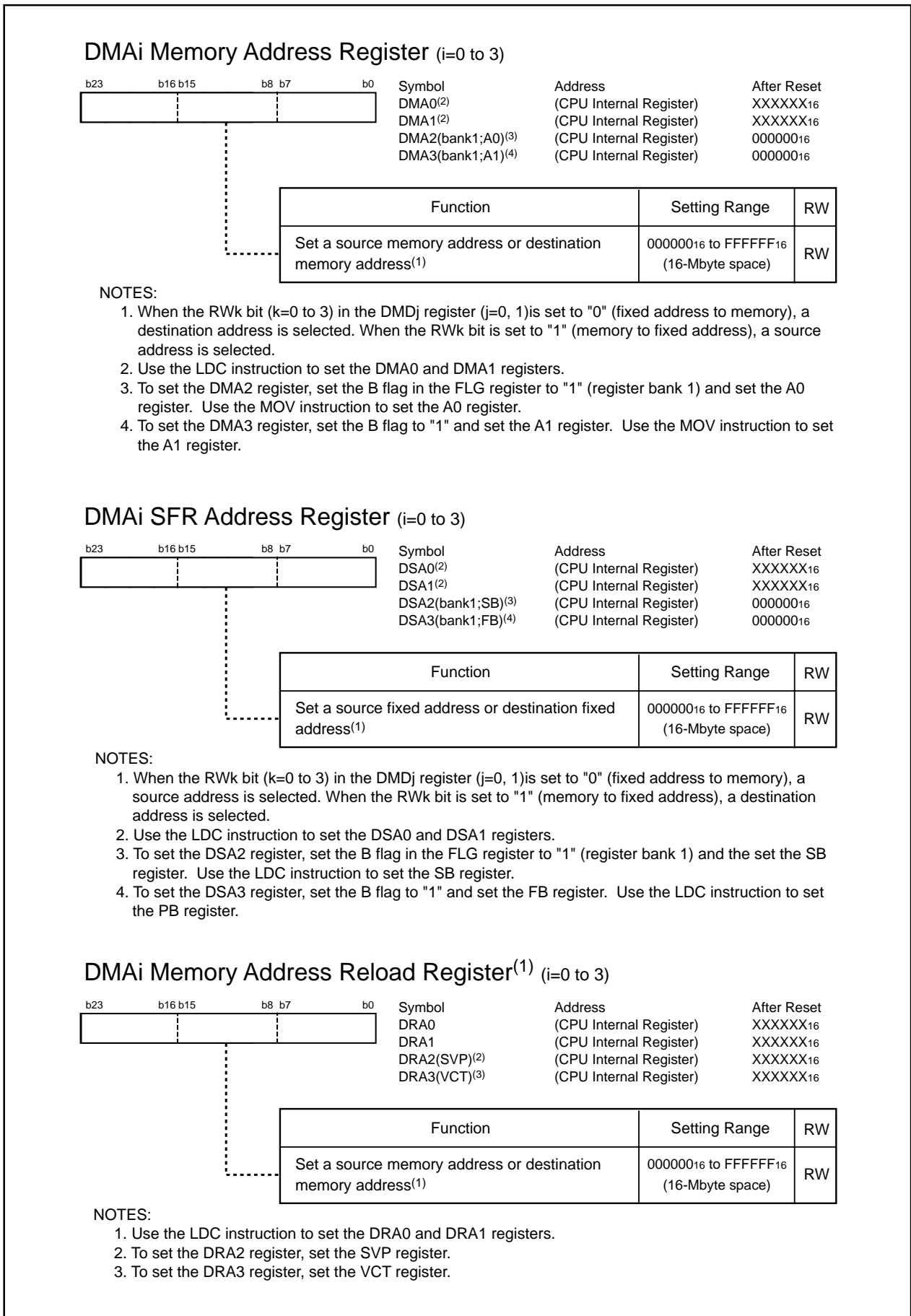


Figure 12.5 DMA0 to DMA3 Registers, DSA0 to DSA3 Registers and DRA0 to DRA3 Registers

## 12.1 Transfer Cycle

Transfer cycle contains a bus cycle to read data from a memory or the SFR area (source read) and a bus cycle to write data to a memory space or the SFR area (destination write). The number of read and write bus cycles depends on source and destination addresses. In memory expansion mode and microprocessor mode, the number of read and write bus cycles also depends on DS register setting. Software wait state insertion and the  $\overline{\text{RDY}}$  signal make a bus cycle longer.

### 12.1.1 Effect of Source and Destination Addresses

When a 16-bit data is transferred with a 16-bit data bus and a source address starting with an odd address, source read cycle is incremented by one bus cycle, compared to a source address starting with an even address.

When a 16-bit data is transferred with a 16-bit data bus and a destination address starting with an odd address, a destination write cycle is incremented by one bus cycle, compared to a destination address starting with an even address.

### 12.1.2 Effect of the DS Register

In an external space in memory expansion or microprocessor mode, transfer cycle varies depending on the data bus used at the source and destination addresses. See **Figure 8.1** for details about the DS register.

- When an 8-bit data bus (the DS<sub>i</sub> bit in the DS register is set to "0" (i=0 to 3)), accessing both source address and destination address, is used to transfer a 16-bit data, 8-bit data is transferred twice. Therefore, two bus cycles are required to read the data and another two bus cycles to write the data.
- When an 8-bit data bus (the DS<sub>i</sub> bit in the DS register is set to "0" (i=0 to 3)), accessing source address, and a 16-bit data bus, accessing destination address, are used to transfer a 16-bit data, 8-bit data is read twice but is written once as 16-bit data. Therefore, two bus cycles are required for reading and one bus cycle is for writing.
- When a 16-bit data bus, accessing source address, and an 8-bit data bus, accessing destination address, are used to transfer a 16-bit data, 16-bit data is read once and 8-bit data is written twice. Therefore, one bus cycle is required for reading and two bus cycles is for writing.

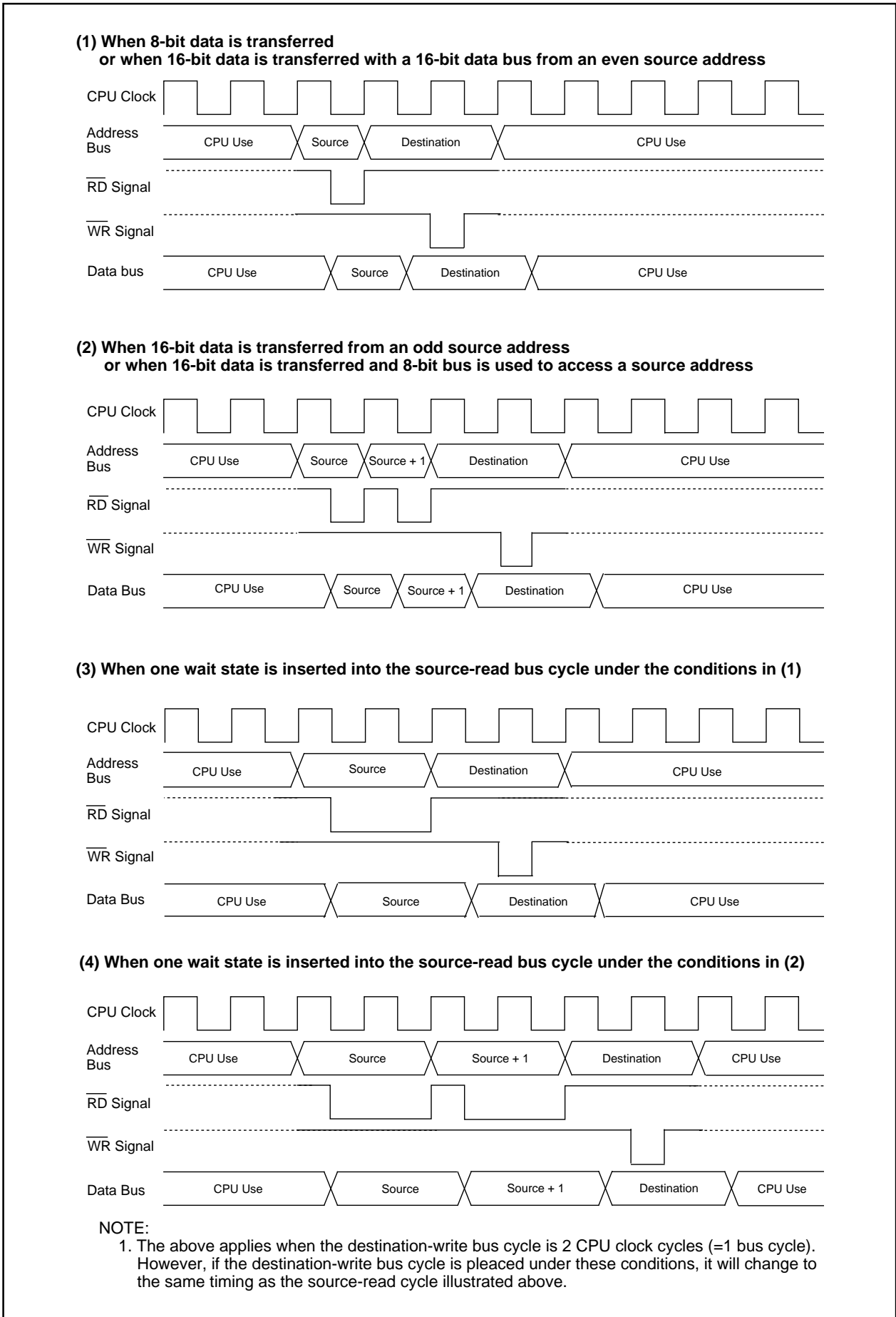
### 12.1.3 Effect of Software Wait State

When the SFR area or memory space with software wait states is accessed, the number of CPU clock cycles is incremented by software wait states.

Figure 12.6 shows an example of a transfer cycle for the source-read bus cycle. In Figure 12.6, the number of source-read bus cycles is illustrated under different conditions, provided that the destination address is an address of an external space with the destination-write cycle as two CPU clock cycles (=one bus cycle). In effect, the destination-write bus cycle is also affected by each condition and the transfer cycles change accordingly. To calculate a transfer cycle, apply respective conditions to both destination-write bus cycle and source-read bus cycle. As shown in example (2) of Figure 12.6, when an 8-bit data bus, accessing both source and destination addresses, is used to transfer a 16-bit data, two bus cycles each are required for the source-read bus cycle and destination-write bus cycle.

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

In memory expansion or microprocessor mode, the  $\overline{\text{RDY}}$  signal affects a bus cycle if a source address or destination address is allocated address in an external space. Refer to **7.2.6 RDY Signal** for details.



**Figure 12.6 Transfer Cycle Examples with the Source-Read Bus Cycle**



## 12.2 DMAC Transfer Cycle

The number of DMAC transfer cycle can be calculated as follows.

Any combination of even or odd transfer read and write addresses are possible. Table 12.3 lists the number of DMAC transfer cycles. Table 12.4 lists coefficient j, k.

$$\text{Transfer cycles per transfer} = \text{Number of read cycle} \times j + \text{Number of write cycle} \times k$$

**Table 12.3 DMAC Transfer Cycles**

Transfer Unit	Bus Width	Access Address	Single-Chip Mode		Memory Expansion Mode Microprocessor Mode	
			Read Cycle	Write Cycle	Read Cycle	Write Cycle
8-bit transfers (BWi bit in the DMDp register = 0)	16-bit	Even	1	1	1	1
		Odd	1	1	1	1
	8-bit	Even	—	—	1	1
		Odd	—	—	1	1
16-bit transfers (BWi bit = 1)	16-bit	Even	1	1	1	1
		Odd	2	2	2	2
	8-bit	Even	—	—	2	2
		Odd	—	—	2	2

i = 0 to 3, p = 0, 1

**Table 12.4 Coefficient j, k**

Internal Space			External Space
Internal RAM with no wait state	Internal RAM with a wait state	SFR area	j and k BCLK cycles shown in Table 7.5. Add one cycle to j or k cycles when inserting a recovery cycle.
j=1	j=2	j=2	
k=1	k=2	k=2	

j, k = 2 to 9

## 12.3 Channel Priority and DMA Transfer Timing

When multiple DMA requests are generated in the same sampling period, between the falling edge of the CPU clock and the next falling edge, the DRQ bit in the DMiSL register (i=0 to 3) is set to "1" (requested) simultaneously. Channel priority in this case is: DMA0 > DMA1 > DMA2 > DMA3.

Figure 12.7 shows an example of the DMA transfer by external source.

In Figure 12.7, the DMA0 request having highest priority is received first to start a transfer when a DMA0 request and DMA1 request are generated simultaneously. After one DMA0 transfer is completed, the bus privilege is returned to the CPU. When the CPU has completed one bus access, the DMA1 transfer starts. After one DMA1 transfer is completed, the privilege is again returned to the CPU.

In addition, DMA requests cannot be counted up since each channel has one DRQ bit. Therefore, when DMA requests, as DMA1 in Figure 12.7, occur more than once before receiving bus privilege, the DRQ bit is set to "0" as soon as privilege is acquired. The bus privilege is returned to the CPU when one transfer is completed.

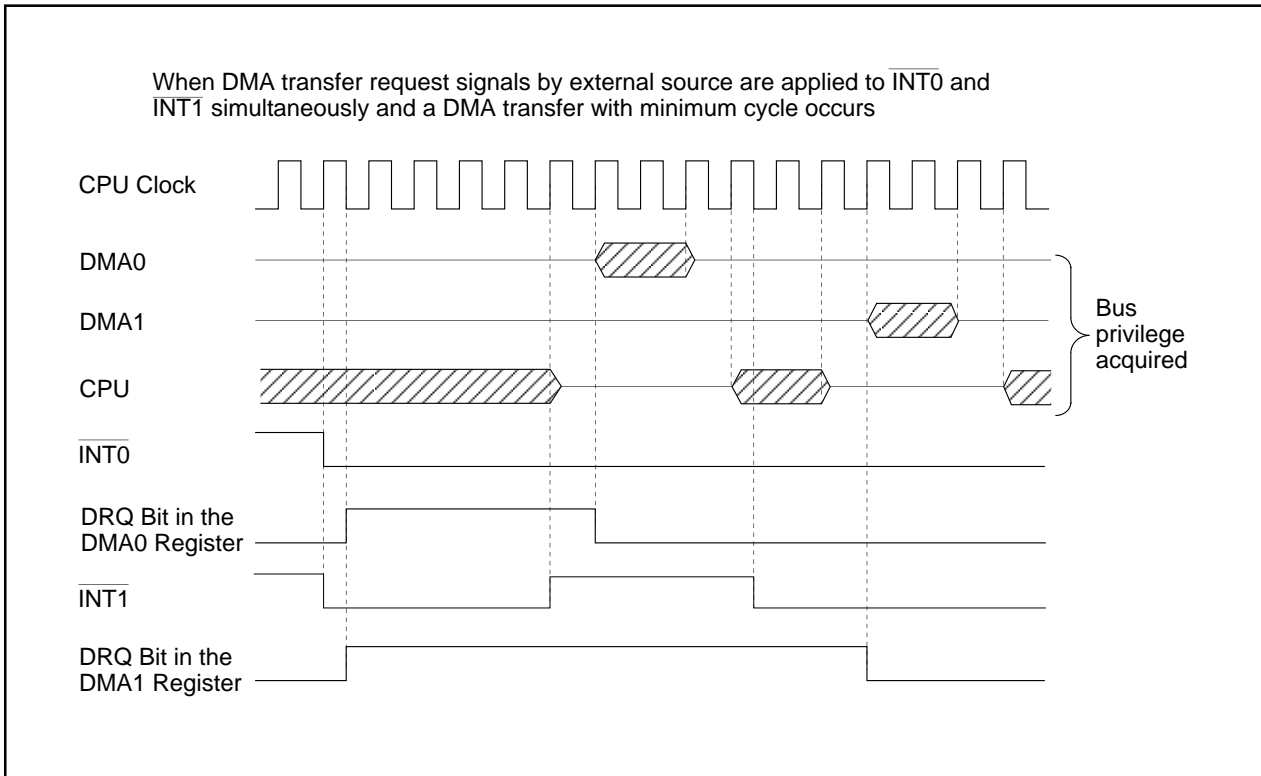


Figure 12.7 DMA Transfer by External Source

## 13. DMAC II

DMAC II performs memory-to-memory transfer, immediate data transfer and calculation transfer, which transfers the sum of two data added by an interrupt request from any peripheral functions.

Table 13.1 lists specifications of DMAC II.

**Table 13.1 DMAC II Specifications**

Item	Specification
DMAC II Request Source	Interrupt requests generated by all peripheral functions when the ILVL2 to ILVL0 bits are set to "1112"
Transfer Data	<ul style="list-style-type: none"> <li>• Data in memory is transferred to memory (memory-to-memory transfer)</li> <li>• Immediate data is transferred to memory (immediate data transfer)</li> <li>• Data in memory (or immediate data) + data in memory are transferred to memory (calculation transfer)</li> </ul>
Transfer Block	8 bits or 16 bits
Transfer Space	64-Kbyte space in addresses 00000 <sub>16</sub> to 0FFFF <sub>16</sub> <sup>(1, 2)</sup>
Transfer Direction	Fixed or forward address Selected separately for each source address and destination address
Transfer Mode	Single transfer, burst transfer
Chained Transfer Function	Parameters (transfer count, transfer address and other information) are switched when transfer counter reaches zero
End-of-Transfer Interrupt	Interrupt occurs when a transfer counter reaches zero
Multiple Transfer Function	Multiple data can be transferred by a generated request for one DMAC II transfer

NOTES:

1. When transferring a 16-bit data to destination address 0FFFF<sub>16</sub>, it is transferred to 0FFFF<sub>16</sub> and 10000<sub>16</sub>. The same transfer occurs when the source address is 0FFFF<sub>16</sub>.
2. The actual space where transfer can occur is limited due to internal RAM capacity.

### 13.1 DMAC II Settings

DMAC II can be made available by setting up the following registers and tables.

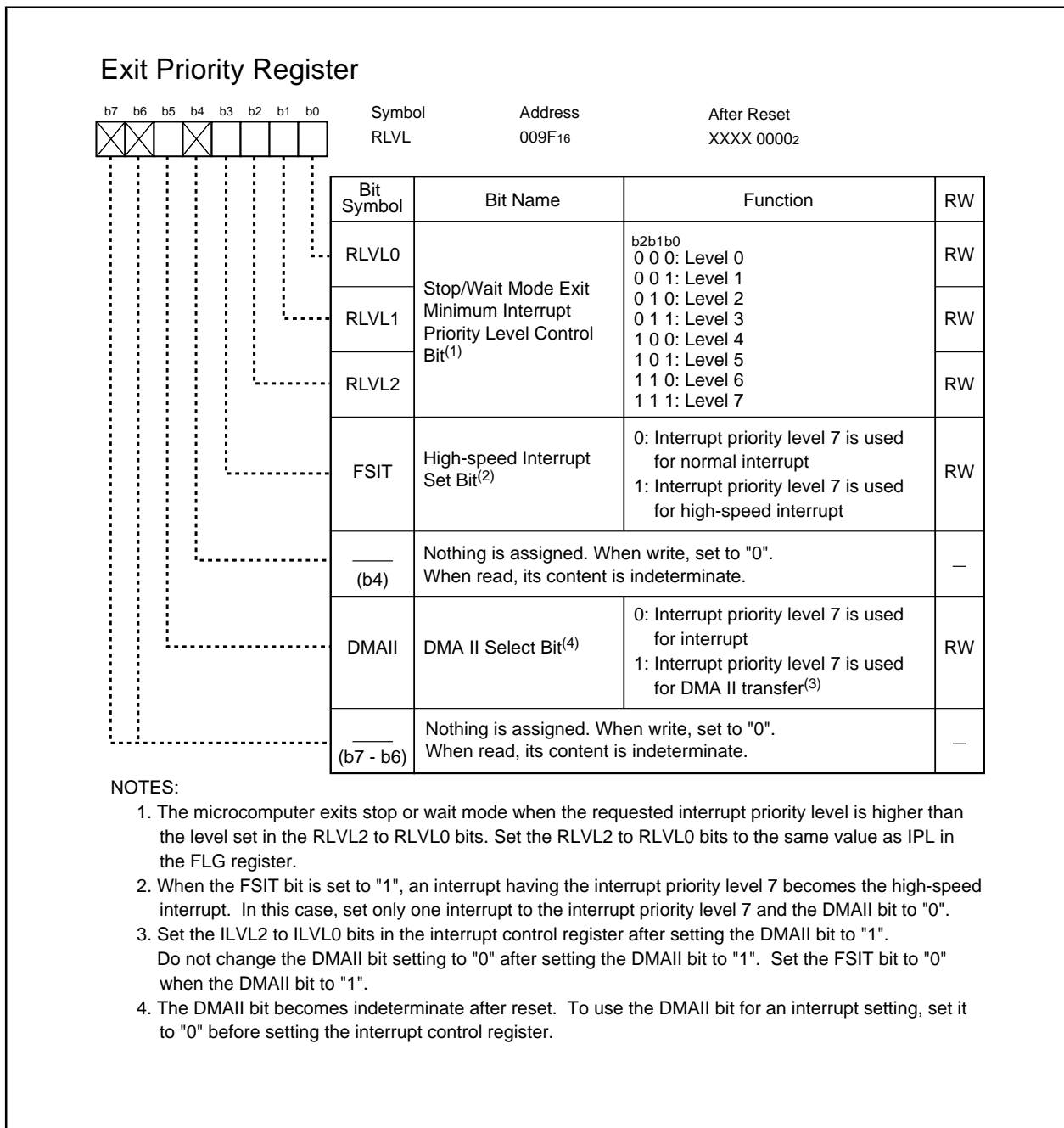
- RLVL register
- DMAC II Index
- Interrupt control register of the peripheral function causing a DMAC II request
- The relocatable vector table of the peripheral function causing a DMAC II request
- IRLT bit in the IIOiE register (i=0 to 4) to use the intelligent I/O

Refer to **10. Interrupts** for details on the IIOiE register.

#### 13.1.1 RLVL Register

When the DMAII bit is set to "1" (DMAC II transfer) and the FSIT bit to "0" (normal interrupt), DMAC II is activated by an interrupt request from any peripheral function with the ILVL2 to ILVL0 bits in the interrupt control register set to "1112" (level 7).

Figure 13.1 shows the RLVL register.



**Figure 13.1 RLVL Register**

### 13.1.2 DMAC II Index

The DMAC II index is a data table which comprises 8 to 18 bytes (maximum 32 bytes when the multiple transfer function is selected). The DMAC II index stores parameters for transfer mode, transfer counter, source address (or immediate data), operation address as an address to be calculated, destination address, chained transfer address, and end-of-transfer interrupt address.

This DMAC II index must be located on the RAM area.

Figure 13.2 shows a configuration of the DMAC II index. Table 13.2 lists a configuration of the DMAC II index in transfer mode.

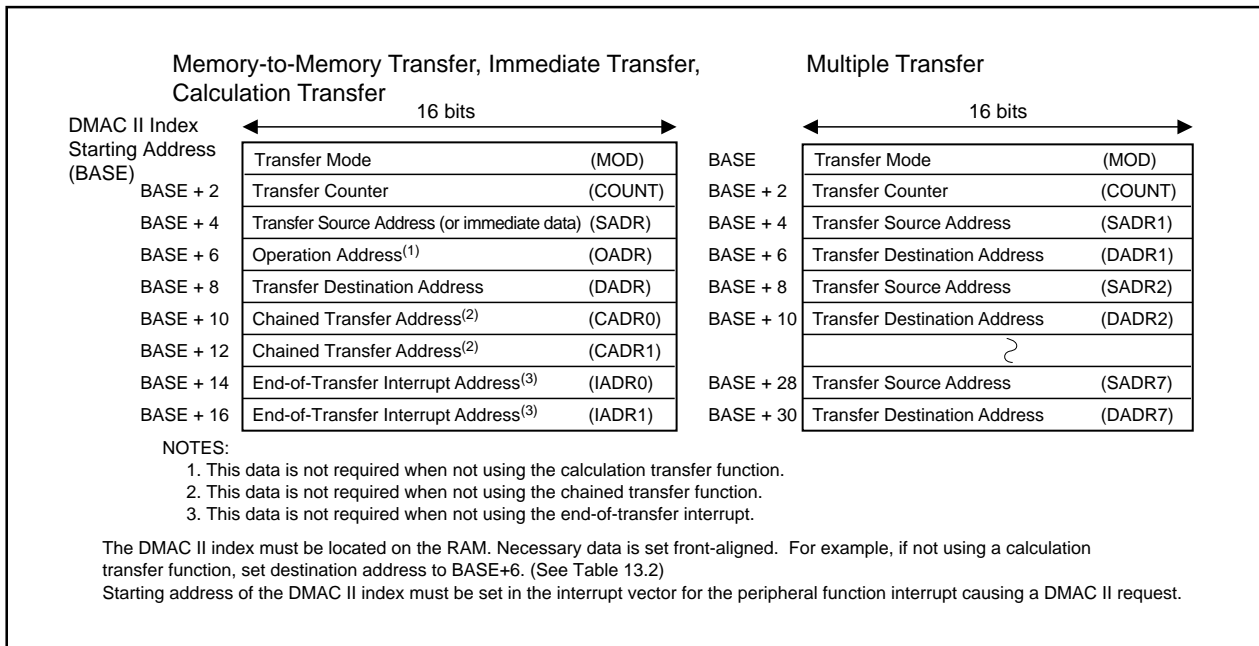


Figure 13.2 DMAC II Index

The followings are details of the DMAC II index. Set these parameters in the specified order listed in Table 13.2, according to DMAC II transfer mode.

- **Transfer mode (MOD)**

Two-byte data is required to set transfer mode. Figure 13.3 shows a configuration for transfer mode.

- **Transfer counter (COUNT)**

Two-byte data is required to set the number of transfer.

- **Transfer source address (SADR)**

Two-byte data is required to set the source memory address or immediate data.

- **Operation address (OADR)**

Two-byte data is required to set a memory address to be calculated. Set this data only when using the calculation transfer function.

- **Transfer destination address (DADR)**

Two-byte data is required to set the destination memory address.

- **Chained transfer address (CADR)**

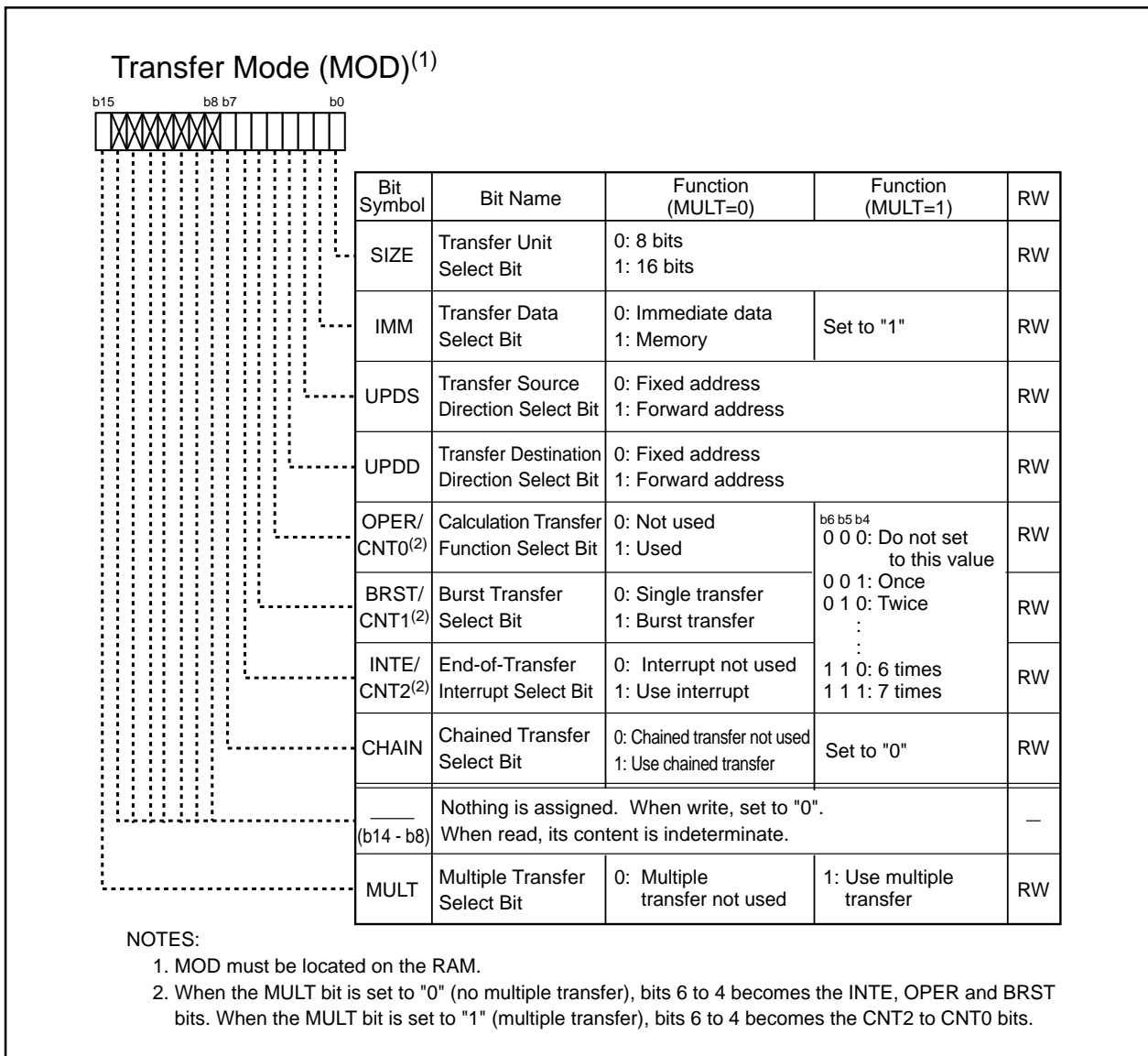
Four-byte data is required to set the starting address of the DMAC II index for the next transfer. Set this data only when using the chained transfer function.

- **End-of-transfer interrupt address (IADR)**

Four-byte data is required to set a jump address for end-of-transfer interrupt processing. Set this data only when using the end-of-transfer interrupt.

**Table 13.2 DMAC II Index Configuration in Transfer Mode**

Transfer Data	Memory-to-Memory Transfer /Immediate Data Transfer				Calculation Transfer				Multiple Transfer																																																		
	Not Used	Used	Not Used	Used	Not Used	Used	Not Used	Used																																																			
Chained Transfer	Not Used	Used	Not Used	Used	Not Used	Used	Not Used	Used	Not Available																																																		
End-of-Transfer Interrupt	Not Used	Not Used	Used	Used	Not Used	Not Used	Used	Used	Not Available																																																		
DMAC II Index	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>DADR</td></tr> </table> <p>8 bytes</p>	MOD	COUNT	SADR	DADR	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>DADR</td></tr> <tr><td>CADR0</td></tr> <tr><td>CADR1</td></tr> </table> <p>12 bytes</p>	MOD	COUNT	SADR	DADR	CADR0	CADR1	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>DADR</td></tr> <tr><td>IADR0</td></tr> <tr><td>IADR1</td></tr> </table> <p>12 bytes</p>	MOD	COUNT	SADR	DADR	IADR0	IADR1	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>DADR</td></tr> <tr><td>CADR0</td></tr> <tr><td>CADR1</td></tr> <tr><td>IADR0</td></tr> <tr><td>IADR1</td></tr> </table> <p>16 bytes</p>	MOD	COUNT	SADR	DADR	CADR0	CADR1	IADR0	IADR1	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>OADR</td></tr> <tr><td>DADR</td></tr> </table> <p>10 bytes</p>	MOD	COUNT	SADR	OADR	DADR	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>OADR</td></tr> <tr><td>DADR</td></tr> <tr><td>CADR0</td></tr> <tr><td>CADR1</td></tr> </table> <p>14 bytes</p>	MOD	COUNT	SADR	OADR	DADR	CADR0	CADR1	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>OADR</td></tr> <tr><td>DADR</td></tr> <tr><td>IADR0</td></tr> <tr><td>IADR1</td></tr> </table> <p>14 bytes</p>	MOD	COUNT	SADR	OADR	DADR	IADR0	IADR1	<table border="1"> <tr><td>MOD</td></tr> <tr><td>COUNT</td></tr> <tr><td>SADR</td></tr> <tr><td>DADR</td></tr> <tr><td>IADR0</td></tr> <tr><td>IADR1</td></tr> <tr><td>SADRi</td></tr> <tr><td>DADRi</td></tr> </table> <p>i=1 to 7 max. 32 bytes (when i=7)</p>	MOD	COUNT	SADR	DADR	IADR0	IADR1	SADRi	DADRi
	MOD																																																										
COUNT																																																											
SADR																																																											
DADR																																																											
MOD																																																											
COUNT																																																											
SADR																																																											
DADR																																																											
CADR0																																																											
CADR1																																																											
MOD																																																											
COUNT																																																											
SADR																																																											
DADR																																																											
IADR0																																																											
IADR1																																																											
MOD																																																											
COUNT																																																											
SADR																																																											
DADR																																																											
CADR0																																																											
CADR1																																																											
IADR0																																																											
IADR1																																																											
MOD																																																											
COUNT																																																											
SADR																																																											
OADR																																																											
DADR																																																											
MOD																																																											
COUNT																																																											
SADR																																																											
OADR																																																											
DADR																																																											
CADR0																																																											
CADR1																																																											
MOD																																																											
COUNT																																																											
SADR																																																											
OADR																																																											
DADR																																																											
IADR0																																																											
IADR1																																																											
MOD																																																											
COUNT																																																											
SADR																																																											
DADR																																																											
IADR0																																																											
IADR1																																																											
SADRi																																																											
DADRi																																																											



**Figure 13.3 MOD**

### 13.1.3 Interrupt Control Register for the Peripheral Function

For the peripheral function interrupt activating DMAC II, set the ILVL2 to ILVL0 bits to "1112" (level 7).

### 13.1.4 Relocatable Vector Table for the Peripheral Function

Set the starting address of the DMAC II index in the interrupt vector for the peripheral function interrupt activating DMAC II.

When using the chained transfer, the relocatable vector table must be located in the RAM.

### 13.1.5 IRLT Bit in the IIOiE Register (i=0 to 4)

When the intelligent I/O interrupt or CAN interrupt is used to activate DMAC II, set the IRLT bit in the IIOiE register of the interrupt to "0".

## 13.2 DMAC II Performance

Function to activate DMAC II is selected by setting the DMA II bit to "1" (DMAC II transfer). DMAC II is activated by all peripheral function interrupts with the ILVL2 to ILVL0 bits set to "1112" (level 7). These peripheral function interrupt request signals become DMAC II transfer request signals and the peripheral function interrupt cannot be used.

When an interrupt request is generated by setting the ILVL2 to ILVL0 bits to "1112" (level 7), DMAC II is activated regardless of what state the I flag and IPL are in.

## 13.3 Transfer Data

DMAC II transfers 8-bit or 16-bit data.

- Memory-to-memory transfer : Data is transferred from a desired memory location in a 64-Kbyte space (Addresses  $00000_{16}$  to  $0FFFF_{16}$ ) to another desired memory location in the same space.
- Immediate data transfer : Immediate data is transferred to a desired memory location in a 64-Kbyte space.
- Calculation transfer : Two 8-bit or 16-bit data are added together and the result is transferred to a desired memory location in a 64-Kbyte space.

When a 16-bit data is transferred to the destination address  $0FFFF_{16}$ , it is transferred to  $0FFFF_{16}$  and  $10000_{16}$ . The same transfer occurs when the source address is  $0FFFF_{16}$ . Actual transferable space varies depending on the internal RAM capacity.

### 13.3.1 Memory-to-memory Transfer

Data transfer between any two memory locations can be:

- a transfer from a fixed address to another fixed address
- a transfer from a fixed address to a relocatable address
- a transfer from a relocatable address to a fixed address
- a transfer from a relocatable address to another relocatable address

When a relocatable address is selected, the address is incremented, after a transfer, for the next transfer. In a 8-bit transfer, the transfer address is incremented by one. In a 16-bit transfer, the transfer address is incremented by two.

When a source or destination address exceeds address  $0FFFF_{16}$  as a result of address incrementation, the source or destination address returns to address  $00000_{16}$  and continues incrementation. Maintain source and destination address at address  $0FFFF_{16}$  or below.

### 13.3.2 Immediate Data Transfer

DMAC II transfers immediate data to any memory location. A fixed or relocatable address can be selected as the destination address. Store the immediate data into SADR. To transfer an 8-bit immediate data, write the data in the low-order byte of SADR (high-order byte is ignored).

### 13.3.3 Calculation Transfer

After two memory data or an immediate data and memory data are added together, DMAC II transfers calculated result to any memory location. SADR must have one memory location address to be calculated or immediate data and OADR must have the other memory location address to be calculated. Fixed or relocatable address can be selected as source and destination addresses when using a memory + memory calculation transfer. If the transfer source address is relocatable, the operation address also becomes relocatable. Fixed or relocatable address can be selected as the transfer destination address when using an immediate data + memory calculation transfer.

## 13.4 Transfer Modes

Single and burst transfers are available. The BRST bit in MOD selects transfer method, either single transfer or burst transfer. COUNT determines how many transfers occur. No transfer occurs when COUNT is set to "0000<sub>16</sub>".

### 13.4.1 Single Transfer

For every transfer request source, DMAC II transfers one transfer unit of 8-bit or 16-bit data once. When the source or destination address is relocatable, the address is incremented, after a transfer, for the next transfer.

COUNT is decremented every time a transfer occurs. When using the end-of-transfer interrupt, the interrupt is acknowledged when COUNT reaches "0".

### 13.4.2 Burst Transfer

For every transfer request source, DMAC II continuously transfers data the number of times determined by COUNT. COUNT is decremented every time a transfer occurs. The burst transfer ends when COUNT reaches "0". The end-of-transfer interrupt is acknowledged when the burst transfer ends if using the end-of-transfer interrupt. All interrupts are ignored while the burst transfer is in progress.

## 13.5 Multiple Transfer

The MULT bit in MOD selects the multiple transfer. When using the multiple transfer, select the memory-to-memory transfer. One transfer request source initiates multiple transfers. The CNT2 to CNT0 bits in MOD selects the number of transfers from "001<sub>2</sub>" (once) to "111<sub>2</sub>" (7 times). Do not set the CNT2 to CNT0 bits to "000<sub>2</sub>".

The transfer source and destination addresses for each transfer must be allocated alternately in addresses following MOD and COUNT. When the multiple transfer is selected, the calculation transfer, burst transfer, end-of-transfer interrupt and chained transfer cannot be used.



### 13.6 Chained Transfer

The CHAIN bit in MOD selects the chained transfer.

The following process initiates the chained transfer.

- (1) Transfer, caused by a transfer request source, occurs according to the content of the DMAC II index. The vectors of the request source indicates where the DMAC II index is allocated. For each request, the BRST bit selects either single or burst transfer.
- (2) When COUNT reaches "0", the contents of CADR1 and CADR0 are written to the vector of the request source. When the INTE bit in MOD is set to "1", the end-of-transfer interrupt is generated simultaneously.
- (3) When the next DMAC II transfer request is generated, transfer occurs according to the contents of the DMAC II index indicated by the peripheral function interrupt vector rewritten in (2).

Figure 13.4 shows the relocatable vector and DMACII index when the chained transfer is in progress. For the chained transfer, the relocatable vector table must be located in the RAM.

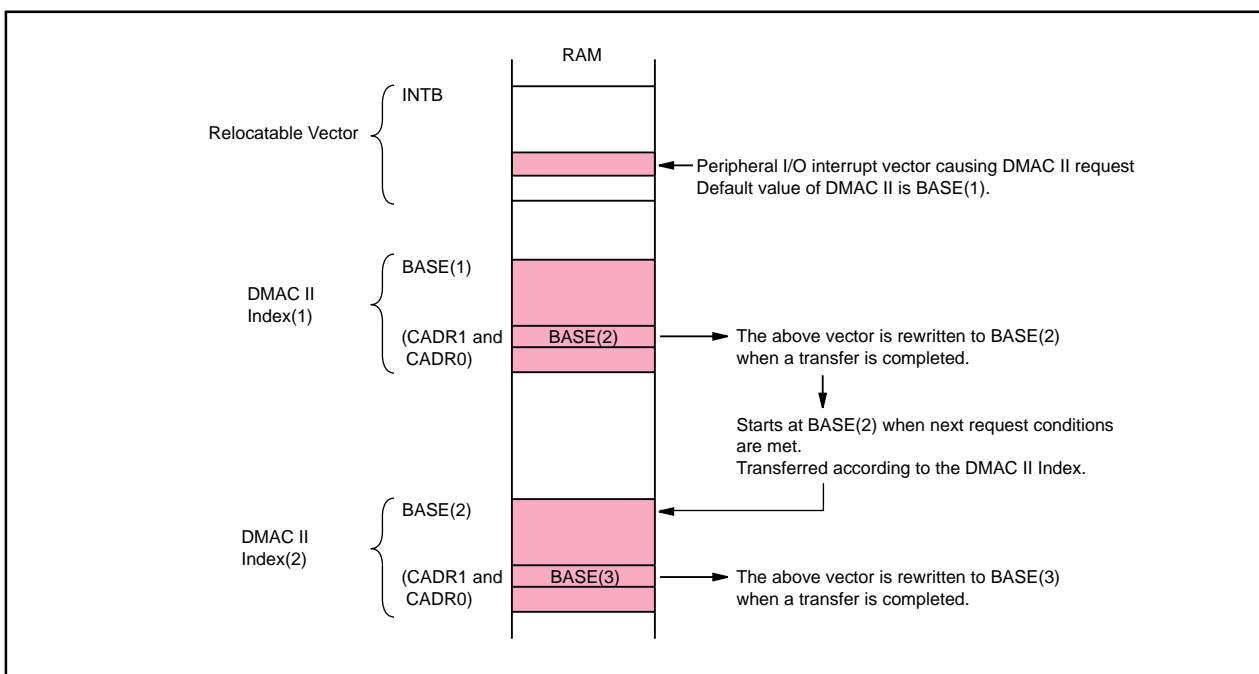


Figure 13.4 Relocatable Vector and DMAC II Index

### 13.7 End-of-Transfer Interrupt

The INTE bit in MOD selects the end-of-transfer interrupt. Set the starting address of the end-of-transfer interrupt routine in IADR1 and IADR0. The end-of-transfer interrupt is generated when COUNT reaches "0."

### 13.8 Execution Time

DMAC II execution cycle is calculated by the following equations:

Multiple transfers:  $t = 21 + (11 + b + c) \times k$  cycles

Other than multiple transfers:  $t = 6 + (26 + a + b + c + d) \times m + (4 + e) \times n$  cycles

a: If IMM = 0 (source of transfer is immediate data), a = 0;

if IMM = 1 (source of transfer is memory), a = -1

b: If UPDS = 1 (source transfer address is a relocatable address), b = 0;

if UPDS = 0 (source transfer address is a fixed address), b = 1

c: If UPDD = 1 (destination transfer address is a relocatable address), c = 0;

if UPDD = 0 (destination transfer address is a fixed address), c = 1

d: If OPER = 0 (calculation function is not selected), d = 0;

if OPER = 1 (calculation function is selected) and UPDS = 0 (source of transfer is immediate data or fixed address memory), d = 7;

if OPER = 1 (calculation function is selected) and UPDS = 1 (source of transfer is relocatable address memory), d = 8

e: If CHAIN = 0 (chained transfer is not selected), e = 0; if CHAIN = 1 (chained transfer is selected), e = 4

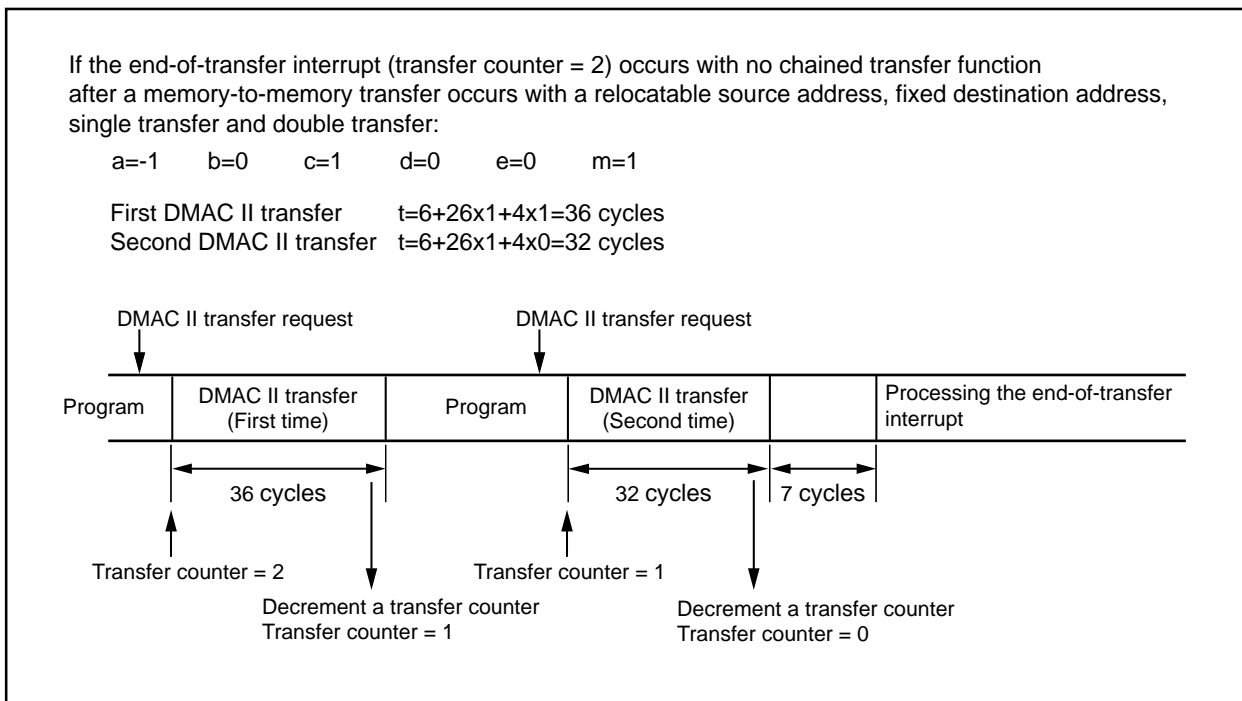
m: BRST = 0 (single transfer), m = 1; BRST = 1 (burst transfer), m = the value set in transfer counter

n: If COUNT = 1, n = 0; if COUNT = 2 or more, n = 1

k: Number of transfers set in the CNT2 to CNT0 bits

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

The first instruction from the end-of-transfer interrupt routine is executed in the eighth cycle after the DMAC II transfer is completed.



**Figure 13.5 Transfer Cycle**

When an interrupt request as a DMAC II transfer request source and another interrupt request with higher priority (e.g.,  $\overline{\text{NMI}}$  or watchdog timer) are generated simultaneously, the interrupt with higher priority takes precedence over the DMAC II transfer. The pending DMAC II transfer starts after the interrupt sequence has been completed.

# 14. Timer

The microcomputer has eleven 16-bit timers. Five timers A and six timers B have different functions. Each timer functions independently. The count source for each timer becomes the clock for timer operations including counting and reloading, etc. Figures 14.1 and 14.2 show block diagrams of timer A and timer B configuration.

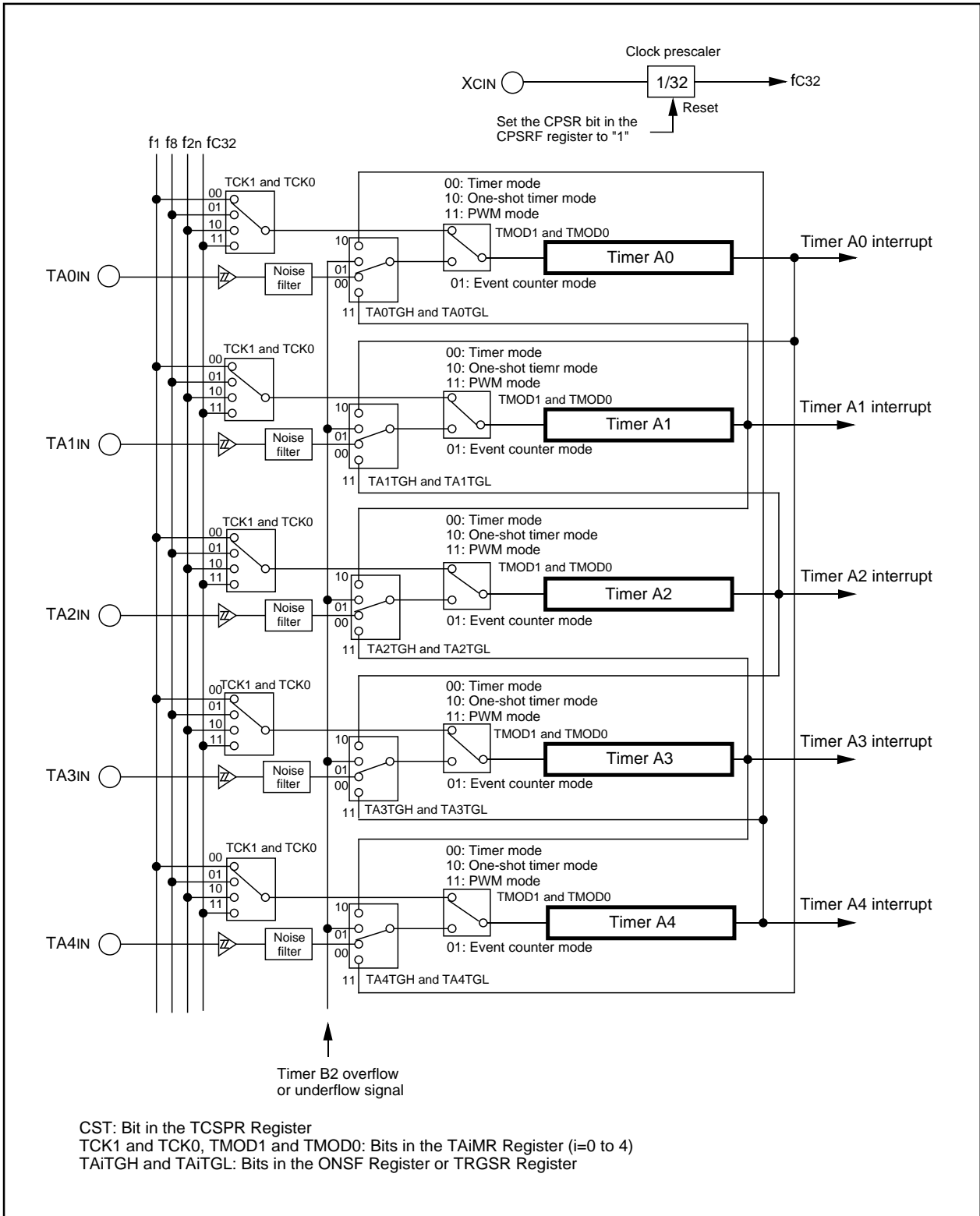


Figure 14.1 Timer A Configuration

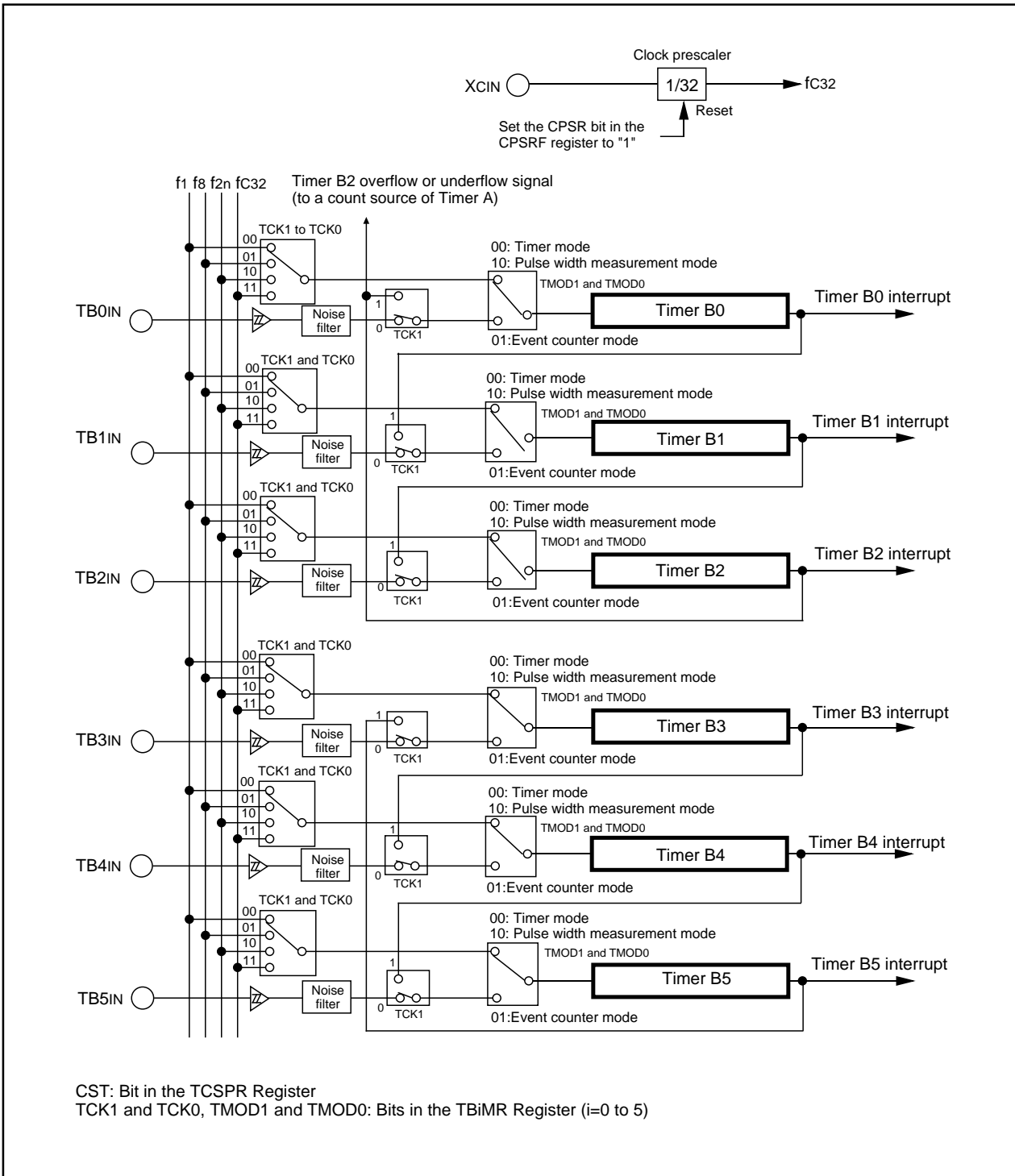


Figure 14.2 Timer B Configuration

## 14.1 Timer A

Figure 14.3 shows a block diagram of the timer A. Figures 14.4 to 14.7 show registers associated with the timer A.

The timer A supports the following four modes. Except in event counter mode, all timers A0 to A4 have the same function. The TMOD1 and TMOD0 bits in the TAI<sub>MR</sub> register (i=0 to 4) determine which mode is used.

- Timer mode: The timer counts an internal count source.
- Event counter mode: The timer counts an external pulse or an overflow and underflow of other timers.
- One-shot timer mode: The timer outputs one valid pulse until a counter value reaches "000016".
- Pulse width modulation mode: The timer continuously outputs desired pulse widths.

Table 14.1 lists TAI<sub>OUT</sub> pin settings when used as an output. Table 14.2 lists TAI<sub>IN</sub> and TAI<sub>OUT</sub> pin settings when used as an input.

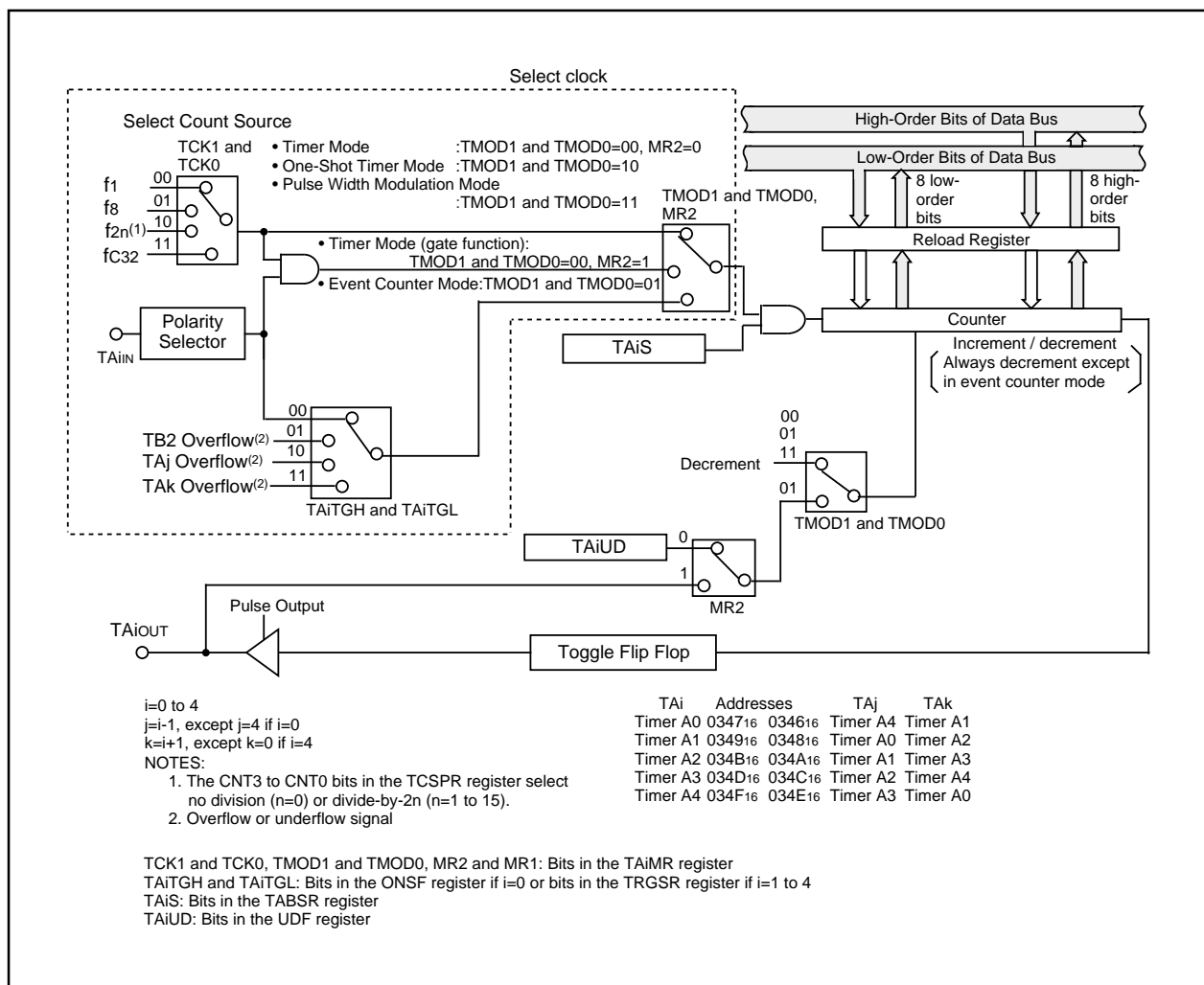
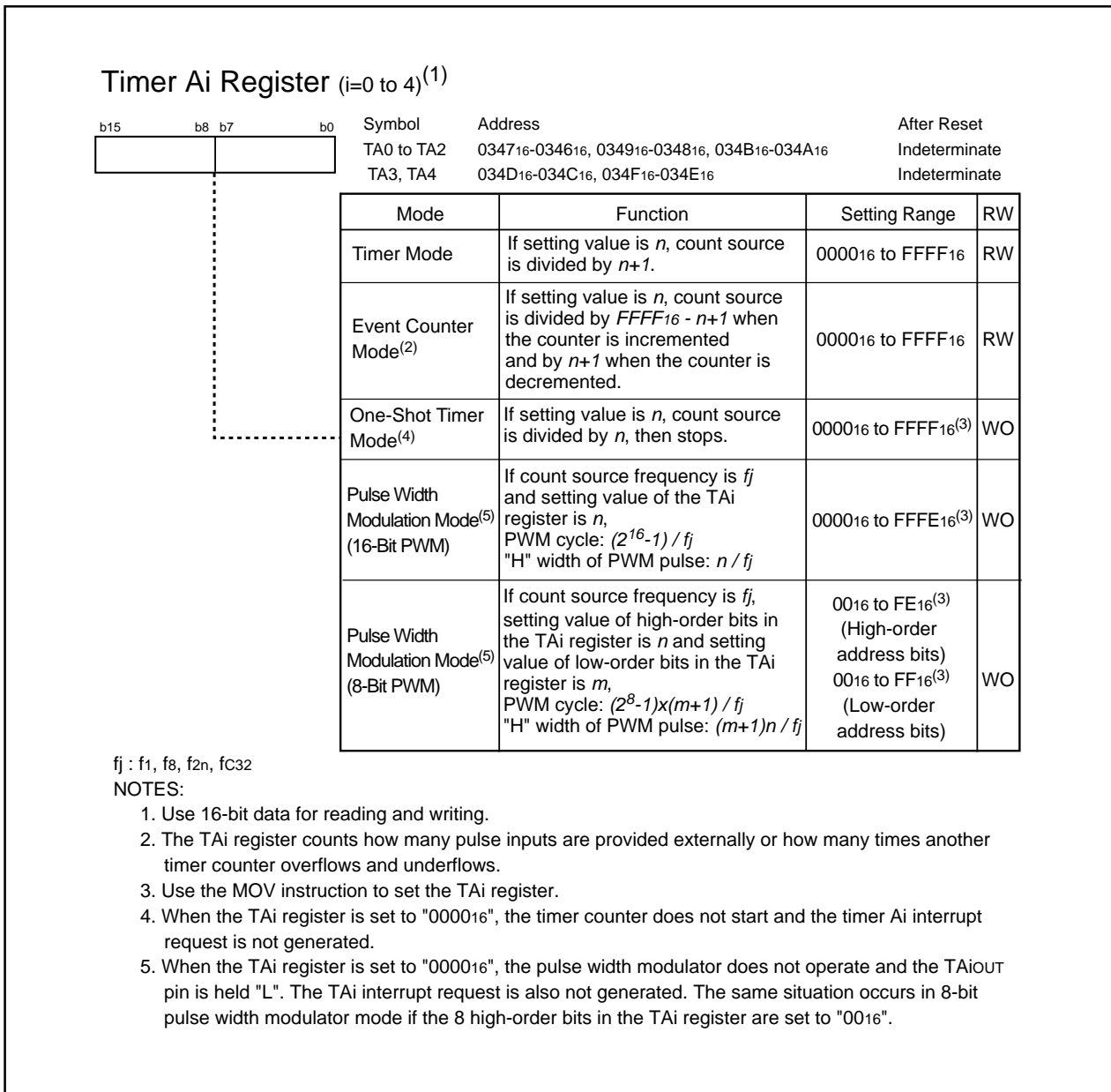


Figure 14.3 Timer A Block Diagram



**Figure 14.4 TA0 to TA4 Registers**

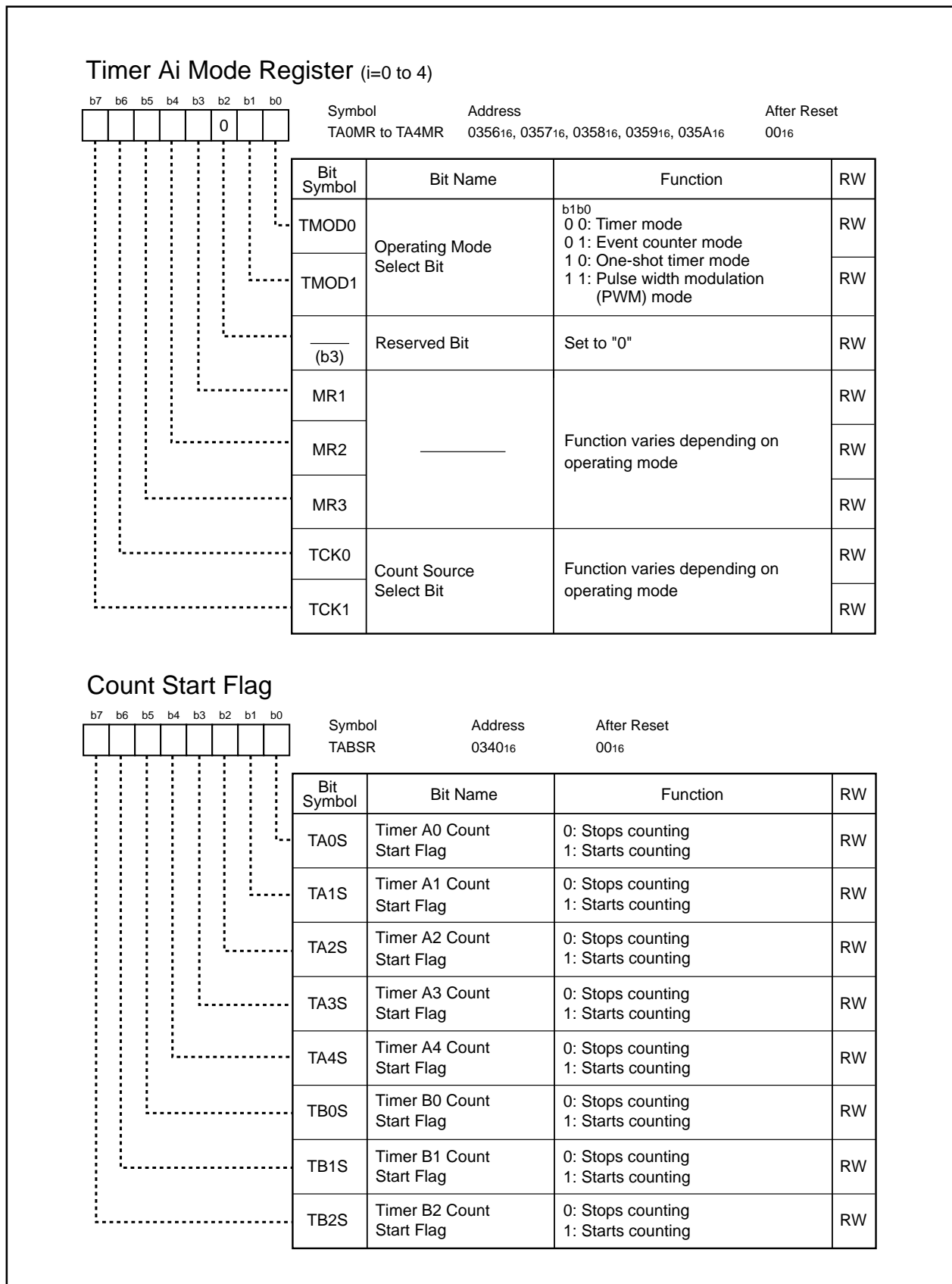


Figure 14.5 TA0MR to TA4MR Registers and TABSR Register

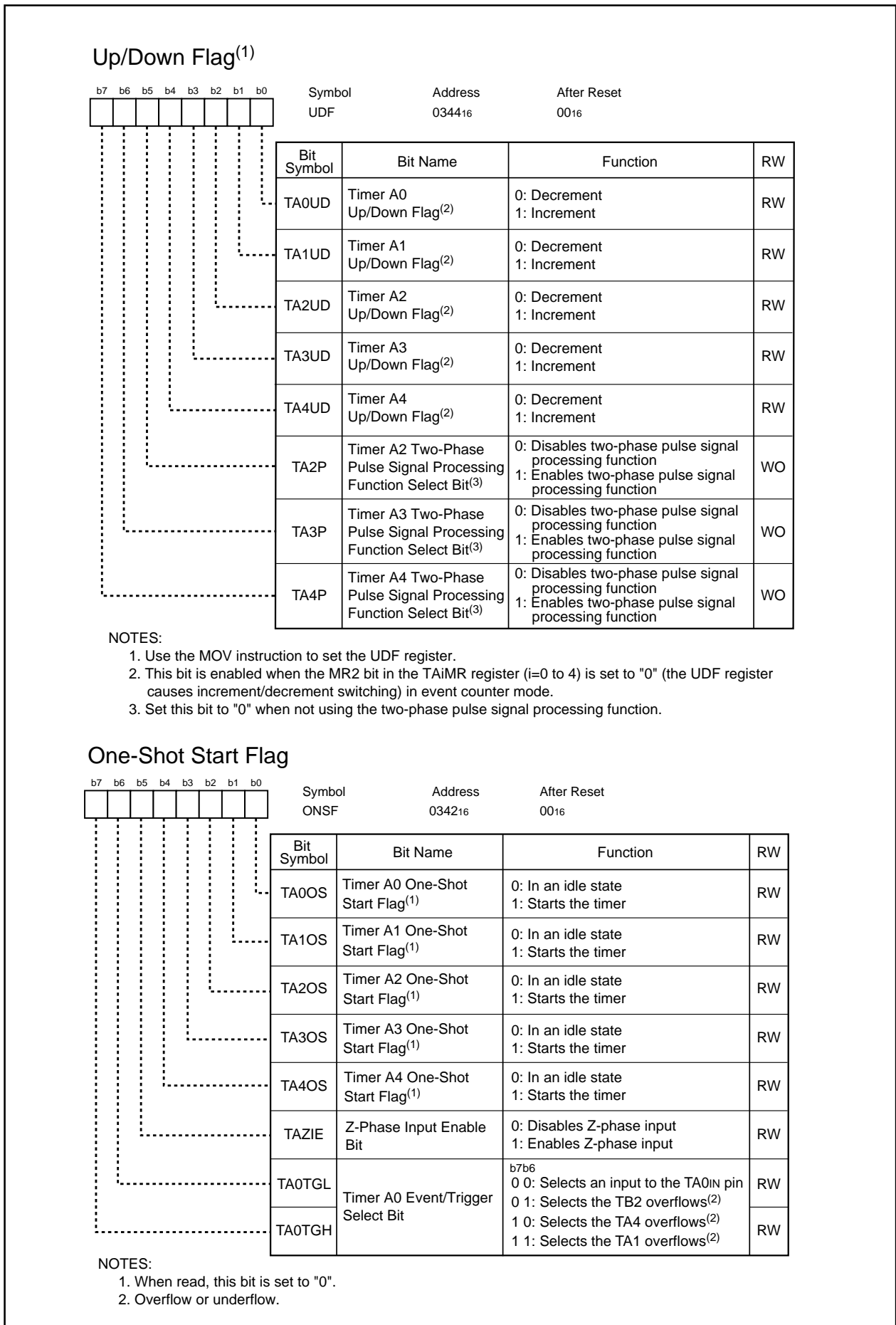


Figure 14.6 UDF Register and ONSF Register



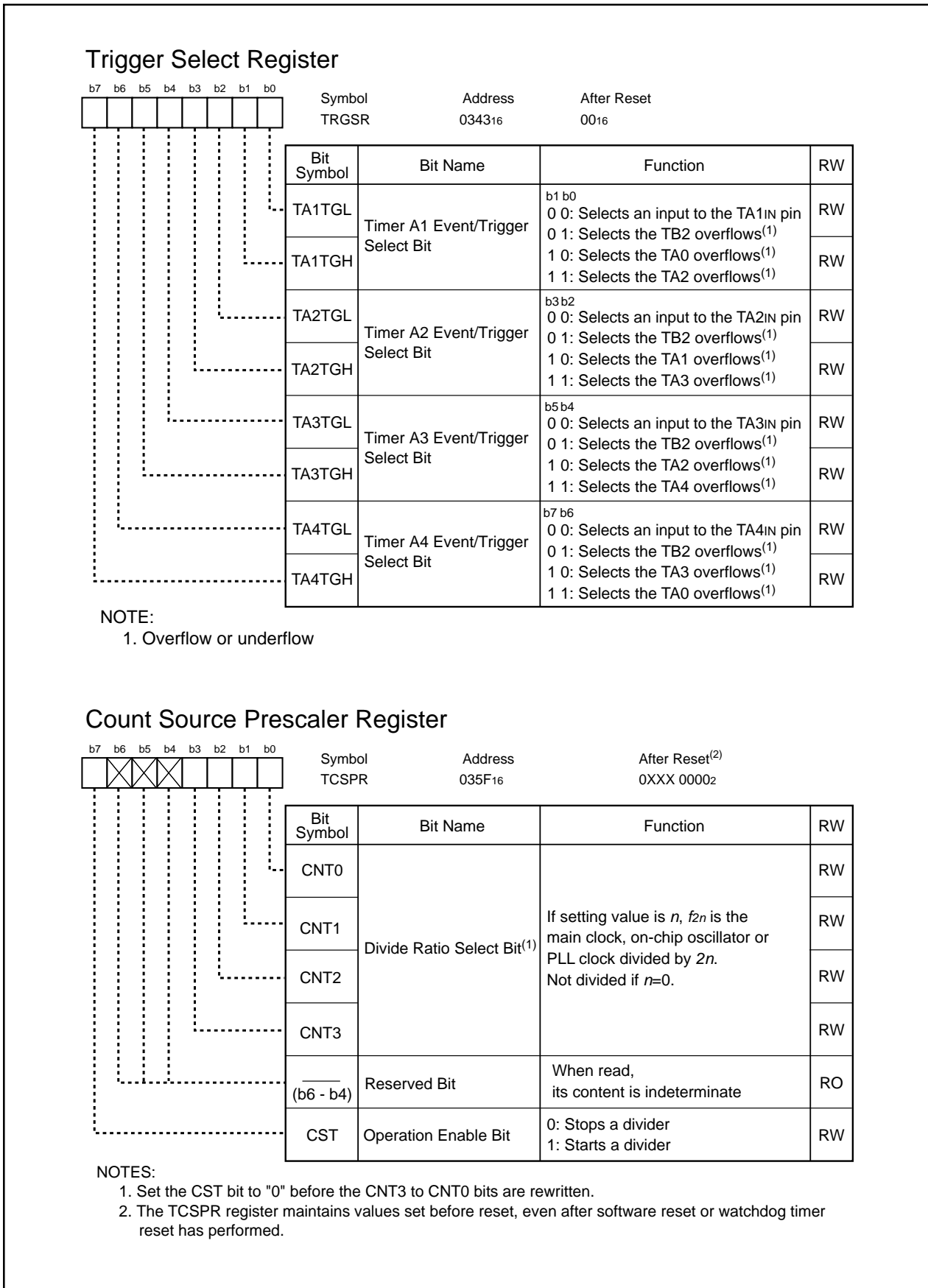


Figure 14.7 TRGSR Register and TCSPR Register

**Table 14.1 Pin Settings for Output from TAIOUT Pin (i=0 to 4)**

Pin	Setting		
	PS1, PS2 Registers	PSL1, PSL2 Registers	PSC Register
P70/TA0OUT <sup>(1)</sup>	PS1_0= 1	PSL1_0=1	PSC_0= 0
P72/TA1OUT	PS1_2= 1	PSL1_2=1	PSC_2= 0
P74/TA2OUT	PS1_4= 1	PSL1_4=0	PSC_4= 0
P76/TA3OUT	PS1_6= 1	PSL1_6=1	PSC_6= 0
P80/TA4OUT	PS2_0= 1	PSL2_0=0	–

**NOTE:**

1. P70/TA0OUT is a port for the N-channel open drain output.

**Table 14.2 Pin Settings for Input to TAIN and TAIOUT Pins (i=0 to 4)**

Pin	Setting	
	PS1, PS2 Registers	PD7, PD8 Registers
P70/TA0OUT	PS1_0=0	PD7_0=0
P71/TA0IN	PS1_1=0	PD7_1=0
P72/TA1OUT	PS1_2=0	PD7_2=0
P73/TA1IN	PS1_3=0	PD7_3=0
P74/TA2OUT	PS1_4=0	PD7_4=0
P75/TA2IN	PS1_5=0	PD7_5=0
P76/TA3OUT	PS1_6=0	PD7_6=0
P77/TA3IN	PS1_7=0	PD7_7=0
P80/TA4OUT	PS2_0=0	PD8_0=0
P81/TA4IN	PS2_1=0	PD8_1=0

### 14.1.1 Timer Mode

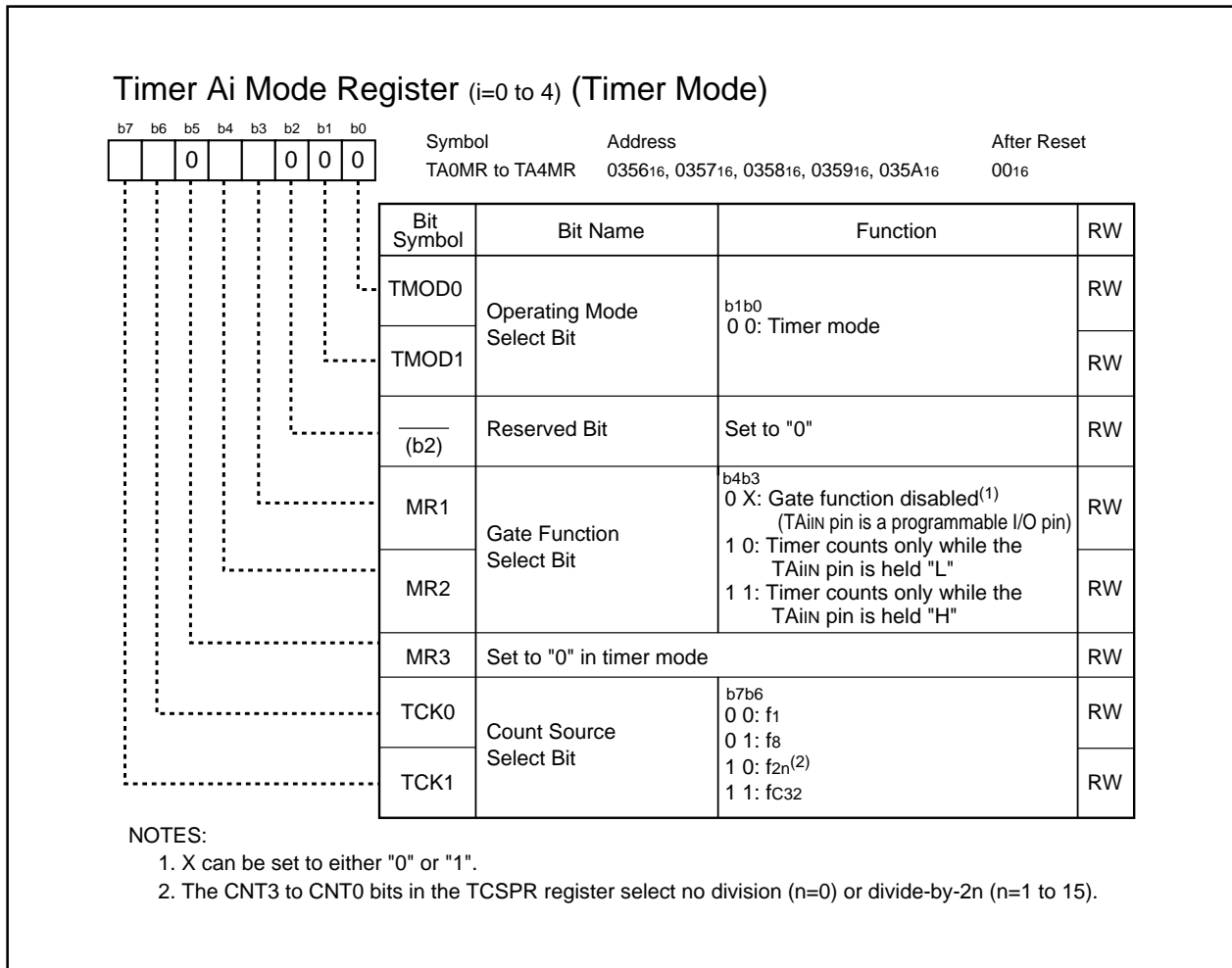
In timer mode, the timer counts an internally generated count source (see **Table 14.3**). Figure 14.8 shows the TAI<sub>MR</sub> register (i=0 to 4) in timer mode.

**Table 14.3 Timer Mode Specifications**

Item	Specification
Count Source	f <sub>1</sub> , f <sub>8</sub> , f <sub>2n</sub> <sup>(1)</sup> , f <sub>C32</sub>
Counting Operation	<ul style="list-style-type: none"> <li>The timer decrements a counter value</li> </ul> When the timer counter underflows, content of the reload register is reloaded into the count register and counting resumes.
Divide Ratio	1/(n+1)    n: setting value of the TAI register (i=0 to 4)    0000 <sub>16</sub> to FFFF <sub>16</sub>
Counter Start Condition	The TAI <sub>S</sub> bit in the TABSR register is set to "1" (starts counting)
Counter Stop Condition	The TAI <sub>S</sub> bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter underflows
TAI <sub>IN</sub> Pin Function	Programmable I/O port or gate input
TAI <sub>OUT</sub> Pin Function	Programmable I/O port or pulse output
Read from Timer	The TAI register indicates counter value
Write to Timer	<ul style="list-style-type: none"> <li>While the timer counter stops, the value written to the TAI register is also written to both reload register and counter</li> <li>While counting, the value written to the TAI register is written to the reload register (It is transferred to the counter at the next reload timing)</li> </ul>
Selectable Function	<ul style="list-style-type: none"> <li>Gate function Input signal to the TAI<sub>IN</sub> pin determines whether the timer counter starts or stops counting</li> <li>Pulse output function The polarity of the TAI<sub>OUT</sub> pin is inverted whenever the timer counter underflows</li> </ul>

**NOTE:**

- The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2<sup>n</sup> (n=1 to 15).



**Figure 14.8 TA0MR to TA4MR Registers**

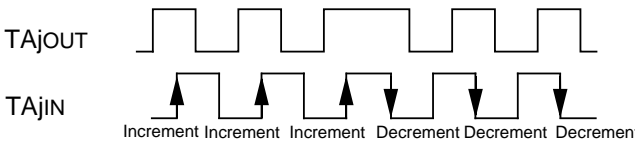
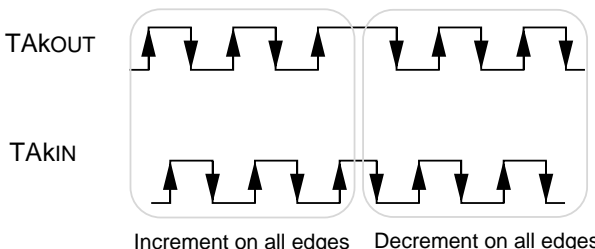
### 14.1.2 Event Counter Mode

In event counter mode, the timer counts how many external signals are applied or how many times another timer counter overflows and underflows. The timers A2, A3 and A4 can count externally generated two-phase signals. Table 14.4 lists specifications in event counter mode (when not handling a two-phase pulse signal). Table 14.5 lists specifications in event counter mode (when handling a two-phase pulse signal with the timers A2, A3 and A4). Figure 14.9 shows the TAI<sub>MR</sub> register (i=0 to 4) in event counter mode.

**Table 14.4 Event Counter Mode Specifications (When Not Processing Two-phase Pulse Signal)**

Item	Specification
Count Source	<ul style="list-style-type: none"> <li>External signal applied to the TAI<sub>IN</sub> pin (i = 0 to 4) (valid edge can be selected by program)</li> <li>Timer B2 overflow or underflow signal, timer A<sub>j</sub> overflow or underflow signal (j=i-1, except j=4 if i=0) and timer A<sub>k</sub> overflow or underflow signal (k=i+1, except k=0 if i=4)</li> </ul>
Counting Operation	<ul style="list-style-type: none"> <li>External signal and program can determine whether the timer increments or decrements a counter value</li> <li>When the timer counter underflows or overflows, content of the reload register is reloaded into the count register and counting resumes. When the free-running count function is selected, the timer counter continues running without reloading.</li> </ul>
Divide Ratio	<ul style="list-style-type: none"> <li><math>1/(FFFF_{16} - n + 1)</math> for counter increment</li> <li><math>1/(n + 1)</math> for counter decrement <math>n</math>: setting value of the TAI register <math>0000_{16}</math> to <math>FFFF_{16}</math></li> </ul>
Counter Start Condition	The TAI <sub>S</sub> bit in the TABSR register is set to "1" (starts counting)
Counter Stop Condition	The TAI <sub>S</sub> bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter overflows or underflows
TAI <sub>IN</sub> Pin Function	Programmable I/O port or count source input
TAI <sub>OUT</sub> Pin Function	Programmable I/O port, pulse output or input selecting a counter increment or decrement
Read from Timer	The TAI register indicates counter value
Write to Timer	<ul style="list-style-type: none"> <li>When the timer counter stops, the value written to the TAI register is also written to both reload register and counter</li> <li>While counting, the value written to the TAI register is written to the reload register (It is transferred to the counter at the next reload timing)</li> </ul>
Selectable Function	<ul style="list-style-type: none"> <li>Free-running count function Content of the reload register is not reloaded even if the timer counter overflows or underflows</li> <li>Pulse output function The polarity of the TAI<sub>OUT</sub> pin is inversed whenever the timer counter overflows or underflows</li> </ul>

**Table 14.5 Event Counter Mode Specifications (When Processing Two-phase Pulse Signal on Timer A2, A3 and A4)**

Item	Specification
Count Source	Two-phase pulse signal applied to the TAIIN and TAIOUT pins (i = 2 to 4)
Counting Operation	<ul style="list-style-type: none"> <li>Two-phase pulse signal determines whether the timer increments or decrements a counter value</li> <li>When the timer counter overflows or underflows, content of the reload register is reloaded into the count register and counting resumes. With the free-running count function, the timer counter continues running without reloading.</li> </ul>
Divide Ratio	<ul style="list-style-type: none"> <li><math>1/(FFFF_{16} - n + 1)</math> for counter increment</li> <li><math>1/(n + 1)</math> for counter decrement <math>n</math>: setting value of the TAI register <math>0000_{16}</math> to <math>FFFF_{16}</math></li> </ul>
Counter Start Condition	The TAI <sub>S</sub> bit in the TABSR register is set to "1" (starts counting)
Counter Stop Condition	The TAI <sub>S</sub> bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter overflows or underflows
TAiIN Pin Function	Two-phase pulse signal is applied
TAiOUT Pin Function	Two-phase pulse signal is applied
Read from Timer	The TAI register indicates the counter value
Write to Timer	<ul style="list-style-type: none"> <li>When the timer counter stops, the value written to the TAI register is also written to both reload register and counter</li> <li>While counting, the value written to the TAI register is written to the reload register (It is transferred to the counter at the next reload timing)</li> </ul>
Selectable Function <sup>(1)</sup>	<ul style="list-style-type: none"> <li>Normal processing operation (the timer A2 and timer A3) While a high-level ("H") signal is applied to the TAJOUT pin (j = 2 or 3), the timer increments a counter value on the rising edge of the TAJIN pin or decrements a counter on the falling edge.</li> </ul>  <ul style="list-style-type: none"> <li>Multiply-by-4 processing operation (the timer A3 and timer A4) While an "H" signal is applied to the TAKOUT pin (k = 3 or 4) on the rising edge of the TAKIN pin, the timer increments a counter value on the rising and falling edges of the TAKOUT and TAKIN pins. While an "H" signal is applied to the TAKOUT pin on the falling edge of the TAKIN pin, the timer decrements a counter value on the rising and falling edges of the TAKOUT and TAKIN pins.</li> </ul> 

**NOTE:**

- Only timer A3 operation can be selected. The timer A2 is for the normal processing operation. The timer A4 is for the multiply-by-4 operation.

