Spansion® Analog and Microcontroller Products

The following document contains information on Spansion analog and microcontroller products. Although the document is marked with the name “Fujitsu”, the company that originally developed the specification, Spansion will continue to offer these products to new and existing customers.

Continuity of Specifications

There is no change to this document as a result of offering the device as a Spansion product. Any changes that have been made are the result of normal document improvements and are noted in the document revision summary, where supported. Future routine revisions will occur when appropriate, and changes will be noted in a revision summary.

Continuity of Ordering Part Numbers

Spansion continues to support existing part numbers beginning with “MB”. To order these products, please use only the Ordering Part Numbers listed in this document.

For More Information

Please contact your local sales office for additional information about Spansion memory, analog, and microcontroller products and solutions.

FUJITSU SEMICONDUCTOR
CONTROLLER MANUAL

MN702-00005-2v0-E

New 8FX

8-BIT MICROCONTROLLER

MB95410H/470H Series
HARDWARE MANUAL

FUJITSU

New 8FX

8-BIT MICROCONTROLLER

MB95410H/470H Series

HARDWARE MANUAL

For the information for microcontroller supports, see the following website.

http://edevice.fujitsu.com/micom/en-support/

FUJITSU SEMICONDUCTOR LIMITED

PREFACE

■ The Purpose and Intended Readership of This Manual

Thank you very much for your continued special support for Fujitsu Semiconductor products.

The MB95410H/470H Series is a line of products developed as general-purpose products in the New 8FX family of proprietary 8-bit single-chip microcontrollers applicable as application-specific integrated circuits (ASICs). The MB95410H/470H Series can be used for a wide range of applications from consumer products including portable devices to industrial equipment.

Intended for engineers who actually develop products using the MB95410H/470H Series of microcontrollers, this manual describes its functions, features, and operations. You should read through the manual.

For details on individual instructions, refer to "F^2MC-8FX Programming Manual".

Note: F^2MC is the abbreviation of FUJITSU Flexible Microcontroller.

■ Trademark

The company names and brand names in this document are the trademarks or registered trademarks of their respective owners.

■ Sample Programs

Fujitsu Semiconductor provides sample programs free of charge to operate the peripheral resources of the New 8FX family of microcontrollers. Feel free to use such sample programs to check the operational specifications and usages of Fujitsu microcontrollers.

Note that sample programs are subject to change without notice. As these pieces of software are offered to show standard operations and usages, evaluate them sufficiently before use with your system. Fujitsu Semiconductor assumes no liability for any damages whatsoever arising out of the use of sample programs.

- FUJITSU SEMICONDUCTOR LIMITED, its subsidiaries and affiliates (collectively, "FUJITSU SEMICONDUCTOR") reserves the right to make changes to the information contained in this document without notice. Please contact your FUJITSU SEMICONDUCTOR sales representatives before order of FUJITSU SEMICONDUCTOR device.
  Customers are advised to consult with sales representatives before ordering.
- Information contained in this document, such as descriptions of function and application circuit examples is presented solely for reference to examples of operations and uses of FUJITSU SEMICONDUCTOR device. FUJITSU SEMICONDUCTOR disclaims any and all warranties of any kind, whether express or implied, related to such information, including, without limitation, quality, accuracy, performance, proper operation of the device or non-infringement. If you develop equipment or product incorporating the FUJITSU SEMICONDUCTOR device based on such information, you must assume any responsibility or liability arising out of or in connection with such information or any use thereof. FUJITSU SEMICONDUCTOR assumes no responsibility or liability for any damages whatsoever arising out of or in connection with such information or any use thereof.
- Nothing contained in this document shall be construed as granting or conferring any right under any patents, copyrights, or any other intellectual property rights of FUJITSU SEMICONDUCTOR or any third party by license or otherwise, express or implied. FUJITSU SEMICONDUCTOR assumes no responsibility or liability for any infringement of any intellectual property rights or other rights of third parties resulting from or in connection with the information contained herein or use thereof.
- The products described in this document are designed, developed and manufactured as contemplated for general use including without limitation, ordinary industrial use, general office use, personal use, and household use, but are not designed, developed and manufactured as contemplated (1) for use accompanying fatal risks or dangers that, unless extremely high levels of safety is secured, could lead directly to death, personal injury, severe physical damage or other loss (including, without limitation, use in nuclear facility, aircraft flight control system, air traffic control system, mass transport control system, medical life support system and military application), or (2) for use requiring extremely high level of reliability (including, without limitation, submersible repeater and artificial satellite). FUJITSU SEMICONDUCTOR shall not be liable for you and/or any third party for any claims or damages arising out of or in connection with above-mentioned uses of the products.
- Any semiconductor devices fail or malfunction with some probability. You are responsible for providing adequate designs and safeguards against injury, damage or loss from such failures or malfunctions, by incorporating safety design measures into your facility, equipments and products such as redundancy, fire protection, and prevention of overcurrent levels and other abnormal operating conditions.
- The products and technical information described in this document are subject to the Foreign Exchange and Foreign Trade Control Law of Japan, and may be subject to export or import laws or regulations in U.S. or other countries. You are responsible for ensuring compliance with such laws and regulations relating to export or re-export of the products and technical information described herein.
- All company names, brand names and trademarks herein are property of their respective owners.

CONTENTS

Major revisions in this edition

A change on a page is indicated by a vertical line drawn on the left of that page.

Page	Revisions (For details, see their respective pages.)	
-	-	Revised the family name. F^2MC-8FX → New 8FX
4	CHAPTER 1 OVERVIEW 1.1 Features of MB95410H/470H Series ■ Features of MB95410H/470H Series	Added the section "● Power-on reset".
5	1.2 Product Line-up of MB95410H/470H Series ■ Product Line-up of MB95410H/ 470H Series Table 1.2-1	Renamed the parameter "ROM capacity" to "Flash memory capacity".
		Added the parameter "Power-on reset".
8	1.2 Product Line-up of MB95410H/470H Series ■ Product Line-up of MB95410H/ 470H Series Table 1.2-2	Renamed the parameter "ROM capacity" to "Flash memory capacity".
		Added the parameter "Power-on reset".
21	1.7 Pin Functions ■ Pin Functions (MB95410H Series) Table 1.7-1	Corrected details of the function of the $\overline{\text{RST}}$ pin. External reset pin → Reset pin Dedicated reset pin for MB95F414H/F416H/F418H
26	1.7 Pin Functions ■ Pin Functions (MB95470H Series) Table 1.7-2	Corrected details of the function of the TO01 pin. 8/16-bit composite timer ch. 0 clock output pin → 8/16-bit composite timer ch. 0 output pin
28		Corrected details of the function of the $\overline{\text{RST}}$ pin. Reset pin → Reset pin Dedicated reset pin for MB95F474H/F476H/F478H
31	1.8 I/O Circuit Types	Corrected the cross reference in the section summary. Table 1.8-1 → Table 1.7-1 and Table 1.7-2
39	CHAPTER 2 NOTES ON DEVICE HANDLING 2.1 Notes on Device Handling ■ Pin Connection	Revised details of "• DBG pin".
		Revised details of "• $\overline{\text{RST}}$ pin".
	2.1 Notes on Device Handling ■ Pin Connection • C pin	Corrected the following statement. The bypass capacitor for the V_{CC} pin must have a capacitance larger than C_S. ↓ The decoupling capacitor for the V_{CC} pin must have a capacitance equal to or larger than the capacitance of C_S.
42	CHAPTER 3 MEMORY SPACE 3.1 Memory Space	Added "an extended I/O area" to the section summary.

Page	Revisions (For details, see their respective pages.)	
62	CHAPTER 6 CLOCK CONTROLLER 6.1 Overview of Clock Controller ■ Overview of Clock Controller	Corrected the following content. This device has four source clocks: → This device has five source clocks:
66	6.1 Overview of Clock Controller ■ Clock Modes	Corrected the following statement. There are five clock modes: main clock (or main PLL clock) mode, main CR clock mode, subclock mode, and sub-CR clock mode. → There are five clock modes: main clock mode, main PLL clock mode, main CR clock mode, subclock mode and sub-CR clock mode.
		Revised Table 6.1-1.
68	6.1 Overview of Clock Controller ■ Combinations of Clock Mode and Standby Mode Table 6.1-4	Renamed the parameter "ROM" to "Flash memory".
69	6.1 Overview of Clock Controller ■ Combinations of Clock Mode and Standby Mode Table 6.1-5	Renamed the parameter "ROM" to "Flash memory".
72	6.3 System Clock Control Register (SYCC) ■ Configuration of System Clock Control Register (SYCC) Figure 6.3-1	Corrected the initial value of the SYCC register. $0000X0011_B$ → $0000X011_B$
81	6.7 System Clock Control Register 2 (SYCC2) ■ Configuration of System Clock Control Register 2 (SYCC2) Figure 6.7-1	Corrected details of the MOSCE bit in the SYCC2 register.
82	6.7 System Clock Control Register 2 (SYCC2) ■ Configuration of System Clock Control Register 2 (SYCC2) Table 6.7-1	Revised the name of the MOSCE bit. Main clock oscillation enable bit → Main clock (or main PLL clock) oscillation enable bit
83	6.8 Clock Modes	Corrected the following statement in the section summary. There are five clock modes: main clock (or main PLL clock) mode, main CR clock mode, subclock mode and sub-CR clock mode. → There are five clock modes: main clock mode, main PLL clock mode, main CR clock mode, subclock mode and sub-CR clock mode.
	6.8 Clock Modes ■ Operations in Subclock Mode	Corrected the following statement. While the device is operating in subclock clock (or main PLL clock) mode, it can be set to transit to one of the following standby mode: sleep mode, stop mode, or watch mode. → While the device is operating in subclock clock mode, it can be set to transit to one of the following standby mode: sleep mode, stop mode, or watch mode.

Page	Revisions (For details, see their respective pages.)	
84	6.8 Clock Modes ■ Clock Mode State Transition Diagram	Corrected the following statement. There are five clock modes: main clock (or main PLL clock) mode, main CR clock mode, subclock mode and sub-CR clock mode. → There are five clock modes: main clock mode, main PLL clock mode, main CR clock mode, subclock mode and sub-CR clock mode.
95	6.9.5 Watch Mode	Corrected the following statement in the section summary. In watch mode, only the subclock, the sub-CR clock and the watch prescaler operate. → In watch mode, only the subclock, the sub-CR clock, the watch prescaler and the LCD controller operate.
97	6.11 Overview of Prescaler ■ Prescaler	Added a remark on F_{CH}.
98	6.12 Configuration of Prescaler ■ Block Diagram of Prescaler Figure 6.12-1	Added a remark on F_{CH}.
	6.12 Configuration of Prescaler ■ Block Diagram of Prescaler	Added a remark on F_{CH}.
99	6.13 Operation of Prescaler ■ Operation of Prescaler	Added a remark on F_{CH}.
102	CHAPTER 7 RESET 7.1 Reset Operation ■ Reset Sources	Corrected the following statement. There are four reset sources for the reset. → There are five reset sources for the reset.
		Revised Table 7.1-1.
		Added the section "● Power-on reset".
103		Deleted the section "● Power-on reset/low-voltage detection reset (optional)".
		Added the section "● Low-voltage detection reset (optional)".
106	7.2 Reset Source Register (RSRR) ■ Configuration of Reset Source Register (RSRR) Figure 7.2-1	Corrected the initial value. $XXXXXXXX_B \rightarrow 000XXXXX_B$
107	7.2 Reset Source Register (RSRR) ■ Configuration of Reset Source Register (RSRR) Table 7.2-1	Corrected the following statement in details of the function of the PONR bit. The low-voltage detection reset function is available only in certain products. → The low-voltage detection reset function is only available on MB95F414K/F416K/F418K/F474K/F476K/F478K.
108	7.2 Reset Source Register (RSRR) ■ State of Reset Source Register (RSRR)	Revised Table 7.2-2.
130	CHAPTER 9 I/O PORTS (MB95410H SERIES) 9.2.2 Operations of Port 0 ■ Operations of Port 0 ● Operation as an analog input pin	Deleted the following statement. In addition, set the corresponding bit in the PUL register to "0".
142	9.4.2 Operations of Port 2 ■ Operations of Port 2	Corrected details of the section "● Operation as an analog input pin".

Page	Revisions (For details, see their respective pages.)	
146	9.5.2 Operations of Port 4 ■ Operations of Port 4 ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
148	9.6 Port 5 ■ Port 5 Pins Table 9.6-1	Corrected the shared peripheral function of the P52/TI0/TO00 pin. TI0: 16-bit reload timer output → TI0: 16-bit reload timer input
151	9.6.2 Operations of Port 5 ■ Operations of Port 5 ● Operation as an input port	Added the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
156	9.7.2 Operations of Port 6 ■ Operations of Port 6 ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
157	9.7.2 Operations of Port 6 ■ Operations of Port 6	Deleted the section "● Operation as a peripheral function output pin".
161	9.8.2 Operations of Port 9 ■ Operations of Port 9 ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
166	9.9.2 Operations of Port A ■ Operations of Port A ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
171	9.10.2 Operations of Port B ■ Operations of Port B ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
	9.10.2 Operations of Port B ■ Operations of Port B	Deleted the section "● Operation as a peripheral function output pin".
176	9.11.2 Operations of Port C ■ Operations of Port C ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
181	9.12.2 Operations of Port E ■ Operations of Port E ● Operation as an input port	Added the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
182	9.12.2 Operations of Port E ■ Operations of Port E ● Operation as a peripheral function input pin	Deleted the following statement. When using the analog input shared pin as another peripheral function input pin, configure it as an input port, which is the same as the operation as an input port.
183	9.13 Port F ■ Port F Pins Table 9.13-1	Corrected the shared peripheral function of the PF2/$\overline{\text{RST}}$ pin. $\overline{\text{RST}}$: External reset pin → $\overline{\text{RST}}$: Reset pin
186	9.13.2 Operations of Port F ■ Operations of Port F ● Operation as an input port	Added the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
191	9.14.2 Operations of Port G ■ Operations of Port G ● Operation as an input port	Added the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
195	CHAPTER 10 I/O PORTS (MB95470H SERIES) 10.1 Overview of I/O Ports ■ Overview of I/O Ports Table 10.1-1	Added details of the "Port G pull-up register".

Page	Revisions (For details, see their respective pages.)	
197	10.2 Port 0 ■ Port 0 Pins Table 10.2-1	Corrected the shared peripheral function of the P01/INT01/AN01/SEG28/UI2/TO00 pin. SEG36: LCDC SEG28 output → SEG28: LCDC SEG28 output
209	10.3.2 Operations of Port 1 ■ Operations of Port 1 ● Operation of the input level select register	Corrected the following statement. When changing the input level of P10, ensure that the peripheral function (UART/SIO ch. 0 output) has been stopped. → When changing the input level of P10, ensure that all its shared peripheral functions have been stopped.
214	10.4.2 Operations of Port 2 ■ Operations of Port 2	Corrected details of the section "● Operation as an analog input pin".
218	10.5.2 Operations of Port 6 ■ Operations of Port 6 ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
	10.5.2 Operations of Port 6 ■ Operations of Port 6	Deleted the section "● Operation as a peripheral function output pin".
223	10.6.2 Operations of Port 9 ■ Operations of Port 9 ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
228	10.7.2 Operations of Port A ■ Operations of Port A ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
233	10.8.2 Operations of Port B ■ Operations of Port B ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
	10.8.2 Operations of Port B ■ Operations of Port B	Deleted the section "● Operation as a peripheral function output pin".
238	10.9.2 Operations of Port C ■ Operations of Port C ● Operation as an output port	Deleted the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
	10.9.2 Operations of Port C ■ Operations of Port C	Deleted the section "● Operation as a peripheral function output pin".
243	10.10.2 Operations of Port E ■ Operations of Port E ● Operation as an input port	Added the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
	10.10.2 Operations of Port E ■ Operations of Port E ● Operation as a peripheral function input pin	Deleted the following statement. When using the analog input shared pin as another peripheral function input pin, configure it as an input port, which is the same as the operation as an input port.
245	10.11 Port F ■ Port F Pins Table 10.11-1	Corrected the shared peripheral function of the PF2/$\overline{\text{RST}}$ pin. $\overline{\text{RST}}$: External reset pin → $\overline{\text{RST}}$: Reset pin
248	10.11.2 Operations of Port F ■ Operations of Port F ● Operation as an input port	Added the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.
253	10.12.2 Operations of Port G ■ Operations of Port G ● Operation as an input port	Added the following statement. For a pin shared with other peripheral functions, disable the output of such peripheral functions.

Page	Revisions (For details, see their respective pages.)	
271	CHAPTER 12 HARDWARE/ SOFTWARE WATCHDOG TIMER 12.2 Configuration of Watchdog Timer ■ Block Diagram of Watchdog Timer	Revised Figure 12.2-1.
274	12.3.1 Watchdog Timer Control Register (WDTC) ■ Watchdog Timer Control Register (WDTC) Figure 12.3-2	Added a remark on F_{CH}.
275	12.3.1 Watchdog Timer Control Register (WDTC) ■ Watchdog Timer Control Register (WDTC) Table 12.3-1	Added a remark on F_{CH}.
294	CHAPTER 14 WATCH COUNTER 14.1 Overview of Watch Counter ■ Watch Counter Table 14.1-1	Corrected the frequencies of the count clock. $F_{CL}/2^{12} \rightarrow 2^{12}/F_{CL}$ $F_{CL}/2^{13} \rightarrow 2^{13}/F_{CL}$ $F_{CL}/2^{14} \rightarrow 2^{14}/F_{CL}$ $F_{CL}/2^{15} \rightarrow 2^{15}/F_{CL}$
295	14.2 Configuration of Watch Counter ■ Block Diagram of Watch Counter Figure 14.2-1	Corrected the frequencies of the count clock. $F_{CL}/2^{12} \rightarrow 2^{12}/F_{CL}$ $F_{CL}/2^{13} \rightarrow 2^{13}/F_{CL}$ $F_{CL}/2^{14} \rightarrow 2^{14}/F_{CL}$ $F_{CL}/2^{15} \rightarrow 2^{15}/F_{CL}$
303	14.5 Operations of Watch Counter and Setting Procedure Example ■ Operation in Main Stop Mode	Corrected the following statement. Moreover, the clock counter stops, too, when subclock oscillation stop bit (SYCC: SUBS) of the system clock control register is set to "1". → Moreover, the watch counter stops, too, when the subclock oscillation enable bit (SOSCE) in the system clock control register 2 (SYCC2) is set to "0".
305	14.7 Sample Settings for Watch Counter ■ Sample Settings ● How to enable/stop the watch counter	Corrected the name of the ISEL bit. watch timer initialization bit → watch counter start & interrupt request enable bit
343	CHAPTER 17 INTERRUPT PIN SELECTION CIRCUIT 17.6 Notes on Using Interrupt Pin Selection Circuit	Deleted the following statement. With multiple interrupt pin selected in the WICR register simultaneously, if any of the signal input to one of the selected interrupt pins is "H", an input to INT00 (ch. 0) of the external interrupt circuit will be treated as "H" (as a result of the "OR" logic of the signals that has been input to the selected pins).

Page	Revisions (For details, see their respective pages.)	
363	CHAPTER 18 8/16-BIT COMPOSITE TIMER 18.5.1 8/16-bit Composite Timer 00/01 Status Control Register 0 (T00CR0/T01CR0) ■ 8/16-bit Composite Timer 00/01 Status Control Register 0 (T00CR0/T01CR0) Table 18.5-1	Added the following statement to details of the function of the IFE bit. During timer operation (T00CR1/T01CR1:STA = 1), the write access to this bit has no effect on operation. Ensure that the timer has stopped before modifying this bit.
		Corrected the following statement. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from either main clock or main CR clock. → Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock.
366	18.5.2 8/16-bit Composite Timer 10/11 Status Control Register 0 (T10CR0/T11CR0) ■ 8/16-bit Composite Timer 10/11 Status Control Register 0 (T10CR0/T11CR0) Table 18.5-2	Added the following statement to details of the function of the IFE bit. During timer operation (T10CR1/T11CR1:STA = 1), the write access to this bit has no effect on operation. Ensure that the timer has stopped before modifying this bit.
		Corrected the following statement. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from either main clock or main CR clock. → Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock.
422	CHAPTER 19 16-BIT RELOAD TIMER 19.2 Configuration of 16-bit Reload Timer	Corrected the register name of the TMRH0 and TMRL0 registers. 16-bit timer register (TMRH0, TMRL0) → 16-bit reload timer timer register (TMRH0, TMRL0)
		Corrected the register name of the TMRLRH0 and TMRLRL0 registers. 16-bit reload register (TMRLRH0, TMRLRL0) → 16-bit reload timer reload register (TMRLRH0, TMRLRL0)
		Corrected the register name of the TMCSRH0 and TMCSRL0 registers. Timer control status register (TMCSRH0, TMCSRL0) → 16-bit reload timer control status register (TMCSRH0, TMCSRL0)

Page	Revisions (For details, see their respective pages.)	
422	19.2 Configuration of 16-bit Reload Timer ■ Block Diagram of 16-bit Reload Timer Figure 19.2-1	Corrected the register name of the TMRLRH0 and TMRLRL0 registers. 16-bit reload register (TMRLRH0, TMRLRL0) → 16-bit reload timer reload register (TMRLRH0, TMRLRL0)
		Corrected the register name of the TMRH0 and TMRL0 registers. 16-bit timer register (TMRH0, TMRL0) → 16-bit reload timer timer register (TMRH0, TMRL0)
		Deleted "Timer control status register (TMCSR).
		Added "16-bit reload timer reload register upper (TMRLRH0)" and "16-bit reload timer reload register lower (TMRLRL0)".
423	19.2 Configuration of 16-bit Reload Timer ■ Block Diagram of 16-bit Reload Timer	Renamed the section "● 16-bit timer register (TMRH0, TMRL0)" to "● 16-bit reload timer timer register (TMRH0, TMRL0)".
		Renamed the section "● 16-bit reload register (TMRLRH0, TMRLRL0)" to "● 16-bit reload timer reload register (TMRLRH0, TMRLRL0)".
		Renamed the section "● Timer control status register (TMCSRH0, TMCSRL0)" to "● 16-bit reload timer control status register (TMCSRH0, TMCSRL0)".
425	19.4 Pins of 16-bit Reload Timer ■ Pins of 16-bit Reload Timer ● TO0 pin	Corrected the following bit number. DDRE:bit5 → DDR1:bit0
429	19.5.1 16-bit Reload Timer Control Status Register Upper (TMCSRH0) ■ 16-bit Reload Timer Control Status Register Upper (TMCSRH0)	Revised Figure 19.5-2.
444	19.8 Notes on Using 16-bit Reload Timer ■ Notes on Using 16-bit Reload Timer ● Note on the event counter operating in event counter operation mode	Deleted the section "● Precaution when Event Counter operates in event counter mode".
		Added the section "● Note on the event counter operating in event counter operation mode".
451	CHAPTER 20 EVENT COUNTER 20.2 Configuration of Event Counter ■ Block Diagram of Event Counter ● Composite timer count clock (CK06/CK16) selection circuit	Corrected the setting of the PCS[1:0] bits in "2.". 00 → 01, 10 or 11
455	20.4 Operation of Event Counter Operation Mode	Corrected the name of the operation mode of the event counter. event counter mode → event counter operation mode
	20.4 Operation of Event Counter Operation Mode ■ Operation of Event Counter Operation Mode	Corrected the following timer name. timer 01 → timer 11

Page	Revisions (For details, see their respective pages.)	
456	20.4 Operation of Event Counter Operation Mode ■ Operation of Event Counter Operation Mode Figure 20.4-2	Corrected the following timer name. timer 01 → timer 11
		Corrected the following interrupt name. Timer 01 compare match interrupt → Timer 11 compare match interrupt
		Corrected the following timer name. timer 01 → timer 11
457	20.5 Setting Procedure Example ■ Setting Procedure Example ● Initial settings	Corrected the following bit names in 12). C2, C1 → C2 to C0
		Corrected the following timer name in 13). timer 01 → timer 11
	20.5 Setting Procedure Example ■ Setting Procedure Example ● Interrupt process of composite timer (timer 11)	Corrected the following timer name. timer 01 → timer 11
459	20.7 Notes on Using Event Counter ■ Notes on Using Event Counter	Corrected the following register names. T00CR1/T01CR1 → T10CR1/T11CR1
463	CHAPTER 21 8/16-BIT PPG 21.2 Configuration of 8/16-bit PPG ■ Block Diagram of 8/16-bit PPG Figure 21.2-1	Added a remark on F_{CH}.
468	21.5 Registers of 8/16-bit PPG ■ Registers of 8/16-bit PPG Figure 21.5-1	Corrected the R/W attribute of bit7 in the PPGS register. R0/WX → R/W
		Corrected the R/W attribute of bit6 in the PPGS register. R0/WX → R/W
		Corrected the R/W attribute of bit5 in the PPGS register. R0/WX → R/W
		Corrected the R/W attribute of bit4 in the PPGS register. R0/WX → R/W
		Corrected the R/W attribute of bit7 in the REVC register. R0/WX → R/W
		Corrected the R/W attribute of bit6 in the REVC register. R0/WX → R/W
		Corrected the R/W attribute of bit5 in the REVC register. R0/WX → R/W
		Corrected the R/W attribute of bit4 in the REVC register. R0/WX → R/W
469	21.5.1 8/16-bit PPG Timer 01 Control Register (PC01) ■ 8/16-bit PPG Timer 01 Control Register (PC01) Figure 21.5-2	Added a remark on F_{CH}.
470	21.5.1 8/16-bit PPG Timer 01 Control Register (PC01) ■ 8/16-bit PPG Timer 01 Control Register (PC01) Table 21.5-1	Added a remark on F_{CH}.
		Corrected the following statement. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from either main clock or main CR clock. → Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock.

Page	Revisions (For details, see their respective pages.)	
471	21.5.2 8/16-bit PPG Timer 00 Control Register (PC00) ■ 8/16-bit PPG Timer 00 Control Register (PC00) Figure 21.5-3	Added a remark on F_{CH}.
472	21.5.2 8/16-bit PPG Timer 00 Control Register (PC00) ■ 8/16-bit PPG Timer 00 Control Register (PC00) Table 21.5-2	Added a remark on F_{CH}.
		Corrected the following statement. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from either main clock or main CR clock. → Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock.
477	21.6 Interrupts of 8/16-bit PPG ■ Registers and Vector Table Addresses Related to Interrupts of 8/16-bit PPG Table 21.6-2	Corrected the name of the UART/SIO channel in the remark. UART/SIO ch. 1 (lower) → UART/SIO ch. 1
495	CHAPTER 22 UART/SIO 22.4 Pins of UART/SIO ■ Block Diagrams of Pins of UART/SIO	Revised Figure 22.4-1.
519	22.7 Operations of UART/SIO and Setting Procedure Example ■ Operating Description of UART/SIO Operation Mode 1 ● Reception in UART/SIO operation mode 1	Revised Figure 22.7-12.
		Added a title "Figure 22.7-13 Overrun Error" to the figure below the section "Overrun error (OVE)".
521	22.7 Operations of UART/SIO and Setting Procedure Example ■ Operating Description of UART/SIO Operation Mode 1 ● Transmission in UART/SIO operation mode 1	Revised Figure 22.7-15.
536	CHAPTER 24 I^{2}C 24.1 Overview of I^{2}C	Deleted the following statement from the section summary. The I^{2}C interface supports the I^{2}C bus specification published by Philips.
544	24.5 Registers of I^{2}C ■ Registers of I^{2}C Figure 24.5-1	Corrected the R/W attribute of bit5 in the IBSR0 register. R/WX→ R0/WX
560	24.7 Operations of I^{2}C and Setting Procedure Example ■ Operations of I^{2}C ● I^{2}C interface	Deleted the following statement. It conforms to the I^{2}C bus specification defined by Philips.
561	24.7.1 I^{2}C Interface	Deleted the following statement from the section summary. It conforms to the I^{2}C bus specification defined by Philips.

Page	Revisions (For details, see their respective pages.)	
575	24.9 Sample Settings for I^{2}C ■ Sample Settings ● Enabling, disabling, and clearing interrupts	Corrected the name of the INTE bit. interrupt request enable bit → transfer completion interrupt enable bit
		Corrected the name of the INT bit. interrupt request flag bit → transfer completion interrupt request flag bit
		Corrected the name of the BEIE bit. interrupt request enable bit → bus error interrupt request enable bit
		Corrected the name of the BER bit. interrupt request flag bit → bus error interrupt request flag bit
		Corrected the name of the SPE bit. interrupt request enable bit → STOP detection interrupt enable bit
		Corrected the name of the SPF bit. interrupt request flag bit → STOP detection interrupt request flag bit
576	24.9 Sample Settings for I^{2}C ■ Sample Settings ● Enabling, disabling, and clearing interrupts	Corrected the name of the ALE bit. interrupt request enable bit → arbitration lost interrupt enable bit
		Corrected the name of the ALF bit. interrupt request flag bit → arbitration lost interrupt request flag bit
		Corrected the name of the WUE bit. interrupt request enable bit → MCU standby-mode wakeup function enable bit
		Corrected the name of the WUF bit. interrupt request flag bit → MCU standby-mode wakeup interrupt request flag bit
594	CHAPTER 25 8/10-BIT A/D CONVERTER 25.7 Notes on Using 8/10-bit A/D Converter ■ Notes on Using 8/10-bit A/D Converter ● Notes on setting the 8/10-bit A/D converter with a program	Corrected the following statement. The start of the reset mode, the stop mode or the watch mode causes the A/D converter to stop and the ADMV bit to be cleared to "0". → A reset, or the start of the stop mode or watch mode causes the A/D converter to stop and the ADMV bit to be cleared to "0".

Page	Revisions (For details, see their respective pages.)	
606	CHAPTER 27 CLOCK SUPERVISOR COUNTER 27.1 Overview of Clock Supervisor Counter	Corrected the following statement. The count clock of this module can be selected from the main oscillation clock and the sub-oscillation clock. → The count clock of this module can be selected from the main oscillation clock, the main PLL clock and the suboscillation clock.
607	27.2 Configuration of Clock Supervisor Counter ■ Block Diagram of Clock Supervisor Counter	Revised Figure 27.2-1.
608	27.2 Configuration of Clock Supervisor Counter ■ Block Diagram of Clock Supervisor Counter ● Counter source clock selector	Corrected the following statement. This block is used to select the counter source clock from the main oscillation clock and the sub-oscillation clock. → This block is used to select the counter source clock from the main oscillation clock, the main PLL clock and the suboscillation clock.
611	27.3.2 Clock Monitoring Control Register (CMCR) ■ Clock Monitoring Control Register (CMCR)	Revised Figure 27.3-3.
612	27.3.2 Clock Monitoring Control Register (CMCR) ■ Clock Monitoring Control Register (CMCR) Table 27.3-2	Revised details of the function of the CMCSEL bit.
613	27.4 Operations of Clock Supervisor Counter ■ Clock Supervisor Counter ● Clock Supervisor Counter Operation 1	Corrected the following statement. The count clock of this module can be selected from the main oscillation clock and the sub-oscillation clock. → The count clock of this module can be selected from the main oscillation clock, the main PLL clock and the suboscillation clock.
624	CHAPTER 28 LCD CONTROLLER (MB95410H SERIES) 28.2 Configuration of LCD Controller ■ LCD Controller Block Diagrams	Added the section "● LCDC blinking setting register 1 (LCDCB1), LCDC blinking setting register 2 (LCDCB2)".
626	28.2.1 Internal Divider Resistors for LCD Controller ■ Internal Divider Resistors	Corrected the following statement. To use only the internal divider resistors without connecting the external divider when using internal split resistors → To use only the internal divider resistors without any external divider resistor
628	28.2.1 Internal Divider Resistors for LCD Controller ■ Use of Internal Divider Resistors and Brightness Control	Corrected the following statement. Figure 28.2-5 shows an example of connecting a VR to internal divider resistors for brightness control. → Figure 28.2-5 illustrates connecting a VR to the V4 pin to control brightness.
		Revised Figure 28.2-4.

Page	Revisions (For details, see their respective pages.)	
629	28.2.1 Internal Divider Resistors for LCD Controller ■ Use of Internal Divider Resistors and Brightness Control Figure 28.2-5	Revised the figure title from "Brightness Control with Internal Divider Resistors Used" to "Brightness Control by Connecting VR to V4 Pin".
630	28.2.2 External Divider Resistors for LCD Controller ■ External Divider Resistors	Corrected the following cross-reference. Figure 28.2-1 → Table 28.2-1
		Revised Figure 28.2-6.
631	28.2.2 External Divider Resistors for LCD Controller ■ Use of External Divider Resistors	Revised Figure 28.2-7.
632	28.3 Pins of LCD Controller ■ Pins of LCD Controller ● COM0 to COM7 pins	Deleted the following statement. In addition, COM0 to COM3 can also function as general-purpose I/O ports.
634	28.3 Pins of LCD Controller ■ Block Diagrams of Pins of LCD Controller	Corrected Figure 28.3-4.
635		Corrected Figure 28.3-5.
		Corrected Figure 28.3-6.
638	28.4.1 LCDC Control Register 1 (LCDCC1) ■ LCDC Control Register 1 (LCDCC1) Figure 28.4-2	Revised details of the FP1 and FP0 bits.
639	28.4.1 LCDC Control Register 1 (LCDCC1) ■ LCDC Control Register 1 (LCDCC1) Table 28.4-1	Revised details of the function of the LCDEN bit.
		Revised details of the function of the VSEL bit.
642	28.4.3 LCDC Enable Register 1 (LCDCE1) ■ LCDC Enable Register 1 (LCDCE1)	Revised Figure 28.4-4.
643	28.4.3 LCDC Enable Register 1 (LCDCE1) ■ LCDC Enable Register 1 (LCDCE1) Table 28.4-3	Revised details of the function of the VE4 bit.
		Revised details of the function of the VE3 bit.
		Revised details of the function of the VE2 bit.
		Revised details of the function of the VE1 bit.
		Revised details of the function of the VE0 bit.
	28.4.3 LCDC Enable Register 1 (LCDCE1) ■ LCDC Enable Register 1 (LCDCE1)	Revised details of "Note".
649	28.4.7 LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2) ■ LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2) Figure 28.4-8.	Corrected the range of numbers represented by "m". 0 to 8 → 0 to 7
651	28.5 LCD Controller Display RAM ■ Display RAM and Output Pins	Corrected the address of "n" in "Note". $0FCD_H$ → $0FBD_H$

Page	Revisions (For details, see their respective pages.)	
654	28.7 Operations of LCD Controller ■ Operations of LCD Controller Figure 28.7-2	Corrected the value to be set to the PICTL bit. 0 → 1
656	28.7 Operations of LCD Controller ■ Operations of LCD Controller Figure 28.7-4	Corrected the value to be set to the PICTL bit. 0 → 1
672	CHAPTER 29 LCD CONTROLLER (MB95470H SERIES) 29.2 Configuration of LCD Controller ■ LCD Controller Block Diagrams	Added the section "● LCDC blinking setting register 1 (LCDCB1), LCDC blinking setting register 2 (LCDCB2)".
674	29.2.1 Internal Divider Resistors for LCD Controller ■ Internal Divider Resistors	Corrected the following content. To use only the internal divider resistors without connecting the external divider when using internal split resistors → To use only the internal divider resistors without any external divider resistor
676	29.2.1 Internal Divider Resistors for LCD Controller ■ Use of Internal Divider Resistors and Brightness Control	Corrected the following content. Figure 29.2-5 shows an example of connecting a VR to internal divider resistors for brightness control. → Figure 29.2-5 illustrates connecting a VR to the V4 pin to control brightness.
		Revised Figure 29.2-4.
677	29.2.1 Internal Divider Resistors for LCD Controller ■ Use of Internal Divider Resistors and Brightness Control Figure 29.2-5	Revised the figure title from "Brightness Control with Internal Divider Resistors Used" to "Brightness Control by Connecting VR to V4 Pin".
678	29.2.2 External Divider Resistors for LCD Controller ■ External Divider Resistors	Corrected a cross-reference. Figure 29.2-1 → Table 29.2-1
		Revised Figure 29.2-6.
	29.2.2 External Divider Resistors for LCD Controller ■ External Divider Resistors Table 29.2-1	Added the legend of "X".
679	29.2.2 External Divider Resistors for LCD Controller ■ Use of External Divider Resistors	Revised Figure 29.2-7.
680	29.3 Pins of LCD Controller ■ Pins of LCD Controller ● COM0 to COM7 pins	Corrected a register abbreviation. LCDCE7 → LCDCE6
		Deleted the following statement. In addition, COM0 to COM3 can also function as general-purpose I/O ports.
683	29.3 Pins of LCD Controller ■ Block Diagrams of Pins of LCD Controller	Corrected Figure 29.3-5.
		Corrected Figure 29.3-6.
684		Corrected Figure 29.3-7.
685		Corrected Figure 29.3-8.
688	29.4.1 LCDC Control Register 1 (LCDCC1) ■ LCDC Control Register 1 (LCDCC1) Figure 29.4-2.	Revised details of the FP1 and FP0 bits.

Page	Revisions (For details, see their respective pages.)	
689	29.4.1 LCDC Control Register 1 (LCDCC1) ■ LCDC Control Register 1 (LCDCC1) Table 29.4-1	Revised details of the function of the LCDEN bit.
		Revised details of the function of the VSEL bit.
692	29.4.3 LCDC Enable Register 1 (LCDCE1) ■ LCDC Enable Register 1 (LCDCE1)	Revised Figure 29.4-4.
693	29.4.3 LCDC Enable Register 1 (LCDCE1) ■ LCDC Enable Register 1 (LCDCE1) Table 29.4-3	Revised details of the function of the VE4 bit.
		Revised details of the function of the VE3 bit.
		Revised details of the function of the VE2 bit.
		Revised details of the function of the VE1 bit.
	29.4.3 LCDC Enable Register 1 (LCDCE1) ■ LCDC Enable Register 1 (LCDCE1)	Revised details of "Note".
696	29.4.5 LCDC Enable Register 3 to LCDC Enable Register 5 (LCDCE3 to LCDCE5)	Corrected the segment output pin name from "SEG31" to "SEG23" in the section summary.
697	29.4.6 LCDC Enable Register 6 (LCDCE6)	Corrected the segment output pin names from "SEG32 to SEG39" to "SEG24 to SEG31" in the section summary.
699	29.4.7 LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2) ■ LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2)	Revised Figure 29.4-8.
701	29.5 LCD Controller Display RAM ■ Display RAM and Output Pins	Corrected the address in "Note". 0FCD$_H$ → 0FBD$_H$
702	29.6 Interrupts of LCD Controller ■ Register and Vector Table Addresses Related to LCD Controller Interrupts Figure 29.6-1	Corrected the legend. V0 to V4: Voltages of V0 to V4 pins → V1 to V4: Voltages of V1 to V4 pins
704	29.7 Operations of LCD Controller ■ Operations of LCD Controller Figure 29.7-2	Corrected the value to be set to the PICTL bit. 0 → 1
706	29.7 Operations of LCD Controller ■ Operations of LCD Controller Figure 29.7-4	Corrected the value to be set to the PICTL bit. 0 → 1
720	CHAPTER 30 DUAL OPERATION FLASH MEMORY 30.1 Overview of Dual Operation Flash Memory	Deleted the following statement from the section summary. The Flash memory interface circuit enables read access and write access from the CPU to the Flash memory.
723	30.3 Registers of Flash Memory ■ Registers of Flash Memory Figure 30.3-1	Corrected the initial value of the FSR3 register. X0000000$_B$ → 00000000$_B$

Page	Revisions (For details, see their respective pages.)	
729	30.3.2 Flash Memory Status Register (FSR) ■ Flash Memory Status Register (FSR)	Revised Figure 30.3-4.
734	30.3.4 Flash Memory Status Register 3 (FSR3) ■ Flash Memory Status Register 3 (FSR3) Figure 30.3-7	Corrected the initial value. $X0000000_B$ → 00000000_B
751	30.6.1 Placing Flash Memory in the Read/Reset State ■ Placing Flash Memory in the Read/Reset State	Deleted the following statement. As is the case with masked ROM, program access from the CPU can be made.
767	CHAPTER 31 EXAMPLE OF SERIAL PROGRAMMING CONNECTION 31.2 Example of Serial Programming Connection ■ Example of Serial Programming Connection	Added a statement related to the use of the pull-up resistor.
776	CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION 32.3.3 Watchdog Timer Selection ID Registers (WDTH, WDTL) ■ Watchdog Timer Selection ID Registers (WDTH, WDTL) Figure 32.3-4	Revised details of the functions of the WDTH and WDTL registers.
782	CHAPTER 33 VOLTAGE COMPARATOR 33.1 Overview of Voltage Comparator	Corrected the following content in the section summary. The voltage comparator is used to monitor the voltages of two analog inputs, which can be either one internal output and one external input or two external inputs, → The voltage comparator is used to monitor the voltages of two analog inputs, which can be either one internal input and one external input or two external inputs,
783	33.2 Configuration of Voltage Comparator ■ Block Diagram of Voltage Comparator	Revised Figure 33.2-1.
787	33.4 Register of Voltage Comparator ■ Register of Voltage Comparator Figure 33.4-1	Corrected the abbreviation of the voltage comparator control register. CMR → CMR0
790	33.5 Interrupts of Voltage Comparator ■ Output Edge Detection Interrupt	Corrected the name of the IF bit in "Note". interrupt flag bit → output edge detection interrupt flag bit
791	33.6 Operations of Voltage Comparator ■ Software Activation of Voltage Comparator	Revised the statement below Figure 33.6-1. After the voltage comparator is activated as shown above, it has to stabilize before starting to operate. → After the voltage comparator is activated as shown above, it has to wait for the stabilization time to elapse before starting to operate. For details of the stabilization wait time, refer to the data sheet of the MB95410H/470H Series.

Page	Revisions (For details, see their respective pages.)	
794	CHAPTER 34 SYSTEM CONFIGURATION CONTROLLER 34.1 Overview of System Configuration Register (SYSC) ■ Functions of SYSC	Corrected the following statement. Selection of the port/reset function for the PF2/$\overline{\text{RST}}$ pin → Selection of the general-purpose I/O port/reset function for the PF2/$\overline{\text{RST}}$ pin
		Corrected the following statement. Selection of the port/reset function for the PG1/X0A pin and that for the PG2/XIA pin → Selection of the general-purpose I/O port/reset function for the PG1/X0A pin and that for the PG2/XIA pin
		Corrected the following statement. Selection of the port/reset function for the PF0/X0 pin and that for the PF1/XI pin → Selection of the general-purpose I/O port/reset function for the PF0/X0 pin and that for the PF1/XI pin
796	34.2 System Configuration Register (SYSC) ■ System Configuration Register (SYSC) Table 34.2-1	Revised details of the function of the VBGRSELX bit.
797	34.3 Notes on Using Controller ■ Notes on Using Controller ● Selecting the reference voltage for the voltage comparator	Corrected the "P21 pin" to the "CMPP pin".

Page	Revisions (For details, see their respective pages.)	
800	APPENDIX APPENDIX A I/O Map ■ I/O Map Table A-1	Corrected the initial value of the RSRR register. XXXXXXXX$_{\text{B}}$ → 000XXXXX$_{\text{B}}$
802		Corrected the initial value of the FSR3 register. X0000000$_{\text{B}}$ → 00000000$_{\text{B}}$
804		Corrected the register name of the BRSR0 register. UART/SIO dedicated baud rate generator setting register ch. 0 → UART/SIO dedicated baud rate generator baud rate setting register ch. 0
		Corrected the register name of the BRSR1 register. UART/SIO dedicated baud rate generator setting register ch. 1 → UART/SIO dedicated baud rate generator baud rate setting register ch. 1
		Corrected the register name of the BRSR2 register. UART/SIO dedicated baud rate generator setting register ch. 2 → UART/SIO dedicated baud rate generator baud rate setting register ch. 2
		Corrected the initial value of the EVCR register. XXXXXXX0$_{\text{B}}$ → 00000000$_{\text{B}}$
		Corrected the register name of the SYSC register. System control register → System configuration register
805		Corrected the register name of the WICR register. Interrupt pin control register → Interrupt pin selection circuit control register

Page	Revisions (For details, see their respective pages.)	
806	APPENDIX A I/O Map ■ I/O Map Table A-2	Corrected the initial value of the RSRR register. $XXXXXXXX_B$ → $000XXXXX_B$
808		Corrected the initial value of the FSR3 register. $X0000000_B$ → 00000000_B
809		Corrected the register name of the BRSR0 register. UART/SIO dedicated baud rate generator setting register ch. 0 → UART/SIO dedicated baud rate generator baud rate setting register ch. 0
		Corrected the register name of the BRSR1 register. UART/SIO dedicated baud rate generator setting register ch. 1 → UART/SIO dedicated baud rate generator baud rate setting register ch. 1
		Corrected the register name of the BRSR2 register. UART/SIO dedicated baud rate generator setting register ch. 2 → UART/SIO dedicated baud rate generator baud rate setting register ch. 2
810		Corrected the register abbreviation of the LCDC control register 1. LCDCC → LCDCC1
		Corrected the initial value of the LCDCC1 register. 00010000_B → 00000000_B
		Corrected the initial value of the EVCR register. $XXXXXXX0_B$ → 00000000_B
		Corrected the register name of the SYSC register. System control register → System configuration register
		Corrected the register name of the WICR register. Interrupt pin control register → Interrupt pin selection circuit control register
812	APPENDIX B Table of Interrupt Sources ■ Table of Interrupt Sources Table B-1	Corrected the interrupt source for IRQ04.

Page	Revisions (For details, see their respective pages.)	
815	APPENDIX D Pin States of MB95410H/470H Series ■ Pin States in Each Mode Table D-1	Deleted "TO01" from the P11/UO0/TO01 pin.
		Added the states of the P17/CMPO pin.
		Added remark *11 to details of the states of the P22/SCL pin and the P23/SDA pin.
816		Revised the remark number from *11 to *12 for the following pins: P40/SEG21, P41/SEG20, P42/SEG19, P43/SEG18, P50/TO01, P51/EC0, P53/TO0.
		Added remark *12 to details of the states of the P52/TI0/TO00 pin.
817		Revised the remark number from *11 to *12 for the following pins: PB2/SEG37, PB3/SEG38, PB4/SEG39, PC4/SEG06, PC5/SEG07, PC6/SEG08, PC7/SEG09
818		Corrected the pin name "TO01" to "TO00" in *2.
		Corrected the pin name "PG0/X0A" to "PG1/X0A", and "PG1/ X1A" to "PG2/X1A" in *5.
		Corrected the pin name "P90" to "P94/V0" in *7.
		Added remark *11.
		Revised the remark number from *11 to *12.
		Revised details of remark *12.

CHAPTER 1

OVERVIEW

This chapter describes the features and basic specifications of the MB95410H/470H Series.

1.1 Features of MB95410H/470H Series

In addition to a compact instruction set, the MB95410H/470H Series is a series of general-purpose single-chip microcontrollers with a variety of peripheral functions.

■ Features of MB95410H/470H Series

● F^{2}MC-8FX CPU core

Instruction set optimized for controllers

- Multiplication and division instructions
- 16-bit arithmetic operations
- Bit test branch instructions
- Bit manipulation instructions, etc.

● Clock

- Selectable main clock source
  - Main oscillation clock (Up to 16.25 MHz, maximum machine clock frequency is 8.125 MHz)
  - External clock (Up to 32.5 MHz, maximum machine clock frequency is 16.25 MHz)
  - Main CR clock (1/8/10/12.5 MHz ±2%, maximum machine clock frequency is 12.5 MHz)
  - Main PLL clock (up to 16.25 MHz, maximum machine clock frequency: 16.25 MHz)
- Selectable subclock source
  - Suboscillation clock (32.768 kHz)
  - External clock (32.768 kHz)
  - Sub-CR clock (Typ: 100 kHz, Min: 50 kHz, Max: 200 kHz)

● Timer

- 8/16-bit composite timer × 2 channels
- 8/16-bit PPG × 2 channels
- 16-bit reload timer × 1 channel
- Event counter × 1 channel
- Time-base timer × 1 channel
- Watch prescaler × 1 channel

● UART/SIO

- Capable of clock asynchronous (UART) and clock synchronous (SIO) serial data transfer
- Full duplex double buffer

● I^{2}C

Built-in wake-up function

● External interrupt

- Interrupt by the edge detection (rising edge, falling edge, and both edges can be selected)
- Can be used to wake up the device from different low-power consumption modes (also called standby modes)

● 8/10-bit A/D converter

8-bit or 10-bit resolution can be selected

● LCD controller (LCDC)

- On MB95F414H/F414K/F416H/F416K/F418H/F418K, LCD output can be selected from 40 SEG × 4 COM and 36 SEG × 8 COM.
- On MB95F474H/F474K/F476H/F476K/F478H/F478K, LCD output can be selected from 32 SEG × 4 COM and 28 SEG × 8 COM.
- Internal divider resistor whose resistance value can be selected from 10 kΩ or 100 kΩ through software
- Interrupt event in sync with the LCD module frame frequency
- Blinking function
- Inverted display function

● Low power consumption (standby) modes

- Stop mode
- Sleep mode
- Watch mode
- Time-base timer mode

● I/O port

- MB95F414H/F416H/F418H (no. of I/O ports: 74)
  - General-purpose I/O ports (CMOS I/O) : 71
  - General-purpose I/O ports (N-ch open drain) : 3
- MB95F414K/F416K/F418K (no. of I/O ports: 75)
  - General-purpose I/O ports (CMOS I/O) : 71
  - General-purpose I/O ports (N-ch open drain) : 4
- MB95F474H/F476H/F478H (no. of I/O ports: 58)
  - General-purpose I/O ports (CMOS I/O) : 55
  - General-purpose I/O ports (N-ch open drain) : 3
- MB95F474K/F476K/F478K (no. of I/O ports: 59)
  - General-purpose I/O ports (CMOS I/O) : 55
  - General-purpose I/O ports (N-ch open drain) : 4

● On-chip debug

- 1-wire serial control
- Serial writing supported (asynchronous mode)

- Hardware/software watchdog timer

  - Built-in hardware watchdog timer
  - Built-in software watchdog timer

- Power-on reset

  A power-on reset is generated when the power is switched on.

- Low-voltage detection reset circuit (only available on MB95F414K/F416K/F418K/F474K/F476K/F478K)

  Built-in low-voltage detector

- Clock supervisor counter

  Built-in clock supervisor counter function

- Programmable port input voltage level

  CMOS input level / hysteresis input level

- Dual operation Flash memory

  The erase/write operation and the read operation can be executed in different banks (upper bank/lower bank) simultaneously.

- Flash memory security function

  Protects the content of the Flash memory

1.2 Product Line-up of MB95410H/470H Series

Table 1.2-1 and Table 1.2-2 list the product line-up of the MB95410H/470H Series.

■ Product Line-up of MB95410H/470H Series

Table 1.2-1 Product Line-up of MB95410H Series (1 / 3)

Part Number Parameter	MB95F414H	MB95F416H	MB95F418H	MB95F414K	MB95F416K	MB95F418K
Type	Flash memory product					
Clock supervisor counter	It supervises the main clock oscillation.					
Flash memory capacity	20 Kbyte	36 Kbyte	60 Kbyte	20 Kbyte	36 Kbyte	60 Kbyte
RAM capacity	496 bytes	1008 bytes	2032 bytes	496 bytes	1008 bytes	2032 bytes
Power-on reset	Yes					
Low-voltage detection reset	No			Yes		
Reset input	Dedicated			Selected through software		
CPU functions	• Number of basic instructions : 136 • Instruction bit length : 8 bits • Instruction length : 1 to 3 bytes • Data bit length : 1, 8, and 16 bits • Minimum instruction execution time : 61.5 ns (machine clock = 16.25 MHz) • Interrupt processing time : 0.6 µs (machine clock = 16.25 MHz)					
General-purpose I/O	• I/O ports : 74 • CMOS I/O : 71 • N-ch open drain : 3			• I/O ports : 75 • CMOS I/O : 71 • N-ch open drain : 4		
Time-base timer	Interval time: 0.256 ms to 8.3 s (external clock = 4 MHz)					
Hardware/ software watchdog timer	• Reset generation cycle - Main oscillation clock at 10 MHz: 105 ms (Min) • The sub-CR clock can be used as the source clock of hardware watchdog timer.					
Wild register	It can be used to replace three bytes of data.					
I²C	1 channel					
	• Master/Slave sending and receiving • Bus error function and arbitration function • Detecting transmitting direction function • Start condition repeated generation and detection functions • Built-in wake-up function					
UART/SIO	3 channels					
	• Data transfer with UART/SIO is enabled. • It has a full duplex double buffer, variable data length (5/6/7/8 bits), a built-in baud rate generator and an error detection function. • It uses the NRZ type transfer format. • LSB-first data transfer and MSB-first data transfer are available to use. • Both clock asynchronous (UART) serial data transfer and clock synchronous (SIO) serial data transfer are enabled.					

Table 1.2-1 Product Line-up of MB95410H Series (2 / 3)

Part Number Parameter	MB95F414H	MB95F416H	MB95F418H	MB95F414K	MB95F416K	MB95F418K
8/10-bit A/D converter	8 channels					
	8-bit or 10-bit resolution can be selected.					
8/16-bit composite timer	2 channels					
	• Each timer can be configured as an "8-bit timer × 2 channels" or a "16-bit timer × 1 channel". • It has built-in timer function, PWC function, PWM function and capture function. • Count clock: it can be selected from internal clocks (7 types) and external clocks. • It can output square wave.					
LCD controller (LCDC)	• COM output: 4 or 8 (selectable) • SEG output: 36 or 40 (selectable) - If the number of COM outputs is 4, the maximum number of SEG outputs is 40, and the maximum number of pixels that can be displayed 160 (4×40). - If the number of COM outputs is 8, the maximum number of SEG outputs is 36, and the maximum number of pixels that can be displayed 288 (8×36). • LCD drive power supply (bias) pins: 5 (Max)					
	• Duty LCD mode • LCD standby mode • Blinking function • Internal divider resistor whose resistance value can be selected from 10 kΩ or 100 kΩ through software • Interrupt event in sync with the LCD module frame frequency • Inverted display function					
16-bit reload timer	1 channel					
	• Two clock modes and two counter operating modes can be selected • Square waveform output • Count clock: 7 internal clocks and external clock can be selected • Counter operating mode: reload mode or one-shot mode can be selected					
Event counter	By configuring the 16-bit reload timer and the 8/16-bit composite timer ch. 1, event counter function can be implemented. When using the event counter function, the 16-bit reload timer and the 8/16-bit composite timer ch. 1 is unavailable					
8/16-bit PPG	2 channels					
	• Each channel of the PPG can be used as "8-bit PPG × 2 channels" or "16-bit PPG × 1 channel" • Counter operating clock: Eight selectable clock sources					
Watch counter	• Count clock: Four selectable clock sources (125 ms, 250 ms, 500 ms or 1 s) • Counter value can be set from 0 to 63. (Capable of counting for 1 minute when selecting clock source of 1 second and setting counter value to 60)					
External interrupt	8 channels					
	• Interrupt by edge detection (The rising edge, falling edge, or both edges can be selected.) • It can be used to wake up the device from standby mode.					
On-chip debug	• 1-wire serial control • It supports serial writing. (asynchronous mode)					
Watch prescaler	Eight different time intervals can be selected. (62.5 ms, 125 ms, 250 ms, 500 ms, 1 s, 2 s, 4 s, 8 s)					
Flash memory	• It supports automatic programming, Embedded Algorithm, write/erase/erase-suspend/erase-resume commands. • It has a flag indicating the completion of the operation of Embedded Algorithm. • Number of write/erase cycles: 100000 • Data retention time: 20 years • Flash security feature for protecting the content of the Flash.					
Standby mode	Sleep mode, stop mode, watch mode, time-base timer mode					

Table 1.2-1 Product Line-up of MB95410H Series (3 / 3)

Part Number / Parameter	MB95F414H	MB95F416H	MB95F418H	MB95F414K	MB95F416K	MB95F418K
Package	FPT-80P-M37					

Table 1.2-2 Product Line-up of MB95470H Series (1 / 2)

Part Number Parameter	MB95F474H	MB95F476H	MB95F478H	MB95F474K	MB95F476K	MB95F478K
Type	Flash memory product					
Clock supervisor counter	It supervises the main clock oscillation.					
Flash memory capacity	20 Kbyte	36 Kbyte	60 Kbyte	20 Kbyte	36 Kbyte	60 Kbyte
RAM capacity	496 bytes	1008 bytes	2032 bytes	496 bytes	1008 bytes	2032 bytes
Power-on reset	Yes					
Low-voltage detection reset	No			Yes		
Reset input	Dedicated			Selected through software		
CPU functions	• Number of basic instructions : 136 • Instruction bit length : 8 bits • Instruction length : 1 to 3 bytes • Data bit length : 1, 8, and 16 bits • Minimum instruction execution time : 61.5 ns (machine clock = 16.25 MHz) • Interrupt processing time : 0.6 µs (machine clock = 16.25 MHz)					
General-purpose I/O	• I/O ports : 58 • CMOS I/O : 55 • N-ch open drain : 3			• I/O ports : 59 • CMOS I/O : 55 • N-ch open drain : 4		
Time-base timer	Interval time: 0.256 ms to 8.3 s (external clock = 4 MHz)					
Hardware/ software watchdog timer	• Reset generation cycle - Main oscillation clock at 10 MHz: 105 ms (Min) • The sub-CR clock can be used as the source clock of hardware watchdog timer.					
Wild register	It can be used to replace three bytes of data.					
I²C	1 channel					
	• Master/Slave sending and receiving • Bus error function and arbitration function • Detecting transmitting direction function • Start condition repeated generation and detection functions • Built-in wake-up function					
UART/SIO	3 channels					
	• Data transfer with UART/SIO is enabled. • It has a full duplex double buffer, variable data length (5/6/7/8 bits), a built-in baud rate generator and an error detection function. • It uses the NRZ type transfer format. • LSB-first data transfer and MSB-first data transfer are available to use. • Both clock asynchronous (UART) serial data transfer and clock synchronous (SIO) serial data transfer are enabled.					
8/10-bit A/D converter	8 channels					
	8-bit or 10-bit resolution can be selected.					
8/16-bit composite timer	2 channels					
	• Each timer can be configured as an "8-bit timer × 2 channels" or a "16-bit timer × 1 channel". • It has built-in timer function, PWC function, PWM function and capture function. • Count clock: it can be selected from internal clocks (7 types) and external clocks. • It can output square wave.					

Table 1.2-2 Product Line-up of MB95470H Series (2 / 2)

Part Number Parameter	MB95F474H	MB95F476H	MB95F478H	MB95F474K	MB95F476K	MB95F478K
LCD controller (LCDC)	• COM output: 4 or 8 (selectable) • SEG output: 28 or 32 (selectable) - If the number of COM outputs is 4, the maximum number of SEG outputs is 32, and the maximum number of pixels that can be displayed 128 (4×32). - If the number of COM outputs is 8, the maximum number of SEG outputs is 28, and the maximum number of pixels that can be displayed 224 (8×28). • LCD drive power supply (bias) pins: 4 (Max)					
	• Duty LCD mode • LCD standby mode • Blinking function • Internal divider resistor whose resistance value can be selected from 10 kΩ or 100 kΩ through software • Inverted display function					
16-bit reload timer	1 channel					
	• Two clock modes and two counter operating modes can be selected • Square waveform output • Count clock: 7 internal clocks and external clock can be selected • Counter operating mode: reload mode or one-shot mode can be selected					
Event counter	By configuring the 16-bit reload timer and the 8/16-bit composite timer ch. 1, event counter function can be implemented. When using the event counter function, the 16-bit reload timer and the 8/16-bit composite timer ch. 1 is unavailable					
8/16-bit PPG	2 channels					
	• Each channel of the PPG can be used as "8-bit PPG × 2 channels" or "16-bit PPG × 1 channel" • Counter operating clock: Eight selectable clock sources					
Watch counter	• Count clock: Four selectable clock sources (125 ms, 250 ms, 500 ms or 1 s) • Counter value can be set from 0 to 63. (Capable of counting for 1 minute when selecting clock source of 1 second and setting counter value to 60)					
External interrupt	8 channels					
	Interrupt by edge detection (The rising edge, falling edge, or both edges can be selected.) It can be used to wake up the device from standby mode.					
On-chip debug	• 1-wire serial control • It supports serial writing. (asynchronous mode)					
Watch prescaler	Eight different time intervals can be selected. (62.5 ms, 125 ms, 250 ms, 500 ms, 1 s, 2 s, 4 s, 8 s)					
Flash memory	• It supports automatic programming, Embedded Algorithm, write/erase/erase-suspend/erase-resume commands. • It has a flag indicating the completion of the operation of Embedded Algorithm. • Number of write/erase cycles: 100000 • Data retention time: 20 years • Flash security feature for protecting the content of the Flash memory.					
Standby mode	Sleep mode, stop mode, watch mode, time-base timer mode					
Package	FPT-64P-M38 FPT-64P-M39					

1.3 Differences among Products and Notes on Product Selection

The following describes differences among the products of the MB95410H/470H Series and notes on product selection.

■ Differences among Products and Notes on Product Selection

- Current consumption

  When using the on-chip debug function, take account of the current consumption of flash erase/write.

  For details of current consumption, refer to "■ ELECTRICAL CHARACTERISTICS" in the data sheet of the MB95410H/470H Series.

- Package

  For details of information on each package, see "1.6 Package Dimension".

- Operating voltage

  The operating voltage varies, depending on whether the on-chip debug function is used or not. For details of the operating voltage, refer to "■ ELECTRICAL CHARACTERISTICS" in the data sheet of the MB95410H/470H Series.

- On-chip debug function

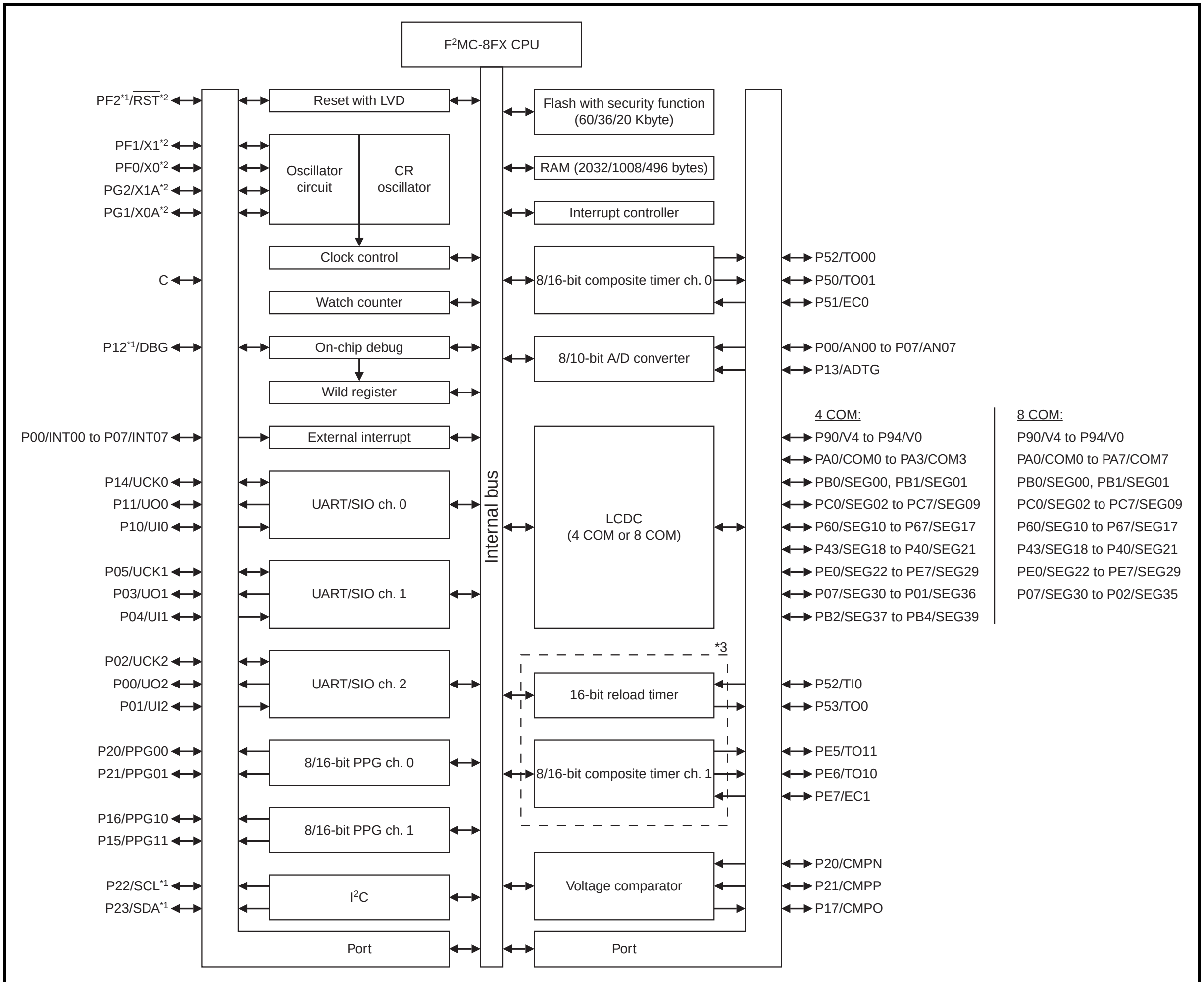
  The on-chip debug function requires that V_{CC}, V_{SS} and 1 serial-wire be connected to an evaluation tool. For details of the connection method, see "CHAPTER 31 EXAMPLE OF SERIAL PROGRAMMING CONNECTION".

1.4 Block Diagrams of MB95410H/470H Series

Figure 1.4-1 and Figure 1.4-2 are block diagrams of the MB95410H/470H Series.

■ Block Diagrams of MB95410H/470H Series

Figure 1.4-1 Block Diagram of MB95410H Series



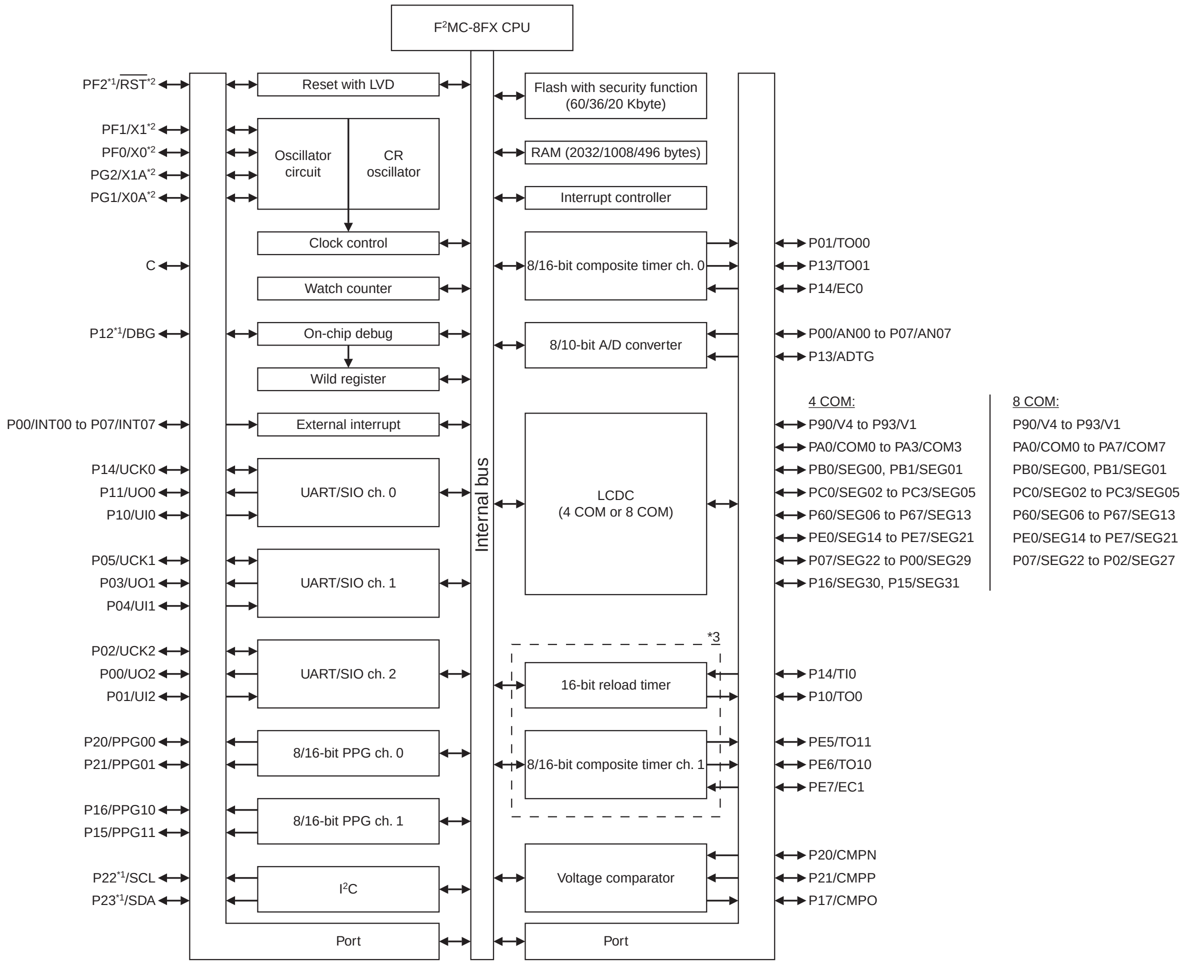
Vcc →
Vss →

*1: PF2, P12, P22 and P23 are N-ch open drain pins.

*2: Software option

*3: 8/16-bit composite timer ch. 1 and 16-bit reload timer can be used as an event counter when the event counter operating mode is enabled.

Figure 1.4-2 Block Diagram of MB95470H Series



*1: PF2, P12, P22 and P23 are N-ch open drain pins.

*2: Software option

*3: 8/16-bit composite timer ch. 1 and 16-bit reload timer can be used as an event counter when the event counter operating mode is enabled.

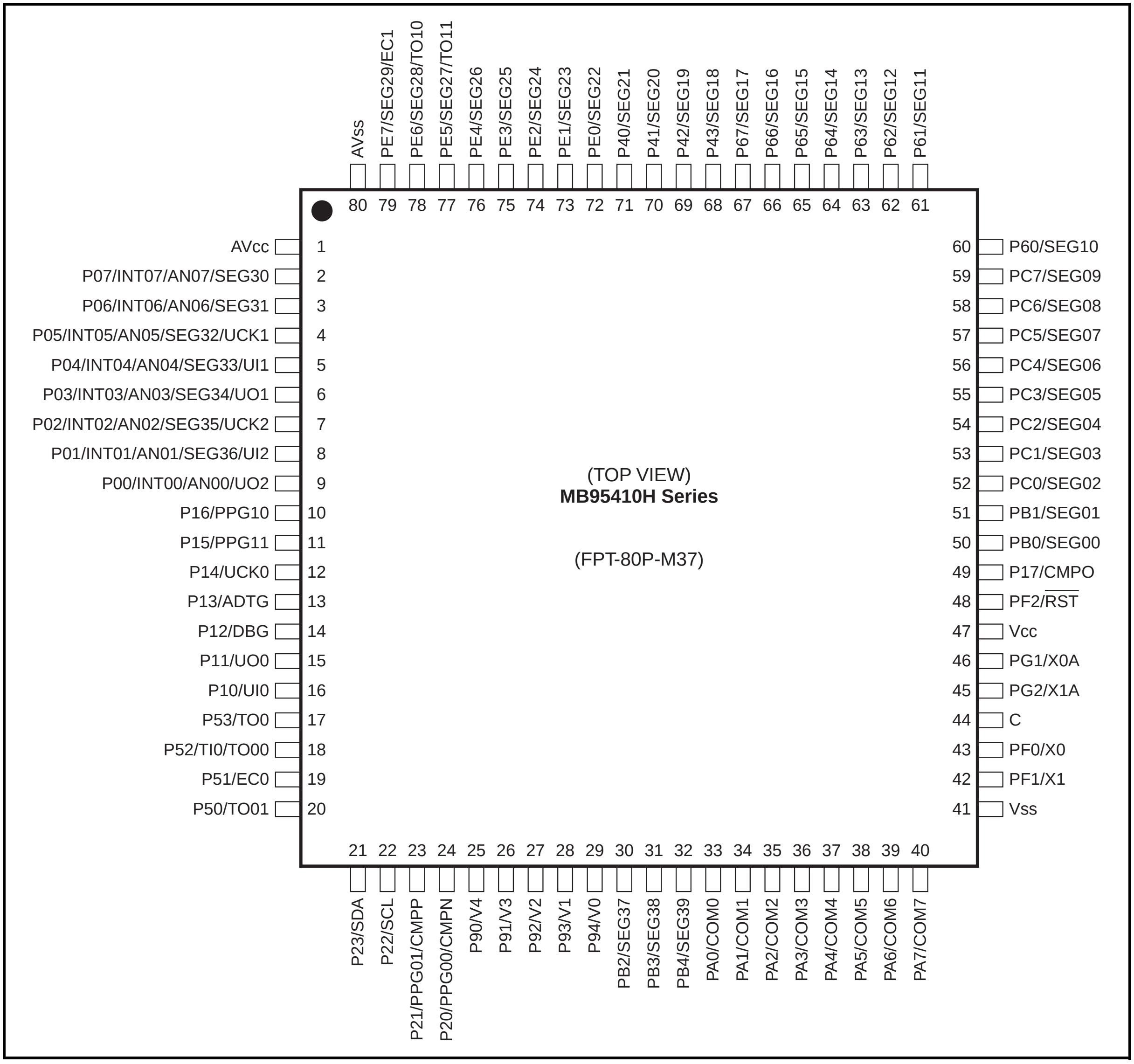
MB95410H/470H Series

1.5 Pin Assignment

Figure 1.5-1 and Figure 1.5-2 show the pin assignment of the MB95410H/470H Series.

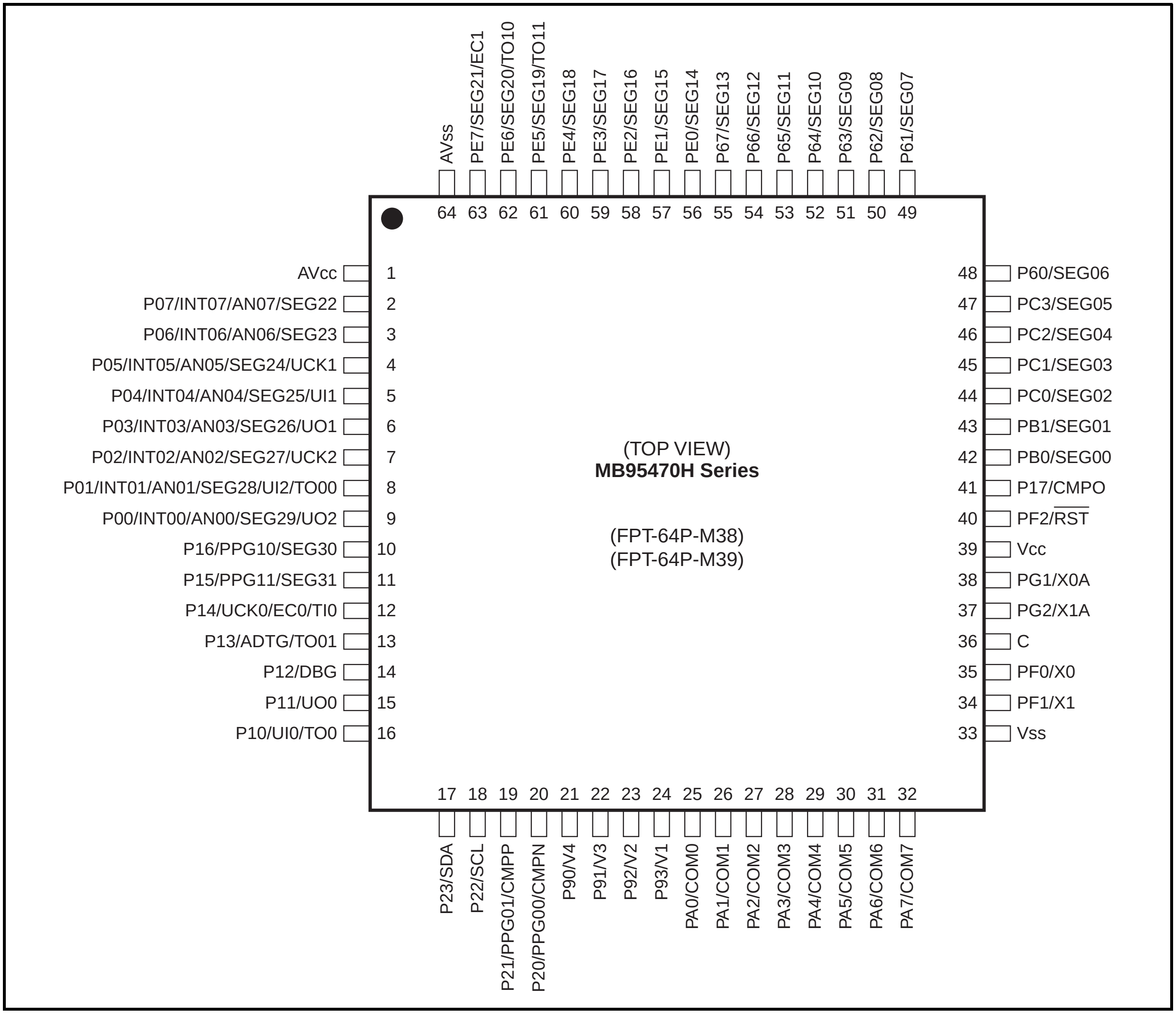
■ Pin Assignment of MB95410H Series

Figure 1.5-1 Pin Assignment of MB95410H Series



■ Pin Assignment of MB95470H Series

Figure 1.5-2 Pin Assignment of MB95470H Series



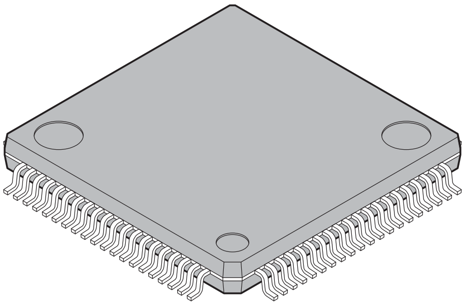
MB95410H/470H Series

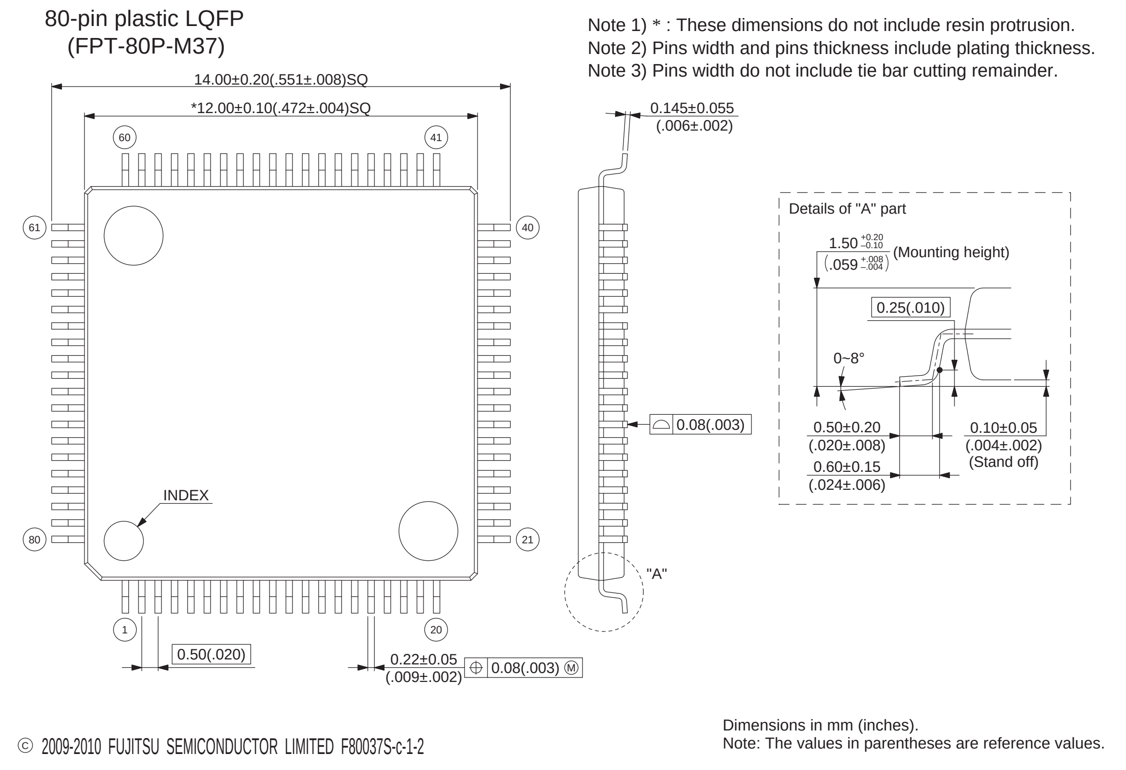
1.6 Package Dimension

The MB95410H/470H Series is available in three types of package.

■ Package Dimension of FPT-80P-M37 (MB95410H Series)

Figure 1.6-1 Package Dimension of FPT-80P-M37

80-pin plastic LQFP  (FPT-80P-M37)	Lead pitch	0.50 mm
	Package width × package length	12.00 mm × 12.00 mm
	Lead shape	Gullwing
	Lead bend direction	Normal bend
	Sealing method	Plastic mold
	Mounting height	1.70 mm MAX
	Weight	0.47 g



Note 1) * : These dimensions do not include resin protrusion.
Note 2) Pins width and pins thickness include plating thickness.
Note 3) Pins width do not include tie bar cutting remainder.

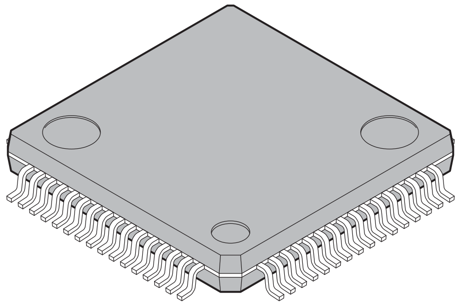
Dimensions in mm (inches).
Note: The values in parentheses are reference values.

Please check the latest package dimension at the following URL.

http://edevice.fujitsu.com/package/en-search/

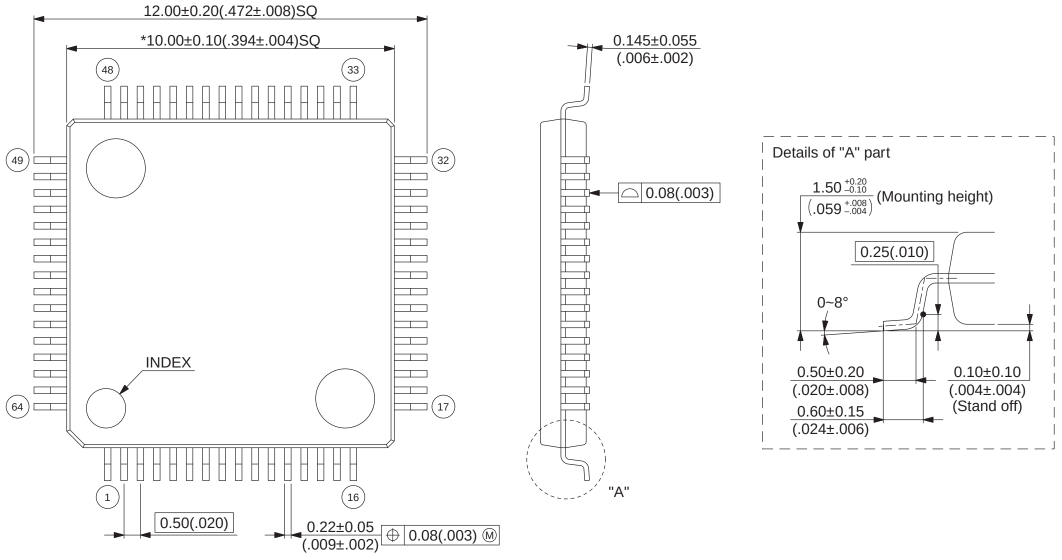
■ Package Dimension of FPT-64P-M38 (MB95470H Series)

Figure 1.6-2 Package Dimension of FPT-64P-M38

64-pin plastic LQFP  (FPT-64P-M38)	Lead pitch	0.50 mm
	Package width × package length	10.00 mm × 10.00 mm
	Lead shape	Gullwing
	Lead bend direction	Normal bend
	Sealing method	Plastic mold
	Mounting height	1.70 mm MAX
	Weight	0.32 g

64-pin plastic LQFP
(FPT-64P-M38)

Note 1) * : These dimensions do not include resin protrusion.
Note 2) Pins width and pins thickness include plating thickness.
Note 3) Pins width do not include tie bar cutting remainder.



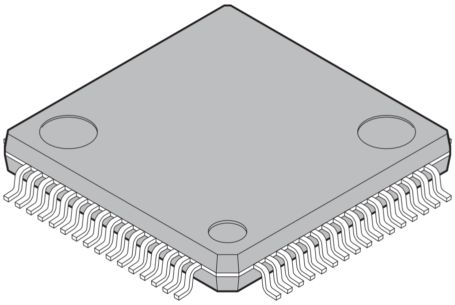
Dimensions in mm (inches).
Note: The values in parentheses are reference values.

Please check the latest package dimension at the following URL.

http://edevice.fujitsu.com/package/en-search/

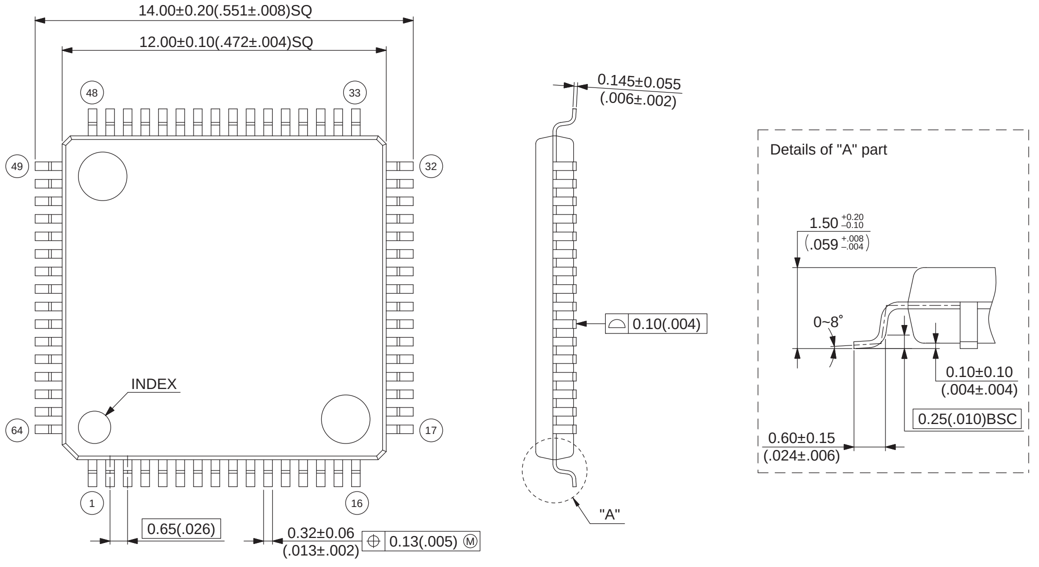
■ Package Dimension of FPT-64P-M39 (MB95470H Series)

Figure 1.6-3 Package Dimension of FPT-64P-M39

64-pin plastic LQFP  (FPT-64P-M39)	Lead pitch	0.65 mm
	Package width × package length	12.00 mm × 12.00 mm
	Lead shape	Gullwing
	Sealing method	Plastic mold
	Mounting height	1.70 mm MAX
	Weight	0.47 g

64-pin plastic LQFP
(FPT-64P-M39)

Note 1) Pins width and pins thickness include plating thickness.



Dimensions in mm (inches).
Note: The values in parentheses are reference values.

Please check the latest package dimension at the following URL.

http://edevice.fujitsu.com/package/en-search/

1.7 Pin Functions

Table 1.7-1 and Table 1.7-2 show pin functions of the MB95410H/470H Series. The alphabets in "I/O circuit type" column of the above tables correspond to those in "Type" column of Table 1.8-1.

■ Pin Functions (MB95410H Series)

Table 1.7-1 Pin Functions (MB95410H Series) (1 / 7)

Pin no.	Pin name	I/O circuit type*	Function
1	AV_{CC}	—	A/D converter power supply pin
2	P07	S	General-purpose I/O port
	INT07		External interrupt input pin
	AN07		A/D analog input pin
	SEG30		LCDC SEG output pin
3	P06	S	General-purpose I/O port
	INT06		External interrupt input pin
	AN06		A/D analog input pin
	SEG31		LCDC SEG output pin
4	P05	S	General-purpose I/O port
	INT05		External interrupt input pin
	AN05		A/D analog input pin
	SEG32		LCDC SEG output pin
	UCK1		UART/SIO ch. 1 clock I/O pin
5	P04	V	General-purpose I/O port
	INT04		External interrupt input pin
	AN04		A/D analog input pin
	SEG33		LCDC SEG output pin
	UI1		UART/SIO ch. 1 data input pin
6	P03	S	General-purpose I/O port
	INT03		External interrupt input pin
	AN03		A/D analog input pin
	SEG34		LCDC SEG output pin
	UO1		UART/SIO ch. 1 data output pin

Table 1.7-1 Pin Functions (MB95410H Series) (2 / 7)

Pin no.	Pin name	I/O circuit type*	Function
7	P02	S	General-purpose I/O port
	INT02		External interrupt input pin
	AN02		A/D analog input pin
	SEG35		LCDC SEG output pin
	UCK2		UART/SIO ch. 2 clock I/O pin
8	P01	V	General-purpose I/O port
	INT01		External interrupt input pin
	AN01		A/D analog input pin
	SEG36		LCDC SEG output pin
	UI2		UART/SIO ch. 2 data input pin
9	P00	W	General-purpose I/O port
	INT00		External interrupt input pin
	AN00		A/D analog input pin
	UO2		UART/SIO ch. 2 data output pin
10	P16	Y	General-purpose I/O port
	PPG10		8/16-bit PPG ch. 1 output pin
11	P15	Y	General-purpose I/O port
	PPG11		8/16-bit PPG ch. 1 output pin
12	P14	H	General-purpose I/O port
	UCK0		UART/SIO ch. 0 clock I/O pin
13	P13	H	General-purpose I/O port
	ADTG		A/D trigger input (ADTG) pin
14	P12	D	General-purpose I/O port
	DBG		DBG input pin
15	P11	H	General-purpose I/O port
	UO0		UART/SIO ch. 0 data output pin
16	P10	G	General-purpose I/O port
	UI0		UART/SIO ch. 0 data input pin
17	P53	H	General-purpose I/O port
	TO0		16-bit reload timer output pin

Table 1.7-1 Pin Functions (MB95410H Series) (3 / 7)

Pin no.	Pin name	I/O circuit type*	Function
18	P52	H	General-purpose I/O port
	TI0		16-bit reload timer input pin
	TO00		8/16-bit composite timer ch. 0 output pin
19	P51	H	General-purpose I/O port
	EC0		8/16-bit composite timer ch. 0 clock input pin
20	P50	H	General-purpose I/O port
	TO01		8/16-bit composite timer ch. 0 output pin
21	P23	I	General-purpose I/O port
	SDA		I^2C data I/O pin
22	P22	I	General-purpose I/O port
	SCL		I^2C clock I/O pin
23	P21	T	General-purpose I/O port
	PPG01		8/16-bit PPG ch. 0 output pin
	CMPP		Voltage comparator input pin
24	P20	T	General-purpose I/O port
	PPG00		8/16-bit PPG ch. 0 output pin
	CMPN		Voltage comparator input pin
25	P90	R	General-purpose I/O port
	V4		LCDC drive power supply pin
26	P91	R	General-purpose I/O port
	V3		LCDC drive power supply pin
27	P92	R	General-purpose I/O port
	V2		LCDC drive power supply pin
28	P93	R	General-purpose I/O port
	V1		LCDC drive power supply pin
29	P94	R	General-purpose I/O port
	V0		LCDC drive power supply pin
30	PB2	M	General-purpose I/O port
	SEG37		LCDC SEG output pin
31	PB3	M	General-purpose I/O port
	SEG38		LCDC SEG output pin

Table 1.7-1 Pin Functions (MB95410H Series) (4 / 7)

Pin no.	Pin name	I/O circuit type*	Function
32	PB4	M	General-purpose I/O port
	SEG39		LCDC SEG output pin
33	PA0	M	General-purpose I/O port
	COM0		LCDC COM output pin
34	PA1	M	General-purpose I/O port
	COM1		LCDC COM output pin
35	PA2	M	General-purpose I/O port
	COM2		LCDC COM output pin
36	PA3	M	General-purpose I/O port
	COM3		LCDC COM output pin
37	PA4	M	General-purpose I/O port
	COM4		LCDC COM output pin
38	PA5	M	General-purpose I/O port
	COM5		LCDC COM output pin
39	PA6	M	General-purpose I/O port
	COM6		LCDC COM output pin
40	PA7	M	General-purpose I/O port
	COM7		LCDC COM output pin
41	V_{SS}	—	Power supply pin (GND)
42	PF1	B	General-purpose I/O port
	X1		Main clock oscillation pin
43	PF0	B	General-purpose I/O port
	X0		Main clock oscillation pin
44	C	—	Capacitor connection pin
45	PG2	C	General-purpose I/O port
	X1A		Subclock oscillation pin (32 kHz)
46	PG1	C	General-purpose I/O port
	X0A		Subclock oscillation pin (32 kHz)
47	V_{CC}	—	Power supply pin
48	PF2	A	General-purpose I/O port
	$\overline{\text{RST}}$		Reset pin Dedicated reset pin for MB95F414H/F416H/F418H

Table 1.7-1 Pin Functions (MB95410H Series) (5 / 7)

Pin no.	Pin name	I/O circuit type*	Function
49	P17	H	General-purpose I/O port
	CMPO		Voltage comparator output pin
50	PB0	M	General-purpose I/O port
	SEG00		LCDC SEG output pin
51	PB1	M	General-purpose I/O port
	SEG01		LCDC SEG output pin
52	PC0	M	General-purpose I/O port
	SEG02		LCDC SEG output pin
53	PC1	M	General-purpose I/O port
	SEG03		LCDC SEG output pin
54	PC2	M	General-purpose I/O port
	SEG04		LCDC SEG output pin
55	PC3	M	General-purpose I/O port
	SEG05		LCDC SEG output pin
56	PC4	M	General-purpose I/O port
	SEG06		LCDC SEG output pin
57	PC5	M	General-purpose I/O port
	SEG07		LCDC SEG output pin
58	PC6	M	General-purpose I/O port
	SEG08		LCDC SEG output pin
59	PC7	M	General-purpose I/O port
	SEG09		LCDC SEG output pin
60	P60	M	General-purpose I/O port
	SEG10		LCDC SEG output pin
61	P61	M	General-purpose I/O port
	SEG11		LCDC SEG output pin
62	P62	M	General-purpose I/O port
	SEG12		LCDC SEG output pin
63	P63	M	General-purpose I/O port
	SEG13		LCDC SEG output pin

Table 1.7-1 Pin Functions (MB95410H Series) (6 / 7)

Pin no.	Pin name	I/O circuit type*	Function
64	P64	M	General-purpose I/O port
	SEG14		LCDC SEG output pin
65	P65	M	General-purpose I/O port
	SEG15		LCDC SEG output pin
66	P66	M	General-purpose I/O port
	SEG16		LCDC SEG output pin
67	P67	M	General-purpose I/O port
	SEG17		LCDC SEG output pin
68	P43	M	General-purpose I/O port
	SEG18		LCDC SEG output pin
69	P42	M	General-purpose I/O port
	SEG19		LCDC SEG output pin
70	P41	M	General-purpose I/O port
	SEG20		LCDC SEG output pin
71	P40	M	General-purpose I/O port
	SEG21		LCDC SEG output pin
72	PE0	M	General-purpose I/O port
	SEG22		LCDC SEG output pin
73	PE1	M	General-purpose I/O port
	SEG23		LCDC SEG output pin
74	PE2	M	General-purpose I/O port
	SEG24		LCDC SEG output pin
75	PE3	M	General-purpose I/O port
	SEG25		LCDC SEG output pin
76	PE4	M	General-purpose I/O port
	SEG26		LCDC SEG output pin
77	PE5	M	General-purpose I/O port
	SEG27		LCDC SEG output pin
	TO11		8/16-bit composite timer ch. 1 output pin

Table 1.7-1 Pin Functions (MB95410H Series) (7 / 7)

Pin no.	Pin name	I/O circuit type*	Function
78	PE6	M	General-purpose I/O port
	SEG28		LCDC SEG output pin
	TO10		8/16-bit composite timer ch. 1 output pin
79	PE7	M	General-purpose I/O port
	SEG29		LCDC SEG output pin
	EC1		8/16-bit composite timer ch. 1 clock input pin
80	AV_{SS}	—	A/D converter power supply pin (GND)

*: For the I/O circuit types, see "1.8 I/O Circuit Types".

■ Pin Functions (MB95470H Series)

Table 1.7-2 Pin Functions (MB95470H Series) (1 / 6)

Pin no.	Pin name	I/O circuit type*	Function
1	AV_{CC}	—	A/D converter power supply pin
2	P07	S	General-purpose I/O port
	INT07		External interrupt input pin
	AN07		A/D analog input pin
	SEG22		LCDC SEG output pin
3	P06	S	General-purpose I/O port
	INT06		External interrupt input pin
	AN06		A/D analog input pin
	SEG23		LCDC SEG output pin
4	P05	S	General-purpose I/O port
	INT05		External interrupt input pin
	AN05		A/D analog input pin
	SEG24		LCDC SEG output pin
	UCK1		UART/SIO ch. 1 clock I/O pin
5	P04	V	General-purpose I/O port
	INT04		External interrupt input pin
	AN04		A/D analog input pin
	SEG25		LCDC SEG output pin
	UI1		UART/SIO ch. 1 data input pin
6	P03	S	General-purpose I/O port
	INT03		External interrupt input pin
	AN03		A/D analog input pin
	SEG26		LCDC SEG output pin
	UO1		UART/SIO ch. 1 data output pin
7	P02	S	General-purpose I/O port
	INT02		External interrupt input pin
	AN02		A/D analog input pin
	SEG27		LCDC SEG output pin
	UCK2		UART/SIO ch. 2 clock I/O pin

Table 1.7-2 Pin Functions (MB95470H Series) (2 / 6)

Pin no.	Pin name	I/O circuit type*	Function
8	P01	V	General-purpose I/O port
	INT01		External interrupt input pin
	AN01		A/D analog input pin
	SEG28		LCDC SEG output pin
	TO00		8/16-bit composite timer ch. 0 output pin
	UI2		UART/SIO ch. 2 data input pin
9	P00	S	General-purpose I/O port
	INT00		External interrupt input pin
	AN00		A/D analog input pin
	SEG29		LCDC SEG output pin
	UO2		UART/SIO ch. 2 data output pin
10	P16	M	General-purpose I/O port
	SEG30		LCDC SEG output pin
	PPG10		8/16-bit PPG ch. 1 output pin
11	P15	M	General-purpose I/O port
	SEG31		LCDC SEG output pin
	PPG11		8/16-bit PPG ch. 1 output pin
12	P14	H	General-purpose I/O port
	UCK0		UART/SIO ch. 0 clock I/O pin
	EC0		8/16-bit composite timer ch. 0 clock input pin
	TI0		16-bit reload timer input pin
13	P13	H	General-purpose I/O port
	ADTG		A/D trigger input (ADTG) pin
	TO01		8/16-bit composite timer ch. 0 output pin
14	P12	D	General-purpose I/O port
	DBG		DBG input pin
15	P11	H	General-purpose I/O port
	UO0		UART/SIO ch. 0 data output pin
16	P10	G	General-purpose I/O port
	UI0		UART/SIO ch. 0 data input pin
	TO0		16-bit reload timer output pin

Table 1.7-2 Pin Functions (MB95470H Series) (3 / 6)

Pin no.	Pin name	I/O circuit type*	Function
17	P23	I	General-purpose I/O port
	SDA		I^2C data I/O pin
18	P22	I	General-purpose I/O port
	SCL		I^2C clock I/O pin
19	P21	T	General-purpose I/O port
	PPG01		8/16-bit PPG ch. 0 output pin
	CMPP		Voltage comparator input pin
20	P20	T	General-purpose I/O port
	PPG00		8/16-bit PPG ch. 0 output pin
	CMPN		Voltage comparator input pin
21	P90	R	General-purpose I/O port
	V4		LCDC drive power supply pin
22	P91	R	General-purpose I/O port
	V3		LCDC drive power supply pin
23	P92	R	General-purpose I/O port
	V2		LCDC drive power supply pin
24	P93	R	General-purpose I/O port
	V1		LCDC drive power supply pin
25	PA0	M	General-purpose I/O port
	COM0		LCDC COM output pin
26	PA1	M	General-purpose I/O port
	COM1		LCDC COM output pin
27	PA2	M	General-purpose I/O port
	COM2		LCDC COM output pin
28	PA3	M	General-purpose I/O port
	COM3		LCDC COM output pin
29	PA4	M	General-purpose I/O port
	COM4		LCDC COM output pin
30	PA5	M	General-purpose I/O port
	COM5		LCDC COM output pin

Table 1.7-2 Pin Functions (MB95470H Series) (4 / 6)

Pin no.	Pin name	I/O circuit type*	Function
31	PA6	M	General-purpose I/O port
	COM6		LCDC COM output pin
32	PA7	M	General-purpose I/O port
	COM7		LCDC COM output pin
33	V_{SS}	—	Power supply pin (GND)
34	PF1	B	General-purpose I/O port
	X1		Main clock oscillation pin
35	PF0	B	General-purpose I/O port
	X0		Main clock oscillation pin
36	C	—	Capacitor connection pin
37	PG2	C	General-purpose I/O port
	X1A		Subclock oscillation pin (32 kHz)
38	PG1	C	General-purpose I/O port
	X0A		Subclock oscillation pin (32 kHz)
39	V_{CC}	—	Power supply pin
40	PF2	A	General-purpose I/O port
	$\overline{\text{RST}}$		Reset pin Dedicated reset pin for MB95F474H/F476H/F478H
41	P17	H	General-purpose I/O port
	CMPO		Voltage comparator output pin
42	PB0	M	General-purpose I/O port
	SEG00		LCDC SEG output pin
43	PB1	M	General-purpose I/O port
	SEG01		LCDC SEG output pin
44	PC0	M	General-purpose I/O port
	SEG02		LCDC SEG output pin
45	PC1	M	General-purpose I/O port
	SEG03		LCDC SEG output pin
46	PC2	M	General-purpose I/O port
	SEG04		LCDC SEG output pin
47	PC3	M	General-purpose I/O port
	SEG05		LCDC SEG output pin

Table 1.7-2 Pin Functions (MB95470H Series) (5 / 6)

Pin no.	Pin name	I/O circuit type*	Function
48	P60	M	General-purpose I/O port
	SEG06		LCDC SEG output pin
49	P61	M	General-purpose I/O port
	SEG07		LCDC SEG output pin
50	P62	M	General-purpose I/O port
	SEG08		LCDC SEG output pin
51	P63	M	General-purpose I/O port
	SEG09		LCDC SEG output pin
52	P64	M	General-purpose I/O port
	SEG10		LCDC SEG output pin
53	P65	M	General-purpose I/O port
	SEG11		LCDC SEG output pin
54	P66	M	General-purpose I/O port
	SEG12		LCDC SEG output pin
55	P67	M	General-purpose I/O port
	SEG13		LCDC SEG output pin
56	PE0	M	General-purpose I/O port
	SEG14		LCDC SEG output pin
57	PE1	M	General-purpose I/O port
	SEG15		LCDC SEG output pin
58	PE2	M	General-purpose I/O port
	SEG16		LCDC SEG output pin
59	PE3	M	General-purpose I/O port
	SEG17		LCDC SEG output pin
60	PE4	M	General-purpose I/O port
	SEG18		LCDC SEG output pin
61	PE5	M	General-purpose I/O port
	SEG19		LCDC SEG output pin
	TO11		8/16-bit composite timer ch. 1 output pin

Table 1.7-2 Pin Functions (MB95470H Series) (6 / 6)

Pin no.	Pin name	I/O circuit type*	Function
62	PE6	M	General-purpose I/O port
	SEG20		LCDC SEG output pin
	TO10		8/16-bit composite timer ch. 1 output pin
63	PE7	M	General-purpose I/O port
	SEG21		LCDC SEG output pin
	EC1		8/16-bit composite timer ch. 1 clock input pin
64	AV_{SS}	—	A/D converter power supply pin (GND)

*: For the I/O circuit types, see "1.8 I/O Circuit Types".

MB95410H/470H Series

1.8 I/O Circuit Types

Table 1.8-1 lists the I/O circuit types. The alphabets in "Type" column of Table 1.8-1 correspond to those in "I/O circuit type" column of Table 1.7-1 and Table 1.7-2.

■ I/O Circuit Types

Table 1.8-1 I/O Circuit Types (1 / 5)

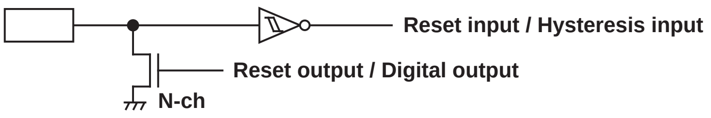
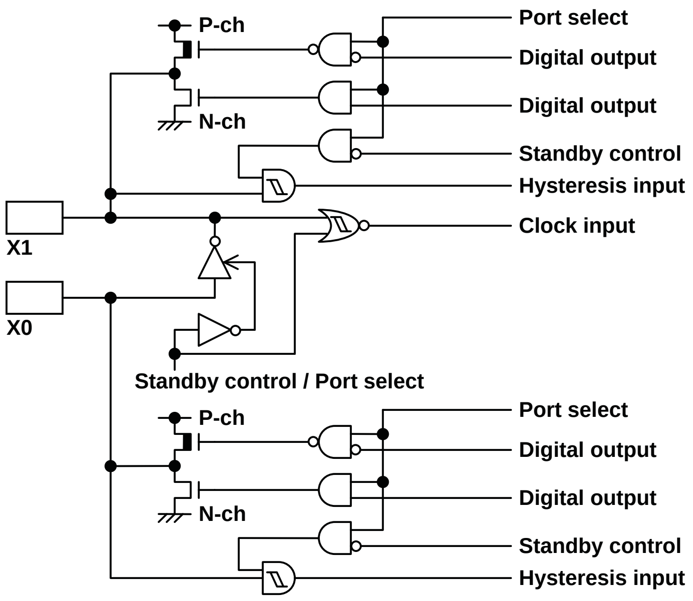
Type	Circuit	Remarks
A	 Reset input / Hysteresis input Reset output / Digital output N-ch	• N-ch open drain output • Hysteresis input • Reset output
B	 P-ch N-ch Port select Digital output Digital output Standby control Hysteresis input Clock input X1 X0 Standby control / Port select P-ch N-ch Port select Digital output Digital output Standby control Hysteresis input	• Oscillation circuit • High-speed side Feedback resistance: approx. 1 MΩ • CMOS output • Hysteresis input

Table 1.8-1 I/O Circuit Types (2 / 5)

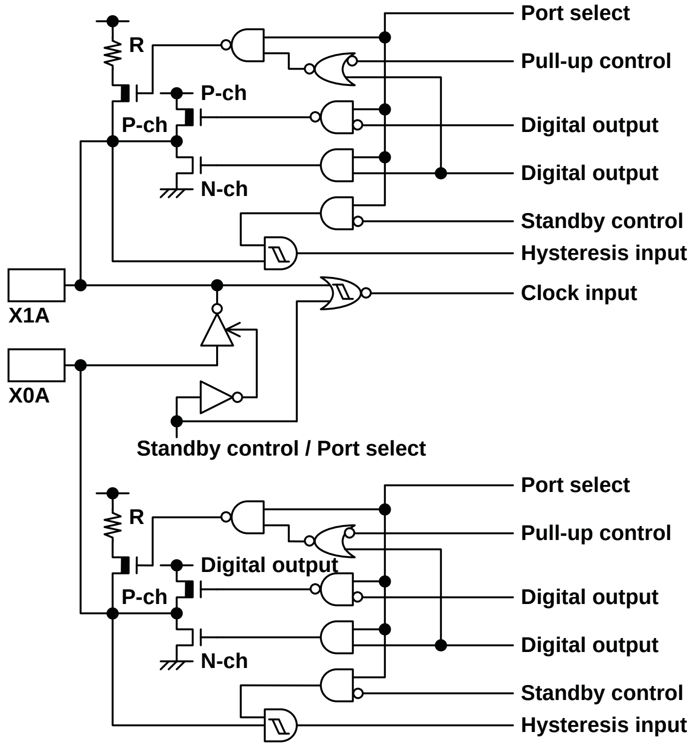
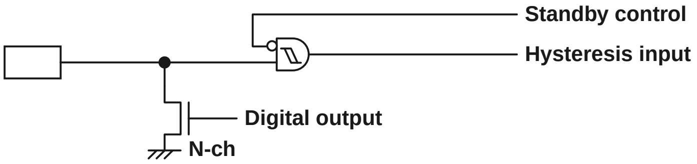
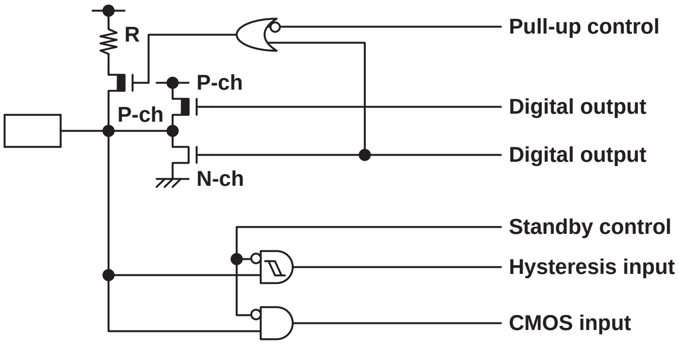
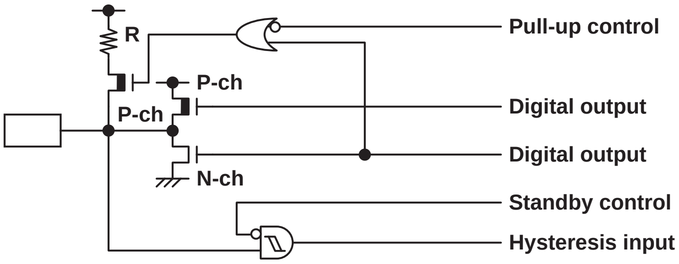
Type	Circuit	Remarks
C	 R P-ch P-ch N-ch Port select Pull-up control Digital output Digital output Standby control Hysteresis input Clock input X1A X0A Standby control / Port select R Digital output P-ch N-ch Port select Pull-up control Digital output Digital output Standby control Hysteresis input	• Oscillation circuit • Low-speed side Feedback resistance: approx. 10 MΩ • CMOS output • Hysteresis input • Pull-up control available
D	 Standby control Hysteresis input Digital output N-ch	• N-ch open drain output • Hysteresis input
G	 R P-ch P-ch N-ch Pull-up control Digital output Digital output Standby control Hysteresis input CMOS input	• CMOS output • Hysteresis input • CMOS input • Pull-up control available
H	 R P-ch P-ch N-ch Pull-up control Digital output Digital output Standby control Hysteresis input	• CMOS output • Hysteresis input • Pull-up control available

Table 1.8-1 I/O Circuit Types (3 / 5)

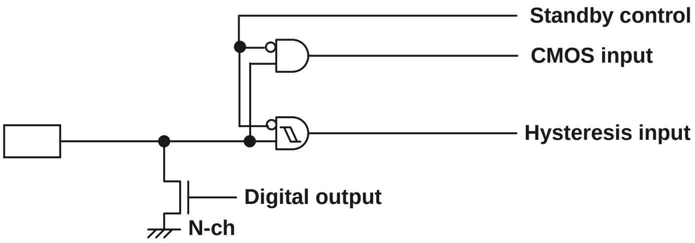
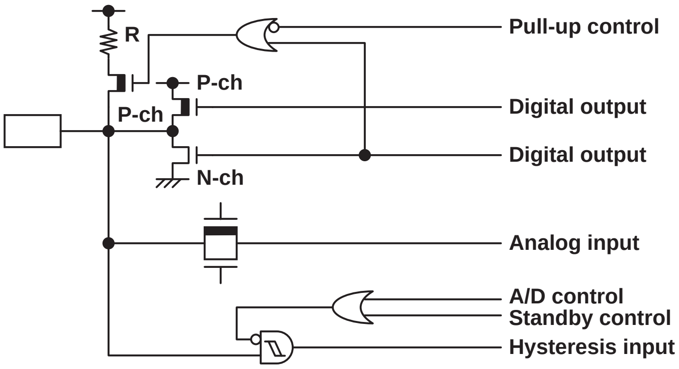
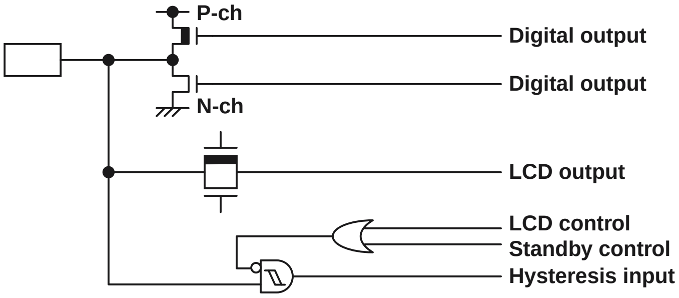
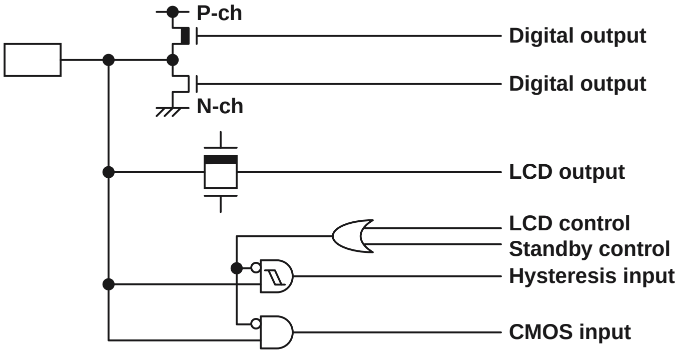
Type	Circuit	Remarks
I	 Standby control CMOS input Hysteresis input Digital output N-ch	• N-ch open drain output • CMOS input • Hysteresis input
J	 R Pull-up control P-ch P-ch Digital output Digital output N-ch Analog input A/D control Standby control Hysteresis input	• CMOS output • Hysteresis input • Analog input • Pull-up control available
M	 P-ch Digital output Digital output N-ch LCD output LCD control Standby control Hysteresis input	• CMOS output • LCD output • Hysteresis input
N	 P-ch Digital output Digital output N-ch LCD output LCD control Standby control Hysteresis input CMOS input	• CMOS output • LCD output • Hysteresis input • CMOS input

Table 1.8-1 I/O Circuit Types (4 / 5)

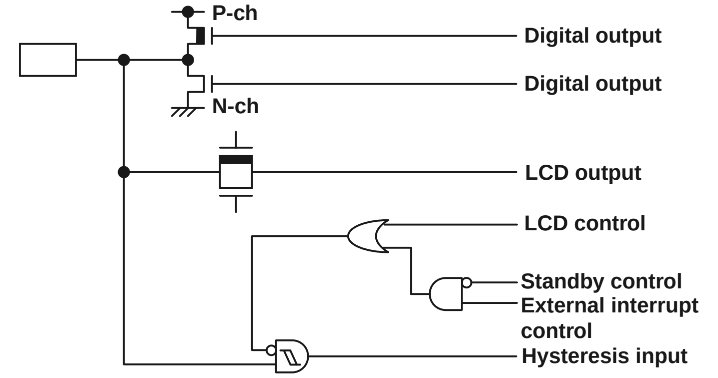
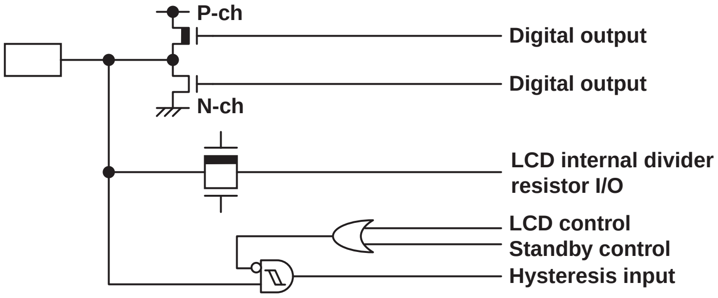
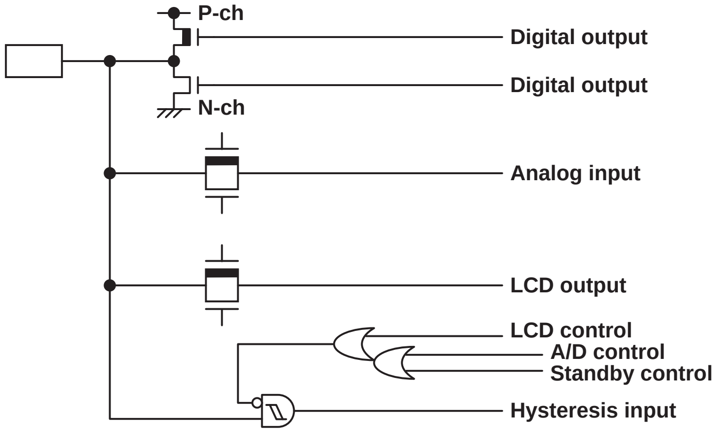
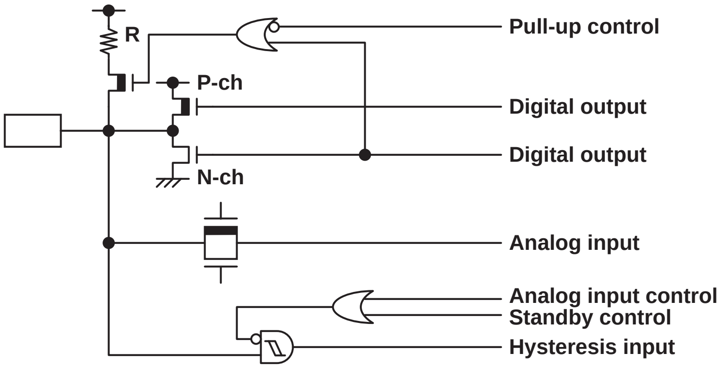
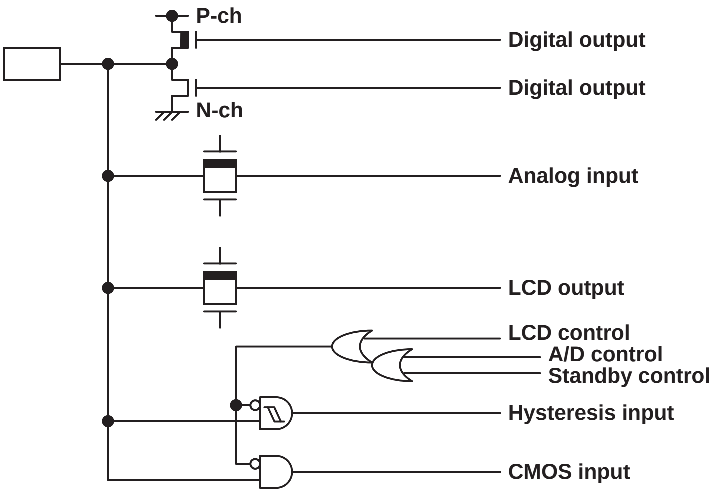
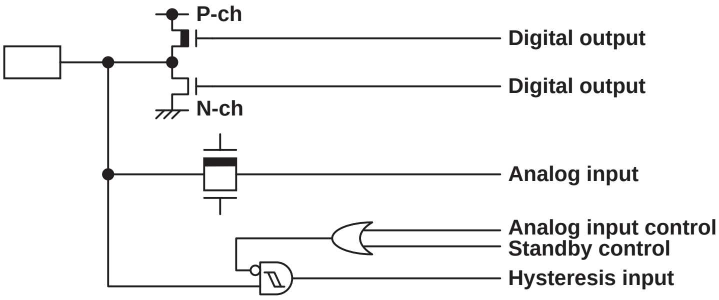
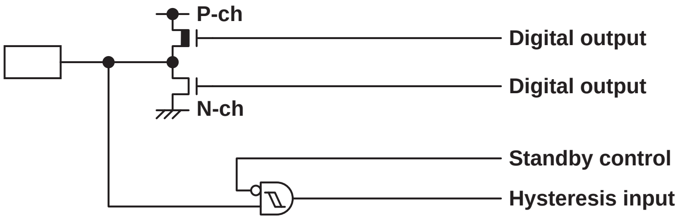
Type	Circuit	Remarks
Q	 P-ch Digital output Digital output N-ch LCD output LCD control Standby control External interrupt control Hysteresis input	• CMOS output • LCD output • Hysteresis input
R	 P-ch Digital output Digital output N-ch LCD internal divider resistor I/O LCD control Standby control Hysteresis input	• CMOS output • LCD power supply • Hysteresis input
S	 P-ch Digital output Digital output N-ch Analog input LCD output LCD control A/D control Standby control Hysteresis input	• CMOS output • LCD output • Hysteresis input • Analog input
T	 R Pull-up control P-ch Digital output Digital output N-ch Analog input Analog input control Standby control Hysteresis input	• CMOS output • Hysteresis input • Analog input • Pull-up control available

Table 1.8-1 I/O Circuit Types (5 / 5)

Type	Circuit	Remarks
V	 P-ch Digital output Digital output N-ch Analog input LCD output LCD control A/D control Standby control Hysteresis input CMOS input	• CMOS output • LCD output • Hysteresis input • Analog input • CMOS input
W	 P-ch Digital output Digital output N-ch Analog input Analog input control Standby control Hysteresis input	• CMOS output • Hysteresis input • Analog input
Y	 P-ch Digital output Digital output N-ch Standby control Hysteresis input	• CMOS output • Hysteresis input

CHAPTER 2

NOTES ON DEVICE HANDLING

This chapter provides notes on using the MB95410H/470H Series.

2.1 Notes on Device Handling

This section provides notes on power supply voltage and pin treatment.

■ Device Handling

- Preventing latch-ups

  When using the device, ensure that the voltage applied does not exceed the maximum voltage rating.

  In a CMOS IC, if a voltage higher than V_{CC} or a voltage lower than V_{SS} is applied to an input/output pin that is neither a medium-withstand voltage pin nor a high-withstand voltage pin, or if a voltage out of the rating range of power supply voltage mentioned in 1. "Absolute Maximum Ratings" of "■ ELECTRICAL CHARACTERISTICS" in the data sheet of the MB95410H/470H Series is applied to the V_{CC} pin or the V_{SS} pin, a latch-up may occur.

  When a latch-up occurs, power supply current increases significantly, which may cause a component to be thermally destroyed.

  Ensure that the analog power supply voltage (AV_{CC}) and the analog input voltage do not exceed the digital power supply voltage (V_{CC}) even when turning on or off the analog system power supply.

- Stabilizing supply voltage

  Supply voltage must be stabilized.

  A malfunction may occur when power supply voltage fluctuates rapidly even though the fluctuation is within the guaranteed operating range of the V_{CC} power supply voltage.

  As a rule of voltage stabilization, suppress voltage fluctuation so that the fluctuation in V_{CC} ripple (p-p value) at the commercial frequency (50 Hz/60 Hz) does not exceed 10% of the standard V_{CC} value, and the transient fluctuation rate does not exceed 0.1 V/ms at a momentary fluctuation such as switching the power supply.

- Notes on using the external clock

  When an external clock is used, oscillation stabilization wait time is required for power-on reset, wakeup from subclock mode or stop mode.

■ Pin Connection

- Treatment of unused pins

  If an unused input pin is left unconnected, a component may be permanently damaged due to malfunctions or latch-ups. Always pull up or pull down an unused input pin through a resistor of at least 2 kΩ. Set an unused input/output pin to the output state and leave it unconnected, or set it to the input state and treat it the same as an unused input pin. If there is an unused output pin, leave it unconnected.

- Treatment of power supply pins on A/D converter

  Ensure that $AV_{CC} = V_{CC}$ and $AV_{SS} = V_{SS}$ when the A/D converter is not in use.

  Any noise riding on the AV_{CC} pin may cause accuracy degradation. Therefore, it is advisable to connect a ceramic bypass capacitor of approximately 0.1 µF between the AV_{CC} pin and the AV_{SS} pin at a location close to this device.

- Power supply pins

  To reduce unnecessary electro-magnetic emission, prevent malfunctions of strobe signals due to an increase in the ground level, and conform to the total output current standard, always connect the V_{CC} pin and the V_{SS} pin to the power supply and ground outside the device. In addition, connect the current supply source to the V_{CC} pin and the V_{SS} pin with low impedance.

  It is also advisable to connect a ceramic bypass capacitor of approximately 0.1 μF between the V_{CC} pin and the V_{SS} pin at a location close to this device.

- DBG pin

  Connect the DBG pin to an external pull-up resistor of 2 kΩ or above.

  After power-on, ensure that the DBG pin does not stay at "L" level until the reset output is released.

  The DBG pin becomes a communication pin in debug mode. Since the actual pull-up resistance depends on the tool used and the interconnection length, refer to the tool document when selecting a pull-up resistor.

- $\overline{\text{RST}}$ pin

  Connect the $\overline{\text{RST}}$ pin to an external pull-up resistor of 2 kΩ or above.

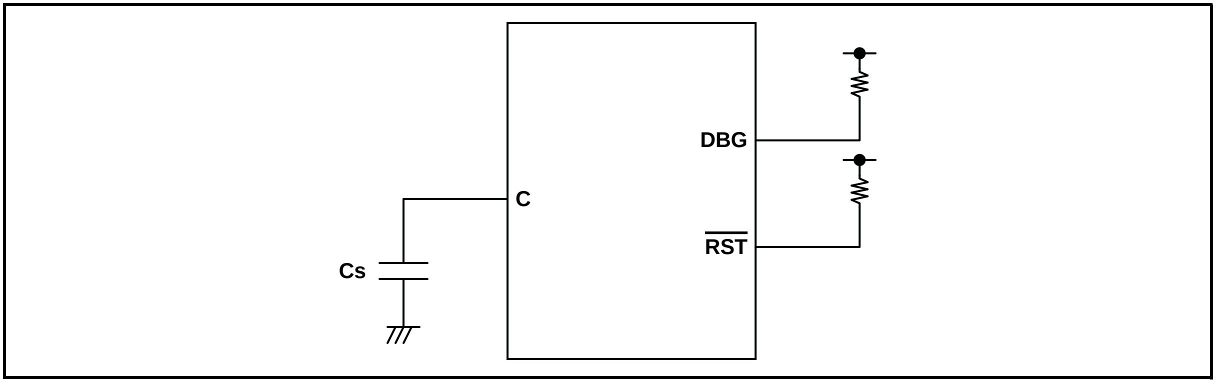
  To prevent the device from unintentionally entering the reset mode due to noise, minimize the interconnection length between a pull-up resistor and the $\overline{\text{RST}}$ pin and that between a pull-up resistor and the V_{CC} pin when designing the layout of the printed circuit board.

  The PF2/$\overline{\text{RST}}$ pin functions as the reset input/output pin after power-on. In addition, the reset output of the PF2/$\overline{\text{RST}}$ pin can be enabled by the RSTOE bit in the SYSC register, and the reset input function and the general purpose I/O function can be selected by the RSTEN bit in the SYSC register.

- C pin

  Use a ceramic capacitor or a capacitor with equivalent frequency characteristics. The decoupling capacitor for the V_{CC} pin must have a capacitance equal to or larger than the capacitance of C_S. For the connection to a smoothing capacitor C_S, see the diagram below. To prevent the device from unintentionally entering an unknown mode due to noise, minimize the distance between the C pin and C_S and the distance between C_S and the V_{SS} pin when designing the layout of a printed circuit board.

Figure 2.1-1 DBG/$\overline{\text{RST}}$/C Pins Connection



- Note on serial communication

  In serial communication, reception of wrong data may occur due to noise or other causes. Therefore, design a printed circuit board to prevent noise from occurring. Taking account of the reception of wrong data, take measures such as adding a checksum to the end of data in order to detect errors. If an error is detected, retransmit the data.

- Analog power supply

  Always use the same potential for the AV_{CC} pin and the V_{CC} pin. If V_{CC} is larger than AV_{CC}, current may flow through the analog input pins (AN).

CHAPTER 3

MEMORY SPACE

This chapter describes the memory space.

3.1 Memory Space

The memory space of the MB95410H/470H Series is 64 Kbyte in size and consists of an I/O area, an extended I/O area, a data area, and a program area. The memory space includes areas for specific applications such as general-purpose registers and a vector table.

■ Configuration of Memory Space

● I/O area (addresses: 0000_H to $007F_H$)

- This area contains the control registers and data registers for built-in peripheral functions.
- As the I/O area forms part of the memory space, it can be accessed in the same way as the memory. It can also be accessed at high-speed by using direct addressing instructions.

● Extended I/O area (addresses: $0F80_H$ to $0FFF_H$)

- This area contains the control registers and data registers for built-in peripheral functions.
- As the extended I/O area forms part of the memory space, it can be accessed in the same way as the memory.

● Data area

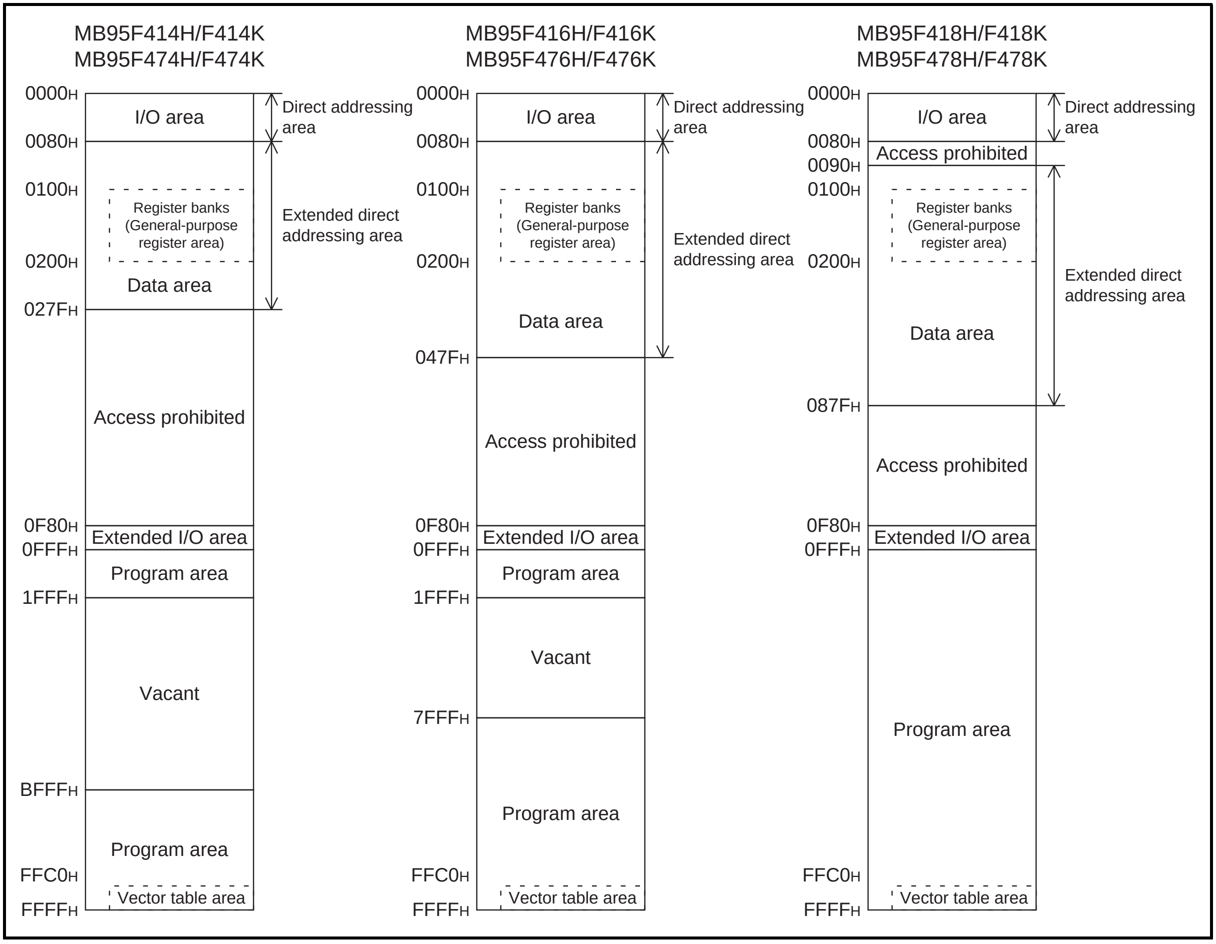
- Static RAM is incorporated in the data area as the internal data area.
- The internal RAM size varies according to the product.
- The RAM area from 0090_H to $00FF_H$ can be accessed at high-speed by using the direct addressing instruction.
- In MB95F414H/F414K/F474H/F474K, the area from 0100_H to $027F_H$ is an extended direct addressing area. It can be accessed at high-speed by the direct addressing instruction with a direct bank pointer set.
- In MB95F416H/F416K/F418H/F418K/F476H/F476K/F478H/F478K, the area from 0100_H to $047F_H$ is an extended direct addressing area. It can be accessed at high-speed by the direct addressing instruction with a direct bank pointer set.
- In MB95F418H/F418K/F478H/F478K, the area from 0480_H to $087F_H$ is an extended direct addressing area. It cannot be accessed at high-speed by the direct addressing instruction with a direct bank pointer set.
- The area from 0100_H to $01FF_H$ can be used as a general-purpose register area.

● Program area

- ROM is incorporated in the program area as the internal program area.
- The internal ROM size varies according to the product.
- The area from $FFC0_H$ to $FFFF_H$ is used as the vector table and $FFFC_H$ is the Flash security byte.
- The area from $FFBC_H$ to $FFBF_H$ is used to store data of the non-volatile register.

■ Memory Maps

Figure 3.1-1 Memory Maps



3.1.1 Areas for Specific Applications

The general-purpose register area and vector table area are used for the specific applications.

■ General-purpose Register Area (Addresses: 0100_H to $01FF_H$)

- This area contains the auxiliary registers used for 8-bit arithmetic operations, transfer, etc.
- As this area forms part of the RAM area, it can also be used as conventional RAM.
- When the area is used as general-purpose registers, general-purpose register addressing enables high-speed access with short instructions.

For details, see "5.1.1 Register Bank Pointer (RP)" and "5.2 General-purpose Register".

■ Non-volatile Register Data Area (Addresses: $FFBC_H$ to $FFBF_H$)

The area from $FFBC_H$ to $FFBF_H$ is used to store data of the non-volatile register. For details, see "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION".

■ Vector Table Area (Addresses: $FFC0_H$ to $FFFF_H$)

- This area is used as the vector table for vector call instructions (CALLV), interrupts, and resets. $FFFC_H$ is the Flash security byte.
- The top of the ROM area is allocated to the vector table area. The start address of a service routine is set to an address in the vector table in the form of data.

Table 8.1-1 in "CHAPTER 8 INTERRUPTS" lists the vector table addresses corresponding to vector call instructions, interrupts, and resets.

For details, see "CHAPTER 7 RESET", "CHAPTER 8 INTERRUPTS", and "● CALLV #vct" in "E.2 Special Instruction" in "APPENDIX".