### Timer Ai Mode Register (i=0 to 4) (Event Counter Mode)

Bit	Symbol	Bit Name	Function (When not processing two-phase pulse signal)	Function (When processing two-phase pulse signal)	RW
b7					
b6					
b5	0				
b4					
b3					
b2	0				
b1	0				
b0	1				
Symbol		Address		After Reset	
TA0MR to TA4MR		0356 <sub>16</sub> , 0357 <sub>16</sub> , 0358 <sub>16</sub> , 0359 <sub>16</sub> , 035A <sub>16</sub>		00 <sub>16</sub>	
TMOD0	Operating Mode Select Bit	b1b0 0 1: Event counter mode <sup>(1)</sup>			RW
TMOD1					RW
(b2)	Reserved Bit	Set to "0"			RW
MR1	Count Polarity Select Bit <sup>(2)</sup>	0: Counts falling edges of an external signal 1: Counts rising edges of an external signal	Set to "0"		RW
MR2	Increment/Decrement Switching Source Select Bit	0: UDF register setting 1: Input signal to TAIOUT pin <sup>(3)</sup>	Set to "1"		RW
MR3	Set to "0" in event counter mode				RW
TCK0	Count Operation Type Select Bit	0: Reloading 1: Free running			RW
TCK1	Two-Phase Pulse Signal Processing Operation Select Bit <sup>(4,5)</sup>	Set to "0"	0: Normal processing operation 1: Multiply-by-4 processing operation		RW

#### NOTES:

1. The TAITGH and TAITGL bits in the ONSF or TRGSR register determine the count source in the event counter mode.
2. MR1 bit setting is enabled only when counting how many times external signals are applied.
3. The timer decrements a counter value when an "L" signal is applied to the TAIOUT pin and the timer increments a counter value when an "H" signal is applied to the TAIOUT pin.
4. The TCK1 bit is enabled only in the TA3MR register.
5. For two-phase pulse signal processing, set the TAJp bit in the UDF register (j=2 to 4) to "1" (two-phase pulse signal processing function enabled). Also, set the TAJTGH and TAJTGL bits to "002" (input to the TAJIN pin).

Figure 14.9 TA0MR to TA4MR Registers

### 14.1.2.1 Counter Reset by Two-Phase Pulse Signal Processing

Z-phase input resets the timer counter when processing a two-phase pulse signal.

This function can be used in timer A3 event counter mode, two-phase pulse signal processing, free-running count operation type or multiply-by-4 processing. The Z-phase signal is applied to the  $\overline{\text{INT2}}$  pin. When the TAZIE bit in the ONSF register is set to "1" (Z-phase input enabled), Z-phase input can reset the timer counter. To reset the counter by a Z-phase input, set the TA3 register to "000016" beforehand.

Z-phase input is enabled when the edge of the signal applied to the  $\overline{\text{INT2}}$  pin is detected. The POL bit in the INT2IC register can determine edge polarity. The Z-phase must have a pulse width of one timer A3 count source cycle or more. Figure 14.10 shows two-phase pulses (A-phase and B-phase) and the Z-phase.

Z-phase input resets the timer counter in the next count source following Z-phase input. Figure 14.11 shows the counter reset timing.

Timer A3 interrupt request is generated twice continuously when a timer A3 overflow or underflow, and a counter reset by  $\overline{\text{INT2}}$  input occur at the same time. Do not use the timer A3 interrupt request when this function is used.

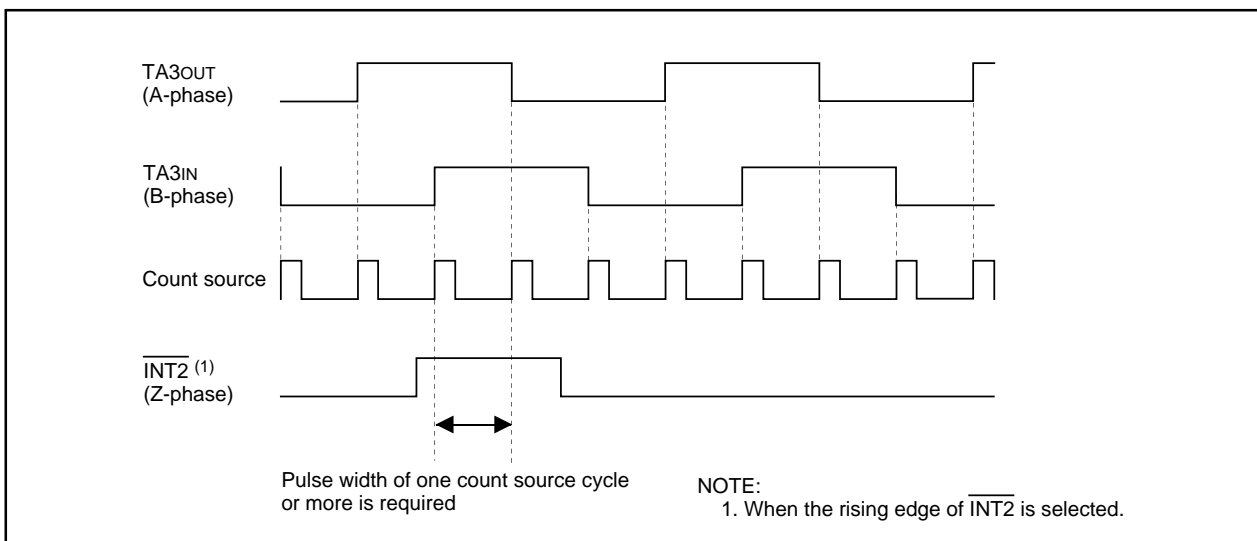


Figure 14.10 Two-Phase Pulse (A-phase and B-phase) and Z-phase

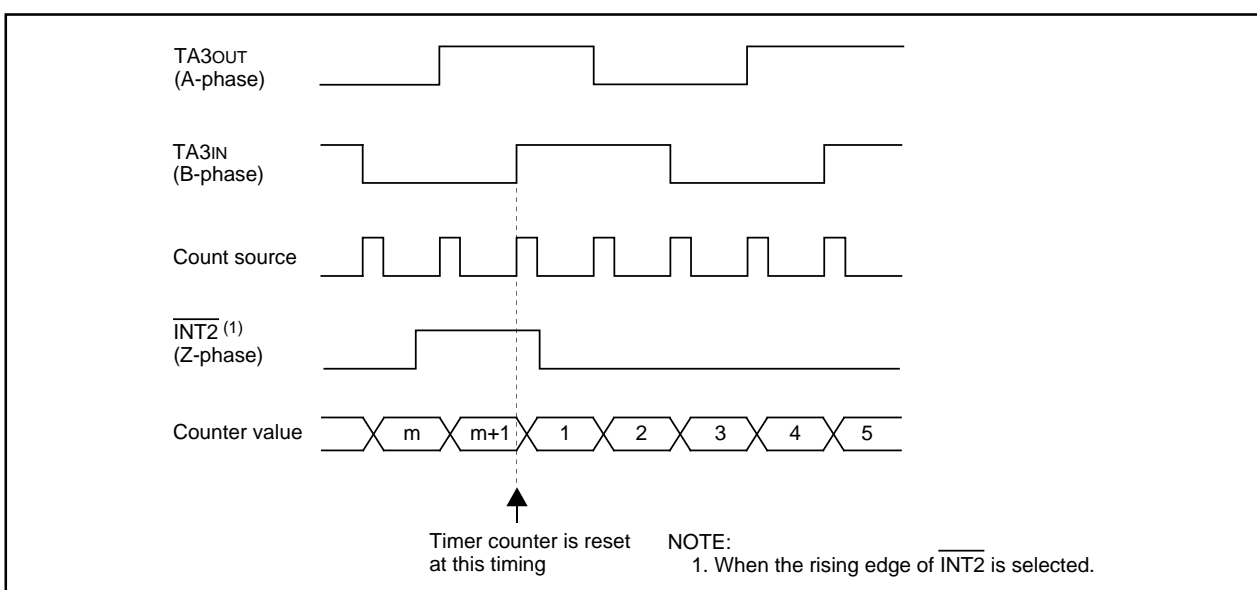


Figure 14.11 Counter Reset Timing



### 14.1.3 One-Shot Timer Mode

In one-shot timer mode, the timer operates only once for each trigger (see **Table 14.6**). Once a trigger occurs, the timer starts and continues operating for a desired period. Figure 14.12 shows the TAI<sub>MR</sub> register (i=0 to 4) in one-shot timer mode.

**Table 14.6 One-Shot Timer Mode Specifications**

Item	Specification
Count Source	f <sub>1</sub> , f <sub>8</sub> , f <sub>2<sup>n</sup></sub> <sup>(1)</sup> , f <sub>C32</sub>
Counting Operation	<ul style="list-style-type: none"> <li>The timer decrements a counter value</li> </ul> When the timer counter reaches "0000 <sub>16</sub> ", it stops counting after reloading. If a trigger occurs while counting, content of the reload register is reloaded into the count register and counting resumes.
Divide Ratio	1/n    n: setting value of the TAI <sub>i</sub> register (i=0 to 4) 0000 <sub>16</sub> to FFFF <sub>16</sub> , but the timer counter does not run if n=0000 <sub>16</sub>
Counter Start Condition	The TAI <sub>S</sub> bit in the TABSR register is set to "1" (starts counting) and following triggers occur: <ul style="list-style-type: none"> <li>External trigger input is provided</li> <li>Timer counter overflows or underflows</li> <li>The TAI<sub>OS</sub> bit in the ONSF register is set to "1" (timer started)</li> </ul>
Counter Stop Condition	<ul style="list-style-type: none"> <li>After the timer counter has reached "0000<sub>16</sub>" and is reloaded</li> <li>When the TAI<sub>S</sub> bit is set to "0" (stops counting)</li> </ul>
Interrupt Request Generation Timing	The timer counter reaches "0000 <sub>16</sub> "
TAI <sub>IN</sub> Pin Function	Programmable I/O port or trigger input
TAI <sub>OUT</sub> Pin Function	Programmable I/O port or pulse output
Read from Timer	The value in the TAI <sub>i</sub> register is indeterminate when read
Write to Timer	<ul style="list-style-type: none"> <li>When the timer counter stops, the value written to the TAI<sub>i</sub> register is also written to both reload register and counter</li> <li>While counting, the value written to the TAI<sub>i</sub> register is written to the reload register (It is transferred to the counter at the next reload timing)</li> </ul>

**NOTE:**

- The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2<sup>n</sup> (n=1 to 15).

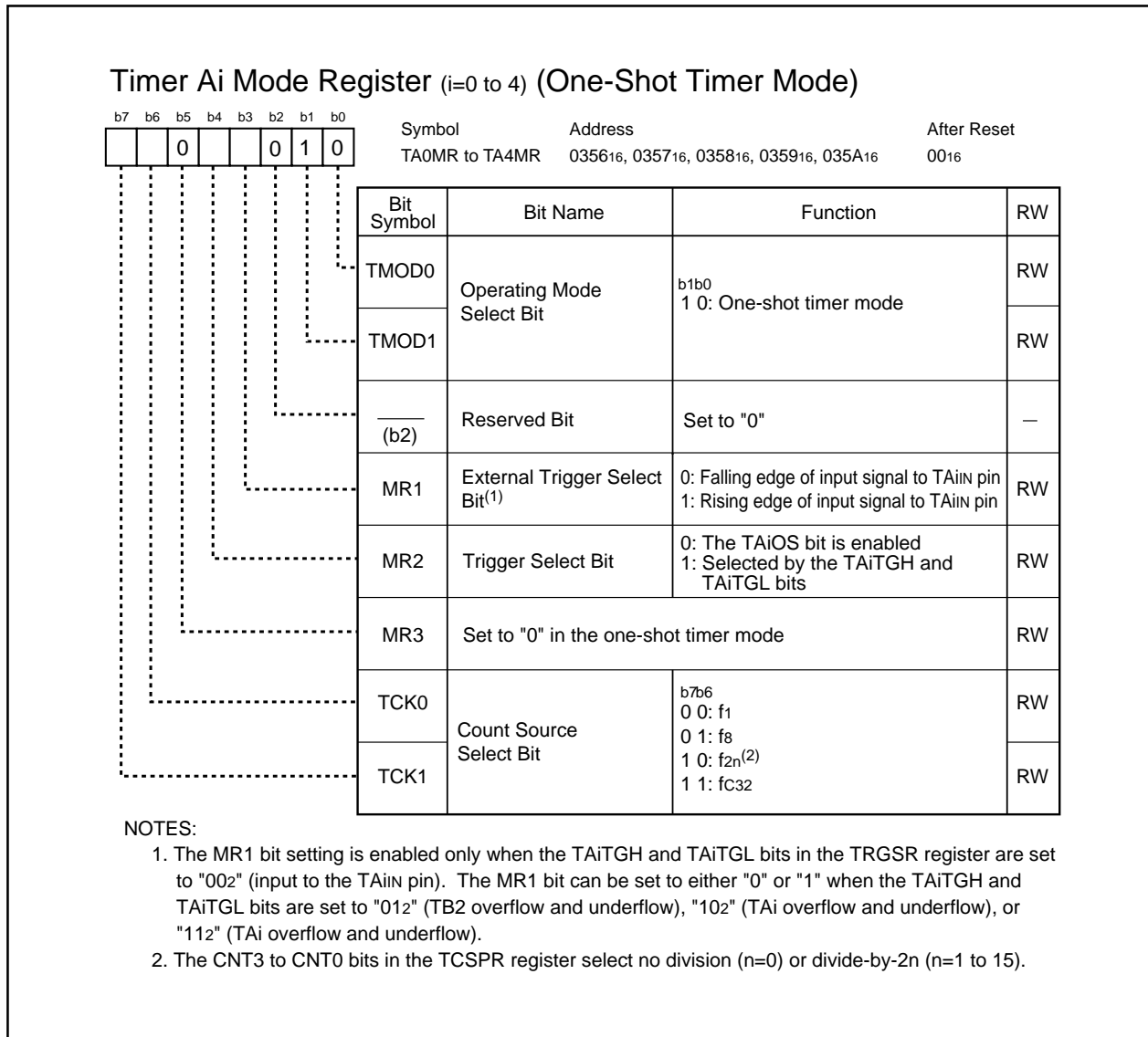


Figure 14.12 TA0MR to TA4MR Registers

### 14.1.4 Pulse Width Modulation Mode

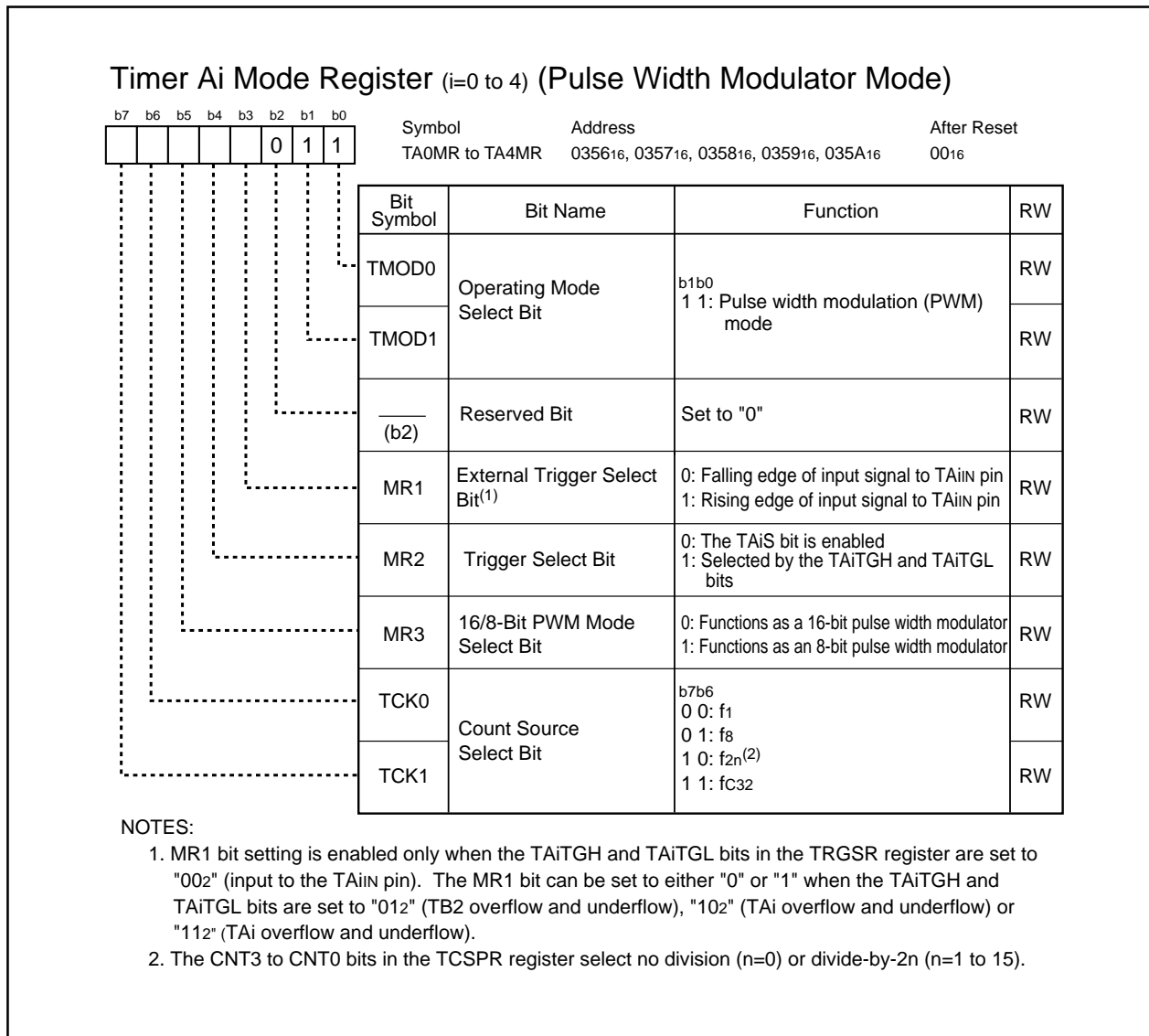
In pulse width modulation mode, the timer outputs pulse of desired width continuously (see **Table 14.7**). The timer counter functions as either 16-bit pulse width modulator or 8-bit pulse width modulator. Figure 14.13 shows the TAI<sub>i</sub>MR register (i=0 to 4) in pulse width modulation mode. Figures 14.14 and 14.15 show examples of how a 16-bit pulse width modulator operates and of how an 8-bit pulse width modulator operates.

**Table 14.7 Pulse Width Modulation Mode Specifications**

Item	Specification
Count Source	f1, f8, f2n <sup>(1)</sup> , fC32
Counting Operation	<ul style="list-style-type: none"> <li>The timer decrements a counter value (The counter functions as an 8-bit or a 16-bit pulse width modulator) Content of the reload register is reloaded on the rising edge of PWM pulse and counting continues.</li> <li>The timer is not affected by a trigger that is generated during counting.</li> </ul>
16-Bit PWM	<ul style="list-style-type: none"> <li>"H" width = <math>n / f_j</math>      <math>n</math>: setting value of the TAI register    0000<sub>16</sub> to FFFE<sub>16</sub> <math>f_j</math>: count source frequency</li> <li>Cycle = <math>(2^{16}-1) / f_j</math> fixed</li> </ul>
8-Bit PWM	<ul style="list-style-type: none"> <li>"H" width = <math>n \times (m+1) / f_j</math></li> <li>Cycles = <math>(2^8-1) \times (m+1) / f_j</math></li> <li><math>m</math>: setting value of low-order bit address of the TAI register    00<sub>16</sub> to FF<sub>16</sub></li> <li><math>n</math>: setting value of high-order bit address of the TAI register    00<sub>16</sub> to FE<sub>16</sub></li> </ul>
Counter Start Condition	<ul style="list-style-type: none"> <li>External trigger input is provided</li> <li>Timer counter overflows or underflows</li> <li>The TAI<sub>S</sub> bit in the TABSR register is set to "1" (starts counting)</li> </ul>
Counter Stop Condition	The TAI <sub>S</sub> bit is set to "0" (stops counting)
Interrupt Request Generation Timing	On the falling edge of the PWM pulse
TAI <sub>i</sub> N Pin Function	Programmable I/O port or trigger input
TAI <sub>i</sub> OUT Pin Function	Pulse output
Read from Timer	The value in the TAI register is indeterminate when read
Write to Timer	<ul style="list-style-type: none"> <li>When the timer counter stops, the value written to the TAI register is also written to both reload register and counter</li> <li>While counting, the value written to the TAI register is written to the reload register (It is transferred to the counter at the next reload timing)</li> </ul>

**NOTE:**

- The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).



**Figure 14.13 TA0MR to TA4MR Registers**

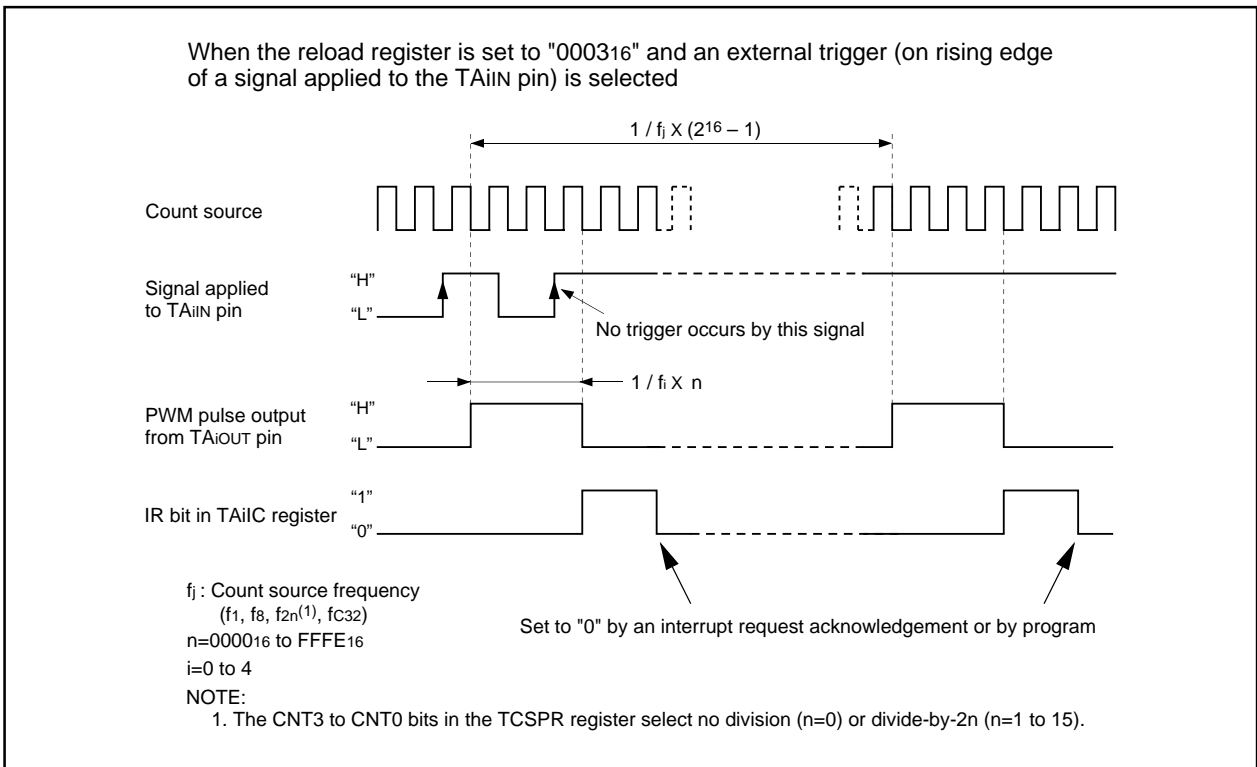


Figure 14.14 16-bit Pulse Width Modulator Operation

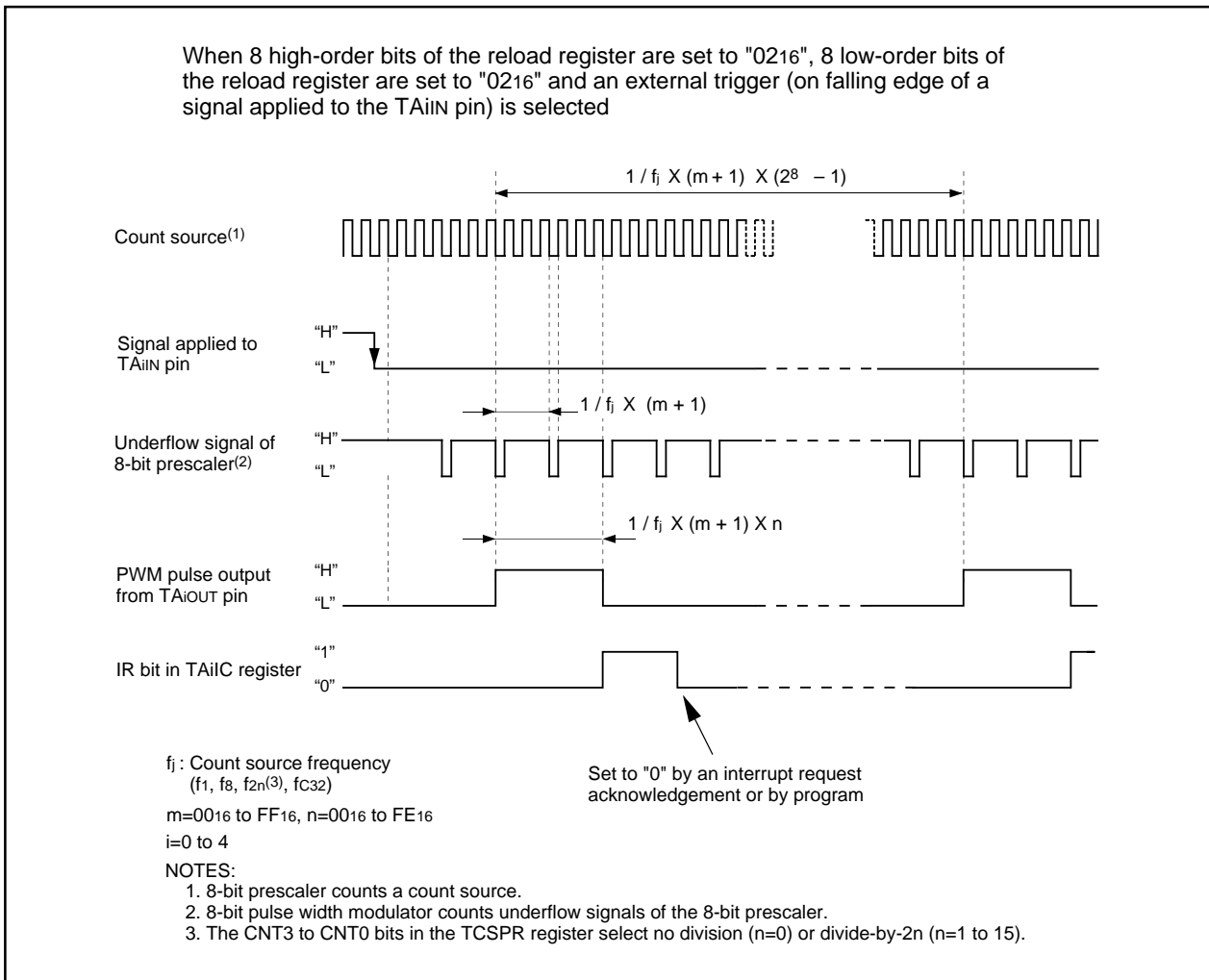


Figure 14.15 8-bit Pulse Width Modulator Operation

## 14.2 Timer B

Figure 14.16 shows a block diagram of the timer B. Figures 14.17 to 14.19 show registers associated with the timer B. The timer B supports the following three modes. The TMOD1 and TMOD0 bits in the TBiMR register ( $i=0$  to 5) determine which mode is used.

- Timer mode : The timer counts an internal count source.
- Event counter mode : The timer counts pulses from an external source or overflow and underflow of another timer.
- Pulse period/pulse width measurement mode : The timer measures pulse period or pulse width of an external signal.

Table 14.8 lists TBiIN pin settings.

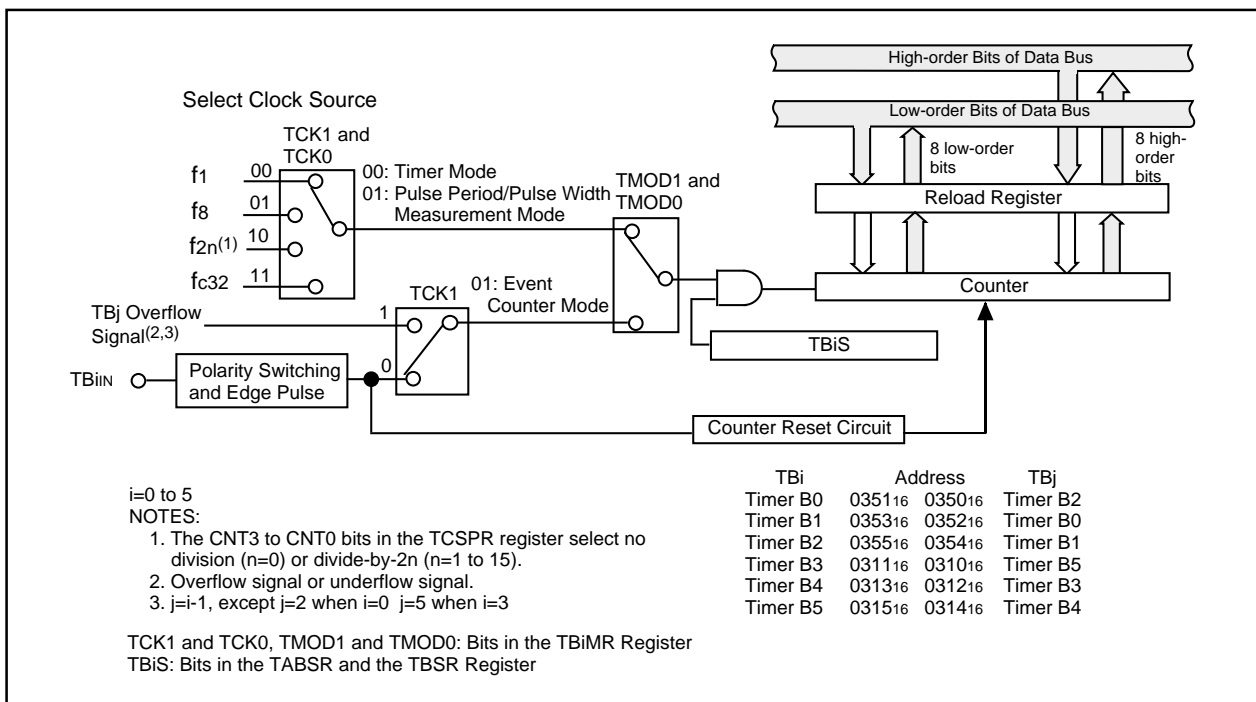


Figure 14.16 Timer B Block Diagram

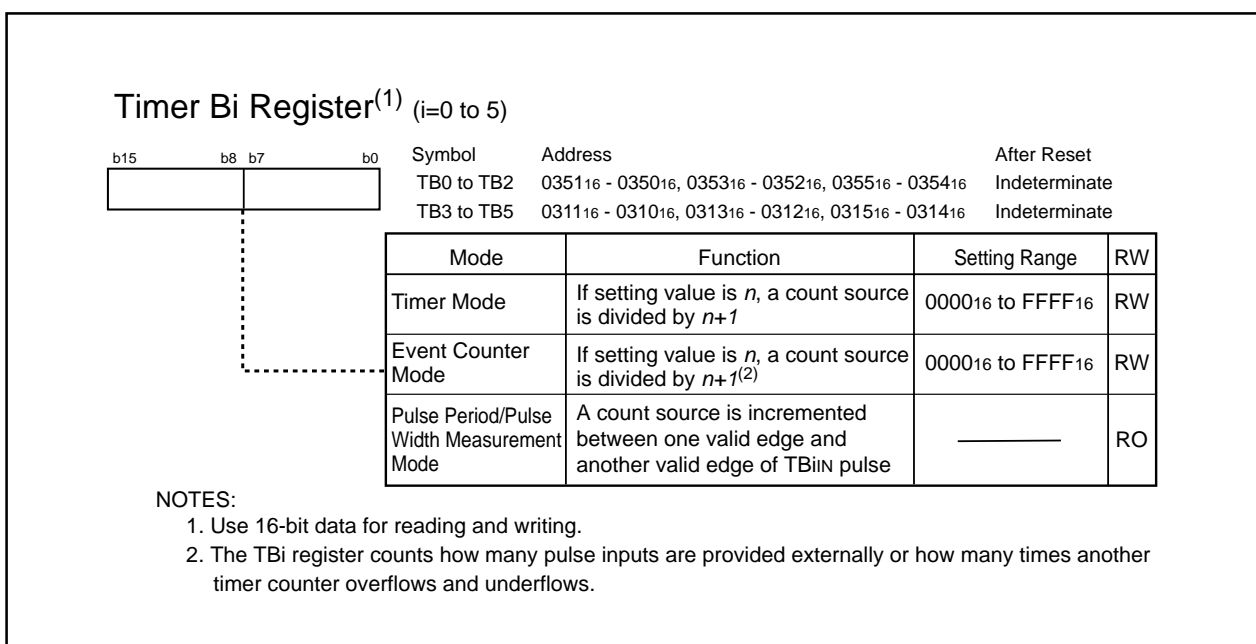


Figure 14.17 TB0 to TB5 Registers

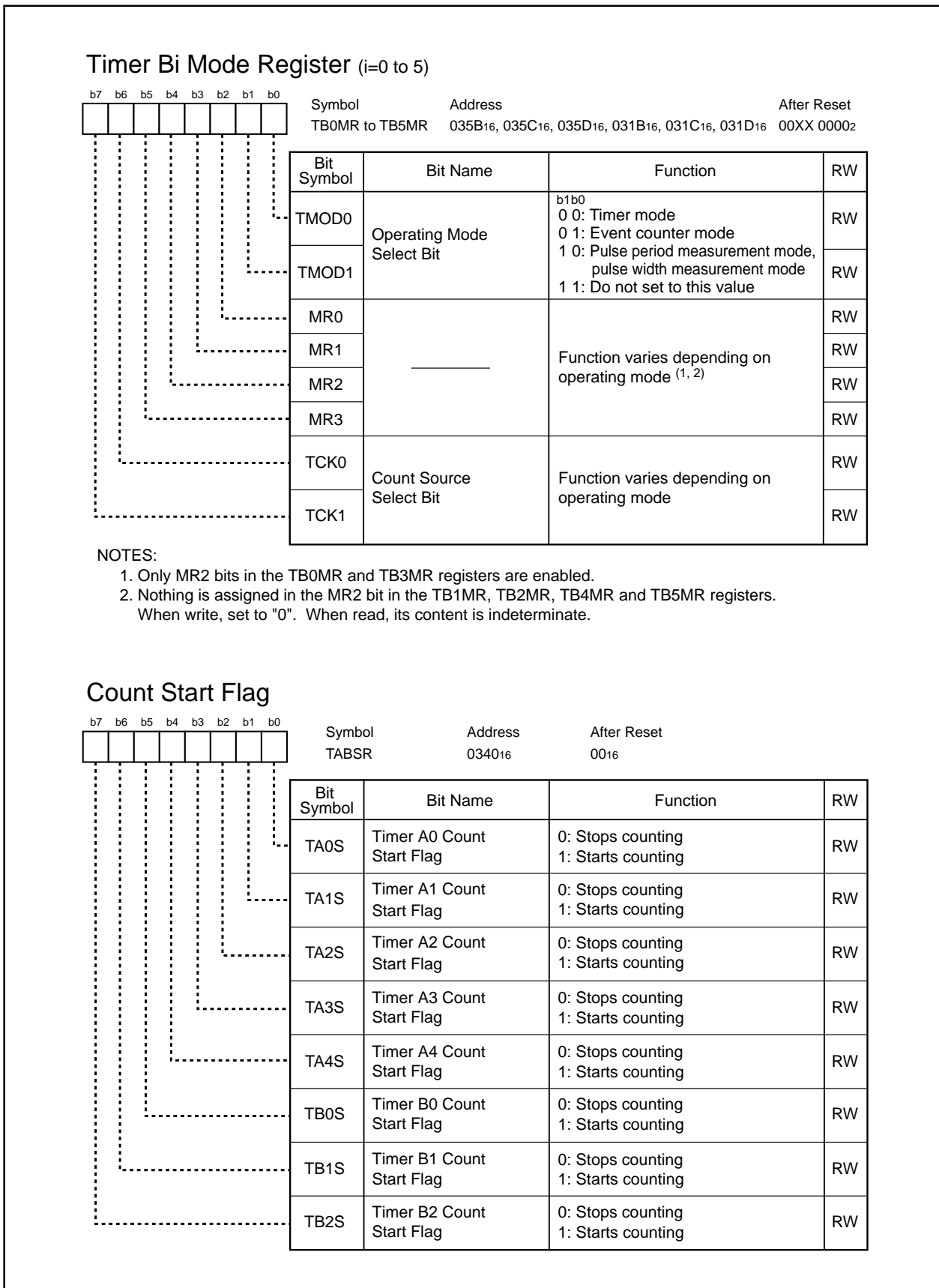


Figure 14.18 TB0MR to TB5MR Registers, TABSR Register

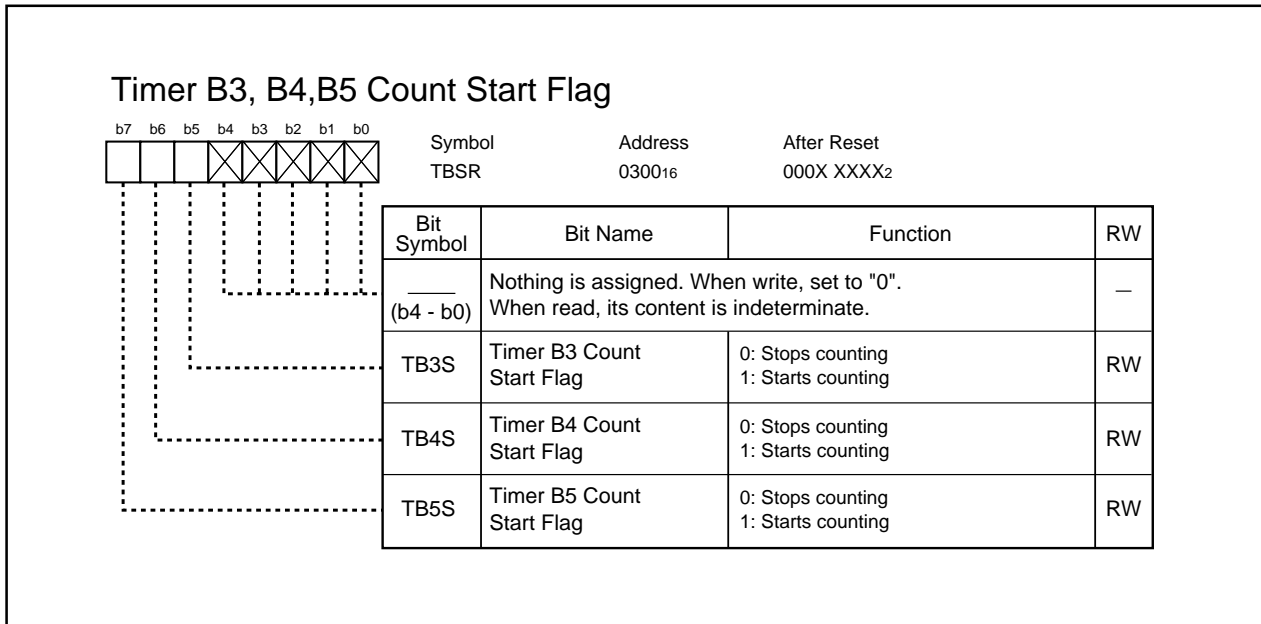


Figure 14.19 TBSR Register

Table 14.8 Settings for the TBiIN Pins (i=0 to 5)

Port Name	Function	Setting	
		PS1, PS3 <sup>(1)</sup> Registers	PD7, PD9 <sup>(1)</sup> Registers
P90	TB0IN	PS3_0=0	PD9_0=0
P91	TB1IN	PS3_1=0	PD9_1=0
P92	TB2IN	PS3_2=0	PD9_2=0
P93	TB3IN	PS3_3=0	PD9_3=0
P94	TB4IN	PS3_4=0	PD9_4=0
P71	TB5IN	PS1_1=0	PD7_1=0

## NOTE:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.



### 14.2.1 Timer Mode

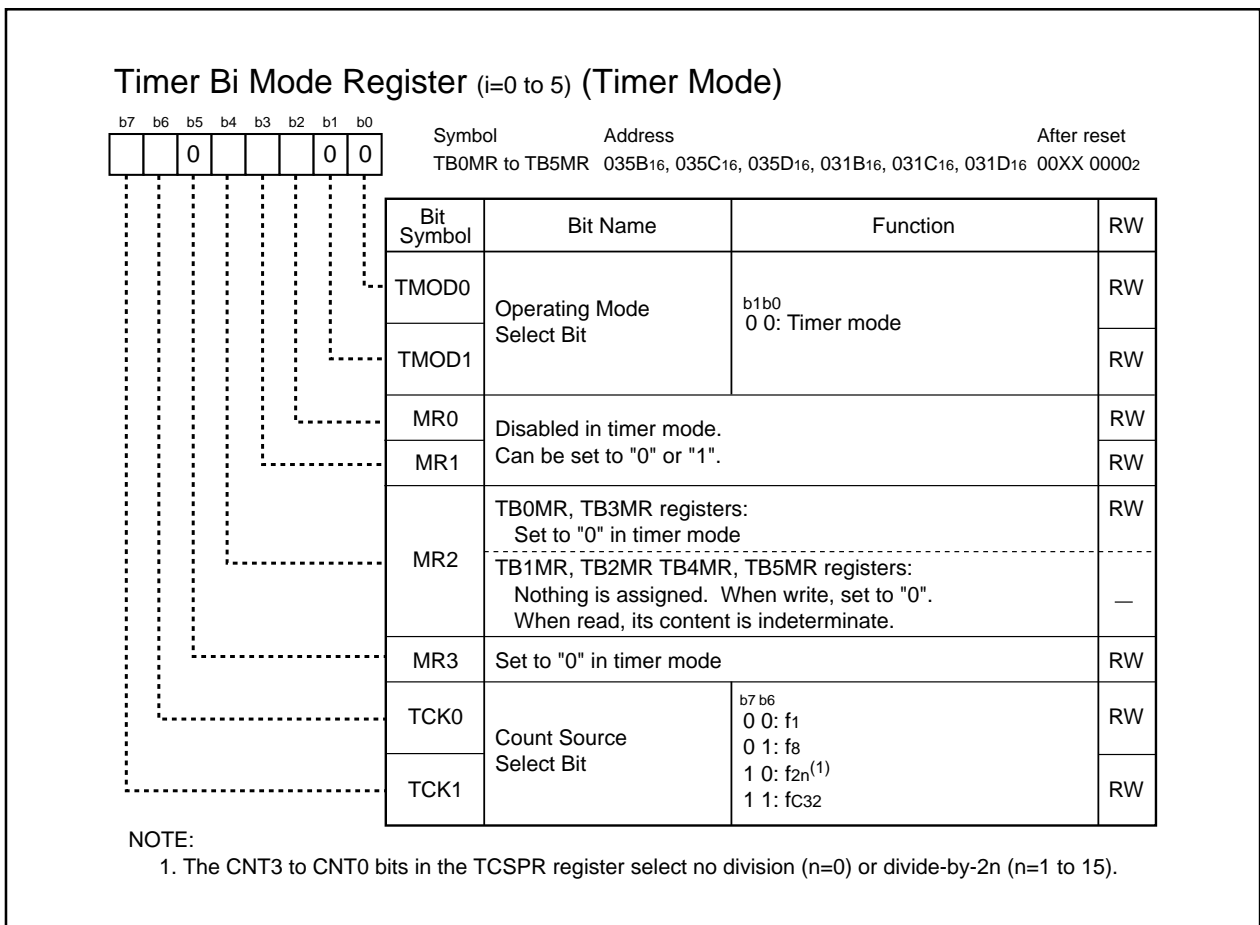
In timer mode, the timer counts an internally generated count source (see **Table 14.9**). Figure 14.20 shows the TBiMR register (i=0 to 5) in timer mode.

**Table 14.9 Timer Mode Specifications**

Item	Specification
Count Source	f1, f8, f2n <sup>(1)</sup> , fc32
Counting Operation	<ul style="list-style-type: none"> <li>The timer decrements a counter value</li> </ul> When the timer counter underflows, content of the reload register is reloaded into the count register and counting resumes
Divide Ratio	$1/(n+1)$ n: setting value of the TBi register (i=0 to 5) 0000 <sub>16</sub> to FFFF <sub>16</sub>
Counter Start Condition	The TBiS bits in the TABSR and TBSR registers are set to "1" (starts counting)
Counter Stop Condition	The TBiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	Timer counter underflows
TBiIn Pin Function	Programmable I/O port
Read from Timer	The TBi register indicates counter value
Write to Timer	<ul style="list-style-type: none"> <li>When the timer counter stops, the value written to the TBi register is also written to both reload register and counter</li> <li>While counting, the value written to the TBi register is written to the reload register (It is transferred to the counter at the next reload timing)</li> </ul>

NOTE:

- The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).



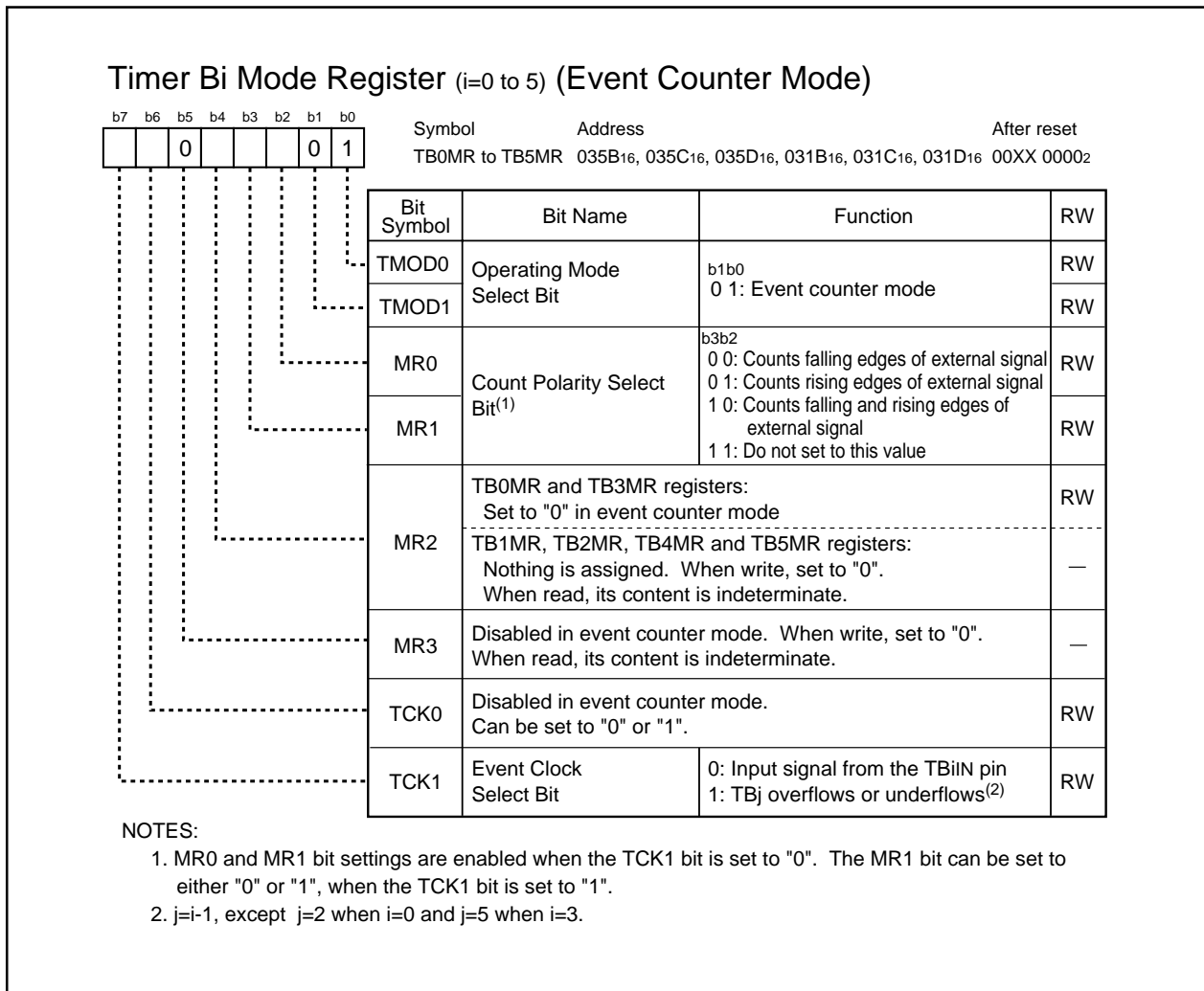
**Figure 14.20 TB0MR to TB5MR Registers**

### 14.2.2 Event Counter Mode

In event counter mode, the timer counts how many external signals are applied or how many times another timer overflows and underflows. (See **Table 14.10**) Figure 14.21 shows the TBiMR register (i=0 to 5) in event counter mode.

**Table 14.10 Event Counter Mode Specifications**

Item	Specification
Count Source	<ul style="list-style-type: none"> <li>External signal applied to the TBiIN pin (i = 0 to 5) (valid edge can be selected by program)</li> <li>TBj overflow or underflow signal (j=i-1, except j=2 when i=0, j=5 when i=3)</li> </ul>
Counting Operation	<ul style="list-style-type: none"> <li>The timer decrements a counter value</li> </ul> When the timer counter underflows, content of the reload register is reloaded into the count register to continue counting
Divide Ratio	$1/(n+1)$ $n$ : setting value of the TBi register      0000 <sub>16</sub> to FFFF <sub>16</sub>
Counter Start Condition	The TBiS bits in the TABSR and TBSR register are set to "1" (starts counting)
Counter Stop Condition	The TBiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter underflows
TBiIN Pin Function	Programmable I/O port or count source input
Read from Timer	The TBi register indicates counter value
Write to Timer	<ul style="list-style-type: none"> <li>When the timer counter stops, the value written to the TBi register is also written to both reload register and counter</li> <li>While counting, the value written to the TBi register is written to the reload register (It is transferred to the counter at the next reload timing)</li> </ul>



**Figure 14.21 TB0MR to TB5MR Registers**

### 14.2.3 Pulse Period/Pulse Width Measurement Mode

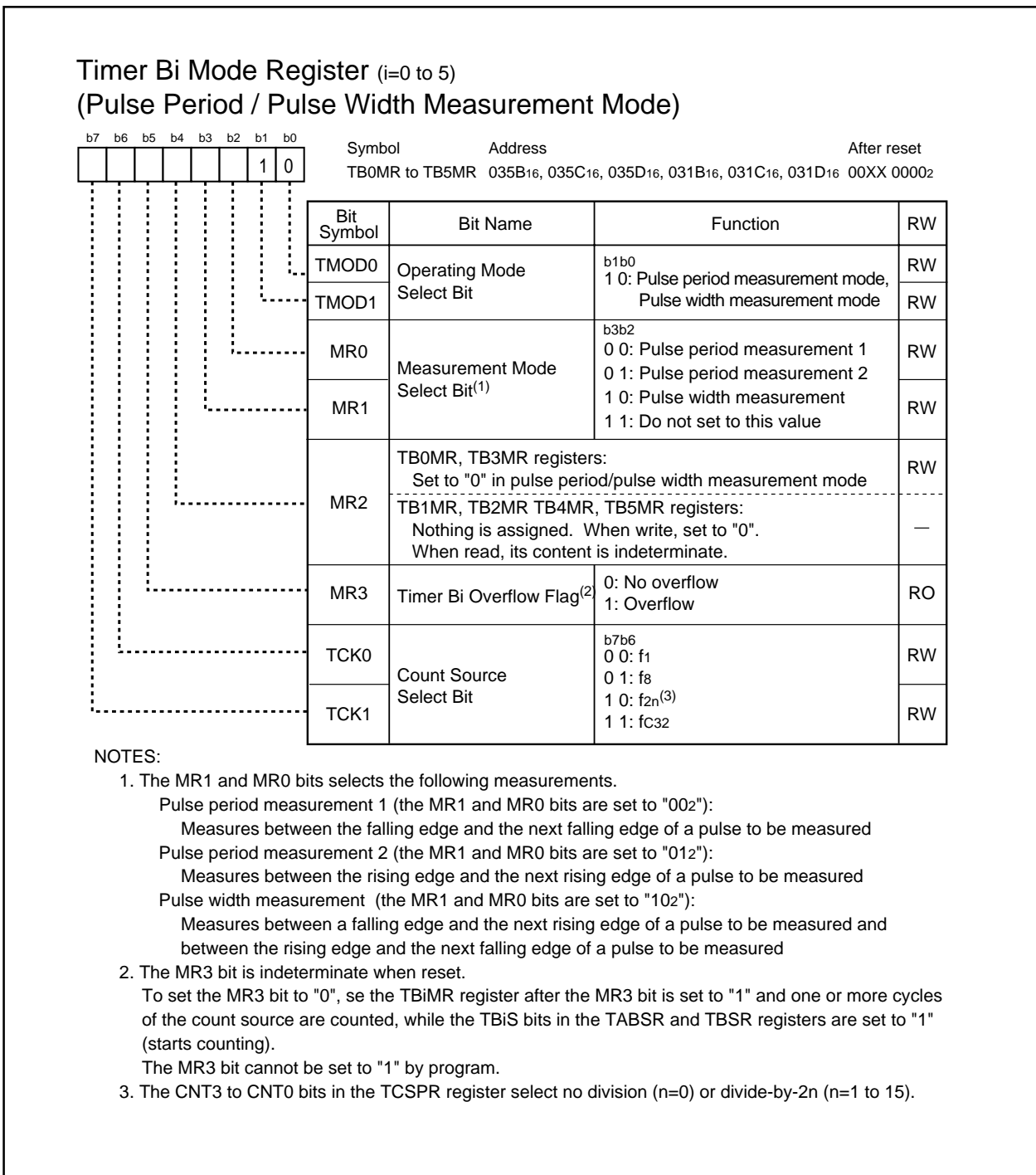
In pulse period/pulse width measurement mode, the timer measures pulse period or pulse width of an external signal. (See **Table 14.11**) Figure 14.22 shows the TBiMR register (i=0 to 5) in pulse period/pulse width measurement mode. Figure 14.23 shows an operation example in pulse period measurement mode. Figure 14.24 shows an operation example in the pulse width measurement mode.

**Table 14.11 Pulse Period/Pulse Width Measurement Mode Specifications**

Item	Specification
Count Source	f1, f8, f2n <sup>(3)</sup> , fc32
Counting Operation	<ul style="list-style-type: none"> <li>The timer increments a counter value</li> </ul> Counter value is transferred to the reload register on the valid edge of a pulse to be measured. It is set to "000016" and the timer continues counting
Counter Start Condition	The TBiS bits (i=0 to 5) in the TABSR and TBSR register are set to "1" (starts counting)
Counter Stop Condition	The TBiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>On the valid edge of a pulse to be measured<sup>(1)</sup></li> <li>The timer counter overflows</li> </ul> The MR3 bit in the TBiMR register is set to "1" (overflow) simultaneously. When the TBiS bit is set to "1" (start counting) and the next count source is counted after setting the MR3 bit to "1" (overflow), the MR3 bit can be set to "0" (no overflow) by writing to the TBiMR register.
TBiIN Pin Function	Input for a pulse to be measured
Read from Timer	The TBi register indicates reload register values (measurement results) <sup>(2)</sup>
Write to Timer	Value written to the TBi register can be written to neither reload register nor counter

**NOTES:**

- No interrupt request is generated when the pulse to be measured is on the first valid edge after the timer has started counting.
- The TBi register is in an indeterminate state until the pulse to be measured is on the second valid edge after the timer has started counting.
- The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).



**Figure 14.22 TB0MR to TB5MR Registers**

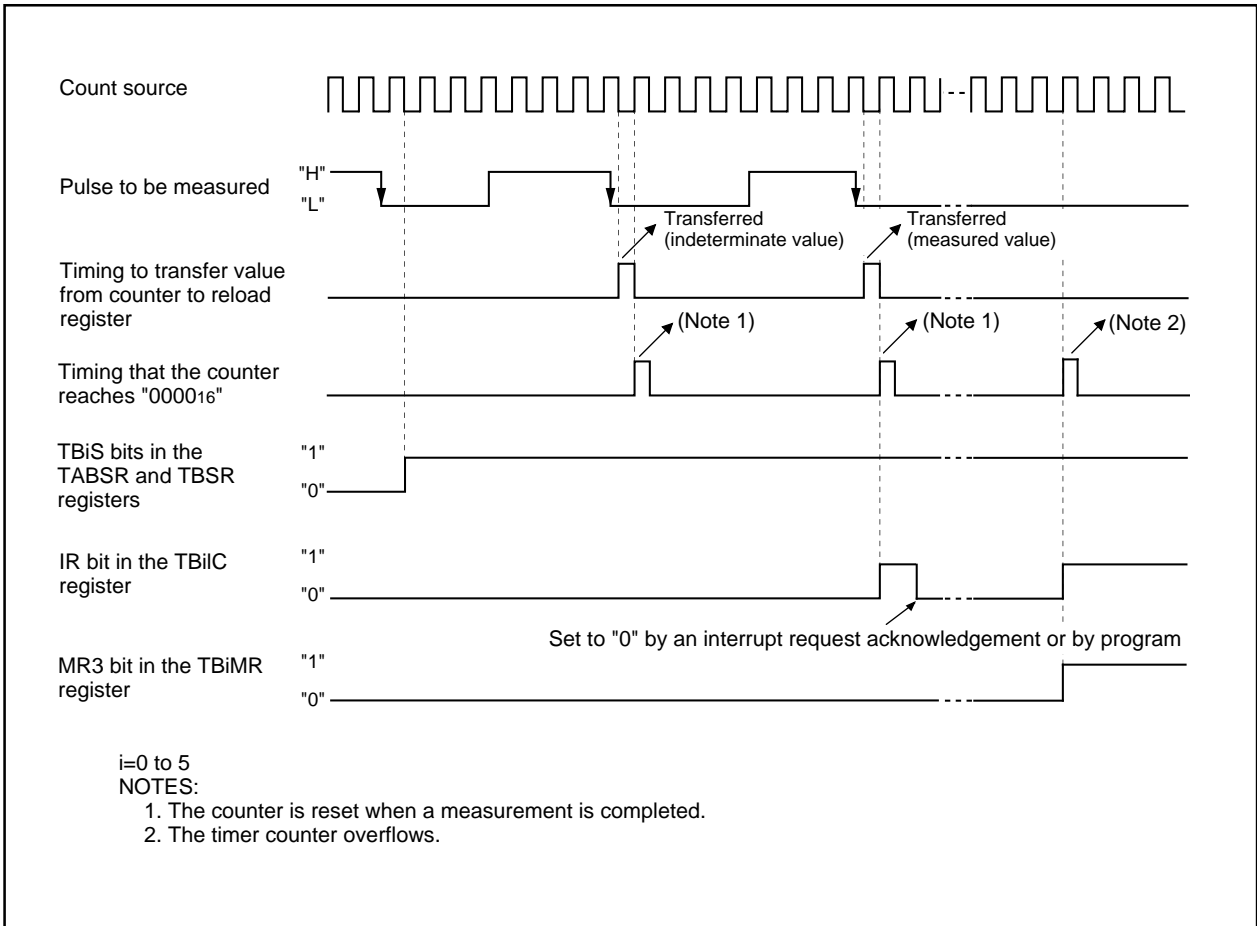


Figure 14.23 Operation Example in Pulse Period Measurement Mode

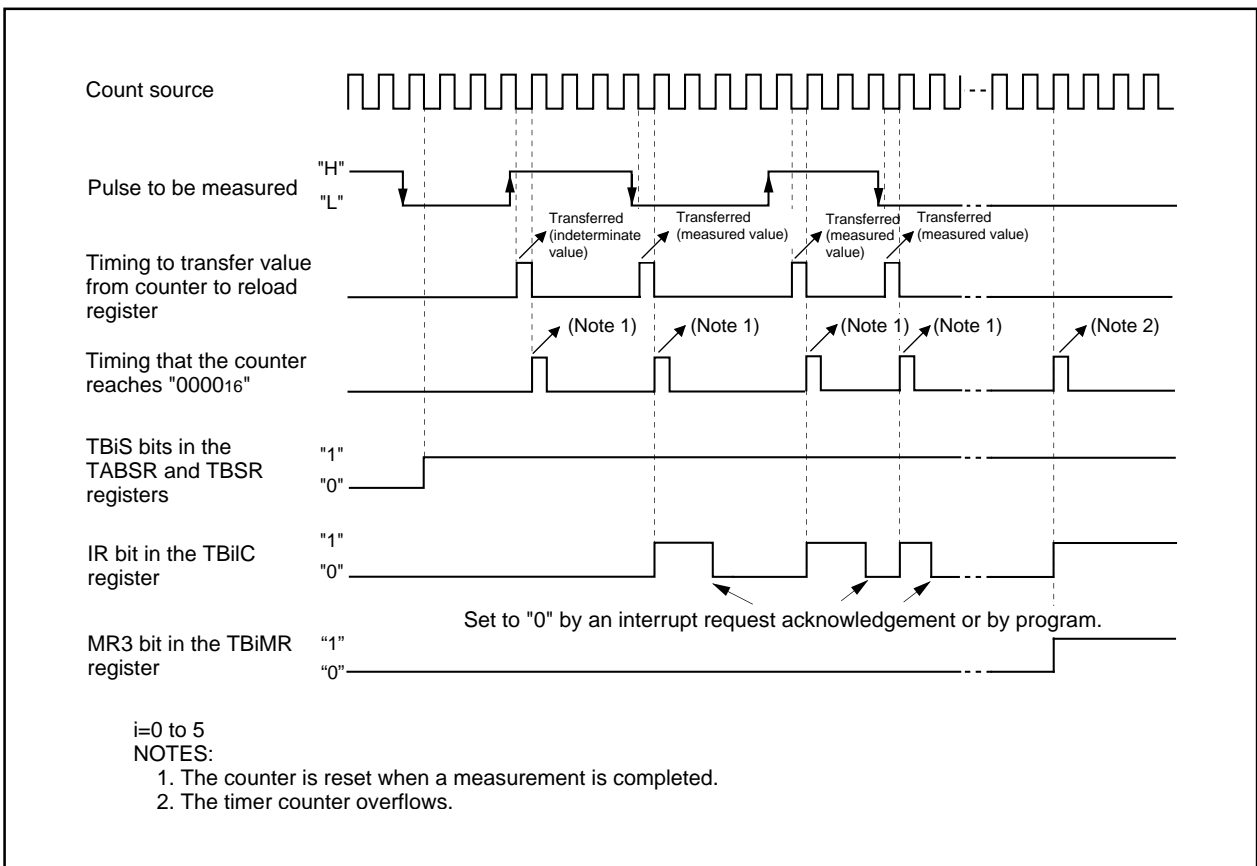


Figure 14.24 Operation Example in Pulse Width Measurement Mode

## 15. Three-Phase Motor Control Timer Functions

Three-phase motor driving waveform can be output by using the timers A1, A2, A4 and B2. Table 15.1 lists specifications of the three-phase motor control timer functions. Table 15.2 lists pin settings. Figure 15.1 shows a block diagram. Figures 15.2 to 15.7 show registers associated with the three-phase motor control timer functions.

**Table 15.1 Three-Phase Motor Control Timer Functions Specification**

Item	Specification
Three-Phase Waveform Output Pin	Six pins (U, $\bar{U}$ , V, $\bar{V}$ , W, $\bar{W}$ )
Forced Cutoff <sup>(1)</sup>	Apply a low-level ("L") signal to the NMI pin
Timers to be Used	Timer A4, A1, A2 (used in 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 timer mode): Carrier wave cycle control Dead time timer (three 8-bit timers share reload register): Dead time control
Output Waveform	Triangular wave modulation, Sawtooth wave modulation Can output a high-level waveform or a low-level waveform for one cycle; Can set positive-phase level and negative-phase level separately
Carrier Wave Cycle	Triangular wave modulation: $count\ source \times (m+1) \times 2$ Sawtooth wave modulation: $count\ source \times (m+1)$ $m$ : setting value of the TB2 register, 0000 <sub>16</sub> to FFFF <sub>16</sub> Count source: f <sub>1</sub> , f <sub>8</sub> , f <sub>2n<sup>(2)</sup></sub> , f <sub>c32</sub>
Three-Phase PWM Output Width	Triangular wave modulation: $count\ source \times n \times 2$ Sawtooth wave modulation: $count\ source \times n$ $n$ : setting value of the TA4, TA1 and TA2 register (of the TA4, TA41, TA1, TA11, TA2 and TA21 registers when setting the INV11 bit to "1"), 0001 <sub>16</sub> to FFFF <sub>16</sub> Count source: f <sub>1</sub> , f <sub>8</sub> , f <sub>2n<sup>(2)</sup></sub> , f <sub>c32</sub>
Dead Time	$Count\ source \times p$ , or no dead time $p$ : setting value of the DTT register, 01 <sub>16</sub> to FF <sub>16</sub> Count source: f <sub>1</sub> , or f <sub>1</sub> divided by 2
Active Level	Selected from a high level ("H") or low level ("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 the timer B2 interrupt, one carrier wave cycle-to-cycle basis through 15 time- carrier wave cycle-to-cycle basis can be selected

**NOTES:**

1. Forced cutoff by the signal applied to the  $\overline{NMI}$  pin is available when the INV02 bit is set to "1" (three-phase motor control timer functions) and the INV03 bit is set to "1" (three-phase motor control timer output enabled).
2. The CNT3 to CNT0 bits in the TCSPPR register select no division (n=0) or divide-by-2n (n=1 to 15).

**Table 15.2 Pin Settings**

Pin	Setting		
	PS1, PS2 Registers <sup>(1)</sup>	PSL1, PSL2 Registers	PSC Register
P72/V	PS1_2 =1	PSL1_2 =0	PSC_2 =1
P73/ $\bar{V}$	PS1_3 =1	PSL1_3 =1	PSC_3 =0
P74/W	PS1_4 =1	PSL1_4 =1	PSC_4 =0
P75/ $\bar{W}$	PS1_5 =1	PSL1_5 =0	—
P80/U	PS2_0 =1	PSL2_0 =1	—
P81/ $\bar{U}$	PS2_1 =1	PSL2_1 =0	—

## NOTE:

1. Set the PS1\_5 to PS1\_2 bits and PS2\_1 and PS2\_0 bits in the PS1 and PS2 registers to "1" after the INV02 bit is set to "1".



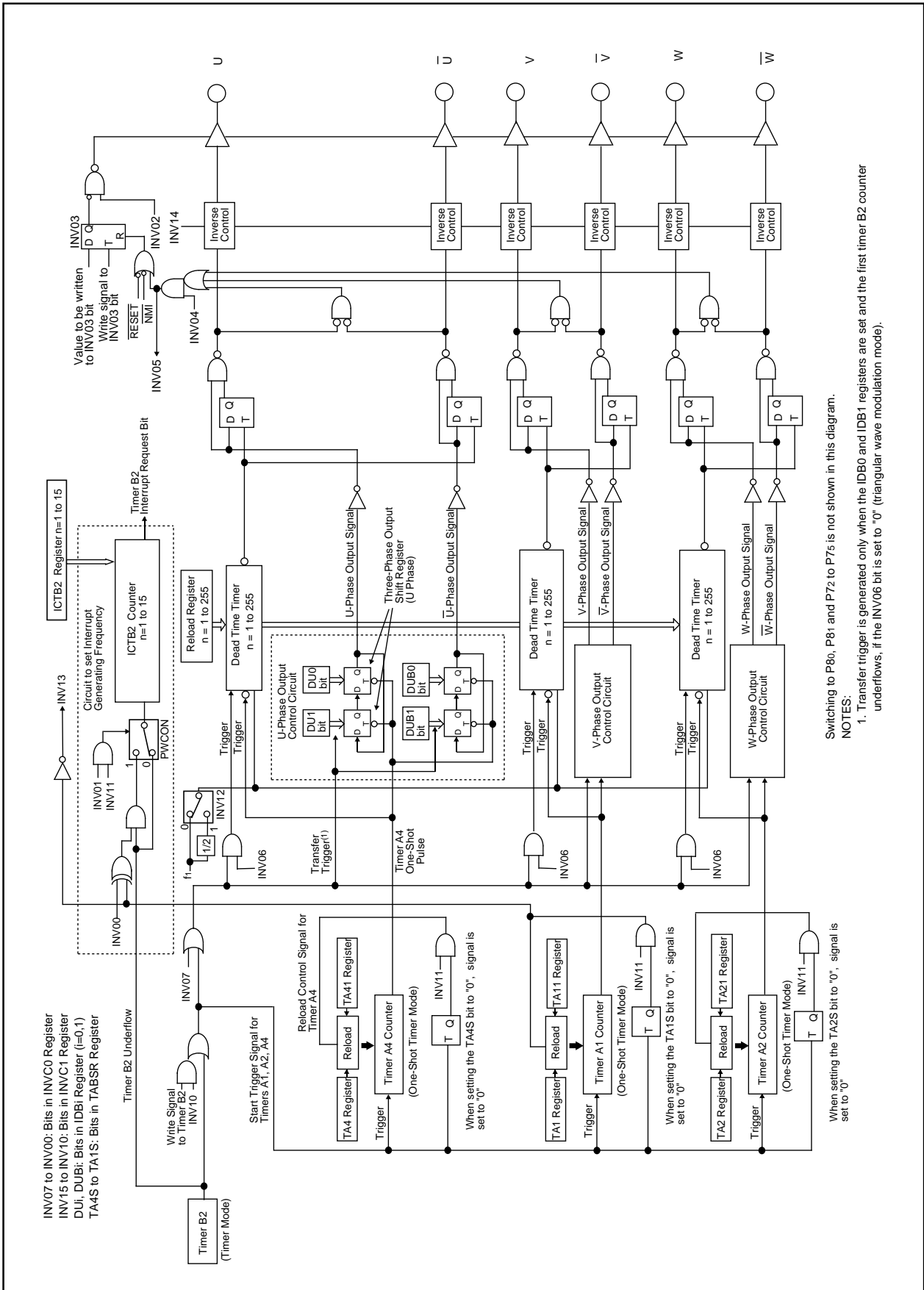
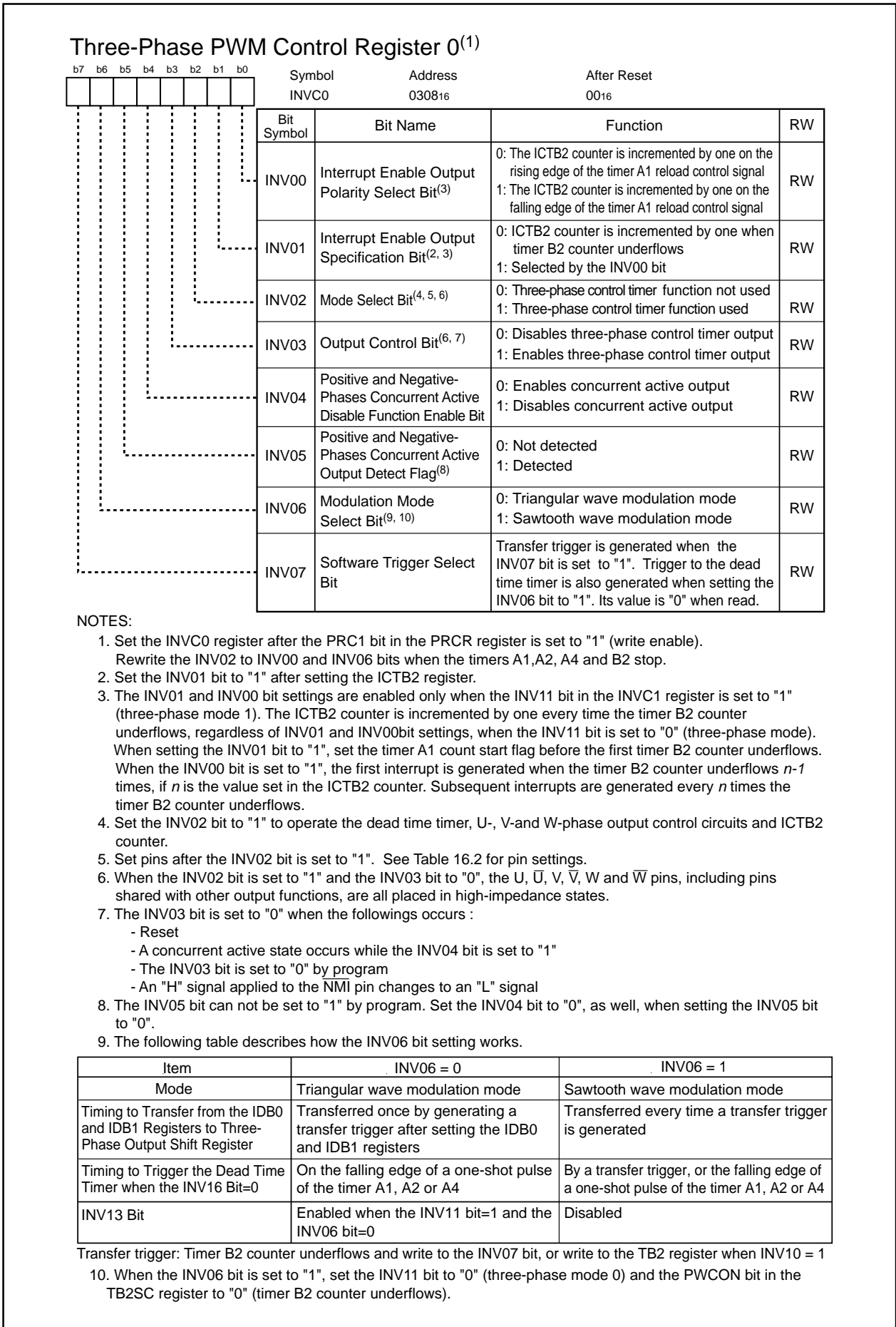
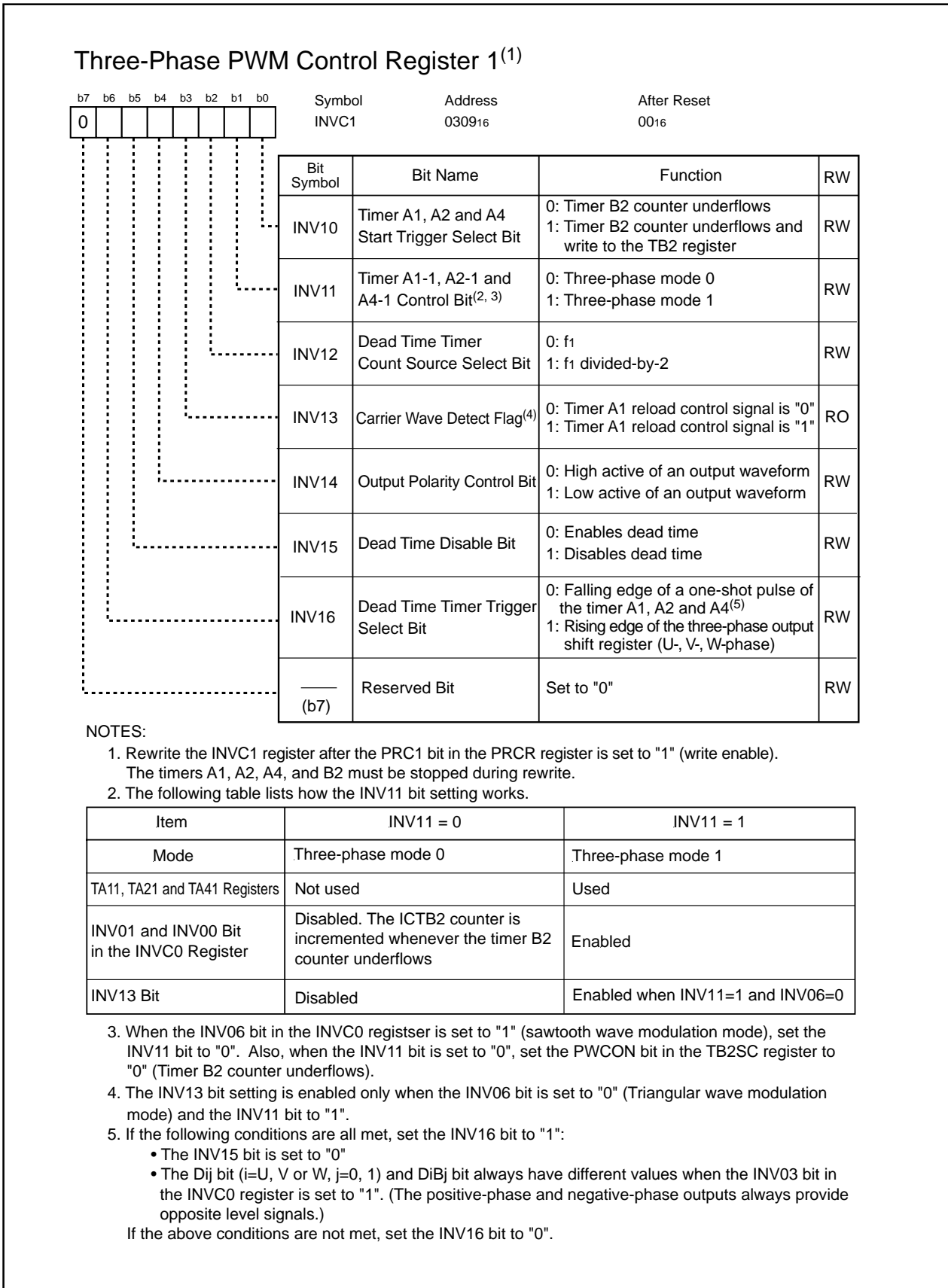


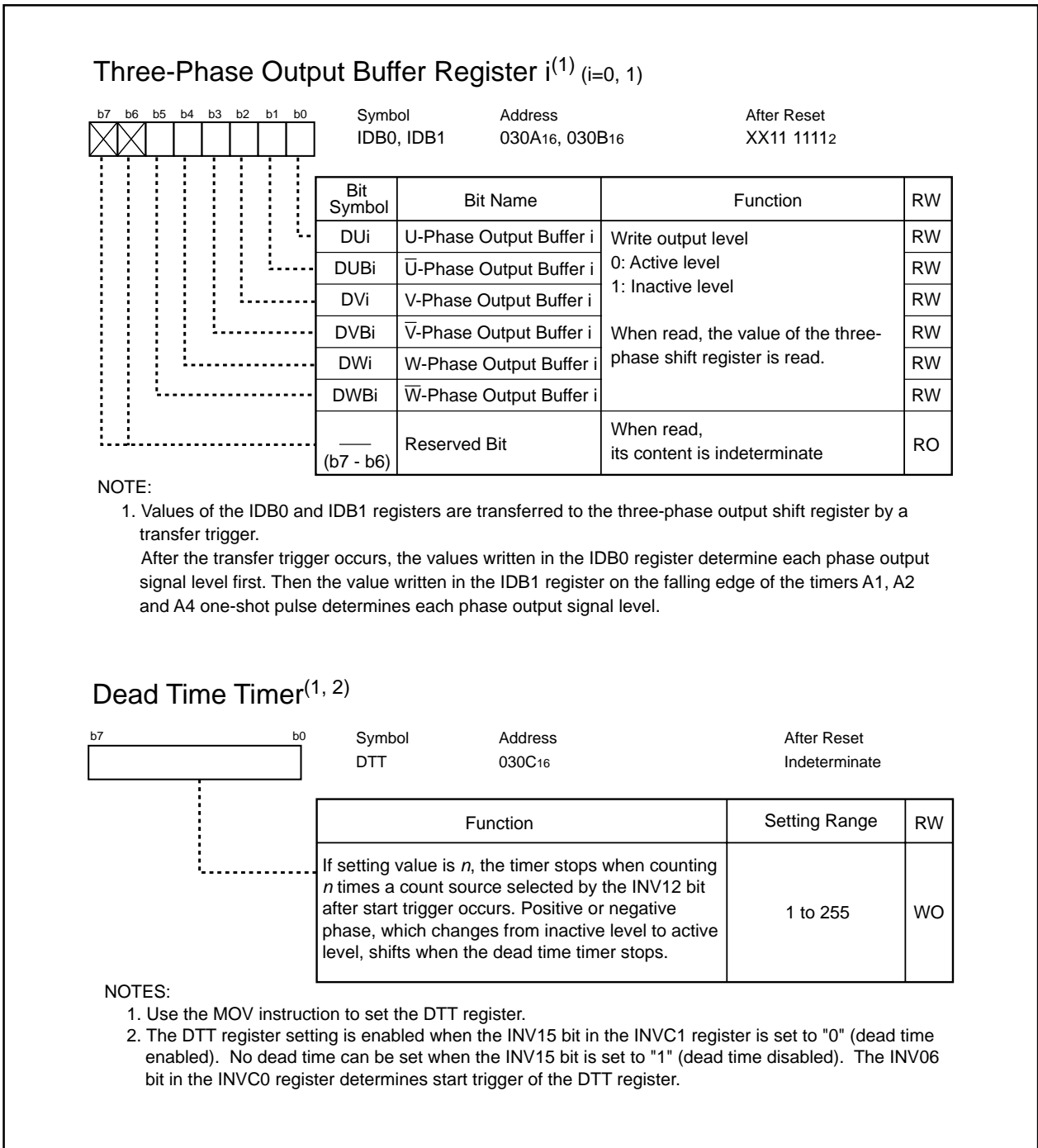
Figure 15.1 Three-Phase Motor Control Timer Functions Block Diagram



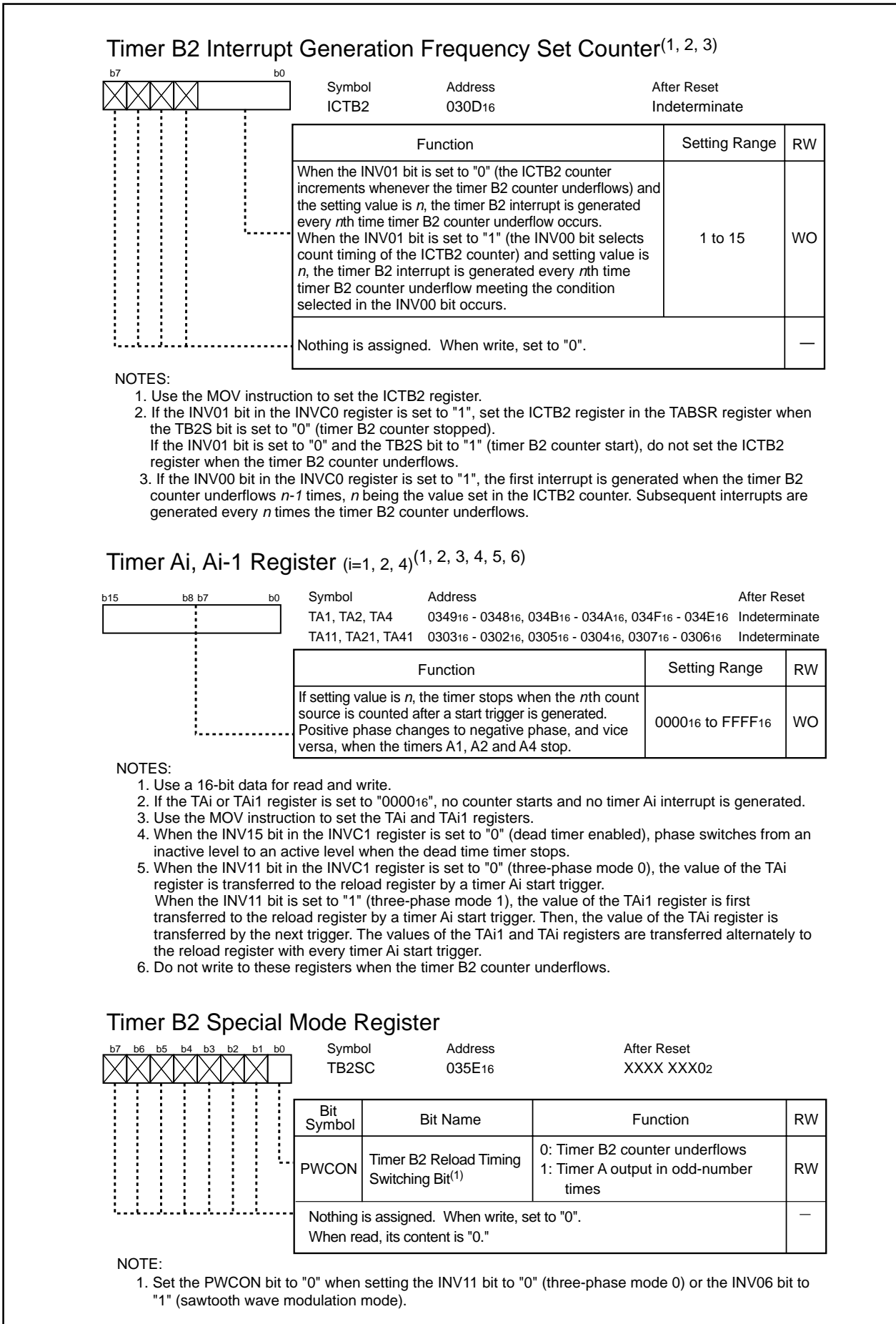
**Figure 15.2 INVC0 Register**



**Figure 15.3 INVC1 Register**



**Figure 15.4 IDB0 and IDB1 registers, DTT Register**



**Figure 15.5 ICTB2 Register, TA1, TA2, TA4, TA11, TA21 and TA41 Registers, TB2SC Register**

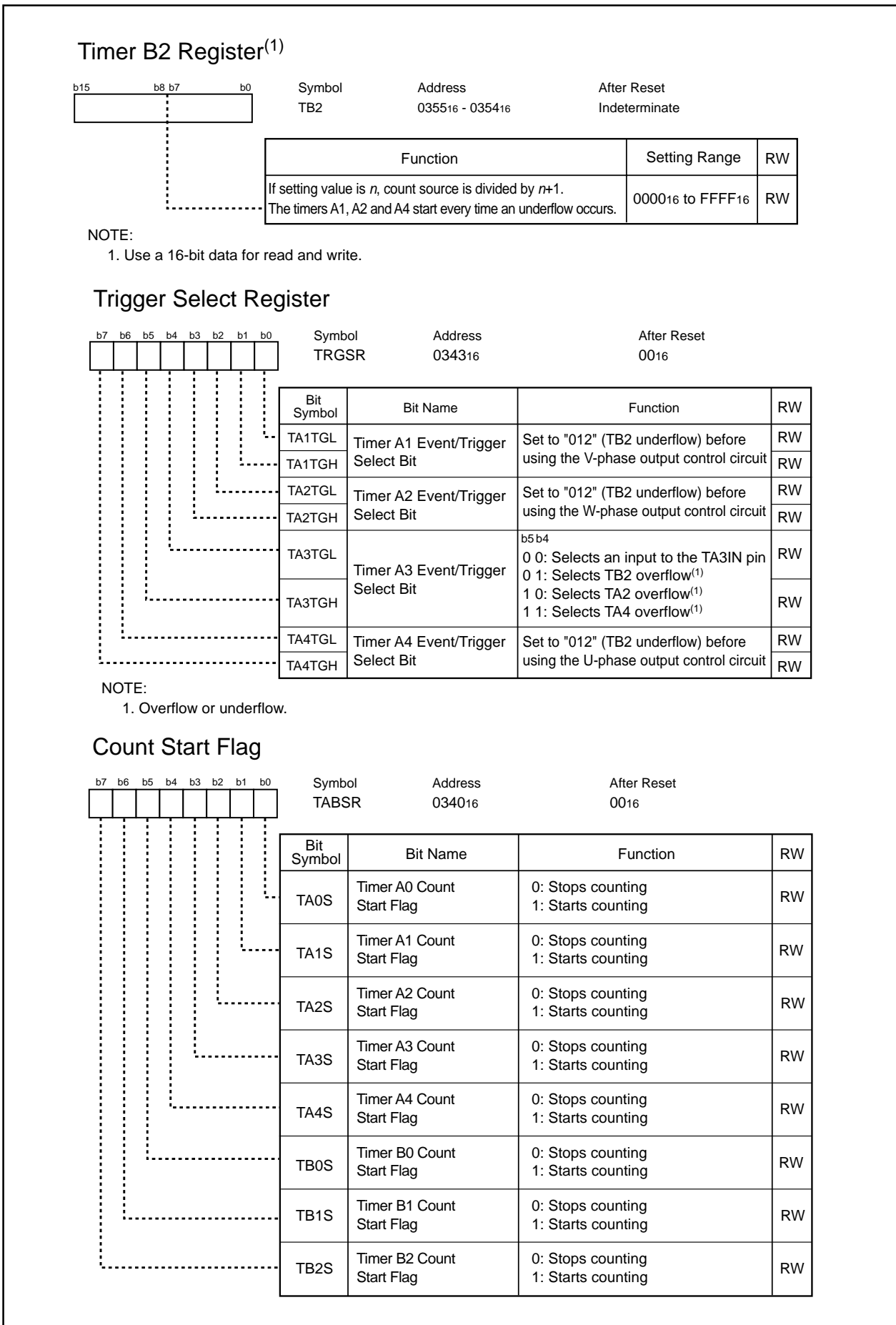


Figure 15.6 TB2, TRGSR and TABSR Registers

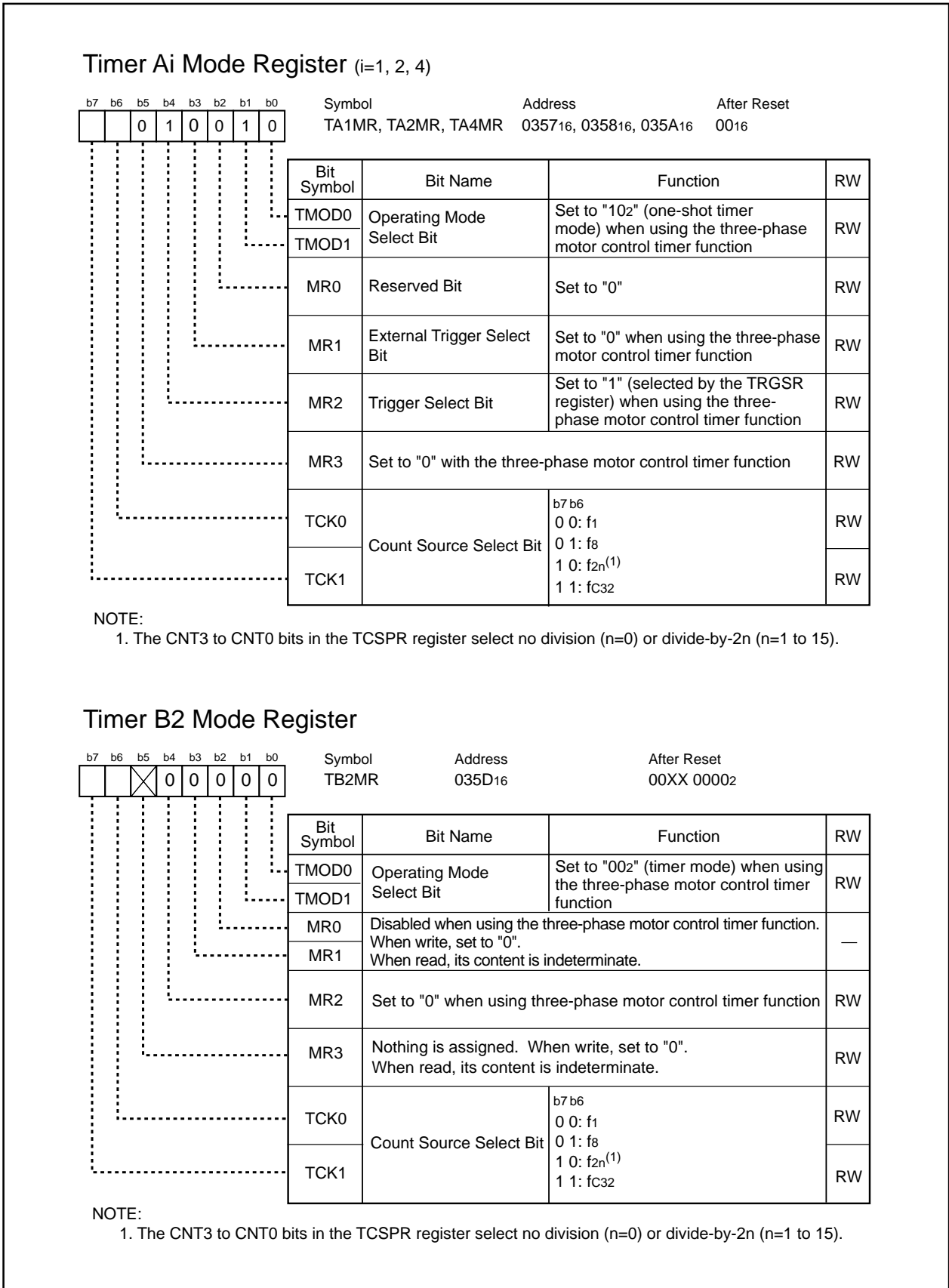


Figure 15.7 TA1MR, TA2MR and TA4MR Registers, TB2MR Register

The three-phase motor control timer function is available by setting the INV02 bit in the INVC0 register to "1". The timer B2 is used for carrier wave control and the timers A1, A2, A4 for three-phase PWM output (U,  $\bar{U}$ , V,  $\bar{V}$ , W,  $\bar{W}$ ) control. An exclusive dead time timer controls dead time. Figure 15.8 shows an example of the triangular modulation waveform. Figure 15.9 shows an example of the sawtooth modulation waveform.

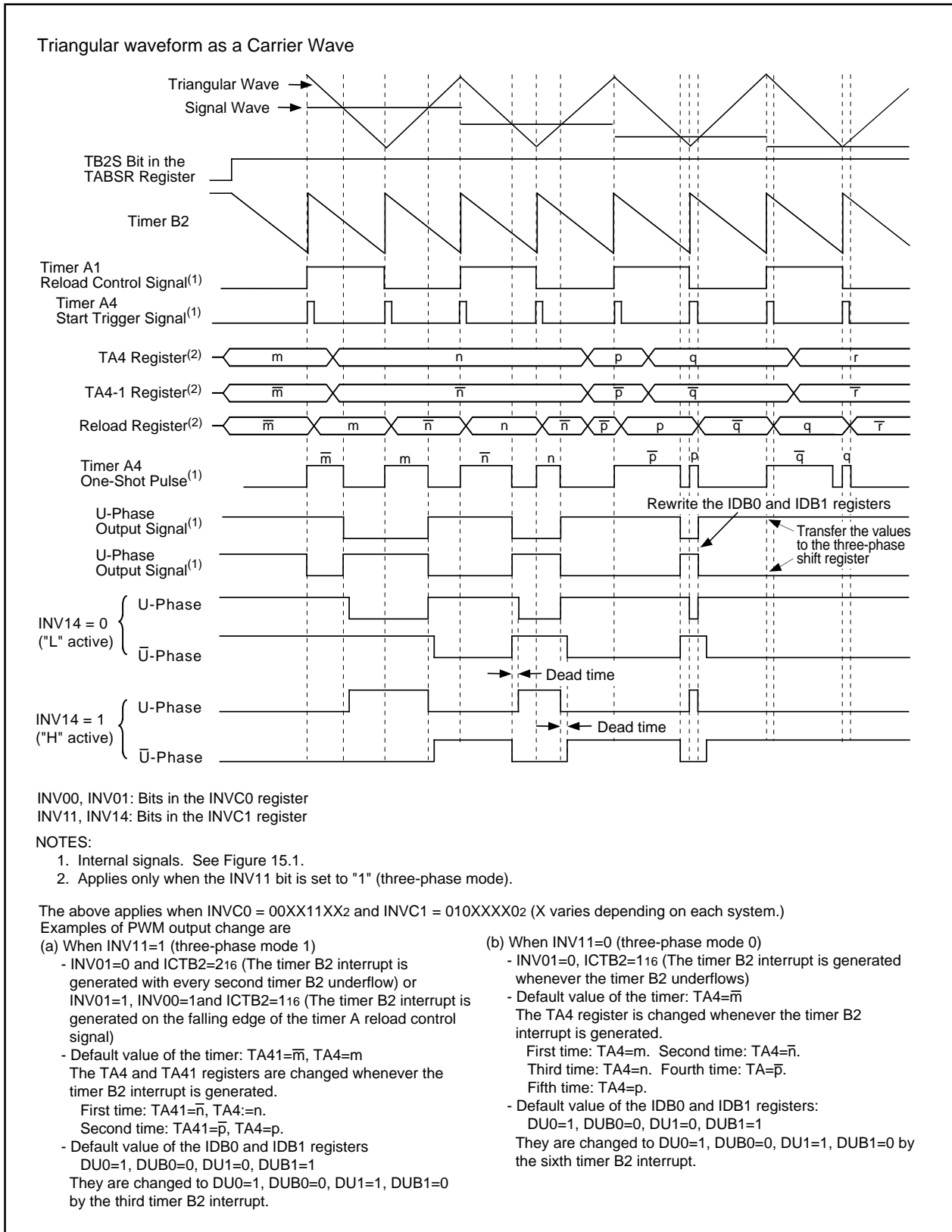
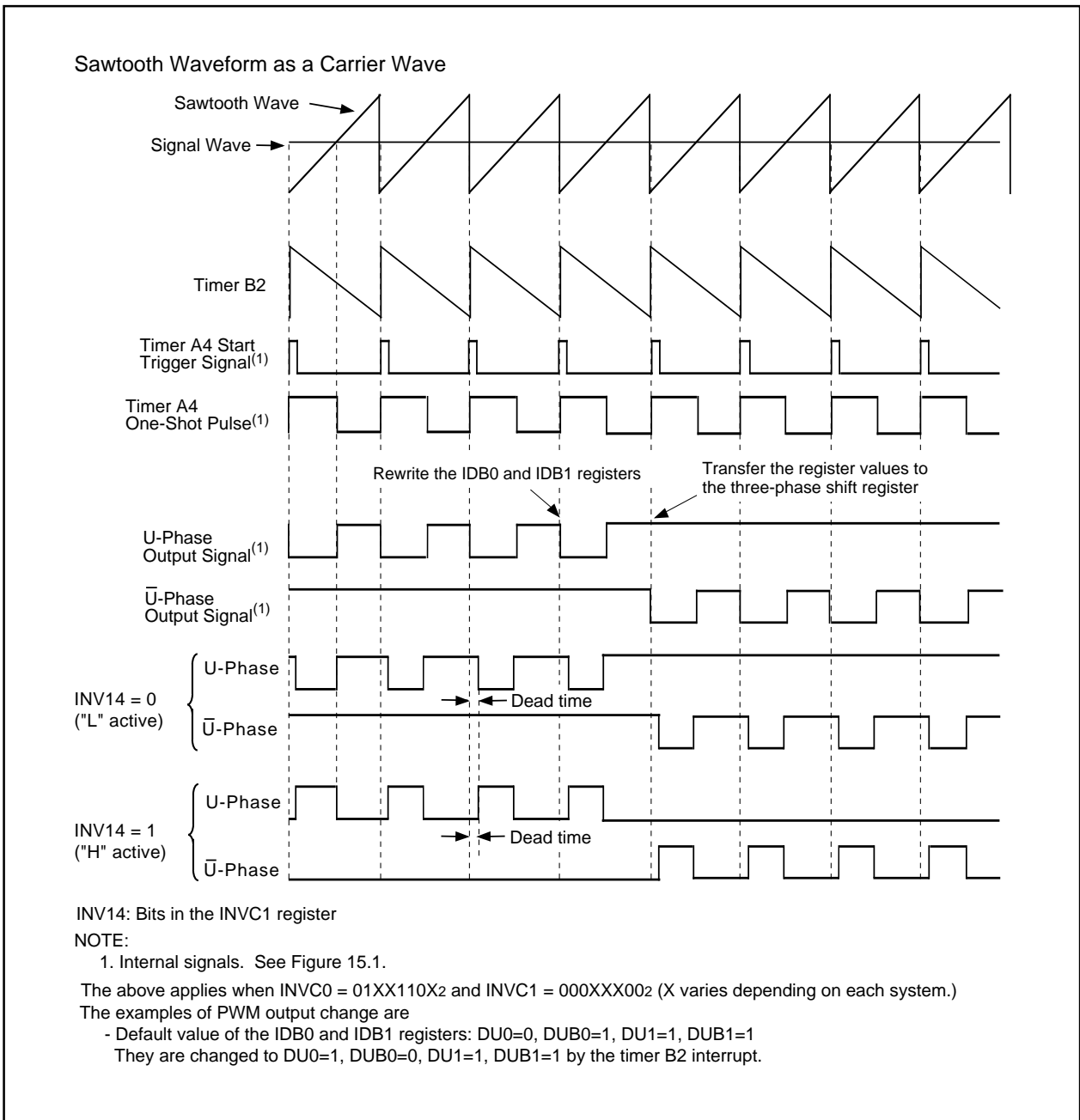


Figure 15.8 Triangular Wave Modulation Operation





**Figure 15.9 Sawtooth Wave Modulation Operation**

## 16. Serial I/O

Serial I/O consists of five channels (UART0 to UART4).

Each UART<sub>i</sub> (i=0 to 4) has an exclusive timer to generate the transfer clock and operates independently.

Figure 16.1 shows a UART<sub>i</sub> block diagram.

UART<sub>i</sub> supports 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 (Clock-divided synchronous function, GCI mode)
- Special mode 4 (Bus conflict detect function, IE mode)
- Special mode 5 (SIM mode)

Figures 16.2 to 16.9 show registers associated with UART<sub>i</sub>.

Refer to the tables listing each mode for register and pin settings.

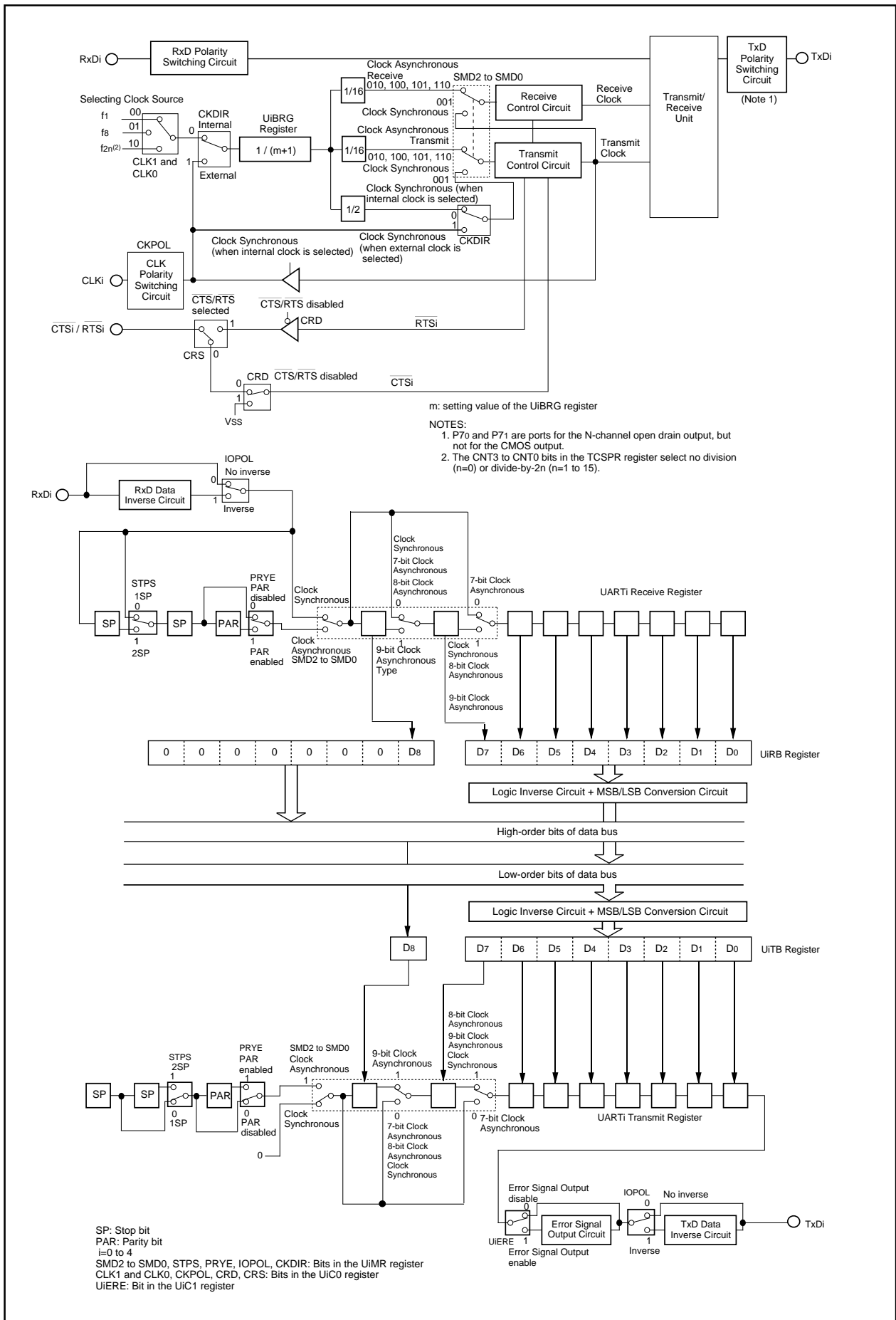


Figure 16.1 UARTi Block Diagram

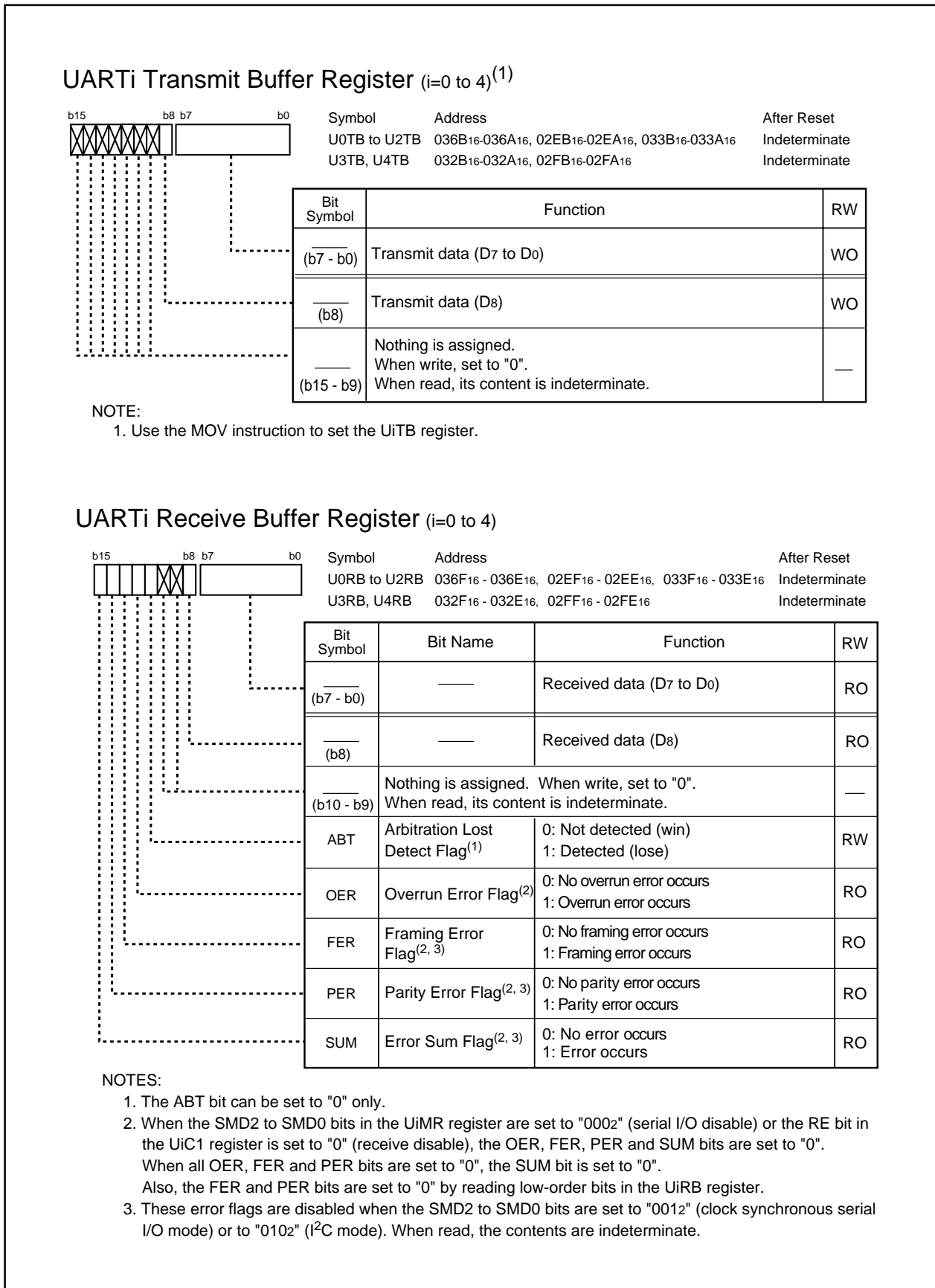


Figure 16.2 U0TB to U4TB Registers and U0RB to U4RB Registers

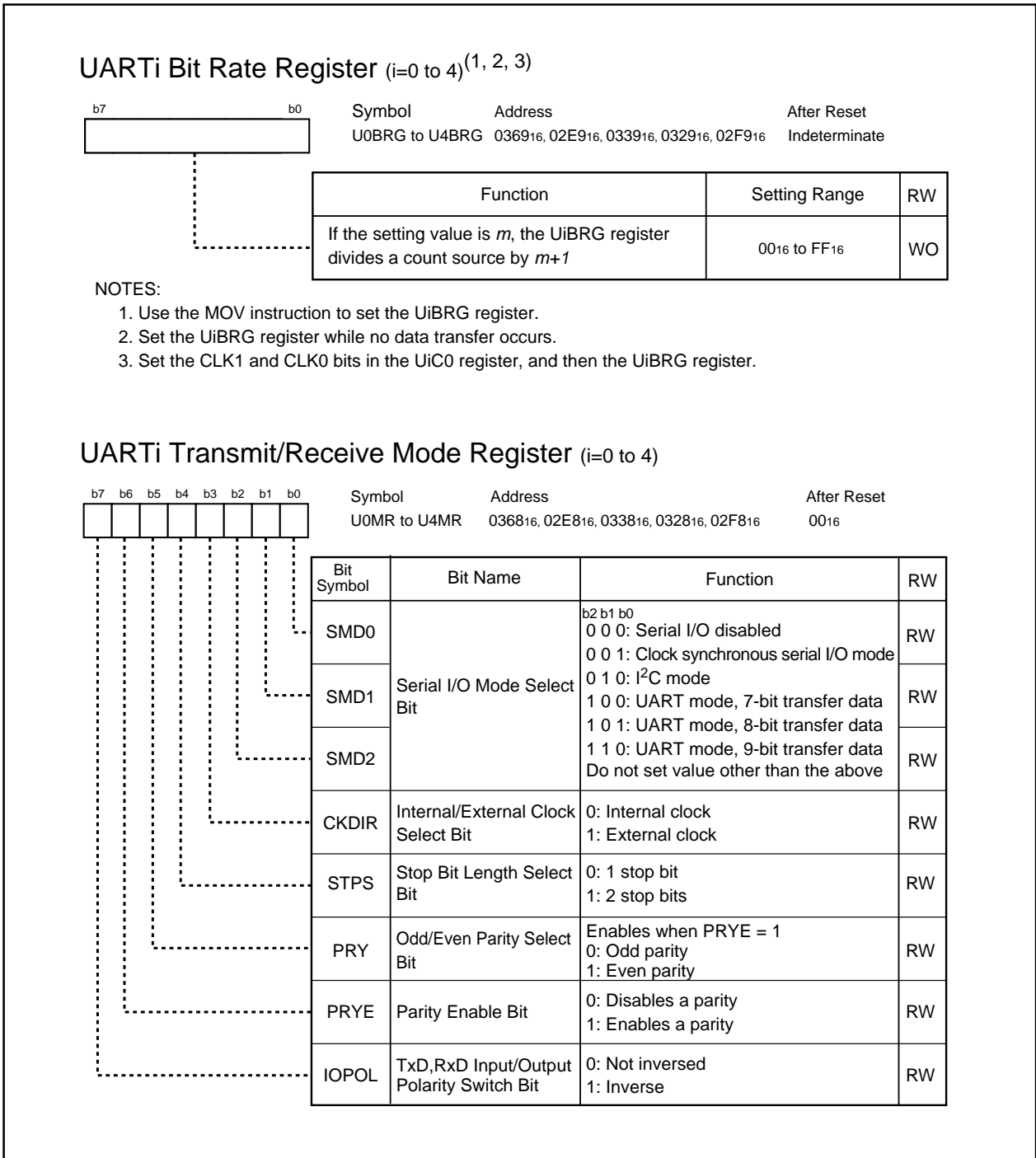
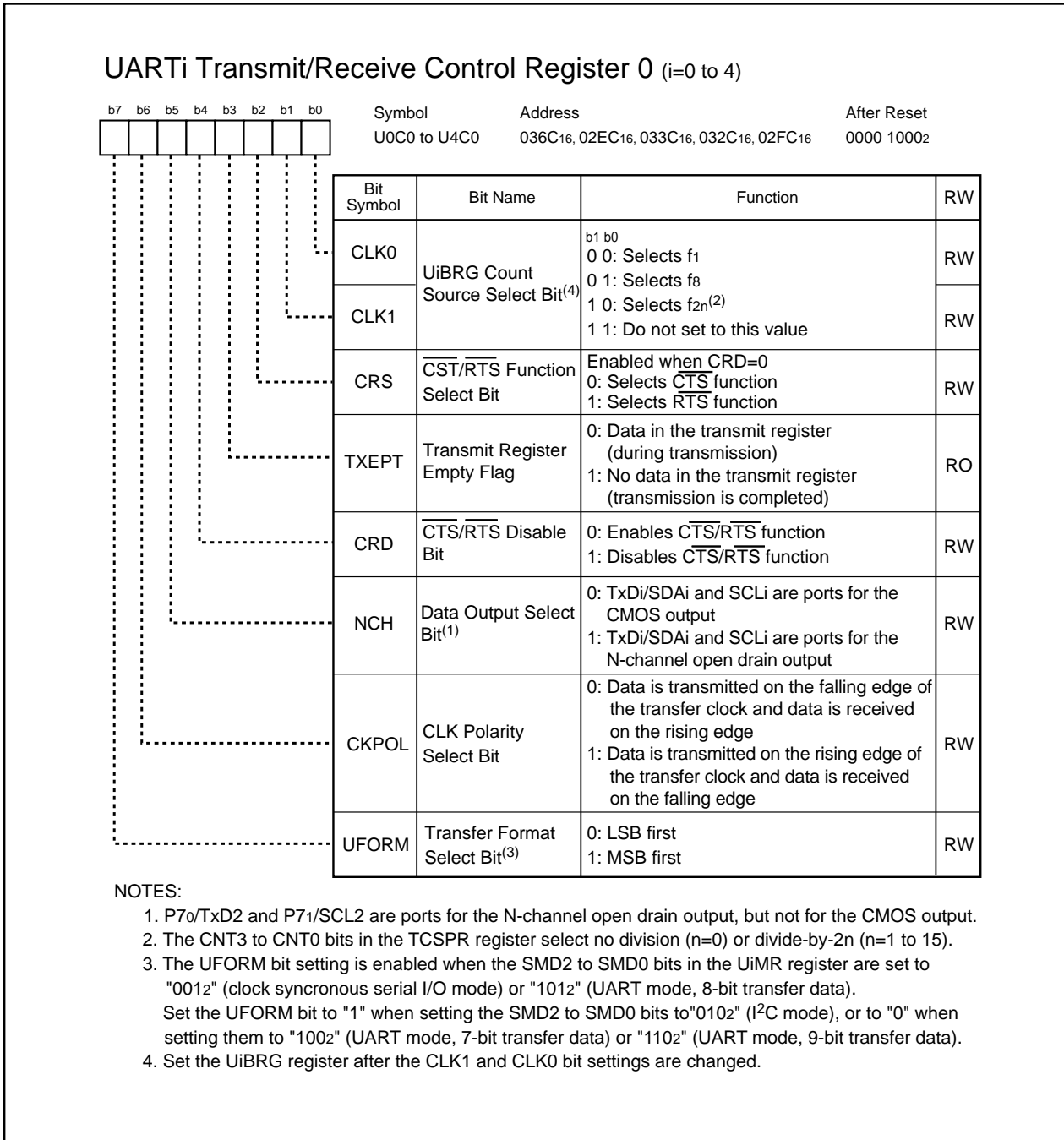
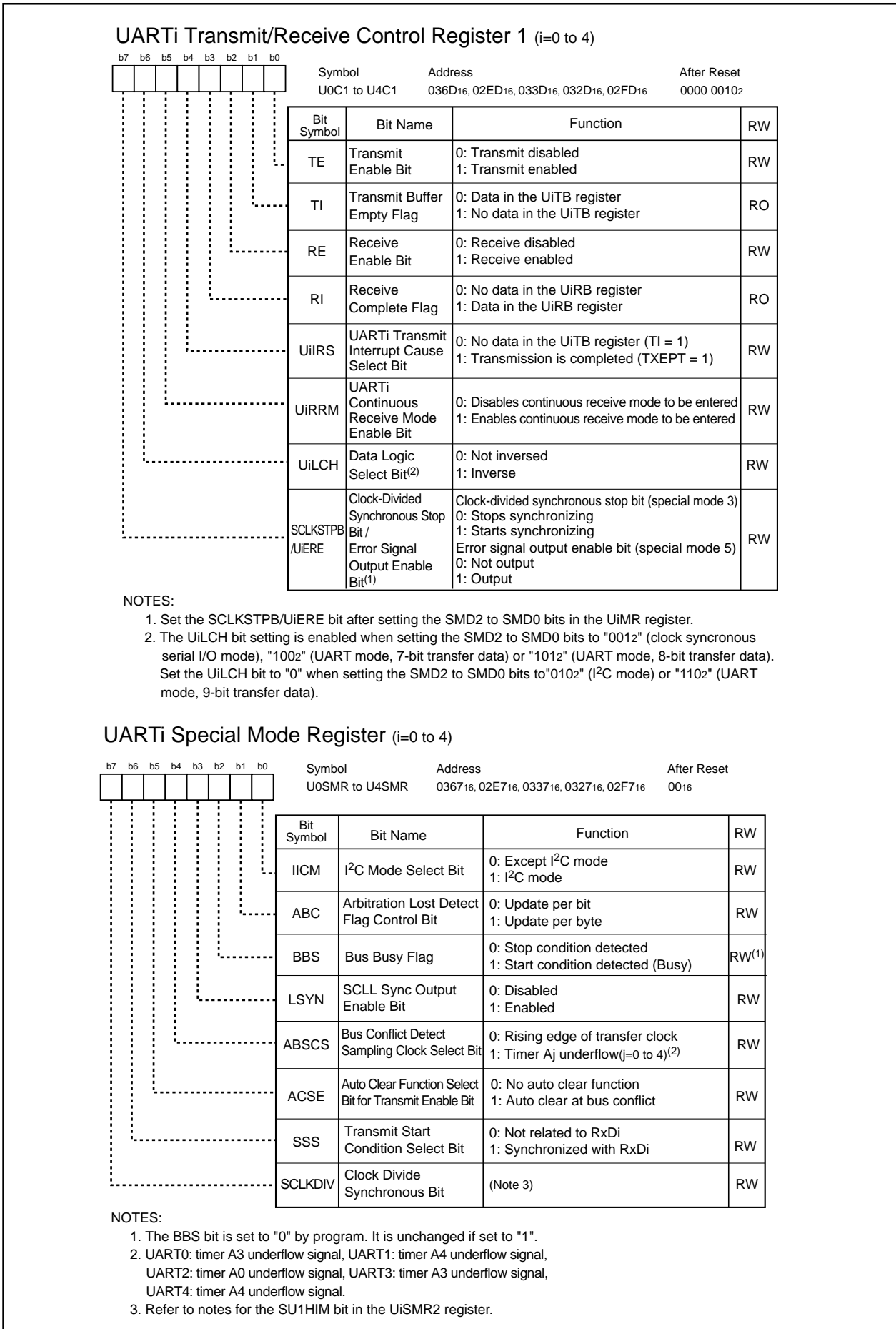


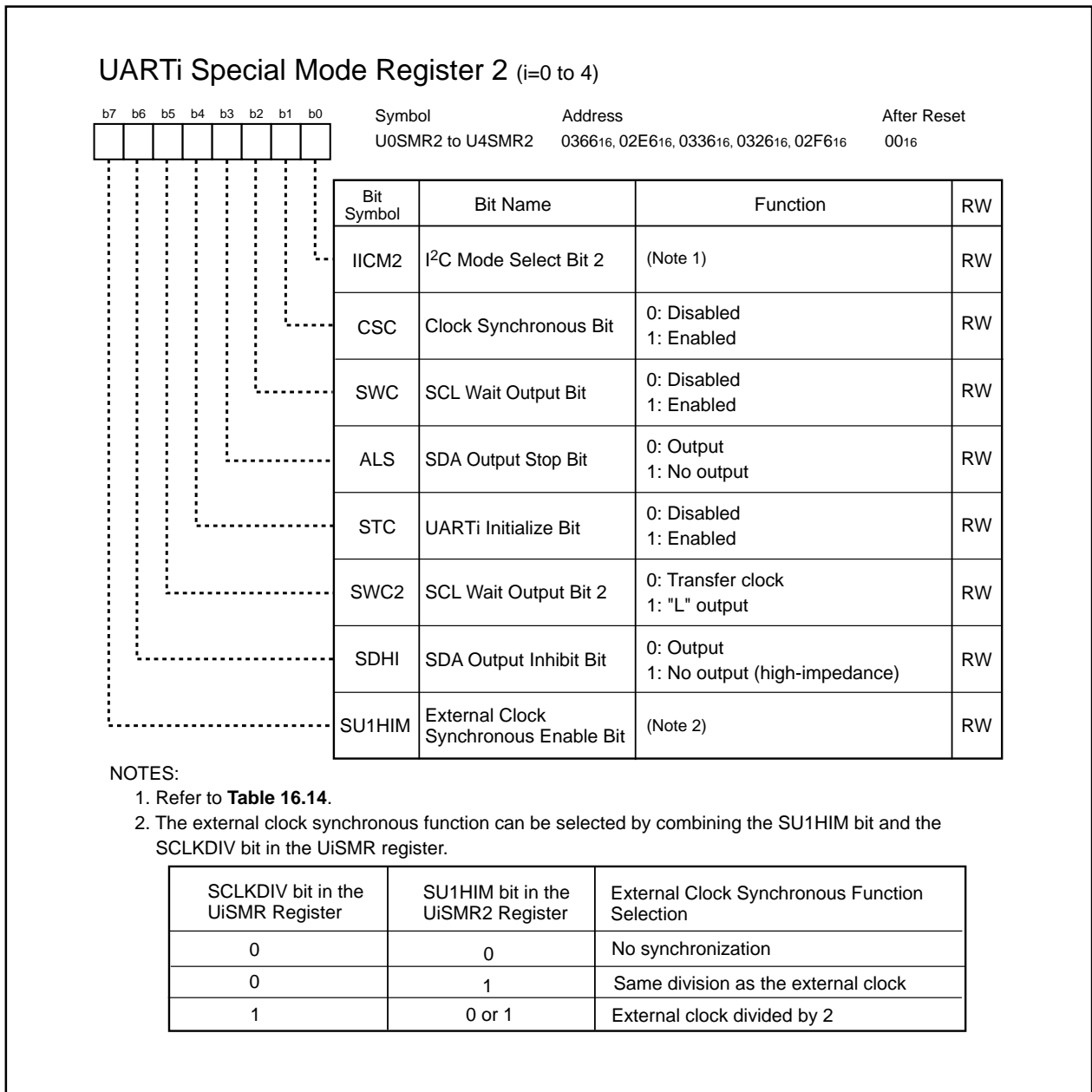
Figure 16.3 U0BRG to U4BRG Registers and U0MR to U4MR Registers



**Figure 16.4 U0C0 to U4C0 Registers**

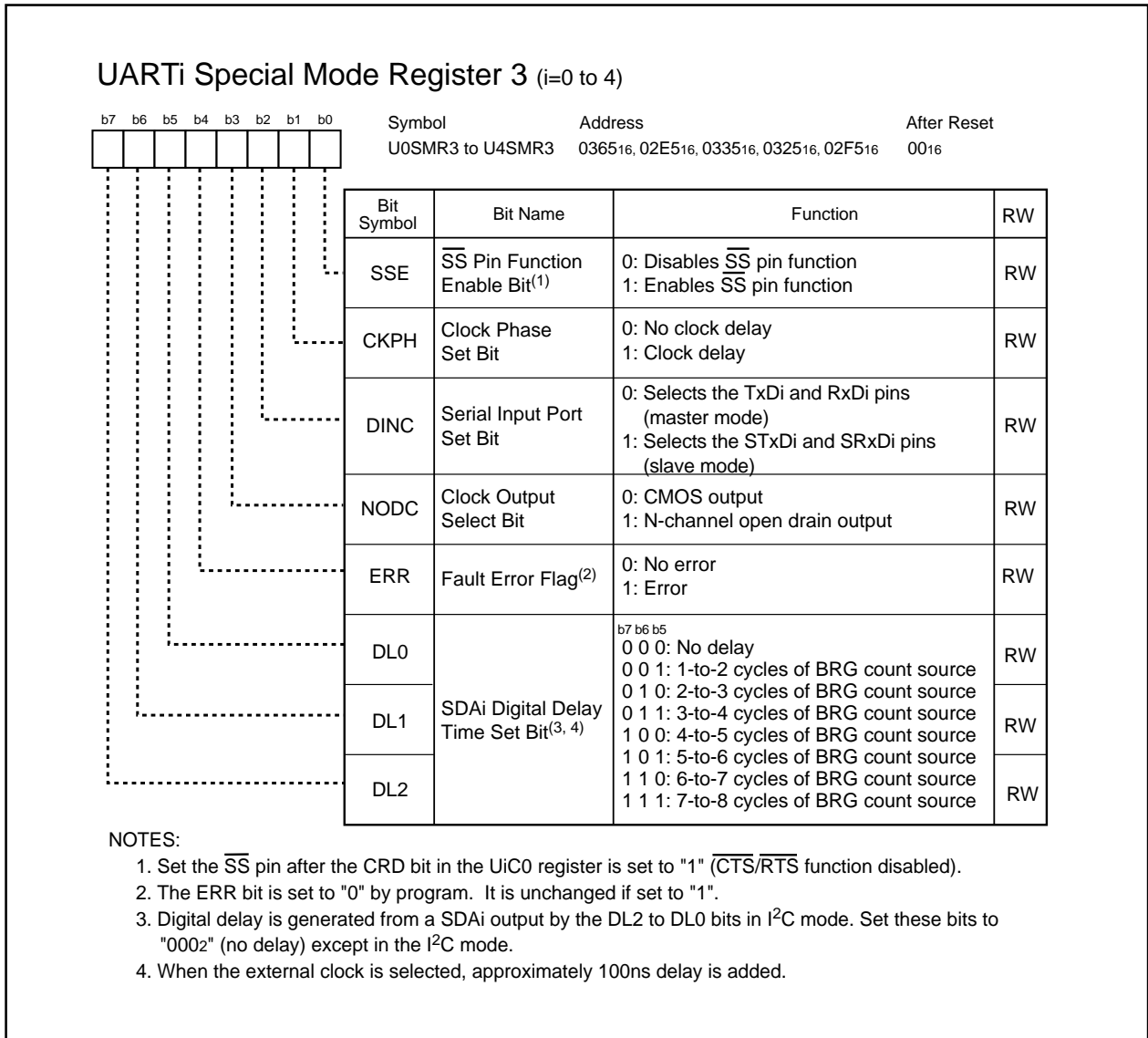


**Figure 16.5 U0C1 to U4C1 Registers and U0SMR to U4SMR Registers**

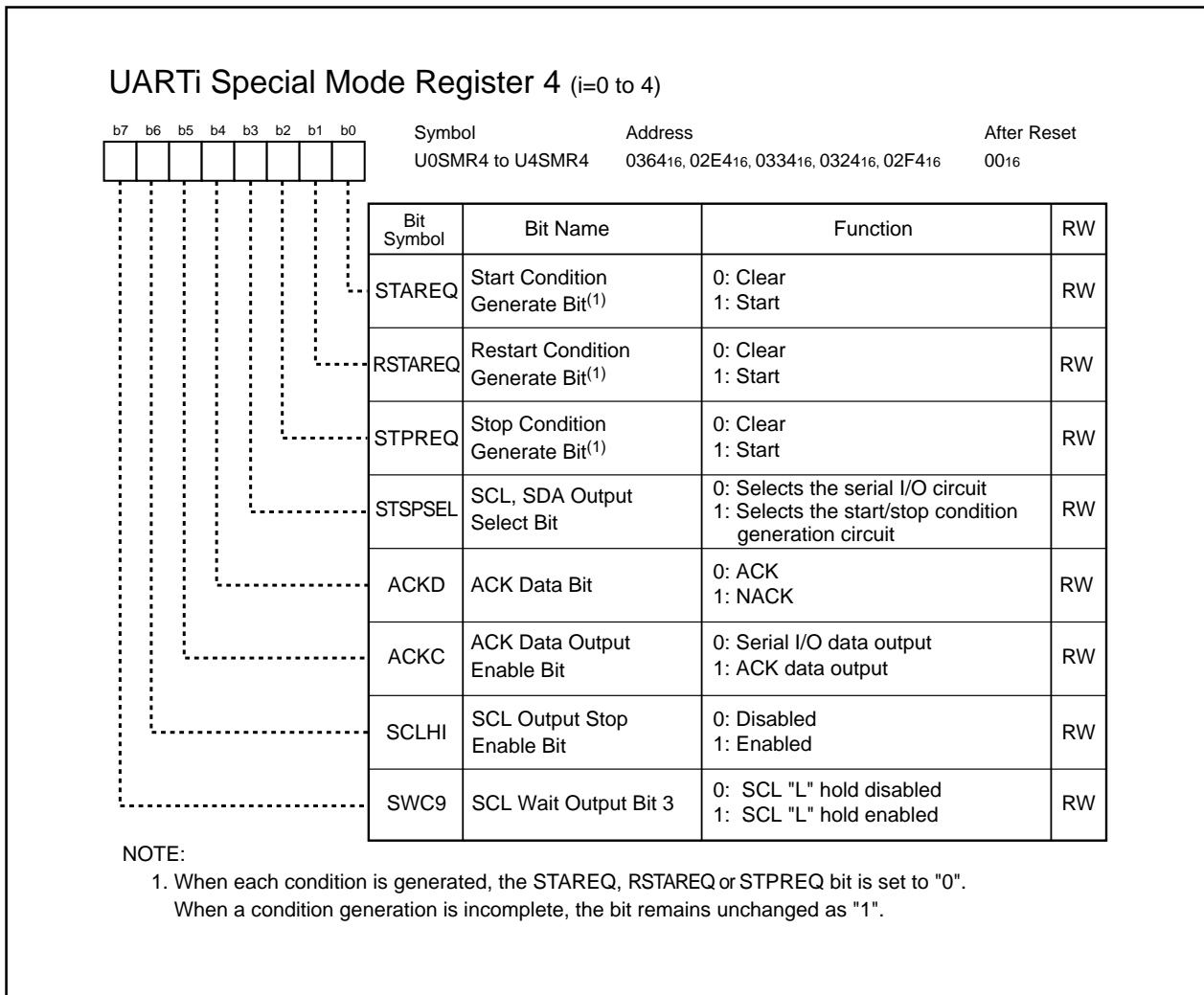


**Figure 16.6 U0SMR2 to U4SMR2 Registers**

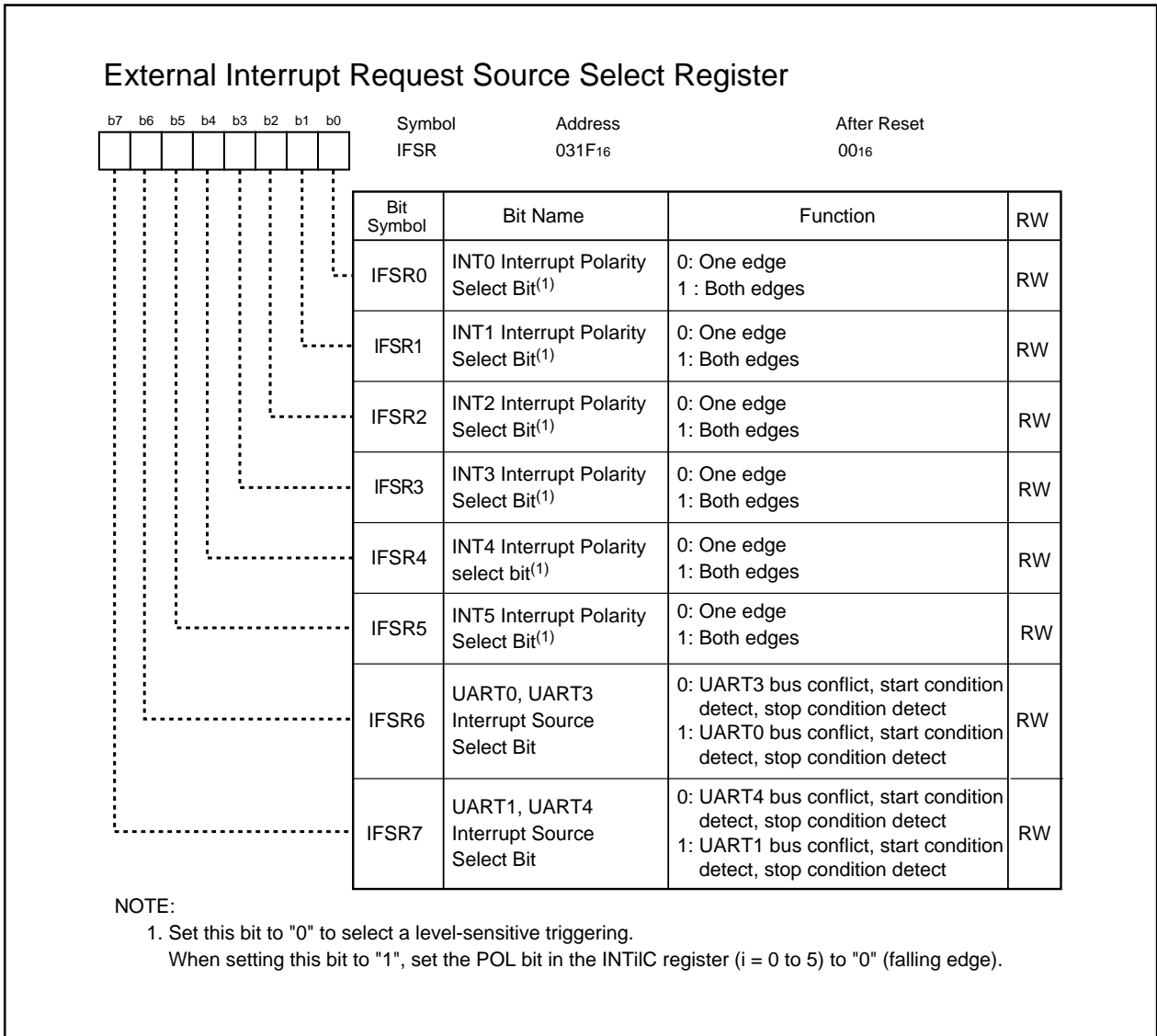




**Figure 16.7 U0SMR3 to U4SMR3 Registers**



**Figure 16.8 U0SMR4 to U4SMR4 Registers**



**Figure 16.9 IFSR Register**

## 16.1 Clock Synchronous Serial I/O Mode

In clock synchronous serial I/O mode, data is transmitted and received with the transfer clock. Table 16.1 lists specifications of clock synchronous serial I/O mode. Table 16.2 lists register settings. Tables 16.3 to 16.5 list pin settings. When UART<sub>i</sub> (i=0 to 4) operating mode is selected, the TxD<sub>i</sub> pin outputs a high-level ("H") signal before transfer starts (the TxD<sub>i</sub> pin is in a high-impedance state when the N-channel open drain output is selected). Figure 16.10 shows transmit and receive timings in clock synchronous serial I/O mode.

**Table 16.1 Clock Synchronous Serial I/O Mode Specifications**

Item	Specification
Transfer Data Format	Transfer data : 8 bits long
Transfer Clock	<ul style="list-style-type: none"> <li>The CKDIR bit in the UiMR register (i=0 to 4) is set to "0" (internal clock selected):  <math display="block">\frac{f_j}{2^{(m+1)}} \quad f_j=f_1, f_8, f_{2^n}^{(1)} \quad m:\text{setting value of the UiBRG register, } 00_{16} \text{ to } FF_{16}</math> </li> <li>The CKDIR bit is set to "1" (external clock selected) : an input from the CLK<sub>i</sub> pin</li> </ul>
Transmit/Receive Control	Selected from the $\overline{\text{CTS}}$ function, RTS function or CTS/RTS function disabled
Transmit Start Condition	To start transmitting, the following requirements must be met <sup>(2)</sup> : <ul style="list-style-type: none"> <li>- Set the TE bit in the UiC1 register to "1" (transmit enabled)</li> <li>- Set the TI bit in the UiC1 register to "0" (data in the UiTB register)</li> <li>- Apply a low-level ("L") signal to the <math>\overline{\text{CTS}}_i</math> pin when the <math>\overline{\text{CTS}}</math> function is selected</li> </ul>
Receive Start Condition	To start receiving, the following requirements must be met <sup>(2)</sup> : <ul style="list-style-type: none"> <li>- Set the RE bit in the UiC1 register to "1" (receive enabled)</li> <li>- Set the TE bit to "1" (transmit enabled)</li> <li>- Set the TI bit to "0" (data in the UiTB register)</li> </ul>
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• While transmitting, the following conditions can be selected: <ul style="list-style-type: none"> <li>- The UiIRS bit in the UiC1 register is set to "0" (no data in the transmit buffer): when data is transferred from the UiTB register to the UART<sub>i</sub> transmit register (transfer started)</li> <li>- The UiIRS bit is set to "1" (transmission completed): when a data transfer from the UART<sub>i</sub> transmit register is completed</li> </ul> </li> <li>• While receiving When data is transferred from the UART<sub>i</sub> receive register to the UiRB register (reception completed)</li> </ul>
Error Detect	Overrun error <sup>(3)</sup> This error occurs when the seventh bit of the next received data is read before reading the UiRB register
Selectable Function	<ul style="list-style-type: none"> <li>• CLK polarity Selectable from the rising edge or falling edge of the transfer clock at transferred data output or input timing</li> <li>• LSB first or MSB first Selectable from data transmission or reception in either bit 0 or in bit 7</li> <li>• Continuous receive mode Data can be received simultaneously by reading the UiRB register</li> <li>• Serial data logic inverse This function inverses transmitted/received data logically</li> </ul>

NOTES:

- The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2<sup>n</sup> (n=1 to 15).
- To start transmission/reception when selecting the external clock, these conditions must be met after the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and data is received on the rising edge) and the CLK<sub>i</sub> pin is held "H", or when the CKPOL bit is set to "1" (data is transmitted on the rising edge of the transfer clock and data is received on the falling edge) and the CLK<sub>i</sub> pin is held "L".
- If an overrun error occurs, the UiRB register is indeterminate. The IR bit setting in the SiRIC register does not change to "1" (interrupt requested).

**Table 16.2 Register Settings in Clock Synchronous Serial I/O Mode**

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data can be read
	OER	Overrun error flag
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "0012"
	CKDIR	Select the internal clock or external clock
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Select $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ when using either
	TXEPT	Transmit register empty flag
	CRD	Enables or disables the CTS or RTS function
	NCH	Select output format of the TxDi pin
	CKPOL	Select transmit clock polarity
	UFORM	Select either LSB first or MSB first
UiC1	TE	Set to "1" to enable data transmission and reception
	TI	Transmit buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM	Set to "1" when using continuous receive mode
	UiLCH	Set to "1" when using data logic inverse
	SCLKSTPB	Set to "0"
UiSMR	7 to 0	Set to "0016"
UiSMR2	7 to 0	Set to "0016"
UiSMR3	2 to 0	Set to "0002"
	NODC	Select clock output format
	7 to 4	Set to "00002"
UiSMR4	7 to 0	Set to "0016"

i=0 to 4

**Table 16.3 Pin Settings in Clock Synchronous Serial I/O Mode (1)**

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P60	CTS0 input	PS0_0=0	-	PD6_0=0
	RTS0 output	PS0_0=1	-	-
P61	CLK0 input	PS0_1=0	-	PD6_1=0
	CLK0 output	PS0_1=1	-	-
P62	RxD0 input	PS0_2=0	-	PD6_2=0
P63	TxD0 output	PS0_3=1	-	-
P64	CTS1 input	PS0_4=0	-	PD6_4=0
	RTS1 output	PS0_4=1	PSL0_4=0	-
P65	CLK1 input	PS0_5=0	-	PD6_5=0
	CLK1 output	PS0_5=1	-	-
P66	RxD1 input	PS0_6=0	-	PD6_6=0
P67	TxD1 output	PS0_7=1	-	-

**Table 16.4 Pin Settings (2)**

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70 <sup>(1)</sup>	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	-
P71 <sup>(1)</sup>	RxD2 input	PS1_1=0	-	-	PD7_1=0
P72	CLK2 input	PS1_2=0	-	-	PD7_2=0
	CLK2 output	PS1_2=1	PSL1_2=0	PSC_2=0	-
P73	CTS2 input	PS1_3=0	-	-	PD7_3=0
	RTS2 output	PS1_3=1	PSL1_3=0	PSC_3=0	-

NOTE:

1. P70 and P71 are ports for the N-channel open drain output.

**Table 16.5 Pin Settings (3)**

Port	Function	Setting			
		PS3 Register <sup>(1)</sup>	PSL3 Register	PSC3 Register	PD9 Register <sup>(1)</sup>
P90	CLK3 input	PS3_0=0	-	-	PD9_0=0
	CLK3 output	PS3_0=1	-	-	-
P91	RxD3 input	PS3_1=0	-	-	PD9_1=0
P92	TxD3 output	PS3_2=1	PSL3_2=0	-	-
P93	CTS3 input	PS3_3=0	PSL3_3=0	-	PD9_3=0
	RTS3 output	PS3_3=1	-	-	-
P94	CTS4 input	PS3_4=0	PSL3_4=0	-	PD9_4=0
	RTS4 output	PS3_4=1	-	-	-
P95	CLK4 input	PS3_5=0	PSL3_5=0	-	PD9_5=0
	CLK4 output	PS3_5=1	-	-	-
P96	TxD4 output	PS3_6=1	-	PSC3_6=0	-
P97	RxD4 input	PS3_7=0	-	-	PD9_7=0

NOTE:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

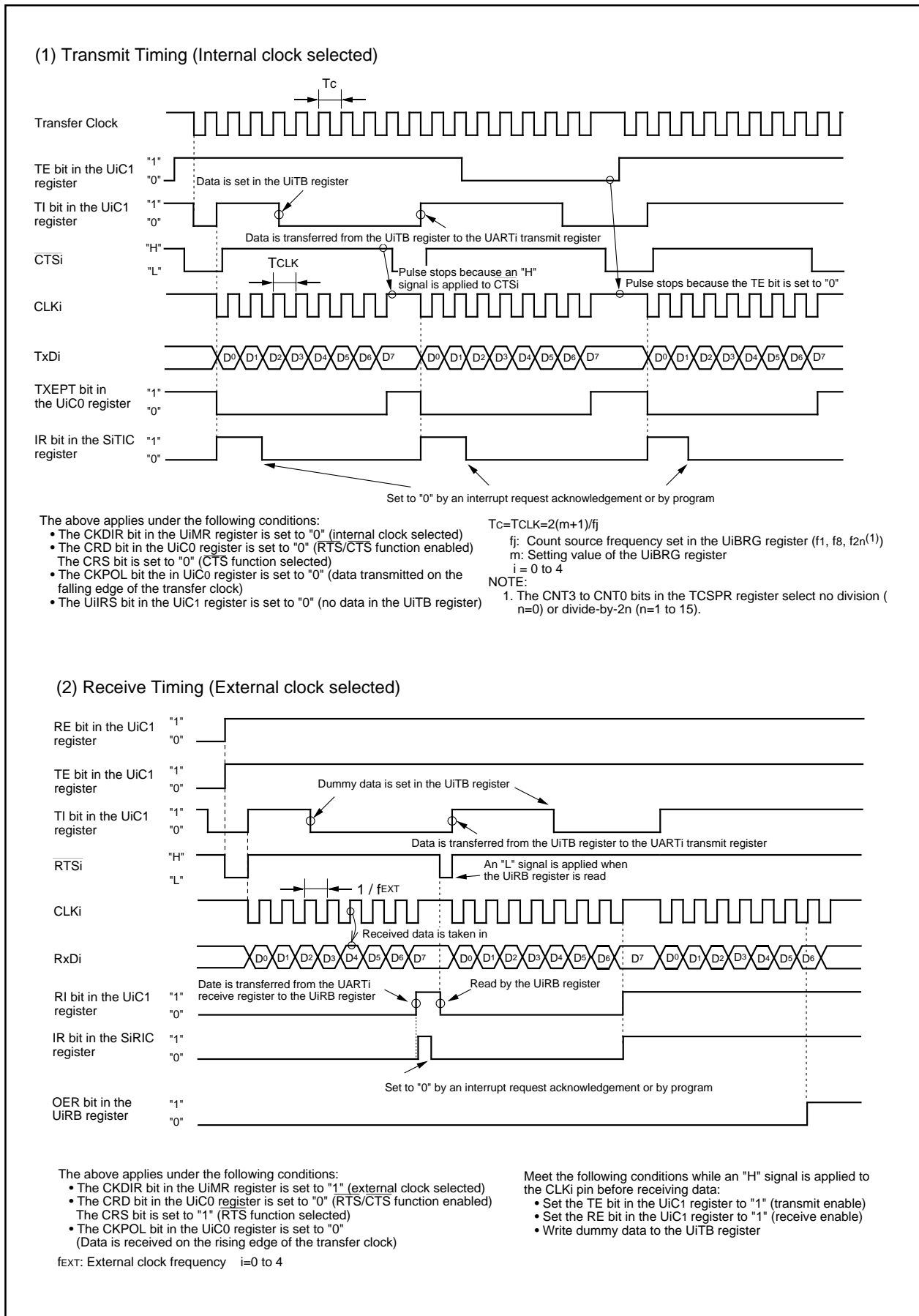


Figure 16.10 Transmit and Receive Operation

### 16.1.1 Selecting CLK Polarity Selecting

As shown in Figure 16.11, the CKPOL bit in the UiC0 register (i=0 to 4) determines the polarity of the transfer clock.

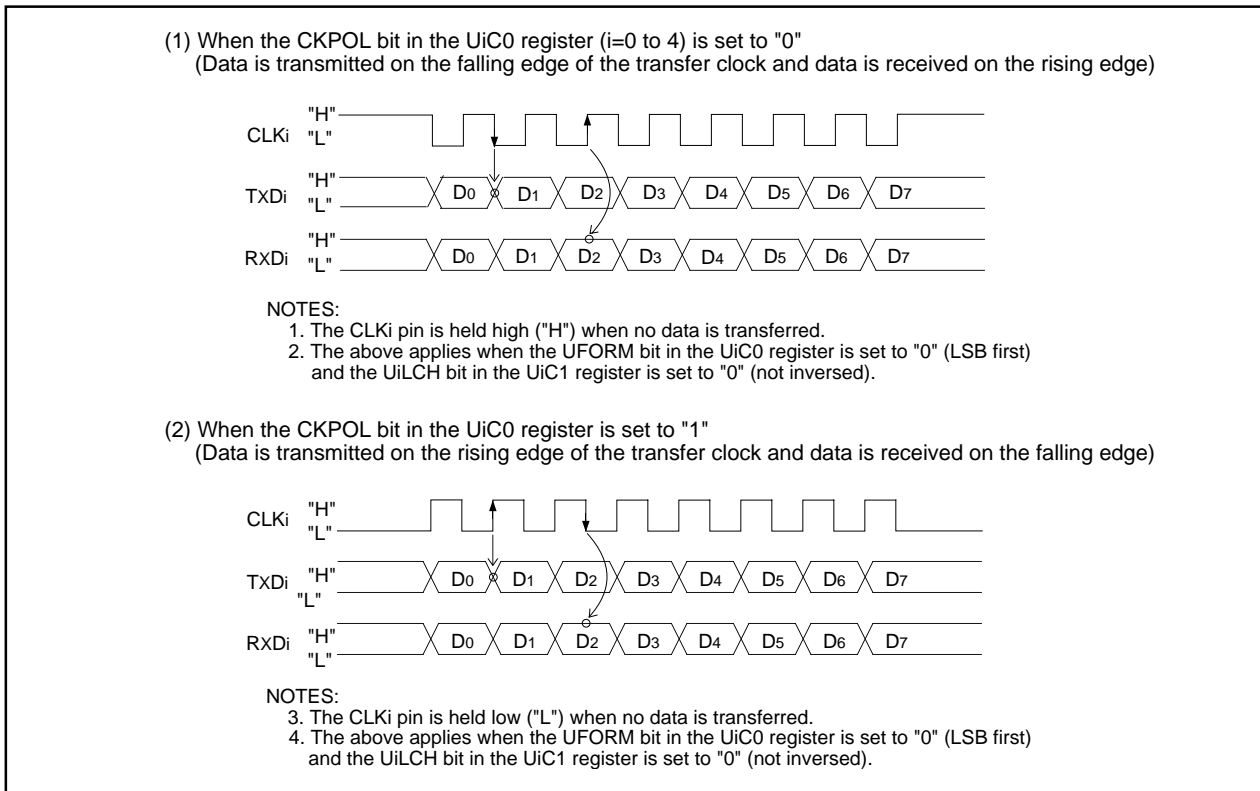


Figure 16.11 Transfer Clock Polarity

### 16.1.2 Selecting LSB First or MSB First

As shown in Figure 16.12, the UFORM bit in the UiC0 register (i=0 to 4) determines a data transfer format.

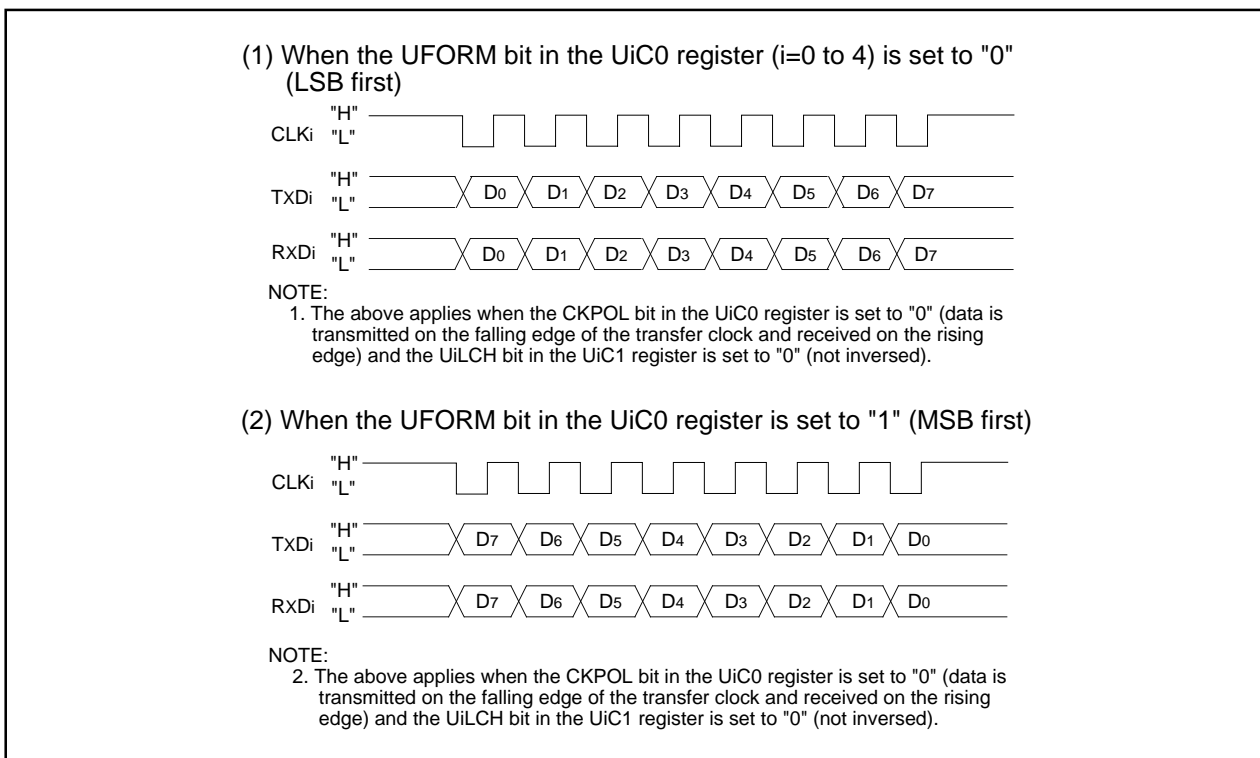


Figure 16.12 Transfer Format

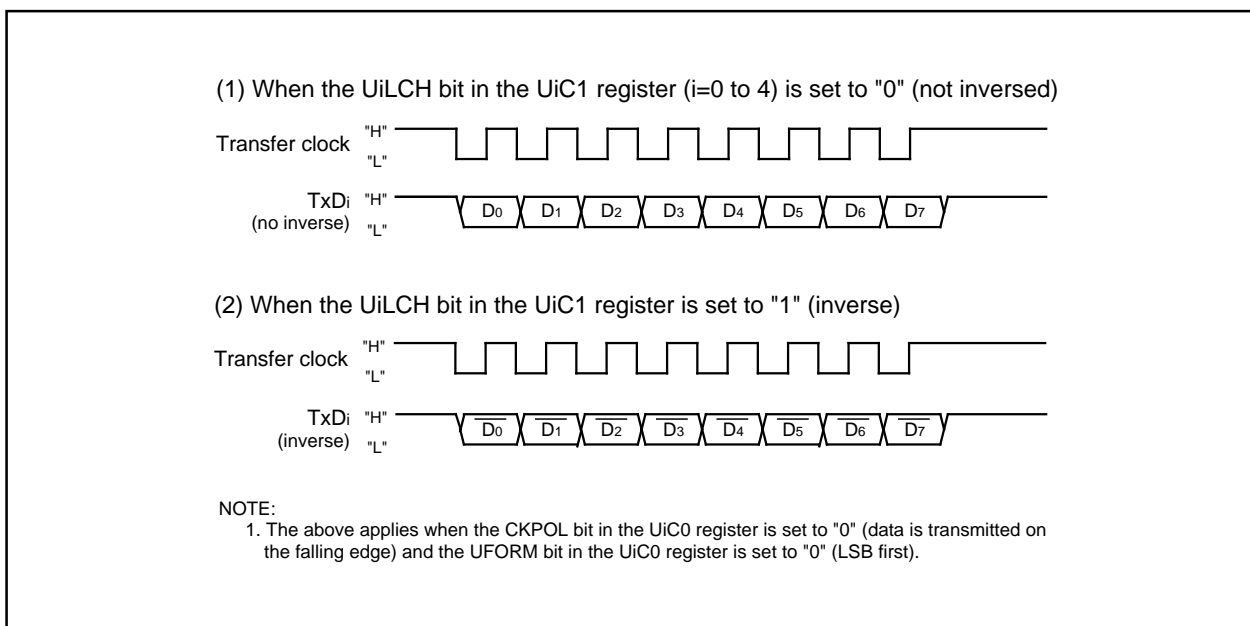


### 16.1.3 Continuous Receive Mode

When the UiRRM bit in the UiC1 register (i=0 to 4) is set to "1" (continuous receive mode), the TI bit is set to "0" (data in the UiTB register) by reading the UiRB register. When the UiRRM bit is set to "1", do not set dummy data in the UiTB register by program.

### 16.1.4 Serial Data Logic Inverse

When the UiLCH bit (i=0 to 4) in the UiC1 register is set to "1" (inverse), data logic written in the UiTB register is inverted when transmitted. The inverted receive data logic can be read by reading the UiRB register. Figure 16.13 shows a switching example of the serial data logic.



**Figure 16.13 Serial Data Logic Inverse**

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

In UART mode, data is transmitted and received after setting a desired bit rate and data transfer format. Table 16.6 lists specifications of UART mode.

**Table 16.6 UART Mode Specifications**

Item	Specification
Transfer Data Format	<ul style="list-style-type: none"> <li>• Character bit (transfer data) : selected from 7 bits, 8 bits, or 9 bits long</li> <li>• Start bit: 1 bit long</li> <li>• Parity bit: selected from odd, even, or none</li> <li>• Stop bit: selected from 1 bit or 2 bits long</li> </ul>
Transfer Clock	<ul style="list-style-type: none"> <li>• The CKDIR bit in the UiMR register is set to "0" (internal clock selected):  <math>f_j/16(m+1)</math> <math>f_j = f_1, f_8, f_{2n}^{(1)}</math> <math>m</math>: setting value of the UiBRG register, 0016 to FF16</li> <li>• The CKDIR bit is set to "1" (external clock selected):  <math>f_{EXT}/16(m+1)</math> <math>f_{EXT}</math>: clock applied to the CLKi pin</li> </ul>
Transmit/Receive Control	Select from CTS function, RTS function or CTS/RTS function disabled
Transmit Start Condition	To start transmitting, the following requirements must be met: <ul style="list-style-type: none"> <li>- Set the TE bit in the UiC1 register to "1" (transmit enabled)</li> <li>- Set the TI bit in the UiC1 register to "0" (data in the UiTB register)</li> <li>- Apply a low-level ("L") signal to the CTSi pin when the CTS function is selected</li> </ul>
Receive Start Condition	To start receiving, the following requirements must be met: <ul style="list-style-type: none"> <li>- Set the RE bit in the UiC1 register to "1" (receive enabled)</li> <li>- The start bit is detected</li> </ul>
Interrupt Request Generation Timing	<p>While transmitting, the following condition can be selected:</p> <ul style="list-style-type: none"> <li>- The UiIRS bit in the UiC1 register is set to "0" (no data in the UiTB register): when data is transferred from the UiTB register to the UARTi transmit register (transfer started)</li> <li>- The UiIRS bit is set to "1" (transmission completed): when data transmission from the UARTi transfer register is completed</li> </ul> <p>While receiving</p> <ul style="list-style-type: none"> <li>- when data is transferred from the UARTi receive register to the UiRB register (reception completed)</li> </ul>
Error Detect	<ul style="list-style-type: none"> <li>• Overrun error<sup>(2)</sup> This error occurs when the bit before the last stop bit of the next received data is read prior to reading the UiRB register (the first stop bit when selecting 2 stop bits)</li> <li>• Framing error This error occurs when the number of stop bits set is not detected</li> <li>• Parity error When parity is enabled, this error occurs when 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 an overrun, framing or parity errors occur</li> </ul>
Selectable Function	<ul style="list-style-type: none"> <li>• LSB first or MSB first Selectable from data transmission or reception in either bit 0 or in bit 7</li> <li>• Serial data logic inverse Logic values of data to be transmitted and received data are inversed. The start bit and stop bit are not inversed</li> <li>• TxD and RxD I/O polarity Inverse TxD pin output and RxD pin input are inversed. All I/O data levels are also inversed</li> </ul>

**NOTES:**

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
2. If an overrun error occurs, the UiRB register is indeterminate. The IR bit setting in the SiRIC register does not change to "1" (interrupt requested).

Table 16.7 lists register settings. Tables 16.8 to 16.10 list pin settings. When UART<sub>i</sub> (i=0 to 4) operating mode is selected, the TxDi pin outputs a high-level ("H") signal before transfer is started (the TxDi pin is in a high-impedance state when the N-channel open drain output is selected). Figure 16.14 shows an example of a transmit operation in UART mode. Figure 16.15 shows an example of a receive operation in UART mode.

**Table 16.7 Register Settings in UART Mode**

Register	Bit	Function
UiTB	8 to 0	Set transmit data <sup>(1)</sup>
UiRB	8 to 0	Received data can be read <sup>(1)</sup>
	OER, FER, PER, SUM	Error flags
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "1002" when transfer data is 7 bits long Set to "1012" when transfer data is 8 bits long Set to "1102" when transfer data is 9 bits long
	CKDIR	Select the internal clock or external clock
	STPS	Select stop bit length
	PRY, PRYE	Select parity enabled or disabled, odd or even
	IOPOL	Select TxD and RxD I/O polarity
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Select either $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ when using either
	TXEPT	Transfer register empty flag
	CRD	Select the $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ function enabled or disabled
	NCH	Select output format of the TxDi pin
	CKPOL	Set to "0"
	UFORM	Select the LSB first or MSB first when a transfer data is 8 bits long Set to "0" when transfer data is 7 bits or 9 bits long
UiC1	TE	Set to "1" to enable data transmission
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UART <sub>i</sub> transmit interrupt to be generated
	UiRRM	Set to "0"
	UiLCH	Select whether data logic is inversed or not inversed when a transfer data is 7 bits or 8 bits long. Set to "0" when transfer data is 9 bits long
	UiERE	Set to either "0" or "1"
UiSMR	7 to 0	Set to "0016"
UiSMR2	7 to 0	Set to "0016"
UiSMR3	7 to 0	Set to "0016"
UiSMR4	7 to 0	Set to "0016"

**NOTE:**

1. Use bits 0 to 6 when transfer data is 7 bits long, bits 0 to 7 when 8 bits long, bits 0 to 8 when 9 bits long.

**Table 16.8 Pin Settings in UART Mode (1)**

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P60	CTS0 input	PS0_0=0	–	PD6_0=0
	RTS0 output	PS0_0=1	–	–
P61	CLK0 input	PS0_1=0	–	PD6_1=0
P62	RxD0 input	PS0_2=0	–	PD6_2=0
P63	TxD0 output	PS0_3=1	–	–
P64	CTS1 input	PS0_4=0	–	PD6_4=0
	RTS1 output	PS0_4=1	PSL0_4=0	–
P65	CLK1 input	PS0_5=0	–	PD6_5=0
P66	RxD1 input	PS0_6=0	–	PD6_6=0
P67	TxD1 output	PS0_7=1	–	–

**Table 16.9 Pin Settings (2)**

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70 <sup>(1)</sup>	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
P71 <sup>(1)</sup>	RxD2 input	PS1_1=0	–	–	PD7_1=0
P72	CLK2 input	PS1_2=0	–	–	PD7_2=0
P73	CTS2 input	PS1_3=0	–	–	PD7_3=0
	RTS2 output	PS1_3=1	PSL1_3=0	PSC_3=0	–

NOTE:

1. P70 and P71 are ports for the N-channel open drain output.

**Table 16.10 Pin Settings (3)**

Port	Function	Setting			
		PS3 Register <sup>(1)</sup>	PSL3 Register	PSC3 Register	PD9 Register <sup>(1)</sup>
P90	CLK3 input	PS3_0=0	–	–	PD9_0=0
P91	RxD3 input	PS3_1=0	–	–	PD9_1=0
P92	TxD3 output	PS3_2=1	PSL3_2=0	–	–
P93	CTS3 input	PS3_3=0	PSL3_3=0	–	PD9_3=0
	RTS3 output	PS3_3=1	–	–	–
P94	CTS4 input	PS3_4=0	PSL3_4=0	–	PD9_4=0
	RTS4 output	PS3_4=1	–	–	–
P95	CLK4 input	PS3_5=0	PSL3_5=0	–	PD9_5=0
P96	TxD4 output	PS3_6=1	–	PSC3_6=0	–
P97	RxD4 input	PS3_7=0	–	–	PD9_7=0

NOTE:

1. Set the PD9 and PS3 registers set immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

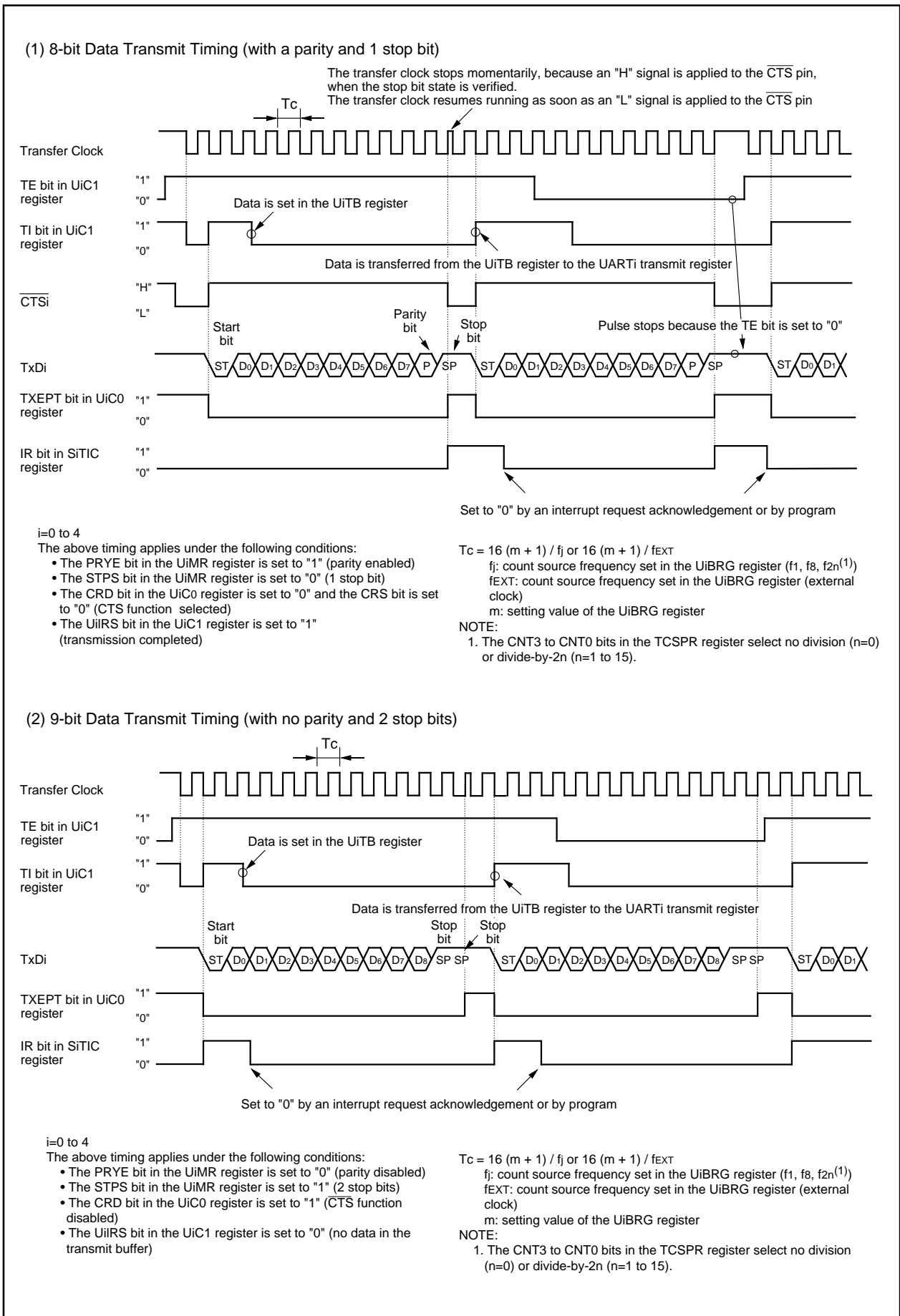


Figure 16.14 Transmit Operation

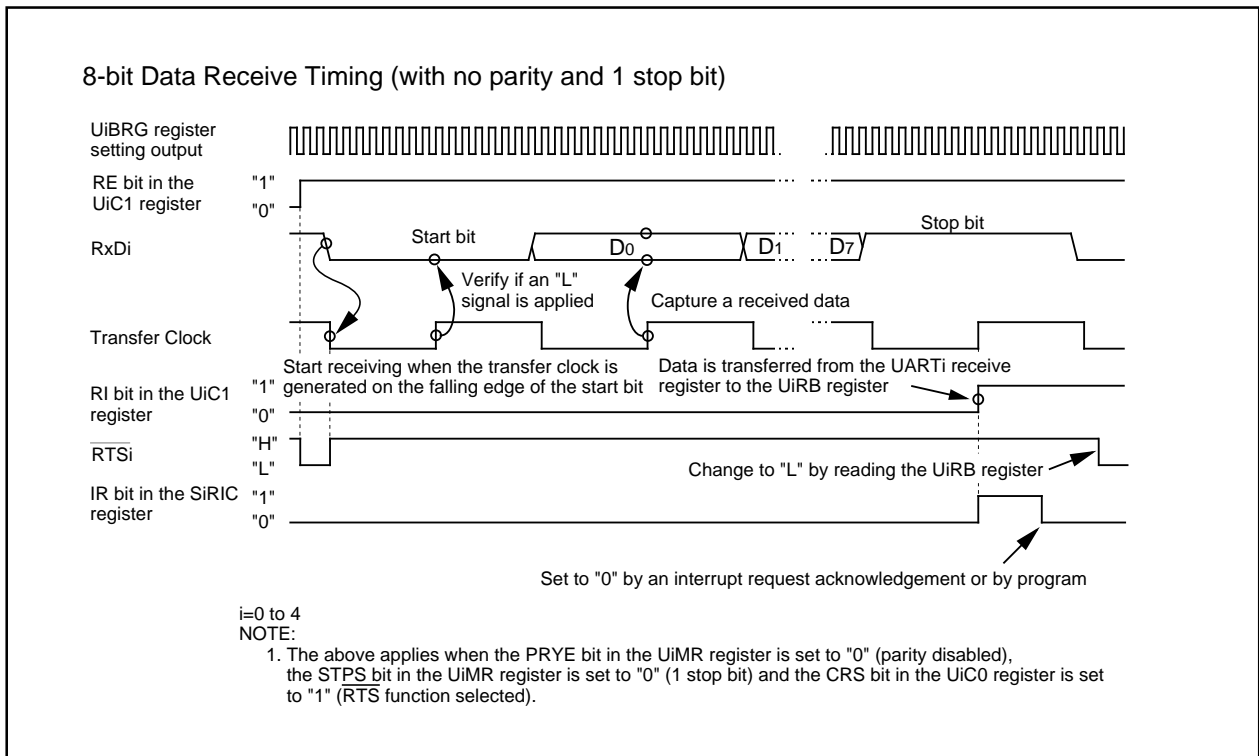


Figure 16.15 Receive Operation

### 16.2.1 Bit Rate

In UART mode, bit rate is clock frequency which is divided by a setting value of the UiBRG ( $i=0$  to 4) register and again divided by 16. Table 16.11 lists an example of bit rate setting.