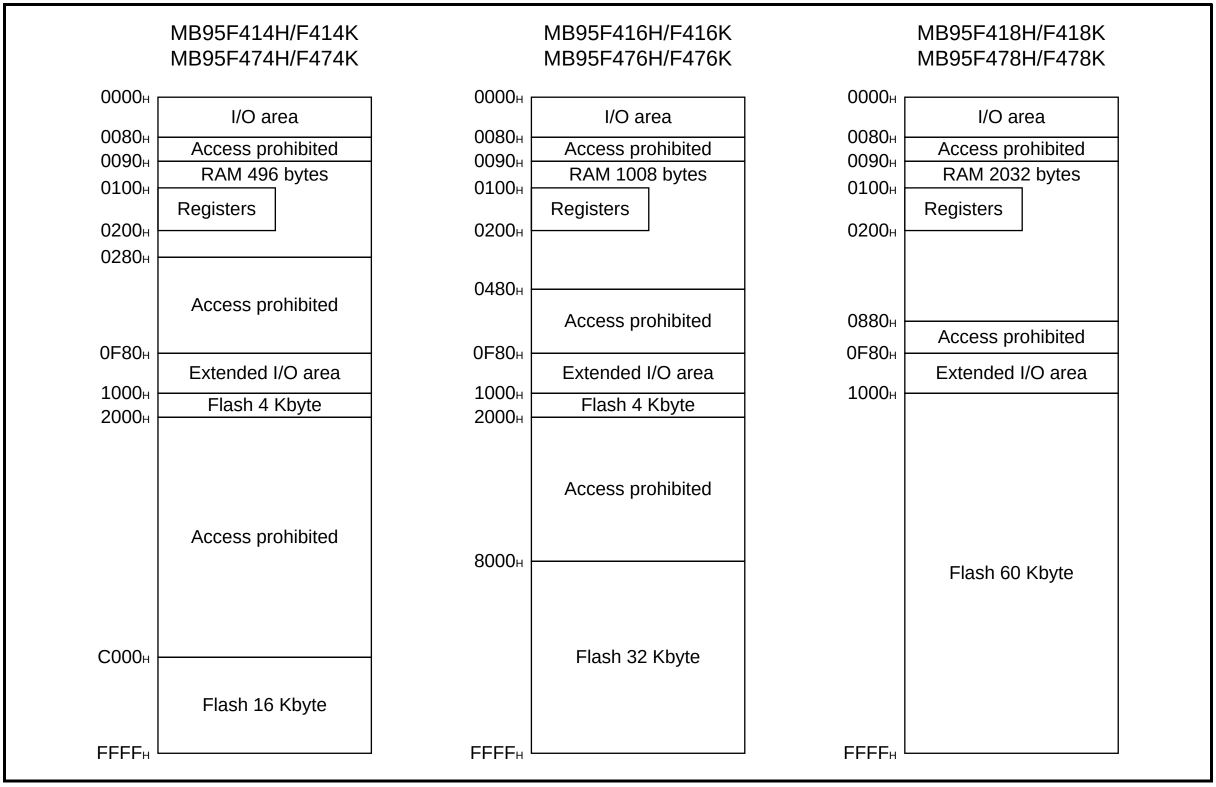
MB95410H/470H Series

3.2 Memory Maps

This section shows the memory maps of the MB95410H/470H Series.

■ Memory Maps

Figure 3.2-1 Memory Maps of Different Products



Parameter / Part number	Flash memory	RAM
MB95F414H/F414K/F474H/F474K	20 Kbyte	496 bytes
MB95F416H/F416K/F476H/F476K	36 Kbyte	1008 bytes
MB95F418H/F418K/F478H/F478K	60 Kbyte	2032 bytes

CHAPTER 4

MEMORY ACCESS MODE

This chapter describes the memory access mode.

4.1 Memory Access Mode

The MB95410H/470H Series supports only one memory access mode: single-chip mode.

■ Single-chip Mode

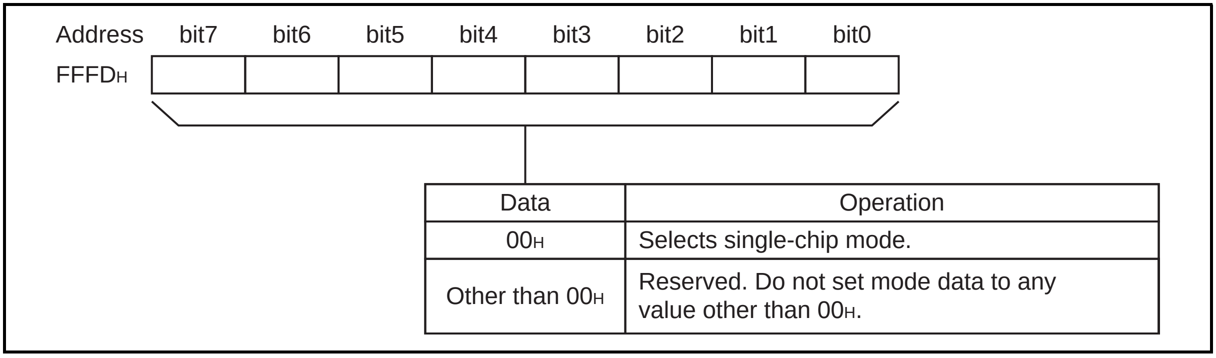
In single-chip mode, only the internal RAM and ROM are used, and no external bus access is executed.

● Mode data

Mode data is the data used to determine the memory access mode of the CPU.

The mode data address is fixed at "FFFD$_H$". Always set the mode data of the internal ROM to "00$_H$" to select the single-chip mode.

Figure 4.1-1 Mode Data Settings



Data	Operation
00H	Selects single-chip mode.
Other than 00H	Reserved. Do not set mode data to any value other than 00H.

After a reset is released, the CPU fetches mode data first.

The CPU then fetches the reset vector after the mode data. It starts executing instructions from the address set in the reset vector.

CHAPTER 5

CPU

This chapter describes the functions and operations of the CPU.

5.1 Dedicated Registers

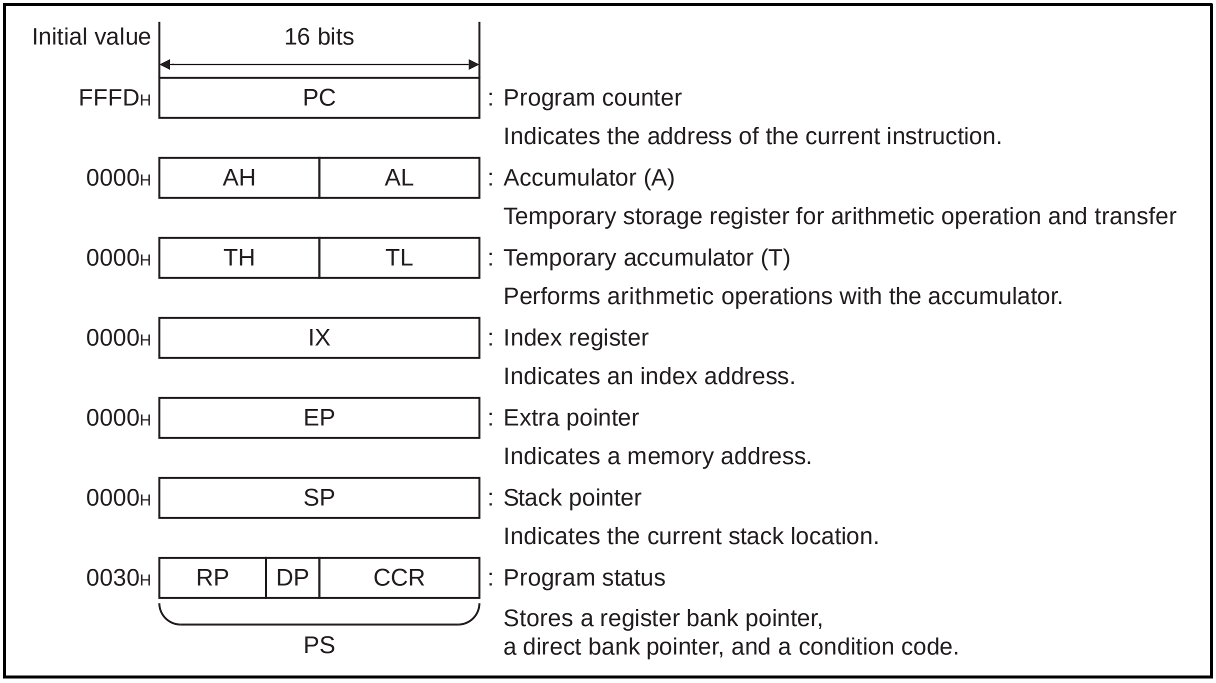
The CPU has dedicated registers: a program counter (PC), two registers for arithmetic operations (A and T), three address pointers (IX, EP, and SP), and the program status (PS) register. Each of the registers is 16 bits long. The PS register consists of the register bank pointer (RP), direct pointer (DP), and condition code register (CCR).

■ Configuration of Dedicated Registers

The dedicated registers in the CPU consist of seven 16-bit registers. As for the accumulator (A) and the temporary accumulator (T), using only the lower eight bits of the respective registers is also supported.

Figure 5.1-1 shows the configuration of the dedicated registers.

Figure 5.1-1 Configuration of Dedicated Registers



Initial value	Register	
FFFD$_H$	PC	: Program counter Indicates the address of the current instruction.
0000$_H$	AH | AL	: Accumulator (A) Temporary storage register for arithmetic operation and transfer
0000$_H$	TH | TL	: Temporary accumulator (T) Performs arithmetic operations with the accumulator.
0000$_H$	IX	: Index register Indicates an index address.
0000$_H$	EP	: Extra pointer Indicates a memory address.
0000$_H$	SP	: Stack pointer Indicates the current stack location.
0030$_H$	RP | DP | CCR	: Program status Stores a register bank pointer, a direct bank pointer, and a condition code.

■ Functions of Dedicated Registers

● Program counter (PC)

The program counter is a 16-bit counter which contains the memory address of the instruction currently executed by the CPU. The program counter is updated whenever an instruction is executed or an interrupt or a reset occurs. The initial value set immediately after a reset is the mode data read address (FFFD$_H$).

● Accumulator (A)

The accumulator is a 16-bit register for arithmetic operation. It is used for a variety of arithmetic and transfer operations of data in memory or data in other registers such as the temporary accumulator (T). The data in the accumulator can be handled either as word (16-bit) data or byte (8-bit) data. For byte-length arithmetic and transfer operations, only the lower eight bits (AL) of the accumulator are used with the upper eight bits (AH) left unchanged. The initial value set immediately after a reset is "0000$_H$".

- Temporary accumulator (T)

The temporary accumulator is an auxiliary 16-bit register for arithmetic operation. It is used to perform arithmetic operations with the data in the accumulator (A). The data in the temporary accumulator is handled as word data for word-length (16-bit) operations with the accumulator (A) and as byte data for byte-length (8-bit) operations. For byte-length operations, only the lower eight bits (TL) of the temporary accumulator are used and the upper eight bits (TH) are not used.

When a MOV instruction is used to transfer data to the accumulator (A), the previous contents of the accumulator are automatically transferred to the temporary accumulator. When transferring byte-length data, the upper eight bits (TH) of the temporary accumulator remain unchanged. The initial value after a reset is "0000_H".

- Index register (IX)

The index register is a 16-bit register used to hold the index address. The index register is used with a single-byte offset (-128 to +127). The offset value is added to the index address to generate the memory address for data access. The initial value after a reset is "0000_H".

- Extra pointer (EP)

The extra pointer is a 16-bit register which contains the value indicating the memory address for data access. The initial value after a reset is "0000_H".

- Stack pointer (SP)

The stack pointer is a 16-bit register which holds the address referenced when an interrupt or a sub-routine call occurs and by the stack push and pop instructions. During program execution, the value of the stack pointer indicates the address of the most recent data pushed onto the stack. The initial value after a reset is "0000_H".

- Program status (PS)

The program status is a 16-bit control register. The upper eight bits consists of the register bank pointer (RP) and direct bank pointer (DP); the lower eight bits consists of the condition code register (CCR).

In the upper eight bits, the upper five bits consists of the register bank pointer used to contain the address of the general-purpose register bank. The lower three bits consists of the direct bank pointer which locates the area to be accessed at high-speed by direct addressing.

The lower eight bits consists of the condition code register (CCR) which consists of flags that represent the state of the CPU.

The instructions that can access the program status are MOVW A,PS and MOVW PS,A. The register bank pointer (RP) and direct bank pointer (DP) in the program status register can also be read from and written to by accessing the mirror address (0078_H).

Note that the condition code register (CCR) is a part of the program status register and cannot be accessed independently.

Refer to the "F^2MC-8FX Programming Manual" for details on using the dedicated registers.

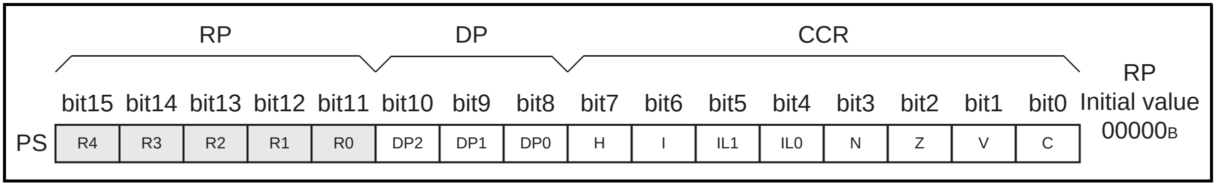
5.1.1 Register Bank Pointer (RP)

The register bank pointer (RP) in bit15 to bit11 of the program status (PS) register contains the address of the general-purpose register bank that is currently in use and is translated into a real address when general-purpose register addressing is used.

■ Configuration of Register Bank Pointer (RP)

Figure 5.1-2 shows the configuration of the register bank pointer.

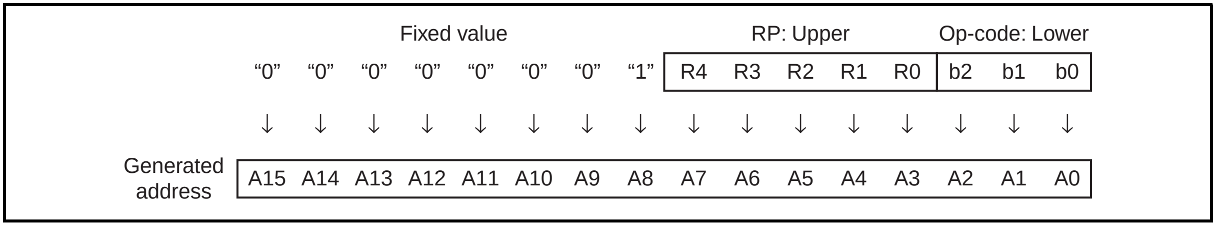
Figure 5.1-2 Configuration of Register Bank Pointer



	RP					DP			CCR								
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	RP Initial value
PS	R4	R3	R2	R1	R0	DP2	DP1	DP0	H	I	IL1	IL0	N	Z	V	C	00000_B

The register bank pointer contains the address of the register bank currently in use. The content of the register bank pointer is translated into a real address according to the rule shown in Figure 5.1-3.

Figure 5.1-3 Rule for Translation into Real Addresses in General-purpose Register Area



	Fixed value								RP: Upper					Op-code: Lower		
	"0"	"0"	"0"	"0"	"0"	"0"	"0"	"1"	R4	R3	R2	R1	R0	b2	b1	b0
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Generated address	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0

The register bank pointer specifies the register bank used as general-purpose registers in the RAM area. There are a total of 32 register banks. The current register bank is specified by setting a value between 0 and 31 in the upper five bits of the register bank pointer. Each register bank has eight 8-bit general-purpose registers which are selected by the lower three bits of the op-code.

The register bank pointer allows the space from "0100_H" to "$01FF_H$"(max) to be used as a general-purpose register area. However, certain products have restrictions on the size of the area available for the general-purpose register area. The initial value of the register bank pointer after a reset is "0000_H".

■ Mirror Address for Register Bank and Direct Bank Pointer

Values can be written to the register bank pointer (RP) and the direct bank pointer (DP) by accessing the program status (PS) register with the "MOVW A,PS" instruction; the two pointers can be read by accessing PS with the "MOVW PS,A" instruction. Values can also be directly written to and read from the two pointers by accessing "0078_H", the mirror address of the register bank pointer.

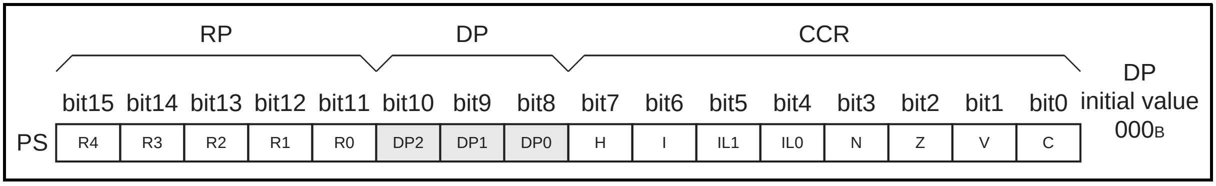
5.1.2 Direct Bank Pointer (DP)

The direct bank pointer (DP) in bit10 to bit8 of the program status (PS) register specifies the area to be accessed by direct addressing.

■ Configuration of Direct Bank Pointer (DP)

Figure 5.1-4 shows the configuration of the direct bank pointer.

Figure 5.1-4 Configuration of Direct Bank Pointer



	RP					DP			CCR							
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PS	R4	R3	R2	R1	R0	DP2	DP1	DP0	H	I	IL1	IL0	N	Z	V	C

DP initial value 000_B

The area of "0000_H to $007F_H$" and that of "0090_H to $047F_H$" can be accessed by direct addressing. Access to 0000_H to $007F_H$ is specified by an operand regardless of the value in the direct bank pointer. Access to 0090_H to $047F_H$ is specified by the value of the direct bank pointer and the operand.

Table 5.1-1 shows the relationship between the direct bank pointer (DP) and the access area; Table 5.1-2 lists the direct addressing instructions.

Table 5.1-1 Direct Bank Pointer and Access Area

Direct bank pointer (DP[2:0])	Operand-specified dir	Access area
XXX_B (It does not affect mapping.)	0000_H to $007F_H$	0000_H to $007F_H$
000_B (Initial value)	0090_H to $00FF_H$	0090_H to $00FF_H$*1
001_B		0100_H to $017F_H$
010_B		0180_H to $01FF_H$
011_B		0200_H to $027F_H$
100_B		0280_H to $02FF_H$*2
101_B		0300_H to $037F_H$
110_B		0380_H to $03FF_H$
111_B		0400_H to $047F_H$

*1: Due to the memory size limit, it is "0090_H to $00FF_H$" in the MB95410H/470H Series.

*2: The available access area is up to "0280_H" in MB95F414H/F414K/F474H/F474K.

Table 5.1-2 Direct Address Instruction List

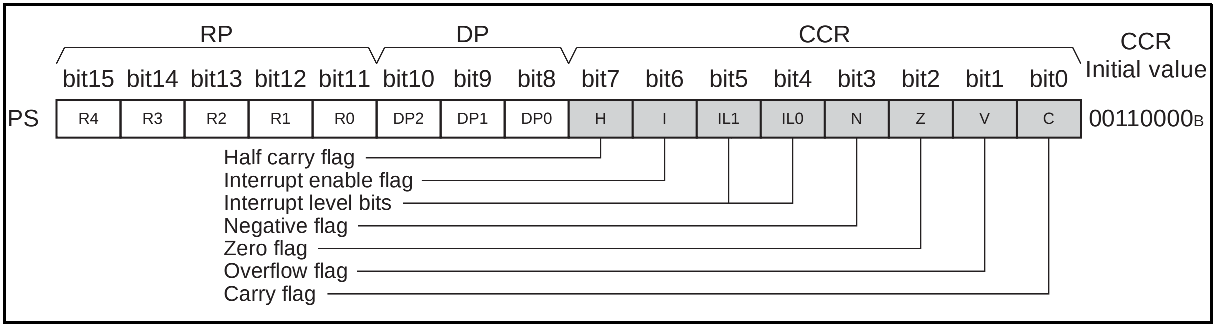
Applicable instructions
CLRB dir:bit
SETB dir:bit
BBC dir:bit,rel
BBS dir:bit,rel
MOV A,dir
CMP A,dir
ADDC A,dir
SUBC A,dir
MOV dir,A
XOR A,dir
AND A,dir
OR A,dir
MOV dir,#imm
CMP dir,#imm
MOVW A,dir
MOVW dir,A

5.1.3 Condition Code Register (CCR)

The condition code register (CCR) in the lower eight bits of the program status (PS) register consists of the bits (H, N, Z, V, and C) containing information about the arithmetic result or transfer data and the bits (I, IL1, and IL0) used to control the acceptance of interrupt requests.

■ Configuration of Condition Code Register (CCR)

Figure 5.1-5 Configuration of Condition Code Register



The condition code register is a part of the program status (PS) register and therefore cannot be accessed independently.

■ Bits Showing Operation Results

- Half carry flag (H)

This flag is set to "1" when a carry from bit3 to bit4 or a borrow from bit4 to bit3 occurs due to the result of an operation. Otherwise, the flag is set to "0". Do not use this flag for any operation other than addition and subtraction as the flag is intended for decimal-adjusted instructions.

- Negative flag (N)

This flag is set to "1" when the value of the most significant bit is "1" due to the result of an operation, and is set to "0" when the value of the most significant bit is "0".

- Zero flag (Z)

This flag is set to "1" when the result of an operation is "0", and is set to "0" when the result is "1".

- Overflow flag (V)

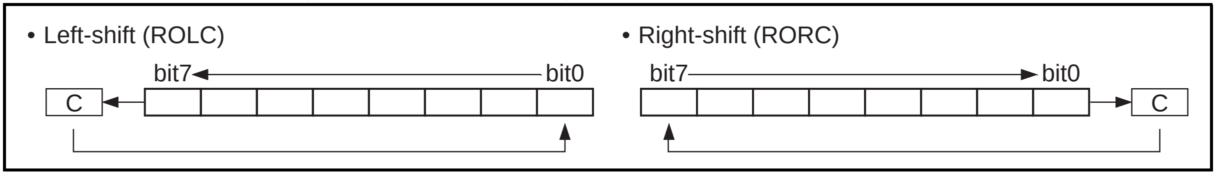
This flag indicates whether the result of an operation has caused an overflow, with the operand used in the operation being regarded as an integer expressed as a complement of two. If an overflow occurs, the overflow flag is set to "1"; otherwise, it is set to "0".

● Carry flag (C)

This flag is set to "1" when a carry from bit7 or a borrow to bit7 occurs due to the result of an operation. Otherwise, the flag is set to "0". When a shift instruction is executed, the flag is set to the shift-out value.

Figure 5.1-6 shows how the carry flag is updated by a shift instruction.

Figure 5.1-6 Carry Flag Updated by Shift Instruction



■ Interrupt Acceptance Control Bits

● Interrupt enable flag (I)

When this flag is set to "1", interrupts are enabled and accepted by the CPU. When this flag is set to "0", interrupts are disabled and rejected by the CPU.

The initial value after a reset is "0".

The SETI and CLRI instructions set and clear the flag to "1" and "0", respectively.

● Interrupt level bits (IL1, IL0)

These bits indicate the level of the interrupt currently accepted by the CPU.

The interrupt level is compared with the value of the interrupt level setting register (ILR0 to ILR5) that corresponds to the interrupt request (IRQ00 to IRQ23) of each peripheral function.

The CPU services an interrupt request only when its interrupt level is smaller than the value of these bits with the interrupt enable flag set (CCR:I = 1). Table 5.1-3 lists interrupt level priorities. The initial value after a reset is "11_B".

Table 5.1-3 Interrupt Levels

IL1	IL0	Interrupt level	Priority
0	0	0	High
0	1	1	↕
1	0	2	
1	1	3	Low (No interrupt)

The interrupt level bits (IL1, IL0) are usually "11_B" when the CPU does not service an interrupt (with the main program running).

For details of interrupts, see "8.1 Interrupts".

5.2 General-purpose Register

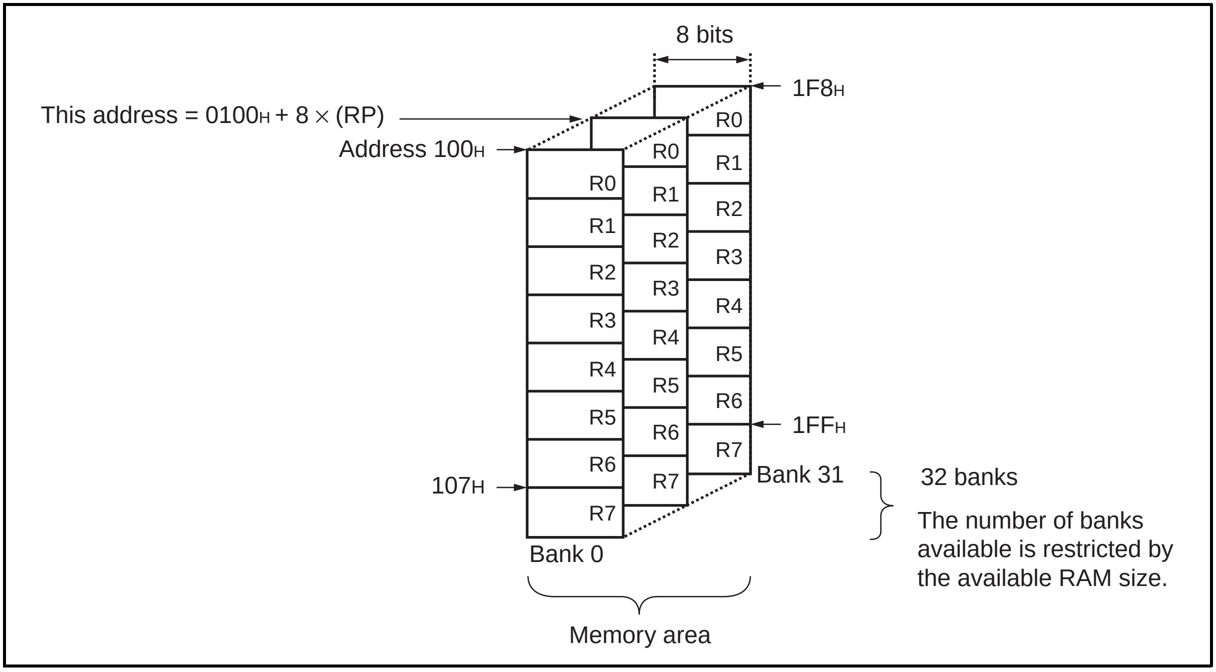
The general-purpose registers are a memory block in which each bank consists of eight 8-bit registers. Up to 32 register banks can be used in total. The register bank pointer (RP) is used to specify a register bank.
Register banks are useful for interrupt handling, vector call processing, and sub-routine calls.

■ Configuration of General-purpose Register

- The general-purpose register is an 8-bit register and is located in a register bank in the general-purpose register area (in RAM).
- Up to 32 banks can be used, each of which consists of eight registers (R0 to R7).
- The register bank pointer (RP) specifies the register bank currently being used and the lower three bits of the op-code specify the general-purpose register 0 (R0) to the general-purpose register 7 (R7).

Figure 5.2-1 shows the configuration of the register banks.

Figure 5.2-1 Configuration of Register Banks


8 bits
1F8H
This address = 0100H + 8 × (RP)
Address 100H
R0
R1
R2
R3
R4
R5
R6
R7
1FFH
107H
Bank 31
Bank 0
32 banks
The number of banks available is restricted by the available RAM size.
Memory area

For information on the general-purpose register area available in each model, see "3.1.1 Areas for Specific Applications".

■ Features of General-purpose Registers

The general-purpose register has the following features.

- High-speed access to RAM with short instructions (general-purpose register addressing).
- Grouping registers into a block of register banks facilitates data protection and division of registers in terms of functions.

A general-purpose register bank can be allocated exclusively to an interrupt service routine or a vector call (CALLV #0 to #7) processing routine. For instance, the fourth register bank is always assigned to the second interrupt.

Data of a general-purpose register before an interrupt can be saved to a dedicated register bank by just specifying that register bank at the beginning of an interrupt service routine. This therefore eliminates the need to save data of a general-purpose register in a stack, thereby enabling the CPU to receive interrupts at high speed.

Notes:

In an interrupt service routine, include one of the following in a program to ensure that values of the interrupt level bits (CCR:IL1, IL0) of the condition code register are not modified when modifying a register bank pointer (RP) to specify a register bank.

- Read the interrupt level bits and save their values before writing a value to the RP.
- Directly write a new value to the RP mirror address "0078_H" to update the RP.

5.3 Placement of 16-bit Data in Memory

This section describes how 16-bit data is stored in memory.

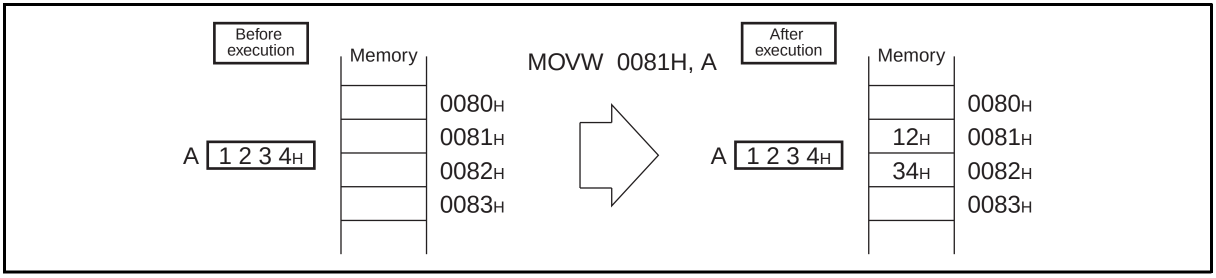
■ Placement of 16-bit Data in Memory

● State of 16-bit data stored in RAM

When 16-bit data is written to memory, the upper byte of the data is stored at a smaller address and the lower byte is stored at the next address. When 16-bit data is read, it is handled in the same way.

Figure 5.3-1 shows how 16-bit data is placed in memory.

Figure 5.3-1 Placement of 16-bit Data in Memory



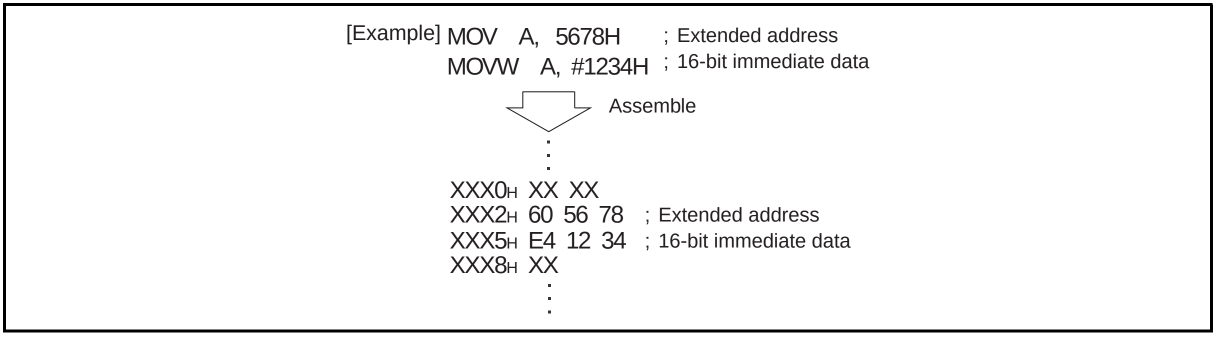
● Storage state of 16-bit data specified by an operand

Even when the operand in an instruction specifies 16-bit data, the upper byte is stored at the address closer to the op-code (instruction) and the lower byte is stored at the address next to the one at which the upper byte is stored.

That is true whether an operand is either a memory address or 16-bit immediate data.

Figure 5.3-2 shows how 16-bit data in an instruction is placed.

Figure 5.3-2 Placement of 16-bit Data in Instruction


```
[Example] MOV   A,  5678H    ; Extended address
          MOVW  A,  #1234H   ; 16-bit immediate data
```

```
    :
XXX0H XX XX
XXX2H 60 56 78  ; Extended address
XXX5H E4 12 34  ; 16-bit immediate data
XXX8H XX
    :
```

● Storage state of 16-bit data in the stack

When 16-bit register data is saved in a stack on an interrupt, the upper byte is stored at a lower address in the same way as 16-bit data specified by an operand.

CHAPTER 6

CLOCK CONTROLLER

This chapter describes the functions and operations of the clock controller.

6.1 Overview of Clock Controller

The New 8FX family has a built-in clock controller that optimizes its power consumption. It supports both the external main clock and the external subclock.
The clock controller enables/disables clock oscillation, enables/disables the supply of clock signals to the internal circuit, selects the clock source, and controls the PLL, the CR oscillator and frequency divider circuits.

■ Overview of Clock Controller

The clock controller enables/disables clock oscillation, enables/disables clock supply to the internal circuit, selects the clock source, and controls the PLL, the CR oscillator and frequency divider circuits.

The clock controller controls the internal clock according to the clock mode, standby mode settings and the reset operation. The clock mode is used to select an internal operating clock; the standby mode is used to enable and disable clock oscillation and signal supply.

The clock controller selects the optimum power consumption and functions depending on the combination of clock mode and standby mode.

This device has five source clocks: a main clock formed by dividing the main oscillation clock by two, a main PLL clock formed by multiplying the main oscillation clock by the PLL multiplier, a subclock formed by dividing the sub-oscillation clock by two, a main CR clock, and a sub-CR clock formed by dividing the sub-CR oscillation by two.

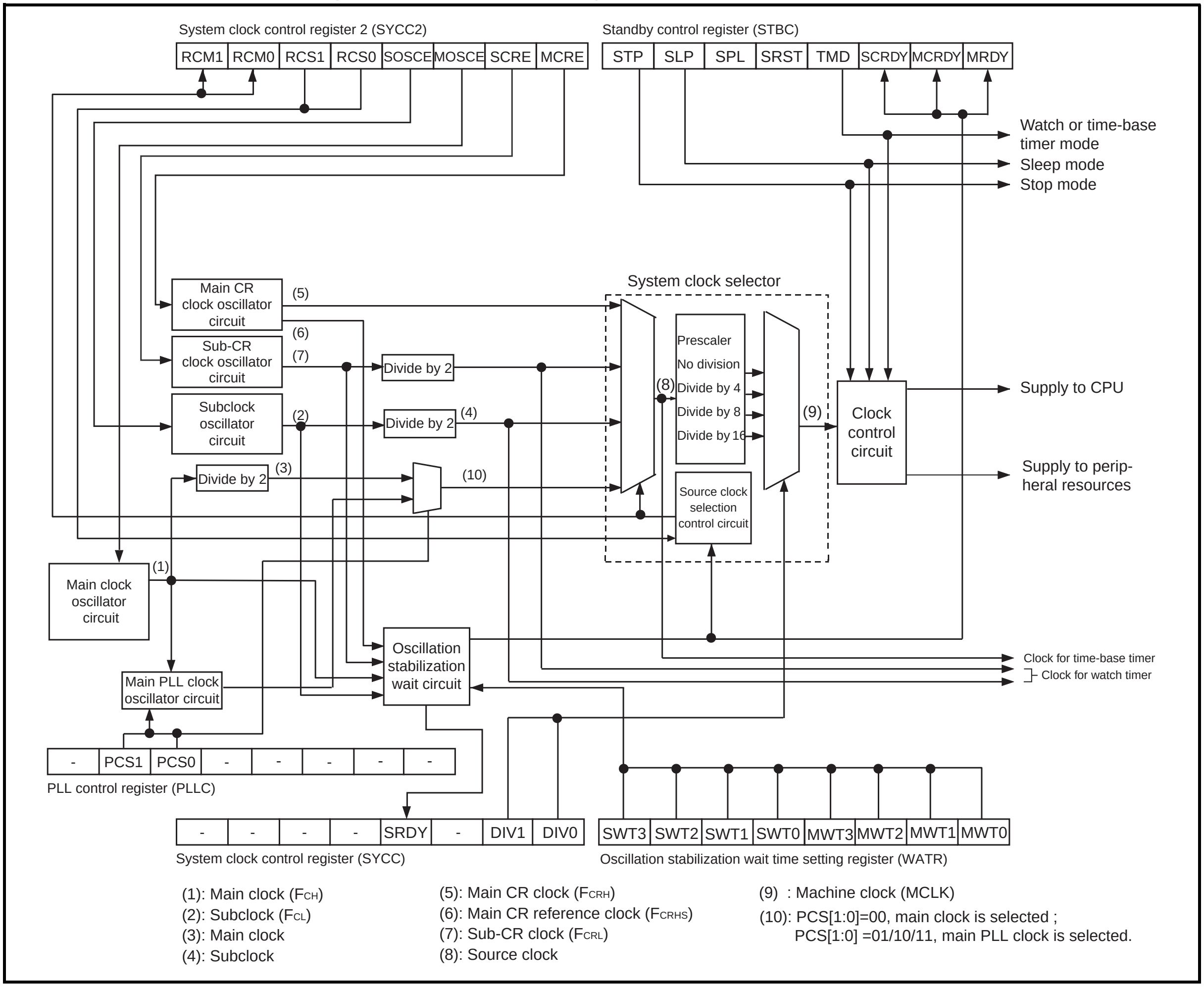
Note:

Only either main clock or main PLL clock can be used at one time. They share the MOSCE bit in the SYCC2 register, and the MRDY bit in the STBC register. The setting of "11_B" in SYCC2:RCS[1:0] and that of "11_B" in SYCC2:RCM[1:0] are applicable to both main clock and main PLL clock.

■ Block Diagrams of Clock Controller

Figure 6.1-1 shows a block diagram of the clock controller.

Figure 6.1-1 Block Diagram of Clock Controller


System clock control register 2 (SYCC2)
RCM1
RCM0
RCS1
RCS0
SOSCE
MOSCE
SCRE
MCRE
Standby control register (STBC)
STP
SLP
SPL
SRST
TMD
SCRDY
MCRDY
MRDY
Watch or time-base timer mode
Sleep mode
Stop mode
Main CR clock oscillator circuit
(5)
(6)
Sub-CR clock oscillator circuit
(7)
Divide by 2
Subclock oscillator circuit
(2)
Divide by 2
(4)
Divide by 2
(3)
(10)
System clock selector
Prescaler
No division
(8)
Divide by 4
Divide by 8
Divide by 16
(9)
Source clock selection control circuit
Clock control circuit
Supply to CPU
Supply to perip-heral resources
Main clock oscillator circuit
(1)
Oscillation stabilization wait circuit
Main PLL clock oscillator circuit
Clock for time-base timer
Clock for watch timer
-
PCS1
PCS0
-
-
-
-
-
PLL control register (PLLC)
-
-
-
-
SRDY
-
DIV1
DIV0
System clock control register (SYCC)
SWT3
SWT2
SWT1
SWT0
MWT3
MWT2
MWT1
MWT0
Oscillation stabilization wait time setting register (WATR)
(1): Main clock (FCH)
(2): Subclock (FCL)
(3): Main clock
(4): Subclock
(5): Main CR clock (FCRH)
(6): Main CR reference clock (FCRHS)
(7): Sub-CR clock (FCRL)
(8): Source clock
(9) : Machine clock (MCLK)
(10): PCS[1:0]=00, main clock is selected ;
PCS[1:0] =01/10/11, main PLL clock is selected.

The clock controller consists of the following blocks:

- Main clock oscillator circuit

  This block is the oscillator circuit for the main clock.

- Subclock oscillator circuit

  This block is the oscillator circuit for the subclock.

- Main PLL clock oscillator circuit

  This block is the oscillator circuit for the main PLL clock.

- Main CR clock oscillator circuit

  This block is the oscillator circuit for the main CR clock.

- Sub-CR clock oscillator circuit

  This block is the oscillator circuit for the sub-CR clock.

- System clock selector

  This block selects a clock according to the clock mode used from the following five types of source clock: main clock, main PLL clock, subclock, main CR clock and sub-CR clock. The source clock selected is divided by the prescaler. The divided clock is called "machine clock", which is to be supplied to the clock control circuit.

- Clock control circuit

  This block controls the supply of the machine clock to the CPU and each peripheral resource according to the standby mode used or oscillation stabilization wait time.

- Oscillation stabilization wait circuit

  This block outputs one of the 14 types of oscillation stabilization signals created by a dedicated timer in the oscillation stabilization wait circuit as the oscillation stabilization signal for the main clock, or one of the 15 types of oscillation stabilization signals created by the same dedicated timer as the oscillation stabilization wait time signal for the subclock.

- System clock control register (SYCC)

  This register is used to select the machine clock divide ratio.

- Standby control register (STBC)

  This register is used to control the transition from RUN state to standby mode, the setting of pin states in stop mode, time-base timer mode, or watch mode, and the generation of software resets.

- PLL control register (PLLC)

  This register is used to set the multiplier of PLL oscillation.

- System clock control register 2 (SYCC2)

This register is used to enable/disable the oscillations of the main clock, main CR clock, subclock, and sub-CR clock, current clock mode display and clock mode selection.

- Oscillation stabilization wait time setting register (WATR)

This register is used to set the oscillation stabilization wait time for the main clock and subclock.

■ Clock Modes

There are five clock modes: main clock mode, main PLL clock mode, main CR clock mode, subclock mode and sub-CR clock mode.

Table 6.1-1 shows the relationships between the clock modes and the machine clock (operating clock for the CPU and peripheral functions).

Table 6.1-1 Clock Modes and Machine Clock Selection

Clock mode	Machine clock
Main clock mode	The machine clock is generated by dividing the main clock by two.
Main PLL clock mode	The machine clock is generated by multiplying the main clock by the PLL multiplier.
Main CR clock mode	The machine clock is generated from the main CR clock.
Subclock mode	The machine clock is generated by dividing the subclock by two.
Sub-CR clock mode	The machine clock is generated by dividing the sub-CR clock by two.

In any clock mode, the frequency of a selected clock can be divided. In addition, in a mode in which the main CR clock is used, the clock frequency can also be selected.

■ Peripheral Function not Affected by Clock Mode

The peripheral function listed in the table below is not affected by the clock mode, division, or CR multiplier settings. Table 6.1-2 lists the peripheral function not affected by the clock mode.

Table 6.1-2 Peripheral Function Not Affected by Clock Mode

Peripheral function	Operating clock
Watchdog timer	Main clock or main PLL clock (with time-base timer output selected) Subclock (with watch prescaler output selected)

For some peripheral functions other than the one listed above, the time-base timer or the watch prescaler can be selected as the count clock. Check the description of each peripheral resource for details.

■ Standby Mode

The clock controller selects whether to enable or disable clock oscillation and clock supply to the internal circuitry according to the standby mode selected. With the exception of time-base timer mode and watch mode, the standby mode can be set independently of the clock mode.

Table 6.1-3 shows the relationships between standby modes and clock supply states.

Table 6.1-3 Standby Mode and Clock Supply States

Standby mode	Clock supply state
Sleep mode	Clock supply to the CPU is stopped. As a result, the CPU stops operating, but other peripheral functions continue operating.
Time-base timer mode	Clock signals are only supplied to the time-base timer and the watch prescaler, while the clock supply to other circuits is stopped. As a result, all the functions other than the time-base timer, watch prescaler, external interrupt, and low-voltage detection reset (option) are stopped. The time-base timer mode can be used in main clock (or main PLL clock) mode and main CR clock mode.
Watch mode	Main clock (or main PLL clock) oscillation is stopped. Clock signals are supplied only to the watch prescaler, while clock supply to other circuits is stopped. As a result, all the functions other than the watch prescaler, external interrupt, and low-voltage detection reset (option) are stopped. The watch mode is the standby mode that can be used in subclock mode and sub-CR clock mode.
Stop mode	Main clock (or main PLL clock) oscillation and subclock oscillation are stopped, and clock supply to all circuits is stopped. As a result, all the functions other than external interrupt and low-voltage detection reset (option) are stopped.

Note:

Clocks that are not mentioned in Table 6.1-3 are supplied under particular settings.

For example, with main clock (or main PLL clock) mode being used in stop mode, when SYCC2:SOSCE and SYCC2:SCRE have been set to "1", the watch prescaler operates.

In addition, with the hardware watchdog timer already started, the watchdog timer operates also in standby mode.

■ Combinations of Clock Mode and Standby Mode

Table 6.1-4 and Table 6.1-5 list the combinations of clock mode and standby mode and the respective operating states of different internal circuits with different combinations of clock mode and standby mode.

Table 6.1-4 Combinations of Standby Mode and Clock Mode and Internal Operating States (1)

Function	RUN				Sleep			
	Main clock (or main PLL clock) mode	Main CR clock mode	Subclock mode	Sub-CR clock mode	Main clock (or main PLL clock) mode	Main CR clock mode	Subclock mode	Sub-CR clock mode
Main clock (or main PLL clock)	Operating	Stopped*1	Stopped		Operating	Stopped*1	Stopped	
Main CR clock	Stopped*2	Operating	Stopped		Stopped*2	Operating	Stopped	
Subclock	Operating*3		Operating	Operating*3	Operating*3		Operating	Operating*3
Sub-CR clock	Operating*4		Operating*4	Operating	Operating*4		Operating*4	Operating
CPU	Operating		Operating		Stopped		Stopped	
Flash memory	Operating		Operating		Value held		Value held	
RAM								
I/O ports	Operating		Operating		Output held		Output held	
Time-base timer	Operating		Stopped		Operating		Stopped	
Watch prescaler	Operating*3, *4		Operating		Operating*3, *4		Operating	
External interrupt	Operating		Operating		Operating		Operating	
Hardware watchdog timer	Operating		Operating		Operating*5		Operating*5	
Software watchdog timer	Operating		Operating		Stopped		Stopped	
Low-voltage detection reset	Operating		Operating		Operating		Operating	
Other peripheral functions	Operating		Operating		Operating		Operating	

*1: The main clock (or main PLL clock) operates when the main clock oscillation enable bit in the system clock control register 2 (SYCC2:MOSCE) is set to "1".

*2: The main CR clock operates when main CR clock oscillation enable bit in the system clock control register 2 (SYCC2:MCRE) is set to "1".

*3: The module operates when the subclock oscillation enable bit in the system clock control register 2 (SYCC2:SOSCE) is set to "1".

*4: The module operates when the sub-CR clock oscillation enable bit in the system clock control register 2 (SYCC2:SCRE) is set to "1".

*5: The hardware watchdog timer stops when the hardware watchdog timer is disabled by the non-volatile register in standby mode.

Table 6.1-5 Combinations of Standby Mode and Clock Mode and Internal Operating States (2)

Function	Time-base timer		Watch prescaler		Stop			
	Main clock (or main PLL clock) mode	Main CR clock mode	Subclock mode	Sub-CR clock mode	Main clock (or main PLL clock) mode	Main CR clock mode	Subclock mode	Sub-CR clock mode
Main clock (or main PLL clock)	Operating	Stopped*1	Stopped		Stopped			
Main CR clock	Stopped*2	Operating	Stopped		Stopped			
Subclock	Operating*3		Operating	Operating*3	Operating*3		Stopped	
Sub-CR clock	Operating*4		Operating*4	Operating	Operating*4		Stopped	
CPU	Stopped		Stopped		Stopped			
Flash memory	Value held		Value held		Value held			
RAM								
I/O ports	Output held / Hi-Z		Output held		Output held/Hi-Z			
Time-base timer	Operating		Stopped		Stopped			
Watch prescaler	Operating*3, *4		Operating		Operating*3, 4		Stopped	
External interrupt	Operating		Operating		Operating			
Hardware watchdog timer	Operating*5		Operating*5		Operating*5			
Software watchdog timer	Stopped		Stopped		Stopped			
Low-voltage detection reset	Operating		Operating		Operating			
Other peripheral functions	Stopped		Stopped		Stopped			

*1: The main clock (or main PLL clock) operates when the main clock oscillation enable bit in the system clock control register 2 (SYCC2:MOSCE) is set to "1".

*2: The main CR clock operates when main CR clock oscillation enable bit in the system clock control register 2 (SYCC2:MCRE) is set to "1".

*3: The module operates when the subclock oscillation enable bit in the system clock control register 2 (SYCC2:SOSCE) is set to "1".

*4: The module operates when the sub-CR clock oscillation enable bit in the system clock control register 2 (SYCC2:SCRE) is set to "1".

*5: The hardware watchdog timer stops when the hardware watchdog timer is disabled by the non-volatile register in standby mode.

6.2 Oscillation Stabilization Wait Time

The oscillation stabilization wait time is the time after the oscillator circuit stops oscillation until the oscillator resumes its stable oscillation at its natural frequency. The clock controller obtains the oscillation stabilization wait time after the start of oscillation by counting a specific number of oscillation clock cycles. During the oscillation stabilization wait time, the clock controller stops clock supply to internal circuits.

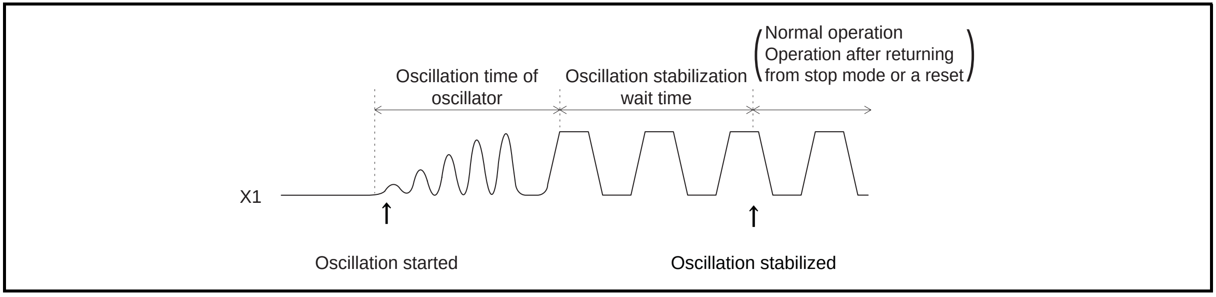
■ Oscillation Stabilization Wait Time

The clock controller obtains the oscillation stabilization wait time after the start of oscillation by counting a specific number of oscillation clock cycles. During the oscillation stabilization wait time, the clock controller stops clock supply to internal circuits.

When the power is switched on, or when a state transition request making the oscillator start from the oscillation stop state is generated due to a change of clock mode caused by a reset, by an interrupt in standby mode or by the software operation, the clock controller automatically waits for the oscillation stabilization wait time of the main clock (or main PLL clock) or of the subclock to elapse before making the clock mode transit to another mode.

Figure 6.2-1 shows how the oscillator operates immediately after starting oscillating.

Figure 6.2-1 Behavior of Oscillator Immediately after Starting Oscillation


Normal operation
Operation after returning
from stop mode or a reset
Oscillation time of oscillator
Oscillation stabilization wait time
X1
Oscillation started
Oscillation stabilized

Oscillation stabilization wait time of main clock (or main PLL clock), subclock, main CR clock, sub-CR clock is counted by using a dedicated counter. The count value can be set in the oscillation stabilization wait time setting register (WATR). Set it in keeping with the oscillator characteristics.

When a power-on reset occurs, the oscillation stabilization wait time is fixed at the initial value.

Table 6.2-1 shows the length of oscillation stabilization wait time.

Table 6.2-1 Oscillation Stabilization Wait Time

Clock	Reset source	Oscillation stabilization wait time
Main clock (or main PLL clock)	Power-on reset	Initial value: $(2^{14}-2)/F_{CH}$ (F_{CH}: main clock frequency)
	Other than power-on reset	Register settings (WATR:MWT3, MWT2, MWT1, MWT0)* *: MWT3-MWT0 are fixed at "1111_B" if the main PLL clock is used.
Subclock	Power-on reset	Initial value: $(2^{15}-2)/F_{CL}$ (F_{CL}: subclock frequency)
	Other than power-on reset	Register settings (WATR:SWT3, SWT2, SWT1, SWT0)

After the oscillation stabilization wait time of the main clock (or main PLL clock) ends, the measurement of the oscillation stabilization wait time of the subclock is started.

■ PLL Clock Oscillation Stabilization Wait Time

As with the oscillation stabilization wait time of the oscillator, the clock controller automatically waits for the PLL clock oscillation stabilization wait time to elapse after a request for state transition from PLL oscillation stopped state to oscillation start is generated via an interrupt in standby mode or a change of clock mode by software. Note that the PLL clock oscillation stabilization wait time changes according to the PLL startup timing.

Table 6.2-2 shows the PLL oscillation stabilization wait time.

Table 6.2-2 PLL Oscillation Stabilization Wait Time

	PLL oscillation stabilization wait time
Main PLL clock	$(2^{14}-2)/F_{CH}$

■ CR Clock Oscillation Stabilization Wait Time

As with the oscillation stabilization wait time of the oscillator, when a state transition request making CR oscillation start from the CR oscillation stop state is generated due to a change of clock mode caused by an interrupt in standby mode or by the software operation, the clock controller automatically waits for the CR oscillation stabilization wait time to elapse.

Table 6.2-3 shows the CR oscillation stabilization wait time.

Table 6.2-3 CR Oscillation Stabilization Wait Time

	CR oscillation stabilization wait time
Main CR clock	$2^8/F_{CRHS}$*
Sub-CR clock	$2^5/F_{CRL}$

*: F_{CRHS}: 1 MHz

■ Oscillation Stabilization Wait Time and Clock Mode/Standby Mode Transition

If state transition occurs, the clock controller automatically waits for the oscillation stabilization wait time to elapse whenever necessary. Depending on the circumstances under which state transition occurs, the clock controller does not wait for the oscillation stabilization wait time to elapse even if state transition occurs.

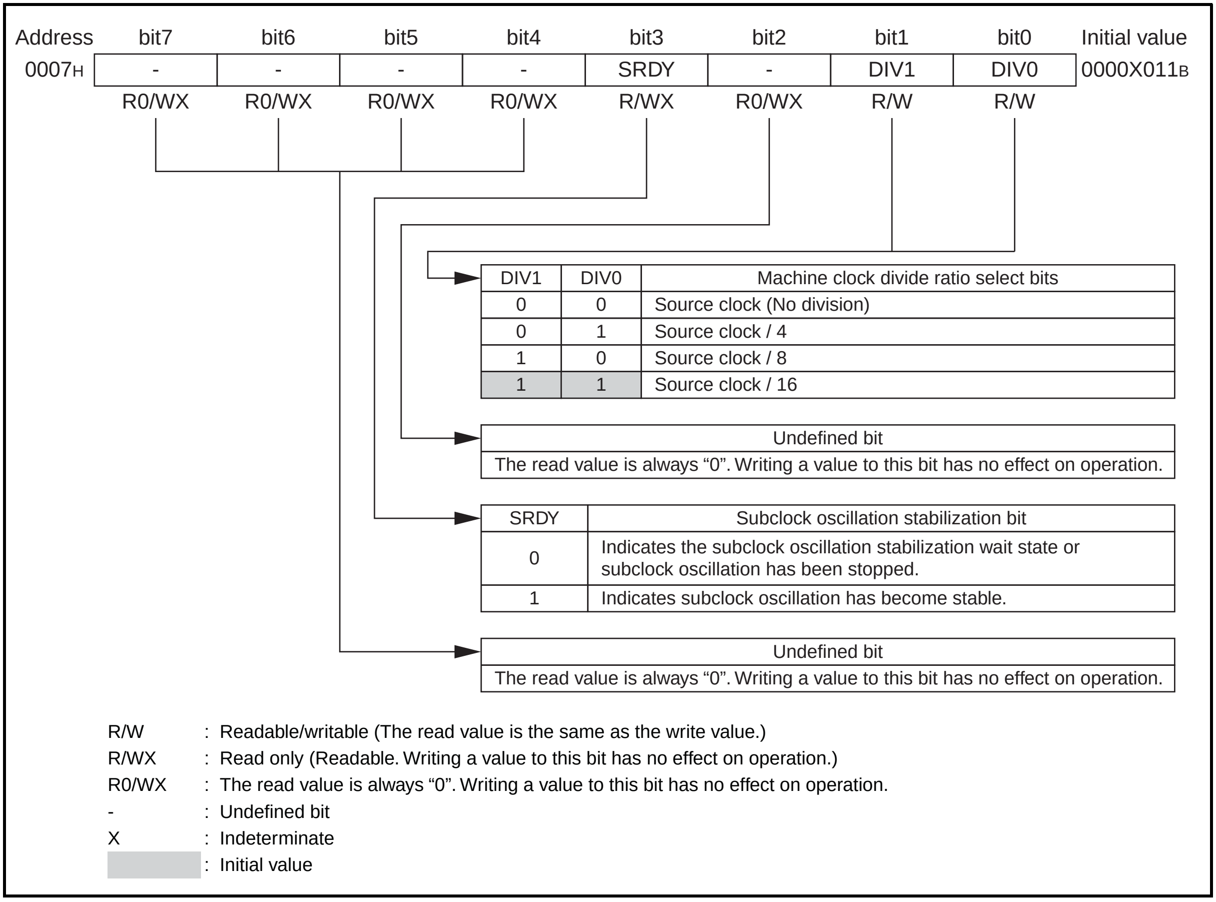
For details on state transition, see "6.8 Clock Modes" and "6.9 Operations in Low-power Consumption Mode (Standby Mode)".

6.3 System Clock Control Register (SYCC)

The system clock control register (SYCC) is used to select the machine clock divide ratio, and indicates the condition of subclock oscillation stabilization.

■ Configuration of System Clock Control Register (SYCC)

Figure 6.3-1 Configuration of System Clock Control Register (SYCC)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0007H	-	-	-	-	SRDY	-	DIV1	DIV0	0000X011B
	R0/WX	R0/WX	R0/WX	R0/WX	R/WX	R0/WX	R/W	R/W	

DIV1	DIV0	Machine clock divide ratio select bits
0	0	Source clock (No division)
0	1	Source clock / 4
1	0	Source clock / 8
▨ 1	▨ 1	Source clock / 16

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

SRDY	Subclock oscillation stabilization bit
0	Indicates the subclock oscillation stabilization wait state or subclock oscillation has been stopped.
1	Indicates subclock oscillation has become stable.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W	:	Readable/writable (The read value is the same as the write value.)
R/WX	:	Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX	:	The read value is always “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
X	:	Indeterminate
▨	:	Initial value

Table 6.3-1 Functions of Bits in System Clock Control Register (SYCC)

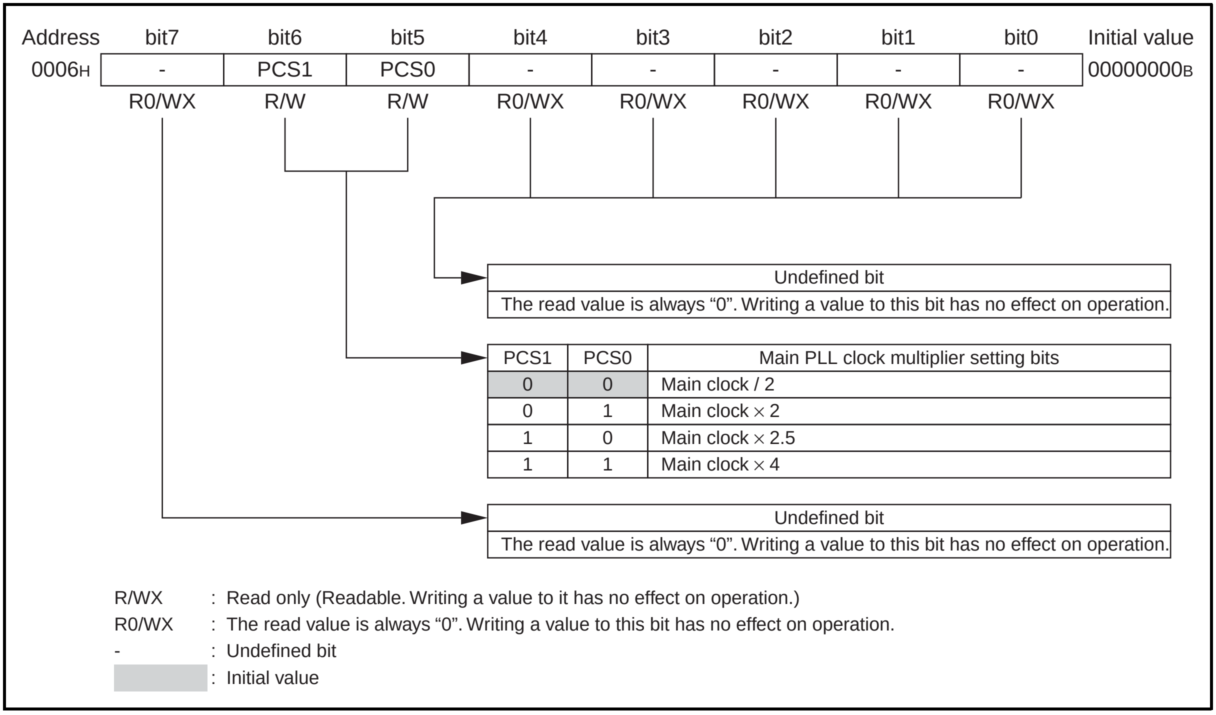
<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7
to
bit4</td><td>Undefined bits</td><td>Their read values are always "0". Writing values to these bits has no effect on operation.</td></tr>
<tr><td>bit3</td><td>SRDY:
Subclock oscillation stabilization bit</td><td>This bit indicates whether subclock oscillation has become stable.
• When the SRDY bit is set to "1", that indicates the oscillation stabilization wait time for the subclock has elapsed.
• When the SRDY bit is set to "0", that indicates that the clock controller is in the subclock oscillation stabilization wait state or that subclock oscillation has been stopped.
This bit is read-only. Writing data to it has no effect on operation.</td></tr>
<tr><td>bit2</td><td>Undefined bit</td><td>The read value is always "0". Writing a value to this bit has no effect on operation.</td></tr>
<tr><td>bit1,
bit0</td><td>DIV1, DIV0:
Machine clock divide ratio select bits</td><td>• These bits select the machine clock divide ratio for the source clock.
• The machine clock is generated from the source clock according to the divide ratio set by these bits.
<table>
<tr><th>DIV1</th><th>DIV0</th><th>Machine clock divide ratio</th></tr>
<tr><td>0</td><td>0</td><td>Source clock (No division)</td></tr>
<tr><td>0</td><td>1</td><td>Source clock / 4</td></tr>
<tr><td>1</td><td>0</td><td>Source clock / 8</td></tr>
<tr><td>1</td><td>1</td><td>Source clock / 16</td></tr>
</table></td></tr>
</table>

6.4 PLL Control Register (PLLC)

The PLL control register (PLLC) controls the main PLL clock multiplier setting.

■ Configuration of PLL Control Register (PLLC)

Figure 6.4-1 Configuration of PLL Control Register (PLLC)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0006H	-	PCS1	PCS0	-	-	-	-	-	00000000B
	R0/WX	R/W	R/W	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

PCS1	PCS0	Main PLL clock multiplier setting bits
0	0	Main clock / 2
0	1	Main clock × 2
1	0	Main clock × 2.5
1	1	Main clock × 4

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
R0/WX : The read value is always “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
▨ : Initial value

Table 6.4-1 Functions of Bits in PLL Control Register (PLLC)

Bit name		Function
bit7	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.
bit6, bit5	PCS1, PCS0: Main PLL clock multiplier setting bits	These bits set the multiplier of the main PLL clock. PCS1 / PCS0 / Main PLL clock multiplier: 0 / 0 / Main clock / 2 0 / 1 / Main clock × 2 1 / 0 / Main clock × 2.5 1 / 1 / Main clock × 4 Note: The value of these bits can only be modified when main PLL clock is stopped. Therefore, they are updated only in main CR clock mode, sub-CR clock mode and subclock mode. If there is a transition from main CR clock mode to main PLL mode and a change of PLL clock multiplier, the MOSCE bit in SYCC2 is not allowed to be set to "1" until the PLLC register is set.
bit4 to bit0	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.

6.5 Oscillation Stabilization Wait Time Setting Register (WATR)

This register is used to set the oscillation stabilization wait time.

■ Configuration of Oscillation Stabilization Wait Time Setting Register (WATR)

Figure 6.5-1 Configuration of Oscillation Stabilization Wait Time Setting Register (WATR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0005_H	SWT3	SWT2	SWT1	SWT0	MWT3	MWT2	MWT1	MWT0	11111111_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

MWT3	MWT2	MWT1	MWT0	Number of cycles	Main Oscillation Clock F_{CH} = 4 MHz	
1	1	1	1	2^{14} - 2	$(2^{14} - 2)/F_{CH}$	About 4.10 ms
1	1	1	0	2^{13} - 2	$(2^{13} - 2)/F_{CH}$	About 2.05 ms
1	1	0	1	2^{12} - 2	$(2^{12} - 2)/F_{CH}$	About 1.02 ms
1	1	0	0	2^{11} - 2	$(2^{11} - 2)/F_{CH}$	511.5 µs
1	0	1	1	2^{10} - 2	$(2^{10} - 2)/F_{CH}$	255.5 µs
1	0	1	0	2^{9} - 2	$(2^{9} - 2)/F_{CH}$	127.5 µs
1	0	0	1	2^{8} - 2	$(2^{8} - 2)/F_{CH}$	63.5 µs
1	0	0	0	2^{7} - 2	$(2^{7} - 2)/F_{CH}$	31.5 µs
0	1	1	1	2^{6} - 2	$(2^{6} - 2)/F_{CH}$	15.5 µs
0	1	1	0	2^{5} - 2	$(2^{5} - 2)/F_{CH}$	7.5 µs
0	1	0	1	2^{4} - 2	$(2^{4} - 2)/F_{CH}$	3.5 µs
0	1	0	0	2^{3} - 2	$(2^{3} - 2)/F_{CH}$	1.5 µs
0	0	1	1	2^{2} - 2	$(2^{2} - 2)/F_{CH}$	0.5 µs
0	0	1	0	2^{1} - 2	$(2^{1} - 2)/F_{CH}$	0.0 µs
0	0	0	1	2^{1} - 2	$(2^{1} - 2)/F_{CH}$	0.0 µs
0	0	0	0	2^{1} - 2	$(2^{1} - 2)/F_{CH}$	0.0 µs

SWT3	SWT2	SWT1	SWT0	Number of cycles	Sub-oscillation Clock F_{CL} = 32.768 kHz	
1	1	1	1	2^{15} - 2	$(2^{15} - 2)/F_{CL}$	About 1.00 s
1	1	1	0	2^{14} - 2	$(2^{14} - 2)/F_{CL}$	About 0.5 s
1	1	0	1	2^{13} - 2	$(2^{13} - 2)/F_{CL}$	About 0.25 s
1	1	0	0	2^{12} - 2	$(2^{12} - 2)/F_{CL}$	About 0.125 s
1	0	1	1	2^{11} - 2	$(2^{11} - 2)/F_{CL}$	About 62.44 ms
1	0	1	0	2^{10} - 2	$(2^{10} - 2)/F_{CL}$	About 31.19 ms
1	0	0	1	2^{9} - 2	$(2^{9} - 2)/F_{CL}$	About 15.56 ms
1	0	0	0	2^{8} - 2	$(2^{8} - 2)/F_{CL}$	About 7.75 ms
0	1	1	1	2^{7} - 2	$(2^{7} - 2)/F_{CL}$	About 3.85 ms
0	1	1	0	2^{6} - 2	$(2^{6} - 2)/F_{CL}$	About 1.89 ms
0	1	0	1	2^{5} - 2	$(2^{5} - 2)/F_{CL}$	About 915.5 µs
0	1	0	0	2^{4} - 2	$(2^{4} - 2)/F_{CL}$	About 427.2 µs
0	0	1	1	2^{3} - 2	$(2^{3} - 2)/F_{CL}$	About 183.1 µs
0	0	1	0	2^{2} - 2	$(2^{2} - 2)/F_{CL}$	About 61.0 µs
0	0	0	1	2^{1} - 2	$(2^{1} - 2)/F_{CL}$	0.0 µs
0	0	0	0	2^{1} - 2	$(2^{1} - 2)/F_{CL}$	0.0 µs

R/W : Readable/writable (The read value is the same as the write value.)

▨ : Initial value

Table 6.5-1 Functions of Bits in Oscillation Stabilization Wait Time Setting Register (WATR) (1 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7
to
bit4</td><td>SWT3, SWT2, SWT1, SWT0: Subclock oscillation stabilization wait time select bits</td><td>These bits set the subclock oscillation stabilization wait time.
<table>
<tr><th>SWT3, SWT2, SWT1, SWT0</th><th>Number of cycles</th><th colspan="2">Subclock F_{CL} = 32.768 kHz</th></tr>
<tr><td>1111_B</td><td>$2^{15}-2$</td><td>$(2^{15}-2)/F_{CL}$</td><td>About 1.0 s</td></tr>
<tr><td>1110_B</td><td>$2^{14}-2$</td><td>$(2^{14}-2)/F_{CL}$</td><td>About 0.5 s</td></tr>
<tr><td>1101_B</td><td>$2^{13}-2$</td><td>$(2^{13}-2)/F_{CL}$</td><td>About 0.25 s</td></tr>
<tr><td>1100_B</td><td>$2^{12}-2$</td><td>$(2^{12}-2)/F_{CL}$</td><td>About 0.125 s</td></tr>
<tr><td>1011_B</td><td>$2^{11}-2$</td><td>$(2^{11}-2)/F_{CL}$</td><td>About 62.44 ms</td></tr>
<tr><td>1010_B</td><td>$2^{10}-2$</td><td>$(2^{10}-2)/F_{CL}$</td><td>About 31.19 ms</td></tr>
<tr><td>1001_B</td><td>$2^{9}-2$</td><td>$(2^{9}-2)/F_{CL}$</td><td>About 15.56 ms</td></tr>
<tr><td>1000_B</td><td>$2^{8}-2$</td><td>$(2^{8}-2)/F_{CL}$</td><td>About 7.75 ms</td></tr>
<tr><td>0111_B</td><td>$2^{7}-2$</td><td>$(2^{7}-2)/F_{CL}$</td><td>About 3.85 ms</td></tr>
<tr><td>0110_B</td><td>$2^{6}-2$</td><td>$(2^{6}-2)/F_{CL}$</td><td>About 1.89 ms</td></tr>
<tr><td>0101_B</td><td>$2^{5}-2$</td><td>$(2^{5}-2)/F_{CL}$</td><td>About 915.5 µs</td></tr>
<tr><td>0100_B</td><td>$2^{4}-2$</td><td>$(2^{4}-2)/F_{CL}$</td><td>About 427.2 µs</td></tr>
<tr><td>0011_B</td><td>$2^{3}-2$</td><td>$(2^{3}-2)/F_{CL}$</td><td>About 183.1 µs</td></tr>
<tr><td>0010_B</td><td>$2^{2}-2$</td><td>$(2^{2}-2)/F_{CL}$</td><td>About 61.0 µs</td></tr>
<tr><td>0001_B</td><td>$2^{1}-2$</td><td>$(2^{1}-2)/F_{CL}$</td><td>0.0 µs</td></tr>
<tr><td>0000_B</td><td>$2^{1}-2$</td><td>$(2^{1}-2)/F_{CL}$</td><td>0.0 µs</td></tr>
</table>
The number of cycles in the above table is the minimum subclock oscillation stabilization wait time. The maximum value is the number of cycles in the above table plus $1/F_{CL}$.
Note: Do not modify these bits during subclock oscillation stabilization wait time. Modify them either when the subclock oscillation stabilization bit in the system clock control register (SYCC:SRDY) has been set to "1", or in main clock (or main PLL clock) mode, main CR clock mode or sub-CR clock mode. These bits can also be modified when the subclock is stopped with the subclock oscillation stop bit in the system clock control register 2 (SYCC2:SOSCE) set to "0" in main clock (or main PLL clock) mode, main CR clock mode or sub-CR clock mode.</td></tr>
</table>

Table 6.5-1 Functions of Bits in Oscillation Stabilization Wait Time Setting Register (WATR) (2 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit3
to
bit0</td><td>MWT3, MWT2, MWT1, MWT0:
Main clock oscillation stabilization wait time select bits</td><td>These bits set the main clock oscillation stabilization wait time.
<table>
<tr><th>MWT3, MWT2, MWT1, MWT0</th><th>Number of cycles</th><th colspan="2">Main clock F_{CH} = 4 MHz</th></tr>
<tr><td>1111_B</td><td>$2^{14}-2$</td><td>$(2^{14}-2)/F_{CH}$</td><td>About 4.10 ms</td></tr>
<tr><td>1110_B</td><td>$2^{13}-2$</td><td>$(2^{13}-2)/F_{CH}$</td><td>About 2.05 ms</td></tr>
<tr><td>1101_B</td><td>$2^{12}-2$</td><td>$(2^{12}-2)/F_{CH}$</td><td>About 1.02 ms</td></tr>
<tr><td>1100_B</td><td>$2^{11}-2$</td><td>$(2^{11}-2)/F_{CH}$</td><td>511.5 µs</td></tr>
<tr><td>1011_B</td><td>$2^{10}-2$</td><td>$(2^{10}-2)/F_{CH}$</td><td>255.5 µs</td></tr>
<tr><td>1010_B</td><td>$2^{9}-2$</td><td>$(2^{9}-2)/F_{CH}$</td><td>127.5 µs</td></tr>
<tr><td>1001_B</td><td>$2^{8}-2$</td><td>$(2^{8}-2)/F_{CH}$</td><td>63.5 µs</td></tr>
<tr><td>1000_B</td><td>$2^{7}-2$</td><td>$(2^{7}-2)/F_{CH}$</td><td>31.5 µs</td></tr>
<tr><td>0111_B</td><td>$2^{6}-2$</td><td>$(2^{6}-2)/F_{CH}$</td><td>15.5 µs</td></tr>
<tr><td>0110_B</td><td>$2^{5}-2$</td><td>$(2^{5}-2)/F_{CH}$</td><td>7.5 µs</td></tr>
<tr><td>0101_B</td><td>$2^{4}-2$</td><td>$(2^{4}-2)/F_{CH}$</td><td>3.5 µs</td></tr>
<tr><td>0100_B</td><td>$2^{3}-2$</td><td>$(2^{3}-2)/F_{CH}$</td><td>1.5 µs</td></tr>
<tr><td>0011_B</td><td>$2^{2}-2$</td><td>$(2^{2}-2)/F_{CH}$</td><td>0.5 µs</td></tr>
<tr><td>0010_B</td><td>$2^{1}-2$</td><td>$(2^{1}-2)/F_{CH}$</td><td>0.0 µs</td></tr>
<tr><td>0001_B</td><td>$2^{1}-2$</td><td>$(2^{1}-2)/F_{CH}$</td><td>0.0 µs</td></tr>
<tr><td>0000_B</td><td>$2^{1}-2$</td><td>$(2^{1}-2)/F_{CH}$</td><td>0.0 µs</td></tr>
</table>
The number of cycles in the above table is the minimum main clock oscillation stabilization wait time. The maximum value is the number of cycles in the above table plus $1/F_{CH}$.
Note: Do not modify these bits during main clock oscillation stabilization wait time. Modify them either when the main clock oscillation stabilization bit in the standby control register (STBC:MRDY) has been set to "1", or in main CR clock mode, subclock mode or sub-CR clock mode. These bits can also be modified when the main clock is stopped with the main clock oscillation stop bit in the system clock control register 2 (SYCC2:MOSCE) set to "0" in main CR clock mode, subclock mode or sub-CR clock mode. In main PLL mode, these bits are not usable, and the PLL clock oscillation stabilization wait time is fixed at $(2^{14}-2)/F_{CH}$.</td></tr>
</table>

■ Note on Setting WATR Register

When using the dual operation Flash function of a device not equipped with the low-voltage detection reset, always set the main clock oscillation stabilization wait time to 90 µs or above (set WATR:MWT[3:0] to "1010_B" or above with the main clock frequency F_{CH} being 4 MHz).

The above setting requirement applies to the following products:

MB95F414H/F416H/F418H/F474H/F476H/F478H

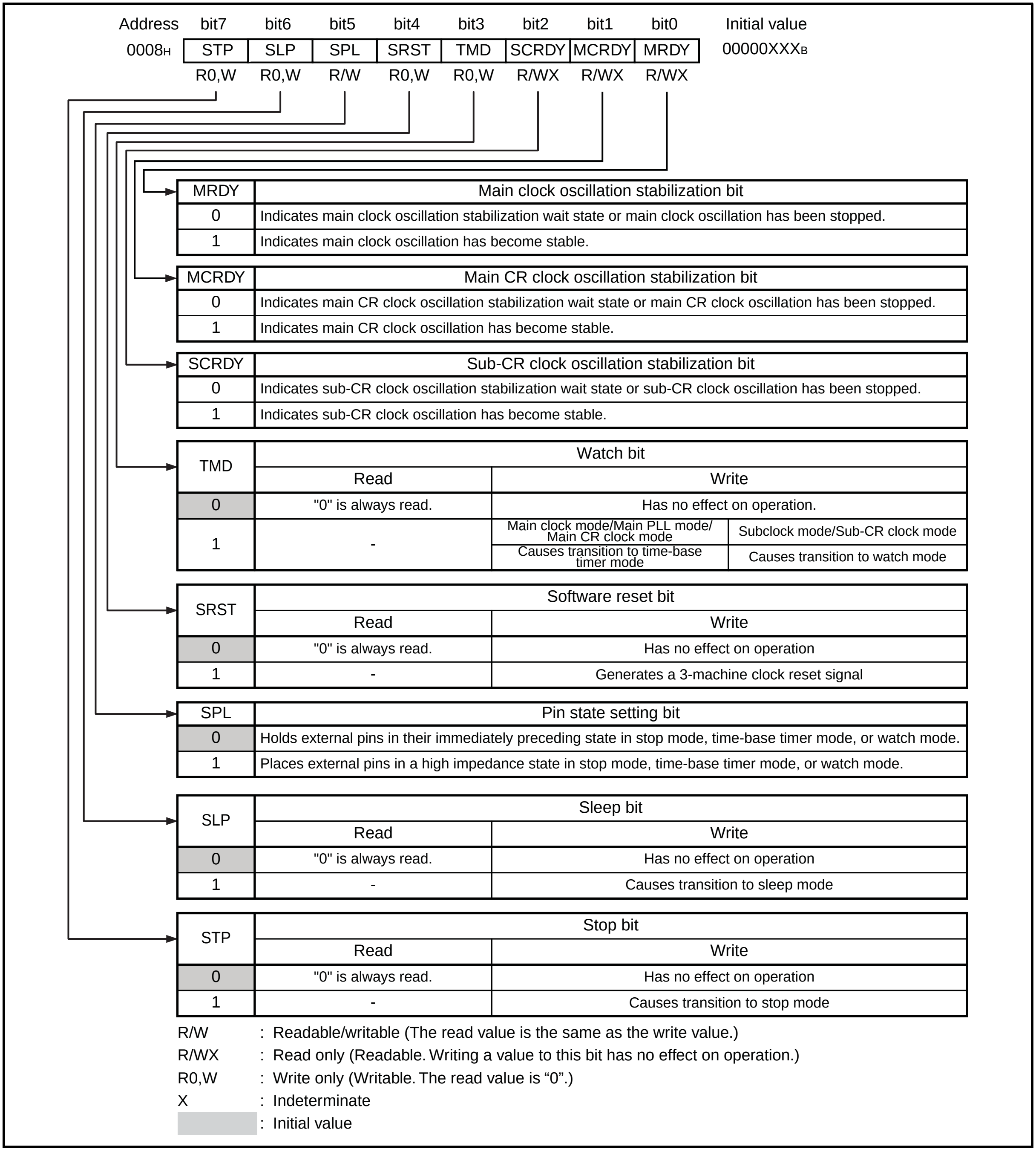
When a flash write/erase operation occurs with the main clock oscillation stabilization wait time having ended within 90 µs, the operation may fail.

6.6 Standby Control Register (STBC)

The standby control register (STBC) is used to control transition from the RUN state to sleep mode, stop mode, time-base timer mode, or watch mode, to set the pin state in stop mode, time-base timer mode, and watch mode, and to control the generation of software resets.

■ Standby Control Register (STBC)

Figure 6.6-1 Standby Control Register (STBC)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0008H	STP	SLP	SPL	SRST	TMD	SCRDY	MCRDY	MRDY	00000XXXB
	R0,W	R0,W	R/W	R0,W	R0,W	R/WX	R/WX	R/WX	

MRDY	Main clock oscillation stabilization bit
0	Indicates main clock oscillation stabilization wait state or main clock oscillation has been stopped.
1	Indicates main clock oscillation has become stable.

MCRDY	Main CR clock oscillation stabilization bit
0	Indicates main CR clock oscillation stabilization wait state or main CR clock oscillation has been stopped.
1	Indicates main CR clock oscillation has become stable.

SCRDY	Sub-CR clock oscillation stabilization bit
0	Indicates sub-CR clock oscillation stabilization wait state or sub-CR clock oscillation has been stopped.
1	Indicates sub-CR clock oscillation has become stable.

TMD	Watch bit		
	Read	Write	
▨0	"0" is always read.	Has no effect on operation.	
1	-	Main clock mode/Main PLL mode/ Main CR clock mode	Subclock mode/Sub-CR clock mode
		Causes transition to time-base timer mode	Causes transition to watch mode

SRST	Software reset bit	
	Read	Write
▨0	"0" is always read.	Has no effect on operation
1	-	Generates a 3-machine clock reset signal

SPL	Pin state setting bit
▨0	Holds external pins in their immediately preceding state in stop mode, time-base timer mode, or watch mode.
1	Places external pins in a high impedance state in stop mode, time-base timer mode, or watch mode.

SLP	Sleep bit	
	Read	Write
▨0	"0" is always read.	Has no effect on operation
1	-	Causes transition to sleep mode

STP	Stop bit	
	Read	Write
▨0	"0" is always read.	Has no effect on operation
1	-	Causes transition to stop mode

R/W	:	Readable/writable (The read value is the same as the write value.)
R/WX	:	Read only (Readable. Writing a value to this bit has no effect on operation.)
R0,W	:	Write only (Writable. The read value is “0”.)
X	:	Indeterminate
▨	:	Initial value

Table 6.6-1 Functions of Bits in Standby Control Register (STBC)

Bit name		Function
bit7	STP: Stop bit	This bit sets the transition to stop mode. **Writing "0":** This bit is meaningless. **Writing "1":** Causes the device to transit to stop mode. When this bit is read, it always returns "0". Note: After an interrupt request is issued, writing "1" to this bit is ignored. For details, see "6.9.1 Notes on Using Standby Mode".
bit6	SLP: Sleep bit	This bit sets the transition to sleep mode. **Writing "0":** This bit is meaningless. **Writing "1":** Causes the device to transit to sleep mode. When this bit is read, it always returns "0". Note: After an interrupt request is issued, writing "1" to this bit is ignored. For details, see "6.9.1 Notes on Using Standby Mode".
bit5	SPL: Pin state setting bit	This bit sets the states of external pins in stop mode, time-base timer mode, and watch mode. **Writing "0":** The state (level) of an external pin is kept in stop mode, time-base timer mode and watch mode. **Writing "1":** An external pin becomes high impedance in stop mode, time-base timer mode and watch mode. (A pin for which connection to a pull-up resistor has been selected in the pull-up register is pulled up.)
bit4	SRST: Software reset bit	This bit sets a software reset. **Writing "0":** Has no effect on operation. **Writing "1":** Generates a 3-machine clock reset signal. When this bit is read, it always returns "0".
bit3	TMD: Watch bit	This bit sets transition to time-base timer mode or watch mode. • Writing "1" to this bit in main clock (or main PLL clock) mode or main CR clock mode causes the device to transit to time-base timer mode. • Writing "1" to this bit in subclock mode or sub-CR clock mode causes the device to transit to watch mode. • Writing "0" to this bit has no effect on operation. • When this bit is read, it always returns "0". Note: After an interrupt request is issued, writing "1" to this bit is ignored. For details, see "6.9.1 Notes on Using Standby Mode".
bit2	SCRDY: Sub-CR clock oscillation stabilization bit	This bit indicates whether sub-CR clock oscillation has become stable. • When the SCRDY bit is set to "1", that indicates the oscillation stabilization wait time for the sub-CR clock has elapsed • When the SCRDY bit is set to "0", that indicates that the clock controller is in the sub-CR clock oscillation stabilization wait state or that sub-CR clock oscillation has been stopped. This bit is read-only. Writing a value to it has no effect on operation.
bit1	MCRDY: Main CR clock oscillation stabilization bit	This bit indicates whether main CR clock oscillation has become stable. • When the MCRDY bit is set to "1", that indicates the oscillation stabilization wait time for the main CR clock has elapsed. • When the MCRDY bit is set to "0", that indicates that the clock controller in the main CR clock oscillation stabilization wait state or that main CR clock stabilization has been stopped. This bit is read-only. Writing a value to it has no effect on operation.
bit0	MRDY: Main clock (or main PLL clock) oscillation stabilization bit	This bit indicates whether main clock (or main PLL clock) oscillation has become stable. • When the MRDY bit is set to "1", that indicates that the oscillation stabilization wait time for the main clock (or main PLL clock) has elapsed. • When the MRDY bit is set to "0", that indicates that the clock controller is in the main clock (or main PLL clock) oscillation stabilization wait state or that main clock (or main PLL clock) oscillation has been stopped. This bit is read-only. Writing a value to it has no effect on operation.

Notes:

- Set the standby mode after making sure that the transition to clock mode has been completed by comparing the values of the clock mode monitor bits (SYCC2:RCM1,RCM0) and clock mode select bits (SYCC2:RCS1,RCS0) in the system clock control register 2.
- If two or more of the following bits, stop bit (STP), sleep bit (SLP), software reset bit (SRST) and watch bit (TMD), are set to "1" together, the order of priority for such bits is as follows:

  (1) Software reset bit (SRST)

  (2) Stop bit (STP)

  (3) Watch bit (TMD)

  (4) Sleep bit (SLP)

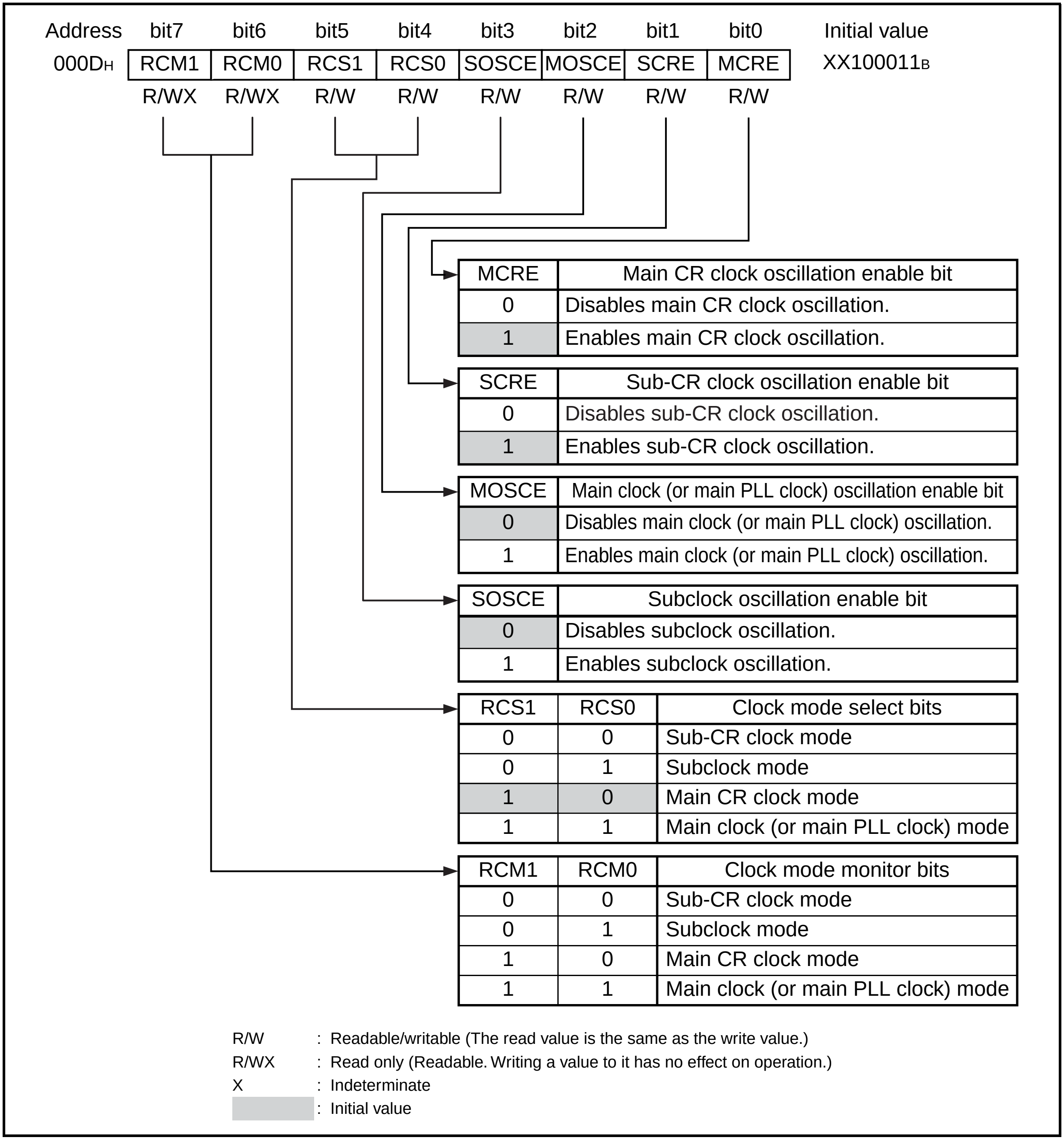
  When released from standby mode, the device returns to the normal operating state.

6.7 System Clock Control Register 2 (SYCC2)

The system clock control register 2 (SYCC2) is used to indicate the current clock mode and switch the clock mode, and control subclock, sub-CR clock, main clock (or main PLL clock), main CR clock oscillations.

■ Configuration of System Clock Control Register 2 (SYCC2)

Figure 6.7-1 Configuration of System Clock Control Register 2 (SYCC2)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
000D$_H$	RCM1	RCM0	RCS1	RCS0	SOSCE	MOSCE	SCRE	MCRE	XX100011$_B$
	R/WX	R/WX	R/W	R/W	R/W	R/W	R/W	R/W	

MCRE	Main CR clock oscillation enable bit
0	Disables main CR clock oscillation.
1	Enables main CR clock oscillation.

SCRE	Sub-CR clock oscillation enable bit
0	Disables sub-CR clock oscillation.
1	Enables sub-CR clock oscillation.

MOSCE	Main clock (or main PLL clock) oscillation enable bit
0	Disables main clock (or main PLL clock) oscillation.
1	Enables main clock (or main PLL clock) oscillation.

SOSCE	Subclock oscillation enable bit
0	Disables subclock oscillation.
1	Enables subclock oscillation.

RCS1	RCS0	Clock mode select bits
0	0	Sub-CR clock mode
0	1	Subclock mode
1	0	Main CR clock mode
1	1	Main clock (or main PLL clock) mode

RCM1	RCM0	Clock mode monitor bits
0	0	Sub-CR clock mode
0	1	Subclock mode
1	0	Main CR clock mode
1	1	Main clock (or main PLL clock) mode

R/W : Readable/writable (The read value is the same as the write value.)
R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
X : Indeterminate
▒ : Initial value

Table 6.7-1 Functions of Bits in System Clock Control Register 2 (SYCC2)

Bit name		Function
bit7, bit6	RCM1, RCM0: Clock mode monitor bits	These bits indicate the current clock mode. **"00$_B$":** Indicates sub-CR clock mode. **"01$_B$":** Indicates subclock mode. **"10$_B$":** Indicates main CR clock mode. **"11$_B$":** Indicates main clock (or main PLL clock) mode. These bits are read-only. Writing values to them has no effect on operation.
bit5, bit4	RCS1, RCS0: Clock mode select bits	These bits select the current clock mode. **Writing "00$_B$":** Selects sub-CR clock mode **Writing "01$_B$":** Selects subclock mode **Writing "10$_B$":** Selects main CR clock mode **Writing "11$_B$":** Selects main clock (or main PLL clock) mode
bit3	SOSCE: Subclock oscillation enable bit	This bit enables/disables the subclock. **Writing "0":** Disables subclock oscillation. **Writing "1":** Enables subclock oscillation. • If the RCS bits are set to "01$_B$", this bit is set to "1". • If the RCS or RCM bits are "01$_B$", writing "0" to this bit is ignored, and its value remains unchanged.
bit2	MOSCE: Main clock (or main PLL clock) oscillation enable bit	This bit enables/disables the main clock (or main PLL clock). **Writing "0":** Disables main clock (or main PLL clock) oscillation. **Writing "1":** Enables main clock (or main PLL clock) oscillation. • If the RCS bits are set to "11$_B$", this bit is set to "1". • If the RCS or RCM bits are "11$_B$", writing "0" to this bit is ignored, and its value remains unchanged. • When the RCM bits are modified to other values from "11$_B$", this bit is set to "0". • If the RCM1 bit is "0", writing "1" to this bit is ignored.
bit1	SCRE: Sub-CR clock oscillation enable bit	This bit enables/disables the sub-CR clock. **Writing "0":** Disables sub-CR clock oscillation. **Writing "1":** Enables sub-CR clock oscillation. • If the RCS bits are set to "00$_B$", this bit is set to "1". • If the RCS or RCM bits are "00$_B$", writing "0" to this bit is ignored, and its value remains unchanged. • If the hardware watchdog timer is used, this bit is set to "1".
bit0	MCRE: Main CR clock oscillation enable bit	This bit enables/disables the main CR clock. **Writing "0":** Disables main CR clock oscillation. **Writing "1":** Enables main CR clock oscillation. • If the RCS bits are set to "10$_B$", the bit is set to "1". • If the RCS or RCM bits are "10$_B$", writing "0" to this bit is ignored, and its value remains unchanged. • When the RCM bits are modified to other values from "10$_B$", the bit is set to "0". • If the RCM1 bit is "0", writing "1" to this bit is ignored.

6.8 Clock Modes

There are five clock modes: main clock, main PLL clock mode, subclock mode, main CR clock mode and sub-CR clock mode. Mode switching occurs according to the settings in the system clock control register 2 (SYCC2).

■ Operations in Main Clock (or main PLL Clock) Mode

In main clock (or main PLL clock) mode, the main clock (or the main PLL clock) is used as the machine clock for the CPU and peripheral functions.

The time-base timer operates using the main clock (or the main PLL clock).

The watch prescaler and watch counter operate with the subclock or the sub-CR clock.

While the device is operating in main clock (or main PLL clock) mode, it can be set to transit to one of the following standby mode: sleep mode, stop mode, or time-base timer mode.

After a reset, the device always enters main CR clock mode regardless of the clock mode used before that reset.

■ Operations in Subclock Mode

In subclock mode, main clock (or main PLL clock) oscillation is stopped* and the subclock is used as the machine clock for the CPU and peripheral functions. In this mode, the time-base timer stops as it requires the main clock (or main PLL clock) for operation.

While the device is operating in subclock clock mode, it can be set to transit to one of the following standby mode: sleep mode, stop mode, or watch mode.

■ Operations in Main CR Clock Mode

In main CR clock mode, the main CR clock is used as the machine clock for the CPU and peripheral functions. The time-base timer and the watchdog timer operate using the main CR clock.

The watch prescaler and watch counter operate with the subclock or the sub-CR clock.

While the device is operating in main CR clock mode, it can be set to transit to one of the following standby mode: sleep mode, stop mode, or time-base timer mode.

■ Operations in Sub-CR Clock Mode

In sub-CR clock mode, main clock (or main PLL clock) oscillation is stopped* and the sub-CR clock is used as the machine clock for the CPU and peripheral functions. In this mode, the time-base timer stops as it requires the main clock (or main PLL clock) for operation. The watch prescaler and watch counter operates using the sub-CR clock.

While the device is operating in sub-CR clock mode, it can be set to transit to one of the following standby mode: sleep mode, stop mode, or watch mode.

*: The main clock (or main PLL clock) and the main CR clock are automatically disabled (SYCC2:MOSCE is set to "0" or SYCC2:MCRE is set to "0") when the clock mode transits from main clock (or main PLL clock) mode or main CR clock mode to another clock mode. If the new clock mode is subclock mode or sub-CR clock mode, the main clock (or main PLL clock) and the main CR clock cannot be enabled by writing "1" to SYCC2:MOSCE and "1" to SYCC2:MCRE respectively.

■ Clock Mode State Transition Diagram

There are five clock modes: main clock mode, main PLL clock mode, subclock mode, main CR clock mode and sub-CR clock mode. The device can switch between these modes according to the settings in the system clock control register 2 (SYCC2).

Figure 6.8-1 Clock Mode State Transition Diagram

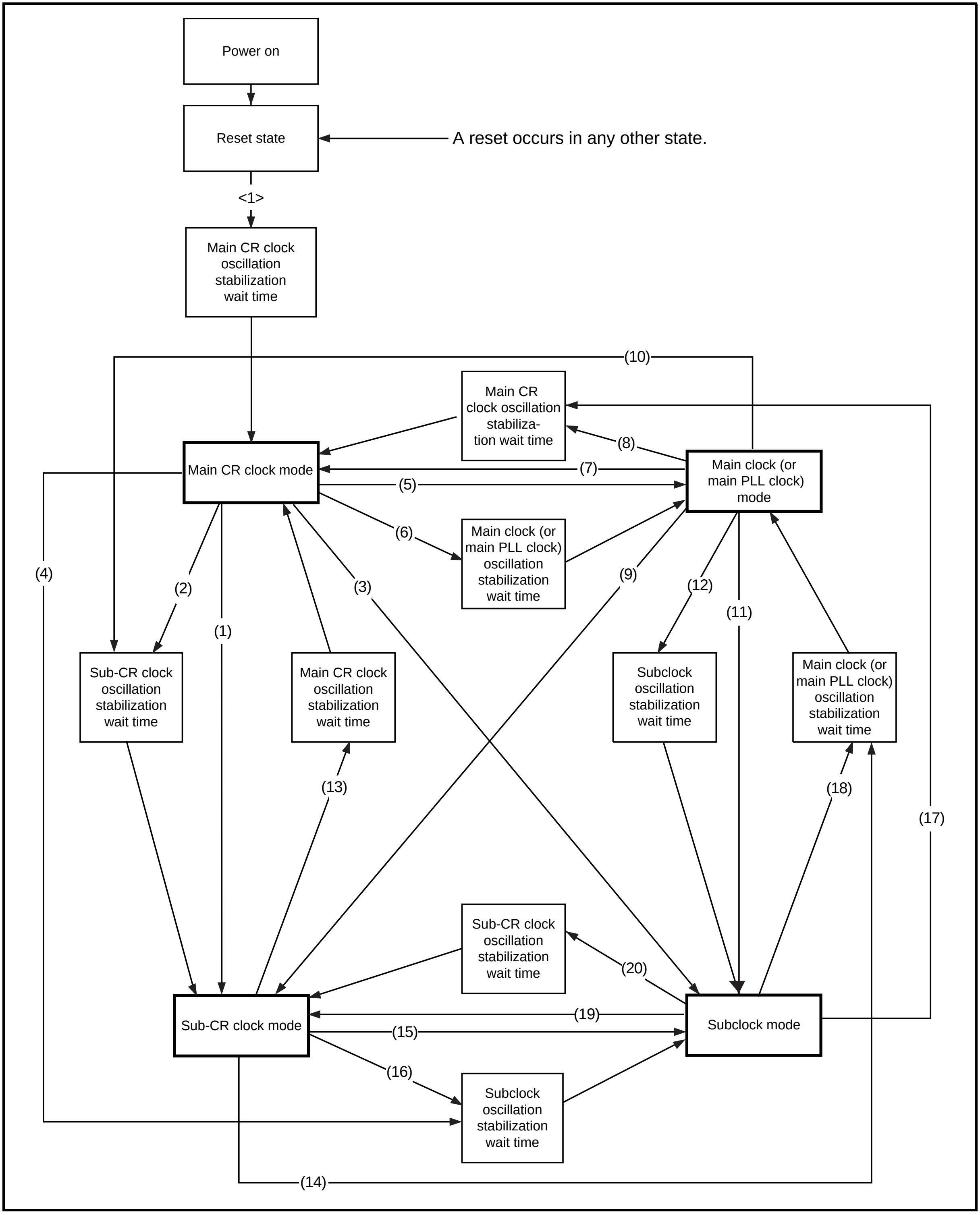

Power on
Reset state
A reset occurs in any other state.
<1>
Main CR clock oscillation stabilization wait time
(10)
Main CR clock oscillation stabiliza- tion wait time
(8)
Main CR clock mode
(7)
(5)
Main clock (or main PLL clock) mode
(6)
Main clock (or main PLL clock) oscillation stabilization wait time
(4)
(2)
(1)
(3)
(9)
(12)
(11)
Sub-CR clock oscillation stabilization wait time
Main CR clock oscillation stabilization wait time
Subclock oscillation stabilization wait time
Main clock (or main PLL clock) oscillation stabilization wait time
(13)
(18)
(17)
Sub-CR clock oscillation stabilization wait time
(20)
Sub-CR clock mode
(19)
(15)
Subclock mode
(16)
Subclock oscillation stabilization wait time
(14)

Table 6.8-1 Clock Mode State Transition Table (1 / 2)

	Current State	Next State	Description
<1>	Reset state	Main CR clock	After a reset, the device waits for the main CR clock oscillation stabilization wait time to elapse and transits to main CR clock mode. Even if that reset is a watchdog reset, software reset or external reset caused in any clock mode, the device waits for the sub-CR clock oscillation stabilization wait time and the main CR clock oscillation stabilization wait time to elapse.
(1)	Main CR clock	Sub-CR clock	The device transits to sub-CR clock mode when the clock mode select bits in the system clock control register 2 (SYCC2:RCS1, RCS0) are set to "00_B". However, if the sub-CR has been stopped according to the setting of the sub-CR clock oscillation enable bit in the system clock control register 2 (SYCC2:SCRE), the device waits for the sub-CR clock oscillation stabilization wait time to elapse before transiting to sub-CR clock mode. In other words, if the sub-CR clock oscillation is enabled in advance and the sub-CR clock oscillation stabilization bit in the standby control register (STBC:SCRDY) is "1", the device transits to sub-CR clock mode immediately after the clock mode select bits (SYCC2:RCS1, RCS0) are set to "00_B".
(2)			
(3)		Subclock	When the clock mode select bits in the system clock control register 2 (SYCC2:RCS1, RCS0) are set to "01_B", the device transits to subclock mode after waiting for the subclock oscillation stabilization wait time. The device does not wait for the subclock oscillation stabilization wait time to elapse if the subclock has been oscillating according to the setting of the subclock oscillation enable bit in the system clock control register 2 (SYCC2:SOSCE). In other words, if subclock oscillation is enabled in advance and the subclock oscillation stabilization bit in the system clock control register (SYCC:SRDY) is "1", the device transits to subclock mode immediately after the clock mode select bits (SYCC2:RCS1, RCS0) are set to "01_B".
(4)			
(5)		Main clock (or main PLL clock)	When the clock mode select bits in the system clock control register 2 (SYCC2:RCS1, RCS0) are set to "11_B", the device transits to main clock (or main PLL clock) mode after waiting for the main clock (or main PLL clock) oscillation stabilization wait time. The device does not wait for the main clock (or main PLL clock) oscillation stabilization wait time to elapse if the main clock (or main PLL clock) has been oscillating according to the setting of the main clock (or main PLL clock) oscillation enable bit in the system clock control register 2 (SYCC2:MOSCE). In other words, if main clock (or main PLL clock) oscillation is enabled in advance and the main clock (or main PLL clock) oscillation stabilization bit in the standby control register (STBC:MRDY) is "1", the device transits to main clock (or main PLL clock) mode immediately after the clock mode select bits (SYCC2:RCS1, RCS0) are set to "11_B".
(6)			

Table 6.8-1 Clock Mode State Transition Table (2 / 2)

	Current State	Next State	Description
(7)	Main clock (or main PLL clock)	Main CR clock	When the clock mode select bits in the system clock control register 2 (SYCC2:RCS1, RCS0) are set to "10_B", the device transits to main CR clock mode after waiting for the main CR clock oscillation stabilization wait time. The device does not wait for the main CR clock oscillation stabilization wait time to elapse if the main CR clock has been oscillating according to the setting of the main CR clock oscillation enable bit in the system clock control register 2 (SYCC2:MCRE). In other words, if main CR clock oscillation is enabled in advance and the main CR clock oscillation stabilization bit in the standby control register (STBC:MCRDY) is "1", the device transits to main CR clock mode immediately after the clock mode select bits (SYCC2:RCS1, RCS0) are set to "10_B".
(8)			
(9)		Sub-CR clock	Same as (1) and (2)
(10)			
(11)		Subclock	Same as (3) and (4)
(12)			
(13)	Sub-CR clock	Main CR clock	When the clock mode select bits in the system clock control register 2 (SYCC2:RCS1, RCS0) are set to "10_B", the device transits to main CR clock mode after waiting for the main CR clock oscillation stabilization wait time.
(14)		Main clock (or main PLL clock)	When the clock mode select bits in the system clock control register 2 (SYCC2:RCS1, RCS0) are set to "11_B", the device transits to main clock (or main PLL clock) mode after waiting for the main clock (or main PLL clock) oscillation stabilization wait time.
(15)		Subclock	Same as (3) and (4)
(16)			
(17)	Subclock	Main CR clock	Same as (13)
(18)		Main clock (or main PLL clock)	Same as (14)
(19)		Sub-CR clock	Same as (1) and (2)
(20)			

6.9 Operations in Low-power Consumption Mode (Standby Mode)

There are four standby modes: sleep mode, stop mode, time-base timer mode and watch mode.

■ Overview of Transiting to and Returning from Standby Mode

There are four standby modes: sleep mode, stop mode, time-base timer mode, and watch mode. The device transits to standby mode according to the settings in the standby control register (STBC).

The device is released from standby mode by an interrupt or a reset. Before transiting to normal operation, the device may wait for the oscillation stabilization wait time to elapse if necessary.

If the clock mode returns from standby mode due to a reset, the device returns to main CR clock mode. If the clock mode returns from standby mode due to an interrupt, before transiting to standby mode, the device returns to the clock mode in which the device was operating.

■ Pin States in Standby Mode

The pin state setting bit (STBC:SPL) of the standby control register can be used to keep the preceding state of an I/O port or a peripheral resource pin before its transition to stop mode, time-base timer mode or watch mode, and to set an I/O port or a peripheral resource pin to high impedance in stop mode, time-base timer mode or watch mode.

See "APPENDIX D Pin States of MB95410H/470H Series" for the states of all pins in standby mode.

6.9.1 Notes on Using Standby Mode

Even if the standby control register (STBC) sets standby mode, transition to standby mode does not occur when an interrupt request has been generated from a peripheral resource. When the device returns from standby mode to the normal operating state in response to an interrupt, the operation that follows varies depending on whether the interrupt request is accepted or not.

■ Insert at least three NOP instructions immediately after a standby mode setting instruction.

The device requires four machine clock cycles before entering standby mode after it is set in the standby control register. During that period, the CPU executes the program. To avoid program execution during this transition to standby mode, insert at least three NOP instructions.

The device still operates normally even if instructions other than NOP instructions are inserted after the instruction that sets the device to transit to standby mode. On this occasion, the following two events may occur. Firstly, an instruction that should be executed after the standby mode is released may be executed before the device transits to standby mode. Secondly, the device may transit to standby mode while an instruction is being executed, and the execution of that same instruction resumes after the device is released from standby mode (increasing the number of instruction execution cycles).

■ Check that clock mode transition has been completed before setting the standby mode.

Before setting the standby mode, ensure that clock-mode transition has been completed by comparing the values of the clock mode monitor bits (SYCC2:RCM1, RCM0) and clock mode select bits (SYCC2:RCS1, RCS0) in the system clock control register 2.

■ An interrupt request may suppress the transition to standby mode.

When the standby mode is set with an interrupt request whose interrupt level is higher than "11_B" having been issued, the device ignores the value written to the standby control register and continues executing instructions without transiting to the standby mode set. Even after the interrupt of that interrupt request is processed, the device does not transit to the standby mode set.

The same operations are executed when interrupts are disabled by the interrupt enable flag (CCR:I) and the interrupt level bits (CCR:IL1, IL0) of the condition code register of the CPU.

■ The standby mode is also released when the CPU rejects interrupts.

When an interrupt request whose interrupt level is higher than "11_B" is issued in standby mode, the device is released from standby mode, regardless of the settings of the interrupt enable flag (CCR:I) and the interrupt level bits (CCR:IL1, IL0) of the condition code register (CCR) of the CPU.

After being released from standby mode, the device processes interrupts if interrupts are to be accepted according to the settings of the condition code register (CCR) of the CPU. If interrupts are not to be accepted according to the settings of CCR, the device resumes instruction execution from the instruction following the one executed before the device transits to standby mode.

■ Standby Mode State Transition Diagram

Figure 6.9-1 shows a standby mode state transition diagram.

Figure 6.9-1 Standby Mode State Transition Diagram

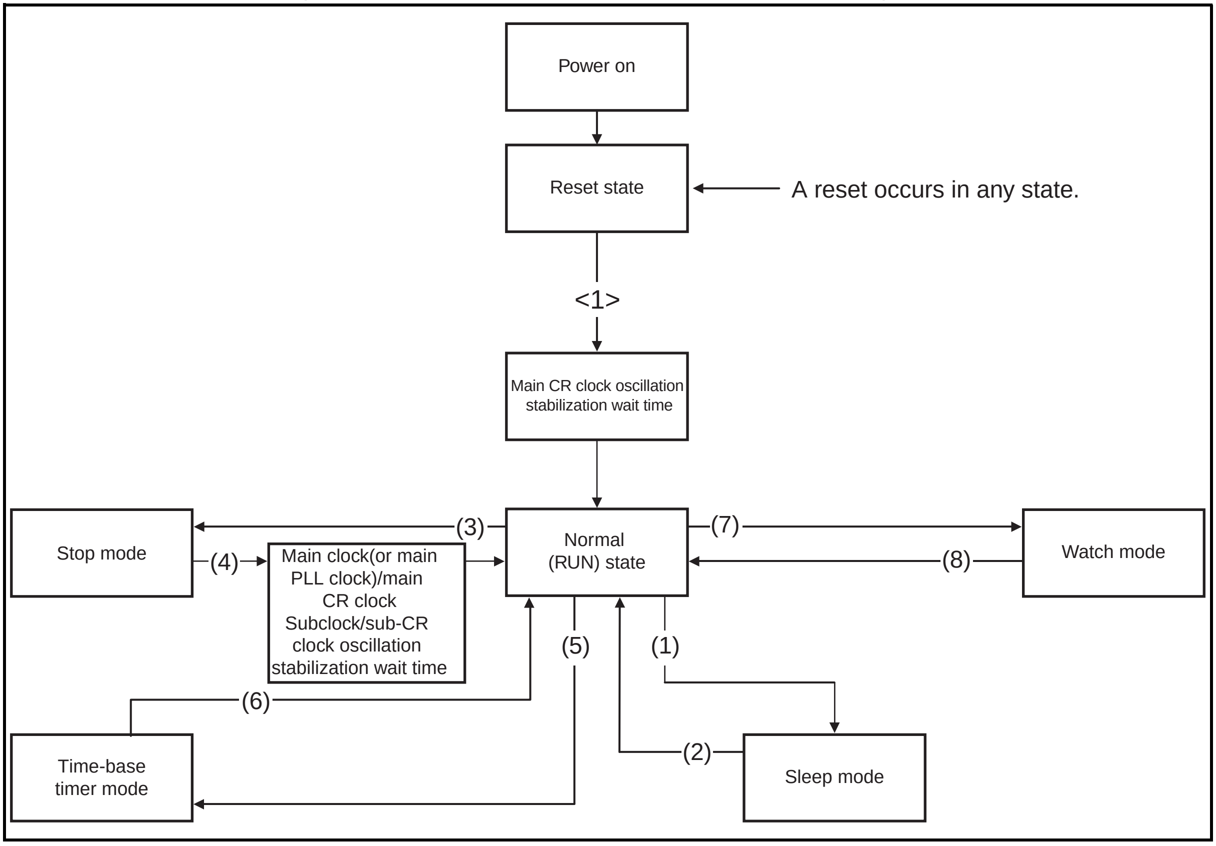

Power on
Reset state
A reset occurs in any state.
<1>
Main CR clock oscillation stabilization wait time
Stop mode
(3)
(4)
Main clock(or main PLL clock)/main CR clock Subclock/sub-CR clock oscillation stabilization wait time
Normal (RUN) state
(7)
(8)
Watch mode
(5)
(1)
(6)
Time-base timer mode
(2)
Sleep mode

Table 6.9-1 State Transition Table (Transitions to and from Standby Modes)

	State Transition	Description
<1>	Normal operation after reset state	After a reset, the device transits to main CR clock mode. If the reset that has occurred is a power-on reset, a watchdog reset, a software reset, or an external reset, the device always waits for the main CR clock oscillation stabilization wait time and the sub-CR clock oscillation stabilization wait time to elapse.
(1)	Sleep mode	The device transits to sleep mode when "1" is written to the sleep bit in the standby control register (STBC:SLP).
(2)		The device returns to the RUN state in response to an interrupt from a peripheral resource.
(3)	Stop mode	The device transits to stop mode when "1" is written to the stop bit in the standby control register (STBC:STP).
(4)		In response to an external interrupt, after waiting for the elapse of the oscillation stabilization wait time required according to the current clock mode, the device returns to the RUN state.
(5)	Time-base timer mode	The device transits to time-base timer mode when "1" is written to the watch bit in the standby control register (STBC:TMD) in main clock (or main PLL clock) mode or main CR clock mode.
(6)		
(7)	Watch mode	The device transits to watch mode when "1" is written to the watch bit in the standby control register (STBC:TMD) in subclock mode or sub-CR clock mode.
(8)		

6.9.2 Sleep Mode

In sleep mode, the operations of the CPU and watchdog timer are stopped.

■ Operations in Sleep Mode

In sleep mode, the CPU and the operating clock for the watchdog timer are stopped. The CPU retains the contents of registers and RAM existing at the point immediately before the device transits to sleep mode and stops; however, all peripheral functions except the watchdog timer continue operating.

In the case of hardware watchdog timer, if it is enabled in standby mode by the non-volatile register function, in sleep mode, the sub-CR clock does not stop and the hardware watchdog timer operates. For details, see "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION".

● Transition to sleep mode

Writing "1" to the sleep bit in the standby control register (STBC:SLP) causes the device to enter sleep mode.

● Release from sleep mode

A reset or an interrupt from a peripheral function releases the device from sleep mode.

6.9.3 Stop Mode

In stop mode, the main clock (or main PLL clock), the main CR clock and the subclock are stopped.

■ Operations in Stop Mode

In stop mode, the main clock (or main PLL clock), the main CR clock, and the subclock are stopped. In this mode, while retaining the contents of registers and RAM existing at the point immediately before the device transits to stop mode, the device stops all functions except external interrupt and low-voltage detection reset.

In the case of hardware watchdog timer, if it is enabled in standby mode by the non-volatile register function, in stop mode, the sub-CR clock does not stop and the hardware watchdog timer operates. For details, see "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION".

● Transition to stop mode

Writing "1" to the stop bit in the standby control register (STBC:STP) causes the device to transit to stop mode. At that point, if the pin state setting bit in the standby control register (STBC:SPL) is "0", the states of the external pins are kept; if the SPL bit is "1", the states of the external pins become high impedance (a pin is pulled up if the pull-up resistor connection for that pin is selected in the pull-up register).

In main clock (or main PLL clock) mode or main CR clock mode, while the device is waiting for main clock (or main PLL clock) oscillation to stabilize after being released from stop mode by an interrupt, a time-base timer interrupt request may be generated. If the interrupt interval time of the time-base timer is shorter than the main clock (or main PLL clock) oscillation stabilization wait time, it is advisable to prevent any unexpected interrupt from occurring by disabling interrupt requests output from the time-base timer before making the device transit to stop mode

It is also advisable to disable interrupt requests output from the watch prescaler before making the device transit to stop mode from subclock mode or sub-CR clock mode.

● Release from stop mode

The device is released from stop mode by a reset or an external interrupt. In any clock mode, if the hardware watchdog timer is enabled in standby mode by the non-volatile register function, the sub-CR clock does not stop, and the watchdog timer and the watch prescaler operate in stop mode. The device can also be released from stop mode by an interrupt from the watch prescaler. For details, see "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION".

Note:

If the device is released from stop mode by an interrupt, a peripheral function having transited to stop mode during operation resumes operating from the point at which it transited to stop mode. Therefore, some settings of that peripheral function, such as the initial interval time of the interval timer, become undefined. Initialize that peripheral function if necessary after releasing the device from stop mode.

6.9.4 Time-base Timer Mode

In time-base timer mode, only the main clock (or main PLL clock) oscillator, the subclock oscillator, the time-base timer, and the watch prescaler operate. The CPU and the operating clock for peripheral functions are stopped in this mode.

■ Operations in Time-base Timer Mode

The time-base timer mode is a mode in which main clock (or main PLL clock) supply is stopped except the clock supply to the time-base timer. In this mode, while retaining the contents of registers and RAM existing at the point immediately before the device transits to time-base timer mode, the device stops all functions except the time-base timer, external interrupt and low-voltage detection reset.

Subclock oscillation and sub-CR clock oscillation can be enabled or disabled by the subclock oscillation enable bit and the sub-CR clock oscillation enable bit in the system clock control register 2 (SYCC2:SOSCE, SCRE) respectively. If the subclock oscillates, the watch prescaler operates.

In the case of hardware watchdog timer, if it is enabled in standby mode by the non-volatile register function, in time-base timer mode, the sub-CR clock does not stop and the hardware watchdog timer operates. For details, see "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION".

● Transition to time-base timer mode

If the clock mode monitor bits in the system clock control register 2 (SYCC2:RCM1, RCM0) are "10_B" or "11_B", writing "1" to the watch bit in the standby control register (STBC:TMD) causes the device to transit to time-base timer mode.

The device can transit to time-base timer mode only when the clock mode is main clock (or main PLL clock) mode or main CR clock mode.

After the device transits to time-base timer mode, if the pin state setting bit in the standby control register (STBC:SPL) is "0", the states of the external pins are kept; if the SPL bit is "1", the states of the external pins become high impedance (a pin is pulled up if the pull-up resistor connection for that pin is selected in the pull-up register).

● Release from time-base timer mode

The device is released from time-base timer mode by a reset, a time-base timer interrupt, or an external interrupt.

Subclock oscillation and sub-CR clock oscillation can be enabled or disabled by setting the subclock oscillation enable bit (SOSCE) and the sub-CR clock oscillation enable bit (SCRE) in the system clock control register 2 (SYCC2). When the subclock oscillates, the device can be released from time-base timer mode by an interrupt from the watch prescaler.

Note:

If the device is released from time-base timer mode by an interrupt, a peripheral function having transited to time-base timer mode during operation resumes operating from the point at which it transited to time-base timer mode. Therefore, some settings of that peripheral function, such as the initial interval time of the interval timer, become undefined. Initialize that peripheral function if necessary after releasing the device from time-base timer mode.

6.9.5 Watch Mode

In watch mode, only the subclock, the sub-CR clock, the watch prescaler and the LCD controller operate. The CPU and the operating clock for peripheral functions are stopped in this mode.

■ Operations in Watch Mode

In watch mode, while retaining the contents of registers and RAM existing at the point immediately before the device transits to watch mode, the device stops all functions except the watch prescaler, external interrupt and low-voltage detection reset.

In the case of hardware watchdog timer, if it is enabled in standby mode by the non-volatile register function, in watch mode, the sub-CR clock does not stop and the hardware watchdog timer operates. For details, see "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION".

● Transition to watch mode

If the clock mode monitor bits in the system clock control register 2 (SYCC2:RCM1, RCM0) are "00_B" or "01_B", writing "1" to the watch bit in the standby control register (STBC:TMD) causes the device to transit to watch mode.

The device can transit to watch mode only when the clock mode is subclock mode or sub-CR clock mode.

After the device transits to watch mode, if the pin state setting bit in the standby control register (STBC:SPL) is "0", the states of the external pins are kept; if the SPL bit is "1", the states of the external pins become high impedance (a pin is pulled up if the pull-up resistor connection for that pin is selected in the pull-up register).

● Release from watch mode

The device is released from watch mode by a reset, a watch interrupt, or an external interrupt.

Note:

If the device is released from watch mode by an interrupt, a peripheral function having transited to watch mode during operation resumes operating from the point at which it transited to watch mode. Therefore, some settings of that peripheral function, such as the initial interval time of the interval timer, become undefined. Initialize that peripheral function if necessary after releasing the device from watch mode.

6.10 Clock Oscillator Circuit

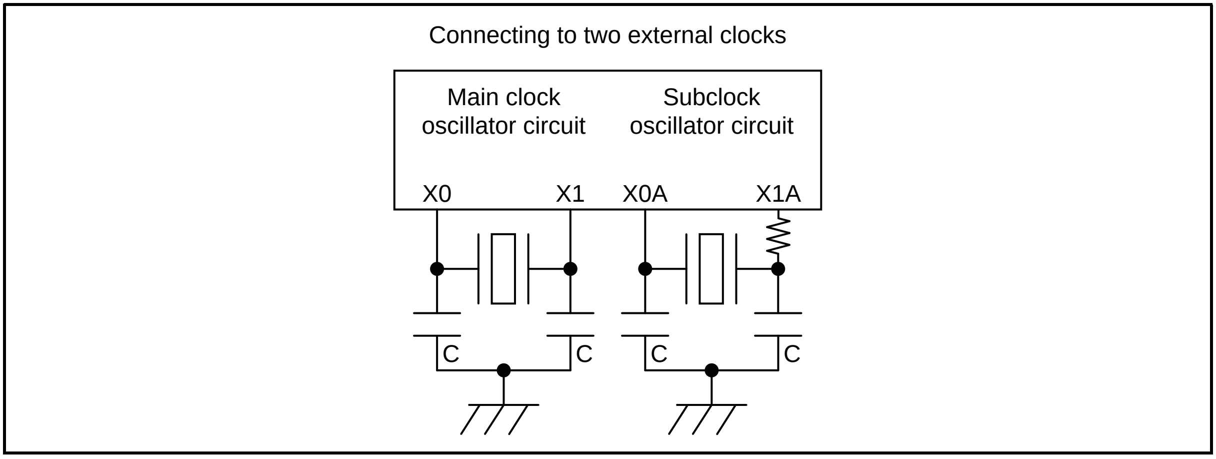
The clock oscillator circuit generates an internal clock with an oscillator connected to the clock oscillation pin or by inputting a clock signal to the clock oscillation pin.

■ Clock Oscillator Circuit

● Using crystal oscillators and ceramic oscillators

Connect crystal oscillators or ceramic oscillators as shown in Figure 6.10-1.

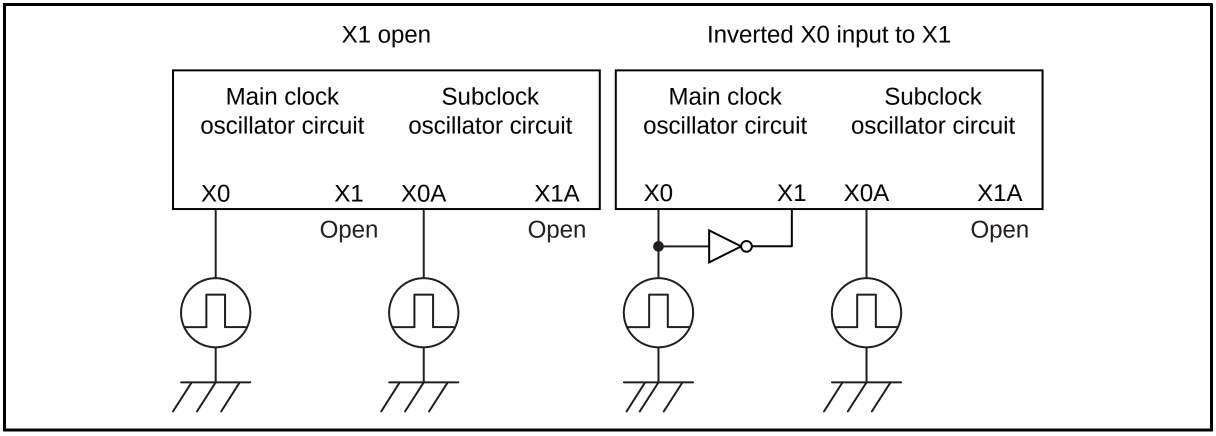
Figure 6.10-1 Sample Connection of Crystal Oscillators and Ceramic Oscillators


Connecting to two external clocks
Main clock oscillator circuit
Subclock oscillator circuit
X0
X1
X0A
X1A
C
C
C
C

● Using external clock

As shown in Figure 6.10-2, connect the external clock to the X0 pin while leaving the X1 pin unconnected or supplying inverted clock of the X0 pin to the X1 pin. (Refer to the data sheet of the MB95410H/470H Series.) To supply clock signals to the subclock from an external clock, connect that external clock to the X0A pin while leaving the X1A pin unconnected.

Figure 6.10-2 Sample Connection of External Clocks


X1 open
Inverted X0 input to X1
Main clock oscillator circuit
Subclock oscillator circuit
X0
X1
X0A
X1A
Open
Open
Main clock oscillator circuit
Subclock oscillator circuit
X0
X1
X0A
X1A
Open

6.11 Overview of Prescaler

The prescaler generates the count clock source to be supplied to various peripheral functions from the machine clock (MCLK) and the count clock output from the time-base timer.

■ Prescaler

The prescaler generates the count clock source to be supplied to various peripheral functions from the machine clock (MCLK) with which the CPU operates and from the count clock (F_{CH}*/2^7, F_{CH}*/2^8, $F_{CRH}/2^6$ or $F_{CRH}/2^7$) output from the time-base timer. The count clock source is a clock whose frequency is divided by the prescaler or a buffered clock. The peripheral functions listed below use the clock whose frequency is divided by the prescaler as the count clock source.

The prescaler has no control register and always operates with the machine clock (MCLK) and the count clock (F_{CH}*/2^7, F_{CH}*/2^8, $F_{CRH}/2^6$ or $F_{CRH}/2^7$) of the time-base timer.

*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used. When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

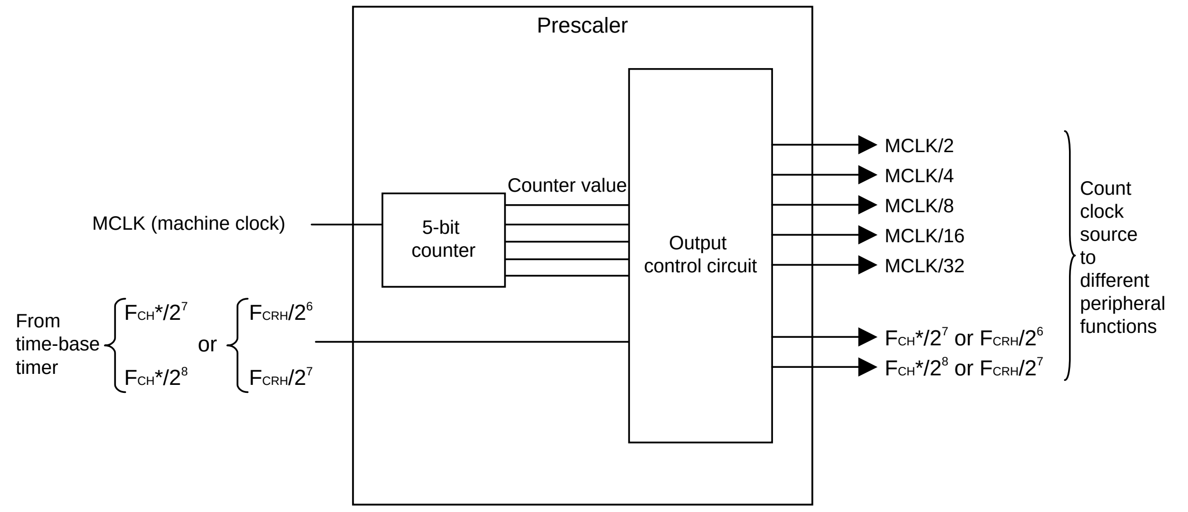
- 8/16-bit composite timer
- 8/10-bit A/D converter

6.12 Configuration of Prescaler

Figure 6.12-1 is the block diagram of the prescaler.

■ Block Diagram of Prescaler

Figure 6.12-1 Block Diagram of Prescaler



MCLK : Machine clock (internal operating frequency)
FCH : Main clock
FCRH : Main CR clock

*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used.
When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

- 5-bit counter

  This counter counts the machine clock (MCLK) and outputs the count value to the output control circuit.

- Output control circuit

  Based on the 5-bit counter value, this circuit supplies clocks generated by dividing the machine clock (MCLK) by 2, 4, 8, 16, or 32 to individual peripheral functions. The circuit also buffers the clock from the time-base timer ($F_{CH}*/2^7$, $F_{CH}*/2^8$, $F_{CRH}/2^6$ or $F_{CRH}/2^7$) and supplies it to peripheral functions.

*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used.
When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

■ Input Clock

The prescaler uses the machine clock, or the output clock of the time-base timer as the input clock.

■ Output Clock

The prescaler supplies clocks to the 8/16-bit composite timer and the 8/10-bit A/D converter.

6.13 Operation of Prescaler

The prescaler generates count clock sources to different peripheral functions.

■ Operation of Prescaler

The prescaler generates count clock sources from a clock whose frequency is generated by dividing the machine clock (MCLK) and from buffered signals from the time-base timer (F_{CH}*/2^7, F_{CH}*/2^8, $F_{CRH}/2^6$ or $F_{CRH}/2^7$), and supplies them to different peripheral functions. The prescaler keeps operating while the machine clock and the clocks from the time-base timer are being supplied.

*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used.
When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

Table 6.13-1 and Table 6.13-2 list the count clock sources generated by the prescaler.

Table 6.13-1 Count Clock Sources Generated by Prescaler (F_{CH})

Count clock source frequency	Frequency (F_{CH} = 20 MHz, MCLK = 10 MHz)	Frequency (F_{CH} = 32 MHz, MCLK = 16 MHz)	Frequency (F_{CH} = 32.5 MHz, MCLK = 16.25 MHz)
MCLK/2	5 MHz	8 MHz	8.125 MHz
MCLK/4	2.5 MHz	4 MHz	4.0625 MHz
MCLK/8	1.25 MHz	2 MHz	2.0313 MHz
MCLK/16	0.625 MHz	1 MHz	1.0156 MHz
MCLK/32	0.3125 MHz	0.5 MHz	0.5078 MHz
$F_{CH}/2^7$	156.25 kHz	250 kHz	253.9 kHz
$F_{CH}/2^8$	78.125 kHz	125 kHz	126.95 kHz

Table 6.13-2 Count Clock Sources Generated by Prescaler (F_{CRH})

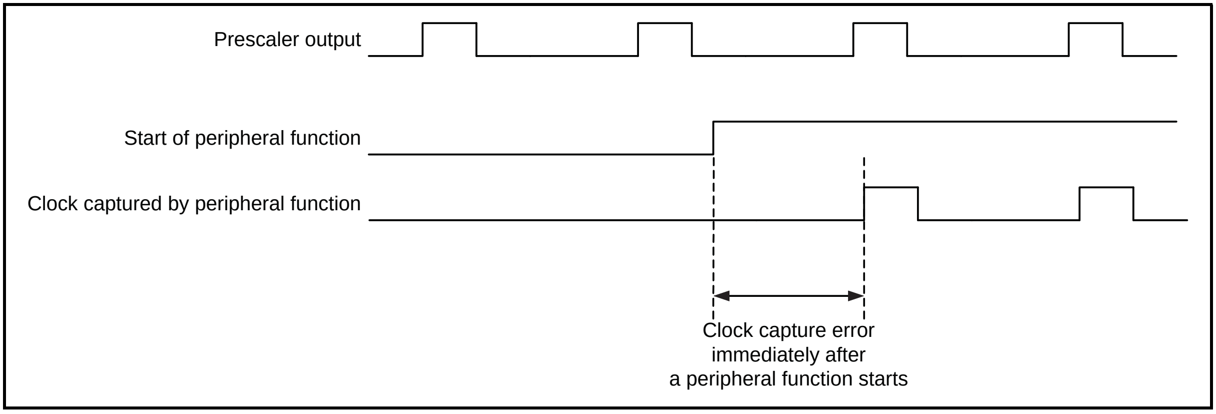
Count clock source frequency	Frequency (F_{CRH} = 1 MHz, MCLK = 1 MHz)	Frequency (F_{CRH} = 8 MHz, MCLK = 8 MHz)	Frequency (F_{CRH} = 10 MHz, MCLK = 10 MHz)	Frequency (F_{CRH} = 12.5 MHz, MCLK = 12.5 MHz)
MCLK/2	500 kHz	4 MHz	5 MHz	6.25 MHz
MCLK/4	250 kHz	2 MHz	2.5 MHz	3.125 MHz
MCLK/8	125 kHz	1 MHz	1.25 MHz	1.5625 MHz
MCLK/16	62.5 kHz	0.5 MHz	0.625 MHz	0.78125 MHz
MCLK/32	31.25 kHz	0.25 MHz	0.3125 MHz	0.390625 MHz
$F_{CRH}/2^6$	15.625 kHz	125 kHz	156.25 kHz	195.3125 kHz
$F_{CRH}/2^7$	7.8125 kHz	62.5 kHz	78.125 kHz	97.65625 kHz

6.14 Notes on Using Prescaler

This section provides notes on using the prescaler.

The prescaler operates with the machine clock and the clock generated from the time-base timer, and keeps operating while those clocks are being supplied. Therefore, in the operation immediately after a peripheral resource is started, an error of up to one cycle of the clock source captured by that peripheral resource will occur, depending on the output value of the prescaler.

Figure 6.14-1 Clock Capture Error Occurring Immediately after a Peripheral Function Starts


Prescaler output
Start of peripheral function
Clock captured by peripheral function
Clock capture error
immediately after
a peripheral function starts

The prescaler count value affects the following peripheral functions:

- 8/16-bit composite timer
- 8/10-bit A/D converter

CHAPTER 7
RESET

This chapter describes the reset operation.

7.1 Reset Operation

When a reset source occurs, the CPU immediately stops the process being executed and enters the reset release wait state. When the reset is released, the CPU reads mode data and the reset vector from the internal ROM (mode fetch). When the power is switched on or when the device is released from a reset in subclock mode, sub-CR clock mode, or stop mode, the CPU performs mode fetch after the oscillation stabilization wait time has elapsed.

■ Reset Sources

There are five reset sources for the reset.

Table 7.1-1 Reset Sources

Reset source	Reset condition
External reset	"L" level is input to the external reset pin.
Software reset	"1" is written to the software reset bit in the standby control register (STBC:SRST).
Watchdog reset	The watchdog timer overflows.
Power-on reset	The power is switched on.
Low-voltage detection reset (optional)	The supply voltage falls below the detection voltage.

● External reset

An external reset is generated if "L" level is input to the external reset pin ($\overline{\text{RST}}$).

An external input reset signal is received asynchronously with the operating clock of the microcontroller via the internal noise filter and then generates an internal reset signal that is synchronized with the machine clock to initialize the internal circuit. Therefore, the operating clock of the microcontroller is necessary for initializing the internal circuit. In order to operate with the external clock, external clock signals must be input. However, the external pins (including I/O ports and peripheral functions) are reset asynchronously. In addition, there is a standard value of the pulse width for external reset input. If the value is below the standard value, a reset signal may not be accepted.

The standard value is shown in the data sheet of this series. Design an external reset circuit that satisfies the standard value.

● Software reset

Writing "1" to the software reset bit of the standby control register (STBC:SRST) generates a software reset.

● Watchdog reset

After the watchdog timer starts, a watchdog reset is generated if the watchdog timer is not cleared within a predetermined period of time.

● Power-on reset

A power-on reset is generated when the power is switched on.

- Low-voltage detection reset (optional)

The low-voltage detection reset circuit is only available on MB95F414K/F416K/F418K/F474K/F476K/F478K.

The low-voltage detection reset circuit generates a reset if the power supply voltage falls below a predetermined level.

The logical function of the low-voltage detection reset is equivalent to that of the power-on reset. All information relating to the power-on reset of this hardware manual also applies to the low-voltage detection reset.

For details of the low-voltage detection reset, see "CHAPTER 26 LOW-VOLTAGE DETECTION RESET CIRCUIT".

■ Reset Time

In the case of a software reset or a watchdog reset, the reset time consists of three machine clock cycles: one machine clock cycle at the machine clock frequency selected before the reset, and two machine clock cycles at the initial machine clock frequency after the reset (1/32 of the main clock frequency). However, the reset time may be extended by the RAM access protection function, which suppresses resets during RAM access, by the machine clock cycle of the frequency selected before the reset. In addition, when in main clock oscillation stabilization standby mode, the reset time is further extended for the oscillation stabilization wait time. Both the external reset and the reset are affected by the RAM access protection function and the main clock oscillation stabilization wait time.

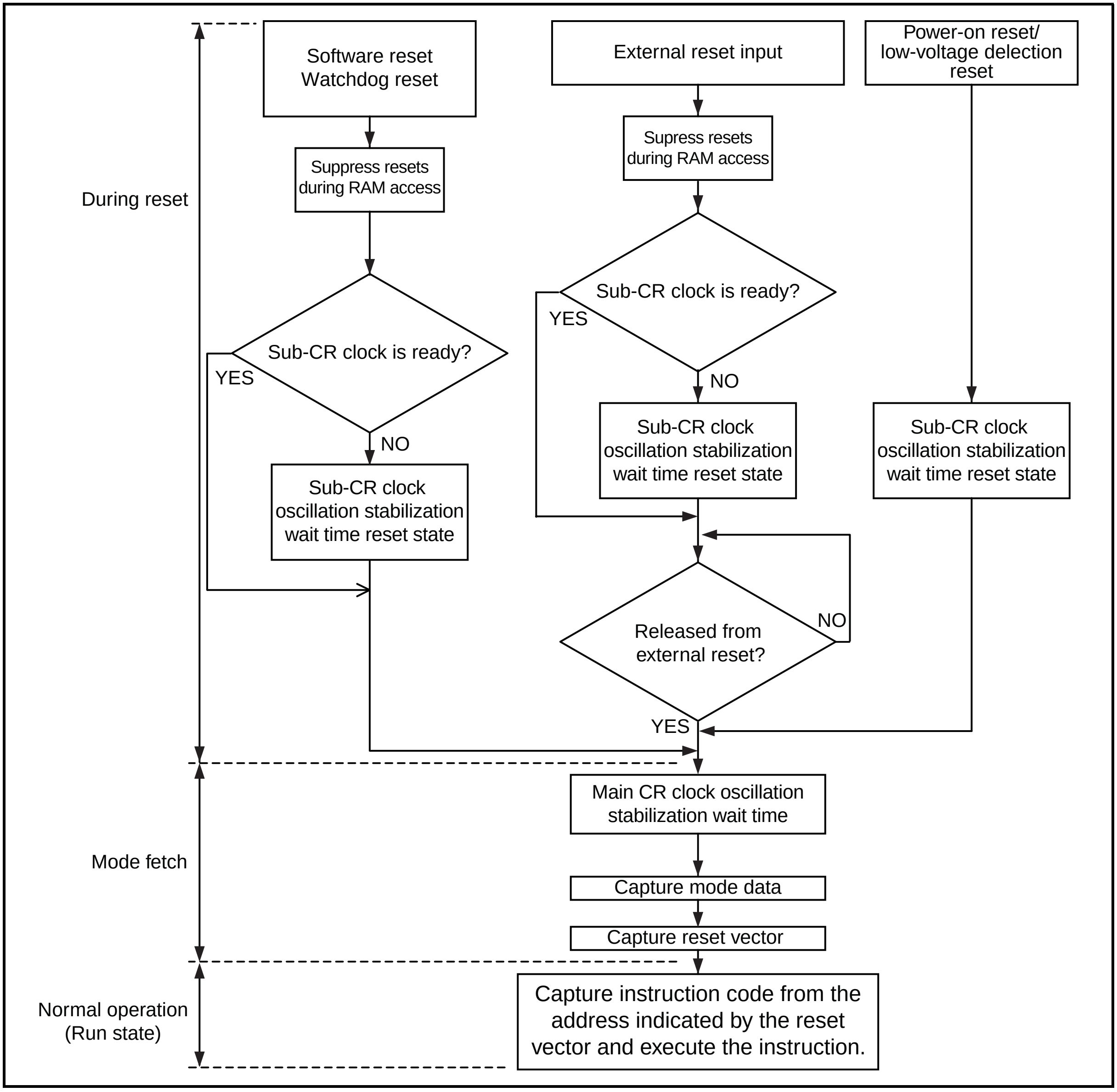
In the case of a power-on reset and a low-voltage detection reset, the reset state continues during the oscillation stabilization wait time.

■ Reset Output

The reset pin outputs "L" level during a reset provided that the reset input function is enabled. However, during an external reset, the reset pin cannot output "L" level. For details of the settings of the reset input function and reset output function, see "CHAPTER 34 SYSTEM CONFIGURATION CONTROLLER".

■ Overview of Reset Operation

Figure 7.1-1 Reset Operation Flow



In any reset, the CPU performs mode fetch after the main CR clock oscillation stabilization wait time elapses.

■ Effect of Reset on RAM Contents

When a reset occurs, the CPU halts the operation of the command currently being executed, and enters the reset state. However, during RAM access execution, in order to protect the RAM access, an internal reset signal synchronized with the machine clock is generated after an RAM access ends. This function prevents a word-data write operation from being interrupted by a reset while data of two bytes is being written.

■ Pin State During a Reset

When a reset occurs, an I/O port or a peripheral resource pin remains high impedance until the setting of that I/O port or that peripheral resource pin by software is executed after the reset is released.

Note:

Connect a pull-up resistor to a pin that becomes high impedance during a reset to prevent the device from malfunctioning.

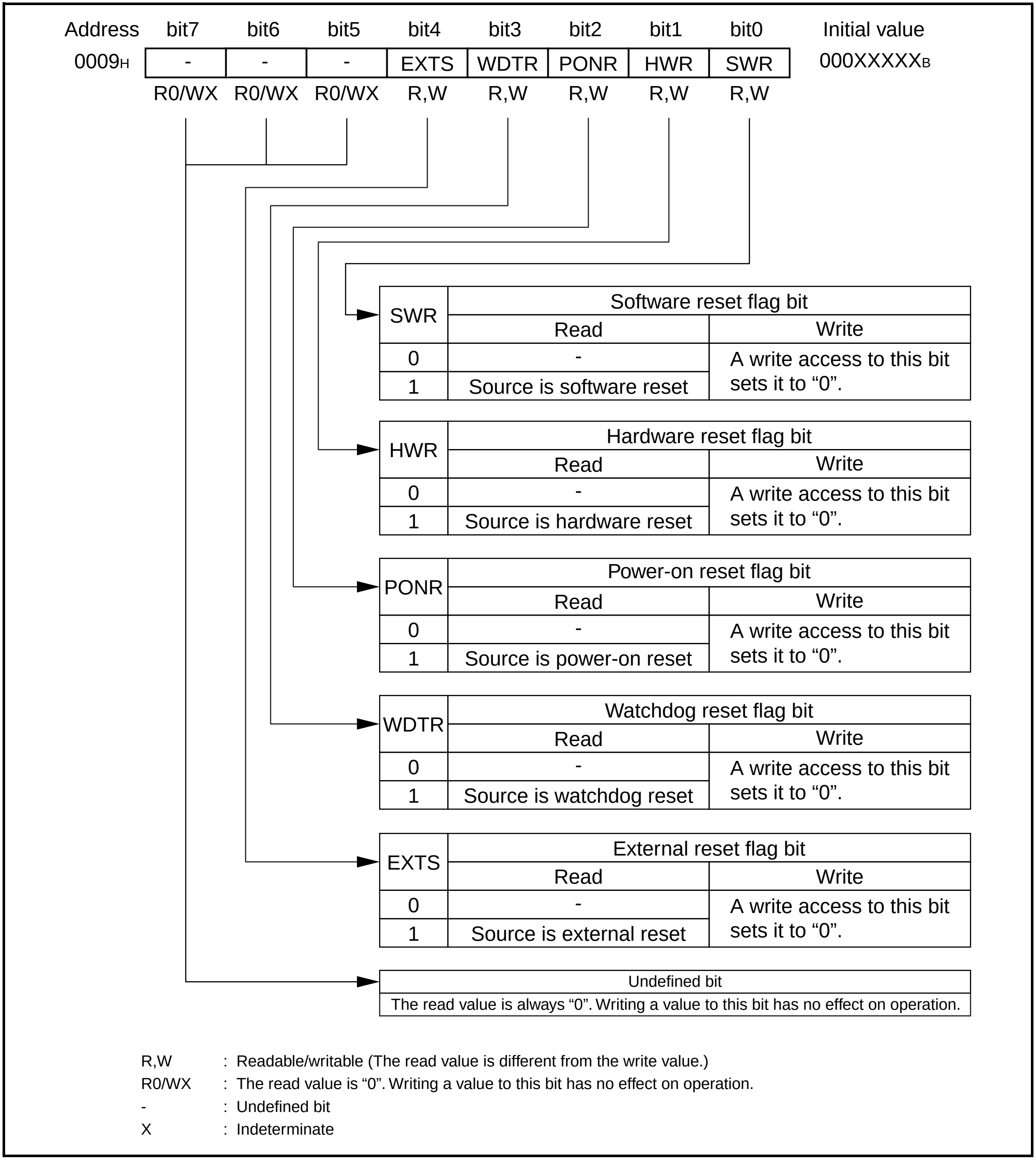
For details of the states of all pins during a reset, see "APPENDIX D Pin States of MB95410H/470H Series".

7.2 Reset Source Register (RSRR)

The reset source register (RSRR) indicates the source of a reset generated.

■ Configuration of Reset Source Register (RSRR)

Figure 7.2-1 Configuration of Reset Source Register (RSRR)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0009H	-	-	-	EXTS	WDTR	PONR	HWR	SWR	000XXXXXB
	R0/WX	R0/WX	R0/WX	R,W	R,W	R,W	R,W	R,W	

SWR	Software reset flag bit	
	Read	Write
0	-	A write access to this bit sets it to “0”.
1	Source is software reset	

HWR	Hardware reset flag bit	
	Read	Write
0	-	A write access to this bit sets it to “0”.
1	Source is hardware reset	

PONR	Power-on reset flag bit	
	Read	Write
0	-	A write access to this bit sets it to “0”.
1	Source is power-on reset	

WDTR	Watchdog reset flag bit	
	Read	Write
0	-	A write access to this bit sets it to “0”.
1	Source is watchdog reset	

EXTS	External reset flag bit	
	Read	Write
0	-	A write access to this bit sets it to “0”.
1	Source is external reset	

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R,W	:	Readable/writable (The read value is different from the write value.)
R0/WX	:	The read value is “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
X	:	Indeterminate

Table 7.2-1 Functions of Bits in Reset Source Register (RSRR)

Bit name		Function
bit7 to bit5	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit4	EXTS: External reset flag bit	When this bit is set to "1", that indicates an external reset has occurred. When any other reset occurs, this bit retains the value that has existed before such reset occurs. • A read access or a write access (writing 0 or 1) to this bit clears it to "0".
bit3	WDTR: Watchdog reset flag bit	When this bit is set to "1", that indicates a watchdog reset has occurred. When any other reset occurs, this bit retains the value that has existed before such reset occurs. • A read access or a write access (writing 0 or 1) to this bit clears it to "0".
bit2	PONR: Power-on reset flag bit	When this bit is set to "1", that indicates a power-on reset or a low-voltage detection reset (optional) has occurred. When any other reset occurs, this bit retains the value that has existed before such reset occurs. • The low-voltage detection reset function is only available on MB95F414K/F416K/F418K/F474K/F476K/F478K. • A read access or a write access (writing 0 or 1) to this bit clears it to "0".
bit1	HWR: Hardware reset flag bit	When this bit is set to "1", that indicates a hardware reset (power-on reset, low-voltage detection reset (optional), external reset or watchdog reset) other than software reset has occurred. Therefore, when any of bit 2 to bit 5 is set to "1", this bit is set to "1" as well. When a software reset occurs, the bit retains the value that has existed before the software reset occurs. • A read access or a write access (writing 0 or 1) to this bit clears it to "0".
bit0	SWR: Software reset flag bit	When this bit is set to "1", that indicates a software reset has occurred. When a hardware reset occurs, the bit retains the value that has existed before the hardware reset occurs. • A read access or a write access (writing 0 or 1) to this bit or a power-on reset clears it to "0".

Note:

Since reading the reset source register clears its contents, save the contents of this register to the RAM before using those contents for calculation.

■ State of Reset Source Register (RSRR)

Table 7.2-2 State of Reset Source Register

Reset source	EXTS	WDTR	PONR	HWR	SWR
Power-on reset	×	×	1	1	0
Low-voltage detection reset (optional)	×	×	1	1	0
Software reset	△	△	△	△	1
Watchdog reset	△	1	△	1	△
External reset	1	△	△	1	△

1: Flag set

△: Previous state kept

×: Indeterminate

EXTS: When this bit is set to "1", that indicates an external reset has occurred.

WDTR: When this bit is set to "1", that indicates a watchdog reset has occurred.

PONR: When this bit is set to "1", that indicates a power-on reset or low-voltage detection reset (optional) has occurred.

HWR: When this bit is set to "1", that indicates one of the following reset has occurred: an external reset, a watchdog reset, a power-on reset or a low-voltage detection reset (optional).

SWR: When this bit is set to "1", that indicates that a software reset has occurred.

7.3 Notes on Using Reset

This section provides notes on using the reset.

■ Notes on Using Reset

● Initialization of registers and bits by reset source

There are registers and bits that are not initialized by a reset source.

- The type of reset source determines which bit in the reset source register (RSRR) is to be initialized.
- The oscillation stabilization wait time setting register (WATR) of the clock controller is initialized only by a power-on reset.

CHAPTER 8

INTERRUPTS

This chapter describes the interrupts.

8.1 Interrupts

This section describes the interrupts.

■ Overview of Interrupts

The New 8FX family has 24 interrupt request inputs for respective peripheral functions, for each of which an interrupt level can be set independently to each other.

When a peripheral resource generates an interrupt request, the interrupt request is output to the interrupt controller. The interrupt controller checks the interrupt level of that interrupt request and then notifies the CPU of the generation of the interrupt. The CPU processes that interrupt according to the interrupt acceptance status. The device is released from standby mode by an interrupt request and resumes executing instructions.

■ Interrupt Requests from Peripheral Functions

Table 8.1-1 lists the interrupt requests of respective peripheral functions. When the CPU receives an interrupt request, it branches to the interrupt service routine with the interrupt vector table address corresponding to the interrupt request as the address of the branch destination.

The priority of each interrupt request in interrupt processing can be set to one of the four levels by the interrupt level setting registers (ILR0 to ILR5).

While an interrupt is being processed in the interrupt service routine, if another interrupt whose interrupt request is of the same level or below the one of the interrupt being processed is generated, it is processed after the current interrupt service routine is completed. In addition, if multiple interrupt requests that are set to the same interrupt level are made, IRQ00 is at the top of the priority order.

Table 8.1-1 Interrupt Requests and Interrupt Vectors

Interrupt request	Vector table address		Bit name in interrupt level setting register	Priority order of interrupt requests of the same level (generated simultaneously)
	Upper	Lower		
IRQ00	FFFA$_H$	FFFB$_H$	L00[1:0]	Highest
IRQ01	FFF8$_H$	FFF9$_H$	L01[1:0]	↑
IRQ02	FFF6$_H$	FFF7$_H$	L02[1:0]	
IRQ03	FFF4$_H$	FFF5$_H$	L03[1:0]	
IRQ04	FFF2$_H$	FFF3$_H$	L04[1:0]	
IRQ05	FFF0$_H$	FFF1$_H$	L05[1:0]	
IRQ06	FFEE$_H$	FFEF$_H$	L06[1:0]	
IRQ07	FFEC$_H$	FFED$_H$	L07[1:0]	
IRQ08	FFEA$_H$	FFEB$_H$	L08[1:0]	
IRQ09	FFE8$_H$	FFE9$_H$	L09[1:0]	
IRQ10	FFE6$_H$	FFE7$_H$	L10[1:0]	
IRQ11	FFE4$_H$	FFE5$_H$	L11[1:0]	
IRQ12	FFE2$_H$	FFE3$_H$	L12[1:0]	
IRQ13	FFE0$_H$	FFE1$_H$	L13[1:0]	
IRQ14	FFDE$_H$	FFDF$_H$	L14[1:0]	
IRQ15	FFDC$_H$	FFDD$_H$	L15[1:0]	
IRQ16	FFDA$_H$	FFDB$_H$	L16[1:0]	
IRQ17	FFD8$_H$	FFD9$_H$	L17[1:0]	
IRQ18	FFD6$_H$	FFD7$_H$	L18[1:0]	
IRQ19	FFD4$_H$	FFD5$_H$	L19[1:0]	
IRQ20	FFD2$_H$	FFD3$_H$	L20[1:0]	
IRQ21	FFD0$_H$	FFD1$_H$	L21[1:0]	
IRQ22	FFCE$_H$	FFCF$_H$	L22[1:0]	↓
IRQ23	FFCC$_H$	FFCD$_H$	L23[1:0]	Lowest

For interrupt sources, see "APPENDIX B Table of Interrupt Sources".

8.1.1 Interrupt Level Setting Registers (ILR0 to ILR5)

The interrupt level setting registers (ILR0 to ILR5) contain 24 pairs of 2-bit data assigned to the interrupt requests of different peripheral functions. Each pair of bits (interrupt level setting bits) is used to set the interrupt level of an interrupt request.

■ Configuration of Interrupt Level Setting Registers (ILR0 to ILR5)

Figure 8.1-1 Configuration of Interrupt Level Setting Registers

Register	Address	bit7 bit6	bit5 bit4	bit3 bit2	bit1 bit0		Initial value
ILR0	00079$_H$	L03[1:0]	L02[1:0]	L01[1:0]	L00[1:0]	R/W	11111111$_B$
ILR1	0007A$_H$	L07[1:0]	L06[1:0]	L05[1:0]	L04[1:0]	R/W	11111111$_B$
ILR2	0007B$_H$	L11[1:0]	L10[1:0]	L09[1:0]	L08[1:0]	R/W	11111111$_B$
ILR3	0007C$_H$	L15[1:0]	L14[1:0]	L13[1:0]	L12[1:0]	R/W	11111111$_B$
ILR4	0007D$_H$	L19[1:0]	L18[1:0]	L17[1:0]	L16[1:0]	R/W	11111111$_B$
ILR5	0007E$_H$	L23[1:0]	L22[1:0]	L21[1:0]	L20[1:0]	R/W	11111111$_B$

The interrupt level setting registers assign a pair of bits to every interrupt request. The values of interrupt level setting bits in these registers represent the priority of an interrupt request (interrupt level: 0 to 3) in interrupt processing.

The interrupt level setting bits are compared with the interrupt level bits in the condition code register (CCR:IL1, IL0).

If the interrupt level of an interrupt request is 3, the CPU ignores that interrupt request.

Table 8.1-2 shows the relationships between interrupt level setting bits and interrupt levels.

Table 8.1-2 Relationships Between Interrupt Level Setting Bits and Interrupt Levels

LXX[1:0]	Interrupt level	Priority
00	0	Highest
01	1	↕
10	2	
11	3	Lowest (No interrupt)

XX:00 to 23 Number of an interrupt request

While the main program is being executed, the interrupt level bits in the condition code register (CCR:IL1, IL0) are "11$_B$".

8.1.2 Interrupt Processing

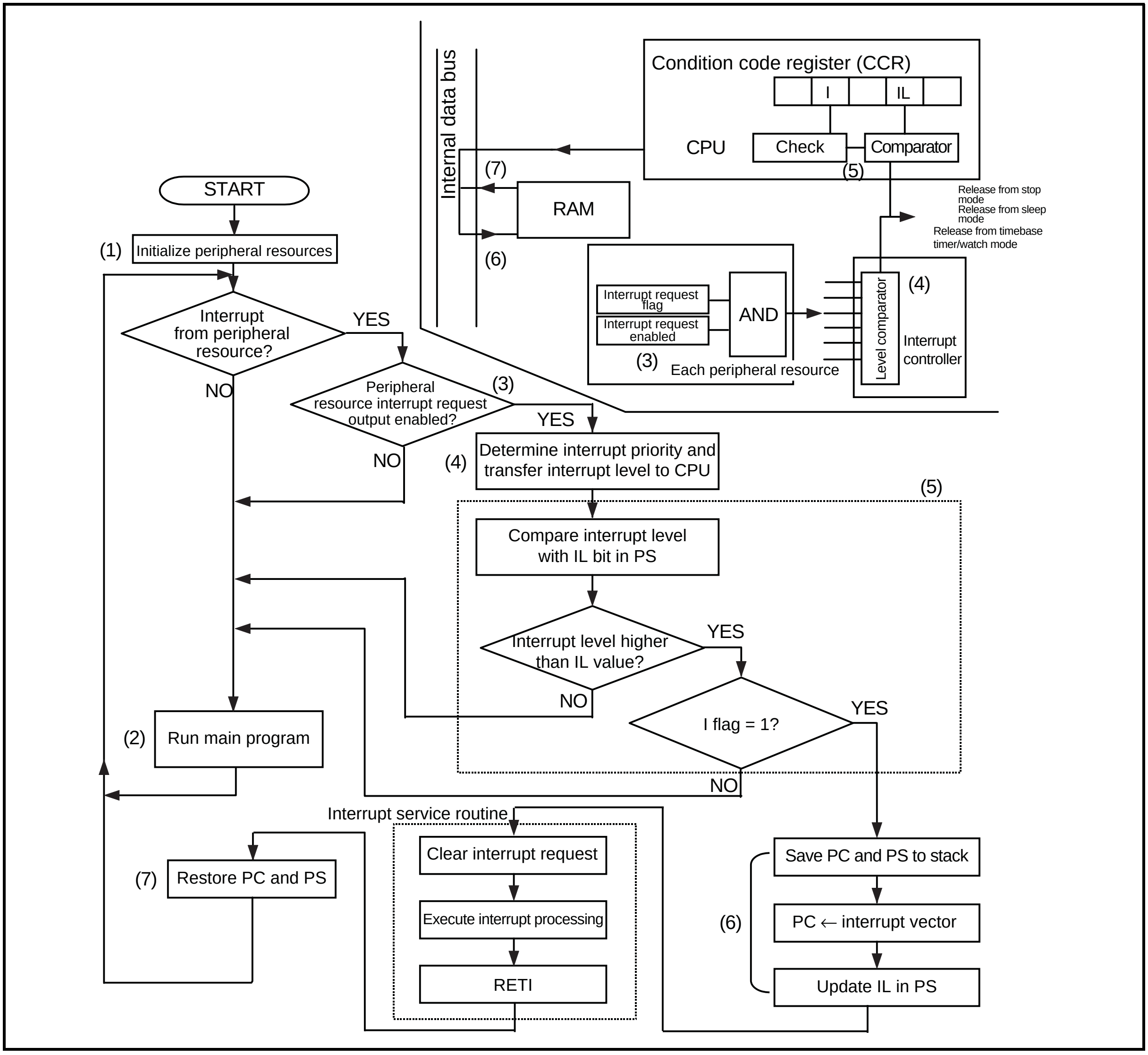
When an interrupt request is made by a peripheral resource, the interrupt controller notifies the CPU of the interrupt level of that interrupt request. When the CPU is ready to accept interrupts, it halts the program it is executing and executes an interrupt service routine.

■ Interrupt Processing

The procedure for processing an interrupt is as follows: the generation of an interrupt source in a peripheral resource, the execution of the main program, the setting of the interrupt request flag bit, the evaluation of the interrupt request enable bit, the evaluation of the interrupt level (ILR0 to ILR5 and CCR:IL1, IL0), the checking for interrupt requests of the same interrupt level made simultaneously, and the evaluation of the interrupt enable flag (CCR:I).

Figure 8.1-2 shows the interrupt processing.

Figure 8.1-2 Interrupt Processing


Internal data bus
Condition code register (CCR)
I
IL
CPU
Check
Comparator
(5)
(7)
RAM
(6)
Release from stop mode
Release from sleep mode
Release from timebase timer/watch mode
START
(1)
Initialize peripheral resources
Interrupt request flag
Interrupt request enabled
AND
(3)
Each peripheral resource
Level comparator
(4)
Interrupt controller
Interrupt from peripheral resource?
YES
NO
Peripheral resource interrupt request output enabled?
(3)
YES
NO
(4)
Determine interrupt priority and transfer interrupt level to CPU
(5)
Compare interrupt level with IL bit in PS
Interrupt level higher than IL value?
YES
NO
I flag = 1?
YES
NO
(2)
Run main program
Interrupt service routine
Clear interrupt request
Execute interrupt processing
RETI
(7)
Restore PC and PS
Save PC and PS to stack
(6)
PC ← interrupt vector
Update IL in PS

(1) All interrupt requests are disabled immediately after a reset. In the peripheral resource initialization program, initialize those peripheral functions that generate interrupts and set their interrupt levels in their respective interrupt level setting registers (ILR0 to ILR5) before starting operating such peripheral functions. The interrupt level can be set to 0, 1, 2, or 3. Level 0 is given the highest priority, and level 1 the second highest. Assigning level 3 to a peripheral resource disables interrupts from that peripheral resource.

(2) Execute the main program (or the interrupt service routine in the case of nested interrupts).

(3) When an interrupt source is generated in a peripheral resource, the interrupt request flag bit for that peripheral resource is set to "1". Provided that the interrupt request enable bit for that peripheral resource has been set to the value that enables interrupts, an interrupt request of that peripheral resource is output to the interrupt controller.

(4) The interrupt controller keeps monitoring interrupt requests from individual peripheral functions and notifies the CPU of the interrupt level having priority over the others among interrupt levels already made. If there are interrupt requests having the same interrupt level, their positions in the priority order are also compared in the interrupt controller.

(5) If the interrupt level received has priority over (smaller interrupt level number) the level set in the interrupt level bits (CCR:IL1, IL0) in the condition code register, the CPU checks the content of the interrupt enable flag (CCR:I), and accepts the interrupt provided that interrupts have been enabled (CCR:I = 1).

(6) The CPU saves the contents of the program counter (PC) and the program status (PS) to the stack, captures the start address of the interrupt service routine from the corresponding interrupt vector table address, modifies the values of the interrupt level bits in the condition code register (CCR:IL1, IL0) to the values of the interrupt level received, then starts executing the interrupt service routine.

(7) Finally, the CPU uses the RETI instruction to restore the values of the program counter (PC) and the program status (PS) from the stack and resumes executing the instruction following the one executed just before the interrupt.

Note:

The interrupt request flag bit for a peripheral resource is not automatically cleared to "0" after an interrupt request is accepted. Therefore, such bit must be cleared to "0" by using a program (writing "0" to the interrupt request flag bit) in the interrupt service routine.

The low-power consumption (standby mode) is released by an interrupt. For details, see "6.9 Operations in Low-power Consumption Mode (Standby Mode)".

8.1.3 Nested Interrupts

Different interrupt levels can be assigned to multiple interrupt requests from peripheral functions in the interrupt level setting registers (ILR0 to ILR5) to process nested interrupts.

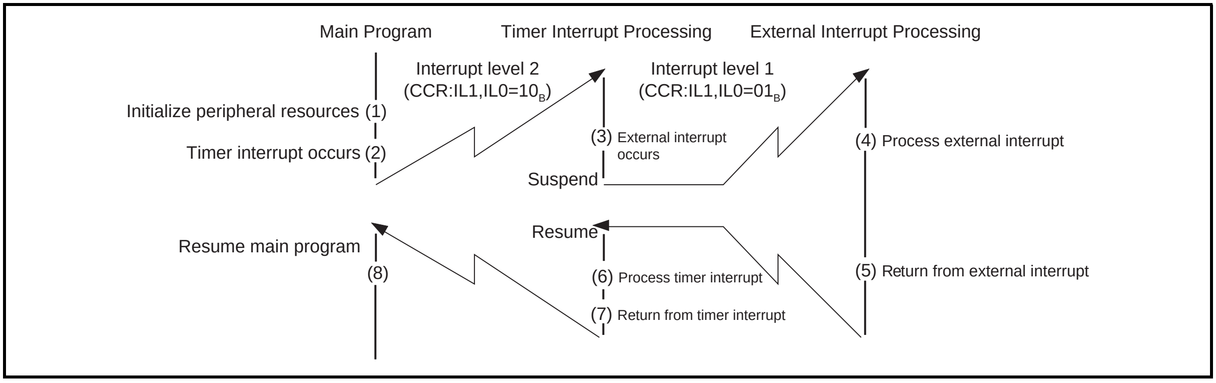
■ Nested Interrupts

During the execution of an interrupt service routine, if another interrupt request whose interrupt level has priority over the interrupt level of the interrupt being processed is made, the CPU suspends the current interrupt processing and accepts the interrupt request given priority. The interrupt level of an interrupt request can be set to 0 to 3. If it is set to 3, the CPU does not accept that interrupt request.

[Example: Nested interrupts]

In the following example of nested interrupts, assuming that the external interrupt is to be given priority over the timer interrupt, the interrupt level of the timer interrupt is set to 2 and that of the external interrupt to 1. If the external interrupt is generated while the timer interrupt is being processed, they are processed as shown in Figure 8.1-3.

Figure 8.1-3 Example of Nested Interrupts


Main Program
Timer Interrupt Processing
External Interrupt Processing
Interrupt level 2
(CCR:IL1,IL0=10B)
Interrupt level 1
(CCR:IL1,IL0=01B)
Initialize peripheral resources (1)
Timer interrupt occurs (2)
(3) External interrupt occurs
Suspend
(4) Process external interrupt
Resume
Resume main program
(8)
(6) Process timer interrupt
(7) Return from timer interrupt
(5) Return from external interrupt

- While the timer interrupt is being processed, the interrupt level bits in the condition code register (CCR: IL1, IL0) hold the same value as that of the interrupt level setting registers (ILR0 to ILR5) corresponding to the timer interrupt (level 2 in this example). If an interrupt request whose interrupt level has priority over the interrupt level of the timer interrupt (level 1 in the example) is made, that interrupt is processed first.
- To temporarily disable nested interrupts processing while the timer interrupt is being processed, disable interrupts by setting the interrupt enable flag in the condition code register (CCR:I) to "0", or set the interrupt level bits (CCR:IL1, IL0) to "00_B".
- After the interrupt processing is completed, if the interrupt return instruction (RETI) is executed, the value of the program counter (PC) and that of the program status (PS) are restored, and the CPU resumes executing the program interrupted. In addition, the values of the condition code register (CCR) return to the ones existing before the interrupt due to the restoration of the value of the program status (PS).

8.1.4 Interrupt Processing Time

Before the CPU enters the interrupt service routine after an interrupt request is made, it needs to wait for the interrupt processing time, which consists of the time between the occurrence of an interrupt request and the end of the execution of the instruction being executed, and the interrupt handling time (the time required to initiate interrupt processing) to elapse. The maximum interrupt processing time is 26 machine clock cycles.

■ Interrupt Processing Time

Before executing the interrupt service routine after an interrupt request is made, the CPU needs to wait for the interrupt request sampling wait time and the interrupt handling time to elapse.

● Interrupt request sampling wait time

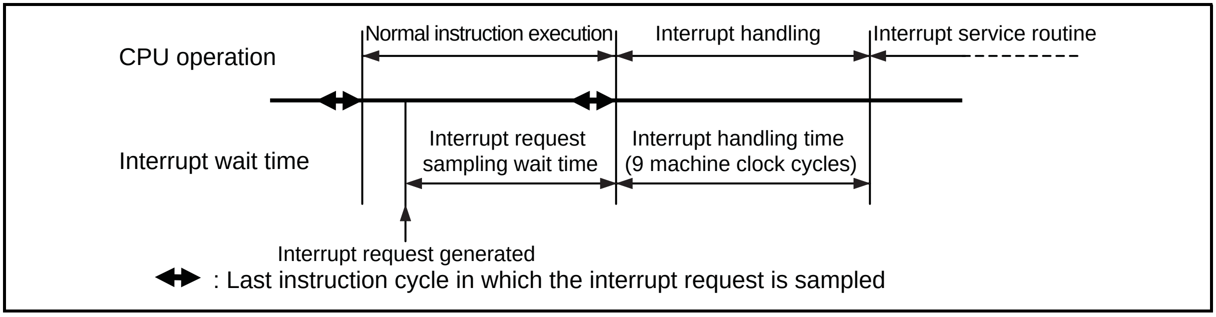
The CPU decides whether an interrupt request has occurred by sampling the interrupt request during the last cycle of an instruction. Therefore, the CPU cannot recognize interrupt requests while executing an instruction. This sampling wait time reaches its maximum when an interrupt request occurs immediately after the CPU starts executing the DIVU instruction, whose execution cycle is the longest (17 machine clock cycles).

● Interrupt handling time

After accepting an interrupt, the CPU requires nine machine clock cycles to perform the following interrupt processing setup:

- Saves the value of the program counter (PC) and that of the program status (PS) to the stack.
- Sets the PC to the start address (interrupt vector) of interrupt service routine.
- Updates the interrupt level bits (PS:CCR:IL1, IL0) in the program status (PS).

Figure 8.1-4 Interrupt Processing Time


CPU operation
Normal instruction execution
Interrupt handling
Interrupt service routine
Interrupt wait time
Interrupt request sampling wait time
Interrupt handling time (9 machine clock cycles)
Interrupt request generated
◆◆ : Last instruction cycle in which the interrupt request is sampled

When an interrupt request occurs immediately after the CPU starts executing the DIVU instruction, whose execution cycle is the longest (17 machine clock cycles), the interrupt processing time spans 26 machine clock cycles.

The span of a machine clock cycle varies depending on the clock mode and main clock speed change (gear function). For details, see "CHAPTER 6 CLOCK CONTROLLER".

8.1.5 Stack Operation During Interrupt Processing

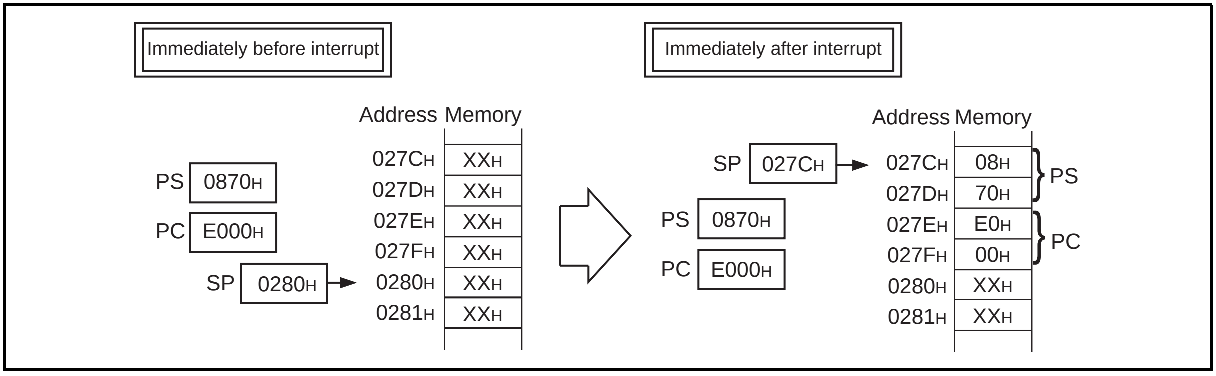
This section describes how the contents of a register are saved and restored during interrupt processing.

■ Stack Operation at the Start of Interrupt Processing

Once the CPU accepts an interrupt, it automatically saves the current value of the program counter (PC) and that of the program status (PS) values to the stack.

Figure 8.1-5 shows the stack operation at the start of interrupt processing.

Figure 8.1-5 Stack Operation at Start of Interrupt Processing



■ Stack Operation after Returning from Interrupt

When the CPU executes the interrupt return instruction (RETI) at the end of interrupt processing, it restores from the stack the value of the program status (PS) first and that of the program counter (PC), which is opposite to the sequence of saving the two values to the stack. After the restoration, both PS and PC return to their states prior to the start of interrupt processing.

Note:

Since the value of the accumulator (A) and that of the temporary accumulator (T) are not automatically saved to the stack, use the PUSHW and POPW instructions to save and restore the values of A and T.

8.1.6 Interrupt Processing Stack Area

The stack area in RAM is used for interrupt processing. The stack pointer (SP) contains the start address of the stack area.

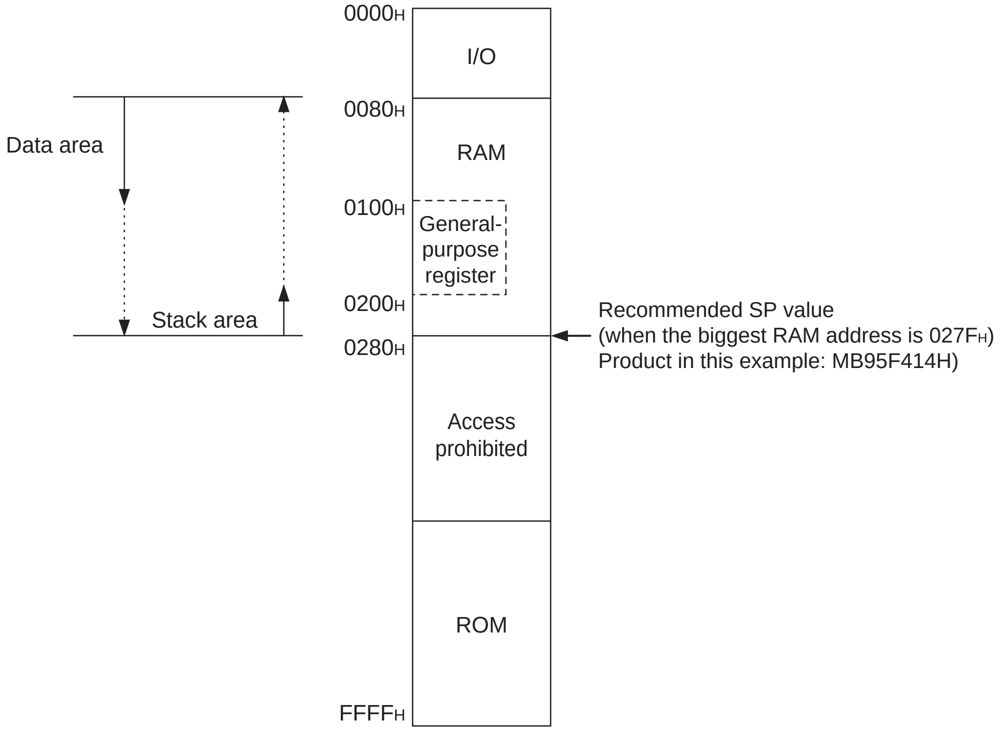
■ Interrupt Processing Stack Area

The stack area is also used for saving and restoring the program counter (PC) when the subroutine call instruction (CALL) or the vector call instruction (CALLV) is executed, and for saving temporarily and restoring register contents by the PUSHW and POPW instructions.

- The stack area is secured on the RAM together with the data area.
- Initialize the stack pointer (SP) so that it indicates the biggest RAM address and make the data area start from the smallest RAM address.

Figure 8.1-6 shows an example of setting the interrupt processing stack area.

Figure 8.1-6 Example of Setting Interrupt Processing Stack Area



Note:

The stack area is utilized by interrupts, sub-routine calls, the PUSHW instruction, etc. in descending of addresses. It is released by return instructions (RETI, RET), the POPW instruction, etc. in ascending order of addresses. If the address value of the stack area used decreases due to nested interrupts or subroutine calls, do not let the stack area overlap the data area and the general-register area, both of which retain other data.

CHAPTER 9

I/O PORTS (MB95410H SERIES)

This chapter describes the functions and operations of the I/O ports.

9.1 Overview of I/O Ports

I/O ports are used to control general-purpose I/O pins.

■ Overview of I/O Ports

The I/O port has functions to output data from the CPU and capture input signals into the CPU with the port data register (PDR). The I/O direction of an individual I/O pin can be set as desired by using the corresponding to that I/O pin in the port direction register (DDR).

Table 9.1-1 lists the registers for each pin.

Table 9.1-1 List of Port Registers (1 / 2)

Register name		Read/Write	Initial value
Port 0 data register	PDR0	R, RM/W	00000000_B
Port 0 direction register	DDR0	R/W	00000000_B
Port 1 data register	PDR1	R, RM/W	00000000_B
Port 1 direction register	DDR1	R/W	00000000_B
Port 2 data register	PDR2	R, RM/W	00000000_B
Port 2 direction register	DDR2	R/W	00000000_B
Port 4 data register	PDR4	R, RM/W	00000000_B
Port 4 direction register	DDR4	R/W	00000000_B
Port 5 data register	PDR5	R, RM/W	00000000_B
Port 5 direction register	DDR5	R/W	00000000_B
Port 6 data register	PDR6	R, RM/W	00000000_B
Port 6 direction register	DDR6	R/W	00000000_B
Port 9 data register	PDR9	R, RM/W	00000000_B
Port 9 direction register	DDR9	R/W	00000000_B
Port A data register	PDRA	R, RM/W	00000000_B
Port A direction register	DDRA	R/W	00000000_B
Port B data register	PDRB	R, RM/W	00000000_B
Port B direction register	DDRB	R/W	00000000_B
Port C data register	PDRC	R, RM/W	00000000_B
Port C direction register	DDRC	R/W	00000000_B
Port E data register	PDRE	R, RM/W	00000000_B
Port E direction register	DDRE	R/W	00000000_B
Port F data register	PDRF	R, RM/W	00000000_B
Port F direction register	DDRF	R/W	00000000_B

Table 9.1-1 List of Port Registers (2 / 2)

Register name		Read/Write	Initial value
Port G data register	PDRG	R, RM/W	00000000_B
Port G direction register	DDRG	R/W	00000000_B
Port 1 pull-up register	PUL1	R/W	00000000_B
Port 2 pull-up register	PUL2	R/W	00000000_B
Port 5 pull-up register	PUL5	R/W	00000000_B
Port G pull-up register	PULG	R/W	00000000_B
A/D input disable register (lower)	AIDRL	R/W	00000000_B
Input level select register	ILSR	R/W	00000000_B

R/W : Readable/writable (The read value is the same as the write value.)

R, RM/W : Readable/writable (The read value is different from the write value. The write value is read by the read-modify-write (RMW) type of instruction.)

9.2 Port 0

**Port 0 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 0 Configuration

Port 0 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 0 data register (PDR0)
- Port 0 direction register (DDR0)
- A/D input disable register lower (AIDRL)
- Input level select register (ILSR)

■ Port 0 Pins

Port 0 has eight I/O pins.

Table 9.2-1 lists the port 0 pins.

Table 9.2-1 Port 0 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P00/INT00/ AN00/UO2	P00: General-purpose I/O	INT00: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN00:Analog input				
		UO2: UART/SIO ch. 2 data output				
P01/INT01/ AN01/SEG36/ UI2	P01: General-purpose I/O	INT01: External interrupt input	Hysteresis/ CMOS/ analog	CMOS/ LCD	-	-
		AN01: Analog input				
		SEG36: LCDC SEG36 output				
		UI2: UART/SIO ch. 2 data input				
P02/INT02/ AN02/SEG35/ UCK2	P02: General-purpose I/O	INT02: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN02: Analog input				
		SEG35: LCDC SEG35 output				
		UCK2: UART/SIO ch. 2 clock I/O				
P03/INT03/ AN03/SEG34/ UO1	P03: General-purpose I/O	INT03: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN03: Analog input				
		SEG34: LCDC SEG34 output				
		UO1: UART/SIO ch. 1 data output				
P04/INT04/ AN04/SEG33/ UI1	P04: General-purpose I/O	INT04: External interrupt input	Hysteresis/ CMOS/ analog	CMOS/ LCD	-	-
		AN04: Analog input				
		SEG33: LCDC SEG33 output				
		UI1: UART/SIO ch. 1 data input				
P05/INT05/ AN05/SEG32/ UCK1	P05: General-purpose I/O	INT05: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN05: Analog input				
		SEG32: LCDC SEG32 output				
		UCK1: UART/SIO ch. 1 clock I/O				
P06/INT06/ AN06/SEG31	P06: General-purpose I/O	INT06: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN06: Analog input				
		SEG31: LCDC SEG31 output				
P07/INT07/ AN07/SEG30	P07: General-purpose I/O	INT07: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN07: Analog input				
		SEG30: LCDC SEG30 output				

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port 0

Figure 9.2-1 Block Diagram of P01 and P04

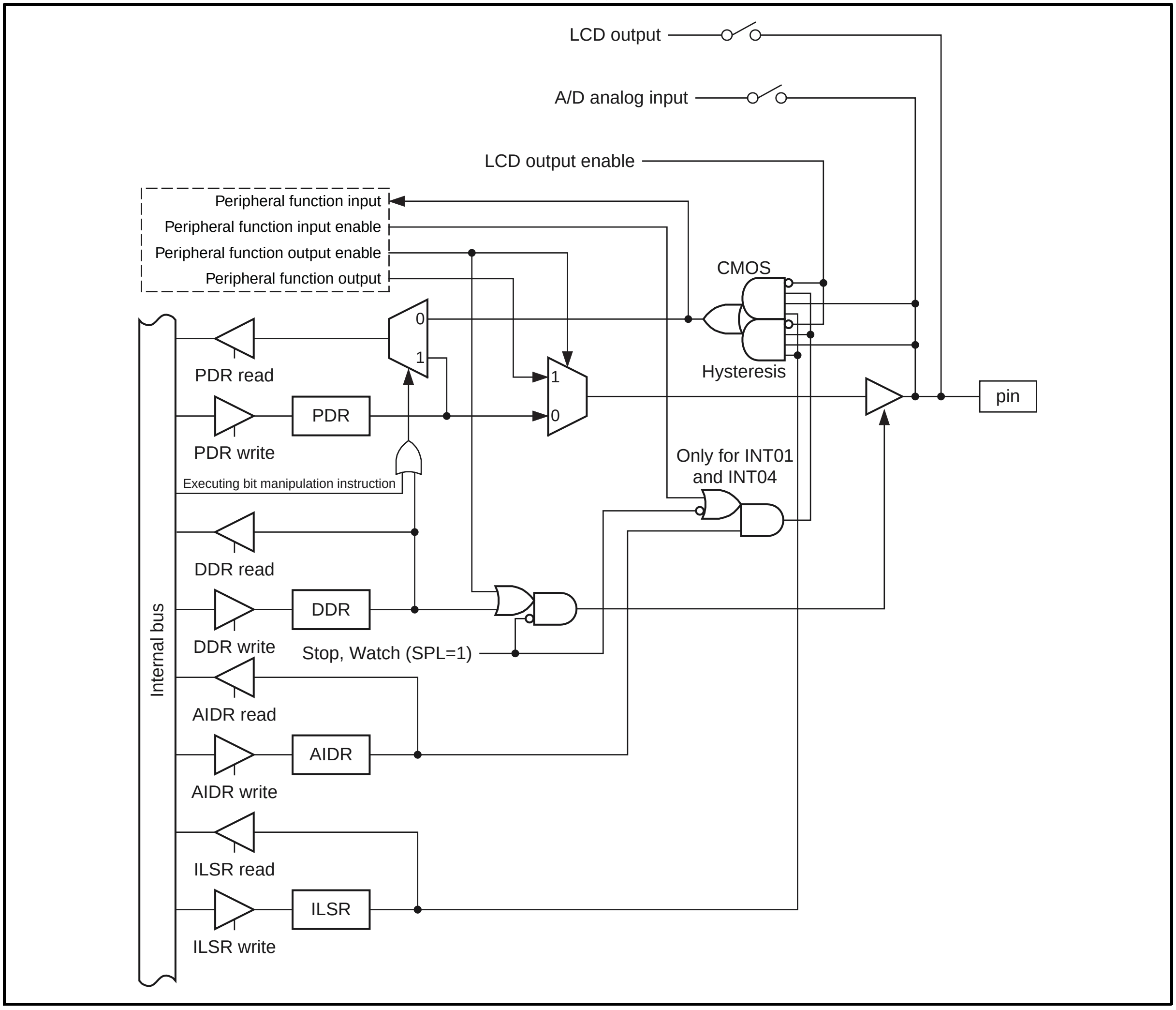
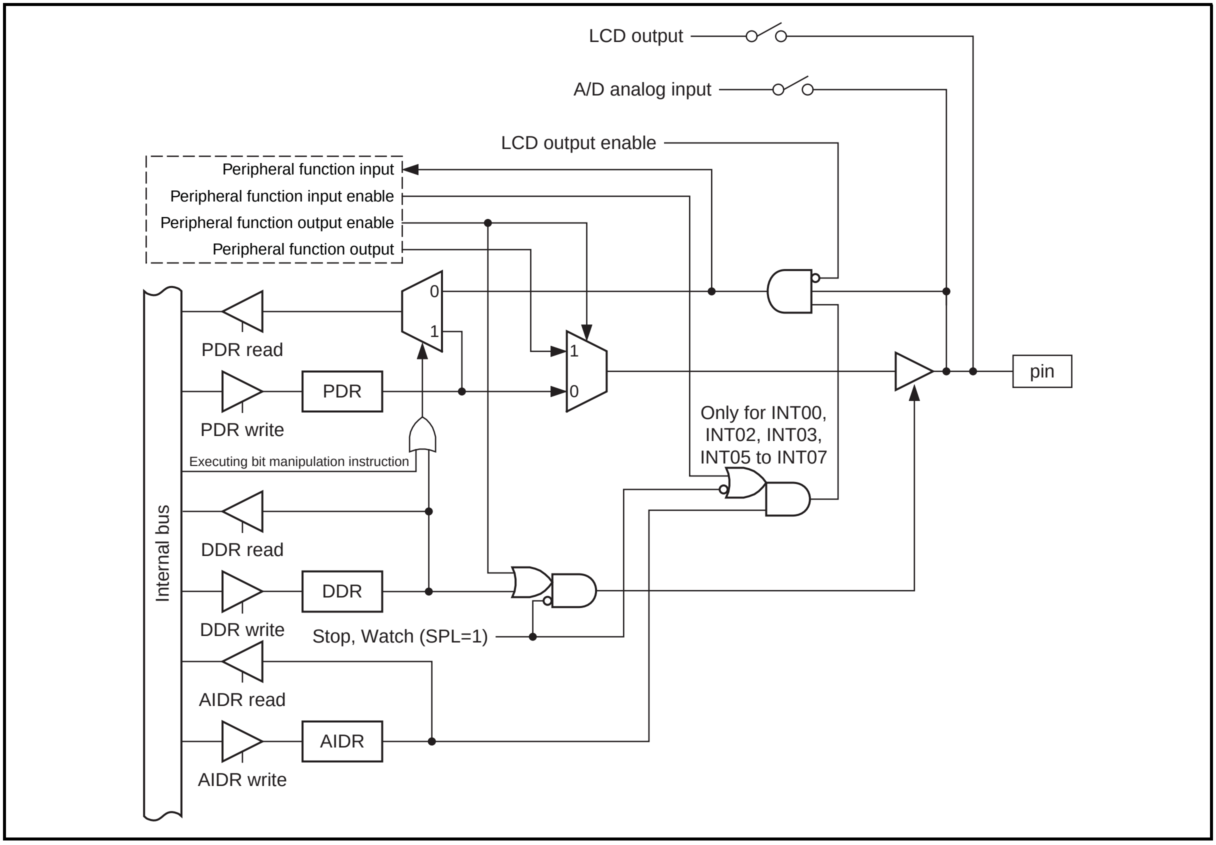


Figure 9.2-2 Block Diagram of P00, P02, P03, P05, P06 and P07



9.2.1 Port 0 Registers

This section describes the registers of port 0.

■ Port 0 Register Functions

Table 9.2-2 lists the functions of the port 0 register.

Table 9.2-2 Port 0 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR0	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR0	0	Port input enabled		
	1	Port output enabled		
AIDRL	0	Analog input enabled		
	1	Port input enabled		
ILSR	0	Hysteresis input level selected		
	1	CMOS input level selected		

Table 9.2-3 lists the correspondence between port 0 pins and each register bit.

Table 9.2-3 Correspondence between Registers and Pins for Port 0

	Correspondence between related register bits and pins							
Pin name	P07	P06	P05	P04	P03	P02	P01	P00
PDR0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDR0								
AIDRL								
ILSR	-	-	-	bit4	-	-	bit1	-

9.2.2 Operations of Port 0

This section describes the operations of port 0.

■ Operations of Port 0

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR register value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 6 (LCDCE6:SEG31, SEG30) or in the LCDC enable register 7 (LCDCE7:SEG36 to SEG32) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When using an analog input shared pin as an input port, set the corresponding bit in the A/D input disable register lower (AIDRL) to "1".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit in the LCDC enable register 6 (LCDCE6:SEG31, SEG30) or in the LCDC enable register 7 (LCDCE7:SEG36 to SEG32) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register bit corresponding to the input pin of a peripheral function to "0".
- When using the analog input shared pin as another peripheral function input pin, configure it as an input port, which is the same as the operation as an input port.
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit in the LCDC enable register 6 (LCDCE6:SEG31, SEG30) or in the LCDC enable register 7 (LCDCE7:SEG36 to SEG32) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled. As for a pin shared with analog input, its port input is disabled because the A/D input disable register lower (AIDRL) is initialized to "0".

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" and the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open. However, if the interrupt input is enabled for the external interrupt (INT07 to INT00), the input is enabled and not blocked.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation as an analog input pin

- Set the bit in the DDR register corresponding to the analog input pin to "0" and the bit corresponding to that pin in the AIDRL register to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.

● Operation as an external interrupt input pin

- Set the bit in the DDR register corresponding to the external interrupt input pin to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- The pin value is always input to the external interrupt circuit. When using a pin for a function other than the interrupt, disable the external interrupt function corresponding to that pin.

● Operation of the input level select register

- Setting bit1 and bit4 in ILSR to "1" changes P01 and P04 respectively from the hysteresis input level to the CMOS input level.
- For pins other than P01 and P04, the CMOS input level cannot be selected, but only the hysteresis input level can be selected.

- When changing the input level of P01 or of P04, ensure that all shared peripheral functions have been stopped.

Table 9.2-4 shows the pin states of port 0.

Table 9.2-4 Pin State of Port 0

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port/ peripheral function I/O	Hi-Z (the pull-up setting is enabled) Input cutoff (If the external interrupt function is enabled, the external interrupt can be input.)	Hi-Z Input disabled*

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input disabled" means the state that the operation of the input gate adjacent to the pin is disabled.

9.3 Port 1

Port 1 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port 1 Configuration

Port 1 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 1 data register (PDR1)
- Port 1 direction register (DDR1)
- Port 1 pull-up register (PUL1)
- Input level select register (ILSR)

■ Port 1 Pins

Port 1 has eight I/O pins.

Table 9.3-1 lists the port 1 pins.

Table 9.3-1 Port 1 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P10/UI0	P10: General-purpose I/O	UI0: UART/SIO ch. 0 data input	Hysteresis/ CMOS	CMOS	-	❍
P11/UO0	P11: General-purpose I/O	UO0: UART/SIO ch. 0 data output	Hysteresis	CMOS	-	❍
P12/DBG	P12: General-purpose I/O	DBG: On-chip debug communication pin	Hysteresis	CMOS	❍	-
P13/ADTG	P13: General-purpose I/O	ADTG: A/D trigger input	Hysteresis	CMOS	-	❍
P14/UCK0	P14: General-purpose I/O	UCK0: UART/SIO ch. 0 clock I/O	Hysteresis	CMOS	-	❍
P15/PPG11	P15: General-purpose I/O	PPG11: 8/16-bit PPG ch. 1 output	Hysteresis	CMOS	-	-
P16/PPG10	P16: General-purpose I/O	PPG10: 8/16-bit PPG ch. 1 output	Hysteresis	CMOS	-	-
P17/CMPO	P17: General-purpose I/O	CMPO: Voltage comparator output	Hysteresis	CMOS	-	❍

OD: N-ch open drain, PU: Pull-up

MB95410H/470H Series

■ Block Diagrams of Port 1

Figure 9.3-1 Block Diagram of P10

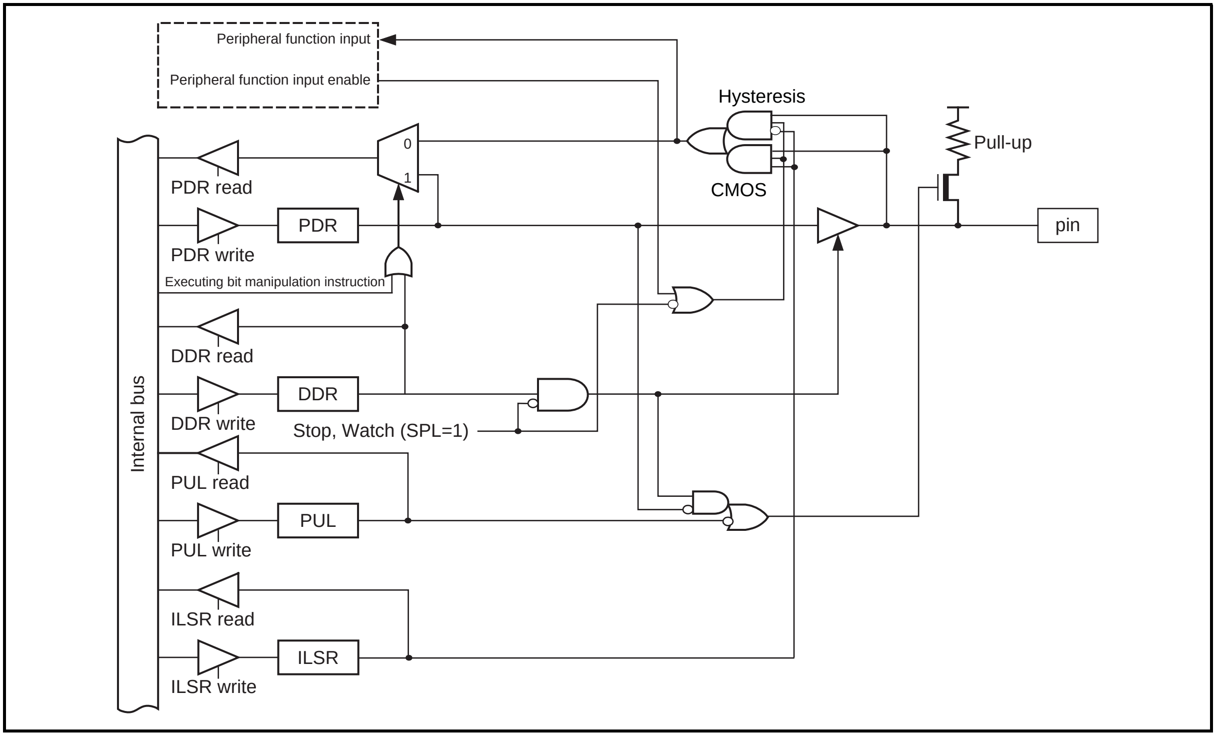


Figure 9.3-2 Block Diagram of P12

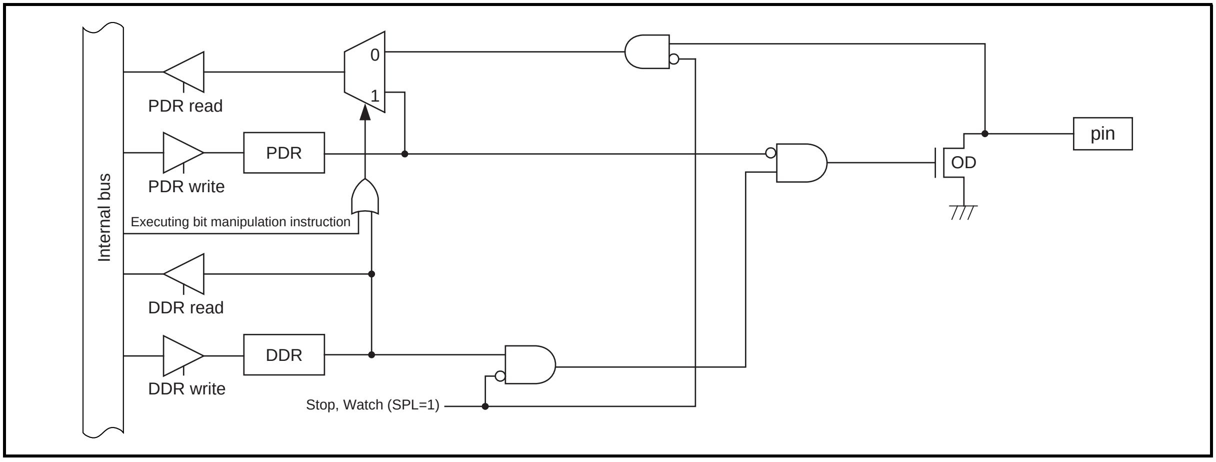


Figure 9.3-3 Block Diagram of P11, P13, P14 and P17

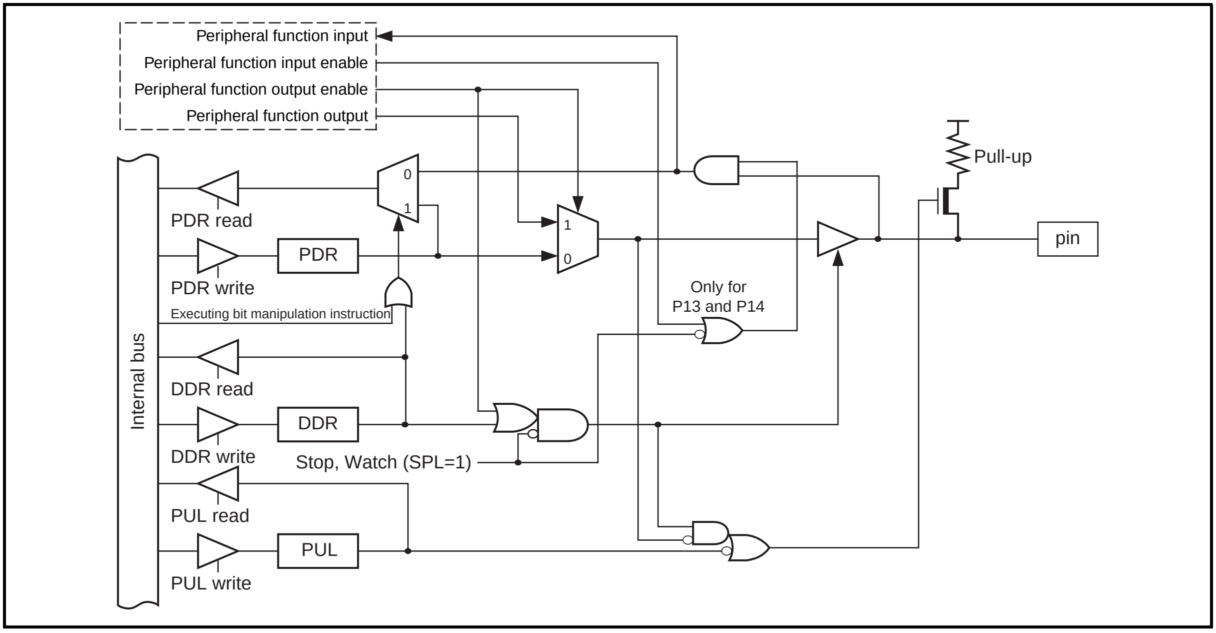
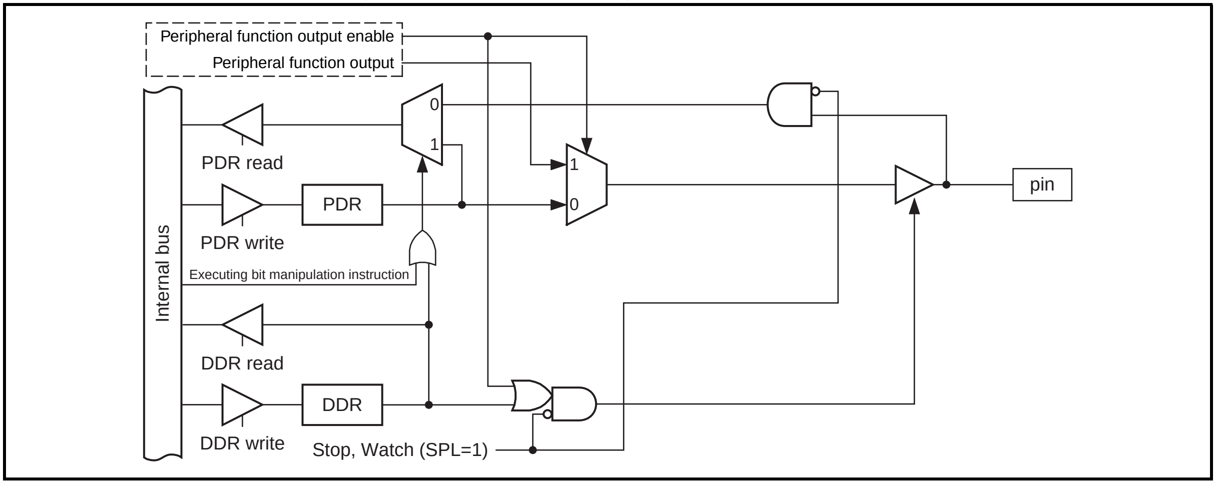

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
Only for
P13 and P14
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write

Figure 9.3-4 Block Diagram of P15 and P16


Peripheral function output enable
Peripheral function output
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)

9.3.1 Port 1 Registers

This section describes the registers of port 1.

■ Port 1 Register Functions

Table 9.3-2 lists the port 1 register functions.

Table 9.3-2 Port 1 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR1	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.*
DDR1	0	Port input enabled		
	1	Port output enabled		
PUL1	0	Pull-up disabled		
	1	Pull-up enabled		
ILSR	0	Hysteresis input level selected		
	1	CMOS input level selected		

*: For the N-ch open drain pin, this should be Hi-Z.

Table 9.3-3 lists the correspondence between port 1 pins and each register bit.

Table 9.3-3 Correspondence between Registers and Pins for Port 1

	Correspondence between related register bits and pins							
Pin name	P17	P16	P15	P14	P13	P12	P11	P10
PDR1 DDR1	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PUL1	bit7	-	-	bit4	bit3	-	bit1	bit0
ILSR	-	-	-	-	-	-	-	bit0

9.3.2 Operations of Port 1

This section describes the operations of port 1.

■ Operations of Port 1

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR register value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register corresponding to the input pin of a peripheral function to "0".
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" and the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation of the pull-up register

Setting the bit in the PUL register to "1" makes the pull-up resistor be internally connected to the pin. When the pin output is "L" level, the pull-up resistor is disconnected regardless of the value of the PUL register.

● Operation of the input level select register

- Setting bit0 in ILSR to "1" changes only P10 from the hysteresis input level to the CMOS input level. When the same bit is set to "0", the input level of P10 should become the hysteresis input level.
- For pins other than P10, the CMOS input level cannot be selected, but only the hysteresis input level can be selected.
- When changing the input level of P10, ensure that the peripheral function (UART/SIO ch. 0 output) has been stopped.

Table 9.3-4 shows the pin states of port 1.

Table 9.3-4 Pin State of Port 1

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port/ peripheral function I/O	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.4 Port 2

**Port 2 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 2 Configuration

Port 2 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 2 data register (PDR2)
- Port 2 direction register (DDR2)
- Port 2 pull-up register (PUL2)
- Input level select register (ILSR)

■ Port 2 Pins

Port 2 has four I/O pins.

Table 9.4-1 lists the port 2 pins.

Table 9.4-1 Port 2 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P20/PPG00/CMPN	P20: General-purpose I/O	PPG00: 8/16-bit PPG ch. 0 output	Hysteresis/analog	CMOS	-	❍
		CMPN: Voltage comparator N ch input				
P21/PPG01/CMPP	P21: General-purpose I/O	PPG01: 8/16-bit PPG ch. 0 output	Hysteresis/analog	CMOS	-	❍
		CMPP: Voltage comparator P ch input				
P22/SCL	P22: General-purpose I/O	SCL: I^{2}C clock I/O	Hysteresis/CMOS	CMOS	❍	-
P23/SDA	P23: General-purpose I/O	SDA: I^{2}C data I/O	Hysteresis/CMOS	CMOS	❍	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port 2

Figure 9.4-1 Block Diagram of P20 and P21

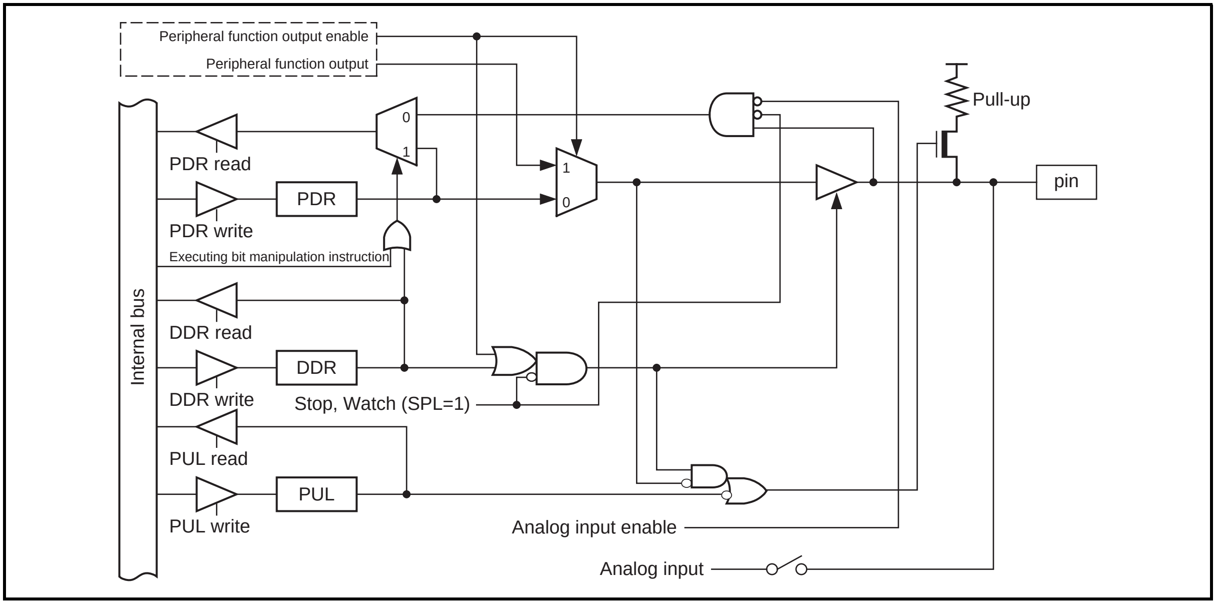
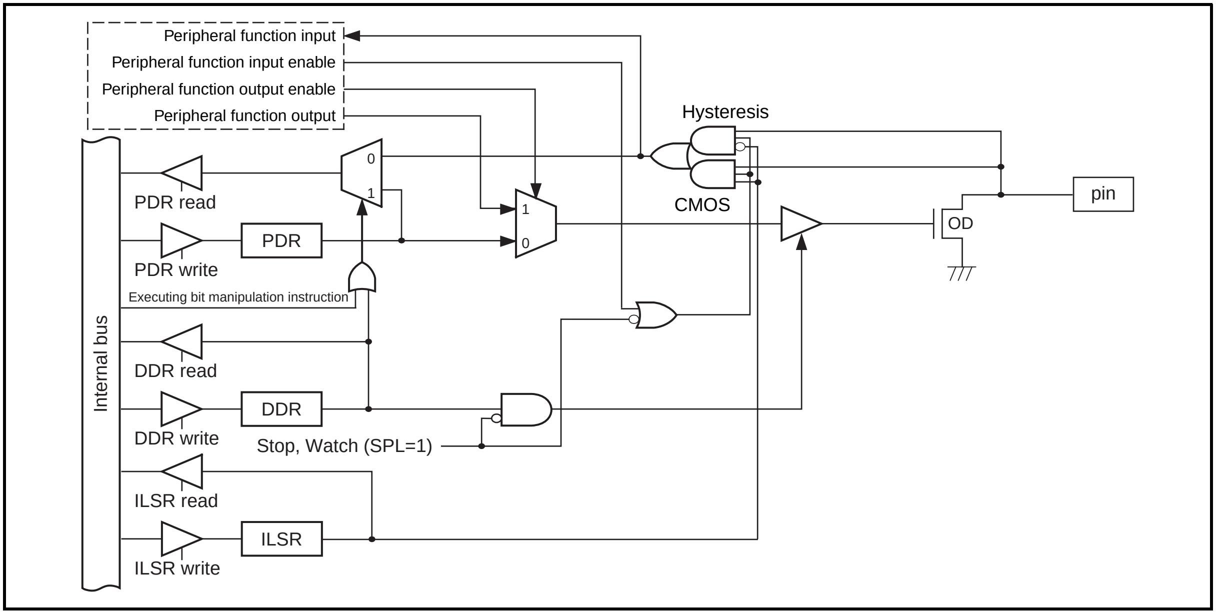


Figure 9.4-2 Block Diagram of P22 and P23


Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Hysteresis
0
1
PDR read
CMOS
pin
1
0
PDR
OD
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
ILSR read
ILSR
ILSR write

9.4.1 Port 2 Registers

This section describes the registers of port 2.

■ Port 2 Register Functions

Table 9.4-2 lists the port 2 register functions.

Table 9.4-2 Port 2 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR2	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.*
DDR2	0	Port input enabled		
	1	Port output enabled		
PUL2	0	Pull-up disabled		
	1	Pull-up enabled		
ILSR	0	Hysteresis input level selected		
	1	CMOS input level selected		

*: For the N-ch open drain pin, this should be Hi-Z.

Table 9.4-3 lists the correspondence between port 2 pins and each register bit.

Table 9.4-3 Correspondence Between Registers and Pins for Port 2

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	P23	P22	P21	P20
PDR2	-	-	-	-	bit3	bit2	bit1	bit0
DDR2								
PUL2	-	-	-	-	-	-	bit1	bit0
ILSR	-	-	-	-	bit3	bit2	-	-

9.4.2 Operations of Port 2

This section describes the operations of port 2.

■ Operations of Port 2

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register corresponding to the input pin of a peripheral function to "0".
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" and the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation as an analog input pin

- Setting the voltage comparator analog input disable bit in the voltage comparator control register (CMR0:VCID) to "0" enables the analog input function of an analog input pin regardless of the settings of the PDR register.
- To disable the analog input function of an analog input pin, set the VCID bit in the CMR0 register to "1".

● Operation of the pull-up register

Setting the bit in the PUL register to "1" makes the pull-up resistor be internally connected to the pin. When the pin output is "L" level, the pull-up resistor is disconnected regardless of the value of the PUL register.

● Operation of the input level select register

- Setting bit2 and bit3 in ILSR to "1" changes P22 and P23 respectively from the hysteresis input level to the CMOS input level. When the same bit is set to "0", the input levels of P22 and P23 become the hysteresis input level.
- For pins other than P22 and P23, the CMOS input level cannot be selected, but only the hysteresis input level can be selected.
- When changing the input levels of P22 and P23, ensure that all shared peripheral functions have been stopped.

Table 9.4-4 shows the pin states of port 2.

Table 9.4-4 Pin State of Port 2

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port/peripheral function I/O	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.5 Port 4

Port 4 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port 4 Configuration

Port 4 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 4 data register (PDR4)
- Port 4 direction register (DDR4)

■ Port 4 Pins

Port 4 has four I/O pins.

Table 9.5-1 list the port 4 pins.

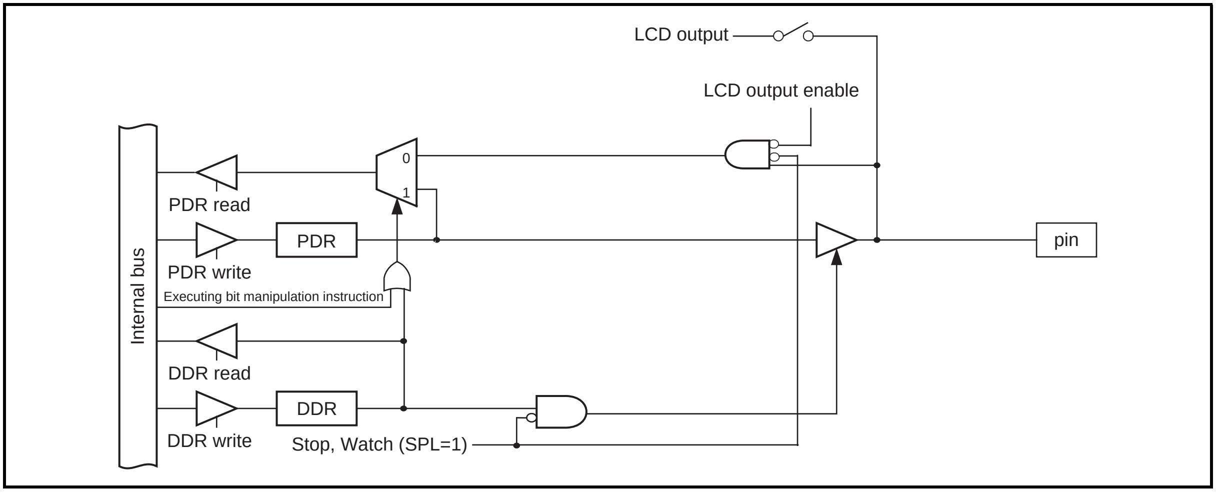
Table 9.5-1 Port 4 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P40/SEG21	P40: General-purpose I/O	SEG21: LCDC SEG21 output	Hysteresis	CMOS/ LCD	-	-
P41/SEG20	P41: General-purpose I/O	SEG20: LCDC SEG20 output	Hysteresis	CMOS/ LCD	-	-
P42/SEG19	P42: General-purpose I/O	SEG19: LCDC SEG19 output	Hysteresis	CMOS/ LCD	-	-
P43/SEG18	P43: General-purpose I/O	SEG18: LCDC SEG18 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port 4

Figure 9.5-1 Block Diagram of P40, P41, P42 and P43



9.5.1 Port 4 Registers

This section describes the registers of port 4.

■ Port 4 Register Functions

Table 9.5-2 lists the port 4 register functions.

Table 9.5-2 Port 4 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR4	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR4	0	Port input enabled		
	1	Port output enabled		

Table 9.5-3 lists the correspondence between port 4 pins and each register bit.

Table 9.5-3 Correspondence between Registers and Pins for Port 4

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	P43	P42	P41	P40
PDR4	-	-	-	-	bit3	bit2	bit1	bit0
DDR4								

9.5.2 Operations of Port 4

This section describes the operations of port 4.

■ Operations of Port 4

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit (SEG21 to SEG18) in the LCDC enable register 5 (LCDCE5) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit (SEG21 to SEG18) in the LCDC enable register 5 (LCDCE5) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit (SEG21 to SEG18) in the LCDC enable register 5 (LCDCE5) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 9.5-4 shows the pin states of port 4.

Table 9.5-4 Pin State of Port 4

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.6 Port 5

**Port 5 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 5 Configuration

Port 5 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 5 data register (PDR5)
- Port 5 direction register (DDR5)
- Port 5 pull-up register (PUL5)

■ Port 5 Pins

Port 5 has four I/O pins.

Table 9.6-1 lists the port 5 pins.

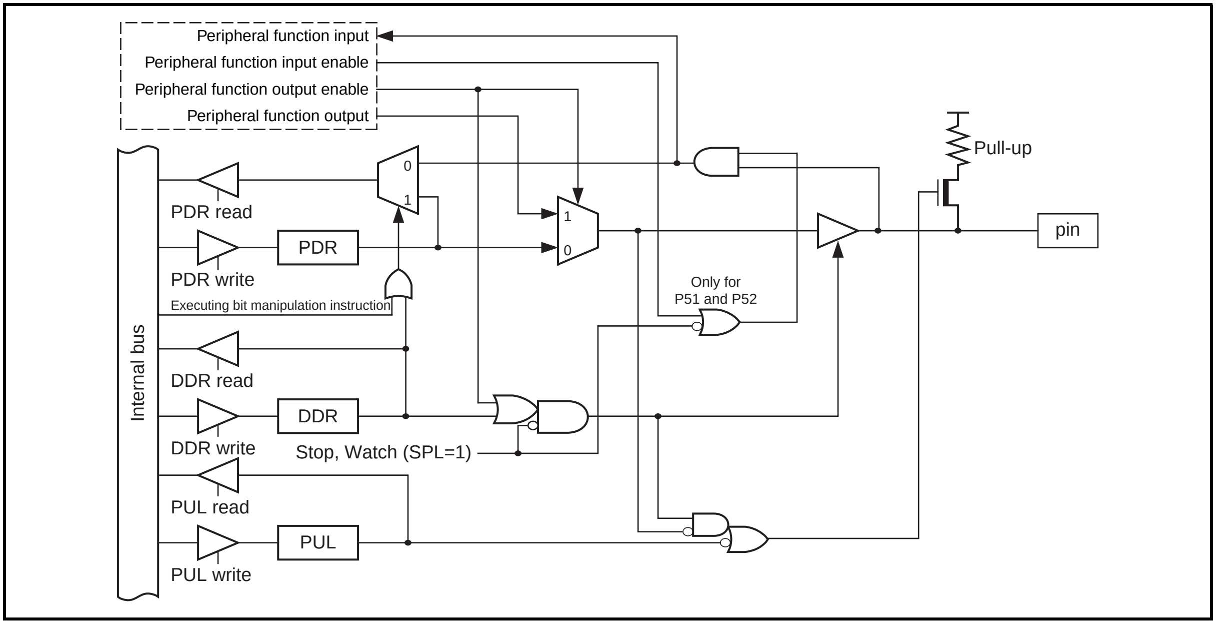
Table 9.6-1 Port 5 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P50/TO01	P50: General-purpose I/O	TO01: 8/16-bit composite timer ch. 0 output	Hysteresis	CMOS	-	○
P51/EC0	P51: General-purpose I/O	EC0: 8/16-bit composite timer ch. 0 clock input	Hysteresis	CMOS	-	○
P52/TI0/TO00	P52: General-purpose I/O	TI0: 16-bit reload timer input TO00: 8/16-bit composite timer ch. 0 output	Hysteresis	CMOS	-	○
P53/TO0	P53: General-purpose I/O	TO0: 16-bit reload timer output	Hysteresis	CMOS	-	○

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port 5

Figure 9.6-1 Block Diagram of P50, P51, P52 and P53


Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
0
1
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write
1
0
Only for
P51 and P52
Pull-up
pin

9.6.1 Port 5 Registers

This section describes the registers of port 5.

■ Port 5 Register Functions

Table 9.6-2 lists the port 5 register functions.

Table 9.6-2 Port 5 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR5	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR5	0	Port input enabled		
	1	Port output enabled		
PUL5	0	Pull-up disabled		
	1	Pull-up enabled		

Table 9.6-3 lists the correspondence between port 5 pins and each register bit.

Table 9.6-3 Correspondence between Registers and Pins for Port 5

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	P53	P52	P51	P50
PDR5	-	-	-	-	bit3	bit2	bit1	bit0
DDR5								
PUL5								

MB95410H/470H Series

9.6.2 Operations of Port 5

This section describes the operations of port 5.

■ Operations of Port 5

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register corresponding to the input pin of a peripheral function to "0".
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation of the pull-up register

Setting the bit in the PUL register to "1" makes the pull-up resistor be internally connected to the pin. When the pin output is "L" level, the pull-up resistor is disconnected regardless of the value of the PUL register.

Table 9.6-4 shows the pin states of port 5.

Table 9.6-4 Pin State of Port 5

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.7 Port 6

**Port 6 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 6 Configuration

Port 6 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 6 data register (PDR6)
- Port 6 direction register (DDR6)

■ Port 6 Pins

Port 6 has eight I/O pins.

Table 9.7-1 lists the port 6 pins.

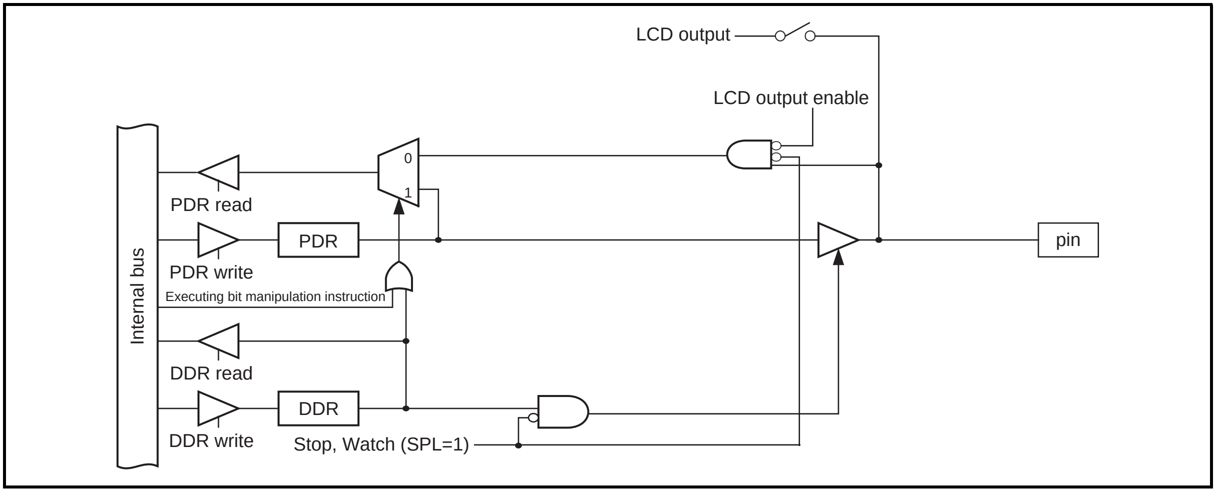
Table 9.7-1 Port 6 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P60/SEG10	P60: General-purpose I/O	SEG10: LCDC SEG10 output	Hysteresis	CMOS/ LCD	-	-
P61/SEG11	P61: General-purpose I/O	SEG11: LCDC SEG11 output	Hysteresis	CMOS/ LCD	-	-
P62/SEG12	P62: General-purpose I/O	SEG12: LCDC SEG12 output	Hysteresis	CMOS/ LCD	-	-
P63/SEG13	P63: General-purpose I/O	SEG13: LCDC SEG13 output	Hysteresis	CMOS/ LCD	-	-
P64/SEG14	P64: General-purpose I/O	SEG14: LCDC SEG14 output	Hysteresis	CMOS/ LCD	-	-
P65/SEG15	P65: General-purpose I/O	SEG15: LCDC SEG15 output	Hysteresis	CMOS/ LCD	-	-
P66/SEG16	P66: General-purpose I/O	SEG16: LCDC SEG16 output	Hysteresis	CMOS/ LCD	-	-
P67/SEG17	P67: General-purpose I/O	SEG17: LCDC SEG17 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port 6

Figure 9.7-1 Block Diagram of P60, P61, P62, P63, P64, P65, P66 and P67



9.7.1 Port 6 Registers

This section describes the registers of port 6.

■ Port 6 Register Functions

Table 9.7-2 lists the port 6 register functions.

Table 9.7-2 Port 6 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR6	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR6	0	Port input enabled		
	1	Port output enabled		

Table 9.7-3 lists the correspondence between port 6 pins and each register bit.

Table 9.7-3 Correspondence between Registers and Pins for Port 6

	Correspondence between related register bits and pins							
Pin name	P67	P66	P65	P64	P63	P62	P61	P60
PDR6	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDR6								

9.7.2 Operations of Port 6

This section describes the operations of port 6.

■ Operations of Port 6

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 4 (LCDCE4:SEG15 to SEG10) or in the LCDC enable register 5 (LCDCE5:SEG17, SEG16) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit in the LCDC enable register 4 (LCDCE4:SEG15 to SEG10) or in the LCDC enable register 5 (LCDCE5:SEG17, SEG16) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit in the LCDC enable register 4 (LCDCE4:SEG15 to SEG10) or in the LCDC enable register 5 (LCDCE5:SEG17, SEG16) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 9.7-4 shows the pin states of port 6.

Table 9.7-4 Pin State of Port 6

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.8 Port 9

**Port 9 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 9 Configuration

Port 9 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 9 data register (PDR9)
- Port 9 direction register (DDR9)

■ Port 9 Pins

Port 9 has five I/O pins.

Table 9.8-1 lists the port 9 pins.

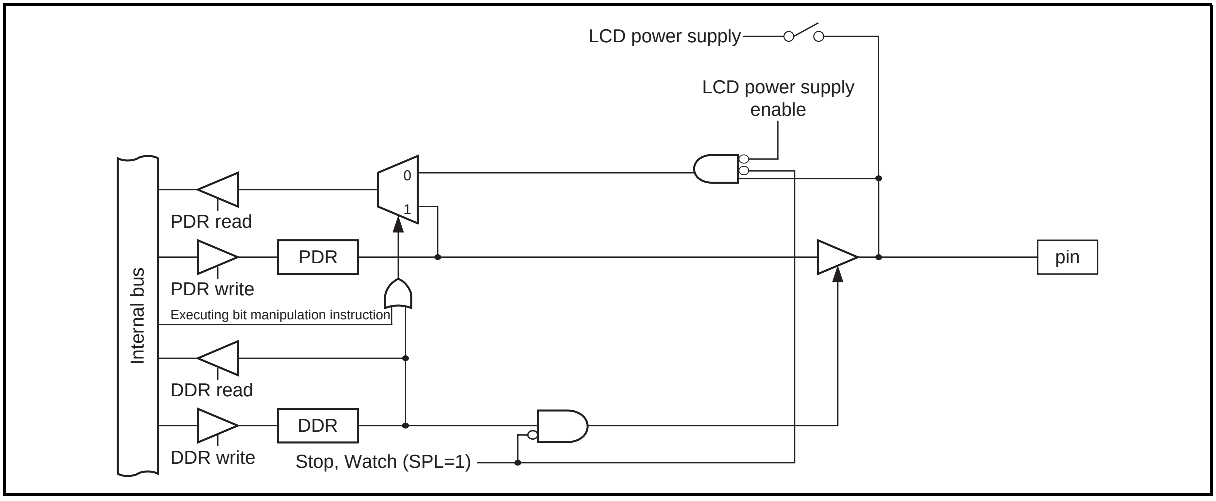
Table 9.8-1 Port 9 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P90/V4	P90: General-purpose I/O	V4: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-
P91/V3	P91: General-purpose I/O	V3: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-
P92/V2	P92: General-purpose I/O	V2: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-
P93/V1	P93: General-purpose I/O	V1: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-
P94/V0	P94: General-purpose I/O	V0: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port 9

Figure 9.8-1 Block Diagram of P90, P91, P92, P93 and P94



9.8.1 Port 9 Registers

This section describes the registers of port 9.

■ Port 9 Register Functions

Table 9.8-2 lists the port 9 register functions.

Table 9.8-2 Port 9 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR9	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR9	0	Port input enabled		
	1	Port output enabled		

Table 9.8-3 lists the correspondence between port 9 pins and each register bit.

Table 9.8-3 Correspondence between Registers and Pins for Port 9

	Correspondence between related register bits and pins							
Pin name	-	-	-	P94	P93	P92	P91	P90
PDR9	-	-	-	bit4	bit3	bit2	bit1	bit0
DDR9								

9.8.2 Operations of Port 9

This section describes the operations of port 9.

■ Operations of Port 9

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set the bit (VE4 to VE0) corresponding to that pin in the LCDC enable register 1 (LCDCE1) to "0".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set the bit (VE4 to VE0) corresponding to that pin in the LCDC enable register 1 (LCDCE1) to "0".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operations as LCDC pins

- Set the DDR register bit corresponding to a desired LCDC pin to "0".
- Set the V0 select bit (VE0), the V1 select bit (VE1), the V2 select bit (VE2), the V3 select bit (VE3) and the V4 select bit (VE4) in the LCDC enable register 1 (LCDCE1) to "1".

Table 9.8-4 shows the pin states of port 9.

Table 9.8-4 Pin State of Port 9

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.9 Port A

**Port A is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port A Configuration

Port A is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port A data register (PDRA)
- Port A direction register (DDRA)

■ Port A Pins

Port A has eight I/O pins.

Table 9.9-1 lists the port A pins.

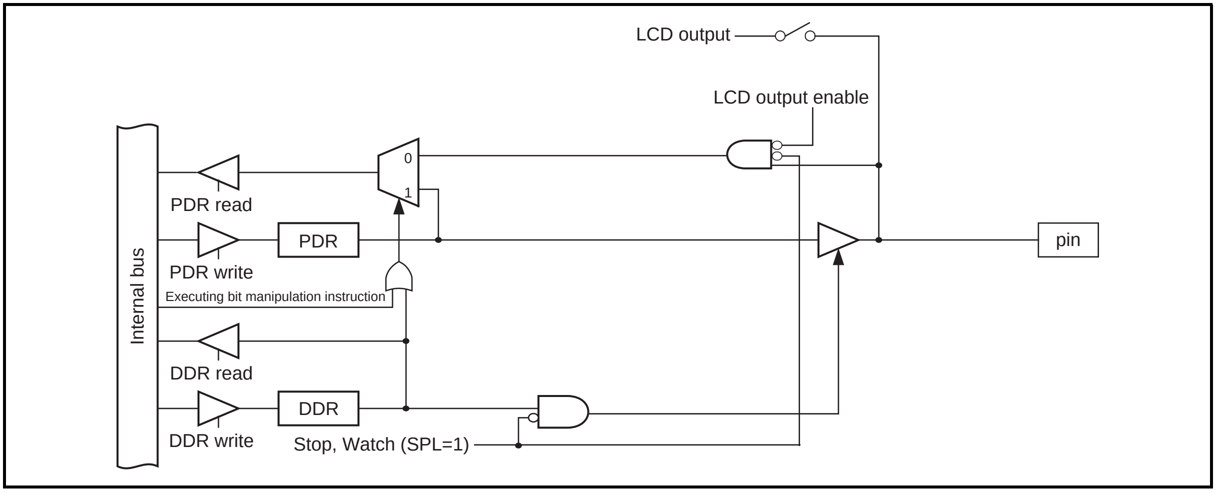
Table 9.9-1 Port A Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PA0/COM0	PA0: General-purpose I/O	COM0: LCDC COM0 output	Hysteresis	CMOS/ LCD	-	-
PA1/COM1	PA1: General-purpose I/O	COM1: LCDC COM1 output	Hysteresis	CMOS/ LCD	-	-
PA2/COM2	PA2: General-purpose I/O	COM2: LCDC COM2 output	Hysteresis	CMOS/ LCD	-	-
PA3/COM3	PA3: General-purpose I/O	COM3: LCDC COM3 output	Hysteresis	CMOS/ LCD	-	-
PA4/COM4	PA4: General-purpose I/O	COM4: LCDC COM4 output	Hysteresis	CMOS/ LCD	-	-
PA5/COM5	PA5: General-purpose I/O	COM5: LCDC COM5 output	Hysteresis	CMOS/ LCD	-	-
PA6/COM6	PA6: General-purpose I/O	COM6: LCDC COM6 output	Hysteresis	CMOS/ LCD	-	-
PA7/COM7	PA7: General-purpose I/O	COM7: LCDC COM7 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port A

Figure 9.9-1 Block Diagram of PA0, PA1, PA2, PA3, PA4, PA5, PA6 and PA7



9.9.1 Port A Registers

This section describes the registers of port A.

■ Port A Register Functions

Table 9.9-2 lists the port A register functions.

Table 9.9-2 Port A Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRA	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRA	0	Port input enabled		
	1	Port output enabled		

Table 9.9-3 lists the correspondence between port A pins and each register bit.

Table 9.9-3 Correspondence between Registers and Pins for Port A

	Correspondence between related register bits and pins							
Pin name	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
PDRA	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDRA								

9.9.2 Operations of Port A

This section describes the operations of port A.

■ Operations of Port A

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding common select bit (COM7 to COM0) in the LCDC enable register 2 (LCDCE2) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding common select bit (COM7 to COM0) in the LCDC enable register 2 (LCDCE2) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation as an LCDC common output

- Set the DDR register bit corresponding to a desired LCDC common output pin to "0".
- Select the common output by setting a corresponding common select bit (COM7 to COM0) in the LCDC enable register 2 (LCDCE2) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

Table 9.9-4 shows the pin states of port A.

Table 9.9-4 Pin State of Port A

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.10 Port B

Port B is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port B Configuration

Port B is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port B data register (PDRB)
- Port B direction register (DDRB)

■ Port B Pins

Port B has five I/O pins.

Table 9.10-1 lists the port B pins.

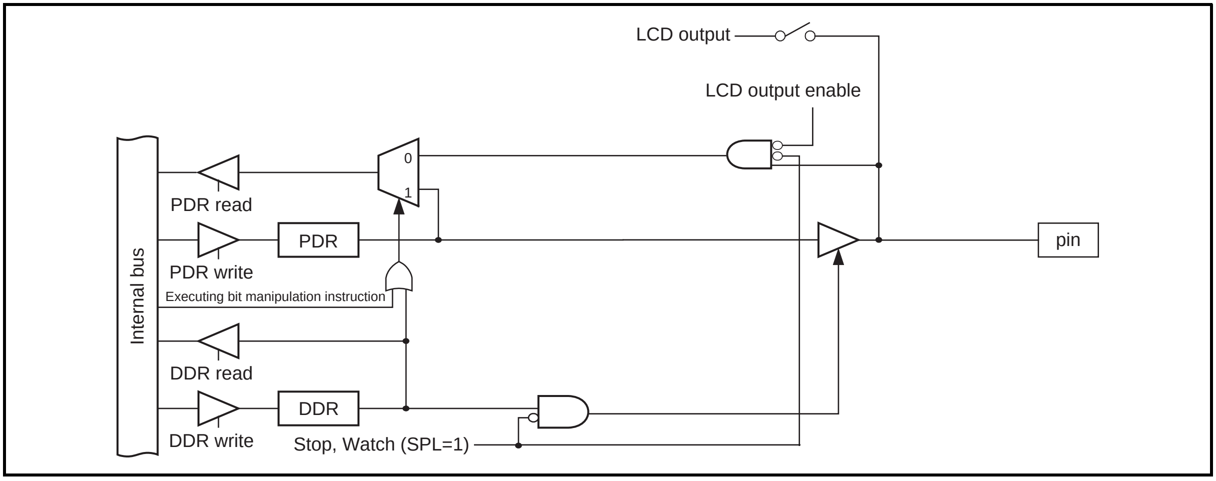
Table 9.10-1 Port B Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PB0/SEG00	PB0: General-purpose I/O	SEG00: LCDC SEG00 output	Hysteresis	CMOS/ LCD	-	-
PB1/SEG01	PB1: General-purpose I/O	SEG01: LCDC SEG01 output	Hysteresis	CMOS/ LCD	-	-
PB2/SEG37	PB2: General-purpose I/O	SEG37: LCDC SEG37 output	Hysteresis	CMOS/ LCD	-	-
PB3/SEG38	PB3: General-purpose I/O	SEG38: LCDC SEG38 output	Hysteresis	CMOS/ LCD	-	-
PB4/SEG39	PB4: General-purpose I/O	SEG39: LCDC SEG39 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port B

Figure 9.10-1 Block Diagram of PB0, PB1, PB2, PB3 and PB4



9.10.1 Port B Registers

This section describes the registers of port B.

■ Port B Register Functions

Table 9.10-2 lists the port B register functions.

Table 9.10-2 Port B Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRB	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRB	0	Port input enabled		
	1	Port output enabled		

Table 9.10-3 lists the correspondence between port B pins and each register bit.

Table 9.10-3 Correspondence between Registers and Pins for Port B

	Correspondence between related register bits and pins							
Pin name	-	-	-	PB4	PB3	PB2	PB1	PB0
PDRB	-	-	-	bit4	bit3	bit2	bit1	bit0
DDRB								

9.10.2 Operations of Port B

This section describes the operations of port B.

■ Operations of Port B

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 3 (LCDCE3:SEG01, SEG00) or in the LCDC enable register 7 (LCDCE7:SEG39 to SEG37) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit the LCDC enable register 3 (LCDCE3:SEG01, SEG00) or in the LCDC enable register 7 (LCDCE7:SEG39 to SEG37) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit the LCDC enable register 3 (LCDCE3:SEG01, SEG00) or in the LCDC enable register 7 (LCDCE7:SEG39 to SEG37) to "1", and then set the port input control bit (PICTL) in LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.

- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 9.10-4 shows the pin states of port B.

Table 9.10-4 Pin State of Port B

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.11 Port C

Port C is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port C Configuration

Port C is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port C data register (PDRC)
- Port C direction register (DDRC)

■ Port C Pins

Port C has eight I/O pins.

Table 9.11-1 lists the port C pins.

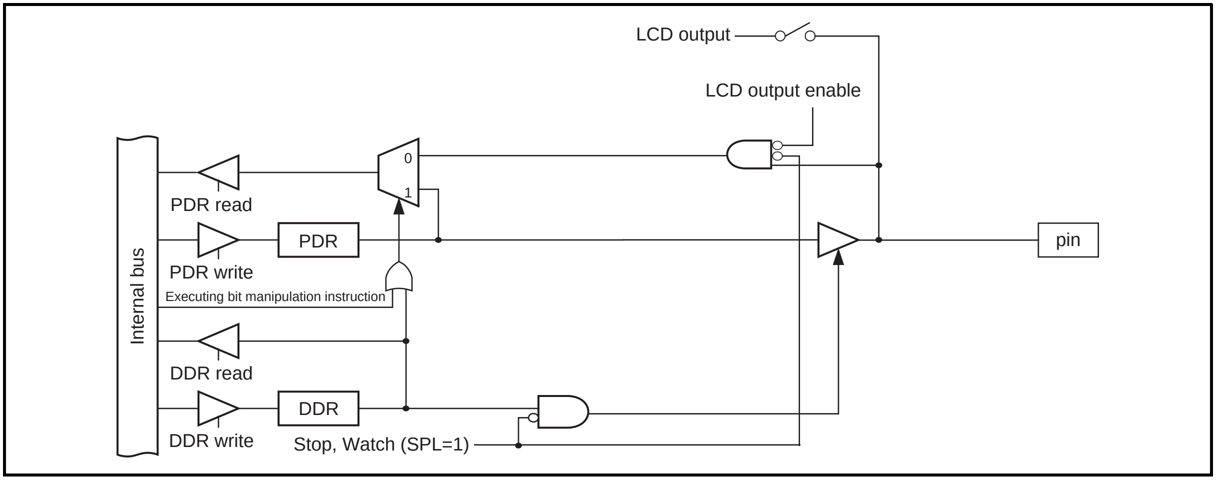
Table 9.11-1 Port C Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PC0/SEG02	PC0: General-purpose I/O	SEG02: LCDC SEG02 output	Hysteresis	CMOS/ LCD	-	-
PC1/SEG03	PC1: General-purpose I/O	SEG03: LCDC SEG03 output	Hysteresis	CMOS/ LCD	-	-
PC2/SEG04	PC2: General-purpose I/O	SEG04: LCDC SEG04 output	Hysteresis	CMOS/ LCD	-	-
PC3/SEG05	PC3: General-purpose I/O	SEG05: LCDC SEG05 output	Hysteresis	CMOS/ LCD	-	-
PC4/SEG06	PC4: General-purpose I/O	SEG06: LCDC SEG06 output	Hysteresis	CMOS/ LCD	-	-
PC5/SEG07	PC5: General-purpose I/O	SEG07: LCDC SEG07 output	Hysteresis	CMOS/ LCD	-	-
PC6/SEG08	PC6: General-purpose I/O	SEG08: LCDC SEG08 output	Hysteresis	CMOS/ LCD	-	-
PC7/SEG09	PC7: General-purpose I/O	SEG09: LCDC SEG09 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port C

Figure 9.11-1 Block Diagram of PC0, PC1, PC2, PC3, PC4, PC5, PC6 and PC7



9.11.1 Port C Registers

This section describes the registers of port C.

■ Port C Register Functions

Table 9.11-2 lists the port C register functions.

Table 9.11-2 Port C Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRC	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRC	0	Port input enabled		
	1	Port output enabled		

Table 9.11-3 lists the correspondence between port C pins and each register bit.

Table 9.11-3 Correspondence between Registers and Pins for Port C

	Correspondence between related register bits and pins							
Pin name	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
PDRC	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDRC								

9.11.2 Operations of Port C

This section describes the operations of port C.

■ Operations of Port C

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 3 (LCDCE3:SEG07 to SEG02) or in the LCDC enable register 4 (LCDCE4:SEG09, SEG08) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit in the LCDC enable register 3 (LCDCE3:SEG07 to SEG02) or in the LCDC enable register 4 (LCDCE4:SEG09, SEG08) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit in the LCDC enable register 3 (LCDCE3:SEG07 to SEG02) or in the LCDC enable register 4 (LCDCE4:SEG09, SEG08) to "1", and then set the port input control bit (PICTL) in LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 9.11-4 shows the pin states of port C.

Table 9.11-4 Pin State of Port C

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

9.12 Port E

Port E is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port E Configuration

Port E is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port E data register (PDRE)
- Port E direction register (DDRE)

■ Port E Pins

Port E has eight I/O pins.

Table 9.12-1 lists the port E pins.

Table 9.12-1 Port E Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PE0/SEG22	PE0: General-purpose I/O	SEG22: LCDC SEG22 output	Hysteresis	CMOS/ LCD	-	-
PE1/SEG23	PE1: General-purpose I/O	SEG23: LCDC SEG23 output	Hysteresis	CMOS/ LCD	-	-
PE2/SEG24	PE2: General-purpose I/O	SEG24: LCDC SEG24 output	Hysteresis	CMOS/ LCD	-	-
PE3/SEG25	PE3: General-purpose I/O	SEG25: LCDC SEG25 output	Hysteresis	CMOS/ LCD	-	-
PE4/SEG26	PE4: General-purpose I/O	SEG26: LCDC SEG26 output	Hysteresis	CMOS/ LCD	-	-
PE5/SEG27/ TO11	PE5: General-purpose I/O	SEG27: LCDC SEG27 output	Hysteresis	CMOS/ LCD	-	-
		TO11: 8/16-bit composite timer ch. 1 output				
PE6/SEG28/ TO10	PE6: General-purpose I/O	SEG28: LCDC SEG28 output	Hysteresis	CMOS/ LCD	-	-
		TO10: 8/16-bit composite timer ch. 1 output				
PE7/SEG29/ EC1	PE7: General-purpose I/O	SEG29: LCDC SEG29 output	Hysteresis	CMOS/ LCD	-	-
		EC1: 8/16-bit composite timer ch. 1 clock input				

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port E

Figure 9.12-1 Block Diagram of PE0, PE1, PE2, PE3 and PE4

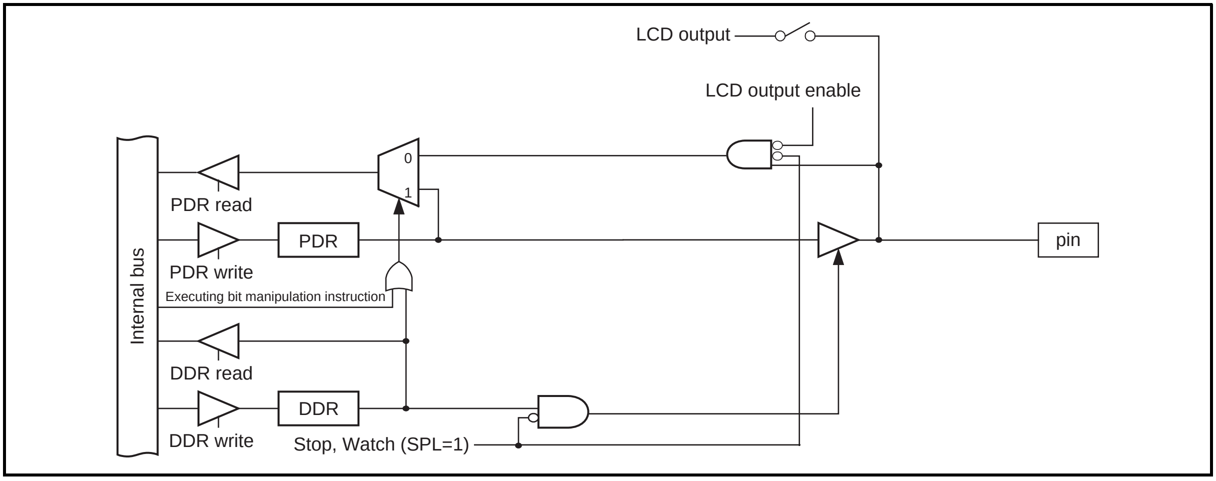
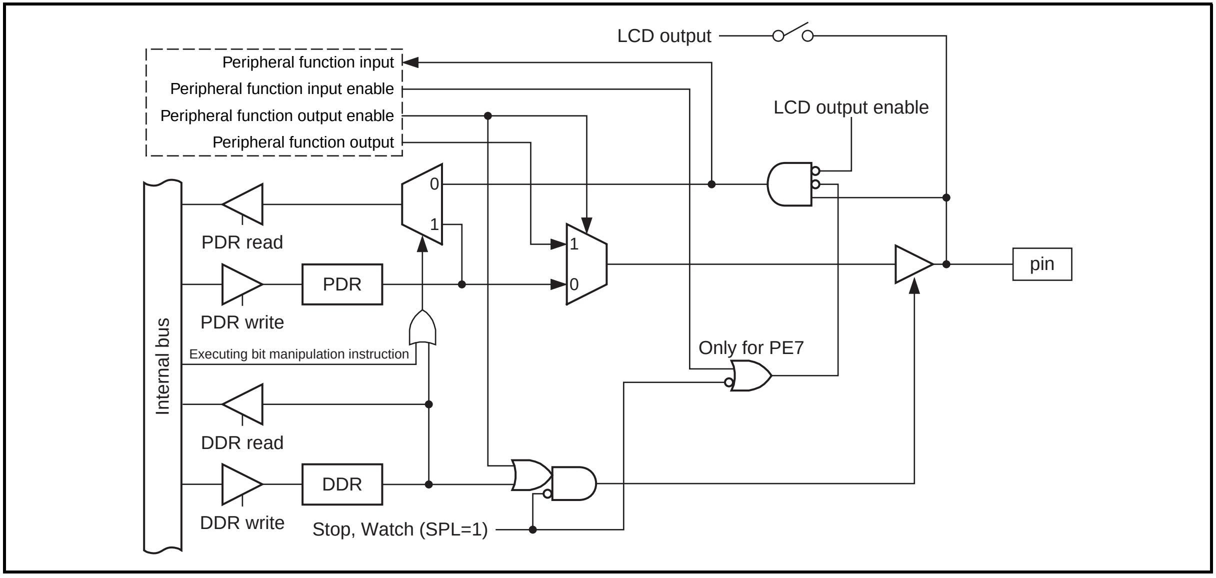


Figure 9.12-2 Block Diagram of PE5, PE6 and PE7


LCD output
Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
LCD output enable
0
1
PDR read
PDR
1
0
pin
PDR write
Executing bit manipulation instruction
Only for PE7
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)

9.12.1 Port E Registers

This section describes the registers of port E.

■ Port E Register Functions

Table 9.12-2 lists the port E register functions.

Table 9.12-2 Port E Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRE	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRE	0	Port input enabled		
	1	Port output enabled		

Table 9.12-3 lists the correspondence between port E pins and each register bit.

Table 9.12-3 Correspondence between Registers and Pins for Port E

	Correspondence between related register bits and pins							
Pin name	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
PDRE	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDRE								

9.12.2 Operations of Port E

This section describes the operations of port E.

■ Operations of Port E

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 5 (LCDCE5:SEG23, SEG22) or in the LCDC enable register 6 (SEG29 to SEG24) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit in the LCDC enable register 5 (LCDCE5:SEG23, SEG22) or in the LCDC enable register 6 (SEG29 to SEG24) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register bit corresponding to the input pin of a peripheral function to "0".
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit in the LCDC enable register 5 (LCDCE5:SEG23, SEG22) or in the LCDC enable register 6 (SEG29 to SEG24) to "1", and then set the port input control bit (PICTL) in LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 9.12-4 shows the pin states of port E.

Table 9.12-4 Pin State of Port E

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

MB95410H/470H Series

9.13 Port F

**Port F is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port F Configuration

Port F is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port F data register (PDRF)
- Port F direction register (DDRF)

■ Port F Pins

Port F has three I/O pins.

Table 9.13-1 lists the port F pins.

Table 9.13-1 Port F Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PF0/X0*1	PF0: General-purpose I/O	X0: Main clock oscillation pin	Hysteresis	CMOS	-	-
PF1/X1*1	PF1: General-purpose I/O	X1: Main clock oscillation pin	Hysteresis	CMOS	-	-
PF2/$\overline{\text{RST}}$*2	PF2: General-purpose I/O	$\overline{\text{RST}}$: Reset pin	Hysteresis	CMOS	❍	-

OD: N-ch open drain, PU: Pull-up

*1: If the main oscillation clock is selected (SYSC:PFSEL = 0), the port function cannot be used.
*2: If the external reset is selected (SYSC:RSTEN = 1), the port function cannot be used. This pin is a dedicated reset pin in MB95F414H/F416H/F418H.

■ Block Diagrams of Port F

Figure 9.13-1 Block Diagram of PF0 and PF1

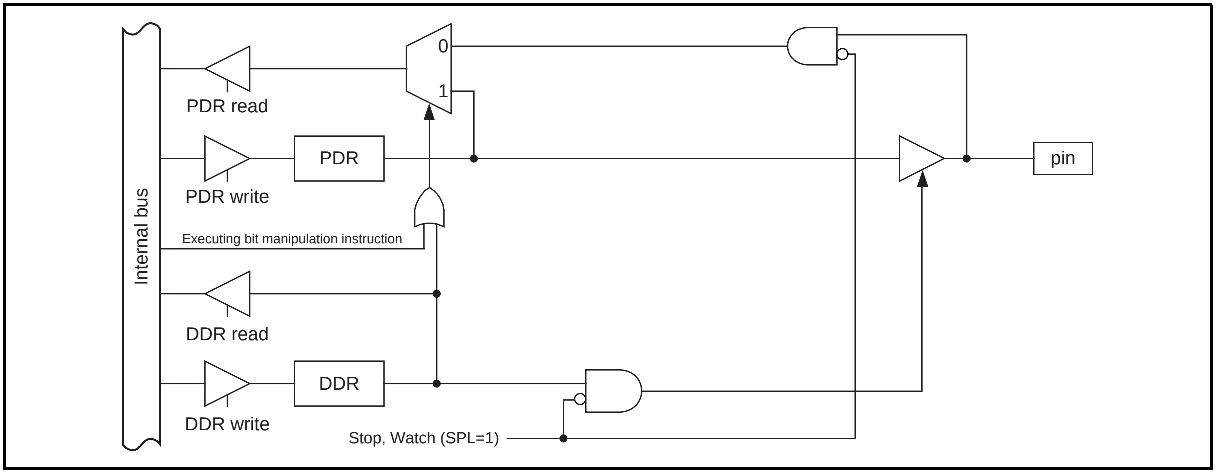
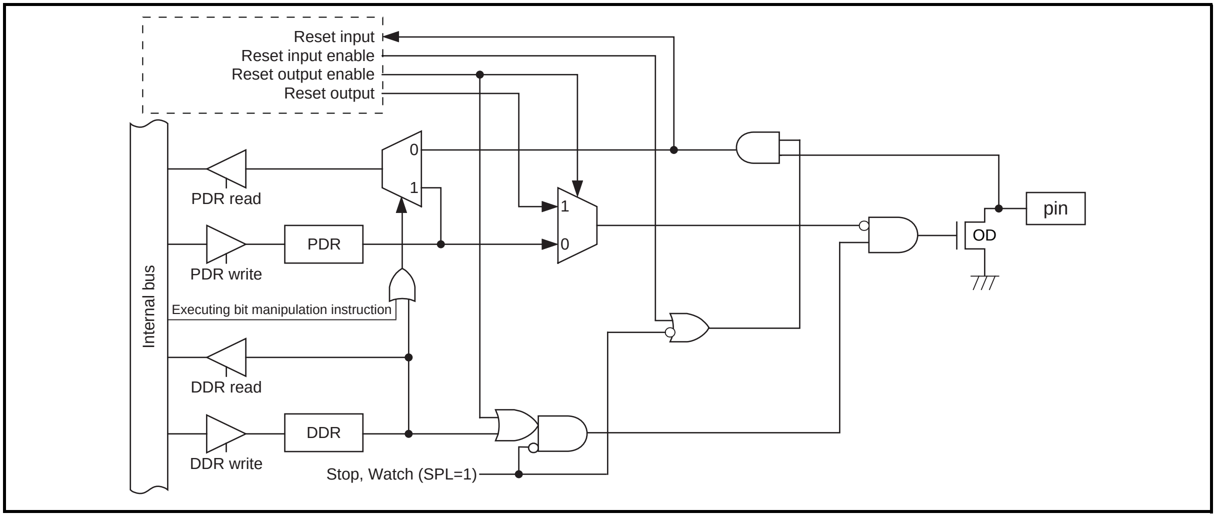


Figure 9.13-2 Block Diagram of PF2



MB95410H/470H Series

9.13.1 Port F Registers

This section describes the registers of port F.

■ Port F Register Functions

Table 9.13-2 lists the port F register functions.

Table 9.13-2 Port F Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRF	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.*
DDRF	0	Port input enabled		
	1	Port output enabled		

*: For the N-ch open drain pin, this should be Hi-Z.

Table 9.13-3 lists the correspondence between port F pins and each register bit.

Table 9.13-3 Correspondence between Registers and Pins for Port F

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	-	PF2*	PF1	PF0
PDRF DDRF	-	-	-	-	-	bit2	bit1	bit0

*: PF2/$\overline{\text{RST}}$ is a dedicated reset pin in MB95F414H/F416H/F418H.

9.13.2 Operations of Port F

This section describes the operations of port F.

■ Operations of Port F

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 9.13-4 shows the pin states of port F.

Table 9.13-4 Pin State of Port F

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled*1 (Not functional) Low*2

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*1: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

*2: Only for PF2 at power-on reset.

9.14 Port G

**Port G is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port G Configuration

Port G is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port G data register (PDRG)
- Port G direction register (DDRG)
- Port G pull-up register (PULG)

■ Port G Pin

Port G has two I/O pin.

Table 9.14-1 lists the port G pins.

Table 9.14-1 Port G Pins

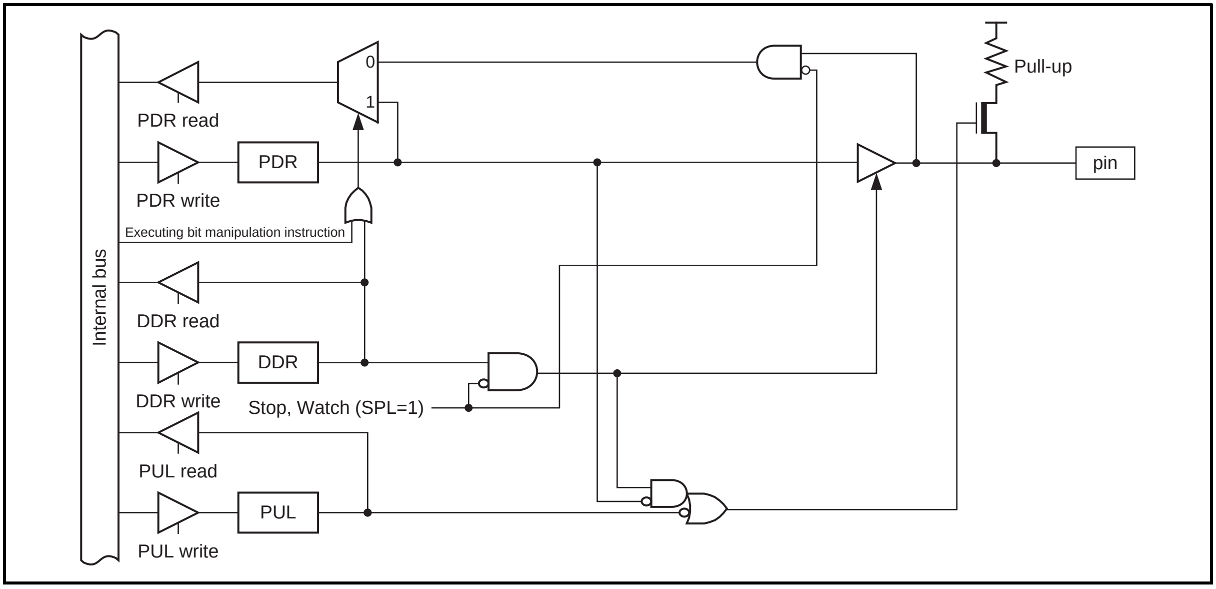
Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PG1/X0A*	PG1: General-purpose I/O	X0A: Subclock oscillation pin	Hysteresis	CMOS	-	❍
PG2/X1A*	PG2: General-purpose I/O	X1A: Subclock oscillation pin	Hysteresis	CMOS	-	❍

OD: N-ch open drain, PU: Pull-up

*: If the sub-oscillation clock is selected (SYSC:PGSEL = 0), the port function cannot be used.

■ Block Diagram of Port G

Figure 9.14-1 Block Diagram of PG2 and PG2



9.14.1 Port G Registers

This section describes the registers of port G.

■ Port G Register Functions

Table 9.14-2 lists the port G register functions.

Table 9.14-2 Port G Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRG	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRG	0	Port input enabled		
	1	Port output enabled		
PULG	0	Pull-up disabled		
	1	Pull-up enabled		

Table 9.14-3 lists the correspondence between port G pins and each register bit.

Table 9.14-3 Correspondence between Registers and Pins for Port G

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	-	PG2	PG1	-
PDRG	-	-	-	-	-	bit2	bit1	-
DDRG								
PULG								

MB95410H/470H Series

9.14.2 Operations of Port G

This section describes the operations of port G.

■ Operations of Port G

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation of the pull-up register

Setting the bit in the PUL register to "1" makes the pull-up resistor be internally connected to the pin. When the pin output is "L" level, the pull-up resistor is disconnected regardless of the value of the PUL register.

Table 9.14-4 shows the pin states of port G.

Table 9.14-4 Pin State of Port G

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

CHAPTER 10

I/O PORTS (MB95470H SERIES)

This chapter describes the functions and operations of the I/O ports.

10.1 Overview of I/O Ports

I/O ports are used to control general-purpose I/O pins.

■ Overview of I/O Ports

The I/O port has functions to output data from the CPU and capture input signals into the CPU with the port data register (PDR). The I/O direction of an individual I/O pin can be set as desired by using the corresponding to that I/O pin in the port direction register (DDR).

Table 10.1-1 lists the registers for each pin.

Table 10.1-1 List of Port Registers (1 / 2)

Register name		Read/Write	Initial value
Port 0 data register	PDR0	R, RM/W	00000000_B
Port 0 direction register	DDR0	R/W	00000000_B
Port 1 data register	PDR1	R, RM/W	00000000_B
Port 1 direction register	DDR1	R/W	00000000_B
Port 2 data register	PDR2	R, RM/W	00000000_B
Port 2 direction register	DDR2	R/W	00000000_B
Port 6 data register	PDR6	R, RM/W	00000000_B
Port 6 direction register	DDR6	R/W	00000000_B
Port 9 data register	PDR9	R, RM/W	00000000_B
Port 9 direction register	DDR9	R/W	00000000_B
Port A data register	PDRA	R, RM/W	00000000_B
Port A direction register	DDRA	R/W	00000000_B
Port B data register	PDRB	R, RM/W	00000000_B
Port B direction register	DDRB	R/W	00000000_B
Port C data register	PDRC	R, RM/W	00000000_B
Port C direction register	DDRC	R/W	00000000_B
Port E data register	PDRE	R, RM/W	00000000_B
Port E direction register	DDRE	R/W	00000000_B
Port F data register	PDRF	R, RM/W	00000000_B
Port F direction register	DDRF	R/W	00000000_B
Port G data register	PDRG	R, RM/W	00000000_B
Port G direction register	DDRG	R/W	00000000_B
Port 1 pull-up register	PUL1	R/W	00000000_B
Port 2 pull-up register	PUL2	R/W	00000000_B

Table 10.1-1 List of Port Registers (2 / 2)

Register name		Read/Write	Initial value
Port G pull-up register	PULG	R/W	00000000_B
A/D input disable register (lower)	AIDRL	R/W	00000000_B
Input level select register	ILSR	R/W	00000000_B

R/W	: Readable/writable (The read value is the same as the write value.)
R, RM/W	: Readable/writable (The read value is different from the write value. The write value is read by the read-modify-write (RMW) type of instruction.)

10.2 Port 0

**Port 0 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 0 Configuration

Port 0 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 0 data register (PDR0)
- Port 0 direction register (DDR0)
- A/D input disable register lower (AIDRL)
- Input level select register (ILSR)

■ Port 0 Pins

Port 0 has eight I/O pins.

Table 10.2-1 lists the port 0 pins.

Table 10.2-1 Port 0 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P00/INT00/ AN00/SEG29/ UO2	P00: General-purpose I/O	INT00: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN00:Analog input				
		SEG29: LCDC SEG29 output				
		UO2: UART/SIO ch. 2 data output				
P01/INT01/ AN01/SEG28/ UI2/TO00	P01: General-purpose I/O	INT01: External interrupt input	Hysteresis/ CMOS/ analog	CMOS/ LCD	-	-
		AN01: Analog input				
		SEG28: LCDC SEG28 output				
		UI2: UART/SIO ch. 2 data input				
		TO00: 8/16-bit composite timer ch. 0 output				
P02/INT02/ AN02/SEG27/ UCK2	P02: General-purpose I/O	INT02: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN02: Analog input				
		SEG27: LCDC SEG27 output				
		UCK2: UART/SIO ch. 2 clock I/O				
P03/INT03/ AN03/SEG26/ UO1	P03: General-purpose I/O	INT03: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN03: Analog input				
		SEG26: LCDC SEG26 output				
		UO1: UART/SIO ch. 1 data output				
P04/INT04/ AN04/SEG25/ UI1	P04: General-purpose I/O	INT04: External interrupt input	Hysteresis/ CMOS/ analog	CMOS/ LCD	-	-
		AN04: Analog input				
		SEG25: LCDC SEG25 output				
		UI1: UART/SIO ch. 1 data input				
P05/INT05/ AN05/SEG24/ UCK1	P05: General-purpose I/O	INT05: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN05: Analog input				
		SEG24: LCDC SEG24 output				
		UCK1: UART/SIO ch. 1 clock I/O				
P06/INT06/ AN06/SEG23	P06: General-purpose I/O	INT06: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN06: Analog input				
		SEG23: LCDC SEG23 output				
P07/INT07/ AN07/SEG22	P07: General-purpose I/O	INT07: External interrupt input	Hysteresis/ analog	CMOS/ LCD	-	-
		AN07: Analog input				
		SEG22: LCDC SEG22 output				

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port 0

Figure 10.2-1 Block Diagram of P01 and P04

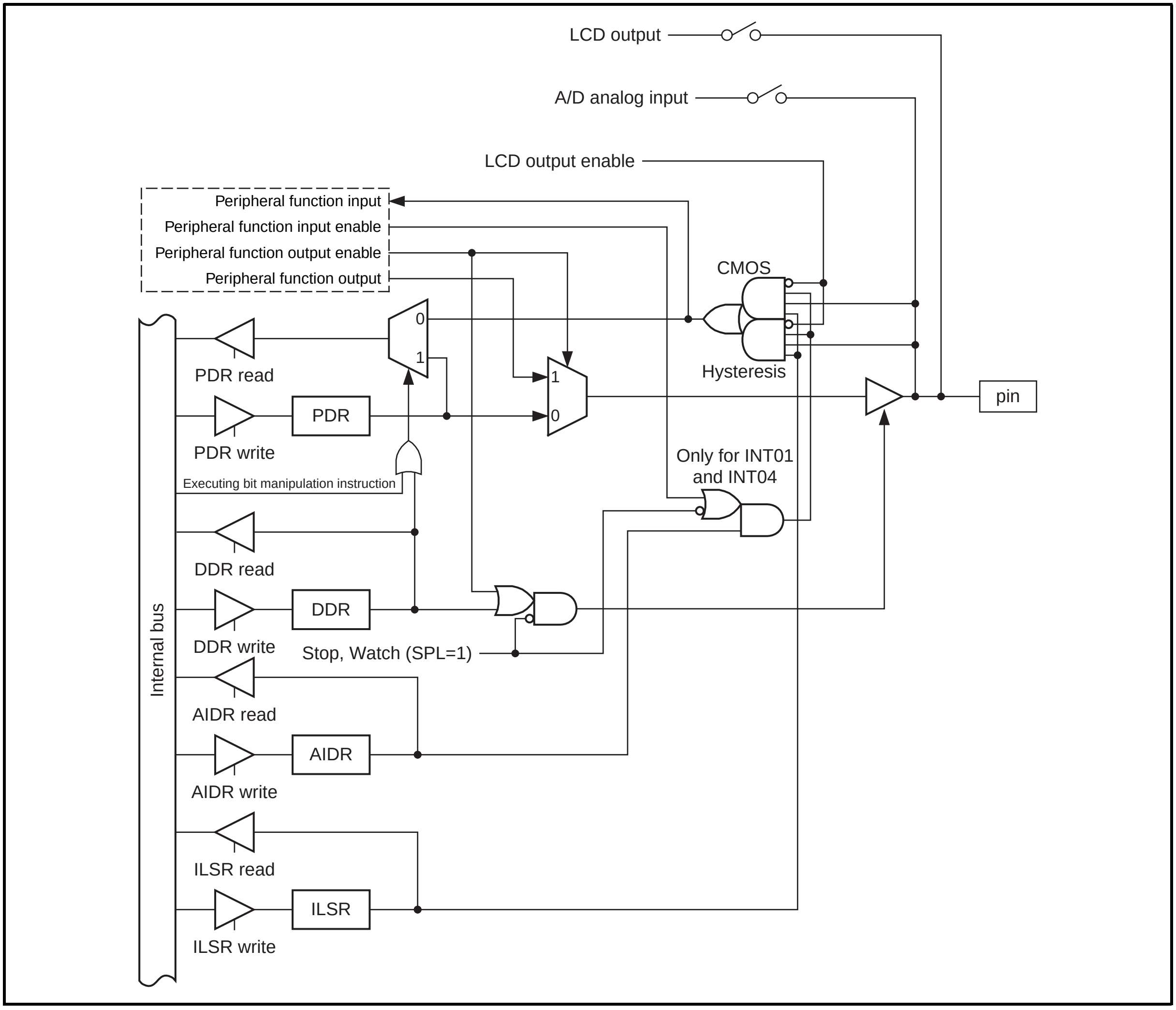
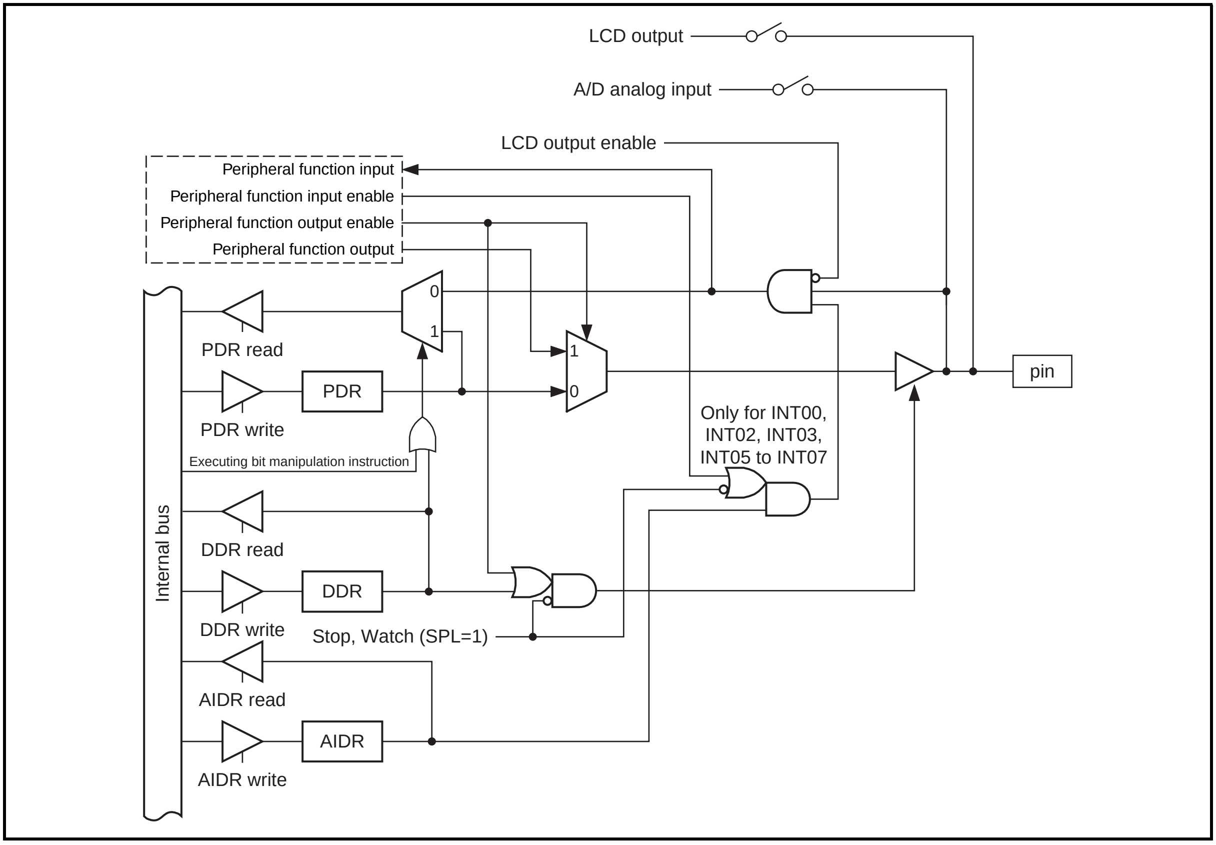


Figure 10.2-2 Block Diagram of P00, P02, P03, P05, P06 and P07


LCD output
A/D analog input
LCD output enable
Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
0
1
PDR read
PDR
PDR write
1
0
pin
Executing bit manipulation instruction
Only for INT00, INT02, INT03, INT05 to INT07
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
AIDR read
AIDR
AIDR write

10.2.1 Port 0 Registers

This section describes the registers of port 0.

■ Port 0 Register Functions

Table 10.2-2 lists the functions of the port 0 register.

Table 10.2-2 Port 0 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR0	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR0	0	Port input enabled		
	1	Port output enabled		
AIDRL	0	Analog input enabled		
	1	Port input enabled		
ILSR	0	Hysteresis input level selected		
	1	CMOS input level selected		

Table 10.2-3 lists the correspondence between port 0 pins and each register bit.

Table 10.2-3 Correspondence between Registers and Pins for Port 0

	Correspondence between related register bits and pins							
Pin name	P07	P06	P05	P04	P03	P02	P01	P00
PDR0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDR0								
AIDRL								
ILSR	-	-	-	bit4	-	-	bit1	-

10.2.2 Operations of Port 0

This section describes the operations of port 0.

■ Operations of Port 0

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR register value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 5 (LCDCE5:SEG23, SEG22) or in the LCDC enable register 6 (LCDCE6:SEG29 to SEG24) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When using an analog input shared pin as an input port, set the corresponding bit in the A/D input disable register lower (AIDRL) to "1".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit in the LCDC enable register 5 (LCDCE5:SEG23, SEG22) or in the LCDC enable register 6 (LCDCE6:SEG29 to SEG24) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register bit corresponding to the input pin of a peripheral function to "0".
- When using the analog input shared pin as another peripheral function input pin, configure it as an input port, which is the same as the operation as an input port.
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit in the LCDC enable register 5 (LCDCE5:SEG23, SEG22) or in the LCDC enable register 6 (LCDCE6:SEG29 to SEG24) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled. As for a pin shared with analog input, its port input is disabled because the A/D input disable register lower (AIDRL) is initialized to "0".

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" and the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open. However, if the interrupt input is enabled for the external interrupt (INT07 to INT00), the input is enabled and not blocked.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation as an analog input pin

- Set the bit in the DDR register corresponding to the analog input pin to "0" and the bit corresponding to that pin in the AIDRL register to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions. In addition, set the corresponding bit in the PUL register to "0".

● Operation as an external interrupt input pin

- Set the bit in the DDR register corresponding to the external interrupt input pin to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- The pin value is always input to the external interrupt circuit. When using a pin for a function other than the interrupt, disable the external interrupt function corresponding to that pin.

● Operation of the input level select register

- Setting bit1 and bit4 in ILSR to "1" changes P01 and P04 respectively from the hysteresis input level to the CMOS input level.
- For pins other than P01 and P04, the CMOS input level cannot be selected, but only the hysteresis input level can be selected.
- When changing the input level of P01 or of P04, ensure that all shared peripheral functions have been stopped.

Table 10.2-4 shows the pin states of port 0.

Table 10.2-4 Pin State of Port 0

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port/ peripheral function I/O	Hi-Z (the pull-up setting is enabled) Input cutoff (If the external interrupt function is enabled, the external interrupt can be input.)	Hi-Z Input disabled*

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input disabled" means the state that the operation of the input gate adjacent to the pin is disabled.

10.3 Port 1

**Port 1 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 1 Configuration

Port 1 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 1 data register (PDR1)
- Port 1 direction register (DDR1)
- Port 1 pull-up register (PUL1)
- Input level select register (ILSR)

■ Port 1 Pins

Port 1 has eight I/O pins.

Table 10.3-1 lists the port 1 pins.