Table 16.11 Bit Rate

Bit Rate (bps)	Count Source of UiBRG	Peripheral Function Clock: 16MHz		Peripheral Function Clock: 24MHz		Peripheral Function Clock: 32MHz	
		Setting Value of UiBRG: $n$	Actual Bit Rate (bps)	Setting Value of UiBRG: $n$	Actual Bit Rate (bps)	Setting Value of UiBRG: $n$	Actual Bit Rate (bps)
1200	f8	103 (67h)	1202	155 (96h)	1202	207 (CFh)	1202
2400	f8	51 (33h)	2404	77 (46h)	2404	103 (67h)	2404
4800	f8	25 (19h)	4808	38 (26h)	4808	51 (33h)	4808
9600	f1	103 (67h)	9615	155 (96h)	9615	207 (CFh)	9615
14400	f1	68 (44h)	14493	103 (67h)	14423	138 (8Ah)	14388
19200	f1	51 (33h)	19231	77 (46h)	19231	103 (67h)	19231
28800	f1	34 (22h)	28571	51 (33h)	28846	68 (44h)	28986
31250	f1	31 (1Fh)	31250	47 (2Fh)	31250	63 (3Fh)	31250
38400	f1	25 (19h)	38462	38 (26h)	38462	51 (33h)	38462
51200	f1	19 (13h)	50000	28 (1Ch)	51724	38 (26h)	51282

### 16.2.2 Selecting LSB First or MSB First

As shown in Figure 16.16, the UFORM bit in the UiC0 register (i=0 to 4) determines data transfer format. This function is available for 8-bit transfer data.

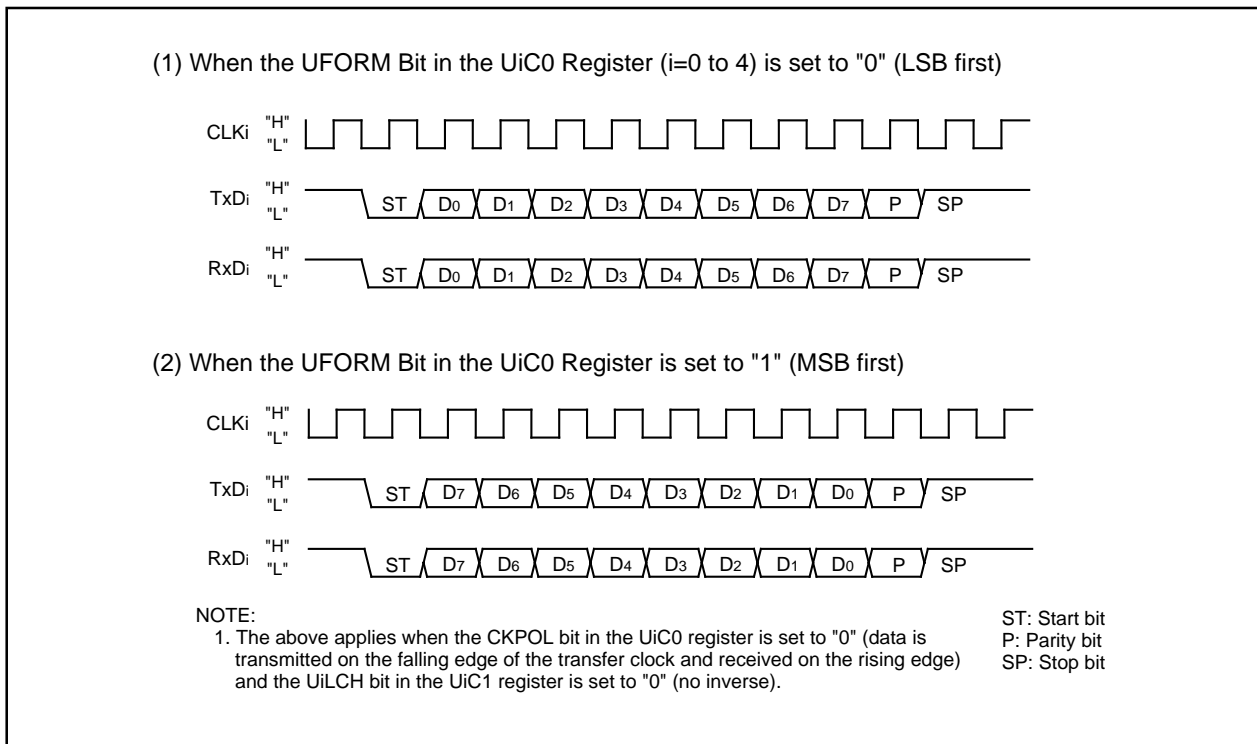


Figure 16.16 Transfer Format

### 16.2.3 Serial Data Logic Inverse

When the UiLCH bit (i=0 to 4) in the UiC1 register is set to "1" (inverse), data logic written in the UiTB register is inverted when transmitted. The inverted receive data logic can be read by reading the UiRB register. Figure 16.17 shows a switching example of the serial data logic.

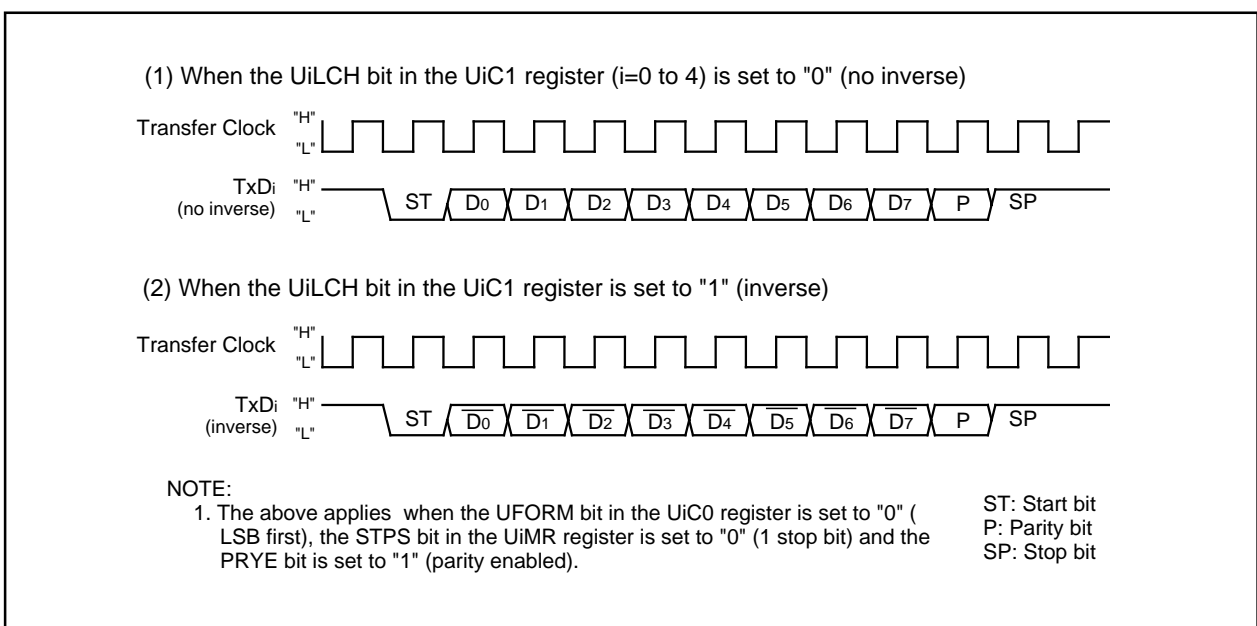


Figure 16.17 Serial Data Logic Inverse

### 16.2.4 TxD and RxD I/O Polarity Inverse

TxD pin output and RxD pin input are inverted. All I/O data level, including the start bit, stop bit and parity bit, are inverted. Figure 16.18 shows TxD and RxD I/O polarity inverse.

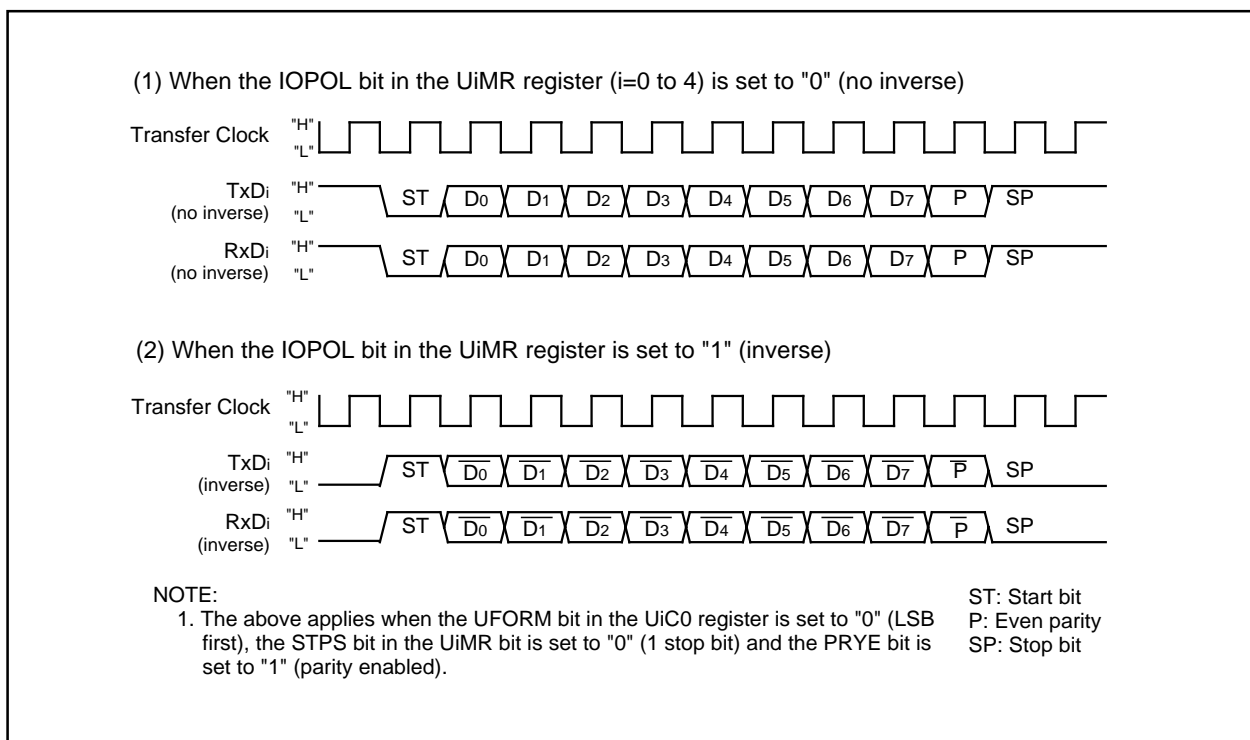


Figure 16.18 TxD and RxD I/O Polarity Inverse



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

I<sup>2</sup>C mode is a mode to communicate with external devices with a simplified I<sup>2</sup>C. Table 16.12 lists specifications of I<sup>2</sup>C mode. Table 16.13 lists register settings, Table 16.14 lists each function. Figure 16.19 shows a block diagram of I<sup>2</sup>C mode. Figure 16.20 shows timings for transfer to the UiRB register (i=0 to 4) and interrupts. Tables 16.15 to 16.17 list pin settings.

As shown in Table 16.12, I<sup>2</sup>C mode is entered when the SMD2 to SMD0 bits in the UiMR register is set to "0102" and the IICM bit in the UiSMR register is set to "1". Output signal from the SDAi pin changes after the SCLi pin level becomes low ("L") and stabilizes due to a SDAi transmit output via the delay circuit.

**Table 16.12 I<sup>2</sup>C Mode Specifications**

Item	Specifications
Interrupt	Start condition detect, stop condition detect, no acknowledgment detect, acknowledgment detect
Selectable Function	<ul style="list-style-type: none"> <li>• Arbitration lost            Selectable from update timing of the ABT bit in the UiRB register.            Refer to <b>16.3.3 Arbitration</b></li> <li>• SDAi digital delay            Selectable from no digital delay or 2 to 8 cycle delay of the count source of of the UiBRG register. Refer to <b>16.3.5 SDA Output</b></li> <li>• Clock phase setting            Selectable from clock delay or no clock delay. Refer to <b>16.3.4 Transfer Clock</b></li> </ul>

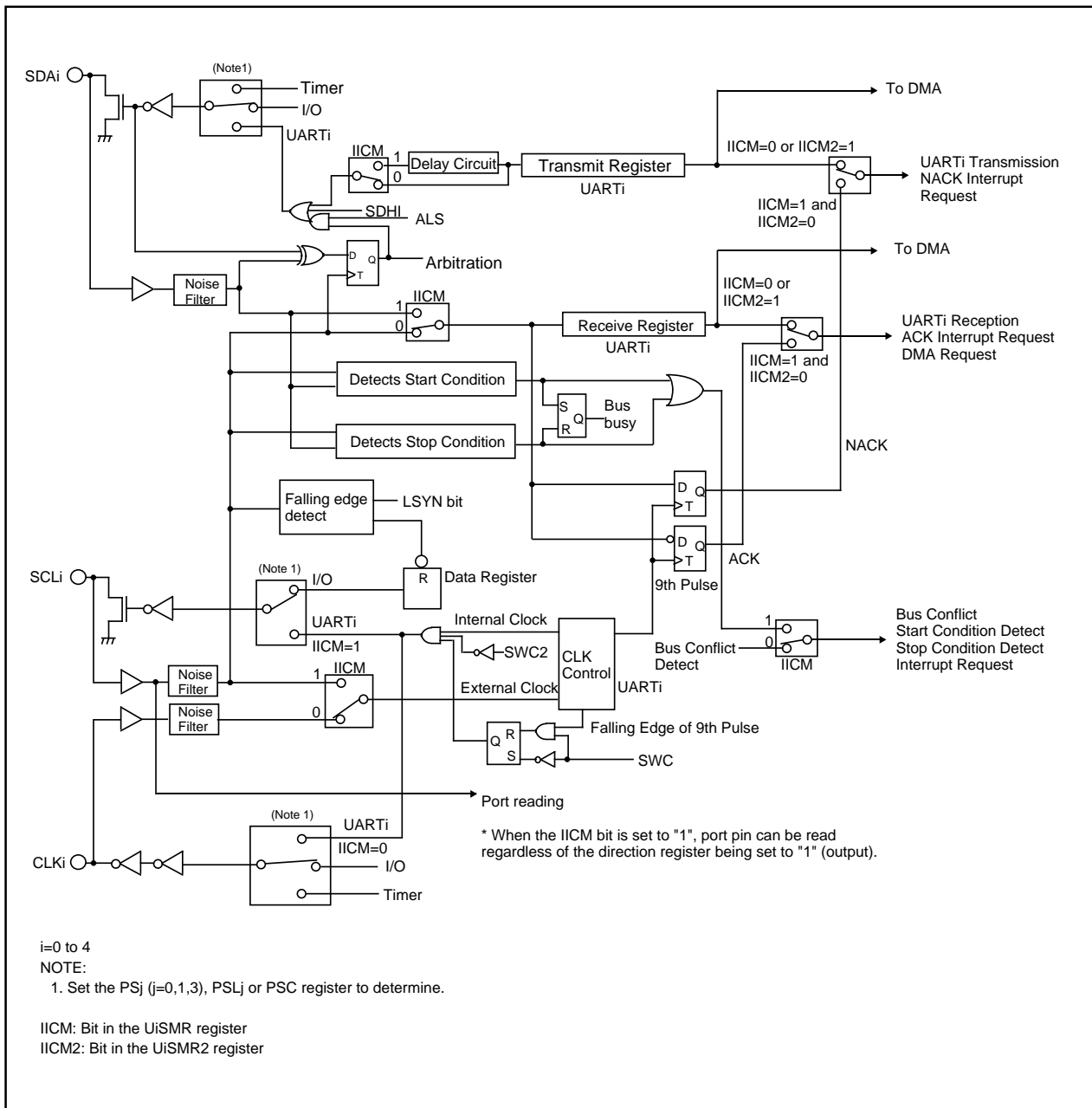


Figure 16.19 I<sup>2</sup>C Mode Block Diagram

**Table 16.13 Register Settings in I<sup>2</sup>C Mode**

Register	Bit	Function	
		Master	Slave
UiTB	7 to 0	Set transmit data	
UiRB	7 to 0	Received data can be read	
	8	ACK or NACK bit can be read	
	ABT	Arbitration lost detect flag	Disabled
	OER	Overrun error flag	
UiBRG	7 to 0	Set bit rate	Disabled
UiMR	SMD2 to SMD0	Set to "0102"	
	CKDIR	Set to "0"	Set to "1"
	IOPOL	Set to "0"	
UiC0	CLK1, CLK0	Select count source of the UiBRG register	Disabled
	CRS	Disabled because the CRD bit is set to "1"	
	TXEPT	Transfer register empty flag	
	CRD, NCH	Set to "1"	
	CKPOL	Set to "0"	
	UFORM	Set to "1"	
UiC1	TE	Set to "1" to enable data transmission	
	TI	Transfer buffer empty flag	
	RE	Set to "1" to enable data reception	
	RI	Reception complete flag	
	UiRRM, UiLCH, UIERE	Set to "0"	
UiSMR	IICM	Set to "1"	
	ABC	Select an arbitration lost detect timing	Disabled
	BBS	Bus busy flag	
	7 to 3	Set to "000002"	
UiSMR2	IICM2	See Table 16.14	
	CSC	Set to "1" to enable clock synchronization	Set to "0"
	SWC	Set to "1" to fix an "L" signal output from SCLi on the falling edge of the ninth bit of the transfer clock	
	ALS	Set to "1" to terminate SDAi output when detecting the arbitration lost	Not used. Set to "0"
	STC	Not used. Set to "0"	Set to "1" to reset UARTi by detecting the start condition
	SWC2	Set to "1" for an "L" signal output from SCL forcibly	
	SDHI	Set to "1" to disable SDA output	
	SU1HIM	Set to "0"	
UiSMR3	SSE	Set to "0"	
	CKPH	See Table 16.14	
	DINC, NODC, ERR	Set to "0"	
	DL2 to DL0	Set digital delay value for SDAi	
UiSMR4	STAREQ	Set to "1" when generating a start condition	Not used. Set to "0"
	RSTAREQ	Set to "1" when generating a restart condition	
	STPREQ	Set to "1" when generating a stop condition	
	STSPSEL	Set to "1" when using a condition generating function	
	ACKD	Select ACK or NACK	
	ACKC	Set to "1" for ACK data output	
	SCLHI	Set to "1" to enable SCL output stop when detecting stop condition	Not used. Set to "0"
	SWC9	Not used. Set to "0"	Set to "1" to fix an "L" signal output from SCLi on the falling edge of the ninth bit of the transfer clock
IFSR	IFSR6, IFSR7	Set to "1"	

i=0 to 4

**Table 16.14 I<sup>2</sup>C Mode Functions**

Function	Clock Synchronous Serial I/O Mode (SMD2 to SMD0=0012, IICM=0)	I <sup>2</sup> C Mode (SMD2 to SMD0=0102, IICM=1)			
		IICM2=0 (NACK/ACK interrupt)		IICM2=1 (UART transmit / UART receive interrupt)	
		CKPH=0 (No clock delay)	CKPH=1 (Clock delay)	CKPH=0 (No clock delay)	CKPH=1 (Clock delay)
Source for Interrupt Numbers 39 to 41 <sup>(1)</sup> (See Figure 16.20)	-	Start condition or stop condition detect (See Table 16.18)			
Source for Interrupt Number 17, 19, 33, 35 and 37 <sup>(1)</sup> (See Figure 16.20)	UARTi Transmission - Transmission started or completed (selected by the UiIRS register)	No Acknowledgement Detection (NACK) - Rising edge of 9th bit of SCLi	UARTi Transmission - Rising edge of 9th bit of SCLi	UARTi Transmission - Next falling edge after the 9th bit of SCLi	
Source for Interrupt Numbers 18, 20, 34, 36 and 38 <sup>(1)</sup> (See Figure 16.20)	UARTi Reception - Receiving at 8th bit CKPOL=0(rising edge) CKPOL=1(falling edge)	Acknowledgement Detection (ACK) - Rising edge of 9th bit of SCLi	UARTi Reception - Falling edge of 9th bit of SCLi		
Data Transfer Timing from the UART Receive Shift Register to the UiRB Register	CKPOL=0(rising edge) CKPOL=1(falling edge)	Rising edge of 9th bit of SCLi	Falling edge of 9th bit of SCLi	Falling edge and rising edge of 9th bit of SCLi	
UARTi Transmit Output Delay	No delay	Delay			
P63, P67, P70, P92, P96 Pin Functions	TxDi output	SDAi input and output			
P62, P66, P71, P91, P97 Pin Functions	RxDi input	SCLi input and output			
P61, P65, P72, P90, P95 Pin Functions	Select CLKi input or output	- (Not used in I <sup>2</sup> C mode)			
Noise Filter Width	15 ns	200 ns			
Reading RxDi and SCLi Pin Levels	Can be read if port direction bit is set to "0"	Can be read regardless of the port direction bit			
Default Value of TxDi, SDAi Output	CKPOL=0 (H) CKPOL=1 (L)	Values set in the port register before entering I <sup>2</sup> C mode <sup>(2)</sup>			
SCLi Default and End Value	-	H	L	H	L
Source for DMA (See Figure 16.20)	UARTi reception	Acknowledgement detection (ACK)	UARTi Reception - Falling edge of 9th bit of SCLi		
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 <sup>(3)</sup>		
Reading Received Data	The UiRB register status is read	Bits 6 to 0 in the UiRB registers <sup>(4)</sup> are read as bit 7 to 1. Bit 8 in the UiRB register is read as bit 0			

i=0 to 4

## NOTES:

- Use the following procedure to change what causes an interrupt to be generated.
  - Disable interrupt of corresponding interrupt number.
  - Change what causes an interrupt to be generated.
  - Set the IR bit of a corresponding interrupt number to "0" (no interrupt requested).
  - Set the ILVL2 to ILVL0 bits of a corresponding interrupt number.
- Set default value of the SDAi output when the SMD2 to SMD0 bits in the UiMR register are set to "0002" (serial I/O disabled).
- Second data transfer to the UiRB register (on the rising edge of the ninth bit of SCLi).
- First data transfer to the UiRB register (on the falling edge of the ninth bit of SCLi).

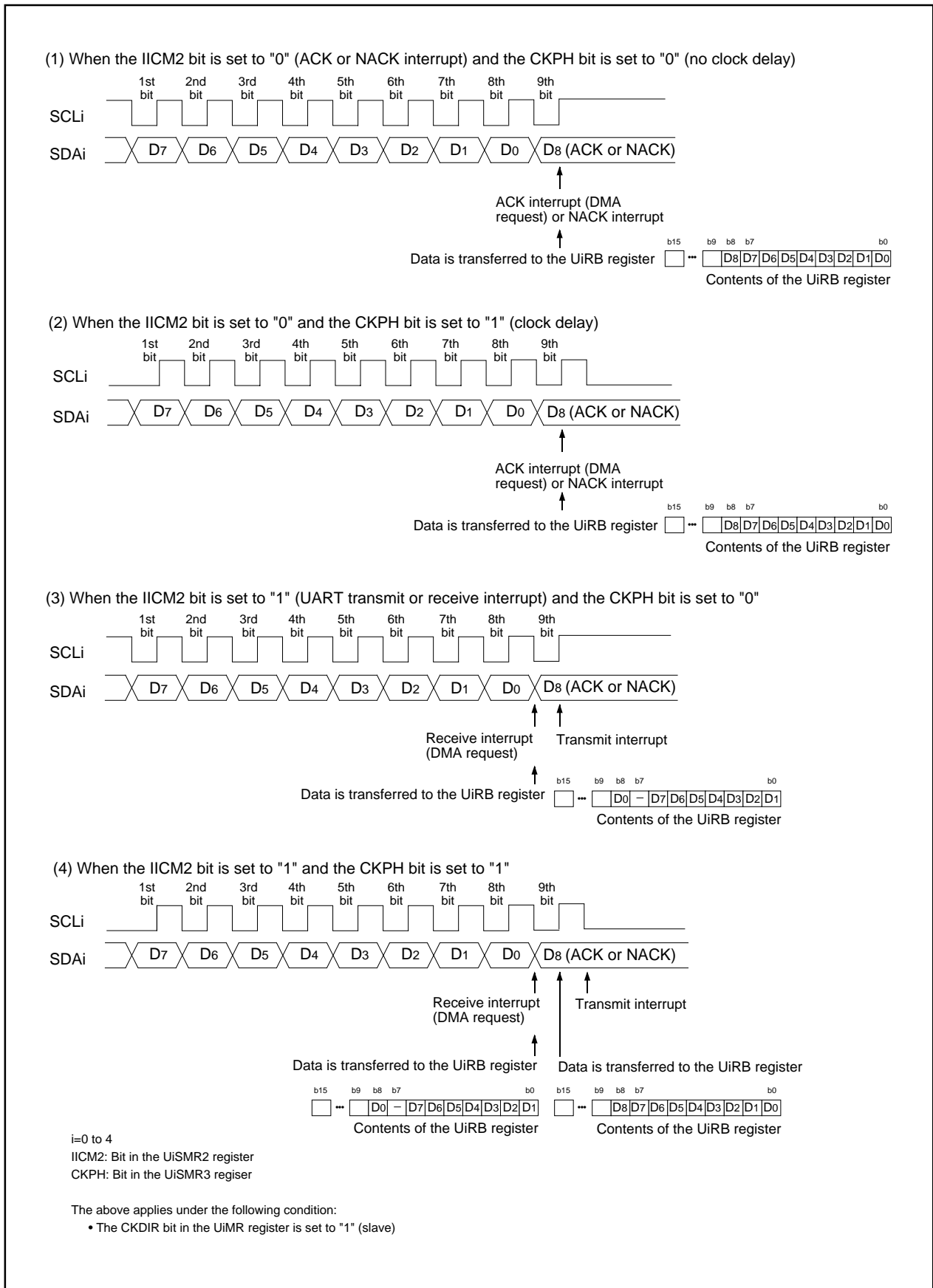


Figure 16.20 SCLi Timing

**Table 16.15 Pin Settings in I<sup>2</sup>C Mode (1)**

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P62	SCL0 output	PS0_2=1	PSL0_2=0	-
	SCL0 input	PS0_2=0	-	PD6_2=0
P63	SDA0 output	PS0_3=1	-	-
	SDA0 input	PS0_3=0	-	PD6_3=0
P66	SCL1 output	PS0_6=1	PSL0_6=0	-
	SCL1 input	PS0_6=0	-	PD6_6=0
P67	SDA1 output	PS0_7=1	-	-
	SDA1 input	PS0_7=0	-	PD6_7=0

**Table 16.16 Pin Settings (2)**

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70 <sup>(1)</sup>	SDA2 output	PS1_0=1	PSL1_0=0	PSC_0=0	-
	SDA2 input	PS1_0=0	-	-	PD7_0=0
P71 <sup>(1)</sup>	SCL2 output	PS1_1=1	PSL1_1=1	PSC_1=0	-
	SCL2 input	PS1_1=0	-	-	PD7_1=0

NOTE:

1. P70 and P71 are ports for the N-channel open drain output.

**Table 16.17 Pin Settings (3)**

Port	Function	Setting			
		PS3 Register <sup>(1)</sup>	PSL3 Register	PSC3 Register	PD9 Register <sup>(1)</sup>
P91	SCL3 output	PS3_1=1	PSL3_1=0	-	-
	SCL3 input	PS3_1=0	-	-	PD9_1=0
P92	SDA3 output	PS3_2=1	PSL3_2=0	-	-
	SDA3 input	PS3_2=0	-	-	PD9_2=0
P96	SDA4 output	PS3_6=1	-	PSC3_6=0	-
	SDA4 input	PS3_6=0	-	-	PD9_6=0
P97	SCL4 output	PS3_7=1	PSL3_7=0	-	-
	SCL4 input	PS3_7=0	-	-	PD9_7=0

NOTE:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

### 16.3.1 Detecting Start Condition and Stop Condition

The microcomputer detects either a start condition or stop condition. The start condition detect interrupt is generated when the SCL<sub>i</sub> (i=0 to 4) pin level is held high ("H") and the SDA<sub>i</sub> pin level changes "H" to low ("L"). The stop condition detect interrupt is generated when the SCL<sub>i</sub> pin level is held "H" and the SDA<sub>i</sub> pin level changes "L" to "H". The start condition detect interrupt shares interrupt control registers and vectors with the stop condition detect interrupt. The BBS bit in the UiSMR register determines which interrupt is requested.

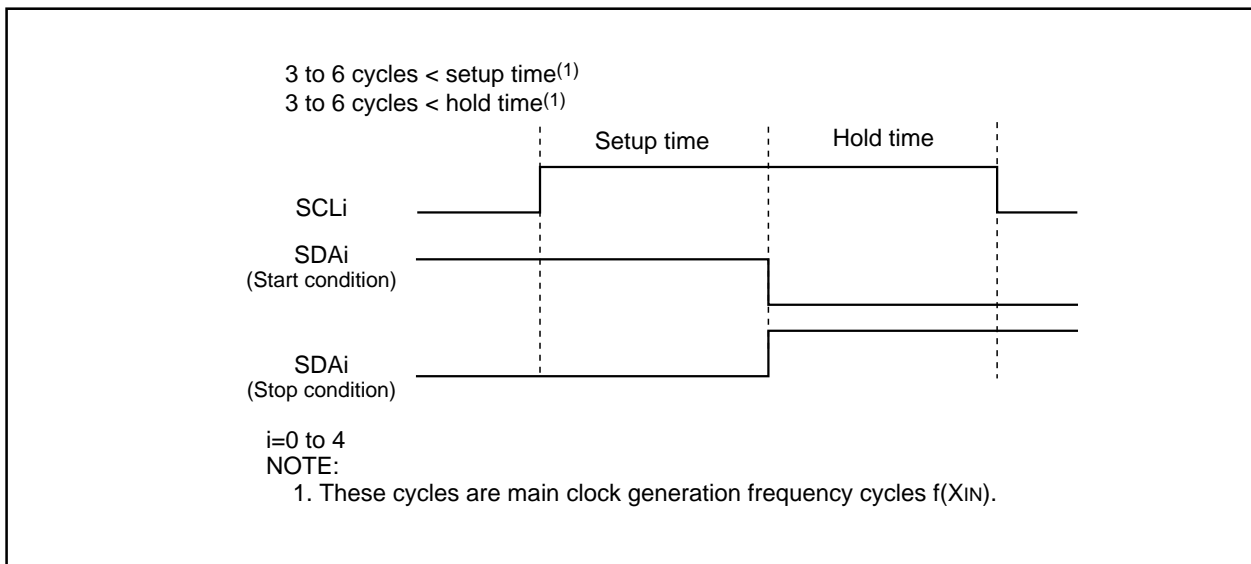


Figure 16.21 Start Condition or Stop Condition Detecting

### 16.3.2 Start Condition or Stop Condition Output

The start condition is generated when the STAREQ bit in the UiSMR4 register (i=0 to 4) is set to "1" (start). The restart condition is generated when the RSTAREQ bit in the UiSMR4 register is set to "1" (start). The stop condition is generated when the STPREQ bit in the UiSMR4 register is set to "1" (start).

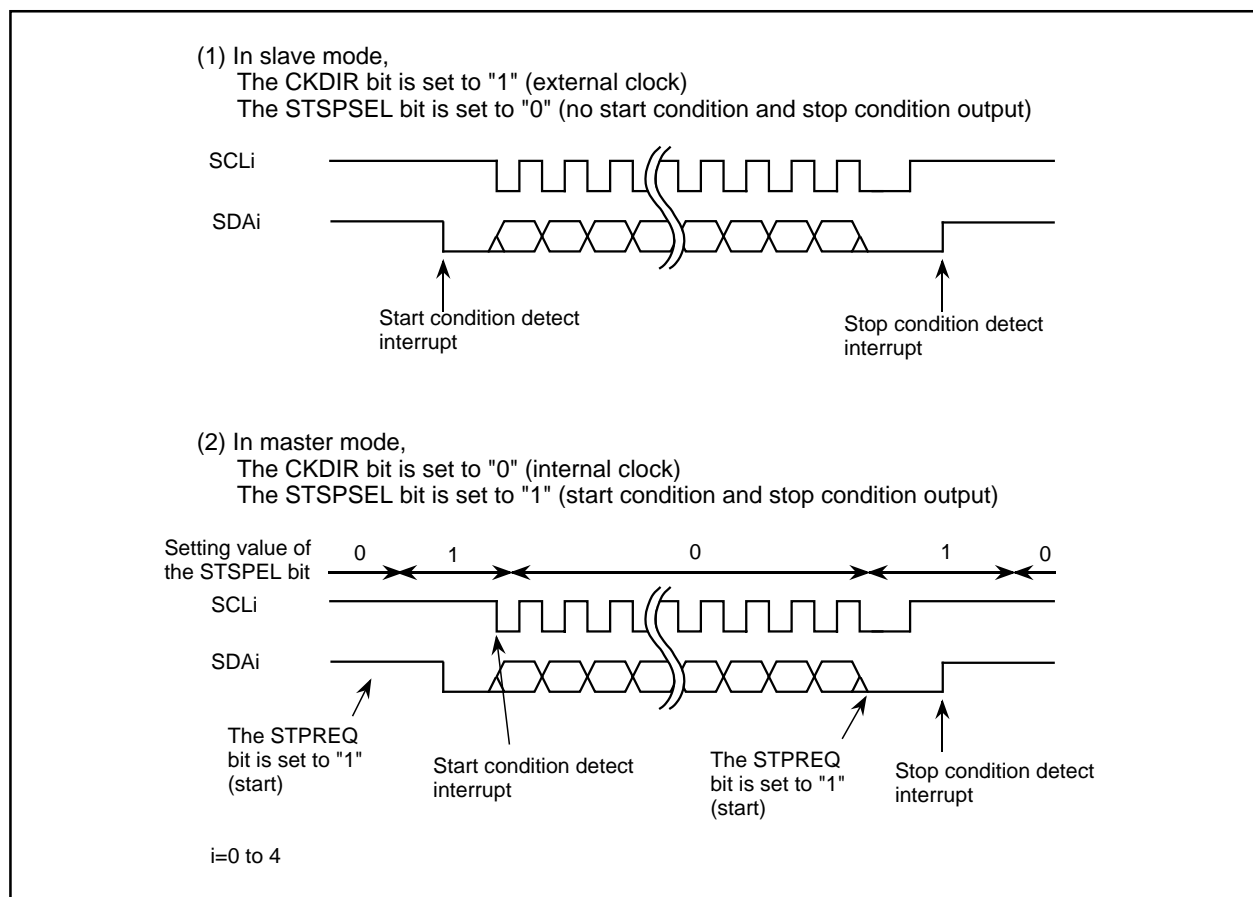
The start condition is output when the STAREQ bit is set to "1" and the STSPSEL bit in the UiSMR4 register is set to "1" (start or stop condition generating circuit selected). The restart condition output is provided when the RSTAREQ bit and STSPSEL bit are set to "1". The stop condition output is provided when the STPREQ bit and the STSPSEL bit are set to "1".

When the start condition, stop condition or restart condition is output, do not generate an interrupt between the instruction to set the STAREQ bit, STPREQ bit or RSTAREQ bit to "1" and the instruction to set the STSPSEL bit to "1". When the start condition is output, set the STAREQ bit to "1" before the STSPSEL bit is set to "1".

Table 16.18 lists function of the STSPSEL bit. Figure 16.22 shows functions of the STSPSEL bit.

**Table 16.18 STSPSEL Bit Function**

Function	STSPSEL = 0	STSPSEL = 1
Start condition and stop condition output	Program with ports determines how the start condition or stop condition output is provided	The STAREQ bit, RSTAREQ bit and STPREQ bit determine how the start condition or stop condition output is provided
Timing to generate start condition and stop condition interrupt requests	The start condition and stop condition are detected	Start condition and stop condition generation are completed



**Figure 16.22 STSPSEL Bit Function**



### 16.3.3 Arbitration

The ABC bit in the UiSMR register (i=0 to 4) determines an update timing for the ABT bit in the UiRB register. On the rising edge of the SCLi pin, the microcomputer determines whether a transmit data matches data input to the SDAi pin.

When the ABC bit is set to "0" (update per bit), the ABT bit is set to "1" (detected-arbitration is lost) as soon as a data discrepancy is detected. The ABT bit is set to "0" (not detected-arbitration is won) if not detected. When the ABC bit is set to "1" (update per byte), the ABT bit is set to "1" on the falling edge of the ninth bit of the transfer clock if any discrepancy is detected. When the ABT bit is updated per byte, set the ABT bit to "0" between an ACK detection in the first byte data and the next byte data to be transferred. When the ALS bit in the UiSMR2 register is set to "1" (SDA output stop enabled), the arbitration lost occurs. As soon as the ABT bit is set to "1", the SDAi pin is placed in a high-impedance state.

### 16.3.4 Transfer Clock

The transfer clock transmits and receives data as is shown in Figure 16.20.

The CSC bit in the UiSMR2 register (i=0 to 4) synchronizes an internally generated clock (internal SCLi) with the external clock applied to the SCLi pin. When the CSC bit is set to "1" (clock synchronous enabled) and the internal SCLi is held high ("H"), the internal SCLi become low ("L") if signal applied to the SCLi pin is on the falling edge. Value of the UiBRG register is reloaded to start counting for low level. A counter stops when the SCLi pin is held "L" and then the internal SCLi changes "L" to "H". Counting is resumed when the SCLi pin become "H". The transfer clock of UARTi is equivalent to the AND for signals from the internal SCLi and the SCLi pin.

The transfer clock is synchronized between a half cycle before the falling edge of first bit of the internal SCLi and the rising edge of the ninth bit. Select the internal clock as the transfer clock while the CSC bit is set to "1".

The SWC bit in the UiSMR2 register determines whether the SCLi pin is fixed to be an "L" signal output on the falling edge of the ninth cycle of the transfer clock or not.

When the SCLHI bit in the UiSMR4 register is set to "1" (enabled), a SCLi output stops when a stop condition is detected (high-impedance).

When the SWC2 bit in the UiSMR2 register is set to "1" (0 output), the SCLi pin forcibly outputs an "L" signal while transmitting and receiving. The fixed "L" signal applied to the SCLi pin is cancelled by setting the SWC2 bit to "0" (transfer clock) and the transfer clock input to and output from the SCLi pin are provided.

When the CKPH bit in the UiSMR3 register is set to "1" and the SWC9 bit in the UiSMR4 register is set to "1" (SCL "L" hold enabled), the SCLi pin is fixed to be an "L" signal output on the next falling edge after the ninth bit of the clock. The fixed "L" signal applied to the SCLi pin is cancelled by setting the SWC9 bit to "0" (SCL "L" hold disabled).

### 16.3.5 SDA Output

Values output set in bits 7 to 0 (D7 to D0) in the UiTB register (i=0 to 4) are provided in descending order from D7. The ninth bit (D8) is ACK or NACK.

Set the default value of SDAi transmit output when the IICM bit is set to "1" (I<sup>2</sup>C mode) and the SMD2 to SMD0 bits in the UiMR register are set to "0002" (serial I/O disabled).

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

When the SDHI bit in the UiSMR2 register is set to "1" (SDA output disabled), the SDAi pin is forcibly placed in a high-impedance state. Do not set the SDHI bit on the rising edge of the UARTi transfer clock. The ABT bit in the UiRB register may be set to "1" (detected).

### 16.3.6 SDA Input

When the IICM2 bit in the UiSMR2 register (i=0 to 4) is set to "0", the first eight bits of received data are stored into bits 7 to 0 (D7 to D0) in the UiRB register. The ninth bit (D8) is ACK or NACK.

When the IICM2 bit is set to "1", the first seven bits (D7 to D1) of received data are stored into bits 6 to 0 in the UiRB register. Store the eighth bit (D0) into bit 8 in the UiRB register.

If the IICM2 bit is set to "1" and the CKPH bit in the UiSMR3 register is set to "1", the same data as that of when setting the IICM2 bit to "0" can be read. To read the data, read the UiRB register after the rising edge of the ninth bit of the transfer clock.

### 16.3.7 ACK, NACK

When the STPSEL bit in the UiSMR4 register (i=0 to 4) is set to "0" (serial I/O circuit selected) and the ACKC bit in the UiSMR4 register is set to "1" (ACK data output), the SDAi pin provides the value output set in the ACKD bit in the UiSMR4 register.

If the IICM2 bit is set to "0", the NACK interrupt request is generated when the SDAi pin is held high ("H") on the rising edge of the ninth bit of the transfer clock. The ACK interrupt request is generated when the SDAi pin is held low ("L") on the rising edge of the ninth bit of the transfer clock.

When ACK is selected to generate a DMA request, the DMA transfer is activated by an ACK detection.

### 16.3.8 Transmit and Receive Reset

When the STC bit in the UiSMR2 register (i=0 to 4) is set to "1" (UARTi initialization enabled) and a start condition is detected,

- the transmit shift register is reset and the content of the UiTB register is transferred to the transmit shift register. The first bit starts transmitting when the next clock is input. UARTi output value remains unchanged between when the clock is applied and when the first bit data output is provided. The value remains the same as when start condition was detected.
- the receive shift register is reset and the first bit start receiving when the next clock is applied.
- the SWC bit is set to "1" (SCL wait output enabled). The SCLi pin becomes "L" on the falling edge of the ninth bit of the transfer clock.

If UARTi transmission and reception are started with this function, the TI bit in the UiC1 register remains unchanged. Select the external clock as the transfer clock when using this function.

## 16.4 Special Mode 2

In special mode 2, serial communication between one or multiple masters and multiple slaves is available. The  $\overline{SSi}$  input pin ( $i=0$  to 4) controls the serial bus communication. Table 16.19 lists specifications of special mode 2. Table 16.20 lists register settings. Tables 16.21 to 16.23 list pin settings.

**Table 16.19 Special Mode 2 Specifications**

Item	Specification
Transfer Data Format	Transfer data : 8 bits long
Transfer Clock	<ul style="list-style-type: none"> <li>The CKDIR bit in the UiMR register (<math>i=0</math> to 4) is set to "0" (internal clock selected):  <math>f_j/2(m+1)</math> <math>f_j = f_1, f_8, f_{2n}^{(1)}</math> <math>m</math>: setting value of the UiBRG register, 0016 to FF16</li> <li>The CKDIR bit to "1" (external clock selected) : input from the CLKi pin</li> </ul>
Transmit/Receive Control	$\overline{SSi}$ input pin function
Transmit Start Condition	To start transmitting, the following requirements must be met <sup>(2)</sup> : <ul style="list-style-type: none"> <li>- Set the TE bit in the UiC1 register to "1" (transmit enabled)</li> <li>- Set the TI bit in the UiC1 register to "0" (data in the UiTB register)</li> </ul>
Receive Start Condition	To start receiving, the following requirement must be met <sup>(2)</sup> : <ul style="list-style-type: none"> <li>- Set the RE bit in the UiC1 register to "1" (receive enabled)</li> <li>- Set the TE bit in the UiC1 register to "1" (transmit enabled)</li> <li>- Set the TI bit in the UiC1 register to "0" (data in the UiTB register)</li> </ul>
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>While transmitting, the following conditions can be selected: <ul style="list-style-type: none"> <li>- The UiIRS bit in the UiC1 register is set to "0" (no data in a transmit buffer) : when data is transferred from the UiTB register to the UAR<i>Ti</i> transmit register (transmission started)</li> <li>- The UiIRS register is set to "1" (transmission completed): when data transmission from UAR<i>Ti</i> transfer register is completed</li> </ul> </li> <li>While receiving When data is transferred from the UAR<i>Ti</i> receive register to the UiRB register (reception completed)</li> </ul>
Error Detection	<ul style="list-style-type: none"> <li>• Overrun error<sup>(3)</sup> This error occurs when the seventh bit of the next received data is read before reading the UiRB register</li> <li>• Fault error In master mode, the fault error occurs an "L" signal is applied to the <math>\overline{SSi}</math> pin</li> </ul>
Selectable Function	<ul style="list-style-type: none"> <li>• CLK polarity Selectable from the rising edge or falling edge of the transfer clock at transferred data output or input timing</li> <li>• LSB first or MSB first Selectable from data transmission or reception in either bit 0 or in bit 7</li> <li>• Continuous receive mode Data reception is enabled simultaneously by reading the UiRB register</li> <li>• Serial data logic inverse This function inverses transmitted or received data logically</li> <li>• TxD and RxD I/O polarity inverse TxD pin output and RxD pin input are inversed. All I/O data levels are also inversed</li> <li>• Clock phase Selectable from one of 4 combinations of transfer data polarity and phases</li> <li>• <math>\overline{SSi}</math> input pin function Output pin is placed in a high-impedance state to avoid data conflict between master and other masters or slaves</li> </ul>

NOTES:

- The CNT3 to CNT0 bits in the TCS*PR* register select no division ( $n=0$ ) or divide-by-2*n* ( $n=1$  to 15).
- To start transmission/reception when selecting the external clock, these conditions must be met after the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and data is received on the rising edge) and the CLKi pin is held high ("H"), or when the CKPOL bit is set to "1" (Data is transmitted on the rising edge of the transfer clock and data is received on the falling edge) and the CLKi pin is held low ("L").
- If an overrun error occurs, the UiRB register is in an indeterminate state. The IR bit setting in the SiR*IC* register does not change to "1" (interrupt requested).

**Table 16.20 Register Settings in Special Mode 2**

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data can be read
	OER	Overflow error flag
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "0012"
	CKDIR	Set to "0" in master mode or "1" in slave mode
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Select the output format of the TxDi pin
	CKPOL	Clock phase can be set by the combination of the CKPOL bit and the CKPH bit in the UiSMR3 register
	UFORM	Select either LSB first or MSB first
UiC1	TE	Set to "1" to enable data transmission and reception
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM	Set to "1" to enable continuous receive mode
	UiLCH, SCLKSTPB	Set to "0"
UiSMR	7 to 0	Set to "0016"
UiSMR2	7 to 0	Set to "0016"
UiSMR3	SSE	Set to "1"
	CKPH	Clock phase can be set by the combination of the CKPH bit and the CKPOL bit in the UiC0 register
	DINC	Set to "0" in master mode or "1" in slave mode
	NODC	Set to "0"
	ERR	Fault error flag
	7 to 5	Set to "0002"
UiSMR4	7 to 0	Set to "0016"

i=0 to 4

**Table 16.21 Pin Settings in Special Mode 2 (1)**

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P60	SS0 input	PS0_0=0	–	PD6_0=0
P61	CLK0 input (slave)	PS0_1=0	–	PD6_1=0
	CLK0 output (master)	PS0_1=1	–	–
P62	RxD0 input (master)	PS0_2=0	–	PD6_2=0
	STxD0 output (slave)	PS0_2=1	PSL0_2=1	–
P63	TxD0 output (master)	PS0_3=1	–	–
	SRxD0 input (slave)	PS0_3=0	–	PD6_3=0
P64	SS1 input	PS0_4=0	–	PD6_4=0
P65	CLK1 input (slave)	PS0_5=0	–	PD6_5=0
	CLK1 output (master)	PS0_5=1	–	–
P66	RxD1 input (master)	PS0_6=0	–	PD6_6=0
	STxD1 output (slave)	PS0_6=1	PSL0_6=1	–
P67	TxD1 output (master)	PS0_7=1	–	–
	SRxD1 input (slave)	PS0_7=0	–	PD6_7=0

**Table 16.22 Pin Settings (2)**

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70 <sup>(1)</sup>	TxD2 output (master)	PS1_0=1	PSL1_0=0	PSC_0=0	–
	SRxD2 input (slave)	PS1_0=0	–	–	PD7_0=0
P71 <sup>(1)</sup>	RxD2 input (master)	PS1_1=0	–	–	PD7_1=0
	STxD2 output (slave)	PS1_1=1	PSL1_1=1	PSC_1=0	–
P72	CLK2 input (slave)	PS1_2=0	–	–	PD7_2=0
	CLK2 output (master)	PS1_2=1	PSL1_2=0	PSC_2=0	–
P73	SS2 input	PS1_3=0	–	–	PD7_3=0

NOTE:

1. P70 and P71 are ports for the N-channel open drain output.

**Table 16.23 Pin Settings (3)**

Port	Function	Setting			
		PS3 Register <sup>(1)</sup>	PSL3 Register	PSC3 Register	PD9 Register <sup>(1)</sup>
P90	CLK3 input (slave)	PS3_0=0	–	–	PD9_0=0
	CLK3 output (master)	PS3_0=1	–	–	–
P91	RxD3 input (master)	PS3_1=0	–	–	PD9_1=0
	STxD3 output (slave)	PS3_1=1	PSL3_1=1	–	–
P92	TxD3 output (master)	PS3_2=1	PSL3_2=0	–	–
	SRxD3 input (slave)	PS3_2=0	–	–	PD9_2=0
P93	SS3 input	PS3_3=0	PSL3_3=0	–	PD9_3=0
P94	SS4 input	PS3_4=0	PSL3_4=0	–	PD9_4=0
P95	CLK4 input (slave)	PS3_5=0	PSL3_5=0	–	PD9_5=0
	CLK4 output (master)	PS3_5=1	–	–	–
P96	TxD4 output (master)	PS3_6=1	–	PSC3_6=0	–
	SRxD4 input (slave)	PS3_6=0	PSL3_6=0	–	PD9_6=0
P97	RxD4 input (master)	PS3_7=0	–	–	PD9_7=0
	STxD4 output (slave)	PS3_7=1	PSL3_7=1	–	–

NOTE:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

### 16.4.1 $\overline{SS}_i$ Input Pin Function (i=0 to 4)

When the SSE bit in the UiSMR3 register is set to "1" ( $\overline{SS}$  function enabled), the special mode 2 is selected, activating the pin function.

The DINC bit in the UiSMR3 register determines which microcomputer performs as master or slave. When multiple microcomputers perform as the masters (multi-master system), the  $\overline{SS}_i$  pin setting determines which master microcomputer is active and when.

#### 16.4.1.1 When Setting the DINC Bit to "1" (Slave Mode)

When a high-level ("H") signal is applied to the  $\overline{SS}_i$  pin, the STxDi and SRxDi pins are placed in a high-impedance state and the transfer clock applied to the CLKi pin is ignored. When a low-level ("L") signal is applied to the  $\overline{SS}_i$  input pin, the transfer clock input is valid and serial communication is enabled.

#### 16.4.1.2 When Setting the DINC Bit to "0" (Master Mode)

When using the  $\overline{SS}_i$  pin function in master mode, set the UilRS bit in the UiC1 register to "1" (transmission completed).

When an "H" signal is applied to the  $\overline{SS}_i$  pin, serial communication is available due to transmission privilege. The master provides the transfer clock output. When an "L" signal is applied to the  $\overline{SS}_i$  pin, it indicates that another master is active. The TxDi and CLKi pins are placed in high-impedance states and the ERR bit in the UiSMR3 register is set to "1" (fault error). Use the transmit complete interrupt routine to verify the ERR bit state.

To resume the serial communication after the fault error occurs, set the ERR bit to "0" while applying the "H" signal to the  $\overline{SS}_i$  pin. The TxDi and CLKi pins become ready for signal outputs.

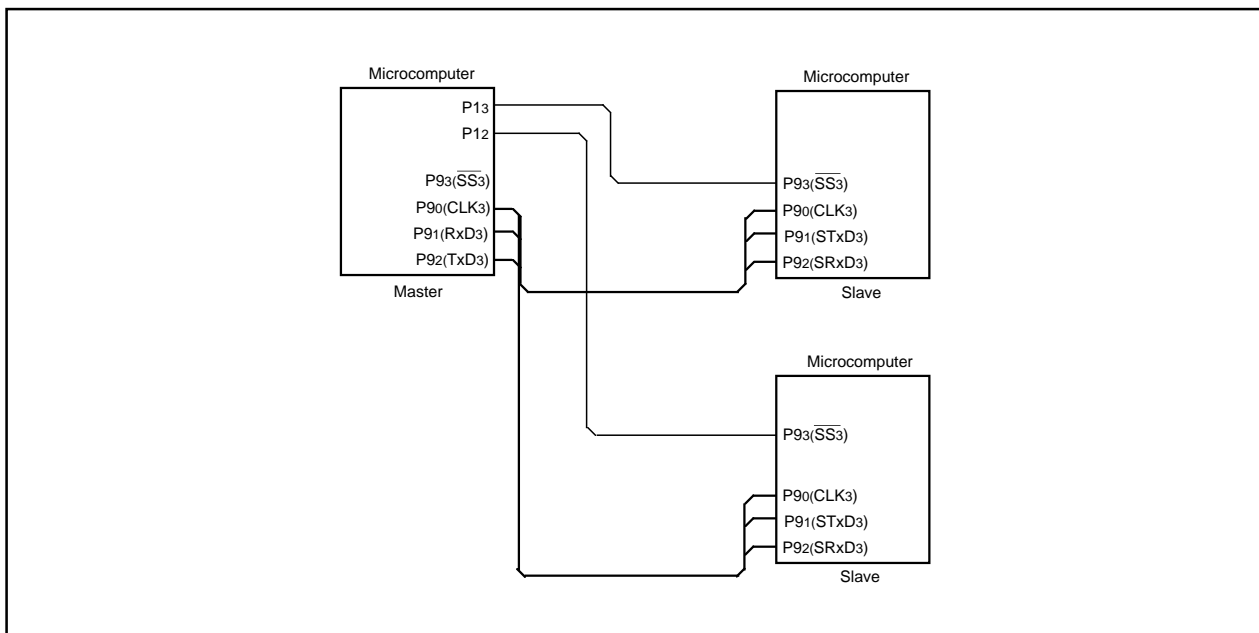


Figure 16.23 Serial Bus Communication Control with SS Pin

### 16.4.2 Clock Phase Setting Function

The CKPH bit in the UiSMR3 register (i=0 to 4) and the CKPOL bit in the UiC0 register select one of four combinations of transfer clock polarity and phases.

The transfer clock phase and polarity must be the same between the master and the slave involved in the transfer.

#### 16.4.2.1 When setting the DINC Bit to "0" (Master (Internal Clock))

Figure 16.24 shows transmit and receive timing.

#### 16.4.2.2 When Setting the DINC Bit to "1" (Slave (External Clock))

When the CKPH bit is set to "0" (no clock delay) and the  $\overline{SSi}$  input pin is held high ("H"), the STxDi pin is placed in a high-impedance state. When the  $\overline{SSi}$  input pin becomes low ("L"), conditions to start a serial transfer are met, but output is indeterminate. The serial transmission is synchronized with the transfer clock. Figure 16.25 shows the transmit and receive timing.

When the CKPH bit is set to "1" (clock delay) and the  $\overline{SSi}$  input pin is held high, the STxDi pin is placed in a high-impedance state. When the  $\overline{SSi}$  pin becomes low, the first data is output. The serial transmission is synchronized with the transfer clock. Figure 16.26 shows the transmit and receive timing.

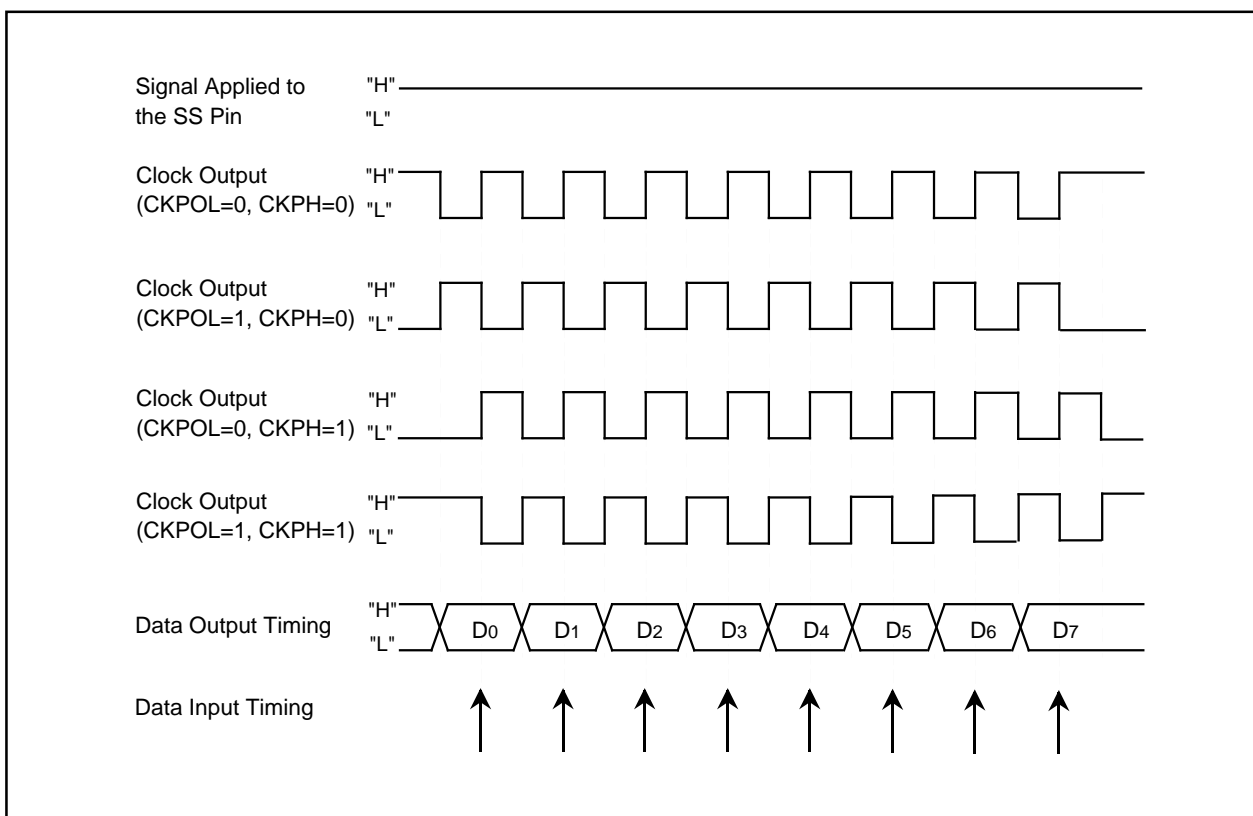


Figure 16.24 Transmit and Receive Timing in Master Mode (Internal Clock)

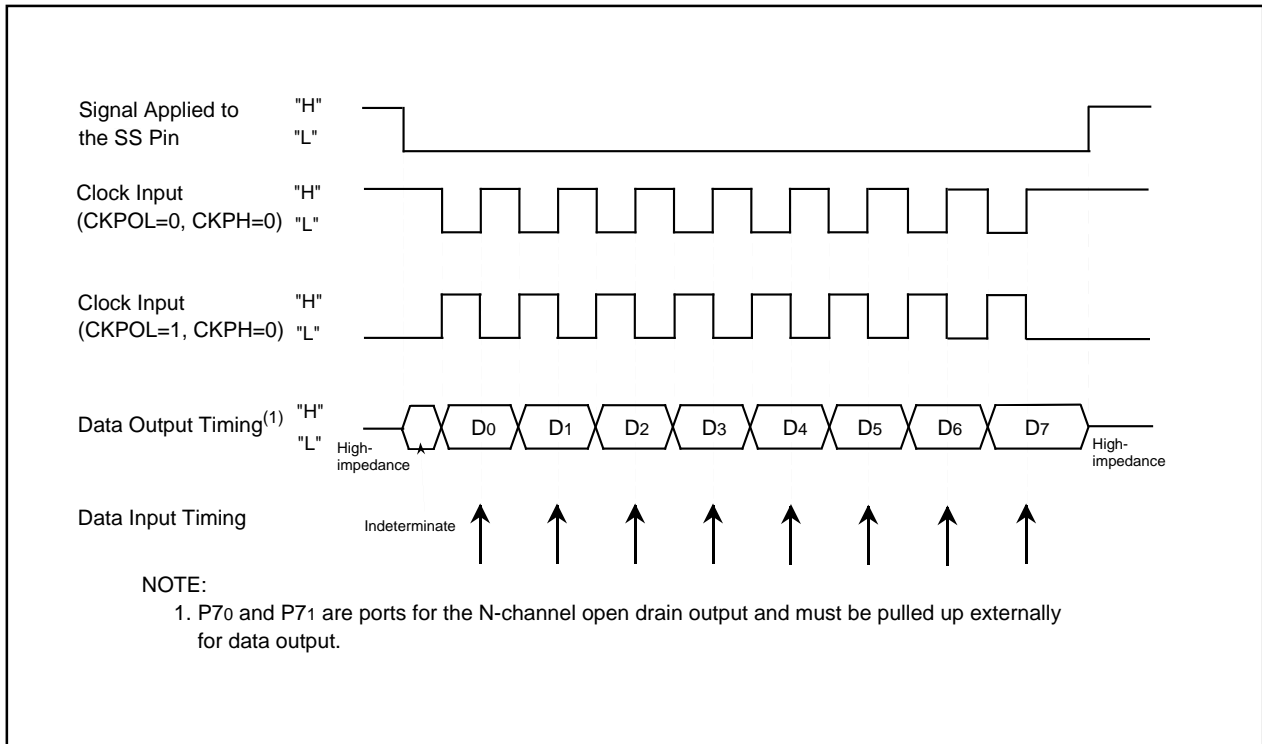


Figure 16.25 Transmit and Receive Timing in Slave Mode (External Clock) (CKPH=0)

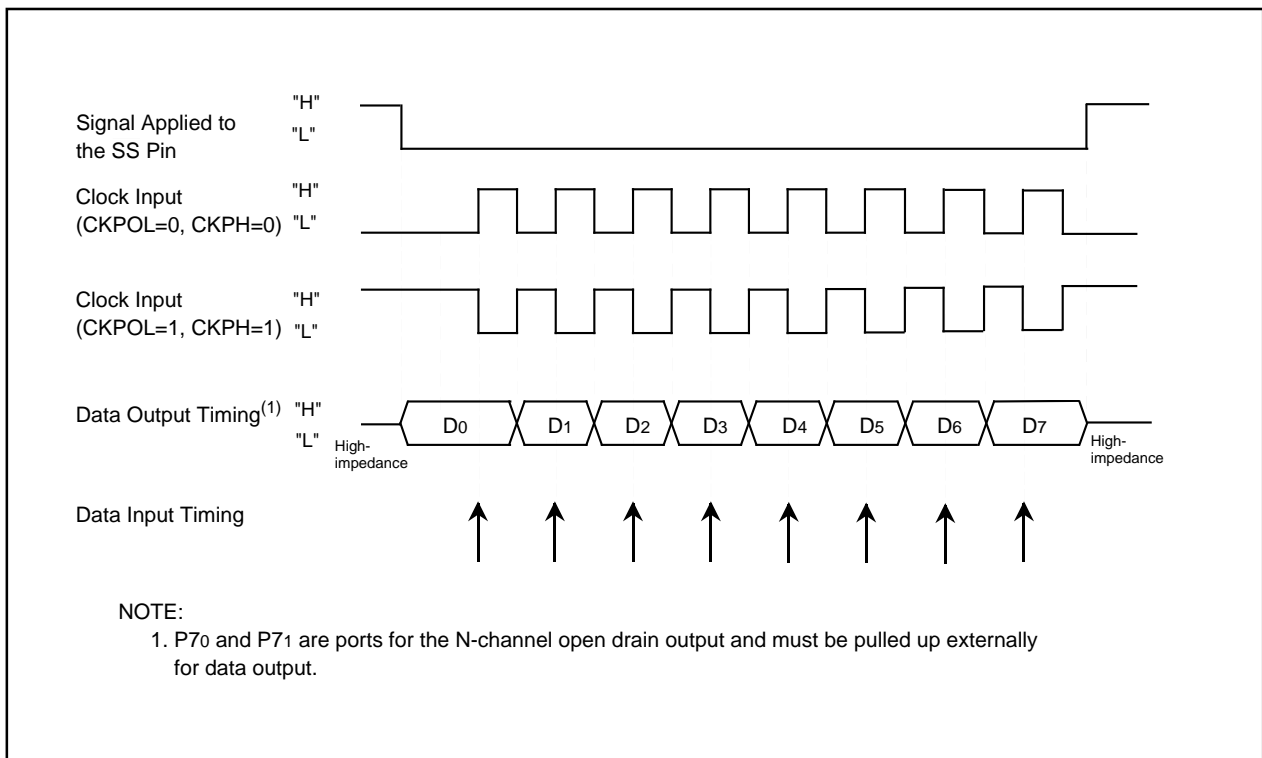


Figure 16.26 Transmit and Receive Timing in Slave Mode (External Clock) (CKPH=1)



### 16.5 Special Mode 3 (GCI Mode)

In GCI mode, the external clock is synchronized with the transfer clock used in the clock synchronous serial I/O mode.

Table 16.24 lists specifications of GCI mode. Table 16.25 lists registers settings. Tables 16.26 to 16.28 list pin settings.

**Table 16.24 GCI Mode Specifications**

Item	Specification
Transfer Data Format	Transfer data : 8 bits long
Transfer Clock	The CKDIR bit in the UiMR register (i=0 to 4) is set to "1" (external clock selected): input from the CLKi pin
Clock Synchronization Function	Trigger signal input from the $\overline{\text{CTS}}_i$ pin
Transmit/Receive Start Condition	To start data transmission and reception, meet the following conditions and then apply a trigger signal to the $\overline{\text{CTS}}_i$ pin: <ul style="list-style-type: none"> <li>- Set the TE bit in the UiC1 register to "1" (transmit enabled)</li> <li>- Set the RE bit in the UiC1 register to "1" (receive enabled)</li> <li>- Set the TI bit in the UiC1 register to "0" (Data in the UiTB register)</li> </ul>
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• While transmitting, the following condition can be selected: <ul style="list-style-type: none"> <li>- The UiIRS bit in the UiC1 register is set to "0" (UiTB register empty): when data is transferred from the UiTB register to the UARTi transmit register (transmission started)</li> <li>- The UiIRS bit is set to "1" (Transmit completed): when a data transmission from the UARTi transfer register is completed</li> </ul> </li> <li>• While receiving, when data is transferred from the UARTi receive register to the UiRB register (reception completed)</li> </ul>
Error Detection	<p>Overflow error<sup>(1)</sup></p> <p>This error occurs when the seventh bit of the next received data is read before reading the UiRB register.</p>

**NOTE:**

1. If an overflow error occurs, the UiRB register is indeterminate. The IR bit setting in the SiRIC register does not change to "1" (interrupt requested).

**Table 16.25 Register Settings in GCI Mode**

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data
	OER	Overflow error flag
UiBRG	7 to 0	Set to "0016"
UiMR	SMD2 to SMD0	Set to "0012"
	CKDIR	Set to "1"
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Set to "002"
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Select the output format of the TxDi pin
	CKPOL	Set to "0"
	UFORM	Set to "0"
UiC1	TE	Set to "1" to enable data transmission and reception
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM, UiLCH	Set to "0"
	SCLKSTPB	Set to "0"
UiSMR	6 to 0	Set to "00000002"
	SCLKDIV	See Table 16.29
UiSMR2	6 to 0	Set to "00000002"
	SU1HIM	See Table 16.29
UiSMR3	2 to 0	Set to "0002"
	NODC	Set to "0"
	7 to 4	Set to "00002"
UiSMR4	7 to 0	Set to "0016"

i=0 to 4

**Table 16.26 Pin Settings in GCI Mode (1)**

Port	Function	Setting	
		PS0 Register	PD6 Register
P60	CTS0 input <sup>(1)</sup>	PS0_0=0	PD6_0=0
P61	CLK0 input	PS0_1=0	PD6_1=0
P62	RxD0 input	PS0_2=0	PD6_2=0
P63	TxD0 output	PS0_3=1	–
P64	CTS1 input <sup>(1)</sup>	PS0_4=0	PD6_4=0
P65	CLK1 input	PS0_5=0	PD6_5=0
P66	RxD1 input	PS0_6=0	PD6_6=0
P67	TxD1 output	PS0_7=1	–

NOTE:

1. CTS input is used as a trigger signal input.

**Table 16.27 Pin Settings (2)**

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70 <sup>(1)</sup>	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
P71 <sup>(1)</sup>	RxD2 input	PS1_1=0	–	–	PD7_1=0
P72	CLK2 input	PS1_2=0	–	–	PD7_2=0
P73	CTS2 input <sup>(2)</sup>	PS1_3=0	–	–	PD7_3=0

NOTES:

1. P70 and P71 are ports for the N-channel open drain output.
2. CTS input is used as a trigger signal input.

**Table 16.28 Pin Settings (3)**

Port	Function	Setting			
		PS3 Register <sup>(1)</sup>	PSL3 Register	PSL3 Register	PD9 Register <sup>(1)</sup>
P90	CLK3 input	PS3_0=0	–	–	PD9_0=0
P91	RxD3 input	PS3_1=0	–	–	PD9_1=0
P92	TxD3 output	PS3_2=1	PSL3_2=0	–	–
P93	CTS3 input <sup>(2)</sup>	PS3_3=0	PSL3_3=0	–	PD9_3=0
P94	CTS4 input <sup>(2)</sup>	PS3_4=0	PSL3_4=0	–	PD9_4=0
P95	CLK4 input	PS3_5=0	PSL3_5=0	–	PD9_5=0
P96	TxD4 output	PS3_6=1	PSL3_6=0	PSL3_6=0	–
P97	RxD4 input	PS3_7=0	–	–	PD9_7=0

NOTES:

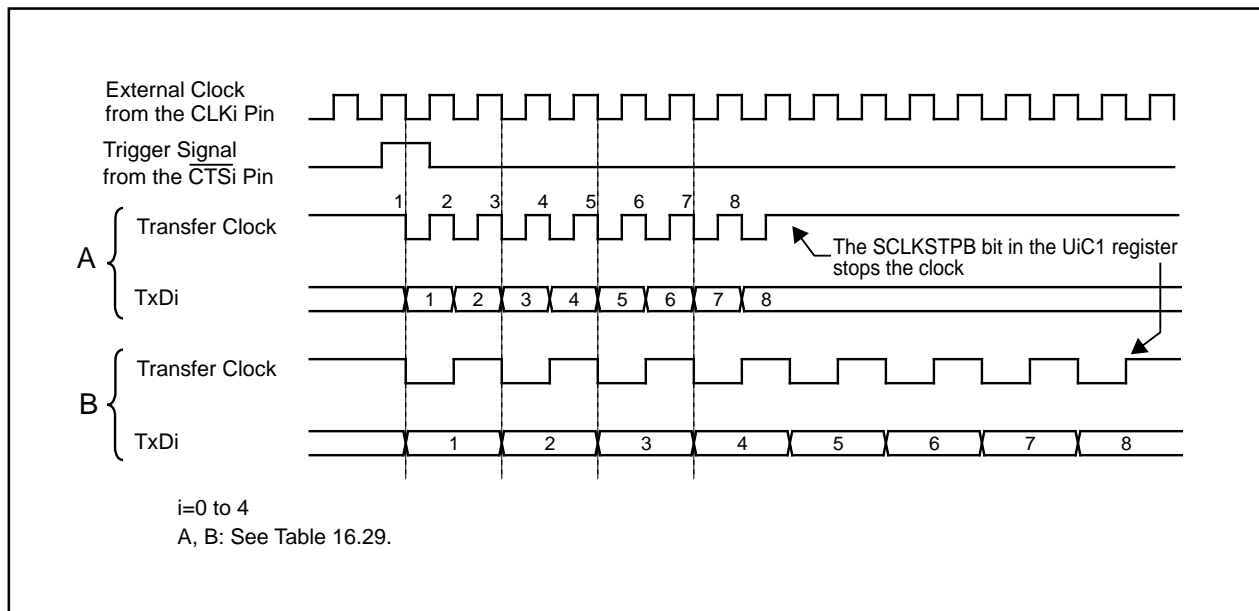
1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.
2. CTS input is used for a trigger signal input.

To generate the internal clock synchronized with the external clock, set the SU1HIM bit in the UiSMR2 register ( $i=0$  to 4) and the SCLKDIV bit in the UiSMR register to values shown in Table 16.29. Then apply a trigger signal to the  $\overline{\text{CTS}}_i$  pin. Either the same clock cycle as the external clock or external clock divided by two can be selected as the transfer clock. The SCLKSTPB bit in the UiC1 register controls the transfer clock. Set the SCLKSTPB bit accordingly, to start or stop the transfer clock during an external clock operation. Figure 16.27 shows an example of the clock-divided synchronous function.

**Table 16.29 Clock-Divided Synchronous Function Select**

SCLKDIV Bit in UiSMR Register	SU1HIM Bit in UiSMR2 Register	Clock-Divided Synchronous Function	Example of Waveform
0	0	Not synchronized	-
0	1	Same division as the external clock	A in Figure 16.27
1	0 or 1	Same division as the external clock divided by 2	B in Figure 16.27

$i=0$  to 4



**Figure 16.27 Clock-Divided Synchronous Function**

## 16.6 Special Mode 4 (IE Mode)

In IE mode, devices connected with the IEBus can communicate in UART mode.

Table 16.30 lists register settings. Tables 16.31 to 16.33 list pin settings.

**Table 16.30 Register Settings in IE Mode**

Register	Bit	Function
UiTB	8 to 0	Set transmit data
UiRB	8 to 0	Received data can be read
	OER, FER, PER, SUM	Error flags
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "1102"
	CKDIR	Select the internal clock or external clock
	STPS	Set to "0"
	PRY	Disabled because the PRYE bit is set to "0"
	PRYE	Set to "0"
	IOPOL	Select TxD and RxD I/O polarity
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Select output format of the TxDi pin
	CKPOL	Set to "0"
	UFORM	Set to "0"
UiC1	TE	Set to "1" to enable data transmission
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM, UiLCH, SCLKSTPB	Set to "0"
UiSMR	3 to 0	Set to "00002"
	ABSCS	Select bus conflict detect sampling timing
	ACSE	Set to "1" to automatically clear the transmit enable bit
	SSS	Select transmit start condition
	SCLKDIV	Set to "0"
UiSMR2	7 to 0	Set to "0016"
UiSMR3	7 to 0	Set to "0016"
UiSMR4	7 to 0	Set to "0016"
IFSR	IFSR6, IFSR7	Select how the bus conflict interrupt occurs

i=0 to 4

**Table 16.31 Pin Settings in IE Mode (1)**

Port	Function	Setting	
		PS0 Register	PD6 Register
P61	CLK0 input	PS0_1=0	PD6_1=0
	CLK0 output	PS0_1=1	–
P62	RxD0 input	PS0_2=0	PD6_2=0
P63	TxD0 output	PS0_3=1	–
P65	CLK1 input	PS0_5=0	PD6_5=0
	CLK1 output	PS0_5=1	–
P66	RxD1 input	PS0_6=0	PD6_6=0
P67	TxD1 output	PS0_7=1	–

**Table 16.32 Pin Settings (2)**

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70 <sup>(1)</sup>	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
P71 <sup>(1)</sup>	RxD2 input	PS1_1=0	–	–	PD7_1=0
P72	CLK2 input	PS1_2=0	–	–	PD7_2=0
	CLK2 output	PS1_2=1	PSL1_2=0	PSC_2=0	–

NOTE:

1. P70 and P71 are ports for the N-channel open drain output.

**Table 16.33 Pin Settings (3)**

Port	Function	Setting			
		PS3 Register <sup>(1)</sup>	PSL3 Register	PSC3 Register	PD9 Register <sup>(1)</sup>
P90	CLK3 input	PS3_0=0	–	–	PD9_0=0
	CLK3 output	PS3_0=1	–	–	–
P91	RxD3 input	PS3_1=0	–	–	PD9_1=0
P92	TxD3 output	PS3_2=1	PSL3_2=0	–	–
P95	CLK4 input	PS3_5=0	PSL3_5=0	–	PD9_5=0
	CLK4 output	PS3_5=1	–	–	–
P96	TxD4 output	PS3_6=1	–	PSC3_6=0	–
P97	RxD4 input	PS3_7=0	–	–	PD9_7=0

NOTE:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

If the output signal level of the TxDi pin (i=0 to 4) differs from the input signal level of the RxDi pin, an interrupt request is generated.

UART0 and UART3 are assigned software interrupt number 40. UART1 and UART4 are assigned number 41. When using the bus conflict detect function of UART0 or UART3, of UART1 or UART4, set the IFSR6 bit and the IFSR7 bit in the IFSR register accordingly.

When the ABSCS bit in the UiSMR register is set to "0" (rising edge of the transfer clock), it is determined, on the rising edge of the transfer clock, if the output level of the TxD pin and the input level of the RxD pin match. When the ABSCS bit is set to "1" (timer Aj underflow), it is determined when the timer Aj (timer A3 in UART0, timer A4 in UART1, timer A0 in UART2, timer A3 in UART3, the timer A4 in UART4) counter overflows. Use the timer Aj in one-shot timer mode.

When the ACSE bit in the UiSMR register is set to "1" (automatic clear at bus conflict) and the IR bit in the BCNiIC register to "1" (discrepancy detected), the TE bit in the UiC1 register is set to "0" (transmit disabled).

When the SSS bit in the UiSMR register is set to "1" (synchronized with RxDi), data is transmitted from the TxDi pin on the falling edge of the RxDi pin. Figure 16.28 shows bits associated with the bus conflict detect function.

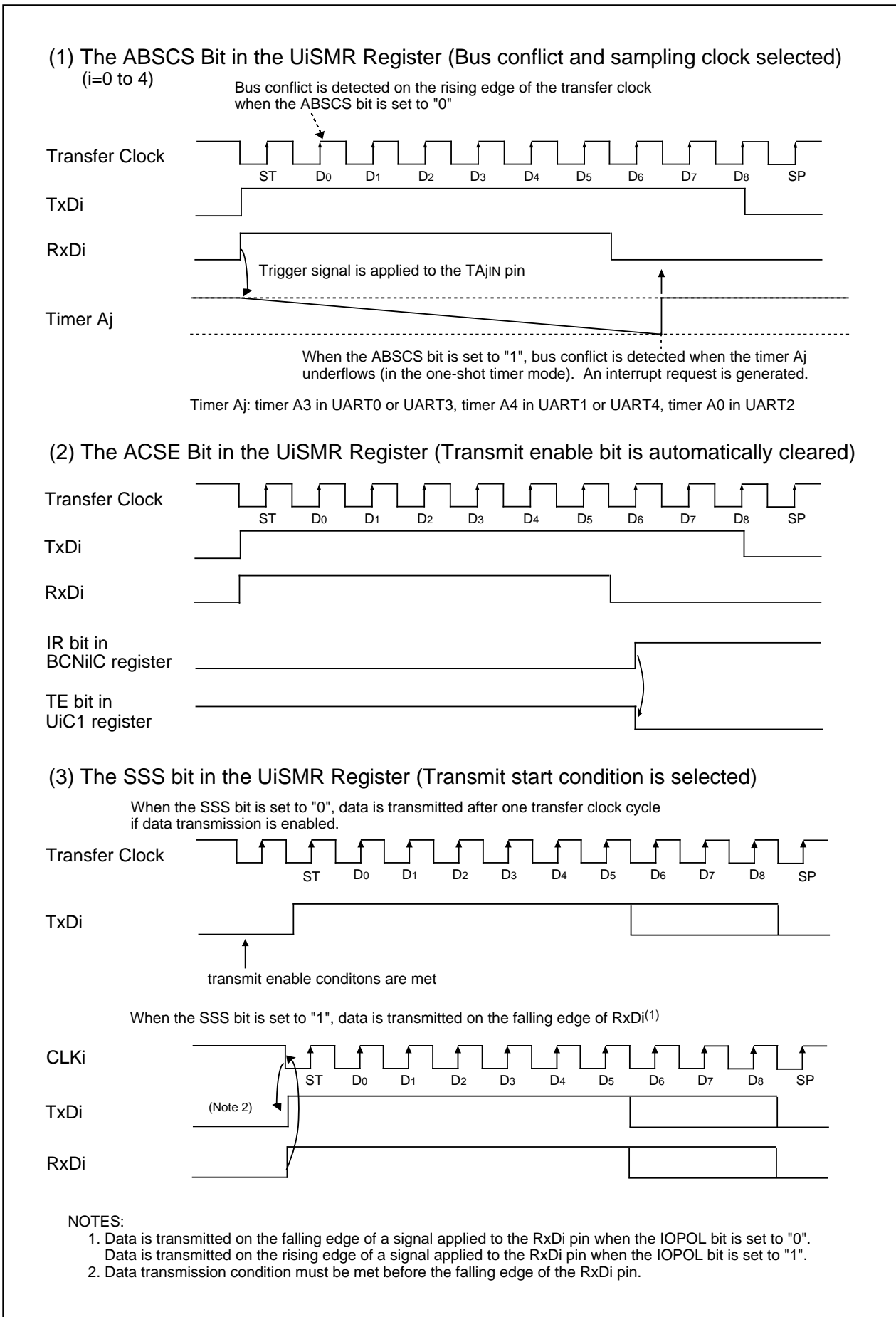


Figure 16.28 Bit Function Related Bus Conflict Detection



## 16.7 Special Mode 5 (SIM Mode)

In SIM mode, SIM interface devices can communicate in UART mode. Both direct and inverse formats are available and a low-level ("L") signal output can be provided from the TxDi pin (i=0 to 4) when a parity error is detected.

Table 16.34 lists specifications of SIM mode. Table 16.35 lists register settings. Tables 16.36 to 16.38 list pin settings.

**Table 16.34 SIM Mode Specifications**

Item	Specification
Transfer Data Format	<ul style="list-style-type: none"> <li>• Transfer data: 8-bit UART mode</li> <li>• In direct format</li> <li>Parity: Even</li> <li>Data logic: Direct</li> <li>Transfer format: LSB first</li> <li>• One stop bit</li> <li>• In inverse format</li> <li>Parity: Odd</li> <li>Data logic: Inverse</li> <li>Transfer format: MSB first</li> </ul>
Transfer Clock	<ul style="list-style-type: none"> <li>• The CKDIR bit in the UiMR register (i=0 to 4) is "0" (internal clock selected):  <math>f_i/16(m+1)^{f_i}</math> <math>f_i = f_1, f_8, f_{2n}^{(2)}</math> <math>m</math>: setting value of the UiBRG register, 00<sub>16</sub> to FF<sub>16</sub></li> <li>Do not set the CKDIR bit to "1" (external clock selected)</li> </ul>
Transmit/Receive Control	The CRD bit in the UiC0 register is set to "1" ( $\overline{\text{CTS}}$ , $\overline{\text{RTS}}$ function disabled)
Other Setting Items	The UiIRS bit in the UiC1 register is set to "1" (transmission completed)
Transmit Start Condition	To start transmitting, the following requirements must be met: <ul style="list-style-type: none"> <li>- Set the TE bit in the UiC1 register to "1" (transmit enabled)</li> <li>- Set the TI bit in the UiC1 register to "0" (data in the UiTB register)</li> </ul>
Receive Start Condition	To start receiving, the following requirements must be met: <ul style="list-style-type: none"> <li>- Set the RE bit in the UiC1 register to "1" (receive enabled)</li> <li>- Detect the start bit</li> </ul>
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• While transmitting, <ul style="list-style-type: none"> <li>-The UiIRS bit is set to "1" (transmission completed): when data transmission from the UARTi transfer register is completed</li> </ul> </li> <li>• While receiving, <ul style="list-style-type: none"> <li>when data is transferred from the UARTi receive register to the UiRB register (reception completed)</li> </ul> </li> </ul>
Error Detection	<ul style="list-style-type: none"> <li>• Overrun error<sup>(1)</sup> <ul style="list-style-type: none"> <li>This error occurs when the eighth bit of the next data is received before reading the UiRB register</li> </ul> </li> <li>• Framing error <ul style="list-style-type: none"> <li>This error occurs when the number of the stop bit set is not detected</li> </ul> </li> <li>• Parity error <ul style="list-style-type: none"> <li>This error occurs when the number of "1" in parity bit and character bits differs from the number set</li> </ul> </li> <li>• Error sum flag <ul style="list-style-type: none"> <li>The SUM bit is set to "1" when an overrun error, framing error or parity error occurs</li> </ul> </li> </ul>

**NOTES:**

1. If an overrun error occurs, the UiRB register is indeterminate. The IR bit setting in the SiRIC register does not change to "1" (interrupt requested).
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2<sup>n</sup> (n=1 to 15).