Table 10.3-1 Port 1 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P10/UI0/TO0	P10: General-purpose I/O	UI0: UART/SIO ch. 0 data input	Hysteresis/ CMOS	CMOS	-	❍
		TO0: 16-bit reload timer output				
P11/UO0	P11: General-purpose I/O	UO0: UART/SIO ch. 0 data output	Hysteresis	CMOS	-	❍
P12/DBG	P12: General-purpose I/O	DBG: On-chip debug communication pin	Hysteresis	CMOS	❍	-
P13/ADTG/ TO01	P13: General-purpose I/O	ADTG: A/D trigger input	Hysteresis	CMOS	-	❍
		TO01: 8/16-bit composite timer ch. 0 output				
P14/UCK0/ EC0/TI0	P14: General-purpose I/O	UCK0: UART/SIO ch. 0 clock I/O	Hysteresis	CMOS	-	❍
		EC0: 8/16-bit composite timer ch. 0 clock input				
		TI0: 16-bit reload timer input				
P15/PPG11/ SEG31	P15: General-purpose I/O	PPG11: 8/16-bit PPG ch. 1 output	Hysteresis	CMOS/ LCD	-	-
		SEG31: LCDC SEG31 output				
P16/PPG10/ SEG30	P16: General-purpose I/O	PPG10: 8/16-bit PPG ch. 1 output	Hysteresis	CMOS/ LCD	-	-
		SEG30: LCDC SEG30 output				
P17/CMPO	P17: General-purpose I/O	CMPO: Voltage comparator output	Hysteresis	CMOS	-	❍

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port 1

Figure 10.3-1 Block Diagram of P10

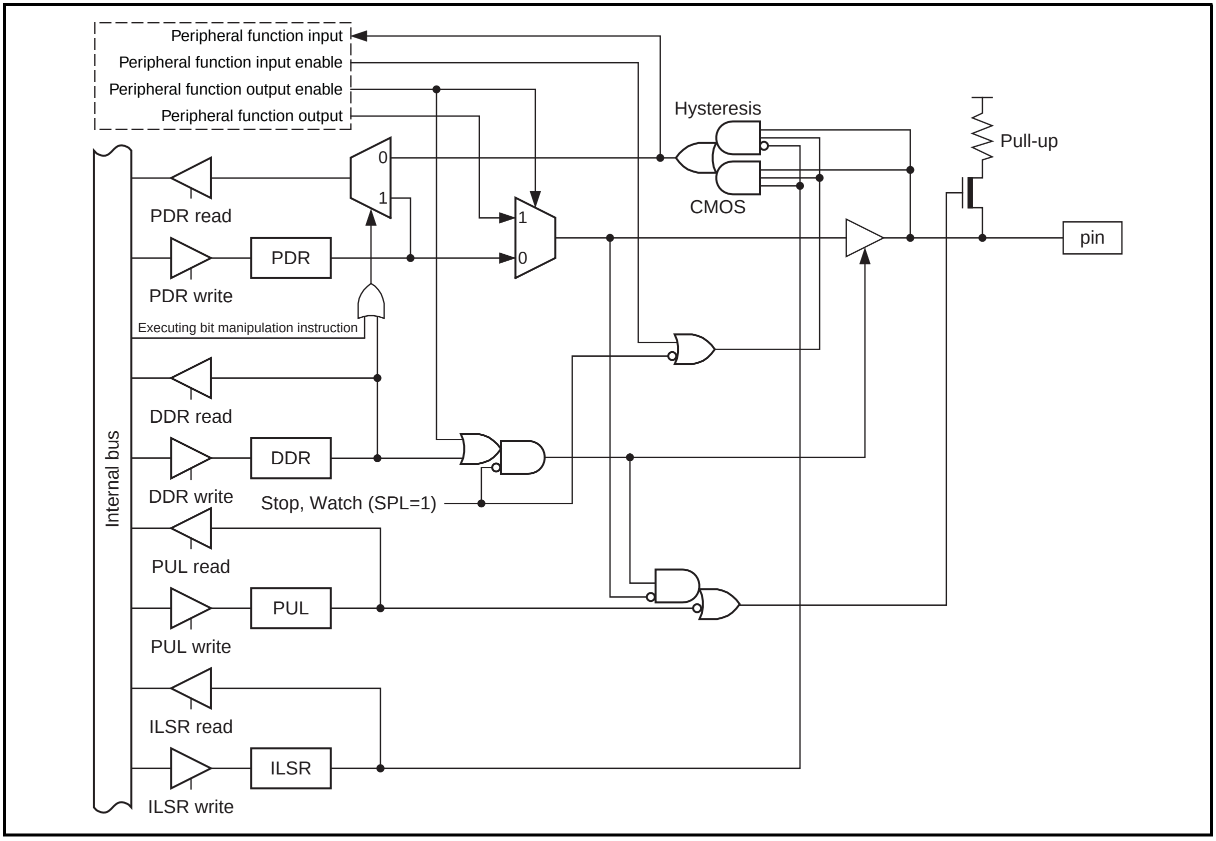

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Hysteresis
CMOS
Pull-up
pin
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write
ILSR read
ILSR
ILSR write
Internal bus

Figure 10.3-2 Block Diagram of P12

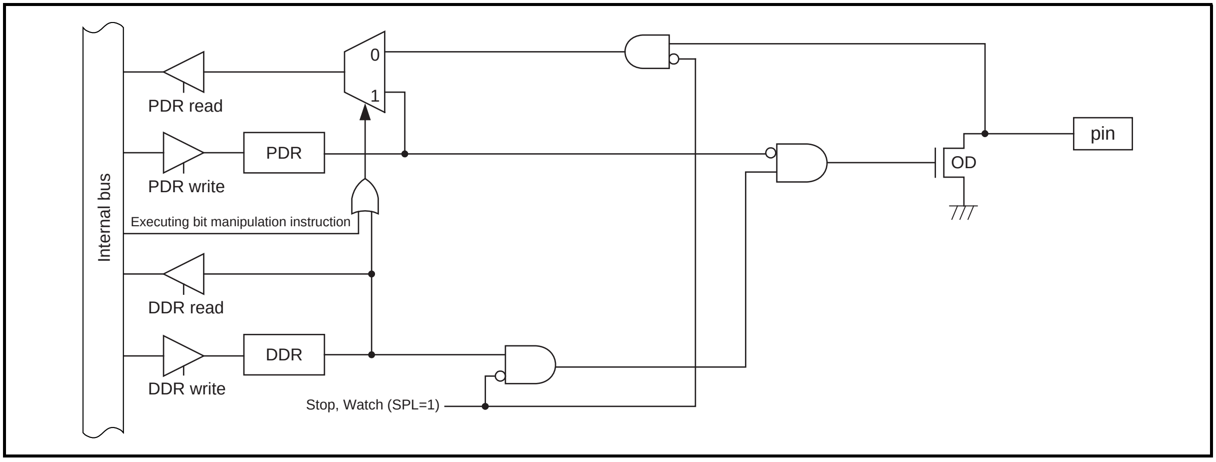


Figure 10.3-3 Block Diagram of P11, P13, P14 and P17

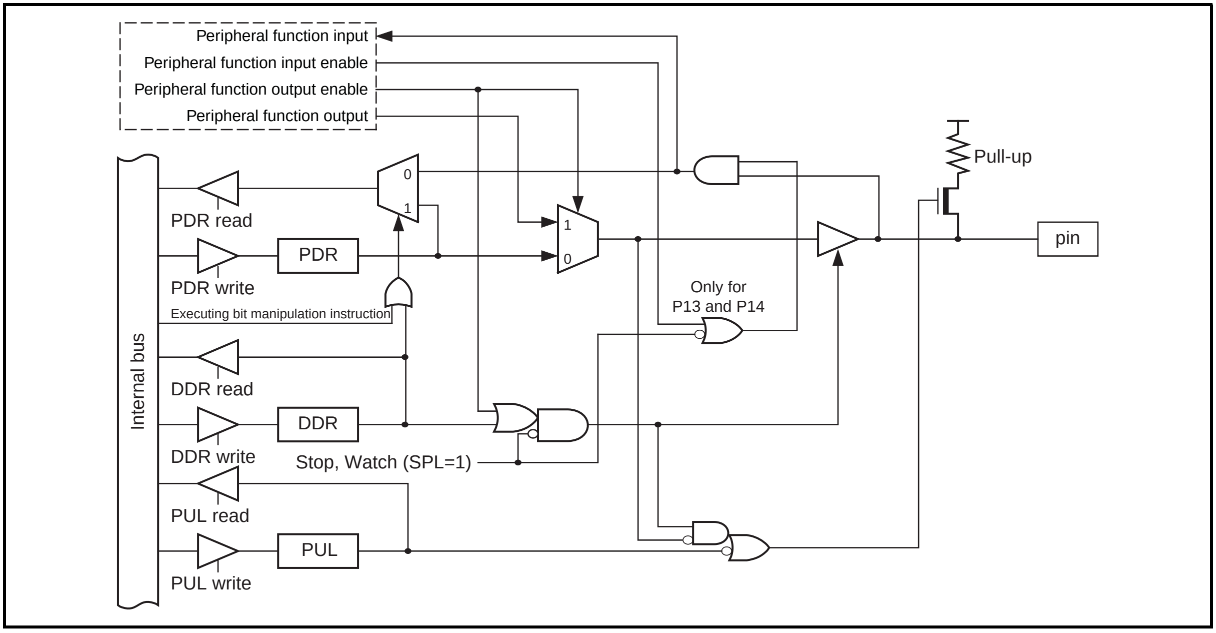
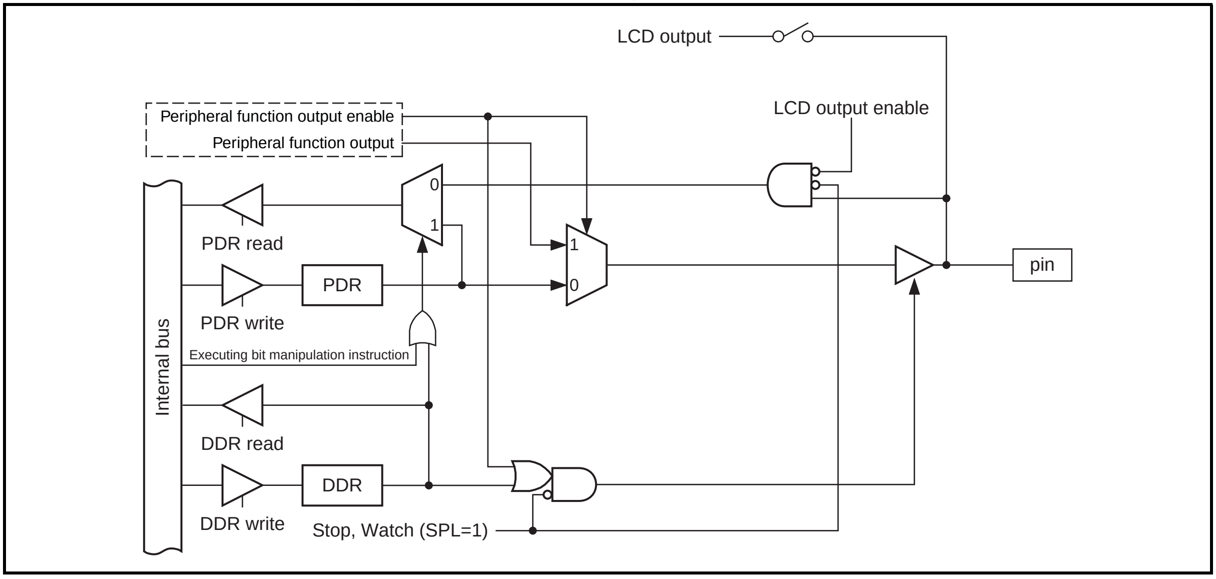

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
Only for
P13 and P14
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write

Figure 10.3-4 Block Diagram of P15 and P16



10.3.1 Port 1 Registers

This section describes the registers of port 1.

■ Port 1 Register Functions

Table 10.3-2 lists the port 1 register functions.

Table 10.3-2 Port 1 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR1	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.*
DDR1	0	Port input enabled		
	1	Port output enabled		
PUL1	0	Pull-up disabled		
	1	Pull-up enabled		
ILSR	0	Hysteresis input level selected		
	1	CMOS input level selected		

*: For the N-ch open drain pin, this should be Hi-Z.

Table 10.3-3 lists the correspondence between port 1 pins and each register bit.

Table 10.3-3 Correspondence between Registers and Pins for Port 1

	Correspondence between related register bits and pins							
Pin name	P17	P16	P15	P14	P13	P12	P11	P10
PDR1	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDR1								
PUL1	bit7	-	-	bit4	bit3	-	bit1	bit0
ILSR	-	-	-	-	-	-	-	bit0

10.3.2 Operations of Port 1

This section describes the operations of port 1.

■ Operations of Port 1

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR register value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit (SEG31, SEG30) in the LCDC enable register 6 (LCDCE6) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set aa corresponding segment select bit (SEG31, SEG30) in the LCDC enable register 6 (LCDCE6) "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register corresponding to the input pin of a peripheral function to "0".
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used

to read the PDR register, the PDR register value is returned.

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit (SEG31, SEG30) in the LCDC enable register 6 (LCDCE6) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" and the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation of the pull-up control register

Setting the bit in the PUL register to "1" makes the pull-up resistor be internally connected to the pin. When the pin output is "L" level, the pull-up resistor is disconnected regardless of the value of the PUL register.

● Operation of the input level select register

- Setting bit0 in ILSR to "1" changes only P10 from the hysteresis input level to the CMOS input level. When the same bit is set to "0", the input level of P10 should become the hysteresis input level.
- For pins other than P10, the CMOS input level cannot be selected, but only the hysteresis input level can be selected.
- When changing the input level of P10, ensure that all its shared peripheral functions have been stopped.

Table 10.3-4 shows the pin states of port 1.

Table 10.3-4 Pin State of Port 1

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port/ peripheral function I/O	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.4 Port 2

Port 2 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port 2 Configuration

Port 2 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 2 data register (PDR2)
- Port 2 direction register (DDR2)
- Port 2 pull-up register (PUL2)
- Input level select register (ILSR)

■ Port 2 Pins

Port 2 has four I/O pins.

Table 10.4-1 lists the port 2 pins.

Table 10.4-1 Port 2 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P20/PPG00/ CMPN	P20: General-purpose I/O	PPG00: 8/16-bit PPG ch. 0 output CMPN: Voltage comparator N ch input	Hysteresis/ analog	CMOS	-	❍
P21/PPG01/ CMPP	P21: General-purpose I/O	PPG01: 8/16-bit PPG ch. 0 output CMPP: Voltage comparator P ch input	Hysteresis/ analog	CMOS	-	❍
P22/SCL	P22: General-purpose I/O	SCL: I^{2}C clock I/O	Hysteresis/ CMOS	CMOS	❍	-
P23/SDA	P23: General-purpose I/O	SDA: I^{2}C data I/O	Hysteresis/ CMOS	CMOS	❍	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port 2

Figure 10.4-1 Block Diagram of P20 and P21

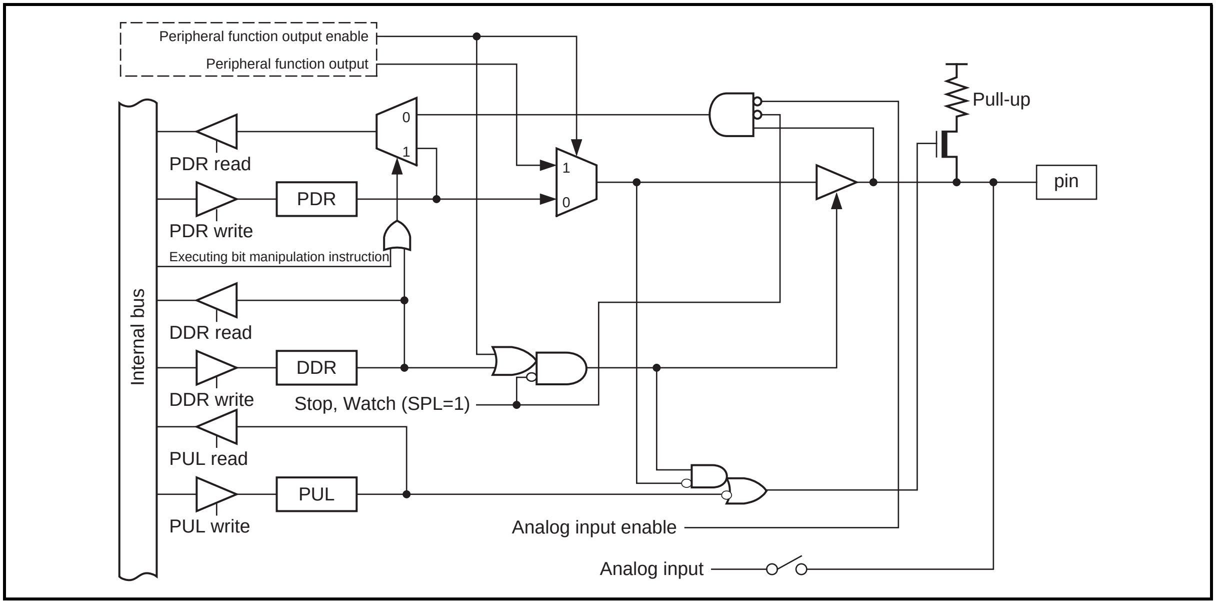
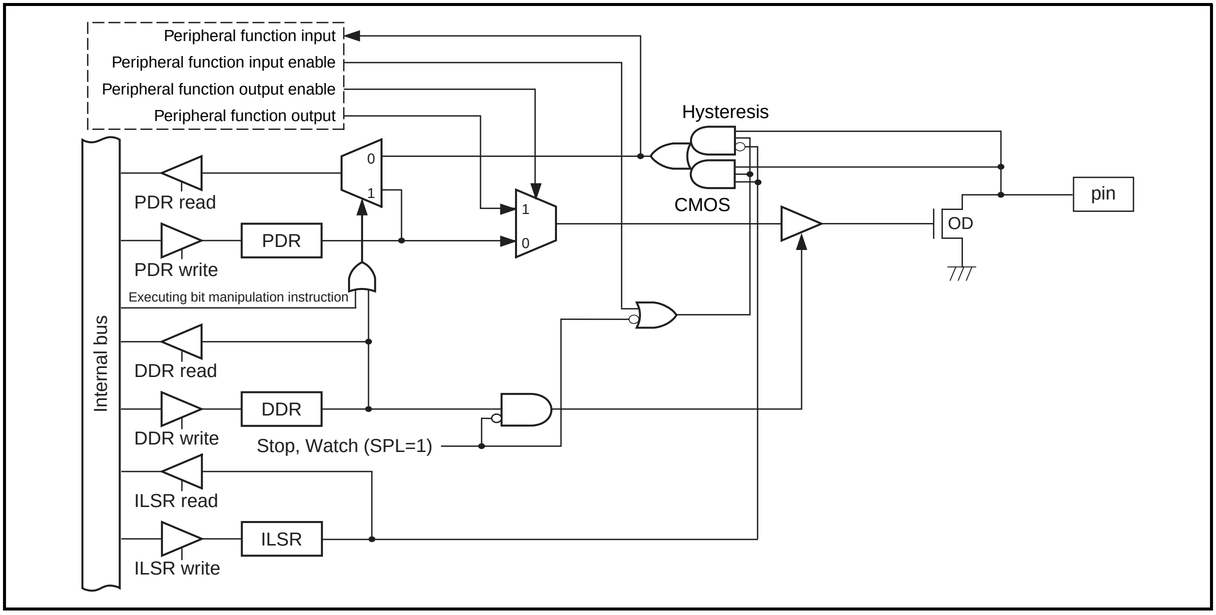


Figure 10.4-2 Block Diagram of P22 and P23


Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Hysteresis
CMOS
pin
OD
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
ILSR read
ILSR
ILSR write

10.4.1 Port 2 Registers

This section describes the registers of port 2.

■ Port 2 Register Functions

Table 10.4-2 lists the port 2 register functions.

Table 10.4-2 Port 2 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR2	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.*
DDR2	0	Port input enabled		
	1	Port output enabled		
PUL2	0	Pull-up disabled		
	1	Pull-up enabled		
ILSR	0	Hysteresis input level selected		
	1	CMOS input level selected		

*: For the N-ch open drain pin, this should be Hi-Z.

Table 10.4-3 lists the correspondence between port 2 pins and each register bit.

Table 10.4-3 Correspondence Between Registers and Pins for Port 2

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	P23	P22	P21	P20
PDR2	-	-	-	-	bit3	bit2	bit1	bit0
DDR2								
PUL2	-	-	-	-	-	-	bit1	bit0
ILSR	-	-	-	-	bit3	bit2	-	-

10.4.2 Operations of Port 2

This section describes the operations of port 2.

■ Operations of Port 2

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register corresponding to the input pin of a peripheral function to "0".
- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" and the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation as an analog input pin

- Setting the voltage comparator analog input disable bit in the voltage comparator control register (CMR0:VCID) to "0" enables the analog input function of an analog input pin regardless of the settings of the PDR register.
- To disable the analog input function of an analog input pin, set the VCID bit in the CMR0 register to "1".

● Operation of the pull-up control register

Setting the bit in the PUL register to "1" makes the pull-up resistor be internally connected to the pin. When the pin output is "L" level, the pull-up resistor is disconnected regardless of the value of the PUL register.

● Operation of the input level select register

- Setting bit2 and bit3 in ILSR to "1" changes P22 and P23 respectively from the hysteresis input level to the CMOS input level. When the same bit is set to "0", the input levels of P22 and P23 become the hysteresis input level.
- For pins other than P22 and P23, the CMOS input level cannot be selected, but only the hysteresis input level can be selected.
- When changing the input levels of P22 and P23, ensure that all shared peripheral functions have been stopped.

Table 10.4-4 shows the pin states of port 2.

Table 10.4-4 Pin State of Port 2

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port/peripheral function I/O	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.5 Port 6

**Port 6 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port 6 Configuration

Port 6 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 6 data register (PDR6)
- Port 6 direction register (DDR6)

■ Port 6 Pins

Port 6 has eight I/O pins.

Table 10.5-1 lists the port 6 pins.

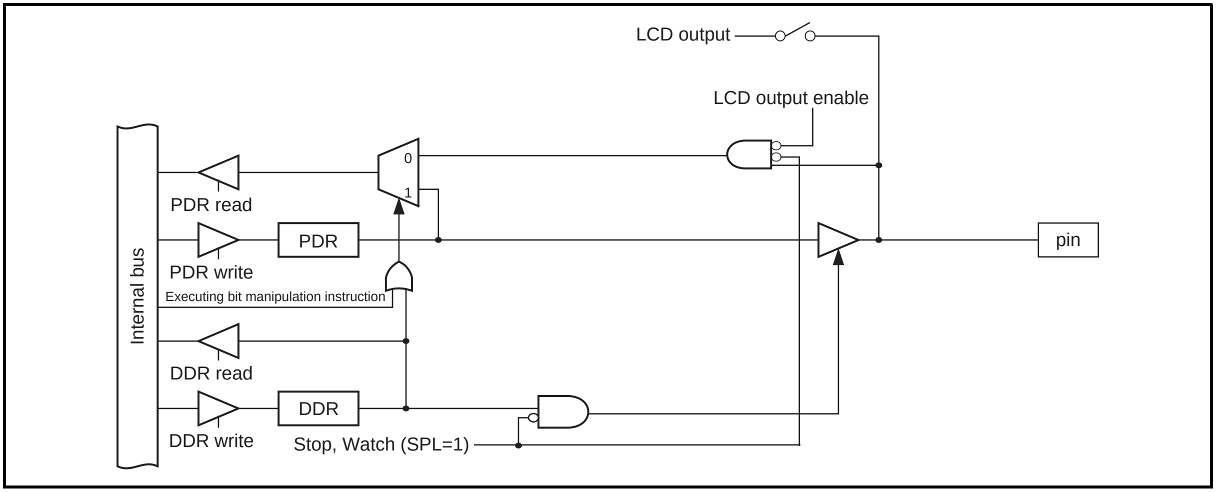
Table 10.5-1 Port 6 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P60/SEG06	P60: General-purpose I/O	SEG06: LCDC SEG06 output	Hysteresis	CMOS/ LCD	-	-
P61/SEG07	P61: General-purpose I/O	SEG07: LCDC SEG07 output	Hysteresis	CMOS/ LCD	-	-
P62/SEG08	P62: General-purpose I/O	SEG08: LCDC SEG08 output	Hysteresis	CMOS/ LCD	-	-
P63/SEG09	P63: General-purpose I/O	SEG09: LCDC SEG09 output	Hysteresis	CMOS/ LCD	-	-
P64/SEG10	P64: General-purpose I/O	SEG10: LCDC SEG10 output	Hysteresis	CMOS/ LCD	-	-
P65/SEG11	P65: General-purpose I/O	SEG11: LCDC SEG11 output	Hysteresis	CMOS/ LCD	-	-
P66/SEG12	P66: General-purpose I/O	SEG12: LCDC SEG12 output	Hysteresis	CMOS/ LCD	-	-
P67/SEG13	P67: General-purpose I/O	SEG13: LCDC SEG13 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port 6

Figure 10.5-1 Block Diagram of P60, P61, P62, P63, P64, P65, P66 and P67



10.5.1 Port 6 Registers

This section describes the registers of port 6.

■ Port 6 Register Functions

Table 10.5-2 lists the port 6 register functions.

Table 10.5-2 Port 6 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR6	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR6	0	Port input enabled		
	1	Port output enabled		

Table 10.5-3 lists the correspondence between port 6 pins and each register bit.

Table 10.5-3 Correspondence between Registers and Pins for Port 6

	Correspondence between related register bits and pins							
Pin name	P67	P66	P65	P64	P63	P62	P61	P60
PDR6	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDR6								

10.5.2 Operations of Port 6

This section describes the operations of port 6.

■ Operations of Port 6

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 3 (LCDCE3:SEG07, SEG06) or in the LCDC enable register 4 (LCDCE4:SEG13 to SEG08) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit in the LCDC enable register 3 (LCDCE3:SEG07, SEG06) or in the LCDC enable register 4 (LCDCE4:SEG13 to SEG08) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit in the LCDC enable register 3 (LCDCE3:SEG07, SEG06) or in the LCDC enable register 4 (LCDCE4:SEG13 to SEG08) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 10.5-4 shows the pin states of port 6.

Table 10.5-4 Pin State of Port 6

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.6 Port 9

Port 9 is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port 9 Configuration

Port 9 is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port 9 data register (PDR9)
- Port 9 direction register (DDR9)

■ Port 9 Pins

Port 9 has four I/O pins.

Table 10.6-1 lists the port 9 pins.

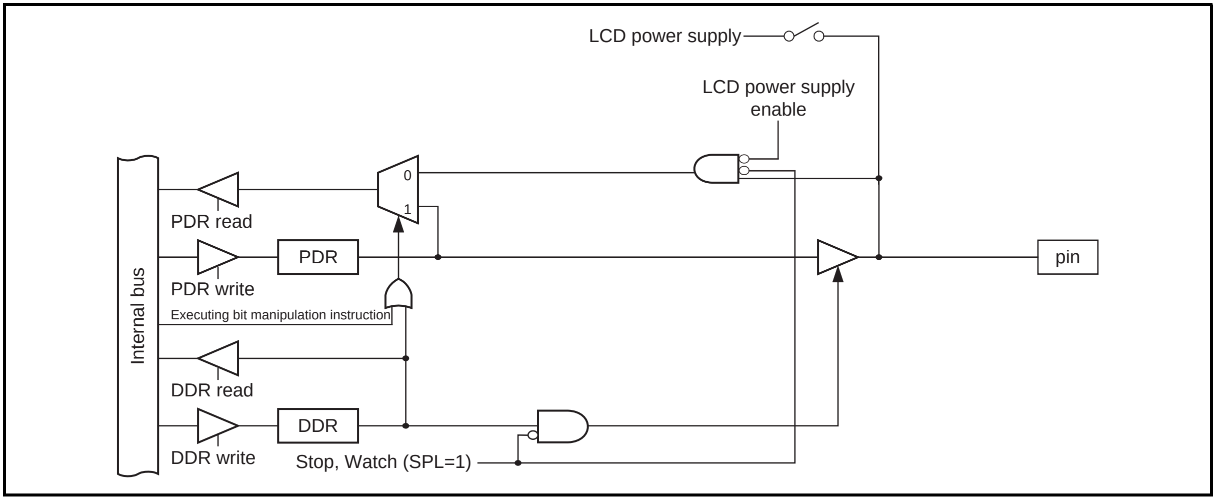
Table 10.6-1 Port 9 Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
P90/V4	P90: General-purpose I/O	V4: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-
P91/V3	P91: General-purpose I/O	V3: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-
P92/V2	P92: General-purpose I/O	V2: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-
P93/V1	P93: General-purpose I/O	V1: Power supply pin for LCDC drive	Hysteresis	CMOS	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port 9

Figure 10.6-1 Block Diagram of P90, P91, P92 and P93



10.6.1 Port 9 Registers

This section describes the registers of port 9.

■ Port 9 Register Functions

Table 10.6-2 lists the port 9 register functions.

Table 10.6-2 Port 9 Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDR9	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDR9	0	Port input enabled		
	1	Port output enabled		

Table 10.6-3 lists the correspondence between port 9 pins and each register bit.

Table 10.6-3 Correspondence between Registers and Pins for Port 9

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	P93	P92	P91	P90
PDR9 DDR9	-	-	-	-	bit3	bit2	bit1	bit0

10.6.2 Operations of Port 9

This section describes the operations of port 9.

■ Operations of Port 9

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set the bit (VE4 to VE1) corresponding to that pin in the LCDC enable register 1 (LCDCE1) to "0".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set the bit (VE4 to VE1) corresponding to that pin in the LCDC enable register 1 (LCDCE1) to "0".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operations as LCDC pins

- Set the DDR register bit corresponding to a desired LCDC pin to "0".
- Set the V1 select bit (VE1), the V2 select bit (VE2), the V3 select bit (VE3) and the V4 select bit (VE4) in the LCDC enable register 1 (LCDCE1) to "1".

Table 10.6-4 shows the pin states of port 9.

Table 10.6-4 Pin State of Port 9

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.7 Port A

**Port A is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port A Configuration

Port A is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port A data register (PDRA)
- Port A direction register (DDRA)

■ Port A Pins

Port A has eight I/O pins.

Table 10.7-1 lists the port A pins.

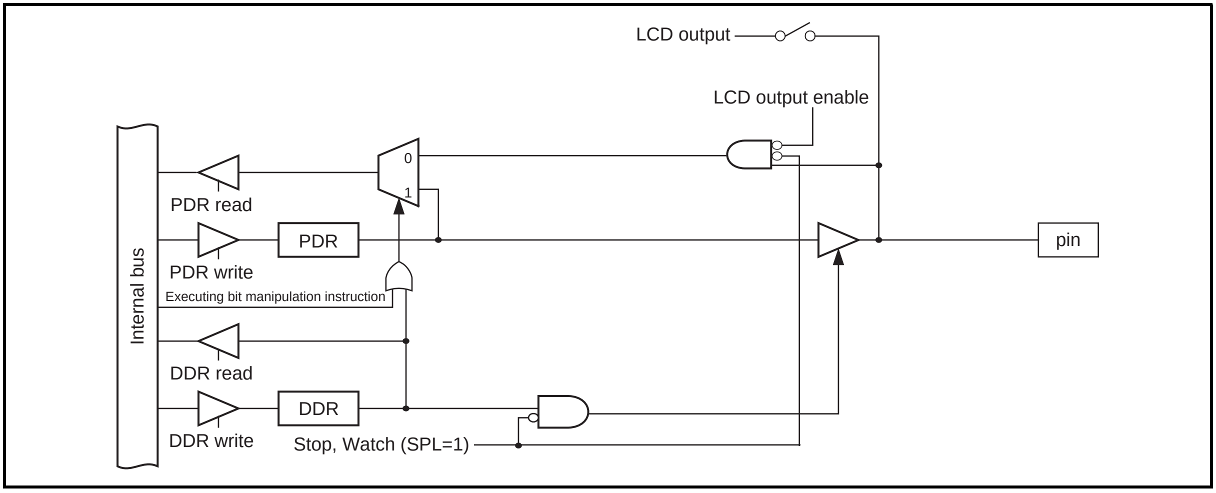
Table 10.7-1 Port A Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PA0/COM0	PA0: General-purpose I/O	COM0: LCDC COM0 output	Hysteresis	CMOS/ LCD	-	-
PA1/COM1	PA1: General-purpose I/O	COM1: LCDC COM1 output	Hysteresis	CMOS/ LCD	-	-
PA2/COM2	PA2: General-purpose I/O	COM2: LCDC COM2 output	Hysteresis	CMOS/ LCD	-	-
PA3/COM3	PA3: General-purpose I/O	COM3: LCDC COM3 output	Hysteresis	CMOS/ LCD	-	-
PA4/COM4	PA4: General-purpose I/O	COM4: LCDC COM4 output	Hysteresis	CMOS/ LCD	-	-
PA5/COM5	PA5: General-purpose I/O	COM5: LCDC COM5 output	Hysteresis	CMOS/ LCD	-	-
PA6/COM6	PA6: General-purpose I/O	COM6: LCDC COM6 output	Hysteresis	CMOS/ LCD	-	-
PA7/COM7	PA7: General-purpose I/O	COM7: LCDC COM7 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port A

Figure 10.7-1 Block Diagram of PA0, PA1, PA2, PA3, PA4, PA5, PA6 and PA7



10.7.1 Port A Registers

This section describes the registers of port A.

■ Port A Register Functions

Table 10.7-2 lists the port A register functions.

Table 10.7-2 Port A Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRA	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRA	0	Port input enabled		
	1	Port output enabled		

Table 10.7-3 lists the correspondence between port A pins and each register bit.

Table 10.7-3 Correspondence between Registers and Pins for Port A

	Correspondence between related register bits and pins							
Pin name	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
PDRA DDRA	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0

10.7.2 Operations of Port A

This section describes the operations of port A.

■ Operations of Port A

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding common select bit (COM7 to COM0) in the LCDC enable register 2 (LCDCE2) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding common select bit (COM7 to COM0) in the LCDC enable register 2 (LCDCE2) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

- ● Operation as an LCDC common output

- Set the DDR register bit corresponding to a desired LCDC common output pin to "0".
- Select the common output by setting a corresponding common select bit (COM7 to COM0) in the LCDC enable register 2 (LCDCE2) to "1", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

Table 10.7-4 shows the pin states of port A.

Table 10.7-4 Pin State of Port A

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.8 Port B

Port B is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port B Configuration

Port B is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port B data register (PDRB)
- Port B direction register (DDRB)

■ Port B Pins

Port B has two I/O pins.

Table 10.8-1 lists the port B pins.

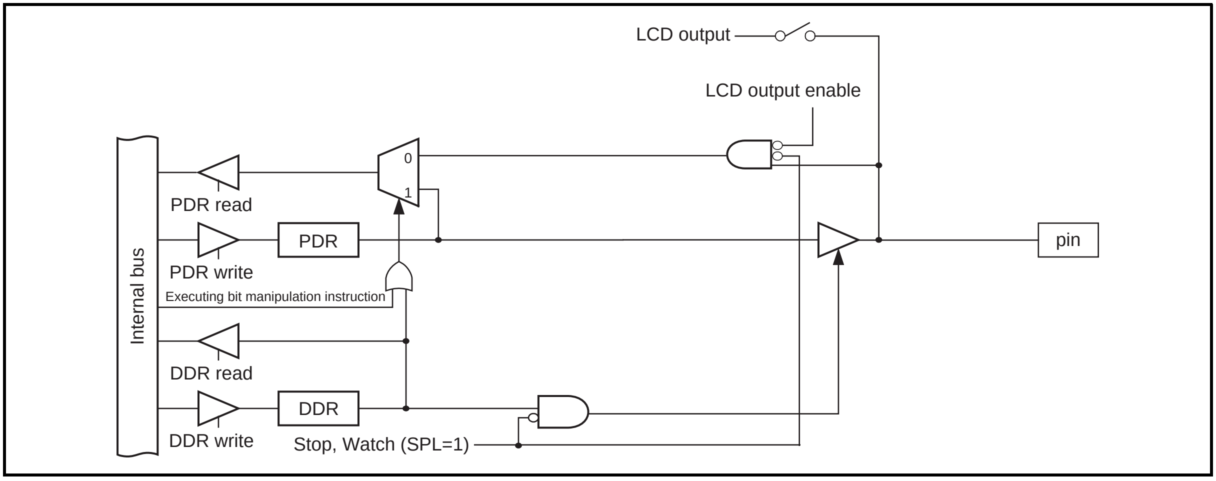
Table 10.8-1 Port B Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PB0/SEG00	PB0: General-purpose I/O	SEG00: LCDC SEG00 output	Hysteresis	CMOS/ LCD	-	-
PB1/SEG01	PB1: General-purpose I/O	SEG01: LCDC SEG01 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port B

Figure 10.8-1 Block Diagram of PB0 and PB1



10.8.1 Port B Registers

This section describes the registers of port B.

■ Port B Register Functions

Table 10.8-2 lists the port B register functions.

Table 10.8-2 Port B Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRB	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRB	0	Port input enabled		
	1	Port output enabled		

Table 10.8-3 lists the correspondence between port B pins and each register bit.

Table 10.8-3 Correspondence between Registers and Pins for Port B

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	-	-	PB1	PB0
PDRB DDRB	-	-	-	-	-	-	bit1	bit0

10.8.2 Operations of Port B

This section describes the operations of port B.

■ Operations of Port B

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit (SEG01, SEG00) in the LCDC enable register 3 (LCDCE3) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit (SEG01, SEG00) in the LCDC enable register 3 (LCDCE3) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit (SEG01, SEG00) in the LCDC enable register 3 (LCDCE3) "1", and then set the port input control bit (PICTL) in LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 10.8-4 shows the pin states of port B.

Table 10.8-4 Pin State of Port B

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.9 Port C

Port C is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port C Configuration

Port C is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port C data register (PDRC)
- Port C direction register (DDRC)

■ Port C Pins

Port C has four I/O pins.

Table 10.9-1 lists the port C pins.

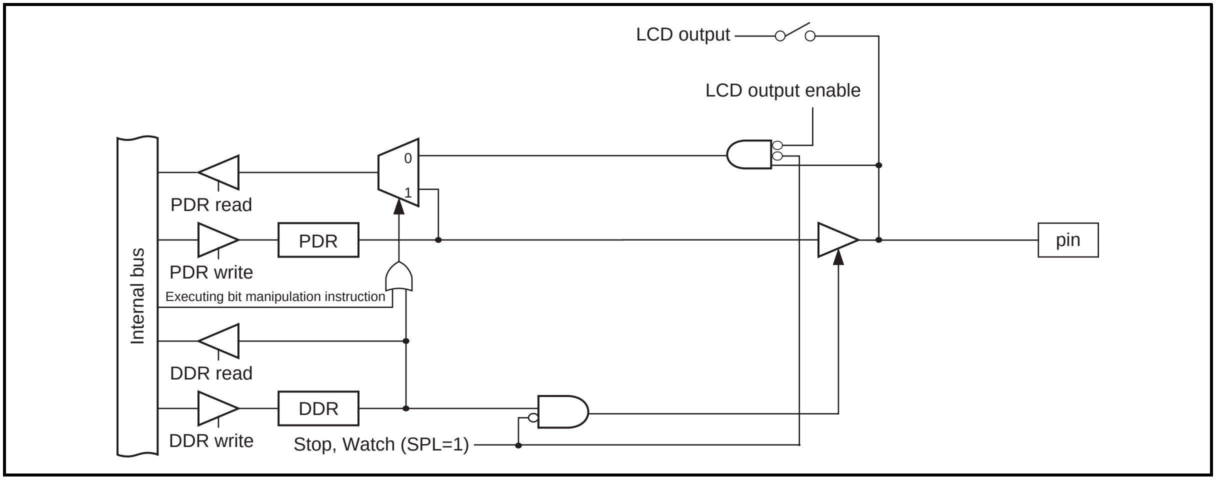
Table 10.9-1 Port C Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PC0/SEG02	PC0: General-purpose I/O	SEG02: LCDC SEG02 output	Hysteresis	CMOS/ LCD	-	-
PC1/SEG03	PC1: General-purpose I/O	SEG03: LCDC SEG03 output	Hysteresis	CMOS/ LCD	-	-
PC2/SEG04	PC2: General-purpose I/O	SEG04: LCDC SEG04 output	Hysteresis	CMOS/ LCD	-	-
PC3/SEG05	PC3: General-purpose I/O	SEG05: LCDC SEG05 output	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagram of Port C

Figure 10.9-1 Block Diagram of PC0, PC1, PC2 and PC3



10.9.1 Port C Registers

This section describes the registers of port C.

■ Port C Register Functions

Table 10.9-2 lists the port C register functions.

Table 10.9-2 Port C Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRC	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRC	0	Port input enabled		
	1	Port output enabled		

Table 10.9-3 lists the correspondence between port C pins and each register bit.

Table 10.9-3 Correspondence between Registers and Pins for Port C

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	PC3	PC2	PC1	PC0
PDRC	-	-	-	-	bit3	bit2	bit1	bit0
DDRC								

10.9.2 Operations of Port C

This section describes the operations of port C.

■ Operations of Port C

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit (SEG05 to SEG02) in the LCDC enable register 3 (LCDCE3) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit (SEG05 to SEG02) in the LCDC enable register 3 (LCDCE3) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit (SEG05 to SEG02) in the LCDC enable register 3 (LCDCE3) to "1", and then set the port input control bit (PICTL) in LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 10.9-4 shows the pin states of port C.

Table 10.9-4 Pin State of Port C

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.10 Port E

**Port E is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port E Configuration

Port E is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port E data register (PDRE)
- Port E direction register (DDRE)

■ Port E Pins

Port E has eight I/O pins.

Table 10.10-1 lists the port E pins.

Table 10.10-1 Port E Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PE0/SEG14	PE0: General-purpose I/O	SEG14: LCDC SEG14 output	Hysteresis	CMOS/ LCD	-	-
PE1/SEG15	PE1: General-purpose I/O	SEG15: LCDC SEG15 output	Hysteresis	CMOS/ LCD	-	-
PE2/SEG16	PE2: General-purpose I/O	SEG16: LCDC SEG16 output	Hysteresis	CMOS/ LCD	-	-
PE3/SEG17	PE3: General-purpose I/O	SEG17: LCDC SEG17 output	Hysteresis	CMOS/ LCD	-	-
PE4/SEG18	PE4: General-purpose I/O	SEG18: LCDC SEG18 output	Hysteresis	CMOS/ LCD	-	-
PE5/SEG19/ TO11	PE5: General-purpose I/O	SEG19: LCDC SEG19 output TO11: 8/16-bit composite timer ch. 1 output	Hysteresis	CMOS/ LCD	-	-
PE6/SEG20/ TO10	PE6: General-purpose I/O	SEG20: LCDC SEG20 output TO10: 8/16-bit composite timer ch. 1 output	Hysteresis	CMOS/ LCD	-	-
PE7/SEG21/ EC1	PE7: General-purpose I/O	SEG21: LCDC SEG21 output EC1: 8/16-bit composite timer ch. 1 clock input	Hysteresis	CMOS/ LCD	-	-

OD: N-ch open drain, PU: Pull-up

■ Block Diagrams of Port E

Figure 10.10-1 Block Diagram of PE0, PE1, PE2, PE3 and PE4

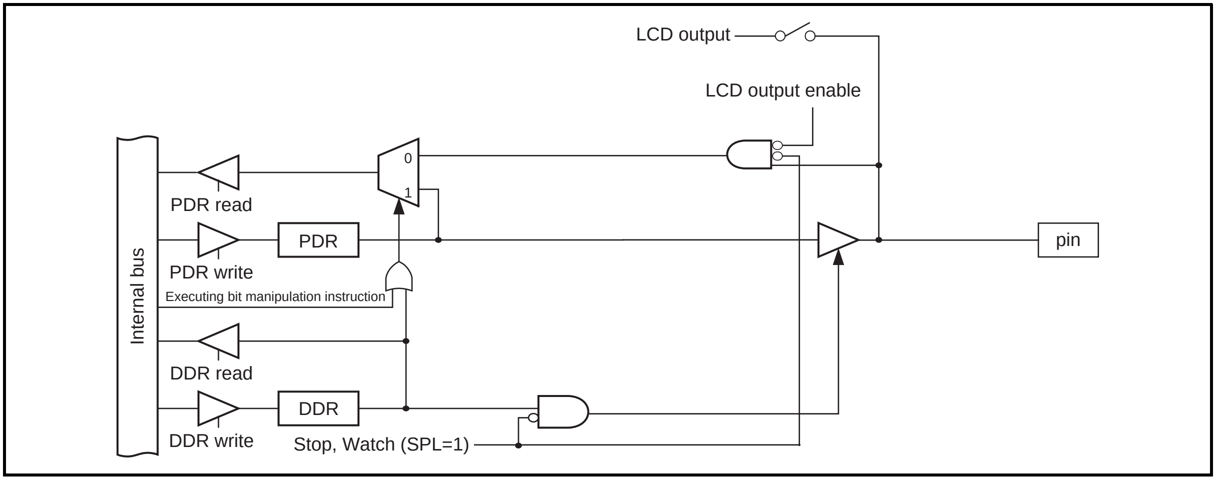
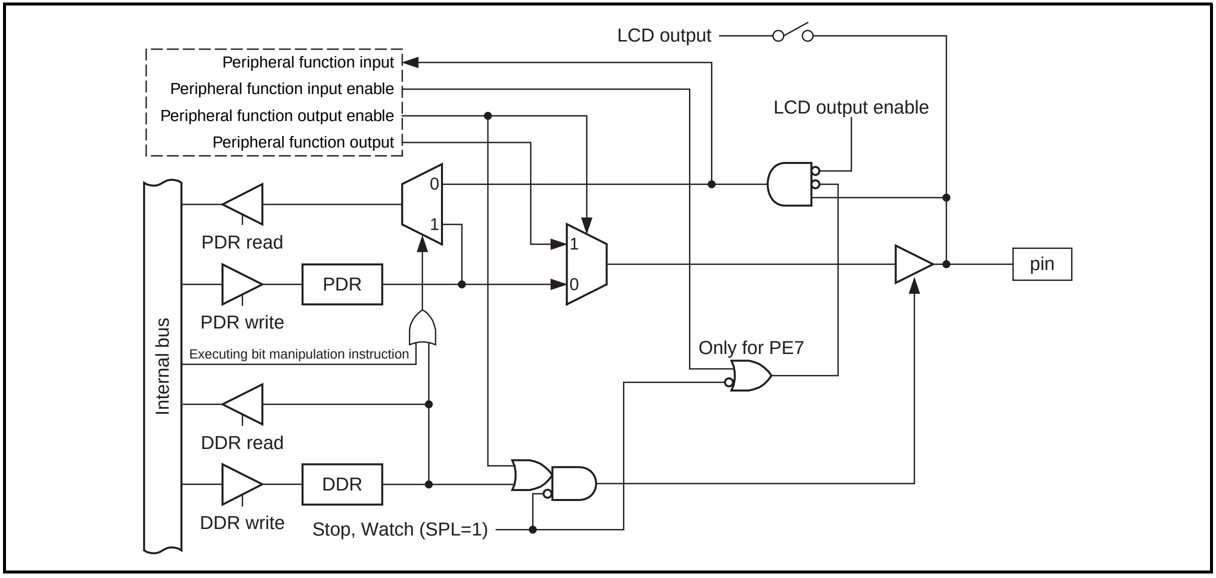


Figure 10.10-2 Block Diagram of PE5, PE6 and PE7


LCD output
Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
LCD output enable
0
1
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
1
0
Only for PE7
pin

10.10.1 Port E Registers

This section describes the registers of port E.

■ Port E Register Functions

Table 10.10-2 lists the port E register functions.

Table 10.10-2 Port E Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRE	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRE	0	Port input enabled		
	1	Port output enabled		

Table 10.10-3 lists the correspondence between port E pins and each register bit.

Table 10.10-3 Correspondence between Registers and Pins for Port E

	Correspondence between related register bits and pins							
Pin name	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
PDRE	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DDRE								

10.10.2 Operations of Port E

This section describes the operations of port E.

■ Operations of Port E

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.
- To use a pin shared with the LCD controller as an output port, set a corresponding segment select bit in the LCDC enable register 4 (LCDCE4:SEG15, SEG14) or in the LCDC enable register 5 (LCDCE5:SEG21 to SEG16) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.
- To use a pin shared with the LCD controller as an input port, set a corresponding segment select bit in the LCDC enable register 4 (LCDCE4:SEG15, SEG14) or in the LCDC enable register 5 (LCDCE5:SEG21 to SEG16) to "0" to select the general-purpose I/O port function, and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● Operation as a peripheral function output pin

- A pin will become a peripheral function output pin if the peripheral output function is enabled by setting the output enable bit of a peripheral function corresponding to that pin.
- The pin value can be read from the PDR register even if the peripheral function output is enabled. Therefore, the output value of a peripheral function can be read by the read operation on the PDR register. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as a peripheral function input pin

- To set a pin as an input port, set the bit in the DDR register bit corresponding to the input pin of a peripheral function to "0".

- Reading the PDR register returns the pin value, regardless of whether the peripheral function uses that pin as its input pin. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation as an LCDC segment output

- Set the DDR register bit corresponding to a desired LCDC segment output pin to "0".
- Select the segment output by setting a corresponding segment select bit in the LCDC enable register 4 (LCDCE4:SEG15, SEG14) or in the LCDC enable register 5 (LCDCE5:SEG21 to SEG16) to "1", and then set the port input control bit (PICTL) in LCDC enable register 1 (LCDCE1) to "1".

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 10.10-4 shows the pin states of port E.

Table 10.10-4 Pin State of Port E

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

10.11 Port F

**Port F is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.**

■ Port F Configuration

Port F is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port F data register (PDRF)
- Port F direction register (DDRF)

■ Port F Pins

Port F has three I/O pins.

Table 10.11-1 lists the port F pins.

Table 10.11-1 Port F Pins

Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PF0/X0[*1]	PF0: General-purpose I/O	X0: Main clock oscillation pin	Hysteresis	CMOS	-	-
PF1/X1[*1]	PF1: General-purpose I/O	X1: Main clock oscillation pin	Hysteresis	CMOS	-	-
PF2/$\overline{\text{RST}}$[*2]	PF2: General-purpose I/O	$\overline{\text{RST}}$: Reset pin	Hysteresis	CMOS	❍	-

OD: N-ch open drain, PU: Pull-up

*1: If the main oscillation clock is selected (SYSC:PFSEL = 0), the port function cannot be used.
*2: If the external reset is selected (SYSC:RSTEN = 1), the port function cannot be used. This pin is a dedicated reset pin in MB95F474H/F476H/F478H.

■ Block Diagrams of Port F

Figure 10.11-1 Block Diagram of PF0 and PF1

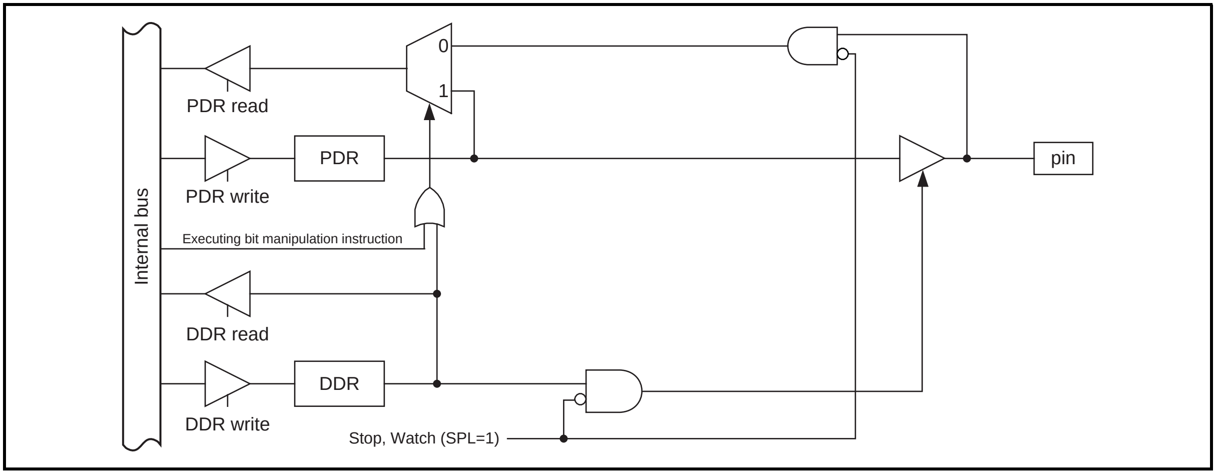
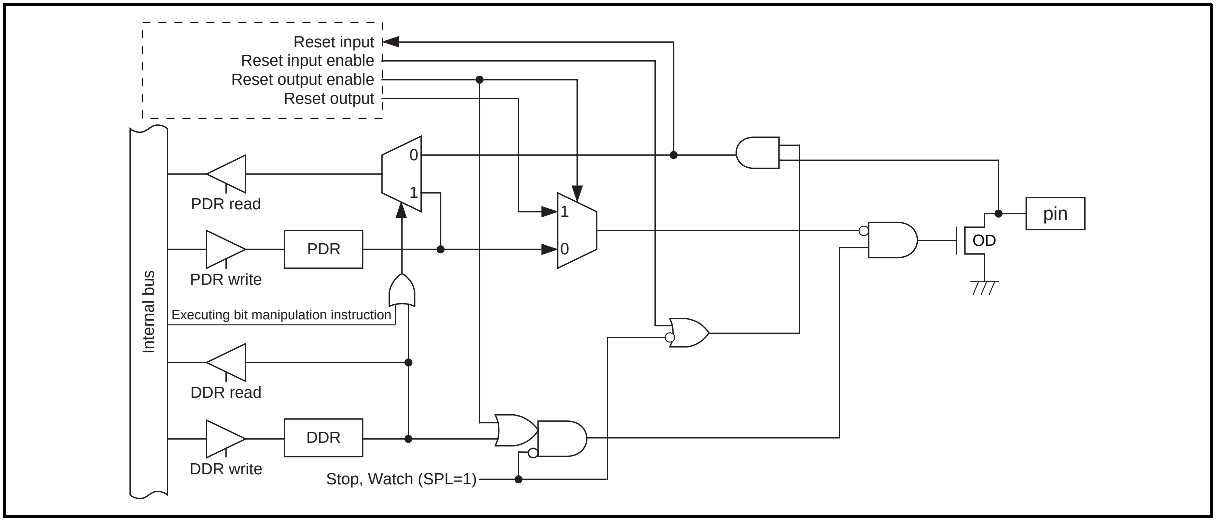


Figure 10.11-2 Block Diagram of PF2



10.11.1 Port F Registers

This section describes the registers of port F.

■ Port F Register Functions

Table 10.11-2 lists the port F register functions.

Table 10.11-2 Port F Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRF	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.*
DDRF	0	Port input enabled		
	1	Port output enabled		

*: For the N-ch open drain pin, this should be Hi-Z.

Table 10.11-3 lists the correspondence between port F pins and each register bit.

Table 10.11-3 Correspondence between Registers and Pins for Port F

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	-	PF2*	PF1	PF0
PDRF	-	-	-	-	-	bit2	bit1	bit0
DDRF								

*: PF2/$\overline{\text{RST}}$ is a dedicated reset pin in MB95F474H/F476H/F478H.

10.11.2 Operations of Port F

This section describes the operations of port F.

■ Operations of Port F

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

Table 10.11-4 shows the pin states of port F.

Table 10.11-4 Pin State of Port F

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled*1 (Not functional) Low*2

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*1: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

*2: Only for PF2 at power-on reset.

10.12 Port G

Port G is a general-purpose I/O port.
This section focuses on its functions as a general-purpose I/O port.
For details of peripheral functions, see their respective chapters.

■ Port G Configuration

Port G is made up of the following elements.

- General-purpose I/O pins/peripheral function I/O pins
- Port G data register (PDRG)
- Port G direction register (DDRG)
- Port G pull-up register (PULG)

■ Port G Pin

Port G has two I/O pin.

Table 10.12-1 lists the port G pins.

Table 10.12-1 Port G Pins

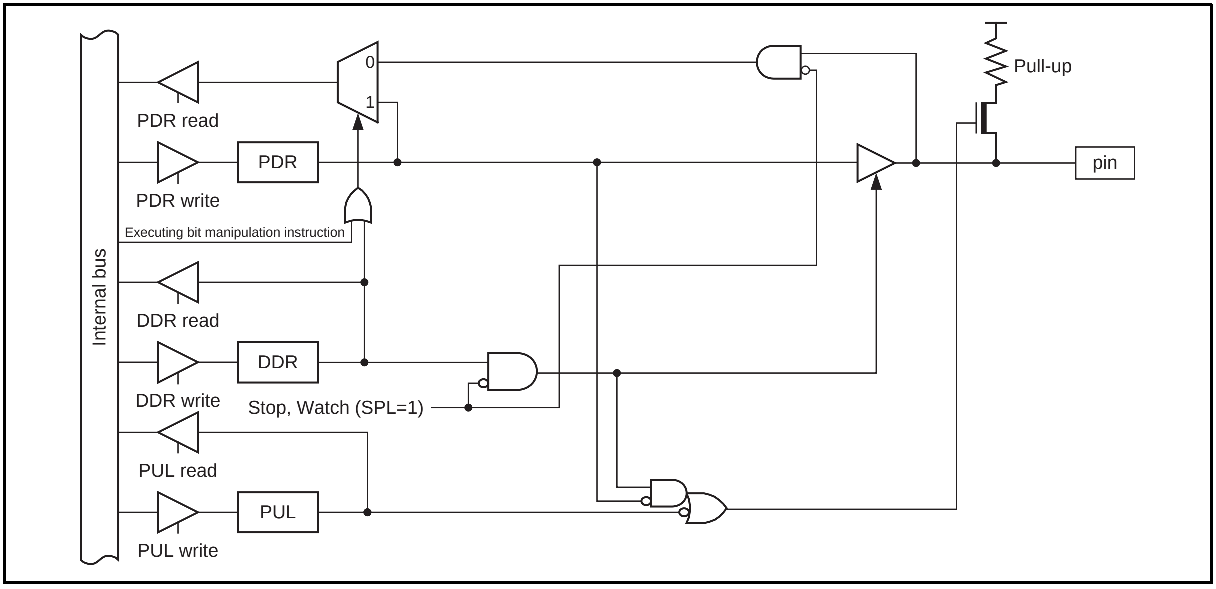
Pin name	Function	Shared peripheral function	I/O type			
			Input	Output	OD	PU
PG1/X0A*	PG1: General-purpose I/O	X0A: Subclock oscillation pin	Hysteresis	CMOS	-	❍
PG2/X1A*	PG2: General-purpose I/O	X1A: Subclock oscillation pin	Hysteresis	CMOS	-	❍

OD: N-ch open drain, PU: Pull-up

*: If the sub-oscillation clock is selected (SYSC:PGSEL = 0), the port function cannot be used.

■ Block Diagram of Port G

Figure 10.12-1 Block Diagram of PG1 and PG2



10.12.1 Port G Registers

This section describes the registers of port G.

■ Port G Register Functions

Table 10.12-2 lists the port G register functions.

Table 10.12-2 Port G Register Functions

Register abbr.	Data	Read	Read by read-modify-write instruction	Write
PDRG	0	Pin state is "L" level.	PDR value is "0".	As output port, outputs "L" level.
	1	Pin state is "H" level.	PDR value is "1".	As output port, outputs "H" level.
DDRG	0	Port input enabled		
	1	Port output enabled		
PULG	0	Pull-up disabled		
	1	Pull-up enabled		

Table 10.12-3 lists the correspondence between port G pins and each register bit.

Table 10.12-3 Correspondence between Registers and Pins for Port G

	Correspondence between related register bits and pins							
Pin name	-	-	-	-	-	PG2	PG1	-
PDRG	-	-	-	-	-	bit2	bit1	-
DDRG								
PULG								

10.12.2 Operations of Port G

This section describes the operations of port G.

■ Operations of Port G

● Operation as an output port

- A pin will become an output port if the bit in the DDR register corresponding to that pin is set to "1".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- When a pin is used as an output port, it outputs the value of the PDR register to external pins.
- If data is written to the PDR register, the value is stored in the output latch and is output to the pin set as an output port as it is.
- Reading the PDR register returns the PDR value.

● Operation as an input port

- A pin will become an input port if the bit in the DDR register corresponding to that pin is set to "0".
- For a pin shared with other peripheral functions, disable the output of such peripheral functions.
- If data is written to the PDR register, the value is stored in the output latch but is not output to the pin set as an input port.
- Reading the PDR register returns the pin value. However, if the read-modify-write instruction is used to read the PDR register, the PDR register value is returned.

● Operation at reset

If the CPU is reset, all bits in the DDR register are initialized to "0" and port input is enabled.

● Operation in stop mode and watch mode

- If the pin state setting bit in the standby control register (STBC:SPL) is set to "1" when the device transits to stop mode or watch mode, the pin is compulsorily made to enter the high impedance state regardless of the DDR register value. The input of that pin is locked to "L" level and blocked in order to prevent leaks due to input open.
- If the pin state setting bit is "0", the state of the port I/O or that of the peripheral function I/O remains unchanged and the output level is maintained.

● Operation of the pull-up register

Setting the bit in the PUL register to "1" makes the pull-up resistor be internally connected to the pin. When the pin output is "L" level, the pull-up resistor is disconnected regardless of the value of the PUL register.

Table 10.12-4 shows the pin states of port G.

Table 10.12-4 Pin State of Port G

Operating state	Normal operation Sleep Stop (SPL=0) Watch (SPL=0)	Stop (SPL=1) Watch (SPL=1)	At reset
Pin state	I/O port	Hi-Z Input cutoff	Hi-Z Input enabled* (Not functional)

SPL: Pin state setting bit in standby control register (STBC:SPL)

Hi-Z: High impedance

*: "Input enabled" means that the input function is enabled. After a reset, setting the port for internal pull-up or as an output pin is recommended.

CHAPTER 11

TIME-BASE TIMER

This chapter describes the functions and operations of the time-base timer.

11.1 Overview of Time-base Timer

The time-base timer is a 24-bit free-run down-counting counter. It is synchronized with the main clock divided by 2, or with the main PLL clock, or with the main CR clock. The clock can be selected by the RCS[1:0] bits in the SYCC2 register and the PCS1 bit and PCS0 bit in the PLLC register. The time-base timer has an interval timer function that can repeatedly generate interrupt requests at regular intervals.

■ Interval Timer Function

The interval timer function repeatedly generates interrupt requests at regular intervals by using the main clock divided by 2, or the main PLL clock, or the main CR clock as the count clock.

- The counter of the time-base timer counts down so that an interrupt request is generated whenever a selected interval time elapses.
- The length of an interval time can be selected from the following 16 values.

Table 11.1-1 shows the interval times available for the time-base timer.

Table 11.1-1 Interval Times of Time-base Timer

	Interval time if the main CR clock is used ($2^n \times 1/F_{CRH}$*1)	Interval time if the main clock is used ($2^n \times 2/F_{CH}$*2, *3)
n=9	64 µs	256 µs
n=10	128 µs	512 µs
n=11	256 µs	1.024 ms
n=12	512 ms	2.048 ms
n=13	1.024 ms	4.096 ms
n=14	2.048 ms	8.192 ms
n=15	4.096 ms	16.384 ms
n=16	8.192 ms	32.768 ms
n=17	16.384 ms	65.536 ms
n=18	32.768 ms	131.072 ms
n=19	65.536 ms	262.144 ms
n=20	131.072 ms	524.288 ms
n=21	262.144 ms	1.049 s
n=22	524.288 ms	2.097 s
n=23	1.049 s	4.194 s
n=24	2.097 s	8.389 s

*1: $1/F_{CRH}$ = 0.125 µs when F_{CRH} = 8 MHz

*2: $2/F_{CH}$ = 0.5 µs when F_{CH} = 4 MHz

*3: When PLLC:PCS[1:0] = 00, the main clock divided by 2 ($F_{CH}/2$) is used as the count clock; when PLLC:PCS[1:0] = 01, 10 or 11, the main PLL clock is used as the count clock.

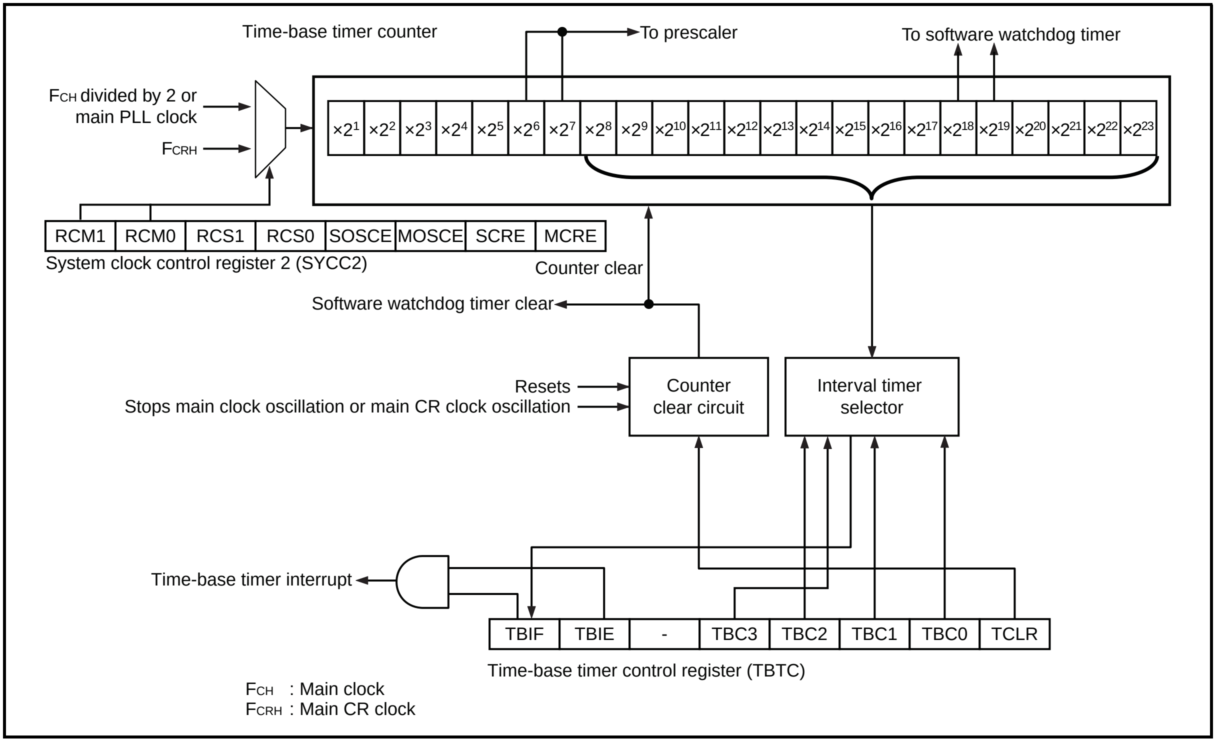
11.2 Configuration of Time-base Timer

The time-base timer consists of the following blocks:

- **Time-base timer counter**
- **Counter clear circuit**
- **Interval timer selector**
- **Time-base timer control register (TBTC)**

■ Block Diagram of Time-base Timer

Figure 11.2-1 Block Diagram of Time-base Timer


Time-base timer counter
To prescaler
To software watchdog timer
FCH divided by 2 or main PLL clock
FCRH
×2¹ ×2² ×2³ ×2⁴ ×2⁵ ×2⁶ ×2⁷ ×2⁸ ×2⁹ ×2¹⁰ ×2¹¹ ×2¹² ×2¹³ ×2¹⁴ ×2¹⁵ ×2¹⁶ ×2¹⁷ ×2¹⁸ ×2¹⁹ ×2²⁰ ×2²¹ ×2²² ×2²³
RCM1 RCM0 RCS1 RCS0 SOSCE MOSCE SCRE MCRE
System clock control register 2 (SYCC2)
Counter clear
Software watchdog timer clear
Resets
Stops main clock oscillation or main CR clock oscillation
Counter clear circuit
Interval timer selector
Time-base timer interrupt
TBIF TBIE - TBC3 TBC2 TBC1 TBC0 TCLR
Time-base timer control register (TBTC)
FCH : Main clock
FCRH : Main CR clock

● Time-base timer counter

This is a 24-bit down-counter using the main clock divided by 2 or the main PLL clock or the main CR clock as its count clock.

● Counter clear circuit

This circuit controls the clearing of the time-base timer counter.

● Interval timer selector

This circuit selects one bit out of 16 bits in the 24 bits of the time-base timer counter as the interval timer.

● Time-base timer control register (TBTC)

This register selects the interval time, clears the counter, controls interrupts and checks the status of the time-base timer.

■ Input Clock

The time-base timer uses the main clock divided by two or the main CR clock as its input clock (count clock).

■ Output Clock

The time-base timer supplies clocks to the main clock, the software watchdog timer and the prescaler.

11.3 Register of Time-base Timer

Figure 11.3-1 shows the register of the time-base timer.

■ Register of Time-base Timer

Figure 11.3-1 Register of Time-base Timer

Time-base timer control (TBTC)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$000A_H$	TBIF	TBIE	-	TBC3	TBC2	TBC1	TBC0	TCLR	00000000_B
	R(RM1),W	R/W	R0/WX	R/W	R/W	R/W	R/W	R0,W	

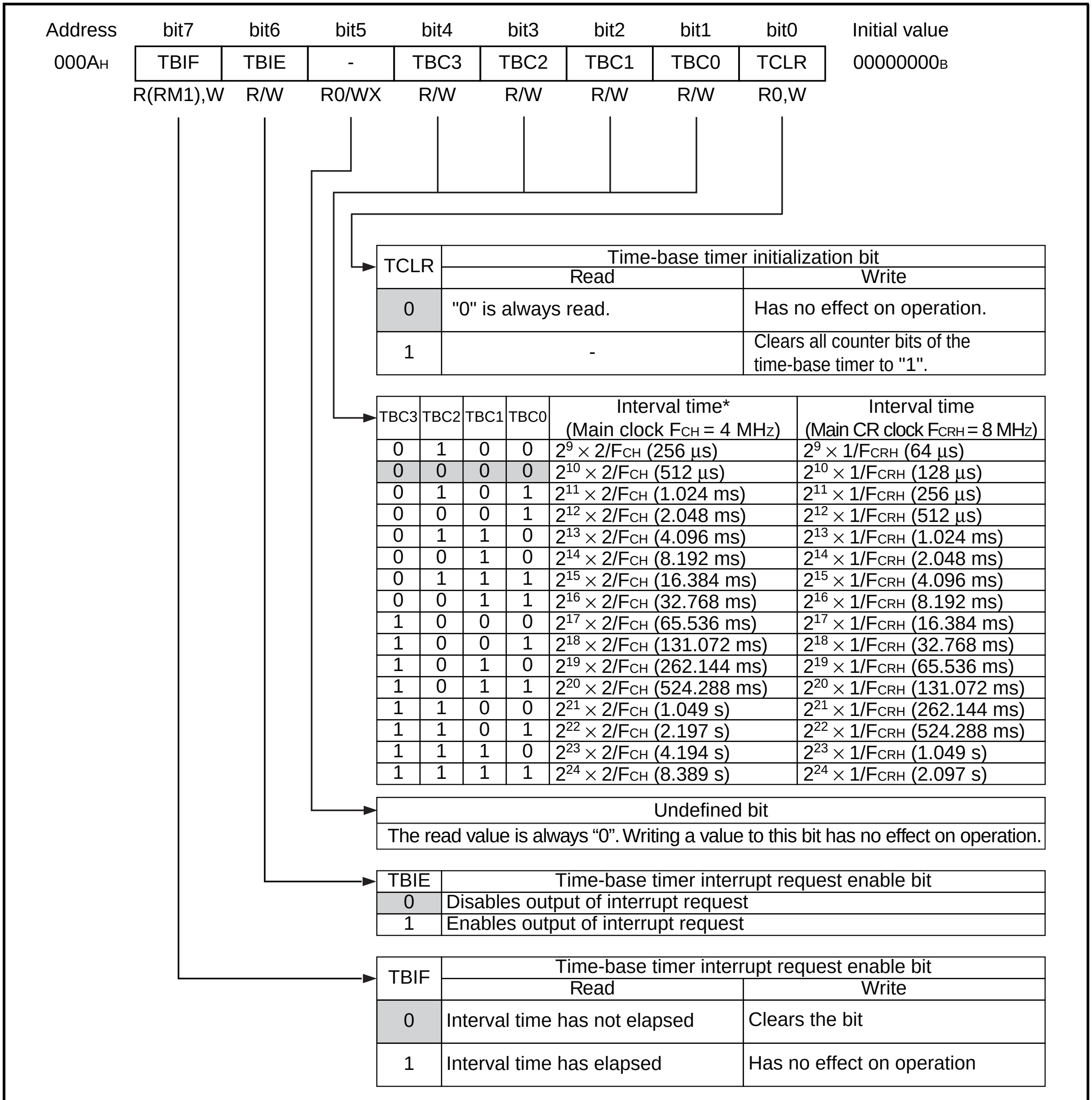
R/W	: Readable/writable (The read value is the same as the write value.)
R(RM1),W	: Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0,W	: Write only (Writable. The read value is "0".
R0/WX	: The read value is "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

11.3.1 Time-base Timer Control Register (TBTC)

The time-base timer control register (TBTC) selects the interval time, clears the counter, controls interrupts and checks the status of the time-base timer.

■ Time-base Timer Control Register (TBTC)

Figure 11.3-2 Time-base Timer Control Register (TBTC)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
000A$_H$	TBIF	TBIE	-	TBC3	TBC2	TBC1	TBC0	TCLR	00000000$_B$
	R(RM1),W	R/W	R0/WX	R/W	R/W	R/W	R/W	R0,W	

TCLR	Time-base timer initialization bit	
	Read	Write
0	"0" is always read.	Has no effect on operation.
1	-	Clears all counter bits of the time-base timer to "1".

TBC3	TBC2	TBC1	TBC0	Interval time* (Main clock F_{CH} = 4 MHz)	Interval time (Main CR clock F_{CRH} = 8 MHz)
0	1	0	0	$2^{9} \times 2/F_{CH}$ (256 µs)	$2^{9} \times 1/F_{CRH}$ (64 µs)
0	0	0	0	$2^{10} \times 2/F_{CH}$ (512 µs)	$2^{10} \times 1/F_{CRH}$ (128 µs)
0	1	0	1	$2^{11} \times 2/F_{CH}$ (1.024 ms)	$2^{11} \times 1/F_{CRH}$ (256 µs)
0	0	0	1	$2^{12} \times 2/F_{CH}$ (2.048 ms)	$2^{12} \times 1/F_{CRH}$ (512 µs)
0	1	1	0	$2^{13} \times 2/F_{CH}$ (4.096 ms)	$2^{13} \times 1/F_{CRH}$ (1.024 ms)
0	0	1	0	$2^{14} \times 2/F_{CH}$ (8.192 ms)	$2^{14} \times 1/F_{CRH}$ (2.048 ms)
0	1	1	1	$2^{15} \times 2/F_{CH}$ (16.384 ms)	$2^{15} \times 1/F_{CRH}$ (4.096 ms)
0	0	1	1	$2^{16} \times 2/F_{CH}$ (32.768 ms)	$2^{16} \times 1/F_{CRH}$ (8.192 ms)
1	0	0	0	$2^{17} \times 2/F_{CH}$ (65.536 ms)	$2^{17} \times 1/F_{CRH}$ (16.384 ms)
1	0	0	1	$2^{18} \times 2/F_{CH}$ (131.072 ms)	$2^{18} \times 1/F_{CRH}$ (32.768 ms)
1	0	1	0	$2^{19} \times 2/F_{CH}$ (262.144 ms)	$2^{19} \times 1/F_{CRH}$ (65.536 ms)
1	0	1	1	$2^{20} \times 2/F_{CH}$ (524.288 ms)	$2^{20} \times 1/F_{CRH}$ (131.072 ms)
1	1	0	0	$2^{21} \times 2/F_{CH}$ (1.049 s)	$2^{21} \times 1/F_{CRH}$ (262.144 ms)
1	1	0	1	$2^{22} \times 2/F_{CH}$ (2.197 s)	$2^{22} \times 1/F_{CRH}$ (524.288 ms)
1	1	1	0	$2^{23} \times 2/F_{CH}$ (4.194 s)	$2^{23} \times 1/F_{CRH}$ (1.049 s)
1	1	1	1	$2^{24} \times 2/F_{CH}$ (8.389 s)	$2^{24} \times 1/F_{CRH}$ (2.097 s)

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

TBIE	Time-base timer interrupt request enable bit
0	Disables output of interrupt request
1	Enables output of interrupt request

TBIF	Time-base timer interrupt request enable bit	
	Read	Write
0	Interval time has not elapsed	Clears the bit
1	Interval time has elapsed	Has no effect on operation

R/W	:	Readable/writable (The read value is the same as the write value.)
R(RM1),W	:	Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R0,W	:	Write only (Writable. The read value is “0”.)
R0/WX	:	The read value is “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

*: When the PLLC:PCS[1:0] bits are set to “00”, the main clock divided by 2 (F_{CH}/2) is used as the count clock.
When the PLLC:PCS[1:0] bits are set to “01”, “10” or “11”, the main PLL clock is used as the count clock.

Table 11.3-1 Functions of Bits in Time-base Timer Control Register (TBTC)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7</td><td>TBIF:
Time-base timer interrupt request flag bit</td><td>This flag is set to "1" when the interval time selected by the time-base timer elapses.
An interrupt request is output if this bit and the time-base timer interrupt request enable bit (TBIE) are set to "1".
<b>Writing "0"</b>: Clears this bit to "0".
<b>Writing "1"</b>: Has no effect on operation.
If this bit is read by the read-modify-write (RMW) type of instruction, it always returns "1".</td></tr>
<tr><td>bit6</td><td>TBIE:
Time-base timer interrupt request enable bit</td><td>This bit enables or disables output of interrupt requests to interrupt controller.
<b>Writing "0"</b>: Disables the output of time-base timer interrupt requests.
<b>Writing "1"</b>: Enables the output of time-base timer interrupt requests.
An interrupt request is output if this bit and the time-base timer interrupt request flag bit (TBIF) are set to "1".</td></tr>
<tr><td>bit5</td><td>Undefined bit</td><td>The read value is always "0". Writing a value to this bit has no effect on operation.</td></tr>
<tr><td>bit4
to
bit1</td><td>TBC3 to TBC0:
Interval time select bits</td><td>These bits select interval time.
<table>
<tr><th>TBC3</th><th>TBC2</th><th>TBC1</th><th>TBC0</th><th>Interval time*
(Main clock F_{CH} = 4 MHz)</th><th>Interval time
(Main CR clock F_{CRH} = 8 MHz)</th></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>$2^{9} \times 2/F_{CH}$ (256 µs)</td><td>$2^{9} \times 1/F_{CRH}$ (64 µs)</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>$2^{10} \times 2/F_{CH}$ (512 µs)</td><td>$2^{10} \times 1/F_{CRH}$ (128 µs)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>$2^{11} \times 2/F_{CH}$ (1.024 ms)</td><td>$2^{11} \times 1/F_{CRH}$ (256 µs)</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>$2^{12} \times 2/F_{CH}$ (2.048 ms)</td><td>$2^{12} \times 1/F_{CRH}$ (512 µs)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>$2^{13} \times 2/F_{CH}$ (4.096 ms)</td><td>$2^{13} \times 1/F_{CRH}$ (1.024 ms)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>$2^{14} \times 2/F_{CH}$ (8.192 ms)</td><td>$2^{14} \times 1/F_{CRH}$ (2.048 ms)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>$2^{15} \times 2/F_{CH}$ (16.384 ms)</td><td>$2^{15} \times 1/F_{CRH}$ (4.096 ms)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>$2^{16} \times 2/F_{CH}$ (32.768 ms)</td><td>$2^{16} \times 1/F_{CRH}$ (8.192 ms)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>$2^{17} \times 2/F_{CH}$ (65.536 ms)</td><td>$2^{17} \times 1/F_{CRH}$ (16.384 ms)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>$2^{18} \times 2/F_{CH}$ (131.072 ms)</td><td>$2^{18} \times 1/F_{CRH}$ (32.768 ms)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>$2^{19} \times 2/F_{CH}$ (262.144 ms)</td><td>$2^{19} \times 1/F_{CRH}$ (65.536 ms)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>$2^{20} \times 2/F_{CH}$ (524.288 ms)</td><td>$2^{20} \times 1/F_{CRH}$ (131.072 ms)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>$2^{21} \times 2/F_{CH}$ (1.049 s)</td><td>$2^{21} \times 1/F_{CRH}$ (262.144 ms)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>$2^{22} \times 2/F_{CH}$ (2.097 s)</td><td>$2^{22} \times 1/F_{CRH}$ (524.288 ms)</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>$2^{23} \times 2/F_{CH}$ (4.194 s)</td><td>$2^{23} \times 1/F_{CRH}$ (1.049 s)</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>$2^{24} \times 2/F_{CH}$ (8.389 s)</td><td>$2^{24} \times 1/F_{CRH}$ (2.097 s)</td></tr>
</table>
*: When the PLLC:PCS[1:0] bits are set to "00", the main clock divided by 2 (F_{CH}/2) is used as the count clock.
When the PLLC:PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used as the count clock.</td></tr>
<tr><td>bit0</td><td>TCLR:
Time-base timer initialization bit</td><td>This bit clears all counter bits of the time-base timer to "1".
<b>Writing "0"</b>: Has no effect on the operation.
<b>Writing "1"</b>: Initializes all counter bits to "1".
When this bit is read, it always returns "0".
Note: When the output of the time-base timer is selected as the count clock for the watchdog timer, using this bit to clear the time-base timer also clears the software watchdog timer.</td></tr>
</table>

11.4 Interrupts of Time-base Timer

An interrupt request is generated when the interval time selected by the time-base timer elapses (interval timer function).

■ Interrupts when Interval Function is in Operation

When the time-base timer counter counts down by using the internal count clock and the selected time-base timer counter underflows, the time-base timer interrupt request flag bit (TBTC:TBIF) is set to "1". With the TBIF bit set to "1", if the time-base timer interrupt request enable bit is also enabled (TBTC:TBIE = 1), an interrupt request (IRQ19) will be generated to the interrupt controller.

- Regardless of the value of the TBIE bit, the TBIF bit is set to "1" when the selected bit underflows.
- With the TBIF bit set to "1", if the TBIE bit is changed from the disable state to the enable state (0 → 1), an interrupt request is generated immediately.
- The TBIF bit will not be set to "1" if the clearing of a counter (TBTC:TCLR = 1) and the underflow of the time-base timer counter occur simultaneously.
- In the interrupt service routine, write "0" to the TBIF bit to clear an interrupt request.

Note:

When enabling the output of interrupt requests after canceling a reset (TBTC:TBIE = 1), always clear the TBIF bit at the same time (TBTC:TBIF = 0).

Table 11.4-1 Interrupts of Time-base Timer

Item	Description
Interrupt condition	The interval time set by "TBTC:TBC3-TBC0" has elapsed.
Interrupt flag	TBTC:TBIF
Interrupt enable	TBTC:TBIE

■ Register and Vector Table Addresses for Interrupts of Time-base Timer

Table 11.4-2 Register and Vector Table Addresses for Interrupts of Time-base Timer

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
Time-base timer	IRQ19	ILR4	L19	$FFD4_H$	$FFD5_H$

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

11.5 Operations of Time-base Timer and Setting Procedure Example

This section describes the operations of the interval timer function of the time-base timer.

■ Operations of Time-base Timer

The counter of the time-base timer is initialized to "FFFFFF$_H$" after a reset and starts counting while being synchronized with the main clock divided by two.

The time-base timer continues to count down as long as the main clock is oscillating. Once the main clock halts, the counter stops counting and is initialized to "FFFFFF$_H$".

The settings shown in Figure 11.5-1 are required to use the interval timer function.

Figure 11.5-1 Settings of Interval Timer Function

Address		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
000A$_H$	TBTC	TBIF	TBIE	-	TBC3	TBC2	TBC1	TBC0	TCLR
		0	1		◎	◎	◎	◎	0

◎: Bit to be used
1 : Set to "1"
0 : Set to "0"

When the time-base timer initialization bit in the time-base timer control register (TBTC:TCLR) is set to "1", the counter of the time-base timer is initialized to "FFFFFF$_H$" and continues to count down. When the selected interval time has elapsed, the time-base timer interrupt request flag bit of the time-base timer control register (TBTC:TBIF) becomes "1". In other words, an interrupt request is generated at each interval time selected, based on the time when the counter was last cleared.

■ Clearing Time-base Timer

If the time-base timer is cleared when the output of the time-base timer is used in other peripheral functions, this will affect the operation by changing the count time or in other manners.

When clearing the counter by using the time-base timer initialization bit (TBTC:TCLR), modify the settings of other peripheral functions whenever necessary so that clearing the counter does not have any unexpected effect on them.

When the output of the time-base timer is selected as the count clock for the watchdog timer, clearing the time-base timer also clears the watchdog timer.

The time-base timer is cleared not only by the time-base timer initialization bit (TBTC:TCLR), but also when the main clock is stopped and the oscillation stabilization wait time is necessary. The time-base timer is cleared in the following situations:

- When the device transits from the main clock mode or main CR clock mode to the stop mode
- When the device transits from the main clock mode or main CR clock mode to the subclock mode or sub-CR clock mode
- At power on
- At low-voltage detection reset

■ Operation Examples of Time-base Timer

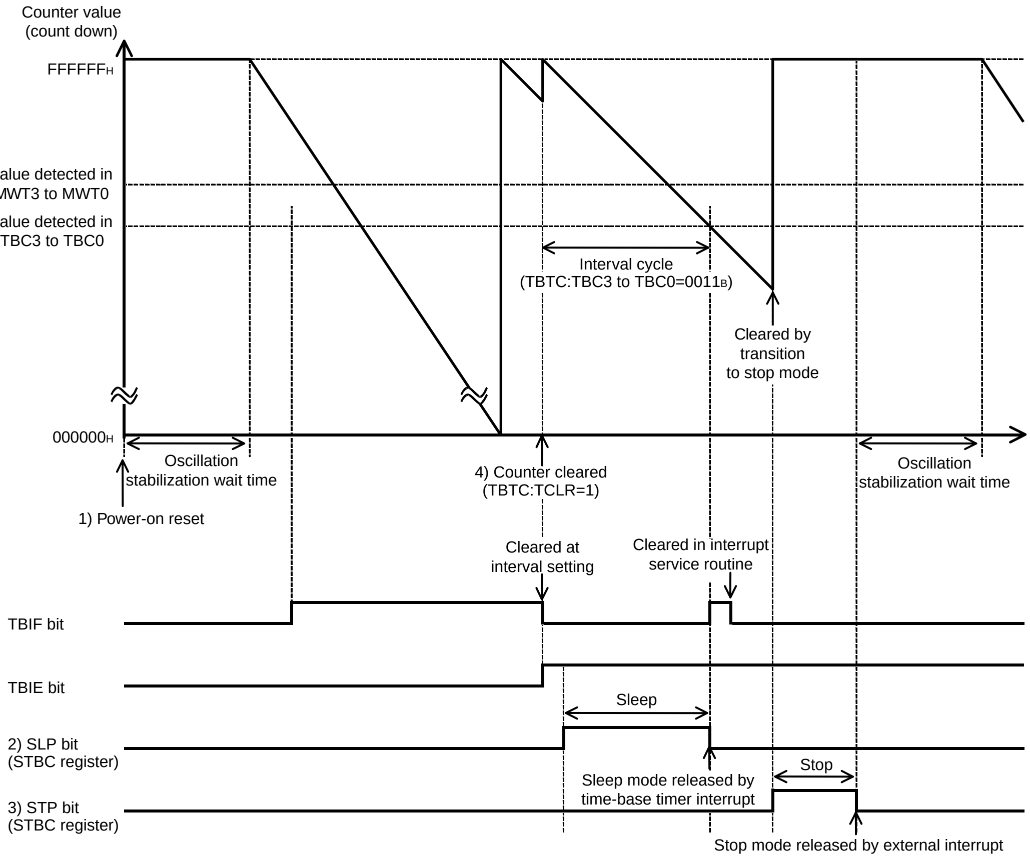
Figure 11.5-2 shows examples of operations under the following conditions:

1) When a power-on reset is generated
2) When the device enters the sleep mode during the operation of the interval timer function in the main clock mode or main CR clock mode
3) When the device enters the stop mode during the main clock mode or main CR clock mode
4) When a request is generated to clear the counter

If the device transits to the time-base time mode, the same operations are executed as those executed when the device transits to the sleep mode.

In stop mode in which the clock mode is subclock mode, sub-CR clock mode, main clock mode or main CR clock mode, the timer operation stops because it is cleared and the main clock stops.

Figure 11.5-2 Operations of Time-base Timer


Counter value
(count down)
FFFFFFH
Count value detected in WATR:MWT3 to MWT0
Count value detected in TBTC:TBC3 to TBC0
Interval cycle
(TBTC:TBC3 to TBC0=0011B)
Cleared by transition to stop mode
000000H
Oscillation stabilization wait time
1) Power-on reset
4) Counter cleared
(TBTC:TCLR=1)
Oscillation stabilization wait time
Cleared at interval setting
Cleared in interrupt service routine
TBIF bit
TBIE bit
Sleep
2) SLP bit
(STBC register)
Stop
Sleep mode released by time-base timer interrupt
3) STP bit
(STBC register)
Stop mode released by external interrupt

- When setting the interval time select bits in time-base timer control register (TBTC:TBC3 to TBC0) to "0011_B" (2^{16} × $2/F_{CH}$)

• TBTC:TBC3 to TBC0	: Interval time select bits in time-base timer control register
• TBTC:TCLR	: Time-base timer initialization bit in time-base timer control register
• TBTC:TBIF	: Time-base timer interrupt request flag bit in time-base timer control register
• TBTC:TBIE	: Time-base timer interrupt request enable bit in time-base timer control register
• STBC:SLP	: Sleep bit in standby control register
• STBC:STP	: Stop bit in standby control register
• WATR:MWT3 to MWT0	: Main clcok oscillation stabilization wait time select bits in oscillation stabilization wait time setting register

■ Setting Procedure Example

Below is an example of procedure for setting the time-base timer.

● Initial settings

1) Disable interrupts. (TBTC:TBIE = 0)
2) Set the interval time. (TBTC:TBC3 to TBC0)
3) Enable interrupts. (TBTC:TBIE = 1)
4) Clear the counter. (TBTC:TCLR = 1)

● Processing interrupts

1) Clear the interrupt request flag. (TBTC:TBIF = 0)
2) Clear the counter. (TBTC:TCLR = 1)

11.6 Notes on Using Time-base Timer

This section provides notes on using the time-base timer.

■ Notes on Using Time-base Timer

● When setting the timer by program

The timer cannot be waken up from interrupt processing when the time-base timer interrupt request flag bit (TBTC:TBIF) is set to "1" and the interrupt request enable bit is enabled (TBTC:TBIE = 1). Always clear the TBIF bit in the interrupt service routine.

● Clearing Time-base Timer

The time-base timer is cleared not only by the time-base timer initialization bit (TBTC:TCLR = 1) but also when the oscillation stabilization wait time of the main clock is required. When the time-base timer is selected as the count clock of the software watchdog timer (WDTC:CS1, CS0 = 00_B or 01_B), clearing the time-base timer also clears the software watchdog timer.

● Peripheral functions receiving clock from time-base timer

In the mode where the source oscillation of the main clock is stopped, the counter is cleared and the time-base timer stops operating. In addition, if the counter of the time-base timer is cleared with the output of the time-base timer being used in other peripheral functions, that will affect the operations of such peripheral operations such as the changing of their operating cycles.

After the counter of the time-base timer is cleared, the clock that is output from the time-base timer for the software watchdog timer returns to the initial state. However, since the software watchdog timer counter is also cleared at the same time as the clock for the software watchdog timer returns to the initial state, the software watchdog timer operates in its normal cycle.

CHAPTER 12

HARDWARE/SOFTWARE WATCHDOG TIMER

This chapter describes the functions and operations of the watchdog timer.

12.1 Overview of Watchdog Timer

The watchdog timer serves as a counter used to prevent programs from running out of control.

■ Watchdog Timer Function

The watchdog timer functions as a counter used to prevent programs from running out of control. Once the watchdog timer is activated, its counter needs to be cleared at specified intervals regularly. A watchdog reset is generated if the timer is not cleared within a certain amount of time due to a problem such as a program entering an infinite loop.

● Count clock for the software/hardware watchdog timer

- For the software watchdog timer, the output of the time-base timer or of the watch prescaler or of the sub-CR timer can be used as the count clock.
- For the hardware watchdog timer, only the output of the sub-CR timer can be used as the count clock.

● Activation of the software/hardware watchdog timer

- The software/hardware watchdog timer is to be activated according to the values at the addresses $FFBE_H$ and $FFBF_H$ on the Flash memory, which are copied to the watchdog timer selection ID registers WDTH/WDTL ($0FEB_H$/$0FEC_H$).
- In the case of software activation (software watchdog), the watchdog timer register (WDTC) must be set to start the watchdog timer function.
- In the case of hardware activation (hardware watchdog), the watchdog timer starts automatically after a reset. It can also stop or run in stop mode according to the values at the addresses $FFBE_H$ and $FFBF_H$ on the Flash memory, which are copied to the watchdog timer selection ID registers WDTH/WDTL ($0FEB_H$/$0FEC_H$). See "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION" for details of the watchdog timer selection ID.
- The intervals of the watchdog timer are shown in Table 12.1-1. If the counter of the watchdog timer is not cleared, a watchdog reset is generated between the minimum time and the maximum time. Clear the counter of the watchdog timer within the minimum time.

Table 12.1-1 Interval Times of Watchdog Timer

Count clock type	Count clock switch bits CS[1:0], CSP	Interval time	
		Minimum time	Maximum time
Time-base timer output (main clock = 4 MHz)	000_B (SWWDT)	524 ms	1.05 s
	010_B (SWWDT)	262 ms	524 ms
Watch prescaler output (subclock = 32.768 kHz)	100_B (SWWDT)	500 ms	1.00 s
	110_B (SWWDT)	250 ms	500 ms
Sub-CR timer (sub-CR clock = 50 kHz to 200 kHz)	$XX1_B$ (SWWDT) or HWWDT*1	328 ms	2.62 s

*1: CS[1:0] = 00_B, CSP = 1 (read only)

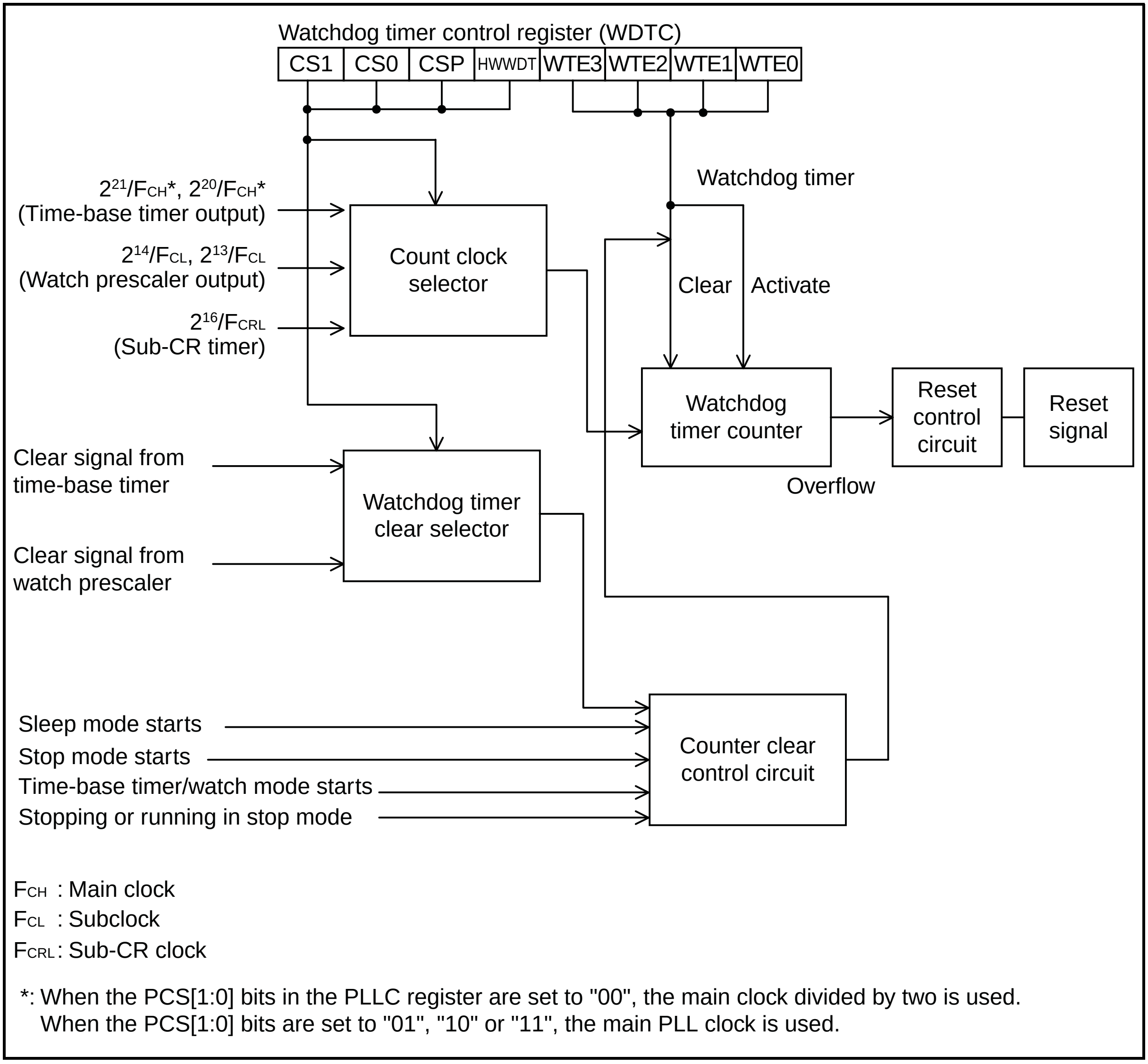
12.2 Configuration of Watchdog Timer

The watchdog timer consists of the following blocks:

- **Count clock selector**
- **Watchdog timer counter**
- **Reset control circuit**
- **Watchdog timer clear selector**
- **Counter clear control circuit**
- **Watchdog timer control register (WDTC)**

■ Block Diagram of Watchdog Timer

Figure 12.2-1 Block Diagram of Watchdog Timer


Watchdog timer control register (WDTC)
CS1
CS0
CSP
HWWDT
WTE3
WTE2
WTE1
WTE0
$2^{21}/F_{CH}$*, $2^{20}/F_{CH}$*
(Time-base timer output)
$2^{14}/F_{CL}$, $2^{13}/F_{CL}$
(Watch prescaler output)
$2^{16}/F_{CRL}$
(Sub-CR timer)
Count clock selector
Watchdog timer
Clear
Activate
Watchdog timer counter
Reset control circuit
Reset signal
Overflow
Clear signal from time-base timer
Clear signal from watch prescaler
Watchdog timer clear selector
Sleep mode starts
Stop mode starts
Time-base timer/watch mode starts
Stopping or running in stop mode
Counter clear control circuit

F_{CH} : Main clock
F_{CL} : Subclock
F_{CRL}: Sub-CR clock

*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used. When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

- Count clock selector

  This selector selects the count clock of the watchdog timer counter.

- Watchdog timer counter

  This is a 1-bit counter that uses the output of the time-base timer or of the watch prescaler or of the sub-CR timer as the count clock.

- Reset control circuit

  This circuit generates a reset signal when the watchdog timer counter overflows.

- Watchdog timer clear selector

  This selector selects the watchdog timer clear signal.

- Counter clear control circuit

  This circuit controls the clearing and stopping of the watchdog timer counter.

- Watchdog timer control register (WDTC)

  This register performs setup for activating/clearing the watchdog timer counter as well as for selecting the count clock.

■ Input Clock

The watchdog timer uses the output clock of the time-base timer or of the watch prescaler or of the sub-CR timer as the input clock (count clock).

12.3 Register of Watchdog Timer

Figure 12.3-1 shows the register of the watchdog timer.

■ Register of Watchdog Timer

Figure 12.3-1 Register of Watchdog Timer

Watchdog timer control register (WDTC)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$000C_H$	CS1	CS0	CSP	HWWDT	WTE3	WTE2	WTE1	WTE0	
Software	R/W	R/W	R/W	R0/WX	R0,W	R0,W	R0,W	R0,W	00000000_B
Hardware	R0/WX	R0/WX	R1/WX	R1/WX	R0,W	R0,W	R0,W	R0,W	00110000_B

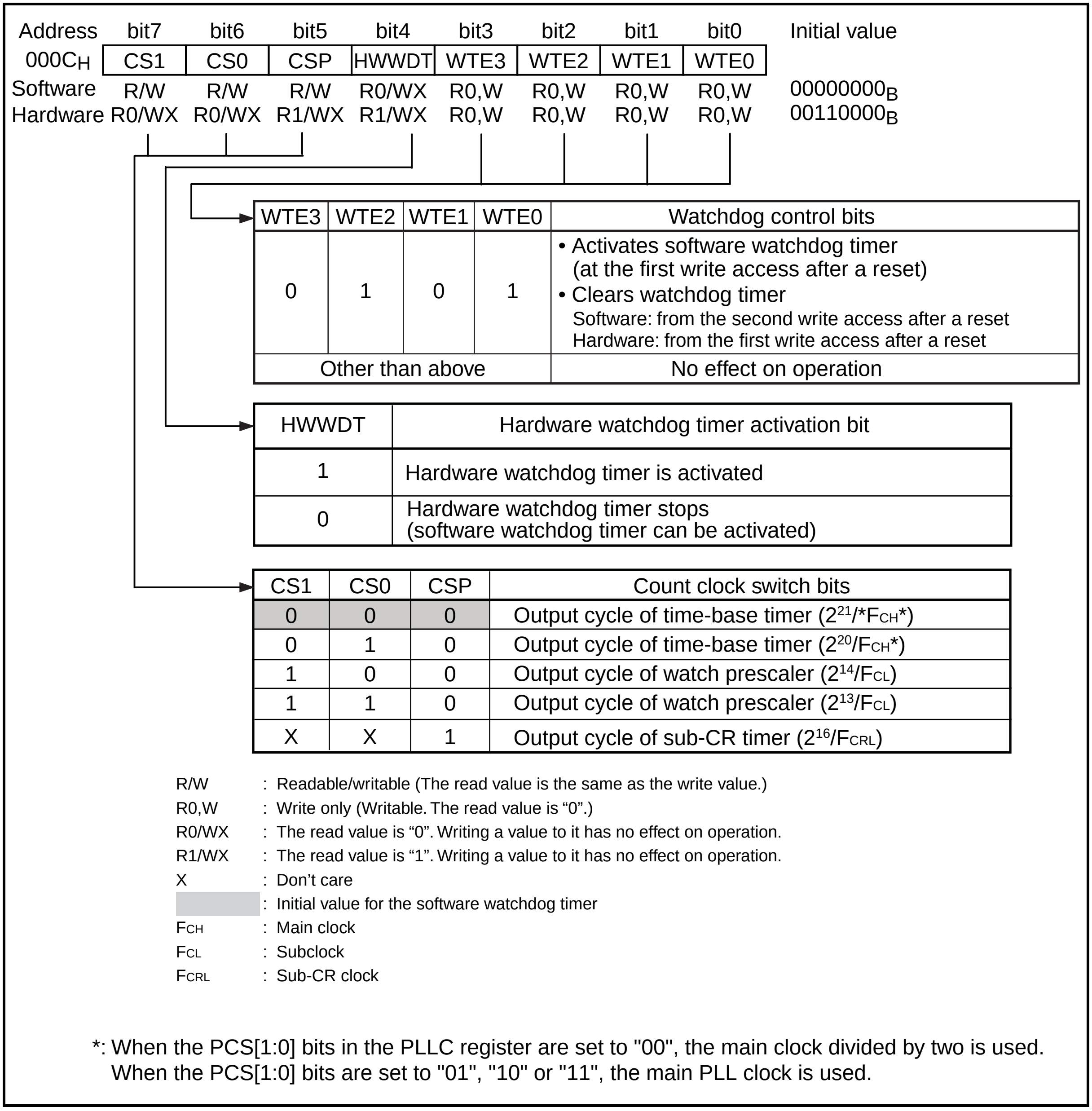
R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is "0". Writing a value to it has no effect on operation.
R1/WX : The read value is "1". Writing a value to it has no effect on operation.
R0,W : Write only (Writable. The read value is "0".)

12.3.1 Watchdog Timer Control Register (WDTC)

The watchdog timer control register (WDTC) activates or clears the watchdog timer.

■ Watchdog Timer Control Register (WDTC)

Figure 12.3-2 Watchdog Timer Control Register (WDTC)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$000C_H$	CS1	CS0	CSP	HWWDT	WTE3	WTE2	WTE1	WTE0	
Software	R/W	R/W	R/W	R0/WX	R0,W	R0,W	R0,W	R0,W	00000000_B
Hardware	R0/WX	R0/WX	R1/WX	R1/WX	R0,W	R0,W	R0,W	R0,W	00110000_B

WTE3	WTE2	WTE1	WTE0	Watchdog control bits
0	1	0	1	• Activates software watchdog timer (at the first write access after a reset) • Clears watchdog timer Software: from the second write access after a reset Hardware: from the first write access after a reset
Other than above				No effect on operation

HWWDT	Hardware watchdog timer activation bit
1	Hardware watchdog timer is activated
0	Hardware watchdog timer stops (software watchdog timer can be activated)

CS1	CS0	CSP	Count clock switch bits
0	0	0	Output cycle of time-base timer (2^{21}/*F_{CH}*)
0	1	0	Output cycle of time-base timer (2^{20}/F_{CH}*)
1	0	0	Output cycle of watch prescaler (2^{14}/F_{CL})
1	1	0	Output cycle of watch prescaler (2^{13}/F_{CL})
X	X	1	Output cycle of sub-CR timer (2^{16}/F_{CRL})

R/W	:	Readable/writable (The read value is the same as the write value.)
R0,W	:	Write only (Writable. The read value is “0”.)
R0/WX	:	The read value is “0”. Writing a value to it has no effect on operation.
R1/WX	:	The read value is “1”. Writing a value to it has no effect on operation.
X	:	Don’t care
▨	:	Initial value for the software watchdog timer
F_{CH}	:	Main clock
F_{CL}	:	Subclock
F_{CRL}	:	Sub-CR clock

*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used. When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

Table 12.3-1 Functions of Bits in Watchdog Timer Control Register (WDTC)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7,
bit6</td><td>CS1, CS0:
Count clock switch bits</td><td rowspan="2">These bits select the count clock of the watchdog timer.
<table>
<tr><th>CS1</th><th>CS0</th><th>CSP</th><th>Count clock switch bits</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Output cycle of time-base timer ($2^{21}/F_{CH}$*)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Output cycle of time-base timer ($2^{20}/F_{CH}$*)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Output cycle of watch prescaler ($2^{14}/F_{CL}$)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Output cycle of watch prescaler ($2^{13}/F_{CL}$)</td></tr>
<tr><td>X</td><td>X</td><td>1</td><td>Output cycle of sub-CR timer ($2^{16}/F_{CRL}$)</td></tr>
</table>
*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used. When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.
• Write to these bits at the same time as activating the watchdog timer by the watchdog control bits.
• No change can be made once the watchdog timer is activated.
Note: Since the time-base timer in stopped in subclock mode, always select the output of the watch prescaler in subclock mode.</td></tr>
<tr><td>bit5</td><td>CSP:
Count clock select sub-CR selector bit</td></tr>
<tr><td>bit4</td><td>HWWDT:
Hardware watchdog activation bit</td><td>The bit is a read-only bit, used to confirm the start/stop of the hardware watchdog timer.
"1": The hardware watchdog timer has been activated.
"0": The hardware watchdog timer has stopped (The software watchdog timer can be activated).</td></tr>
<tr><td>bit3
to
bit0</td><td>WTE3, WTE2, WTE1, WTE0:
Watchdog control bits</td><td>These bits are used to control the watchdog timer.
<b>Writing "0101$_B$"</b>:Activates the watchdog timer (in first write after reset) or clears it (from second write after reset).
<b>Writing other than "0101$_B$"</b>: Has no effect on operation.
• When these bits are read, they always return "0000$_B$".</td></tr>
</table>

Note:

Using the read-modify-write (RMW) type of instruction to access the WDTC register is prohibited.

12.4 Operations of Watchdog Timer and Setting Procedure Example

The watchdog timer generates a watchdog reset when the watchdog timer counter overflows.

■ Operations of Watchdog Timer

● How to activate the watchdog timer

To activate the software watchdog timer

- The watchdog timer is activated when "0101_B" is written to the watchdog control bits of the watchdog timer control register (WDTC:WTE3 to WTE0) for the first time after a reset. The count clock switch bits of the watchdog timer control register (WDTC:CS1,CS0,CSP) should also be set at the same time.
- Once the watchdog timer is activated, a reset is the only way to stop its operation.

To activate the hardware watchdog timer

- To activate the hardware watchdog timer, write any value except "$A596_H$" to the addresses $FFBE_H$ and $FFBF_H$ on the Flash memory. After a reset, the data in $FFBE_H$ and $FFBF_H$ on the Flash memory are copied to the watchdog timer selection ID registers WDTH/WDTL ($0FEB_H$ /$0FEC_H$). Writing "$A597_H$" to the addresses $FFBE_H$ and $FFBF_H$ on the Flash memory enables the hardware watchdog timer except in one of the standby modes; writing any value other than "$A596_H$" and "$A597_H$" enables the hardware watchdog timer in all modes. See "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION" for details of the watchdog timer selection ID.
- Start operation after a reset.
- CS1, CS0, CSP bits are read-only bits, fixed at "001_B".
- The timer is cleared by a reset and resumes operation after the reset is released.

● Clearing the watchdog timer

- When the counter of the watchdog timer is not cleared within the interval time, it overflows, allowing the watchdog timer to generate a watchdog reset.
- The counter of the hardware watchdog timer is cleared when "0101_B" is written to the watchdog control bits of the watchdog timer control register (WDTC:WTE3 to WTE0). The counter of the software watchdog timer is cleared when "0101_B" is written to the watchdog control bits of the watchdog timer control register (WDTC:WTE3 to WTE0) for the second time and from the second time onward.
- The watchdog timer is cleared at the same time as the timer selected as the count clock (time-base timer or watch prescaler) is cleared.

● Operation in standby mode

Regardless of the clock mode selected, the watchdog timer clears its counter and stops the operation when transiting to standby mode (sleep/stop/time-base timer/watch), except in the case of selecting the hardware activation with the hardware watchdog timer running in standby mode.

Once released from standby mode, the timer restarts the operation, except in the case of selecting the hardware activation with the hardware watchdog timer running in standby mode.

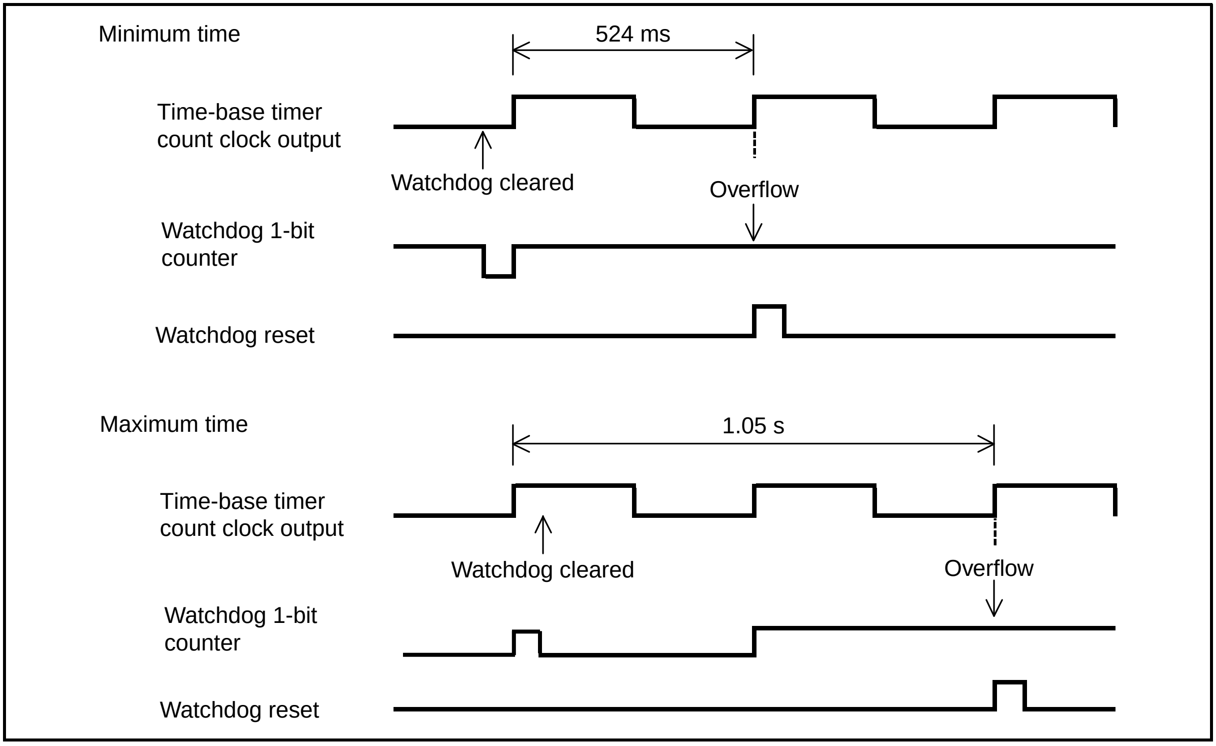
Note:

The watchdog timer is also cleared when the timer selected as the count clock (time-base timer or watch prescaler) is cleared. For this reason, the watchdog timer cannot function if the software is set to repeatedly clear the timer selected as the count clock of the watchdog timer at the interval time selected for the watchdog timer.

● Interval time

The interval time varies depending on the timing of clearing the watchdog timer. Figure 12.4-1 shows the relation between the timing of clearing timing the watchdog timer and the interval time when the time-base timer output $2^{21}/F_{CH}$ (F_{CH}: main clock) is selected as the count clock (main clock = 4 MHz).

Figure 12.4-1 Clearing Timing and Interval Time of Watchdog Timer



● Operation in subclock mode

When a watchdog reset is generated in subclock mode, the timer starts operating in main clock mode after the oscillation stabilization wait time has elapsed. The reset signal is output during this oscillation stabilization wait time.

■ Setting Procedure Example

Below is an example of procedure for setting the software watchdog timer.

1) Select the count clock. (WDTC:CS1, CS0, CSP)

2) Activate the watchdog timer. (WDTC:WTE3 to WTE0 = 0101_B)

3) Clear the watchdog timer. (WDTC:WTE3 to WTE0 = 0101_B)

Below is the procedure for setting the hardware watchdog timer.

1) Write any value except "A596$_H$" to the addresses FFBE$_H$ and FFBF$_H$ on the Flash memory. After a reset, the data in FFBE$_H$ and FFBF$_H$ on the Flash memory are copied to the watchdog timer selection ID registers WDTH/WDTL (0FEB$_H$ /0FEC$_H$). Writing "A597$_H$" to the addresses FFBE$_H$ and FFBF$_H$ on the Flash memory enables the hardware watchdog timer except in one of the standby modes; writing any value other than "A596$_H$" and "A597$_H$" enables the hardware watchdog timer in all modes. See "CHAPTER 32 NON-VOLATILE REGISTER (NVR) FUNCTION" for details of the watchdog timer selection ID.
2) Clear the watchdog timer. (WDTC:WTE3 to WTE0 = 0101$_B$)

12.5 Notes on Using Watchdog Timer

This section provides notes on using the watchdog timer.

■ Notes on Using Watchdog Timer

● Stopping the watchdog timer

Software watchdog timer

Once activated, the watchdog timer cannot be stopped until a reset is generated.

● Selecting the count clock

Software watchdog timer

The count clock switch bits (WDTC:CS1, CS0, CSP) can be modified only when the watchdog control bits (WDTC:WTE3 to WTE0) are set to "0101_B" after the activation of the watchdog timer. The count clock switch bits cannot be set by a bit manipulation instruction. Moreover, the bit settings should not be changed once the timer is activated.

In subclock mode, the time-base timer does not operate because the main clock stops oscillating.

In order to make the watchdog timer operate in subclock mode, it is necessary to select the watch prescaler as the count clock beforehand and set WDTC:CS1,CS0,CSP to "100_B" or "110_B" or "$XX1_B$".

● Clearing the watchdog timer

Clearing the counter used as the count clock of the watchdog timer (time-base timer or watch prescaler or sub-CR timer) also clears the counter of the watchdog timer.

The counter of the watchdog timer is cleared when the watchdog timer transits to the sleep mode, stop mode or watch mode, except in the case of selecting the hardware activation with the hardware watchdog timer running in standby mode.

● Programming precaution

When creating a program in which the watchdog timer is cleared repeatedly in the main loop, set the processing time of the main loop including the interrupt processing time to the minimum watchdog timer interval time or shorter.

● Hardware watchdog timer (with timer running in standby mode)

The watchdog timer does not stop in stop mode, sleep mode, time-base timer mode or watch mode. Therefore, the watchdog timer is not to be cleared by the CPU even if the internal clock stops. (in stop mode, sleep mode, time-base timer mode or watch mode).

Regularly release the device from standby mode and clear the watchdog timer. However, depending on the setting of the oscillation stabilization wait time setting register, a watchdog reset may be generated after the CPU wakes up from stop mode in subclock mode or sub-CR clock mode.

Take account of the setting of the subclock stabilization wait time when selecting the subclock.

CHAPTER 13
WATCH PRESCALER

This chapter describes the functions and operations of the watch prescaler.

13.1 Overview of Watch Prescaler

The watch prescaler is a 16-bit down-counting, free-run counter, which is synchronized with the subclock divided by two or the sub-CR clock divided by two. It has an interval timer function that continuously generates interrupt requests at regular intervals.

■ Interval Timer Function

The interval timer function continuously generates interrupt requests at regular intervals, using the subclock divided by two or the sub-CR clock divided by two as its count clock.

- The counter of the watch prescaler counts down and an interrupt request is generated whenever the selected interval time has elapsed.
- The interval time can be selected from the following eight types:

Table 13.1-1 shows the interval times of the watch prescaler.

Table 13.1-1 Interval Times of Watch Prescaler

	Interval time (Sub-CR clock) ($2^n \times 2/F_{CRL}$ *1)	Interval time (Subclock) ($2^n \times 2/F_{CL}$ *2)
n=10	20.48 ms	62.5 ms
n=11	40.96 ms	125 ms
n=12	81.92 ms	250 ms
n=13	163.84 ms	500 ms
n=14	327.68 ms	1 s
n=15	655.36 ms	2 s
n=16	1.311 s	4 s
n=17	2.621 s	8 s

*1: $2/F_{CRL}$ = 20 µs when F_{CRL} = 100 kHz

*2: $2/F_{CL}$ = 61.035 µs when F_{CL} = 32.768 kHz

Note:

Refer to the data sheet of the MB95410H/470H Series for the accuracy of the sub-CR clock frequency.

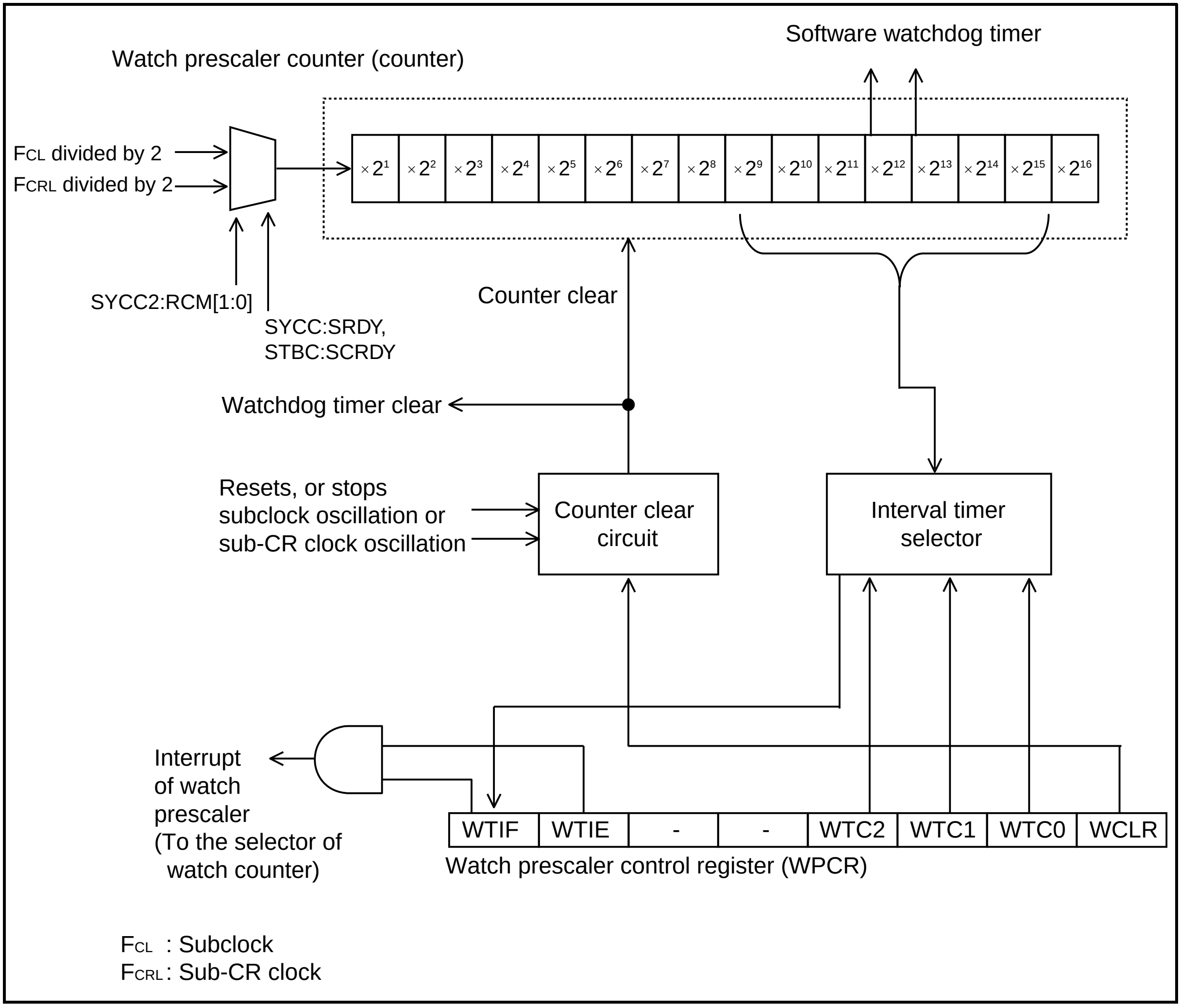
13.2 Configuration of Watch Prescaler

The watch prescaler consists of the following blocks:

- **Watch prescaler counter**
- **Counter clear circuit**
- **Interval timer selector**
- **Watch prescaler control register (WPCR)**

■ Block Diagram of Watch Prescaler

Figure 13.2-1 Block Diagram of Watch Prescaler



- Watch prescaler counter (counter)

  This is a 16-bit down-counter that uses the subclock divided by two or the sub-CR clock divided by two as its count clock.

- Counter clear circuit

  This circuit controls the clearing of the watch prescaler.

- Interval timer selector

  This circuit selects one out of the eight bits used for the interval timer among 16 bits available in the watch prescaler counter.

- Watch prescaler control register (WPCR)

  This register selects the interval time, clears the counter, controls interrupts and checks the status.

■ Input Clock

The watch prescaler uses the subclock divided by two or the sub-CR clock divided by two as its input clock (count clock).

■ Output Clock

The watch prescaler supplies its clock to the timer for the software watchdog timer and the watch counter.

13.3 Register of Watch Prescaler

Figure 13.3-1 shows the register of the watch prescaler.

■ Register of Watch Prescaler

Figure 13.3-1 Register of Watch Prescaler

Watch prescaler control register (WPCR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$000B_H$	WTIF	WTIE	-	-	WTC2	WTC1	WTC0	WCLR	00000000_B
	R(RM1),W	R/W	R0/WX	R0/WX	R/W	R/W	R/W	R0,W	

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0,W : Write only (Writable. The read value is "0".)
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

13.3.1 Watch Prescaler Control Register (WPCR)

The watch prescaler control register (WPCR) is a register used to select the interval time, clear the counter, control interrupts and check the status of the watch prescaler.

■ Watch Prescaler Control Register (WPCR)

Figure 13.3-2 Watch Prescaler Control Register (WPCR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$000B_H$	WTIF	WTIE	-	-	WTC2	WTC1	WTC0	WCLR	00000000_B
	R(RM1),W	R/W	R0/WX	R0/WX	R/X	R/W	R/W	R0,W	

WCLR	Watch timer initialization bit	
	Read	Write
0	"0" is always read.	No change No effect on operation
1	-	Clears all counter bits of the watch prescaler to "1".

WTC2	WTC1	WTC0	Interval time (Subclock FCL=32.768 kHz)	Interval time (Sub-CR clock FCRL=100 kHz)
1	0	0	$2^{10} \times 2/F_{CL}$ (62.5ms)	$2^{10} \times 2/F_{CRL}$ (20.48 ms)
0	0	0	$2^{11} \times 2/F_{CL}$ (125 ms)	$2^{11} \times 2/F_{CRL}$ (40.96 ms)
0	0	1	$2^{12} \times 2/F_{CL}$ (250 ms)	$2^{12} \times 2/F_{CRL}$ (81.92 ms)
0	1	0	$2^{13} \times 2/F_{CL}$ (500 ms)	$2^{13} \times 2/F_{CRL}$ (163.84 ms)
0	1	1	$2^{14} \times 2/F_{CL}$ (1 s)	$2^{14} \times 2/F_{CRL}$ (327.68 ms)
1	0	1	$2^{15} \times 2/F_{CL}$ (2 s)	$2^{15} \times 2/F_{CRL}$ (655.36 ms)
1	1	0	$2^{16} \times 2/F_{CL}$ (4 s)	$2^{16} \times 2/F_{CRL}$ (1.311 s)
1	1	1	$2^{17} \times 2/F_{CL}$ (8 s)	$2^{17} \times 2/F_{CRL}$ (2.621 s)

WTIE	Interrupt request enable bit
0	Disables interrupt request output.
1	Enables interrupt request output.

WTIF	Watch interrupt request flag bit	
	Read	Write
0	Interval time has not elapsed.	Clears the bit.
1	Interval time has elapsed.	No change No effect on operation

R/W	:	Readable/writable (The read value is the same as the write value.)
R(RM1),W	:	Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0,W	:	Write only (Writable. The read value is "0".)
R0/WX	:	The read value is "0". Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

Table 13.3-1 Functions of Bits in Watch Prescaler Control Register (WPCR)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7</td><td>WTIF:
Watch interrupt request flag bit</td><td>This bit becomes "1" when the selected interval time of the watch prescaler has elapsed.
• An interrupt request is generated when this bit and the interrupt request enable bit (WTIE) are set to "1".
<b>Writing "0"</b>: Clears this bit to "0".
<b>Writing "1"</b>: Has no effect on operation.
• If this bit is read by the read-modify-write (RMW) type of instruction, it always returns "1".</td></tr>
<tr><td>bit6</td><td>WTIE:
Interrupt request enable bit</td><td>This bit enables or disables the output of interrupt requests to interrupt controller.
<b>Writing "0"</b>: Disables the interrupt request output of the watch prescaler.
<b>Writing "1"</b>: Enables the interrupt request output of the watch prescaler.
An interrupt request is output when this bit and the watch interrupt request flag bit (WTIF) are set to "1".</td></tr>
<tr><td>bit5,
bit4</td><td>Undefined bits</td><td>Their read values are always "0". Writing a value to these bits has no effect on operation.</td></tr>
<tr><td>bit3
to
bit1</td><td>WTC2 to WTC0:
Watch interrupt interval time select bits</td><td>These bits select the interval time.
<table>
<tr><th>WTC2</th><th>WTC1</th><th>WTC0</th><th>Interval time
(Subclock F_{CL} = 32.768 kHz)</th><th>Interval time
(Sub-CR clock F_{CRL} = 100 kHz)</th></tr>
<tr><td>1</td><td>0</td><td>0</td><td>$2^{10} \times 2/F_{CL}$ (62.5 ms)</td><td>$2^{10} \times 2/F_{CRL}$ (20.48 ms)</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>$2^{11} \times 2/F_{CL}$ (125 ms)</td><td>$2^{11} \times 2/F_{CRL}$ (40.96 ms)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>$2^{12} \times 2/F_{CL}$ (250 ms)</td><td>$2^{12} \times 2/F_{CRL}$ (81.92 ms)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>$2^{13} \times 2/F_{CL}$ (500 ms)</td><td>$2^{13} \times 2/F_{CRL}$ (163.84 ms)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>$2^{14} \times 2/F_{CL}$ (1 s)</td><td>$2^{14} \times 2/F_{CRL}$ (327.68 ms)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>$2^{15} \times 2/F_{CL}$ (2 s)</td><td>$2^{15} \times 2/F_{CRL}$ (655.36 ms)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>$2^{16} \times 2/F_{CL}$ (4 s)</td><td>$2^{16} \times 2/F_{CRL}$ (1.311 s)</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>$2^{17} \times 2/F_{CL}$ (8 s)</td><td>$2^{17} \times 2/F_{CRL}$ (2.621 s)</td></tr>
</table></td></tr>
<tr><td>bit0</td><td>WCLR:
Watch timer initialization bit</td><td>This bit clears all counter bits of the watch prescaler to "1".
<b>Writing "0"</b>: Has no effect on operation.
<b>Writing "1"</b>: Initializes all counter bits to "1".
When this bit is read, it always returns "0".
Note: When the output of the watch prescaler is selected as the count clock of the software watchdog timer, clearing the watch prescaler with this bit also clears the software watchdog timer.</td></tr>
</table>

13.4 Interrupts of Watch Prescaler

An interrupt request is generated when the selected interval time of the watch prescaler has elapsed (interval timer function).

■ Interrupts in Operation of Interval Timer Function (Watch Interrupts)

In any mode except the stop mode in which the subclock mode is used, if the watch prescaler counter counts up using the source oscillation of the subclock and the time of the interval timer has elapsed, the watch interrupt request flag bit is set to "1" (WPCR:WTIF = 1). At that time, if the interrupt request enable bit has been enabled (WPCR:WTIE = 1), an interrupt request (IRQ20) is output from the watch prescaler to the interrupt controller.

- Regardless of the value in the WTIE bit, the WTIF bit is set to "1" as soon as the time set by the watch interrupt interval time select bits has elapsed.
- When the WTIF bit is set to "1", changing the WTIE bit from the disable state to the enable state (WPCR:WTIE = 0 → 1) immediately generates an interrupt request.
- The WTIF bit will not be set to "1" if the counter is cleared (WPCR:WCLR = 1) at the same time as the selected bit overflows.
- Write "0" to the WTIF bit in the interrupt service routine to clear an interrupt request to "0".

Note:

To enable the output of interrupt requests after releasing a reset, set the WTIE bit in the WPCR register to "1" and clear the WTIF bit in the same register simultaneously.

■ Interrupts of Watch Prescaler

Table 13.4-1 Interrupts of Watch Prescaler

Item	Description
Interrupt condition	Interval time set by "WPCR:WTC2 to WTC0" has elapsed.
Interrupt flag	WPCR:WTIF
Interrupt enable	WPCR:WTIE

■ Register and Vector Table Addresses Related to Interrupts of Watch Prescaler

Table 13.4-2 Register and Vector Table Addresses Related to Interrupts of Watch Prescaler

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
Watch prescaler*	IRQ20	ILR5	L20	FFD2$_H$	FFD3$_H$

*: The watch prescaler uses the same interrupt request number and vector table addresses as the watch counter.

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

13.5 Operations of Watch Prescaler and Setting Procedure Example

The watch prescaler operates as an interval timer.

■ Operations of Interval Timer Function (Watch Prescaler)

The counter of the watch prescaler continues to count down using the subclock divided by two as its count clock as long as the subclock oscillates.

When cleared (WPCR:WCLR = 1), the counter starts counting down from "FFFF_H". Once it reaches "0000_H", it returns to "FFFF_H" to continue counting. As soon as the time set by the interrupt interval time select bits has elapsed during the counting down, the watch interrupt request flag bit (WPCR:WTIF) is set to "1" in any mode except the stop mode in which the subclock mode is used. In other words, a watch interrupt request is generated at every selected interval time, based on the time when the counter was last cleared.

■ Clearing Watch Prescaler

If the watch prescaler is cleared, other peripheral functions that are using the watch prescaler output are affected by changes in count time and by other factors.

When clearing the counter using the watch prescaler initialization bit (WPCR:WCLR), modify the settings of other peripheral functions so that clearing the counter does not have any unexpected effect on them.

When the output of the watch prescaler is selected as the count clock, clearing the watch prescaler also clears the watchdog timer.

The watch prescaler is cleared not only by the watch prescaler initialization bit (WPCR:WCLR) but also when the subclock is stopped and the oscillation stabilization wait time is necessary. The watch prescaler is cleared in the following situations:

- When the device transits from the subclock mode or sub-CR clock mode to the stop mode
- When the subclock oscillation enable bits in the system clock control register 2 (SYCC2:SOSCE or SCRE) is set to "0" in main clock mode or main CR clock mode.

In addition, the counter of the watch prescaler is cleared and stops operating when a reset is generated.

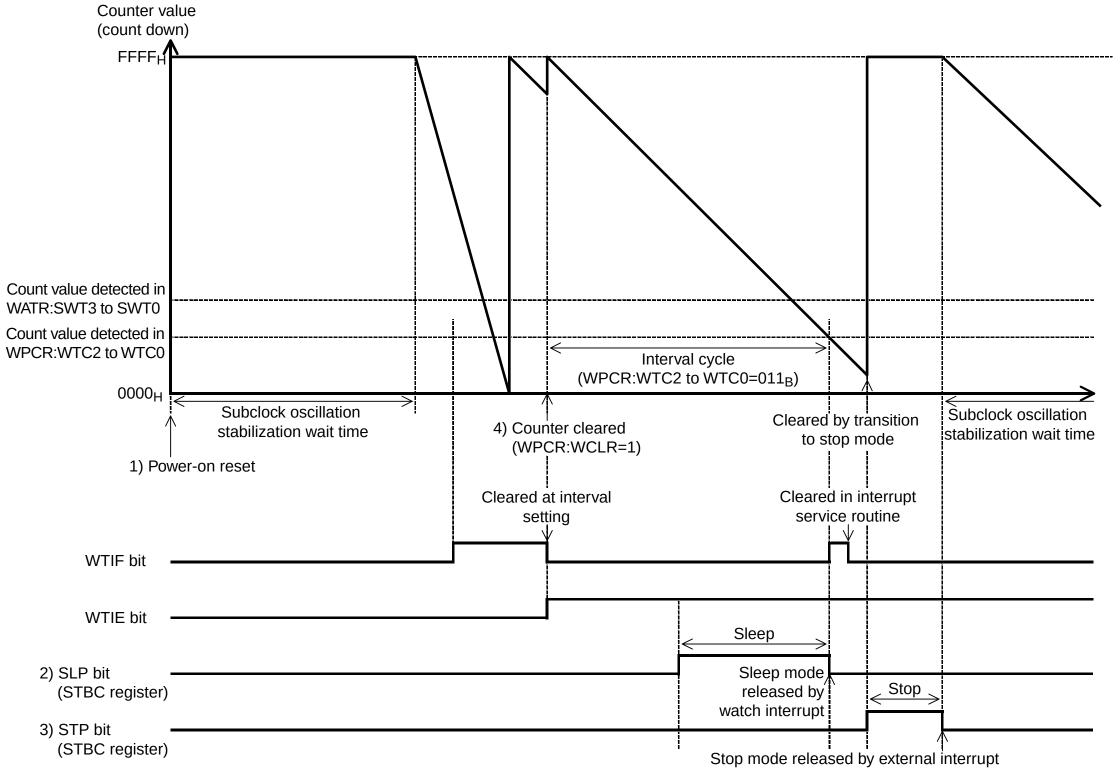
■ Operation Examples of Watch Prescaler

Figure 13.5-1 shows operating examples under the following conditions:

1) When a power-on reset occurs
2) When the device transits to the sleep mode during the operation of the interval timer function in subclock mode or sub-CR clock mode
3) When the device transits to the stop mode during the operation of the interval timer function in subclock mode or sub-CR clock mode
4) When a request for clearing the counter is issued

The same operation is performed when changing to the watch mode as for when changing to the sleep mode.

Figure 13.5-1 Operation Examples of Watch Prescaler


Counter value
(count down)
FFFF$_H$
Count value detected in WATR:SWT3 to SWT0
Count value detected in WPCR:WTC2 to WTC0
0000$_H$
Interval cycle
(WPCR:WTC2 to WTC0=011$_B$)
Subclock oscillation stabilization wait time
1) Power-on reset
4) Counter cleared
(WPCR:WCLR=1)
Cleared by transition to stop mode
Subclock oscillation stabilization wait time
Cleared at interval setting
Cleared in interrupt service routine
WTIF bit
WTIE bit
Sleep
2) SLP bit
(STBC register)
Sleep mode released by watch interrupt
Stop
3) STP bit
(STBC register)
Stop mode released by external interrupt

• When setting interval time select bits in the watch prescaler control register (WPCR:WTC2 to WTC0) to "011$_B$" (2^{14} × 2/F$_{CL}$)

• WPCR:WTC2 to WTC0	: Interval time select bits in watch prescaler control register
• WPCR:WCLR	: Watch timer initialization bit in watch prescaler control register
• WPCR:WTIF	: Watch interrupt request flag bit in watch prescaler control register
• WPCR:WTIE	: Watch interrupt request enable bit in watch prescaler control register
• STBC:SLP	: Sleep bit in standby control register
• STBC:STP	: Stop bit in standby control register
• WATR:SWT3 to SWT0	: Subclock oscillation stabilization wait time select bits in oscillation stabilization wait time setting register

■ Setting Procedure Example

Below is an example of procedure for setting the watch prescaler.

● Initial settings

1) Set the interrupt level.	(ILR5)
2) Set the interval time.	(WPCR:WTC2 to WTC0)
3) Enable interrupts.	(WPCR:WTIE = 1)
4) Clear the counter.	(WPCR:WCLR = 1)

● Processing interrupts

1) Clear the interrupt request flag.	(WPCR:WTIF = 0)
2) Process an interrupt.	

13.6 Notes on Using Watch Prescaler

This section provides notes on using the watch prescaler.

■ Notes on Using Watch Prescaler

● When setting interrupt processing in a program

The watch prescaler cannot be waken up from interrupt processing if the watch interrupt request flag bit (WPCR:WTIF) is set to "1" and the interrupt request is enabled (WPCR:WTIE = 1). Always clear the WTIF bit in the interrupt routine.

● Clearing the watch prescaler

When the watch prescaler is selected as the count clock of the software watchdog timer (WDTC:CS1, CS0, CSP = 100_B or 110_B), clearing the watch prescaler also clears the software watchdog timer.

● Watch interrupts

In stop mode in which the main clock is used, the watch prescaler performs counting and generates the watch prescaler interrupt (IRQ20).

● Peripheral functions receiving clock from the watch prescaler

If the counter of the watch prescaler is cleared when the output of the watch prescaler is used in other peripheral functions, the operations of such peripheral functions may be affected such as the changing of their operating cycles.

After the counter of the watch prescaler is cleared, the clock for the software watchdog timer output from the watch prescaler returns to the initial state. However, since the software watchdog timer counter is also cleared at the same time as the clock for the software watchdog timer returns to the initial state, the software watchdog timer operates in its normal cycle.

13.7 Sample Settings for Watch Prescaler

This section provides sample settings for the watch prescaler.

■ Sample Settings

● How to initialize the watch prescaler

The watch timer initialization bit (WPCR:WCLR) is used.

Operation	Watch timer initialization bit (WCLR)
To initialize the watch prescaler	Set the bit to "1".

● How to select the interval time

The watch interrupt interval time select bits (WPCR:WTC2 to WTC0) are used to select the interval time.

● Interrupt-related register

The interrupt level setting register shown in the following table is used to select the interrupt level.

Interrupt source	Interrupt level setting register	Interrupt vector
Watch prescaler	Interrupt level setting register (ILR5) Address: $0007E_H$	#20 Address: $0FFD2_H$

● How to enable/disable/clear interrupts

Interrupt request enable bit, Watch interrupt request flag

The interrupt request enable bit (WPCR:WTIE) is used to enable interrupts.

Operation	Interrupt request enable bit (WTIE)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

The watch interrupt request flag (WPCR:WTIF) is used to clear interrupt requests.

Operation	Watch interrupt request flag (WTIF)
To clear an interrupt request	Set the bit to "0".

CHAPTER 14

WATCH COUNTER

This chapter describes the functions and operations of the watch counter.

14.1 Overview of Watch Counter

The watch counter can generate interrupt requests ranging from min. 125 ms to max. 63 s intervals.

■ Watch Counter

The watch counter performs counting for the number of times specified in the register by using the selected count clock and generates an interrupt request. The count clock can be selected from the four types shown in Table 14.1-1. The count value can be set to any number from 0 to 63. When "0" is selected, no interrupt is generated.

When the count clock is set to 1s and the count value is set to "60", an interrupt is generated every one minute.

Table 14.1-1 Count Clock Types

Count clock	Count cycle when F_{CL} operates at 32.768 kHz
$2^{12}/F_{CL}$	125 ms
$2^{13}/F_{CL}$	250 ms
$2^{14}/F_{CL}$	500 ms
$2^{15}/F_{CL}$	1 s

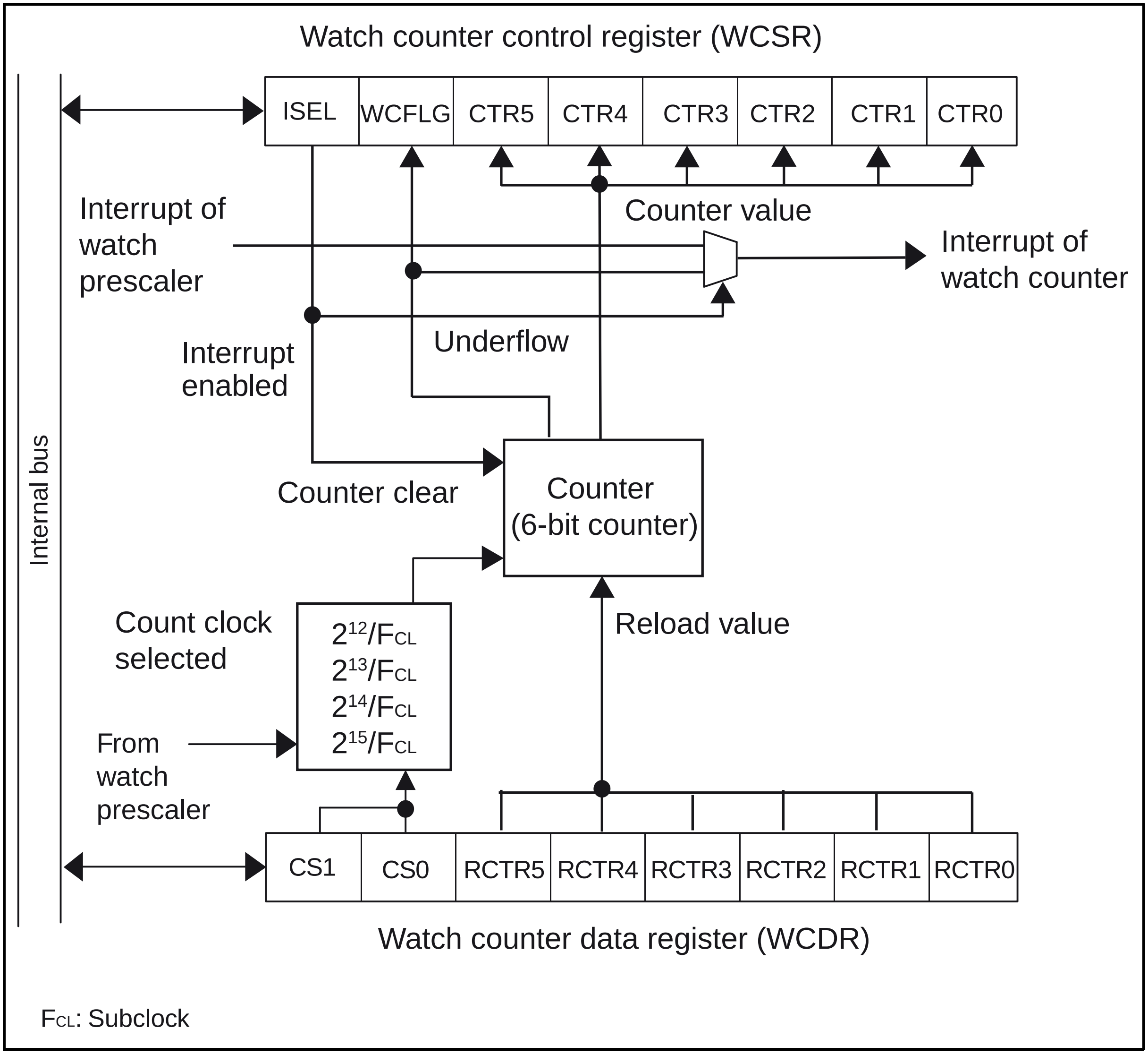
F_{CL}: Subclock

14.2 Configuration of Watch Counter

Figure 14.2-1 shows the block diagram of the watch counter.

■ Block Diagram of Watch Counter

Figure 14.2-1 Block Diagram of Watch Counter



- Counter

This is a 6-bit down-counter that uses the output clock of the watch prescaler as its count clock.

- Watch counter control register (WCSR)

This register controls interrupts and checks the status.

- Watch counter data register (WCDR)

This register sets the interval time and selects the count clock.

■ Input Clock

The watch counter uses the output clock of the watch prescaler as its input clock (count clock).

14.3 Registers of Watch Counter

Figure 14.3-1 shows the registers of the watch counter.

■ Registers of Watch Counter

Figure 14.3-1 Registers of Watch Counter

Watch counter data register (WCDR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FE3$_H$	CS1	CS0	RCTR5	RCTR4	RCTR3	RCTR2	RCTR1	RCTR0	00111111$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Watch counter control register (WCSR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0070$_H$	ISEL	WCFLG	CTR5	CTR4	CTR3	CTR2	CTR1	CTR0	00000000$_B$
	R/W	R(RM1),W	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

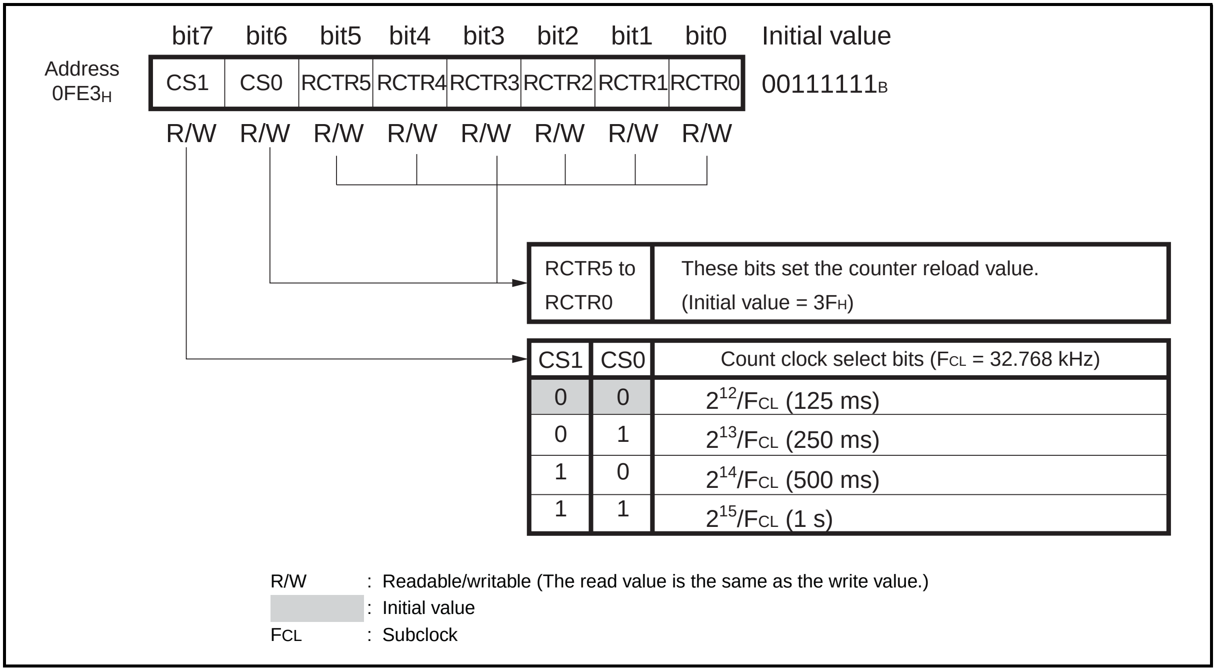
R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to it has no effect on operation.)

14.3.1 Watch Counter Data Register (WCDR)

The watch counter data register (WCDR) is used to select the count clock and set the counter reload value.

■ Watch Counter Data Register (WCDR)

Figure 14.3-2 Watch Counter Data Register (WCDR)



	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address 0FE3$_H$	CS1	CS0	RCTR5	RCTR4	RCTR3	RCTR2	RCTR1	RCTR0	00111111$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

RCTR5 to RCTR0	These bits set the counter reload value. (Initial value = 3F$_H$)

CS1	CS0	Count clock select bits (F_{CL} = 32.768 kHz)
0	0	$2^{12}/F_{CL}$ (125 ms)
0	1	$2^{13}/F_{CL}$ (250 ms)
1	0	$2^{14}/F_{CL}$ (500 ms)
1	1	$2^{15}/F_{CL}$ (1 s)

R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value
F_{CL} : Subclock

Table 14.3-1 Functions of Bits in Watch Counter Data Register (WCDR)

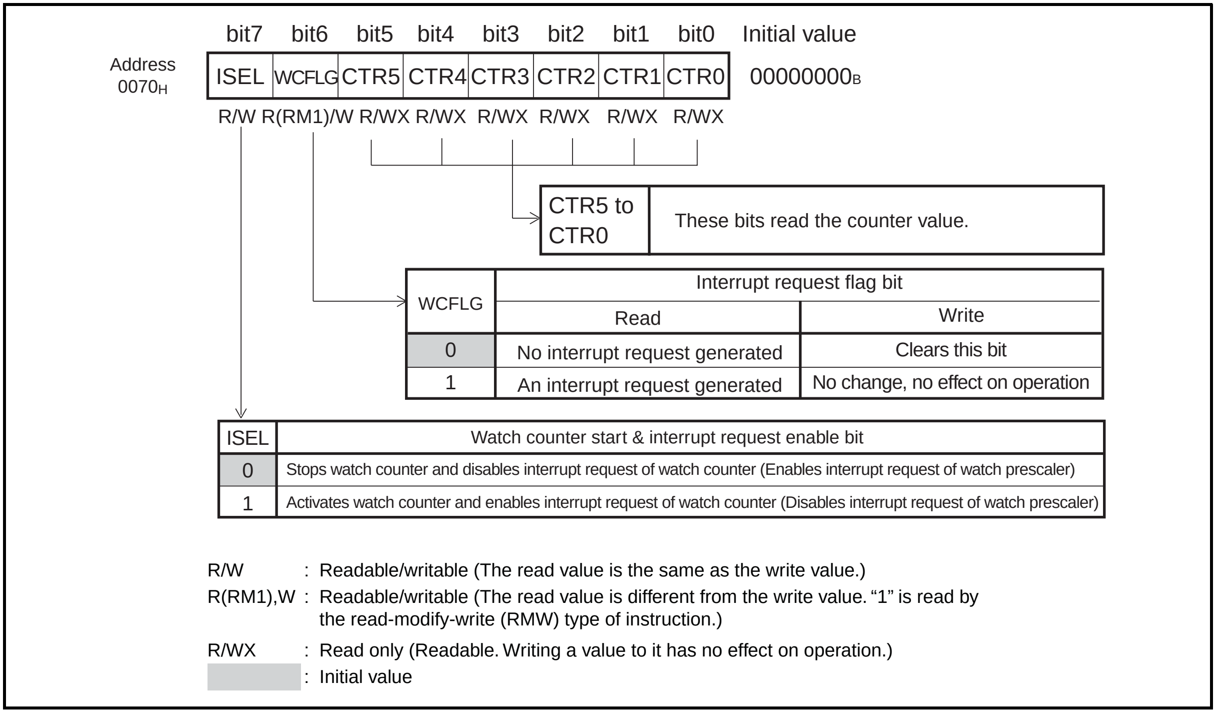
Bit name		Function
bit7, bit6	CS1, CS0: Count clock select bits	These bits select the clock for the watch counter. "00" = $2^{12}/F_{CL}$, "01" = $2^{13}/F_{CL}$, "10" = $2^{14}/F_{CL}$, "11" = $2^{15}/F_{CL}$ (F_{CL}: Subclock) These bits should be modified when the WCSR:ISEL bit is "0".
bit5 to bit0	RCTR5 to RCTR0: Counter reload value setting bits	These bits set the counter reload value. If the value is modified during counting, the modified value will become effective upon a reload after the counter underflows. **Writing "0"**: Generates no interrupt request. If the reload value (RCTR5 to RCTR0) is modified at the same time as an interrupt is generated (WCSR:WCFLG = 1), the correct value will not be reloaded. Therefore, the reload value must be modified before an interrupt is generated, such as when the watch counter is stopped (WCSR:ISEL = 0), or during the interrupt routine.

14.3.2 Watch Counter Control Register (WCSR)

The watch counter control register (WCSR) is used to control the operation and interrupts of the watch counter. It can also read the count value.

■ Watch Counter Control Register (WCSR)

Figure 14.3-3 Watch Counter Control Register (WCSR)



Address		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0070H		ISEL	WCFLG	CTR5	CTR4	CTR3	CTR2	CTR1	CTR0	00000000B
		R/W	R(RM1)/W	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

CTR5 to CTR0	These bits read the counter value.

WCFLG	Interrupt request flag bit	
	Read	Write
0	No interrupt request generated	Clears this bit
1	An interrupt request generated	No change, no effect on operation

ISEL	Watch counter start & interrupt request enable bit
0	Stops watch counter and disables interrupt request of watch counter (Enables interrupt request of watch prescaler)
1	Activates watch counter and enables interrupt request of watch counter (Disables interrupt request of watch prescaler)

R/W : Readable/writable (The read value is the same as the write value.)

R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)

R/WX : Read only (Readable. Writing a value to it has no effect on operation.)

▨ : Initial value

Table 14.3-2 Functions of Bits in Watch Counter Status Register (WCSR)

Bit name		Function
bit7	ISEL: Watch counter start & interrupt request enable bit	• This bit activates the watch counter and selects whether to enable interrupts of the watch counter or those of the watch prescaler. **Writing "0":** The watch counter is cleared and stopped. Moreover, interrupt requests of the watch counter are disabled, while interrupt requests of the watch prescaler are enabled. **Writing "1":** The interrupt request output of the watch counter is enabled and the counter starts operation. On the other hand, interrupt requests of the watch prescaler are disabled. • Always disable interrupts of the watch prescaler before setting this bit to "1" to select interrupts of the watch counter. • The watch counter performs counting, using an asynchronous clock from the watch prescaler. For this reason, an error of up to one count clock may occur at the beginning of a count cycle, depending on the timing for setting ISEL bit to "1".
bit6	WCFLG: Interrupt request flag bit	• This bit is set to "1" when the counter underflows. • When this bit and the ISEL bit are both set to "1", a watch counter interrupt is generated. **Writing "0":** Clears the bit. **Writing "1":** Has no effect on the operation. • "1" is always read in read-modify-write operation.
bit5 to bit0	CTR5 to CTR0: Counter read bits	• These bits can read the counter value during counting. It should be noted that the correct counter value may not be read if a read is attempted while the counter value is being changed. Therefore, read the counter value twice to check if the same value is read on both occasions before using it. • Write has no effect on the operation.

14.4 Interrupts of Watch Counter

The watch counter outputs interrupt requests when the counter underflows (counter value = 000001_B).

■ Interrupts of Watch Counter

When the counter of the watch counter underflows, the interrupt request flag bit (WCFLG) in the watch counter control register (WCSR) is set to "1". If the interrupt request enable bit (ISEL) of the watch counter is set to "1", an interrupt request of the watch counter is outputted to the interrupt controller.

Table 14.4-1 shows the interrupt control bits and interrupt sources of the watch counter.

Table 14.4-1 Interrupt Control Bits and Interrupt Sources of Watch Counter

Item	Description
Interrupt request flag bit	WCFLG bit in the WCSR register
Interrupt request enable bit	ISEL bit in the WCSR register
Interrupt source	Counter underflow

■ Register and Vector Table Addresses Related to Interrupts of Watch Counter

Table 14.4-2 Register and Vector Table Addresses Related to Interrupts of Watch Counter

Interrupt source	Interrupt request no.	Interrupt level setup register		Vector table address	
		Register	Setting bit	Upper	Lower
Watch counter*	IRQ20	ILR5	L20	$FFD2_H$	$FFD3_H$

*: The watch counter uses the same interrupt request number and vector table addresses as the watch prescaler.

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

14.5 Operations of Watch Counter and Setting Procedure Example

The watch counter counts down for the number of times specified in the count value by RCTR5 to RCTR0 bits, using the count clock selected by CS1 and CS0 bits, when the ISEL bit is set to "1". Once the counter underflows, WCFLG bit in the WCSR register is set to "1", generating an interrupt.

■ Setting Procedure of Watch Counter

The setting procedure of the watch counter is described below.

(1) Select the count clock (CS1 and CS0 bits) and set the counter reload value (RCTR5 to RCTR0 bits).

(2) Set the ISEL bit in the WCSR register to "1" to start a down count and enable interrupts. Also disable interrupts of the watch prescaler.
The watch counter performs counting by using a divided clock (asynchronous) from the watch prescaler. An error of up to one count clock may occur at the beginning of a count cycle, depending on the timing for setting the ISEL bit to "1".

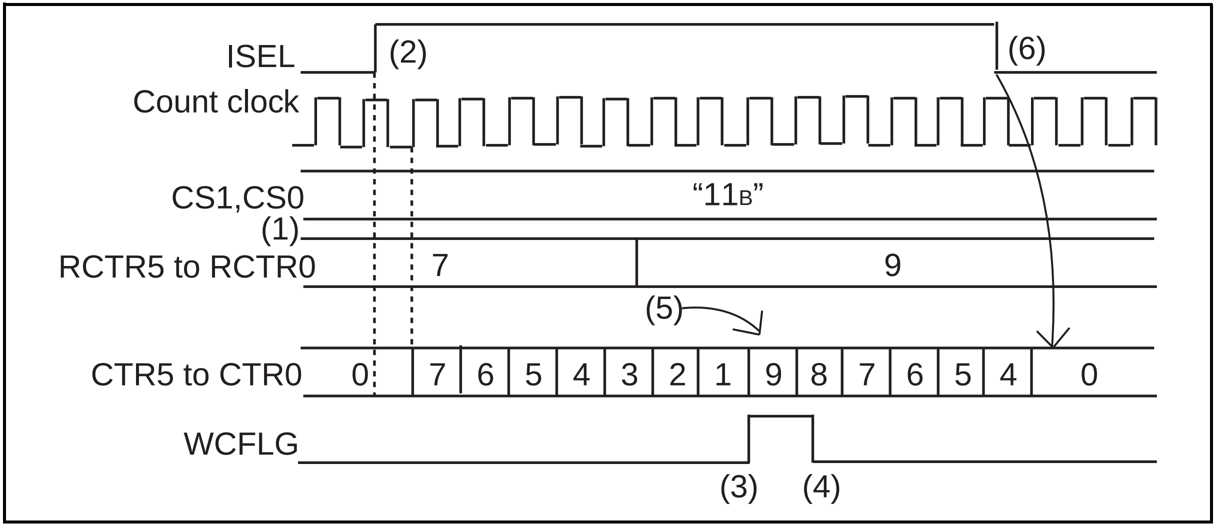
(3) When the counter underflows, the WCFLG bit in the WCSR register is set to "1", generating an interrupt.

(4) Write "0" to the WCFLG bit to clear it.

(5) If RCTR5 to RCTR0 bits are modified during counting, the reload value will be updated during a reload after the counter is set to "1".

(6) When writing "0" to the ISEL bit, the counter becomes "0" and stops operation.

Figure 14.5-1 Watch Counter Operation



Note:

To re-activate the counter by setting WCSR:ISEL to "1" after stopping it by setting WCSR:ISEL to "0", read WCSR:CTR[5:0] twice to ensure that WCSR:CTR[5:0] have been cleared to "000000$_B$".

■ Operation in Substop Mode

When the device enters the substop mode, the watch counter stops the count operation and the watch prescaler is also cleared. Therefore, the watch counter cannot count the correct value after the substop mode is cancelled. After the substop mode is cancelled, the ISEL bit must always be set to "0" to clear the counter before use. In any standby mode other than the substop mode, the watch counter continues to operate.

■ Operation in Main Stop Mode

The interrupt is not generated though the watch counter continues the count operation when entering the main stop mode. Moreover, the watch counter stops, too, when the subclock oscillation enable bit (SOSCE) in the system clock control register 2 (SYCC2) is set to "0".

■ Setting Procedure Example

Below is an example of procedure for setting the watch counter.

● Initial settings

1) Set the interrupt level. (ILR5)
2) Select the count clock. (WCDR:CS1, CS0)
3) Set the counter reload value. (WCDR:RCTR5 to RCTR0)
4) Activate the watch counter and enable interrupts. (WCSR:ISEL = 1)

● Interrupt processing

1) Clear the interrupt request flag. (WCSR:WCFLG = 0)
2) Process any interrupt.

14.6 Notes on Using Watch Counter

This section provides notes on using the watch counter.

- If the watch prescaler is cleared during the operation of the watch counter, the watch counter may not be able to perform normal operation. When clearing the watch prescaler, set the ISEL bit in the WCSR register to "0" to stop the watch counter in advance.
- To re-activate the counter by setting WCSR:ISEL to "1" after stopping it by setting WCSR:ISEL to "0", read WCSR:CTR[5:0] twice to ensure that WCSR:CTR[5:0] have been cleared to "000000_B".

14.7 Sample Settings for Watch Counter

This section provides sample settings for the watch counter.

■ Sample Settings

● How to enable/stop the watch counter

Use the watch counter start & interrupt request enable bit (WCSR:ISEL).

Operation	Watch counter start & interrupt request enable bit (ISEL)
To enable the watch counter	Set the bit to "1".
To stop the watch counter	Set the bit to "0".

● How to select the count clock

Use the count clock select bits (WCDR:CS1, CS0) to select a count clock.

● Interrupt-related register

The interrupt level is set in the interrupt level setting register shown in the following table.

Interrupt source	Interrupt level setting register	Interrupt vector
Watch counter	Interrupt level setting register (ILR5) Address: $0007E_H$	#20 Address: $0FFD2_H$

● How to enable/disable/clear interrupts

Interrupt request enable bit, Interrupt request flag bit

Use the watch counter start & interrupt request enable bit (WCSR:ISEL) to enable interrupts.

Operation	Watch counter start & interrupt request enable bit (ISEL)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

Use the interrupt request flag bit (WCSR:WCFLG) to clear the interrupt request.

Operation	Interrupt request flag bit (WCFLG)
To clear an interrupt request	Set the bit to "0".

CHAPTER 15 WILD REGISTER FUNCTION

This chapter describes the functions and operations of the wild register function.

15.1 Overview of Wild Register Function

The wild register function can be used to patch bugs in a program with addresses and amendment data, both of which are to be set in built-in registers. This section describes the wild register function.

■ Wild Register Function

The wild register consists of three wild register data setting registers, three wild register address setting registers, a 1-byte address compare enable register and a 1-byte wild register data test setting register. If addresses and data that are to be modified are set to these registers, the ROM data can be replaced with modification data set in the registers. Data of up to three different addresses can be modified.

The wild register function can be used to debug a program after creating the mask and to patch bugs in the program.

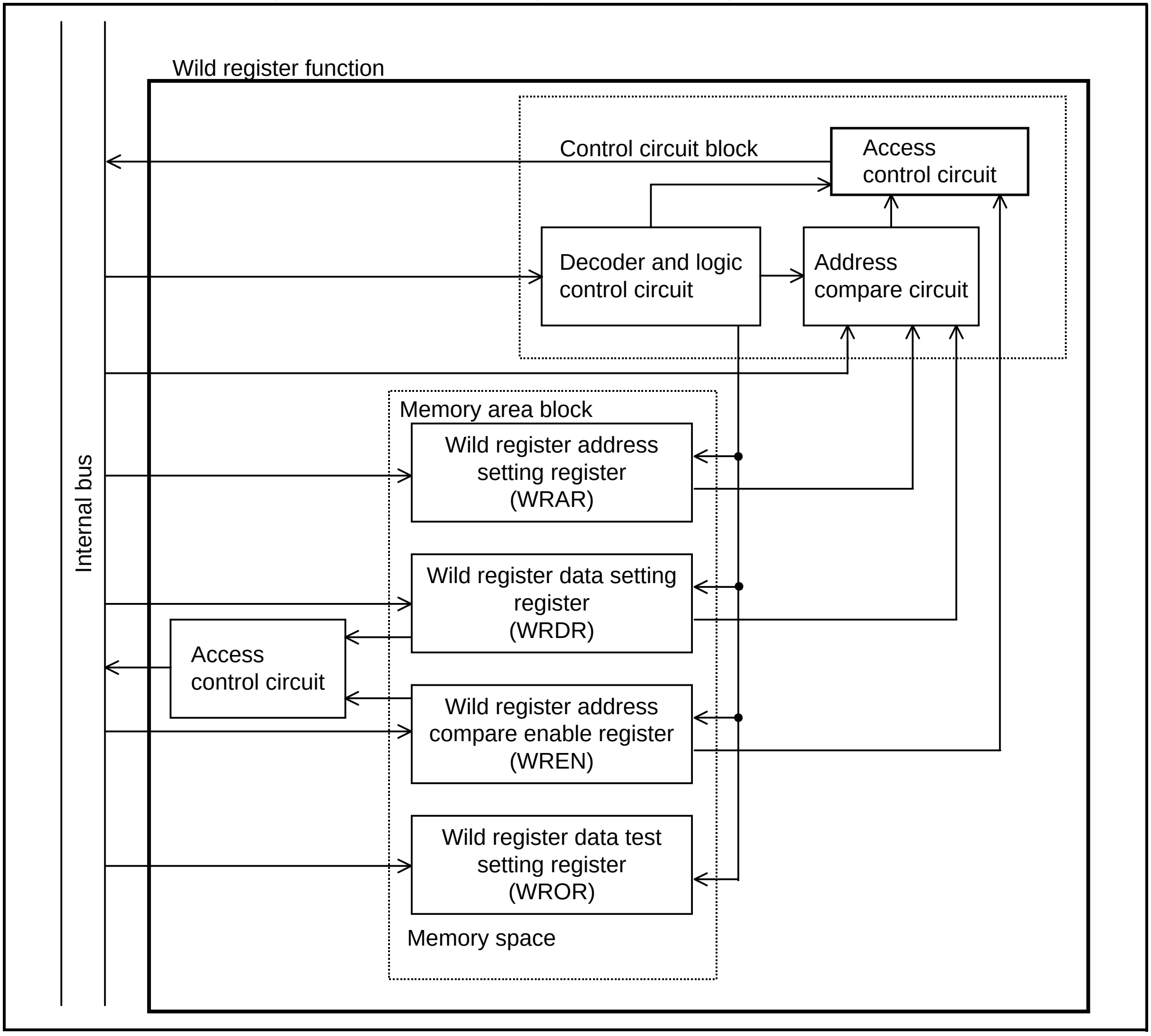
15.2 Configuration of Wild Register Function

The block diagram of the wild register is shown below. The wild register consists of the following blocks:

- **Memory area block**
  Wild register data setting register (WRDR0 to WRDR2)
  Wild register address setting register (WRAR0 to WRAR2)
  Wild register address compare enable register (WREN)
  Wild register data test setting register (WROR)
- **Control circuit block**

■ Block Diagram of Wild Register Function

Figure 15.2-1 Block Diagram of Wild Register Function


Wild register function
Control circuit block
Access control circuit
Decoder and logic control circuit
Address compare circuit
Memory area block
Wild register address setting register (WRAR)
Wild register data setting register (WRDR)
Wild register address compare enable register (WREN)
Wild register data test setting register (WROR)
Access control circuit
Memory space
Internal bus

- Memory area block

The memory area block consists of the wild register data setting registers (WRDR), wild register address setting registers (WRAR), wild register address compare enable register (WREN) and wild register data test setting register (WROR). The wild register function is used to specify the addresses and data that need to be replaced. The wild register address compare enable register (WREN) enables the wild register function for each wild register data setting register (WRDR). In addition, the wild register data test setting register (WROR) enables the normal read function for each wild register data setting register (WRDR).

- Control circuit block

This circuit compares the actual address data with addresses set in the wild register address setting registers (WRAR). If they match, the circuit outputs the data from the wild register data setting register (WRDR) to the data bus. The operation of the control circuit block is controlled by the wild register address compare enable register (WREN).

15.3 Registers of Wild Register Function

The registers of the wild register function include the wild register data setting registers (WRDR), wild register address setting registers (WRAR), wild register address compare enable register (WREN) and wild register data test setting register (WROR).

■ Registers of Wild Register Function

Figure 15.3-1 Registers of Wild Register Function

Wild register data setting registers (WRDR0 to WRDR2)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
WRDR0	$0F82_H$	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	00000000_B
WRDR1	$0F85_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
WRDR2	$0F88_H$									

Wild register address setting registers (WRAR0 to WRAR2)

	Address	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	Initial value
WRAR0	$0F80_H$, $0F81_H$	RA15	RA14	RA13	RA12	RA11	RA10	RA9	RA8	00000000_B
WRAR1	$0F83_H$, $0F84_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
WRAR2	$0F86_H$, $0F87_H$	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
		RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	00000000_B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Wild register address compare enable register (WREN)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0076_H	-	-	Reserved	Reserved	Reserved	EN2	EN1	EN0	00000000_B
	R0/WX	R0/WX	R/W0	R/W0	R/W0	R/W	R/W	R/W	

Wild register data test setting register (WROR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0077_H	-	-	Reserved	Reserved	Reserved	DRR2	DRR1	DRR0	00000000_B
	R0/WX	R0/WX	R/W0	R/W0	R/W0	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)
R/W0 : The write value is "0". The read value is the same as the write value.
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

■ Wild Register Number

A wild register number is assigned to each wild register address setting register (WRAR) and each wild register data setting register (WRDR).

Table 15.3-1 Wild Register Numbers Corresponding to Wild Register Address Setting Registers and Wild Register Data Setting Registers

Wild register number	Wild register address setting register (WRAR)	Wild register data setting register (WRDR)
0	WRAR0	WRDR0
1	WRAR1	WRDR1
2	WRAR2	WRDR2

15.3.1 Wild Register Data Setting Registers (WRDR0 to WRDR2)

The wild register data setting registers (WRDR0 to WRDR2) are used to specify the data to be amended by the wild register function.

■ Wild Register Data Setting Registers (WRDR0 to WRDR2)

Figure 15.3-2 Wild Register Data Setting Registers (WRDR0 to WRDR2)

WRDR0

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F82_H$	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

WRDR1

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F85_H$	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

WRDR2

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F88_H$	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

Table 15.3-2 Functions of Bits in Wild Register Data Setting Register (WRDR)

Bit name		Function
bit7 to bit0	RD7 to RD0: Wild register data setting bits	These bits specify the data to be amended by the wild register function. • These bits are used to set the amendment data at the address assigned by the wild register address setting register (WRAR). Data is valid at an address corresponding to one of the wild register numbers. • The read access to one of these bits is enabled only when the data test setting bit in the wild register data test setting register (WROR) corresponding to the bit to be read is set to "1".

15.3.2 Wild Register Address Setting Registers (WRAR0 to WRAR2)

The wild register address setting registers (WRAR0 to WRAR2) are used to set the address to be amended by the wild register function.

■ Wild Register Address Setting Registers (WRAR0 to WRAR2)

Figure 15.3-3 Wild Register Address Setting Registers (WRAR0 to WRAR2)

WRAR0

Address	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	Initial value
$0F80_H$	RA15	RA14	RA13	RA12	RA11	RA10	RA9	RA8	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F81_H$	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

WRAR1

Address	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	Initial value
$0F83_H$	RA15	RA14	RA13	RA12	RA11	RA10	RA9	RA8	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F84_H$	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

WRAR2

Address	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	Initial value
$0F86_H$	RA15	RA14	RA13	RA12	RA11	RA10	RA9	RA8	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F87_H$	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

Table 15.3-3 Functions of Bits in Wild Register Address Setting Register (WRAR)

Bit name		Function
bit15 to bit0	RA15 to RA0: Wild register address setting bits	These bits set the address to be amended by the wild register function. The address to be assigned to amendment data is set to these bits. The address is to be specified according to the wild register number corresponding to a wild register address setting register.

15.3.3 Wild Register Address Compare Enable Register (WREN)

The wild register address compare enable register (WREN) enables/disables the operations of wild register functions using their respective wild register numbers.

■ Wild Register Address Compare Enable Register (WREN)

Figure 15.3-4 Wild Register Address Compare Enable Register (WREN)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0076_H	-	-	Reserved	Reserved	Reserved	EN2	EN1	EN0	00000000_B
	R0/WX	R0/WX	R/W0	R/W0	R/W0	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)
R/W0 : The write value is "0". The read value is the same as the write value.
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

Table 15.3-4 Functions of Bits in Wild Register Address Compare Enable Register (WREN)

Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5 to bit3	Reserved bits	Always set these bits to "0".
bit2 to bit0	EN2, EN1, EN0: Wild register address compare enable bits	These bits enable/disable the operation of the wild register. • EN0 corresponds to wild register number 0. • EN1 corresponds to wild register number 1. EN2 corresponds to wild register number 2. **Writing "0"**: Disables the operation of the wild register function. **Writing "1"**: Enables the operation of the wild register function.

15.3.4 Wild Register Data Test Setting Register (WROR)

The wild register data test setting register (WROR) enables/disables reading data from the corresponding wild register data setting register (WRDR0 to WRDR2).

■ Wild Register Data Test Setting Register (WROR)

Figure 15.3-5 Wild Register Data Test Setting Register (WROR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0077_H	-	-	Reserved	Reserved	Reserved	DRR2	DRR1	DRR0	00000000_B
	R0/WX	R0/WX	R/W0	R/W0	R/W0	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)
R/W0 : The write value is "0". The read value is the same as the write value.
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

Table 15.3-5 Functions of Bits in Wild Register Data Test Setting Register (WROR)

Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5 to bit3	Reserved bits	Always set these bits to "0".
bit2 to bit0	DRR2, DRR1, DRR0: Wild register data test setting bits	These bits enable/disable the normal reading from the corresponding data setting register of the wild register. • DRR0 enables/disables reading from the wild register data setting register (WRDR0). • DRR1 enables/disables reading from the wild register data setting register (WRDR1). • DRR2 enables/disables reading from the wild register data setting register (WRDR2). **Writing "0"**: Disables reading. **Writing "1"**: Enables reading.

15.4 Operations of Wild Register Function

This section describes the procedure for setting the wild register function.

■ Procedure for Setting Wild Register Function

Prepare a program that can read the value to be set in the wild register from external memory (e.g. EEPROM or FRAM) in the user program before using the wild register function. The setting method for the wild register is shown below.

This section does not include information on the method of communications between the external memory and the device.

- Write the address of the built-in ROM code that will be modified to the wild register address setting register (WRAR0 to WRAR2).
- Write a new code to the wild register data setting register (WRDR0 to WRDR2) corresponding to the wild register address setting register to which the address has been written.
- Write "1" to the EN bit in the wild register address compare enable register (WREN) corresponding to the wild register number to enable the wild register function represented by that wild register number.

Table 15.4-1 shows the procedure for setting the registers of the wild register function.

Table 15.4-1 Procedure for Setting Registers of Wild Register Function

Step	Operation	Operation example
1	Read replacement data from a peripheral function outside through a certain communication method.	Suppose the built-in ROM code to be modified is at the address F011$_H$ and the data to be modified is "B5$_H$", and there are three built-in ROM codes to be modified.
2	Write the replacement address to a wild register address setting register (WRAR0 to WRAR2).	Set wild register address setting registers (WRAR0 = F011$_H$, WRAR1 = ..., WRAR2 = ...).
3	Write a new ROM code (replacement for the built-in ROM code) to a wild register data setting register (WRDR0 to WRDR2).	Set the wild register data setting registers (WRDR0 = B5$_H$, WRDR1 =..., WRDR2 =...).
4	Enable the EN bit in the wild register address compare enable register (WREN) corresponding to the wild register number of the wild register function used.	Setting bit 0 of the address compare enable register (WREN) to "1" enables the wild register function of the wild register number 0. If the address matches the value set in the wild register address setting register (WRAR), the value of the wild register data setting register (WRDR) will be replaced with the built-in ROM code. When replacing more than one built-in ROM code, enable the related EN bits in the wild register address compare enable register (WREN) corresponding to respective built-in ROM codes.

■ Wild Register Function Applicable Addresses

The wild register function can be applied to all address space except the address "0078$_H$".

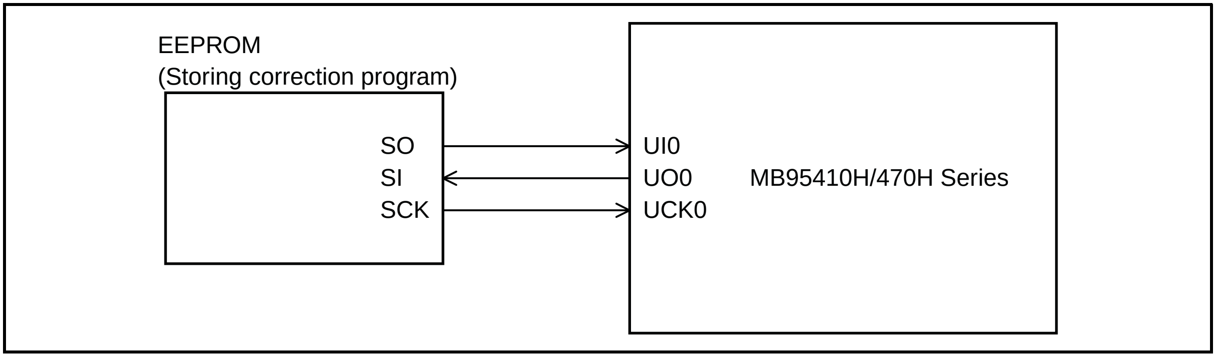
Since the address "0078$_H$" is used as a mirror address for the register bank pointer and the direct bank pointer, this address cannot be patched.

15.5 Typical Hardware Connection Example

Below is an example of typical hardware connection for the application of the wild register function.

■ Hardware Connection Example

Figure 15.5-1 Typical Hardware Connection Example



CHAPTER 16
EXTERNAL INTERRUPT CIRCUIT

This chapter describes the functions and operations of the external interrupt circuit.

16.1 Overview of External Interrupt Circuit

The external interrupt circuit detects edges on the signal that is input to the external interrupt pin, and outputs interrupt requests to the interrupt controller.

■ Function of External Interrupt Circuit

The function of the external interrupt circuit is to detect any edge of a signal that is input to an external interrupt pin and to generate an interrupt request to the interrupt controller. The interrupt generated according to this interrupt request can cause the device to wake up from standby mode and return to its normal operating state. Therefore, the operating mode of the device can be changed when a signal is input to the external interrupt pin.

16.2 Configuration of External Interrupt Circuit

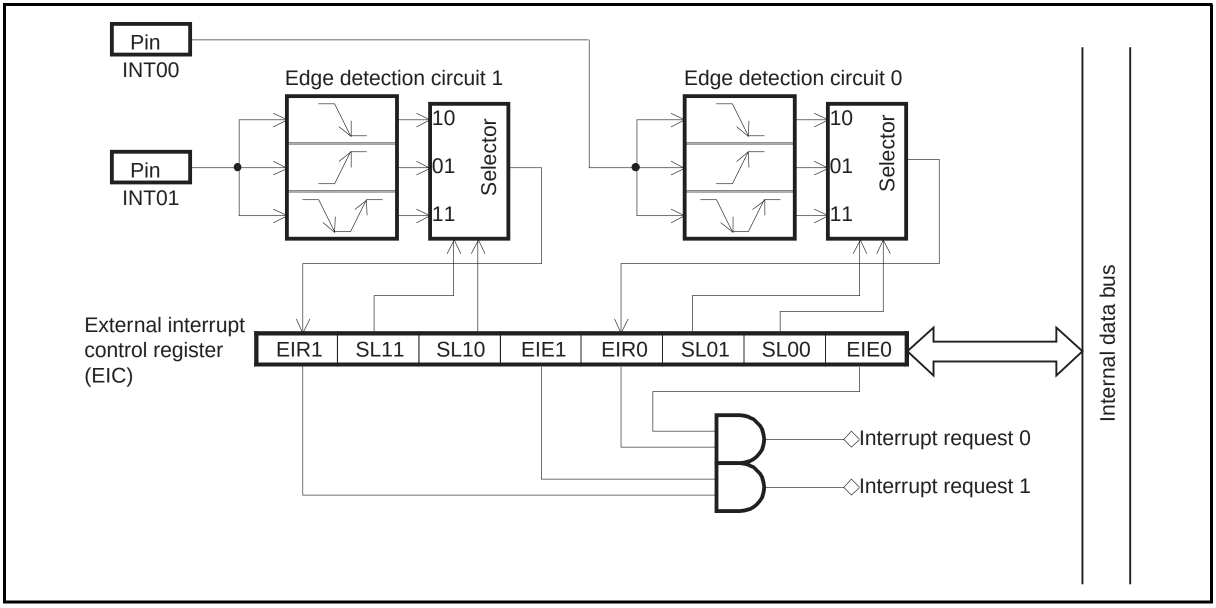
The external interrupt circuit consists of the following blocks:

- **Edge detection circuit**
- **External interrupt control register**

■ Block Diagram of External Interrupt Circuit

Figure 16.2-1 is the block diagram of the external interrupt circuit.

Figure 16.2-1 Block Diagram of External Interrupt Circuit



● Edge detection circuit

When the polarity of the edge detected on a signal input to an external interrupt circuit pin (INT) matches the polarity of the edge selected in the interrupt control register (EIC), a corresponding external interrupt request flag bit (EIR) is set to "1".

● External interrupt control register (EIC)

This register is used to select an edge, enable or disable interrupt requests, check for interrupt requests, etc.

16.3 Channels of External Interrupt Circuit

This section describes the channels of the external interrupt circuit.

■ Channels of External Interrupt Circuit

The MB95410H/470H Series has 4 units of external interrupt circuit.

Table 16.3-1 shows the pins of the external interrupt circuit.

Table 16.3-1 Pins of External Interrupt Circuit

Unit	Pin name	Pin function
1	INT00	External interrupt input ch. 0
	INT01	External interrupt input ch. 1
2	INT02	External interrupt input ch. 2
	INT03	External interrupt input ch. 3
3	INT04	External interrupt input ch. 4
	INT05	External interrupt input ch. 5
4	INT06	External interrupt input ch. 6
	INT07	External interrupt input ch. 7

Table 16.3-2 Registers of External Interrupt Circuit

Unit	Register abbreviation	Corresponding register (Name in this manual)
1	EIC00	EIC: External Interrupt Control register
2	EIC10	
3	EIC20	
4	EIC30	

In the following sections, only details of unit 1of the external interrupt circuit are provided.

Details of other units of the external interrupt circuit are the same as those of unit 1.

16.4 Pins of External Interrupt Circuit

This section provides details of the pins of the external interrupt circuit and the block diagrams of such pins.

■ Pins of External Interrupt Circuit

In the MB95410H/470H Series, the pins of the external interrupt circuit are the INT00 to INT07 pins.

● INT00 to INT07 pins

These pins serve both as external interrupt input pins and as general-purpose I/O ports.

INT00 to INT07: If a pin of INT00 to INT07 is set as an input port by the port direction register (DDR) and the corresponding external interrupt input is enabled by the external interrupt control register (EIC), that pin functions as an external interrupt input pin (INT00 to INT07).

The state of a pin can always be read from the port data register (PDR) when that pin is set as an input port. However, the value of PDR is read when the read-modify-write (RMW) type of instruction is used.

■ Block Diagrams of Pins of External Interrupt Circuit

Figure 16.4-1 Block Diagram of INT01 and INT04 of External Interrupt Circuit

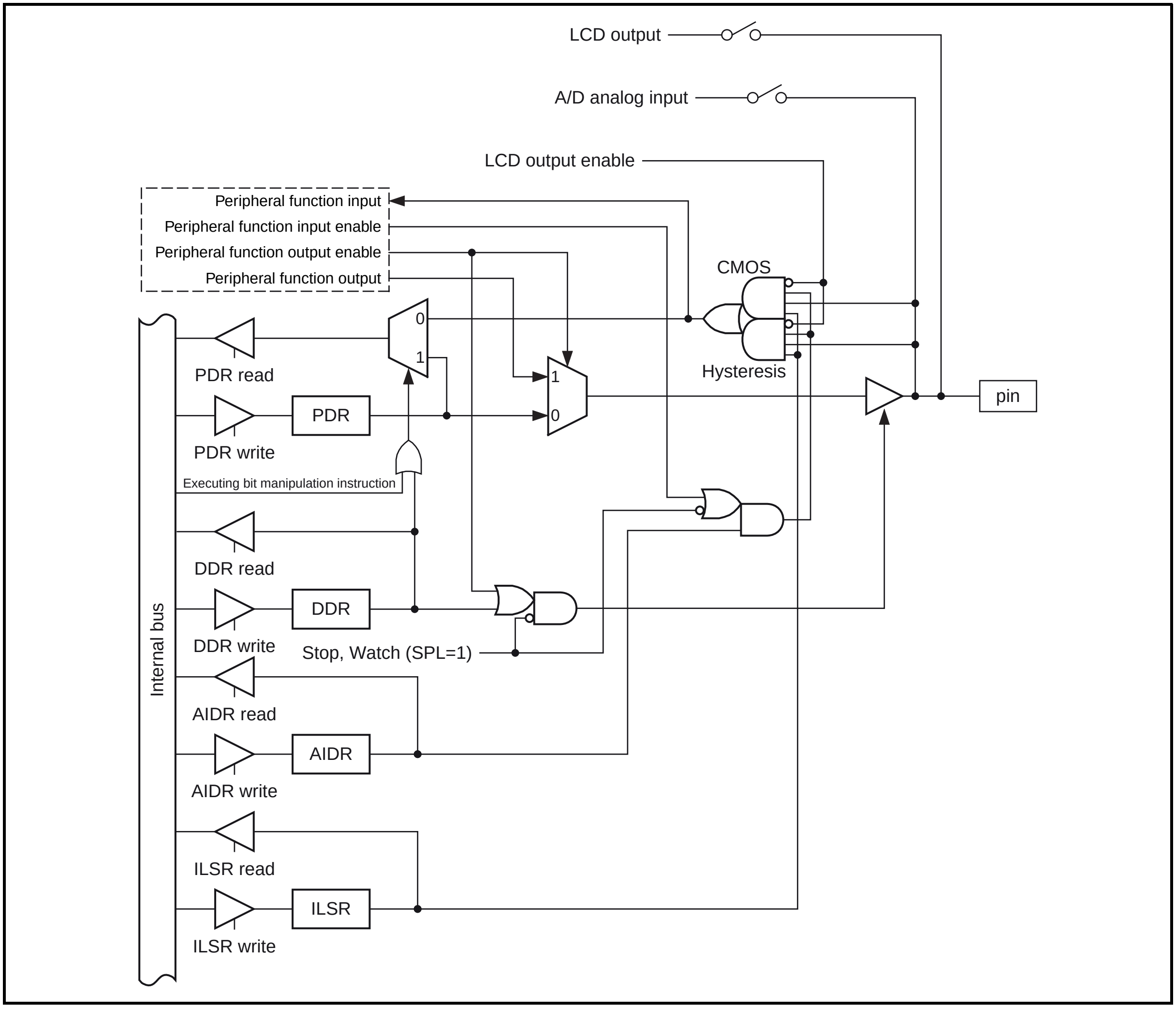
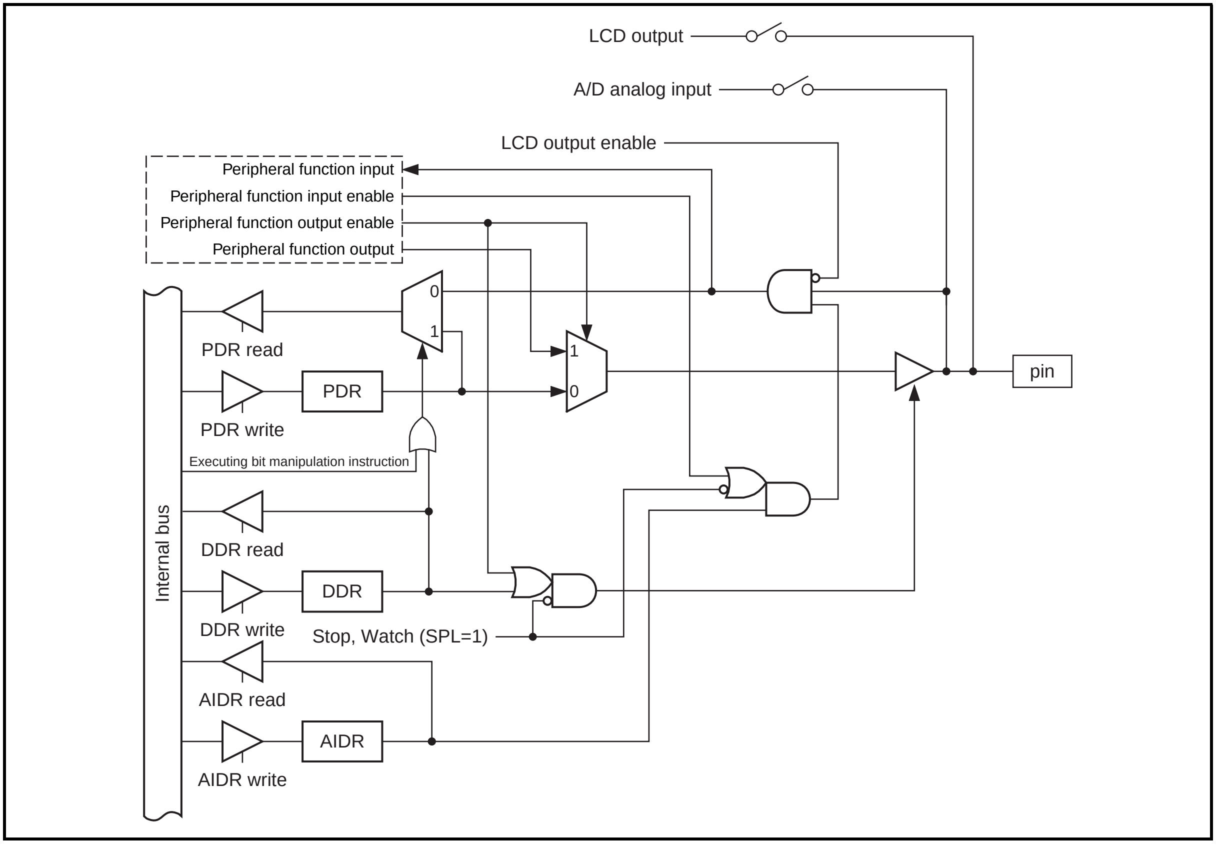


Figure 16.4-2 Block Diagram of Pins INT00, INT02, INT03, INT05, INT06 and INT07 of External Interrupt Circuit


LCD output
A/D analog input
LCD output enable
Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
pin
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
AIDR read
AIDR
AIDR write

16.5 Registers of External Interrupt Circuit

This section describes the registers of the external interrupt circuit.

■ Registers of External Interrupt Circuit

Figure 16.5-1 shows the registers of the external interrupt circuit.

Figure 16.5-1 Registers of External Interrupt Circuit

External interrupt control register (EIC)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
EIC00	0048_H	EIR1	SL11	SL10	EIE1	EIR0	SL01	SL00	EIE0	00000000_B
		R(RM1),W	R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W	
		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
EIC10	0049_H	EIR1	SL11	SL10	EIE1	EIR0	SL01	SL00	EIE0	00000000_B
		R(RM1),W	R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W	
		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
EIC20	$004A_H$	EIR1	SL11	SL10	EIE1	EIR0	SL01	SL00	EIE0	00000000_B
		R(RM1),W	R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W	
		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
EIC30	$004B_H$	EIR1	SL11	SL10	EIE1	EIR0	SL01	SL00	EIE0	00000000_B
		R(RM1),W	R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W	

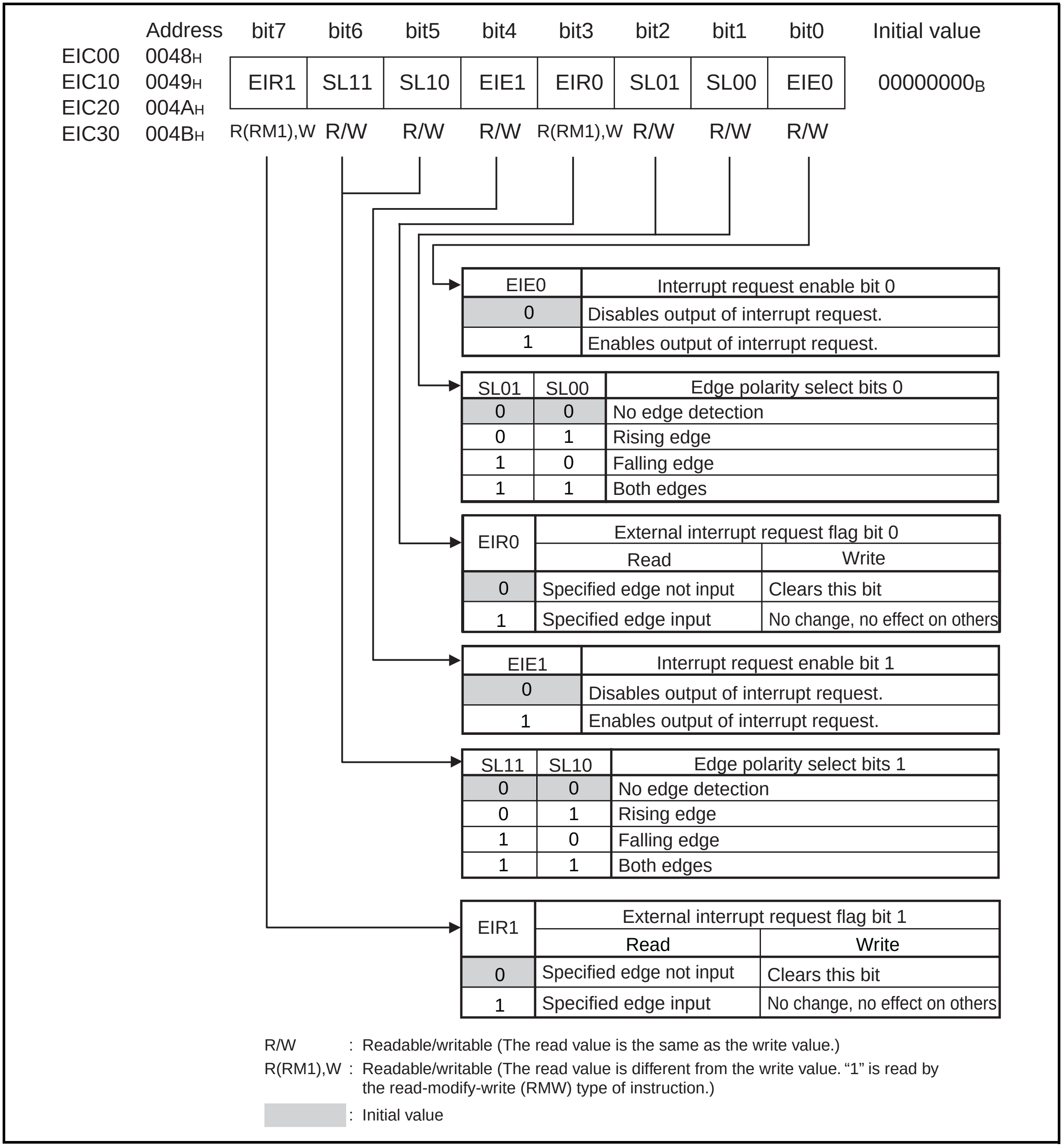
R/W : Readable/writable (The read value is the same as the write value.)
R(RM1), W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)

16.5.1 External Interrupt Control Register (EIC00)

The external interrupt control register (EIC00) is used to select the edge polarity for the external interrupt input and control interrupts.

■ External Interrupt Control Register (EIC00)

Figure 16.5-2 External Interrupt Control Register (EIC00)



	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
EIC00	0048$_H$	EIR1	SL11	SL10	EIE1	EIR0	SL01	SL00	EIE0	00000000$_B$
EIC10	0049$_H$									
EIC20	004A$_H$									
EIC30	004B$_H$	R(RM1),W	R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W	

EIE0	Interrupt request enable bit 0
0	Disables output of interrupt request.
1	Enables output of interrupt request.

SL01	SL00	Edge polarity select bits 0
0	0	No edge detection
0	1	Rising edge
1	0	Falling edge
1	1	Both edges

EIR0	External interrupt request flag bit 0	
	Read	Write
0	Specified edge not input	Clears this bit
1	Specified edge input	No change, no effect on others

EIE1	Interrupt request enable bit 1
0	Disables output of interrupt request.
1	Enables output of interrupt request.

SL11	SL10	Edge polarity select bits 1
0	0	No edge detection
0	1	Rising edge
1	0	Falling edge
1	1	Both edges

EIR1	External interrupt request flag bit 1	
	Read	Write
0	Specified edge not input	Clears this bit
1	Specified edge input	No change, no effect on others

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
▨ : Initial value

Table 16.5-1 Functions of Bits in External Interrupt Control Register (EIC00)

Bit name		Function
bit7	EIR1: External interrupt request flag bit 1	This flag is set to "1" when the edge selected by the edge polarity select bits (SL11, SL10) is input to the external interrupt pin INT01. • When this bit and the interrupt request enable bit 1 (EIE1) are set to "1", an interrupt request is output. • Writing "0" clears the bit. Writing "1" has no effect on operation. • If this bit is read by the read-modify-write (RMW) type of instruction, it returns "1".
bit6, bit5	SL11, SL10: Edge polarity select bits 1	These bits select the polarity of an edge of the pulse input to the external interrupt pin INT01. The edge selected is to be the interrupt source. • If these bits are set to "00_B", edge detection is not performed and no interrupt request is made. • If these bits are set to "01_B", rising edges are to be detected; if "10_B", falling edges are to be detected; if "11_B", both edges are to be detected.
bit4	EIE1: Interrupt request enable bit 1	This bit is used to enable and disable output of interrupt requests to the interrupt controller. When this bit and the external interrupt request flag bit 1 (EIR1) are "1", an interrupt request is output. • When using an external interrupt pin, write "0" to the corresponding bit in the port direction register (DDR) to set the pin as an input port. • The status of the external interrupt pin can be read directly from the port data register, regardless of the status of the interrupt request enable bit.
bit3	EIR0: External interrupt request flag bit 0	This flag is set to "1" when the edge selected by the edge polarity select bits (SL01, SL00) is input to the external interrupt pin INT00. • When this bit and the interrupt request enable bit 0 (EIE0) are set to "1", an interrupt request is output. • Writing "0" clears the bit. Writing "1" has no effect on operation. • If this bit is read by the read-modify-write (RMW) type of instruction, it returns "1".
bit2, bit1	SL01, SL00: Edge polarity select bits 0	These bits select the polarity of an edge of the pulse input to the external interrupt pin INT00. The edge selected is to be the interrupt source. • If these bits are set to "00_B", edge detection is not performed and no interrupt request is made. • If these bits are set to "01_B", rising edges are to be detected; if "10_B", falling edges are to be detected; if "11_B", both edges are to be detected.
bit0	EIE0: Interrupt request enable bit 0	This bit enables or disables the output of interrupt requests to the interrupt controller. An interrupt request is output when this bit and the external interrupt request flag bit 0 (EIR0) are "1". • When using an external interrupt pin, write "0" to the corresponding bit in the port direction register (DDR) to set the pin as an input port. • The status of the external interrupt pin can be read directly from the port data register, regardless of the status of the interrupt request enable bit.

16.6 Interrupts of External Interrupt Circuit

The interrupt sources for the external interrupt circuit include detection of the specified edge of the signal input to an external interrupt pin.

■ Interrupt During Operation of External Interrupt Circuit

When the specified edge of external interrupt input is detected, the corresponding external interrupt request flag bit (EIC: EIR0, EIR1) is set to "1". In this case, if the interrupt request enable bit (EIC: EIE0, EIE1 = 1) corresponding to that external interrupt request flag bit is enabled, an interrupt request is generated to the interrupt controller. In an interrupt service routine, write "0" to the external interrupt request flag bit corresponding to that interrupt request generated to clear the interrupt request.

■ Registers and Vector Table Addresses Related to Interrupts of External Interrupt Circuit

Table 16.6-1 Registers and Vector Table Addresses Related to Interrupts of External Interrupt Circuit

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
External interrupt ch. 0	IRQ00	ILR0	L00	$FFFA_H$	$FFFB_H$
External interrupt ch. 4					
External interrupt ch. 1	IRQ01	ILR1	L01	$FFF8_H$	$FFF9_H$
External interrupt ch. 5					
External interrupt ch. 2	IRQ02	ILR2	L02	$FFF6_H$	$FFF7_H$
External interrupt ch. 6					
External interrupt ch. 3	IRQ03	ILR3	L03	$FFF4_H$	$FFF5_H$
External interrupt ch. 7					

ch.: Channel

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

16.7 Operations of External Interrupt Circuit and Setting Procedure Example

This section describes the operations of the external interrupt circuit.

■ Operations of External Interrupt Circuit

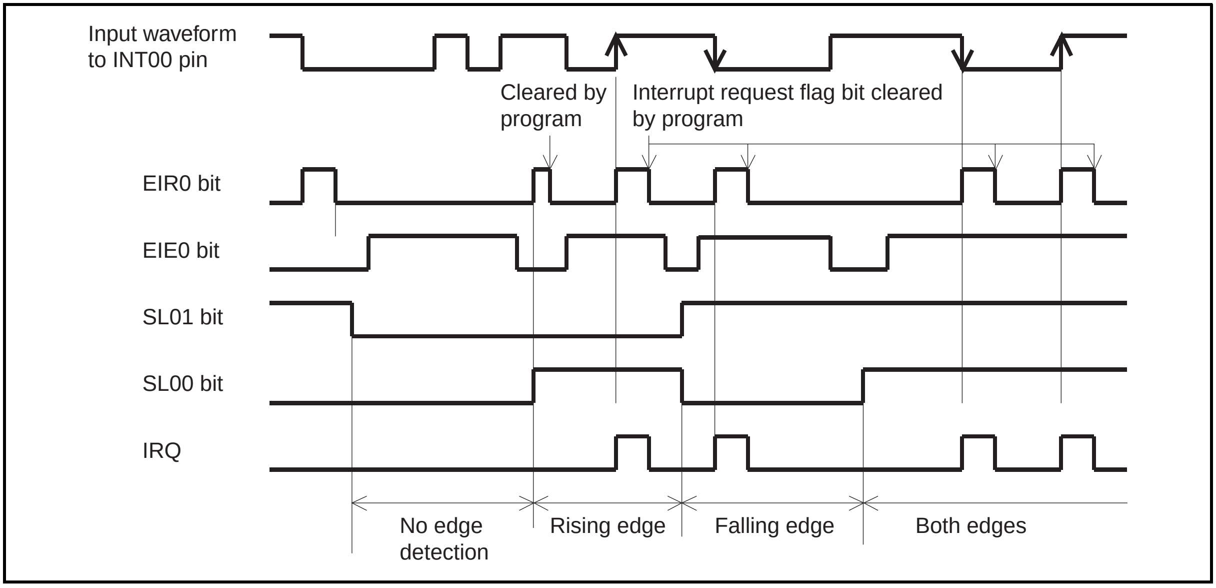
When the polarity of an edge of a signal input from one of the external interrupt pins (INT00, INT01) matches the polarity of the edge selected by the external interrupt control register (EIC:SL00, SL01, SL10, SL11), the corresponding external interrupt request flag bit (EIC:EIR0, EIR1) is set to "1" and the interrupt request is generated.

Always set the interrupt request enable bit to "0" when not using an external interrupt to wake up the device from standby mode.

When setting the edge polarity select bit (SL), set the interrupt request enable bit (EIE) to "0" to prevent the interrupt request from being generated accidentally. Also clear the interrupt request flag bit (EIR) to "0" after changing the edge polarity.

Figure 16.7-1 shows the operations for setting the INT00 pin as an external interrupt input.

Figure 16.7-1 Operations of External Interrupt



■ Setting Procedure Example

Below is an example of procedure for setting the external interrupt circuit.

● Initial settings

1) Set the interrupt level. (ILR0)
2) Select the edge polarity. (EIC:SL01, SL00)
3) Enable interrupt requests. (EIC:EIE0 = 1)

● Interrupt processing

1) Clear the interrupt request flag. (EIC:EIR0 = 0)
2) Process any interrupt.

Note:

An external interrupt input port shares the same pin with an I/O port. Therefore, when using the pin as an external interrupt input port, set the bit in the port direction register (DDR) corresponding to that pin to "0" (input).

16.8 Notes on Using External Interrupt Circuit

This section provides notes on using the external interrupt circuit.

■ Notes on Using External Interrupt Circuit

- Prior to setting the edge polarity select bit (SL), set the interrupt request enable bit (EIE) to "0" (disabling interrupt requests). In addition, clear the external interrupt request flag bit (EIR) to "0" after setting the edge polarity.
- The external interrupt circuit cannot wake up from the interrupt service routine if the external interrupt request flag bit is "1" and the interrupt request enable bit is enabled. In the interrupt service routine, always clear the external interrupt request flag bit.

16.9 Sample Settings for External Interrupt Circuit

This section provides sample settings for the external interrupt circuit.

■ Sample Settings

● Detection levels and setting methods

Four detection levels are available: no edge detection, rising edge, falling edge, both edges

The detection level bits (EIC:SL01, SL00 or EIC:SL11, SL10) are used.

Operating mode	Detection level bits (SL01, SL00 or SL11, SL10)
No edge detection	Set the bits to "00_B"
Detecting rising edges	Set the bits to "01_B"
Detecting falling edges	Set the bits to "10_B"
Detecting both edges	Set the bits to "11_B"

● How to use the external interrupt pin

Set a corresponding bit in the data direction register (DDR0) to "0".

Operation	Direction bit (P00 to P07)	Setting
Using INT00 pin for external interrupt	DDR0: P00	Set the bit to "0".
Using INT01 pin for external interrupt	DDR0: P01	Set the bit to "0".
Using INT02 pin for external interrupt	DDR0: P02	Set the bit to "0".
Using INT03 pin for external interrupt	DDR0: P03	Set the bit to "0".
Using INT04 pin for external interrupt	DDR0: P04	Set the bit to "0".
Using INT05 pin for external interrupt	DDR0: P05	Set the bit to "0".
Using INT06 pin for external interrupt	DDR0: P06	Set the bit to "0".
Using INT07 pin for external interrupt	DDR0: P07	Set the bit to "0".

● Interrupt-related registers

The interrupt level is set by the interrupt level setting registers shown in the following table.

Channel	Interrupt level setting register	Interrupt vector
ch. 0	Interrupt level setting register (ILR0) Address: 00079_H	#0 Address: $0FFFA_H$
ch. 1	Interrupt level setting register (ILR0) Address: 00079_H	#1 Address: $0FFF8_H$
ch. 2	Interrupt level setting register (ILR0) Address: 00079_H	#2 Address: $0FFF6_H$
ch. 3	Interrupt level setting register (ILR0) Address: 00079_H	#3 Address: $0FFF4_H$
ch. 4	Interrupt level setting register (ILR0) Address: 00079_H	#0 Address: $0FFFA_H$
ch. 5	Interrupt level setting register (ILR0) Address: 00079_H	#1 Address: $0FFF8_H$
ch. 6	Interrupt level setting register (ILR0) Address: 00079_H	#2 Address: $0FFF6_H$
ch. 7	Interrupt level setting register (ILR0) Address: 00079_H	#3 Address: $0FFF4_H$

● How to enable/disable/clear interrupt requests

Interrupts requests are enabled/disabled by the interrupt request enable bit (EIC00:EIE0 or EIE1).

Operation	Interrupt request enable bit (EIE0 or EIE1)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

Interrupt requests are cleared by the interrupt request bit (EIC00: EIR0 or EIR1).

Operation	Interrupt request bit (EIR0 or EIR1)
To clear an interrupt request	Set the bit to "0".

CHAPTER 17

INTERRUPT PIN SELECTION CIRCUIT

This chapter describes the functions and operations of the interrupt pin selection circuit.

17.1 Overview of Interrupt Pin Selection Circuit

The interrupt pin selection circuit selects pins to be used as interrupt input pins from among various peripheral input pins.

■ Interrupt Pin Selection Circuit

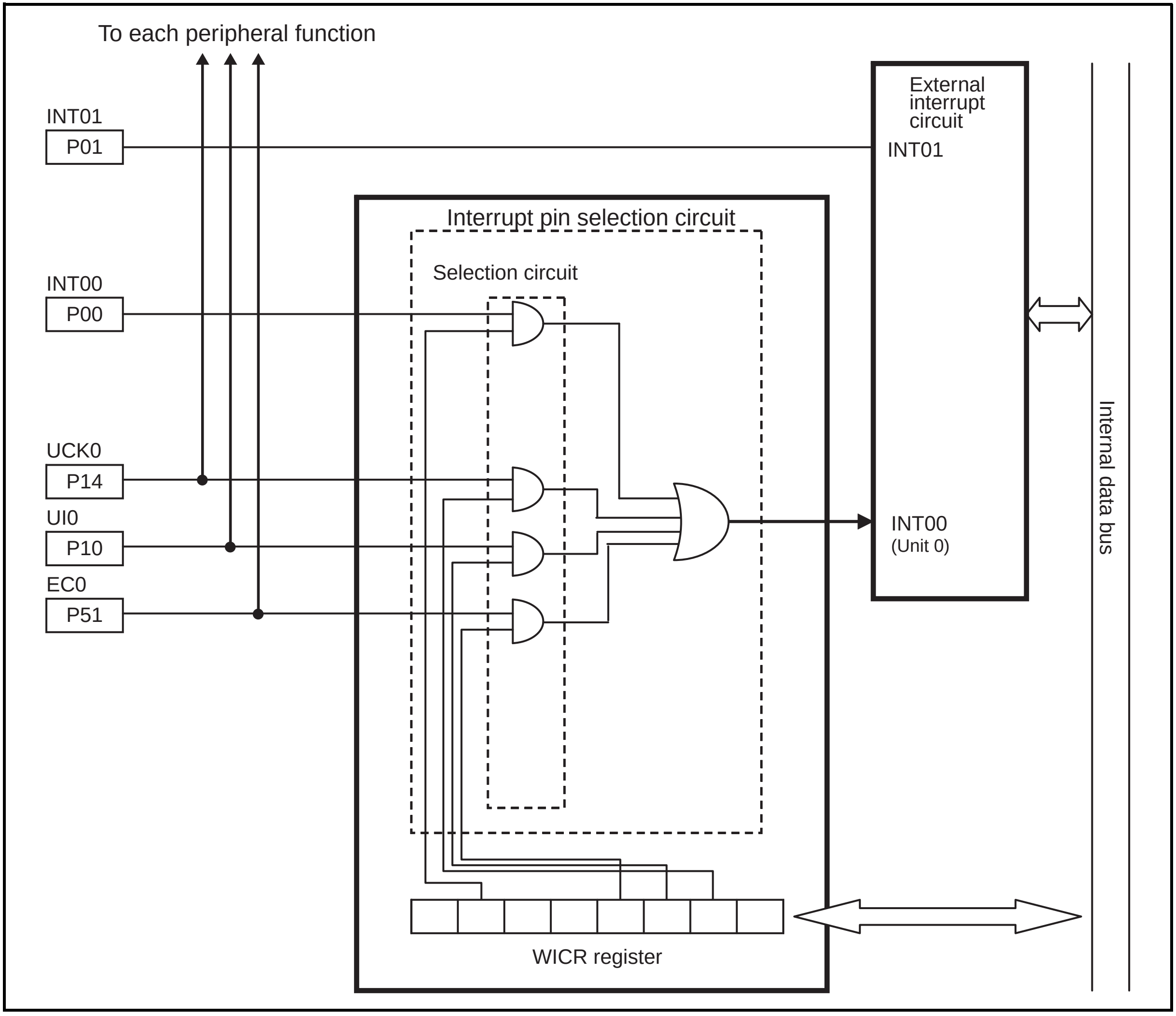
The interrupt pin selection circuit is used to select interrupt input pins from amongst various peripheral inputs (UCK0, UI0, EC0, INT00). The input signal from each peripheral function pin is selected by this circuit and the signal is used as the INT00 (channel 0) input of external interrupt. This enables the input signals to the peripheral function pins to also serve as external interrupt pins.

17.2 Configuration of Interrupt Pin Selection Circuit

Figure 17.2-1 shows the block diagram of the interrupt pin selection circuit.

■ Block Diagram of Interrupt Pin Selection Circuit

Figure 17.2-1 Block Diagram of Interrupt Pin Selection Circuit



- WICR register (interrupt pin selection circuit control register)

  This register is used to determine which of the available peripheral input pins should be outputted to the interrupt circuit and which interrupt pins they should serve as.

- Selection circuit

  This circuit outputs the input from the pin selected by the WICR register to the INT00 input of the external interrupt circuit (channel 0).

17.3 Pins of Interrupt Pin Selection Circuit

This section describes the pins of the interrupt pin selection circuit.

■ Pins of Interrupt Pin Selection Circuit

The peripheral function pins of the interrupt pin selection circuit are the UCK0, UI0, EC0 and INT00 pins. These inputs (except INT00) are also connected to their respective peripheral units in parallel and can be used for both functions simultaneously. Table 17.3-1 shows the correspondence between the peripheral functions and peripheral input pins.

Table 17.3-1 Correspondence between Peripheral Functions and Peripheral Input Pins

Peripheral input pin name	Peripheral functions name
INT00	Interrupt pin selection circuit
UCK0	UART/SIO (clock input/output)
UI0	UART/SIO (data input)
EC0	8/16-bit composite timer (event input)

17.4 Register of Interrupt Pin Selection Circuit

Figure 17.4-1 shows the register of the interrupt pin selection circuit.

■ Register of Interrupt Pin Selection Circuit

Figure 17.4-1 Register of Interrupt Pin Selection Circuit

Interrupt pin selection circuit control register (WICR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FEF_H$	-	INT00	-	-	EC0	UI0	UCK0	-	01000000_B
	R0/WX	R/W	R0/WX	R0/WX	R/W	R/W	R/W	R0/WX	

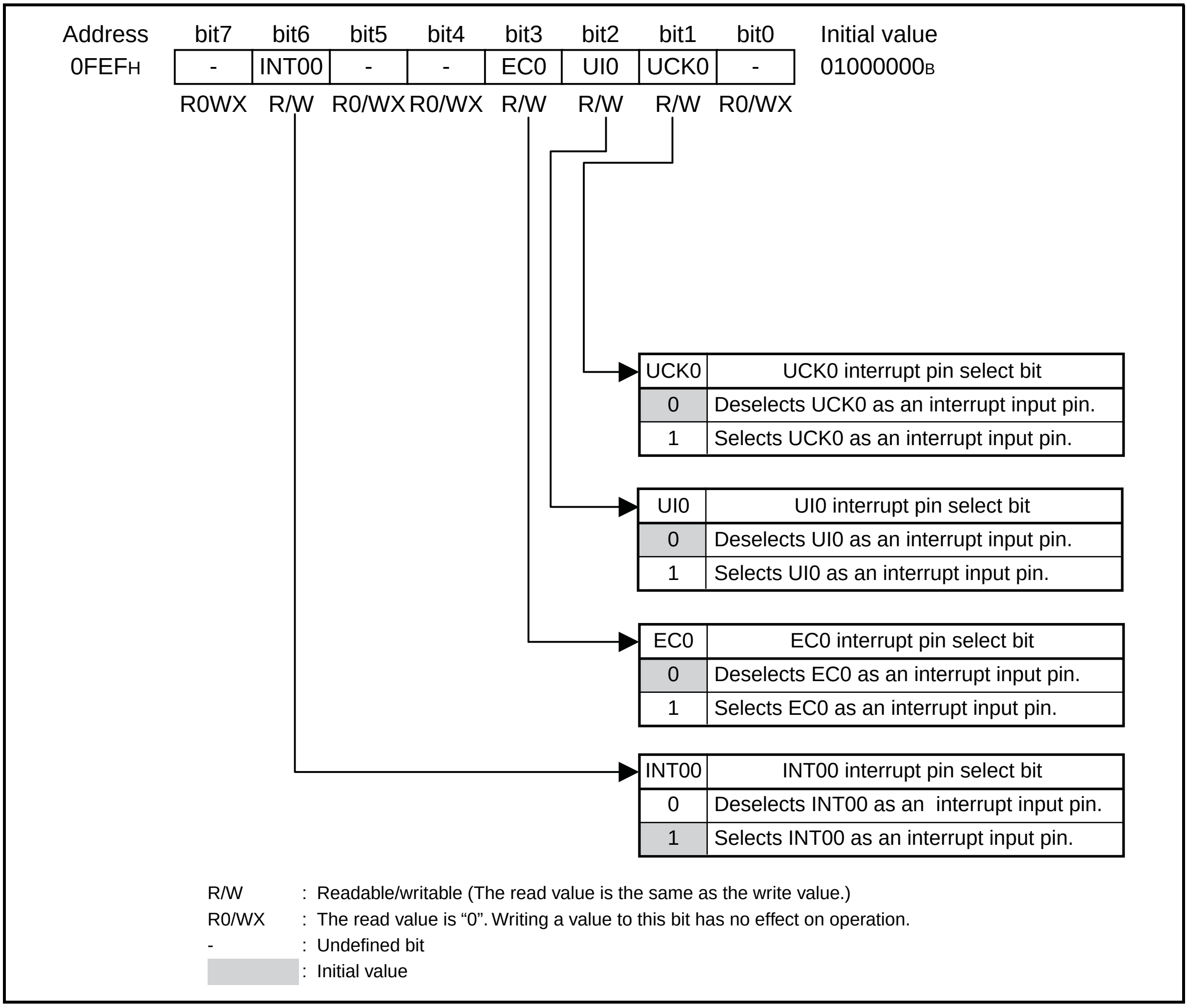
R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

17.4.1 Interrupt Pin Selection Circuit Control Register (WICR)

This register is used to determine which of the available peripheral input pins should be outputted to the interrupt circuit and which interrupt pins they should serve as.

■ Interrupt Pin Selection Circuit Control Register (WICR)

Figure 17.4-2 Interrupt Pin Selection Circuit Control Register (WICR)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FEFH	-	INT00	-	-	EC0	UI0	UCK0	-	01000000B
	R0WX	R/W	R0/WX	R0/WX	R/W	R/W	R/W	R0/WX	

UCK0	UCK0 interrupt pin select bit
0	Deselects UCK0 as an interrupt input pin.
1	Selects UCK0 as an interrupt input pin.

UI0	UI0 interrupt pin select bit
0	Deselects UI0 as an interrupt input pin.
1	Selects UI0 as an interrupt input pin.

EC0	EC0 interrupt pin select bit
0	Deselects EC0 as an interrupt input pin.
1	Selects EC0 as an interrupt input pin.

INT00	INT00 interrupt pin select bit
0	Deselects INT00 as an interrupt input pin.
1	Selects INT00 as an interrupt input pin.

R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
▨ : Initial value

Table 17.4-1 Functions of Bits in Interrupt Pin Selection Circuit Control Register (WICR)

Bit name		Function
bit7	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.
bit6	INT00: INT00 interrupt pin select bit	This bit is used to determine whether to select the INT00 pin as an interrupt input pin. **Writing "0":** Deselects the INT00 pin as an interrupt input pin and the circuit treats the INT00 pin input as being fixed at "0". **Writing "1":** Selects the INT00 pin as an interrupt input pin and the circuit passes the INT00 pin input to INT00 (channel 0) of the external interrupt circuit. In this case, the input signal to the INT00 pin can generate an external interrupt if INT00 (channel 0) operation is enabled in the external interrupt circuit.
bit5, bit4	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit3	EC0: EC0 interrupt pin select bit	This bit is used to determine whether to select the EC0 pin as an interrupt input pin. **Writing "0":** Deselects the EC0 pin as an interrupt input pin and the circuit treats the EC0 pin input as being fixed at "0". **Writing "1":** Selects the EC0 pin as an interrupt input pin and the circuit passes the EC0 pin input to INT00 (channel 0) of the external interrupt circuit. In this case, the input signal to the EC0 pin can generate an external interrupt if INT00 (channel 0) operation is enabled in the external interrupt circuit.
bit2	UI0: UI0 interrupt pin select bit	This bit is used to determine whether to select the UI0 pin as an interrupt input pin. **Writing "0":** Deselects the UI0 pin as an interrupt input pin and the circuit treats the UI0 pin input as being fixed at "0". **Writing "1":** Selects the UI0 pin as an interrupt input pin and the circuit passes the UI0 pin input to INT00 (channel 0) of the external interrupt circuit. In this case, the input signal to the UI0 pin can generate an external interrupt if INT00 (channel 0) operation is enabled in the external interrupt circuit.
bit1	UCK0: UCK0 interrupt pin select bit	This bit is used to determine whether to select the UCK0 pin as an interrupt input pin. **Writing "0":** Deselects the UCK0 pin as an interrupt input pin and the circuit treats the UCK0 pin input as being fixed at "0". **Writing "1":** Selects the UCK0 pin as an interrupt input pin and the circuit passes the UCK0 pin input to INT00 (channel 0) of the external interrupt circuit. In this case, the input signal to the UCK0 pin can generate an external interrupt if INT00 (channel 0) operation is enabled in the external interrupt circuit.
bit0	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.

When these bits are set to "1" and the operation of INT00 (ch. 0) of the external interrupt circuit is enabled in MCU standby mode, the selected pins are enabled to perform input operation. The MCU wakes up from the standby mode when a valid edge pulse is input to the pins. For information about the standby modes, see "6.9 Operations in Low-power Consumption Mode (Standby Mode)".

Note:

The input signals to the peripheral pins do not generate an external interrupt even when "1" is written to these bits if the INT00 (ch. 0) of the external interrupt circuit is disabled.

Do not modify the values of these bits while the INT00 (ch. 0) of the external interrupt circuit is enabled. If modified, the external interrupt circuit may detect a valid edge, depending on the pin input level.

If multiple interrupt pins are selected in the WICR register simultaneously and the operation of INT00 (channel 0) of the external interrupt circuit is enabled (the values other than "00_B" are set to SL01, SL00 bits in EIC00 register of external interrupt circuit.), the selected pins will remain enabled to perform input so as to accept interrupts even in a standby mode.

17.5 Operation of Interrupt Pin Selection Circuit

The interrupt pins are selected by setting WICR (interrupt pin selection circuit control register).

■ Operation of Interrupt Pin Selection Circuit

The WICR (interrupt pin selection circuit control register) setting is used to select the input pins to be input to INT00 of the external interrupt circuit (ch. 0). Shown below is the setup procedure for the interrupt pin selection circuit and external interrupt circuit (channel 0), which must be followed when selecting the UCK0 pin as an interrupt pin.

1) Write "0" to the corresponding bit in the port direction register (DDR) to set the pin as an input.
2) Select the UCK0 pin as an interrupt input pin in WICR (interrupt pin selection circuit control register).
   - Write "02_H" to the WICR register. At this point, after writing "0" in the EIE0 bit of the EIC00 register of the external interrupt circuit, the operation of the external interrupt circuit is disabled.)
3) Enable the operation of INT00 of the external interrupt circuit (ch. 0).
   - Set the SL01 and SL00 bits of the EIC00 register to any value other than "00_B" in the external interrupt circuit to select the valid edge. Also write "1" to the EIE0 bit to enable interrupts).
4) The subsequent interrupt operation is the same as for the external interrupt circuit.
   - When a reset is released, the WICR register is initialized to "40_H" and the INT00 bit is selected as the only available interrupt pin. Update the value of this register before enabling the operation of the external interrupt circuit, when using any pins other than the INT00 pin as external interrupt pins.

17.6 Notes on Using Interrupt Pin Selection Circuit

This section provides notes on using the interrupt pin selection circuit.

- The WICR register is initialized to "40_H" after a reset. This selects the INT00 bit only as an interrupt pin. If using pins other than the INT00 pin as external interrupt pins, update the value of this register before enabling the operation of the external interrupt circuit.
- If multiple interrupt pins are selected in the WICR register simultaneously and the operation of INT00 (ch. 0) of the external interrupt circuit is enabled (Set the SL01 and SL00 bits in the EIC00 register to any value other than "00_B" in the external interrupt circuit to select the valid edge. Also write "1" to the EIE0 bit to enable interrupts.), the selected pins will remain enabled to perform input so as to accept interrupts even in a standby mode.
- If multiple interrupt pins are selected in the WICR register simultaneously, an input to INT00 (ch. 0) of the external interrupt circuit is treated as "H" if a signal input to one of the selected interrupt pins is "H". (It becomes "OR" of the signals input to the selected pins.)

CHAPTER 18

8/16-BIT COMPOSITE TIMER

This chapter describes the functions and operations of the 8/16-bit composite timer.

18.1 Overview of 8/16-bit Composite Timer

The 8/16-bit composite timer consists of two 8-bit counters. It can be used as two 8-bit timers, or as a 16-bit timer if the two counters are connected in cascade.

The 8/16-bit composite timer has the following functions:

- **Interval timer function**
- **PWM timer function**
- **PWC timer function (pulse width measurement)**
- **Input capture function**

■ Interval Timer Function (One-shot Mode)

When the interval timer function (one-shot mode) is selected, the counter starts counting from "00_H" as the timer is started. When the counter value matches the value of the 8/16-bit composite timer 00/01 data register, the timer output is inverted, an interrupt request occurs, and the counter stops counting.

■ Interval Timer Function (Continuous Mode)

When the interval timer function (continuous mode) is selected, the counter starts counting from "00_H" as the timer is started. When the counter value matches the value of the 8/16-bit composite timer 00/01 data register, the timer output is inverted, an interrupt request occurs, and the counter counts from "00_H" again. The timer outputs square wave as a result of this repeated operation.

■ Interval Timer Function (Free-run Mode)

When the interval timer function (free-run mode) is selected, the counter starts counting from "00_H". When the counter value matches the value of the 8/16-bit composite timer 00/01 data register, the timer output is inverted and an interrupt request occurs. Under these conditions, if the counter continues to count and reaches "FF_H", it restarts counting from "00_H". The timer outputs square wave as a result of this repeated operation.

■ PWM Timer Function (Fixed-cycle Mode)

When the PWM timer function (fixed-cycle mode) is selected, a PWM signal with a variable "H" pulse width is generated in fixed cycles. The cycle is fixed to be "FF_H" during 8-bit operation or "$FFFF_H$" during 16-bit operation. The time is determined by the count clock selected. The "H" pulse width is specified by setting a specific register.

■ PWM Timer Function (Variable-cycle Mode)

When the PWM timer function (variable-cycle mode) is selected, two 8-bit counters are used to generate an 8-bit PWM signal of variable cycle and duty depending on the cycle and "L" pulse width specified by registers.

In this operating mode, since the two 8-bit counters have to be used separately, the composite timer cannot operate as a 16-bit counter.

■ PWC Timer Function

When the PWC timer function is selected, the width and cycle of an external input pulse can be measured.

In this operating mode, the counter starts counting from "00_H" immediately after a count start edge of an external input signal is detected. Afterward, when a count end edge is detected, the counter transfers its value to a register to generate an interrupt.

■ Input Capture Function

When the input capture function is selected, the counter value is stored in a register immediately after the detection of an edge of an external input signal.

This function is available in either free-run mode or clear mode for count operation.

In clear mode, the counter starts counting from "00_H", and transfers its value to a register to generate an interrupt after an edge is detected. Afterward, the counter restarts counting from "00_H".

In free-run mode, the counter transfers its value to a register to generate an interrupt immediately after the detection of an edge. Afterward, unlike in clear mode, the counter continues to count without being cleared to "00_H".

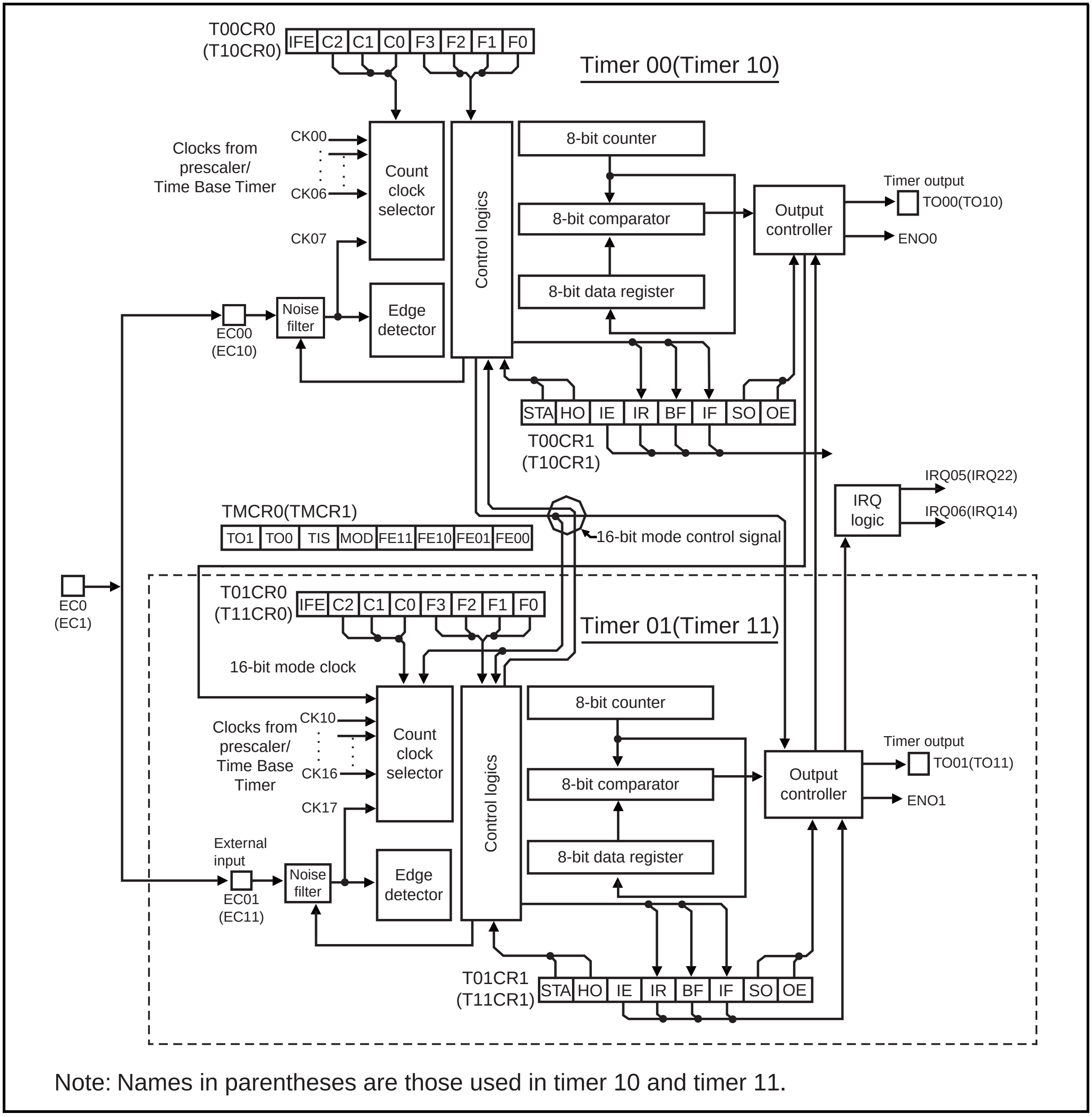
18.2 Configuration of 8/16-bit Composite Timer

The 8/16-bit composite timer consists of the following blocks:

- **8-bit counter × 2 channels**
- **8-bit comparator (including a temporary latch) × 2 channels**
- **8/16-bit composite timer 00/01 data register × 2 channels (T00DR/T01DR), (T10DR/T11DR)**
- **8/16-bit composite timer 00/01 status control register 0 × 2 channels (T00CR0/T01CR0), (T10CR0/T11CR0)**
- **8/16-bit composite timer 00/01 status control register 1 × 2 channels (T00CR1/T01CR1), (T10CR1/T11CR1)**
- **8/16-bit composite timer 00/01 timer mode control register (TMCR0), (TMCR1)**
- **Output controller × 2 channels**
- **Control logic × 2 channels**
- **Count clock selector × 2 channels**
- **Edge detector × 2 channels**
- **Noise filter × 2 channels**

■ Block Diagram of 8/16-bit Composite Timer

Figure 18.2-1 Block Diagram of 8/16-bit Composite Timer



Note: Names in parentheses are those used in timer 10 and timer 11.

- 8-bit counter

This counter serves as the basis for various timer operations. It can be used either as two 8-bit counters or as a 16-bit counter.

- 8-bit comparator

The comparator compares the value in the 8/16-bit composite timer 00/01 data register and that in the counter. It incorporates a latch that temporarily stores the 8/16-bit composite timer 00/01 data register value.

● 8/16-bit composite timer 00/01 data register (T00DR/T01DR) [8/16-bit composite timer 10/11 data register (T10DR/T11DR)]

This register is used to write the maximum value counted during interval timer operation or PWM timer operation and to read the count value during PWC timer operation or input capture operation.

● 8/16-bit composite timer 00/01 status control registers 0 (T00CR0/T01CR0) [8/16-bit composite timer 10/11 status control registers 0 (T10CR0/T11CR0)]

These registers are used to select the timer operating mode and the count clock, and to enable or disable IF flag interrupts.

● 8/16-bit composite timer 00/01 status control registers 1 (T00CR1/T01CR1) [8/16-bit composite timer 10/11 status control registers 1 (T10CR1/T11CR1)]

These registers are used to control interrupt flags, timer output, and timer operation.

● 8/16-bit composite timer 00/01 timer mode control register (TMCR0) [8/16-bit composite timer 10/11 timer mode control register (TMCR1)]

This register is used to select the noise filter function, 8-bit or 16-bit operating mode, and signal input to timer 00 and to indicate the timer output value.

● Output controller

The output controller controls timer output. The timer output is supplied to the external pin when the pin output has been enabled.

● Control logic

The control logic controls timer operation.

● Count clock selector

The selector selects the counter operating clock signal from different prescaler output signals (divided machine clock signal and time-base timer output signal).

● Edge detector

The edge detector selects the edge of an external input signal to be used as an event for PWC timer operation or input capture operation.

● Noise filter

This filter serves as a noise filter for external input signals. The filter function can be selected from "H" pulse noise elimination, "L" pulse noise elimination, and "H"/"L"-pulse noise elimination.

■ Input Clock

The 8/16-bit composite timer uses the output clock from the prescaler as its input clock (count clock).

18.3 Channels of 8/16-bit Composite Timer

This section describes the channels of the 8/16-bit composite timer.

■ Channels of 8/16-bit Composite Timer

The MB95410H/470H Series has two channels of 8/16-bit composite timer.

In a channel, there are two 8-bit counters. They can be used as two 8-bit timers or one 16-bit timer. The following table lists the external pins and registers corresponding to each channel.

Table 18.3-1 8/16-bit Composite Timer Channels and Corresponding External Pins

Channel	Pin name	Pin function
0	TO00	Timer 00 output
	TO01	Timer 01 output
	EC0	Timer 00 input and timer 01 input
1	TO10	Timer 10 output
	TO11	Timer 11 output
	EC1	Timer 10 input and timer 11 input

Table 18.3-2 8/16-bit Composite Timer Channels and Corresponding Registers

Channel	Register abbreviation	Corresponding register (Name in this manual)
0	T00CR0	Timer 00 status control register 0
	T01CR0	Timer 01 status control register 0
	T00CR1	Timer 00 status control register 1
	T01CR1	Timer 01 status control register 1
	T00DR	Timer 00 data register
	T01DR	Timer 01 data register
	TMCR0	Timer 00/01 timer mode control register
1	T10CR0	Timer 10 status control register 0
	T11CR0	Timer 11 status control register 0
	T10CR1	Timer 10 status control register 1
	T11CR1	Timer 11 status control register 1
	T10DR	Timer 10 data register
	T11DR	Timer 11 data register
	TMCR1	Timer 10/11 timer mode control register

In the following sections in this chapter, only details of channel 0 of the 8/16-bit composite timer are provided.

Channel 0 and channel 1 have identical configuration. The 2-digit number in a pin name and a register abbreviation corresponds to channel and timer. The upper number corresponds to channel and the lower number corresponds to timer.

18.4 Pins of 8/16-bit Composite Timer

This section describes the pins of the 8/16-bit composite timer.

■ Pins of 8/16-bit Composite Timer

The external pins of the 8/16-bit composite timer are TO00, TO01, TO10, TO11, EC0 and EC1.

● TO00 pin

TO00:

This pin serves as the timer output pin for timer 00 in 8-bit operation or for timers 00 and 01 in 16-bit operation. When the output is enabled (T00CR1:OE = 1) in the interval timer function, PWM timer function, or PWC timer function, this pin becomes an output pin automatically regardless of the setting of the port direction register (DDR5:bit2 in the MB95410H Series, DDR0:bit1 in the MB95470H Series) and functions as the timer output TO00 pin.

The output becomes undetermined if output is enabled with the input capture function in use.

● TO01 pin

TO01:

This pin serves as the timer output pin for timer 01 in 8-bit operation. When the output is enabled (T01CR1:OE = 1) in interval timer function, PWM timer function (fixed-cycle mode), the pin becomes an output pin automatically regardless of the setting of the port direction register (DDR5:bit0 in the MB95410H Series, DDR1:bit3 in the MB95470H Series) and functions as the timer output TO01 pin.

In 16-bit operation, if output is enabled with the PWM timer function (variable-cycle mode) or input capture function in use, the output becomes undetermined.

● EC0 pin

The EC0 pin is connected to the EC00 and EC01 internal pins.

EC00 internal pin:

This pin serves as the external count clock input pin for timer 00 when the interval timer function or PWM timer function is selected, or as the signal input pin for timer 00 when the PWC timer function or input capture function is selected. The pin cannot be set to serve as the external count clock input pin when the PWC timer function or input capture function is selected.

To use the input function mentioned above, set the bit in the port direction register corresponding to EC0 pin to "0" to make the pin as an input port.

EC01 internal pin:

This pin serves as the external count clock input pin for timer 01 when the interval timer function or PWM timer function is selected, or as the signal input pin for timer 01 when the PWC timer function or input capture function is selected. The pin cannot be set to serve as the external count clock input pin when the PWC timer function or input capture function is selected.

In 16-bit operation, the input function of this pin is not used. If the PWM timer function (variable-cycle mode) is selected, the input function of this pin can also be used.

To use the input function mentioned above, set the bit in the port direction register corresponding to EC0 pin to "0" to make the pin as an input port.

● TO10 pin

TO10:

This pin serves as the timer output pin for timer 10 in 8-bit operation or for timers 10 and 11 in 16-bit operation. When the output is enabled (T10CR1:OE = 1) in the interval timer function, PWM timer function, or PWC timer function, this pin becomes an output pin automatically regardless of the setting of the port direction register (DDRE:bit6) and functions as the timer output TO10 pin.

The output becomes undetermined if output is enabled with the input capture function in use.

● TO11 pin

TO11:

This pin serves as the timer output pin for timer 11 in 8-bit operation. When the output is enabled (T11CR1:OE = 1) in interval timer function, PWM timer function (fixed-cycle mode), or PWC timer function, the pin becomes an output pin automatically regardless of the setting of the port direction register (DDRE:bit5) and functions as the timer output TO11 pin.

In 16-bit operation, if output is enabled with the PWM timer function (variable-cycle mode) or input capture function in use, the output becomes undetermined.

● EC1 pin

The EC1 pin is connected to the EC10 and EC11 internal pins.

EC10 internal pin:

This pin serves as the external count clock input pin for timer 10 when the interval timer function or PWM timer function is selected, or as the signal input pin for timer 10 when the PWC timer function or input capture function is selected. The pin cannot be set to serve as the external count clock input pin when the PWC timer function or input capture function is selected.

To use the input function mentioned above, set the bit in the port direction register corresponding to EC1 pin to "0" to make the pin as an input port.

EC11 internal pin:

This pin serves as the external count clock input pin for timer 11 when the interval timer function or PWM timer function is selected, or as the signal input pin for timer 11 when the PWC timer function or input capture function is selected. The pin cannot be set to serve as the external count clock input pin when the PWC timer function or input capture function is selected.

In 16-bit operation, the input function of this pin is not used. If the PWM timer function (variable-cycle mode) is selected, the input function of this pin can also be used.

To use the input function mentioned above, set the bit in the port direction register corresponding to EC1 pin to "0" to make the pin as an input port.

■ Block Diagrams of Pins of 8/16-bit Composite Timer (MB95410H Series)

Figure 18.4-1 Block Diagram of EC0 of 8/16-bit Composite Timer

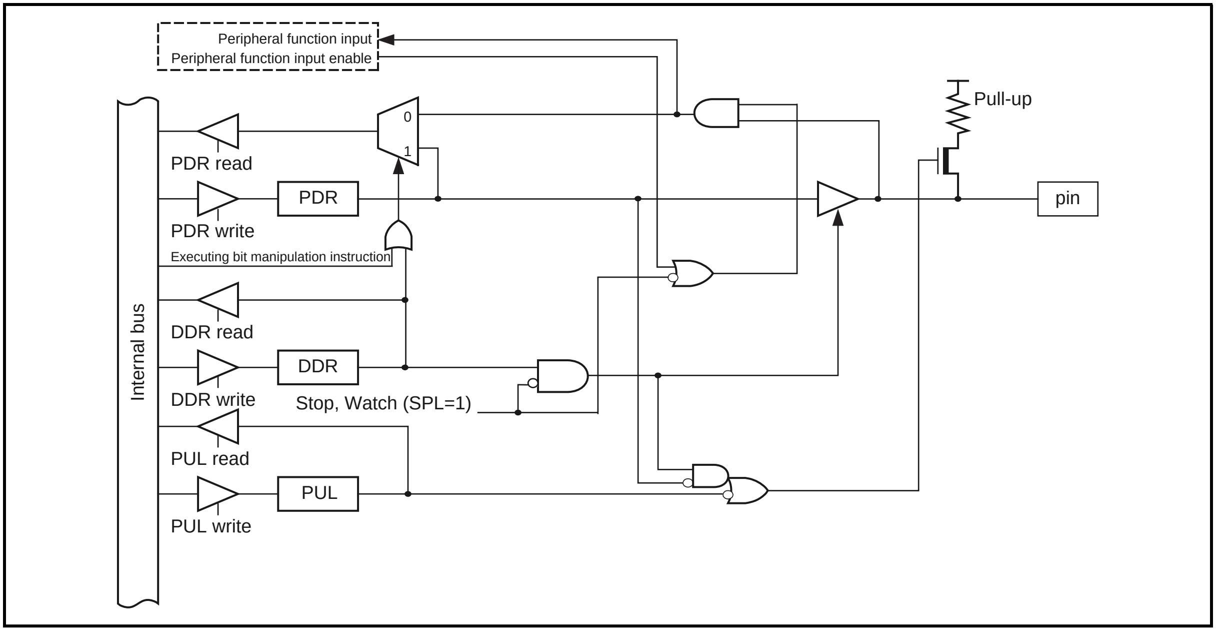


Figure 18.4-2 Block Diagram of TO00 and TO01 of 8/16-bit Composite Timer

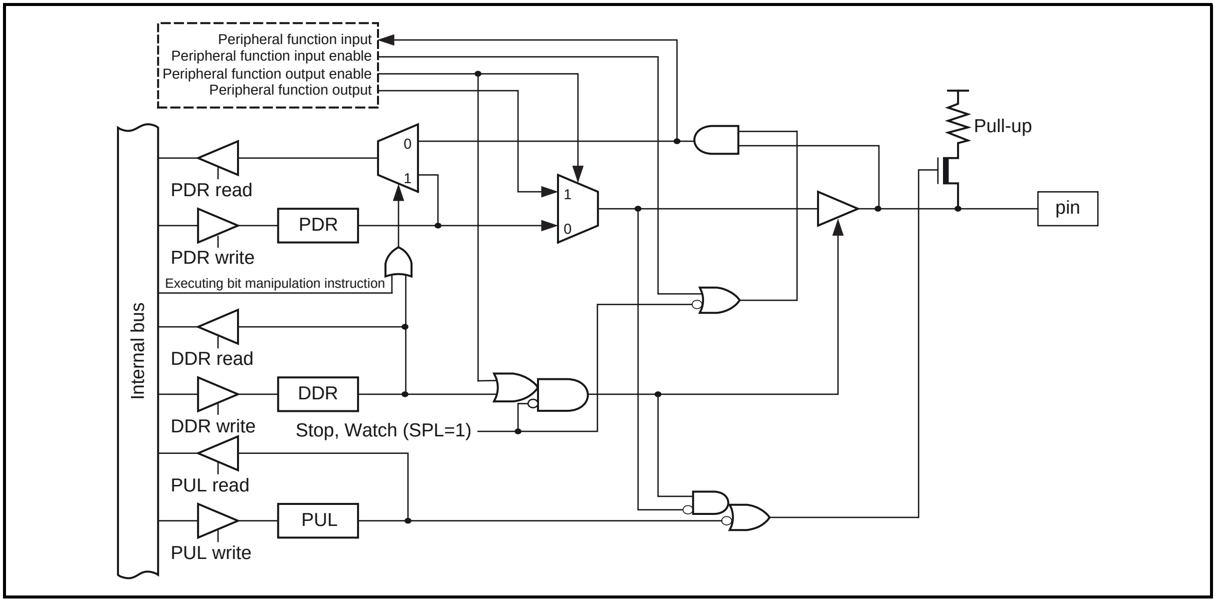

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
0
1
1
0
PDR read
PDR
PDR write
pin
Executing bit manipulation instruction
DDR read
Internal bus
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write

Figure 18.4-3 Block Diagram of EC1 of 8/16-bit Composite Timer

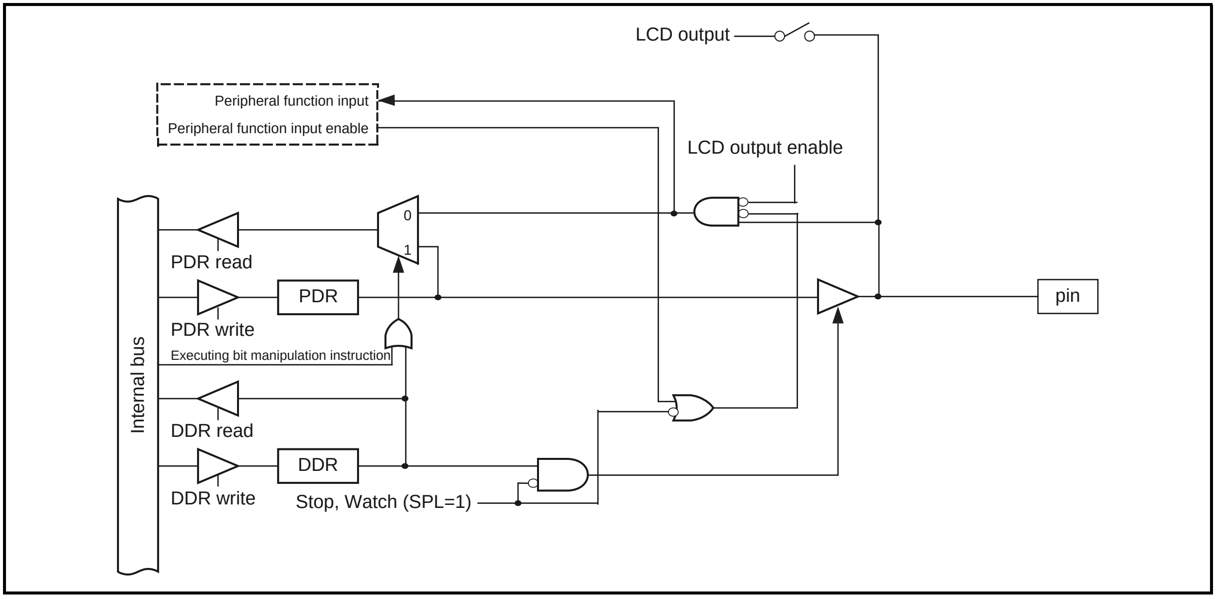
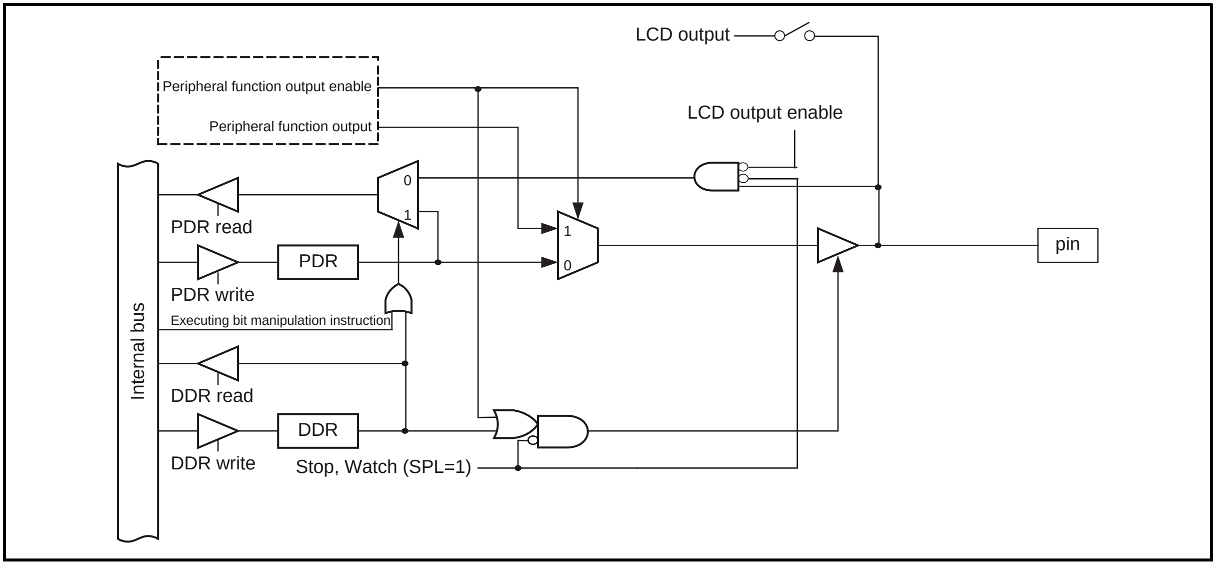


Figure 18.4-4 Block Diagram of TO10 and TO11 of 8/16-bit Composite Timer



■ Block Diagrams of Pins of 8/16-bit Composite Timer (MB95470H Series)

Figure 18.4-5 Block Diagram of EC0 of 8/16-bit Composite Timer

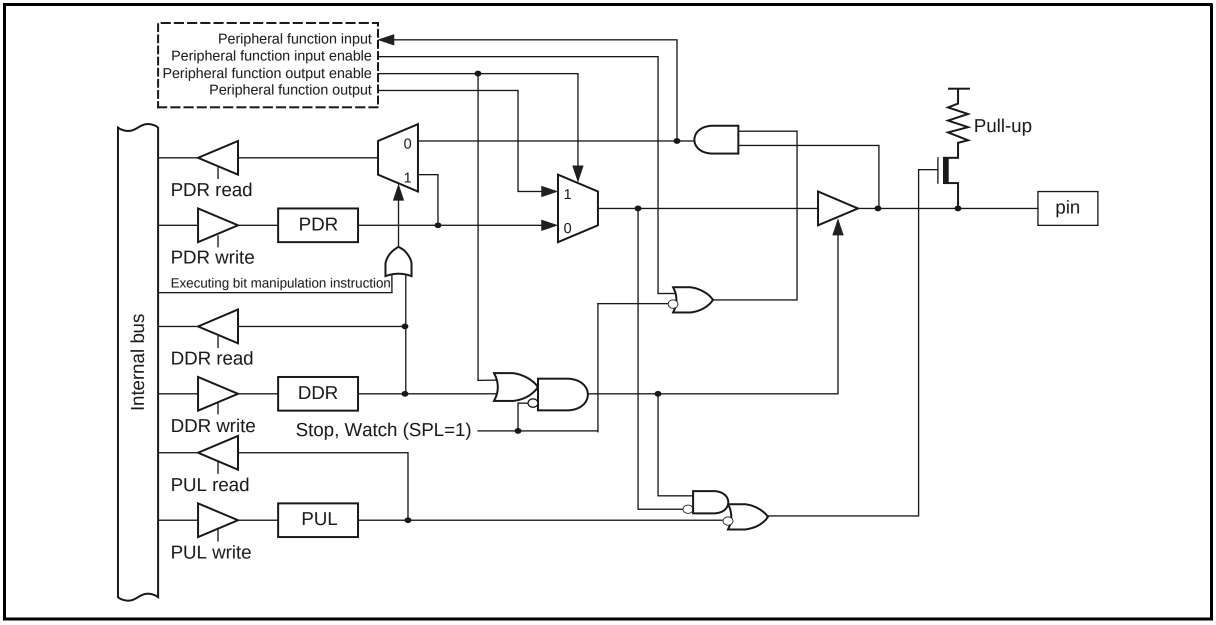

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write

Figure 18.4-6 Block Diagram of TO00 of 8/16-bit Composite Timer

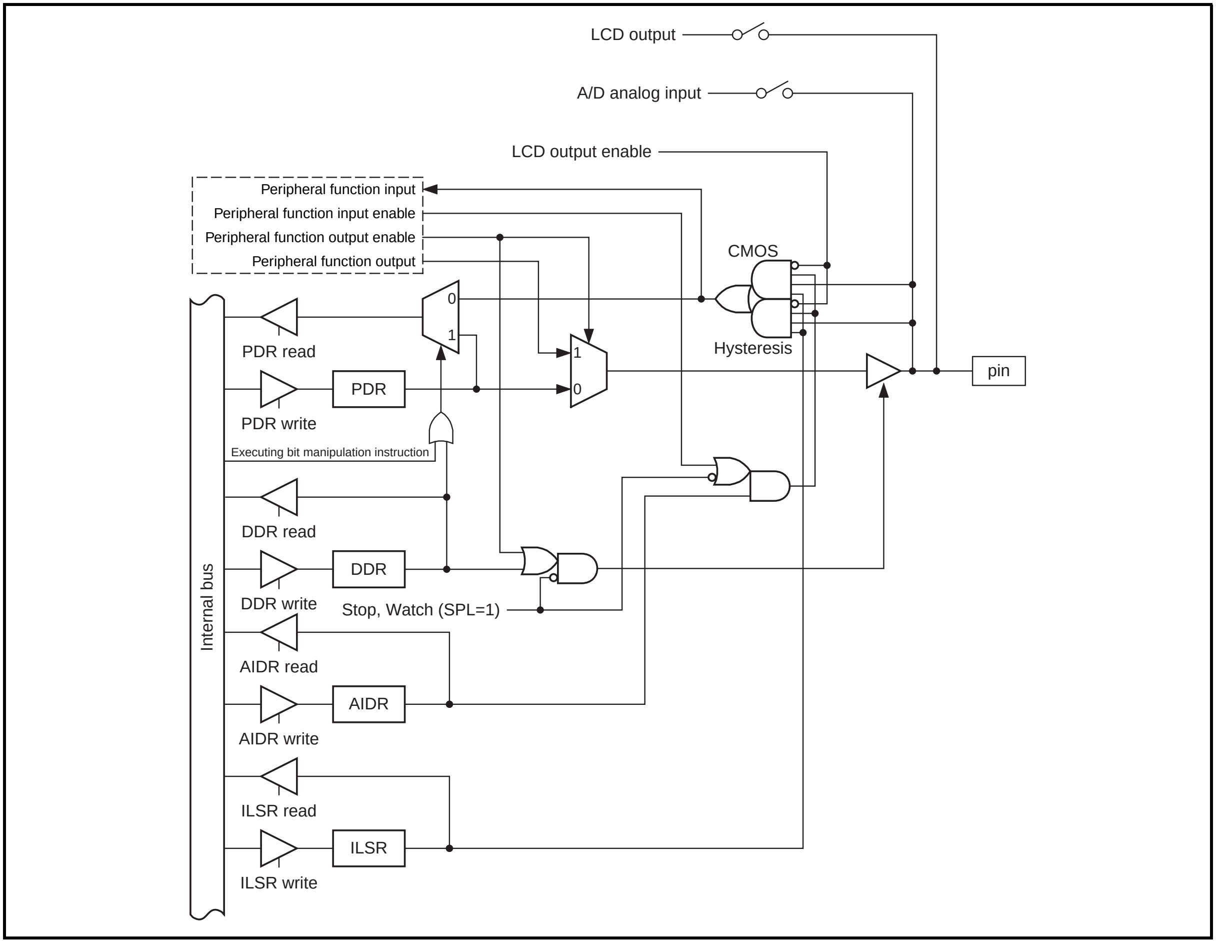


Figure 18.4-7 Block Diagram of TO01 of 8/16-bit Composite Timer

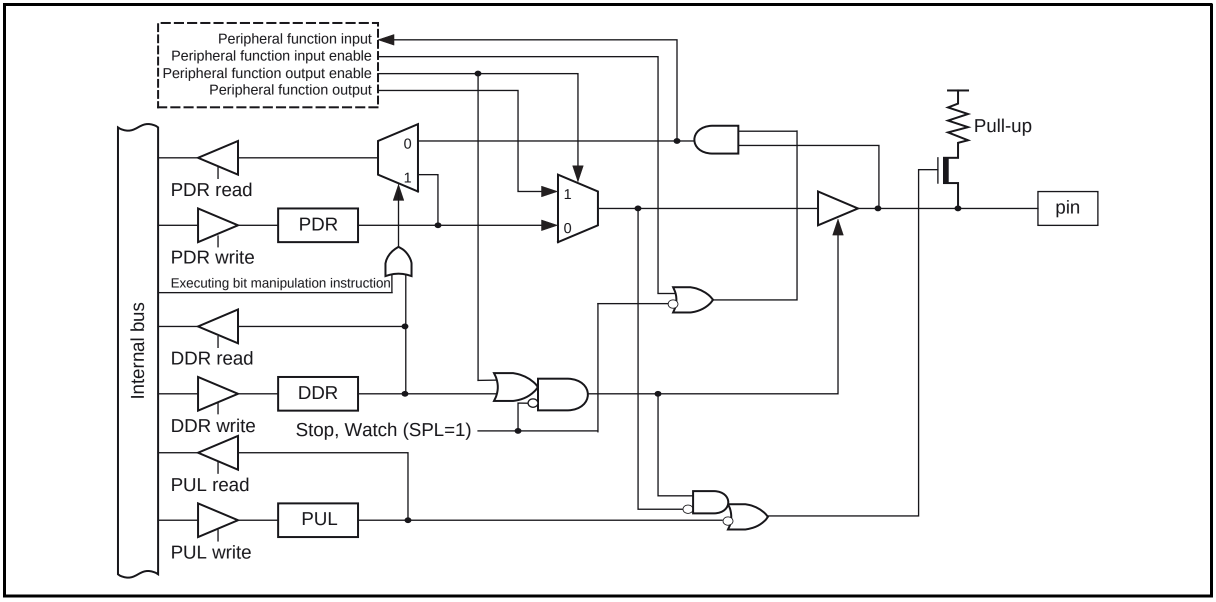

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write

Figure 18.4-8 Block Diagram of EC1 of 8/16-bit Composite Timer

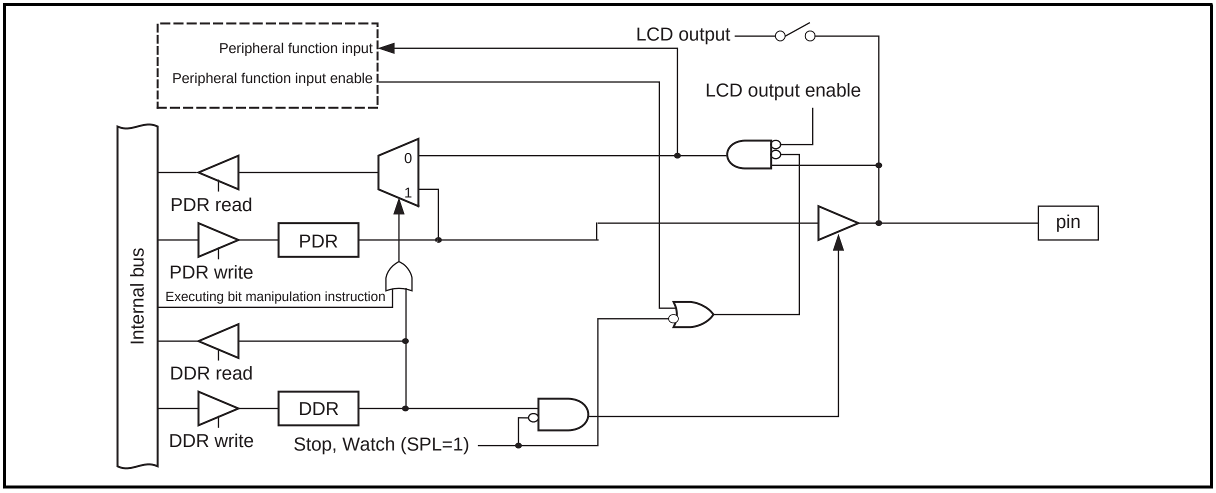
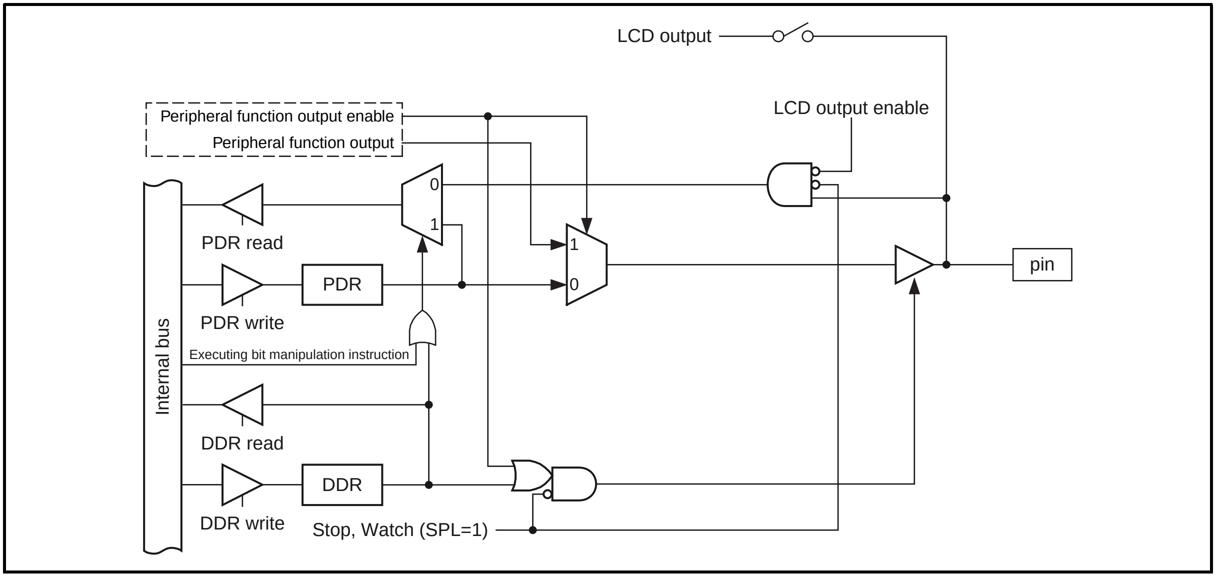


Figure 18.4-9 Block Diagram of Pins TO10 and TO11 of 8/16-bit Composite Timer



18.5 Registers of 8/16-bit Composite Timer

This section describes the registers of the 8/16-bit composite timer.

■ Registers of 8/16-bit Composite Timer 0

Figure 18.5-1 Registers of 8/16-bit Composite Timer 0

8/16-bit composite timer 00/01 status control register 0 (T00CR0/T01CR0)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
T01CR0	0F92_H	IFE	C2	C1	C0	F3	F2	F1	F0	00000000_B
T00CR0	0F93_H	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/16-bit composite timer 00/01 status control register 1 (T00CR1/T01CR1)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
T01CR1	0036_H	STA	HO	IE	IR	BF	IF	SO	OE	00000000_B
T00CR1	0037_H	R/W	R/W	R/W	R(RM1),W	R/WX	R(RM1),W	R/W	R/W	

8/16-bit composite timer 00/01 data register (T00DR/T01DR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
T01DR	0F94_H	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	00000000_B
T00DR	0F95_H	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W	

8/16-bit composite timer 00/01 timer mode control register (TMCR0)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
	0F96_H	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00	00000000_B
		R/WX	R/WX	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1), W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
R,W : Readable/writable (The read value is different from the write value.)

■ Registers of 8/16-bit Composite Timer 1

Figure 18.5-2 Registers of 8/16-bit Composite Timer 1

8/16-bit composite timer 10/11 status control register 0 (T10CR0/T11CR0)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
T11CR0	$0F97_H$	IFE	C2	C1	C0	F3	F2	F1	F0	00000000_B
T10CR0	$0F98_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/16-bit composite timer 10/11 status control register 1 (T10CR1/T11CR1)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
T11CR1	0038_H	STA	HO	IE	IR	BF	IF	SO	OE	00000000_B
T10CR1	0039_H	R/W	R/W	R/W	R(RM1),W	R/WX	R(RM1),W	R/W	R/W	

8/16-bit composite timer 10/11 data register (T10DR/T11DR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
T11DR	$0F99_H$	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	00000000_B
T10DR	$0F9A_H$	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W	

8/16-bit composite timer 10/11 timer mode control register (TMCR1)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
	$0F9B_H$	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00	00000000_B
		R/WX	R/WX	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
R,W : Readable/writable (The read value is different from the write value.)

18.5.1 8/16-bit Composite Timer 00/01 Status Control Register 0 (T00CR0/T01CR0)

The 8/16-bit composite timer 00/01 status control register 0 (T00CR0/T01CR0) selects the timer operation mode, selects the count clock, and enables or disables IF flag interrupts. The T00CR0 and T01CR0 registers correspond to timers 00 and 01 respectively.

■ 8/16-bit Composite Timer 00/01 Status Control Register 0 (T00CR0/T01CR0)

Figure 18.5-3 8/16-bit Composite Timer 00/01 Status Control Register 0 (T00CR0/T01CR0)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
T01CR0 T00CR0	$0F92_H$ $0F93_H$	IFE	C2	C1	C0	F3	F2	F1	F0	00000000_B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

F3	F2	F1	F0	Timer operating mode select bits
0	0	0	0	Interval timer (one-shot mode)
0	0	0	1	Interval timer (continuous mode)
0	0	1	0	Interval timer (free-run mode)
0	0	1	1	PWM timer (fixed-cycle mode)
0	1	0	0	PWM timer (variable-cycle mode)
0	1	0	1	PWC timer ("H" pulse = rising to falling)
0	1	1	0	PWC timer ("L" pulse = falling to rising)
0	1	1	1	PWC timer (cycle = rising to rising)
1	0	0	0	PWC timer (cycle = falling to falling)
1	0	0	1	PWC timer ("H" pulse = rising to falling; Cycle = rising to rising)
1	0	1	0	Input capture (rising, free-run counter)
1	0	1	1	Input capture (falling, free-run counter)
1	1	0	0	Input capture (both edges, free-run counter)
1	1	0	1	Input capture (rising, counter clear)
1	1	1	0	Input capture (falling, counter clear)
1	1	1	1	Input capture (both edges, counter clear)

C2	C1	C0	Count clock select bits
0	0	0	1 × MCLK (machine clock)
0	0	1	1/2 × MCLK (machine clock)
0	1	0	1/4 × MCLK (machine clock)
0	1	1	1/8 × MCLK (machine clock)
1	0	0	1/16 × MCLK (machine clock)
1	0	1	1/32 × MCLK (machine clock)
1	1	0	1/128 × F_{CH} or 1/64 × F_{CRH}*
1	1	1	External clock

IFE	IF flag interrupt enable bit
0	Disables the IF flag interrupt
1	Enables the IF flag interrupt

R/W : Readable/writable (The read value is the same as the write value.)

▨ : Initial value

* : The value to be used as the count clock is decided according to the settings of the SYCC2 register.

Table 18.5-1 Functions of Bits in 8/16-bit Composite Timer 00/01 Status Control Register 0 (T00CR0/T01CR0) (1 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7</td><td>IFE:
IF flag interrupt enable bit</td><td>This bit enables or disables IF flag interrupts.
During timer operation (T00CR1/T01CR1:STA = 1), the write access to this bit has no effect on operation. Ensure that the timer has stopped before modifying this bit.
<b>Writing "0":</b> Disables IF flag interrupts.
<b>Writing "1":</b> An IF flag interrupt request is output when both the IE bit (T00CR1/T01CR1:IE) and the IF flag (T00CR1/T01CR1:IF) are set to "1".</td></tr>
<tr><td>bit6
to
bit4</td><td>C2, C1, C0:
Count clock select bits</td><td>These bits select the count clock.
• The count clock is generated by the prescaler. See "6.13 Operation of Prescaler".
• Write access to these bits is nullified in timer operation (T00CR1/T01CR1:STA = 1).
• The clock selection of T01CR0 (timer 01) is nullified in 16-bit operation.
• These bits cannot be set to "111_B" when the PWC function or input capture function is used. An attempt to write "111_B" with the PWC function or input capture function in use resets the bits to "000_B". The bits are also reset to "000_B" if the timer enters the input capture operation mode with the bits set to "111_B".
• When these bits are set to "110_B", the count clock from the time-base timer will be used as the count clock. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock. In the case of using the count clock from the time-base timer as the count clock, resetting the time-base timer by writing "1" to the time-base timer initialization bit in the time-base timer control register (TBTC:TCLR) will affect the count time.
<table>
<tr><th>C2</th><th>C1</th><th>C0</th><th>Count clock</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1 × MCLK (machine clock)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1/2 × MCLK (machine clock)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1/4 × MCLK (machine clock)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1/8 × MCLK (machine clock)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1/16 × MCLK (machine clock)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1/32 × MCLK (machine clock)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1/128 × F_{CH} or 1/64 × F_{CRH}</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>External clock</td></tr>
</table></td></tr>
</table>

Table 18.5-1 Functions of Bits in 8/16-bit Composite Timer 00/01 Status Control Register 0 (T00CR0/T01CR0) (2 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit3
to
bit0</td><td>F3, F2, F1, F0:
Timer operating mode select bits</td><td>These bits select the timer operating mode.
• The PWM timer function (variable-cycle mode; F3, F2, F1, F0 = 0100_B) is set by either the T00CR0 (timer 00) register or T01CR0 (timer 01) register. If one of the timers starts operating (T00CR1/T01CR1: STA= 1), the F3, F2, F1 and F0 bits of the other timer are automatically set to "0100_B".
• With the 16-bit operation having been selected (TMCR0:MOD = 1), if the composite timer starts operating using the PWM timer function (variable-cycle mode) (T00CR1/T01CR1:STA = 1), the MOD bit is set to "0" automatically.
• Write access to these bits is nullified in timer operation (T00CR1/T01CR1:STA = 1).
<table>
<tr><th>F3</th><th>F2</th><th>F1</th><th>F0</th><th>Timer operating mode select bits</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>Interval timer (one-shot mode)</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Interval timer (continuous mode)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Interval timer (free-run mode)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>PWM timer (fixed-cycle mode)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>PWM timer (variable-cycle mode)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>PWC timer ("H" pulse = rising to falling)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>PWC timer ("L" pulse = falling to rising)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>PWC timer (cycle = rising to rising)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>PWC timer (cycle = falling to falling)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>PWC timer
("H" pulse = rising to falling; Cycle = rising to rising)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>Input capture
(rising, free-run counter)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>Input capture
(falling, free-run counter)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>Input capture
(both edges, free-run counter)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>Input capture
(rising, counter clear)</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>Input capture
(falling, counter clear)</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>Input capture
(both edges, counter clear)</td></tr>
</table></td></tr>
</table>

18.5.2 8/16-bit Composite Timer 10/11 Status Control Register 0 (T10CR0/T11CR0)

The 8/16-bit composite timer 10/11 status control register 0 (T10CR0/T11CR0) selects the timer operation mode, selects the count clock, and enables or disables IF flag interrupts. The T10CR0 and T11CR0 registers correspond to timers 10 and 11 respectively.