**Table 16.35 Register Settings in SIM Mode**

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data can be read
	OER, FER, PER, SUM	Error flags
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "1012"
	CKDIR	Set to "0"
	STPS	Set to "0"
	PRY	Set to "1" for direct format or "0" for inverse format
	PRYE	Set to "1"
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Set to "1"
	CKPOL	Set to "0"
	UFORM	Set to "0" for direct format or "1" for inverse format
UiC1	TE	Set to "1" to enable data transmission
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Set to "1"
	UiRRM	Set to "0"
	UiLCH	Set to "0" for direct format or "1" for inverse format
	UiERE	Set to "1"
UiSMR	7 to 0	Set to "0016"
UiSMR2	7 to 0	Set to "0016"
UiSMR3	7 to 0	Set to "0016"
UiSMR4	7 to 0	Set to "0016"

i=0 to 4

**Table 16.36 Pin Settings in SIM Mode (1)**

Port	Function	Setting	
		PS0 Register	PD6 Register
P62	RxD0 input	PS0_2=0	PD6_2=0
P63	TxD0 output	PS0_3=1	–
P66	RxD1 input	PS0_6=0	PD6_6=0
P67	TxD1 output	PS0_7=1	–

**Table 16.37 Pin Settings (2)**

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70 <sup>(1)</sup>	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
P71 <sup>(1)</sup>	RxD2 input	PS1_1=0	–	–	PD7_1=0

NOTE:

1. P70 and P71 are ports for the N-channel open drain output.

**Table 16.38 Pin Settings (3)**

Port	Function	Setting			
		PS3 Register <sup>(1)</sup>	PSL3 Register	PSC3 Register	PD9 Register <sup>(1)</sup>
P91	RxD3 input	PS3_1=0	–		PD9_1=0
P92	TxD3 output	PS3_2=1	PSL3_2=0		–
P96	TxD4 output	PS3_6=1	–	PSC3_6=0	–
P97	RxD4 input	PS3_7=0	–		PD9_7=0

NOTE:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

Figure 16.29 shows an example of a SIM interface operation. Figure 16.30 shows an example of a SIM interface connection. Connect the TxDi pin to the RxDi pin for a pull-up.

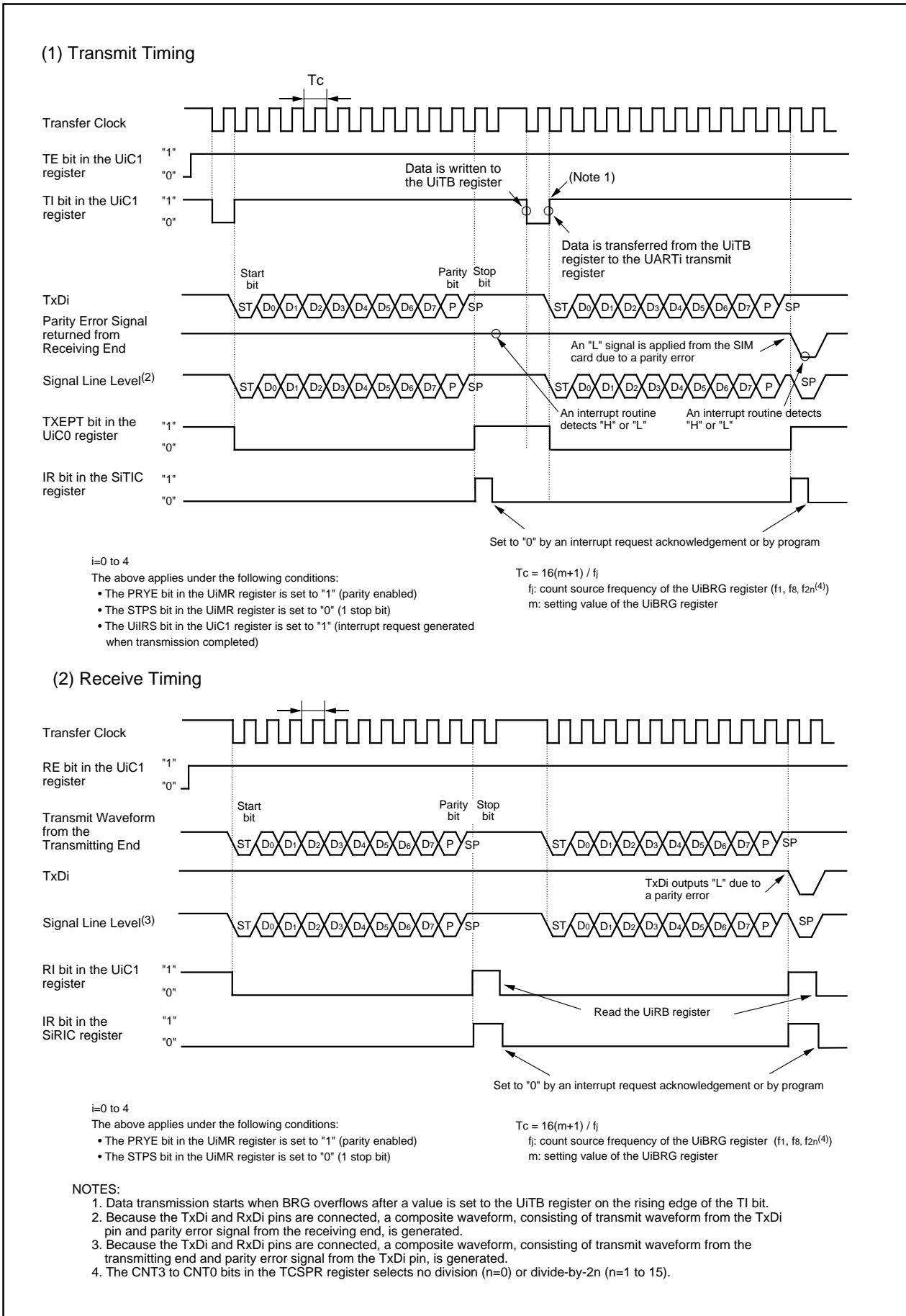


Figure 16.29 SIM Interface Operation

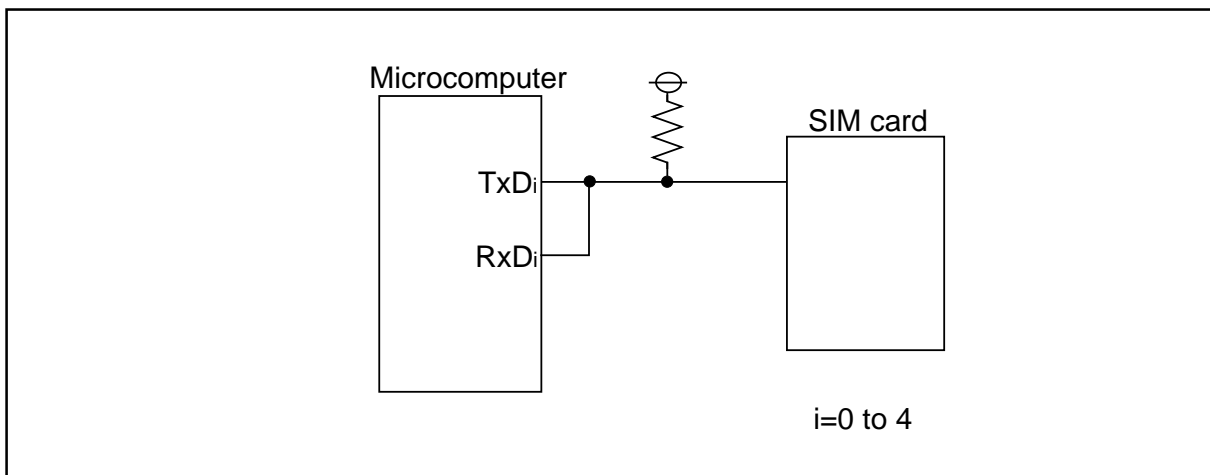


Figure 16.30 SIM Interface Connection

### 16.7.1 Parity Error Signal

#### 16.7.1.1 Parity Error Signal Output Function

When the UiERE bit in the UiC1 register (i=0 to 4) is set to "1" (output), the parity error signal output can be provided. The parity error signal output is provided when a parity error is detected upon receiving data. A low-level ("L") signal output is provided from the TxDi pin in the timing shown in Figure 16.31. When reading the UiRB register during a parity error output, the PER bit in the UiRB register is set to "0" (no error occurs) and a high-level ("H") signal output is again provided simultaneously.

#### 16.7.1.2 Parity Error Signal

To determine whether the parity error signal is output, the port that shares a pin with the RxDi pin is read by using an end-of-transmit interrupt routine.

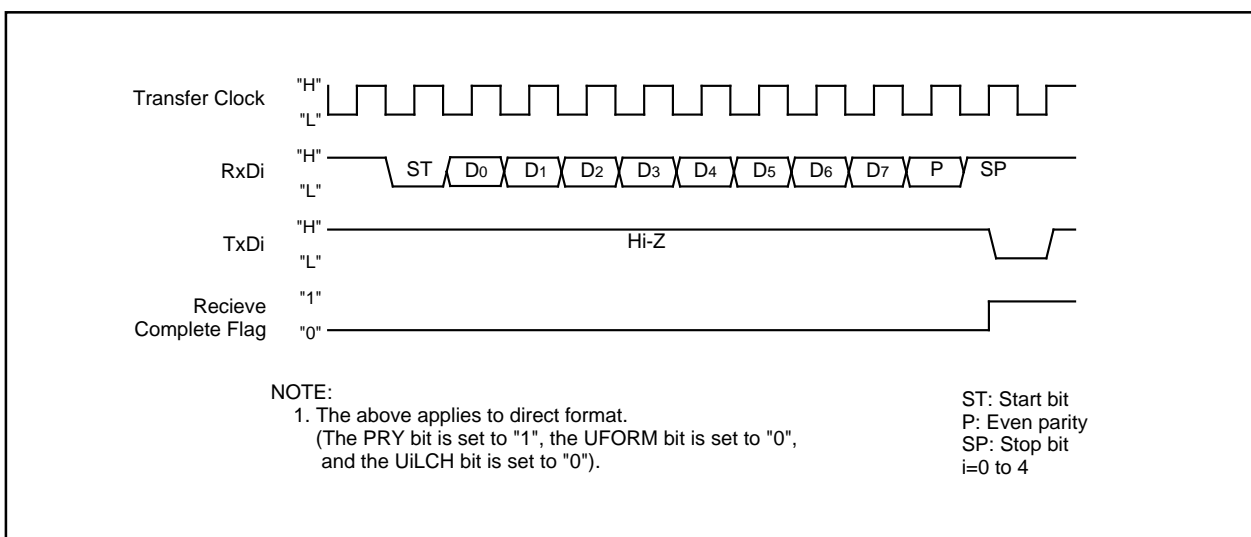


Figure 16.31 Parity Error Signal Output Timing (LSB First)

## 16.7.2 Format

### 16.7.2.1 Direct Format

Set the PRYE bit in the UiMR register ( $i=0$  to 4) to "1" (parity enabled), the PRY bit to "1" (even parity), the UFORM bit in the UiC0 register to "0" (LSB first) and the UiLCH bit in the UiC1 register to "0" (not inverted). When data are transmitted, data set in the UiTB register are transmitted with the even-numbered parity, starting from D0. When data are received, received data are stored in the UiRB register, starting from D0. The even-numbered parity determines whether a parity error occurs.

### 16.7.2.2 Inverse Format

Set the PRYE bit to "1", the PRY bit to "0" (odd parity), the UFORM bit to "1" (MSB first) and the UiLCH bit to "1" (inverted). When data are transmitted, values set in the UiTB 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 UiRB register, starting from D7. The odd-numbered parity determines whether a parity error occurs.

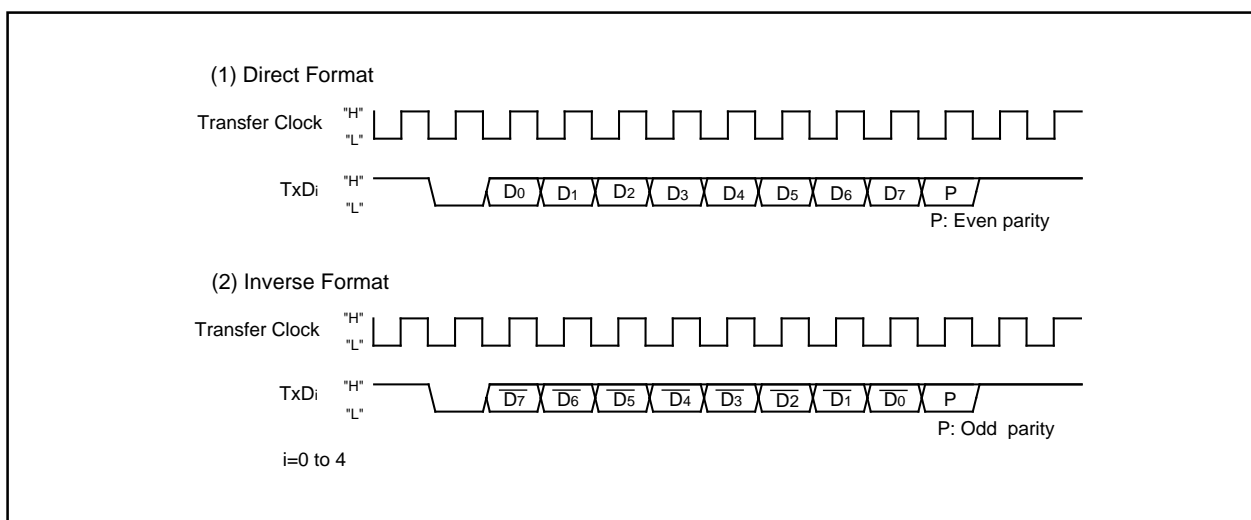


Figure 16.32 SIM Interface Format

## 17. A/D Converter

The A/D converter consists of one 10-bit successive approximation A/D converter with a capacitive coupling amplifier.

The result of an A/D conversion is stored into the A/D registers corresponding to selected pins. It is stored into the AD00 register only when DMAC operating mode is entered.

Table 17.1 lists specifications of the A/D converter. Figure 17.1 shows a block diagram of the A/D converter. Figures 17.2 to 17.6 show registers associated with the A/D converter.

**Table 17.1 A/D Converter Specifications**

Item	Specification
A/D Conversion Method	Successive approximation (with a capacitive coupling amplifier)
Analog Input Voltage <sup>(1)</sup>	0V to AVCC (VCC1)
Operating Clock, $\phi_{AD}$ <sup>(2)</sup>	fAD, fAD/2, fAD/3, fAD/4, fAD/6, fAD/8
Resolution	8 bits or 10 bits
Operating Mode	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, repeat sweep mode 1
Analog Input Pins <sup>(3)</sup>	10 pins 8 pins for AN0 to AN7 2 extended input pins (ANEX0 and ANEX1)
A/D Conversion Start Condition	<ul style="list-style-type: none"> <li>• Software trigger The ADST bit in the AD0CON0 register is set to "1" (A/D conversion started) by program</li> <li>• External trigger (re-trigger is enabled) When a falling edge is applied to the <math>\overline{ADTRG}</math> pin after the ADST bit is set to "1" by program</li> <li>• Hardware trigger (re-trigger is enabled) The timer B2 interrupt request of the three-phase motor control timer functions (after the ICTB2 counter completes counting) is generated after the ADST bit is set to "1" by program</li> </ul>
Conversion Rate Per Pin	<ul style="list-style-type: none"> <li>• Without the sample and hold function 8-bit resolution : 49 <math>\phi_{AD}</math> cycles 10-bit resolution : 59 <math>\phi_{AD}</math> cycles</li> <li>• With the sample and hold function 8-bit resolution : 28 <math>\phi_{AD}</math> cycles 10-bit resolution : 33 <math>\phi_{AD}</math> cycles</li> </ul>

**NOTES:**

1. Analog input voltage is not affected by the sample and hold function status.
2.  $\phi_{AD}$  frequency must be under 16 MHz when VCC1=5V.  
 $\phi_{AD}$  frequency must be under 10 MHz when VCC1=3.3V.  
Without the sample and hold function, the  $\phi_{AD}$  frequency is 250 kHz or more.  
With the sample and hold function, the  $\phi_{AD}$  frequency is 1 MHz or more.
3. AVCC=VREF=VCC1, A/D input voltage (for AN0 to AN7, ANEX0, and ANEX1)  $\leq$  VCC1.

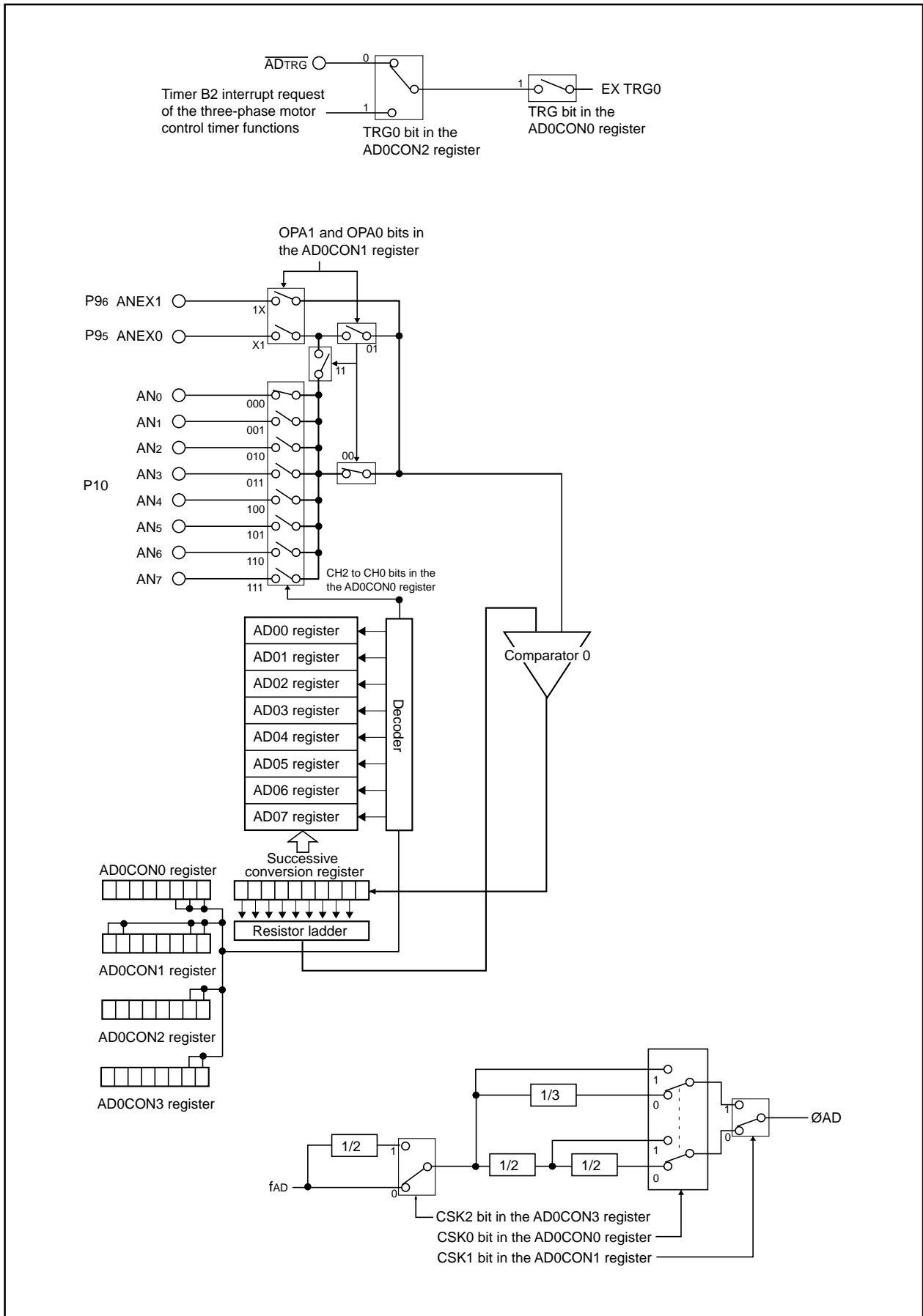


Figure 17.1 A/D Converter Block Diagram



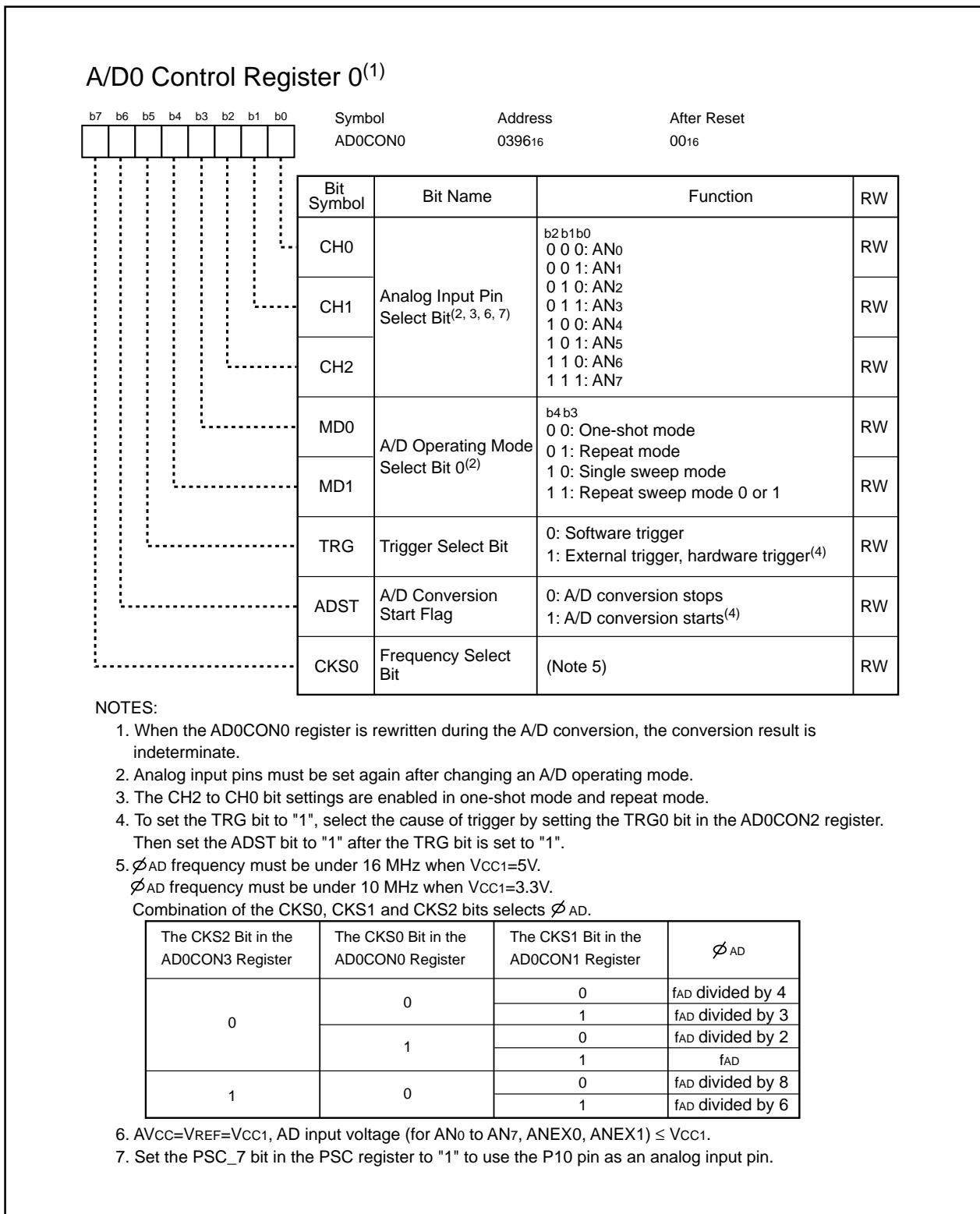


Figure 17.2 AD0CON0 Register

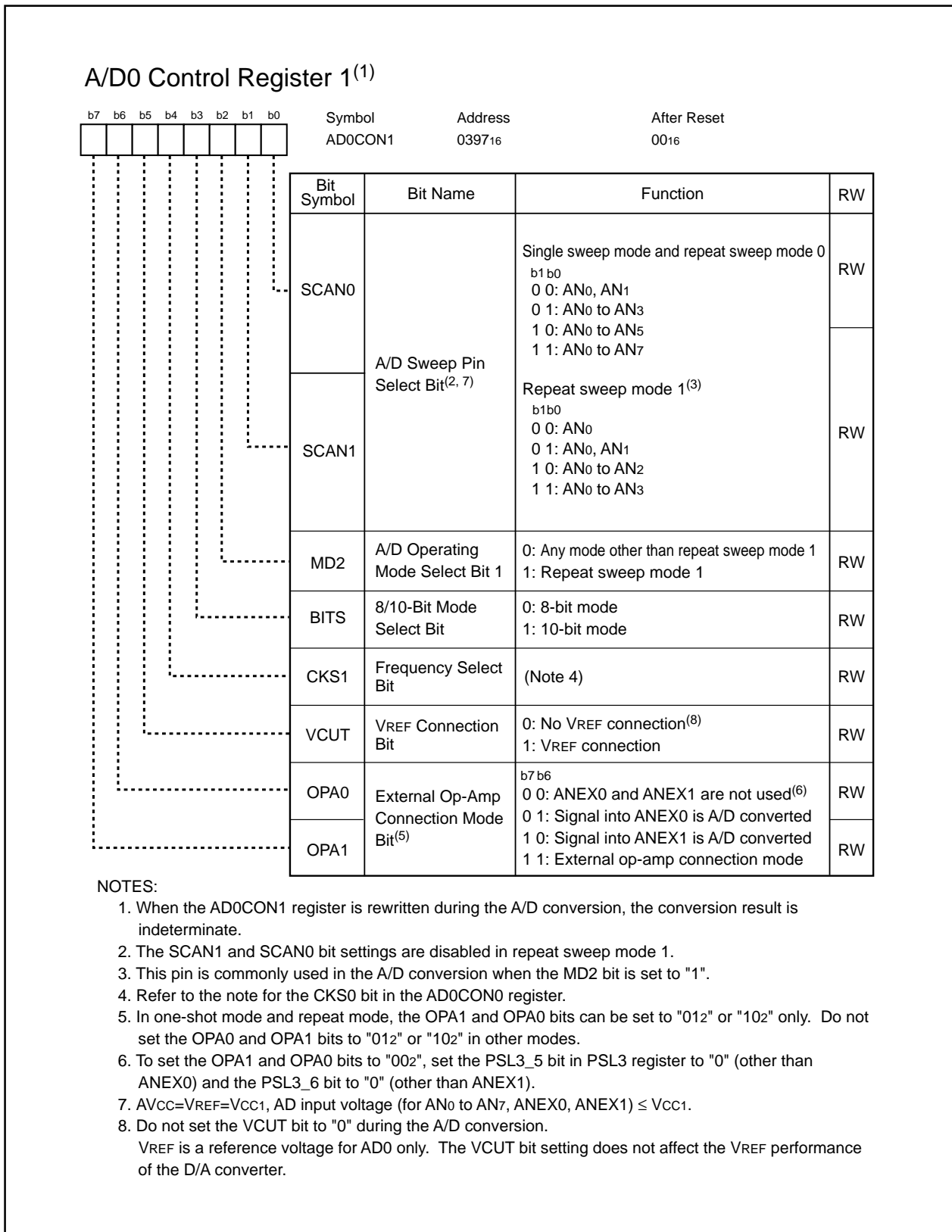


Figure 17.3 AD0CON1 Register

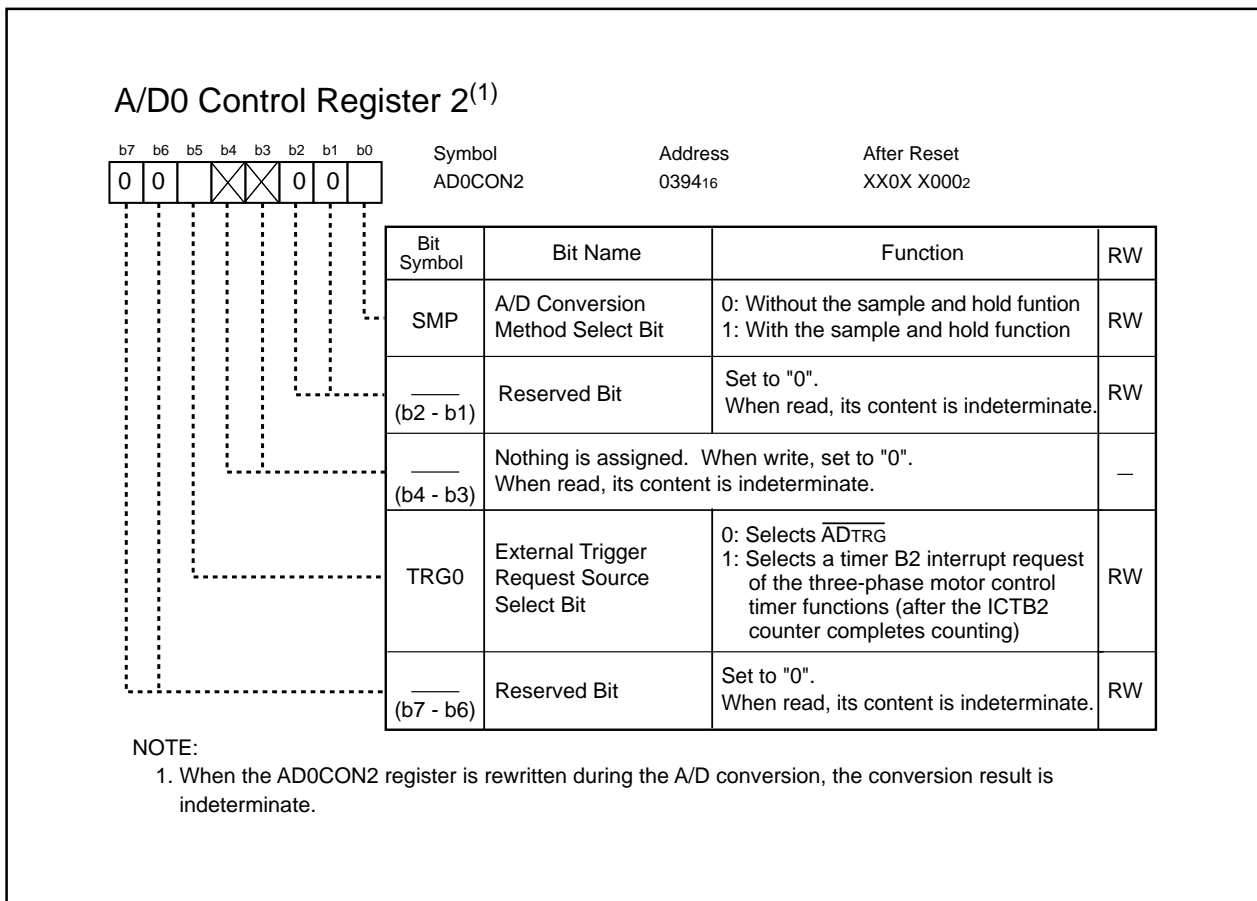
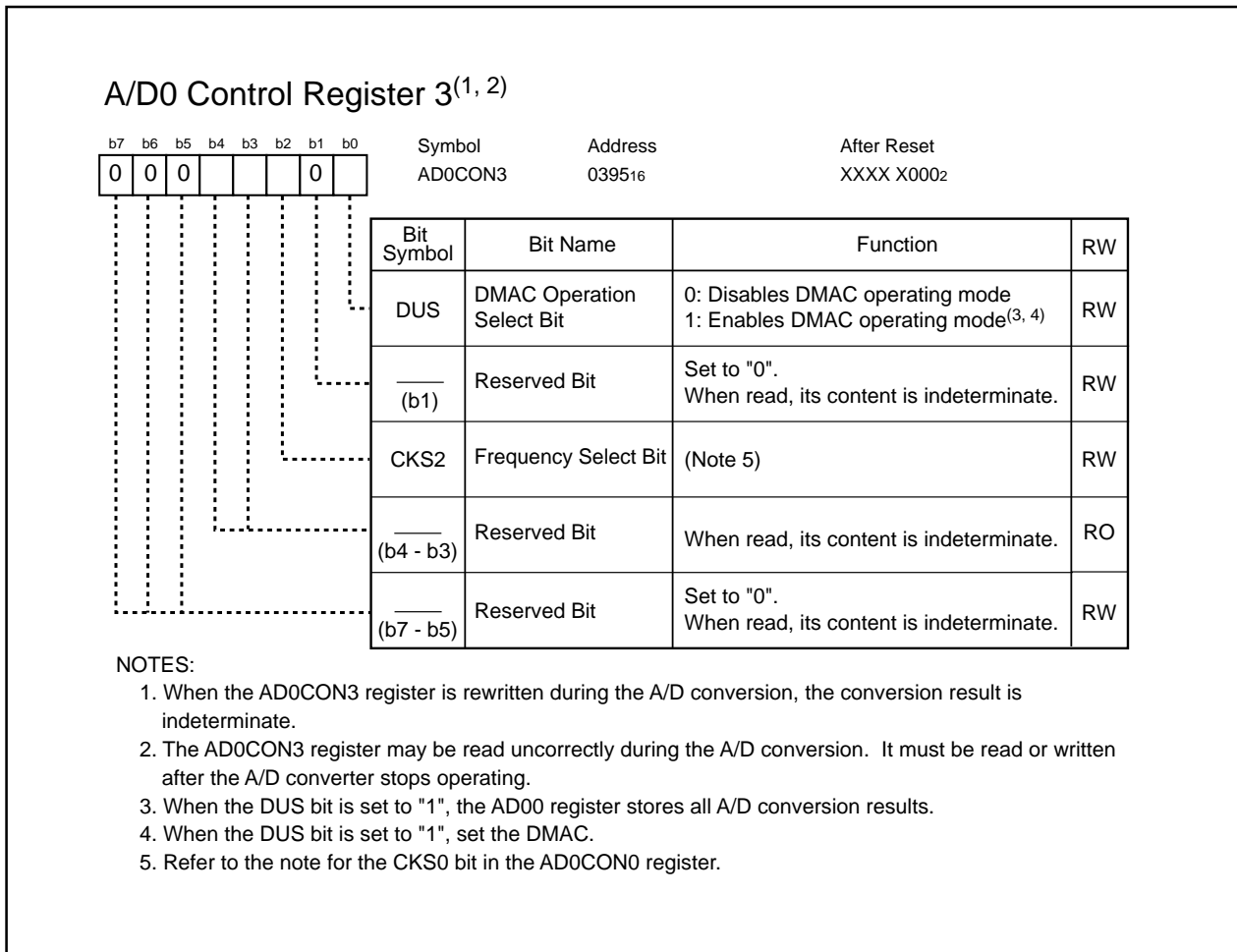
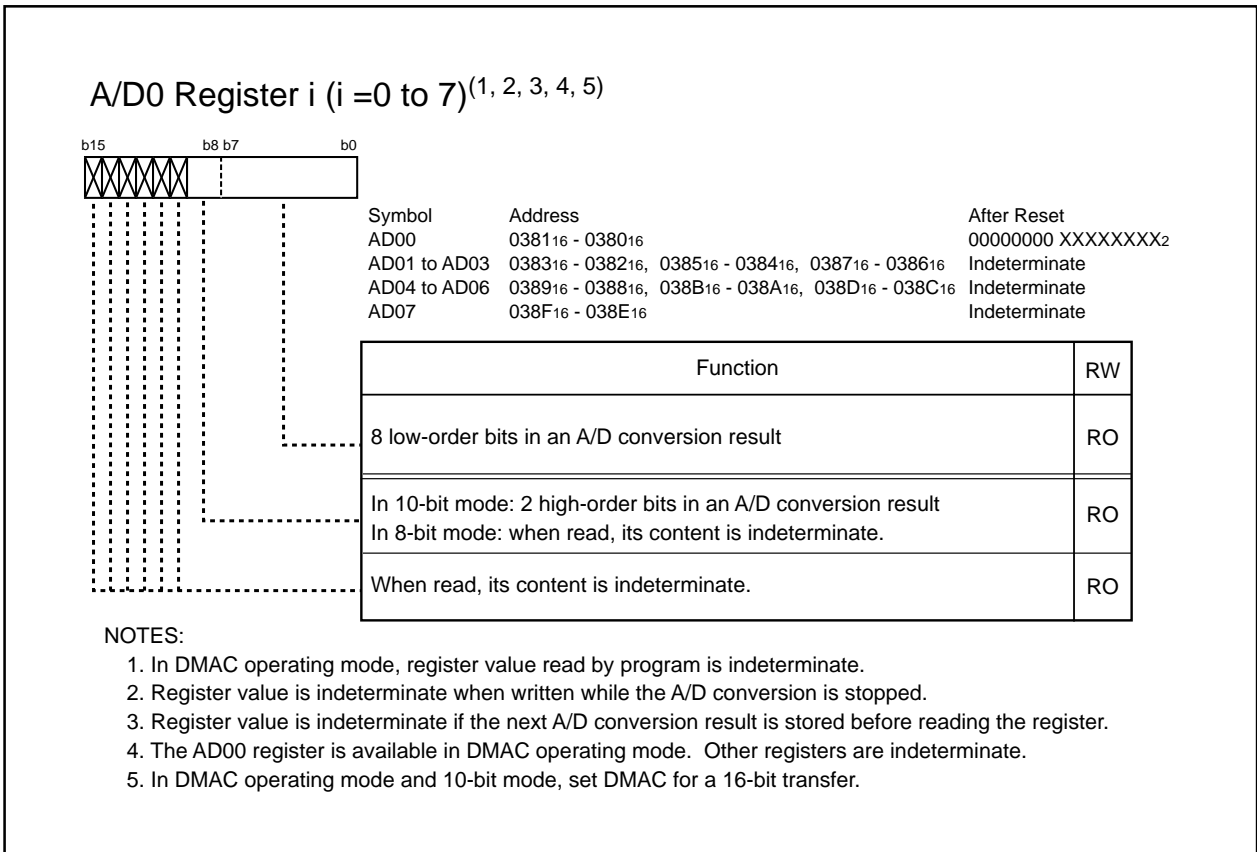


Figure 17.4 AD0CON2 Register



**Figure 17.5 AD0CON3 Register**



**Figure 17.6 AD00 to AD07 Registers**

## 17.1 Mode Description

### 17.1.1 One-shot Mode

In one-shot mode, analog voltage applied to a selected pin is converted to a digital code once. Table 17.2 lists specifications of one-shot mode.

**Table 17.2 One-shot Mode Specifications**

Item	Specification
Function	The CH2 to CH0 bits in the AD0CON0 register and the OPA1 and OPA0 bits in the AD0CON1 register select a pin. Analog voltage applied to the pin is converted to a digital code once
Start Condition	<ul style="list-style-type: none"> <li>• When the TRG bit in the AD0CON0 register is set to "0" (software trigger), the ADST bit in the AD0CON0 register is set to "1" (A/D conversion starts) by program</li> <li>• When the TRG bit is set to "1" (external trigger, hardware trigger): <ul style="list-style-type: none"> <li>- a falling edge is applied to the <math>\overline{ADTRG}</math> pin after the ADST bit is set to "1" by program</li> <li>- The timer B2 interrupt request of three-phase motor control timer functions (after the ICTB2 register counter completes counting) is generated after the ADST bit is set to "1" by program</li> </ul> </li> </ul>
Stop Condition	<ul style="list-style-type: none"> <li>• A/D conversion is completed (the ADST bit is set to "0" when the software trigger is selected)</li> <li>• The ADST bit is set to "0" (A/D conversion stopped) by program</li> </ul>
Interrupt Request Generation Timing	A/D conversion is completed
Analog Voltage Input Pins	Select one pin from AN0 to AN7, ANEX0, or ANEX1
Reading of A/D Conversion Result	<ul style="list-style-type: none"> <li>• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), the microcomputer reads the AD0j register (j=0 to 7) corresponding to selected pin</li> <li>• When the DUS bit is set to "1" (DMAC operating mode enabled), do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to <b>12. DMAC</b> for DMAC settings</li> </ul>

### 17.1.2 Repeat Mode

In repeat mode, analog voltage applied to a selected pin is repeatedly converted to a digital code. Table 17.3 lists specifications of repeat mode.

**Table 17.3 Repeat Mode Specifications**

Item	Specification
Function	The CH2 to CH0 bits in the AD0CON0 register and the OPA1 and OPA0 bits in the AD0CON1 register select a pin. Analog voltage applied to the pin is repeatedly converted to a digital code
Start Condition	Same as one-shot mode
Stop Condition	The ADST bit in the AD0CON0 register is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), no interrupt request is generated.</li> <li>• When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed.</li> </ul>
Analog Voltage Input Pins	Select one pin from AN0 to AN7, ANEX0, or ANEX1
Reading of A/D Conversion Result	<ul style="list-style-type: none"> <li>• When the DUS bit is set to "0", the microcomputer reads the AD0j register (j=0 to 7) corresponding to the selected pin.</li> <li>• When DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space.</li> </ul> <p>Refer to <b>12. DMAC</b> for DMAC settings</p>

### 17.1.3 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 17.4 lists specifications of single sweep mode.

**Table 17.4 Single Sweep Mode Specifications**

Item	Specification
Function	The SCAN1 and SCAN0 bits in the AD0CON1 register select pins. Analog voltage applied to the pin is converted one-by-one to a digital code
Start Condition	Same as one-shot mode
Stop Condition	Same as one-shot mode
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), an interrupt request is generated after a sweep is completed.</li> <li>• When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed</li> </ul>
Analog Voltage Input Pins	Select from AN0 and AN1 (2 pins), AN0 to AN3 (4 pins), AN0 to AN5 (6 pins) or AN0 to AN7 (8 pins)
Reading of A/D Conversion Result	<ul style="list-style-type: none"> <li>• When the DUS bit is set to "0", the microcomputer reads the AD0j register (j=0 to 7) corresponding to selected pins</li> <li>• When DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to <b>12. DMAC</b> for DMAC settings</li> </ul>



### 17.1.4 Repeat Sweep Mode 0

In repeat sweep mode 0, analog voltage applied to selected pins is repeatedly converted to a digital code. Table 17.5 lists specifications of repeat sweep mode 0.

**Table 17.5 Repeat Sweep Mode 0 Specifications**

Item	Specification
Function	The SCAN1 and SCAN0 bits in the AD0CON1 register select pins. Analog voltage applied to the pins is repeatedly converted to a digital code
Start Condition	Same as one-shot mode
Stop Condition	The ADST bit in the AD0CON0 register is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), no interrupt request is generated</li> <li>• When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed</li> </ul>
Analog Voltage Input Pins	Select from AN0 and AN1 (2 pins), AN0 to AN3 (4 pins), AN0 to AN5 (6 pins) or AN0 to AN7 (8 pins)
Reading of A/D Conversion Result	<ul style="list-style-type: none"> <li>• When the DUS bit is set to "0", the microcomputer reads the AD0j register (j=0 to 7) corresponding to selected pins</li> <li>• When the DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to <b>12. DMAC</b> for DMAC settings</li> </ul>

### 17.1.5 Repeat Sweep Mode 1

In repeat sweep mode 1, analog voltage selectively applied to eight pins is repeatedly converted to a digital code. Table 17.6 lists specifications of repeat sweep mode 1.

**Table 17.6 Repeat Sweep Mode 1 Specifications**

Item	Specification
Function	The SCAN1 and SCAN0 bits in the AD0CON1 register select 8 pins. Analog voltage selectively applied to 8 pins is repeatedly converted to a digital code e.g., When AN <sub>i</sub> is selected (i = none, 0, 2, 15), analog voltage is converted to a digital code in the following order: AN <sub>0</sub> → AN <sub>1</sub> → AN <sub>0</sub> → AN <sub>2</sub> → AN <sub>0</sub> → AN <sub>3</sub> ..... etc.
Start Condition	Same as one-shot mode (Any trigger generated during an A/D conversion is invalid)
Stop Condition	The ADST bit is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	<ul style="list-style-type: none"> <li>• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), no interrupt request is generated</li> <li>• When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed</li> </ul>
Analog Voltage Input Pins	AN <sub>0</sub> to AN <sub>7</sub> (8 pins)
Prioritized Pins	AN <sub>0</sub> (1 pin), AN <sub>0</sub> and AN <sub>1</sub> (2 pins), AN <sub>0</sub> to AN <sub>2</sub> (3 pins) or AN <sub>0</sub> to AN <sub>3</sub> (4 pins)
Reading of A/D Conversion Result	<ul style="list-style-type: none"> <li>• When the DUS bit is set to "0", the microcomputer reads the AD0<sub>j</sub> register (j=0 to 7) corresponding to selected pins</li> <li>• When the DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to <b>12. DMAC</b> for DMAC settings</li> </ul>

## 17.2 Functions

### 17.2.1 Resolution Select Function

The BITS bit in the AD0CON1 register determines the resolution. When the BITS bit is set to "1" (10-bit precision), the A/D conversion result is stored into bits 9 to 0 in the AD0j register (j = 0 to 7). When the BITS bit is set to "0" (8-bit precision), the A/D conversion result is stored into bits 7 to 0 in the AD0j register.

### 17.2.2 Sample and Hold Function

When the SMP bit in the AD0CON2 register is set to "1" (with the sample and hold function), A/D conversion rate per pin increases to  $28 \varnothing_{AD}$  cycles for 8-bit resolution and  $33 \varnothing_{AD}$  cycles for 10-bit resolution. The sample and hold function is available in all operating modes. Start the A/D conversion after selecting whether the sample and hold function is to be used or not.

### 17.2.3 Trigger Select Function

The TRG bit in the AD0CON0 register and the TRG0 bit in the AD0CON2 register select the trigger to start the A/D conversion. Table 17.9 lists settings of the trigger select function.

**Table 17.9 Trigger Select Function Settings**

Bit and Setting		Trigger
AD0CON0 Register	AD0CON2 Register	
TRG = 0	-	Software trigger The A/D0 starts the A/D conversion when the ADST bit in the AD0CON0 register is set to "1"
TRG = 1 <sup>(1)</sup>	TRG0 = 0	External trigger <sup>(2)</sup> Falling edge of a signal applied to $\overline{ADTRG}$
	TRG0 = 1	Hardware trigger <sup>(2)</sup> The timer B2 interrupt request of three-phase motor control timer functions (after the ICTB2 counter completes counting)

NOTES:

1. A/D0 starts the A/D conversion when the ADST bit is set to "1" (A/D conversion started) and a trigger is generated.
2. The A/D conversion is restarted if an external trigger or a hardware trigger is inserted during the A/D conversion. (The A/D conversion in process is aborted.)

### 17.2.4 DMAC Operating Mode

DMAC operating mode is available with all operating modes. When the A/D converter is in multi-port single sweep mode or multi-port repeat sweep mode 0, the DMAC operating mode must be used. When the DUS bit in the AD0CON3 register is set to "1" (DMAC operating mode enabled), all A/D conversion results are stored into the AD00 register. DMAC transfers data from the AD00 register to any memory space every time an A/D conversion is completed in each pin. 8-bit DMA transfer must be selected for 8-bit resolution and 16-bit DMA transfer for 10-bit resolution. Refer to **12. DMAC** for instructions.

### 17.2.5 Extended Analog Input Pins

In one-shot mode and repeat mode, the ANEX0 and ANEX1 pins can be used as analog input pins. The OPA1 and OPA0 bits in the AD0CON1 register select which pins to use as analog input pins. An A/D conversion result for the ANEX0 pin is stored into the AD00 register. The result for the ANEX1 pin is stored into the AD01 register, but is stored into the AD00 register when the DUS bit in the AD0CON3 register is set to "1" (DMAC operating mode enabled).

### 17.2.6 External Operating Amplifier (Op-Amp) Connection Mode

In external op-amp connection mode, multiple analog voltage can be amplified by one external op-amp using extended analog input pins ANEX0 and ANEX1.

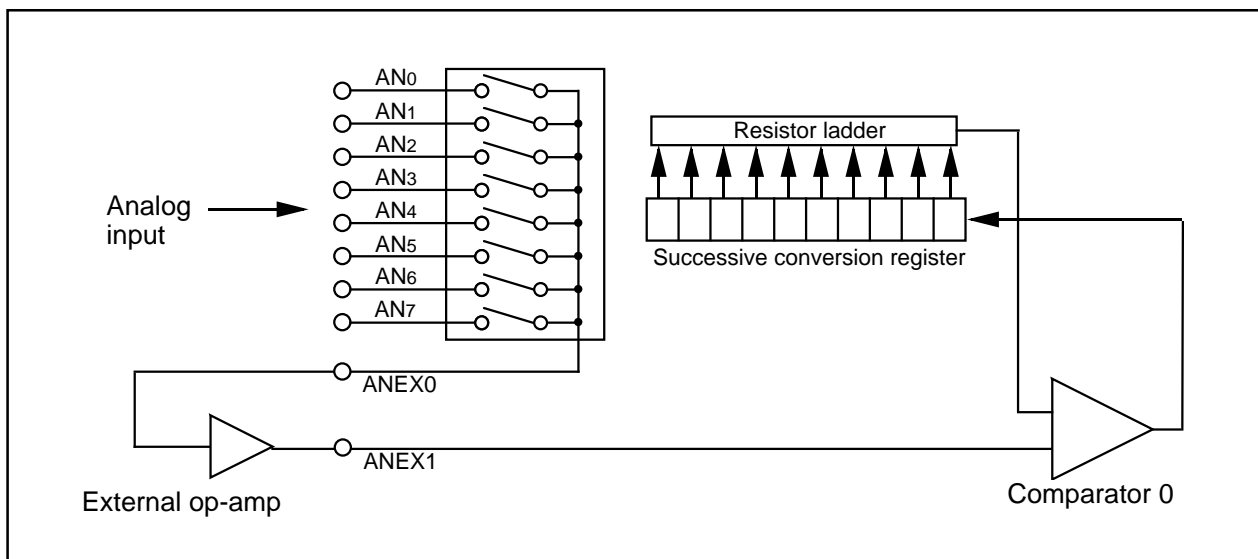
When the OPA1 and OPA0 bits in the AD0CON1 register are set to "112" (external op-amp connection), voltage applied to the AN<sub>0</sub> to AN<sub>7</sub> pins are output from ANEX0. Amplify this output signal by an external op-amp and apply it to ANEX1.

Analog voltage applied to ANEX1 is converted to a digital code and the A/D conversion result is stored into the corresponding AD0<sub>j</sub> register (j=0 to 7). A/D conversion rate varies depending on the response of the external op-amp. The ANEX0 pin cannot be connected to the ANEX1 pin directly.

Figure 17.7 shows an example of an external op-amp connection.

**Table 17.10 Extended Analog Input Pin Settings**

AD0CON1 Register		ANEX0 Function	ANEX1 Function
OPA1 Bit	OPA0 Bit		
0	0	Not used	Not used
0	1	P9 <sub>5</sub> as an analog input	Not used
1	0	Not used	P9 <sub>6</sub> as an analog input
1	1	Output to an external op-amp	Input from an external op-amp



**Figure 17.7 External Op-Amp Connection**

### 17.2.7 Power Consumption Reducing Function

When the A/D converter is not used, the VCUT bit in the AD0CON1 register isolates the resistor ladder of the A/D converter from the reference voltage input pin (VREF). Power consumption is reduced by shutting off any current flow into the resistor ladder from the VREF pin.

When using the A/D converter, set the VCUT bit to "1" (VREF connection) before setting the ADST bit in the AD0CON0 register to "1" (A/D conversion started). Do not set the ADST bit and VCUT bit to "1" simultaneously, nor set the VCUT bit to "0" (no VREF connection) during the A/D conversion. The VCUT bit does not affect the VREF performance of the D/A converter.

### 17.2.8 Output Impedance of Sensor Equivalent Circuit under A/D Conversion

For perfect A/D converter performance, complete internal capacitor (C) charging, shown in Figure 17.8, for the specified period (T) as sampling time. Output Impedance of the sensor equivalent circuit (R<sub>0</sub>) is determined by the following equations:

$$V_C = V_{IN} \left\{ 1 - e^{-\frac{1}{C(R_0 + R)} t} \right\}$$

$$\text{When } t = T, \quad V_C = V_{IN} - \frac{X}{Y} V_{IN} = V_{IN} \left( 1 - \frac{X}{Y} \right)$$

$$e^{-\frac{1}{C(R_0 + R)} T} = \frac{X}{Y}$$

$$-\frac{1}{C(R_0 + R)} T = \ln \frac{X}{Y}$$

$$R_0 = -\frac{T}{C \cdot \ln \frac{X}{Y}} - R$$

where:

V<sub>C</sub> = Voltage between pins

R = Internal resistance of the microcomputer

X = Precision (error) of the A/D converter

Y = Resolution of the A/D converter (1024 in 10-bit mode, and 256 in 8-bit mode)

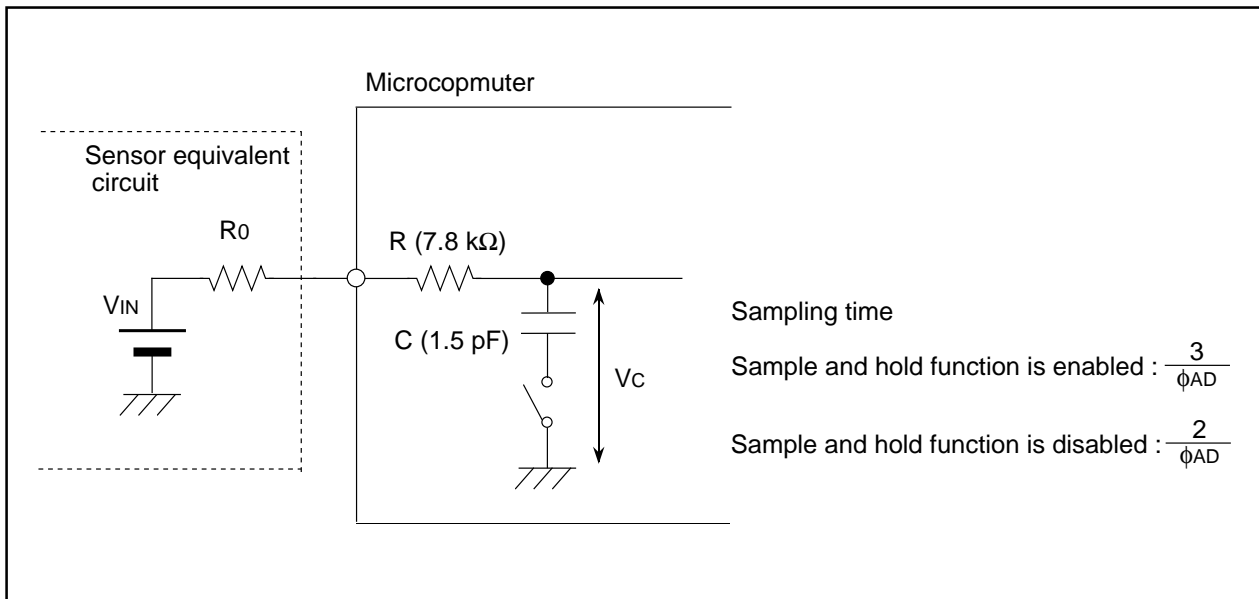
Figure 17.8 shows analog input pin and external sensor equivalent circuit. The impedance (R<sub>0</sub>) can be obtained if the voltage between pins (V<sub>C</sub>) changes from 0 to V<sub>IN</sub> - (0.1/1024) V<sub>IN</sub> in the time (T), when the difference between V<sub>IN</sub> and V<sub>C</sub> becomes 0.1LSB.

(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  $\phi_{AD} = 10$  MHz, T = 0.3  $\mu$ s in the A/D conversion mode with the sample and hold function. Output impedance (R<sub>0</sub>) for sufficiently charging capacitor (C) in the time (T) is determined by the following equation:

Using T = 0.3  $\mu$ s, R = 7.8 k $\Omega$ , C = 1.5 pF, X = 0.1, Y = 1024,

$$R_0 = -\frac{0.3 \times 10^{-6}}{1.5 \times 10^{-12} \cdot \ln \frac{0.1}{1024}} - 7.8 \times 10^3 = 13.9 \times 10^3$$

Thus, the allowable output impedance of the sensor equivalent circuit, making the precision (error) 0.1LSB or less, is approximately 13.9 k $\Omega$  maximum.



**Figure 17.8 Analog Input Pin and External Sensor Equivalent Circuit**

## 18. D/A Converter

The D/A converter consists of two separate 8-bit R-2R ladder D/A converters.

Digital code is converted to an analog voltage when a value is written to the corresponding DAi registers (i=0,1). The DAiE bit in the DACON register determines whether the D/A conversion result output is provided or not. Set the DAiE bit to "1" (output enabled) to disable a pull-up of a corresponding port.

Output analog voltage ( $V$ ) is calculated from value  $n$  ( $n$ =decimal) set in the DAi register.

$$V = \frac{V_{REF} \times n}{256} \quad (n = 0 \text{ to } 255)$$

$V_{REF}$  : reference voltage (not related to VCUT bit setting in the AD0CON1 register)

Table 18.1 lists specifications of the D/A converter. Table 18.2 lists the DA0 and DA1 pin settings. Figure 18.1 shows a block diagram of the D/A converter. Figure 18.2 shows the D/A control register. Figure 18.3 shows a D/A converter equivalent circuit.

When the D/A converter is not used, set the DAi register to "0016" and the DAiE bit to "0" (output disabled).

**Table 18.1 D/A Converter Specifications**

Item	Specification
D/A Conversion Method	R-2R
Resolution	8 bits
Analog Output Pin	2 channels

**Table 18.2 Pin Settings**

Port	Function	Setting		
		PD9 Register <sup>(1)</sup>	PS3 Register <sup>(1)</sup>	PSL3 Register
P93	DA0 output	PD9_3=0	PS3_3=0	PSL3_3=1
P94	DA1 output	PD9_4=0	PS3_4=0	PSL3_4=1

NOTE:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

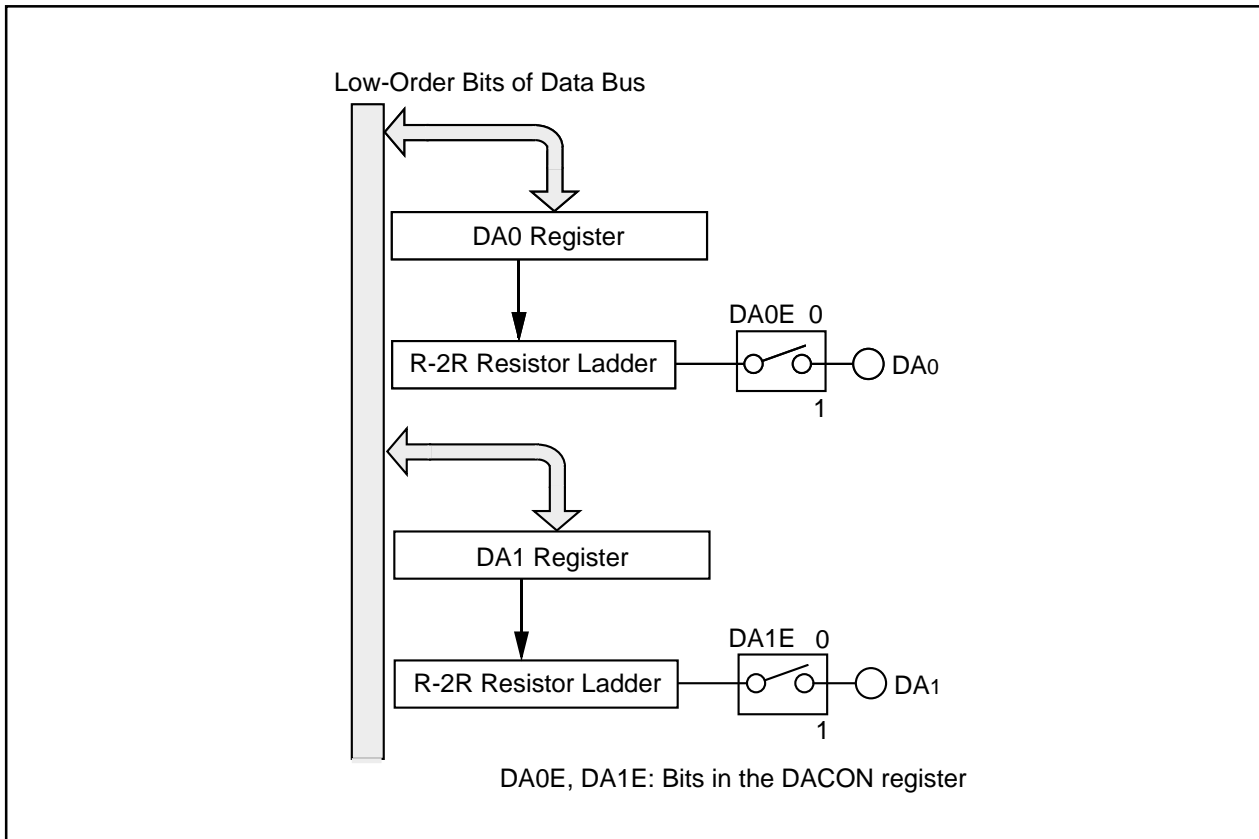
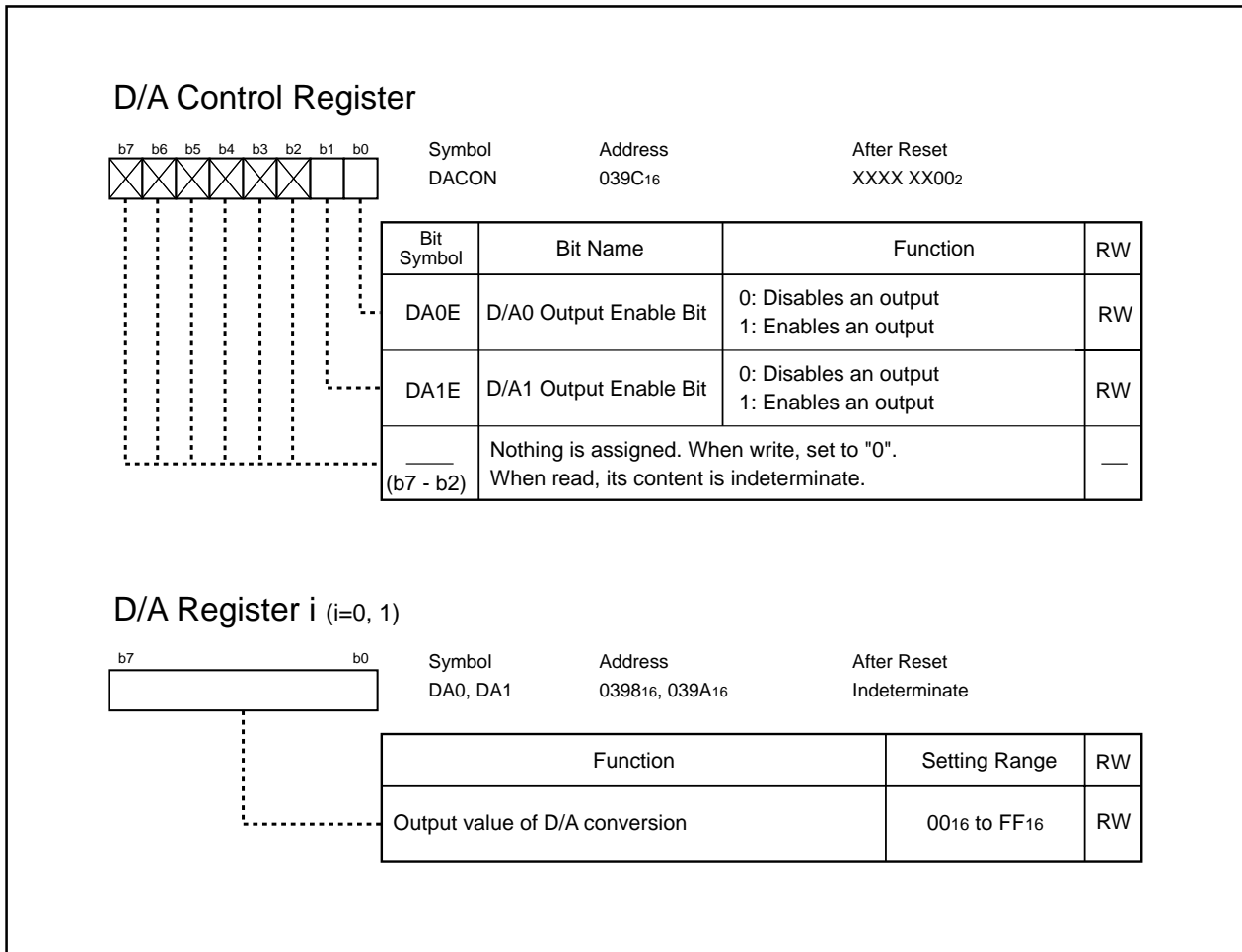
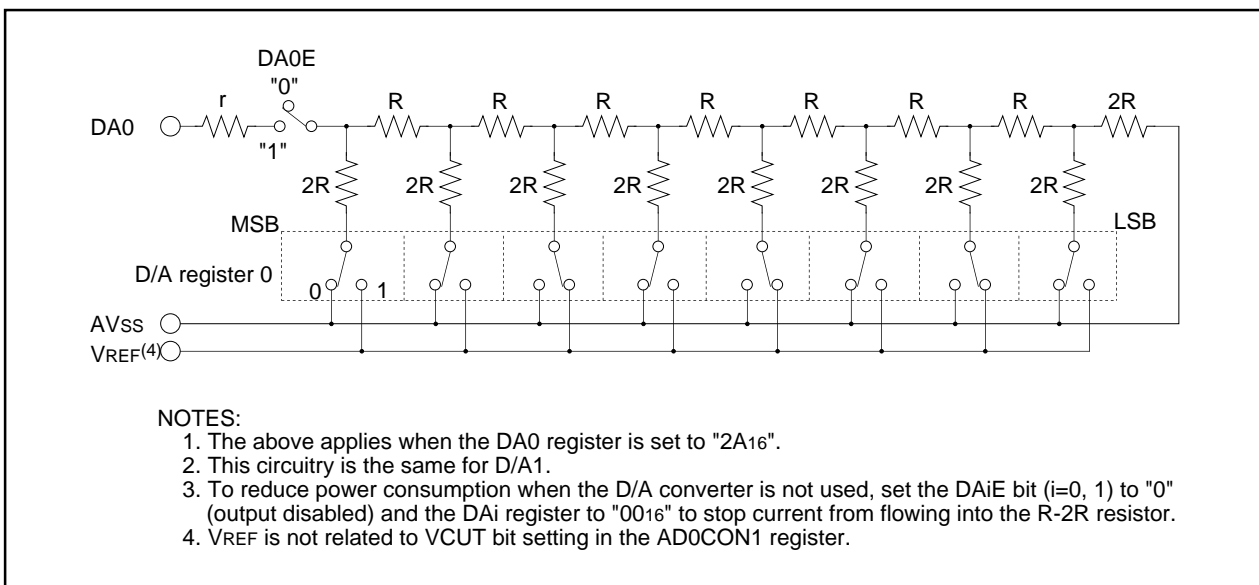


Figure 18.1 D/A Converter





**Figure 18.2 DACON Register, DA0 and DA1 Registers**



**Figure 18.3 D/A Converter Equivalent Circuit**

# 19. CRC Calculation

The CRC (Cyclic Redundancy Check) calculation detects an error in data blocks. A generator polynomial of CRC\_CCITT ( $X^{16} + X^{12} + X^5 + 1$ ) generates CRC code.

The CRC code is a 16-bit code generated for a block of data of desired length. This block of data is in 8-bit units. The CRC code is set in the CRCD register every time one-byte data is transferred to the CRCIN register after a default value is written to the CRCD register. CRC code generation for one-byte data is completed in two cycles.

Figure 19.1 shows a block diagram of a CRC circuit. Figure 19.2 shows CRC-associated registers. Figure 19.3 shows an example of the CRC calculation.

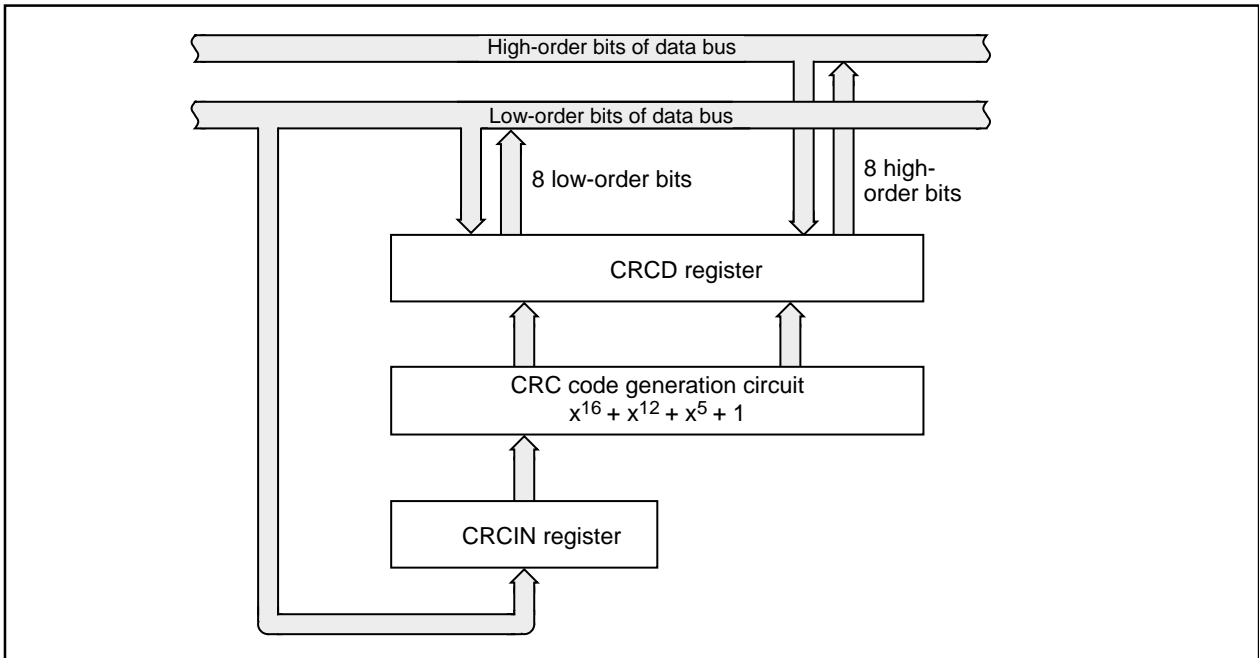


Figure 19.1 CRC Calculation Block Diagram

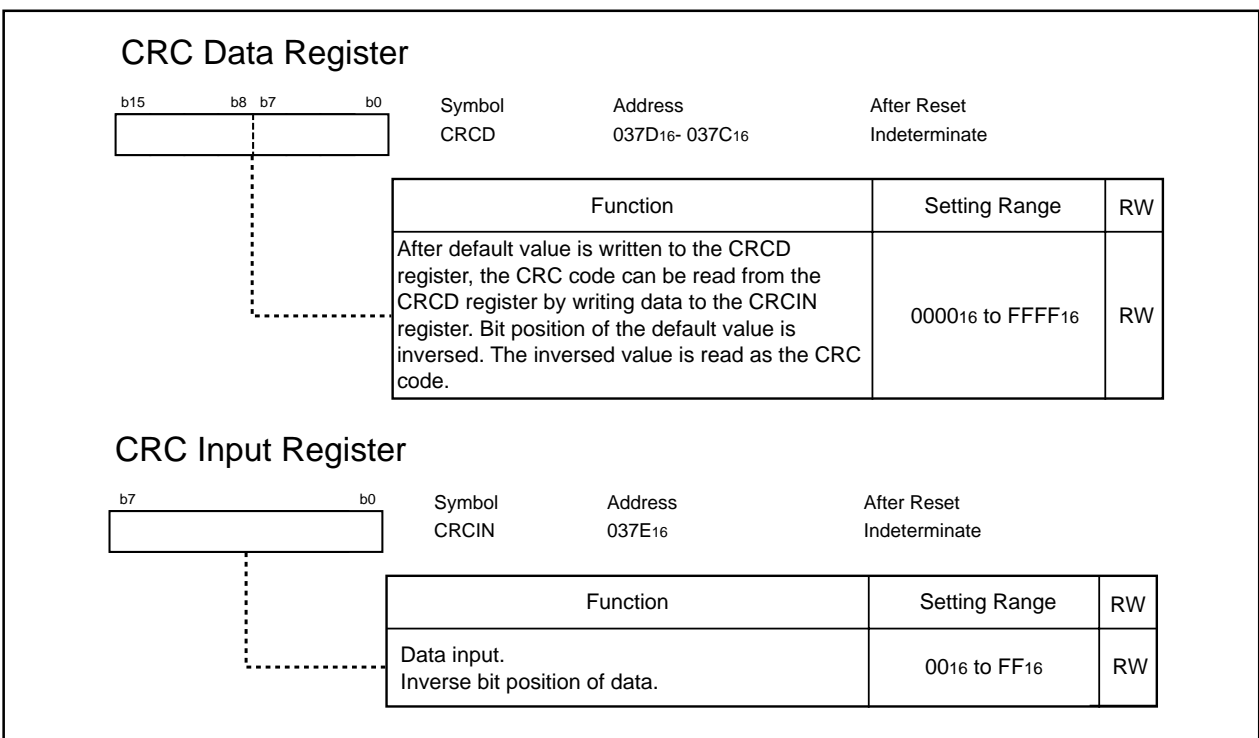


Figure 19.2 CRCD Register and CRCIN Register

### CRC Calculation and Setup Procedure to Generate CRC Code for "80C416"

○ CRC Calculation for M32C

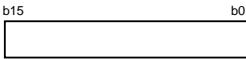
CRC Code : a remainder of a division,  $\frac{\text{value of the CRCIN register with inverted bit position}}{\text{generator polynomial}}$

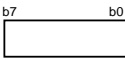
Generator Polynomial :  $X^{16} + X^{12} + X^5 + 1$  (1 0001 0000 0010 0001<sub>2</sub>)

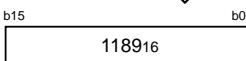
○ Setting Steps


(1) Inverse a bit position of "80C416" per byte by program

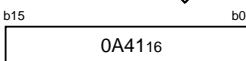
"8016" → "0116", "C416" → "2316"

(2) Set "000016" (default value) →  CRCD register

(3) Set "0116" →  CRCIN register  
 Bit position of the CRC code for "8016" (918816) is inverted to "118916", which is stored into the CRCD register in 3rd cycle.

 CRCD register

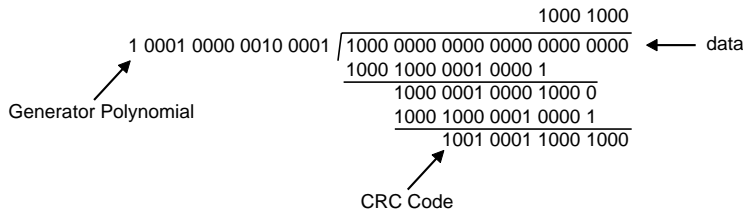
(4) Set "2316" →  CRCIN register  
 Bit position of the CRC code for "80C416" (825016) is inverted to "0A4116", which is stored into the CRCD register in 3rd cycle.

 CRCD register

○ Details of CRC Calculation

As shown in (3) above, bit position of "0116" (00000001<sub>2</sub>) written to the CRCIN register is inverted and becomes "10000002".

Add "1000 0000 0000 0000 0000 0000<sub>2</sub>", as "10000002" plus 16 digits, to "000016" as the default value of the CRCD register to perform the modulo-2 division.



Modulo-2 Arithmetic is calculated on the law below.

- 0 + 0 = 0
- 0 + 1 = 1
- 1 + 0 = 1
- 1 + 1 = 0
- 1 = 1

"0001 0001 1000 1001<sub>2</sub> (118916)", the remainder "1001 0001 1000 1000<sub>2</sub> (918816)" with inverted bit position, can be read from the CRCD register.

When going on to (4) above, "2316 (00100011<sub>2</sub>)" written in the CRCIN register is inverted and becomes "11000100<sub>2</sub>".

Add "1100 0100 0000 0000 0000 0000<sub>2</sub>", as "11000100<sub>2</sub>" plus 16 digits, to "1001 0001 1000 1000<sub>2</sub>" as a remainder of (3) left in the CRCD register to perform the modulo-2 division.

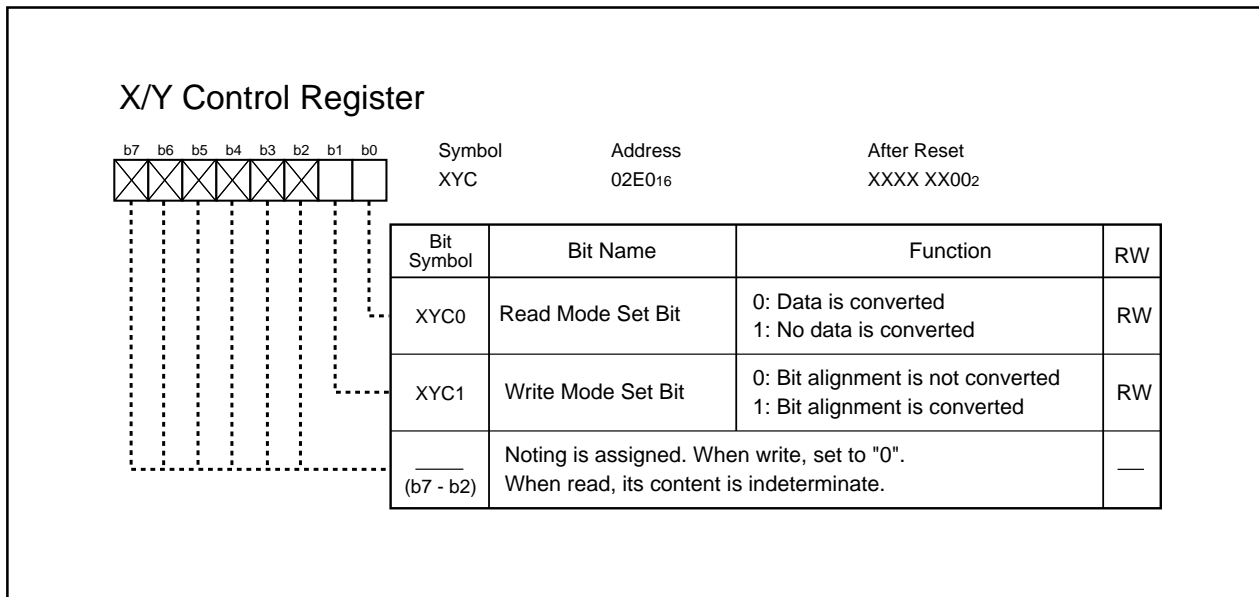
"0000 1010 0100 0001<sub>2</sub> (0A4116)", the remainder with inverted bit position, can be read from CRCD register.

Figure 19.3 CRC Calculation

## 20. X/Y Conversion

The X/Y conversion rotates a 16 x 16 matrix data by 90 degrees and inverses high-order bits and low-order bits of a 16-bit data. Figure 20.1 shows the XYC register.

The 16-bit XiR register (i=0 to 15) and 16-bit YjR register (j=0 to 15) are allocated to the same address. The XiR register is a write-only register, while the YjR register is a read-only register. Access the XiR and YjR registers from an even address in 16-bit units. Performance cannot be guaranteed if the XiR and YiR registers are accessed in 8-bit units.



**Figure 20.1** XYC Register

The XYC0 bit in the XYC register determines how to read the YjR register.

By reading the YjR register when the XYC0 bit is set to "0" (data conversion), bit j in the X0R to X15R registers can be read simultaneously.

For example, bit 0 in the X0R register can be read if reading bit 0 in the Y0R register, bit 0 in the X1R register if reading bit 1 in the Y0R register..., bit 0 in the X14R register if reading bit 14 in the Y0R register and bit 0 in the X15R register if reading bit 15 in the Y0R register.

Figure 20.2 shows the conversion table when the XYC0 bit is set to "0". Figure 20.3 shows an example of the X/Y conversion.

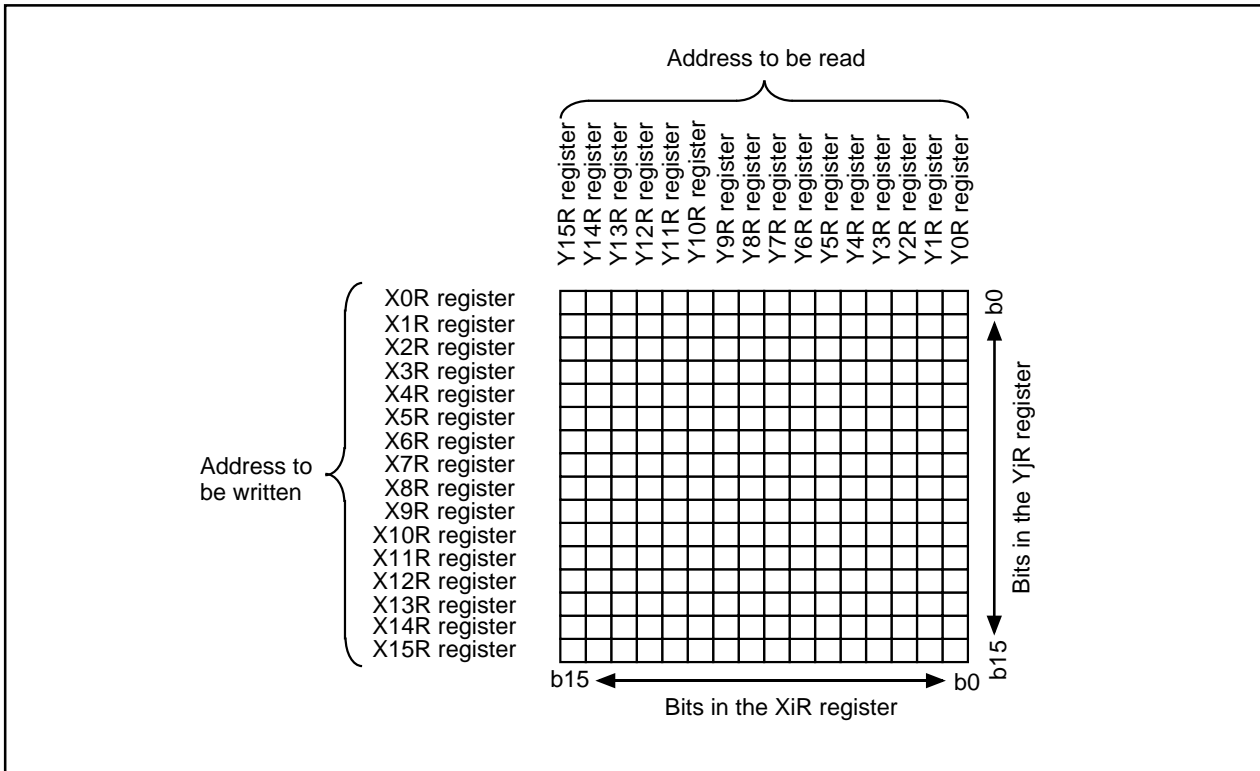


Figure 20.2 Conversion Table when Setting the XYC0 Bit to "0"

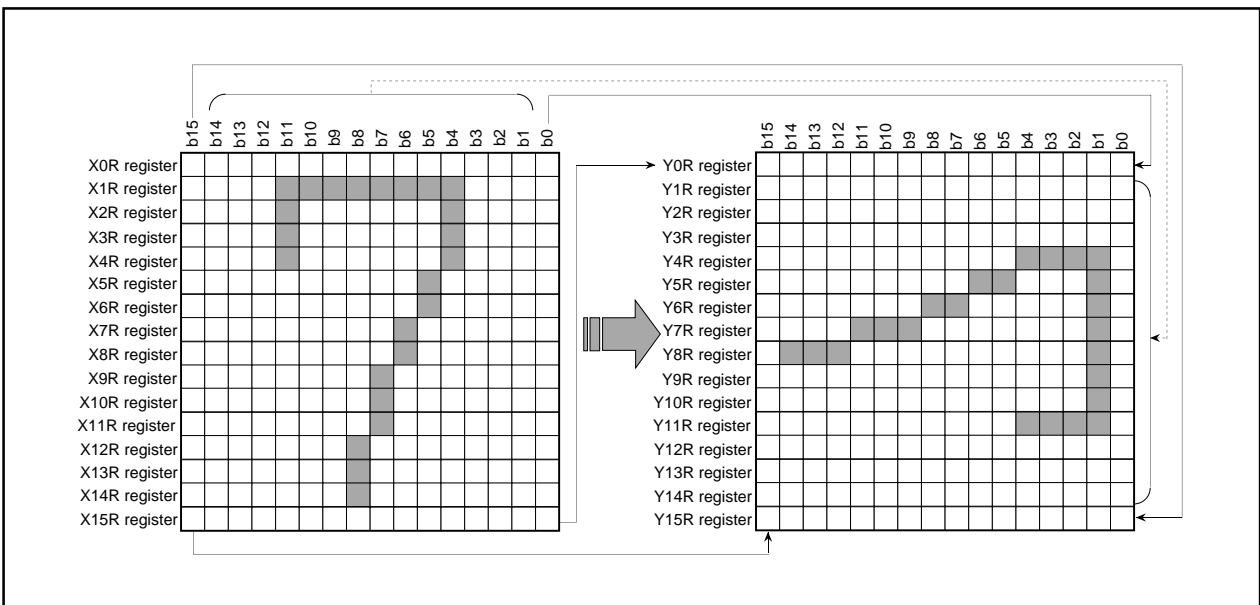


Figure 20.3 X/Y Conversion

By reading the YjR register when the XYC0 bit in the XYC register is set to "1" (no data conversion), the value written to the XiR register can be read directly. Figure 20.4 shows the conversion table when the XYC0 bit is set to "1."