■ 8/16-bit Composite Timer 10/11 Status Control Register 0 (T10CR0/T11CR0)

Figure 18.5-4 8/16-bit Composite Timer 10/11 Status Control Register 0 (T10CR0/T11CR0)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
T11CR0	0F97H	IFE	C2	C1	C0	F3	F2	F1	F0	00000000B
T10CR0	0F98H									
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

F3	F2	F1	F0	Timer operating mode select bits
0	0	0	0	Interval timer (one-shot mode)
0	0	0	1	Interval timer (continuous mode)
0	0	1	0	Interval timer (free-run mode)
0	0	1	1	PWM timer (fixed-cycle mode)
0	1	0	0	PWM timer (variable-cycle mode)
0	1	0	1	PWC timer ("H" pulse = rising to falling)
0	1	1	0	PWC timer ("L" pulse = falling to rising)
0	1	1	1	PWC timer (cycle = rising to rising)
1	0	0	0	PWC timer (cycle = falling to falling)
1	0	0	1	PWC timer ("H" pulse = rising to falling; Cycle = rising to rising)
1	0	1	0	Input capture (rising, free-run counter)
1	0	1	1	Input capture (falling, free-run counter)
1	1	0	0	Input capture (both edges, free-run counter)
1	1	0	1	Input capture (rising, counter clear)
1	1	1	0	Input capture (falling, counter clear)
1	1	1	1	Input capture (both edges, counter clear)

C2	C1	C0	Count clock select bits
0	0	0	1 × MCLK (machine clock)
0	0	1	1/2 × MCLK (machine clock)
0	1	0	1/4 × MCLK (machine clock)
0	1	1	1/8 × MCLK (machine clock)
1	0	0	1/16 × MCLK (machine clock)
1	0	1	1/32 × MCLK (machine clock)
1	1	0	CK06/CK16 clock*
1	1	1	External clock

IFE	IF flag interrupt enable bit
0	Disables the IF flag interrupt
1	Enables the IF flag interrupt

R/W : Readable/writable (The read value is the same as the write value.)

▨ : Initial value

* : In main clock mode, the CK06/CK16 clock is a clock source to be selected from the following three prescaler clocks: $1/2^7 \times F_{CH}$, $1/2^9 \times F_{CH}$ and $1/2^{11} \times F_{CH}$, in the SEL[1:0] bits in the EVCR register of the event counter. In main CR clock mode, the CK06/CK16 clock is a clock source to be selected from the following three prescaler clocks: $1/2^6 \times F_{CRH}$, $1/2^8 \times F_{CRH}$ and $1/2^{10} \times F_{CRH}$, in the SEL[1:0] bits in the EVCR register of the event counter.

Table 18.5-2 Functions of Bits in 8/16-bit Composite Timer 10/11 Status Control Register 0 (T10CR0/T11CR0) (1 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7</td><td>IFE:
IF flag interrupt enable bit</td><td>This bit enables or disables IF flag interrupts.
During timer operation (T10CR1/T11CR1:STA = 1), the write access to this bit has no effect on operation. Ensure that the timer has stopped before modifying this bit.
<b>Writing "0":</b> Disables IF flag interrupts.
<b>Writing "1":</b> An IF flag interrupt request is output when both the IE bit (T10CR1/T11CR1:IE) and the IF flag (T10CR1/T11CR1:IF) are set to "1".</td></tr>
<tr><td>bit6
to
bit4</td><td>C2, C1, C0:
Count clock select bits</td><td>These bits select the count clock.
• The count clock is generated by the prescaler. See "6.13 Operation of Prescaler".
• Write access to these bits is nullified in timer operation (T10CR1/T11CR1:STA = 1).
• The clock selection of T11CR0 (timer 11) is nullified in 16-bit operation.
• These bits cannot be set to "111_B" when the PWC function or input capture function is used. An attempt to write "111_B" with the PWC function or input capture function in use resets the bits to "000_B". The bits are also reset to "000_B" if the timer enters the input capture operation mode with the bits set to "111_B".
• When these bits are set to "110_B", the count clock from the time-base timer will be used as the count clock. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock. In the case of using the count clock from the time-base timer as the count clock, resetting the time-base timer by writing "1" to the time-base timer initialization bit in the time-base timer control register (TBTC:TCLR) will affect the count time.
<table>
<tr><th>C2</th><th>C1</th><th>C0</th><th>Count clock</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1 × MCLK (machine clock)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1/2 × MCLK (machine clock)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1/4 × MCLK (machine clock)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1/8 × MCLK (machine clock)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1/16 × MCLK (machine clock)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1/32 × MCLK (machine clock)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>CK06/CK16 clock*</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>External clock</td></tr>
</table>
*: In main clock mode, the CK06/CK16 clock is a clock source to be selected from the following three prescaler clocks: $1/2^7 \times F_{CH}$, $1/2^9 \times F_{CH}$ and $1/2^{11} \times F_{CH}$, in the SEL[1:0] bits in the EVCR register of the event counter.
In main CR clock mode, the CK06/CK16 clock is a clock source to be selected from the following three prescaler clocks: $1/2^6 \times F_{CRH}$, $1/2^8 \times F_{CRH}$ and $1/2^{10} \times F_{CRH}$, in the SEL[1:0] bits in the EVCR register of the event counter.
For details of the CK06/CK16 clock, see "20.3.1 Event Counter Control Register (EVCR)".</td></tr>
</table>

Table 18.5-2 Functions of Bits in 8/16-bit Composite Timer 10/11 Status Control Register 0 (T10CR0/T11CR0) (2 / 2)

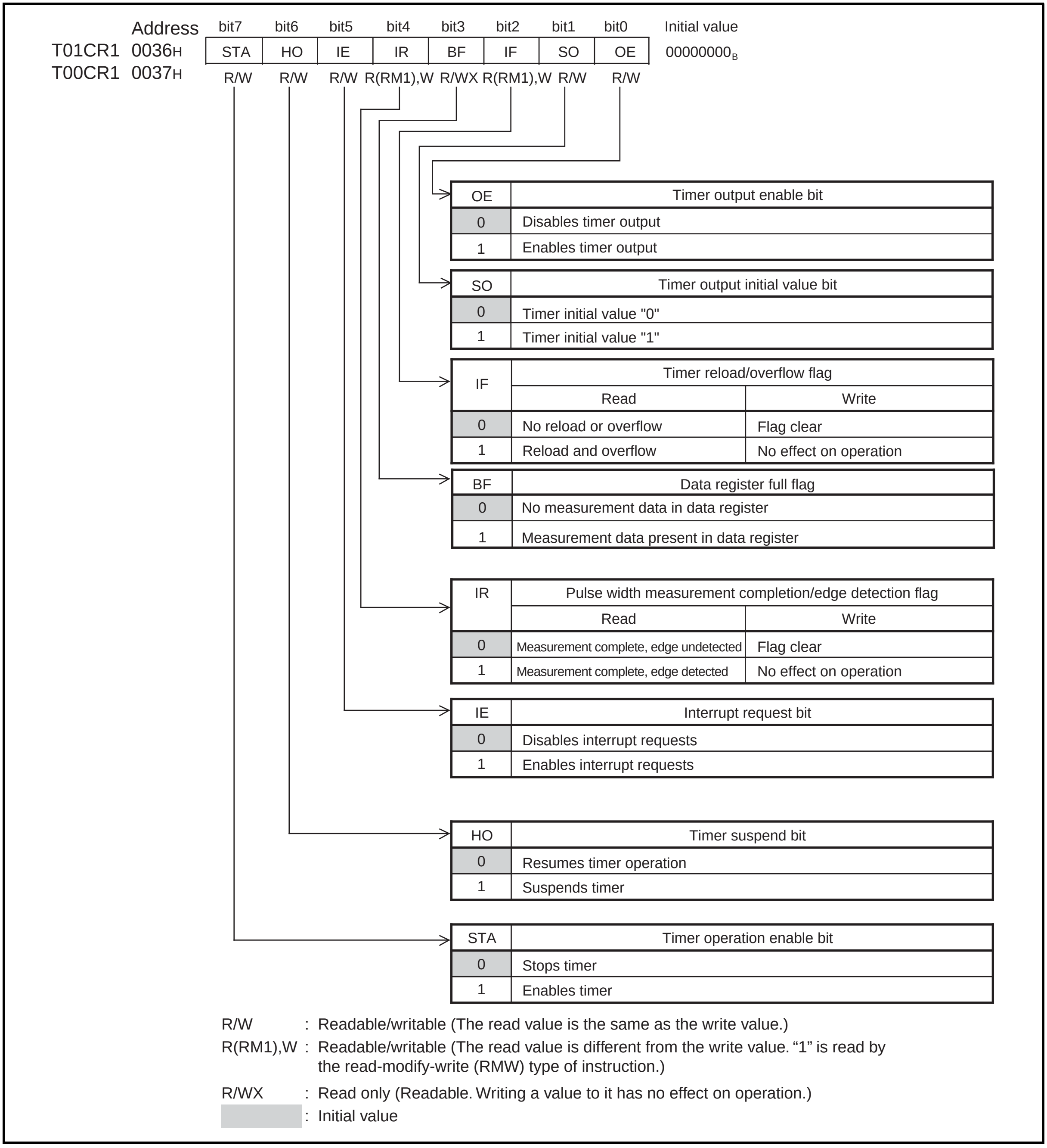
<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit3
to
bit0</td><td>F3, F2, F1, F0:
Timer operating mode select bits</td><td>These bits select the timer operating mode.
• The PWM timer function (variable-cycle mode; F3, F2, F1, F0 = 0100_B) is set by either the T10CR0 (timer 10) register or T11CR0 (timer 11) register. If one of the timers starts operating (T10CR1/T11CR1: STA= 1), the F3, F2, F1 and F0 bits of the other timer are automatically set to "0100_B".
• With the 16-bit operation having been selected (TMCR1:MOD = 1), if the composite timer starts operating using the PWM timer function (variable-cycle mode) (T10CR1/ T11CR1:STA = 1), the MOD bit is set to "0" automatically.
• Write access to these bits is nullified in timer operation (T10CR1/T11CR1:STA = 1).
<table>
<tr><th>F3</th><th>F2</th><th>F1</th><th>F0</th><th>Timer operating mode select bits</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>Interval timer (one-shot mode)</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Interval timer (continuous mode)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Interval timer (free-run mode)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>PWM timer (fixed-cycle mode)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>PWM timer (variable-cycle mode)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>PWC timer ("H" pulse = rising to falling)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>PWC timer ("L" pulse = falling to rising)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>PWC timer (cycle = rising to rising)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>PWC timer (cycle = falling to falling)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>PWC timer
("H" pulse = rising to falling; Cycle = rising to rising)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>Input capture
(rising, free-run counter)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>Input capture
(falling, free-run counter)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>Input capture
(both edges, free-run counter)</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>Input capture
(rising, counter clear)</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>Input capture
(falling, counter clear)</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>Input capture
(both edges, counter clear)</td></tr>
</table>
</td></tr>
</table>

18.5.3 8/16-bit Composite Timer 00/01 Status Control Register 1 (T00CR1/T01CR1)

The 8/16-bit composite timer 00/01 status control register 1 (T00CR1/T01CR1) controls the interrupt flag, timer output, and timer operations. T00CR1 and T01CR1 registers correspond to timers 00 and 01 respectively.

■ 8/16-bit Composite Timer 00/01 Status Control Register 1 (T00CR1/T01CR1)

Figure 18.5-5 8/16-bit Composite Timer 00/01 Status Control Register 1 (T00CR1/T01CR1)



	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
T01CR1	0036H	STA	HO	IE	IR	BF	IF	SO	OE	00000000_B
T00CR1	0037H	R/W	R/W	R/W	R(RM1),W	R/WX	R(RM1),W	R/W	R/W	

OE	Timer output enable bit
0	Disables timer output
1	Enables timer output

SO	Timer output initial value bit
0	Timer initial value "0"
1	Timer initial value "1"

IF	Timer reload/overflow flag	
	Read	Write
0	No reload or overflow	Flag clear
1	Reload and overflow	No effect on operation

BF	Data register full flag
0	No measurement data in data register
1	Measurement data present in data register

IR	Pulse width measurement completion/edge detection flag	
	Read	Write
0	Measurement complete, edge undetected	Flag clear
1	Measurement complete, edge detected	No effect on operation

IE	Interrupt request bit
0	Disables interrupt requests
1	Enables interrupt requests

HO	Timer suspend bit
0	Resumes timer operation
1	Suspends timer

STA	Timer operation enable bit
0	Stops timer
1	Enables timer

R/W : Readable/writable (The read value is the same as the write value.)

R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)

R/WX : Read only (Readable. Writing a value to it has no effect on operation.)

▨ : Initial value

Table 18.5-3 Functions of Bits in 8/16-bit Composite Timer 00/01 Status Control Register 1 (T00CR1/T01CR1) (1 / 2)

Bit name		Function
bit7	STA: Timer operation enable bit	This bit enables or stops the timer operation. **Writing "0":** Stops the timer operation and sets the count value to "00_H". • With the PWM timer function (variable-cycle mode) in use (T00CR0/T01CR0: F3, F2, F1, F0 = 0100_B), the STA bit in either the T00CR1 (timer 10) or the T01CR1 (timer 11) register can be used to enable or disable the timer operation. If the STA bit in one of the registers is set to "0", the STA bit in the other one is automatically set to the same value. • During 16-bit operation (TMCR0:MOD = 1), use the STA bit in the T00CR1 (timer 10) register to enable or disable timer operation. If the STA bit of one of the timers is set to "0", the STA bit in the other one is automatically set to the same value. **Writing "1":** allows timer operation to start from count value "00_H". • Before setting this bit to "1", set the count clock select bits (T00CR0/T01CR0:C2, C1, C0), timer operation select bits (T00CR0/T01CR0:F3, F2, F1, F0), timer output initial value bit (T00CR1/T01CR1:SO), 16-bit mode enable bit (TMCR0:MOD), and filter function select bits (TMCR0:FE11, FE10, FE01, FE00).
bit6	HO: Timer suspend bit	This bit suspends or resumes the timer operation. • Writing "1" to this bit during timer operation suspends the timer operation. • When the timer operation has been enabled (T00CR1/T01CR1:STA = 1), writing "0" to the bit resumes the timer operation. • With the PWM timer function (variable-cycle mode) in used (T00CR0/T01CR0: F3, F2, F1, F0=0100_B), the HO bit in either T00CR1 (timer 00) or T01CR1 (timer 01) can be used to suspend or resume timer operation. If the HO bit in one of the registers is set to "0" or "1", the HO bit in the other one is automatically set to the same value. • In 16-bit operation (TMCR0:MOD = 1), use the HO bit in the T00CR1 (timer 00) register to suspend or resume timer operation. If the HO bit in one of the registers is set to "0" or "1", the HO bit in the other one is automatically set to the same value.
bit5	IE: Interrupt request enable bit	This bit enables or disables the output of interrupt requests. **Writing "0":** Disables interrupt request. **Writing "1":** Outputs an interrupt request when the pulse width measurement completion/ edge detection flag (T00CR1/T01CR1:IR) or timer reload/overflow flag (T00CR1/T01CR1:IF) is "1". However, an interrupt request from the timer reload/overflow flag (T00CR1/ T01CR1:IF) is not output unless the IF flag interrupt enable (T00CR0/ T01CR0:IFE) bit is also set to "1".
bit4	IR: Pulse width measurement completion/edge detection flag	This bit indicates the completion of pulse width measurement or the detection of an edge. • With the PWC timer function in use, this bit is set to "1" immediately after pulse width measurement is complete. • With the input capture function in use, this bit is set to "1" immediately after an edge is detected. • The bit is set to "0" when the function of the composite timer selected is neither the PWC timer function nor the input capture function. • If this bit is read by the read-modify-write (RMW) type of instruction, it always returns "1". • The IR bit in the T01CR1 (timer 01) register is set to "0" in 16-bit operation. • Writing "0" to this bit sets it to "0". • Writing "1" to this bit is ignored.

Table 18.5-3 Functions of Bits in 8/16-bit Composite Timer 00/01 Status Control Register 1 (T00CR1/T01CR1) (2 / 2)

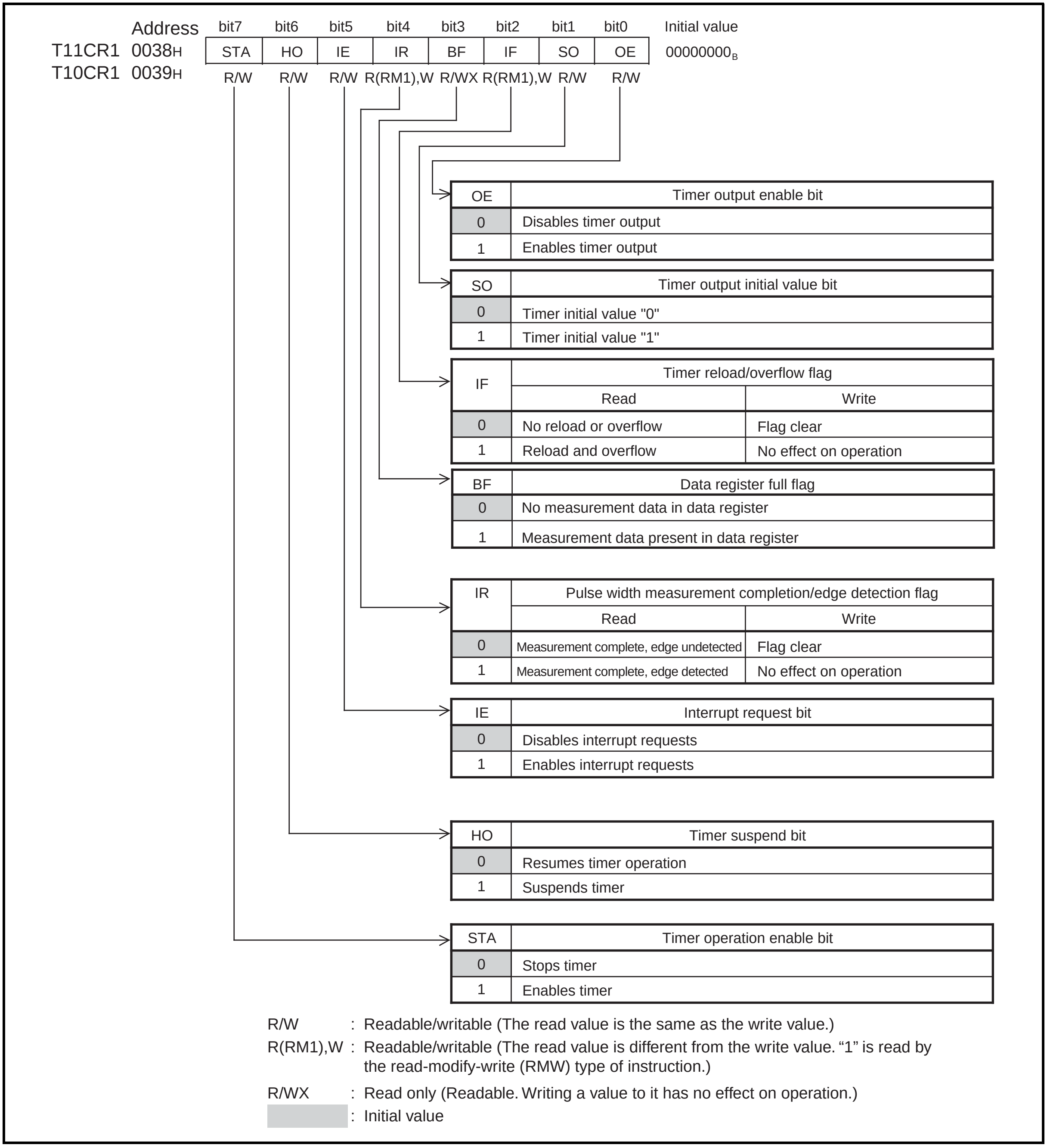
Bit name		Function
bit3	BF: Data register full flag	• With the PWC timer function in use, this bit is set to "1" when a count value is stored in the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) immediately after pulse width measurement is complete. • In 8-bit operation, this bit is set to "0" when the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is read. • The 8/16-bit composite timer 00/01 data register (T00DR/T01DR) holds data if this bit is set to "1". With this bit being "1", even when the next edge is detected, the count value is not transferred to the 8/16-bit composite timer 00/01 data register (T00DR/T01DR), and the next measurement result is thus lost. Nonetheless, there is an exception. With the F3 bit to F0 bit in the T00CR0/T01CR0 register having been set to "1001_B", even though the BF bit is set to "1", the "H" pulse measurement result is transferred to the 8/16-bit composite timer 00/01 data register (T00DR/T01DR), while the cycle measurement result is not transferred to the 8/16-bit composite timer 00/01 data register. Therefore, in order to perform cycle measurement, the "H" pulse measurement result must be read before a cycle is completed. In addition, the result of "H" pulse measurement and that of cycle measurement are lost if they are not read before the completion of the next "H" pulse. • The BF bit in the T00CR1 (timer 00) register is set to "0" when the T01DR (timer 01) register is read during 16-bit operation. • The BF bit in T01CR1 (timer 01) register is set to "0" during 16-bit operation. • This bit is "0" when any timer function other than the PWC timer function is selected. • Writing a value to this bit has no effect on operation.
bit2	IF: Timer reload/overflow flag	This bit is used to detect the count value match and the counter overflow. • With the interval timer function (one-shot or continuous) or the PWM timer function (variable-cycle mode) in use, this bit is set to "1" if the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) value matches the count value. • With the PWC timer function of the input capture function in use, this bit is set to "1" if a counter overflow occurs. • If this bit is read by a read-modify-write (RMW) instruction, it always returns "1". • Writing "0" to this bit sets it to "0". • Writing "1" to this bit has no effect on operation. • The bit becomes "0" if the PWM function (variable-cycle mode) is selected. • The IF bit in the T01CR1 (timer 01) register is "0" in 16-bit operation.
bit1	SO: Timer output initial value bit	The timer output (TMCR0:TO1/TO0) initial value is set by writing a value to this bit. The value in this bit is reflected in the timer output when the timer operation enable bit (T00CR1/T01CR1:STA) changes from "0" to "1". • In 16-bit operation (TMCR0:MOD = 1), use the SO bit in the T00CR1 (timer 00) register to set the timer output initial value. In this case, the value of the SO bit in the other one has no effect on operation. • During timer operation (T00CR1:STA = 1 or T01CR1:STA = 1), the write access to this bit is invalid. However, in 16-bit operation, although a value can be written to the SO bit in the T01CR1 (timer 01) register even during timer operation, the value written has no direct effect on the timer output. • When the PWM timer function (fixed cycle mode or variable cycle mode) or the input capture function is in use, the value of this bit has no effect on operation.
bit0	OE: Timer output enable bit	This bit enables or disables timer output. **Writing "0":** No timer output is supplied to the external pin. In this case, the external pin serves as a general-purpose port. **Writing "1":** The time output (TMCR0:TO1/TO0) is supplied to the external pin.

18.5.4 8/16-bit Composite Timer 10/11 Status Control Register 1 (T10CR1/T11CR1)

The 8/16-bit composite timer 10/11 status control register 1 (T10CR1/T11CR1) controls the interrupt flag, timer output, and timer operations. T10CR1 and T11CR1 registers correspond to timers 10 and 11 respectively.

■ 8/16-bit Composite Timer 10/11 Status Control Register 1 (T10CR1/T11CR1)

Figure 18.5-6 8/16-bit Composite Timer 10/11 Status Control Register 1 (T10CR1/T11CR1))



	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
T11CR1	0038H	STA	HO	IE	IR	BF	IF	SO	OE	00000000_B
T10CR1	0039H	R/W	R/W	R/W	R(RM1),W	R/WX	R(RM1),W	R/W	R/W	

OE	Timer output enable bit
0	Disables timer output
1	Enables timer output

SO	Timer output initial value bit
0	Timer initial value "0"
1	Timer initial value "1"

IF	Timer reload/overflow flag	
	Read	Write
0	No reload or overflow	Flag clear
1	Reload and overflow	No effect on operation

BF	Data register full flag
0	No measurement data in data register
1	Measurement data present in data register

IR	Pulse width measurement completion/edge detection flag	
	Read	Write
0	Measurement complete, edge undetected	Flag clear
1	Measurement complete, edge detected	No effect on operation

IE	Interrupt request bit
0	Disables interrupt requests
1	Enables interrupt requests

HO	Timer suspend bit
0	Resumes timer operation
1	Suspends timer

STA	Timer operation enable bit
0	Stops timer
1	Enables timer

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
▨ : Initial value

Table 18.5-4 Functions of Bits in 8/16-bit Composite Timer 10/11 Status Control Register 1 (T10CR1/T11CR1) (1 / 2)

Bit name		Function
bit7	STA: Timer operation enable bit	This bit enables or stops the timer operation. **Writing "0":** Stops the timer operation and sets the count value to "00_H". • With the PWM timer function (variable-cycle mode) in use (T10CR0/T11CR0: F3, F2, F1, F0 = 0100_B), the STA bit in either the T10CR1 (timer 10) or the T11CR1 (timer 11) register can be used to enable or disable the timer operation. If the STA bit in one of the registers is set to "0", the STA bit in the other one is automatically set to the same value. • During 16-bit operation (TMCR1:MOD = 1), use the STA bit in the T10CR1 (timer 10) register to enable or disable timer operation. If the STA bit of one of the timers is set to "0", the STA bit in the other one is automatically set to the same value. **Writing "1":** Allows timer operation to start from count value "00_H". • Before setting this bit to "1", set the count clock select bits (T10CR0/T11CR0:C2, C1, C0), timer operation select bits (T10CR0/T11CR0:F3, F2, F1, F0), timer output initial value bit (T10CR1/T11CR1:SO), 16-bit mode enable bit (TMCR1:MOD), and filter function select bits (TMCR1:FE11, FE10, FE01, FE00).
bit6	HO: Timer suspend bit	This bit suspends or resumes the timer operation. • Writing "1" to this bit during timer operation suspends the timer operation. • When the timer operation has been enabled (T10CR1/T11CR1:STA = 1), writing "0" to the bit resumes the timer operation. • With the PWM timer function (variable-cycle mode) in used (T10CR0/T11CR0: F3, F2, F1, F0=0100_B), the HO bit in either T10CR1 (timer 10) or T11CR1 (timer 11) can be used to suspend or resume timer operation. If the HO bit in one of the registers is set to "0" or "1", the HO bit in the other one is automatically set to the same value. • In 16-bit operation (TMCR1:MOD = 1), use the HO bit in the T10CR1 (timer 10) register to suspend or resume timer operation. If the HO bit in one of the registers is set to "0" or "1", the HO bit in the other one is automatically set to the same value.
bit5	IE: Interrupt request enable bit	This bit enables or disables the output of interrupt requests. **Writing "0":** Disables interrupt request. **Writing "1":** Outputs an interrupt request when the pulse width measurement completion/ edge detection flag (T10CR1/T11CR1:IR) or timer reload/overflow flag (T10CR1/T11CR1:IF) is "1". However, an interrupt request from the timer reload/overflow flag (T10CR1/ T11CR1:IF) is not output unless the IF flag interrupt enable (T10CR0/ T11CR0:IFE) bit is also set to "1".
bit4	IR: Pulse width measurement completion/edge detection flag	This bit indicates the completion of pulse width measurement or the detection of an edge. • With the PWC timer function in use, this bit is set to "1" immediately after pulse width measurement is complete. • With the input capture function in use, this bit is set to "1" immediately after an edge is detected. • The bit is set to "0" when the function of the composite timer selected is neither the PWC timer function nor the input capture function. • If this bit is read by the read-modify-write (RMW) type of instruction, it always returns "1". • The IR bit in the T11CR1 (timer 01) register is set to "0" in 16-bit operation. • Writing "0" to this bit sets it to "0". • Writing "1" to this bit is ignored.

Table 18.5-4 Functions of Bits in 8/16-bit Composite Timer 10/11 Status Control Register 1 (T10CR1/T11CR1) (2 / 2)

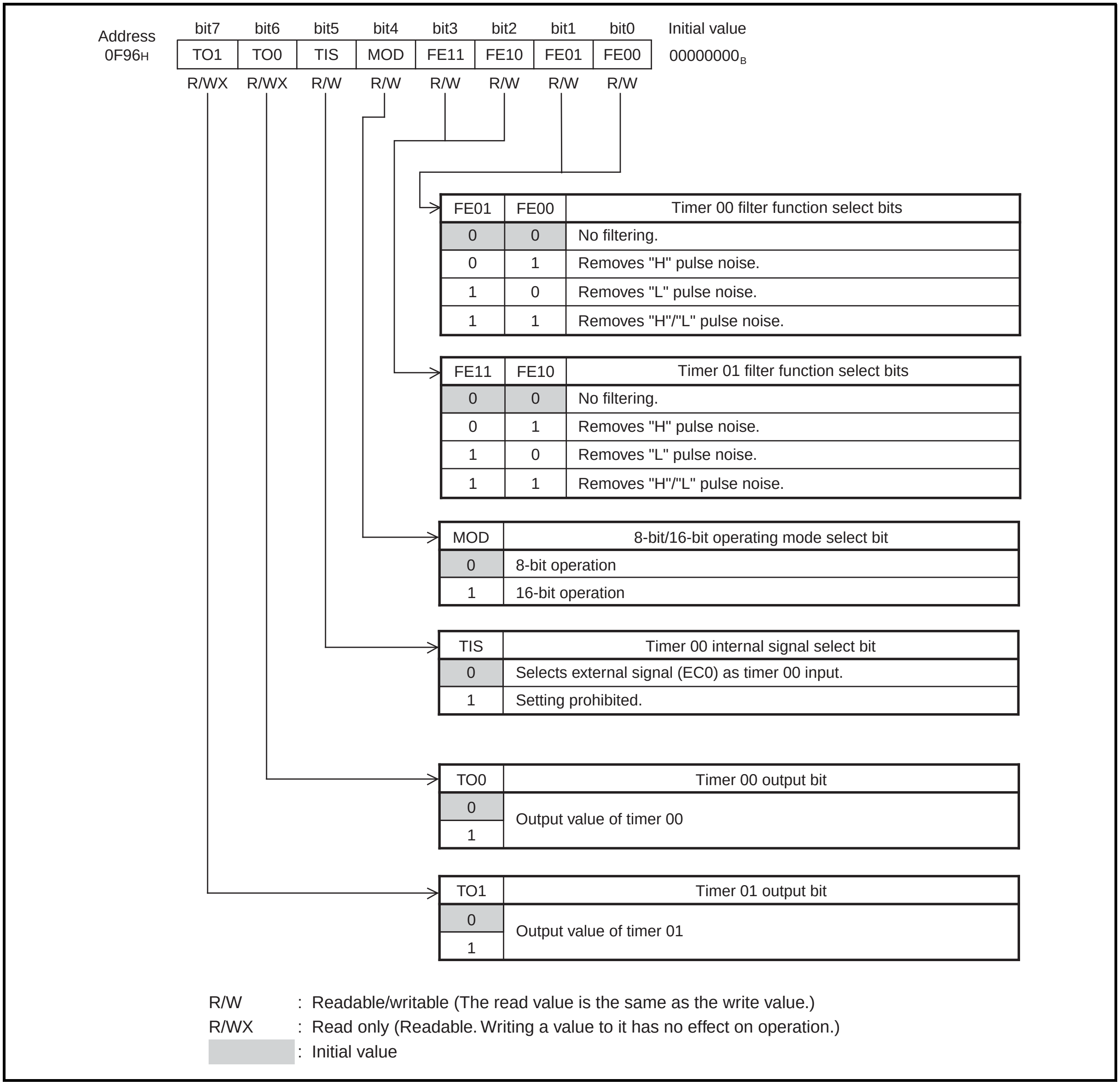
Bit name		Function
bit3	BF: Data register full flag	• With the PWC timer function in use, this bit is set to "1" when a count value is stored in the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) immediately after pulse width measurement is complete. • In 8-bit operation, this bit is set to "0" when the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is read. • The 8/16-bit composite timer 10/11 data register (T10DR/T11DR) holds data if this bit is set to "1". With this bit being "1", even when the next edge is detected, the count value is not transferred to the 8/16-bit composite timer 10/11 data register (T10DR/T11DR), and the next measurement result is thus lost. Nonetheless, there is an exception. With the F3 bit to F0 bit in the T10CR0/T11CR0 register having been set to "1001_B", even though the BF bit is set to "1", the "H" pulse measurement result is transferred to the 8/16-bit composite timer 10/11 data register (T10DR/T11DR), while the cycle measurement result is not transferred to the 8/16-bit composite timer 10/11 data register. Therefore, in order to perform cycle measurement, the "H" pulse measurement result must be read before a cycle is completed. In addition, the result of "H" pulse measurement and that of cycle measurement are lost if they are not read before the completion of the next "H" pulse. • The BF bit in the T10CR1 (timer 10) register is set to "0" when the T11DR (timer 11) register is read during 16-bit operation. • The BF bit in T11CR1 (timer 11) register is set to "0" during 16-bit operation. • This bit is "0" when any timer function other than the PWC timer function is selected. • Writing a value to this bit has no effect on operation.
bit2	IF: Timer reload/overflow flag	This bit is used to detect the count value match and the counter overflow. • With the interval timer function (one-shot or continuous) or the PWM timer function (variable-cycle mode) in use, this bit is set to "1" if the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) value matches the count value. • With the PWC timer function of the input capture function in use, this bit is set to "1" if a counter overflow occurs. • If this bit is read by a read-modify-write (RMW) instruction, it always returns "1". • Writing "0" to this bit sets it to "0". • Writing "1" to this bit has no effect on operation. • The bit becomes "0" if the PWM function (variable-cycle mode) is selected. • The IF bit in the T11CR1 (timer 11) register is "0" in 16-bit operation.
bit1	SO: Timer output initial value bit	The timer output (TMCR1:TO1/TO0) initial value is set by writing a value to this bit. The value in this bit is reflected in the timer output when the timer operation enable bit (T10CR1/T11CR1:STA) changes from "0" to "1". • In 16-bit operation (TMCR1:MOD = 1), use the SO bit in the T10CR1 (timer 10) register to set the timer output initial value. In this case, the value of the SO bit in the other one has no effect on operation. • During timer operation (T10CR1:STA = 1 or T11CR1:STA = 1), the write access to this bit is invalid. However, in 16-bit operation, although a value can be written to the SO bit in the T11CR1 (timer 11) register even during timer operation, the value written has no direct effect on the timer output. • When the PWM timer function (fixed cycle mode or variable cycle mode) or the input capture function is in use, the value of this bit has no effect on operation.
bit0	OE: Timer output enable bit	This bit enables or disables timer output. **Writing "0":** No timer output is supplied to the external pin. In this case, the external pin serves as a general-purpose port. **Writing "1":** The time output (TMCR1:TO1/TO0) is supplied to the external pin.

18.5.5 8/16-bit Composite Timer 00/01 Timer Mode Control Register (TMCR0)

The 8/16-bit composite timer 00/01 timer mode control register (TMCR0) selects the filter function, 8-bit or 16-bit operating mode, and signal input to timer 00 and indicates the timer output value. This register serves both timer 00 and timer 01.

■ 8/16-bit Composite Timer 00/01 Timer Mode Control Register (TMCR0)

Figure 18.5-7 8/16-bit Composite Timer 00/01 Timer Mode Control Register (TMCR0)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0F96H	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00	00000000_B
	R/WX	R/WX	R/W	R/W	R/W	R/W	R/W	R/W	

FE01	FE00	Timer 00 filter function select bits
0	0	No filtering.
0	1	Removes "H" pulse noise.
1	0	Removes "L" pulse noise.
1	1	Removes "H"/"L" pulse noise.

FE11	FE10	Timer 01 filter function select bits
0	0	No filtering.
0	1	Removes "H" pulse noise.
1	0	Removes "L" pulse noise.
1	1	Removes "H"/"L" pulse noise.

MOD	8-bit/16-bit operating mode select bit
0	8-bit operation
1	16-bit operation

TIS	Timer 00 internal signal select bit
0	Selects external signal (EC0) as timer 00 input.
1	Setting prohibited.

TO0	Timer 00 output bit
0	Output value of timer 00
1	

TO1	Timer 01 output bit
0	Output value of timer 01
1	

R/W : Readable/writable (The read value is the same as the write value.)
R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
▨ : Initial value

Table 18.5-5 Functions of Bits in 8/16-bit Composite Timer 00/01 Timer Mode Control Register (TMCR0) (1 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7</td><td>TO1:
Timer 01 output bit</td><td>This bit indicates the output value of timer 01. When the timer starts operation (T00CR1/ T01CR1:STA = 1), the value in the bit changes depending on the timer function selected.
• Writing a value to this bit has no effect on operation.
• In 16-bit operation, if the PWM timer function (variable-cycle mode) or the input capture function is selected, the value in the bit becomes undefined.
• With the interval timer function or the PWC timer function having been selected, if the timer stops operating (T00CR1/T01CR1:STA = 0), this bit holds the last value.
• With the PWM timer function (variable-cycle mode) having been selected, if the timer stops operating (T00CR1/T01CR1:STA = 0), this bit holds the last value.
• When the timer operating mode select bits (T00CR0/T01CR0: F3, F2, F1, F0) are modified with the timer stopping operating, this bit indicates the last value of timer operation if the same timer operation has been performed; otherwise it indicates "0", its initial value.</td></tr>
<tr><td>bit6</td><td>TO0:
Timer 00 output bit</td><td>This bit indicates the output value of timer 00. When the timer starts operation (T00CR1/ T01CR1:STA = 1), the value in the bit changes depending on the selected timer function.
• Writing a value to this bit has no effect on operation.
• If the input capture function is selected, the value in the bit becomes undefined.
• With the interval timer function or the PWM timer (variable-cycle mode) or the PWC timer function having been selected, if the timer stops operating (T00CR1/T01CR1:STA = 0), this bit holds the last value.
• With the PWM timer function (variable-cycle mode) having been selected, if the timer stops operating (T00CR1/T01CR1:STA = 0), this bit holds the last value.
• When the timer operating mode select bits (T00CR0/T01CR0: F3, F2, F1, F0) are modified with the timer stopping operating, this bit indicates the last value of timer operation if the same timer operation has been performed; otherwise it indicates "0", its initial value.</td></tr>
<tr><td>bit5</td><td>TIS:
Timer 00 internal signal select bit</td><td>This bit selects the signal input to timer 00 when the PWC timer function or input capture function is selected.
<b>Writing "0":</b> Selects the external signal (EC0) as the signal input for timer 00.
<b>Writing "1":</b> Setting prohibited.</td></tr>
<tr><td>bit4</td><td>MOD:
16-bit mode enable bit</td><td>This bit selects 8-bit or 16-bit operation mode.
<b>Writing "0":</b> Allows timers 00 and 01 to operate as separate 8-bit timers.
<b>Writing "1":</b> Allows timers 00 and 01 to operate as a 16-bit timer.
• While this bit is "1", if the timer starts operating (T00CR1/T01CR1:STA = 1) with the PWM timer function (variable-cycle mode), this bit is automatically set to "0".
• During timer operation (T00CR1:STA = 1 or T01CR1:STA = 1), the write access to this bit is invalid.</td></tr>
<tr><td>bit3,
bit2</td><td>FE11, FE10:
Timer 01 filter function select bits</td><td>These bits select the filter function for the external signal (EC0) to timer 01 when the PWC timer function or the input capture function is selected.
<table>
<tr><th>FE11</th><th>FE10</th><th>Timer 01 filter</th></tr>
<tr><td>0</td><td>0</td><td>No filtering out.</td></tr>
<tr><td>0</td><td>1</td><td>Filters out "H" pulse noise.</td></tr>
<tr><td>1</td><td>0</td><td>Filters out "L" pulse noise.</td></tr>
<tr><td>1</td><td>1</td><td>Filters out both "H" pulse noise and "L" pulse noise.</td></tr>
</table>
• During timer operation (T00CR1:STA = 1), the write access to these bits is invalid.
• The settings of the bits have no effect on operation when the interval timer function or the PWM timer function is selected (the filter function does not operate.).</td></tr>
</table>

Table 18.5-5 Functions of Bits in 8/16-bit Composite Timer 00/01 Timer Mode Control Register (TMCR0) (2 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit1,
bit0</td><td>FE01, FE00:
Timer 00 filter function select bits</td><td>These bits select the filter function for the external signal (EC0) to timer 00 when the PWC timer function or the input capture function is selected.
<table>
<tr><th>FE01</th><th>FE00</th><th>Timer 00 filter</th></tr>
<tr><td>0</td><td>0</td><td>No filtering out.</td></tr>
<tr><td>0</td><td>1</td><td>Filters out "H" pulse noise.</td></tr>
<tr><td>1</td><td>0</td><td>Filters out "L" pulse noise.</td></tr>
<tr><td>1</td><td>1</td><td>Filters out both "H" pulse noise and "L" pulse noise.</td></tr>
</table>

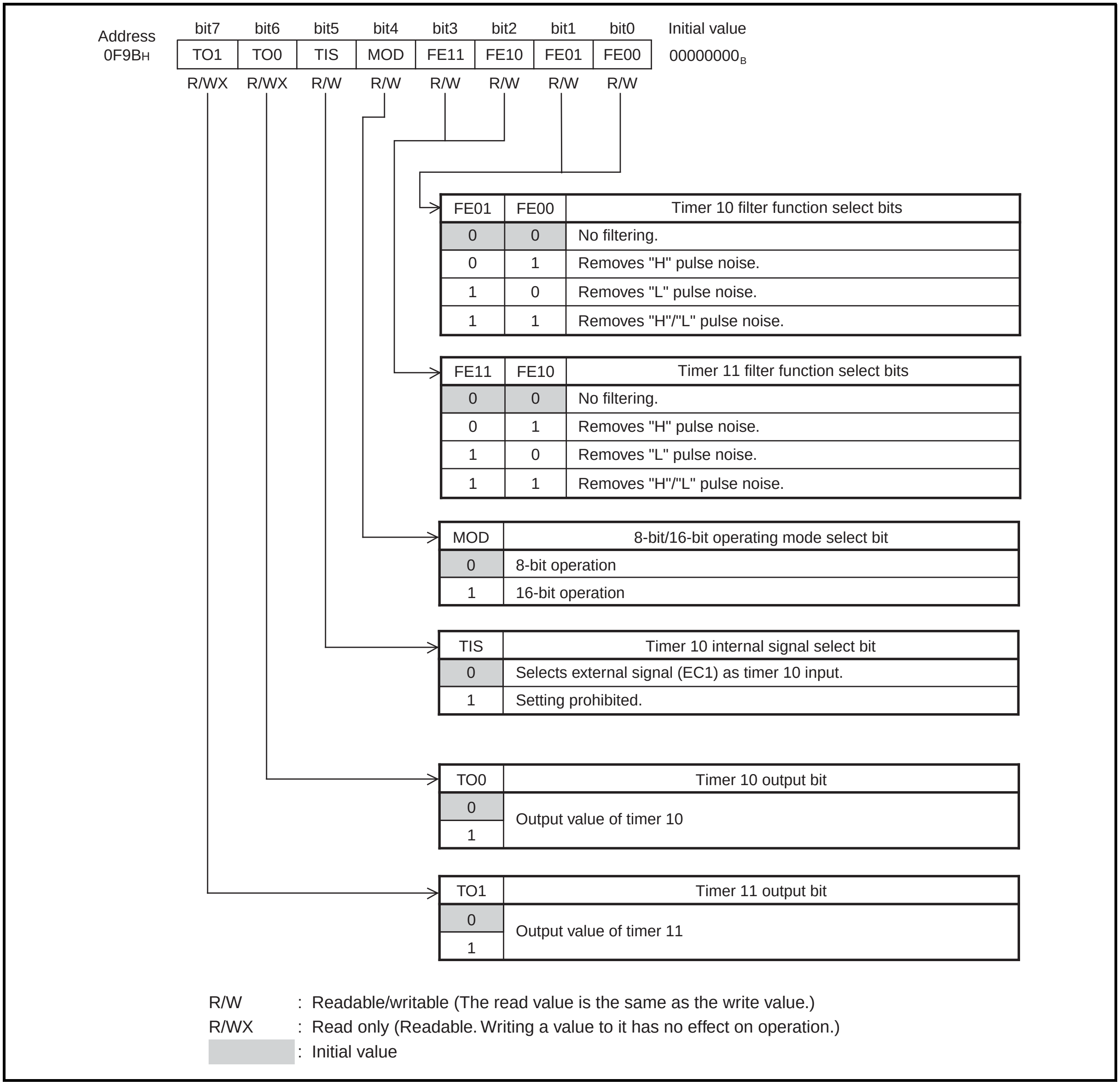
• During timer operation (T00CR1:STA = 1), the write access to these bits is invalid.
• The settings of these bits have no effect on operation when the interval timer function or the PWM timer function is selected (the filter function does not operate.).</td></tr>
</table>

18.5.6 8/16-bit Composite Timer 10/11 Timer Mode Control Register (TMCR1)

The 8/16-bit composite timer 10/11 timer mode control register (TMCR1) selects the filter function, 8-bit or 16-bit operating mode, and signal input to timer 10 and indicates the timer output value. This register serves both timer 10 and timer 11.

■ 8/16-bit Composite Timer 10/11 Timer Mode Control Register (TMCR1)

Figure 18.5-8 8/16-bit Composite Timer 10/11 Timer Mode Control Register (TMCR1)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0F9BH	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00	00000000B
	R/WX	R/WX	R/W	R/W	R/W	R/W	R/W	R/W	

FE01	FE00	Timer 10 filter function select bits
0	0	No filtering.
0	1	Removes "H" pulse noise.
1	0	Removes "L" pulse noise.
1	1	Removes "H"/"L" pulse noise.

FE11	FE10	Timer 11 filter function select bits
0	0	No filtering.
0	1	Removes "H" pulse noise.
1	0	Removes "L" pulse noise.
1	1	Removes "H"/"L" pulse noise.

MOD	8-bit/16-bit operating mode select bit
0	8-bit operation
1	16-bit operation

TIS	Timer 10 internal signal select bit
0	Selects external signal (EC1) as timer 10 input.
1	Setting prohibited.

TO0	Timer 10 output bit
0	Output value of timer 10
1	

TO1	Timer 11 output bit
0	Output value of timer 11
1	

R/W : Readable/writable (The read value is the same as the write value.)
R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
▨ : Initial value

.

Table 18.5-6 Functions of Bits in 8/16-bit Composite Timer 10/11 Timer Mode Control Register (TMCR1) (1 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7</td><td>TO1:
Timer 11 output bit</td><td>This bit indicates the output value of timer 11. When the timer starts operation (T10CR1/T11CR1:STA = 1), the value in the bit changes depending on the timer function selected.
• Writing a value to this bit has no effect on operation.
• In 16-bit operation, if the PWM timer function (variable-cycle mode) or the input capture function is selected, the value in the bit becomes undefined.
• With the interval timer function or the PWC timer function having been selected, if the timer stops operating (T10CR1/T11CR1:STA = 0), this bit holds the last value.
• With the PWM timer function (variable-cycle mode) having been selected, if the timer stops operating (T10CR1/T11CR1:STA = 0), this bit holds the last value.
• When the timer operating mode select bits (T10CR0/T11CR0: F3, F2, F1, F0) are modified with the timer stopping operating, this bit indicates the last value of timer operation if the same timer operation has been performed; otherwise it indicates "0", its initial value.</td></tr>
<tr><td>bit6</td><td>TO0:
Timer 10 output bit</td><td>This bit indicates the output value of timer 10. When the timer starts operation (T10CR1/T11CR1:STA = 1), the value in the bit changes depending on the selected timer function.
• Writing a value to this bit has no effect on operation.
• If the input capture function is selected, the value in the bit becomes undefined.
• With the interval timer function or the PWM timer (variable-cycle mode) or the PWC timer function having been selected, if the timer stops operating (T10CR1/T11CR1:STA = 0), this bit holds the last value.
• With the PWM timer function (variable-cycle mode) having been selected, if the timer stops operating (T10CR1/T11CR1:STA = 0), this bit holds the last value.
• When the timer operating mode select bits (T10CR0/T11CR0: F3, F2, F1, F0) are modified with the timer stopping operating, this bit indicates the last value of timer operation if the same timer operation has been performed; otherwise it indicates "0", its initial value.</td></tr>
<tr><td>bit5</td><td>TIS:
Timer 10 internal signal select bit</td><td>This bit selects the signal input to timer 10 when the PWC timer function or input capture function is selected.
<b>Writing "0":</b> Selects the external signal (EC1) as the signal input for timer 10.
<b>Writing "1":</b> Setting prohibited.</td></tr>
<tr><td>bit4</td><td>MOD:
16-bit mode enable bit</td><td>This bit selects 8-bit or 16-bit operation mode.
<b>Writing "0":</b> Allows timers 10 and 11 to operate as separate 8-bit timers.
<b>Writing "1":</b> Allows timers 10 and 11 to operate as a 16-bit timer.
• While this bit is "1", if the timer starts operating (T10CR1/T11CR1:STA = 1) with the PWM timer function (variable-cycle mode), this bit is automatically set to "0".
• During timer operation (T10CR1:STA = 1 or T11CR1:STA = 1), the write access to this bit is invalid.</td></tr>
<tr><td>bit3,
bit2</td><td>FE11, FE10:
Timer 11 filter function select bits</td><td>These bits select the filter function for the external signal (EC1) to timer 11 when the PWC timer function or the input capture function is selected.
<table>
<tr><th>FE11</th><th>FE10</th><th>Timer 11 filter</th></tr>
<tr><td>0</td><td>0</td><td>No filtering out.</td></tr>
<tr><td>0</td><td>1</td><td>Filters out "H" pulse noise.</td></tr>
<tr><td>1</td><td>0</td><td>Filters out "L" pulse noise.</td></tr>
<tr><td>1</td><td>1</td><td>Filters out both "H" pulse noise and "L" pulse noise.</td></tr>
</table>
• During timer operation (T10CR1:STA = 1), the write access to these bits is invalid.
• The settings of the bits have no effect on operation when the interval timer function or the PWM timer function is selected (the filter function does not operate.).</td></tr>
</table>

Table 18.5-6 Functions of Bits in 8/16-bit Composite Timer 10/11 Timer Mode Control Register (TMCR1) (2 / 2)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit1,
bit0</td><td>FE01, FE00:
Timer 10 filter function select bits</td><td>These bits select the filter function for the external signal (EC1) to timer 10 when the PWC timer function or the input capture function is selected.
<table>
<tr><th>FE01</th><th>FE00</th><th>Timer 10 filter</th></tr>
<tr><td>0</td><td>0</td><td>No filtering out.</td></tr>
<tr><td>0</td><td>1</td><td>Filters out "H" pulse noise.</td></tr>
<tr><td>1</td><td>0</td><td>Filters out "L" pulse noise.</td></tr>
<tr><td>1</td><td>1</td><td>Filters out both "H" pulse noise and "L" pulse noise.</td></tr>
</table>
• During timer operation (T10CR1:STA = 1), the write access to these bits is invalid.
• The settings of these bits have no effect on operation when the interval timer function or the PWM timer function is selected (the filter function does not operate.).</td></tr>
</table>

18.5.7 8/16-bit Composite Timer 00/01 Data Register (T00DR/T01DR)

The 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is used to set the maximum count value during the interval timer operation or the PWM timer operation and to read the count value during the PWC timer operation or the input capture operation. The T00DR and T01DR registers correspond to timers 00 and 01 respectively.

■ 8/16-bit Composite Timer 00/01 Data Register (T00DR/T01DR)

Figure 18.5-9 8/16-bit Composite Timer 00/01 Data Register (T00DR/T01DR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
T01DR	0F94$_H$	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	00000000$_B$
T00DR	0F95$_H$	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W	

R,W : Readable/writable (The read value is different from the write value.)

● Interval timer function

The 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is used to set the interval time. When the timer starts operating (T00CR1/T01CR1:STA = 1), the value of this register is transferred to the latch in the 8-bit comparator and the counter starts counting. When the count value matches the value held in the latch in the 8-bit comparator, the value of this register is transferred again to the latch, and the counter returns to "00$_H$" and continues to count.

The current count value can be read from this register.

An attempt to write "00$_H$" to this register is disabled in interval timer function.

In 16-bit operation, write the upper timer data to T01DR and lower timer data to T00DR, and write or read T01DR first and then T00DR.

● PWM timer function (fixed-cycle)

The 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is used to set "H" pulse width time. When the timer starts operating (T00CR1/T01CR1:STA = 1), the value of this register is transferred to the latch in the 8-bit comparator and the counter starts counting from timer output "H". When the count value matches the value transferred to the latch, the timer output becomes "L" and the counter continues to count until the count value reaches "FF$_H$". When an overflow occurs, the value of this register is transferred again to the latch in the 8-bit comparator and the counter performs the next cycle of counting.

The current value can be read from this register. In 16-bit operation, write the upper timer data to T01DR and lower timer data to T00DR, and write or read T01DR first and then T00DR.

- PWM timer function (variable-cycle)

The 8/16-bit composite timer 00 data register (T00DR) and 8/16-bit composite timer 01 data register (T01DR) are used to set "L" pulse width time and cycle respectively. When the timer starts operating (T00CR1/T01CR1:STA = 1), the value of each register is transferred to the latch in the 8-bit comparator and the two counters start counting from timer output "L". When the T00DR value transferred to the latch matches the timer 00 counter value, the timer output becomes "H" and the counting continues until the T01DR value transferred to the latch matches the timer 01 counter value. When the T01DR value transferred to the latch of the 8-bit comparator matches the timer 01 counter value, the values of the T00DR register and the T01DR register are transferred again to the latch and the counter performs the next PWM cycle of counting.

The current count value can be read from this register. In 16-bit operation, write the upper timer data to T01DR and lower timer data to T00DR, and read T01DR first and then T00DR.

- PWC timer function

The 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is used to read PWC measurement results. When PWC measurement is completed, the counter value is transferred to this register and the BF bit is set to "1".

When the 8/16-bit composite timer 00/01 data register is read, the BF bit is set to "0". While the BF bit is "1", no data is transferred to the 8/16-bit composite timer 00/01 data register.

There is an exception. With the F3 bit to F0 bit in the T00CR0/T01CR0 register having been set to "1001_B", even though the BF bit is set to "1", the "H" pulse measurement result is transferred to the 8/16-bit composite timer 00/01 data register, while the cycle measurement result is not transferred to the 8/16-bit composite timer 00/01 data register. Therefore, in order to perform cycle measurement, the "H" pulse measurement result must be read before a cycle is completed. In addition, the result of "H" pulse measurement and that of cycle measurement are lost if they are not read before the completion of the next "H" pulse.

When reading the 8/16-bit composite timer 00/01 data register, ensure that the BF bit is not cleared accidentally.

If new data is written to the 8/16-bit composite timer 00/01 data register, the stored measurement data is replaced with the new data. Therefore, do not write data to the register. In 16-bit operation, write the upper timer data to T01DR and lower timer data to T00DR, and read T01DR first and then T00DR.

- Input capture function

The 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is used to read input capture results. When an edge specified is detected, the counter value is transferred to the 8/16-bit composite timer 00/01 data register.

If new data is written to the 8/16-bit composite timer 00/01 data register, the stored measurement data is replaced with the new data. Therefore, do not write data to the register. In 16-bit operation, write the upper timer data to T01DR and lower timer data to T00DR, and read T01DR first and then T00DR.

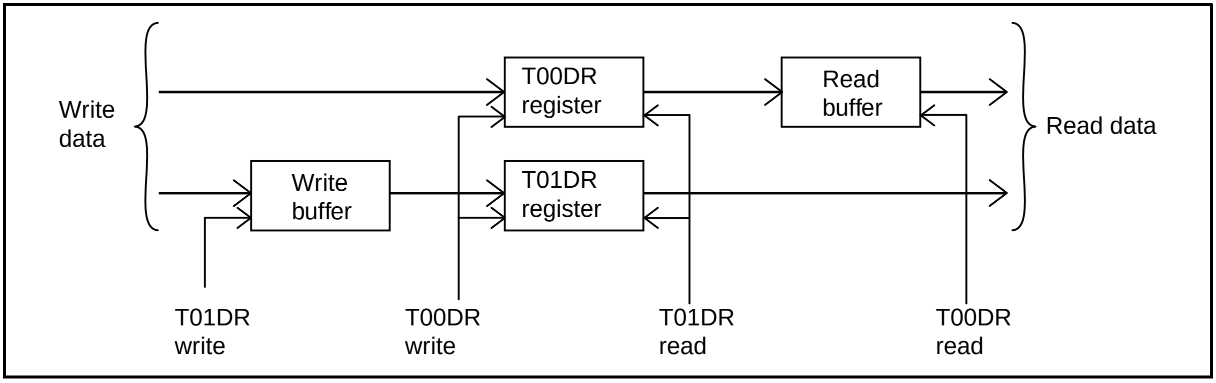
● Read and write operations

Read and write operations of T00DR and T01DR are performed in the following manner in 16-bit operation or when the PWM timer function (variable-cycle) is selected.

- Read from T01DR: In addition to the read access to T01DR, the value of T00DR is also stored in the internal read buffer at the same time.
- Read from T00DR: The internal read buffer is read.
- Write to T01DR: Data is written to the internal write buffer.
- Write to T00DR: In addition to the write access to T00DR, the value of the internal write buffer is stored in T01DR at the same time.

Figure 18.5-10 shows the T00DR and T01DR registers read from and written to during 16-bit operation.

Figure 18.5-10 Read and write operations of T00DR and T01DR registers during 16-bit operation



18.5.8 8/16-bit Composite Timer 10/11 Data Register (T10DR/T11DR)

The 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is used to set the maximum count value during the interval timer operation or the PWM timer operation and to read the count value during the PWC timer operation or the input capture operation. The T10DR and T11DR registers correspond to timers 10 and 11 respectively.

■ 8/16-bit Composite Timer 10/11 Data Register (T10DR/T11DR)

Figure 18.5-11 8/16-bit Composite Timer 10/11 Data Register (T10DR/T11DR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
T11DR	0F99$_H$	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	00000000$_B$
T10DR	0F9A$_H$	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W	

R,W : Readable/writable (The read value is different from the write value.)

● Interval timer function

The 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is used to set the interval time. When the timer starts operating (T10CR1/T11CR1:STA = 1), the value of this register is transferred to the latch in the 8-bit comparator and the counter starts counting. When the count value matches the value held in the latch in the 8-bit comparator, the value of this register is transferred again to the latch, and the counter returns to "00$_H$" and continues to count.

The current count value can be read from this register.

An attempt to write "00$_H$" to this register is disabled in interval timer function.

In 16-bit operation, write the upper timer data to T11DR and lower timer data to T10DR, and write or read T11DR first and then T10DR.

● PWM timer function (fixed-cycle)

The 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is used to set "H" pulse width time. When the timer starts operating (T10CR1/T11CR1:STA = 1), the value of this register is transferred to the latch in the 8-bit comparator and the counter starts counting from timer output "H". When the count value matches the value transferred to the latch, the timer output becomes "L" and the counter continues to count until the count value reaches "FF$_H$". When an overflow occurs, the value of this register is transferred again to the latch in the 8-bit comparator and the counter performs the next cycle of counting.

The current value can be read from this register. In 16-bit operation, write the upper timer data to T11DR and lower timer data to T10DR, and write or read T11DR first and then T10DR.

● PWM timer function (variable-cycle)

The 8/16-bit composite timer 10 data register (T10DR) and 8/16-bit composite timer 11 data register (T11DR) are used to set "L" pulse width time and cycle respectively. When the timer starts operating (T10CR1/T11CR1:STA = 1), the value of each register is transferred to the latch in the 8-bit comparator and the two counters start counting from timer output "L". When the T10DR value transferred to the latch matches the timer 10 counter value, the timer output becomes "H" and the counting continues until the T11DR value transferred to the latch matches the timer 11 counter value. When the T11DR value transferred to the latch of the 8-bit comparator matches the timer 11 counter value, the values of the T10DR register and the T11DR register are transferred again to the latch and the counter performs the next PWM cycle of counting.

The current count value can be read from this register. In 16-bit operation, write the upper timer data to T11DR and lower timer data to T10DR, and read T11DR first and then T10DR.

● PWC timer function

The 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is used to read PWC measurement results. When PWC measurement is completed, the counter value is transferred to this register and the BF bit is set to "1".

When the 8/16-bit composite timer 10/11 data register is read, the BF bit is set to "0". While the BF bit is "1", no data is transferred to the 8/16-bit composite timer 10/11 data register.

There is an exception. With the F3 bit to F0 bit in the T10CR0/T11CR0 register having been set to "1001_B", even though the BF bit is set to "1", the "H" pulse measurement result is transferred to the 8/16-bit composite timer 10/11 data register, while the cycle measurement result is not transferred to the 8/16-bit composite timer 10/11 data register. Therefore, in order to perform cycle measurement, the "H" pulse measurement result must be read before a cycle is completed. In addition, the result of "H" pulse measurement and that of cycle measurement are lost if they are not read before the completion of the next "H" pulse.

When reading the 8/16-bit composite timer 10/11 data register, ensure that the BF bit is not cleared accidentally.

If new data is written to the 8/16-bit composite timer 10/11 data register, the stored measurement data is replaced with the new data. Therefore, do not write data to the register. In 16-bit operation, write the upper timer data to T11DR and lower timer data to T10DR, and read T11DR first and then T10DR.

● Input capture function

The 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is used to read input capture results. When an edge specified is detected, the counter value is transferred to the 8/16-bit composite timer 10/11 data register.

If new data is written to the 8/16-bit composite timer 10/11 data register, the stored measurement data is replaced with the new data. Therefore, do not write data to the register. In 16-bit operation, write the upper timer data to T11DR and lower timer data to T10DR, and read T11DR first and then T10DR.

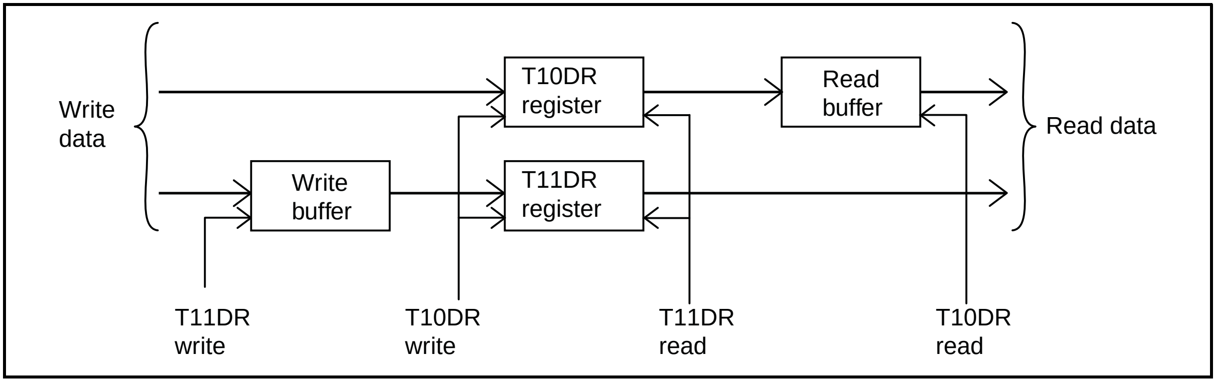
● Read and write operations

Read and write operations of T10DR and T11DR are performed in the following manner in 16-bit operation or when the PWM timer function (variable-cycle) is selected.

- Read from T11DR: In addition to the read access to T11DR, the value of T10DR is also stored in the internal read buffer at the same time.
- Read from T10DR: The internal read buffer is read.
- Write to T11DR: Data is written to the internal write buffer.
- Write to T10DR: In addition to the write access to T10DR, the value of the internal write buffer is stored in T11DR at the same time.

Figure 18.5-12 shows the T10DR and T11DR registers read from and written to during 16-bit operation.

Figure 18.5-12 Read and write operations of T10DR and T11DR registers during 16-bit operation



18.6 Interrupts of 8/16-bit Composite Timer

The 8/16-bit composite timer generates the following types of interrupts. An interrupt number and an interrupt vector are assigned to each type of interrupts.

- **Timer 00 interrupt**
- **Timer 01 interrupt**
- **Timer 10 interrupt**
- **Timer 11 interrupt**

■ Timer 00 Interrupt

Table 18.6-1 shows the timer 00 interrupt and its sources.

Table 18.6-1 Timer 00 Interrupt

Item	Description		
Interrupt generating condition	Comparison match in the interval timer operation or the PWM timer operation (variable-cycle mode)	Overflow in the PWC timer operation or the input capture operation	Completion of measurement in the PWC timer operation or edge detection in the input capture operation
Interrupt flag	T00CR1:IF	T00CR1:IF	T00CR1:IR
Interrupt enable	T00CR1:IE and T00CR0:IFE	T00CR1:IE and T00CR0:IFE	T00CR1:IE

■ Timer 01 Interrupt

Table 18.6-2 shows the timer 01 interrupt and its sources.

Table 18.6-2 Timer 01 Interrupt

Item	Description		
Interrupt generating condition	Comparison match in the interval timer operation or the PWM timer operation (variable-cycle mode), except in 16-bit operation	Overflow in the PWC timer operation or the input capture operation, except in 16-bit operation	Completion of measurement in the PWC timer operation or edge detection in the input capture operation, except in 16-bit operation
Interrupt flag	T01CR1:IF	T01CR1:IF	T01CR1:IR
Interrupt enable	T01CR1:IE and T01CR0:IFE	T01CR1:IE and T01CR0:IFE	T01CR1:IE

■ Timer 10 Interrupt

Table 18.6-3 shows the timer 10 interrupt and its sources.

Table 18.6-3 Timer 10 Interrupt

Item	Description		
Interrupt generating condition	Comparison match in the interval timer operation or the PWM timer operation (variable-cycle mode)	Overflow in the PWC timer operation or the input capture operation	Completion of measurement in the PWC timer operation or edge detection in the input capture operation
Interrupt flag	T10CR1:IF	T10CR1:IF	T10CR1:IR
Interrupt enable	T10CR1:IE and T10CR0:IFE	T10CR1:IE and T10CR0:IFE	T10CR1:IE

■ Timer 11 Interrupt

Table 18.6-4 shows the timer 11 interrupt and its sources.

Table 18.6-4 Timer 11 Interrupt

Item	Description		
Interrupt generating condition	Comparison match in the interval timer operation or the PWM timer operation (variable-cycle mode), except in 16-bit operation	Overflow in the PWC timer operation or the input capture operation, except in 16-bit operation	Completion of measurement in the PWC timer operation or edge detection in the input capture operation, except in 16-bit operation
Interrupt flag	T11CR1:IF	T11CR1:IF	T11CR1:IR
Interrupt enable	T11CR1:IE and T11CR0:IFE	T11CR1:IE and T11CR0:IFE	T11CR1:IE

■ Registers and Vector Table Addresses Related to Interrupts of 8/16-bit Composite Timer

Table 18.6-5 Registers and Vector Table Addresses Related to Interrupts of 8/16-bit Composite Timer

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
8/16-bit composite timer ch. 0 (lower) / Timer 00	IRQ05	ILR1	L05	$FFF0_H$	$FFF1_H$
8/16-bit composite timer ch. 0 (upper) / Timer 01	IRQ06	ILR1	L06	$FFEE_H$	$FFEF_H$
8/16-bit composite timer ch. 1 (lower) / Timer 10	IRQ22	ILR5	L22	$FFCE_H$	$FFCF_H$
8/16-bit composite timer ch. 1 (upper) / Timer 11	IRQ14	ILR3	L14	$FFDE_H$	$FFDF_H$

ch.: Channel

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

18.7 Operation of Interval Timer Function (One-shot Mode)

This section describes the operation of the interval timer function (one-shot mode) of the 8/16-bit composite timer.

■ Operation of Interval Timer Function (One-shot Mode) (Timer 0)

To use the interval timer function (one-shot mode), do the settings shown in Figure 18.7-1.

Figure 18.7-1 Settings of Interval Timer Function (One-shot Mode) (Timer 0)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T00CR0/T01CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	0	0
T00CR1/T01CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	❍	❍	❍	❍	❍
TMCR0	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	❍	❍	❍	❍	❍	❍
T00DR/T01DR	Sets interval time (counter compare value)							

❍: Used bit

×: Unused bit

1: Set to "1"

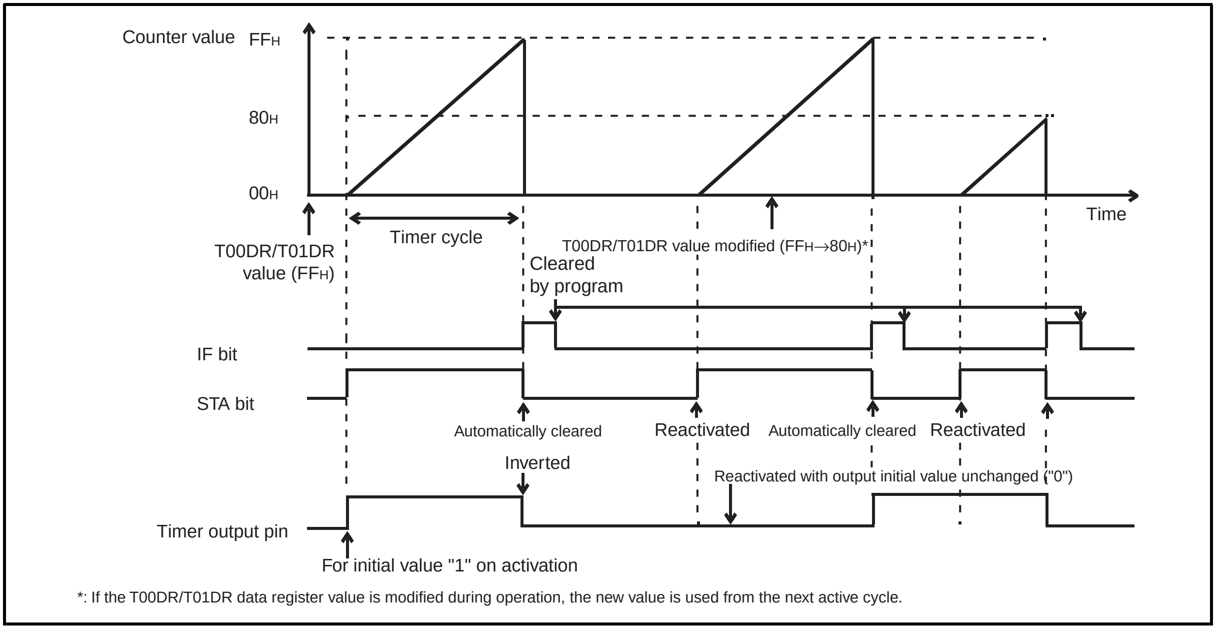
0: Set to "0"

As for the interval timer function (one-shot mode), enabling timer operation (T00CR1/T01CR1:STA = 1) causes the counter to start counting from "00_H" at the rising edge of a selected count clock signal. When the counter value matches the value of the 8/16-bit composite timer 00/01 data register (T00DR/T01DR), the timer output (TMCR0:TO0/TO1) is inverted, the interrupt flag (T00CR1/T01CR1:IF) is set to "1", the start bit (T00CR1/T01CR1:STA) is set to "0", and the counter stops counting.

The value of the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator when the counter starts counting. Do not write "00_H" to the 8/16-bit composite timer 00/01 data register.

Figure 18.7-2 shows the operation of the interval timer function in 8-bit operation (Timer 0).

Figure 18.7-2 Operation of Interval Timer Function in 8-bit Mode (One-shot Mode) (Timer 0)



*: If the T00DR/T01DR data register value is modified during operation, the new value is used from the next active cycle.

■ Operation of Interval Timer Function (One-shot Mode) (Timer 1)

To use the interval timer function (one-shot mode), do the settings shown in Figure 18.7-3.

Figure 18.7-3 Settings of Interval Timer Function (One-shot Mode) (Timer 1)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	0	0
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	❍	❍	❍	❍	❍
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	❍	❍	❍	❍	❍	❍
T10DR/T11DR	Sets interval time (counter compare value)							

❍: Used bit

×: Unused bit

1: Set to "1"

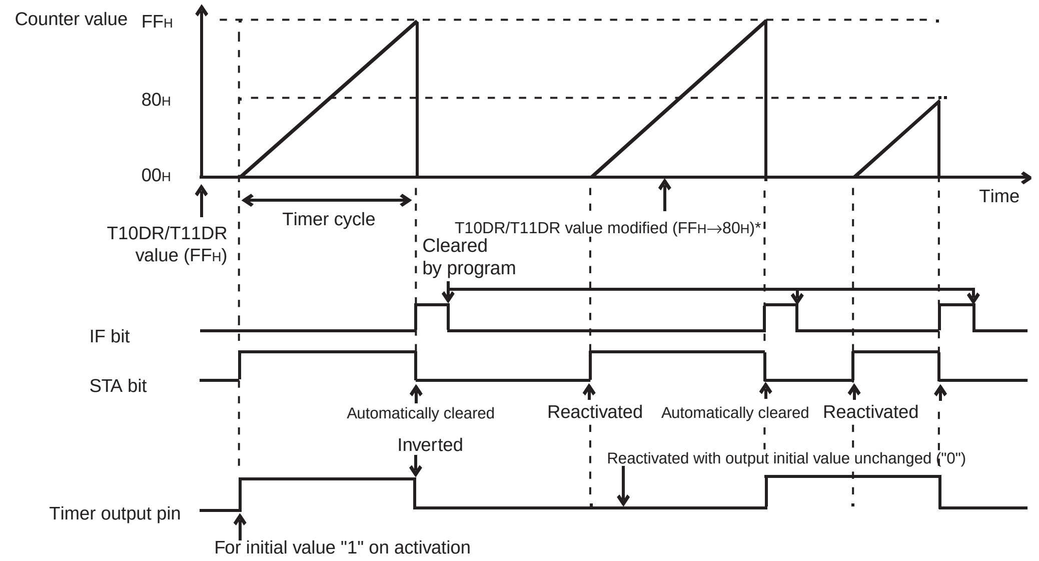
0: Set to "0"

As for the interval timer function (one-shot mode), enabling timer operation (T10CR1/ T11CR1:STA = 1) causes the counter to start counting from "00_H" at the rising edge of a selected count clock signal. When the counter value matches the value of the 8/16-bit composite timer 10/11 data register (T10DR/T11DR), the timer output (TMCR1:TO0/TO1) is inverted, the interrupt flag (T10CR1/T11CR1:IF) is set to "1", the start bit (T10CR1/ T11CR1:STA) is set to "0", and the counter stops counting.

The value of the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator when the counter starts counting. Do not write "00_H" to the 8/16-bit composite timer 10/11 data register.

Figure 18.7-4 shows the operation of the interval timer function in 8-bit operation (Timer 1).

Figure 18.7-4 Operation of Interval Timer Function in 8-bit Mode (One-shot Mode) (Timer 1)



*: If the T10DR/T11DR data register value is modified during operation, the new value is used from the next active cycle.

18.8 Operation of Interval Timer Function (Continuous Mode)

This section describes the interval timer function (continuous mode operation) of the 8/16-bit composite timer.

■ Operation of Interval Timer Function (Continuous Mode) (Timer 0)

To use the interval timer function (continuous mode), do the settings shown in Figure 18.8-1.

Figure 18.8-1 Settings for Interval Timer Function (Continuous Mode) (Timer 0)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T00CR0/T01CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	0	1
T00CR1/T01CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	×	×	❍	❍	❍
TMCR0	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	❍	❍	❍	❍	❍
T00DR/T01DR	Sets interval time (counter compare value)							

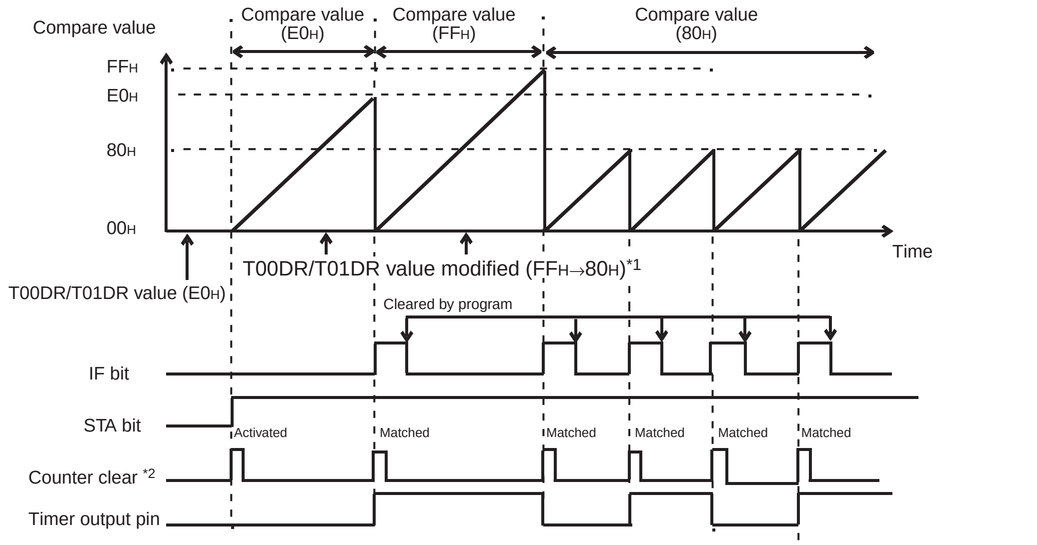
❍: Bit to be used
×: Unused bit
1: Set to "1"
0: Set to "0"

As for the interval timer function (continuous mode), enabling timer operation (T00CR1/T01CR1:STA = 1) causes the counter to start counting from "00_H" at the rising edge of a selected count clock signal. When the counter value matches the value in the 8/16-bit composite timer 00/01 data register (T00DR/T01DR), the timer output bit (TMCR0:TO0/TO1) is inverted, the interrupt flag (T00CR1/T01CR1:IF) is set to "1", and the counter returns to "00_H" and restarts counting. The timer outputs square wave as a result of this continuous operation.

The value of the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a counter value comparison match is detected. Do not write "00_H" to the 8/16-bit composite timer 00/01 data register while the counter is counting.

When the timer stops operating, the timer output bit (TMCR0:TO0/TO1) holds the last value.

Figure 18.8-2 Operation Diagram of Interval Timer Function (Continuous Mode) (Timer 0)



*1: If the T00DR/T01DR data register value is modified during operation, the new value is used from the next active cycle.

*2: The counter is cleared and the data register settings are loaded into the comparison data latch whenever a match is detected during operation.

■ Operation of Interval Timer Function (Continuous Mode) (Timer 1)

To use the interval timer function (continuous mode), do the settings shown in Figure 18.8-3.

Figure 18.8-3 Settings for Interval Timer Function (Continuous Mode) (Timer 1)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	0	1
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	×	×	❍	❍	❍
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	❍	❍	❍	❍	❍
T10DR/T11DR	Sets interval time (counter compare value)							

❍: Bit to be used

×: Unused bit

1: Set to "1"

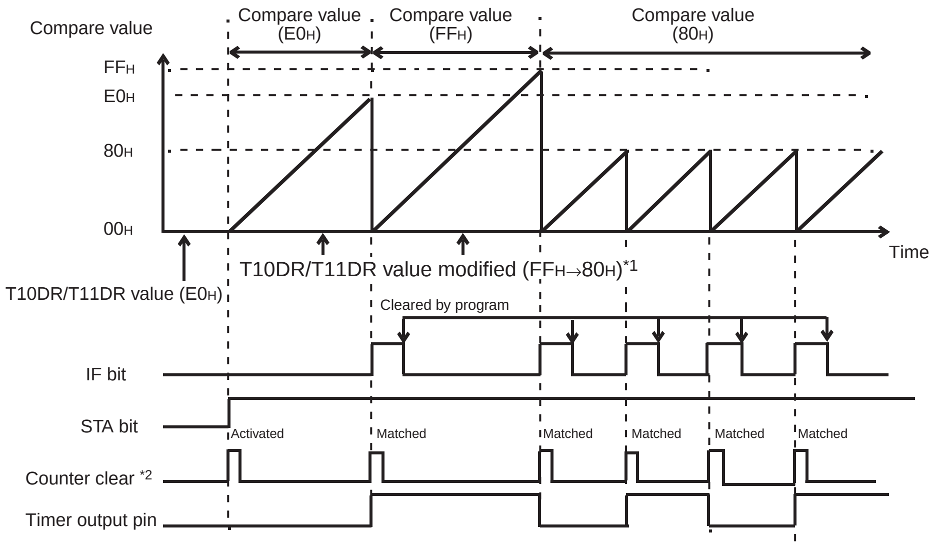
0: Set to "0"

As for the interval timer function (continuous mode), enabling timer operation (T10CR1/T11CR1:STA = 1) causes the counter to start counting from "00_H" at the rising edge of a selected count clock signal. When the counter value matches the value in the 8/16-bit composite timer 10/11 data register (T10DR/T11DR), the timer output bit (TMCR1:TO0/TO1) is inverted, the interrupt flag (T10CR1/T11CR1:IF) is set to "1", and the counter returns to "00_H" and restarts counting. The timer outputs square wave as a result of this continuous operation.

The value of the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a counter value comparison match is detected. Do not write "00_H" to the 8/16-bit composite timer 10/11 data register while the counter is counting.

When the timer stops operating, the timer output bit (TMCR1:TO0/TO1) holds the last value.

Figure 18.8-4 Operation Diagram of Interval Timer Function (Continuous Mode) (Timer 1)



*1: If the T10DR/T11DR data register value is modified during operation, the new value is used from the next active cycle.

*2: The counter is cleared and the data register settings are loaded into the comparison data latch whenever a match is detected during operation.

18.9 Operation of Interval Timer Function (Free-run Mode)

This section describes the operation of the interval timer function (free-run mode) of the 8/16-bit composite timer.

■ Operation of Interval Timer Function (Free-run Mode) (Timer 0)

To use the interval timer function (free-run mode), do the settings shown in Figure 18.9-1.

Figure 18.9-1 Settings for Interval Timer Function (Free-run Mode) (Timer 0)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T00CR0/T01CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	1	0
T00CR1/T01CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	×	×	❍	❍	❍
TMCR0	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	❍	❍	❍	❍	❍
T00DR/T01DR	Sets interval time (counter compare value)							

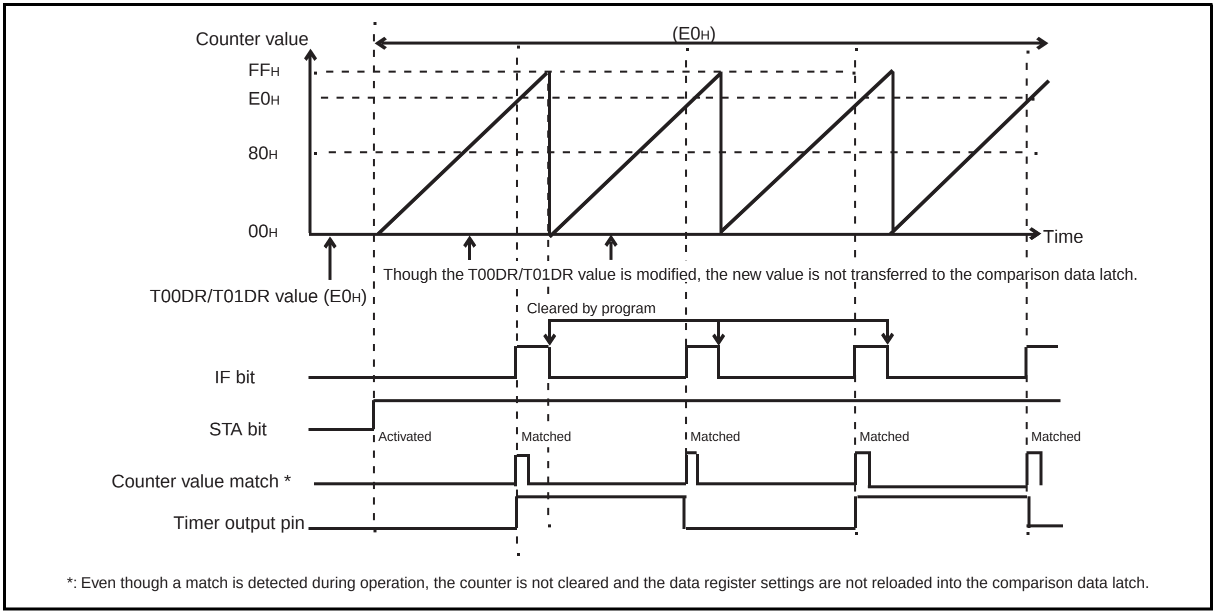
❍: Bit to be used
×: Unused bit
1: Set to "1"
0: Set to "0"

As for the interval timer function (free-run mode), enabling timer operation (T00CR1/ T00CR1:STA = 1) causes the counter to start counting from "00_H" at the rising edge of a selected count clock signal. When the counter value matches the value in the 8/16-bit composite timer 00/01 data register (T00DR/T01DR), the timer output bit (TMCR0:TO0/TO1) is inverted and the interrupt flag (T00CR1/T01CR1:IF) is set to "1". If the counter continues to count with the above settings and then reaches "FF_H", it returns to "00_H" and restarts counting. The timer outputs square wave as a result of this continuous operation.

The value of the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a counter value comparison match is detected. Do not write "00_H" to the 8/16-bit composite timer 00/01 data register.

When the timer stops operation, the timer output bit (TMCR0:TO0/TO1) holds the last value.

Figure 18.9-2 Operation Diagram of Interval Timer Function (Free-run Mode) (Timer 0)



*: Even though a match is detected during operation, the counter is not cleared and the data register settings are not reloaded into the comparison data latch.

■ Operation of Interval Timer Function (Free-run Mode) (Timer 1)

To use the interval timer function (free-run mode), do the settings shown in Figure 18.9-3.

Figure 18.9-3 Settings for Interval Timer Function (Free-run Mode) (Timer 1)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	1	0
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	×	×	❍	❍	❍
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	❍	❍	❍	❍	❍
T10DR/T11DR	Sets interval time (counter compare value)							

❍: Bit to be used

×: Unused bit

1: Set to "1"

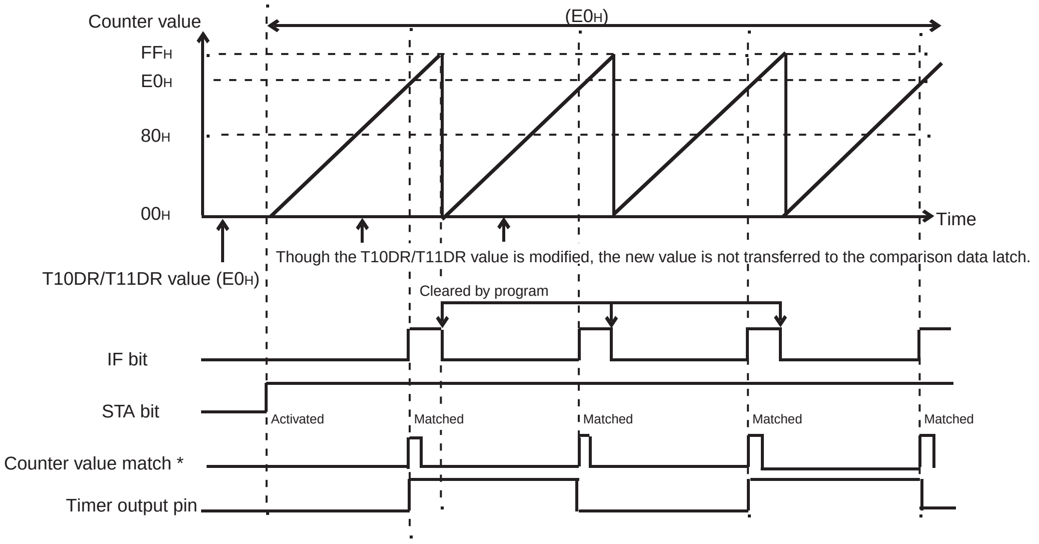
0: Set to "0"

As for the interval timer function (free-run mode), enabling timer operation (T10CR1/T11CR1:STA = 1) causes the counter to start counting from "00_H" at the rising edge of a selected count clock signal. When the counter value matches the value in the 8/16-bit composite timer 10/11 data register (T10DR/T11DR), the timer output bit (TMCR1:TO0/TO1) is inverted and the interrupt flag (T10CR1/T11CR1:IF) is set to "1". If the counter continues to count with the above settings and then reaches "FF_H", it returns to "00_H" and restarts counting. The timer outputs square wave as a result of this continuous operation.

The value of the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a counter value comparison match is detected. Do not write "00_H" to the 8/16-bit composite timer 10/11 data register.

When the timer stops operation, the timer output bit (TMCR1:TO0/TO1) holds the last value.

Figure 18.9-4 Operation Diagram of Interval Timer Function (Free-run Mode) (Timer 1)



*: Even though a match is detected during operation, the counter is not cleared and the data register settings are not reloaded into the comparison data latch.

18.10 Operation of PWM Timer Function (Fixed-cycle Mode)

This section describes the operation of the PWM timer function (fixed-cycle mode) of the 8/16-bit composite timer.

■ Operation of PWM Timer Function (Fixed-cycle Mode) (Timer 0)

To use the PWM timer function (fixed-cycle mode), do the settings shown in Figure 18.10-1.

Figure 18.10-1 Settings for PWM Timer Function (Fixed-cycle Mode) (Timer 0)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T00CR0/T01CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	1	1
T00CR1/T01CR1	STA	HO	IE	IR	BF	IF	SO	OE
	❍	❍	×	×	×	×	×	❍
TMCR0	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	❍	❍	❍	❍	❍
T00DR/T01DR	Sets "H" pulse width (compare value)							

❍: Bit to be used
×: Unused bit
1: Set to "1"
0: Set to "0"

As for the PWM timer function (fixed-cycle mode), PWM signal that has a fixed cycle and variable "H" pulse width is output from the timer output pin (TO00/TO01). The cycle is fixed at "FF$_H$" in 8-bit operation or "FFFF$_H$" in 16-bit operation. The time is determined by the count clock selected. The "H" pulse width is specified by the value in the 8/16-bit composite timer 00/01 data register (T00DR/T01DR).

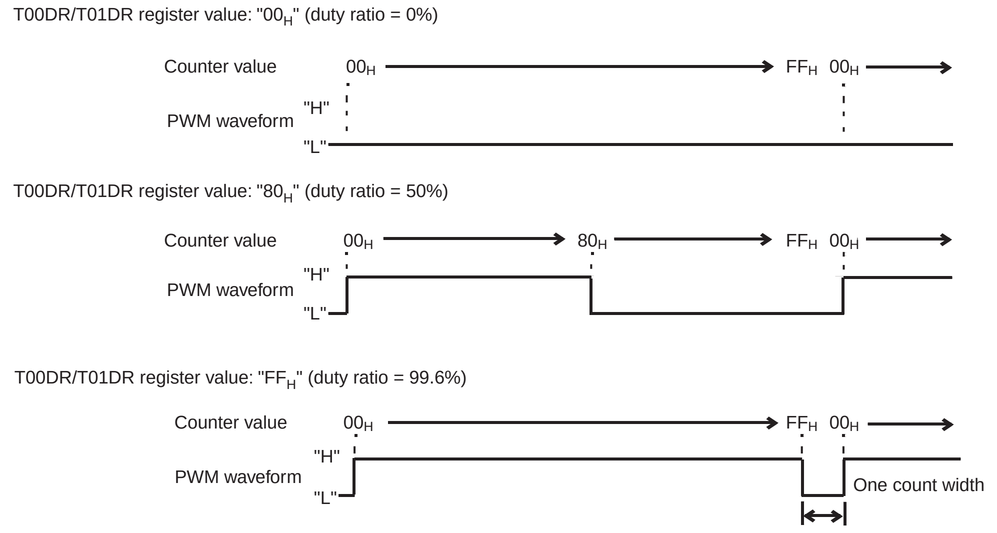
This function has no effect on the interrupt flag (T00CR1/T01CR1:IF). Since each cycle always starts with "H" pulse output, the timer output initial value setting bit (T00CR1/T01CR1:SO) has no effect on operation.

The value of the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a counter value comparison match is detected.

When the timer stops operation, the timer output bit (TMCR0:TO0/TO1) holds the last value.

The "H" pulse is one count clock shorter than the setting value in the output waveform immediately after activation of the timer (write "1" to the STA bit), the "H" pulse is one count clock shorter than the value set in the T00DR/T01DR register.

Figure 18.10-2 Operation Diagram of PWM Timer Function (Fixed-cycle Mode) (Timer 0)


T00DR/T01DR register value: "00H" (duty ratio = 0%)
Counter value
00H
FFH
00H
PWM waveform
"H"
"L"
T00DR/T01DR register value: "80H" (duty ratio = 50%)
Counter value
00H
80H
FFH
00H
PWM waveform
"H"
"L"
T00DR/T01DR register value: "FFH" (duty ratio = 99.6%)
Counter value
00H
FFH
00H
PWM waveform
"H"
"L"
One count width

Note: When the PWM function has been selected, the timer output pin holds the level at the point when the counter stops (T00CR0/T01CR0:STA = 0).

■ Operation of PWM Timer Function (Fixed-cycle Mode) (Timer 1)

To use the PWM timer function (fixed-cycle mode), do the settings shown in Figure 18.10-3.

Figure 18.10-3 Settings for PWM Timer Function (Fixed-cycle Mode) (Timer 1)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	0	1	1
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	❍	❍	×	×	×	×	×	❍
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	❍	❍	❍	❍	❍
T10DR/T11DR	Sets "H" pulse width (compare value)							

❍: Bit to be used

×: Unused bit

1: Set to "1"

0: Set to "0"

As for the PWM timer function (fixed-cycle mode), PWM signal that has a fixed cycle and variable "H" pulse width is output from the timer output pin (TO10/TO11). The cycle is fixed at "$\mathrm{FF_H}$" in 8-bit operation or "$\mathrm{FFFF_H}$" in 16-bit operation. The time is determined by the count clock selected. The "H" pulse width is specified by the value in the 8/16-bit composite timer 10/11 data register (T10DR/T11DR).

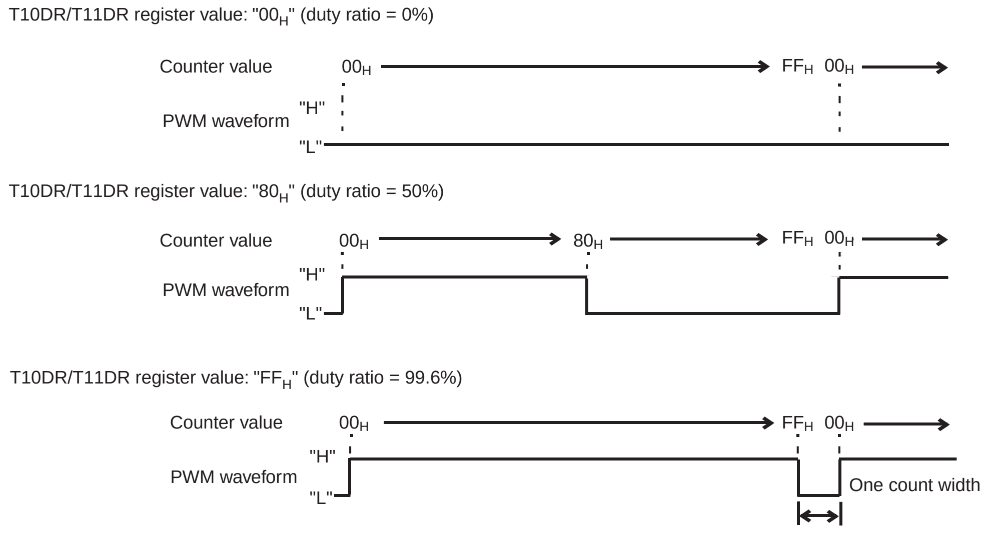
This function has no effect on the interrupt flag (T10CR1/T11CR1:IF). Since each cycle always starts with "H" pulse output, the timer output initial value setting bit (T10CR1/T11CR1:SO) has no effect on operation.

The value of the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a counter value comparison match is detected.

When the timer stops operation, the timer output bit (TMCR1:TO0/TO1) holds the last value.

The "H" pulse is one count clock shorter than the setting value in the output waveform immediately after activation of the timer (write "1" to the STA bit), the "H" pulse is one count clock shorter than the value set in the T10DR/T11DR register.

Figure 18.10-4 Operation Diagram of PWM Timer Function (Fixed-cycle Mode) (Timer 1)


T10DR/T11DR register value: "00H" (duty ratio = 0%)
Counter value
00H
FFH
00H
PWM waveform
"H"
"L"
T10DR/T11DR register value: "80H" (duty ratio = 50%)
Counter value
00H
80H
FFH
00H
PWM waveform
"H"
"L"
T10DR/T11DR register value: "FFH" (duty ratio = 99.6%)
Counter value
00H
FFH
00H
PWM waveform
"H"
"L"
One count width

Note: When the PWM function has been selected, the timer output pin holds the level at the point when the counter stops (T10CR0/T11CR0:STA = 0).

18.11 Operation of PWM Timer Function (Variable-cycle Mode)

This section describes the operation of the PWM timer function (variable-cycle mode) of the 8/16-bit composite timer.

■ Operation of PWM Timer Function (Variable-cycle Mode) (Timer 0)

To use the PWM timer function (variable-cycle mode), do the settings shown in Figure 18.11-1.

Figure 18.11-1 Settings for PWM Timer Function (Variable-cycle Mode) (Timer 0)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T00CR0/T01CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	1	0	0
T00CR1/T01CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	×	×	❍	×	×
TMCR0	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	×	❍	❍	❍	❍
T00DR	Sets "L" pulse width (compare value)							
T01DR	Sets the cycle of PWM waveform (compare value)							

❍: Bit to be used
×: Unused bit
1: Set to "1"
0: Set to "0"

As for the PWM timer function (variable-cycle mode), both timers 00 and 01 are used. PWM signal of any cycle and of any duty is output from the timer output pin (TO00). The cycle is specified by the 8/16-bit composite timer 01 data register (T01DR), and the "L" pulse width is specified by the 8/16-bit composite timer 00 data register (T00DR).

Since both the 8-bit counters are used for this function, the composite timer cannot form a 16-bit counter.

Enabling timer operation (by setting either T00CR1:STA = 1 or T01CR1:STA = 1) sets the mode bit (TMCR0:MOD) to "0". As the first cycle always begins with "L" pulse output, the timer initial value setting bit (T00CR1/T01CR1:SO) has no effect on operation.

An interrupt flag (T00CR1/T01CR1:IF) is set when the 8-bit counter corresponding to that interrupt flag matches the value in its corresponding 8/16-bit composite timer 00/01 data register (T00DR/T01DR).

The 8/16-bit composite timer 00/01 data register value is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a comparison match with each counter value is detected.

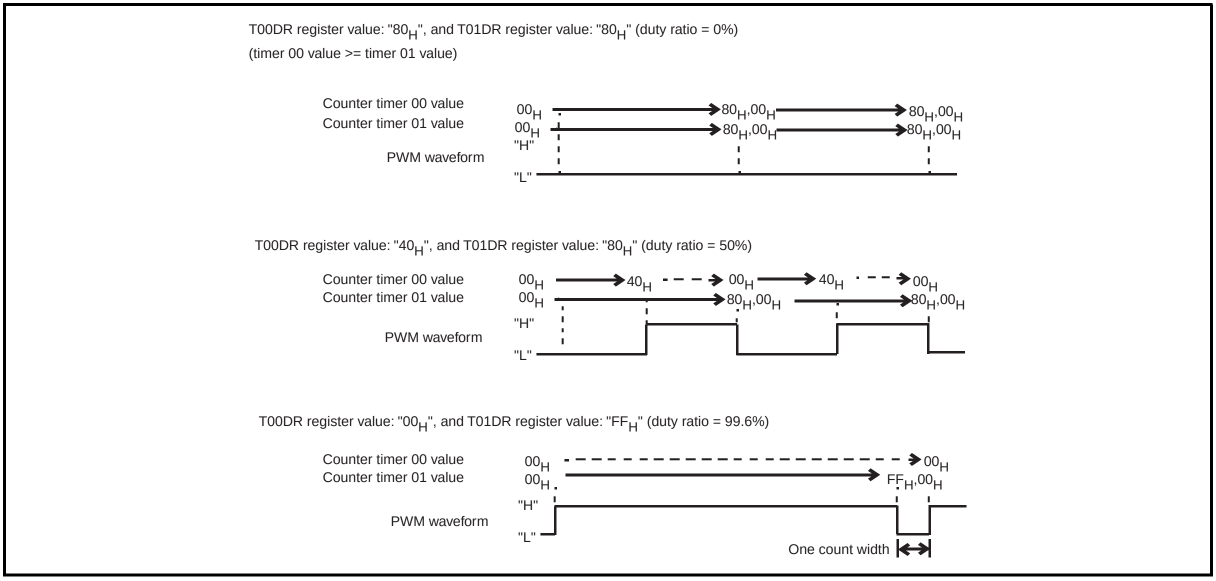
"H" is not output when the "L" pulse width setting value is greater than the cycle setting value.

The count clock must be selected for both of timers 00 and 01. Selecting different count clocks for the two timers is prohibited.

When the timer stops operating, the timer output bit (TMCR0:TO0) holds the last output value.

If the 8/16-bit composite timer 00/01 data register is modified during operation, the data written will become valid from the cycle immediately after the detection of a synchronous match.

Figure 18.11-2 Operation Diagram of PWM Timer Function (Variable-cycle Mode) (Timer 0)



■ Operation of PWM Timer Function (Variable-cycle Mode) (Timer 1)

To use the PWM timer function (variable-cycle mode), do the settings shown in Figure 18.11-3.

Figure 18.11-3 Settings for PWM Timer Function (Variable-cycle Mode) (Timer 1)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	0	1	0	0
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	×	×	❍	×	×
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	×	❍	❍	❍	❍
T10DR	Sets "L" pulse width (compare value)							
T11DR	Sets the cycle of PWM waveform (compare value)							

❍: Bit to be used
×: Unused bit
1: Set to "1"
0: Set to "0"

As for the PWM timer function (variable-cycle mode), both timers 10 and 11 are used. PWM signal of any cycle and of any duty is output from the timer output pin (TO10). The cycle is specified by the 8/16-bit composite timer 11 data register (T11DR), and the "L" pulse width is specified by the 8/16-bit composite timer 10 data register (T10DR).

Since both the 8-bit counters are used for this function, the composite timer cannot form a 16-bit counter.

Enabling timer operation (by setting either T10CR1:STA = 1 or T11CR1:STA = 1) sets the mode bit (TMCR1:MOD) to "0". As the first cycle always begins with "L" pulse output, the timer initial value setting bit (T10CR1/T11CR1:SO) has no effect on operation.

An interrupt flag (T10CR1/T11CR1:IF) is set when the 8-bit counter corresponding to that interrupt flag matches the value in its corresponding 8/16-bit composite timer 10/11 data register (T10DR/T11DR).

The 8/16-bit composite timer 10/11 data register value is transferred to the temporary storage latch (comparison data storage latch) in the comparator either when the counter starts counting or when a comparison match with each counter value is detected.

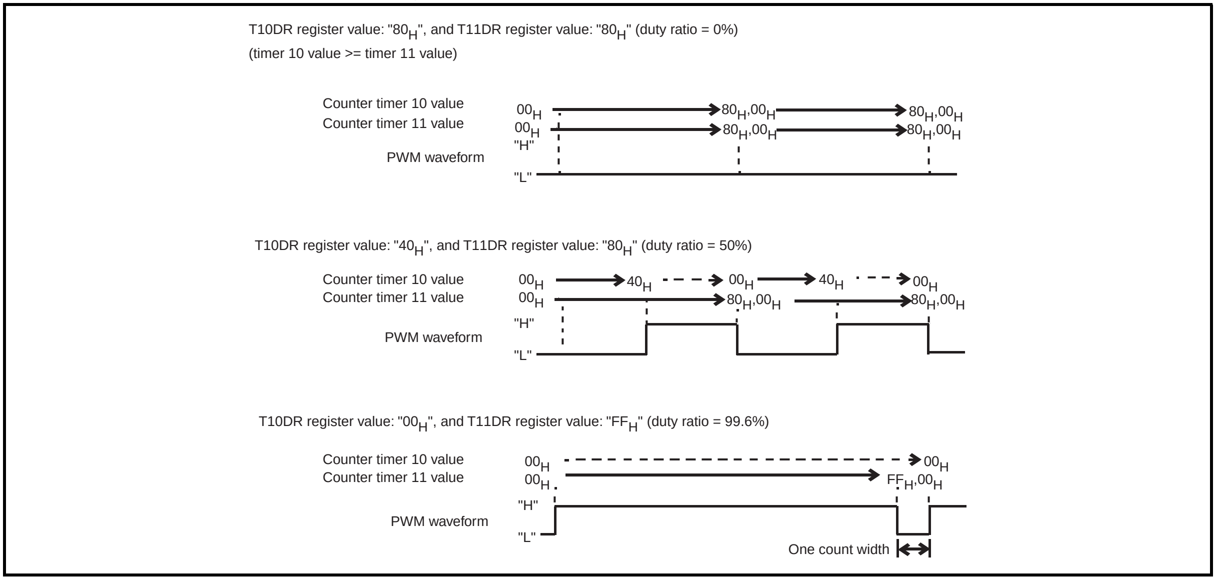
"H" is not output when the "L" pulse width setting value is greater than the cycle setting value.

The count clock must be selected for both of timers 10 and 11. Selecting different count clocks for the two timers is prohibited.

When the timer stops operating, the timer output bit (TMCR1:TO0) holds the last output value.

If the 8/16-bit composite timer 10/11 data register is modified during operation, the data written will become valid from the cycle immediately after the detection of a synchronous match.

Figure 18.11-4 Operation Diagram of PWM Timer Function (Variable-cycle Mode) (Timer 1)



18.12 Operation of PWC Timer Function

This section describes the operation of the PWC timer function of the 8/16-bit composite timer.

■ Operation of PWC Timer Function (Timer 0)

To use the PWC timer function, do the settings shown in Figure 18.12-1.

Figure 18.12-1 Settings for PWC Timer Function (Timer 0)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T00CR0/T01CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	❍	❍	❍	❍
T00CR1/T01CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	❍	❍	❍	❍	×
TMCR0	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	❍	❍	❍	❍	❍	❍
T00DR/T01DR	Holds pulse width measurement value							

❍: Bit to be used

×: Unused bit

1: Set to "1"

When the PWC timer function is selected, the width and cycle of an external input pulse can be measured. The edges at which counting starts and ends are selected by the timer operating mode select bits (T00CR0/T01CR0:F3, F2, F1, F0).

In the operation of this function, the counter starts counting from "00_H" immediately after a specified count start edge of an external input signal is detected. Upon the detection of a specified count end edge, the count value is transferred to the 8/16-bit composite timer 00/01 data register (T00DR/T01DR), and the interrupt flag (T00CR1/T01CR1:IR) and the buffer full flag (T00CR1/T01CR1:BF) are set to "1". The buffer full flag is set to "0" when the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) is read.

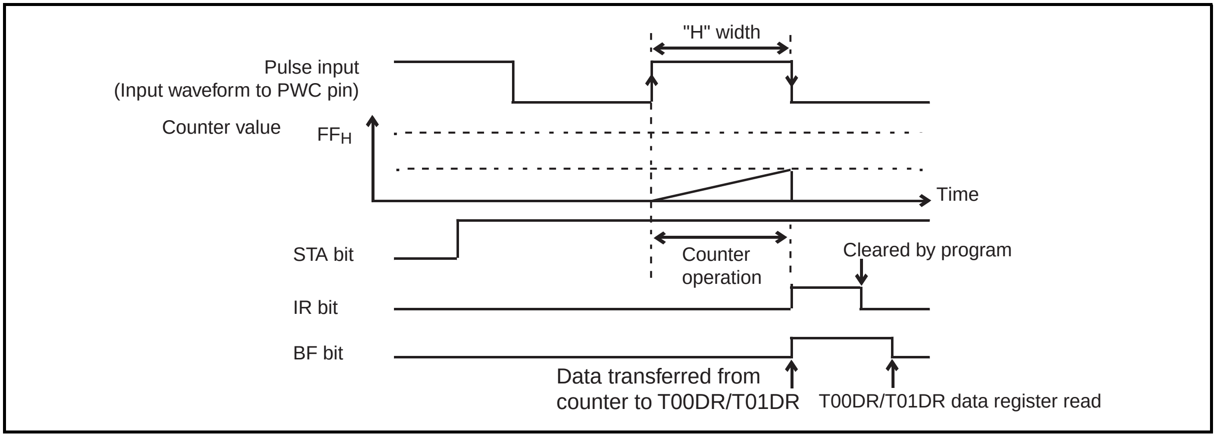
If the buffer full flag is set to "1", the 8/16-bit composite timer 00/01 data register holds data. Even if the next edge is detected during that time, the next measurement result is lost since the count value has not been transferred to the 8/16-bit composite timer 00/01 data register.

There is an exception. With the F3 bit to F0 bit in the T00CR0/T01CR0 register having been set to "1001_B", even though the BF bit is set to "1", the "H" pulse measurement result is transferred to the 8/16-bit composite timer 00/01 data register, while the cycle measurement result is not transferred to the 8/16-bit composite timer 00/01 data register. Therefore, in order to perform cycle measurement, the "H" pulse measurement result must be read before a cycle is completed. In addition, the result of "H" pulse measurement and that of cycle measurement are lost if they are not read before the completion of the next "H" pulse.

To measure the time exceeding the range of the counter, software can be used to count the number of counter overflows. When the counter overflows, the interrupt flag (T00CR1/T01CR1:IF) is set to "1". The interrupt service routine can therefore be used to count the number of overflows. In addition, the timer output is inverted due to the overflow. The timer output initial value can be set by the timer output initial value bit (T00CR1/T01CR1:SO).

When the timer stops operating, the timer output bit (TMCR0:TO1/TO0) holds the last value.

Figure 18.12-2 Operation Diagram of PWC Timer (Example of H-pulse Width Measurement) (Timer 0)



■ Operation of PWC Timer Function (Timer 1)

To use the PWC timer function, do the settings shown in Figure 18.12-3.

Figure 18.12-3 Settings for PWC Timer Function (Timer 1)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	❍	❍	❍	❍
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	❍	❍	❍	❍	×
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	❍	❍	❍	❍	❍	❍
T10DR/T11DR	Holds pulse width measurement value							

❍: Bit to be used

×: Unused bit

1: Set to "1"

When the PWC timer function is selected, the width and cycle of an external input pulse can be measured. The edges at which counting starts and ends are selected by the timer operating mode select bits (T10CR0/T11CR0:F3, F2, F1, F0).

In the operation of this function, the counter starts counting from "00_H" immediately after a specified count start edge of an external input signal is detected. Upon the detection of a specified count end edge, the count value is transferred to the 8/16-bit composite timer 10/11 data register (T10DR/T11DR), and the interrupt flag (T10CR1/T11CR1:IR) and the buffer full flag (T10CR1/T11CR1:BF) are set to "1". The buffer full flag is set to "0" when the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) is read.

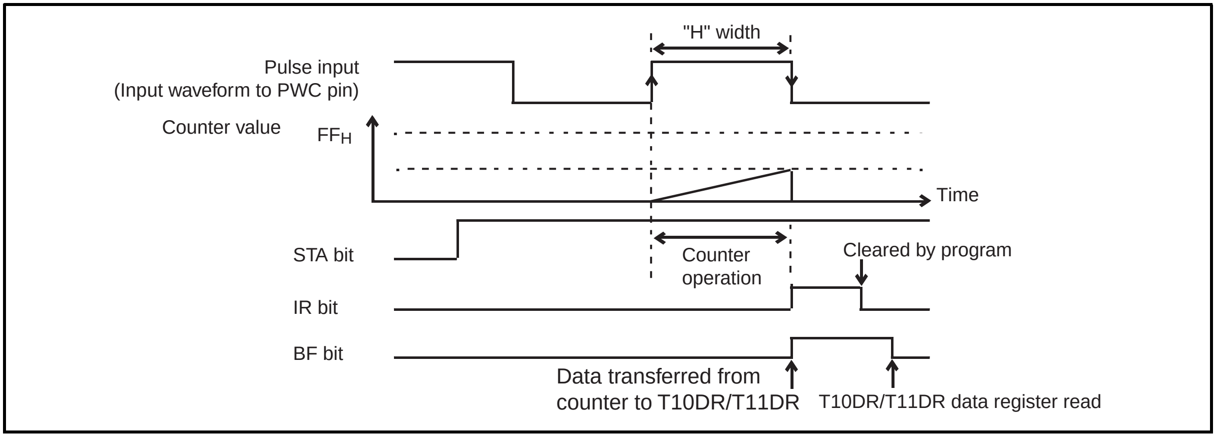
If the buffer full flag is set to "1", the 8/16-bit composite timer 10/11 data register holds data. Even if the next edge is detected during that time, the next measurement result is lost since the count value has not been transferred to the 8/16-bit composite timer 10/11 data register.

There is an exception. With the F3 bit to F0 bit in the T10CR0/T11CR0 register having been set to "1001_B", even though the BF bit is set to "1", the "H" pulse measurement result is transferred to the 8/16-bit composite timer 10/11 data register, while the cycle measurement result is not transferred to the 8/16-bit composite timer 10/11 data register. Therefore, in order to perform cycle measurement, the "H" pulse measurement result must be read before a cycle is completed. In addition, the result of "H" pulse measurement and that of cycle measurement are lost if they are not read before the completion of the next "H" pulse.

To measure the time exceeding the range of the counter, software can be used to count the number of counter overflows. When the counter overflows, the interrupt flag (T10CR1/T11CR1:IF) is set to "1". The interrupt service routine can therefore be used to count the number of overflows. In addition, the timer output is inverted due to the overflow. The timer output initial value can be set by the timer output initial value bit (T10CR1/T11CR1:SO).

When the timer stops operating, the timer output bit (TMCR1:TO1/TO0) holds the last value.

Figure 18.12-4 Operation Diagram of PWC Timer (Example of H-pulse Width Measurement) (Timer 1)



18.13 Operation of Input Capture Function

This section describes the operation of the input capture function of the 8/16-bit composite timer.

■ Operation of Input Capture Function (Timer 0)

To use the input capture function, do the settings shown in Figure 18.13-1.

Figure 18.13-1 Settings for Input Capture Function (Timer 0)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T00CR0/T01CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	❍	❍	❍	❍
T00CR1/T01CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	❍	×	❍	×	×
TMCR0	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	×	×	❍	❍	❍	❍	❍	❍
T00DR/T01DR	Holds pulse width measurement value							

❍: Bit to be used

×: Unused bit

1: Set to "1"

When the input capture function is selected, the counter value is stored to the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) immediately after an edge of the external signal input is detected. The target edge to be detected is selected by the timer operating mode select bits (T00CR0/T01CR0:F3, F2, F1, F0).

This function is available in free-run mode and clear mode, which can be selected by the timer operating mode select bits.

In clear mode, the counter starts counting from "00_H". When an edge is detected, the counter value is transferred to the 8/16-bit composite timer 00/01 data register (T00DR/T01DR), the interrupt flag (T00CR1/T01CR1:IR) is set to "1", and the counter returns to "00_H" and restarts counting.

In free-run mode, when an edge is detected, the counter value is transferred to the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) and the interrupt flag (T00CR1/T01CR1:IR) is set to "1". In this case, the counter continues to count without being cleared.

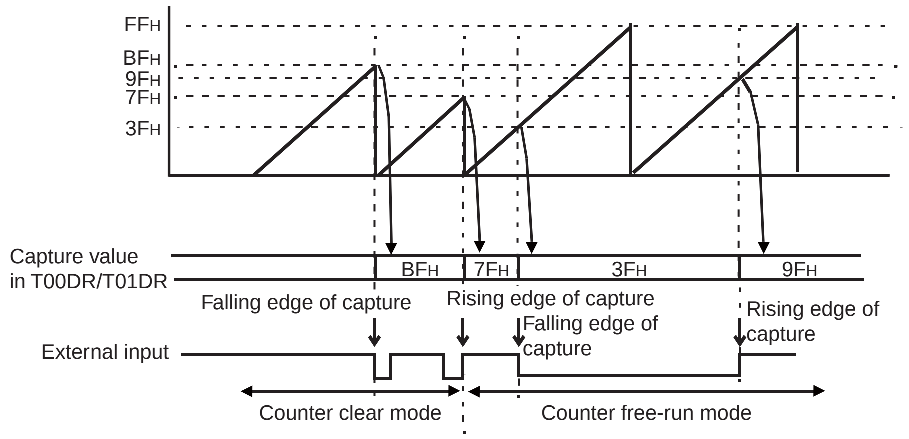
This function has no effect on the buffer full flag (T00CR1/T01CR1:BF).

To measure the time exceeding the range of the counter, software can be used to count the number of counter overflows. When the counter overflows, the interrupt flag (T00CR1/T01CR1:IF) is set to "1". The interrupt service routine can therefore be used to count the number of overflows. In addition, the timer output is inverted due to the overflow. The timer output initial value can be set by the timer output initial value bit (T00CR1/T01CR1:SO).

Note:

See "18.16 Notes on Using 8/16-bit Composite Timer" for notes on using the input capture function.

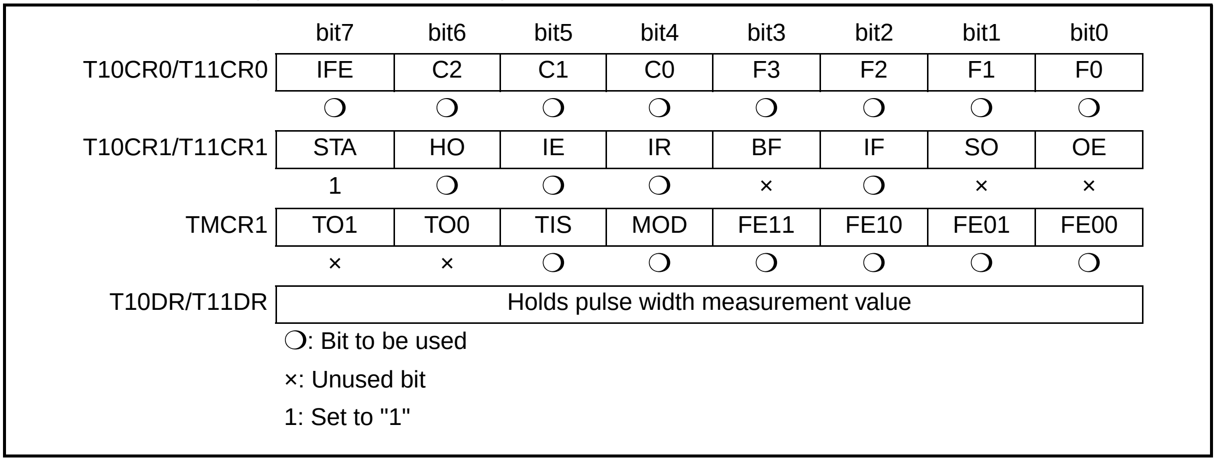
Figure 18.13-2 Operating Diagram of Input Capture Function (Timer 0)



■ Operation of Input Capture Function (Timer 1)

To use the input capture function, do the settings shown in Figure 18.13-3.

Figure 18.13-3 Settings for Input Capture Function (Timer 1)



	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	❍	❍	❍	❍	❍	❍	❍
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	❍	×	❍	×	×
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	×	×	❍	❍	❍	❍	❍	❍
T10DR/T11DR	Holds pulse width measurement value							

❍: Bit to be used

×: Unused bit

1: Set to "1"

When the input capture function is selected, the counter value is stored to the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) immediately after an edge of the external signal input is detected. The target edge to be detected is selected by the timer operating mode select bits (T10CR0/T11CR0:F3, F2, F1, F0).

This function is available in free-run mode and clear mode, which can be selected by the timer operating mode select bits.

In clear mode, the counter starts counting from "00_H". When an edge is detected, the counter value is transferred to the 8/16-bit composite timer 10/11 data register (T10DR/T11DR), the interrupt flag (T10CR1/T11CR1:IR) is set to "1", and the counter returns to "00_H" and restarts counting.

In free-run mode, when an edge is detected, the counter value is transferred to the 8/16-bit composite timer 10/11 data register (T10DR/T11DR) and the interrupt flag (T10CR1/T11CR1:IR) is set to "1". In this case, the counter continues to count without being cleared.

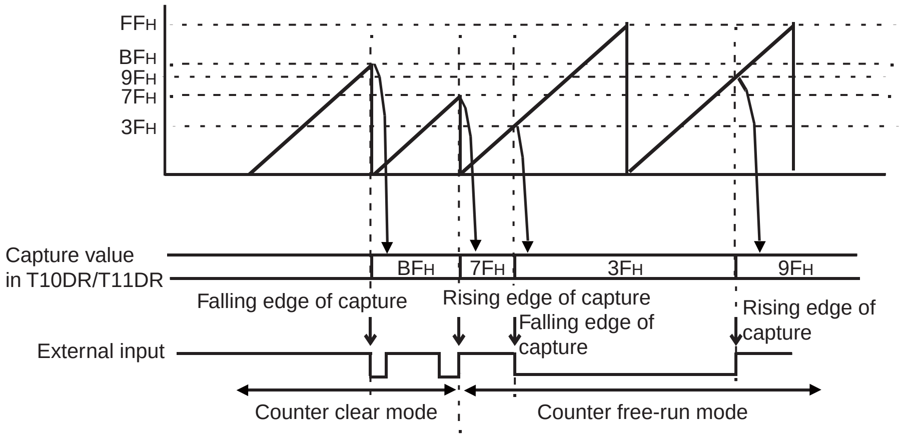
This function has no effect on the buffer full flag (T10CR1/T11CR1:BF).

To measure the time exceeding the range of the counter, software can be used to count the number of counter overflows. When the counter overflows, the interrupt flag (T10CR1/T11CR1:IF) is set to "1". The interrupt service routine can therefore be used to count the number of overflows. In addition, the timer output is inverted due to the overflow. The timer output initial value can be set by the timer output initial value bit (T10CR1/T11CR1:SO).

Note:

See "18.16 Notes on Using 8/16-bit Composite Timer" for notes on using the input capture function.

Figure 18.13-4 Operating Diagram of Input Capture Function (Timer 1)

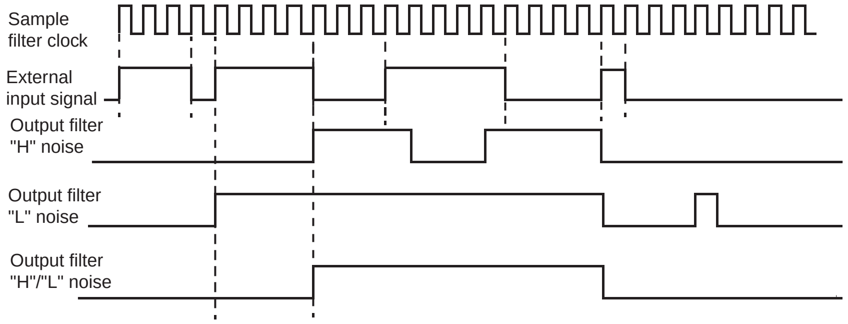


18.14 Operation of Noise Filter

This section describes the operation of the noise filter of the 8/16-bit composite timer.

When the input capture function or PWC timer function is selected, a noise filter can be used to eliminate the pulse noise of the signal from the external input pin (EC0/EC1). H-pulse noise, L-pulse noise, or H/L-pulse noise elimination can be selected by setting the FE11, FE10, FE01 and FE00 bits in the TMCR0 and TMCR1 register. The maximum pulse width that can be eliminated is three machine clock cycles. If the noise filter function is activated, the signal input will be delayed for four machine clock cycles.

Figure 18.14-1 Operation of Noise Filter



18.15 States in Each Mode during Operation

This section describes how the 8/16-bit composite timer behaves when the microcontroller transits to watch mode or stop mode or when a suspend (T00CR1/T01CR1/T10CR1/T11CR1:HO = 1) request is made during operation.

■ When Interval Timer, Input Capture, or PWC Function Is Selected

Figure 18.15-1 shows how the counter value changes when the microcontroller transits to watch mode or stop mode, or a suspend request is made during the operation of the 8/16-bit composite timer.

The counter stops operating while holding the value when the microcontroller transits to stop mode or watch mode. When the stop mode or watch mode is released by an interrupt, the counter resumes operating with the last value that it holds. Therefore, the first interval time or the initial external clock count value is incorrect. Always initialize the counter value after the microcontroller is released from stop mode or watch mode.

Figure 18.15-1 Operations of Counter in Standby Mode or in Pause (Not Serving as PWM Timer)

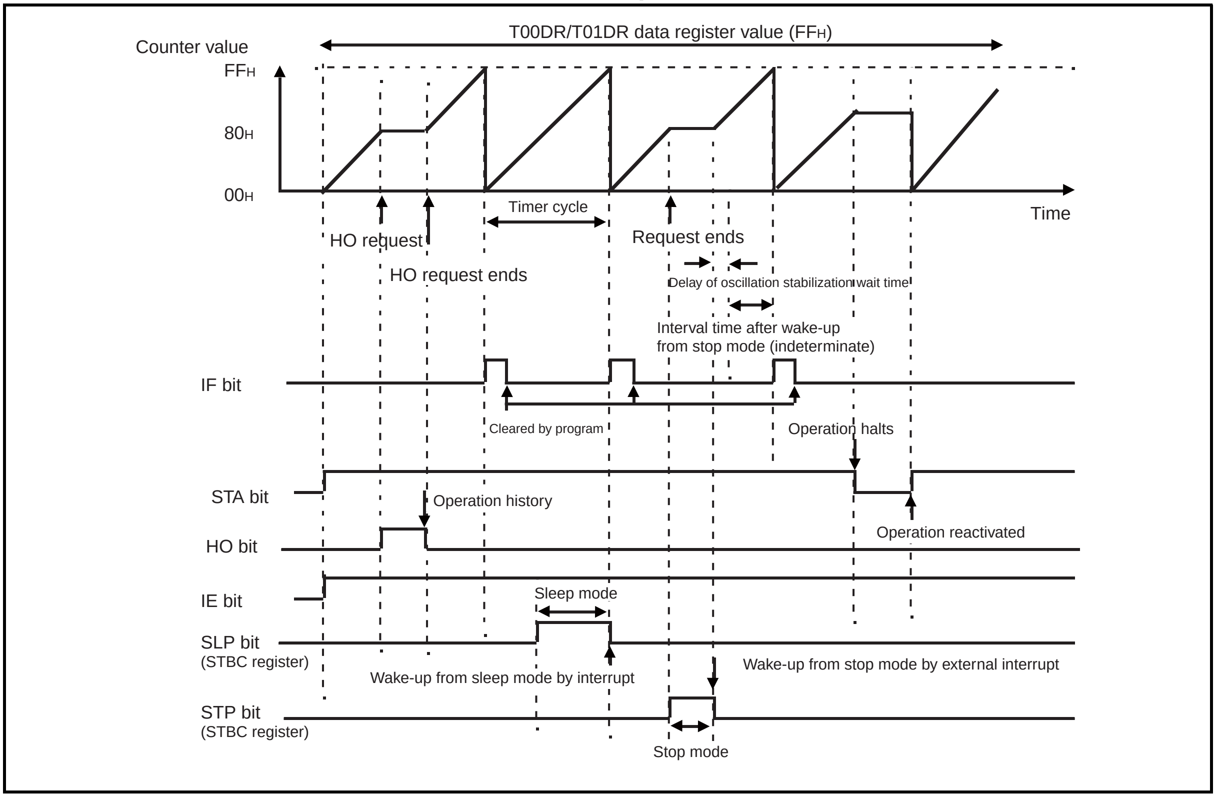
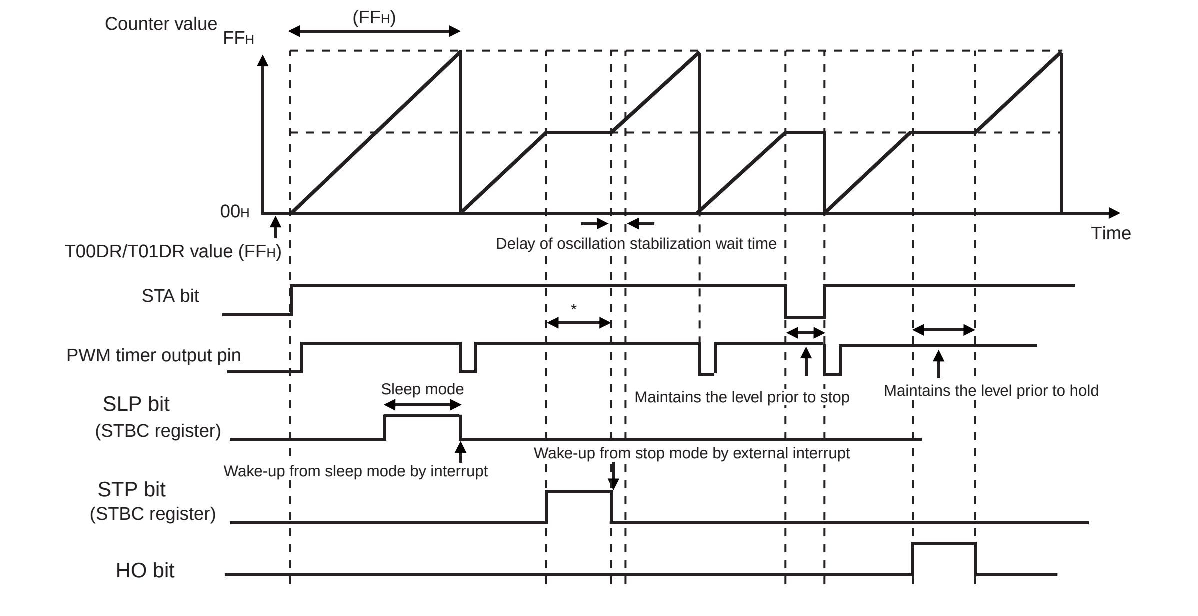


Figure 18.15-2 Operations of Counter in Standby Mode or in Pause (Serving as PWM Timer)


Counter value
FFH
(FFH)
00H
Time
T00DR/T01DR value (FFH)
Delay of oscillation stabilization wait time
STA bit
*
PWM timer output pin
Sleep mode
Maintains the level prior to stop
Maintains the level prior to hold
SLP bit
(STBC register)
Wake-up from stop mode by external interrupt
Wake-up from sleep mode by interrupt
STP bit
(STBC register)
HO bit

*: The PWM timer output maintains the value held before it enters the stop mode.

18.16 Notes on Using 8/16-bit Composite Timer

This section provides notes on using the 8/16-bit composite timer.

■ Notes on Using 8/16-bit Composite Timer

- To switch the timer function with the timer operating mode select bits (T00CR0/T01CR0/T10CR0/T11CR0:F3, F2, F1, F0), stop the timer operation first (T00CR1/T01CR1/T10CR1/T11CR1:STA = 0), then clear the interrupt flag (T00CR1/T01CR1/T10CR1/T11CR1:IF, IR), the interrupt enable bits (T00CR1/T01CR1/T10CR1/T11CR1:IE, T00CR0/T01CR0/T10CR0/T11CR0:IFE) and the buffer full flag (T00CR1/T01CR1/T10CR1/T11CR1:BF).
- In the case of using the input capture function, when both edges of the external input signal is selected as the timing at which the 8/16-bit composite timer captures a counter value (T00CR0/T01CR0/T10CR0/T11CR0:F3, F2, F1, F0 = 1100_B or 1111_B) while "H" level external input signal is being input, the first falling edge will be ignored, no counter value will be transferred to the data register (T00DR/T01DR/T10DR/T11DR), and pulse width measurement completion/edge detection flag (T00CR1/T01CR1/T10CR1/T11CR1:IR) will not be set either.
  - In counter clear mode, the counter will not be cleared at the first falling edge and no data will be transferred to the data register either. The 8/16-bit composite timer will start the input capture operation from the next rising edge.
  - In counter free-run mode, no data will be transferred to the data register at the first falling edge. The 8/16-bit composite timer will start the input capture operation from the next rising edge.
- In 8-bit operating mode (TMCR0/TMCR1:MOD = 0) of the PWM timer function (variable-cycle mode), when modifying the 8/16-bit composite timer 00/01 data register (T00DR/T01DR) during counter operation, modify T01DR first and then T00DR. The same setting sequence requirement is also applicable to the 8/16-bit composite timer 10/11 data register (T10DR/T11DR).
- Note that 8/16-bit composite timer ch. 1 is to be used when Event Counter operates in event counter mode. For details on Event Counter, see "CHAPTER 20 EVENT COUNTER".

CHAPTER 19
16-BIT RELOAD TIMER

This chapter describes the functions and operations of the 16-bit reload timer.

19.1 Overview of 16-bit Reload Timer

The 16-bit reload timer has two counter operation modes available in the following two clock modes.
The 16-bit reload timer can be used as an interval timer by generating an interrupt when an underflow occurs in the timer.

■ Operation Modes of 16-bit Reload Timer

Table 19.1-1 shows the operation modes of the 16-bit reload timer.

Table 19.1-1 Operation Modes of 16-bit Reload Timer

Clock mode	Counter operation mode	Trigger operation mode
Internal clock mode	Reload mode	Software trigger operation External trigger input operation External gate input operation
	One-shot mode	
Event count mode (external clock mode)	Reload mode	Software trigger operation
	One-shot mode	

■ Internal Clock Mode

Internal clock mode is selected when any value other than "111_B" is set in the count clock setting bits (CSL2 to CSL0) of the 16-bit reload timer control status register upper (TMCSRH0).

In internal clock mode, the following three trigger operation modes are available.

● Software trigger operation

With the count enable bit (CNTE) in the 16-bit reload timer control status register lower (TMCSRL0) set to "1", the counter starts when the software trigger bit (TRG) in the TMCSRL0 register is set to "1".

● External trigger input operation

When the count enable bit (CNTE) in the 16-bit reload timer control status register lower (TMCSRL0) is set to "1", the count will start if a valid edge (rising, falling, or both selectable) specified by the operation mode setting bits (MOD2 to MOD0) is input to the TI0 pin.

● External gate input operation

When the count enable bit (CNTE) in the 16-bit reload timer control status register lower (TMCSRL0) is set to "1", the count will start if a valid trigger input level ("L" or "H" selectable) specified by the operation mode setting bits (MOD2 to MOD0) is input to the TI0 pin.

■ Event Count Mode (External Clock Mode)

When the count clock setting bits (CSL2 to CSL0) in the 16-bit reload timer control status register upper (TMCSRH0) are set to "111_B", the count will start if a valid edge of trigger input (rising, falling, or both) specified by the operation mode setting bits (MOD2 to MOD0) is input to the TI0 pin. When an external clock is input in regular cycles, the reload timer can also be used as an interval timer.

■ Counter Operation Mode

● Reload mode

The value of the 16-bit reload timer reload register (TMRLRH0/TMRLRL0) is loaded to the 16-bit down-counter and the count continues when an underflow occurs on the 16-bit down-counter ("0000_H" → "$FFFF_H$"). Also, the interrupt request is output by an underflow, so the mode can be used as the interval timer.

● One-shot mode

An interrupt is outputted when an underflow occurs on the 16-bit down-counter.

During counter operation, the TO0 pin outputs a square waveform indicating that the counter is currently running.

19.2 Configuration of 16-bit Reload Timer

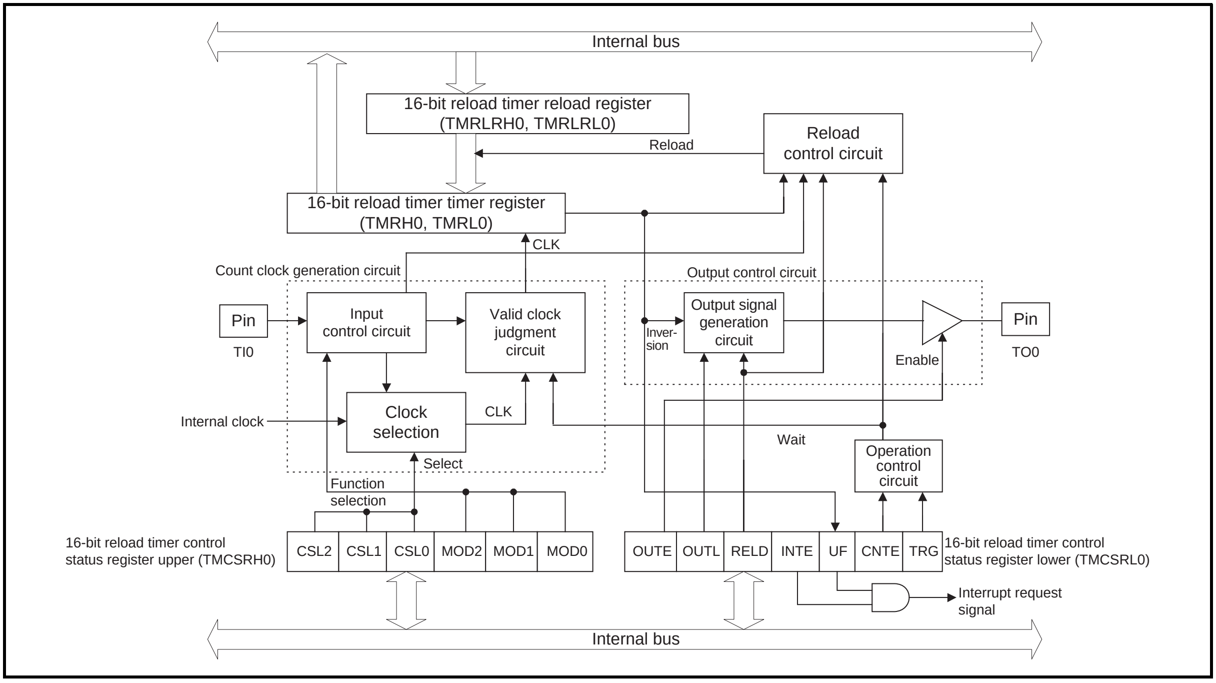
The 16-bit reload timer consists of the following blocks:

- **Count clock generation circuit**
- **Reload control circuit**
- **Output control circuit**
- **Operation control circuit**
- **16-bit reload timer timer register (TMRH0, TMRL0)**
- **16-bit reload timer reload register (TMRLRH0, TMRLRL0)**
- **16-bit reload timer control status register (TMCSRH0, TMCSRL0)**

■ Block Diagram of 16-bit Reload Timer

Figure 19.2-1 shows the block diagram of the 16-bit reload timer.

Figure 19.2-1 Block Diagram of 16-bit Reload Timer



● Count clock generation circuit

The count clock for the 16-bit reload timer is outputted from the internal clock or TI0 pin input signal.

● Reload control circuit

This circuit controls reload operation when the timer is started or an underflow occurs.

● Output control circuit

This circuit controls the inversion of TO0 pin output by an underflow of the 16-bit down-counter and the enabling and disabling of TO0 pin output.

● Operation control circuit

This circuit controls the starting and stopping of the 16-bit down-counter.

● 16-bit reload timer timer register (TMRH0, TMRL0)

TMRH0 and TMRL0 form a 16-bit down-counter. Reading returns the current count value.

● 16-bit reload timer reload register (TMRLRH0, TMRLRL0)

This register sets the load value to the 16-bit down-counter. The register loads the setting value of the 16-bit reload timer reload register to the 16-bit down-counter to down count.

● 16-bit reload timer control status register (TMCSRH0, TMCSRL0)

This register controls the count clock operation mode, clock selection, interrupts and other aspects of the 16-bit reload timer as well as indicates the current operation status.

■ Input Clock

The 16-bit reload timer uses the output clock from the prescaler or the input signal from the TI0 pin as its input clock (count clock).

When Event Counter operates in event counter mode, external clock input from the TI0 pin is gated by the PWM output signal of 8/16-bit composite timer ch.1, and then input to the 16-bit reload timer as count clock. For details on this function, See "CHAPTER 20 EVENT COUNTER".

19.3 Channels of 16-bit Reload Timer

This section describes the channels of the 16-bit reload timer.

■ Channels of 16-bit Reload Timer

The MB95410H/470H Series has one channel of the 16-bit reload timer. Table 19.3-1 and Table 19.3-2 show the correspondence of the channel, pin and register.

Table 19.3-1 Pins of 16-bit Reload Timer

Channel	Pin name	Pin function
0	TO0	16-bit reload timer output
	TI0	16-bit reload timer input

Table 19.3-2 Registers of 16-bit Reload Timer

Channel	Register abbreviation	Register
0	TMCSRH0	16-bit reload timer control status register upper
	TMCSRL0	16-bit reload timer control status register lower
	TMRH0	16-bit reload timer timer register upper
	TMRL0	16-bit reload timer timer register lower
	TMRLRH0	16-bit reload timer reload register upper
	TMRLRL0	16-bit reload timer reload register lower

19.4 Pins of 16-bit Reload Timer

This section describes the pins of the 16-bit reload timer and shows the block diagram of these pins.

■ Pins of 16-bit Reload Timer

The pins of the 16-bit reload timer are namely the TI0 and TO0 pins.

● TI0 pin

This pin is used both as a general-purpose I/O port and as an external pulse input pin for the counter (TI0).

TI0: Any pulse edge input to this pin is counted during counter operation. To use it as the TI0 pin in counter operation, set the port direction register (DDR5:bit2 in the MB95410H Series, DDR1:bit4 in the MB95470H Series) to "0" and use the pin as an input port.

● TO0 pin

This pin is used both as a general-purpose I/O port and as the output pin of the 16-bit reload timer (TO0).

TO0: The pin outputs a waveform of the 16-bit reload timer.

When using this pin as the TO0 pin for the 16-bit reload timer, enabling timer output (TMCSRL0:OUTE = 1) allows output to be performed automatically regardless of the setting of the port direction register (DDR5:bit3 in the MB95410H Series, DDR1:bit0 in the MB95470H Series) and the pin to serve as the TO0 pin of the timer output.

■ Block Diagrams of Pins of 16-bit Reload Timer (MB95410H Series)

Figure 19.4-1 Block Diagram of TO0 of 16-bit Reload Timer

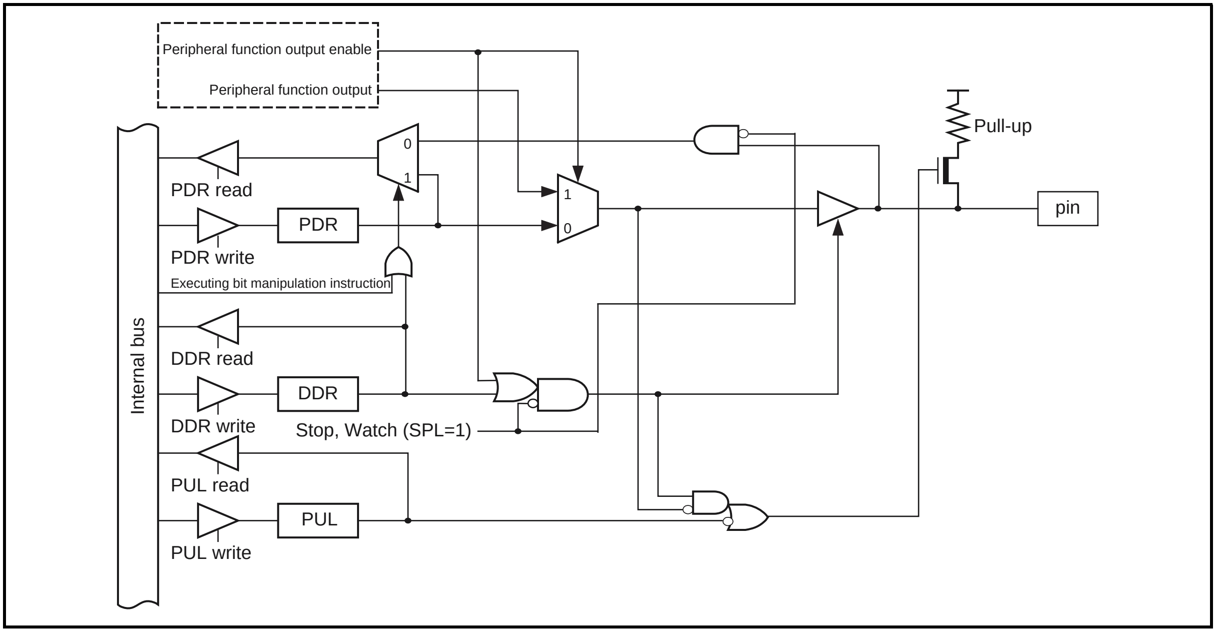
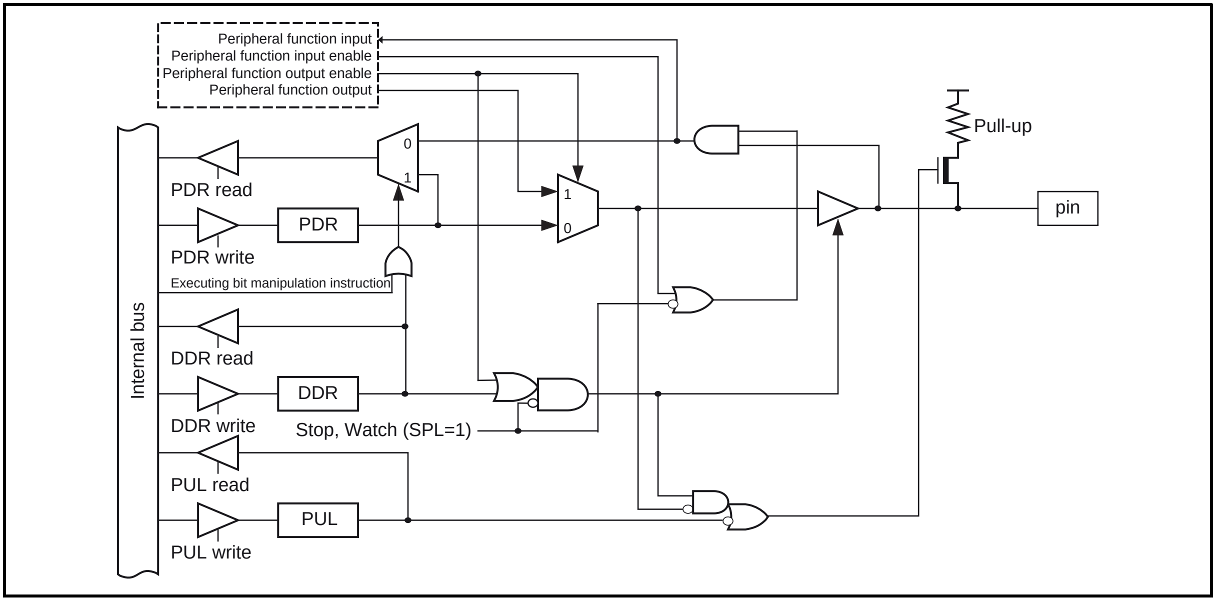


Figure 19.4-2 Block Diagram of TI0 of 16-bit Reload Timer


Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write

■ Block Diagrams of Pins of 16-bit Reload Timer (MB95470H Series)

Figure 19.4-3 Block Diagram of TO0 of 16-bit Reload Timer

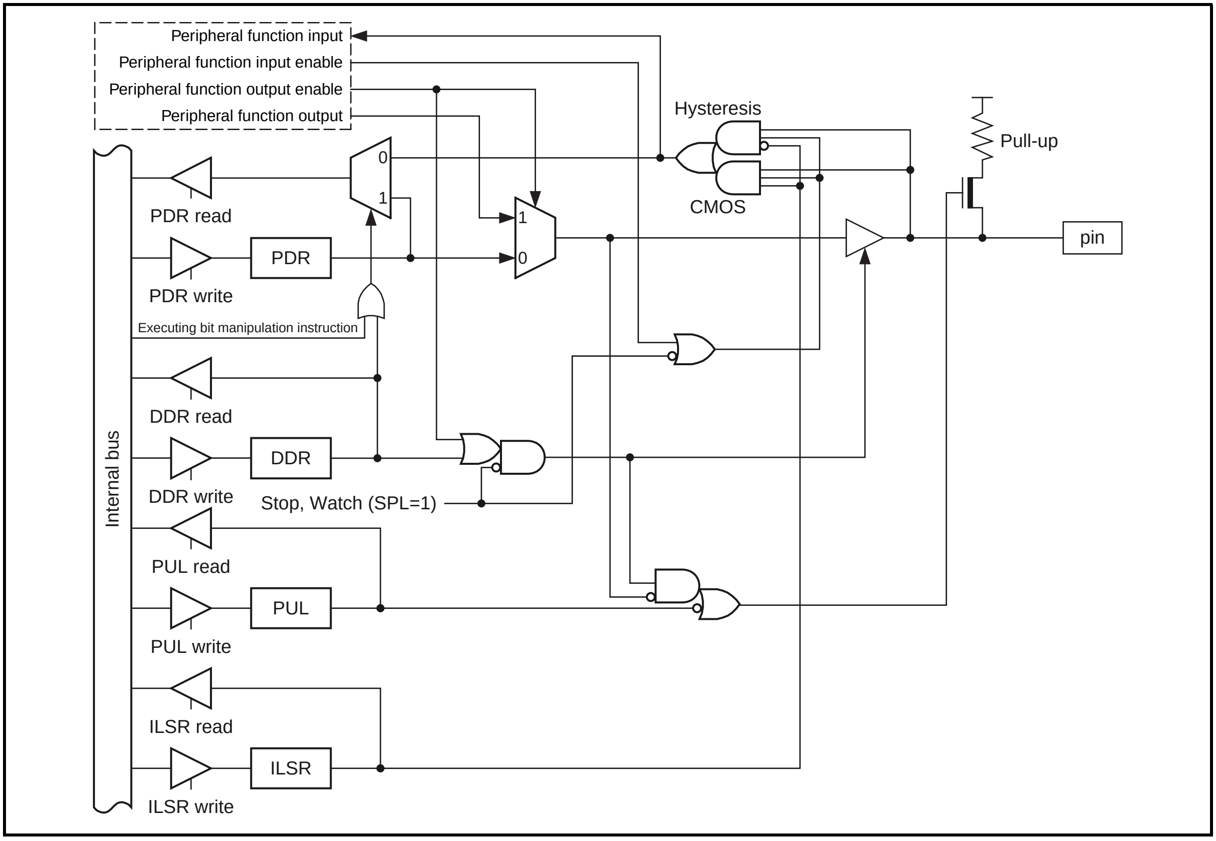
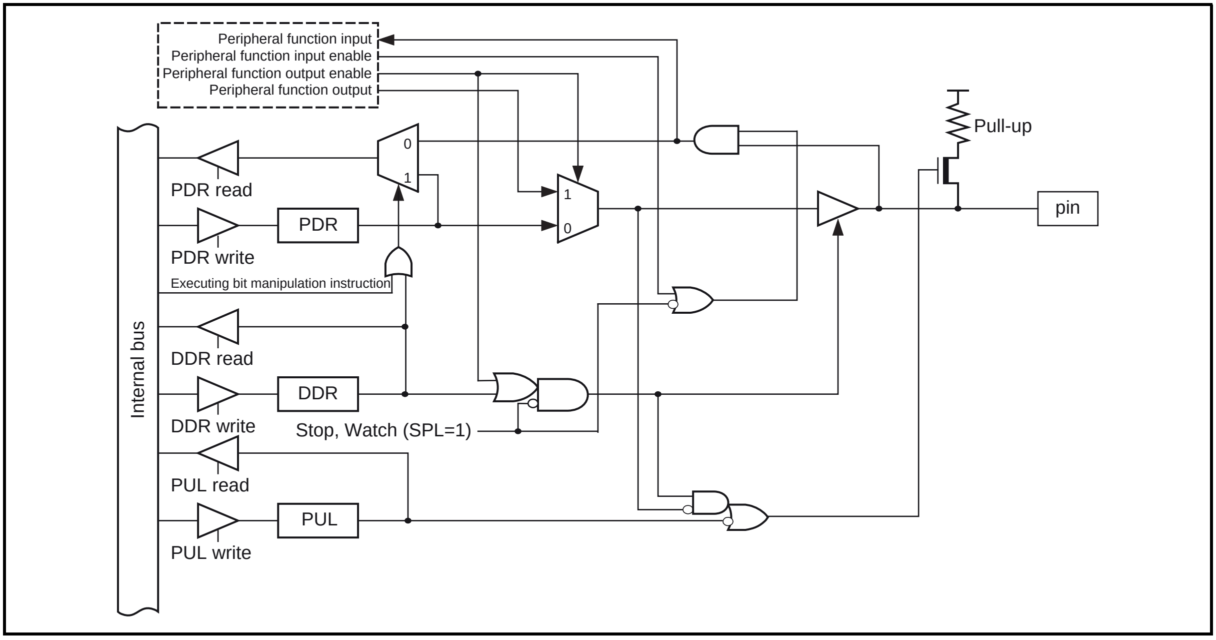

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Hysteresis
CMOS
Pull-up
pin
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write
ILSR read
ILSR
ILSR write
Internal bus

Figure 19.4-4 Block Diagram of TI0 of 16-bit Reload Timer


Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
pin
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write
Internal bus

19.5 Registers of 16-bit Reload Timer

This section describes the registers of the 16-bit reload timer.

■ Registers of 16-bit Reload Timer

Figure 19.5-1 shows the registers of the 16-bit reload timer.

Figure 19.5-1 Registers of 16-bit Reload Timer

16-bit reload timer control status register upper (TMCSRH0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
003E$_H$	-	-	CSL2	CSL1	CSL0	MOD2	MOD1	MOD0	00000000$_B$
	R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	

16-bit reload timer control status register lower (TMCSRL0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
003F$_H$	-	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG	00000000$_B$
	R0/WX	R/W	R/W	R/W	R/W	R(RM1),W	R/W	R0,W	

16-bit reload timer timer register upper (TMRH0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FA6$_H$	D15	D14	D13	D12	D11	D10	D9	D8	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

16-bit reload timer timer register lower (TMRL0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FA7$_H$	D7	D6	D5	D4	D3	D2	D1	D0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

16-bit reload timer reload register upper (TMRLRH0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FA6$_H$	D15	D14	D13	D12	D11	D10	D9	D8	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

16-bit reload timer reload register lower (TMRLRL0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FA7$_H$	D7	D6	D5	D4	D3	D2	D1	D0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W	: Readable/writable (The read value is the same as the write value.)
R(RM1), W	: Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0,W	: Write only (Writable. The read value is "0".)
R0/WX	: The read value is "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

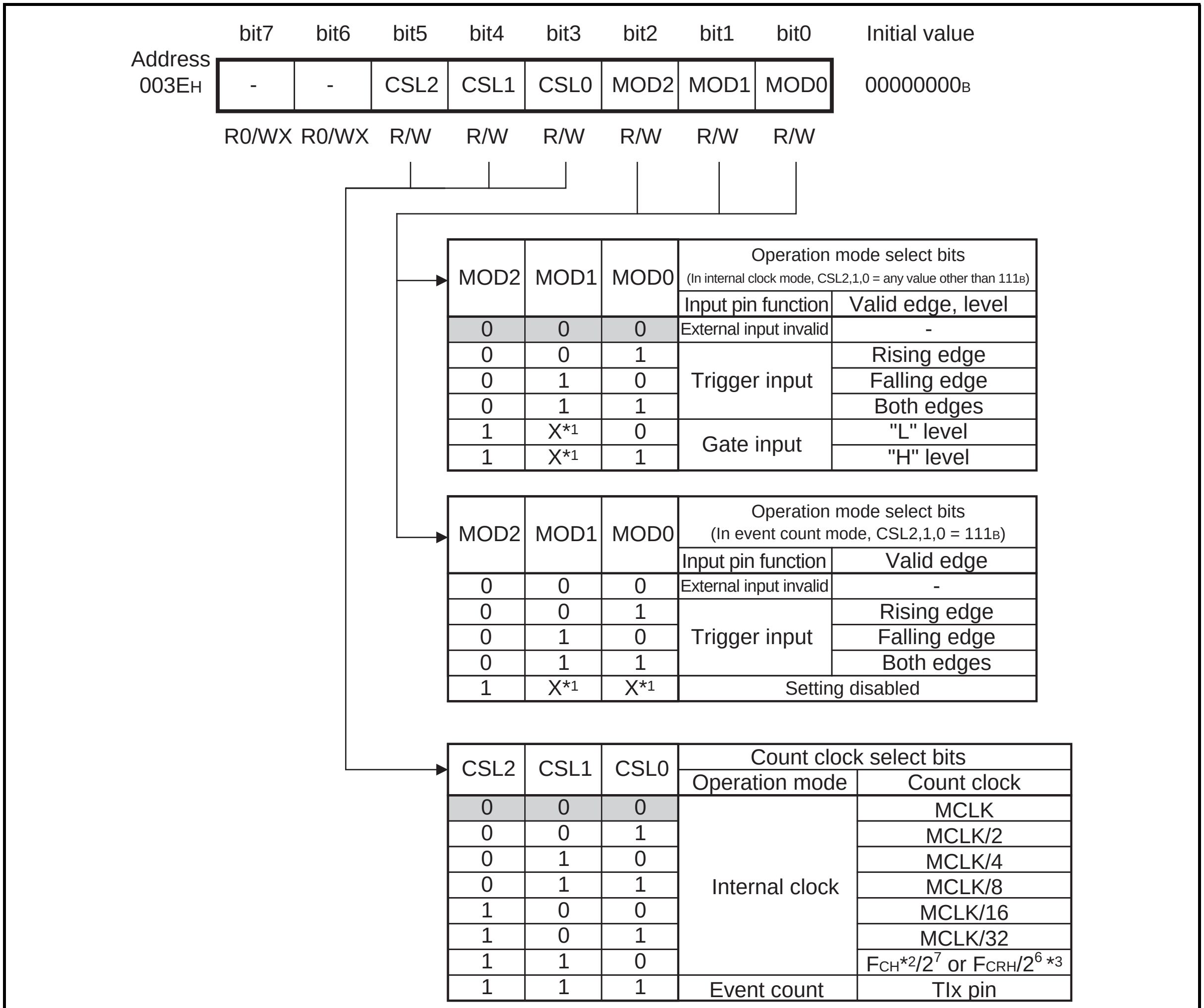
Note: TMRH0 and TMRLRH0 are assigned to the same address.
TMRL0 and TMRLRL0 are assigned to the same address.

19.5.1 16-bit Reload Timer Control Status Register Upper (TMCSRH0)

The 16-bit reload timer control status register upper (TMCSRH0) sets the operation mode and operating conditions of the 16-bit reload timer.

■ 16-bit Reload Timer Control Status Register Upper (TMCSRH0)

Figure 19.5-2 16-bit Reload Timer Control Status Register Upper (TMCSRH0)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$003E_H$	-	-	CSL2	CSL1	CSL0	MOD2	MOD1	MOD0	00000000_B
	R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	

MOD2	MOD1	MOD0	Operation mode select bits (In internal clock mode, CSL2,1,0 = any value other than 111_B)	
			Input pin function	Valid edge, level
0	0	0	External input invalid	-
0	0	1	Trigger input	Rising edge
0	1	0		Falling edge
0	1	1		Both edges
1	X*1	0	Gate input	"L" level
1	X*1	1		"H" level

MOD2	MOD1	MOD0	Operation mode select bits (In event count mode, CSL2,1,0 = 111_B)	
			Input pin function	Valid edge
0	0	0	External input invalid	-
0	0	1	Trigger input	Rising edge
0	1	0		Falling edge
0	1	1		Both edges
1	X*1	X*1	Setting disabled	

CSL2	CSL1	CSL0	Count clock select bits	
			Operation mode	Count clock
0	0	0	Internal clock	MCLK
0	0	1		MCLK/2
0	1	0		MCLK/4
0	1	1		MCLK/8
1	0	0		MCLK/16
1	0	1		MCLK/32
1	1	0		F_{CH}*2/2^7 or F_{CRH}/2^6 *3
1	1	1	Event count	TIx pin

MCLK	:	Machine clock
F_{CH}	:	Main clock
F_{CRH}	:	Main CR clock
R/W	:	Readable/writable (The read value is the same as the write value.)
R0/WX	:	The read value is "0". Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

*1 : Either "0" or "1" can be selected.

*2 : When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used.
When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

*3 : The value to be used as the count clock is decided according to the settings of the SYCC2 register.

Table 19.5-1 Functions of Bits in 16-bit Reload Timer Timer Control Status Register Upper (TMCSRH0)

Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5 to bit3	CSL2, CSL1, CSL0: Count clock select bits	These bits select the count clock for the 16-bit reload timer. **Writing any value other than "111":** Internal clock is counted (internal clock mode). The internal clock is generated by the prescaler. See "6.13 Operation of Prescaler". **Writing "111":** Edge of the external event clock is counted (event count mode).
bit2 to bit0	MOD2, MOD1, MOD0: Operation mode select bits	These bits set the operating conditions of the 16-bit reload timer. • **Internal clock mode (CSL2 to CSL0 = any value other than 111_B)** The MOD2 bit selects the input pin function. **Writing "0" to the MOD2 bit:** - TI0 pin serves as a trigger input. - MOD1 and MOD0 bits are used to select the edge to be detected. - When the edge is detected, the value set in the 16-bit reload timer reload register is reloaded in the 16-bit reload timer timer register (TMR) and the TMR starts counting. **Writing "1" to the MOD2 bit:** - TI0 pin serves as a gate input. - Setting the MOD1 bit is invalid. - The MOD0 bit is used to select the valid signal level (H or L). The TMR only counts while the valid signal level is being input. Note: External input is disabled when MOD2 to MOD0 are "000". In this case, the TRG bit is used to start operation by software. • **Event count mode (CSL2 to CSL0 = 111_B)** - The MOD2 bit is always fixed to "0". - The external event clock is input from the TI0 pin. - The MOD1 and MOD0 bits are used to select the edge to be detected. *: When Event Counter operates in event counter mode, external clock input from the TI0 pin is gated by the PWM output signal of 8/16-bit composite timer ch.1, and then input to 16-bit reload timer as count clock. For detail on this function, see "CHAPTER 20 EVENT COUNTER".

19.5.2 16-bit Reload Timer Control Status Register Lower (TMCSRL0)

The 16-bit reload timer control status register lower (TMCSRL0) sets the operating conditions of the 16-bit reload timer, enables or disables counting, controls interrupts, and checks the interrupt request status.

■ 16-bit Reload Timer Control Status Register Lower (TMCSRL0)

Figure 19.5-3 16-bit Reload Timer Control Status Register Lower (TMCSRL0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
003F$_H$	-	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG	00000000$_B$
	R0/WX	R/W	R/W	R/W	R/W	R(RM1),W	R/W	R0,W	

TRG	Software trigger bit	
	Read	Write
0	Always reads "0"	No effect on operation
1		Starts counting after reloading

CNTE	Count enable bit
0	Stops count
1	Enables count (waiting for start trigger)

UF	Underflow interrupt request flag bit	
	Read	Write
0	No underflow	Clears this bit
1	Underflow	No effect on operation

INTE	Underflow interrupt request enable bit
0	Disables underflow interrupt
1	Enables underflow interrupt

RELD	Reload selection bit
0	One-shot mode
1	Reload mode

OUTL	Pin output level selection bit	
	One-shot mode	Reload mode
0	Outputs "H" square waveform during counting	Outputs "L" toggle when counting starts
1	Outputs "L" square waveform during counting	Outputs "H" toggle when counting starts

OUTE	Timer output enable bit
0	Disables timer output (general-purpose I/O port)
1	Enables timer output

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0,W : Write only (Writable. The read value is "0".)
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit
▨ : Initial value

Table 19.5-2 Functions of Bits in 16-bit Reload Timer Control Status Register Lower (TMCSRL0)

Bit name		Function
bit7	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.
bit6	OUTE: Timer output enable bit	This bit sets the TO0 pin function of the 16-bit reload timer. **Writing "0":** Sets the pin as a general-purpose I/O port. **Writing "1":** Sets the pin as the TO0 pin of the 16-bit reload timer.
bit5	OUTL: Pin output level selection bit	This bit sets the output level of the output pin of the 16-bit reload timer. • **When one-shot mode is selected (RELD = 0):** **"0":** Outputs "H" level square waveform while the 16-bit reload timer counts. **"1":** Outputs "L" level square waveform while the 16-bit reload timer counts. • **When reload mode is selected (RELD = 1):** **"0":** Outputs an "L" when the 16-bit reload timer is started and then toggles each time an underflow occurs. **"1":** Outputs an "H" when the 16-bit reload timer is started and then toggles each time an underflow occurs.
bit4	RELD: Reload selection bit	This bit sets reload operation when an underflow occurs. **"0":** When an underflow occurs, counting is suspended. (One-shot mode) **"1":** When an underflow occurs, the value that has been set to the 16-bit reload timer reload register is loaded to the 16-bit reload timer timer register, and counting continues. (Reload mode)
bit3	INTE: Underflow interrupt request enable bit	This bit enables or disables underflow interrupts. **Writing "0":** Disables interrupt requests. **Writing "1":** Enables interrupt requests.
bit2	UF: Underflow interrupt request flag bit	This bit indicates that an underflow has occurred on the 16-bit reload timer. **Writing "0":** Clears the UF bit. **Writing "1":** Has no effect on operation. • "1" is always read in read-modify-write instructions.
bit1	CNTE: Count enable bit	This bit enables or disables the operation of the 16-bit reload timer. **Writing "0":** Stops counting. **Writing "1":** The unit goes to standby to wait for a start trigger. When a start trigger is input, the 16-bit reload timer timer register starts counting.
bit0	TRG: Software trigger bit	This bit allows the 16-bit reload timer to be started by software. The TRG bit is valid only when timer operation is enabled (CNTE = 1). **Writing "0":** Has no effect on operation. **Writing "1":** The value set in the 16-bit reload timer reload register is reloaded to the 16-bit reload timer timer register and then the 16-bit reload timer timer register starts counting from the next count clock input. Note: This bit can be set to "1" at the same time as the CNTE bit without affecting the operation. • This bit always returns "0" when read. However, "1" is read during the time between writing "1" to start the timer and the timer count actually starting.

19.5.3 16-bit Reload Timer Timer Register Upper (TMRH0)/Lower (TMRL0)

The 16-bit reload timer timer register upper (TMRH0)/lower (TMRL0) can be used to read the value of the 16-bit down-counter.

■ 16-bit Reload Timer Timer Register Upper (TMRH0)/Lower (TMRL0)

Figure 19.5-4 16-bit Reload Timer Timer Register Upper (TMRH0)/Lower (TMRL0)

TMRH0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address	D15	D14	D13	D12	D11	D10	D9	D8	00000000_B
$0FA6_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

TMRL0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address	D7	D6	D5	D4	D3	D2	D1	D0	00000000_B
$0FA7_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

The 16-bit reload timer timer registers can read the count value of the 16-bit down-counter.

If counting is enabled (TMCSRL0:CNTE=1) at the beginning of a count, the value written in the 16-bit reload timer reload registers will be reloaded to these registers and the timer will start counting down.

Notes:

- The registers can read the count value even during counting. To make a read access to these registers, use a word transfer instruction, or read the upper byte first and the lower byte second. The circuit is configured so that the value in the lower byte is saved when the upper byte is read.
- The registers are read-only and located at the same addresses as the 16-bit reload timer reload registers. Accordingly, a write access to these registers is also a write access to the 16-bit reload timer reload registers.

19.5.4 16-bit Reload Timer Reload Register Upper (TMRLRH0)/Lower (TMRLRL0)

The 16-bit reload timer reload upper (TMRLRH0)/lower (TMRLRL0) register set the reload value for the 16-bit down-counter. The value set in the 16-bit reload timer reload registers is reloaded to the 16-bit down-counter to down count.

■ 16-bit Reload Timer Reload Register Upper (TMRLRH0)/Lower (TMRLRL0)

Figure 19.5-5 16-bit Reload Timer Reload Register Upper (TMRLRH0)/Lower (TMRLRL0)

TMRLRH0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address	D15	D14	D13	D12	D11	D10	D9	D8	00000000_B
$0FA6_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

TMRLRL0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address	D7	D6	D5	D4	D3	D2	D1	D0	00000000_B
$0FA7_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

These registers set the reload value to the 16-bit down-counter.

The value set in the 16-bit reload timer reload registers is reloaded to the 16-bit down-counter to start down-counting at the timing of start or underflow. (The value can be modified during counter operation)

Notes:

- The registers can be written to even while the counter is running. To make a write access to these registers, use a word transfer instruction or write the upper byte first and lower byte second. (The circuit is implemented so that the upper byte is not used until the lower byte is written.)
- The registers are write-only and located at the same addresses as the 16-bit reload timer timer registers. Therefore, a read access to these registers is also a read access to the 16-bit reload timer timer registers.

19.6 Interrupts of 16-bit Reload Timer

The 16-bit reload timer outputs an interrupt request when an underflow occurs on the 16-bit down-counter.

■ Interrupts of 16-bit Reload Timer

Table 19.6-1 shows the interrupt control bits and interrupt sources of the 16-bit reload timer.

Table 19.6-1 Interrupt Control Bits and Interrupt Sources of 16-bit Reload Timer

Item	Description
Interrupt request flag bit	UF bit in TMCSRL0 register
Interrupt request enable bit	INTE bit in TMCSRL0 register
Interrupt source	Underflow of down-counter (TMRH0/TMRL0)

The 16-bit reload timer sets the underflow interrupt request flag bit (UF) in the 16-bit reload timer control status register lower (TMCSRL0) to "1" when an underflow occurs in the 16-bit down-counter ("0000_H" → "$FFFF_H$"). If the underflow interrupt request enable bit is enabled (INTE = 1), the interrupt request will be outputted to the interrupt controller.

■ Register and Vector Table Addresses Related to Interrupts of 16-bit Reload Timer

Table 19.6-2 Register and Vector Table Addresses Related to Interrupts of 16-bit Reload Timer

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
16-bit reload timer ch. 0	IRQ11	ILR2	L11	$FFE4_H$	$FFE5_H$

ch.: Channel

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

19.7 Operations of 16-bit Reload Timer and Setting Procedure Example

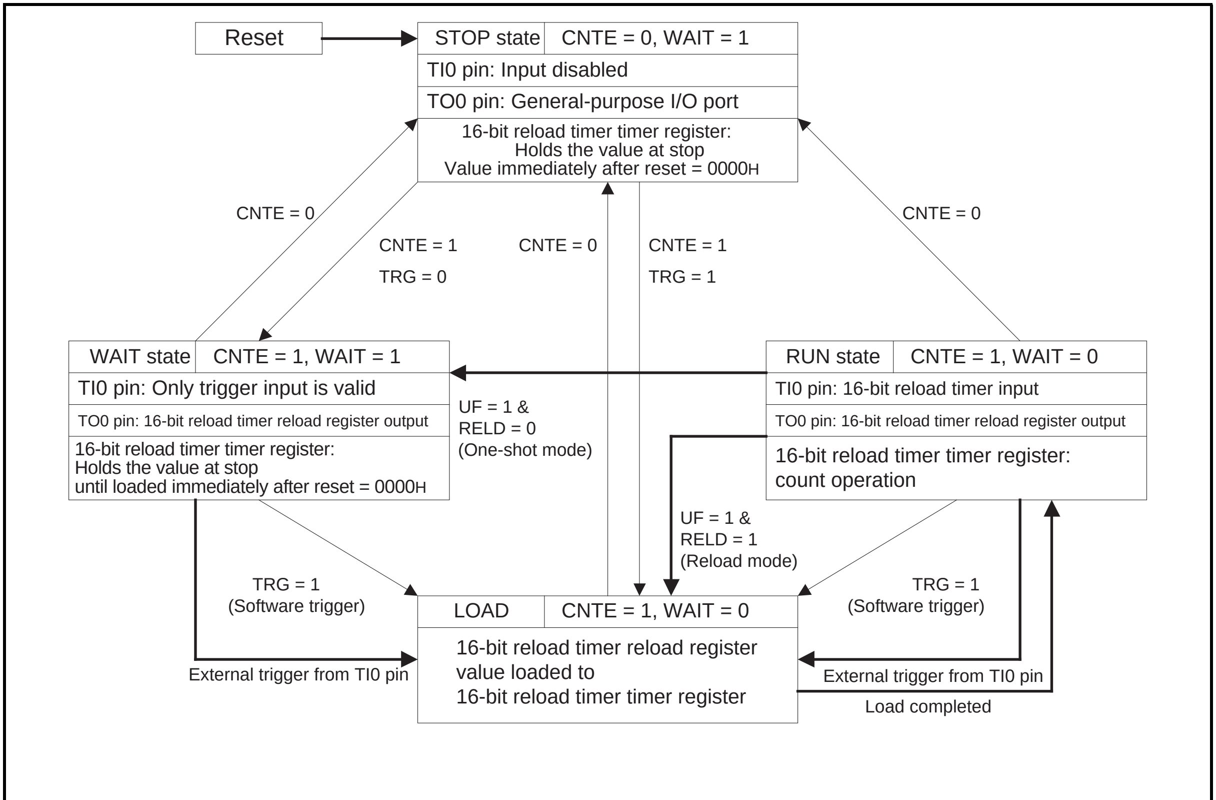
This section describes the operating status of the 16-bit reload timer counter.

■ Operating Status of Counter

The counter status is determined by the value of the count enable bit (CNTE) in the 16-bit reload timer control status register (TMCSRL0) and the internal signal start trigger wait signal (WAIT). The STOP state (halted), WAIT state (waiting for a start trigger) and RUN state (operating state) can be set.

Figure 19.7-1 shows the status transition of these counters.

Figure 19.7-1 Diagram of Counter State Transition


Reset
STOP state
CNTE = 0, WAIT = 1
TI0 pin: Input disabled
TO0 pin: General-purpose I/O port
16-bit reload timer timer register:
Holds the value at stop
Value immediately after reset = 0000H
CNTE = 0
CNTE = 1
TRG = 0
CNTE = 0
CNTE = 1
TRG = 1
CNTE = 0
WAIT state
CNTE = 1, WAIT = 1
TI0 pin: Only trigger input is valid
TO0 pin: 16-bit reload timer reload register output
16-bit reload timer timer register:
Holds the value at stop
until loaded immediately after reset = 0000H
UF = 1 &
RELD = 0
(One-shot mode)
RUN state
CNTE = 1, WAIT = 0
TI0 pin: 16-bit reload timer input
TO0 pin: 16-bit reload timer reload register output
16-bit reload timer timer register:
count operation
UF = 1 &
RELD = 1
(Reload mode)
TRG = 1
(Software trigger)
TRG = 1
(Software trigger)
LOAD
CNTE = 1, WAIT = 0
16-bit reload timer reload register
value loaded to
16-bit reload timer timer register
External trigger from TI0 pin
External trigger from TI0 pin
Load completed

⟶	: State transition by hardware
→	: State transition by register access
WAIT	: WAIT signal (internal signal)
TRG	: Software trigger bit (TMCSRL0)
CNTE	: Timer operation enable bit (TMCSRL0)
UF	: Underflow generation flag bit (TMCSRL0)
RELD	: Reload selection bit (TMCSRL0)

■ Setting Procedure Example

Below is an example of procedure for setting the 16-bit reload timer.

● Initial settings

1) Set the interrupt level. (ILR2)
2) Set the reload value. (TMR0)
3) Select the clock. (TMCSRH0:CSL2 to CSL0)
4) Select the operation mode. (TMCSRH0:MOD2 to MOD0)
5) Enable the output. (TMCSRL0:OUTE = 1)
6) Select the output level. (TMCSRL0:OUTL)
7) Select reload. (TMCSRL0:RELD)
8) Enable a count. (TMCSRL0:CNTE = 1)
9) Perform the software trigger. (TMCSRL0:TRG = 1)
10) Enable underflow interrupt. (TMCSRL0:INTE = 1)

● Interrupt processing

1) Clear the underflow interrupt request flag. (TMCSRL0:UF=0)
2) Disable underflow interrupt. (TMCSRL0:INTE = 0)
3) Process any interrupt.
4) Enable underflow interrupt. (TMCSRL0:INTE = 1)

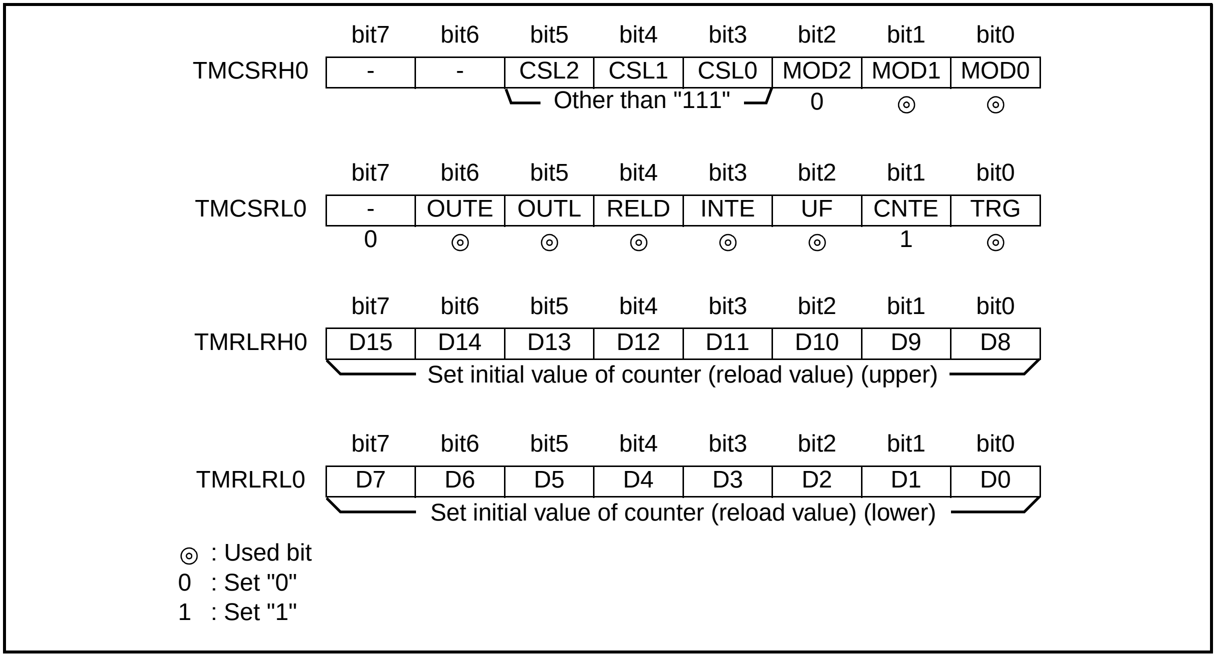
19.7.1 Internal Clock Mode

In this mode, the 16-bit down-counter counts down while being synchronized with the internal count clock, and outputs an interrupt request to the interrupt controller every time an underflow occurs ("0000$_H$" → "FFFF$_H$"). In addition, the TO0 pin can output the toggle waveform.

■ Setting Internal Clock Mode

The timer requires the register settings shown in Figure 19.7-2 to operate as an interval timer.,

Figure 19.7-2 Internal Clock Mode Setup



	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMCSRH0	-	-	CSL2	CSL1	CSL0	MOD2	MOD1	MOD0
			Other than "111"			0	◎	◎

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMCSRL0	-	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG
	0	◎	◎	◎	◎	◎	1	◎

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMRLRH0	D15	D14	D13	D12	D11	D10	D9	D8
	Set initial value of counter (reload value) (upper)							

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMRLRL0	D7	D6	D5	D4	D3	D2	D1	D0
	Set initial value of counter (reload value) (lower)							

◎ : Used bit
0 : Set "0"
1 : Set "1"

■ Operation of Internal Clock Mode (Reload Mode)

When "1" is set to the count enable bit (CNTE) to enable counting, and the timer is started by setting "1" to the software trigger bit (TRG) or by an external trigger, the value set in the 16-bit reload timer reload register lower (TMRLR0) is reloaded to the 16-bit down-counter and down-counting starts. If counting is enabled when the count enable bit (CNTE) and software trigger bit (TRG) are set to "1" at the same time, the count is started at the same time.

If the reload selection bit (RELD) is "1", the value of the 16-bit reload timer reload register lower (TMRLR0) is reloaded to the 16-bit down-counter and the count continues when the 16-bit counter underflows ("0000$_H$" → "FFFF$_H$"). If the underflow interrupt request flag bit (UF) is "1" when the underflow interrupt request enable bit (INTE) is set to "1", an interrupt request is outputted.

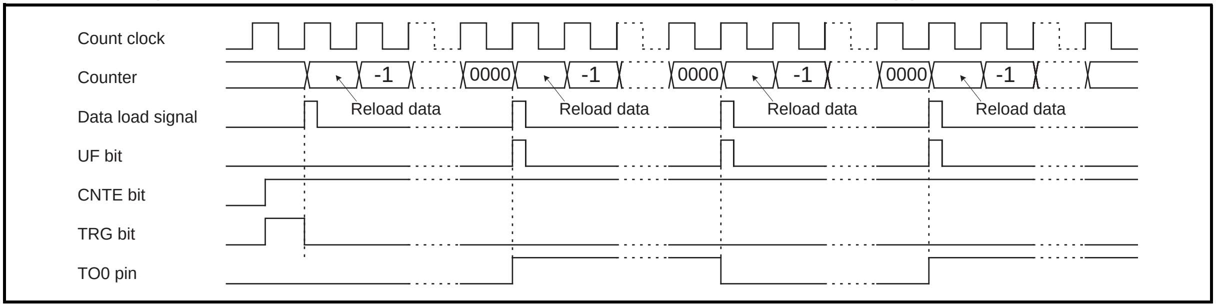
The TO0 pin can output a toggle waveform that is inverted every time an underflow occurs.

● Software trigger operation

When the count enable bit (CNTE) is set to "1", setting "1" to the software trigger bit (TRG) starts counting.

Figure 19.7-3 shows the software trigger operation in reload mode.

Figure 19.7-3 Count Operation in Reload Mode (Software Trigger Operation)



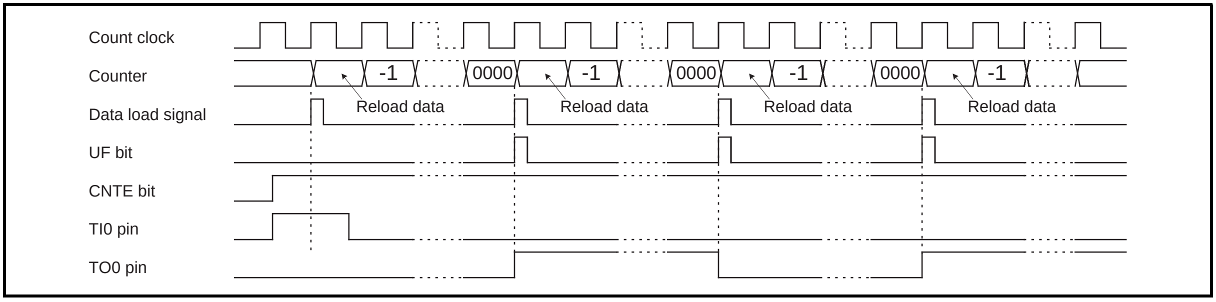
● External trigger input operation

The count starts when the count enable bit (CNTE) is set to "1" and a valid edge of trigger input (rising, falling, or both selectable) set by the operation mode selection bits (MOD2 to MOD0) is input to the TI0 pin.

The timer which starts with the software trigger also becomes effective as well as the start with an external trigger.

Figure 19.7-4 shows the external trigger input operation in reload mode.

Figure 19.7-4 Count Operation in Reload Mode (External Trigger Input Operation)



● Gate input operation

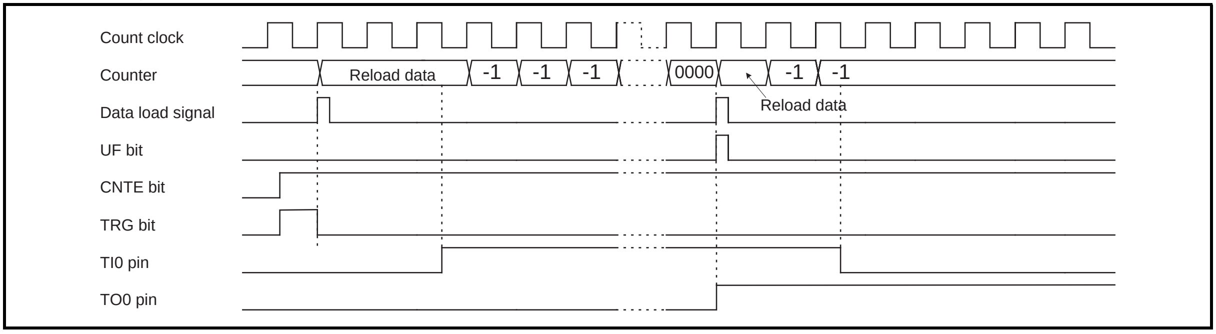
The count starts when the count enable bit (CNTE) is set to "1" and the software trigger bit (TRG) is also set to "1".

The timer continues counting while the valid gate input level ("L" or "H" selectable) set by the operation mode selection bits (MOD2 to MOD0) is being input to the TI0 pin.

The timer start with the software trigger becomes effective as well as the start with an external trigger, too.

Figure 19.7-5 shows the gate input operation in reload mode.

Figure 19.7-5 Count Operation in Reload Mode (External Gate Input Operation)



■ Operation of Internal Clock Mode (One-shot Mode)

When the count enable bit (CNTE) is set to "1" and the software trigger bit (TRG) is set to "1" or the valid edge (rising, falling or both edges selectable) specified by the operation mode selection bits (MOD2 to MOD0) is input to the TI0 pin, the value set in the 16-bit reload timer reload register is reloaded to the 16-bit down-counter and down-counting starts. When the count enable bit (CNTE) and software trigger bit (TRG) are set to "1" at the same time and then counting is enabled, the count is started simultaneously.

If the reload selection bit (RELD) is "0", the 16-bit counter halts at "FFFF$_H$" when the 16-bit counter underflows ("0000$_H$" → "FFFF$_H$"). In this case, the underflow interrupt request flag bit (UF) is set to "1" and if the underflow interrupt request enable bit (INTE) is "1", an interrupt request is outputted.

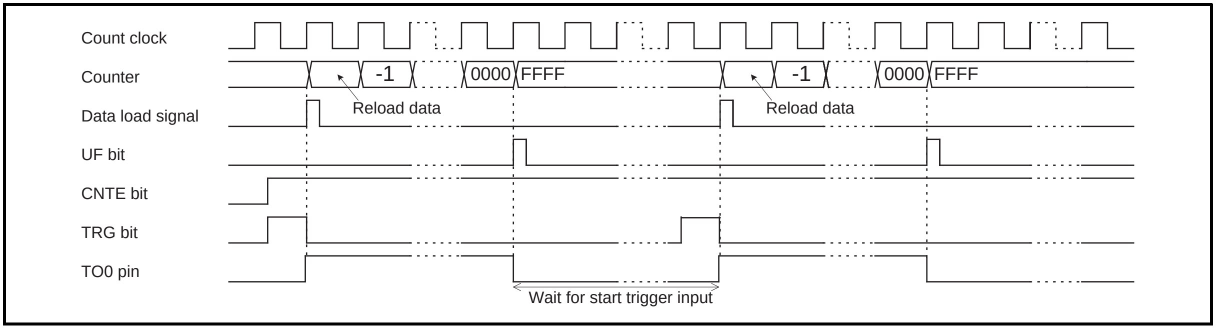
A square waveform can be outputted from the TO0 pin to indicate that the count is in progress.

● Software trigger operation

The count starts when the count enable bit (CNTE) is "1" and the software trigger bit (TRG) is set to "1".

Figure 19.7-6 shows the software trigger operation in one-shot mode.

Figure 19.7-6 Count Operation in One-shot Mode (Software Trigger Operation)

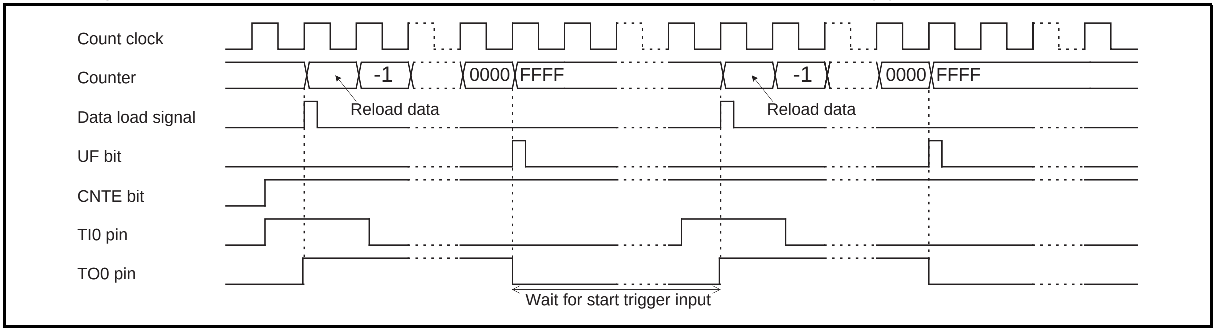


- External trigger input

The count starts when the count enable bit (CNTE) is "1" and the valid edge of trigger input (rising, falling, or both edges) specified by the operation mode selection bits (MOD2 to MOD0) is input to the TI0 pin.

Figure 19.7-7 shows the external trigger input operation in one-shot mode.

Figure 19.7-7 Count Operation in One-shot Mode (External Trigger Input Operation)



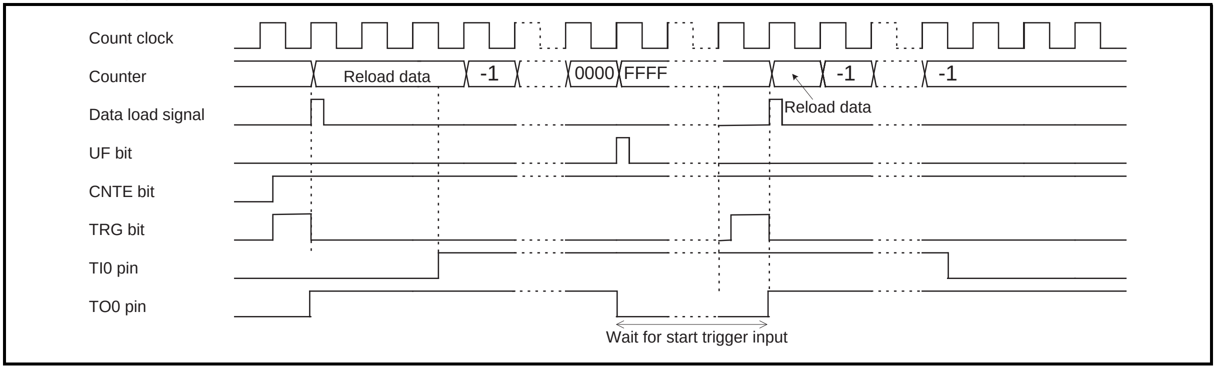
- Gate input operation

The count starts when the count enable bit (CNTE) is "1" and the software trigger bit (TRG) is also set to "1".

The timer continues counting as long as the trigger input enable level ("L" or "H" selectable) specified by the operation mode selection bits (MOD2 to MOD0) is input to the TI0 pin.

Figure 19.7-8 shows the external gate input operation in one-shot mode.

Figure 19.7-8 Count Operation in One-shot Mode (External Gate Input Operation)



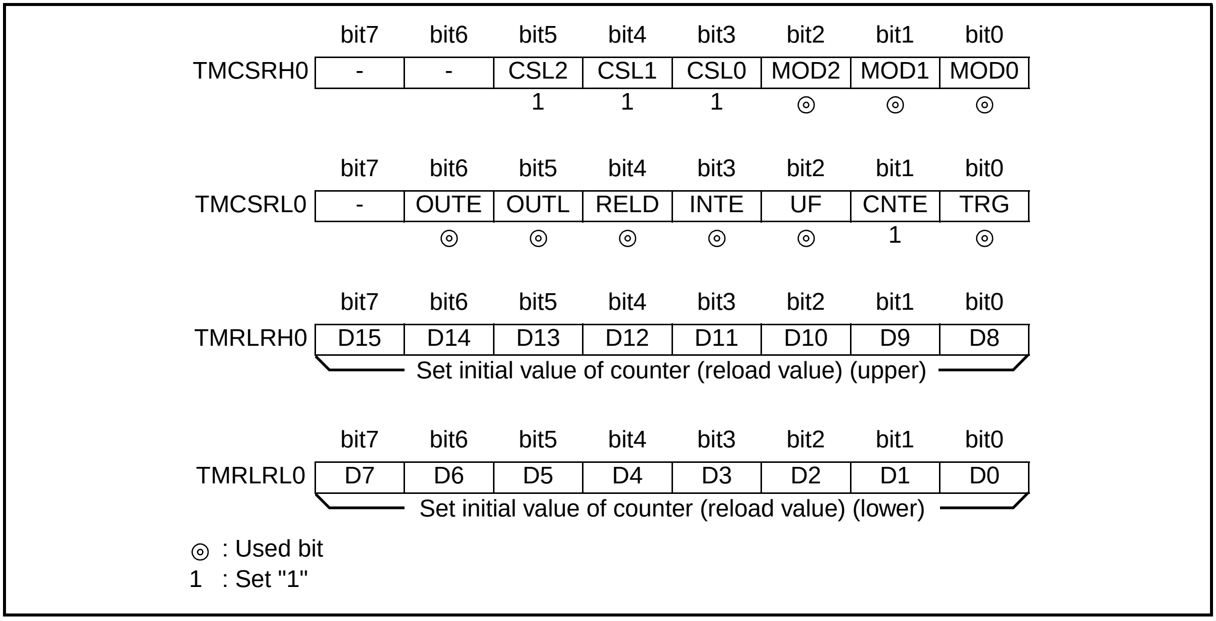
19.7.2 Event Count Mode

In this mode, the 16-bit down-counter counts down each time the valid edge is detected on the pulses input to the TI0 pin, and an interrupt request is outputted to the interrupt controller when an underflow occurs ("0000$_H$" → "FFFF$_H$"). In addition, a toggle waveform or square waveform can be outputted from the TO0 pin.

■ Event Count Mode Setup

The timer requires the register settings shown in Figure 19.7-9 to operate as an event counter.

Figure 19.7-9 Event Count Mode Setup



	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMCSRH0	-	-	CSL2	CSL1	CSL0	MOD2	MOD1	MOD0
			1	1	1	◎	◎	◎

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMCSRL0	-	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG
		◎	◎	◎	◎	◎	1	◎

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMRLRH0	D15	D14	D13	D12	D11	D10	D9	D8

Set initial value of counter (reload value) (upper)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMRLRL0	D7	D6	D5	D4	D3	D2	D1	D0

Set initial value of counter (reload value) (lower)

◎ : Used bit
1 : Set "1"

■ Event Count Mode

The value set in the 16-bit reload timer reload register (TMRLRH0/TMRLRL0) is reloaded to the 16-bit counter when the count enable bit (CNTE) is set to "1" and the software trigger bit (TRG) is set to "1". The counter counts each time the valid edge (rising, falling, or both edges selectable) is detected on the pulses input to the TI0 pin (external count clock).

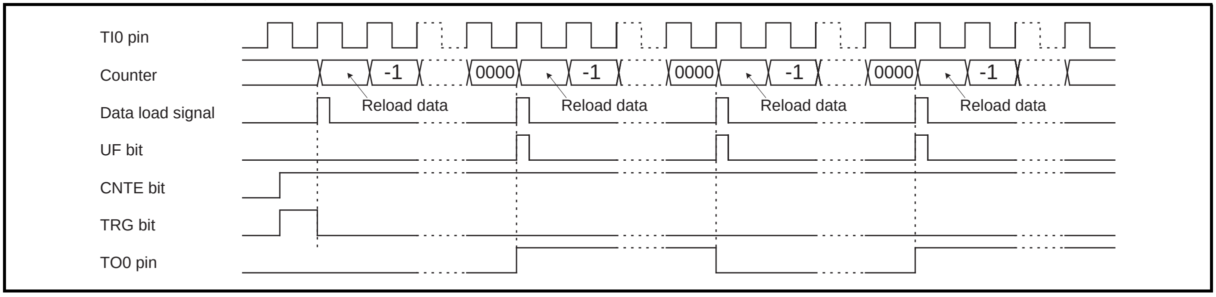
● Operation of reload mode

If the reload selection bit (RELD) is "1", the value set in the 16-bit reload timer reload register (TMRLRH0/TMRLRL0) is reloaded to the 16-bit counter and the count continues when the 16-bit counter underflows ("0000$_H$" → "FFFF$_H$").

The underflow interrupt request flag bit (UF) in the 16-bit reload timer control status register lower(TMCSRL0) is set to "1" when an underflow occurs ("0000$_H$" → "FFFF$_H$") in the 16-bit counter, and an interrupt request is outputted if the underflow interrupt enable bit (INTE) is set to "1".

The TO0 pin can output a toggle waveform that is inverted each time an underflow occurs. Figure 19.7-10 shows the count operation in reload mode.

Figure 19.7-10 Count Operation in Reload Mode (Event Count Mode)



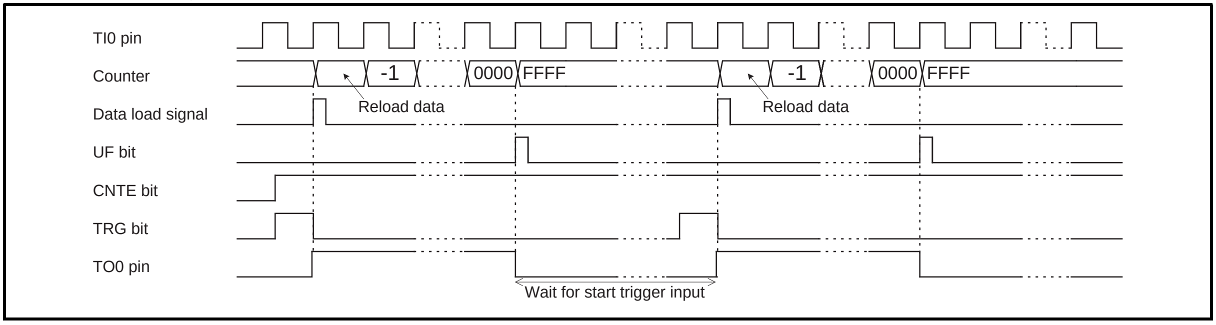
● Operation of one-shot mode

If the reload selection bit (RELD) is "0", the value of the 16-bit counter halts at "FFFF$_H$" when the 16-bit counter underflows ("0000$_H$" → "FFFF$_H$").

An interrupt request is outputted when the underflow request flag bit (UF) in the lower timer control status register (TMCSRL0) is set to "1" with the underflow interrupt enable bit (INTE) set to "1".

The TO0 pin outputs a square waveform indicating that counting is in progress. Figure 19.7-11 shows the count operation in one-shot mode.

Figure 19.7-11 Count Operation in One-shot Mode (Event Count Mode)



19.8 Notes on Using 16-bit Reload Timer

This section provides notes on using the 16-bit reload timer.

■ Notes on Using 16-bit Reload Timer

● Precautions when setting the program

- A value can be read from the 16-bit reload timer timer register even during counting. As for read access, use a word transfer instruction or read the upper byte first and the lower byte second.
- A value can be written to the 16-bit reload timer reload register even during counting. As for write access, use a word transfer instruction or write the upper byte first and the lower byte second.

● Precaution for interrupts

The unit cannot recover from interrupt processing when the underflow interrupt request enable bit (INTE) is set to "1" and "1" is set to the underflow interrupt request flag bit (UF) in the 16-bit reload timer control status register lower (TMCSRL0). Always set the underflow interrupt request flag bit (UF) to "0".

● Note on the event counter operating in event counter operation mode

When the event counter operates in event counter operation mode, the 16-bit reload timer cannot be used.

19.9 Sample Settings for 16-bit Reload Timer

This section provides sample settings for the 16-bit reload timer.

■ Sample Settings

● How to select the count clock

The count clock selection bits (TMCSRH0:CSL[2:0]) are used.

Operation	Count clock selection bits (CSL[2:0])
To select the internal clock	Set the bits to any value other than "111_B".
To select the external event clock	Set the bits to "111_B".

● How to select the operating conditions of internal clock mode

The operation mode selection bits (TMCSRH0:MOD[2:0]) are used to set the conditions.

Operating condition	Operation mode selection bits (MOD[2:0])
Trigger input from TI0 pin (rising edge)	Set the bits to "001_B".
Trigger input from TI0 pin (falling edge)	Set the bits to "010_B".
Trigger input from TI0 pin (both edges)	Set the bits to "011_B".
Gate input from TI0 pin (L level)	Set the bits to "$1x0_B$".
Gate input from TI0 pin (H level)	Set the bits to "$1x1_B$".

● How to select the operating conditions of event count mode

The operation mode selection bits (TMCSRH0:MOD[1:0]) are used to set the conditions.

Operating condition	Operation mode selection bits (MOD[1:0])
Rising edge	Set the bits to "01_B".
Falling edge	Set the bits to "10_B".
Both edges	Set the bits to "11_B".

The setting of MOD2 has no effect on operation, whether it is "0" or "1".

- How to enable/stop the count operation of the reload timer

The count enable bit of the timer (TMCSRL0:CNTE) is used.

Operation	Operation enable bit (CNTE)
To stop the reload timer	Set the bit to "0".
To enable the count operation of the reload timer	Set the bit to "1".

The count cannot be resumed from the stop state. Enable the operation before or at the same time as the activation.

- How to set reload the timer mode (reload/one-shot)

The reload selection bit (TMCSRL0:RELD) is used.

Operation	Reload selection bit (RELD)
To select one-shot mode	Set the bit to "0".
To select reload mode	Set the bit to "1".

- How to invert the output level

The output level is specified as shown in the following table.

The pin output level selection bit (TMCSRL0:OUTL) is used to set the output level.

Output level	Pin output level selection bit (OUTL)
"L" toggle output when count starts in reload mode	Set the bit to "0".
"H" toggle output when count starts in reload mode	Set the bit to "1".
Outputting "H" square waveform during counting in one-shot mode	Set the bit to "0".
Outputting "L" square waveform during counting in one-shot mode	Set the bit to "1".

- How to switch the TI0 pin to an external event input pin or to an external trigger input pin

"0" is set to the data direction specification bit (DDR5:bit2 in the MB95410H Series and DDR1:bit4 in the MB95470H Series).

Pin	Control bit	
TI0 pin	Data direction register (DDR5) (MB95410H Series)	Data direction specification bit (P52) (MB95410H Series)
	Data direction register (DDR1) (MB95470H Series)	Data direction specification bit (P14) (MB95470H Series)

● How to enable/disable the TO0 pin

The timer output enable bit (TMCSRL0:OUTE) is used.

Operation	Timer output enable bit (OUTE)
To enable the TO0 pin	Set the bit to "1".
To disable the TO0 pin	Set the bit to "0".

● How to generate a start trigger

- How to generate the software trigger

  The software trigger bit (TMCSRL0:TRG) is used.

  Writing "1" to the software trigger bit (TRG) generates a trigger.

  When enabling and starting operation at the same time, set the count enable bit (TMCSLR0:CNTE) and the software trigger bit (TMCSRL0:TRG) at the same time.
- How to generate an external trigger

  An external trigger is outputted when the edge specified by the operation mode selection bits is input to the trigger pin corresponding to each reload timer.

Timer	Trigger pin
Reload timer	TI0

● Interrupt-related registers

The interrupt level is set by the interrupt level setting register shown in the following table.

	Interrupt level setting register	Interrupt vector
Reload timer ch. 0	Interrupt level setting register (ILR2) Address: 0007B$_H$	#11 Address: 0FFE4$_H$

● How to enable interrupts

Interrupt request enable bit, Interrupt request flag

The interrupt request enable bit (TMCSRL0:INTE) is used to enable interrupts.

	Interrupt request enable bit (INTE)
When disabling interrupt requests	Set the bit to "0".
When enabling interrupt requests	Set the bit to "1".

The interrupt request bit (TMCSRL0:UF) is used to clear interrupt requests.

	Interrupt request bit (UF)
When disabling interrupt requests	Set the bit to "0".

CHAPTER 20

EVENT COUNTER

This chapter describes the functions and operations of the event counter.

20.1 Overview of Event Counter

The event counter is mainly used to measure the frequency of external clock with configurable measure period. 16-bit reload timer and 8/16-bit composite timer ch. 1 are configured to provide an event counter operation mode in the event counter.

■ Overview of Event Counter

The function of the event counter is summarized below.

● Event counter operation mode

In this mode, 8/16-bit composite timer ch. 1 is used to generate a PWM signal. Then external clock will be gated by this PWM signal, and then input to the 16-bit reload timer as count clock. 16-bit reload timer operates in external clock mode (reload mode). The frequency of external clock could be calculated with configured measure period in the interrupt service subroutine of 8/16-bit composite timer ch. 1.

Note:

In the following sections of this chapter, the term "composite timer" represents "8/16-bit composite timer ch. 1" and the term "reload timer" "16-bit reload timer".

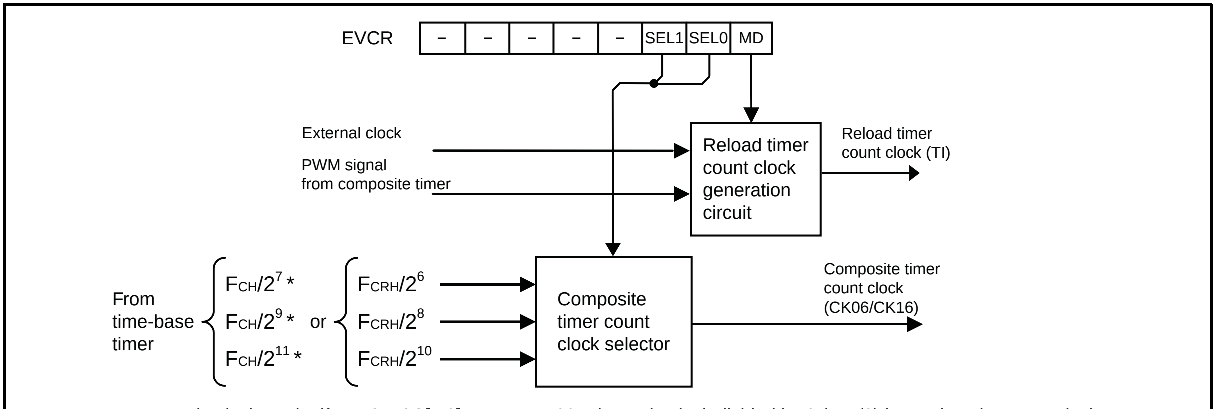
20.2 Configuration of Event Counter

The event counter consists of the following blocks:

- **Reload timer count clock generation circuit**
- **Composite timer count clock (CK06/CK16) selection circuit**
- **Event counter control register (EVCR)**

■ Block Diagram of Event Counter

Figure 20.2-1 Block Diagram of Event Counter


EVCR
–
–
–
–
–
SEL1
SEL0
MD
External clock
PWM signal
from composite timer
Reload timer count clock generation circuit
Reload timer count clock (TI)
From time-base timer
FCH/2^7 *
FCH/2^9 *
FCH/2^11 *
or
FCRH/2^6
FCRH/2^8
FCRH/2^10
Composite timer count clock selector
Composite timer count clock (CK06/CK16)

*: In main clock mode, if PLLC:PCS[1:0] are set to “00”, the main clock divided by 2 (F_{CH}/2) is used as the count clock; if PLLC:PCS[1:0] are set to “01” or “10” or “11”, the main PLL clock is used as the count clock.

● Reload timer count clock generation circuit

When the MD bit in the EVCR register is set to "1", external clock input is gated by PWM output from composite timer, then output to reload timer as count clock. When the MD bit is set to "0", external clock is output to reload timer directly as external clock.

● Composite timer count clock (CK06/CK16) selection circuit

The event counter uses one of the following time-base timer output signals (divided machine clock signal) as the CK06/CK16 count clock according to the settings of the SEL[1:0] bits in the EVCR register:

1. $F_{CH}/2^7$ or $F_{CH}/2^9$ or $F_{CH}/2^{11}$ (Main clock mode, PCS[1:0] = 00)
2. Main PLL clock divided by 2^6 or 2^8 or 2^{10} (Main clock mode, PCS[1:0] = 01, 10 or 11)
3. $F_{CRH}/2^6$, $F_{CRH}/2^8$ or $F_{CRH}/2^{10}$ (Main CR clock mode)

● Event counter control register (EVCR)

The event counter control register enables or disables the event counter operation mode and selects composite timer count clock source (CK06/CK16).

20.3 Register of Event Counter

This section describes the register of the event counter.

■ Event Counter Register

Figure 20.3-1 shows the register of the event counter.

Figure 20.3-1 Register of Event Counter

Event counter control register (EVCR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit0	bit0	Initial value
0FE2$_H$	-	-	-	-	-	SEL1	SEL0	MD	00000000$_B$
	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R/W	R/W	R/W	

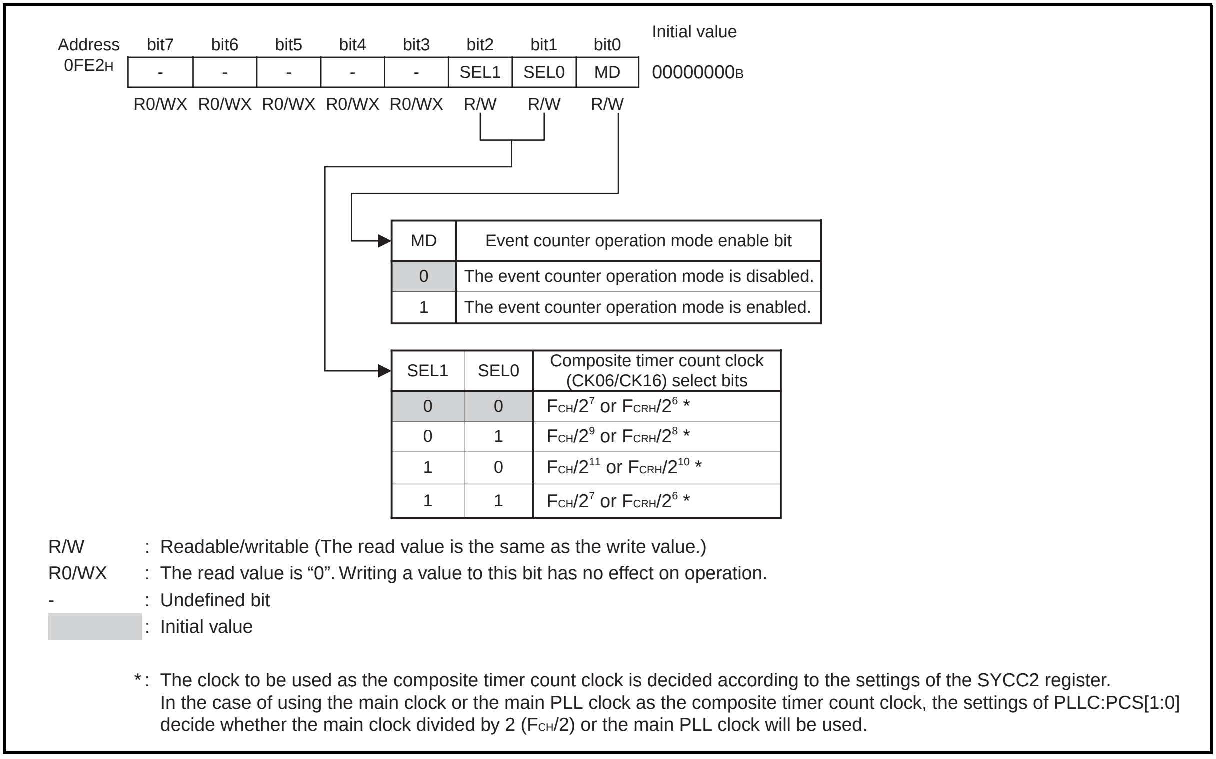
R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

20.3.1 Event Counter Control Register (EVCR)

The event counter control register (EVCR) enables or disables the event counter operation mode, and selects a count clock from the CK06/CK16 clock sources of the composite timer.

■ Event Counter Control Register (EVCR)

Figure 20.3-2 Event Counter Control Register (EVCR)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FE2H	-	-	-	-	-	SEL1	SEL0	MD	00000000B
	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R/W	R/W	R/W	

MD	Event counter operation mode enable bit
0	The event counter operation mode is disabled.
1	The event counter operation mode is enabled.

SEL1	SEL0	Composite timer count clock (CK06/CK16) select bits
0	0	$F_{CH}/2^7$ or $F_{CRH}/2^6$ *
0	1	$F_{CH}/2^9$ or $F_{CRH}/2^8$ *
1	0	$F_{CH}/2^{11}$ or $F_{CRH}/2^{10}$ *
1	1	$F_{CH}/2^7$ or $F_{CRH}/2^6$ *

R/W : Readable/writable (The read value is the same as the write value.)

R0/WX : The read value is “0”. Writing a value to this bit has no effect on operation.

- : Undefined bit

▨ : Initial value

*: The clock to be used as the composite timer count clock is decided according to the settings of the SYCC2 register.
In the case of using the main clock or the main PLL clock as the composite timer count clock, the settings of PLLC:PCS[1:0] decide whether the main clock divided by 2 (F_{CH}/2) or the main PLL clock will be used.

Table 20.3-1 Functions of Bits in Event Counter Control Register (EVCR)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7
to
bit3</td><td>Undefined bits</td><td>Their read values are always "0". Writing values to these bits has no effect on operation.</td></tr>
<tr><td>bit2,
bit1</td><td>SEL1, SEL0:
Composite timer count clock (CK06/CK16) select bits</td><td>These bits select the composite timer count clock (CK06/CK16).
• The count clock is generated by the prescaler. See "6.13 Operation of Prescaler".
• Write access to these bits is prohibited when composite timer and reload timer are in timer operation (T00CR1/T01CR1:STA = 1 or TMCSRL0:CNTE=1).
• These bits are in effect even if MD bit in the EVCR register is set to "0".
• The count clock from the time-base timer will be used as the count clock. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from either main clock or main CR clock. When the count clock from the time-base timer is used as the count clock, resetting the time-base timer by writing "1" to the time-base timer initialization bit in the time-base timer control register (TBTC:TCLR) will affect the count time.
<table>
<tr><th>SEL1</th><th>SEL0</th><th>Composite timer count clock (CK06/CK16)</th></tr>
<tr><td>0</td><td>0</td><td>$F_{CH}/2^7$or $F_{CRH}/2^6$ *</td></tr>
<tr><td>0</td><td>1</td><td>$F_{CH}/2^9$ or $F_{CH}/2^8$ *</td></tr>
<tr><td>1</td><td>0</td><td>$F_{CH}/2^{11}$ or $F_{CH}/2^{10}$ *</td></tr>
<tr><td>1</td><td>1</td><td>$F_{CH}/2^7$or $F_{CRH}/2^6$ *</td></tr>
</table>
*: The clock to be used as the composite timer count clock is decided according to the settings of the SYCC2 register. In the case of using the main clock or the main PLL clock as the composite timer count clock, the settings of PLLC:PCS[1:0] decide whether the main clock divided by 2 ($F_{CH}/2$) or the main PLL clock will be used.</td></tr>
<tr><td>bit0</td><td>MD:
Event counter operation mode select bit</td><td>This bit selects the event counter operation mode.
<b>Writing "0":</b> The event counter operation mode is disabled, and the composite timer and reload timer will work independently.
<b>Writing "1":</b> The event counter operation mode is enabled, and the composite timer and reload timer work together to implement the event counter function.
• Write access to this bit is prohibited when composite timer and reload timer are in timer operation (T00CR1/T01CR1:STA = 1 or TMCSRL0:CNTE=1).</td></tr>
</table>

20.4 Operation of Event Counter Operation Mode

This section describes the operation of the event counter operation mode.

■ Operation of Event Counter Operation Mode

The event counter, reload timer and composite timer require the register setting shown in Figure 20.4-1 to serve as an event counter (for frequency measurement).

Figure 20.4-1 Settings of Event Counter Operation Mode

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
Event Counter Register								
EVCR	-	-	-	-	-	SEL1	SEL0	MD
						❍	❍	1
Reload Timer Registers								
TMCSRH0	-	-	CSL2	CSL1	CSL0	MOD2	MOD1	MOD0
			1	1	1	❍*	❍*	❍*
TMCSRL0	-	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG
		×	×	1	❍*	❍	1	❍
TMRLRH0	Sets the reload value (upper)							
TMRLRL0	Sets the reload value (lower)							
Composite Timer Registers								
T10CR0/T11CR0	IFE	C2	C1	C0	F3	F2	F1	F0
	❍	1	1	0	0	1	0	0
T10CR1/T11CR1	STA	HO	IE	IR	BF	IF	SO	OE
	1	❍	❍	×	×	❍	×	×
TMCR1	TO1	TO0	TIS	MOD	FE11	FE10	FE01	FE00
	❍	❍	×	×	×	×	×	×
T10DR	Sets "L" pulse width (compare value)							
T11DR	Sets the cycle of PWM waveform (compare value)							

❍: Used bit

×: Unused bit

1: Set to "1"

0: Set to "0"

In event counter operation mode, the reload timer and the composite timer are used, therefore, they cannot be used for other function any more.

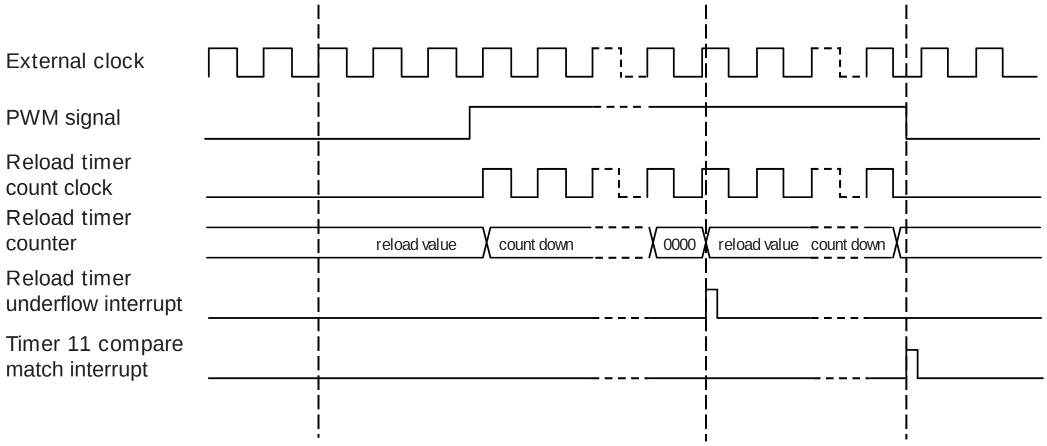
The reload timer should operate in event count mode (reload mode). In other words, TMCSRL0:MOD2 to MOD0 should be set one of the following values: "001_B", "010_B", "011_B", and TMCSRL0:RELD should be set "1". The reload timer interrupt should be enabled in order to record the reload timer underflow times.

The composite timer should operate in PWM operation mode (variable-cycle mode), count clock select source must be selected from CK06/CK16. It means T10CR0/11CR0:C2 to C0 must be set to "110_B", and T10CR0/11CR0:F3 to F0 must be set to "0100_B". In the composite timer, timer 11 interrupt should be enabled in order to calculate the frequency of external clock.

When the reload timer underflows, record the underflow times and clear underflow flag (UF) in reload timer interrupt service subroutine. When timer 11 interrupt occurs in the composite timer, clear IF flag in T11CR1, read the reload timer count value, and calculate the frequency of external clock in the interrupt service subroutine.

Figure 20.4-2 shows the operation of event counter operation mode.

Figure 20.4-2 Operation of Event Counter Operation Mode



Notes:

- In reload timer underflow interrupt service subroutine, clear the UF flag, count the times of underflow and write the reload register with reload value.
- In the timer 11 (composite timer ch. 1) compare match interrupt service subroutine, clear IF flag, read reload timer value and calculate the frequency.

Figure 20.4-3 shows the calculation of external clock frequency.

Figure 20.4-3 Calculation of external clock frequency

$$\text{Frequency of external clock} = \frac{\text{Count value of reload timer}}{\text{“H” pulse width of PWM signal}}$$

In the above expression:

$$\begin{aligned}\text{(count value of reload timer)} = {} & \text{(TMRLRH0/TMRLRL0 set value)} \times \text{(underflow times)} \\ & + \text{(TMRLRH0/TMRLRL0 set value)} - \text{(read value of TMRH0/TMRL0)}\end{aligned}$$

$$\text{(“H” pulse width of PWM signal)} = \frac{\text{(T11DR set value - T10DR set value)}}{\text{(frequency of composite timer count clock source)}}$$

20.5 Setting Procedure Example

This section describes the setting procedure example of the event counter function.

■ Setting Procedure Example

Below is an example of procedure for setting the event counter.

● Initial settings

1) Select the event counter operation mode. (EVCR:MD)

2) Select the composite timer CK06/CK16 source. (EVCR:SEL1, SEL0)

3) Set the interrupt level of reload timer and composite timer. (ILRx)

4) Set the reload value of reload timer. (TMRLRH0 and TMRLRL0)

5) Select the reload timer count clock. (TMCSRH0:CSL2 to CSL0)

6) Select the reload timer operation mode. (TMCSRH0:MOD2 to MOD0)

7) Select reload mode. (TMCSRL0:RELD)

8) Enable underflow interrupt. (TMCSRL0:INTE)

9) Enable reload timer count. (TMCSRL0:CNTE)

10) Perform the software trigger. (TMCSRL0:TRG=1)

11) Select composite timer operation mode. (T10CR0/T11CR0:F3 to F0)

12) Select composite timer count clock. (T10CR0/T11CR0:C2 to C0)

13) Enable the interrupt of timer 11. (T11CR1:IE)

14) Start the composite timer operation. (either T10CR1:STA or T11CR1:STA)

● Interrupt process of reload timer

1) Clear the underflow interrupt request flag. (TMCSRL0:UF)

2) Disable underflow interrupt. (TMCSRL0:INTE)

3) Record the underflow times.

4) Enable underflow interrupt. (TMCSRL0:INTE)

● Interrupt process of composite timer (timer 11)

1) Clear the interrupt request flag. (T11CR1:IF)

2) Disable the interrupt. (T11CR1:IE)

3) Read counter value of reload timer. (TMRH0, TMRL0)

4) Calculate the frequency of external clock.

5) Enable the interrupt. (T11CR1:IE)

20.6 Frequency Measurement Range and Precision

This section describes the frequency measurement range and precision of the event counter.

■ Frequency measurement range

The maximum measurable frequency is limited by peripheral resource clock. When peripheral resource clock frequency is F_{PCLK}, the maximum measurable frequency is $F_{PCLK}/4$.

The minimum measurable frequency is limited by the measure period, in order to ensure the frequency measurement precision.

■ Frequency measurement precision

The frequency measurement precision is determined by the main clock frequency and the precision of the reload timer counter. The more the reload timer counter counts, the more precise the calculated frequency becomes.

20.7 Notes on Using Event Counter

This section provides notes on using the event counter.

■ Notes on Using Event Counter

To switch the event counter operation mode with MD bit in the EVCR register, stop the composite timer and the reload timer first (T10CR1/T11CR1:STA=0, TMCSRL0:CNTE=0), then clear the interrupt flags (T10CR1/T11CR1:IF, IR, TMCSRL0:UF), and interrupt enable bits (T10CR1/T11CR1:IE, T10CR0/T11CR0:IFE, TMCSRL0:INTE) in the composite timer and the reload timer.

Set the L pulse width of PWM long enough so that the external clock frequency can be calculated within the interrupt service subroutine.

CHAPTER 21

8/16-BIT PPG

This chapter describes the functions and operations of the 8/16-bit PPG.

21.1 Overview of 8/16-bit PPG

The 8/16-bit PPG is an 8-bit reload timer module that uses pulse output control based on timer operation to perform PPG output. The 8/16-bit PPG also operates in cascade (8 bits + 8 bits) as a 16-bit PPG.

■ Overview of 8/16-bit PPG

The functions of the 8/16-bit PPG are summarized below.

● 8-bit PPG output independent operation mode

In this mode, the unit can operate as 2 8-bit PPG (PPG timer 00 and PPG timer 01).

● 8-bit prescaler + 8-bit PPG output operation mode

The rising and falling edge detection pulses from the PPG timer 01 output can be input to the downcounter of the PPG timer 00 to enable variable-cycle 8-bit PPG output.

● 16-bit PPG output operation mode

The unit can also operate in cascade (PPG timer 01 (upper 8 bits) + PPG timer 00 (lower 8 bits)) as 16-bit PPG output.

● PPG output operation

In this operation, a variable-cycle pulse waveform is output in any duty ratio.

The unit can also be used as a D/A converter in conjunction with an external circuit.

● Output inversion mode

This mode can invert the PPG output value.

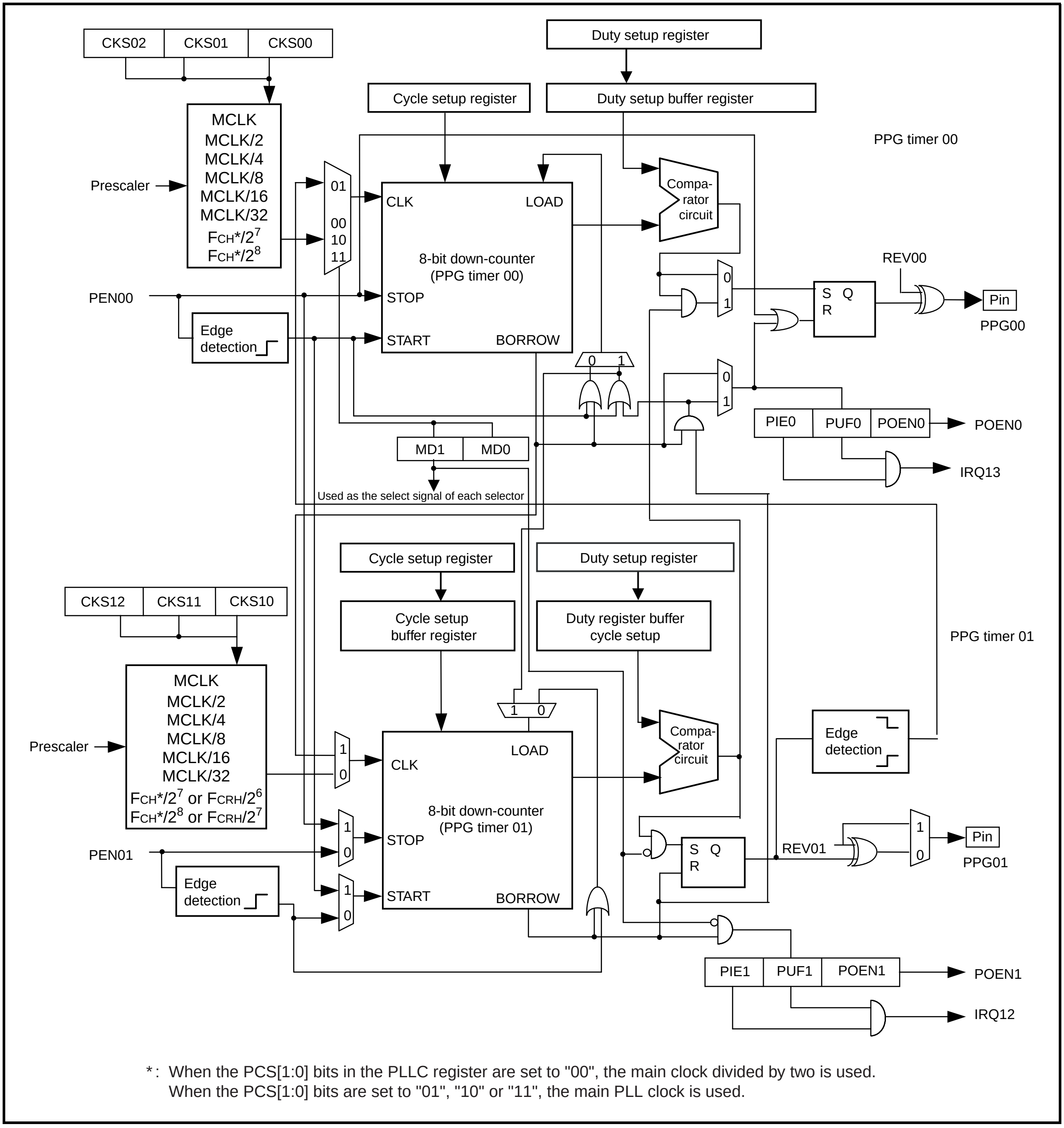
21.2 Configuration of 8/16-bit PPG

This section shows the block diagram of 8/16-bit PPG.

■ Block Diagram of 8/16-bit PPG

Figure 21.2-1 shows the block diagram of the 8/16-bit PPG.

Figure 21.2-1 Block Diagram of 8/16-bit PPG



*: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used.
When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

- Counter clock selector

  The clock for the countdown of 8-bit down counter is selected from eight types of internal count clocks.

- 8-bit downcounter

  It counts down with the count clock selected with the count clock selector.

- Comparator circuit

  The output is kept "H" level until the value of 8-bit down counter is corresponding to the value of 8/16-bit PPG duty setup buffer register from the value of 8/16-bit set buffer register of PPG cycle.

  Afterwards, after keep "L" level the output until the counter value is corresponding to "1", it keeps counting 8-bit down counter from the value of 8/16-bit PPG cycle setup buffer register.

- 8/16-bit PPG timer 01 control register (PC01)

  The operation condition on the PPG timer 01 side of 8/16-bit PPG timer is set.

- 8/16-bit PPG timer 00 control register (PC00)

  The operation mode of 8/16-bit PPG timer and the operation condition on the PPG timer 00 side are set.

- 8/16-bit PPG timer 01/00 cycle setup buffer register ch. 0 (PPS01, PPS00)

  The compare value for the cycle of 8/16-bit PPG timer is set.

- 8/16-bit PPG timer 01/00 duty setup buffer register ch. 0 (PDS01, PDS00)

  The compare value for "H" width of 8/16-bit PPG timer is set.

- 8/16-bit PPG start register

  The start or the stop of 8/16-bit PPG timer is set.

- 8/16-bit PPG output inversion register

  An initial level also includes the output of 8/16-bit PPG timer and it is reversed.

■ Input Clock

The 8/16-bit PPG uses the output clock from the prescaler as its input clock (count clock).

21.3 Channels of 8/16-bit PPG

This section describes the channels of the 8/16-bit PPG.

■ Channels of 8/16-bit PPG

The MB95410H/470H Series has two channels of 8/16-bit PPG. There are 8-bit PPG timer 00 and 8-bit PPG timer 01 in each channel. They can be used respectively as two 8-bit PPGs. Also, they can be used as a 16-bit PPG.

Table 21.3-1 and Table 21.3-2 show the channels and their corresponding pins and registers.

Table 21.3-1 Pins of 8/16-bit PPG

Channel	Pin name	Pin function
0	PPG00	PPG timer 00 output (8-bit PPG (00), 16-bit PPG)
	PPG01	PPG timer 01 output (8-bit PPG (01), 8-bit prescaler)
1	PPG10	PPG timer 10 output (8-bit PPG (10), 16-bit PPG)
	PPG11	PPG timer 11 output (8-bit PPG (11), 8-bit prescaler)

Table 21.3-2 Registers of 8/16-bit PPG

Channel	Register abbreviation	Corresponding register (Name in this manual)
0	PC01	8/16-bit PPG timer 01 control register
	PC00	8/16-bit PPG timer 00 control register
	PPS01	8/16-bit PPG timer 01 cycle setup buffer register
	PPS00	8/16-bit PPG timer 00 cycle setup buffer register
	PDS01	8/16-bit PPG timer 01 duty setup buffer register
	PDS00	8/16-bit PPG timer 00 duty setup buffer register
1	PC11	8/16-bit PPG timer 11 control register
	PC10	8/16-bit PPG timer 10 control register
	PPS11	8/16-bit PPG timer 11 cycle setup buffer register
	PPS10	8/16-bit PPG timer 10 cycle setup buffer register
	PDS11	8/16-bit PPG timer 11 duty setup buffer register
	PDS10	8/16-bit PPG timer 10 duty setup buffer register
Both channels	PPGS	8/16-bit PPG start register
	REVC	8/16-bit PPG output inversion register

The following sections describe only the 8/16-bit PPG on ch. 0.

21.4 Pins of 8/16-bit PPG

This section describes the pins of the 8/16-bit PPG.

■ Pins of 8/16-bit PPG

● PPG00 pin and PPG01 pin

These pins function both as general-purpose I/O ports and 8/16-bit PPG outputs.

PPG00, PPG01: A PPG waveform is output to these pins. The PPG waveform can be output by enabling the output by the 8/16-bit PPG timer 01/00 control registers (PC00: POEN0 = 1, PC01: POEN1 = 1).

■ Block Diagrams of Pins of 8/16-bit PPG

Figure 21.4-1 Block Diagram of PPG00 and PPG01 of 8/16-bit PPG

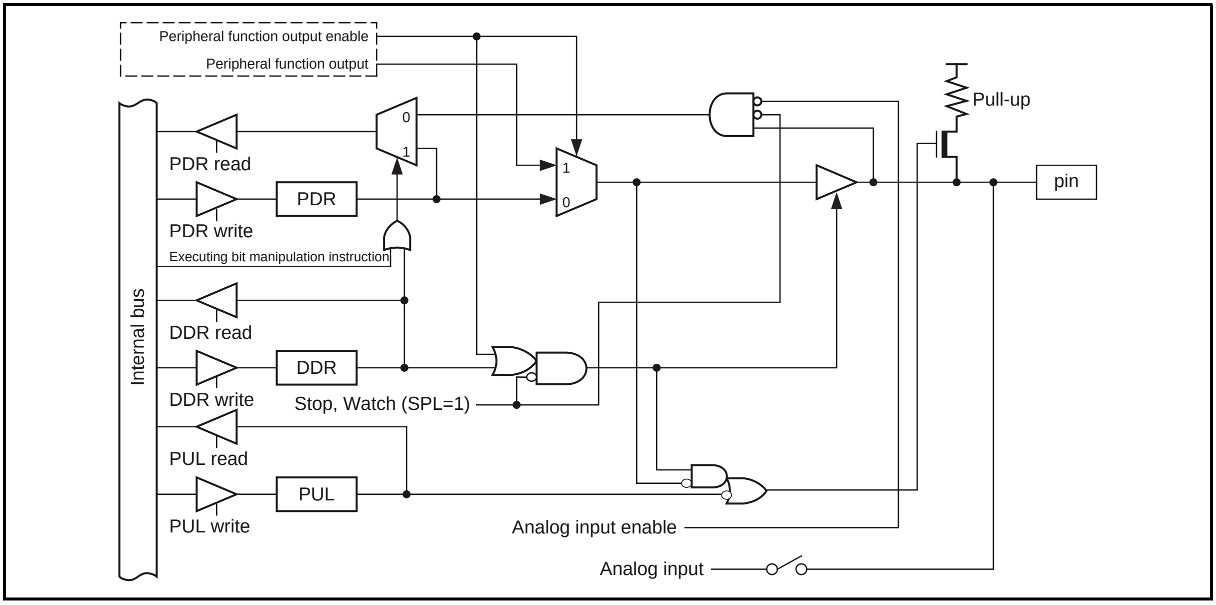
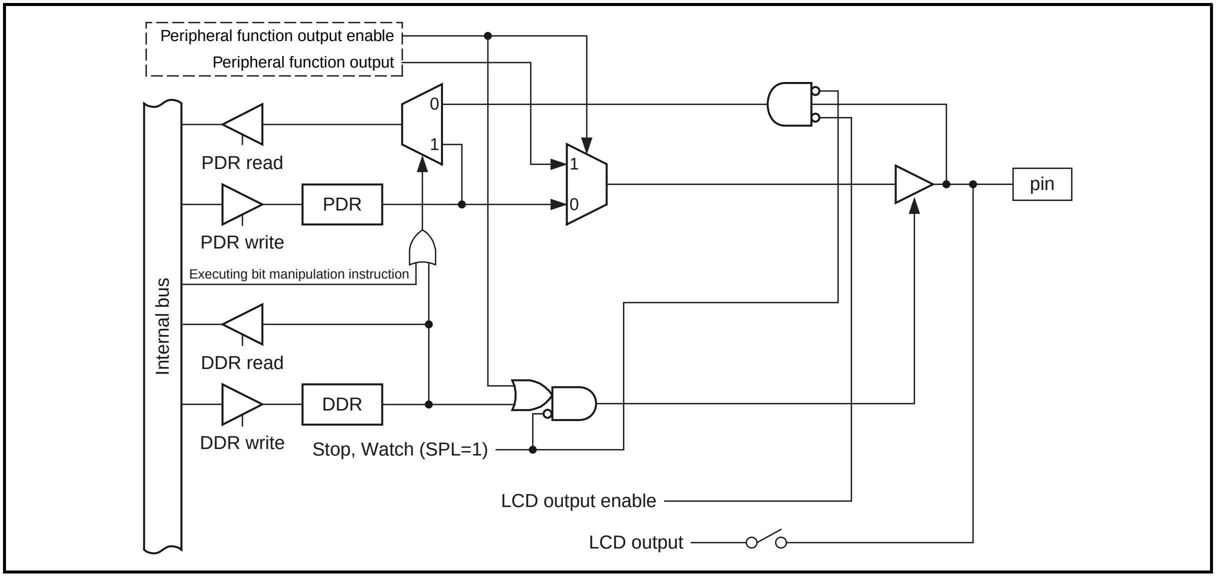


Figure 21.4-2 Block Diagram of PPG10 and PPG11 of 8/16-bit PPG



21.5 Registers of 8/16-bit PPG

This section describes the registers of the 8/16-bit PPG.

■ Registers of 8/16-bit PPG

Figure 21.5-1 shows the registers of the 8/16-bit PPG.

Figure 21.5-1 Registers of 8/16-bit PPG

8/16-bit PPG timer 01 control register (PC01)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$003A_H$	-	-	PIE1	PUF1	POEN1	CKS12	CKS11	CKS10	00000000_B
	R0/WX	R0/WX	R/W	R(RM1),W	R/W	R/W	R/W	R/W	

8/16-bit PPG timer 00 control register (PC00)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$003B_H$	MD1	MD0	PIE0	PUF0	POEN0	CKS02	CKS01	CKS00	00000000_B
	R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W	R/W	

8/16-bit PPG timer 01 cycle setup buffer register (PPS01)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F9C_H$	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0	11111111_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/16-bit PPG timer 00 cycle setup buffer register (PPS00)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F9D_H$	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	11111111_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/16-bit PPG timer 01 duty setup buffer register (PDS01)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F9E_H$	DH7	DH6	DH5	DH4	DH3	DH2	DH1	DH0	11111111_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/16-bit PPG timer 00 duty setup buffer register (PDS00)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0F9F_H$	DL7	DL6	DL5	DL4	DL3	DL2	DL1	DL0	11111111_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/16-bit PPG start register (PPGS)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FA4_H$	-	-	-	-	PEN11	PEN10	PEN01	PEN00	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/16-bit PPG output inversion register (REVC)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FA5_H$	-	-	-	-	REV11	REV10	REV01	REV00	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

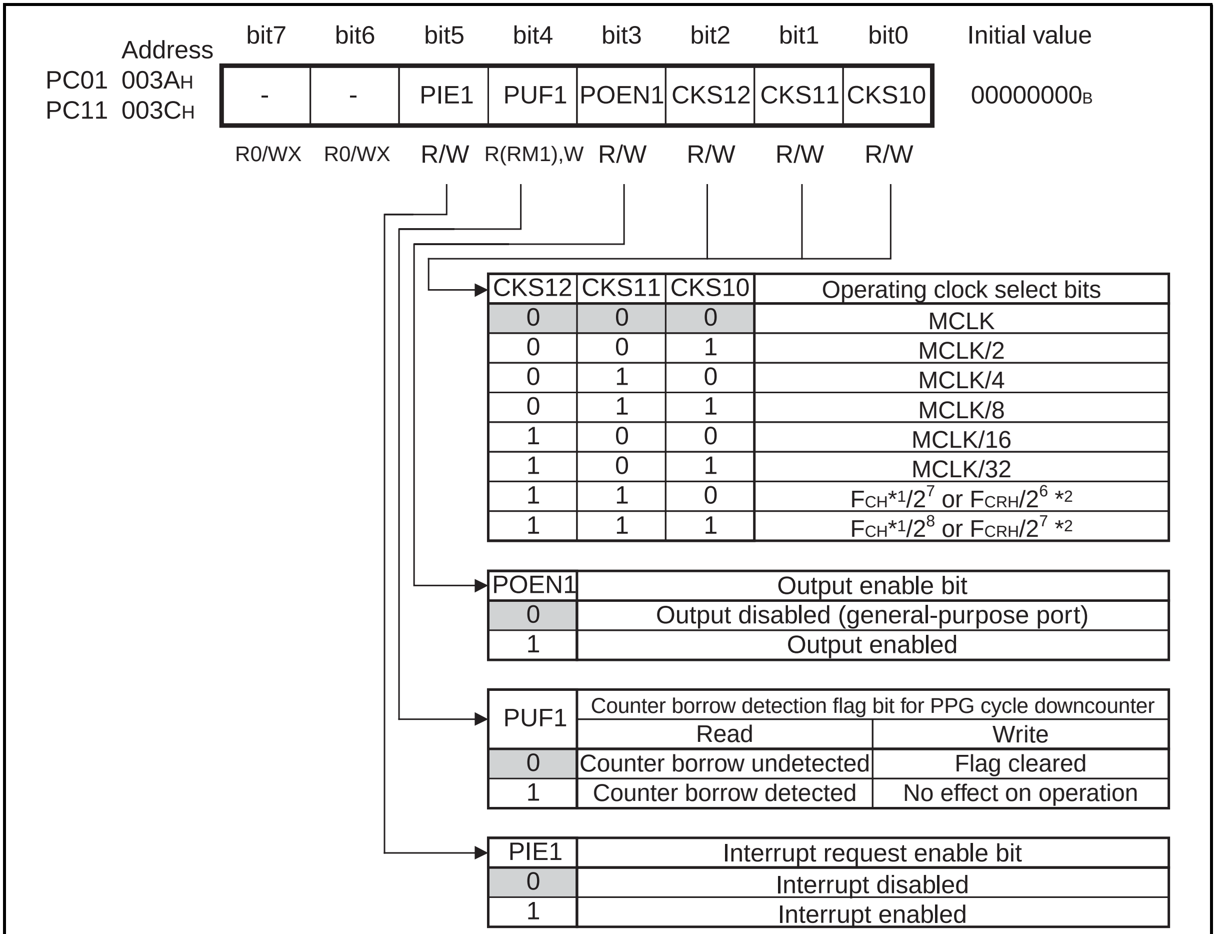
R/W	: Readable/writable (The read value is the same as the write value.)
R(RM1), W	: Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0/WX	: The read value is "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

21.5.1 8/16-bit PPG Timer 01 Control Register (PC01)

The 8/16-bit PPG timer 01 control register (PC01) sets the operating conditions for PPG timer 01.

■ 8/16-bit PPG Timer 01 Control Register (PC01)

Figure 21.5-2 8/16-bit PPG Timer 01 Control Register (PC01)



	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
PC01 PC11	003AH 003CH	-	-	PIE1	PUF1	POEN1	CKS12	CKS11	CKS10	00000000B
		R0/WX	R0/WX	R/W	R(RM1),W	R/W	R/W	R/W	R/W	

CKS12	CKS11	CKS10	Operating clock select bits
0	0	0	MCLK
0	0	1	MCLK/2
0	1	0	MCLK/4
0	1	1	MCLK/8
1	0	0	MCLK/16
1	0	1	MCLK/32
1	1	0	FCH*1/2^7 or FCRH/2^6 *2
1	1	1	FCH*1/2^8 or FCRH/2^7 *2

POEN1	Output enable bit
0	Output disabled (general-purpose port)
1	Output enabled

PUF1	Counter borrow detection flag bit for PPG cycle downcounter	
	Read	Write
0	Counter borrow undetected	Flag cleared
1	Counter borrow detected	No effect on operation

PIE1	Interrupt request enable bit
0	Interrupt disabled
1	Interrupt enabled

MCLK	:	Machine clock
FCH	:	Main clock
FCRH	:	Main CR clock
R/W	:	Readable/writable (The read value is the same as the write value.)
R(RM1),W	:	Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0/WX	:	The read value is "0". Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

*1 : When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used. When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

*2 : The value to be used as the operating clock is decided according to the settings of the SYCC2 register.

Table 21.5-1 Functions of Bits in 8/16-bit PPG Timer 01 Control Register (PC01)

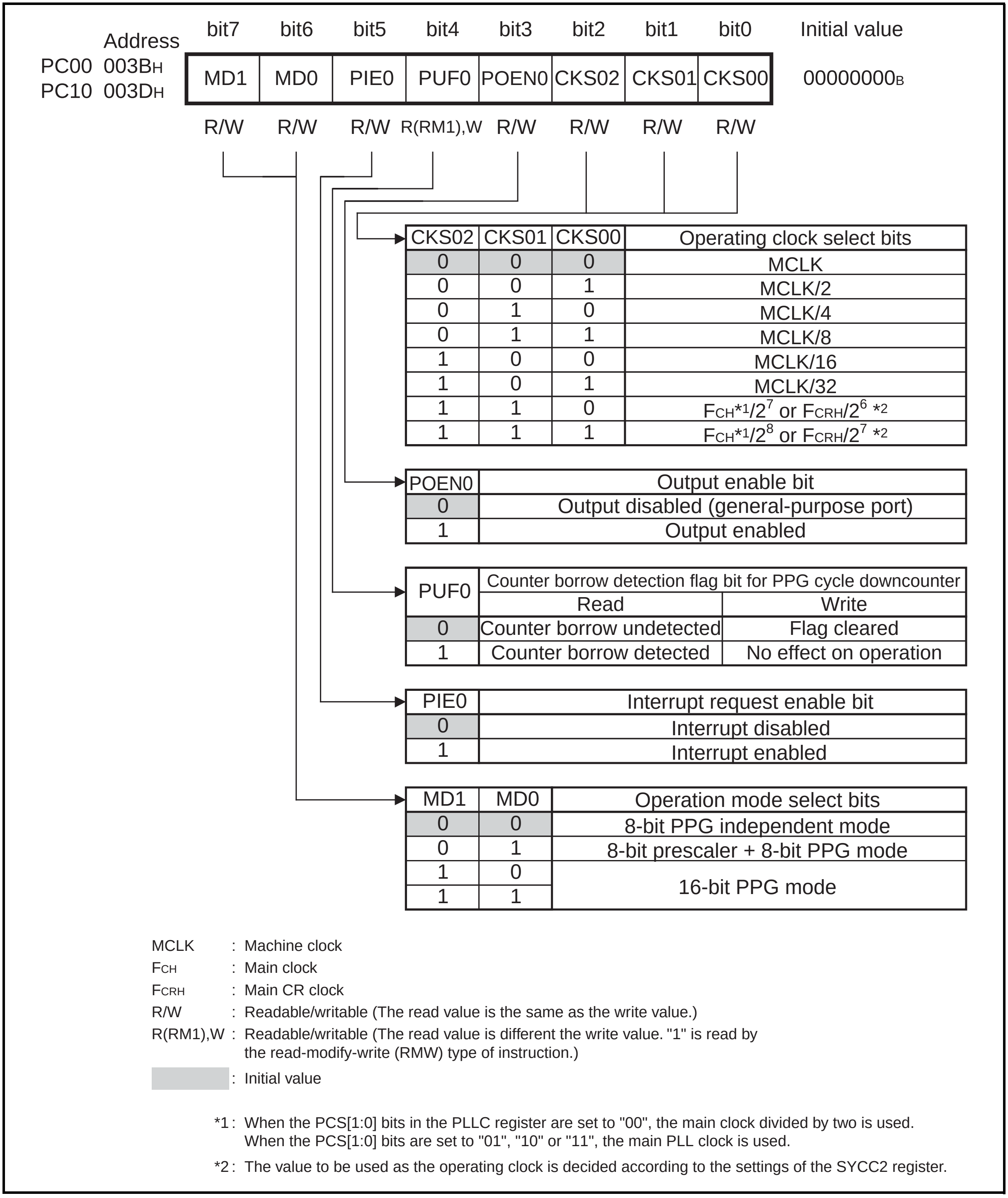
Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5	PIE1: Interrupt request enable bit	This bit controls interrupts of PPG timer 01. **Writing "0":** Disables interrupts of PPG timer 01. **Writing "1":** Enables interrupts of PPG timer 01. The bit outputs an interrupt request (IRQ12) when the counter borrow detection bit (PUF1) and the PIE1 bit are both set to "1".
bit4	PUF1: Counter borrow detection flag bit for PPG cycle downcounter	This bit serves as the counter borrow detection flag for the PPG cycle downcounter of PPG timer 01. • This bit is set to "1" when a counter borrow occurs during 8-bit prescaler + 8-bit PPG mode. • In 16-bit PPG mode, this bit is not set to "1" even when a counter borrow occurs. • Writing "1" to the bit is meaningless. • Writing "0" clears the bit. • "1" is read in read-modify-write (RMW) instruction. **Reading "0":** No counter borrow of PPG timer 01 is detected. **Reading "1":** A counter borrow of PPG timer 01 is detected.
bit3	POEN1: Output enable bit	This bit enables or disables the output of PPG timer 01 pin. **Writing "0":** The PPG timer 01 pin is used as a general-purpose port. **Writing "1":** The PPG timer 01 pin is used as the PPG output pin. Setting this bit to "1" during 16-bit PPG operation mode sets the PPG timer 01 pin as an output pin. (The setting value of REV01 is output. "L" output is supplied when REV01 is "0".)
bit2 to bit0	CKS12, CKS11, CKS10: Operating clock select bits	These bits select the operating clock for 8-bit downcounter of PPG timer 01. • The operating clock is generated from the prescaler. See "CHAPTER 6 CLOCK CONTROLLER". • In 16-bit PPG operation mode, the settings of these bits have no effect on the operation. **"000_B":** MCLK **"001_B":** MCLK/2 **"010_B":** MCLK/4 **"011_B":** MCLK/8 **"100_B":** MCLK/16 **"101_B":** MCLK/32 **"110_B":** F_{CH}*/2^7 or $F_{CRH}/2^6$ **"111_B":** F_{CH}*/2^8 or $F_{CRH}/2^7$ Note: • The use of the subclock will stop the time-base timer operation. Therefore, selecting "110_B" or "111_B" is prohibited. • When these bits are set to "110_B" or "111_B", the count clock from the time-base timer will be used as the operating clock. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock. In the case of using the count clock from the time-base timer as the operating clock, resetting the time-base timer by writing "1" to the time-base timer initialization bit in the time-base timer control register (TBTC:TCLR) will affect the count time. *: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used. When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

21.5.2 8/16-bit PPG Timer 00 Control Register (PC00)

The 8/16-bit PPG timer 00 control register (PC00) sets the operating conditions and the operation mode for PPG timer 00.

■ 8/16-bit PPG Timer 00 Control Register (PC00)

Figure 21.5-3 8/16-bit PPG Timer 00 Control Register (PC00)



	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
PC00	003B$_H$	MD1	MD0	PIE0	PUF0	POEN0	CKS02	CKS01	CKS00	00000000$_B$
PC10	003D$_H$									
		R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W	R/W	

CKS02	CKS01	CKS00	Operating clock select bits
0	0	0	MCLK
0	0	1	MCLK/2
0	1	0	MCLK/4
0	1	1	MCLK/8
1	0	0	MCLK/16
1	0	1	MCLK/32
1	1	0	F_{CH}*1/2^7 or F_{CRH}/2^6 *2
1	1	1	F_{CH}*1/2^8 or F_{CRH}/2^7 *2

POEN0	Output enable bit
0	Output disabled (general-purpose port)
1	Output enabled

PUF0	Counter borrow detection flag bit for PPG cycle downcounter	
	Read	Write
0	Counter borrow undetected	Flag cleared
1	Counter borrow detected	No effect on operation

PIE0	Interrupt request enable bit
0	Interrupt disabled
1	Interrupt enabled

MD1	MD0	Operation mode select bits
0	0	8-bit PPG independent mode
0	1	8-bit prescaler + 8-bit PPG mode
1	0	16-bit PPG mode
1	1	

MCLK	:	Machine clock
F_{CH}	:	Main clock
F_{CRH}	:	Main CR clock
R/W	:	Readable/writable (The read value is the same as the write value.)
R(RM1),W	:	Readable/writable (The read value is different the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
▨	:	Initial value

*1 : When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used.
When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

*2 : The value to be used as the operating clock is decided according to the settings of the SYCC2 register.

Table 21.5-2 Functions of Bits in 8/16-bit PPG Timer 00 Control Register (PC00)

Bit name		Function
bit7, bit6	MD1, MD0: Operation mode select bits	These bits select the PPG operation mode. Do not modify the bit settings during counting. **Writing "00_B":** 8-bit PPG independent mode **Writing "01_B":** 8-bit prescaler + 8-bit PPG mode **Writing "10_B" or "11_B":** 16-bit PPG mode
bit5	PIE0: Interrupt request enable bit	This bit controls interrupts of PPG timer 00. • Set this bit in 16-bit PPG operation mode. **Writing "0":** Disables interrupts of PPG timer 00. **Writing "1":** Enables interrupts of PPG timer 00. • An interrupt request (IRQ13) is output when the counter borrow detection bit (PUF0) and PIE0 bit are both set to "1".
bit4	PUF0: Counter borrow detection flag bit for PPG cycle downcounter	This is the counter borrow detection flag for the PPG cycle downcounter of PPG timer 00. • Only this bit is effective in 16-bit PPG operation mode (PC1:PUF1 is not operable). Note: Always effective in 8-bit mode • Writing "1" to this bit has no effect on operation. • Writing "0" clears the bit. • "1" is read by the read-modify-write (RMW) type of instruction. **Reading "0":** No counter borrow of PPG timer 00 is detected. **Reading "1":** A counter borrow of PPG timer 00 has been detected.
bit3	POEN0: Output enable bit	This bit enables or disables the output of PPG timer 00 pin. **Writing "0":** PPG timer 00 pin is used as a general-purpose port. **Writing "1":** PPG timer 00 pin is used as the PPG output pin. As the output is supplied from the PPG timer 00 pin in 16-bit PPG operation mode, this bit is used to control the operation.
bit2 to bit0	CKS02, CKS01, CKS00: Operating clock select bits	These bits select the operating clock for PPG downcounter of PPG timer 00. • The operating clock is generated from the prescaler. See "CHAPTER 6 CLOCK CONTROLLER". • The rising and falling edge detection pulses from the PPG timer 01 output are used as the count clock for PPG timer 00 when the 8-bit prescaler + 8-bit PPG mode has been selected. Therefore, the setting of this bit has no effect on the operation. • Set these bits in 16-bit PPG operation mode. **"000_B":** MCLK **"001_B":** MCLK/2 **"010_B":** MCLK/4 **"011_B":** MCLK/8 **"100_B":** MCLK/16 **"101_B":** MCLK/32 **"110_B":** F_{CH}*****/2^7 or $F_{CRH}/2^6$ **"111_B":** F_{CH}*****/2^8 or $F_{CRH}/2^7$ Note: • The use of the subclock will stop the time-base timer operation. Therefore, selecting "110_B" or "111_B" is prohibited. • When these bits are set to "110_B" or "111_B", the count clock from the time-base timer will be used as the operating clock. Depending on the settings of the SYCC2 register, the count clock from the time-base timer can be generated from the main clock, the main PLL clock or the main CR clock. In the case of using the count clock from the time-base timer as the operating clock, resetting the time-base timer by writing "1" to the time-base timer initialization bit in the time-base timer control register (TBTC:TCLR) will affect the count time. *: When the PCS[1:0] bits in the PLLC register are set to "00", the main clock divided by two is used. When the PCS[1:0] bits are set to "01", "10" or "11", the main PLL clock is used.

21.5.3 8/16-bit PPG Timer 00/01 Cycle Setup Buffer Register (PPS01, PPS00)

The 8/16-bit PPG timer 00/01 cycle setup buffer register (PPS01, PPS00) sets the PPG output cycle.

■ 8/16-bit PPG Timer 00/01 Cycle Setup Buffer Register (PPS01, PPS00)

Figure 21.5-4 8/16-bit PPG Timer 00/01 Cycle Setup Buffer Register (PPS01, PPS00)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
PPS01	0F9C$_H$	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0	11111111$_B$
PPS11	0FA0$_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
PPS00	0F9D$_H$	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	11111111$_B$
PPS10	0FA1$_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

This register is used to set the PPG output cycle.

- In 16-bit PPG mode, PPS01 serves as the upper 8 bits, while PPS00 serves as the lower 8 bits.
- In 16-bit PPG mode, write the upper bits before the lower bits. When only the upper bits are written, the previously written value is reused in the next load.
- 8-bit mode: Cycle = max. 255 (FF$_H$) × Input clock cycle
- 16-bit mode: Cycle = max. 65535 (FFFF$_H$) × Input clock cycle
- PPS01 and PPS00 are initialized upon reset.
- Do not set the cycle to "00$_H$" or "01$_H$" when using the unit in 8-bit PPG independent mode, or in 8-bit prescaler mode + 8-bit PPG mode
- Do not set the cycle to "0000$_H$" or "0001$_H$" when using the unit in 16-bit PPG mode.
- If the cycle settings are modified during the operation, the modified settings will be effective from the next PPG cycle.

21.5.4 8/16-bit PPG Timer 00/01 Duty Setup Buffer Register (PDS01, PDS00)

The 8/16-bit PPG timer 00/01 duty setup buffer register (PDS01, PDS00) sets the duty of the PPG output.

■ 8/16-bit PPG Timer 00/01 Duty Setup Buffer Register (PDS01, PDS00)

Figure 21.5-5 8/16-bit PPG Timer 00/01 Duty Setup Buffer Register (PDS01, PDS00)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
PDS01	$0F9E_H$	DH7	DH6	DH5	DH4	DH3	DH2	DH1	DH0	11111111_B
PDS11	$0FA2_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
PDS00	$0F9F_H$	DL7	DL6	DL5	DL4	DL3	DL2	DL1	DL0	11111111_B
PDS10	$0FA3_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

This register is used to set the duty of the PPG output ("H" pulse width when normal polarity).

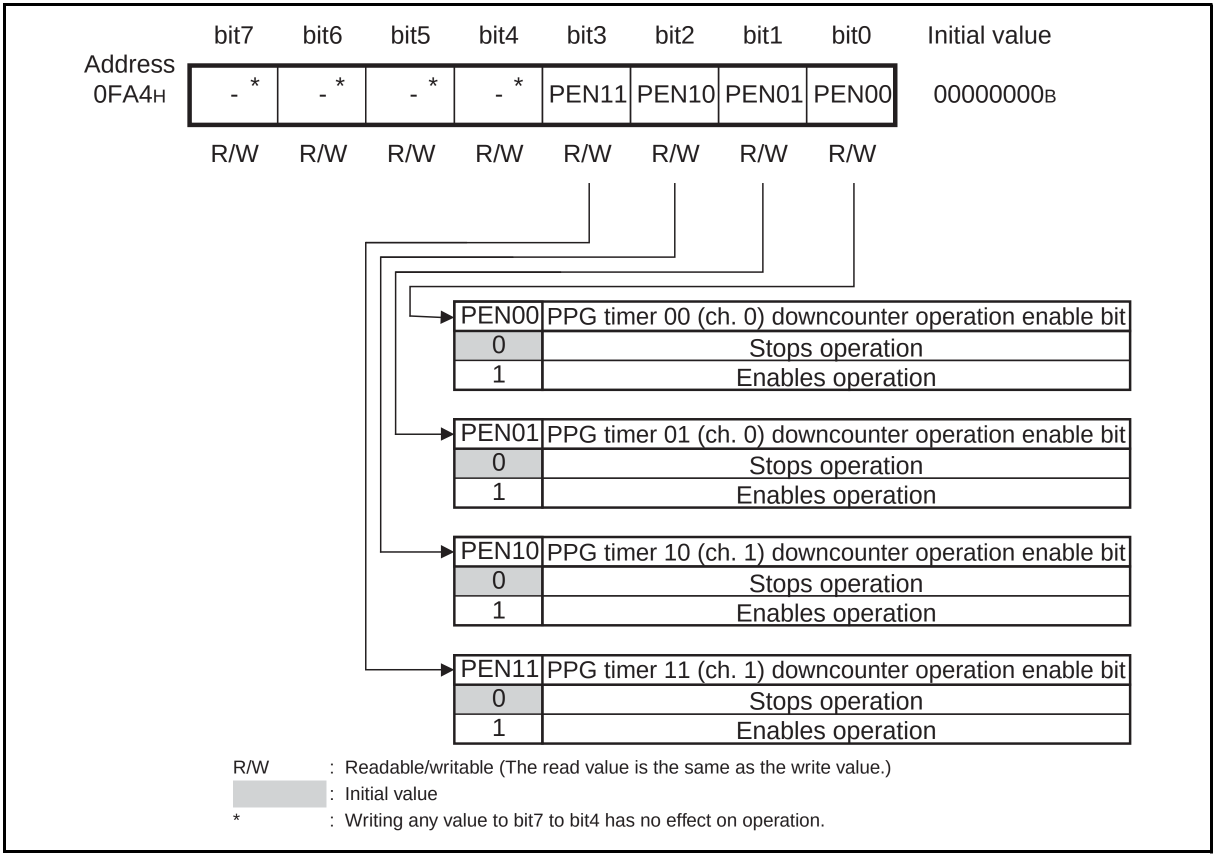
- In 16-bit PPG mode, PDS01 serves as the upper 8 bits while PDS00 serves as the lower 8 bits.
- In 16-bit PPG mode, write the upper bits before the lower bits. When only the upper bits are written, the previously written value is reused in the next load. Writing data to PDF00 also updates PDS01 at the same time.
- PDS01 and PDS00 are initialized at reset.
- To set the duty to 0%, select "00_H".
- To set the duty to 100%, set it to the same value as the 8/16-bit PPG timer 00/01 cycle setup register (PPS00, PPS01).
- When the 8/16-bit PPG timer 00/01 duty setup register (PDS) is set to a larger value than the setting value of the 8/16-bit PPG cycle setup buffer register (PPS), the PPG output becomes "L" output in the normal polarity (when the output level inversion bit of 8/16-bit PPG output inversion register is "0").
- If the duty settings are modified during operation, the modified value will be effective from the next PPG cycle.

21.5.5 8/16-bit PPG Start Register (PPGS)

The 8/16-bit PPG start register (PPGS) starts or stops the downcounter. The operation enable bit of each channel is assigned to the PPGS register, allowing simultaneous activation of the PPG channels.

■ 8/16-bit PPG Start Register (PPGS)

Figure 21.5-6 8/16-bit PPG Start Register (PPGS)



	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address 0FA4H	- *	- *	- *	- *	PEN11	PEN10	PEN01	PEN00	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

PEN00	PPG timer 00 (ch. 0) downcounter operation enable bit
0	Stops operation
1	Enables operation

PEN01	PPG timer 01 (ch. 0) downcounter operation enable bit
0	Stops operation
1	Enables operation

PEN10	PPG timer 10 (ch. 1) downcounter operation enable bit
0	Stops operation
1	Enables operation

PEN11	PPG timer 11 (ch. 1) downcounter operation enable bit
0	Stops operation
1	Enables operation

R/W : Readable/writable (The read value is the same as the write value.)

▨ : Initial value

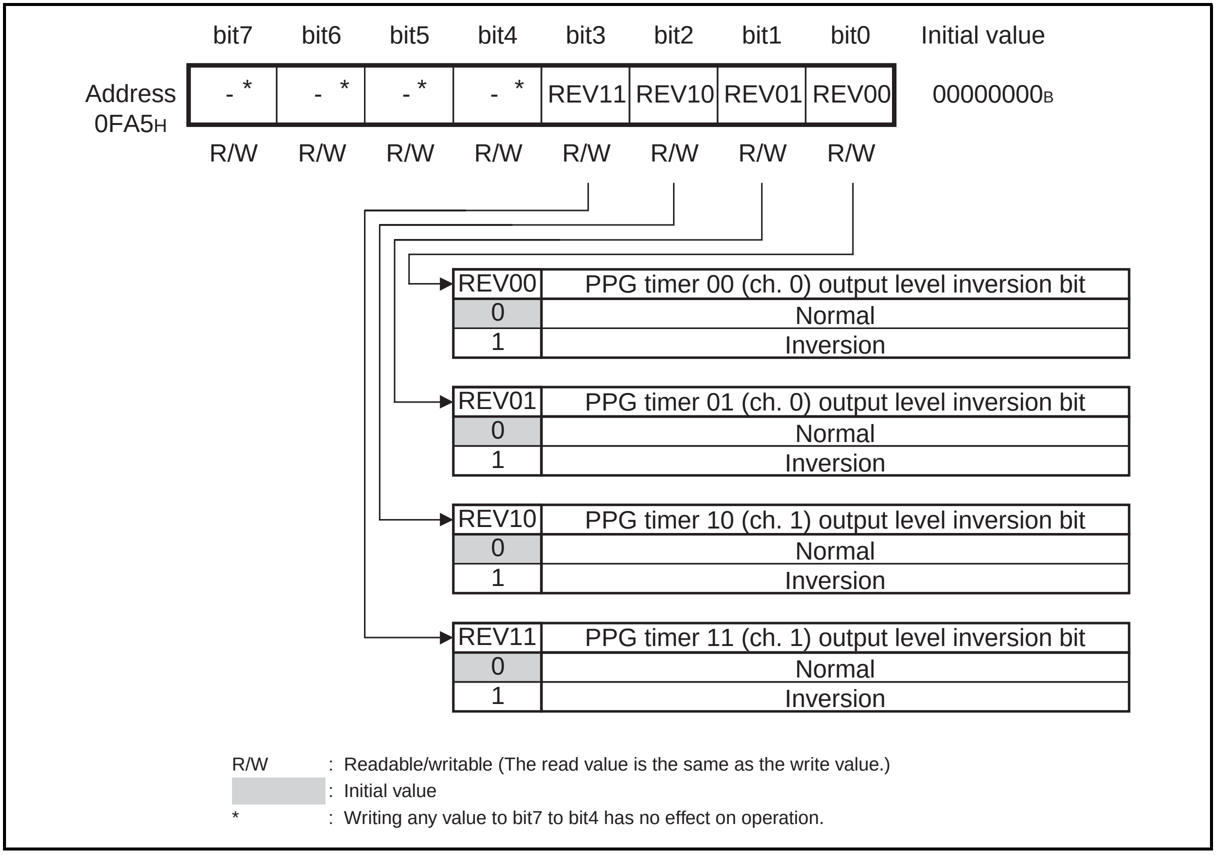
* : Writing any value to bit7 to bit4 has no effect on operation.

21.5.6 8/16-bit PPG Output Inversion Register (REVC)

The 8/16-bit PPG output inversion register (REVC) inverts the PPG output including the initial level.

■ 8/16-bit PPG Output Inversion Register (REVC)

Figure 21.5-7 8/16-bit PPG Output Inversion Register (REVC)



	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address 0FA5H	- *	- *	- *	- *	REV11	REV10	REV01	REV00	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

REV00	PPG timer 00 (ch. 0) output level inversion bit
0	Normal
1	Inversion

REV01	PPG timer 01 (ch. 0) output level inversion bit
0	Normal
1	Inversion

REV10	PPG timer 10 (ch. 1) output level inversion bit
0	Normal
1	Inversion

REV11	PPG timer 11 (ch. 1) output level inversion bit
0	Normal
1	Inversion

R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value
* : Writing any value to bit7 to bit4 has no effect on operation.

21.6 Interrupts of 8/16-bit PPG

The 8/16-bit PPG outputs an interrupt request when a counter borrow is detected.

■ Interrupts of 8/16-bit PPG

Table 21.6-1 shows the interrupt control bits and interrupt sources of the 8/16-bit PPG.

Table 21.6-1 Interrupt Control Bits and Interrupt Sources of 8/16-bit PPG

Item	Description	
	PPG timer 01 (8-bit PPG, 8-bit prescaler)	PPG timer 00 (8-bit PPG, 16-bit PPG)
Interrupt request flag bit	PUF1 bit in PC01	PUF0 bit in PC00
Interrupt request enable bit	PIE1 bit in PC01	PIE0 bit in PC00
Interrupt source	Counter borrow of PPG cycle downcounter	

When a counter borrow occurs on the downcounter, the 8/16-bit PPG sets the counter borrow detection flag bit (PUF) in the 8/16-bit PPG timer 00/01 control register (PC) to "1". When the interrupt request enable bit is enabled (PIE = 1), an interrupt request is output to the interrupt controller.

In 16-bit PPG mode, the 8/16-bit PPG timer 00 control register (PC00) is available.

■ Registers and Vector Table Addresses Related to Interrupts of 8/16-bit PPG

Table 21.6-2 Registers and Vector Table Addresses Related to Interrupts of 8/16-bit PPG

Interrupt source	Interrupt request no.	Interrupt level setup register		Vector table address	
		Register	Setting bit	Upper	Lower
8/16-bit PPG ch. 1 (lower)*	IRQ09	ILR2	L09	FFE8$_H$	FFE9$_H$
8/16-bit PPG ch. 1 (upper)	IRQ10	ILR2	L10	FFE6$_H$	FFE7$_H$
8/16-bit PPG ch. 0 (upper)	IRQ12	ILR3	L12	FFE2$_H$	FFE3$_H$
8/16-bit PPG ch. 0 (lower)	IRQ13	ILR3	L13	FFE0$_H$	FFE1$_H$

ch.: Channel

*: 8/16-bit PPG ch. 1 (lower) uses the same interrupt request number and vector table addresses as UART/SIO ch. 1.

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

21.7 Operations of 8/16-bit PPG and Setting Procedure Example

This section describes the operations of the 8/16-bit PPG.

■ Setting Procedure Example

Below is an example of procedure for setting the 8/16-bit PPG.

● Initial settings

1) Set the port output. (DDR1, DDR2)
2) Set the interrupt level (ILR2, ILR3)
3) Select the operating clock, enable the output and interrupt. (PC01)
4) Select the operating clock, enable the output and interrupt, select the operation mode. (PC00)
5) Set the cycle. (PPS)
6) Set the duty. (PDS)
7) Set the output inversion. (REVC)
8) Start PPG. (PPGS)

● Interrupt processing

1) Process any interrupt.
2) Clear the interrupt request flag. (PC01: PUF1, PC00: PUF0)
3) Start PPG. (PPGS)

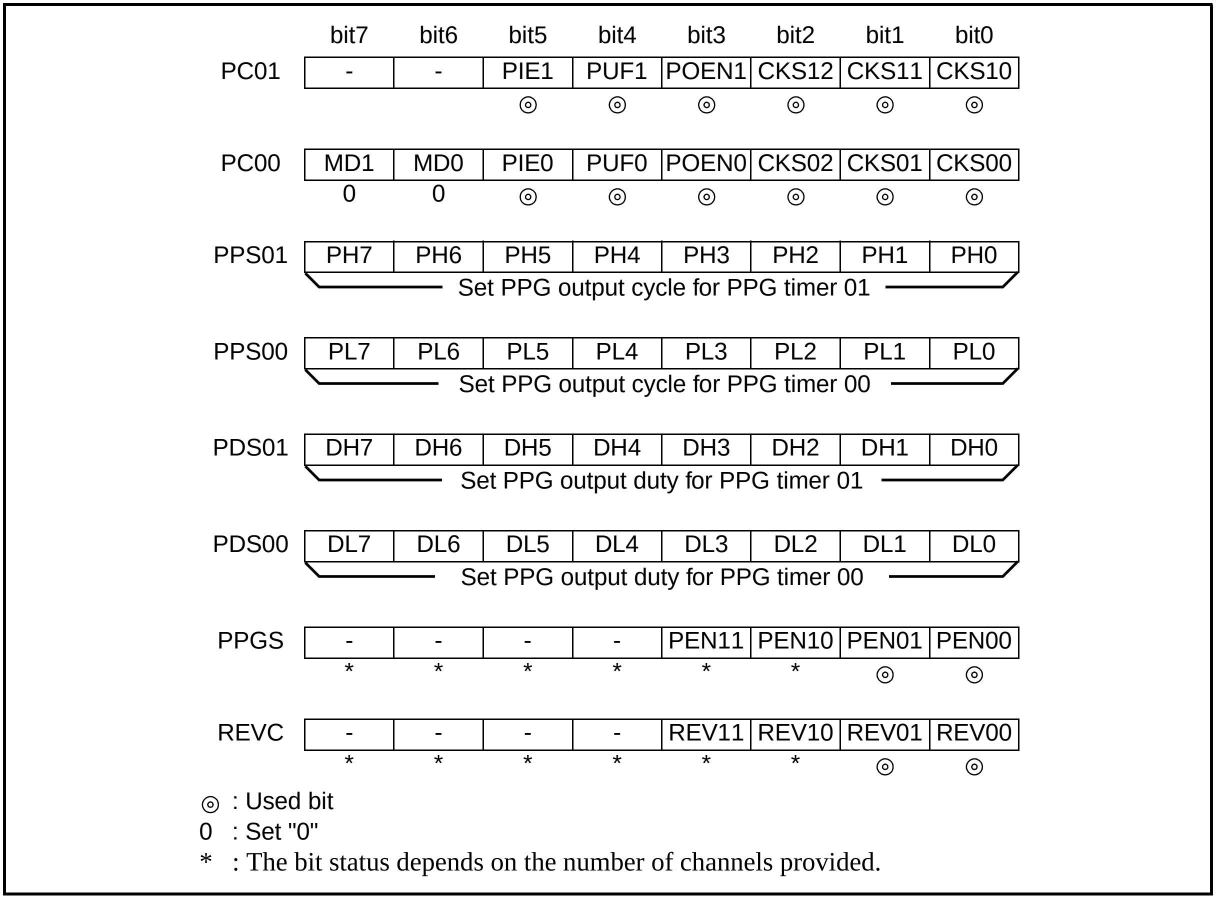
21.7.1 8-bit PPG Independent Mode

In this mode, the unit operates as two channels (PPG timer 00 and PPG timer 01) of the 8-bit PPG.