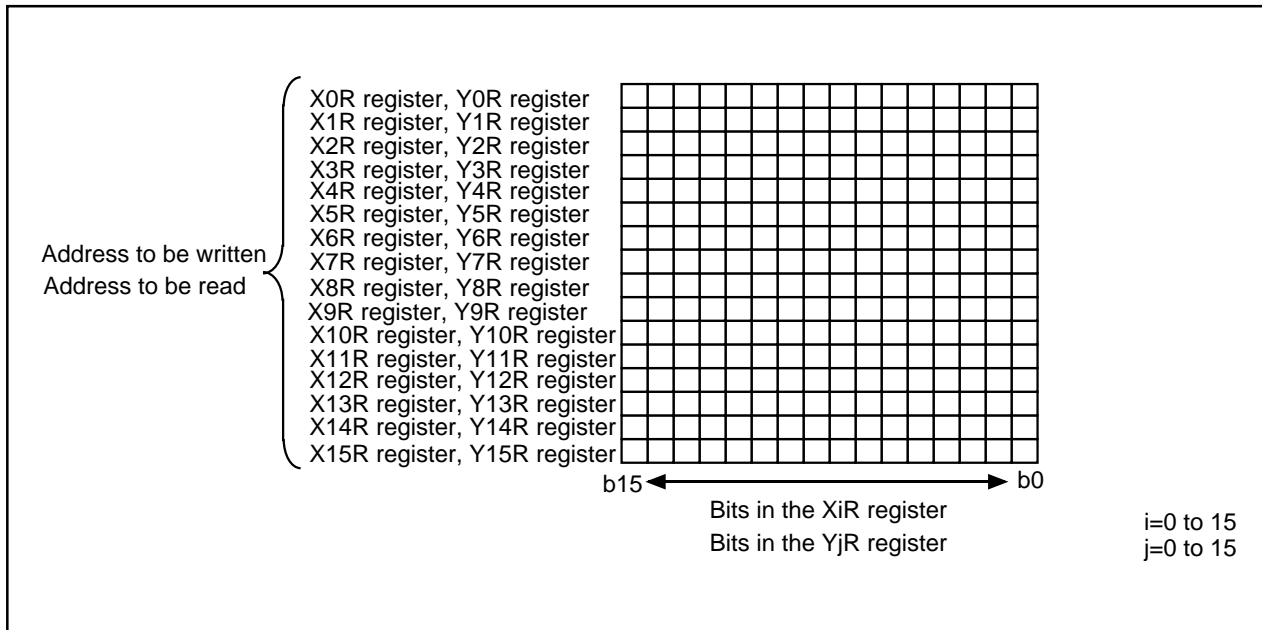


Figure 20.4 Conversion Table when Setting the XYC0 Bit to "1"

The XYC1 bit in the XYC register selects bit alignment of the value in the XiR register.

By writing to the XiR register while the XYC1 bit is set to "0" (no bit alignment conversion), bit alignment is written as is. By writing to the XiR register while the XYC1 bit is set to "1" (bit sequence replaced), bit alignment is written inversed.

Figure 20.5 shows the conversion table when the XYC1 bit is set to "1".

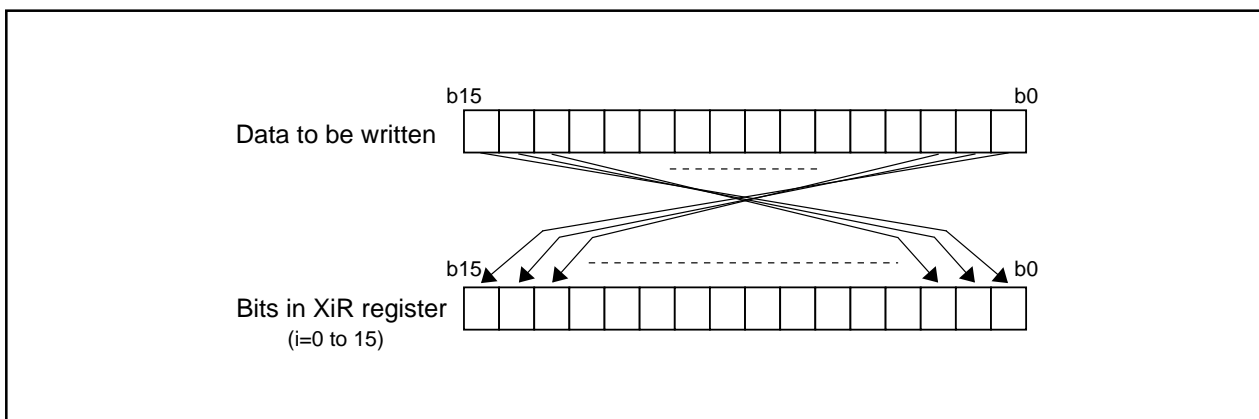


Figure 20.5 Conversion Table when Setting the XYC1 Bit to "1"

## 21. Intelligent I/O

The intelligent I/O is a multifunctional I/O port for clock synchronous serial I/O and HDLC data processing.

The intelligent I/O has two sets of two 8-bit shift registers for communications.

Table 21.1 lists functions and channels of the intelligent I/O.

**Table 21.1 Intelligent I/O Functions and Channels**

Function	Description	
Communication	Communication unit 0	Communication unit 1
Clock Synchronous Serial I/O Mode	Available	Available
HDLC Data Processing Mode	Available	

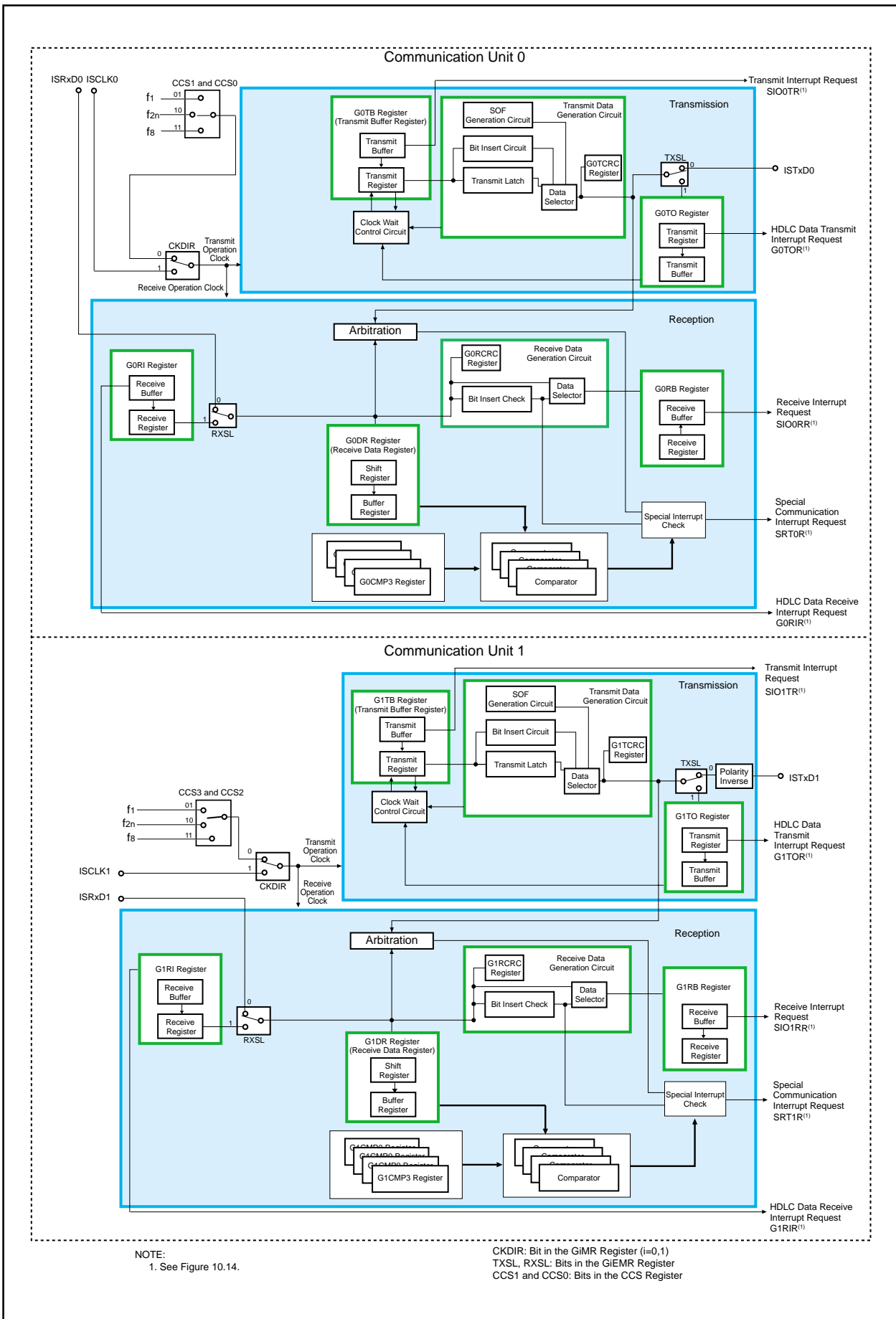


Figure 21.1 Intelligent I/O Communication Unit Block Diagram



### 21.1 Communication Unit 0 and 1 Communication Function

In the intelligent I/O communication units, 8-bit clock synchronous serial I/O or HDLC data processing is available.

Figures 21.2 to 21.11 show registers associated with the communication function.

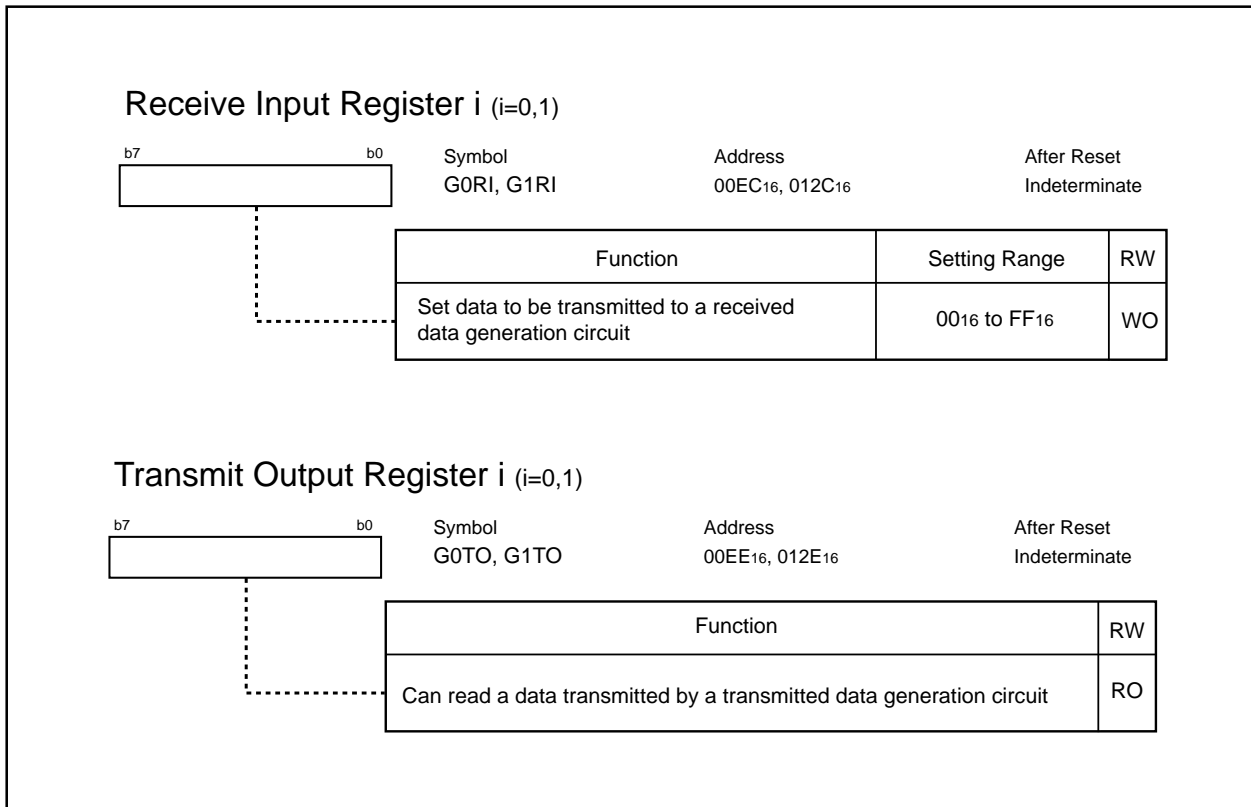
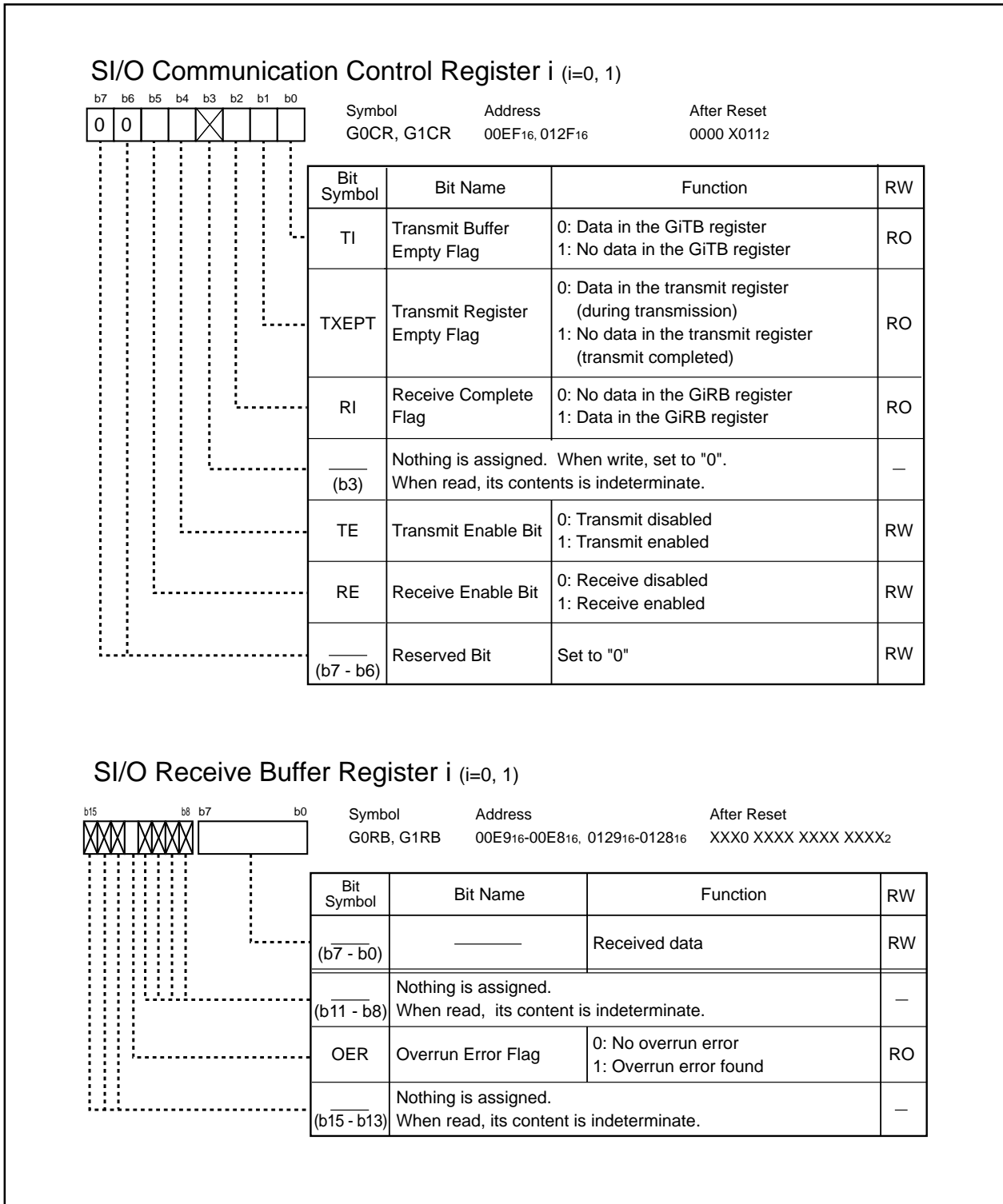


Figure 21.2 G0RI and G1RI Registers, G0TO and G1TO Registers



**Figure 21.3 G0CR and G1CR Registers, G0RB and G1RB Registers**

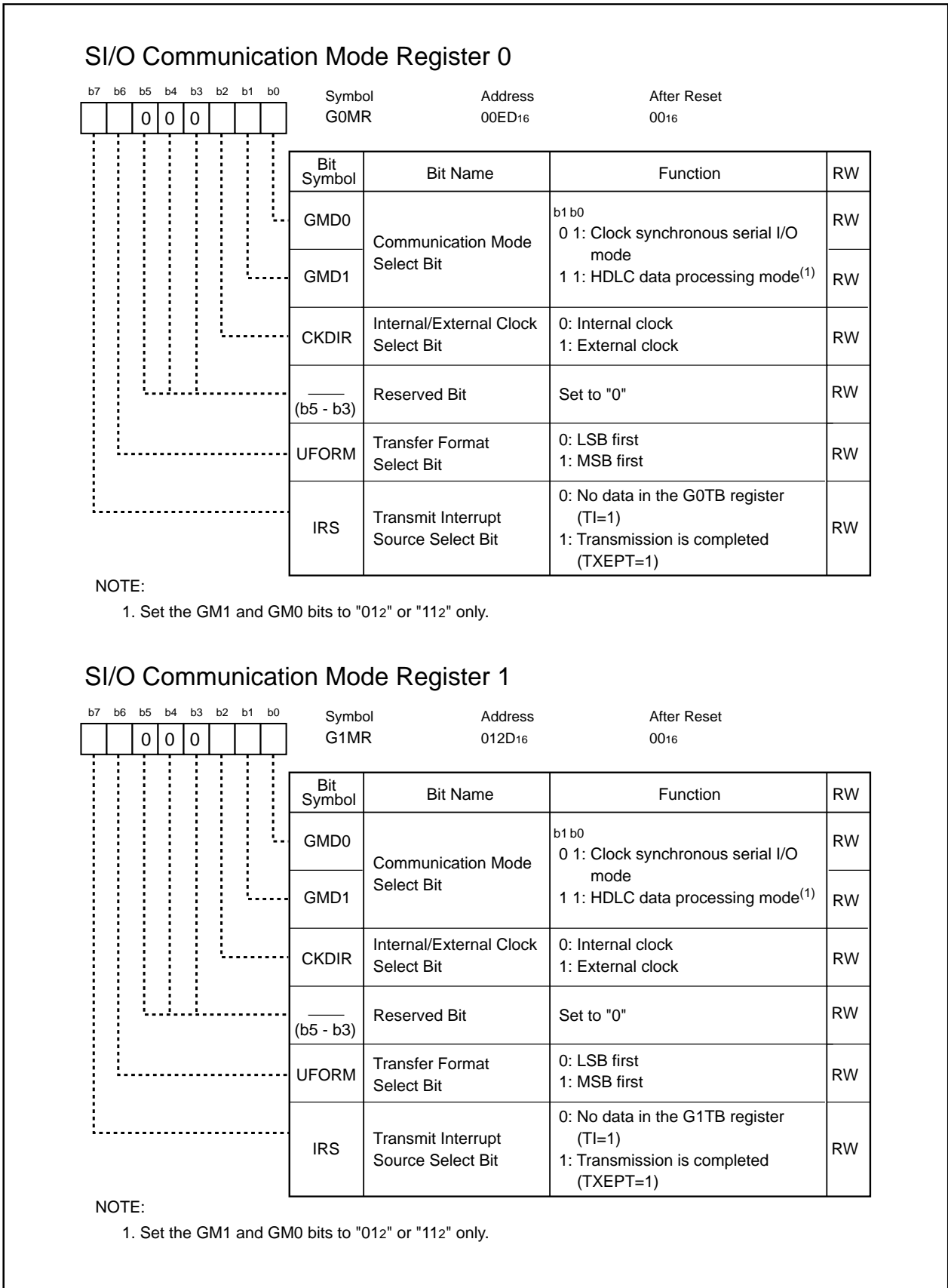
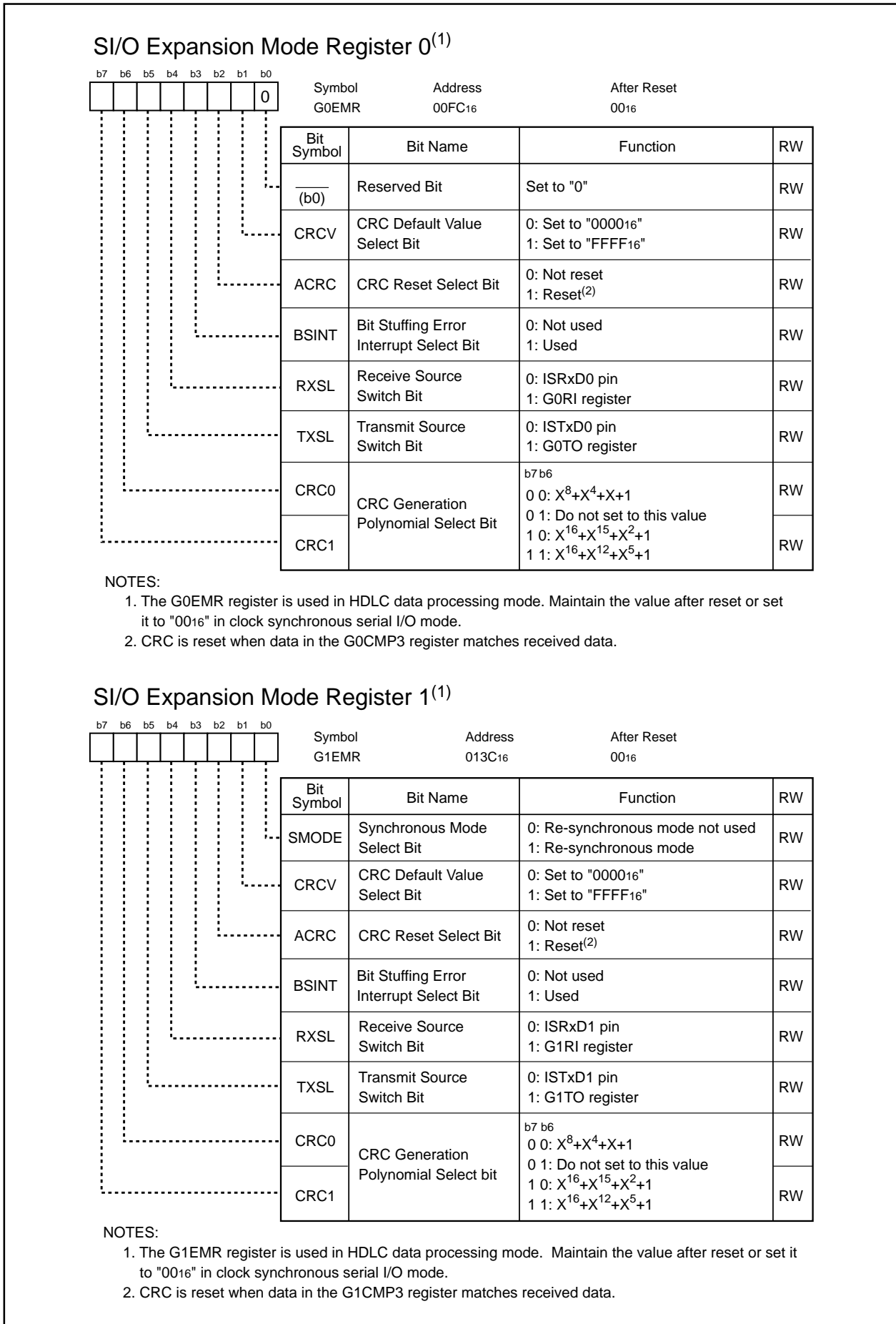
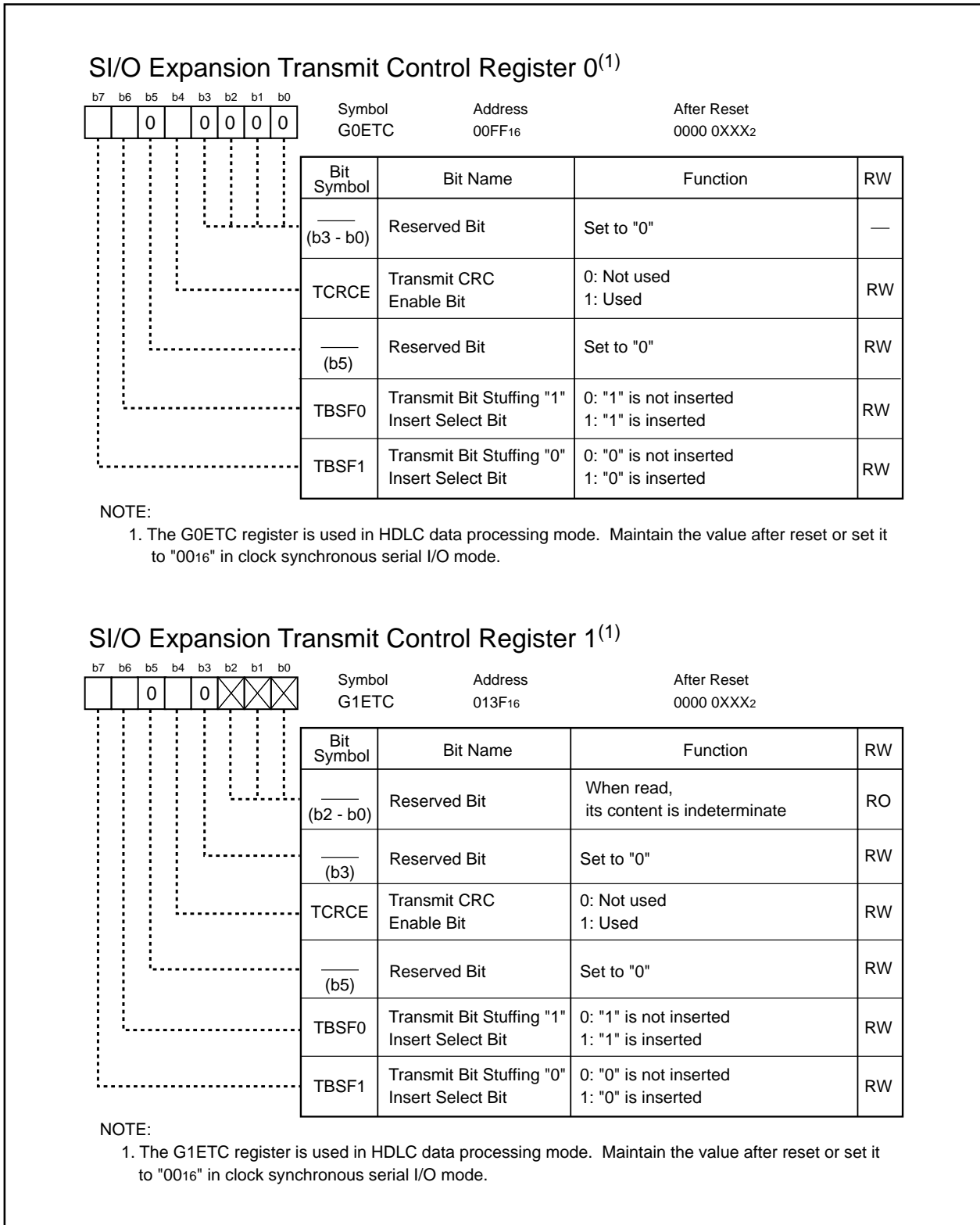


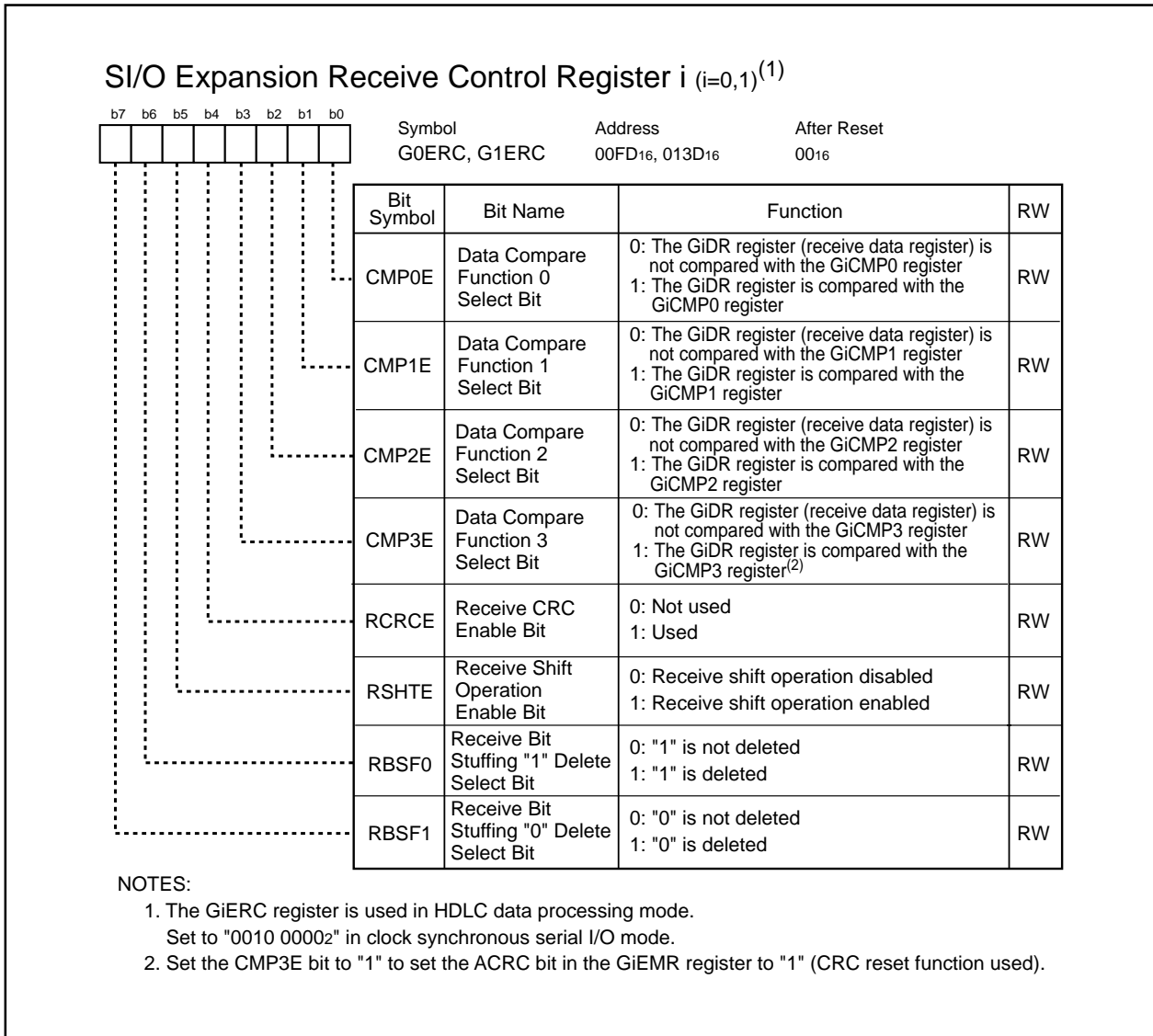
Figure 21.4 G0MR and G1MR Registers



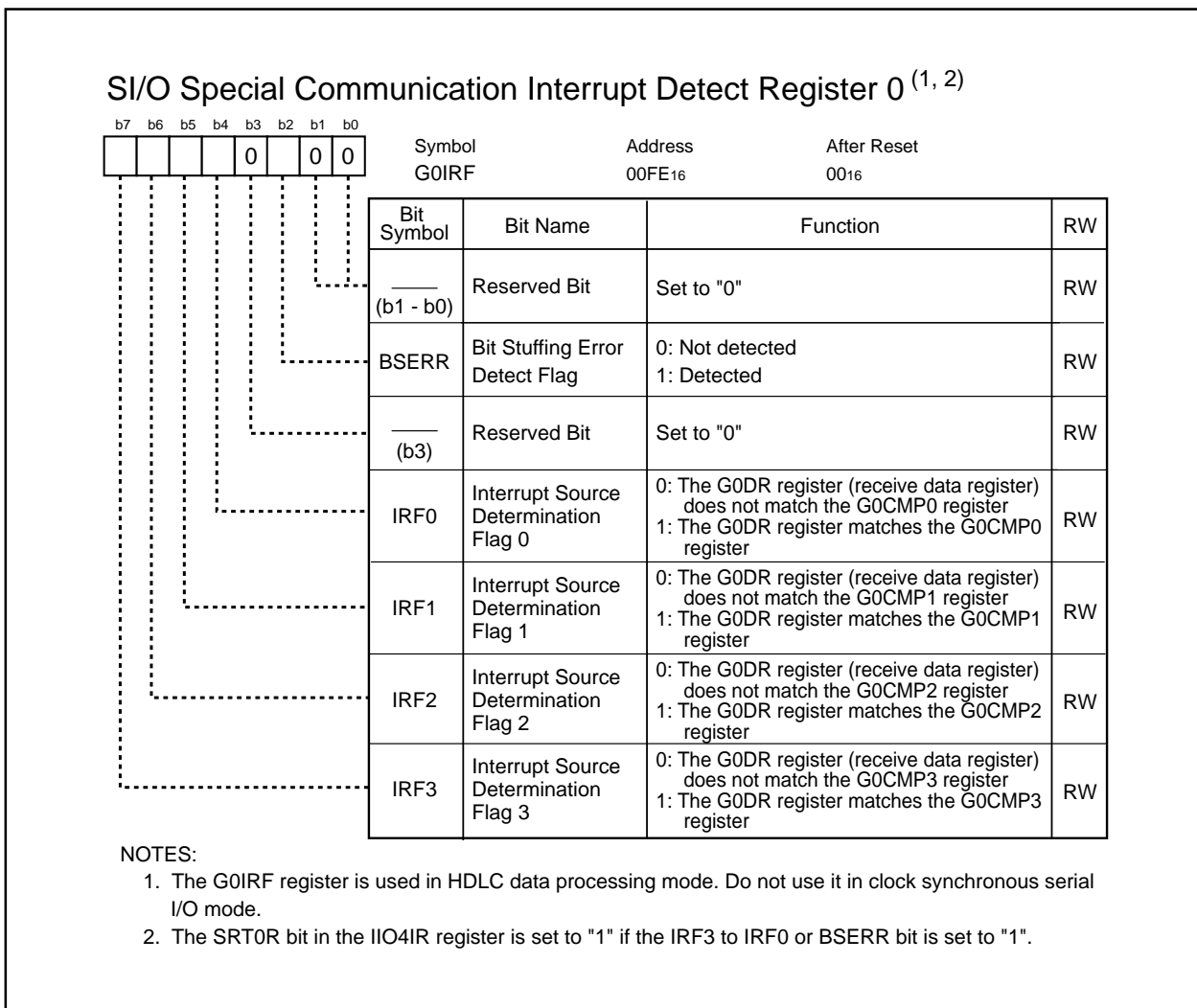
**Figure 21.5 G0EMR and G1EMR Registers**



**Figure 21.6 G0ETC and G1ETC Registers**



**Figure 21.7 G0ERC and G1ERC Registers**



**Figure 21.8 G0IRF Register**

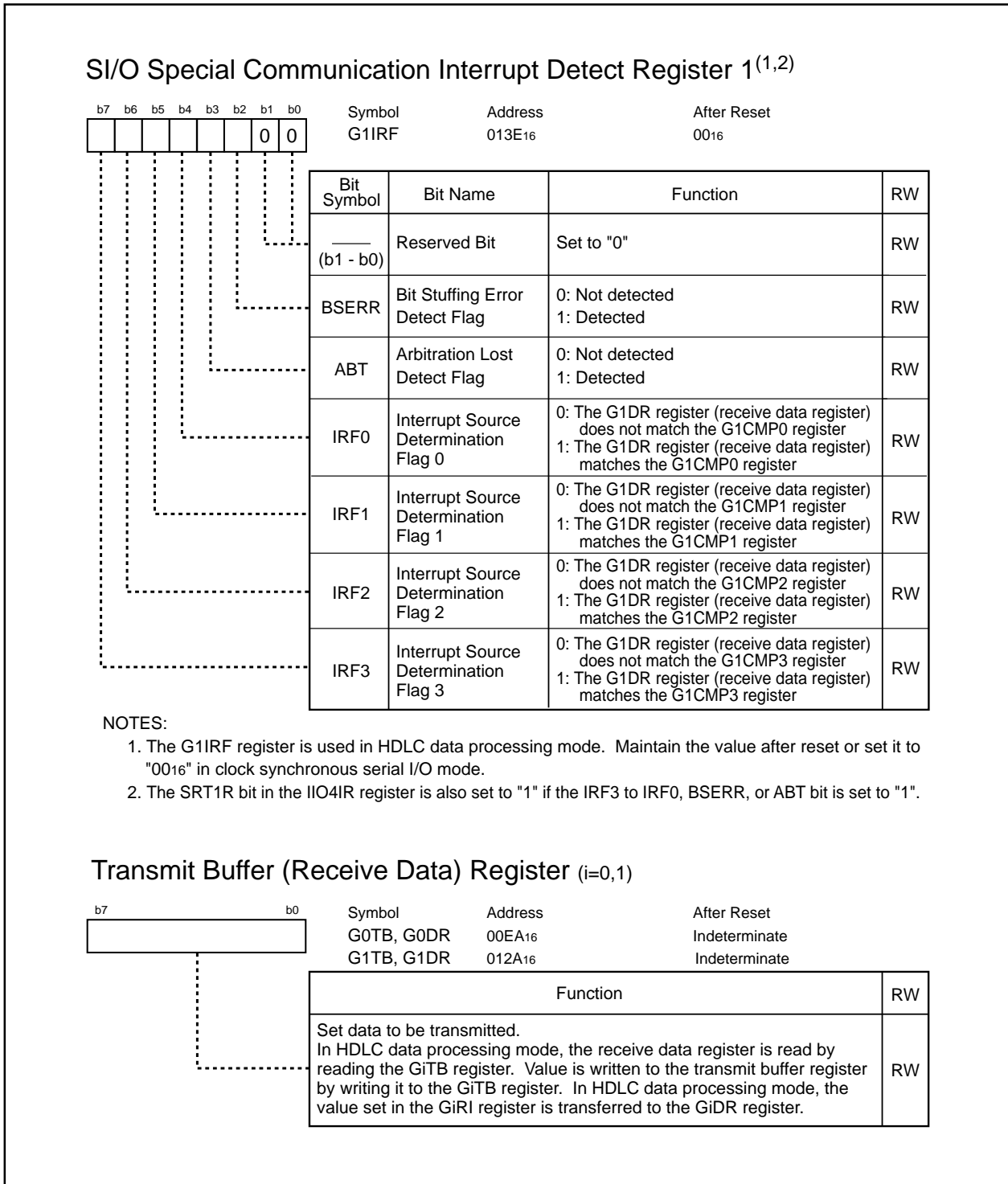


Figure 21.9 G1IRF Register, G0TB and G1TB / G0DR and G1DR Registers



### Data Compare Register ij (i=0,1, j=0 to 3)

b7	b0	Symbol	Address	After Reset
		G0CMP0 to G0CMP3	00F0 <sub>16</sub> , 00F1 <sub>16</sub> , 00F2 <sub>16</sub> , 00F3 <sub>16</sub>	Indeterminate
		G1CMP0 to G1CMP3	0130 <sub>16</sub> , 0131 <sub>16</sub> , 0132 <sub>16</sub> , 0133 <sub>16</sub>	Indeterminate
		Function	Setting Range	RW
		Data to be compared	00 <sub>16</sub> to FF <sub>16</sub>	RW

**NOTE:**

- Set the GiMSK0 register to use the GiCMP0 register.  
Set the GiMSK1 register to use the GiCMP1 register.

### Data Mask Register ij (i=0,1, j=0,1)

b7	b0	Symbol	Address	After Reset
		G0MSK0, G0MSK1	00F4 <sub>16</sub> , 00F5 <sub>16</sub>	Indeterminate
		G1MSK0, G1MSK1	0134 <sub>16</sub> , 0135 <sub>16</sub>	Indeterminate
		Function	Setting Range	RW
		Masked data for received data Set incomparable bit to "1"	00 <sub>16</sub> to FF <sub>16</sub>	RW

### Transmit CRC Code Register i (i=0,1)

b15	b8	b7	b0	Symbol	Address	After Reset
				G0TCRC, G1TCRC	00FB <sub>16</sub> -00FA <sub>16</sub> , 013B <sub>16</sub> -013A <sub>16</sub>	0000 <sub>16</sub>
				Function	RW	
				Result of the transmit CRC calculation <sup>(1, 2)</sup>	RO	

**NOTES:**

- The calculated result is reset by setting the TE bit in the GiCR register to "0" (transmit disabled).  
The CRCV bit in the GiEMR register selects a default value.
- Transmit CRC calculation is performed with each bit of data transmitted while the TCRCE bit in the GiETC register is set to "1" (used).

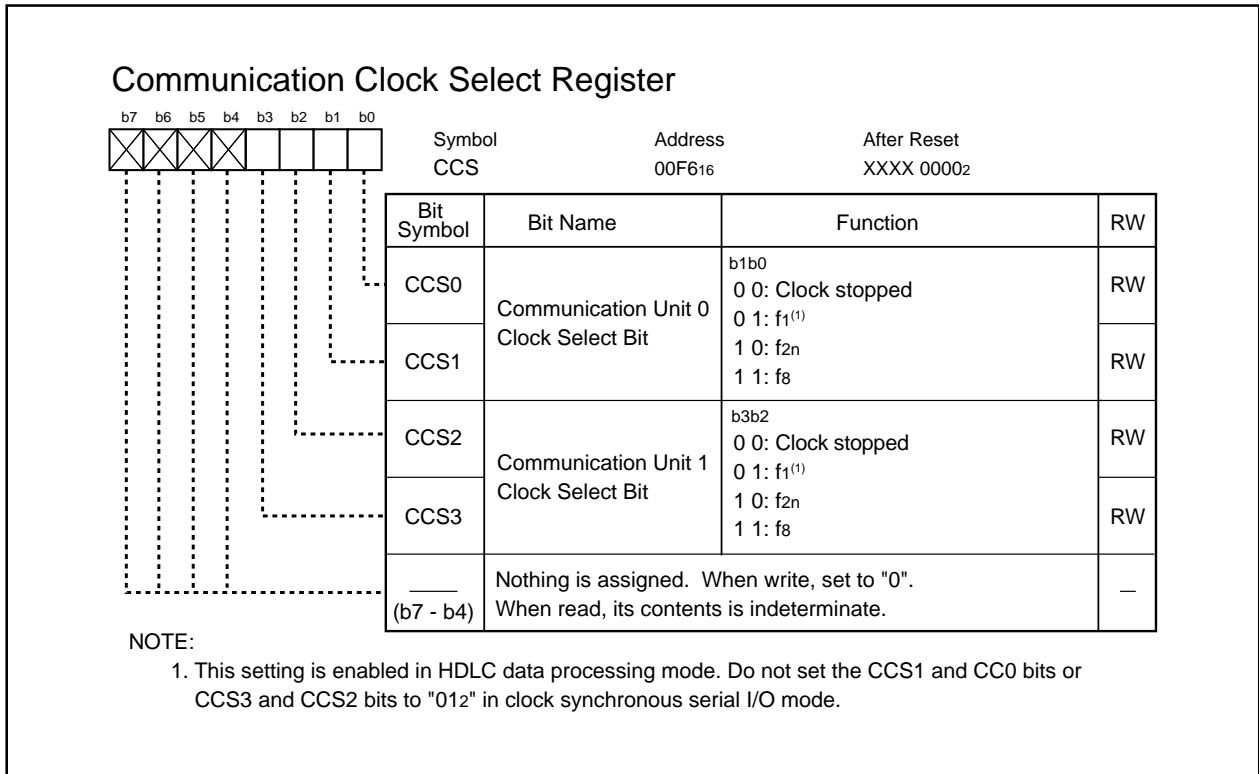
### Receive CRC Code Register i (i=0,1)

b15	b8	b7	b0	Symbol	Address	After Reset
				G0RCRC, G1RCRC	00F9 <sub>16</sub> -00F8 <sub>16</sub> , 0139 <sub>16</sub> -0138 <sub>16</sub>	Indeterminate
				Function	RW	
				Result of the receive CRC calculation <sup>(1, 2, 3)</sup>	RO	

**NOTES:**

- The calculated result is reset by setting the RCRCE bit in the GiERC register to "0" (not used).  
If the ACRC bit in the GiEMR register is set to "1" (reset), the result is reset by matching data in the GiCMPj register (j=0 to 3) with the received data.
- The result is reset to the default value selected by the CRCV bit in the GiEMR register before reception starts.
- Receive CRC calculation is performed with every bit of data received while the RCRCE bit in the GiERC register is set to "1" (used).

**Figure 21.10 G0CMP0 to G0CMP3 Registers and G1CMP0 to G1CMP3 Registers  
G0MSK0 and G0MSK1 Registers, G1MSK0 and G1MSK1 Registers  
G0TCRC and G1TCRC Registers, G0RCRC and G1RCRC Registers**



**Figure 21.11 CCS Register**

### 21.1.1 Clock Synchronous Serial I/O Mode (Communication Units 0 and 1)

In clock synchronous serial I/O mode, data is transmitted and received with the transfer clock.  $f_8$  or  $f_{2n}$  can be selected as the transfer clock.

Table 21.2 lists specifications of clock synchronous serial I/O mode for the communication units 0 and 1. Tables 21.3 and 21.4 list clock settings. Table 21.5 lists register settings. Tables 21.6 and 21.7 list pin settings. Figure 21.12 shows an example of transmit and receive operation.

**Table 21.2 Clock Synchronous Serial I/O Mode Specifications (Communication Units 0 and 1)**

Item	Specification
Transfer Data Format	Transfer data: 8 bits long
Transfer Clock <sup>(1)</sup>	See Tables 21.3 and 21.4
Transmit Start Condition	Set registers associated with the waveform generating function, the GiMR and GiERC registers ( $i=0,1$ ). Then, set as is written below after at least one transfer clock cycle. <ul style="list-style-type: none"> <li>• Set the TE bit in the GiCR register to "1" (transmit enabled)</li> <li>• Set the TI bit in the GiCR register to "0" (data in the GiTB register)</li> </ul>
Receive Start Condition	Set registers associated with the waveform generating function, the GiMR and GiERC registers. Then, set as is written below after at least one transfer clock cycle. <ul style="list-style-type: none"> <li>• Set the RE bit in the GiCR register to "1" (receive enabled)</li> <li>• Set the TE bit to "1" (transmit enabled)</li> <li>• Set the TI bit to "0" (data in the GiTB register)</li> </ul>
Interrupt Request	<ul style="list-style-type: none"> <li>• While transmitting, one of the following conditions can be selected to set the SIOiTR bit to "1" (interrupt requested) (See Figure 10.14): <ul style="list-style-type: none"> <li>– The IRS bit in the GiMR register is set to "0" (no data in the GiTB register) and data is transferred to the transmit register from the GiTB register</li> <li>– The IRS bit is set to "1" (transmission completed) and data transfer from the transmit register is completed</li> </ul> </li> <li>• While receiving, the following condition can be selected to set SIOiRR bit is set to "1" (data reception is completed): <ul style="list-style-type: none"> <li>– Data is transferred from the receive register to the GiRB register (See Figure 10.14)</li> </ul> </li> </ul>
Error Detection	<p>Overrun error<sup>(2)</sup></p> <p>This error occurs, when the next data reception is started and the 8th bit of the next data is received before reading the GiRB register</p>
Selectable Function	<ul style="list-style-type: none"> <li>• LSB first or MSB first</li> </ul> <p>Select either bit 0 or bit 7 to transmit or receive data</p>

**NOTES:**

1. In clock synchronous serial I/O mode, set the RSHTTE bit in the GiERC register ( $i=0, 1$ ) to "1" (receive shift operation enabled).
2. When an overrun error occurs, the GiRB register is indeterminate.

The ISTxDi pin outputs a high-level ("H") signal between selecting operating mode and starting transfer.

**Table 21.3 Clock Settings (Communication Unit 0)**

Transfer Clock	GOMR Register	CCS Register	
	CKDIR Bit	CCS0 Bit	CCS1 Bit
$f_8$	0	1	1
$f_{2n}^{(1)}$	0	0	1
Input from ISCLK0	1	-	-

**NOTE:**

1. The CNT3 to CNT0 bits in the TCSPR register select no division ( $n=0$ ) or divide-by- $2^n$  ( $n=1$  to 15).

**Table 21.4 Clock Settings (Communication Unit 1)**

Transfer Clock	G1MR Register	CCS Register	
	CKDIR Bit	CCS2 Bit	CCS3 Bit
f <sub>8</sub>	0	1	1
f <sub>2n</sub> <sup>(2)</sup>	0	0	1
Input from ISCLK1	1	-	-

NOTE:

- The CNT3 to CNT0 bits in the TCSPR register select no division ( $n=0$ ) or divide-by- $2^n$  ( $n=1$  to 15).

**Table 21.5 Register Settings in Clock Synchronous Serial I/O Mode**

Register	Bit	Function	
		Communication Unit 1	Communication Unit 0
CCS	CCS1, CCS0	Setting not required when using the communication unit 1 only	Select transfer clock
	CCS3, CSS2	Select transfer clock	Setting not required when using the communication unit 0 only
GIERC	7 to 0	Set to "0010 00002"	
GiMR	GMD1, GMD0	Set to "012"	
	CKDIR	Select internal clock or external clock	
	UFORM	Select either LSB first or MSB first	
	IRS	Select what cause the transmit interrupt to be generated	
GiCR	TI	Transmit buffer empty flag	
	TXEPT	Transmit register empty flag	
	RI	Receive complete flag	
	TE	Set to "1" to enable transmission and reception	
	RE	Set to "1" to enable reception	
GiTB	-	Write data to be transmitted	
GiRB	-	Received data and error flag are stored	

i=0, 1

**Table 21.6 Pin Settings in Clock Synchronous Serial I/O Mode (1)**

Port	Function	Setting				
		PS1 Register	PSL1 Register	PSC Register	PSD1 Register	PD7 Register
P73	ISTxD1 Output	PS1_3=1	PSL1_3=0	PSC_3=1	-	-
P74	ISCLK1 Input	PS1_4=0	-	-	-	PD7_4=0
	ISCLK1 Output	PS1_4=1	PSL1_4=0	PSC_4=1	-	-
P75	ISRxD1 Input	PS1_5=0	-	-	-	PD7_5=0
P76	ISTxD0 Output	PS1_6=1	PSL1_6=0	PSC_6=0	PSD1_6=0	-
P77	ISCLK0 Input	PS1_7=0	-	-	-	PD7_7=0
	ISCLK0 Output	PS1_7=1	PSL1_7=0	-	-	-

**Table 21.7 Pin Settings (2)**

Port	Function	Setting	
		PS2 Register	PD8 Register
P80	ISRxD0 Input	PS2_0=0	PD8_0=0

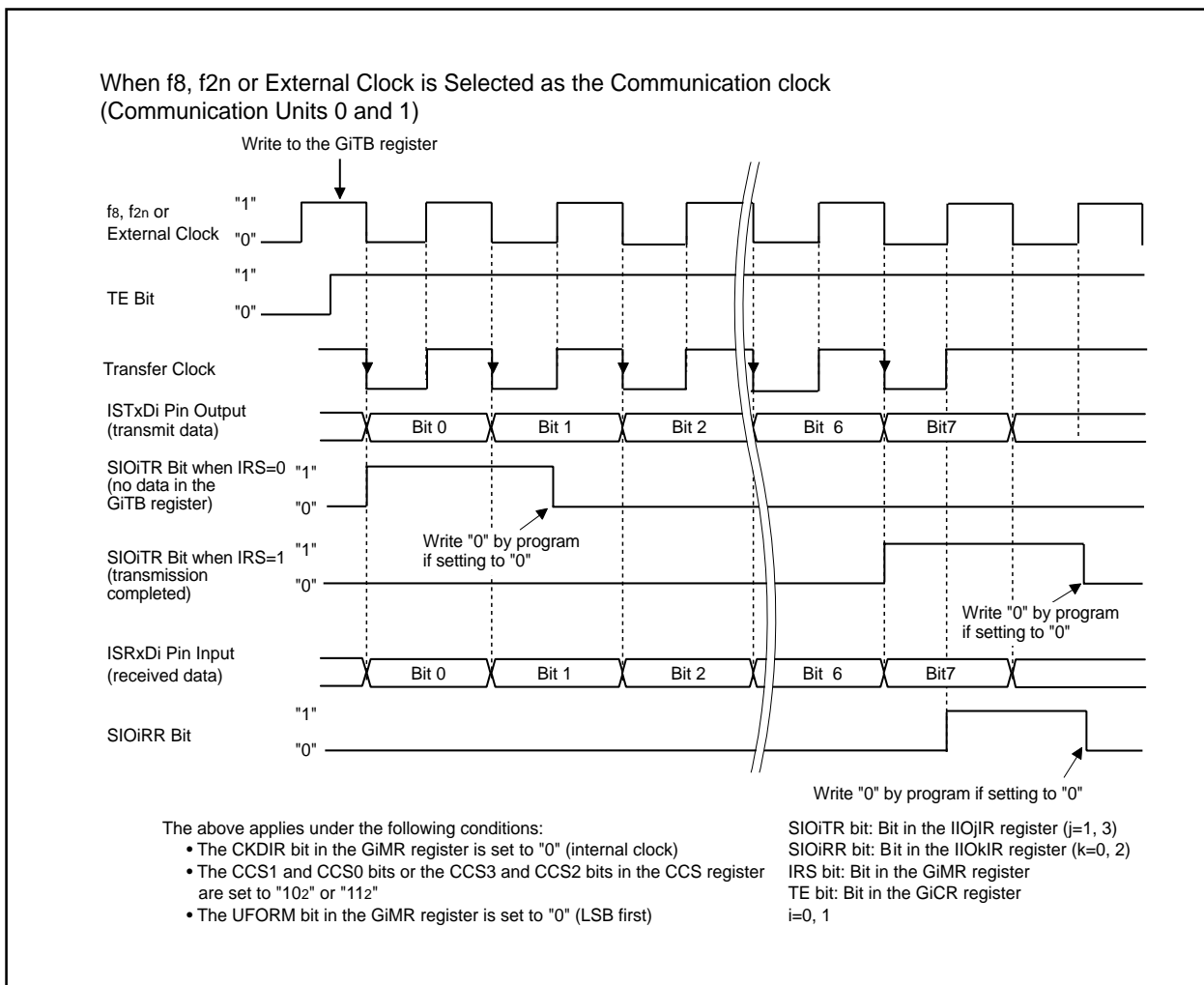


Figure 21.12 Transmit and Receive Operation

### 21.1.2 HDLC Data Processing Mode (Communication Units 0 and 1)

In HDLC data processing mode, bit stuffing, flag detection, abort detection and CRC calculation are available for HDLC control. f1, f8 or f2n can be selected as the transfer clock. No pin is used.

To convert data, data to be transmitted is written to the GiTB register (i=0,1) and the data conversion result is restored after data conversion. If any data are in the GiTO register after data conversion, the conversion is terminated. If no data is in the GiTO register, bit stuffing processing is executed regardless of no data available in the transmit output buffer. A CRC value is calculated every time one bit is converted. If no data is in the GiRI register, received data conversion is terminated.

Table 21.8 list specifications of the HDLC data processing mode. Tables 21.9 and 21.10 list clock settings. Table 21.11 lists register settings.

**Table 21.8 HDLC Processing Mode Specifications (Communication Units 0 and 1)**

Item	Specification
Input Data Format	8-bit data fixed, bit alignment is optional
Output Data Format	8-bit data fixed
Transfer Clock	See Tables 21.9 and 21.10
I/O Method	<ul style="list-style-type: none"> <li>• During transmit data processing, value set in the GiTB register is converted in HDLC data processing mode and transferred to the GiTO register.</li> <li>• During received data processing, value set in the GiRI register is converted in HDLC data processing mode and transferred to the GiRB register. The value in the GiRI register is also transferred to the GiTB register (received data register).</li> </ul>
Bit Stuffing	During transmit data processing, "0" following five continuous "1" is inserted. During received data processing, "0" following five continuous "1" is deleted.
Flag Detection	Write the flag data "7E16" to the GiCMPj register (j=0 to 3) to use the special communication interrupt (the SRTiR bit in the IIO4IR register)
Abort Detection	Write the masked data "0116" to the GiMSKj register
CRC	<p>The CRC1 and CRC0 bits are set to "112" (<math>X^{16}+X^{12}+X^5+1</math>). The CRCV bit is set to "1" (set to "FFFF16").</p> <ul style="list-style-type: none"> <li>• During transmit data processing, CRC calculation result is stored into the GiTCRC register. The TCRCE bit in the GiETC register is set to "1" (transmit CRC used). The CRC calculation result is reset when the TE bit in the GiCR register is set to "0" (transmit disabled).</li> <li>• During received data processing, CRC calculation result is stored into the GiRCRC register. The RCRCE bit in the GiERC register is set to "1" (receive CRC used). The CRC calculation result is reset by comparing the flag data "7E16" and matching the result with the value in the GiCMP3 register. The ACRC bit in the GiEMR register is set to "1" (CRC reset).</li> </ul>
Data Processing Start Condition	<p>The following conditions are required to start transmit data processing:</p> <ul style="list-style-type: none"> <li>• The TE bit in the GiCR register is set to "1" (transmit enabled)</li> <li>• Data is written to the GiTB register</li> </ul> <p>The following conditions are required to start receive data processing:</p> <ul style="list-style-type: none"> <li>• The RE bit in the GiCR register is set to "1" (receive enabled)</li> <li>• Data is written to the GiRI register</li> </ul>

**Table 21.8 HDLC Processing Mode Specifications (Continued)**

Item	Specification
Interrupt Request <sup>(1)</sup>	<p>During transmit data processing,</p> <ul style="list-style-type: none"> <li>• One of the following conditions can be selected to set the GiTOR bit in the interrupt request register to "1" (interrupt request) (see Figure 10.14). <ul style="list-style-type: none"> <li>– When the IRS bit in the GiMR register is set to "0" (no data in the GiTB register) and data is transferred from the GiTB register to the transmit register (transmit start).</li> <li>– When the IRS bit is set to "1" (transmission completed) and data transfer from the transmit register to the GiTO register is completed.</li> </ul> </li> <li>• When data, which is already converted to HDLC data, is transferred from the receive register of the GiTO register to the transmit buffer, the GiTOR bit is set to "1"</li> </ul> <p>During received data processing,</p> <ul style="list-style-type: none"> <li>• When data is transferred from the GiRI register to the GiRB register (reception completed), the GiRIR bit is set to "1" (See Figure 10.14).</li> <li>• When received data is transferred from the receive buffer of the GiRI register to the receive register, the GiRIR bit is set to "1".</li> <li>• When the GiTB register is compared to the GiCMPj register (j=0 to 3), the SRTiR bit is set to "1".</li> </ul>

**NOTE:**

1. See **Figure 10.14** for details on the GiTOR bit, GiRIR bit and SRTiR bit.

**Table 21.9 Clock Settings (Communication Unit 0)**

Transfer Clock <sup>(1)</sup>	CCS Register	
	CCS0 Bit	CCS1 Bit
f1	1	0
f8	1	1
f <sub>2n</sub> <sup>(2)</sup>	0	1

**NOTES:**

1. The transfer clock for reception is generated when the RSHTTE bit in the G0ERC register is set to "1" (receive shift operation enabled).
2. The CNT3 to CNT0 bits in the TCSPR register select no division ( $n=0$ ) or divide-by- $2^n$  ( $n=1$  to 15).

**Table 21.10 Clock Settings (Communication Unit 1)**

Transfer Clock <sup>(1)</sup>	CCS Register	
	CCS2 Bit	CCS3 Bit
f1	1	0
f8	1	1
f <sub>2n</sub> <sup>(2)</sup>	0	1

**NOTES:**

1. The transfer clock for reception is generated when the RSHTTE bit in the G1ERC register is set to "1" (receive shift operation enabled).
2. The CNT3 to CNT0 bits in the TCSPR register select no division ( $n=0$ ) or divide-by- $2^n$  ( $n=1$  to 15).

**Table 21.11 Register Settings in HDLC Processing Mode**

Register	Bit	Function
GiMR	GMD1, GMD0	Set to "112"
	CKDIR	Set to "0"
	UFORM	Set to "0"
	IRS	Select what causes the transmit interrupt to be generated
GiEMR	7 to 0	Set to "1111 01102"
GiCR	TI	Transmit buffer empty flag
	TXEPT	Transmit register empty flag
	RI	Receive complete flag
	TE	Transmit enable bit
	RE	Receive enable bit
GiETC	TCRCE	Select whether transmit CRC is used or not
	TBSF1, TBSF0	Transmit bit stuffing
GiERC	CMP2E to CMP0E	Select whether received data is compared or not
	CMP3E	Set to "1"
	RCRCE	Select whether receive CRC is used or not
	RSHTE	Set to "1" to use it in the receiver
	RBSF1, RBSF0	Receive bit stuffing
GiIRF	BSERR	Set to "0"
	IRF3 to IRF0	Select what causes an interrupt to be generated
GiCMP0, GiCMP1	7 to 0	Write "FE16" to abort processing
GiCMP2	7 to 0	Data to be compared
GiCMP3	7 to 0	Write "7E16"
GiMSK0, GiMSK1	7 to 0	Write "0116" to abort processing
GiTCRC	15 to 0	Transmit CRC calculation result can be read
GiRCRC	15 to 0	Receive CRC calculation result can be read
GiTO	7 to 0	Data, which is output from a transmit data generation circuit, can be read
GiRI	7 to 0	Set data input to a receive data generation circuit
GiRB	7 to 0	Received data is stored
GiTB	7 to 0	For transmission: write data to be transmitted For reception: received data for comparison is stored
CCS	CCS1, CCS0	Select the HDLC processing clock
	CCS3, CCS2	Select the HDLC processing clock

i=0, 1



## 22. Programmable I/O Ports

87 programmable I/O ports from ports P0 to P10 (excluding P85) are available. The direction registers determine each port status, input or output. The pull-up control registers determine whether the ports, divided into groups of four ports, are pulled up or not. P85 is an input port and no pull-up for this port is allowed. The P8\_5 bit in the P8 register indicates an  $\overline{\text{NMI}}$  input level since P85 shares pins with  $\overline{\text{NMI}}$ .

Figures 22.1 to 22.4 show programmable I/O port configurations.

Each pin functions as the programmable I/O port, an I/O pin for internal peripheral functions or the bus control pin.

To use the pins as input or output pins for internal peripheral functions, refer to the explanations for each function. Refer to **7. Bus** when used as the bus control pin.

The registers associated with the programmable I/O ports are as follows.

### 22.1 Port Pi Direction Register (PDi Register, i=0 to 10)

Figure 22.5 shows the PDi register.

The PDi register selects input or output status of a programmable I/O port. Each bit in the PDi register corresponds to a port.

In memory expansion and microprocessor mode, the PDi register cannot control pins being used as bus control pins (A0 to A22,  $\overline{\text{A23}}$ , D0 to D15,  $\overline{\text{CS0}}$  to  $\overline{\text{CS3}}$ ,  $\overline{\text{WRL/WR}}$ ,  $\overline{\text{WRH/BHE}}$ ,  $\overline{\text{RD}}$ , BCLK/ALE/CLKOUT,  $\overline{\text{HLDA}}$ /ALE, HOLD, ALE and RDY). No bit controlling P85 is provided in the direction registers.

### 22.2 Port Pi Register (Pi Register, i=0 to 10)

Figure 22.6 shows the Pi register.

The Pi register writes and reads data to communicate with external devices. The Pi register consists of a port latch to hold output data and a circuit to read pin states. Each bit in the Pi register corresponds to a port. In memory expansion and microprocessor mode, the Pi register cannot control pins being used as bus control pins (A0 to A22,  $\overline{\text{A23}}$ , D0 to D15,  $\overline{\text{CS0}}$  to  $\overline{\text{CS3}}$ ,  $\overline{\text{WRL/WR}}$ ,  $\overline{\text{WRH/BHE}}$ ,  $\overline{\text{RD}}$ , BCLK/ALE/CLKOUT,  $\overline{\text{HLDA}}$ /ALE, HOLD, ALE and RDY).

### 22.3 Function Select Register Aj (PSj Register) (j=0 to 3)

Figures 22.7 and 22.8 show the PSj registers.

The PSj register selects either I/O port or peripheral function output if an I/O port shares pins with a peripheral function output (excluding DA0 and DA1.)

When multiple peripheral function outputs are assigned to a pin, set the PSL0 to PSL3, PSC, PSC3, and PSD1 registers to select which function is used.

Tables 22.3 to 22.10 list peripheral function output control settings for each pin.

### 22.4 Function Select Register B0 to B3 (PSL0 to PSL3 Registers)

Figures 22.9 and 22.10 show the PSL0 to PSL3 registers.

When multiple peripheral function outputs are assigned to a pin, the PSL0 to PSL3 registers select which peripheral function output is used.

Refer to **22.10 Analog Input and Other Peripheral Function Input** for the PSL3\_6 to PSL3\_3 bits in the PSL3 register.

## 22.5 Function Select Register C (PSC and PSC3 Registers)

Figures 22.11 and 22.12 show the PSC and PSC3 registers.

When multiple peripheral function outputs are assigned to a pin, the PSC and PSC3 registers select which peripheral function output is used.

Refer to **22.10 Analog Input and Other Peripheral Function Input** for the PSC\_7 bit in the PSC register.

## 22.6 Function Select Register D (PSD1 Register)

Figure 22.12 shows the PSD1 register.

When multiple peripheral function outputs are assigned to a pin, the PSD1 register selects which peripheral function output is used.

## 22.7 Pull-up Control Register 0 to 3 (PUR0 to PUR3 Registers)

Figures 22.13 and 22.14 show the PUR0 to PUR3 registers.

The PUR0 to PUR3 registers select whether the ports, divided into groups of four ports, are pulled up or not. Ports with bits in the PUR0 to PUR3 registers set to "1" (pull-up) and the direction registers set to "0" (input mode) are pulled up.

Set bits in the PUR0 and PUR1 registers in ports P0 to P5, running as bus, to "0" (no pull-up) in memory expansion mode and microprocessor mode. Ports P0, P1 and P40 to P43 can be pulled up when they are used as input ports in memory expansion mode and microprocessor mode.

## 22.8 Port Control Register (PCR Register)

Figure 22.14 shows the PCR register.

The PCR register selects either CMOS output or N-channel open drain output as port P1 output format. If the PCR0 bit is set to "1", N-channel open drain output is selected because the P-channel in the CMOS port is turned off. This is, however, not a perfect open drain. Therefore, the absolute maximum rating of the input voltage is between -0.3V and  $V_{CC2} + 0.3V$ .

If P1 is used as a port for data bus in memory expansion mode and microprocessor mode, set the PCR0 bit to "0". If P1 is used as a port in memory expansion mode and microprocessor mode, the PCR0 bit determines the output format.

## 22.9 Analog Input and Other Peripheral Function Input

The PSL3\_6 to PSL3\_3 bits in the PSL3 register and the PSC\_7 bit in the PSC register each separate analog I/O ports from other peripheral functions. Setting the corresponding bit to "1" (analog I/O) to use the analog I/O port (DA0, DA1, ANEX0, ANEX1, AN4 to AN7) prevents an intermediate potential from being impressed to other peripheral functions. The impressed intermediate potential may cause increase in power consumption.

Set the corresponding bit to "0" (except analog I/O) when analog I/O is not used. All peripheral function inputs except the analog I/O port are available when the corresponding bit is set to "0". These inputs are indeterminate when the bit is set to "1". When the PSC\_7 bit is set to "1", key input interrupt request remains unchanged regardless of  $\overline{KI0}$  to  $\overline{KI3}$  pin input level change.

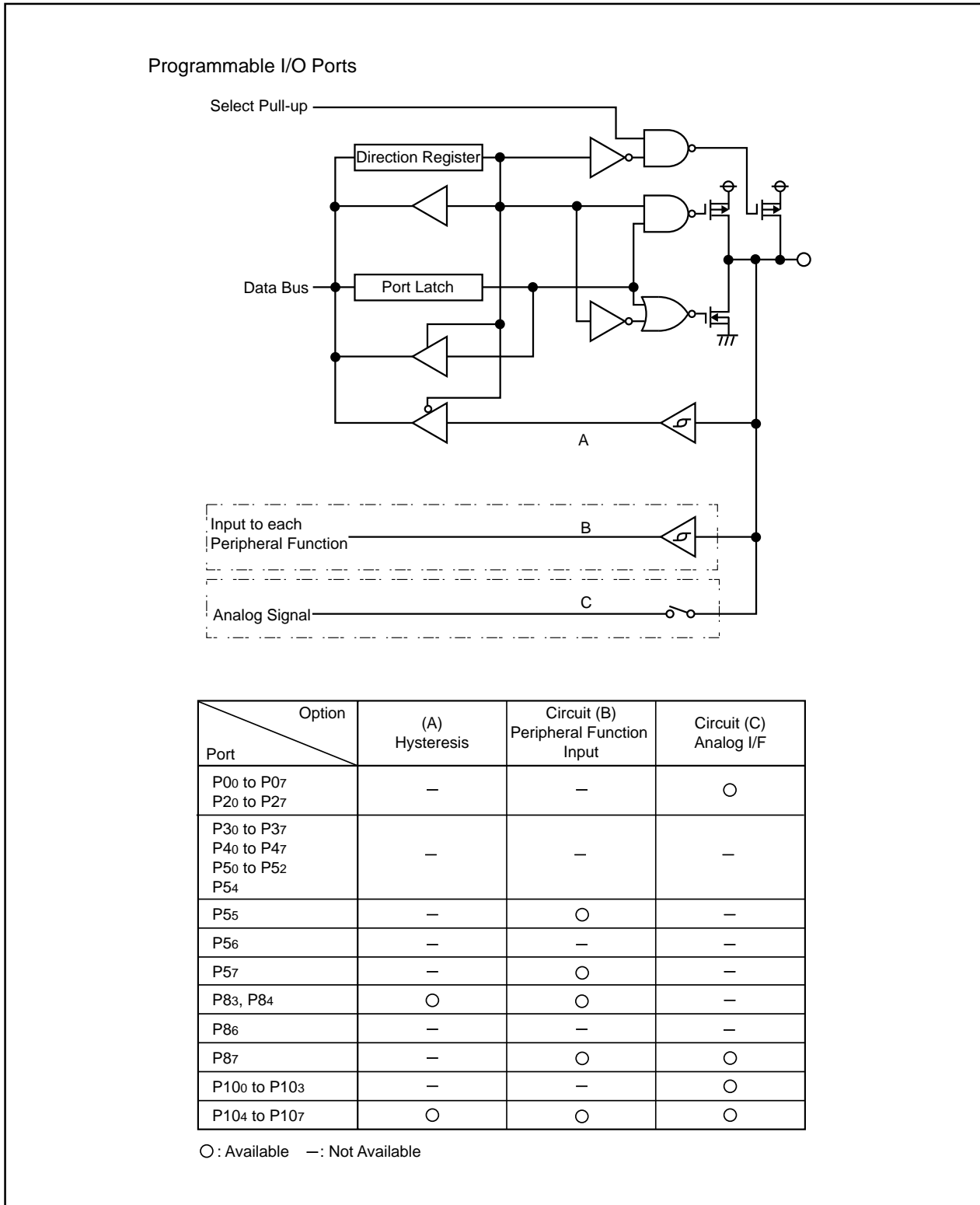
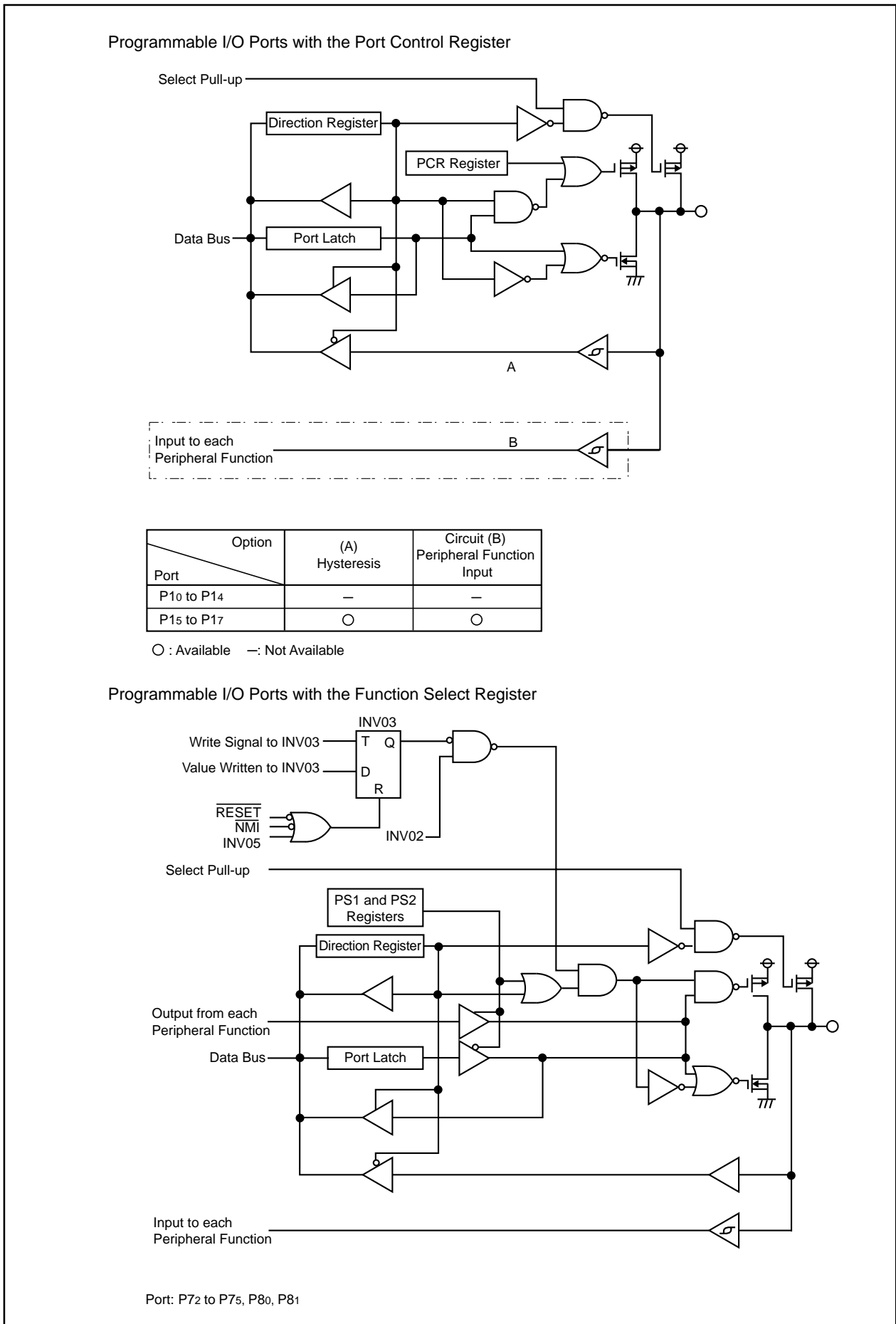


Figure 22.1 Programmable I/O Ports (1)



**Figure 22.2 Programmable I/O Ports (2)**

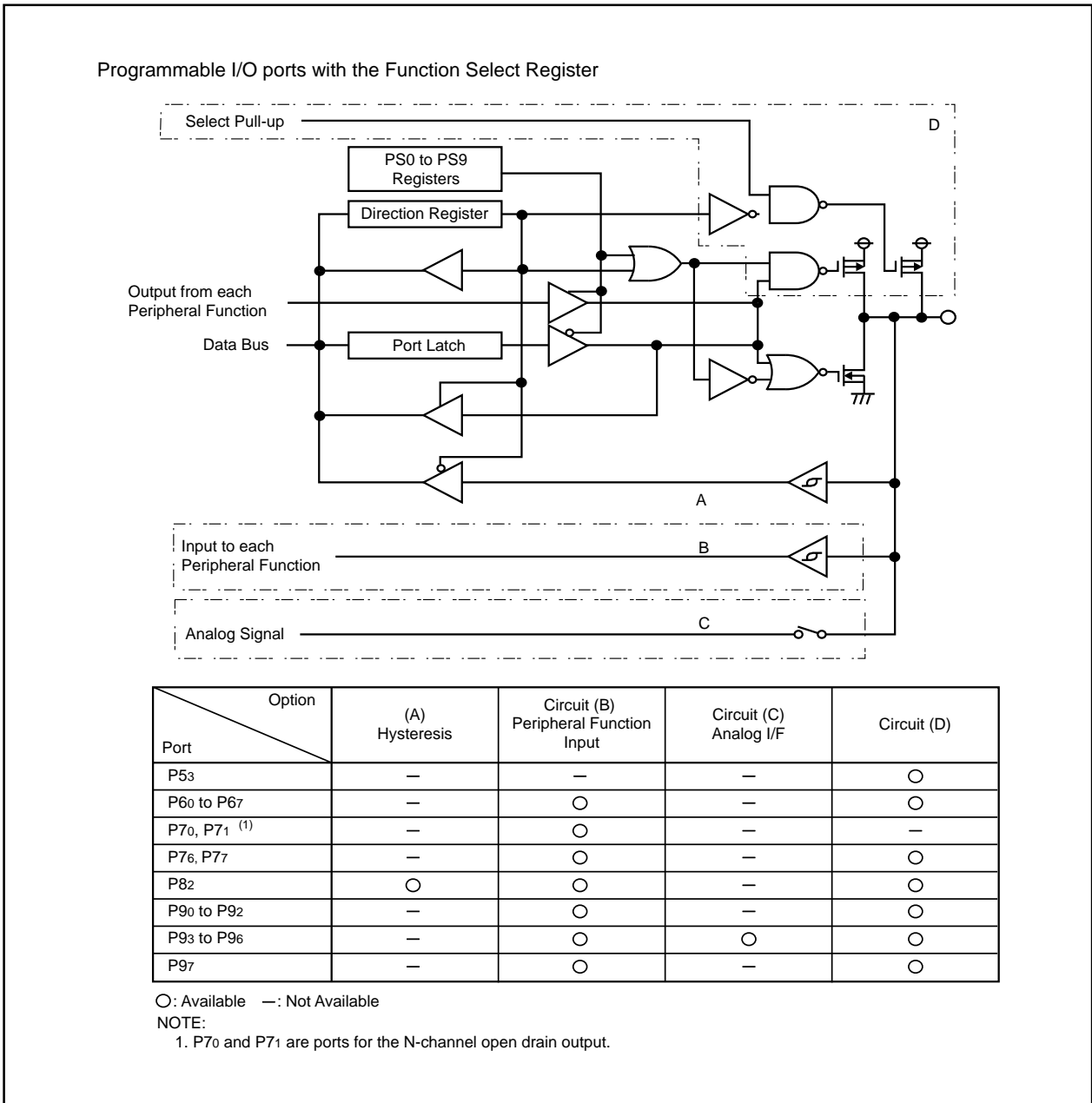


Figure 22.3 Programmable I/O Ports (3)

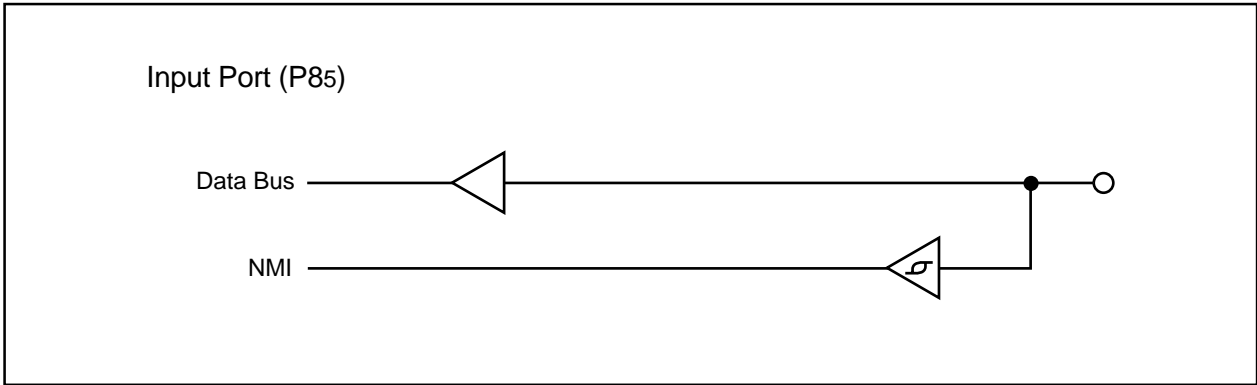


Figure 22.4 Programmable I/O Ports (4)

**Port Pi Direction Register (i=0 to 10) <sup>(2)</sup>**

Bit	Symbol	Address	After Reset
b7			
b6			
b5			
b4			
b3			
b2			
b1			
b0			
Symbol		Address	After Reset
PD0 to PD3		03E2 <sub>16</sub> , 03E3 <sub>16</sub> , 03E6 <sub>16</sub> , 03E7 <sub>16</sub>	00 <sub>16</sub>
PD4 to PD7		03EA <sub>16</sub> , 03EB <sub>16</sub> , 03C2 <sub>16</sub> , 03C3 <sub>16</sub>	00 <sub>16</sub>
PD8		03C6 <sub>16</sub> <sup>(3)</sup>	00X0 0000 <sub>2</sub>
PD9, PD10		03C7 <sub>16</sub> <sup>(1)</sup> , 03CA <sub>16</sub>	00 <sub>16</sub>

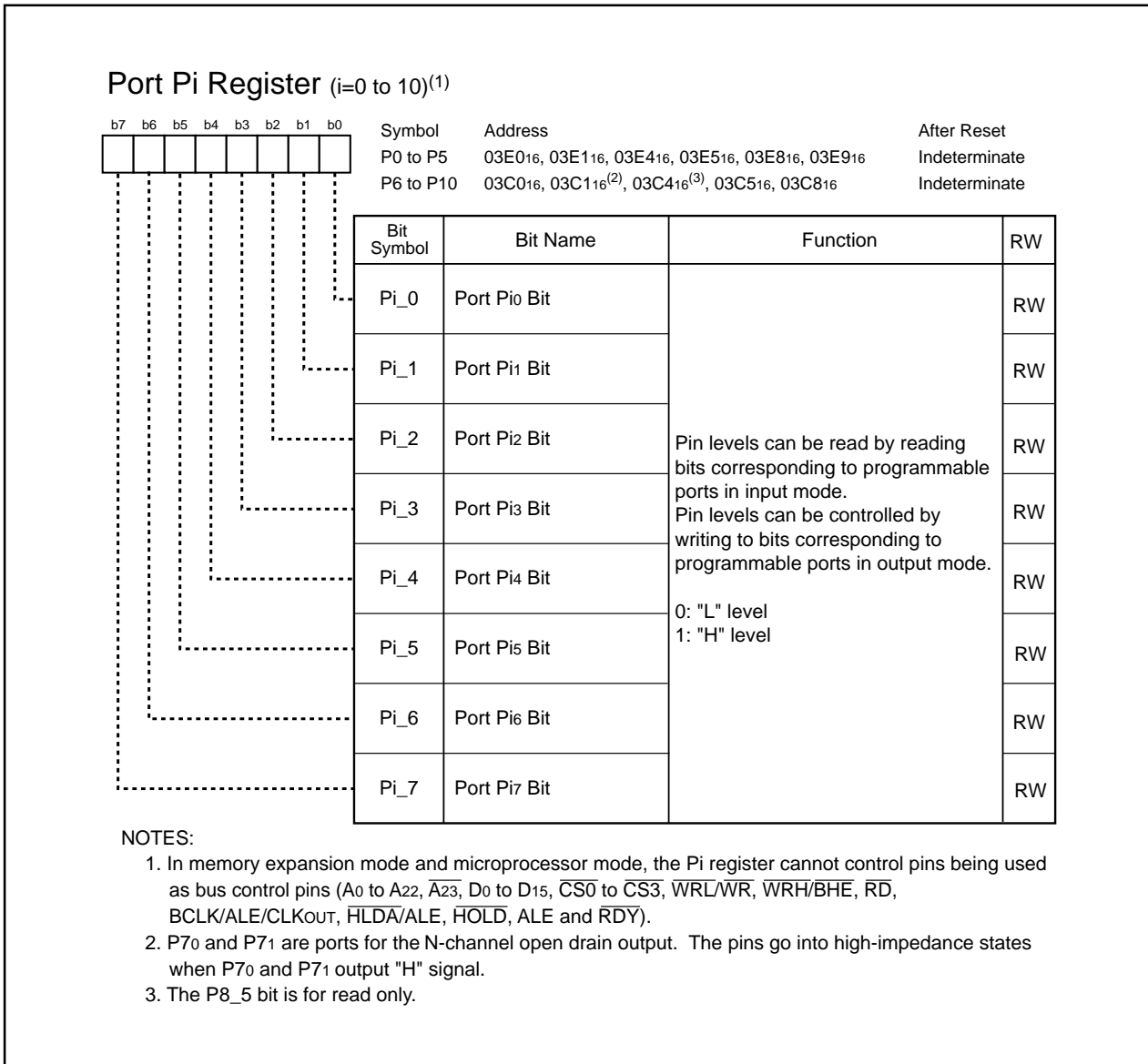
  

Bit Symbol	Bit Name	Function	RW
PDi_0	Port Pi0 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW
PDi_1	Port Pi1 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW
PDi_2	Port Pi2 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW
PDi_3	Port Pi3 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW
PDi_4	Port Pi4 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW
PDi_5	Port Pi5 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW
PDi_6	Port Pi6 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW
PDi_7	Port Pi7 Direction Bit	0: Input mode (Functions as input port) 1: Output mode (Functions as output port)	RW

**NOTES:**

- Set the PD9 register immediately after the PRC2 bit in the PRCR register is set to "1" (write enabled). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 register.
- In memory expansion mode and microprocessor mode, the PDi register cannot control pins being used as bus control pins (A0 to A22,  $\overline{A23}$ , D0 to D15,  $\overline{CS0}$  to  $\overline{CS3}$ ,  $\overline{WRL}/\overline{WR}$ ,  $\overline{WRH}/\overline{BHE}$ ,  $\overline{BCLK}/\overline{ALE}/\overline{CLKOUT}$ ,  $\overline{RD}$ ,  $\overline{HLDA}/\overline{ALE}$ ,  $\overline{HOLD}$ ,  $\overline{ALE}$  and  $\overline{RDY}$ ).
- Nothing is assigned in the PD8\_5 bit in the PD8 register. If write, set these bits to "0". When read, their contents are indeterminate.

Figure 22.5 PD0 to PD10 Registers



**Figure 22.6 P0 to P10 Registers**

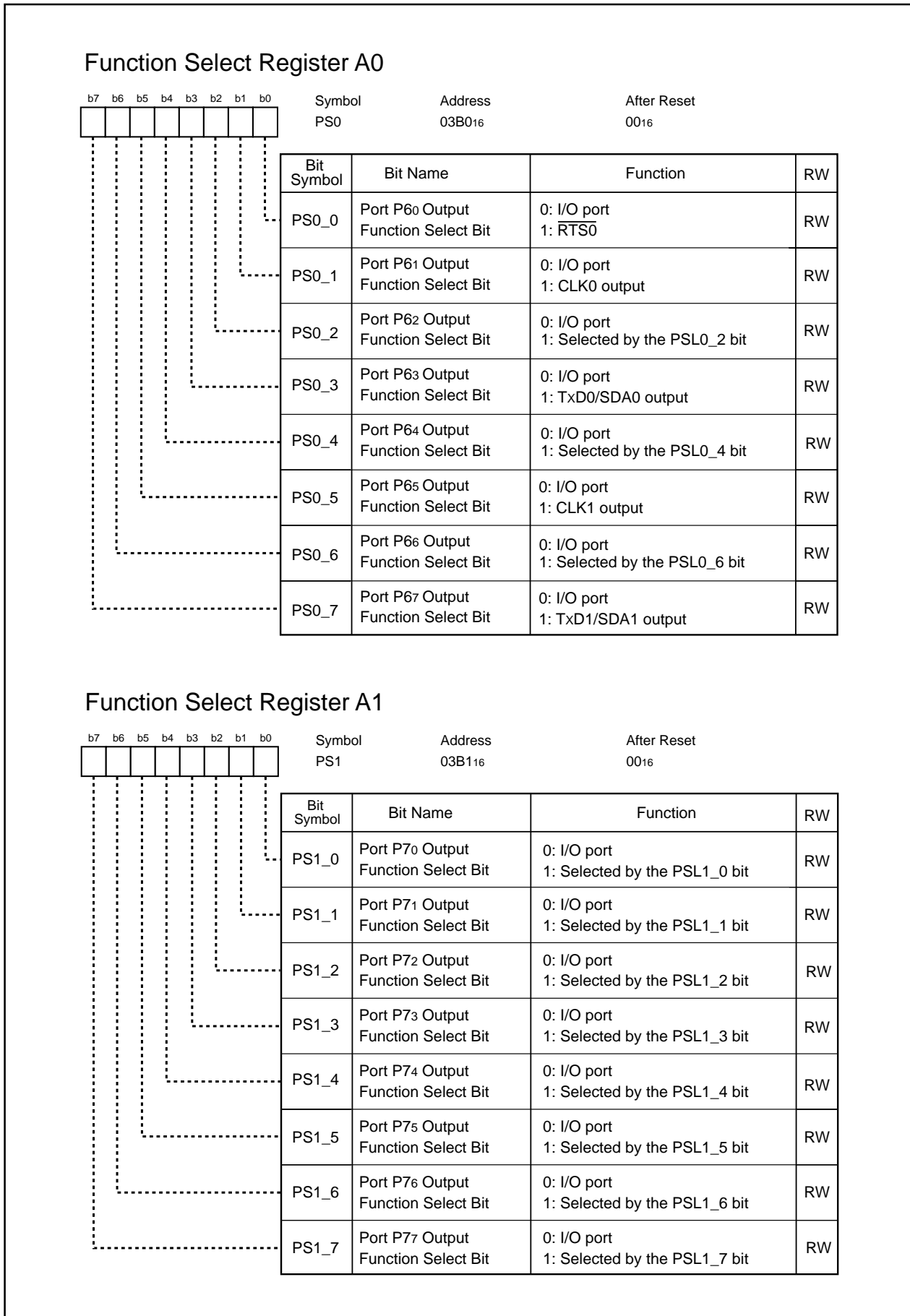


Figure 22.7 PS0 Register and PS1 Register



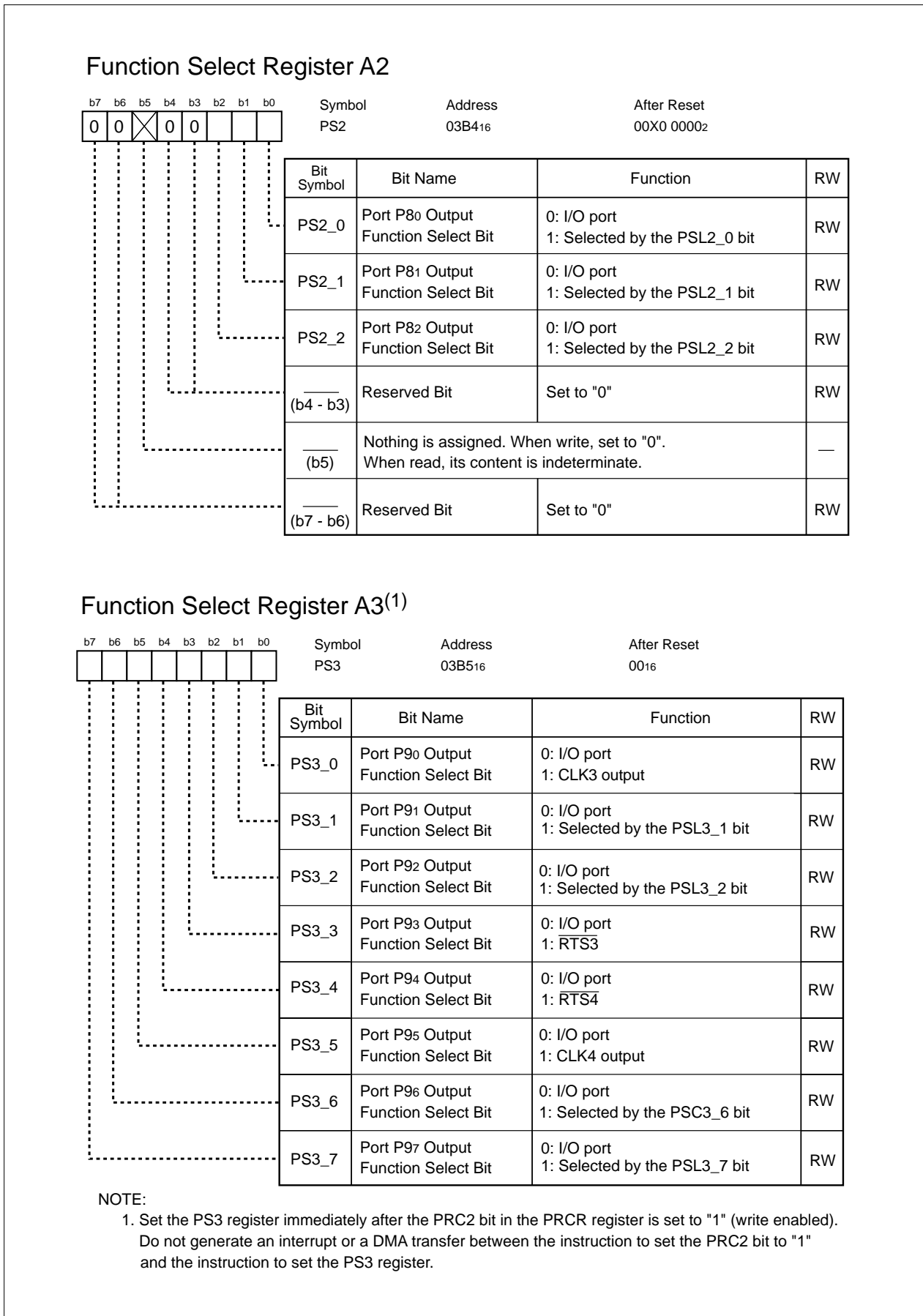


Figure 22.8 PS2 Register and PS3 Register

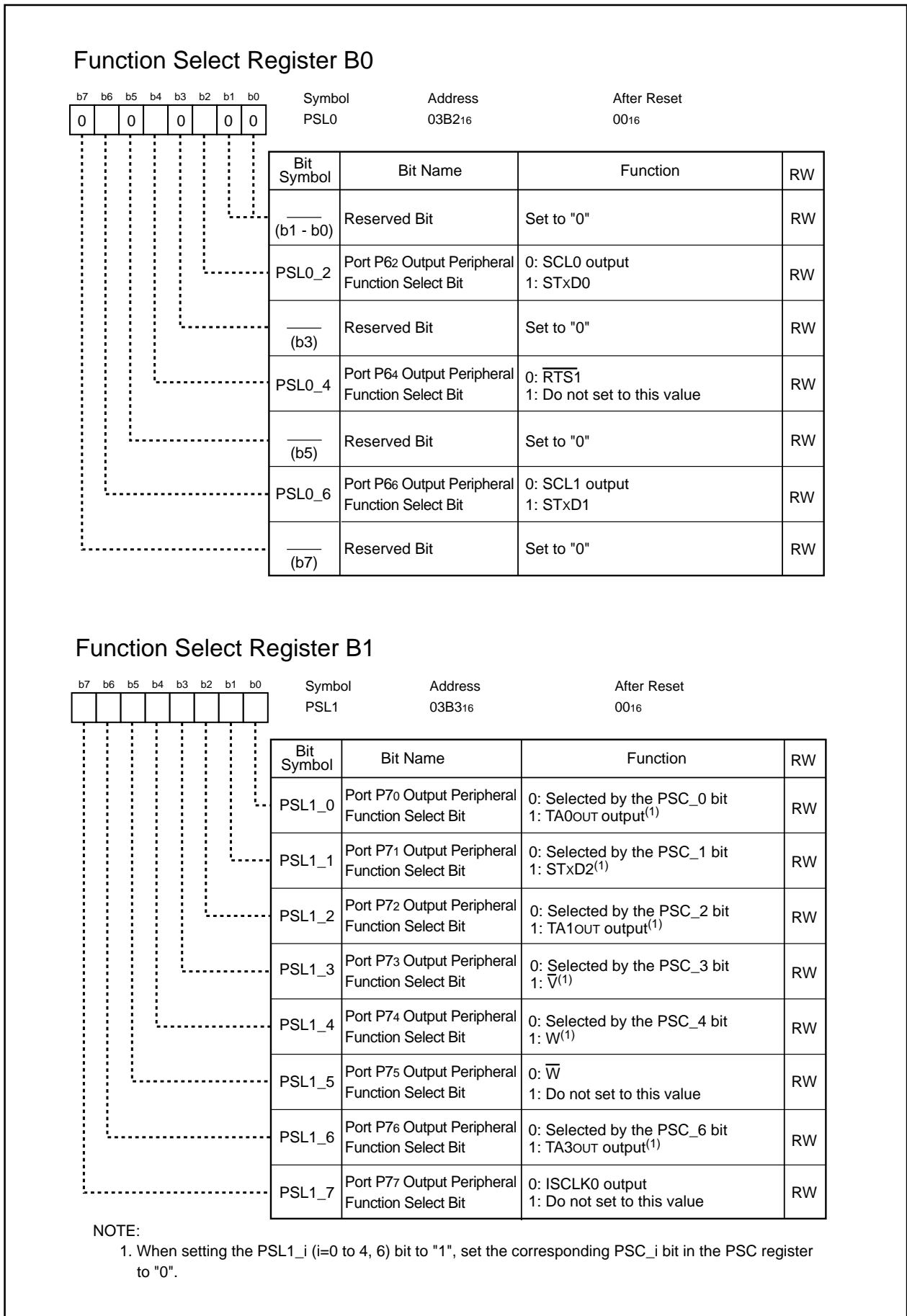


Figure 22.9 PSL0 Register and PSL1 Register

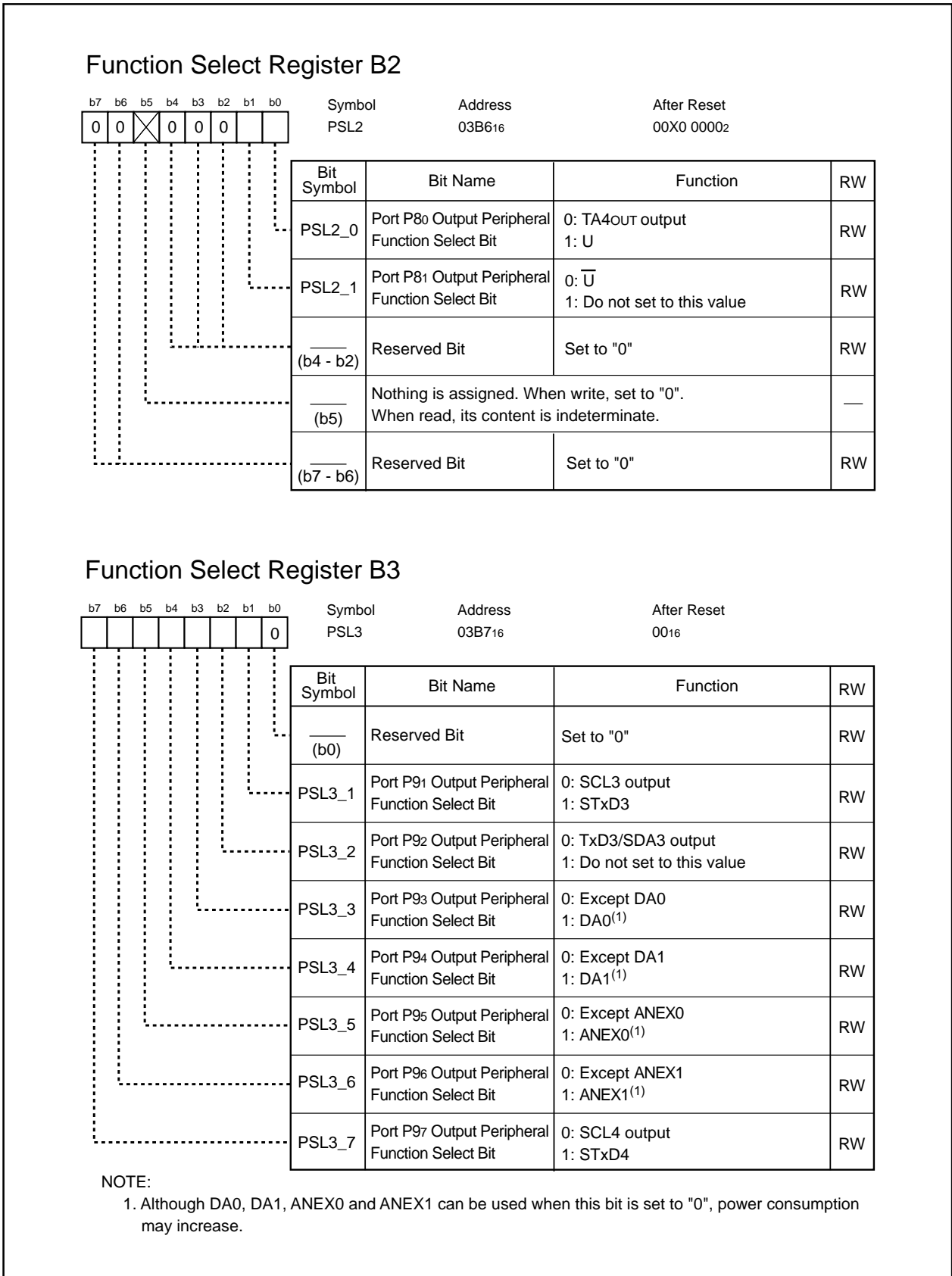
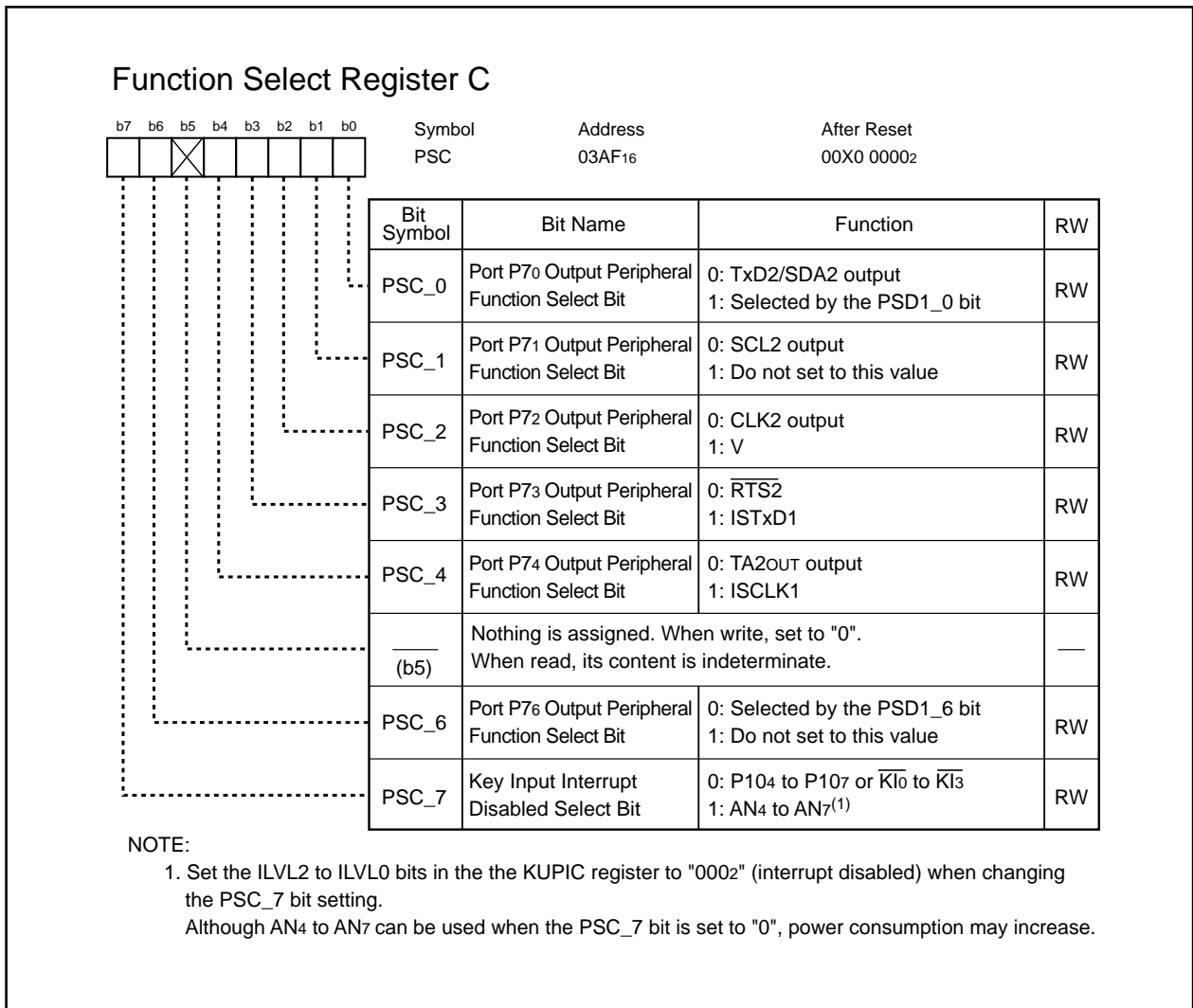


Figure 22.10 PSL2 Register and PSL3 Register



**Figure 22.11 PSC Register**

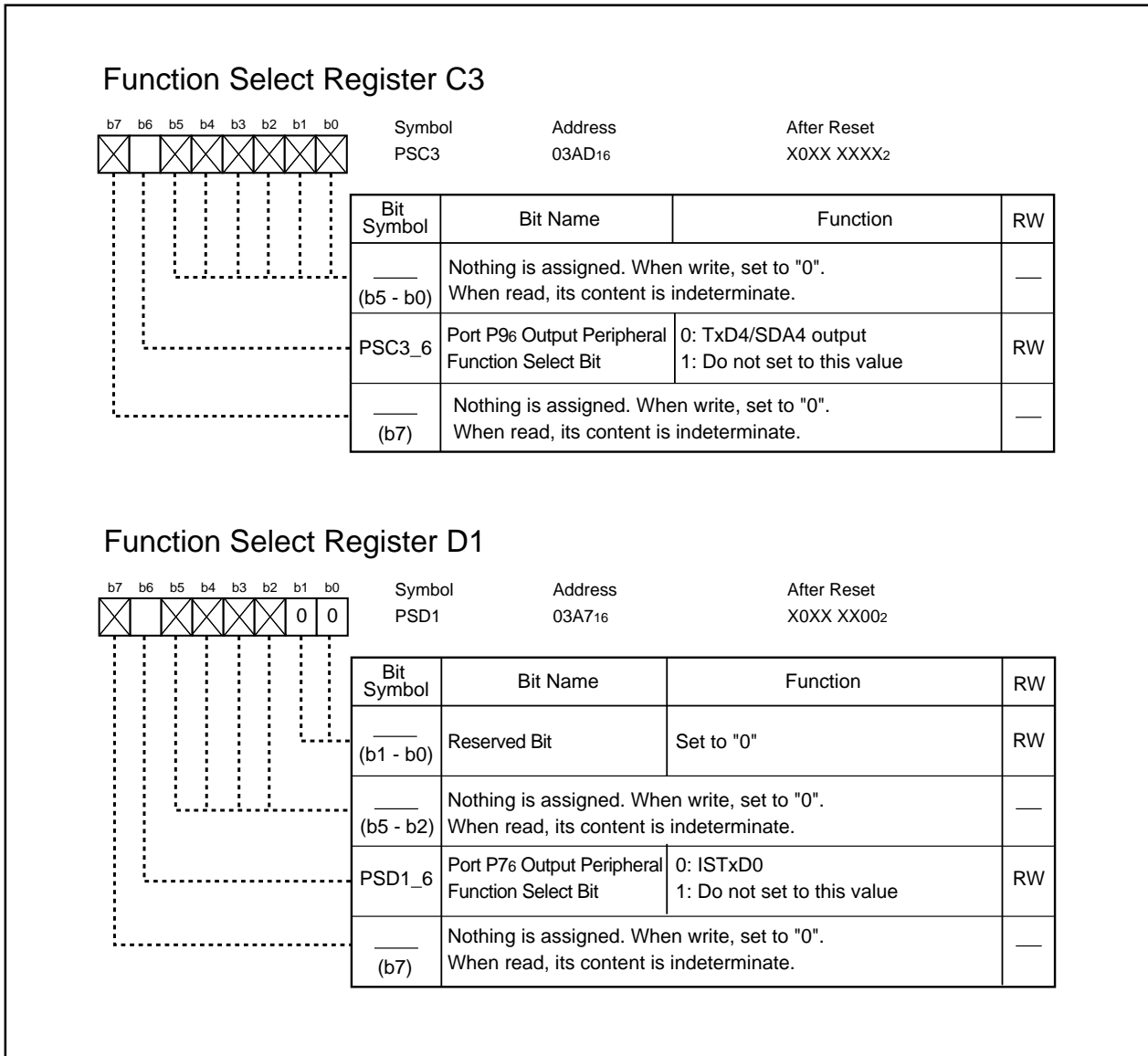
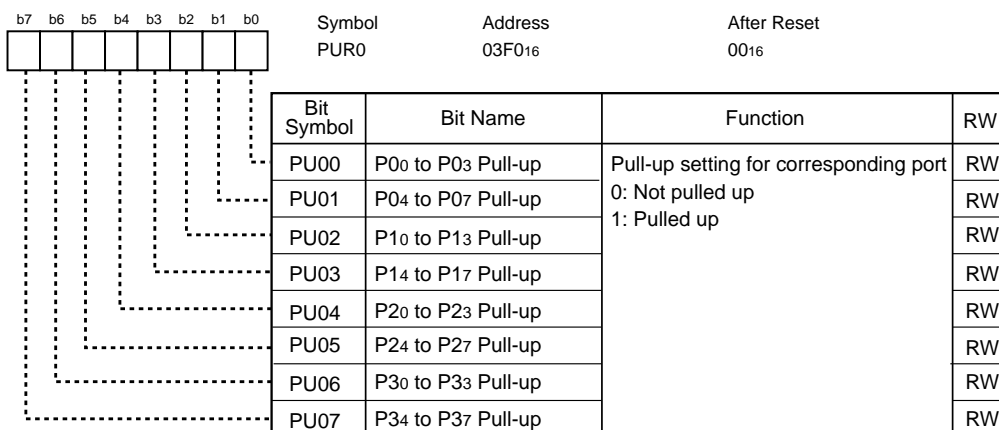


Figure 22.12 PSC3 Register and PSD1 Register

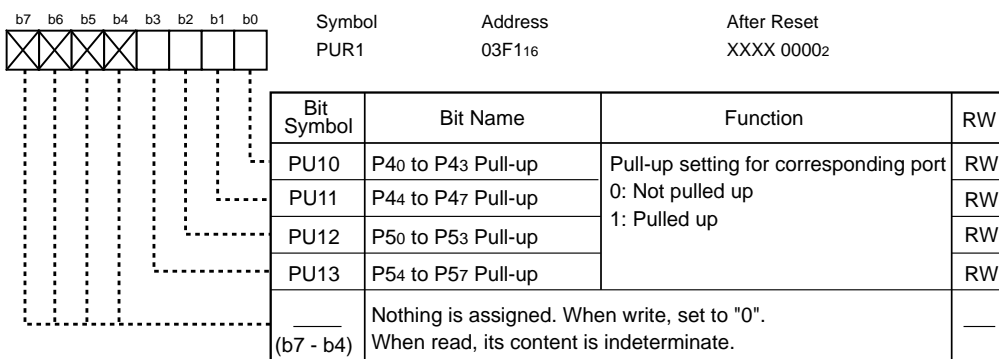
### Pull-Up Control Register 0<sup>(1)</sup>



**NOTE:**

- Set each bit in the PUR0 register to "0" when ports P0 to P5 become bus control pins in memory expansion mode and microprocessor mode. When using the ports as I/O ports, pull-up or no pull-up setting can be selected.

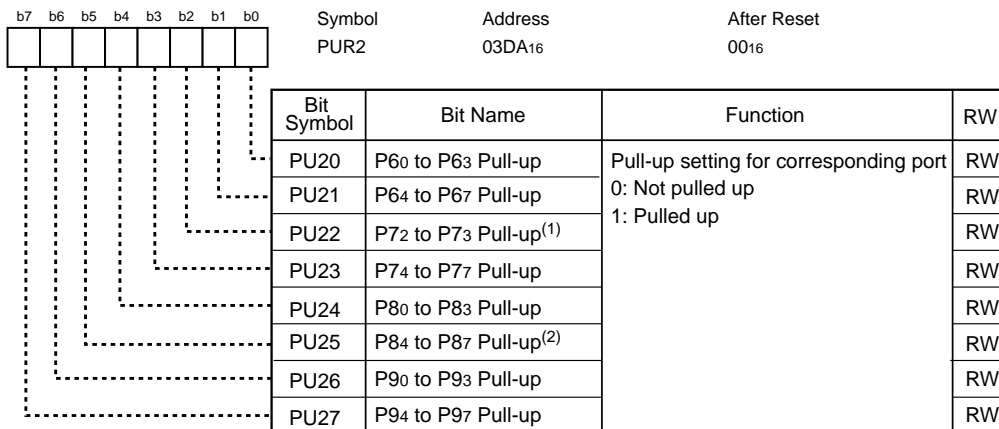
### Pull-Up Control Register 1<sup>(1)</sup>



**NOTES:**

- Set each bit in the PUR1 register to "0" when ports P0 to P5 become bus control pins in memory expansion mode and microprocessor mode. When using the ports as I/O ports, pull-up or no pull-up setting can be selected.

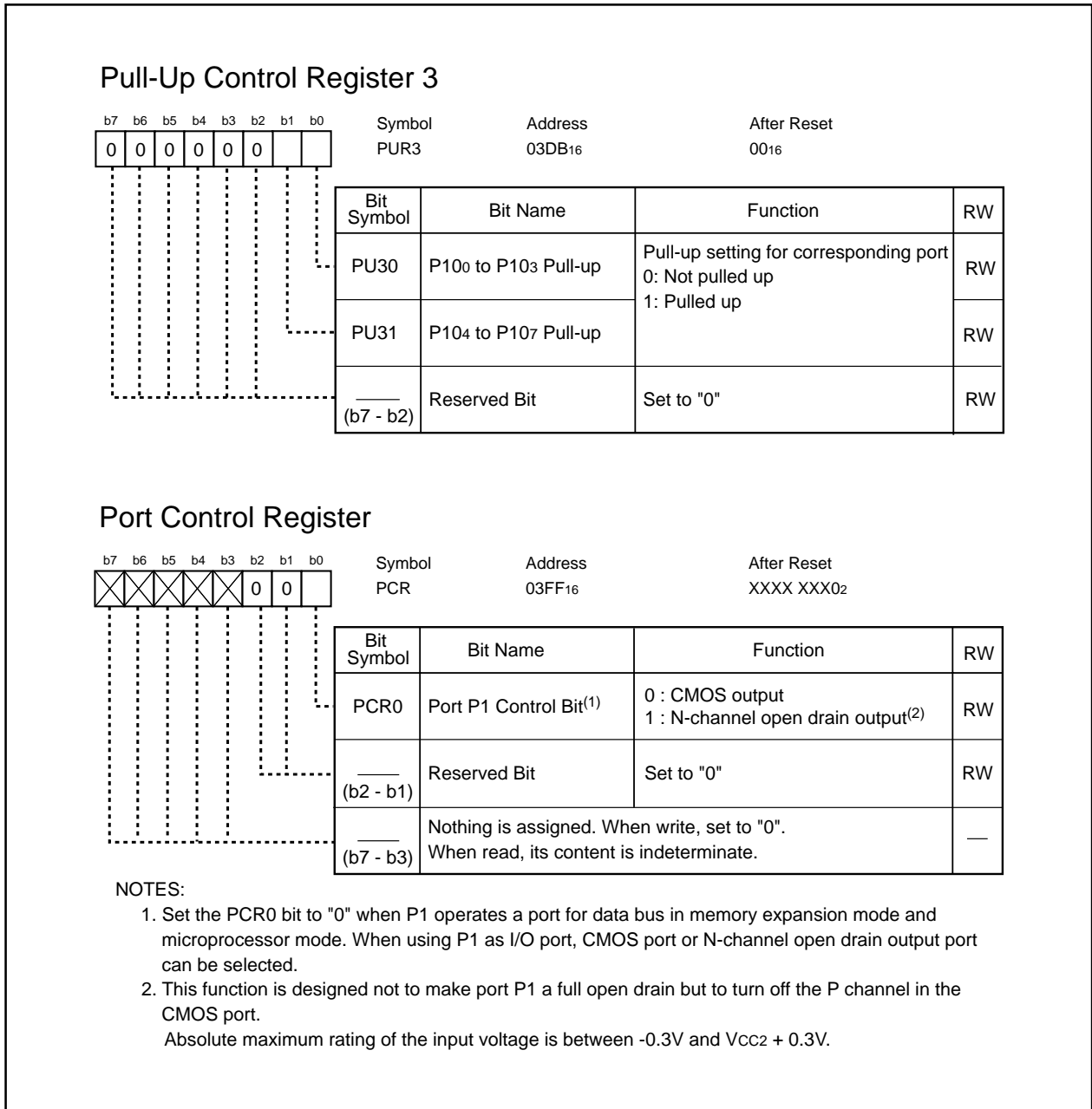
### Pull-Up Control Register 2



**NOTES:**

- P7<sub>0</sub> and P7<sub>1</sub> cannot be pulled up.
- P8<sub>5</sub> cannot be pulled up.

**Figure 22.13 PUR0 Register, PUR1 Register and PUR2 Register**



**Figure 22.14 PUR3 Register and PCR Register**

**Table 22.1 Unassigned Pin Settings in Single-Chip Mode**

Pin Name	Setting
P0 to P10 (excluding P85) <sup>(1,2,3)</sup>	Enter input mode and connect each pin to VSS via a resistor (pull-down); or enter output mode and leave pins open
XOUT <sup>(4)</sup>	Leave pin open
NMI(P85)	Connect pin to VCC1 via a resistor (pull-up)
AVCC	Connect pin to VCC1
AVSS, VREF, BYTE	Connect pins to VSS

## NOTES:

1. If the port enters output mode and is left open, it is in input mode before output mode is entered by program after reset. While the port is in input mode, voltage level on the pins is indeterminate and power consumption may increase.  
Direction register settings may be changed by noise or failure caused by noise. Configure direction register settings regularly to increase the reliability of the program.
2. Use the shortest possible wiring to connect the microcomputer pins to unassigned pins (within 2 cm).
3. P70 and P71 must output low-level ("L") signals if they are in output mode. They are ports N-channel open drain outputs.
4. When the external clock is applied to the XIN pin, set the pin as written above.

**Table 22.2 Unassigned Pin Setting in Memory Expansion Mode and Microprocessor Mode**

Pin Name	Setting
P6 to P10 (excluding P85) <sup>(1,2,3)</sup>	Enter input mode and connect each pin to VSS via a resistor (pull-down); or enter output mode and leave pins open
BHE, ALE, HLDA, XOUT <sup>(5)</sup> , BCLK	Leave pin open
NMI(P85)	Connect pin to VCC1 via a resistor (pull-up)
RDY, HOLD	Connect pins to VCC2 via a resistor (pull-up)
AVCC	Connect pin to VCC1
AVSS, VREF	Connect pins to VSS

## NOTES:

1. If the port enters output mode and is left open, it is in input mode before output mode is entered by program after reset. While the port is in input mode, voltage level on the pins is indeterminate and power consumption may increase.  
Direction register settings may be changed by noise or failure caused by noise. Configure direction register settings regularly to increase the reliability of the program.
2. Use the shortest possible wiring to connect the microcomputer pins to unassigned pins (within 2 cm).
3. P70 and P71 must outputs low-level ("L") signals if they are in output mode. They are N-channel open-drain outputs.
4. When the external clock is applied to the XIN pin, set the pin as written above.



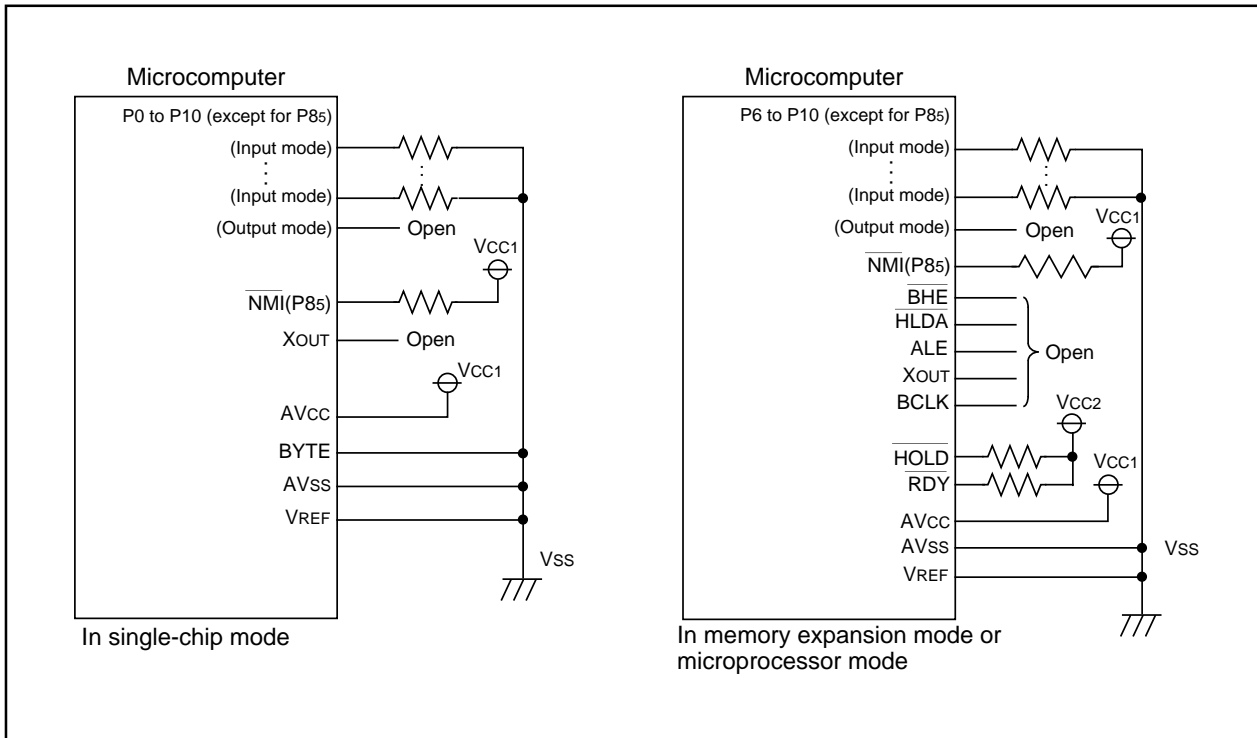


Figure 22.15 Unassigned Pin Handling

**Table 22.3 Port P6 Peripheral Function Output Control**

	PS0 Register	PSL0 Register
Bit 0	0: P60/CTS0/SS0 1: RTS0	Set to "0"
Bit 1	0: P61/CLK0(input) 1: CLK0(output)	Set to "0"
Bit 2	0: P62/RxD0/SCL0(input) 1: Selected by the PSL0 register	0: SCL0(output) 1: STxD0
Bit 3	0: P63/SRxD0/SDA0(input) 1: TxD0/SDA0 (output)	Set to "0"
Bit 4	0: P64/CTS1/SS1 1: Selected by the PSL0 register	0: RTS1 1: Do not set this value
Bit 5	0: P65/CLK1(input) 1: CLK1(output)	Set to "0"
Bit 6	0: P66/RxD1/SCL1(input) 1: Selected by the PSL0 register	0: SCL1(output) 1: STxD1
Bit 7	0: P67/SRxD1/SDA1(input) 1: TxD1/SDA1(output)	Set to "0"

**Table 22.4 Port P7 Peripheral Function Output Control**

	PS1 Register	PSL1 Register	PSC Register <sup>(1)</sup>	PSD1 Register
Bit 0	0: P70/TA0OUT(input)/SRxD2/ SDA2 (input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: TA0OUT(output)	0: TxD2/SDA2(output) 1: Do not set to this value	Set to "0"
Bit 1	0: P71/TB5IN/TA0IN/RxD2/ SCL2 (input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: STxD2	0: SCL2(output) 1: Do not set to this value	Set to "0"
Bit 2	0: P72/TA1OUT(input)/ CLK2(input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: TA1OUT(output)	0: CLK2(output) 1: V	Set to "0"
Bit 3	0: P73/TA1IN/CTS2/SS2/ 1: Selected by the PSL1 register	0: Selected by the PSC register 1: $\bar{V}$	0: RTS2 1: ISTxD1	Set to "0"
Bit 4	0: P74/ISCLK1(input)/ TA2OUT(input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: W	0: TA2OUT(output) 1: ISCLK1(output)	Set to "0"
Bit 5	0: P75/TA2IN/ISRxD1 1: Selected by the PSL1 register	0: $\bar{W}$ 1: Do not set to this value	Set to "0"	Set to "0"
Bit 6	0: P76/TA3OUT(input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: TA3OUT(output)	0: Selected by the PSD1 register 1: Do not set to this value	0: ISTxD0 1: Do not set to this value
Bit 7	0: P77/TA3IN/ISCLK0(input) 1: Selected by the PSL1 register	0: ISCLK0(output) 1: Do not set to this value	0: P104 to P107 or $\bar{K}I0$ to $\bar{K}I3$ 1: AN4 to AN7 (No relation to P77)	Set to "0"

**NOTE:**

1. When setting the PSL1\_i bit (i=0 to 4, 6) to "1", set the corresponding PSC\_i bit to "0".

**Table 22.5 Port P8 Peripheral Function Output Control**

	PS2 Register	PSL2 Register
Bit 0	0: P80/ISRxD0/TA4OUT(input) 1: Selected by the PSL2 register	0: TA4OUT(output) 1: U
Bit 1	0: P81/TA4IN 1: Selected by the PSL2 register	0: $\bar{U}$ 1: Do not set to this value
Bit 2	0: P82/INT0 1: Selected by the PSL2 register	Set to "0"
Bit 3 to 7	Set to "000002"	

**Table 22.6 Port P9 Peripheral Function Output Control**

	PS3 Register	PSL3 Register	PSC3 Register
Bit 0	0: P90/TB0IN/CLK3(input) 1: CLK3(output)	Set to "0"	Set to "0"
Bit 1	0: P91/TB1IN/RxD3/SCL3(input) 1: Selected by the PSL3 register	0: SCL3(output) 1: STxD3	Set to "0"
Bit 2	0: P92/TB2IN/SRxD3/SDA3(input) 1: Selected by the PSL3 register	0: TxD3/SDA3(output) 1: Do not set to this value	Set to "0"
Bit 3	0: P93/TB3IN/CTS3/SS3/DA0(output) 1: RTS3	0: Except DA0 1: DA0	Set to "0"
Bit 4	0: P94/TB4IN/CTS4/SS4/DA1(output) 1: RTS4	0: Except DA1 1: DA1	Set to "0"
Bit 5	0: P95/ANEX0/CLK4(input)/ 1: CLK4(output)	0: Except ANEX0 1: ANEX0	Set to "0"
Bit 6	0: P96/SRxD4/ANEX1/SDA4(input) 1: Selected by the PSC3 register	0: Except ANEX1 1: ANEX1	0: TxD4/SDA4 output 1: Do not set to this value
Bit 7	0: P97/RxD4/ $\overline{ADTRG}$ /SCL4(input) 1: Selected by the PSL3 register	0: SCL4(output) 1: STxD4	Set to "0"

**Table 22.7 Port P10 Peripheral Function Output Control**

	PSC Register
Bit 7	0: P104 to P107 or $\bar{KI}0$ to $\bar{KI}3$ 1: AN4 to AN7

## 23. Electrical Characteristics

**Table 23.1 Absolute Maximum Ratings**

Symbol	Parameter	Condition	Value	Unit
V <sub>CC1</sub> , V <sub>CC2</sub>	Supply Voltage	V <sub>CC1</sub> =AV <sub>CC</sub>	-0.3 to 6.0	V
V <sub>CC2</sub>	Supply Voltage	-	-0.3 to V <sub>CC1</sub>	V
AV <sub>CC</sub>	Analog Supply Voltage	V <sub>CC1</sub> =AV <sub>CC</sub>	-0.3 to 6.0	V
V <sub>I</sub>	Input Voltage	RESET, CNV <sub>SS</sub> , BYTE, P60-P67, P72-P77, P80-P87, P90-P97, P100-P107, V <sub>REF</sub> , X <sub>IN</sub>	-0.3 to V <sub>CC1</sub> +0.3	V
		P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57	-0.3 to V <sub>CC2</sub> +0.3	
		P70, P71	-0.3 to 6.0	
V <sub>O</sub>	Output Voltage	P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107, X <sub>OUT</sub>	-0.3 to V <sub>CC1</sub> +0.3	V
		P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57	-0.3 to V <sub>CC2</sub> +0.3	
		P70, P71	-0.3 to 6.0	
P <sub>d</sub>	Power Dissipation	T <sub>opr</sub> =25° C	500	mW
T <sub>opr</sub>	Operating Ambient Temperature		-20 to 85/ -40 to 85 <sup>(1)</sup>	° C
T <sub>stg</sub>	Storage Temperature		-65 to 150	° C

**NOTE:**

- Contact our sales office if temperature range of -40 to 85° C is required.

**Table 23.2 Recommended Operating Conditions**  
**(V<sub>CC1</sub>= V<sub>CC2</sub>=3.0V to 5.5V at Topr=- 20 to 85°C unless otherwise specified)**

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> )	3.0	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>	Input High ("H") Voltage	P20-P27, P30-P37, P40-P47, P50-P57	0.8V <sub>CC2</sub>		V <sub>CC2</sub>	V
		P60-P67, P72-P77, P80-P87 <sup>(3)</sup> , P90-P97, P100-P107, X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE	0.8V <sub>CC1</sub>		V <sub>CC1</sub>	
		P70, P71	0.8V <sub>CC1</sub>		6.0	
		P00-P07, P10-P17 (in single-chip mode)	0.8V <sub>CC2</sub>		V <sub>CC2</sub>	
		P00-P07, P10-P17 (in memory expansion mode and microprocessor mode)	0.5V <sub>CC2</sub>		V <sub>CC2</sub>	
V <sub>IL</sub>	Input Low ("L") Voltage	P20-P27, P30-P37, P40-P47, P50-P57	0		0.2V <sub>CC2</sub>	V
		P60-P67, P70-P77, P80-P87 <sup>(3)</sup> , P90-P97, P100-P107, X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE	0		0.2V <sub>CC1</sub>	
		P00-P07, P10-P17 (in single-chip mode)	0		0.2V <sub>CC2</sub>	
		P00-P07, P10-P17 (in memory expansion mode and microprocessor mode)	0		0.16V <sub>CC2</sub>	
I <sub>OH(peak)</sub>	Peak Output High ("H") Current <sup>(2)</sup>	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107			-10.0	mA
I <sub>OH(avg)</sub>	Average Output High ("H") Current <sup>(1)</sup>	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107			-5.0	mA
I <sub>OL(peak)</sub>	Peak Output Low ("L") Current <sup>(2)</sup>	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P84, P86, P87, P90-P97, P100-P107			10.0	mA
I <sub>OL(avg)</sub>	Average Output Low ("L") Current <sup>(1)</sup>	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P84, P86, P87, P90-P97, P100-P107			5.0	mA

## NOTES:

- Typical values when average output current is 100 ms.
- Total I<sub>OL(peak)</sub> for P0, P1, P2, P86, P87, P9, and P10 must be 80 mA or less.  
 Total I<sub>OL(peak)</sub> for P3, P4, P5, P6, P7, and P80 to P84 must be 80 mA or less.  
 Total I<sub>OH(peak)</sub> for P0, P1, and P2 must be -40 mA or less.  
 Total I<sub>OH(peak)</sub> for P86, P87, P9, and P10 must be -40 mA or less.  
 Total I<sub>OH(peak)</sub> for P3, P4, and P5 must be -40 mA or less.  
 Total I<sub>OH(peak)</sub> for P6, P7, and P80 to P84 must be -40 mA or less.
- V<sub>IH</sub> and V<sub>IL</sub> reference for P87 applies when P87 is used as a programmable input port.  
 It does not apply when P87 is used as X<sub>CIN</sub>.

**Table 23.2 Recommended Operating Conditions (Continued)**  
**(V<sub>CC1</sub>=V<sub>CC2</sub>=3.0V to 5.5V at T<sub>opr</sub>=-20 to 85°C unless otherwise specified)**

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
f(BCLK)	CPU Operation Frequency	V <sub>CC1</sub> =4.2 to 5.5 V	0		32	MHz
		V <sub>CC1</sub> =3.0 to 5.5 V	0		24	MHz
f(XIN)	Main Clock Input Frequency	V <sub>CC1</sub> =4.2 to 5.5 V	0		32	MHz
		V <sub>CC1</sub> =3.0 to 5.5 V	0		24	MHz
f(XCIN)	Sub Clock Frequency		32.768	50	kHz	
f(Ring)	On-chip Oscillator Frequency (T <sub>opr</sub> =25° C)		0.5	1	2	MHz
f(PLL)	PLL Clock Frequency	V <sub>CC1</sub> =4.2 to 5.5 V	10		32	MHz
		V <sub>CC1</sub> =3.0 to 5.5 V	10		24	MHz
t <sub>SU(PLL)</sub>	Wait Time to Stabilize PLL Frequency Synthesizer	V <sub>CC1</sub> =5.0 V			5	ms
		V <sub>CC1</sub> =3.3 V			10	ms

$$V_{CC1}=V_{CC2}=5V$$

Table 23.3 Electrical Characteristics

( $V_{CC1}=V_{CC2}=4.2$  to  $5.5V$ ,  $V_{SS}=0V$  at  $T_{opr}=-20$  to  $85^{\circ}C$ ,  $f(BCLK)=32MHz$  unless otherwise specified)

Symbol	Parameter		Condition	Standard			Unit	
				Min.	Typ.	Max.		
V <sub>OH</sub>	Output High ("H") Voltage	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57	I <sub>OH</sub> =-5mA	V <sub>CC2</sub> -2.0		V <sub>CC2</sub>	V	
		P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107	I <sub>OH</sub> =-5mA	V <sub>CC1</sub> -2.0		V <sub>CC1</sub>		
		P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57	I <sub>OH</sub> =-200μA	V <sub>CC2</sub> -0.3		V <sub>CC2</sub>	V	
		P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107	I <sub>OH</sub> =-200μA	V <sub>CC1</sub> -0.3		V <sub>CC1</sub>		
		X <sub>OUT</sub>	I <sub>OH</sub> =-1mA	3.0		V <sub>CC1</sub>	V	
		X <sub>COUT</sub>	High Power	No load applied		2.5		V
		Low Power	No load applied		1.6			
V <sub>OL</sub>	Output Low ("L") Voltage	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P84, P86, P87, P90-P97, P100-P107	I <sub>OL</sub> =5mA			2.0	V	
		P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P84, P86, P87, P90-P97, P100-P107	I <sub>OL</sub> =200μA			0.45	V	
		X <sub>OUT</sub>	I <sub>OL</sub> =1mA			2.0	V	
		X <sub>COUT</sub>	High Power	No load applied		0		V
			Low Power	No load applied		0		
V <sub>T+</sub> -V <sub>T-</sub>	Hysteresis	HOLD, RDY, TA0 <sub>IN</sub> -TA4 <sub>IN</sub> , TB0 <sub>IN</sub> -TB5 <sub>IN</sub> , INT0-INT5, AD <sub>TRG</sub> , CTS0-CTS4, CLK0-CLK4, TA0 <sub>OUT</sub> -TA4 <sub>OUT</sub> , NMI, K10-K13, RxD0-RxD4, SCL0-SCL4, SDA0-SDA4		0.2		1.0	V	
		RESET		0.2		1.8	V	
I <sub>IH</sub>	Input High ("H") Current	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P87, P90-P97, P100-P107, X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE	V <sub>I</sub> =5V			5.0	μA	
I <sub>IL</sub>	Input Low ("L") Current	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P87, P90-P97, P100-P107, X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE	V <sub>I</sub> =0V			-5.0	μA	
R <sub>PULLUP</sub>	Pull-up Resistance	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107	V <sub>I</sub> =0V	20	40	167	kΩ	
R <sub>fXIN</sub>	Feedback Resistance	X <sub>IN</sub>			1.5		MΩ	
R <sub>fXCIN</sub>	Feedback Resistance	X <sub>CIN</sub>			15		MΩ	
V <sub>RAM</sub>	RAM Standby Voltage	In stop mode		2.0			V	
I <sub>CC</sub>	Power Supply Current	In single-chip mode, output pins are left open and other pins are connected to V <sub>SS</sub> .	f(BCLK)=32 MHz, Square wave, No division		22	60	mA	
			f(BCLK)=32 kHz, In wait mode, T <sub>opr</sub> =25° C		10		μA	
			While clock stops, T <sub>opr</sub> =25° C		0.8	5	μA	
			While clock stops, T <sub>opr</sub> =85° C			20	μA	

$$V_{CC1}=V_{CC2}=5V$$

**Table 23.4 A/D Conversion Characteristics ( $V_{CC1}=V_{CC2}=AV_{CC}=V_{REF}=4.2$  to  $5.5V$ ,  $V_{SS}=AV_{SS}=0V$  at  $T_{opr}=-20$  to  $85^{\circ}C$ ,  $f(BCLK) = 32MHz$  unless otherwise specified)**

Symbol	Parameter	Measurement Condition	Standard			Unit	
			Min.	Typ.	Max.		
-	Resolution	$V_{REF}=V_{CC1}$			10	Bits	
INL	Integral Nonlinearity Error	$V_{REF}=V_{CC1}=V_{CC2}=5V$	AN <sub>0</sub> to AN <sub>7</sub> , ANEX <sub>0</sub> , ANEX <sub>1</sub>			±3	LSB
							LSB
		External op-amp connection mode			±7	LSB	
DNL	Differential Nonlinearity Error				±1	LSB	
-	Offset Error				±3	LSB	
-	Gain Error				±3	LSB	
RLADDER	Resistor Ladder	$V_{REF}=V_{CC1}$	8		40	kΩ	
t <sub>CONV</sub>	10-bit Conversion Time <sup>(1, 2)</sup>		2.06			μs	
t <sub>CONV</sub>	8-bit Conversion Time <sup>(1, 2)</sup>		1.75			μs	
t <sub>SAMP</sub>	Sampling Time <sup>(1)</sup>		0.188			μs	
V <sub>REF</sub>	Reference Voltage		2		V <sub>CC1</sub>	V	
V <sub>IA</sub>	Analog Input Voltage		0		V <sub>REF</sub>	V	

## NOTES:

1. Divide  $f(X_{IN})$ , if exceeding 16 MHz, to keep  $\phi_{AD}$  frequency at 16 MHz or less.
2. With using the sample and hold function.

**Table 23.5 D/A Conversion Characteristics ( $V_{CC1}=V_{CC2}=V_{REF}=4.2$  to  $5.5V$ ,  $V_{SS}=AV_{SS}=0V$  at  $T_{opr}=-20$  to  $85^{\circ}C$ ,  $f(BCLK) = 32MHz$  unless otherwise specified)**

Symbol	Parameter	Measurement 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 1)			1.5	mA

## NOTE:

1. Measurement when using one D/A converter. The DA<sub>i</sub> register (i=0, 1) of the D/A converter, not being used, is set to "00<sub>16</sub>". The resistor ladder in the A/D converter is excluded.  
I<sub>VREF</sub> flows even if the VCUT bit in the AD0CON1 register is set to "0" (no V<sub>REF</sub> connection).



$$V_{CC1}=V_{CC2}=5V$$

### Timing Requirements

( $V_{CC1}=V_{CC2}=4.2$  to  $5.5V$ ,  $V_{SS}=0V$  at  $T_{opr}=-20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.6 External Clock Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_c$	External Clock Input Cycle Time	31.25		ns
$t_{w(H)}$	External Clock Input High ("H") Width	13.75		ns
$t_{w(L)}$	External Clock Input Low ("L") Width	13.75		ns
$t_r$	External Clock Rise Time		5	ns
$t_f$	External Clock Fall Time		5	ns

**Table 23.7 Memory Expansion Mode and Microprocessor Mode**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data Input Access Time (RD standard)		(Note 1)	ns
$t_{ac1(AD-DB)}$	Data Input Access Time (AD standard, CS standard)		(Note 1)	ns
$t_{ac2(RD-DB)}$	Data Input Access Time (RD standard, when accessing a space with the multiplexrd bus)		(Note 1)	ns
$t_{ac2(AD-DB)}$	Data Input Access Time (AD standard, when accessing a space with the multiplexed bus)		(Note 1)	ns
$t_{su(DB-BCLK)}$	Data Input Setup Time	26		ns
$t_{su(RDY-BCLK)}$	$\overline{RDY}$ Input Setup Time	26		ns
$t_{su(HOLD-BCLK)}$	$\overline{HOLD}$ Input Setup Time	30		ns
$t_{h(RD-DB)}$	Data Input Hold Time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{RDY}$ Input Hold Time	0		ns
$t_{h(BCLK-HOLD)}$	$\overline{HOLD}$ Input Hold Time	0		ns
$t_{d(BCLK-HLDA)}$	$\overline{HLDA}$ Output Delay Time		25	ns

**NOTE:**

1. Values can be obtained from the following equations, according to BCLK frequency and external bus cycles. Insert a wait state or lower the operation frequency,  $f_{(BCLK)}$ , if the calculated value is negative.

$$t_{ac1(RD-DB)} = \frac{10^9 \times m}{f_{(BCLK)} \times 2} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, m=(bx2)+1)$$

$$t_{ac1(AD-DB)} = \frac{10^9 \times n}{f_{(BCLK)}} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, n=a+b)$$

$$t_{ac2(RD-DB)} = \frac{10^9 \times m}{f_{(BCLK)} \times 2} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, m=(bx2)-1)$$

$$t_{ac2(AD-DB)} = \frac{10^9 \times p}{f_{(BCLK)} \times 2} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, p=\{(a+b-1)x2\}+1)$$

$V_{CC1}=V_{CC2}=5V$

**Timing Requirements****( $V_{CC1}=V_{CC2}=4.2$  to  $5.5V$ ,  $V_{SS}=0V$  at  $T_{op}=-20$  to  $85^{\circ}C$  unless otherwise specified)****Table 23.8 Timer A Input (Count Source 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 ("H") Width	40		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	40		ns

**Table 23.9 Timer A Input (Gate 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 ("H") Width	200		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	200		ns

**Table 23.10 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 ("H") Width	100		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	100		ns

**Table 23.11 Timer A Input (External Trigger Input in Pulse Width Modulation Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN Input High ("H") Width	100		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	100		ns

**Table 23.12 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 ("H") Width	1000		ns
$t_{w(UPL)}$	TAiOUT Input Low ("L") Width	1000		ns
$t_{su(UP-TIN)}$	TAiOUT Input Setup Time	400		ns
$t_{h(TIN-UP)}$	TAiOUT Input Hold Time	400		ns

$$V_{CC1}=V_{CC2}=5V$$

### Timing Requirements

( $V_{CC1} = V_{CC2} = 4.2$  to  $5.5V$ ,  $V_{SS} = 0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.13 Timer B Input (Count Source 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 ("H") Width (counted on one edge)	40		ns
$t_{w(TBL)}$	TBiIN Input Low ("L") 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 ("H") Width (counted on both edges)	80		ns
$t_{w(TBL)}$	TBiIN Input Low ("L") Width (counted on both edges)	80		ns

**Table 23.14 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 ("H") Width	200		ns
$t_{w(TBL)}$	TBiIN Input Low ("L") Width	200		ns

**Table 23.15 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 ("H") Width	200		ns
$t_{w(TBL)}$	TBiIN Input Low ("L") Width	200		ns

**Table 23.16 A/D Trigger Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	ADTRG Input Cycle Time (required for trigger)	1000		ns
$t_{w(ADL)}$	ADTRG Input Low ("L") Width	125		ns

**Table 23.17 Serial I/O**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi Input Cycle Time	200		ns
$t_{w(CKH)}$	CLKi Input High ("H") Width	100		ns
$t_{w(CKL)}$	CLKi Input Low ("L") 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	30		ns
$t_{h(C-Q)}$	RxDi Input Hold Time	90		ns

**Table 23.18 External Interrupt INTi Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	INTi Input High ("H") Width	250		ns
$t_{w(INL)}$	INTi Input Low ("L") Width	250		ns

$$V_{CC1}=V_{CC2}=5V$$

### Switching Characteristics

( $V_{CC1} = V_{CC2} = 4.2$  to  $5.5V$ ,  $V_{SS} = 0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.19 Memory Expansion Mode and Microprocessor Mode**  
(when accessing external memory space)

Symbol	Parameter	Measurement Condition	Standard		Unit
			Min.	Max.	
td(BCLK-AD)	Address Output Delay Time	See Figure 23.1		18	ns
th(BCLK-AD)	Address Output Hold Time (BCLK standard)		-3		ns
th(RD-AD)	Address Output Hold Time (RD standard)		0		ns
th(WR-AD)	Address Output Hold Time (WR standard)		(Note 1)		ns
td(BCLK-CS)	Chip-Select Signal Output Delay Time			18	ns
th(BCLK-CS)	Chip-Select Signal Output Hold Time (BCLK standard)		-3		ns
th(RD-CS)	Chip-Select Signal Output Hold Time (RD standard)		0		ns
th(WR-CS)	Chip-Select Signal Output Hold Time (WR standard)		(Note 1)		ns
td(BCLK-RD)	RD Signal Output Delay Time			18	ns
th(BCLK-RD)	RD Signal Output Hold Time		-5		ns
td(BCLK-WR)	WR Signal Output Delay Time			18	ns
th(BCLK-WR)	WR Signal Output Hold Time		-5		ns
td(DB-WR)	Data Output Delay Time (WR standard)		(Note 2)		ns
th(WR-DB)	Data Output Hold Time (WR standard)		(Note 1)		ns
tW(WR)	WR Output Width		(Note 2)		ns

#### NOTES:

1. Values can be obtained from the following equations, according to BCLK frequency.

$$t_{h(WR-DB)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

$$t_{h(WR-AD)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

$$t_{h(WR-CS)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

2. Values can be obtained from the following equations, according to BCLK frequency and external bus cycles.

$$t_{W(WR)} = \frac{10^9 \times n}{f_{(BCLK)} \times 2} - 15 \quad [ns] \quad (\text{if external bus cycle is } a\phi + b\phi, n=(bx2)-1)$$

$$t_{d(DB-WR)} = \frac{10^9 \times m}{f_{(BCLK)}} - 20 \quad [ns] \quad (\text{if external bus cycle is } a\phi + b\phi, m= b)$$

$$V_{CC1}=V_{CC2}=5V$$

### Switching Characteristics

( $V_{CC} = 4.2$  to  $5.5V$ ,  $V_{SS} = 0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.20 Memory Expansion Mode and Microprocessor Mode**  
(when accessing an external memory space with the multiplexed bus)

Symbol	Parameter	Measurement Condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address Output Delay Time	See Figure 23.1		18	ns
$t_{h(BCLK-AD)}$	Address Output Hold Time (BCLK standard)		-3		ns
$t_{h(RD-AD)}$	Address Output Hold Time (RD standard)		(Note 1)		ns
$t_{h(WR-AD)}$	Address Output Hold Time (WR standard)		(Note 1)		ns
$t_{d(BCLK-CS)}$	Chip-Select Signal Output Delay Time			18	ns
$t_{h(BCLK-CS)}$	Chip-Select Signal Output Hold Time (BCLK standard)		-3		ns
$t_{h(RD-CS)}$	Chip-Select Signal Output Hold Time (RD standard)		(Note 1)		ns
$t_{h(WR-CS)}$	Chip-Select Signal Output Hold Time (WR standard)		(Note 1)		ns
$t_{d(BCLK-RD)}$	RD Signal Output Delay Time			18	ns
$t_{h(BCLK-RD)}$	RD Signal Output Hold Time		-5		ns
$t_{d(BCLK-WR)}$	WR Signal Output Delay Time			18	ns
$t_{h(BCLK-WR)}$	WR Signal Output Hold Time		-5		ns
$t_{d(DB-WR)}$	Data Output Delay Time (WR standard)		(Note 2)		ns
$t_{h(WR-DB)}$	Data Output Hold Time (WR standard)		(Note 1)		ns
$t_{d(BCLK-ALE)}$	ALE Signal Output Delay Time (BCLK standard)			18	ns
$t_{h(BCLK-ALE)}$	ALE Signal Output Hold Time (BCLK standard)		-5		ns
$t_{d(AD-ALE)}$	ALE Signal Output Delay Time (address standard)		(Note 3)		ns
$t_{h(ALE-AD)}$	ALE Signal Output Hold Time (address standard)		(Note 4)		ns
$t_{dz(RD-AD)}$	Address Output Float Start Time			8	ns

#### NOTES:

1. Values can be obtained from the following equations, according to BCLK frequency.

$$t_{h(RD-AD)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

$$t_{h(WR-AD)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

$$t_{h(RD-CS)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

$$t_{h(WR-CS)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

$$t_{h(WR-DB)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

2. Values can be obtained from the following equations, according to BCLK frequency and external bus cycle.

$$t_{d(DB-WR)} = \frac{10^9 \times m}{f_{(BCLK)} \times 2} - 25 \quad [\text{ns}] \quad (\text{if external bus cycle is } a\phi + b\phi, m = (bx2)-1)$$

3. Values can be obtained from the following equations, according to BCLK frequency and external bus cycle.

$$t_{d(AD-ALE)} = \frac{10^9 \times n}{f_{(BCLK)} \times 2} - 20 \quad [\text{ns}] \quad (\text{if external bus cycle is } a\phi + b\phi, n = a)$$

4. Values can be obtained from the following equations, according to BCLK frequency and external bus cycle.

$$t_{h(ALE-AD)} = \frac{10^9 \times n}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}] \quad (\text{if external bus cycle is } a\phi + b\phi, n = a)$$

$$V_{CC1}=V_{CC2}=5V$$

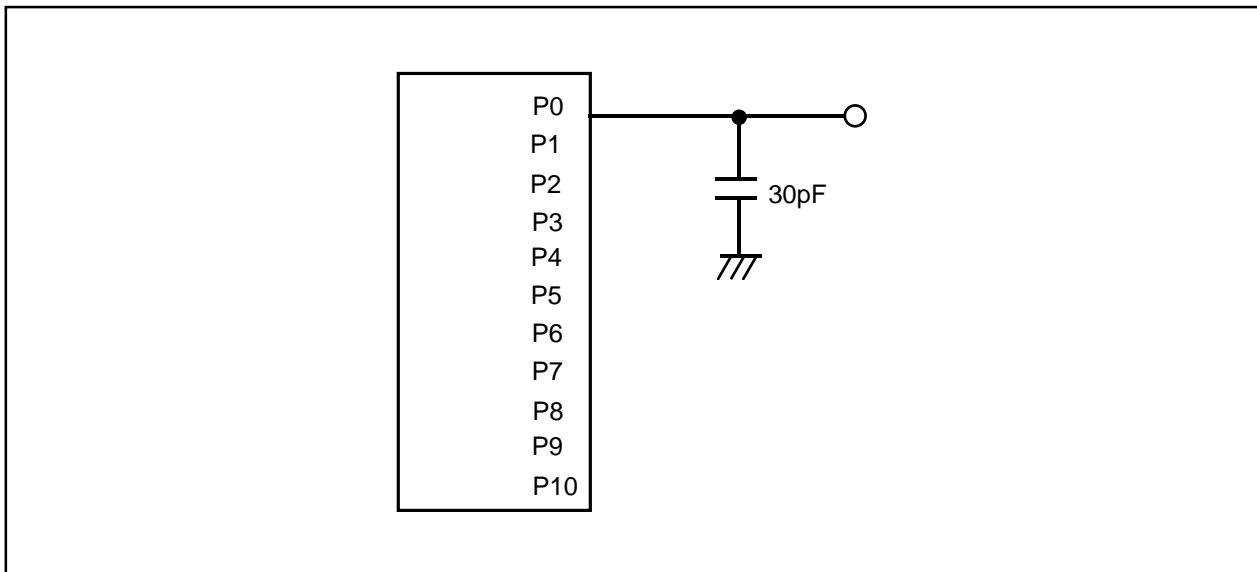


Figure 23.1 P0 to P10 Measurement Circuit

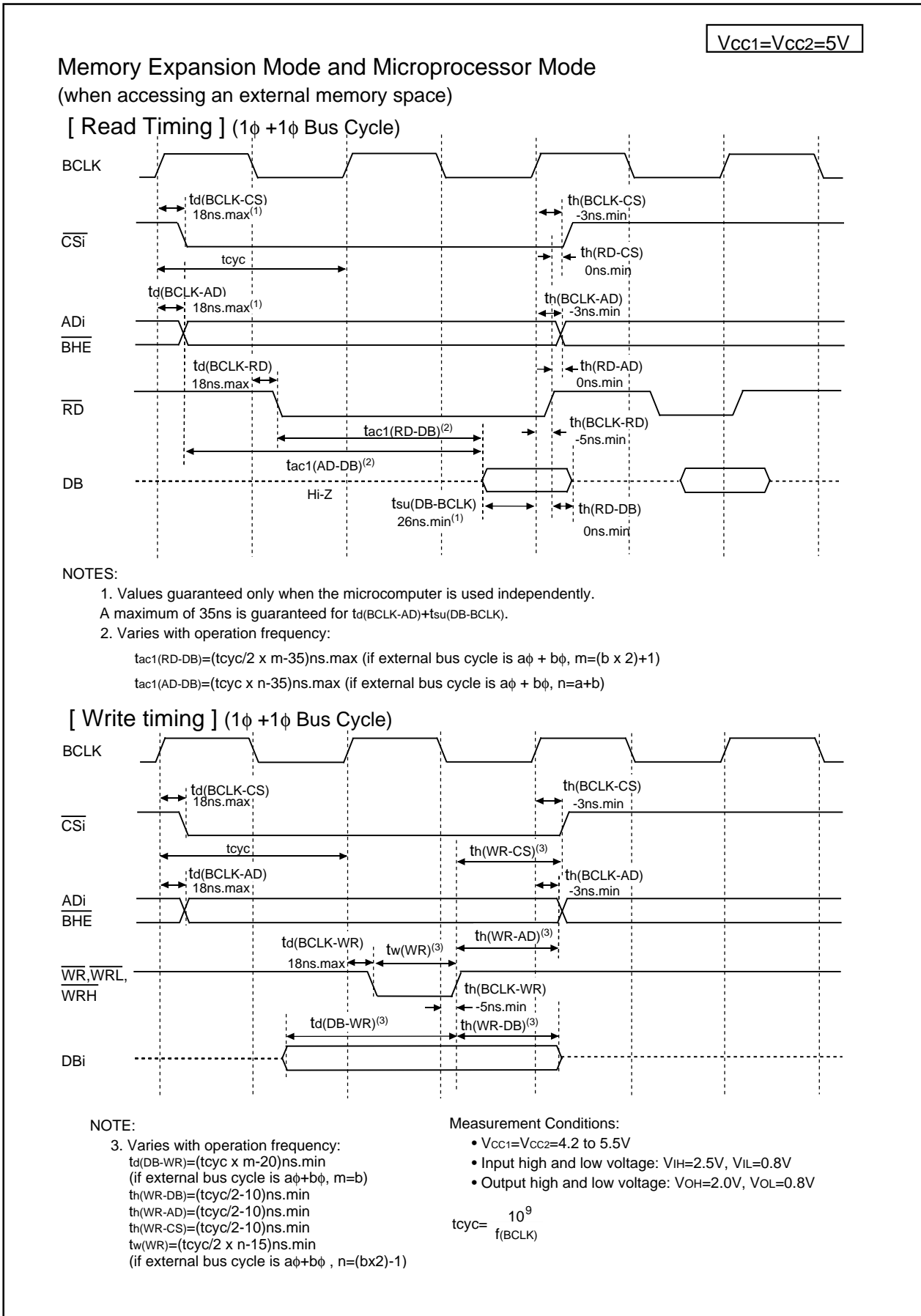
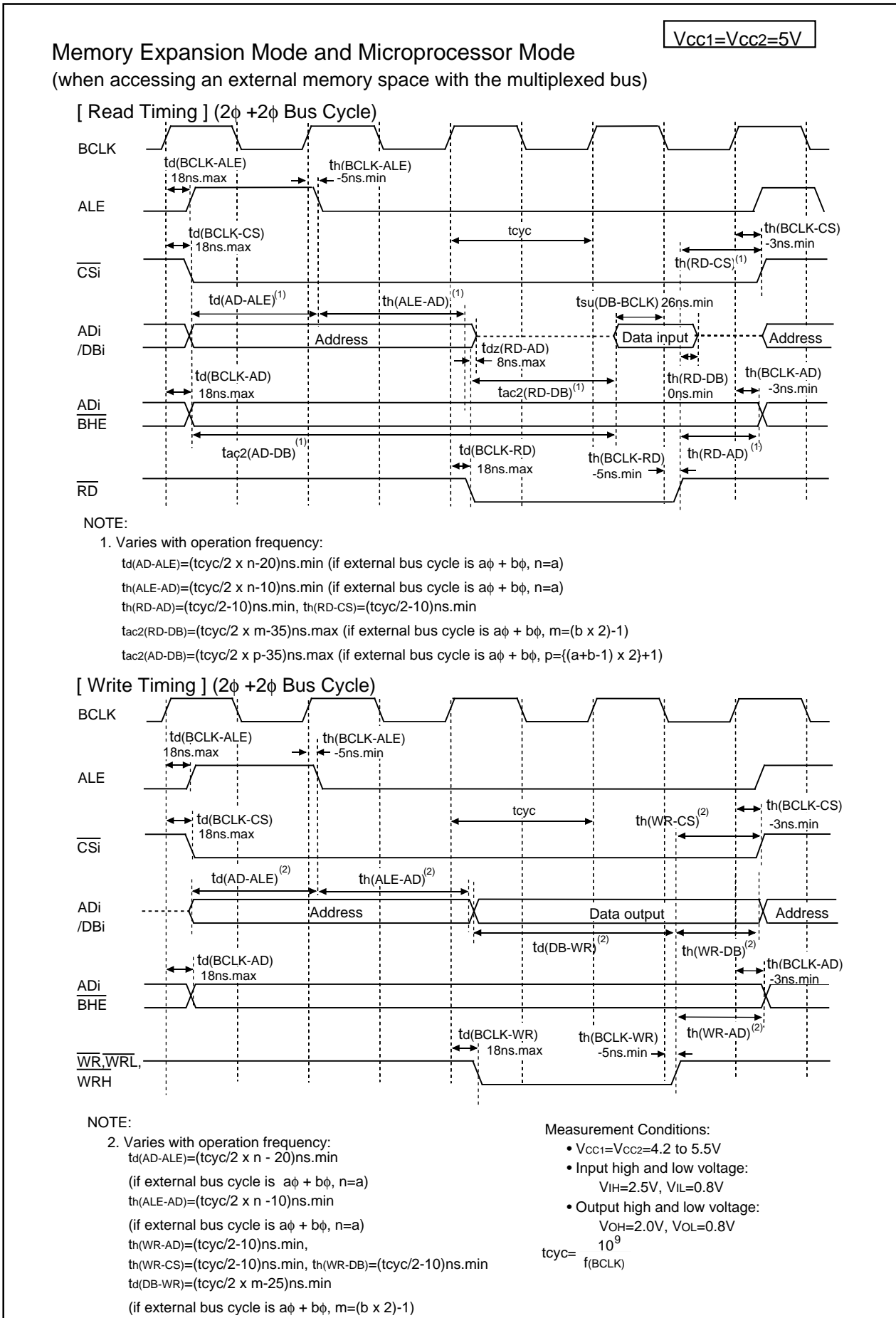


Figure 23.2 V<sub>CC1</sub>=V<sub>CC2</sub>=5V Timing Diagram (1)



**Figure 23.3 Vcc1=Vcc2=5V Timing Diagram (2)**



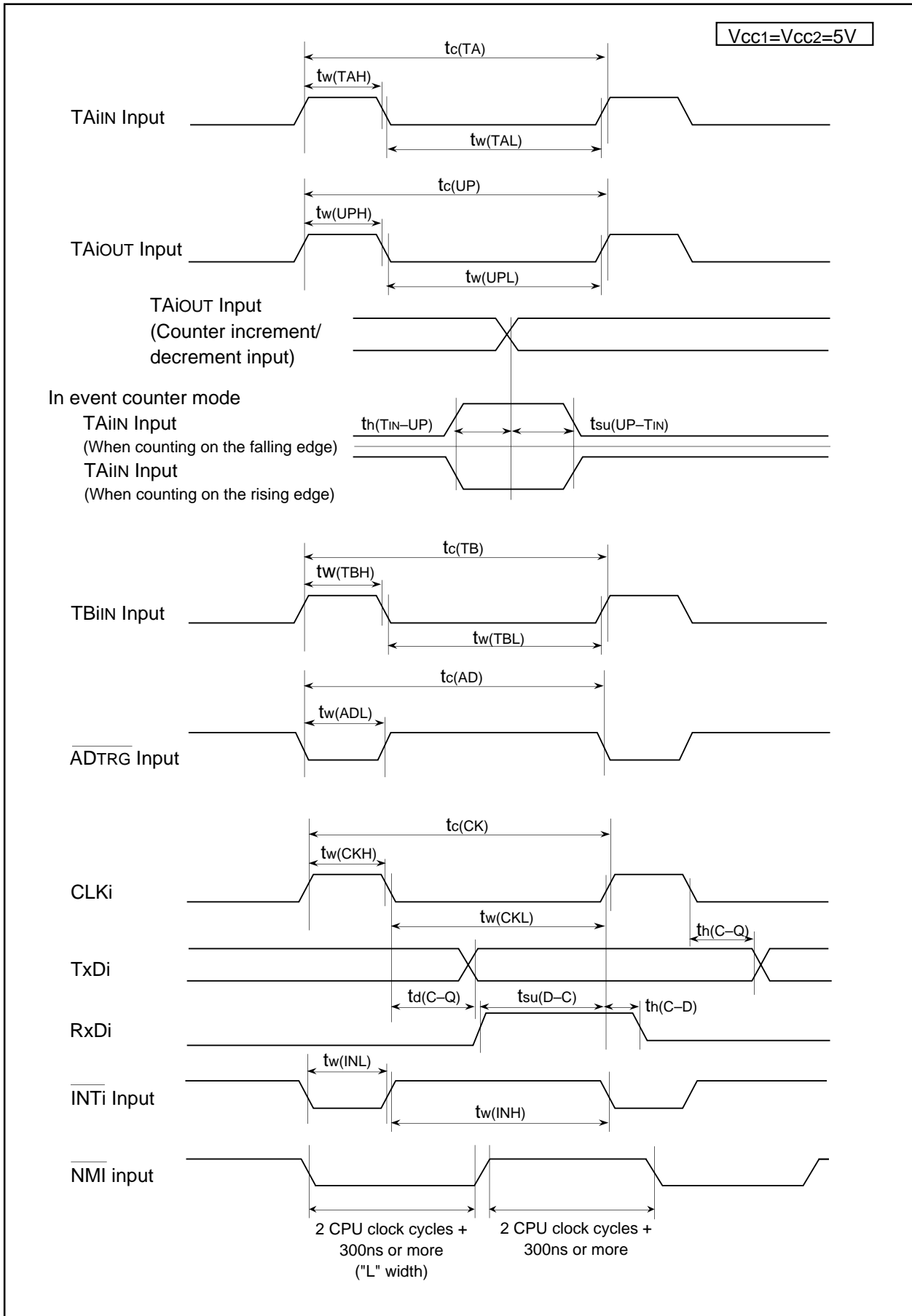


Figure 23.4 VCC1=VCC2=5V Timing Diagram (3)

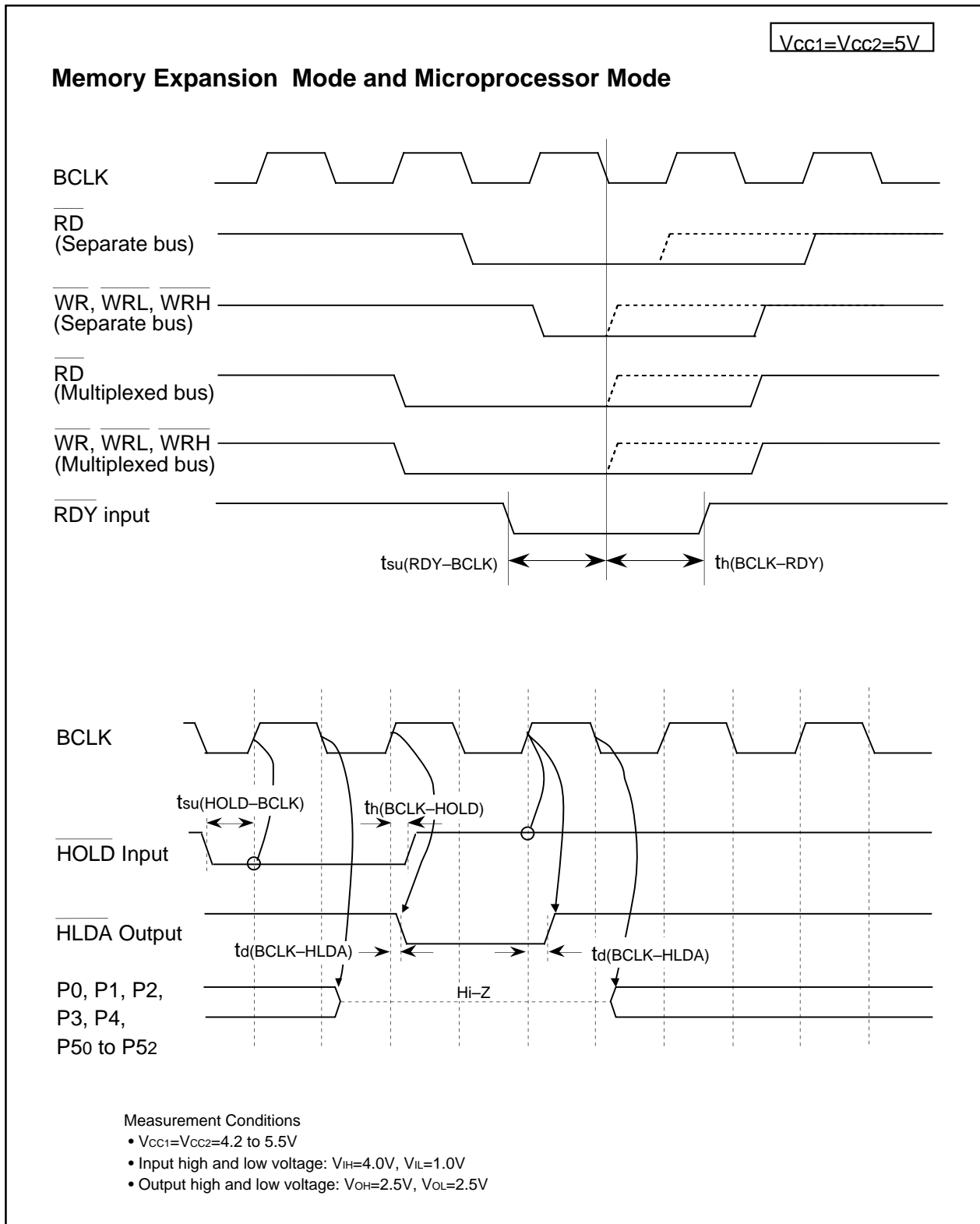


Figure 23.5  $V_{CC1}=V_{CC2}=5V$  Timing Diagram (4)

$$V_{CC1}=V_{CC2}=3.3V$$

**Table 23.21 Electrical Characteristics ( $V_{CC1}=V_{CC2}=3.0$  to  $3.6V$ ,  $V_{SS}=0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$ ,  $f(BCLK)=24MHz$  unless otherwise specified)**

Symbol	Parameter		Condition	Standard			Unit
				Min.	Typ.	Max.	
V <sub>OH</sub>	Output High ("H") Voltage	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57	I <sub>OH</sub> =-1mA	V <sub>CC2</sub> -0.6		V <sub>CC2</sub>	V
		P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107		V <sub>CC1</sub> -0.6		V <sub>CC1</sub>	V
	X <sub>OUT</sub>	I <sub>OH</sub> =-0.1mA	2.7		V <sub>CC1</sub>	V	
	X <sub>COUT</sub>	High Power	No load applied		2.5		V
		Low Power	No load applied		1.6		V
V <sub>OL</sub>	Output Low ("L") Voltage	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P84, P86, P87, P90-P97, P100-P107	I <sub>OL</sub> =1mA			0.5	V
		X <sub>OUT</sub>	I <sub>OL</sub> =0.1mA			0.5	V
	X <sub>COUT</sub>	High Power	No load applied		0		V
		Low Power	No load applied		0		V
	V <sub>T+</sub> -V <sub>T-</sub>	Hysteresis	HOLD, RDY, TA0 <sub>IN</sub> -TA4 <sub>IN</sub> , TB0 <sub>IN</sub> -TB5 <sub>IN</sub> , INT0-INT5, AD <sub>TRG</sub> , CTS0-CTS4, CLK0-CLK4, TA0 <sub>OUT</sub> -TA4 <sub>OUT</sub> , NMI, KI0-KI3, RxD0-RxD4, SCL0-SCL4, SDA0-SDA4		0.2		1.0
RESET				0.2		1.8	V
I <sub>IH</sub>	Input High ("H") Current	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P87, P90-P97, P100-P107, X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE	V <sub>I</sub> =3V			4.0	μA
I <sub>IL</sub>	Input Low ("L") Current	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P70-P77, P80-P87, P90-P97, P100-P107, X <sub>IN</sub> , RESET, CNV <sub>SS</sub> , BYTE	V <sub>I</sub> =0V			-4.0	μA
R <sub>PULLUP</sub>	Pull-up Resistance	P00-P07, P10-P17, P20-P27, P30-P37, P40-P47, P50-P57, P60-P67, P72-P77, P80-P84, P86, P87, P90-P97, P100-P107	V <sub>I</sub> =0V	40	70	500	kΩ
R <sub>fXIN</sub>	Feedback Resistance	X <sub>IN</sub>			3.0		MΩ
R <sub>fXCIN</sub>	Feedback Resistance	X <sub>CIN</sub>			30.0		MΩ
V <sub>RAM</sub>	RAM Standby Voltage	in stop mode		2.0			V
I <sub>CC</sub>	Power Supply Current	Measurement condition: In single-chip mode, output pins are left open and other pins are connected to V <sub>SS</sub> .	f(BCLK)=24 MHz, Square wave, No division		17	35	mA
			f(BCLK)=32 kHz, In wait mode, T <sub>opr</sub> =25° C		10		μA
			While clock stops, T <sub>opr</sub> =25° C		0.8	5	μA
			While clock stops, T <sub>opr</sub> =85° C			50	μA

$V_{CC1}=V_{CC2}=3.3V$

**Table 23.22 A/D Conversion Characteristics ( $V_{CC1}=V_{CC2}=AV_{CC}=V_{REF}=3.0$  to  $3.6V$ ,  $V_{SS}=AV_{SS}=0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$ ,  $f(BCLK) = 24MHz$  unless otherwise specified)**

Symbol	Parameter		Measurement Condition	Standard			Unit
				Min.	Typ.	Max.	
-	Resolution		$V_{REF}=V_{CC1}$			10	Bits
INL	Integral Nonlinearity Error	No S&H (8-bit)	$V_{CC1}=V_{CC2}=V_{REF}=3.3V$			$\pm 2$	LSB
DNL	Differential Nonlinearity Error	No S&H (8-bit)				$\pm 1$	LSB
-	Offset Error	No S&H (8-bit)				$\pm 2$	LSB
-	Gain Error	No S&H (8-bit)				$\pm 2$	LSB
RLADDER	Resistor Ladder		$V_{REF}=V_{CC1}$	8.0		40	k $\Omega$
tCONV	8-bit Conversion Time <sup>(1, 2)</sup>			6.1			$\mu s$
VREF	Reference Voltage			3.3		$V_{CC1}$	V
VIA	Analog Input Voltage			0		$V_{REF}$	V

S&amp;H: Sample and Hold

## NOTES:

1. Divide  $f(X_{IN})$ , if exceeding 10 MHz, to keep  $\phi AD$  frequency at 10 MHz or less.
2. S&H not available.

**Table 23.23 D/A Conversion Characteristics ( $V_{CC1}=V_{CC2}=V_{REF}=3.0$  to  $3.6V$ ,  $V_{SS}=AV_{SS}=0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$ ,  $f(BCLK) = 24MHz$  unless otherwise specified)**

Symbol	Parameter		Measurement Condition	Standard			Unit
				Min.	Typ.	Max.	
-	Resolution					8	Bits
-	Absolute Accuracy					1.0	%
tsu	Setup Time					3	$\mu s$
Ro	Output Resistance			4	10	20	k $\Omega$
I <sub>VREF</sub>	Reference Power Supply Input Current		(Note 1)			1.0	mA

## NOTE:

1. Measurement results when using one D/A converter. The DA<sub>i</sub> register (i=0, 1) of the D/A converter, not being used, is set to "00<sub>16</sub>". The resistor ladder in the A/D converter is excluded.  
I<sub>VREF</sub> flows even if the VCUT bit in the AD0CON1 register is set to "0" (no V<sub>REF</sub> connection).

$$V_{CC1}=V_{CC2}=3.3V$$

### Timing Requirements

( $V_{CC1}=V_{CC2}= 3.0$  to  $3.6V$ ,  $V_{SS} = 0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.24 External Clock Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_c$	External Clock Input Cycle Time	41		ns
$t_{w(H)}$	External Clock Input High ("H") Width	18		ns
$t_{w(L)}$	External Clock Input Low ("L") Width	18		ns
$t_r$	External Clock Rise Time		5	ns
$t_f$	External Clock Fall Time		5	ns

**Table 23.25 Memory Expansion Mode and Microprocessor Mode**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data Input Access Time (RD standard)		(Note 1)	ns
$t_{ac1(AD-DB)}$	Data Input Access Time (AD standard, CS standard)		(Note 1)	ns
$t_{ac2(RD-DB)}$	Data Input Access Time (RD standard, when accessing a space with the multiplexed bus)		(Note 1)	ns
$t_{ac2(AD-DB)}$	Data Input Access Time (AD standard, when accessing a space with the multiplexed bus)		(Note 1)	ns
$t_{su(DB-BCLK)}$	Data Input Setup Time	30		ns
$t_{su(RDY-BCLK)}$	$\overline{RDY}$ Input Setup Time	40		ns
$t_{su(HOLD-BCLK)}$	HOLD Input Setup Time	60		ns
$t_{h(RD-DB)}$	Data Input Hold Time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{RDY}$ Input Hold Time	0		ns
$t_{h(BCLK-HOLD)}$	HOLD Input Hold Time	0		ns
$t_{d(BCLK-HLDA)}$	HLDA Output Delay Time		25	ns

**NOTE:**

1. Values can be obtained from the following equations, according to BCLK frequency and external bus cycles. Insert a wait state or lower the operation frequency,  $f_{(BCLK)}$ , if the calculated value is negative.

$$t_{ac1(RD-DB)} = \frac{10^9 \times m}{f_{(BCLK)} \times 2} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, m=(bx2)+1)$$

$$t_{ac1(AD-DB)} = \frac{10^9 \times n}{f_{(BCLK)}} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, n=a+b)$$

$$t_{ac2(RD-DB)} = \frac{10^9 \times m}{f_{(BCLK)} \times 2} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, m=(bx2)-1)$$

$$t_{ac2(AD-DB)} = \frac{10^9 \times p}{f_{(BCLK)} \times 2} - 35 \quad [\text{ns}] \text{ (if external bus cycle is } a\phi + b\phi, p=((a+b-1)x2)+1)$$

$$V_{CC1}=V_{CC2}=3.3V$$

### Timing Requirements

( $V_{CC1}=V_{CC2}= 3.0$  to  $3.6V$ ,  $V_{SS}= 0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.26 Timer A Input (Count Source 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 ("H") Width	40		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	40		ns

**Table 23.27 Timer A Input (Gate 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 ("H") Width	200		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	200		ns

**Table 23.28 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 ("H") Width	100		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	100		ns

**Table 23.29 Timer A Input (External Trigger Input in Pulse Width Modulation Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN Input High ("H") Width	100		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	100		ns

**Table 23.30 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 ("H") Width	1000		ns
$t_{w(UPL)}$	TAiOUT Input Low ("L") Width	1000		ns
$t_{su(UP-TIN)}$	TAiOUT Input Setup Time	400		ns
$t_{h(TIN-UP)}$	TAiOUT Input Hold Time	400		ns

$V_{CC1}=V_{CC2}=3.3V$

**Timing Requirements**(V<sub>CC1</sub>=V<sub>CC2</sub>= 3.0 to 3.6V, V<sub>SS</sub> = 0V at Topr = –20 to 85°C unless otherwise specified)**Table 23.31 Timer B Input (Count Source Input in Event Counter Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c(TB)</sub>	TB <sub>iIN</sub> Input Cycle Time (counted on one edge)	100		ns
t <sub>w(TBH)</sub>	TB <sub>iIN</sub> Input High ("H") Width (counted on one edge)	40		ns
t <sub>w(TBL)</sub>	TB <sub>iIN</sub> Input Low ("L") Width (counted on one edge)	40		ns
t <sub>c(TB)</sub>	TB <sub>iIN</sub> Input Cycle Time (counted on both edges)	200		ns
t <sub>w(TBH)</sub>	TB <sub>iIN</sub> Input High ("H") Width (counted on both edges)	80		ns
t <sub>w(TBL)</sub>	TB <sub>iIN</sub> Input Low ("L") Width (counted on both edges)	80		ns

**Table 23.32 Timer B Input (Pulse Period Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c(TB)</sub>	TB <sub>iIN</sub> Input Cycle Time	400		ns
t <sub>w(TBH)</sub>	TB <sub>iIN</sub> Input High ("H") Width	200		ns
t <sub>w(TBL)</sub>	TB <sub>iIN</sub> Input Low ("L") Width	200		ns

**Table 23.33 Timer B Input (Pulse Width Measurement Mode)**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c(TB)</sub>	TB <sub>iIN</sub> Input Cycle Time	400		ns
t <sub>w(TBH)</sub>	TB <sub>iIN</sub> Input High ("H") Width	200		ns
t <sub>w(TBL)</sub>	TB <sub>iIN</sub> Input Low ("L") Width	200		ns

**Table 23.34 A/D Trigger Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c(AD)</sub>	$\overline{AD}_{TRG}$ Input Cycle Time (required for trigger)	1000		ns
t <sub>w(ADL)</sub>	$\overline{AD}_{TRG}$ Input Low ("L") Width	125		ns

**Table 23.35 Serial I/O**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>c(CK)</sub>	CLK <sub>i</sub> Input Cycle Time	200		ns
t <sub>w(CKH)</sub>	CLK <sub>i</sub> Input High ("H") Width	100		ns
t <sub>w(CKL)</sub>	CLK <sub>i</sub> Input Low ("L") Width	100		ns
t <sub>d(C-Q)</sub>	TxD <sub>i</sub> Output Delay Time		80	ns
t <sub>h(C-Q)</sub>	TxD <sub>i</sub> Hold Time	0		ns
t <sub>su(D-C)</sub>	RxD <sub>i</sub> Input Setup Time	30		ns
t <sub>h(C-Q)</sub>	RxD <sub>i</sub> Input Hold Time	90		ns

**Table 23.36 External Interrupt  $\overline{INT}_i$  Input**

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t <sub>w(INH)</sub>	$\overline{INT}_i$ Input High ("H") Width	250		ns
t <sub>w(INL)</sub>	$\overline{INT}_i$ Input Low ("L") Width	250		ns

$$V_{CC1}=V_{CC2}=3.3V$$

### Switching Characteristics

( $V_{CC1}=V_{CC2}=3.0$  to  $3.6V$ ,  $V_{SS} = 0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.37 Memory Expansion Mode and Microprocessor Mode  
(when accessing external memory space)**

Symbol	Parameter	Measurement Condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address Output Delay Time	See Figure 23.1		18	ns
$t_{h(BCLK-AD)}$	Address Output Hold Time (BCLK standard)		0		ns
$t_{h(RD-AD)}$	Address Output Hold Time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address Output Hold Time (WR standard)		(Note 1)		ns
$t_{d(BCLK-CS)}$	Chip-Select Signal Output Delay Time			18	ns
$t_{h(BCLK-CS)}$	Chip-Select Signal Output Hold Time (BCLK standard)		0		ns
$t_{h(RD-CS)}$	Chip-Select Signal Output Hold Time (RD standard)		0		ns
$t_{h(WR-CS)}$	Chip-Select Signal Output Hold Time (WR standard)		(Note 1)		ns
$t_{d(BCLK-RD)}$	RD Signal Output Delay Time			18	ns
$t_{h(BCLK-RD)}$	RD Signal Output Hold Time		-3		ns
$t_{d(BCLK-WR)}$	WR Signal Output Delay Time			18	ns
$t_{h(BCLK-WR)}$	WR Signal Output Hold Time		0		ns
$t_{d(DB-WR)}$	Data Output Delay Time (WR standard)		(Note 2)		ns
$t_{h(WR-DB)}$	Data Output Hold Time (WR standard)		(Note 1)		ns
$t_{w(WR)}$	WR Output Width		(Note 2)		ns

#### NOTES:

1. Values can be obtained from the following equations, according to BCLK frequency.

$$t_{h(WR-DB)} = \frac{10^9}{f_{(BCLK)} \times 2} - 20 \quad [ns]$$

$$t_{h(WR-AD)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

$$t_{h(WR-CS)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

2. Values can be obtained from the following equations, according to BCLK frequency and external bus cycles.

$$t_{w(WR)} = \frac{10^9 \times n}{f_{(BCLK)} \times 2} - 15 \quad [ns] \quad (\text{if external bus cycle is } a\phi + b\phi, n=(b \times 2)-1)$$

$$t_{d(DB-WR)} = \frac{10^9 \times m}{f_{(BCLK)}} - 20 \quad [ns] \quad (\text{if external bus cycle is } a\phi + b\phi, m=b)$$



$$V_{CC1}=V_{CC2}=3.3V$$

### Switching Characteristics

( $V_{CC1} = V_{CC2} = 3.0$  to  $3.6V$ ,  $V_{SS} = 0V$  at  $T_{opr} = -20$  to  $85^{\circ}C$  unless otherwise specified)

**Table 23.38 Memory Expansion Mode and Microprocessor Mode**  
(when accessing an external memory space with the multiplexed bus)

Symbol	Parameter	Measurement Condition	Standard		Unit
			Min.	Max.	
td(BCLK-AD)	Address Output Delay Time	See Figure 23.1		18	ns
th(BCLK-AD)	Address Output Hold Time (BCLK standard)		0		ns
th(RD-AD)	Address Output Hold Time (RD standard)		(Note 1)		ns
th(WR-AD)	Address Output Hold Time (WR standard)		(Note 1)		ns
td(BCLK-CS)	Chip-Select Signal Output Delay Time			18	ns
th(BCLK-CS)	Chip-Select Signal Output Hold Time (BCLK standard)		0		ns
th(RD-CS)	Chip-Select Signal Output Hold Time (RD standard)		(Note 1)		ns
th(WR-CS)	Chip-Select Signal Output Hold Time (WR standard)		(Note 1)		ns
td(BCLK-RD)	RD Signal Output Delay Time			18	ns
th(BCLK-RD)	RD Signal Output Hold Time		-3		ns
td(BCLK-WR)	WR Signal Output Delay Time			18	ns
th(BCLK-WR)	WR Signal Output Hold Time		0		ns
td(DB-WR)	Data Output delay Time (WR standard)		(Note 2)		ns
th(WR-DB)	Data Output Hold Time (WR standard)		(Note 1)		ns
td(BCLK-ALE)	ALE Signal Output Delay Time (BCLK standard)			18	ns
th(BCLK-ALE)	ALE Signal Output Hold Time (BCLK standard)		-2		ns
td(AD-ALE)	ALE Signal Output Delay Time (address standard)		(Note 3)		ns
th(ALE-AD)	ALE Signal Output Hold Time (address standard)		(Note 4)		ns
tdz(RD-AD)	Address Output Float Start Time			8	ns

#### NOTES:

1. Values can be obtained by the following equations, according to BCLK frequency.

$$th(RD - AD) = \frac{10^9}{f(BCLK) \times 2} - 10 \quad [ns]$$

$$th(WR - AD) = \frac{10^9}{f(BCLK) \times 2} - 10 \quad [ns]$$

$$th(RD - CS) = \frac{10^9}{f(BCLK) \times 2} - 10 \quad [ns]$$

$$th(WR - CS) = \frac{10^9}{f(BCLK) \times 2} - 10 \quad [ns]$$

$$th(WR - DB) = \frac{10^9}{f(BCLK) \times 2} - 20 \quad [ns]$$

2. Values can be obtained by the following equations, according to BCLK frequency and external bus cycles.

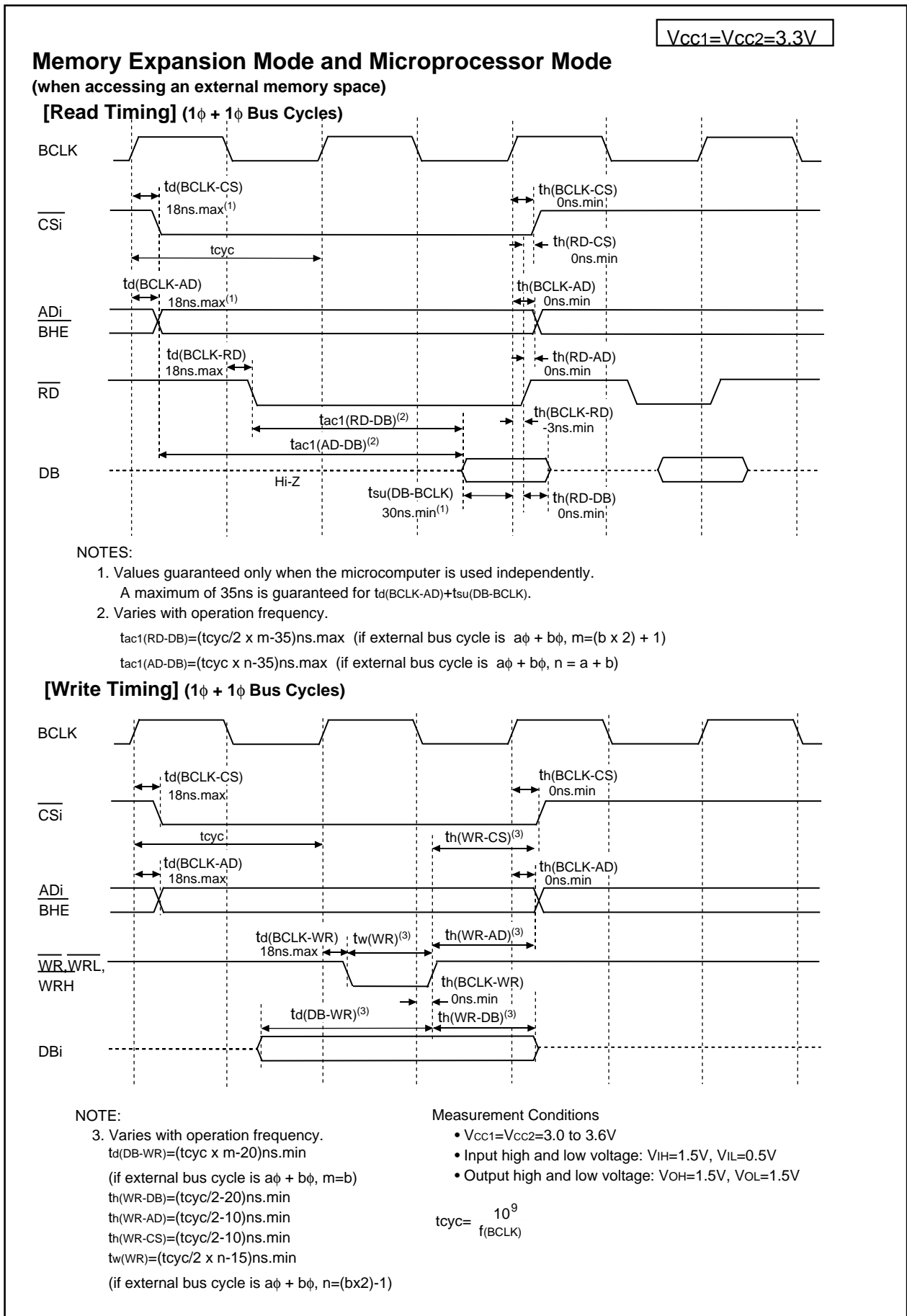
$$td(DB - WR) = \frac{10^9 \times m}{f(BCLK) \times 2} - 25 \quad [ns] \quad (\text{if external bus cycle is } a\phi + b\phi, m=(b+2)-1)$$

3. Values can be obtained by the following equations, according to BCLK frequency and external bus cycles.

$$td(AD - ALE) = \frac{10^9 \times n}{f(BCLK) \times 2} - 20 \quad [ns] \quad (\text{if external bus cycle is } a\phi + b\phi, n=a)$$

4. Values can be obtained by the following equations, according to BCLK frequency and external bus cycles.

$$th(ALE - AD) = \frac{10^9 \times n}{f(BCLK) \times 2} - 10 \quad [ns] \quad (\text{if external bus cycle is } a\phi + b\phi, n=a)$$



**Figure 23.6 V<sub>CC1</sub>=V<sub>CC2</sub>=3.3V Timing Diagram (1)**

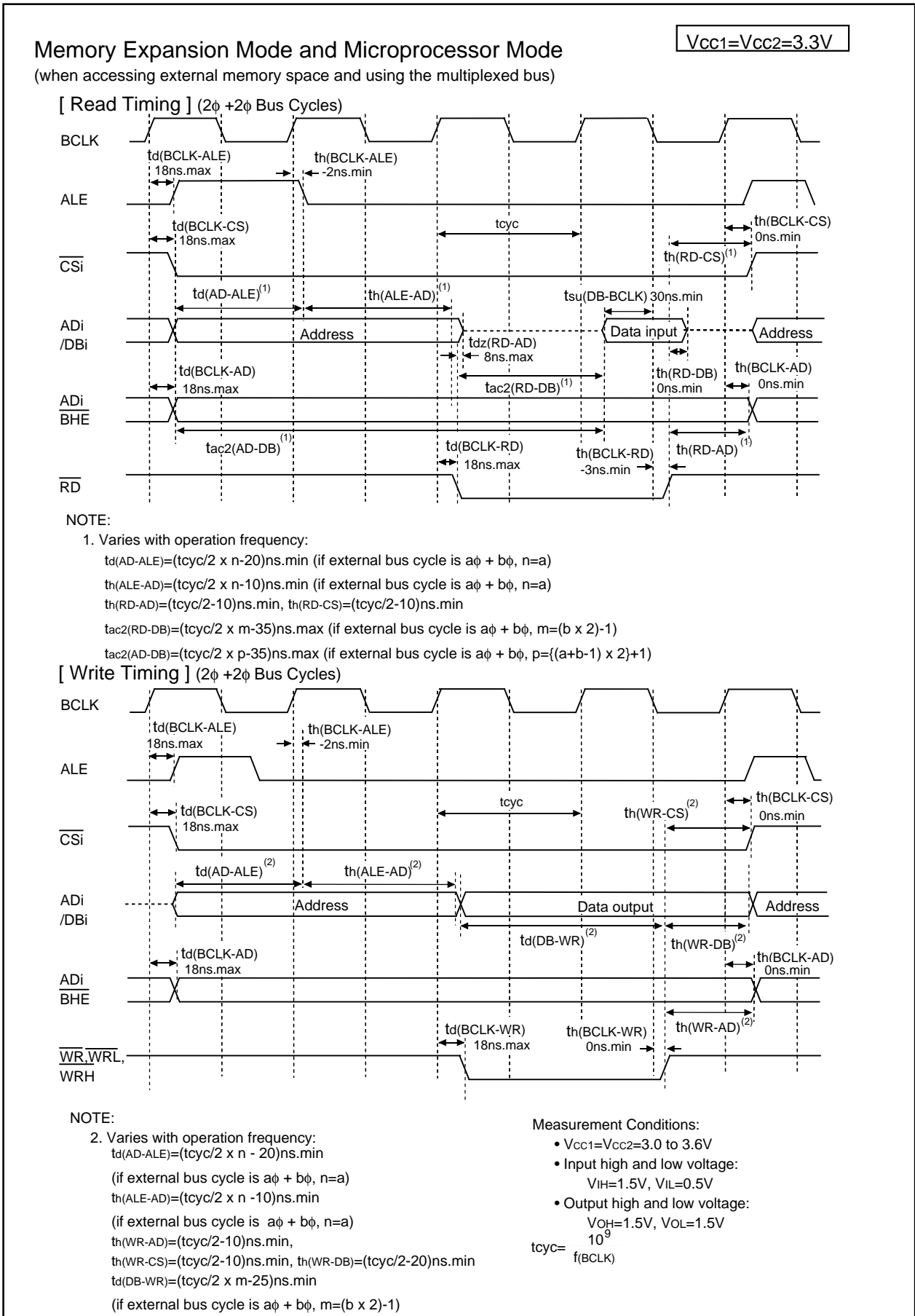


Figure 23.7 V<sub>CC1</sub>=V<sub>CC2</sub>=3.3V Timing Diagram (2)

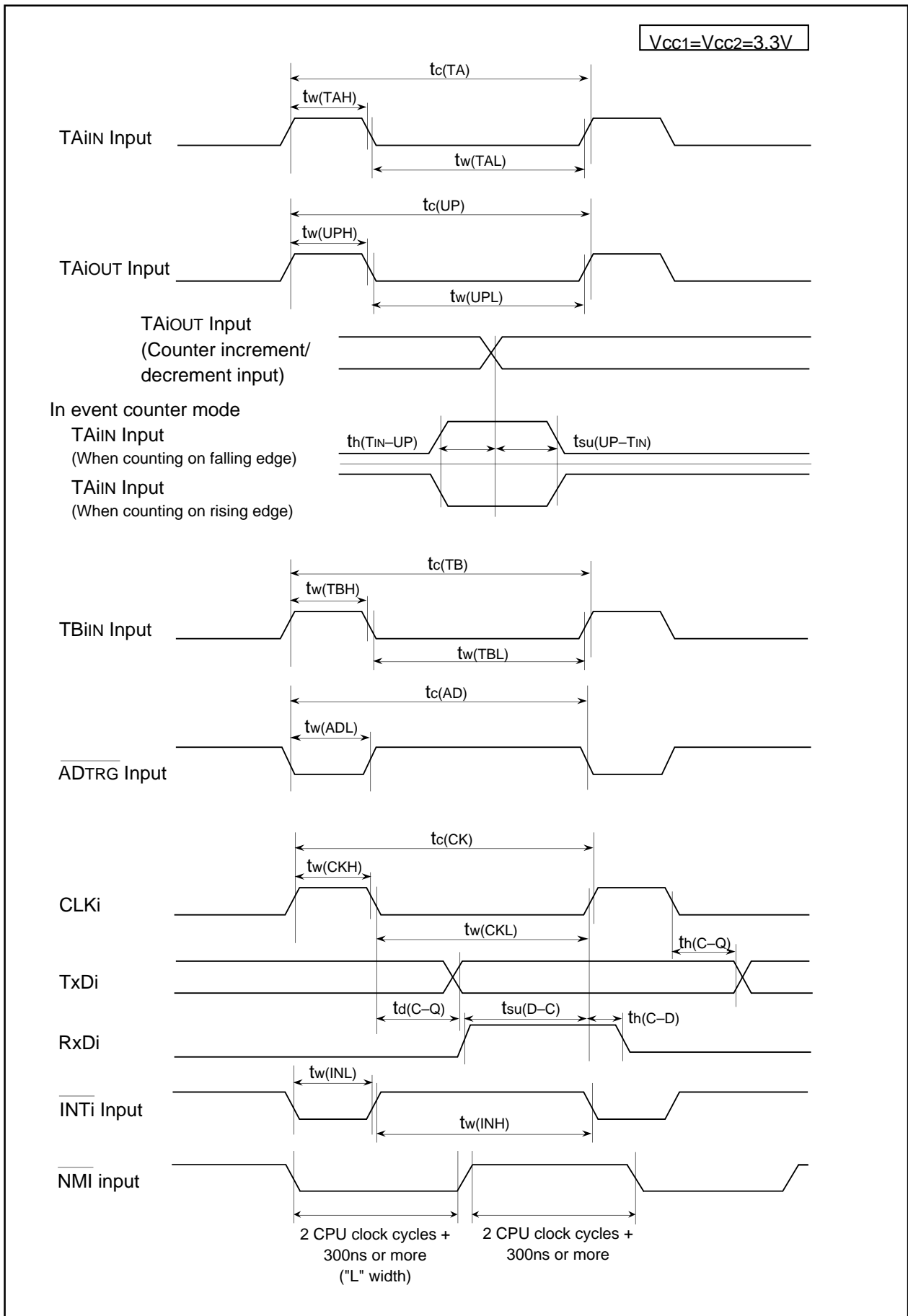


Figure 23.8 V<sub>CC1</sub>=V<sub>CC2</sub>=3.3V Timing Diagram (3)

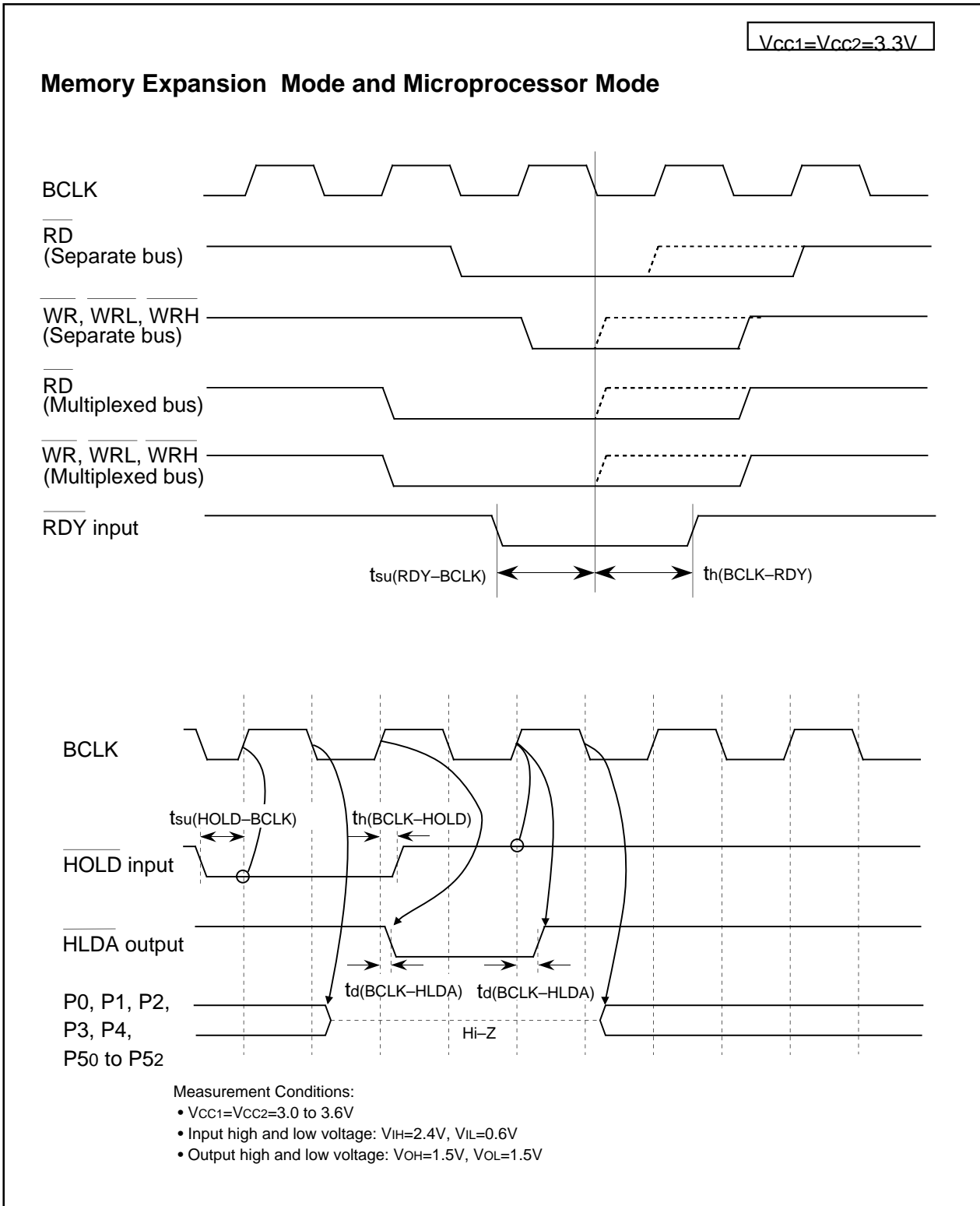


Figure 23.9  $V_{CC1}=V_{CC2}=3.3V$  Timing Diagram (4)

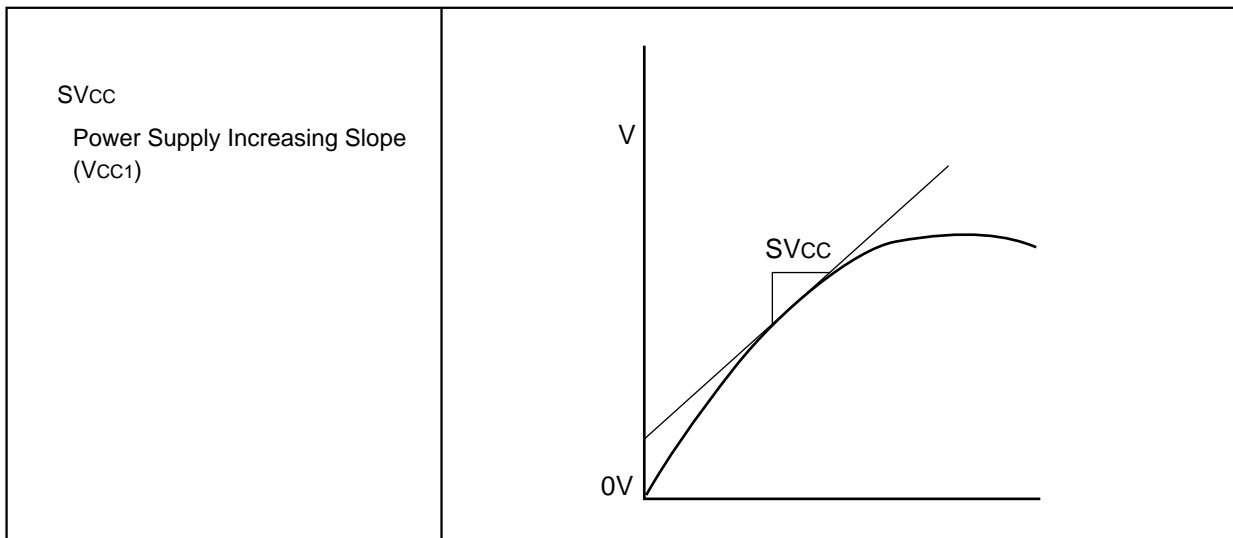
## 24. Precautions

### 24.1 Reset

Voltage applied to the VCC1 pin must meet the SVcc standard.

**Table 24.1 Power Supply Increasing Slope**

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
SVcc	Power Supply Increasing Slope (Vcc1)	0.05			V/ms



**Figure 24.1 SVcc Timing**

## 24.2 Bus

### 24.2.1 $\overline{\text{HOLD}}$ Signal

When entering microprocessor mode or memory expansion mode from single-chip mode and using  $\overline{\text{HOLD}}$  input, set the PM01 and PM00 bits to "112" (microprocessor mode) or to "012" (memory expansion mode) after setting the PD4\_7 to PD4\_0 bits in the PD4 register and the PD5\_2 to PD5\_0 bits in the PD5 register to "0" (input mode).

P40 to P47 (A16 to A22,  $\overline{\text{A23}}$ ,  $\overline{\text{CS0}}$  to  $\overline{\text{CS3}}$ , MA8 to MA12) and P50 to P52 ( $\overline{\text{RD/WR/BHE}}$ ,  $\overline{\text{RD/WRL/WRH}}$ ) are not placed in high-impedance states even when a low-level ("L") signal is applied to the  $\overline{\text{HOLD}}$  pin, if the PM01 and PM00 bits are set to "112" (microprocessor mode) or to "012" (memory expansion mode) after setting the PD4\_7 to PD4\_0 bits in the PD4 register and the PD5\_2 to PD5\_0 bits in the PD5 register to "1" (output mode) in single-chip mode.

## 24.3 Special Function Registers (SFRs)

### 24.3.1 Register Settings

Table 24.2 lists registers containing 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 24.2 Registers with Write-only Bits**

Register	Address	Register	Address
WDTS Register	000E <sub>16</sub>	U3BRG Register	0329 <sub>16</sub>
G0RI Register	00EC <sub>16</sub>	U3TB Register	032B <sub>16</sub> , 032A <sub>16</sub>
G1RI Register	012C <sub>16</sub>	U2BRG Register	0339 <sub>16</sub>
U1BRG Register	02E9 <sub>16</sub>	U2TB Register	033B <sub>16</sub> , 033A <sub>16</sub>
U1TB Register	02EB <sub>16</sub> , 02EA <sub>16</sub>	UDF Register	0344 <sub>16</sub>
U4BRG Register	02F9 <sub>16</sub>	TA0 Register <sup>(1)</sup>	0347 <sub>16</sub> , 0346 <sub>16</sub>
U4TB Register	02FB <sub>16</sub> , 02FA <sub>16</sub>	TA1 Register <sup>(1)</sup>	0349 <sub>16</sub> , 0348 <sub>16</sub>
TA11 Register	0303 <sub>16</sub> , 0302 <sub>16</sub>	TA2 Register <sup>(1)</sup>	034B <sub>16</sub> , 034A <sub>16</sub>
TA21 Register	0305 <sub>16</sub> , 0304 <sub>16</sub>	TA3 Register <sup>(1)</sup>	034D <sub>16</sub> , 034C <sub>16</sub>
TA41 Register	0307 <sub>16</sub> , 0306 <sub>16</sub>	TA4 Register <sup>(1)</sup>	034F <sub>16</sub> , 034E <sub>16</sub>
DTT Register	030C <sub>16</sub>	U0BRG Register	0369 <sub>16</sub>
ICTB2 Register	030D <sub>16</sub>	U0TB Register	036B <sub>16</sub> , 36A <sub>16</sub>

**NOTE:**

1. In one-shot timer mode and pulse width modulation mode only.



## 24.4 Clock Generation Circuit

### 24.4.1 CPU Clock

- When the CPU operating frequency is 24 MHz or more, use the following procedure for better EMC (Electromagnetic Compatibility) performance.
  - 1) Oscillator connected between the XIN and XOUT pins, or external clock applied to the XIN pin, has less than 24 MHz frequency.
  - 2) Use the PLL frequency synthesizer to multiply the main clock.

### 24.4.2 Sub Clock

Set the CM03 bit to "0" (XCIN-XCOUT drive capacity "LOW") when selecting the sub clock (XCIN-XCOUT) as the CPU clock, or Timer A or Timer B count source (fc32).

#### 24.4.2.1 Sub Clock Oscillation

When oscillating the sub clock, set the CM04 bit in the CM0 register to "1" (XCIN-XCOUT oscillation function) after setting the CM07 bit in the CM0 register to "0" (clock other than sub clock) and the CM03 bit to "1" (XCIN-XCOUT drive capacity "HIGH"). Set the CM03 bit to "0" after sub clock oscillation stabilizes.

Set the sub clock as the CPU clock, or Timer A or Timer B count source (fc32) after the above settings are completed.

#### 24.4.2.2 Using Stop Mode

When the microcomputer enters stop mode, the CM03 bit is automatically set to "1" (XCIN-XCOUT drive capacity "HIGH"). Use the following procedure to select the main clock as the CPU clock when entering stop mode.

- 1) Set the CM17 bit in the CM1 register to "0" (main clock).
- 2) Set the CM21 bit in the CM2 register to "0" (clock selected by the CM17 bit).
- 3) Set the CM07 bit in the CM0 register to "0" (clock selected by the CM21 bit divided by the MCD register setting).

After exiting stop mode, wait for the sub clock oscillation to stabilize. Then set the CM03 bit to "0" and the CM07 bit to "1" (sub clock).

#### 24.4.2.3 Oscillation Parameter Matching

If the sub clock oscillation parameters have only been evaluated with the drive capacity "HIGH", the parameters should be reevaluated for drive capacity "LOW".

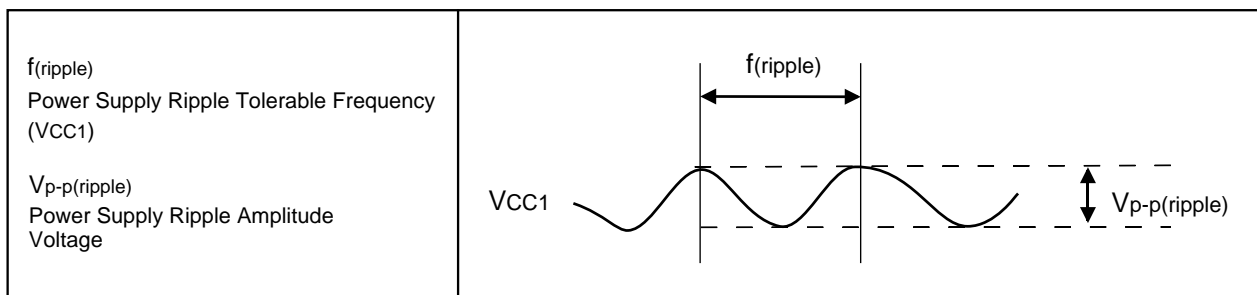
Contact your oscillator manufacturer for details on matching parameters.

### 24.4.3 PLL Frequency Synthesizer

Stabilize supply voltage to meet the power supply standard when using the PLL frequency synthesizer.

**Table 24.3 Power Supply Ripple**

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
$f_{(ripple)}$	Power Supply Ripple Tolerable Frequency ( $V_{CC1}$ )			10	kHz
$V_{P-P(ripple)}$	Power Supply Ripple Voltage Fluctuation Range	$V_{CC1}=5V$		0.5	V
		$V_{CC1}=3.3V$		0.3	V
$V_{CC1} ( \Delta V / \tau )$	Power Supply Ripple Voltage Fluctuation Rate	$V_{CC1}=5V$		1	V/ms
		$V_{CC1}=3.3V$		0.3	V/ms



**Figure 24.2 Power Supply Fluctuation Timing**

### 24.4.4 External Clock

Do not stop an external clock running if the main clock is selected as the CPU clock while the external clock is applied to the XIN pin.

Do not set the CM05 bit in the CM0 register to "1" (main clock stopped) while the external clock input is used for the CPU clock.

### 24.4.5 Clock Divide Ratio

Set the PM12 bit in the PM1 register to "0" (no wait state) when changing the MCD4 to MCD0 bit settings in the MCD register.

### 24.4.6 Power Consumption Control

Stabilize the main clock, sub clock or PLL clock to switch the CPU clock source to each clock.

#### 24.4.6.1 Wait Mode

When entering wait mode while the CM02 bit in the CM0 register is set to "1" (peripheral function stop in wait mode), set the MCD4 to MCD0 bits in the MCD register to maintain the 10-MHz CPU clock frequency or less.

When entering wait mode, the instruction queue reads ahead to instructions following the WAIT instruction, and the program stops. Write at least 4 NOP instructions after the WAIT instruction.

### 24.4.6.2 Stop Mode

- Use the following procedure to select the main clock as the CPU clock when entering stop mode.
  - 1) Set the CM17 bit in the CM1 register to "0" (main clock).
  - 2) Set the CM21 bit in the CM2 register to "0" (clock selected by the CM17 bit).
  - 3) Set the CM07 bit in the CM0 register to "0" (clock selected by the CM21 bit divided by the MCD register setting).

If the PLL clock is selected as the CPU clock source, set the CM17 bit to "0" (main clock) and the PLC07 bit in the PLC0 register to "0" (PLL off) before entering stop mode.

- The microcomputer cannot enter stop mode if a low-level signal ("L") is applied to the  $\overline{\text{NMI}}$  pin. Apply a high-level ("H") signal instead.
- If stop mode is exited by any reset, apply an "L" signal to the  $\overline{\text{RESET}}$  pin until a main clock oscillation is stabilized enough.
- If using the  $\overline{\text{NMI}}$  interrupt to exit stop mode, use the following procedure to set the CM10 bit in the CM1 register (all clocks stopped).
  - 1) Exit stop mode with using the  $\overline{\text{NMI}}$  interrupt.
  - 2) Generate a dummy interrupt.
  - 3) Set the CM10 bit to "1".

```
e.g.,      int   #63                ; dummy interrupt
           bset  cm1                ; all clocks stopped
```

```
/* dummy interrupt handling */
```

```
dummy
  reit
```

- When entering stop mode, the instruction queue reads ahead to instructions following the instruction setting the CM10 bit in the CM1 register to "1" (all clocks stopped), and the program stops. When the microcomputer exits stop mode, the instruction lined in the instruction queue is executed before the interrupt routine for recovery is done.

Write the JMP.B instruction, as follows, after the instruction setting the CM10 bit in the CM1 register to "1" (all clocks stopped).

```
e.g.,      bset 0, prcr              ; protection removed
           bset 0, cm1              ; all clocks stopped
           jmp.b LABEL_001          ; JMP.B instruction executed (no instuction between JMP.B
                                   ; and LABEL.)
```

```
LABEL_001:
  nop                ; NOP (1)
  nop                ; NOP (2)
  nop                ; NOP (3)
  nop                ; NOP (4)
  mov.b #0, prcr     ; Protection set
  .
  .
  .
```

### 24.4.6.3 Suggestions for Reducing Power Consumption

The followings are suggestions for reducing power consumption when programming or designing systems.

**Ports:** I/O ports maintains the same state despite the microcomputer entering wait mode or stop mode. Current flows through active output ports. Feedthrough current flows through input ports in a high-impedance state. Set unassigned ports as input ports and stabilize electrical potential before entering wait mode or stop mode.

**A/D Converter:** If the A/D conversion is not performed, set the VCUT bit in the AD0CON1 register to "0" (no VREF connection). Set the VCUT bit to "1" (VREF connection) and wait at least 1 $\mu$ s before starting the A/D conversion.

**D/A Converter:** Set the DAI bit ( $i=0, 1$ ) in the DACON register to "0" (output disabled) and set the DAI register to "00<sub>16</sub>" when the D/A conversion is not performed.

**Peripheral Function Stop:** Set the CM02 bit in the CM0 register while in wait mode to stop unnecessary peripheral functions. However, this does not reduce power consumption because the peripheral function clock (fc32) generating from the sub clock does not stop. When in low-speed mode and low-power consumption mode, do not enter wait mode when the CM02 bit is set to "1" (peripheral clock stops in wait mode).

## 24.5 Protection

The PRC2 bit setting in the PRCR register is changed to "0" (write disabled) when an instruction is written to any address after the PRC2 bit is set to "1" (write enabled). Write instruction immediately after setting the PRC2 bit to "1" to change registers protected by the PRC2 bit. Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the following instruction.

## 24.6 Interrupts

### 24.6.1 ISP Setting

After reset, the ISP is set to "00000016". The program runs out of control if an interrupt is acknowledged before the ISP is set. Therefore, the ISP must be set before an interrupt request is generated. Set the ISP to an even address, which allows interrupt sequences to be executed at a higher speed.

To use  $\overline{\text{NMI}}$  interrupt, set the ISP at the beginning of the program. The  $\overline{\text{NMI}}$  interrupt can be acknowledged after the first instruction has been executed after reset.

### 24.6.2 $\overline{\text{NMI}}$ Interrupt

- $\overline{\text{NMI}}$  interrupt cannot be denied. Connect the  $\overline{\text{NMI}}$  pin to Vcc via a resistor (pull-up) when not in use.
- The P8\_5 bit in the P8 register indicates the  $\overline{\text{NMI}}$  pin value. Read the P8\_5 bit only to determine the pin level after a  $\overline{\text{NMI}}$  interrupt occurs.
- "H" and "L" signals applied to the  $\overline{\text{NMI}}$  pin must be over 2 CPU clock cycles + 300 ns wide.
- $\overline{\text{NMI}}$  interrupt request may not be acknowledged if this and other interrupt requests are generated simultaneously.

### 24.6.3 $\overline{\text{INT}}$ Interrupt

- Edge Sensitive  
"H" and "L" signals applied to the  $\overline{\text{INT}}0$  to  $\overline{\text{INT}}5$  pins must be at least 250 ns wide, regardless of the CPU clock.
- Level Sensitive  
"H" and "L" signals applied to the  $\overline{\text{INT}}0$  to  $\overline{\text{INT}}5$  pins must be at least 1 CPU clock cycle + 200 ns wide. For example, "H" and "L" must be at least 234ns wide if  $X_{IN}=30\text{MHz}$  with no division.
- The IR bit setting may change to "1" (interrupt requested) when switching the polarity of the  $\overline{\text{INT}}0$  to  $\overline{\text{INT}}5$  pins. Set the IR bit to "0" (no interrupt requested) after selecting the polarity. Figure 24.3 shows an example of the switching procedure for the  $\overline{\text{INT}}$  interrupt.

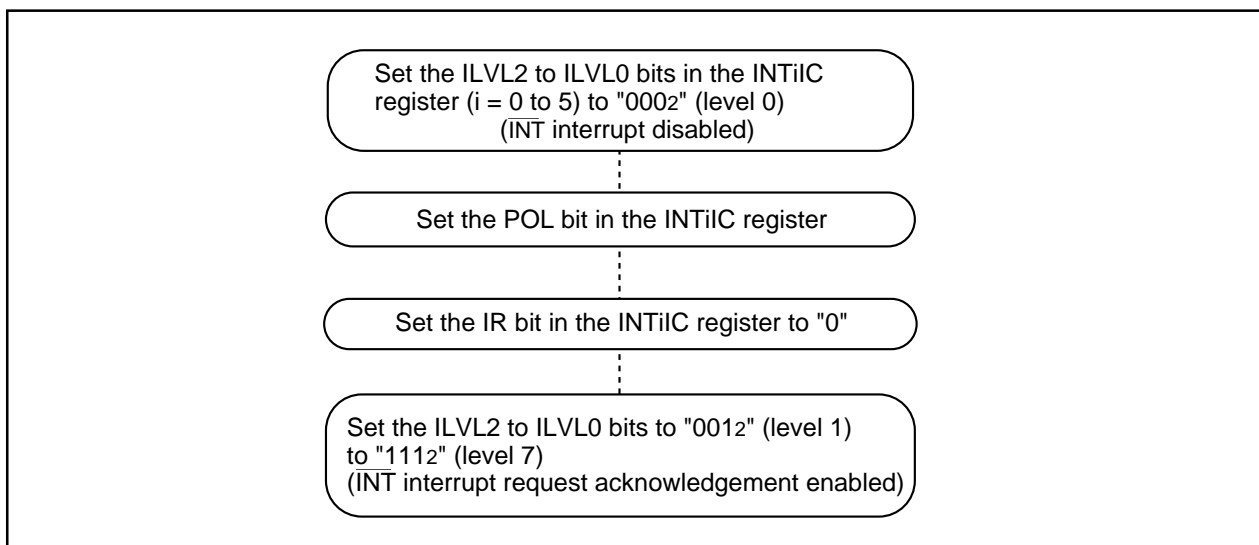


Figure 24.3 Switching Procedure for  $\overline{\text{INT}}$  Interrupt

#### 24.6.4 Watchdog Timer Interrupt

Reset the watchdog timer after a watchdog timer interrupt occurs.

#### 24.6.5 Changing Interrupt Control Register

To change the interrupt control register while the interrupt request is denied, follow the instructions below.

##### Changing IR bit

The IR bit setting may not change to "0" (no interrupt requested) depending on the instructions written. If this is a problem, use the following instruction to change the register: MOV

##### Changing Bits Except IR Bit

When an interrupt request is generated while executing an instruction, the IR bit may not be set to "1" (interrupt requested) and the interrupt may be ignored. If this is a problem, use the following instructions to change the register: AND, OR, BCLR, BSET

#### 24.6.6 Changing IIOiIR Register (i = 0 to 4)

Use the following instructions to set bits 1 to 7 in the IIOiIR register to "0" (no interrupt requested): AND, BCLR

#### 24.6.7 Changing RLVL Register

The DMAII bit is indeterminate after reset. When using the DMAII bit to generate an interrupt, set the interrupt control register after setting the DMAII bit to "0" (interrupt priority level 7 available for interrupts).

## 24.7 DMAC

- Set DMAC-associated registers while the MDi1 and MDi0 bits (i=0 to 3) in the channel to be used are set to "002" (DMA disabled). Set the MDi1 and MDi0 bits to "012" (single transfer) or "112" (repeat transfer) at the end of setup procedure to start DMA requests.
- Do not set the DRQ bit in the DMiSL register to "0" (no request).  
If a DMA request is generated but the receiving channel is not ready to receive<sup>(1)</sup>, the DMA transfer does not occur and the DRQ bit is set to "0".

### NOTE:

1. The MDi1 and MDi0 bits are set to "002" or the DCTi register is set to "0000<sub>16</sub>" (transferred 0 times).

- To start a DMA transfer by a software trigger, set the DSR bit and DRQ bit in the DMiSL register to "1" simultaneously.

e.g.,

OR.B #0A0h,DMiSL ; Set the DSR and DRQ bits to "1" simultaneously

- Do not generate a channel i DMA request when setting the MDi1 and MDi0 bits in the DMDj register (j=0,1) corresponding to channel i to "012" (single transfer) or "112" (repeat transfer), if the DCTi register of channel i is set to "1".
- Select the peripheral function which causes the DMA request after setting the DMA-associated registers. If none of the conditions above (setting  $\overline{\text{INT}}$  interrupt as DMA request source) apply, do not write "1" to the DCTi register.
- Enable DMA<sup>(2)</sup> after setting the DMiSL register (i=0 to 3) and waiting six BCLK cycles or more by program.

### NOTE:

2. DMA is enabled when the values set in the MDi1 and MDi0 bits in the DMDj register are changed from "002" (DMA disabled) to "012" (single transfer) or "112" (repeat transfer).



## 24.8 Timer

### 24.8.1 Timers A and B

Timers stop after reset. Set the TAI<sub>S</sub>(i=0 to 4) bit or TB<sub>j</sub>S(j=0 to 5) bit in the TABSR register or TBSR register to "1" (starts counting) after setting operating mode, count source and counter.

The following registers and bits must be set while the TAI<sub>S</sub> bit or TB<sub>j</sub>S bit is set to "0" (stops counting).

- TAI<sub>MR</sub>, TB<sub>j</sub>MR register
- TAI, TB<sub>j</sub> register
- UDF register
- TAZIE, TA0TGL, TA0TGH bits in the ONSF register
- TRGSR register

### 24.8.2 Timer A

The TA1<sub>OUT</sub>, TA2<sub>OUT</sub> and TA4<sub>OUT</sub> pins are placed in high-impedance states when a low-level ("L") signal is applied to the  $\overline{\text{NMI}}$  pin while the INV03 and INV02 bits in the INVC0 register are set to "112" (forced cutoff of the three-phase output by an "L" signal applied to the  $\overline{\text{NMI}}$  pin).

#### 24.8.2.1 Timer A (Timer Mode)

- The TAI<sub>S</sub> bit (i=0 to 4) in the TABSR register is set to "0" (stops counting) after reset. Set the TAI<sub>S</sub> bit to "1" (starts counting) after selecting an operating mode and setting the TAI register.
- The TAI register indicates the counter value during counting at any given time. However, the counter is "FFFF<sub>16</sub>" when reloading. The setting value can be read after setting the TAI register while the counter stops and before the counter starts counting.

#### 24.8.2.2 Timer A (Event Counter Mode)

- The TAI<sub>S</sub> (i=0 to 4) bit in the TABSR register is set to "0" (stops counting) after reset. Set the TAI<sub>S</sub> bit to "1" (starts counting) after selecting an operating mode and setting the TAI register.
- The TAI register indicates the counter values during counting at any given time. However, the counter will be "FFFF<sub>16</sub>" during underflow and "0000<sub>16</sub>" during overflow, when reloading. The setting value can be read after setting the TAI register while the counter stops and before the counter starts counting.

### 24.8.2.3 Timer A (One-shot Timer Mode)

- The TAI<sub>S</sub> (i=0 to 4) bit in the TABSR register is set to "0" (stops counting) after reset. Set the TAI<sub>S</sub> bit to "1" (starts counting) after selecting an operating mode and setting the TAI register.
- The followings occur when the TABSR register is set to "0" (stops counting) while counting:
  - The counter stops counting and the microcomputer reloads contents of the reload register.
  - The TAI<sub>OUT</sub> pin becomes low ("L").
  - The IR bit in the TAI<sub>IC</sub> register is set to "1" (interrupt requested) after one CPU clock cycle.
- The output of the one-shot timer is synchronized with an internal count source. When set to an external trigger, there is a delay of one count source cycle maximum, from trigger input to the TAI<sub>IN</sub> pin to the one-shot timer output.
- The IR bit is set to "1" when the following procedures are performed to set timer mode:
  - selecting one-shot timer mode after reset.
  - switching from timer mode to one-shot timer mode.
  - switching from event counter mode to one-shot timer mode.
 Therefore, set the IR bit to "0" to generate a timer A<sub>i</sub> interrupt (IR bit) after performing these procedures.
- When a trigger is generated while counting, the reload register reloads and continues counting after the counter has decremented once following a re-trigger. To generate a trigger while counting, wait at least 1 count source cycle after the previous trigger has been generated and generate a re-trigger.
- If an external trigger input is selected to start counting in timer A one-shot timer mode, do not provide another external trigger input again for 300 ns before the timer A counter value reaches "0000<sub>16</sub>". One-shot timer may stop counting.

### 24.8.2.4 Timer A (Pulse Width Modulation Mode)

- The TAI<sub>S</sub>(i=0 to 4) bit in the TABSR register is set to "0" (stops counting) after reset. Set the TAI<sub>S</sub> bit to "1" (starts counting) after selecting an operating mode and setting the TAI register.
- The IR bit is set to "1" when the following procedures are performed to set timer mode:
  - Selecting PWM mode after reset
  - Switching from timer mode to PWM mode
  - Switching from event counter mode to PWM mode
 Therefore, set the IR bit to "0" by program to generate a timer A<sub>i</sub> interrupt (IR bit) after performing these procedures.
- The followings occur when the TAI<sub>S</sub> bit is set to "0" (stops counting) while PWM pulse is output:
  - The counter stops counting
  - Output level changes to low ("L") and the IR bit changes to "1" when the TAI<sub>OUT</sub> pin is held high ("H")
  - The IR bit and the output level remain unchanged when TAI<sub>OUT</sub> pin is held "L"

### 24.8.3 Timer B

#### 24.8.3.1 Timer B (Timer Mode, Event Counter Mode)

- The TBiS (i=0 to 5) bit is set to "0" (stops counting) after reset. Set the TBiS bit to "1" (starts counting) after selecting an operating mode and setting TBi register.  
The TB2S to TB0S bits are bits 7 to 5 in the TABSR register. The TB5S to TB3S bits are bits 7 to 5 in the TBSR register.
- The TBi register indicates the counter value during counting at any given time. However, the counter is "FFFF16" when reloading. The setting value can be read after setting the TBi register while the counter stops and before the counter starts counting.

#### 24.8.3.2 Timer B (Pulse Period/Pulse Width Measurement Mode)

- The IR bit in the TBiIC (i=0 to 5) register is set to "1" (interrupt requested) when the valid edge of a pulse to be measured is input and when the timer Bi counter overflows. The MR3 bit in the TBiMR register determines the interrupt source within an interrupt routine.
- Use another timer to count how often the timer counter overflows when an interrupt source cannot be determined by the MR3 bit, such as when a pulse to be measured is input at the same time the timer counter overflows.
- To set the MR3 bit in the TBiMR register to "0" (no overflow), set the TBiMR register after the MR3 bit is set to "1" (overflow) and one or more cycles of the count source are counted, while the TBiS bits in the TABSR and TBSR registers are set to "1" (starts counting).
- The IR bit in the TBiIC register is used to detect overflow only. Use the MR3 bit only to determine interrupt source within an interrupt routine.
- Indeterminate values are transferred to the reload register during the first valid edge input after counting is started. Timer Bi interrupt request is not generated at this time.
- The counter value is indeterminate when counting is started. Therefore, the MR3 bit setting may change to "1" (overflow) and causes timer Bi interrupt requests to be generated until a valid edge is input after counting is started.
- The IR bit may be set to "1" (interrupt requested) if the MR1 and MR0 bits in the TBiMR register are set to a different value after a count begins. If the MR1 and MR0 bits are rewritten, but to the same value as before, the IR bit remains unchanged.
- Pulse width measurement measures pulse width continuously. Use program to determine whether measurement results are high ("H") or low ("L").

## 24.9 Serial I/O

### 24.9.1 Clock Synchronous Serial I/O Mode

The  $\overline{\text{RTS}}_2$  and  $\text{CLK}_2$  pins are placed in high-impedance states when a low-level ("L") signal is applied to the  $\overline{\text{NMI}}$  pin while the  $\text{INV03}$  and  $\text{INV02}$  bits in the  $\text{INVC0}$  register are set to "112" (forced cutoff of the three-phase output by an "L" signal applied to the  $\overline{\text{NMI}}$  pin).

#### 24.9.1.1 Transmission /Reception

When the  $\overline{\text{RTS}}$  function is used while an external clock is selected, the output level of the  $\overline{\text{RTS}}_i$  pin is held "L" indicating that the microcomputer is ready for reception. The transmitting microcomputer is notified that reception is possible. The output level of the  $\overline{\text{RTS}}_i$  pin becomes high ("H") when reception begins. Therefore, connecting the  $\overline{\text{RTS}}_i$  pin to the  $\overline{\text{CTS}}_i$  pin of the transmitting microcomputer synchronizes transmission and reception. The  $\overline{\text{RTS}}$  function is disabled if an internal clock is selected.

#### 24.9.1.2 Transmission

When an external clock is selected while the  $\text{CKPOL}$  bit in the  $\text{UiC0}$  ( $i=0$  to 4) register is set to "0" (data is transmitted on the falling edge of the transfer clock and received on the rising edge) and the external clock is held "H", or when the  $\text{CKPOL}$  bit is set to "1" (data is transmitted on the rising edge of the transfer clock and received on the falling edge) and the external clock is held "L", meet the following conditions:

- Set the  $\text{TE}$  bit in the  $\text{UiC1}$  register to "1" (receive enabled)
- Set the  $\text{TI}$  bit in the  $\text{UiC1}$  register to "0" (data in the  $\text{UiTB}$  register)
- Apply "L" signal to the  $\overline{\text{CTS}}_i$  pin if the  $\overline{\text{CTS}}$  function is selected

#### 24.9.1.3 Reception

Activating the transmitter in clock synchronous serial I/O mode generates the shift clock. Therefore, set for transmission even if the microcomputer is used for reception only. Dummy data is output from the  $\text{TxDi}$  pin while receiving.

If an internal clock is selected, the shift clock is generated when the  $\text{TE}$  bit in the  $\text{UiC1}$  registers is set to "1" (receive enabled) and dummy data is set in the  $\text{UiTB}$  register. If an external clock is selected, the shift clock is generated when the external clock is input into  $\text{CLK}_i$  pin while the  $\text{TE}$  bit is set to "1" (receive enabled) and dummy data is set in the  $\text{UiTB}$  register.

When receiving data consecutively while the  $\text{RE}$  bit in the  $\text{UiC1}$  register is set to "1" (data in the  $\text{UiRB}$  register) and the next data is received by the  $\text{UART}_i$  reception register, an overrun error occurs and the  $\text{OER}$  bit in the  $\text{UiRB}$  register is set to "1" (overrun error). In this case, the  $\text{UiRB}$  register is indeterminate. When overrun error occurs, program both reception and transmission registers to retransmit earlier data. The  $\text{IR}$  bit in the  $\text{SiRIC}$  does not change when an overrun error occurs.

When receiving data consecutively, feed dummy data to the low-order byte in the  $\text{UiTB}$  register every time a reception is made.

When an external clock is selected while the  $\text{CKPOL}$  bit in the  $\text{UiC0}$  register is set to "0" (data is transmitted on the falling edge of the transfer clock and received on the rising edge) and the external clock is held "H" or when the  $\text{CKPOL}$  bit is set to "1" (data is transmitted on the rising edge of the transfer clock and received on the falling edge) and the external clock is held "L", meet the following conditions:

- Set the  $\text{RE}$  bit in the  $\text{UiC1}$  register to "1" (receive enabled)
- Set the  $\text{TE}$  bit in the  $\text{UiC1}$  register to "1" (transmit enabled)
- Set the  $\text{TI}$  bit in the  $\text{UiC1}$  register to "0" (data in the  $\text{UiTB}$  register)

### 24.9.2 UART Mode

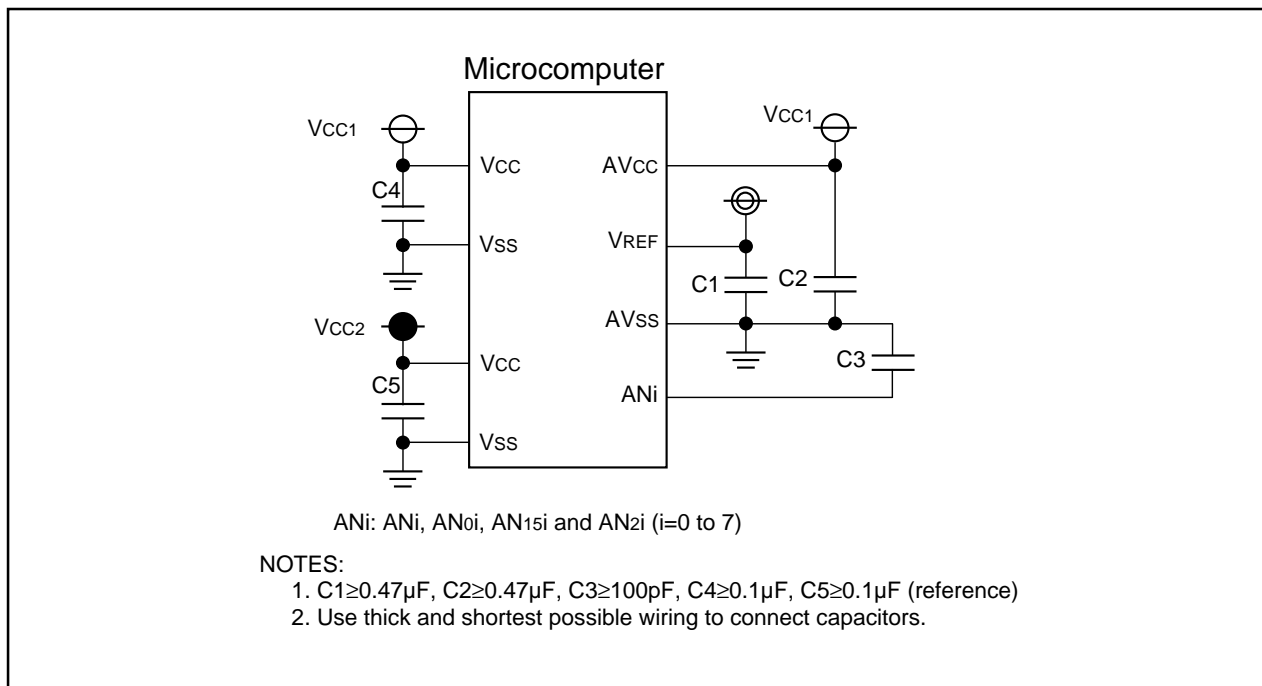
Set the UiERE bit (i=0 to 4) in the UiC1 register after setting the UiMR register.

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

To generate the start condition, stop condition or restart condition, set the STSPSEL bit in the UiSMR4 register to "0" first. Then, change each condition generating bit (the STAREQ bit, STPREQ bit or RSTAREQ bit) setting from "0" to "1" after going through a half cycle of the transfer clock.

## 24.10 A/D Converter

- Set the AD0CON0 (bit 6 excluded), AD0CON1, AD0CON2, AD0CON3, and AD0CON4 registers while the A/D conversion is stopped (before a trigger is generated).
- Wait a minimum of 1 $\mu$ s before starting the A/D conversion when changing the VCUT bit setting in the AD0CON1 register from "0" (VREF no connection) to "1" (VREF connection). Change the VCUT bit setting from "1" to "0" after the A/D conversion is completed.
- Insert capacitors between the AVCC pin, VREF pin, analog input pin ANi (i=0 to 7) and AVSS pin to prevent latch-ups and malfunctions due to noise, and to minimize conversion errors. The same applies to the VCC and VSS pins. Figure 24.4 shows the use of capacitors to reduce noise.



**Figure 24.4 Use of Capacitors to Reduce Noise**

- Set the bit in the port direction register, which corresponds to the pin being used as the analog input, to "0" (input mode). Set the bit in the port direction register, which corresponds to the  $\overline{\text{ADTRG}}$  pin, to "0" (input mode) if the TRG bit in the AD0CON0 register is set to "1" (external trigger).
- When generating a key input interrupt, do not use the AN4 to AN7 pins as analog input pins (key input interrupt request is generated when the A/D input voltage becomes "L").
- The  $\phi_{\text{AD}}$  frequency must be 16MHz or less. When the sample and hold function is not activated, the  $\phi_{\text{AD}}$  frequency must be 250 kHz or more. If the sample and hold function is activated, the  $\phi_{\text{AD}}$  frequency must be 1MHz or more.
- Set the CH2 to CH0 bits in the AD0CON0 register or the SCAN1 and SCAN0 bits in the AD0CON1 register to re-select analog input pins when changing A/D conversion mode.

- $AVCC = VREF = VCC1$ , A/D input voltage (for AN0 to AN7, ANEX0, and ANEX1)  $\leq VCC1$ .
- Wrong values are stored in the AD0i register (i=0 to 7) if the CPU reads the AD0i register while the AD0i register stores results from a completed A/D conversion. This occurs when the CPU clock is set to a divided main clock or a sub clock.  
In one-shot mode or single sweep mode, read the corresponding AD0i register after verifying that the A/D conversion has been completed. The IR bit in the AD0IC register determines the completion of the A/D conversion.  
In repeat mode, repeat sweep mode 0 and repeat sweep mode 1 use an undivided main clock as the CPU clock.
- Conversion results of the A/D converter are indeterminate if the ADST bit in the AD0CON0 register is set to "0" (A/D conversion stopped) and the conversion is forcibly terminated by program during the A/D conversion. The AD0i register not performing the A/D conversion may also be indeterminate.  
If the ADST bit is changed to "0" by program, during the A/D conversion, do not use any values obtained from the AD0i registers.
- External triggers cannot be used in DMAC operating mode. Do not read the AD00 register by program.
- Do not perform the A/D conversion in wait mode.
- Set the MCD4 to MCD0 bits in the MCD register to "100102" (no division) if using the sample and hold function.
- Do not acknowledge any interrupt requests, even if generated, before setting the ADST bit, if the A/D conversion is terminated by setting the ADST bit in the AD0CON0 register to "0" (A/D conversion stopped) while the microcomputer is A/D converting in single sweep mode.

## 24.11 Intelligent I/O

### 24.11.1 Register Setting

Operations, controlled by the values written to the G0RI and G1RI, G0TO and G1TO, G0CR and G1CR, G0RB and G1RB, G0MR and G1MR, G0EMR and G1EMR, G0ETC and G1ETC, G0ERC and G1ERC, G0IRF, G1IRF, G0TB and G1TB, G0CMP0 to G0CMP3, G1CMP0 to G1CMP3, G0MSK0 and G0MSK1, G1MSK0 and G1MSK1, G0TCRC and G1TCRC, G0RCRC and G1RCRC registers are affected by the transfer clock.

Set transfer clock before setting the G0RI and G1RI, G0TO and G1TO, G0CR and G1CR, G0RB and G1RB, G0MR and G1MR, G0EMR and G1EMR, G0ETC and G1ETC, G0ERC and G1ERC, G0IRF and G1IRF, G0TB and G1TB, G0CMP0 to G0CMP3, G1CMP0 to G1CMP3, G0MSK0 and G0MSK1, G1MSK0 and G1MSK1, G0TCRC and G1TCRC, G0RCRC and G1RCRC registers.



## 24.12 Programmable I/O Ports

- Because ports P72 to P75, P80, and P81 have three-phase PWM output forced cutoff function, they are affected by the three-phase motor control timer function and the  $\overline{\text{NMI}}$  pin when these ports are set for output functions (port output, timer output, three-phase PWM output, serial I/O output, intelligent I/O output).

Table 24.4 shows the INVC0 register setting, the  $\overline{\text{NMI}}$  pin input level and the state of output ports.

**Table 24.4 INVC0 Register and the  $\overline{\text{NMI}}$  Pin**

Setting Value of the INVC0 Register		Signal level Applied to the $\overline{\text{NMI}}$ Pin	P72 to P75, P80, P81 Pin States (When Setting Them as Output Pins)
INV02 Bit	INV03 Bit		
0 (Not Using the Three-Phase Motor Control Timer Functions)	-	-	Provides functions selected by the PS1, PSL1, PSC, PS2, PSL2 registers
1 (Using the Three-Phase Motor Control Timer Functions)	0 (Three-Phase Motor Control Timer Output Disabled)	-	High-impedance state
	1 (Three-Phase Motor Control Timer Output Enabled) <sup>(1)</sup>	H	Provides functions selected by the PS1, PSL1, PSC, PS2, PSL2 registers
		L (Forcibly Terminated)	High-impedance state

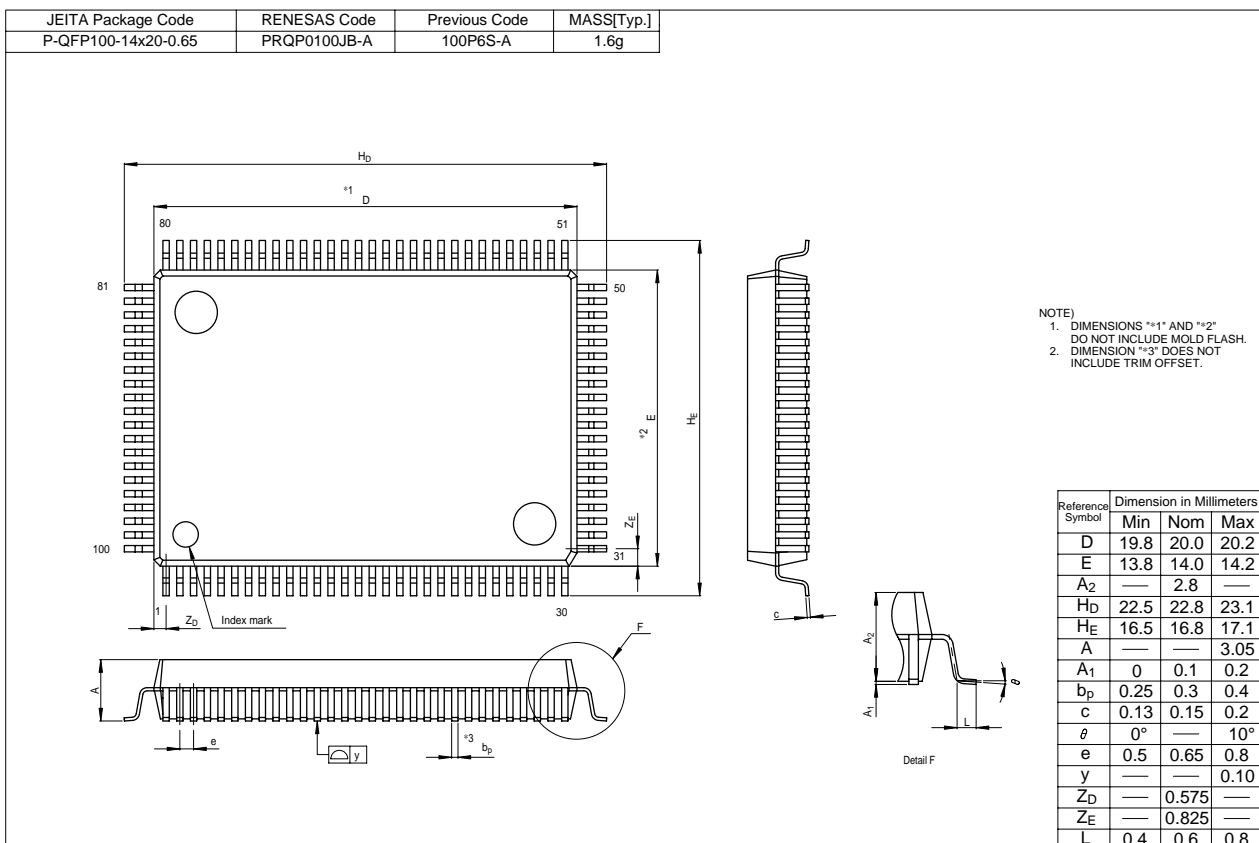
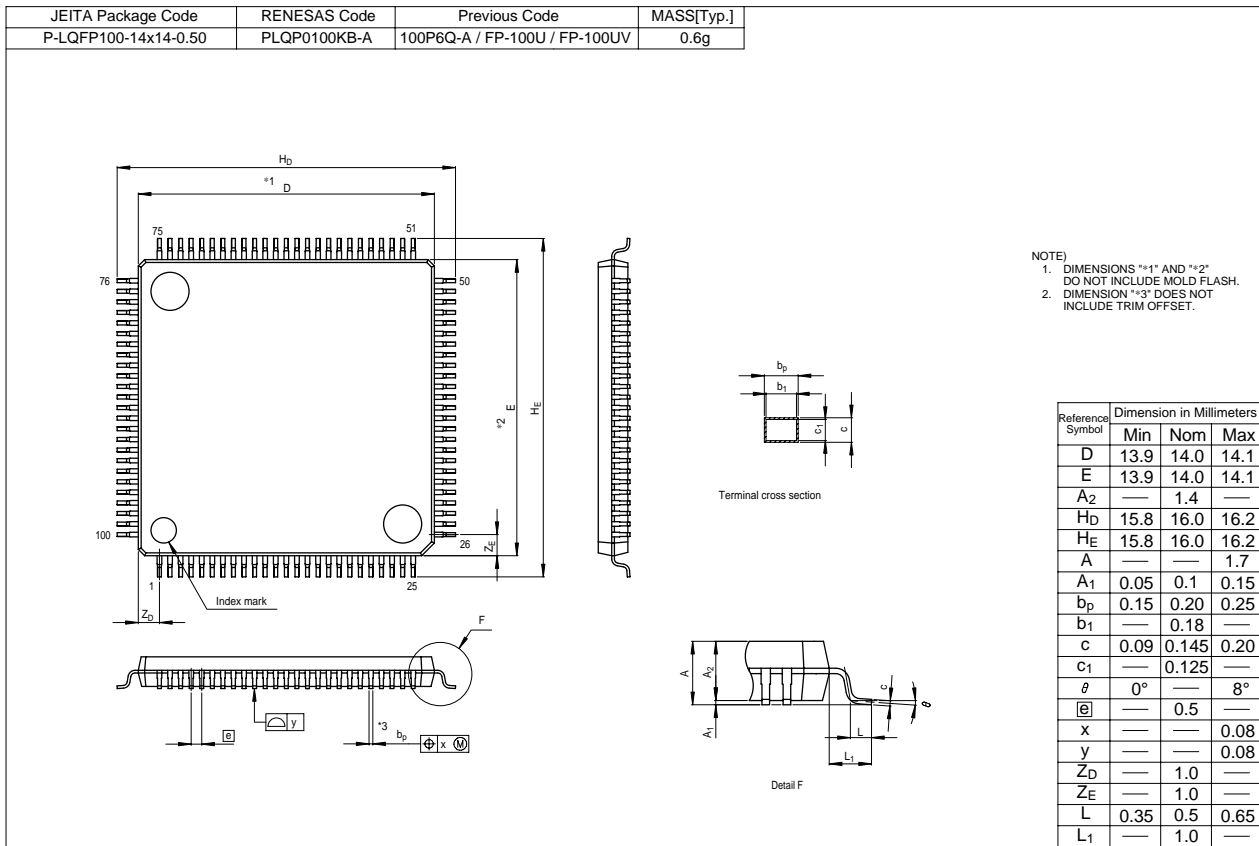
**NOTE:**

1. The INV03 bit is set to "0" after a low-level ("L") signal is applied to the  $\overline{\text{NMI}}$  pin.
- The availability of pull-up resistors is indeterminate until internal power voltage stabilizes, if the RESET pin is held "L".
  - The input threshold voltage varies between programmable I/O ports and peripheral functions. Therefore, if the level of the voltage applied to a pin shared by both programmable I/O ports and peripheral functions is not within the recommended operating condition,  $V_{IH}$  and  $V_{IL}$  (neither "H" nor "L"), the level may vary depending on the programmable ports and peripheral functions.

### 24.13 Noise

Connect a bypass capacitor (0.1 $\mu$ F or more) between Vcc and Vss by shortest path, using thick wires.

# Package Dimensions



# Register Index

## A

AD00 to AD07 **226**  
 AD0CON0 **222**  
 AD0CON1 **223**  
 AD0CON2 **224**  
 AD0CON3 **225**  
 AIER **101**

## C

CCS **255**  
 CM0 **58, 107**  
 CM1 **59**  
 CM2 **61**  
 CPSRF **62**  
 CRCD **239**  
 CRCIN **239**

## D

DA0, DA1 **238**  
 DACON **238**  
 DCT0 to DCT3 **114**  
 DM0SL to DM3SL **111**  
 DMA0 to DMA3 **115**  
 DMD0, DMD1 **113**  
 DRA0 to DRA3 **115**  
 DRC0 to DRC3 **114**  
 DS **39**  
 DSA0 to DSA3 **115**  
 DTT **161**

## E

EWCR0 to EWCR3 **45**

## G

G0CMP0 to G0CMP3 **254**  
 G0CR, G1CR **247**  
 G0DR, G1DR **253**  
 G0EMR **249**  
 G0ERC, G1ERC **251**  
 G0ETC **250**  
 G0IRF **252**  
 G0MR **248**  
 G0MSK0, G0MSK1 **254**

G0RB, G1RB **247**  
 G0RCRC, G1RCRC **254**  
 G0RI, G1RI **246**  
 G0TB, G1TB **253**  
 G0TCRC, G1TCRC **254**  
 G0TO, G1TO **246**  
 G1CMP0 to G1CMP3 **254**  
 G1EMR **249**  
 G1ETC **250**  
 G1IRF **253**  
 G1MR **248**  
 G1MSK0, G1MSK1 **254**

## I

ICTB2 **162**  
 IDB0, IDB1 **161**  
 IFSR **99, 176**  
 IIO0IE to IIO4IE **104**  
 IIO0IR to IIO4IR **103**  
 Interrupt Control **90, 91**  
 INVC0 **159**  
 INVC1 **160**

## M

MCD **60**

## O

ONSF **133**

## P

P0 to P15 **268**  
 PCR **276**  
 PD0 to PD15 **267**  
 PLC0 **63**  
 PLC1 **63**  
 PM0 **36**  
 PM1 **37**  
 PM2 **64**  
 PRCR **81**  
 PS0 **269**  
 PS1 **269**  
 PS2 **270**  
 PS3 **270**

PSC 273  
PSC3 274  
PSD1 274  
PSL0 271  
PSL1 271  
PSL2 272  
PSL3 272  
PUR0 275  
PUR1 275  
PUR2 275  
PUR3 276

## R

RLVL 92, 121  
RMAD0 to RMAD7 101

## T

TA0 to TA4 131  
TA0MR to TA4MR 132, 137, 140, 143, 145  
TA1, TA2, TA4, TA11, TA21, TA41 162  
TA1MR, TA2MR, TA4MR 164  
TABSR 132, 148, 163  
TB0 to TB5 147  
TB0MR to TB5MR 148, 150, 152, 154  
TB2 163  
TB2MR 164  
TB2SC 162  
TBSR 149  
TCSPR 62, 134  
TRGSR 134, 163

## U

U0BRG to U4BRG 170  
U0C0 to U4C0 171  
U0C1 to U4C1 172  
U0MR to U4MR 170  
U0RB to U4RB 169  
U0SMR to U4SMR 172  
U0SMR2 to U4SMR2 173  
U0SMR3 to U4SMR3 174  
U0SMR4 to U4SMR4 175  
U0TB to U4TB 169  
UDF 133

## W

WDC 106  
WDTS 106

## X

X0R to X15R 241  
XYC 241

## Y

Y0R to Y15R 241



---

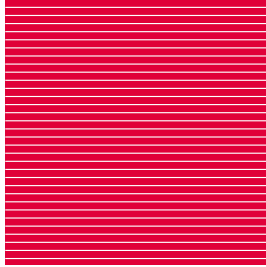
**RENESAS 16/32-BIT SINGLE-CHIP MICROCOMPUTER  
HARDWARE MANUAL  
M32C/80 Group**

**Publication Data : Rev.1.00 Nov. 01, 2005**

**Published by : Sales Strategic Planning Div.  
Renesas Technology Corp.**

---

# M32C/80 Group Hardware Manual



Renesas Technology Corp.  
2-6-2, Ote-machi, Chiyoda-ku, Tokyo, 100-0004, Japan