■ Setting 8-bit Independent Mode

The unit requires the register settings shown in Figure 21.7-1 to operate in 8-bit independent mode.

Figure 21.7-1 8-bit Independent Mode



	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PC01	-	-	PIE1	PUF1	POEN1	CKS12	CKS11	CKS10
			◎	◎	◎	◎	◎	◎
PC00	MD1	MD0	PIE0	PUF0	POEN0	CKS02	CKS01	CKS00
	0	0	◎	◎	◎	◎	◎	◎
PPS01	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
	Set PPG output cycle for PPG timer 01							
PPS00	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
	Set PPG output cycle for PPG timer 00							
PDS01	DH7	DH6	DH5	DH4	DH3	DH2	DH1	DH0
	Set PPG output duty for PPG timer 01							
PDS00	DL7	DL6	DL5	DL4	DL3	DL2	DL1	DL0
	Set PPG output duty for PPG timer 00							
PPGS	-	-	-	-	PEN11	PEN10	PEN01	PEN00
	*	*	*	*	*	*	◎	◎
REVC	-	-	-	-	REV11	REV10	REV01	REV00
	*	*	*	*	*	*	◎	◎

◎ : Used bit
0 : Set "0"
* : The bit status depends on the number of channels provided.

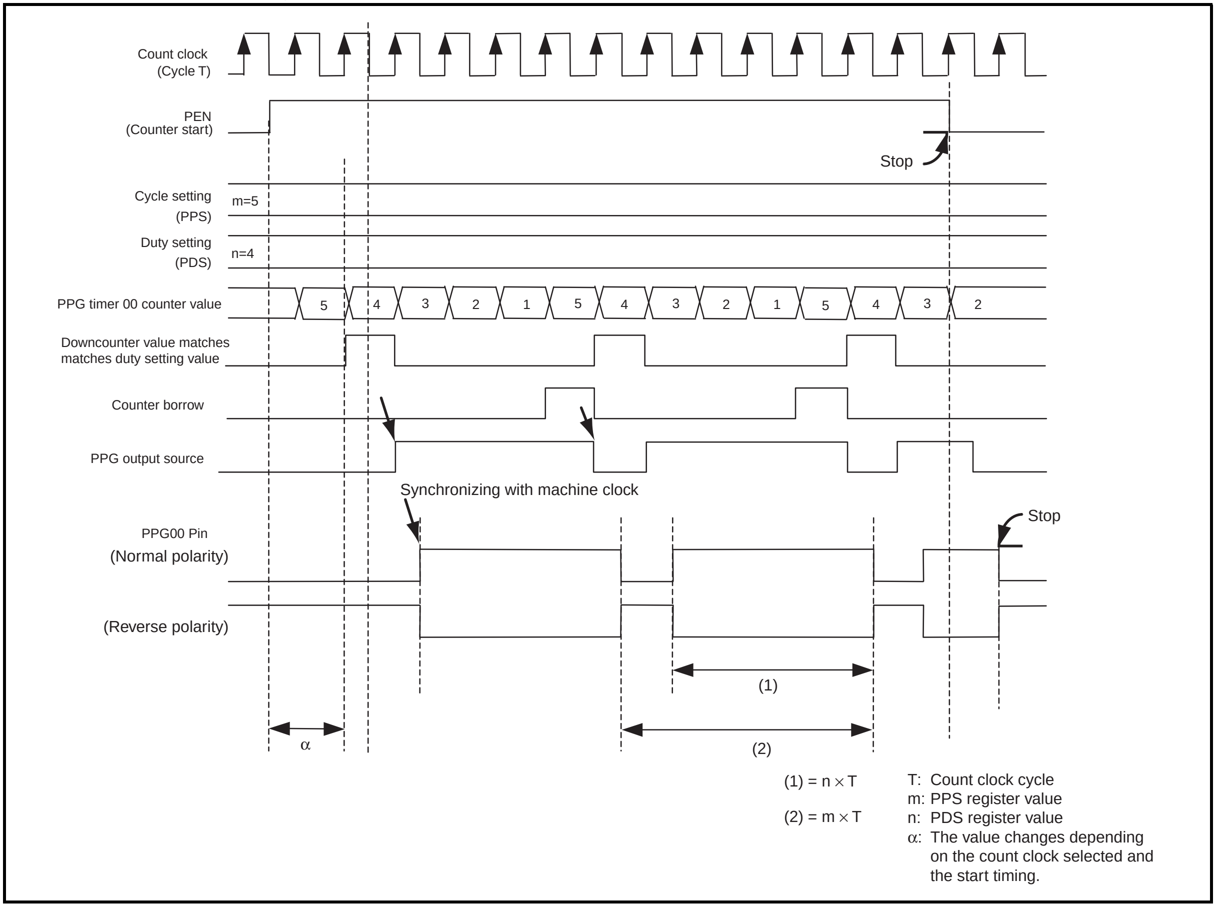
■ Operation of 8-bit PPG Independent Mode

- This mode is selected when the operation mode select bits (MD1, MD0) in the 8/16-bit PPG timer 00 control register (PC00) are set to "00_B".
- When the corresponding bit (PEN) in the 8/16-bit PPG start register (PPGS) is set to "1", the value in the 8/16-bit PPG cycle setup buffer register (PPS) is loaded to start down-count operation. When the count value reaches "1", the value in the cycle setup register is reloaded to repeat the counting.
- "H" is output to the PPG output synchronizing with the count clock. When the downcounter value matches the value in the 8/16-bit PPG timer 00/01 duty setup buffer register (PDS). After "H" which is the value of duty setting is output, "L" is output to the PPG output.

If, however, the PPG output inversion bit is set to "1", the PPG output is set and reset inversely from the above process.

Figure 21.7-2 shows the operation of the 8-bit PPG independent mode.

Figure 21.7-2 Operation of 8-bit PPG Independent Mode



Example of setting the duty to 50%

When PDS is set to "02_H" with PPS set to "04_H", the PPG output is set at a duty ratio of 50% (PPS setting value /2 set to PDS).

21.7.2 8-bit Prescaler + 8-bit PPG Mode

In this mode, the rising and falling edge detection pulses from the PPG timer 01 output can be used as the count clock of the PPG timer 00 downcounter to allow variable-cycle 8-bit PPG output from PPG timer 00.

■ Setting 8-bit Prescaler + 8-bit PPG Mode

The unit requires the register settings shown in Figure 21.7-3 to operate in 8-bit prescaler + 8-bit PPG mode.

Figure 21.7-3 Setting 8-bit Prescaler + 8-bit PPG Mode

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PC01	-	-	PIE1	PUF1	POEN1	CKS12	CKS11	CKS10
			◎	◎	◎	◎	◎	◎
PC00	MD1	MD0	PIE0	PUF0	POEN0	CKS02	CKS01	CKS00
	0	1	◎	◎	◎	×	×	×
PPS01	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
	Set PPG output cycle for PPG timer 01							
PPS00	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
	Set PPG output cycle for PPG timer 00							
PDS01	DH7	DH6	DH5	DH4	DH3	DH2	DH1	DH0
	Set PPG output duty for PPG timer 01							
PDS00	DL7	DL6	DL5	DL4	DL3	DL2	DL1	DL0
	Set PPG output duty for PPG timer 00							
PPGS	-	-	-	-	PEN11	PEN10	PEN01	PEN00
	*	*	*	*	*	*	◎	◎
REVC	-	-	-	-	REV11	REV10	REV01	REV00
	*	*	*	*	*	*	◎	◎

◎ : Used bit
0 : Set "0"
1 : Set "1"
× : Setting nullified
* : The bit status varies depending of the number of channels implemented

■ Operation of 8-bit Prescaler + 8-bit PPG Mode

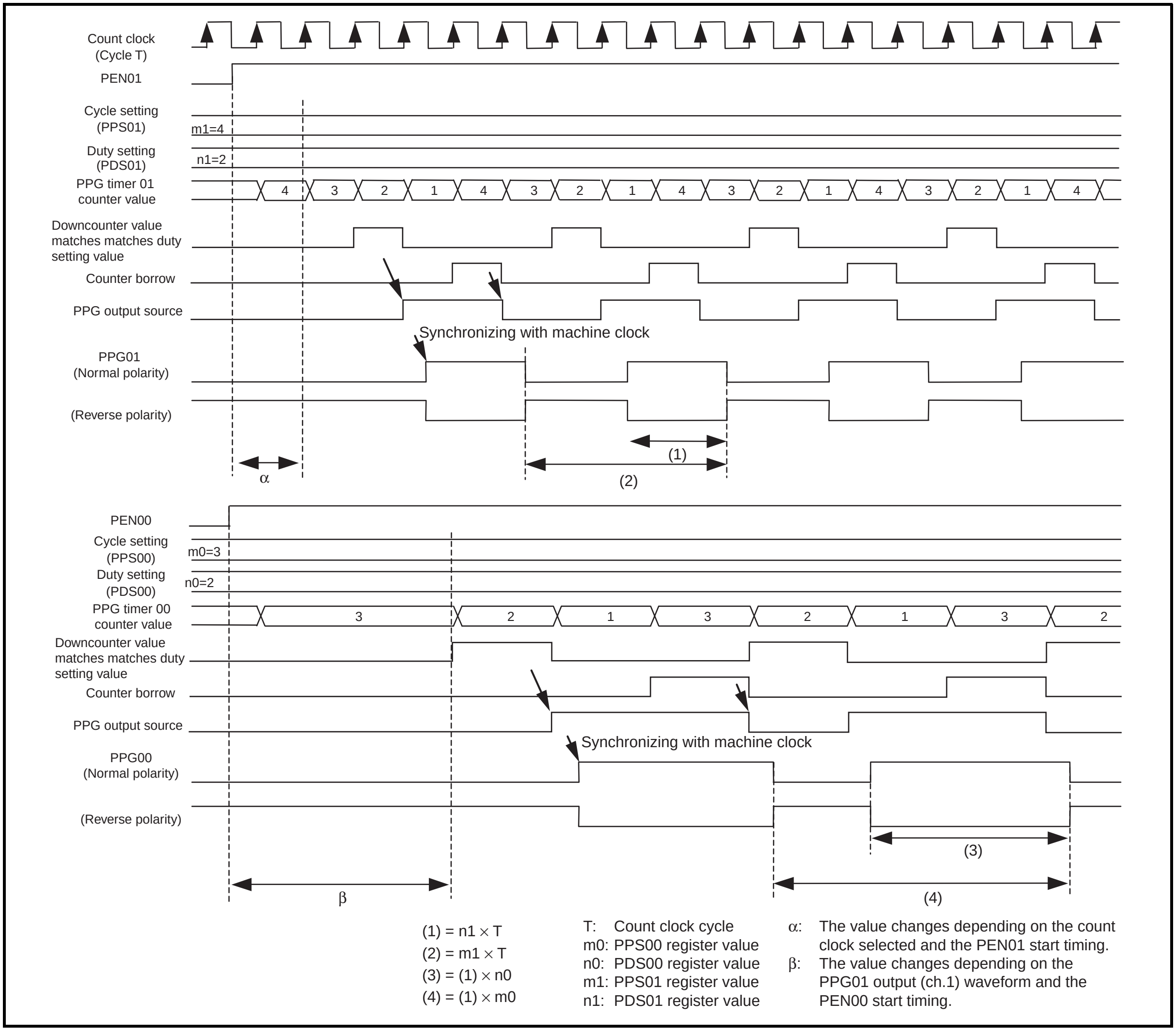
- This mode is selected by setting the operation mode select bits (MD1, MD0) of the 8/16-bit PPG timer 00 control register (PC00) to "01_B". This allows PPG timer 01 to be used as an 8-bit prescaler and PPG timer 00 to be used as an 8-bit PPG.
- When the PPG timer 01 (ch. 0) down counter operation enable bit (PEN01) is set to "1", the 8-bit prescaler (PPG timer 01) loads the value in the 8/16-bit PPG timer 01 cycle setup buffer register (PPS01) and starts down-count operation. When the value of the downcounter matches the value in the 8/16-bit PPG timer 01 duty setup buffer register (PDS01), the PPG01 output is set to "H" synchronizing with the count clock. After "H" which is the value of duty setting is output, the PPG01 output is set to "L". If the output inversion signal (REV01) is "0", the polarity will remain the same. If it is "1", the polarity

will be inverted and the signal will be output to the PPG pin.

- When the PPG operation enable bit (PEN00) is set to "1", the 8-bit PPG (PPG timer 00) loads the value in the 8/16-bit PPG timer 00 cycle setup buffer register (PPS00) and starts down-count operation (count clock = rising and falling edge detection pulses of PPG01 output after PPG timer 01 operation is enabled). When the count value reaches "1", the value in the 8/16-bit PPG timer 00 cycle setup buffer register is reloaded to repeat the counting. When the value of the downcounter matches the value in the 8/16-bit PPG timer 00 duty setup buffer register (PDS00), the PPG00 output is set to "H" synchronizing with the count clock. After "H" which is the value of duty setting is output, the PPG00 output is reset to "L". If the output inversion signal (REV00) is "0", the polarity will remain the same. If the output inversion signal (REV00) is "1", the polarity will be inverted and the signal will be output to the PPG00 pin.
- Set that the duty of the 8-bit prescaler (PPG timer 01) output to 50%.
- When PPG timer 00 is started with the 8-bit prescaler (PPG timer 01) being stopped, PPG timer 00 does not count.
- When the duty of the 8-bit prescaler (PPG timer 01) is set to 0% or 100%, PPG timer 00 does not perform counting as the 8-bit prescaler (PPG timer 01) output does not toggle.

Figure 21.7-4 shows the operation of 8-bit prescaler + 8-bit PPG mode.

Figure 21.7-4 Operation of 8-bit Prescaler + 8-bit PPG Mode



21.7.3 16-bit PPG Mode

In this mode, the unit can operate as a 16-bit PPG when PPG timer 01 and PPG timer 00 are assigned to the upper and lower bits respectively.

■ Setting 16-bit PPG Mode

The unit requires the register settings shown in Figure 21.7-5 to operate in 16-bit PPG mode.

Figure 21.7-5 Setting 16-bit PPG Mode

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PC01	-	-	PIE1	PUF1	POEN1	CKS12	CKS11	CKS10
			◎	◎	◎	◎	◎	◎
PC00	MD1	MD0	PIE0	PUF0	POEN0	CKS02	CKS01	CKS00
	0	0/1	◎	◎	◎	◎	◎	◎
PPS01	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
	Set PPG output cycle (Upper 8 bits) for PPG timer 01							
PPS00	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
	Set PPG output cycle (Lower 8 bits) for PPG timer 00							
PDS01	DH7	DH6	DH5	DH4	DH3	DH2	DH1	DH0
	Set PPG output duty (Upper 8 bits) for PPG timer 01							
PDS00	DL7	DL6	DL5	DL4	DL3	DL2	DL1	DL0
	Set PPG output duty (Lower 8 bits) for PPG timer 00							
PPGS	-	-	-	-	PEN11	PEN10	PEN01	PEN00
	*	*	*	*	*	*	×	◎
REVC	-	-	-	-	REV11	REV10	REV01	REV00
	*	*	*	*	*	*	×	◎

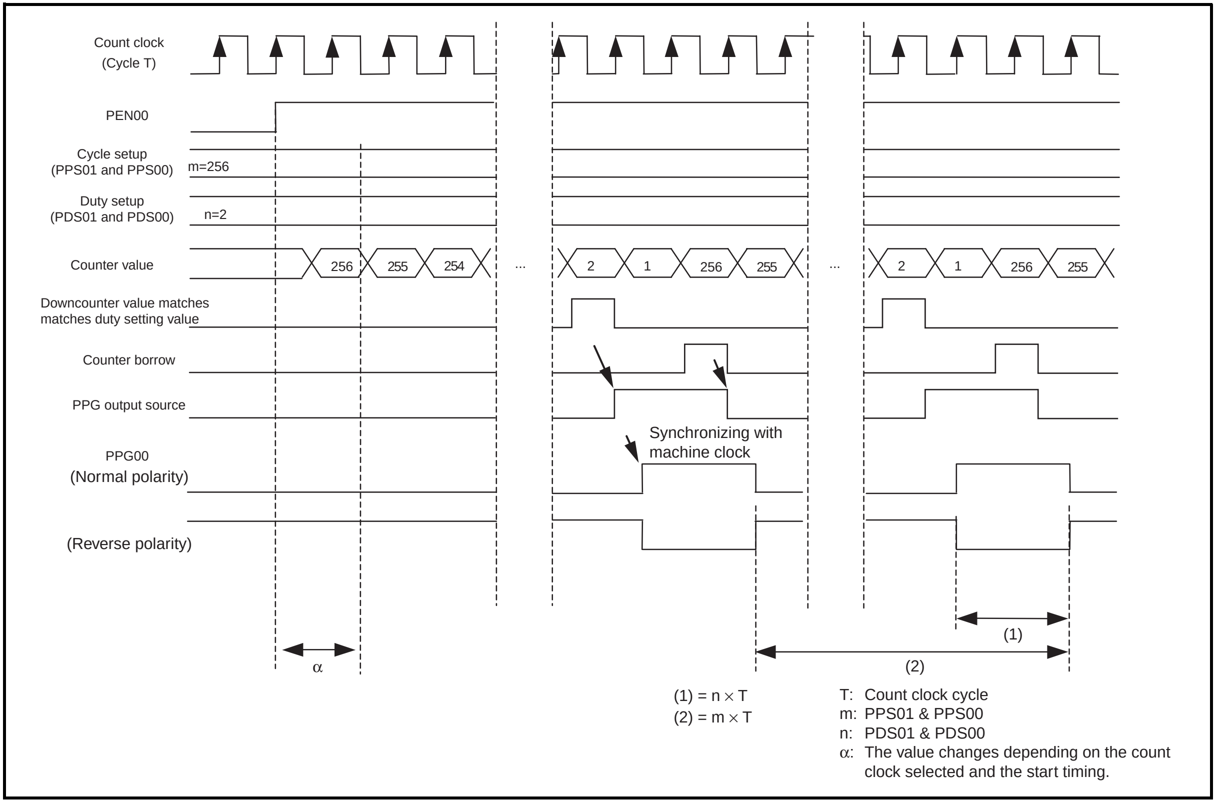
◎ : Used bit
0 : Set "0"
1 : Set "1"
× : Setting nullified
* : The bit status changes depending on the number of channels implemented.

■ Operation of 16-bit PPG Mode

- This mode is selected by setting the operation mode select bits (MD1, MD0) of the PPG timer 00 control register (PC00) to "10_B" or "11_B".
- When the PPG operation enable bit (PEN00) is set to "1" in 16-bit PPG mode, the 8-bit downcounters (PPG timer 00) and 8-bit downcounter (PPG timer 01) load the values in the 8/16-bit PPG timer 00/01 cycle setup buffer registers (PPS01 for PPG timer 01 and PPS00 for PPG timer 00) and start down-count operation. When the count value reaches "1", the values in the cycle setup register are reloaded and the counters repeat the counting.
- When the values of the downcounters match the values in the 8/16-bit PPG timer duty setup buffer registers (both the value in PDS01 for PPG timer 01 and the value in PDS00 for PPG timer 00), the PPG00 pin is set to "H" synchronizing with the count clock. After "H" which is the value of duty setting is output, the PPG00 pin is set to "L". If the output inversion signal (REV00) is "0", the signal will be output to the PPG00 with the polarity unchanged. If it is set to "1", the polarity will be inverted and the signal will be output to the PPG00 pin. (This applies to ch. 0 only. Ch. 1 will be set to the initial value <"L" if REV01 is "0", or "H" if it is "1">.)

Figure 21.7-6 shows the operation of 16-bit PPG mode.

Figure 21.7-6 Operation of 16-bit PPG Mode



21.8 Notes on Using 8/16-bit PPG

This section provides notes on using the 8/16-bit PPG.

■ Notes on Using 8/16-bit PPG

- Operational precaution

  Depending on the timing between the activation of PPG and count clock, an error may occur in the first cycle of the PPG output immediately after the activation. The error varies depending on the count clock selected. The output, however, is performed properly in the succeeding cycles.

- Precaution regarding interrupts

  A PPG interrupt is generated when the interrupt enable bit (PIE1/PIE0) is set to "1" and the interrupt request flag bit (PUF1/PUF0) in the 8/16-bit PPG timer 01/00 control register (PC01/PC00) is also set to "1". Always clear the interrupt request flag bit (PUF1/PUF0) to "0" in the interrupt routine.

21.9 Sample Settings for 8/16-bit PPG Timer

This section provides sample settings for the 8/16-bit PPG timer.

■ Sample Settings

● How to enable/stop PPG operation

The PPG operation enable bit (PPGS:PEN00 or PEN10) is used for PPG00.

Operation	PPG operation enable bit (PEN00 or PEN10)
To stop PPG operation	Set the bit to "0".
To enable PPG operation	Set the bit to "1".

PPG operation must be enabled before the PPG is activated.

The PPG operation enable bit (PPGS:PEN01 or PEN11) is used for PPG01.

Operation	PPG operation enable bit (PEN01 or PEN11)
To stop PPG operation	Set the bit to "0".
To enable PPG operation	Set the bit to "1".

PPG operation must be enabled before the PPG is activated.

● How to set the PPG operation mode

The operation mode select bits (PC00:MD[1:0]) are used.

● How to select the operating clock

Ch. 1 is selected by the operating clock select bits (PC01:CKS12/CKS11/CKS10).

Ch. 0 is selected by the operating clock select bits (PC00:CKS02/CKS01/CKS00).

● How to enable/disable the PPG output pin

The output enable bit (PC00:POEN0 or PC01:POEN1) is used.

Operation	Output enable bit (POEN0 or POEN1)
To enable PPG output	Set the bit to "1".
To disable PPG output	Set the bit to "0".

● How to invert the PPG output

The output level inversion bit (REVC:REV00 or REV10) is used for PPG00.

Operation	Output level inversion bit (REV00 or REV10)
To invert PPG output	Set the bit to "1".

The output level inversion bit (REVC:REV01 or REV11) is used for PPG01.

Operation	Output level inversion bit (REV01 or REV11)
To invert PPG output	Set the bit to "1".

● Interrupt-related register

The interrupt level is set by the interrupt level setting register shown in the following table.

Interrupt source	Interrupt level setting register	Interrupt vector
ch. 1 (lower)	Interrupt level setting register (ILR2) Address: $0007B_H$	#09 Address: $0FFE8_H$
ch. 1 (upper)	Interrupt level setting register (ILR2) Address: $0007B_H$	#10 Address: $0FFE6_H$
ch. 0 (lower)	Interrupt level setting register (ILR3) Address: $0007C_H$	#13 Address: $0FFE0_H$
ch. 0 (upper)	Interrupt level setting register (ILR3) Address: $0007C_H$	#12 Address: $0FFE2_H$

● How to enable/disable/clear interrupts

Interrupt request enable flag, Interrupt request flag

The interrupt request enable bit (PC00:PIE0 or PC01:PIE1) is used to enable or disable interrupts.

Operation	Interrupt request enable bit (PIE0 or PIE1)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

The interrupt request flag (PC00:PUF0 or PC01:PUF1) is used to clear interrupt requests.

Operation	Interrupt request flag (PUF0 or PUF1)
To clear an interrupt request	Set the bit to "0".

CHAPTER 22
UART/SIO

This chapter describes the functions and operations of UART/SIO.

22.1 Overview of UART/SIO

The UART/SIO is a general-purpose serial data communication interface. Serial data transfers of variable-length data can be made with a synchronous or asynchronous clock. The transfer format is NRZ. The transfer rate can be set with the dedicated baud rate generator or external clock (in clock synchronous mode).

■ Functions of UART/SIO

The UART/SIO is capable of serial data transmission/reception (serial input/output) to and from another CPU or peripheral device.

- Equipped with a full-duplex double buffer that allows 2-way full-duplex communication.
- The synchronous or asynchronous transfer mode can be selected.
- The optimum baud rate can be selected with the dedicated baud rate generator.
- The data length is variable; it can be set to 5 to 8 bits when no parity is used or to 6 to 9 bits when parity is used. (See Table 22.1-1.)
- The serial data direction (endian) can be selected.
- The data transfer format is NRZ (Non-Return-to-Zero).
- Two operation modes (operation modes 0 and 1) are available.
  Operation mode 0 operates as asynchronous clock mode (UART).
  Operation mode 1 operates as clock synchronous mode (SIO).

Table 22.1-1 UART/SIO Operation Modes

Operation mode	Data length		Synchronization mode	Length of stop bit
	No parity	With parity		
0	5	6	Asynchronous	1 bit or 2 bits
	6	7		
	7	8		
	8	9		
1	5	-	Synchronous	1 bit or 2 bits
	6	-		
	7	-		
	8	-		

MB95410H/470H Series

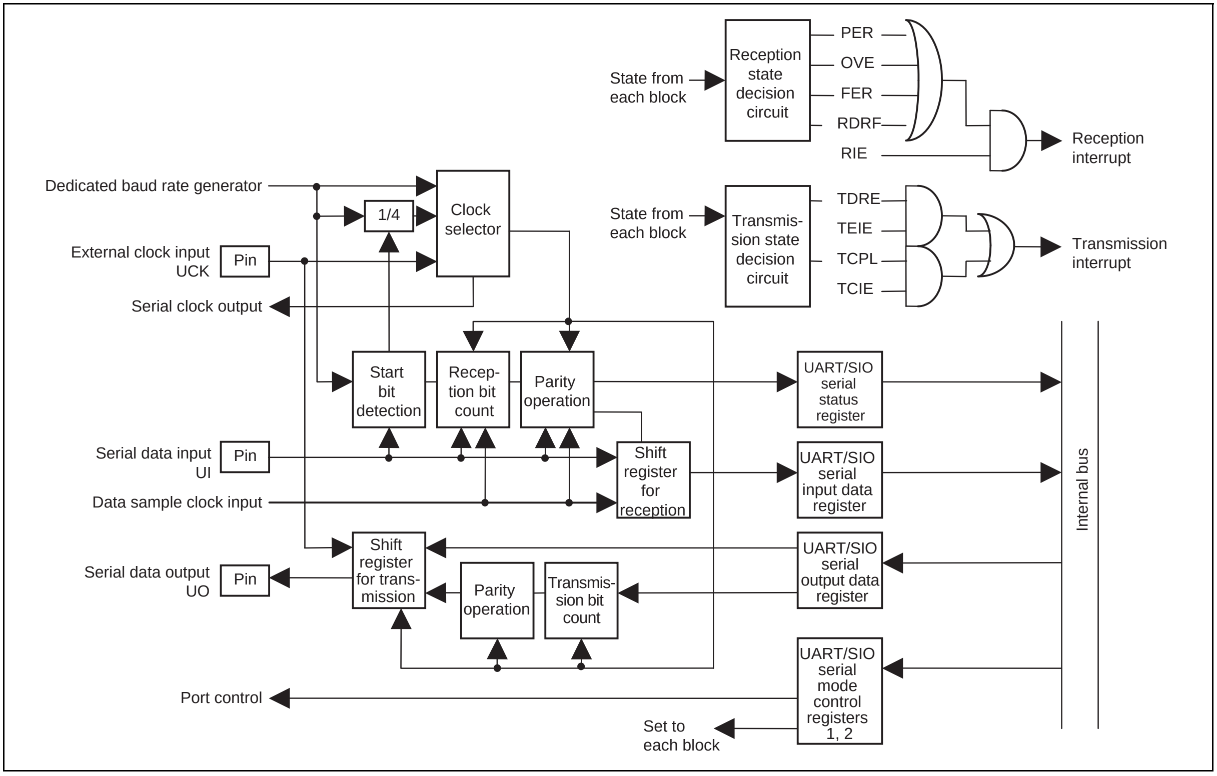
22.2 Configuration of UART/SIO

The UART/SIO consists of the following blocks:

- **UART/SIO serial mode control register 1 (SMC10/SMC11/SMC12)**
- **UART/SIO serial mode control register 2 (SMC20/SMC21/SMC22)**
- **UART/SIO serial status register (SSR0/SSR1/SSR2)**
- **UART/SIO serial input data register (RDR0/RDR1/RDR2)**
- **UART/SIO serial output data register (TDR0/TDR1/TDR2)**

■ Block Diagram of UART/SIO

Figure 22.2-1 Block Diagram of UART/SIO



- UART/SIO serial mode control register 1 (SMC10/SMC11/SMC12)

This register controls UART/SIO operation mode. It is used to set the serial data direction (endian), parity and its polarity, stop bit length, operation mode (synchronous/asynchronous), data length, and serial clock.

- UART/SIO serial mode control register 2 (SMC20/SMC21/SMC22)

This register controls UART/SIO operation mode. It is used to enable/disable serial clock output, serial data output, transmission/reception, and interrupts and to clear the reception error flag.

- UART/SIO serial status register (SSR0/SSR1/SSR2)

This register indicates the transmission/reception status and error status of UART/SIO.

- UART/SIO serial input data register (RDR0/RDR1/RDR2)

This register holds the receive data. The serial input is converted and then stored in this register.

- UART/SIO serial output data register (TDR0/TDR1/TDR2)

This register sets the transmit data. Data written to this register is serial-converted and then outputted.

■ Input Clock

The UART/SIO uses the output clock (internal clock) from the dedicated baud rate generator or the input signal (external clock) from the UCK pin as its input clock (serial clock).

22.3 Channels of UART/SIO

This section describes the channels of UART/SIO.

■ Channels of UART/SIO

The MB95410H/470H Series has 3 channels of UART/SIO. The following table shows the correspondence of the channel, pin, and register.

Table 22.3-1 Pins of UART/SIO

Channel	Pin name	Pin function
0	UCK0	Clock input/output
	UO0	Data output
	UI0	Data input
1	UCK1	Clock input/output
	UO1	Data output
	UI1	Data input
2	UCK2	Clock input/output
	UO2	Data output
	UI2	Data input

Table 22.3-2 Registers of UART/SIO

Channel	Register abbreviation	Corresponding register (Name in this manual)
0	SMC10	UART/SIO serial mode control register 1
	SMC20	UART/SIO serial mode control register 2
	SSR0	UART/SIO serial status register
	TDR0	UART/SIO serial output data register
	RDR0	UART/SIO serial input data register
1	SMC11	UART/SIO serial mode control register 1
	SMC21	UART/SIO serial mode control register 2
	SSR1	UART/SIO serial status register
	TDR1	UART/SIO serial output data register
	RDR1	UART/SIO serial input data register
2	SMC12	UART/SIO serial mode control register 1
	SMC22	UART/SIO serial mode control register 2
	SSR2	UART/SIO serial status register
	TDR2	UART/SIO serial output data register
	RDR2	UART/SIO serial input data register

22.4 Pins of UART/SIO

This section describes the pins of the UART/SIO.

■ Pins of UART/SIO

The pins associated with UART/SIO are the clock input and output pin (UCK), serial data output pin (UO) and serial data input pin (UI).

The following sections describe only ch. 0 of UART/SIO.

The functions of UCK1, UO1 and UI1 of ch. 1 and those of UCK2, UO2 and UI2 of ch. 2 are identical to those of UCK0, UO0 and UI0 of ch. 0 respectively.

UCK0:

Clock input/output pin for UART/SIO.

When the clock output is enabled (SMC20:SCKE=1), it serves as a UART/SIO clock output pin regardless of the value of the corresponding port direction register. At this time, do not select the external clock (SMC10:CKS = 0).

When it is to be used as a UART/SIO clock input pin, disable the clock output (SMC20:SCKE = 0) and make sure that it is set as input port by the corresponding port direction register. At this time, be sure to select the external clock (SMC10:CKS = 0).

UO0:

Serial data output pin for UART/SIO. When the serial data output is enabled (SMC20:TXOE = 1), it serves as a UART/SIO serial data output pin regardless of the value of the corresponding port direction register.

UI0:

Serial data input pin for UART/SIO. When it is to be used as a UART/SIO serial data input pin, make sure that it is set as input port by the corresponding port direction register.

■ Block Diagrams of Pins of UART/SIO

Figure 22.4-1 Block Diagram of UO0 of UART/SIO

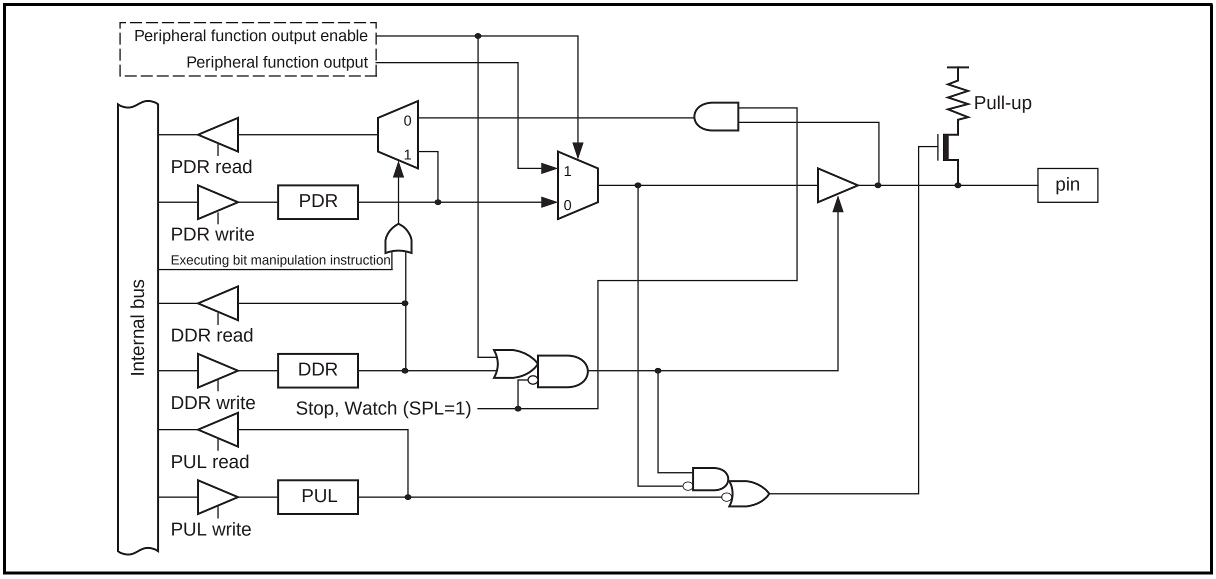


Figure 22.4-2 Block Diagram of UCK0 of UART/SIO

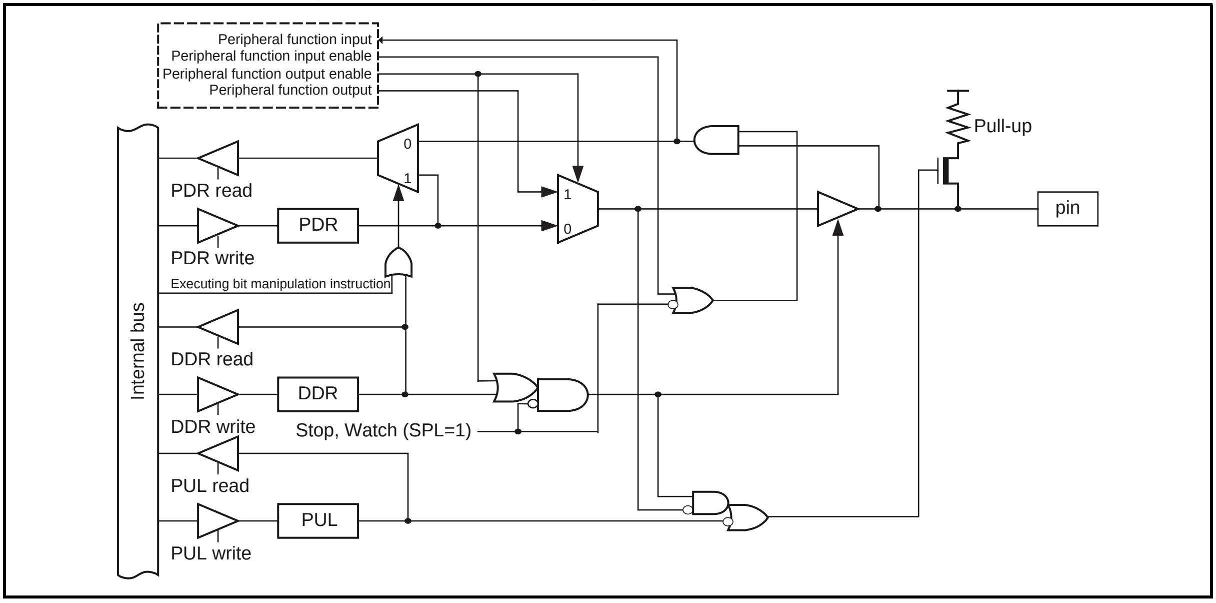
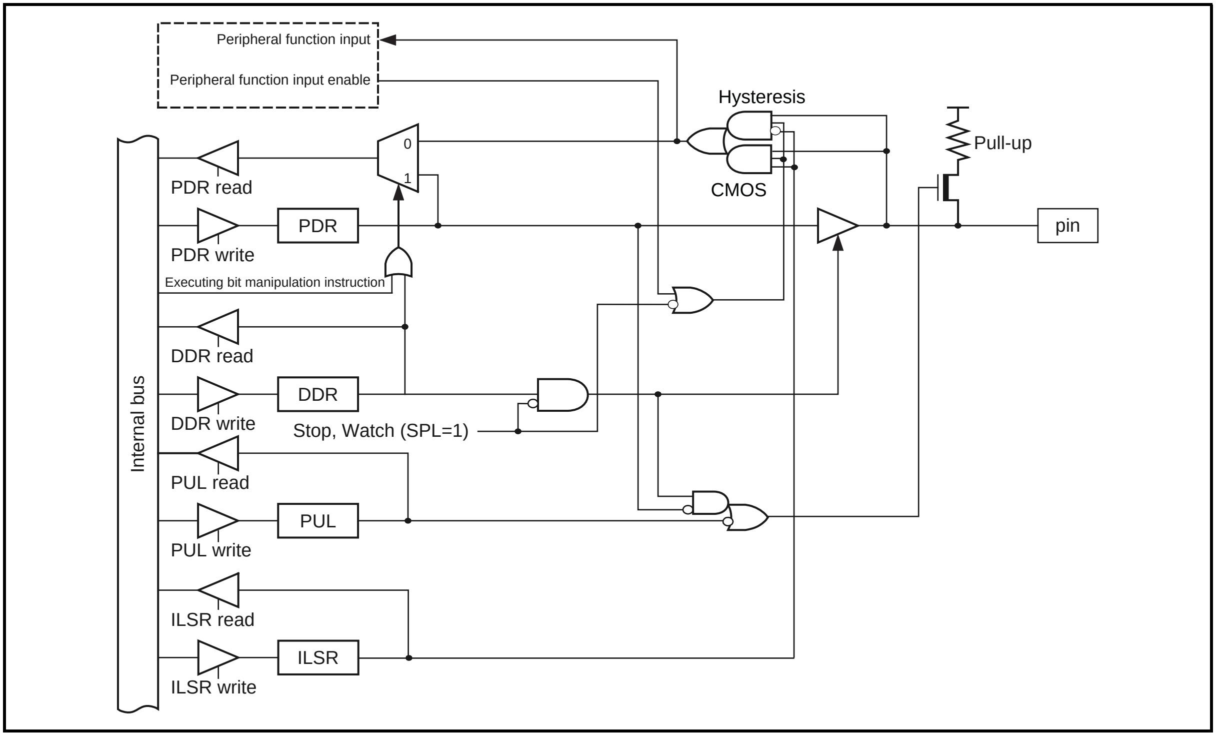

Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Pull-up
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
PUL read
PUL
PUL write

Figure 22.4-3 Block Diagram of UI0 of UART/SIO



22.5 Registers of UART/SIO

The registers of UART/SIO are UART/SIO serial mode control register 1 (SMC1), UART/SIO serial mode control register 2 (SMC2), UART/SIO serial status register (SSR), UART/SIO serial output data register (TDR), and UART/SIO serial input data register (RDR).

■ Registers of UART/SIO

Figure 22.5-1 Registers of UART/SIO

UART/SIO serial mode control register 1 (SMC1)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
SMC10	0056_H	BDS	PEN	TDP	SBL	CBL1	CBL0	CKS	MD	00000000_B
SMC11	$005B_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
SMC12	0066_H									

UART/SIO serial mode control register 2 (SMC2)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
SMC20	0057_H	SCKE	TXOE	RERC	RXE	TXE	RIE	TCIE	TEIE	00100000_B
SMC21	$005C_H$	R/W	R/W	R1/W	R/W	R/W	R/W	R/W	R/W	
SMC22	0067_H									

UART/SIO serial status register (SSR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
SSR0	0058_H	-	-	PER	OVE	FER	RDRF	TCPL	TDRE	00000001_B
SSR1	$005D_H$	R0/WX	R0/WX	R/WX	R/WX	R/WX	R/WX	R(RM1), W	R/WX	
SSR2	0068_H									

UART/SIO serial output data register (TDR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
TDR0	0059_H	TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0	$XXXXXXXX_B$
TDR1	$005E_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
TDR2	0069_H									

UART/SIO serial input data register (RDR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
RDR0	$005A_H$	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	00000000_B
RDR1	$005F_H$	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	
RDR2	$006A_H$									

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1), W : Readable/writable (The read value is different from write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
R1/W : Readable/writable (The read value is "1".)
- : Undefined bit

The following sections describe only UART/SIO ch. 0.

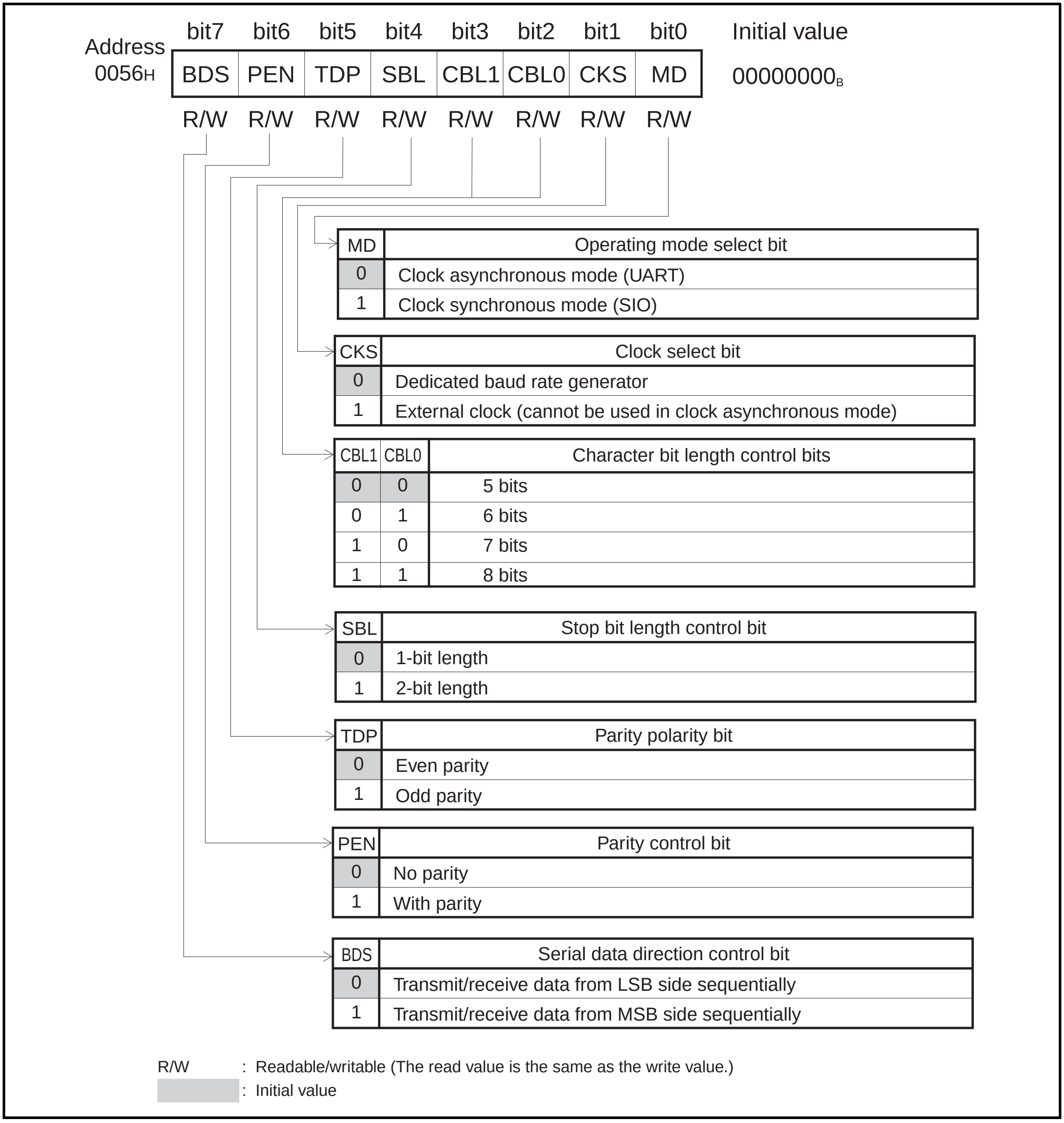
Ch. 1 and ch. 2 have the same configuration as ch. 0.

22.5.1 UART/SIO Serial Mode Control Register 1 (SMC10)

The UART/SIO serial mode control register 1(SMC10) controls the UART/SIO operation mode. The register is used to set the serial data direction (endian), parity and its polarity, stop bit length, operation mode (clock synchronous mode/clock asynchronous mode), data length, and serial clock.

■ UART/SIO Serial Mode Control Register 1 (SMC10)

Figure 22.5-2 UART/SIO Serial Mode Control Register 1 (SMC10)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0056H	BDS	PEN	TDP	SBL	CBL1	CBL0	CKS	MD	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

MD	Operating mode select bit
0	Clock asynchronous mode (UART)
1	Clock synchronous mode (SIO)

CKS	Clock select bit
0	Dedicated baud rate generator
1	External clock (cannot be used in clock asynchronous mode)

CBL1	CBL0	Character bit length control bits
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

SBL	Stop bit length control bit
0	1-bit length
1	2-bit length

TDP	Parity polarity bit
0	Even parity
1	Odd parity

PEN	Parity control bit
0	No parity
1	With parity

BDS	Serial data direction control bit
0	Transmit/receive data from LSB side sequentially
1	Transmit/receive data from MSB side sequentially

R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value

Table 22.5-1 Functions of Bits in UART/SIO Serial Mode Control Register 1 (SMC10)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7</td><td>BDS:
Serial data direction control bit</td><td>This bit sets the serial data direction (endian).
<b>Writing "0":</b> The bit specifies transmission or reception to be performed sequentially starting from the LSB side in the serial data register.
<b>Writing "1":</b> The bit specifies transmission or reception to be performed sequentially starting from the MSB side in the serial data register.</td></tr>
<tr><td>bit6</td><td>PEN:
Parity control bit</td><td>This bit enables or disables parity in clock asynchronous mode.
<b>Writing "0":</b> No parity
<b>Writing "1":</b> With parity</td></tr>
<tr><td>bit5</td><td>TDP:
Parity polarity bit</td><td>This bit controls even/odd parity.
<b>Writing "0":</b> Specifies even parity.
<b>Writing "1":</b> Specifies odd parity.</td></tr>
<tr><td>bit4</td><td>SBL:
Stop bit length control bit</td><td>This bit controls the length of the stop bit in clock asynchronous mode.
<b>Writing "0":</b> Sets the stop bit length to "1".
<b>Writing "1":</b> Sets the stop bit length to "2".
Note: The setting of this bit is only valid for transmission operation in clock asynchronous mode.
For receiving operation, reception data register full flag is set to "1" after detecting stop bit(1-bit) and completing the reception regardless of this bit.</td></tr>
<tr><td>bit3,
bit2</td><td>CBL1, CBL0:
Character bit length control bits</td><td>These bits select the character bit length as shown in the following table:
<table>
<tr><th>CBL1</th><th>CBL0</th><th>Character bit length</th></tr>
<tr><td>0</td><td>0</td><td>5</td></tr>
<tr><td>0</td><td>1</td><td>6</td></tr>
<tr><td>1</td><td>0</td><td>7</td></tr>
<tr><td>1</td><td>1</td><td>8</td></tr>
</table>
• The above setting is valid in both clock asynchronous mode and clock synchronous modes.</td></tr>
<tr><td>bit1</td><td>CKS:
Clock select bit</td><td>This bit selects the external clock or dedicated baud rate generator.
<b>Writing "0":</b> Selects the dedicated baud rate generator.
<b>Writing "1":</b> Selects the external clock.
Note: Setting this bit to "1" forcibly disables the output of the UCK0 pin.
The external clock cannot be used in clock asynchronous mode (UART).</td></tr>
<tr><td>bit0</td><td>MD:
Operating mode select bit</td><td>This bit selects clock asynchronous mode (UART) or clock synchronous mode (SIO).
<b>Writing "0":</b> Selects clock asynchronous mode (UART).
<b>Writing "1":</b> Selects clock synchronous mode (SIO).</td></tr>
</table>

Note:

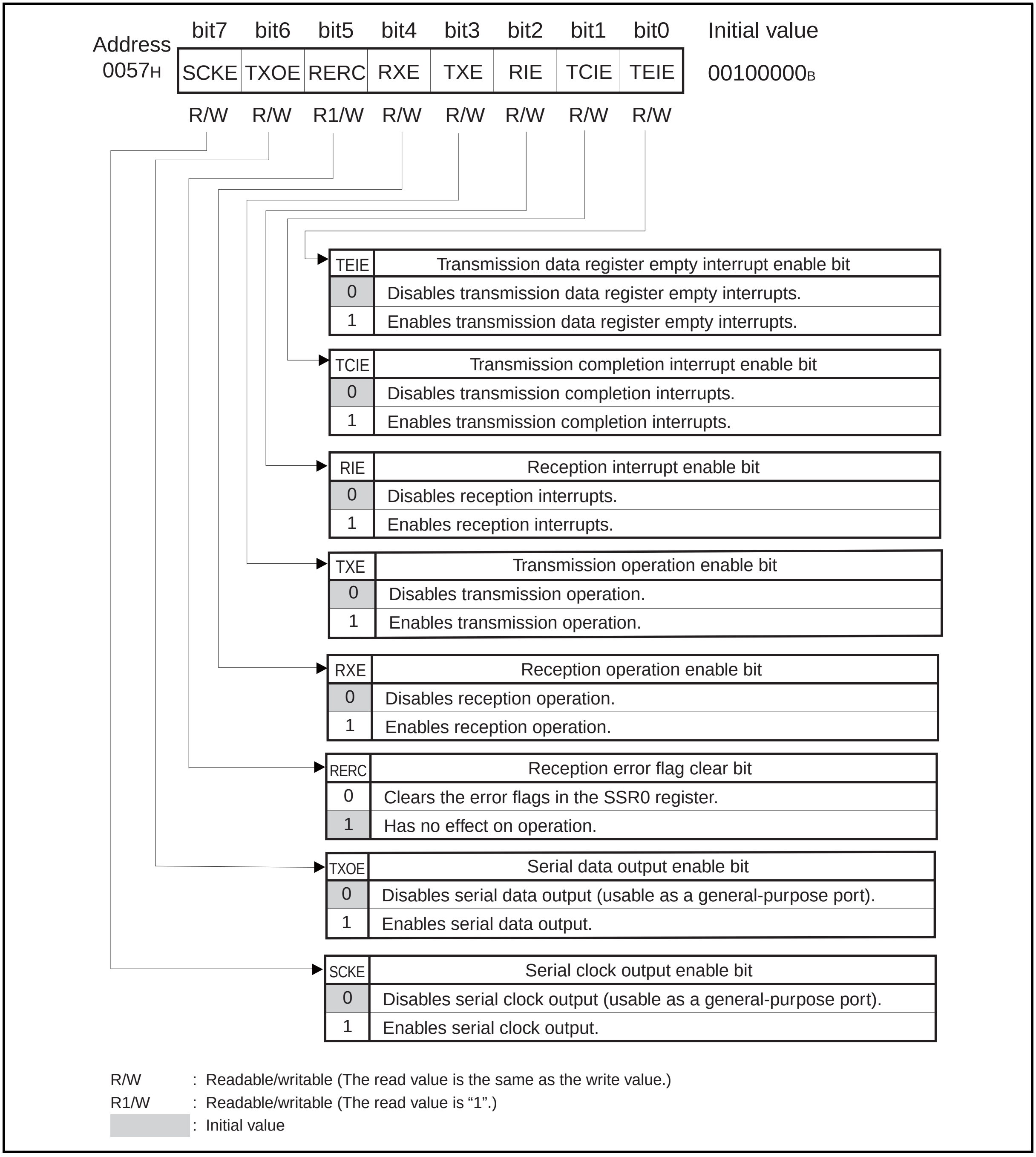
When modifying the UART/SIO serial mode control register 1 (SMC10), do not perform the modification during data transmission or reception.

22.5.2 UART/SIO Serial Mode Control Register 2 (SMC20)

The UART/SIO serial mode control register 2 (SMC20) controls the UART/SIO operation mode. The register is used to enable/disable serial clock output, serial data output, transmission/reception, and interrupts and to clear the reception error flag.

■ UART/SIO Serial Mode Control Register 2 (SMC20)

Figure 22.5-3 UART/SIO Serial Mode Control Register 2 (SMC20)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0057H	SCKE	TXOE	RERC	RXE	TXE	RIE	TCIE	TEIE	00100000B
	R/W	R/W	R1/W	R/W	R/W	R/W	R/W	R/W	

TEIE	Transmission data register empty interrupt enable bit
0	Disables transmission data register empty interrupts.
1	Enables transmission data register empty interrupts.

TCIE	Transmission completion interrupt enable bit
0	Disables transmission completion interrupts.
1	Enables transmission completion interrupts.

RIE	Reception interrupt enable bit
0	Disables reception interrupts.
1	Enables reception interrupts.

TXE	Transmission operation enable bit
0	Disables transmission operation.
1	Enables transmission operation.

RXE	Reception operation enable bit
0	Disables reception operation.
1	Enables reception operation.

RERC	Reception error flag clear bit
0	Clears the error flags in the SSR0 register.
1	Has no effect on operation.

TXOE	Serial data output enable bit
0	Disables serial data output (usable as a general-purpose port).
1	Enables serial data output.

SCKE	Serial clock output enable bit
0	Disables serial clock output (usable as a general-purpose port).
1	Enables serial clock output.

R/W : Readable/writable (The read value is the same as the write value.)
R1/W : Readable/writable (The read value is "1".)
▨ : Initial value

Table 22.5-2 Functions of Bits in UART/SIO Serial Mode Control Register 2 (SMC20)

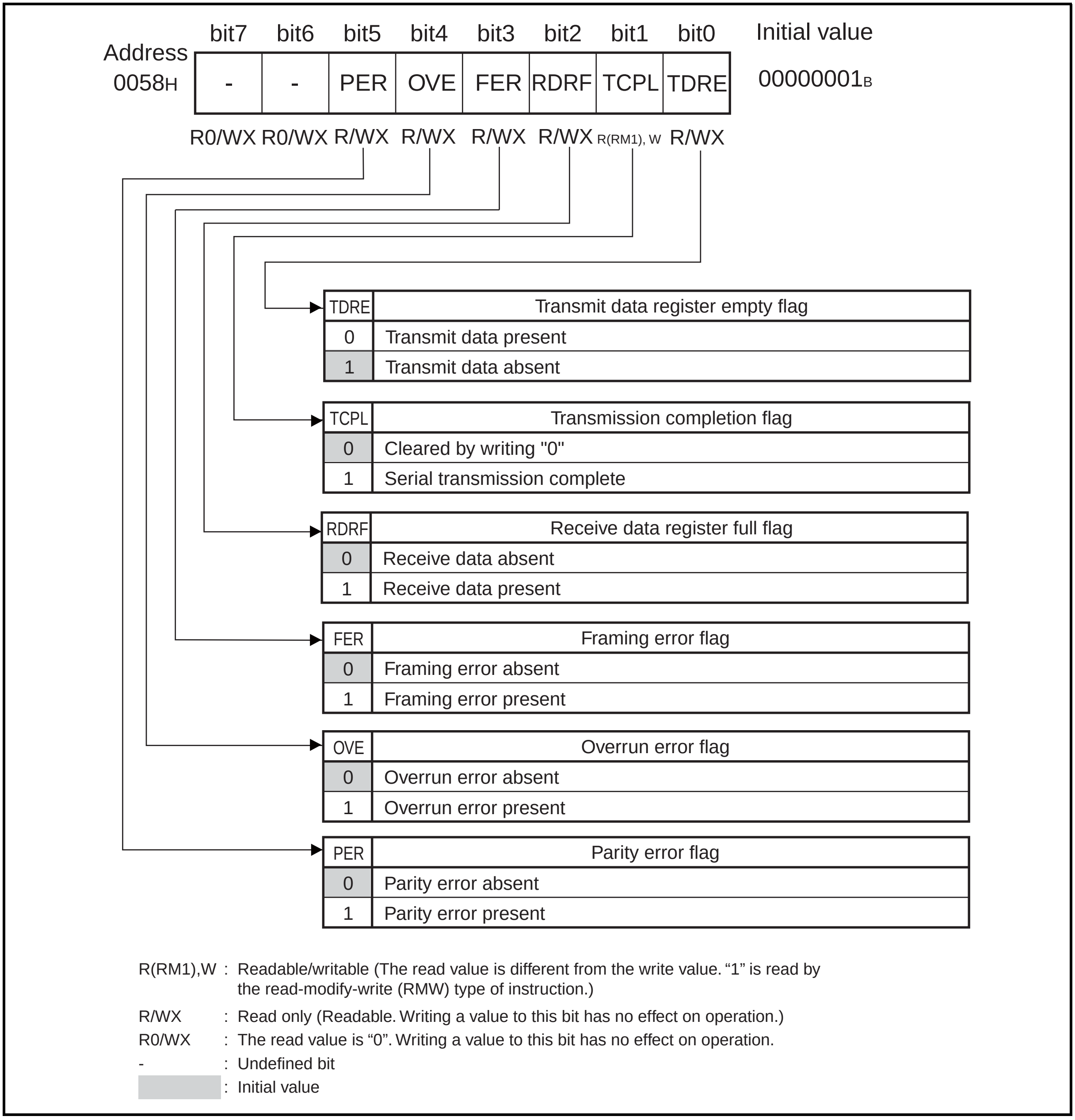
Bit name		Function
bit7	SCKE: Serial clock output enable bit	This bit controls the input/output of the serial clock pin (UCK0) in clock synchronous mode. **Writing "0":** Allows the pin to be used as a general-purpose port. **Writing "1":** Enables clock output. Note: When CKS is "1", the internal clock signal is not outputted even with this bit set to "1". If this bit is set to "1" with SMC10:MD set to "0"(asynchronous mode), the output from the port will always be "H".
bit6	TXOE: Serial data output enable bit	This bit controls the output of the serial data pin (UO0). **Writing "0":** Allows the pin to be used as a general-purpose port. **Writing "1":** Enables serial data output.
bit5	RERC: Reception error flag clear bit	**Writing "0":** Clears the error flags (PER, OVE, FER) in the SSR0 register. **Writing "1":** Has no effect on operation. • Reading this bit always returns "1".
bit4	RXE: Reception operation enable bit	**Writing "0":** Disables the reception of serial data. **Writing "1":** Enables the reception of serial data. • If this bit is set to "0" during reception, the reception operation will be immediately disabled and initialization will be performed. The data received up to that point will not be transferred to the UART/SIO serial input data register. Note: Setting this bit to "0" initializes reception operation. It has no effect on the error flags (PER, OVE, FER, RDRF).
bit3	TXE: Transmission operation enable bit	**Writing "0":** Disables the transmission of serial data. **Writing "1":** Enables the transmission of serial data. • If this bit is set to "0" during transmission, the transmission operation will be immediately disabled and initialization will be performed. The transmission completion flag (TCPL) will be set to "1" and the transmission data register empty (TDRE) bit will also be set to "1".
bit2	RIE: Reception interrupt enable bit	**Writing "0":** Disables reception interrupts. **Writing "1":** Enables reception interrupts. • A reception interrupt occurs immediately after either the receive data register full (RDRF) bit or an error flag (PER, OVE, FER, or RDRF) is set to "1" with this bit set to "1" (enabled).
bit1	TCIE: Transmission completion interrupt enable bit	**Writing "0":** Disables interrupts by the transmission completion flag. **Writing "1":** Enables interrupts by the transmission completion flag. • A transmission interrupt occurs immediately after the transmission completion flag (TCPL) bit is set to "1" with this bit set to "1" (enabled).
bit0	TEIE: Transmission data register empty interrupt enable bit	**Writing "0":** Disables interrupts by the transmission data register empty. **Writing "1":** Enables interrupts by the transmission data register empty. • A transmission interrupt occurs immediately after the transmission data register empty (TDRE) bit is set to "1" with this bit set to "1" (enabled).

22.5.3 UART/SIO Serial Status Register (SSR0)

The UART/SIO serial status register (SSR0) indicates the transmission/reception status and error status of the UART/SIO.

■ UART/SIO Serial Status Register (SSR0)

Figure 22.5-4 UART/SIO Serial Status Register (SSR0)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0058H	-	-	PER	OVE	FER	RDRF	TCPL	TDRE	00000001B
	R0/WX	R0/WX	R/WX	R/WX	R/WX	R/WX	R(RM1), W	R/WX	

TDRE	Transmit data register empty flag
0	Transmit data present
1	Transmit data absent

TCPL	Transmission completion flag
0	Cleared by writing "0"
1	Serial transmission complete

RDRF	Receive data register full flag
0	Receive data absent
1	Receive data present

FER	Framing error flag
0	Framing error absent
1	Framing error present

OVE	Overrun error flag
0	Overrun error absent
1	Overrun error present

PER	Parity error flag
0	Parity error absent
1	Parity error present

R(RM1),W	:	Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R/WX	:	Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX	:	The read value is “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

Table 22.5-3 Functions of Bits in UART/SIO Serial Status Register (SSR0)

Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5	PER: Parity error flag	This flag detects a parity error in receive data. • The bit is set when a parity error occurs during reception. Writing "0" to the RERC bit clears this flag. • If error detection and clearing by RERC occur at the same time, the error flag is set preferentially.
bit4	OVE: Overrun error flag	This flag detects an overrun error in receive data. • The flag is set when an overrun error occurs during reception. Writing "0" to the RERC bit clears this flag. • If error detection and clearing by RERC occur at the same time, the error flag is set preferentially.
bit3	FER: Framing error flag	This flag detects a framing error in receive data. • The bit is set when a framing error occurs during reception. Writing "0" to the RERC bit clears this flag. • If error detection and clearing by RERC occur at the same time, the error flag is set preferentially.
bit2	RDRF: Receive data register full flag	This flag indicates the status of the UART/SIO serial input data register. • The bit is set to "1" when receive data is loaded to the serial input data register. • The bit is cleared to "0" when data is read from the serial input data register.
bit1	TCPL: Transmission completion flag	This flag indicates the data transmission status. • The bit is set to "1" upon completion of serial transmission. Note, however, that the bit is not set to "1" even upon completion of transmission when the UART/SIO serial output data register contains data to be transmitted in succession. • Writing "0" to this bit clears its flag. • If events to set and clear the flag occur at the same time, it is set preferentially. • Writing "1" to this bit has no effect on operation.
bit0	TDRE: Transmit data register empty flag	This flag indicates the status of the UART/SIO serial output data register. • The bit is set to "0" when transmit data is written to the serial output register. • The bit is set to "1" when data is loaded to the transmission shift register and transmission starts.

22.5.4 UART/SIO Serial Input Data Register (RDR0)

The UART/SIO serial input data register (RDR0) is used to input (receive) serial data.

■ UART/SIO Serial Input Data Register (RDR0)

Figure 22.5-5 shows the bit configuration of the UART/SIO serial input data register.

Figure 22.5-5 UART/SIO Serial Input Data Register (RDR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
005A$_H$	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	00000000$_B$
	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

R/WX : Read only (Readable. Writing a value to it has no effect on operation.)

This register stores received data. The serial data signals sent to the serial data input pin (UI0) is converted by the shift register and stored in this register.

When received data is set correctly in this register, the receive data register full (RDRF) bit is set to "1". At this time, an interrupt occurs if reception interrupt requests have been enabled. If an RDRF bit check by the program or using an interruption shows that received data is stored in this register, the reading of the content for this register clears the RDRF flag to "0".

When the character bit length (CBL1, CBL0) is set to shorter than 8 bits, the excess upper bits (beyond the set bit length) are set to "0".

22.5.5 UART/SIO Serial Output Data Register (TDR0)

The UART/SIO serial output data register (TDR0) is used to output (transmit) serial data.

■ UART/SIO Serial Output Data Register (TDR0)

Figure 22.5-6 shows the bit configuration of the UART/SIO serial output data register.

Figure 22.5-6 UART/SIO Serial Output Data Register (TDR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0059_H	TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

This register holds data to be transmitted. The register accepts a write when the transmission data register empty (TDRE) bit contains "1". An attempt to write to the bit is ignored when the bit contains "0".

When this register is updated at writting complete the transmission data and TDRE=0 (without depending on TXE of the UART/SIO serial mode control register2 is "1" or "0"), the transmission operation is initialized by writing "0" to TXE, TDRE becomes "1", and updating this register is enabled.

Moreover, when "0" is written in TXE without the starting transmission (when the transmission data is written in TDR0, and it has not transmitted TXE to "1" yet), TCPL is not set in "1". The transmission data is transferred to the shift register for the transmission, it is converted into the serial data, and it is transmitted from the serial data output terminal.

When transmit data is written to the UART/SIO serial output data register (TDR0), the transmission data register empty bit (TDRE) is set to "0". Upon completion of transfer of transmit data to the transmission shift register, the transmission data register empty bit (TDRE) is set to "1", allowing the next piece of transmit data to be written. At this time, an interrupt occurs if transmission data register empty interrupts have been enabled. Write the next transmit data when transmit data empty occurs or the TDRE bit is set to "1".

When the character bit length (CBL1, CBL0) is set to shorter than 8 bits, the excess upper bits (beyond the set bit length) are ignored.

Note:

The data in this register cannot be updated when TDRE in UART/SIO serial status data register is "0".

When this register is updated at writing complete the transmission data and TDRE=0 (without depending on TXE of the UART/SIO serial mode control register 2 is "1" or "0"), the transmission operation is initialized by writing "0" to TXE, TDRE becomes "1", and the update of this register becomes possible.

Moreover, when "0" is written in TXE without the starting transmission (when the transmission data is written in TDR0, and it has not transmitted TXE to "1" yet), TCPL is not set in "1". And, to change data, please write it after making TDRE "1" once by writing TXE =0.

22.6 Interrupts of UART/SIO

The UART/SIO has six interrupt-related bits: error flag bits (PER, OVE, FER), receive data register full bit (RDRF), transmission data register empty bit (TDRE), and transmission completion flag (TCPL).

■ Interrupts of UART/SIO

Table 22.6-1 lists the UART/SIO interrupt control bits and interrupt sources.

Table 22.6-1 UART/SIO Interrupt Control Bits and Interrupt Sources

Item	Description					
Interrupt request flag bit	SSR0: TDRE	SSR0: TCPL	SSR0: RDRF	SSR0: PER	SSR0: OVE	SSR0: FER
Interrupt request enable bit	SMC20: TEIE	SMC20: TCIE	SMC20: RIE	SMC20: RIE	SMC20: RIE	SMC20: RIE
Interrupt source	Transmission data register empty	Transmission completion	Receive data full	Parity error	Overrun error	Framing error

■ Transmission Interrupt

When transmit data is written to the UART/SIO serial output data register (TDR0), the data is transferred to the transmission shift register. When the next piece of data can be written, the TDRE bit is set to "1". At this time, an interrupt request to the interrupt controller occurs when transmit data register empty interrupt enable bit has been enabled (SMC20:TEIE = 1).

The TCPL bit is set to "1" upon completion of transmission of all pieces of transmit data. At this time, an interrupt request to the interrupt controller occurs when transmission completion interrupt enable bit has been enabled (SMC20:TCIE = 1).

■ Reception Interrupt

If the data is input successfully up to the stop bit, the RDRF bit is set to "1". If an overrun, parity, or framing error occurs, the corresponding error flag bit (PER, OVE, or FER) is set to "1".

These bits are set when a stop bit is detected. If reception interrupt enable bit has been enabled (SMC20:RIE = 1), an interrupt request to the interrupt controller will be generated.

■ Registers and Vector Table Addresses Related to UART/SIO Interrupts

Table 22.6-2 Registers and Vector Table Addresses Related to UART/SIO Interrupts

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
UART/SIO ch. 0	IRQ04	ILR1	L04	$FFF2_H$	$FFF3_H$
UART/SIO ch. 1*	IRQ09	ILR2	L09	$FFE8_H$	$FFE9_H$
UART/SIO ch. 2	IRQ07	ILR1	L07	$FFEC_H$	$FFED_H$

ch.: Channel

*: UART/SIO ch. 1 uses the same interrupt request number and vector table addresses as 8/16-bit PPG ch. 1 (lower).

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

22.7 Operations of UART/SIO and Setting Procedure Example

The UART/SIO has a serial communication function (operation mode 0, 1).

■ Operation of UART/SIO

● Operation mode

Two operation modes are available in the UART/SIO. Clock synchronous mode (SIO) or clock asynchronous mode (UART) can be selected (see Table 22.7-1).

Table 22.7-1 Operation Modes of UART/SIO

Operation mode	Data length		Synchronization mode	Length of stop bit
	No parity	With parity		
0	5	6	Asynchronous	1 bit or 2 bits
	6	7		
	7	8		
	8	9		
1	5	6	Synchronous	1 bit or 2 bits
	6	7		
	7	8		
	8	9		

■ Setting Procedure Example

Below is an example of procedure for setting the UART/SIO.

● Initial settings

1) Set the port input. (DDR1, DDR9, DDRG)
2) Set the interrupt level. (ILR1, ILR2)
3) Set the prescaler. (PSSR0)
4) Set the baud rate. (BRSR0)
5) Select the clock. (SMC10:CKS)
6) Set the operation mode. (SMC10:MD)
7) Enable/disable the serial clock output. (SMC20:SCKE)
8) Enable reception. (SMC20:RXE = 1)
9) Enable interrupts. (SMC20:RIE = 1)

● Interrupt processing

Read receive data. (RDR0)

22.7.1 Operations in Operation Mode 0

Operation mode 0 operates as clock asynchronous mode (UART).

■ Operating Description of UART/SIO Operation Mode 0

Clock asynchronous mode (UART) is selected when the MD bit in the UART/SIO serial mode control register 1 (SMC10) is set to "0".

● Baud rate

The serial clock is selected by the CKS bit in the SMC10 register. Be sure to select the dedicated baud rate generator at this time.

The baud rate is equivalent to the output clock frequency of the dedicated baud rate generator, divided by four. The UART can perform communication within the range from -2% to +2% of the selected baud rate.

The baud rate generated by the dedicated baud rate generator is obtained from the equation illustrated below. (For information about the dedicated baud rate generator, see "CHAPTER 23 UART/SIO DEDICATED BAUD RATE GENERATOR".)

Figure 22.7-1 Baud Rate Calculation when Using Dedicated Baud Rate Generator

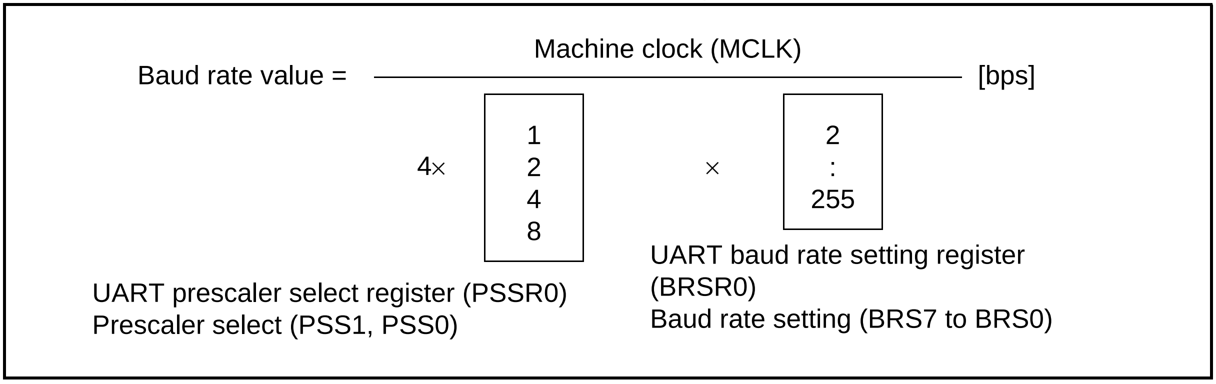

Baud rate value =
Machine clock (MCLK)
[bps]
4×
1
2
4
8
×
2
:
255
UART prescaler select register (PSSR0)
Prescaler select (PSS1, PSS0)
UART baud rate setting register
(BRSR0)
Baud rate setting (BRS7 to BRS0)

Table 22.7-2 Sample Asynchronous Transfer Rates Based on Dedicated Baud Rate Generator (Clock Gear = 4/F_{CH}, Machine Clock = 10 MHz)

Dedicated baud rate generator setting		UART internal division	Total division ratio (PSS × BRS × 4)	Baud rate (10 MHz / Total division ratio)
Prescaler select PSS[1:0]	Baud rate counter setting BRS[7:0]			
1 (Setting value: 0,0)	20	4	80	125000
1 (Setting value: 0,0)	22	4	88	113636
1 (Setting value: 0,0)	44	4	176	56818
1 (Setting value: 0,0)	87	4	348	28736
1 (Setting value: 0,0)	130	4	520	19231
2 (Setting value: 0,1)	130	4	1040	9615
4 (Setting value: 1,0)	130	4	2080	4808
8 (Setting value: 1,1)	130	4	4160	2404

The baud rate in clock asynchronous mode can be set in the following range.

Table 22.7-3 Baud Rate Setting Range in Clock Asynchronous Mode

PSS[1:0]	BRS[7:0]
"00_B" to "11_B"	02_H (2) to FF_H (255)

● Transfer data format

UART can treat data only in NRZ (Non-Return-to-Zero) format. Figure 22.7-2 shows the data format.

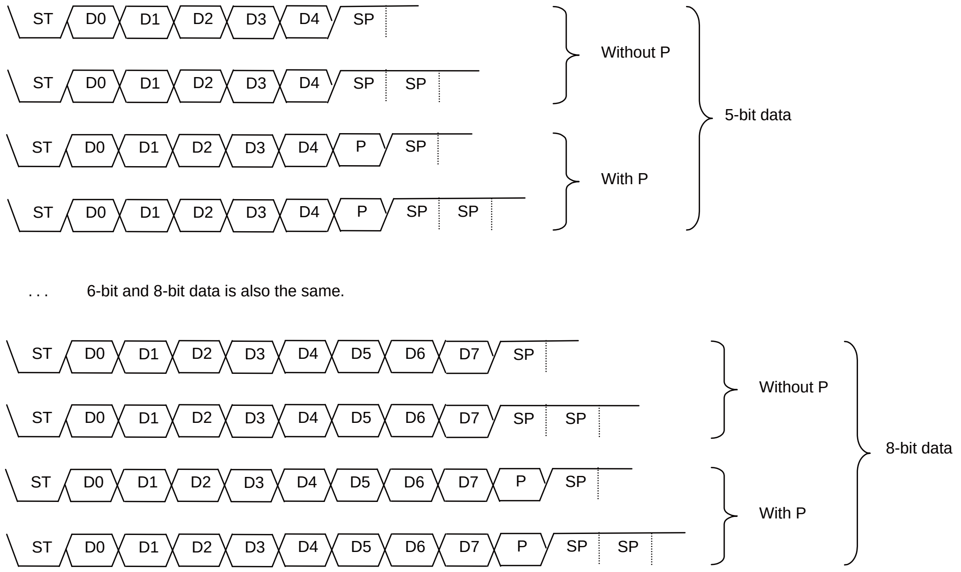
The character bit length can be selected from among 5 to 8 bits depending on the CBL1 and CBL0 settings.

The stop bit length can be set to 1 or 2 bits depending on the SBL setting.

PEN and TDP can be used to enable/disable parity and to select parity polarity.

As shown in Figure 22.7-2, the transfer data always starts from the start bit ("L" level) and ends with the stop bit ("H" level) by performing the specified data bit length transfer with MSB or LSB first ("LSB first" or "MSB first" can be selected by the BDS bit). It becomes "H" level at the idle state.

Figure 22.7-2 Transfer Data Format


ST D0 D1 D2 D3 D4 SP
ST D0 D1 D2 D3 D4 SP SP
Without P
ST D0 D1 D2 D3 D4 P SP
ST D0 D1 D2 D3 D4 P SP SP
With P
5-bit data
. . . 6-bit and 8-bit data is also the same.
ST D0 D1 D2 D3 D4 D5 D6 D7 SP
ST D0 D1 D2 D3 D4 D5 D6 D7 SP SP
Without P
ST D0 D1 D2 D3 D4 D5 D6 D7 P SP
ST D0 D1 D2 D3 D4 D5 D6 D7 P SP SP
With P
8-bit data

ST : Start bit
SP : Stop bit
P : Parity bit
D0 to D7: Data. The sequence can be selected from "LSB first" or "MSB first" by the direction control register (BDS bit)

● Receiving operation in asynchronous clock mode (UART)

Use UART/SIO serial mode control register 1 (SMC10) to select the serial data direction (endian), parity/non-parity, parity polarity, stop bit length, character bit length, and clock.

Reception remains performed as long as the reception operation enable bit (RXE) contains "1".

Upon detection of a start bit in receive data with the reception operation enable bit (RXE) set to "1", one frame of data is received according to the data format set in UART/SIO serial control register 1 (SMC10).

When the reception of one frame of data has been completed, the received data is transferred to the UART/SIO serial input data register (RDR0) and the next frame of serial data can be received.

When the UART/SIO serial input data register (RDR0) stores data, the receive data register full (RDRF) bit is set to "1".

A reception interrupt occurs the moment the receive data register full (RDRF) bit is set to "1" when the reception interrupt enable bit (RIE) contains "1".

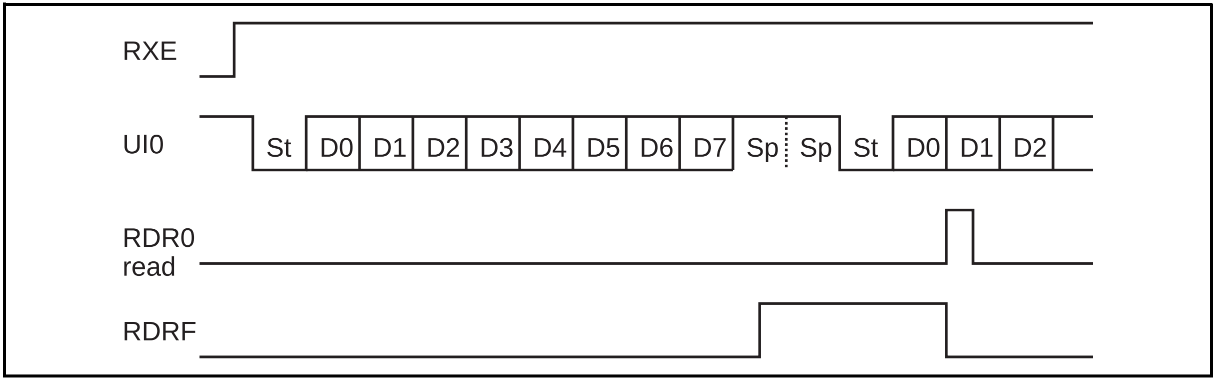
Received data is read from the UART/SIO serial input data register (RDR0) after each error flag (PER, OVE, FER) in the UART/SIO serial status register is checked.

When received data is read from the UART/SIO serial input data register (RDR0), the receive data register full (RDRF) bit is cleared to "0".

Note that modifying UART/SIO serial mode control register 1 (SMC10) during reception may result in unpredictable operation.

If the RXE bit is set to "0" during reception, the reception is immediately disabled and initialization will be performed. The data received up to that point will not be transferred to the serial input data register.

Figure 22.7-3 Receiving Operation in Asynchronous Clock Mode



● Reception error in asynchronous clock mode (UART)

If any of the following three error flags (PER, FER, OVE) has been set, receive data is not transferred to the UART/SIO serial input data register (RDR0) and the receive data register full (RDRF) bit is not set to "1" either.

1. Parity error (PER)

The parity error (PER) bit is set to "1" if the parity bit in received serial data does not match the parity polarity bit (TDP) when the parity control bit (PEN) contains "1".

2. Framing error (FER)

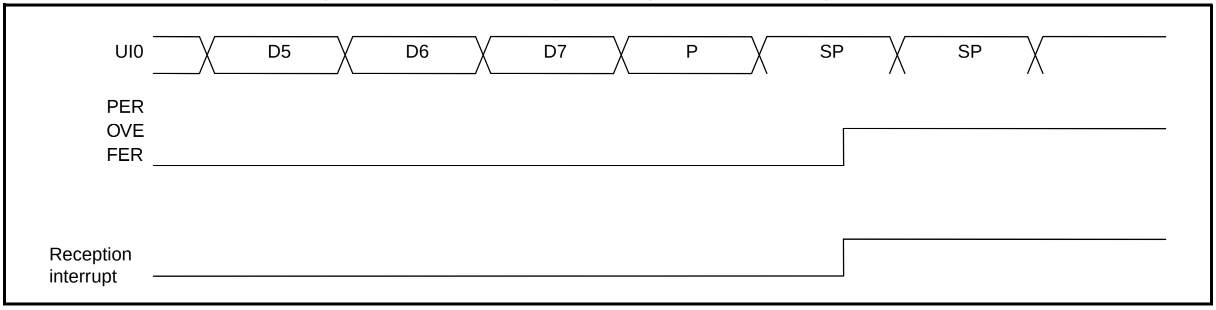
The framing error (FER) bit is set to "1" if "1" is not detected at the position of the first stop bit in serial data received in the set character bit length (CBL) under parity control (PEN). Note that the stop bit is not checked if it appears at the second bit or later.

3. Overrun error (OVE)

Upon completion of reception of serial data, the overrun error (OVE) bit is set to "1" if the reception of the next data is performed before the previous receive data is read.

Each flag is set at the position of the first stop bit.

Figure 22.7-4 Setting Timing for Receiving Errors



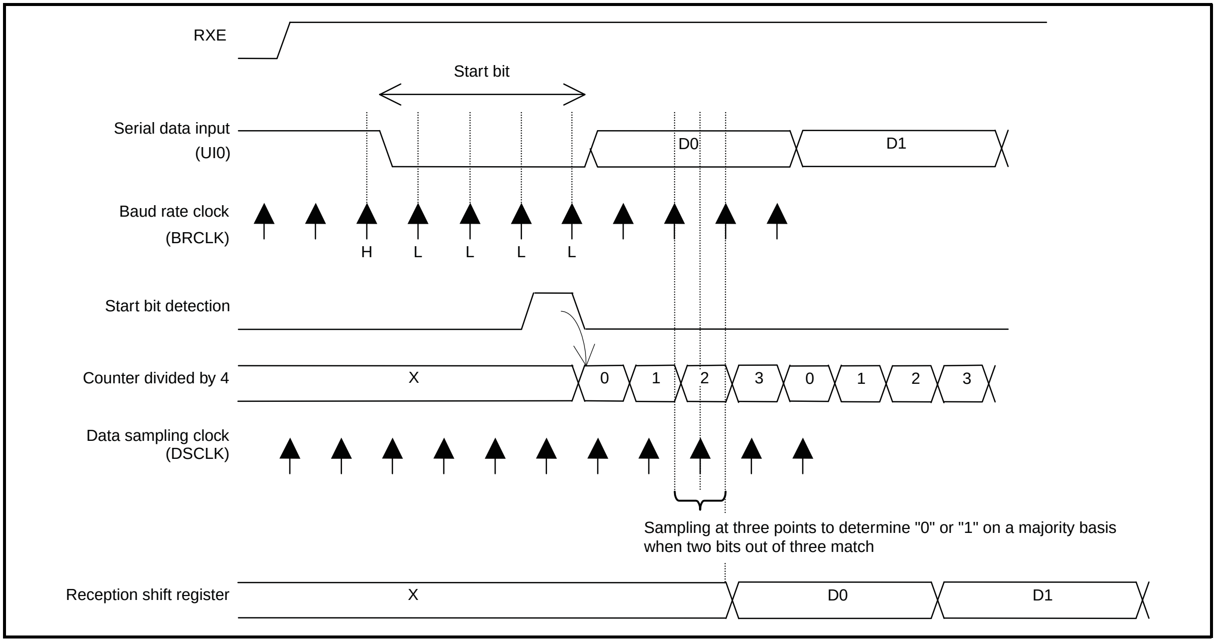
● Start bit detection and confirmation of receive data during reception

The start bit is detected by a falling of the serial input followed by a succession of three "L" levels after the serial data input is sampled according to the clock (BRCLK) signal provided by the dedicated baud rate generator with the reception operation enable bit (RXE) set to "1". When the first "H, L, L, L" train is detected in a BRCLK sample, therefore, the current bit is regarded as the start bit.

The frequency-quartered circuit is activated upon detection of the start bit and serial data is input to the reception shift register at intervals of four periods of BRCLK.

When data is received, sampling is performed at three points of the baud rate clock (BRCLK) and data sampling clock (DSCLK) and received data is confirmed on a majority basis when two bits out of three match.

Figure 22.7-5 Start Bit Detection and Serial Data Input



● Transmission in asynchronous clock mode

Use UART/SIO serial mode control register 1 (SMC10) to select the serial data direction (endian), parity/non-parity, parity polarity, stop bit length, character bit length, and clock.

Either of the following two procedures can be used to initiate the transmission process:

- Set the transmission operation enable bit (TXE) to "1", and then write transmit data to the serial output data register to start transmission.
- Write transmit data to the UART/SIO serial output data register, and then set the transmission operation enable bit (TXE) to "1" to start transmission.

Transmit data is written to the UART/SIO serial output data register (TDR0) after it is checked that the transmit data register empty (TDRE) bit set to "1".

When the transmit data is written to the UART/SIO serial output data register (TDR0), the transmit data register empty (TDRE) bit is cleared to "0".

The transmit data is transferred from the UART/SIO serial output data register (TDR0) to the transmission shift register, and the transmit data register empty (TDRE) is set to "1".

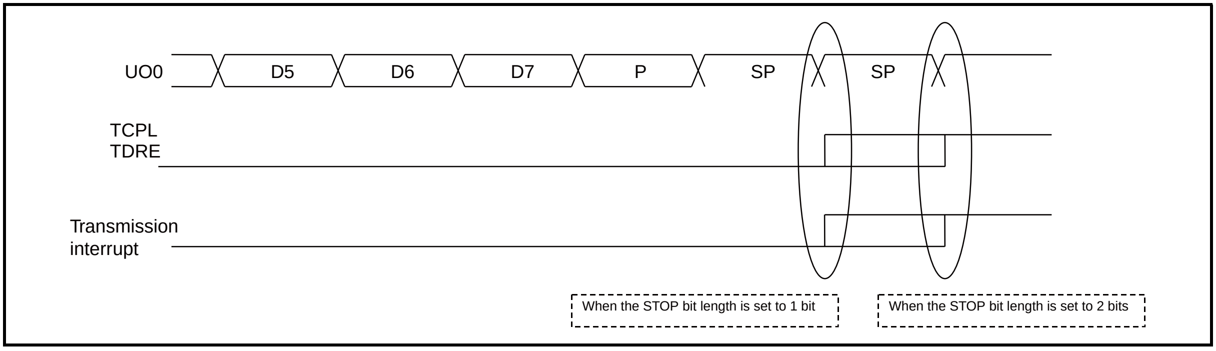
When the transmission interrupt enable bit (TIE) contains "1", a transmission interrupt occurs if the transmit data register empty (TDRE) bit is set to "1". This allows the next piece of transmit data to be written to the UART/SIO serial output data register (TDR0) by interrupt handling.

To detect the completion of serial transmission by transmission interrupt, set the transmission completion interrupt enable bits as follows: TEIE = 0, TCIE = 1. Upon completion of transmission, the transmission completion flag (TCPL) is set to 1 and a transmission interrupt occurs.

Both the transmission completion flag (TCPL) and the transmission data register empty flag (TDRE), when transmitting data consecutively, are set at the position which the transmission of the last bit was completed (it varies depending on the data length, parity enable, or stop bit length setting), as shown in Figure 22.7-6 below.

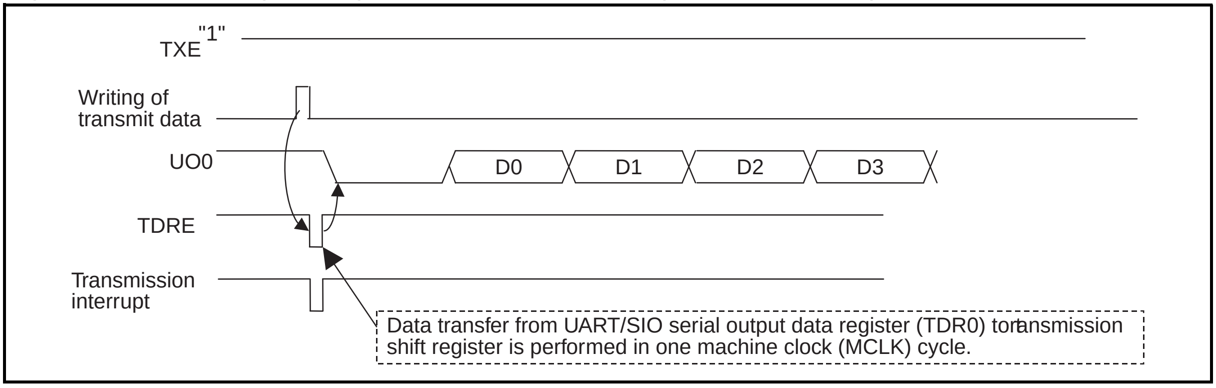
Note that modifying UART/SIO serial mode control register 1 (SMC10) during transmission may result in unpredictable operation.

Figure 22.7-6 Transmission in Asynchronous Clock Mode (UART)


UO0
D5
D6
D7
P
SP
SP
TCPL
TDRE
Transmission
interrupt
When the STOP bit length is set to 1 bit
When the STOP bit length is set to 2 bits

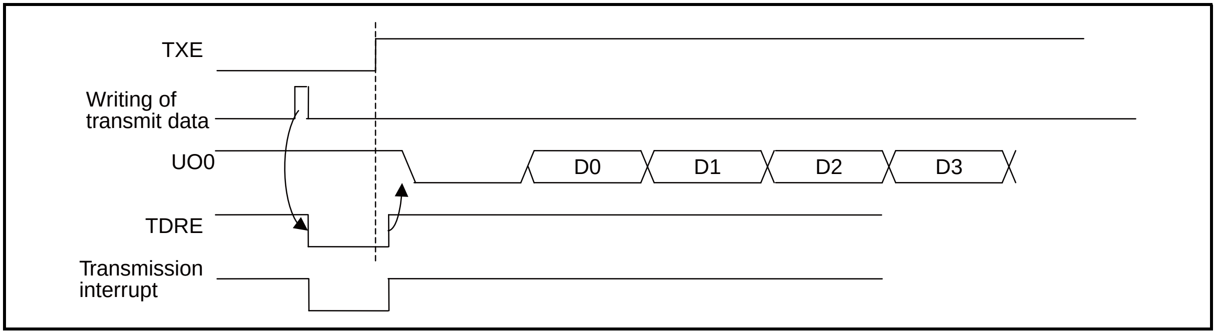
The TDRE flag is set at the point indicated in the following figure if the preceding piece of transmit data does not exist in the transmission shift register.

Figure 22.7-7 Setting Timing 1 for Transmit Data Register Empty Flag (TDRE) (When TXE is "1")



Data transfer from UART/SIO serial output data register (TDR0) tortansmission shift register is performed in one machine clock (MCLK) cycle.

Figure 22.7-8 Setting Timing 2 for Transmit Data Register Empty Flag (TDRE) (When TXE Is Switched from "0" to "1")



- Concurrent transmission and reception

In asynchronous clock mode (UART), transmission and reception can be performed independently. Therefore, transmission and reception can be performed at the same time or even with transmitting and receiving frames overlapping each other in shifted phases.

22.7.2 Operations in Operation Mode 1

Operation mode 1 operates in synchronous clock mode.

■ Operating Description of UART/SIO Operation Mode 1

Setting the MD bit in UART/SIO serial mode control register 1 (SMC10) to "1" selects synchronous clock mode (SIO).

The character bit length in synchronous clock mode (SIO) is variable between 5 and 8 bits.

Note, however, that parity is disabled and no stop bit is used.

The serial clock is selected by the CKS bit in the SMC10 register. Select the dedicated baud rate generator or external clock. The SIO performs shift operation using the selected serial clock as a shift clock.

To input the external clock signal, set the SCKE bit to "0".

To output the dedicated baud rate generator output as a shift clock signal, set the SCKE bit to "1". The serial clock signal is obtained by dividing clock by two, which is supplied by the dedicated baud rate generator. The baud rate in the SIO mode can be set in the following range. (For more information about the dedicated baud rate generator, also see "CHAPTER 23 UART/SIO DEDICATED BAUD RATE GENERATOR".)

Table 22.7-4 Baud Rate Setting Range in SIO Mode

PSS[1:0]	BRS[7:0]
"00_B" to "11_B"	01_H(1) to FF_H(255), 00_H(256) (The highest and lowest baud rate settings are 01_H and 00_H, respectively.)

The baud rate applied when the external clock or dedicated baud rate generator is used is obtained from the corresponding equation illustrated below.

Figure 22.7-9 Calculating Baud Rate Based on External Clock

$$\text{Baud rate value} = \frac{1}{\text{External clock*}} \text{ [bps]}$$

*: External clock

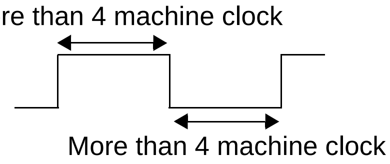


Figure 22.7-10 Baud Rate Calculation Formula for Using Dedicated Baud Rate Generator

$$\text{Baud rate value} = \frac{\text{Machine clock (MCLK)}}{2\times \boxed{\begin{array}{c}1\\2\\4\\8\end{array}} \times \boxed{\begin{array}{c}1\\:\\256\end{array}}}\ [\text{bps}]$$

UART prescaler select register(PSSR0)
Prescaler select (PSS1, PSS0)

UART baud rate setting register
(BRSR0)
Baud rate setting (BRS7 to BRS0)

- Serial clock

The serial clock signal is outputted under control of the output for transmit data. When only reception is performed, therefore, set transmission control (TXE = 1) to write dummy transmit data to the UART/SIO serial output register.MB95410H/470H Series

Refer to the data sheet of the MB95410H/470H Series for the UCK0 clock value.

● Reception in UART/SIO operation mode 1

For reception in operation mode 1, each register is used as follows.

Figure 22.7-11 Registers Used for Reception in Operation Mode 1

SMC10 (UART/SIO serial mode control register 1)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
BDS	PEN	TDP	SBL	CBL1	CBL0	CKS	MD
◎	×	×	×	◎	◎	◎	1

SMC20 (UART/SIO serial mode control register 2)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SCKE	TXOE	RERC	RXE	TXE	RIE	TCIE	TEIE
◎	0	◎	◎	◎	◎	×	×

SSR0 (UART/SIO serial status register)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
-	-	PER	OVE	FER	RDRF	TCPL	TDRE
×	×	×	◎	×	◎	×	×

TDR0 (UART/SIO serial output data register)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0
×	×	×	×	×	×	×	×

RDR0 (UART/SIO serial input data register)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
◎	◎	◎	◎	◎	◎	◎	◎

◎ : Used bit
× : Unused bit
1 : Set "1"
0 : Set "0"

The reception depends on whether the serial clock has been set to external or internal clock.

<When external clock is enabled>

When the reception operation enable bit (RXE) contains "1", serial data is received always at the rising edge of the external clock signal.

<When internal clock is enabled>

The serial clock signal is outputted in accordance with transmission. Therefore, transmission must be performed even when only performing reception. The following two procedures can be used.

- Set the transmission operation enable bit (TXE) to "1", then write transmit data to the UART/SIO serial output data register to generate the serial clock signal and start reception.
- Write transmit data to the UART/SIO serial output data register, then set the transmission operation enable bit (TXE) to "1" to generate the serial clock signal and start reception.

When 5 to 8-bit serial data is received by the reception shift register, the received data is transferred to the UART/SIO serial input data register (RDR0) and the next piece of serial data can be received.

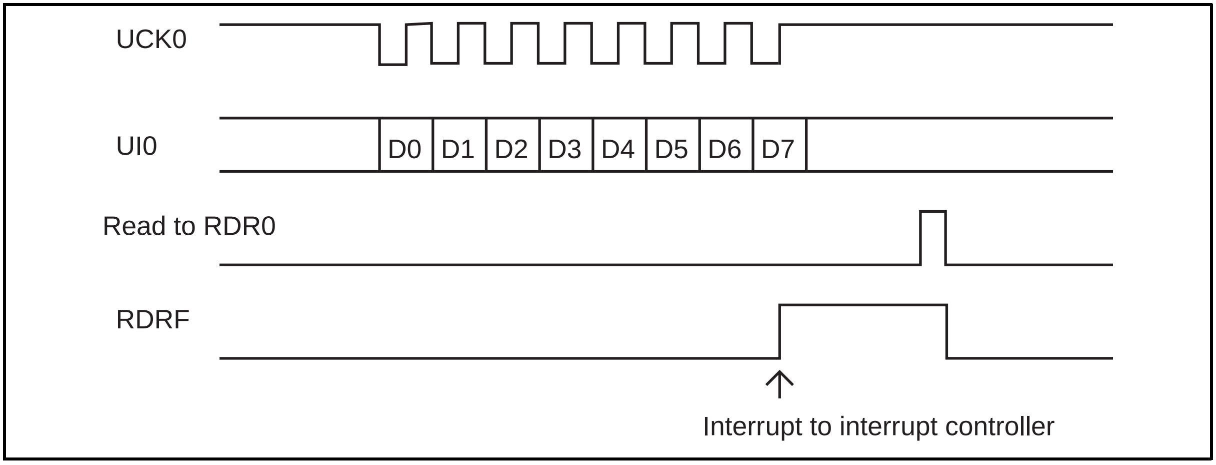
When the UART/SIO serial input data register stores data, the receive data register full (RDRF) bit is set to "1".

A reception interrupt occurs the moment the receive data register full (RDRF) bit is set to "1" when the reception interrupt enable bit (RIE) contains "1".

To read received data, read it from the UART/SIO serial input data register after checking the error flag (OVE) in the UART/SIO serial status register.

When received data is read from the UART/SIO serial input data register (RDR0), the receive data register full (RDRF) bit is cleared to "0".

Figure 22.7-12 8-bit Reception of Synchronous Clock Mode



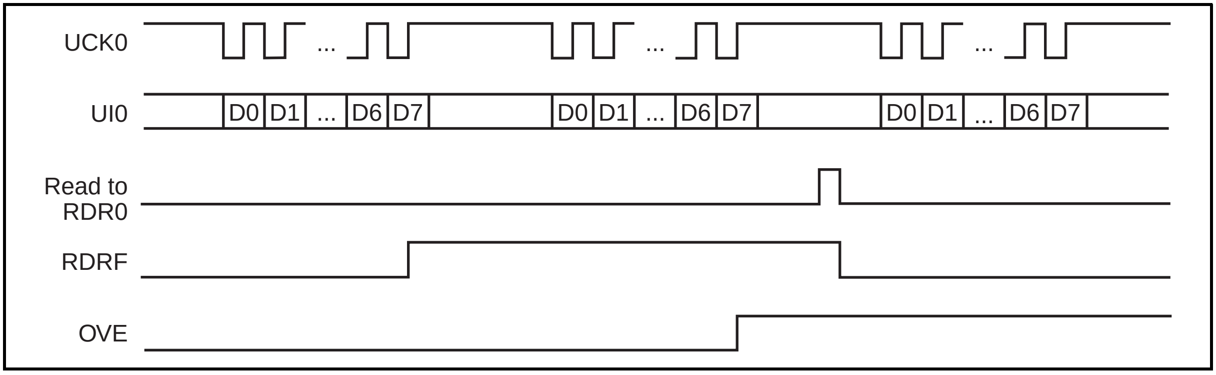
Operation when reception error occurs

When an overrun error (OVE) exists, received data is not transferred to the UART/SIO serial input data register (RDR0).

Overrun error (OVE)

Upon completion of reception for serial data, the overrun error (OVE) bit is set to "1" if the receive data register full (RDRF) bit has been set to "1" by the reception for the preceding piece of data.

Figure 22.7-13 Overrun Error



● Transmission in UART/SIO operation mode 1

For transmission in operation mode 1, each register is used as follows.

Figure 22.7-14 Registers Used for Transmission in Operation Mode 1

SMC10 (UART/SIO serial mode control register 1)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
BDS	PEN	TDP	SBL	CBL1	CBL0	CKS	MD
◎	×	×	×	◎	◎	◎	1

SMC20 (UART/SIO serial mode control register 2)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SCKE	TXOE	RERC	RXE	TXE	RIE	TCIE	TEIE
◎	0	◎	◎	◎	◎	×	×

SSR0 (UART/SIO serial status register)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
-	-	PER	OVE	FER	RDRF	TCPL	TDRE
×	×	×	◎	×	◎	×	×

TDR0 (UART/SIO serial output data register)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0
×	×	×	×	×	×	×	×

RDR0 (UART/SIO serial input data register)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
◎	◎	◎	◎	◎	◎	◎	◎

◎ : Used bit
× : Unused bit
1 : Set "1"
0 : Set "0"

The following two procedures can be used to initiate the transmission process:

- Set the transmission operation enable bit (TXE) to "1", then write transmit data to the UART/SIO serial output data register to start transmission.
- Write transmit data to the UART/SIO serial output data register, then set the transmission operation enable bit (TXE) to "1" to start transmission.

Transmit data is written to the UART/SIO serial output data register (TDR0) after it is checked that the transmit data register empty (TDRE) bit is set to "1".

When the transmit data is written to the UART/SIO serial output data register (TDR0), the transmit data register empty (TDRE) bit is cleared to "0".

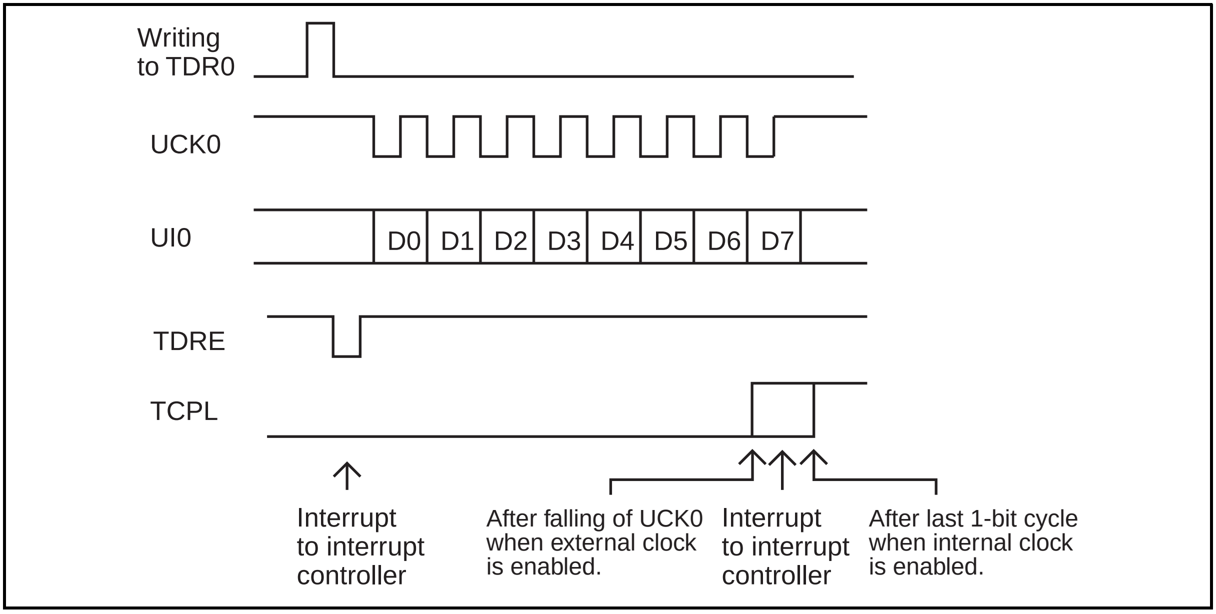
When serial transmission is started after transmit data is transferred from the UART/SIO serial output data register (TDR0) to the transmission shift register, the transmit data register empty (TDRE) bit is set to "1".

When the use of the external clock signal has been set, serial data transmission starts at the fall of the first serial clock signal after the transmission process is started.

A transmission completion interrupt occurs the moment the transmit data register empty (TDRE) bit is set to "1" when the transmission interrupt enable bit (TIE) contains "1". At this time, the next piece of transmit data can be written to the UART/SIO serial output data register (TDR0). Serial transmission can be continued with the transmission operation enable bit (TXE) set to "1".

To use a transmission completion interrupt to detect the completion of serial transmission, enable transmission completion interrupt output this way: TEIE = 0, TCIE = 1. Upon completion of transmission, the transmission completion flag (TCPL) is set to "1" and a transmission completion interrupt occurs.

Figure 22.7-15 8-bit Transmission in Synchronous Clock Mode


Writing to TDR0
UCK0
UI0
D0
D1
D2
D3
D4
D5
D6
D7
TDRE
TCPL
Interrupt to interrupt controller
After falling of UCK0 when external clock is enabled.
Interrupt to interrupt controller
After last 1-bit cycle when internal clock is enabled.

- ● Concurrent transmission and reception

<When external clock is enabled>

Transmission and reception can be performed independently of each other. Transmission and reception can therefore be performed at the same time or even when their phases are shifted from each other and overlapping.

<When internal clock is enabled>

As the transmitting side generates a serial clock, reception is influenced.

If transmission stops during reception, the receiving side is suspended. It resumes reception when the transmitting side is restarted.

- See "22.4 Pins of UART/SIO" for the operation with serial clock output and the operation with serial clock input.

22.8 Sample Settings for UART/SIO

This section provides sample settings for the UART/SIO.

■ Sample Settings

● How to select the operation mode

The operation mode selection bit (SMC10:MD) is used.

Operating mode		Operating mode select bit (MD)
Mode 0	Asynchronous clock mode (UART)	Set the bit to "0".
Mode 1	Synchronous clock mode (SIO)	Set the bit to "1".

● Operating clock types and selection method

The clock select bit (SMC10:CKS) is used.

Operation	Clock select bit (CKS)
To select the dedicated baud rate generator	Set the bit to "0".
To select the external clock	Set the bit to "1".

● How to use the UCK0, UI0, or UO0 pin

The following settings are used.

	UART
To set the UCK0 pin as an input pin	DDR1:P14 = 0 SMC20:SCKE = 0
To set the UCK0 pin as an output pin	SMC20:SCKE = 1
To use the UI0 pin	DDR1:P10 = 0
To use the UO0 pin	SMC20:TXOE = 1

● How to enable/stop UART operation

The reception operation enable bit (SMC20:RXE) is used.

Operation	Reception operation enable bit (RXE)
To disable (stop) reception	Set the bit to "0".
To enable reception	Set the bit to "1".

The transmission operation control bit (SMC20:TXE) is used.

Operation	Transmission operation control bit (TXE)
To disable (stop) transmission	Set the bit to "0".
To enable transmission	Set the bit to "1".

- How to set parity

The parity control (SMC10:PEN) and parity polarity (SMC10:TDP) bits are used.

Operation	Parity control (SMC10:PEN)	Parity polarity (SMC10:TDP)
To select no parity	Set the bit to "0"	-
To select even parity	Set the bit to "1"	Set the bit to "0".
To select odd parity	Set the bit to "1"	Set the bit to "1".

- How to set the data length

The character bit length control bits (SMC10:CBL[1:0]) are used.

Operation	Character bit length control bits (CBL[1:0])
To select 5-bit length	Set the bits to "00_B".
To select 6-bit length	Set the bits to "01_B".
To select 7-bit length	Set the bits to "10_B".
To select 8-bit length	Set the bits to "11_B".

- How to select the STOP bit length

The STOP bit length control bit (SMC10:SBL) is used.

Operation	STOP bit length control (SBL)
To set the STOP bit length to 1 bit	Set the bit to "0".
To set the STOP bit length to 2 bits	Set the bit to "1".

- How to clear error flags

The reception error flag clear bit (SMC20:RERC) is used.

Operation	Reception error flag clear bit (RERC)
To clear error flags (PER, OVE, FER)	Set the bit to "0".

- How to set the transfer direction

The serial data direction control bit (SMC10:BDS) is used.

LSB or MSB can be selected for the transfer direction in any operation mode.

Operation	Serial data direction control (BDS)
To select LSB transfer (from least significant bit)	Set the bit to "0".
To select MSB transfer (from most significant bit)	Set the bit to "1".

- How to clear the reception completion flag

The following setup is performed.

Operation	Method
To clear the reception completion flag	Read from the RDR0 register.

When the first read from the RDR0 register is performed, reception starts.

- How to clear the transmission buffer empty flag

The following operation is performed.

Operation	Method
To clear the transmission buffer empty flag	Write to the TDR0 register.

When the first write to TDR0 register is performed, transmission starts.

- How to set the baud rate

See "22.7.1 Operations in Operation Mode 0".

- Interrupt-related registers

The interrupt level setting registers shown in the following table are used to set the interrupt level.

	Interrupt level setting register	Interrupt vector
ch. 0	Interrupt level setting register (ILR1) Address: 0007A$_H$	#4 Address: 0FFF2$_H$
ch. 1	Interrupt level setting register (ILR2) Address: 0007B$_H$	#9 Address: 0FFE8$_H$
ch. 2	Interrupt level setting register (ILR1) Address: 0007A$_H$	#7 Address: 0FFEC$_H$

● How to enable/disable/clear interrupts

Interrupt request enable flag, interrupt request flag

The interrupt request enable bits (SMC20:RIE, SMC20:TCIE, SMC20:TEIE) are used to enable interrupts.

	UART reception	UART transmission	
	Reception interrupt enable bit (RIE)	Transmission completion interrupt enable bit (TCIE)	Transmission data register empty interrupt enable bit (TEIE)
To disable interrupt requests	Select "0".		
To enable interrupt requests	Select "1".		

Interrupt requests are cleared in the following setup procedure.

	UART reception	UART transmission
To clear an interrupt request	Read from UART/SIO serial input register (RDR0) to clear reception data register full bit (RDRF).	Write data to UART/SIO serial output data register (TDR0) to clear transmission data register empty bit (TDRE) to "0".
	Write "0" to error flag clear bit (RERC) to clear error flags (PER, OVE, FER) to "0".	

CHAPTER 23

UART/SIO DEDICATED BAUD RATE GENERATOR

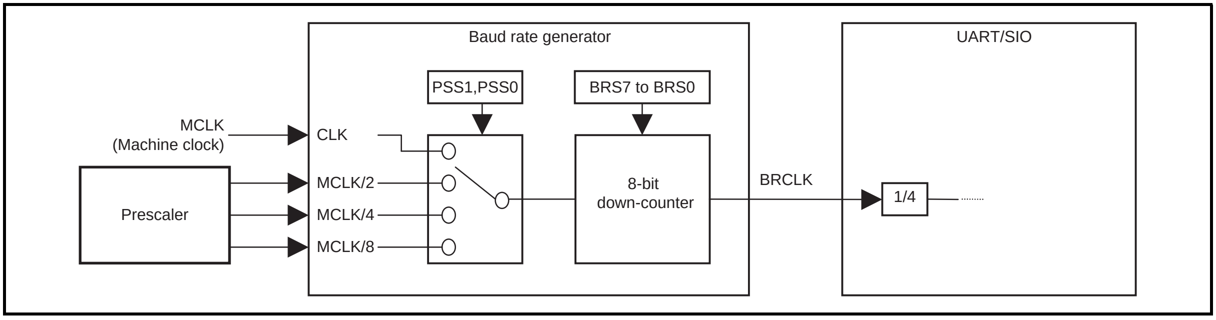
This chapter describes the functions and operations of the dedicated baud rate generator of UART/SIO.

23.1 Overview of UART/SIO Dedicated Baud Rate Generator

**The UART/SIO dedicated baud rate generator generates the baud rate for the UART/SIO.
The generator consists of the UART/SIO dedicated baud rate generator prescaler select register (PSSR) and UART/SIO dedicated baud rate generator baud rate setting register (BRSR).**

■ Block Diagram of UART/SIO Dedicated Baud Rate Generator

Figure 23.1-1 Block Diagram of UART/SIO Dedicated Baud Rate Generator



■ Input Clock

The UART/SIO dedicated baud rate generator uses the output clock from the prescaler or the machine clock as its input clock.

■ Output Clock

The UART/SIO dedicated baud rate generator supplies its clock to the UART/SIO.

23.2 Channels of UART/SIO Dedicated Baud Rate Generator

This section describes the channels of the UART/SIO dedicated baud rate generator.

■ Channels of UART/SIO Dedicated Baud Rate Generator

The MB95410H/470H Series has 3 channels of UART/SIO dedicated baud rate generator.

The following table shows the correspondence the channel and registers.

Table 23.2-1 Registers of Dedicated Baud Rate Generator

Channel	Register abbreviation	Corresponding register (Name in this manual)
0	PSSR0	UART/SIO dedicated baud rate generator prescaler select register ch. 0
	BRSR0	UART/SIO dedicated baud rate generator baud rate setting register ch. 0
1	PSSR1	UART/SIO dedicated baud rate generator prescaler select register ch. 1
	BRSR1	UART/SIO dedicated baud rate generator baud rate setting register ch. 1
2	PSSR2	UART/SIO dedicated baud rate generator prescaler select register ch. 2
	BRSR2	UART/SIO dedicated baud rate generator baud rate setting register ch. 2

ch.: Channel

23.3 Registers of UART/SIO Dedicated Baud Rate Generator

The registers of the UART/SIO dedicated baud rate generator are namely the UART/SIO dedicated baud rate generator prescaler select register (PSSR) and UART/SIO dedicated baud rate generator baud rate setting register (BRSR).

■ Registers of UART/SIO Dedicated Baud Rate Generator

Figure 23.3-1 Registers of UART/SIO Dedicated Baud Rate Generator

UART/SIO dedicated baud rate generator prescaler select register (PSSR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
PSSR0	$0FA8_H$	-	-	-	-	-	BRGE	PSS1	PSS0	00000000_B
PSSR1	$0FAA_H$	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R/W	R/W	R/W	
PSSR2	$0FAC_H$									

UART/SIO dedicated baud rate generator baud rate setting register (BRSR)

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
BRSR0	$0FA9_H$	BRS7	BRS6	BRS5	BRS4	BRS3	BRS2	BRS1	BRS0	00000000_B
BRSR1	$0FAB_H$	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
BRSR2	$0FAD_H$									

R/W	: Readable/writable (The read value is the same as the write value.)
R0/WX	: The read value is "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

The following sections describe only UART/SIO ch. 0.

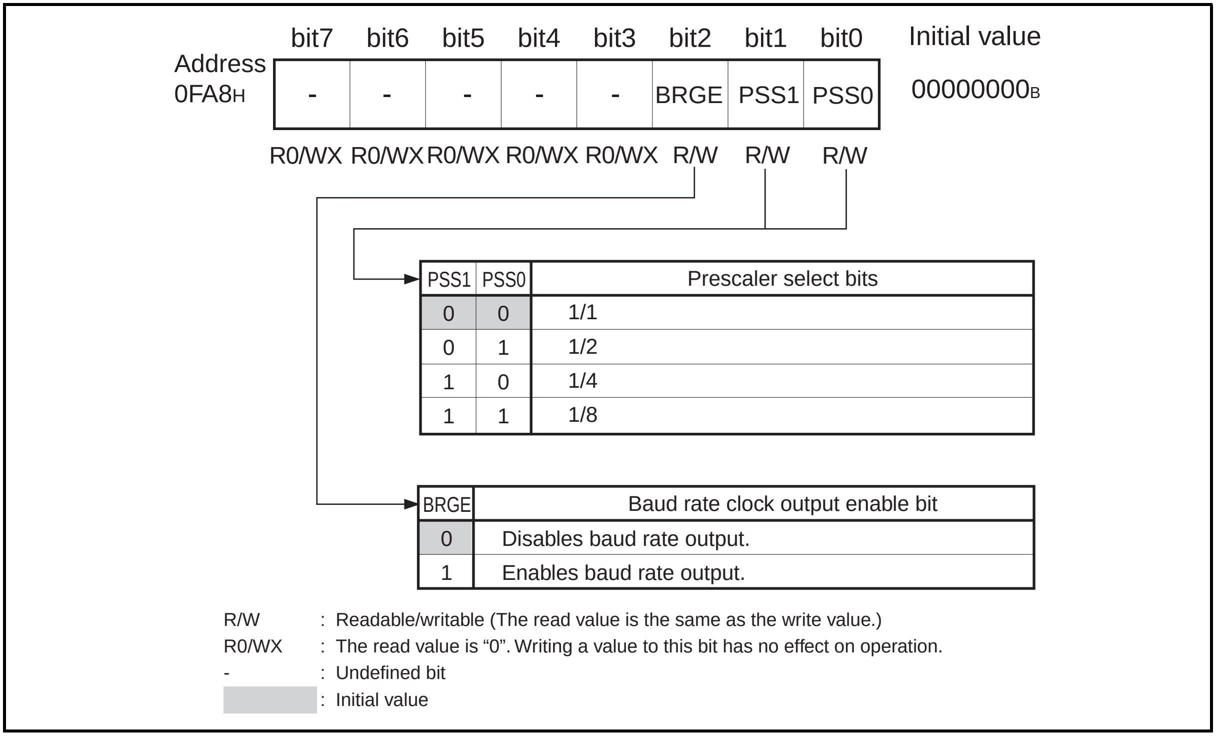
Ch. 1 and ch. 2 have the same configuration as ch. 0.

23.3.1 UART/SIO Dedicated Baud Rate Generator Prescaler Select Register (PSSR0)

The UART/SIO dedicated baud rate generator prescaler select register (PSSR0) controls the output of the baud rate clock and the prescaler.

■ UART/SIO Dedicated Baud Rate Generator Prescaler Select Register (PSSR0)

Figure 23.3-2 UART/SIO Dedicated Baud Rate Generator Prescaler Select Register (PSSR0)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FA8H	-	-	-	-	-	BRGE	PSS1	PSS0	00000000B
	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R/W	R/W	R/W	

PSS1	PSS0	Prescaler select bits
0	0	1/1
0	1	1/2
1	0	1/4
1	1	1/8

BRGE	Baud rate clock output enable bit
0	Disables baud rate output.
1	Enables baud rate output.

R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
▨ : Initial value

Table 23.3-1 Functions of Bits in UART/SIO Dedicated Baud Rate Generator Prescaler Select Register (PSSR0)

Bit name		Function
bit7 to bit3	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit2	BRGE: Baud rate clock output enable	• This bit enables the output of the baud rate clock "BRCLK". **Writing "0":** Stops the output of "BRCLK". **Writing "1":** Loads BRS[7:0] to the 8-bit down-counter and outputs "BRCLK", which is supplied to the UART/SIO.
bit1, bit0	PSS1, PSS0: Prescaler select bits	PSS1 / PSS0 / Prescaler select 0 / 0 / 1/1 0 / 1 / 1/2 1 / 0 / 1/4 1 / 1 / 1/8

23.3.2 UART/SIO Dedicated Baud Rate Generator Baud Rate Setting Register (BRSR0)

The UART/SIO dedicated baud rate generator dedicated baud rate generator baud rate setting register (BRSR0) controls the baud rate settings.

■ UART/SIO Dedicated Baud Rate Generator Baud Rate Setting Register (BRSR0)

Figure 23.3-3 UART/SIO Dedicated Baud Rate Generator Baud Rate Setting Register (BRSR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FA9$_H$	BRS7	BRS6	BRS5	BRS4	BRS3	BRS2	BRS1	BRS0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

This register sets the cycle of the 8-bit down-counter and can be used to set any baud rate clock. Write to the register when the UART is stopped.

Do not set BRS[7:0] to "00$_H$" or "01$_H$" in clock asynchronous mode.

23.4 Operations of UART/SIO Dedicated Baud Rate Generator

The UART/SIO dedicated baud rate generator serves as the baud rate generator for asynchronous clock mode.

■ Baud Rate Setting

The SMC10 register (CKS bit) of the UART/SIO is used to select the serial clock. This selects the UART/SIO dedicated baud rate generator.

In asynchronous clock mode, the shift clock that is selected by the CKS bit and divided by four is used and transfers can be performed within the range from -2% to +2%. The baud rate calculation formula for the UART/SIO dedicated baud rate generator is shown below.

Figure 23.4-1 Baud Rate Calculation Formula when UART/SIO Dedicated Baud Rate Generator Is Used

$$\text{Baud rate} = \frac{\text{Machine clock (MCLK)}}{4 \times \boxed{\begin{matrix}1\\2\\4\\8\end{matrix}} \times \boxed{\begin{matrix}2\\:\\255\end{matrix}}} \ \text{[bps]}$$

UART dedicated baud rate generator
prescaler select register (PSSR0)
Prescaler select (PSS1, PSS0)

UART dedicated baud rate generator
baud rate setting register (BRSR0)
Baud rate setting (BRS7 to BRS0)

Table 23.4-1 Sample Asynchronous Transfer Rates by Baud Rate Generator (Machine Clock = 10 MHz)

UART/SIO dedicated baud rate generator setting		UART internal division	Total division ratio (PSS × BRS × 4)	Baud rate (10 MHz/Total division ratio)
Prescaler select PSS[1:0]	Baud rate counter setting BRS[7:0]			
1 (Setting value: 0, 0)	20	4	80	125000
1 (Setting value: 0, 0)	22	4	88	113636
1 (Setting value: 0, 0)	44	4	176	56818
1 (Setting value: 0, 0)	87	4	348	28736
1 (Setting value: 0, 0)	130	4	520	19231
2 (Setting value: 0, 1)	130	4	1040	9615
4 (Setting value: 1, 0)	130	4	2080	4808
8 (Setting value: 1, 1)	130	4	4160	2404

The baud rate can be set in UART mode within the following range.

Table 23.4-2 Permissible Baud Rate Range in UART Mode

PSS[1:0]	BRS[7:0]
"00_B" to "11_B"	02_H (2) to FF_H (255)

CHAPTER 24

I^2C

This chapter describes functions and operations of the I^2C.

24.1 Overview of I^{2}C

The I^{2}C interface provides the functions of transmission and reception in master and slave modes, detection of arbitration lost, detection of slave address and general call address, generation and detection of start and stop conditions, bus error detection, and MCU standby wakeup.

■ I^{2}C Functions

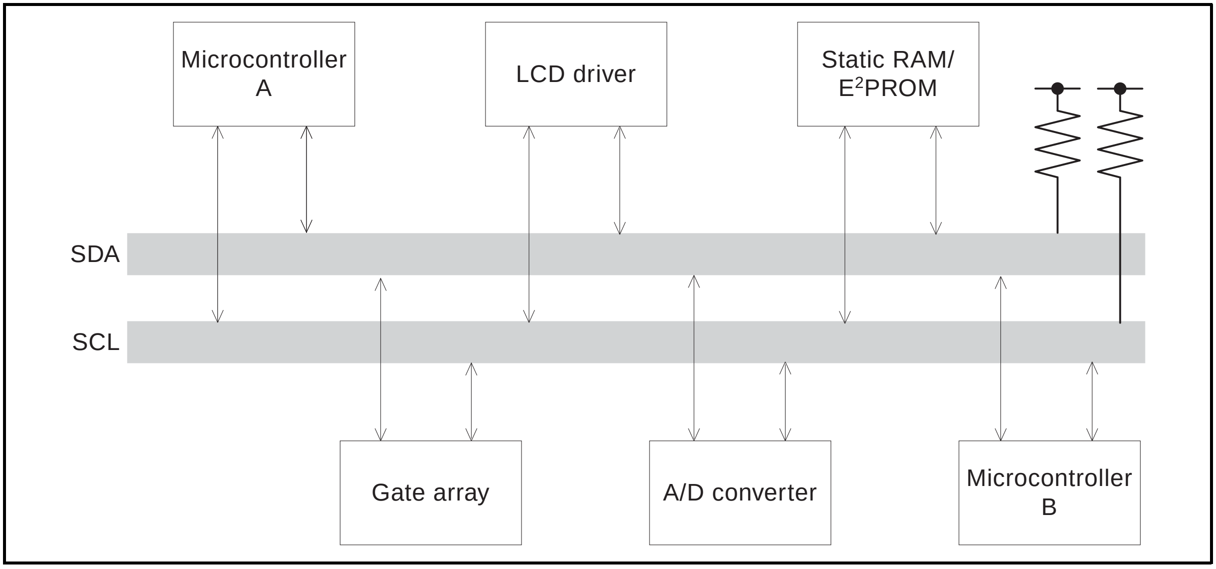
The I^{2}C interface is a two-wire, bi-directional bus consisting of a serial data line (SDA) and serial clock line (SCL). The devices connected to the bus via these two wires can exchange data, and each device can operate as a sender or receiver in accordance with their respective functions based on the unique address assigned to each device. Furthermore, the interface establishes a master/slave relationship between devices.

Also, the I^{2}C interface can connect multiple devices provided the bus capacitance does not exceed an upper limit of 400 pF. The I^{2}C interface is a true multi-master bus with collision detection and a communication control protocol that prevent loss of data even if more than one master attempts to start a data transfer at the same time.

The communication control protocol ensures that only one master is able to take control of the bus at a time, even if multiple masters attempt to take control of the bus simultaneously, without messages being lost or data being altered. Multi-master means that more than one master can attempt to take control of the bus at the same time without causing messages to be lost.

Also, the I^{2}C interface includes a function to wake up the MCU from standby mode.

Figure 24.1-1 I^{2}C Interface Configuration



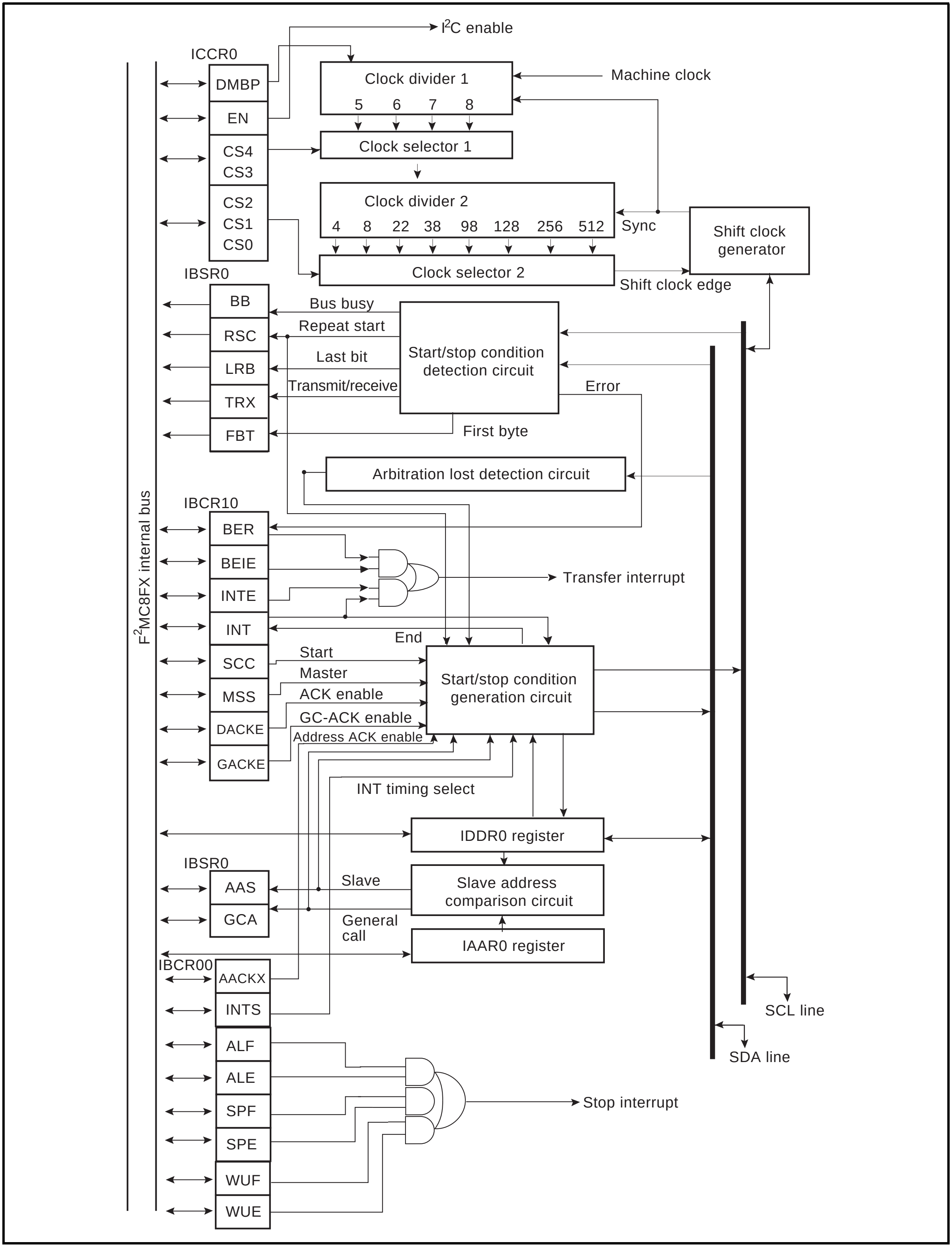
24.2 I^{2}C Configuration

I^{2}C consists of the following blocks:

- **Clock selector**
- **Clock divider**
- **Shift clock generator**
- **Start/stop condition generation circuit**
- **Start/stop condition detection circuit**
- **Arbitration lost detection circuit**
- **Slave address comparison circuit**
- **IBSR register**
- **IBCR registers (IBCR00, IBCR10)**
- **ICCR0 register**
- **IAAR0 register**
- **IDDR0 register**

■ I^{2}C Block Diagram

Figure 24.2-1 I^{2}C Block Diagram



- Clock selector, clock divider, and shift clock generator

This circuit uses the machine clock to generate the shift clock for the I^{2}C bus.

- Start/stop condition generation circuit

When a start condition is transmitted with the bus idle (SCL and SDA at the "H" level), a master starts communications. When SCL = "H", a start condition is generated by changing the SDA line from "H" to "L". The master can terminate its communication by generating a stop condition. When SCL = "H", a stop condition is generated by changing the SDA line from "L" to "H".

- Start/stop condition detection circuit

This circuit detects a start/stop condition for data transfer.

- Arbitration lost detection circuit

This interface circuit supports multi-master systems. If two or more masters attempt to transmit at the same time, the arbitration lost condition (if logic level "1" is sent when the SDA line goes to the "L" level) occurs. When the arbitration lost is detected, IBCR00:ALF is set to "1" and the master changes to a slave automatically.

- Slave address comparison circuit

The slave address comparison circuit receives the slave address after the start condition to compare it with its own slave address. The address is seven-bit data followed by a data direction (R/W) bit in the eighth bit position. If the received address matches the own slave address, the comparison circuit transmits an acknowledgment.

- IBSR0 register

The IBSR0 register shows the status of the I^{2}C interface.

- IBCR registers (IBCR00, IBCR10)

The IBCR registers are used to select the operating mode and to enable or disable interrupts, acknowledgment, general call acknowledgment, and the function to wake up the MCU from standby mode.

- ICCR0 register

The ICCR0 register is used to enable I^{2}C interface operations and select the shift clock frequency.

- IAAR0 register

The IAAR0 register is used to set the slave address.

- IDDR0 register

The IDDR0 register holds the transmit or receive shift data or address. When transmitted, the data or address written to this register is transferred from the MSB to the bus.

■ Input Clock

I^{2}C uses the machine clock as the input clock (shift clock).

24.3 I^{2}C Channel

This section describes the I^{2}C channel.

■ I^{2}C Channel

The MB95410H/470H Series has one channel of I^{2}C.

Table 24.3-1 and Table 24.3-2 show the correspondence among the channel, pins, and registers respectively.

Table 24.3-1 I^{2}C Pins

Channel	Pin name	Pin function
0	SCL SDA	I^{2}C bus I/O

Table 24.3-2 I^{2}C Registers

Channel	Register abbreviation	Corresponding register (Name in this manual)
0	IBCR00	I^{2}C bus control register 0
	IBCR10	I^{2}C bus control register 1
	IBSR0	I^{2}C bus status register
	IDDR0	I^{2}C data register
	IAAR0	I^{2}C address register
	ICCR0	I^{2}C clock control register

24.4 Pins of I^{2}C Bus Interface

This section describes the pins of the I^{2}C bus interface and gives their block diagram.

■ Pins of I^{2}C Bus Interface

The pins of the I^{2}C bus interface are the SDA and SCL pins.

● SDA pin

The SDA pin can serve as a general-purpose I/O port, external interrupt input (hysteresis input), serial data output pin (N-ch open-drain) for 8-bit serial I/O, and I^{2}C data I/O pin (SDA).

SDA: When I^{2}C is enabled (ICCR0:EN = 1), the SDA pin is automatically set as a data I/O pin to function as the SDA terminal.

To use it as an input pin, enable the I^{2}C operation (ICCR0: EN = 1) and write "0" to the corresponding bit in the port direction register (DDR).

● SCL pin

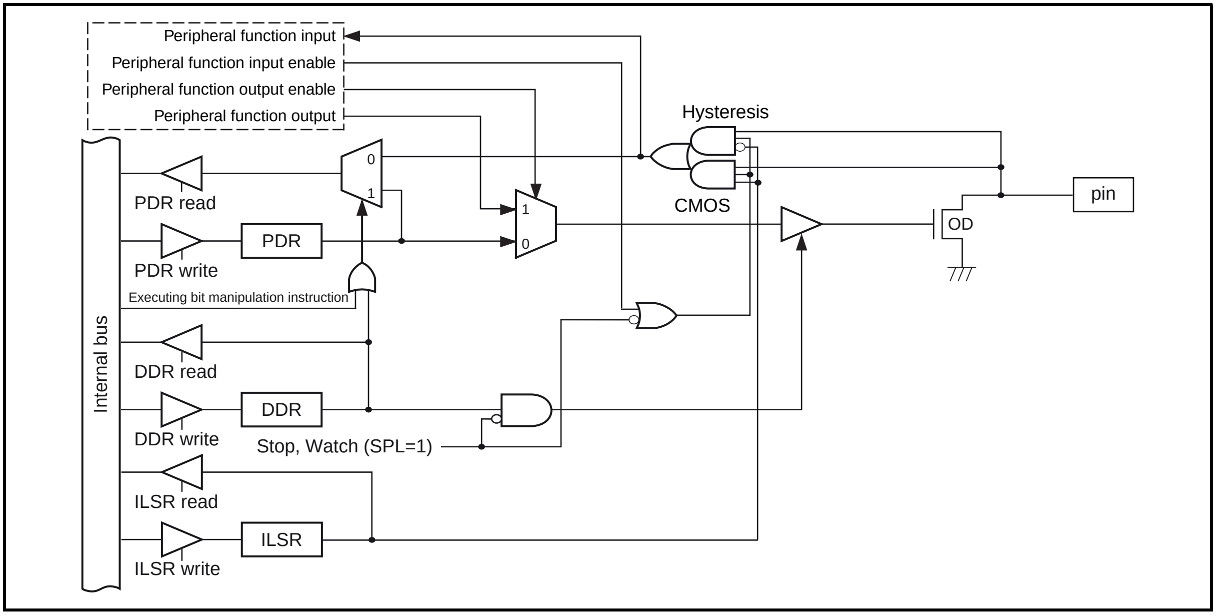
The SCL pin can serve as a N-ch open drain I/O port, external interrupt input (hysteresis input), serial data input (hysteresis input) for eight-bit serial I/O, or I^{2}C serial clock I/O pin (SCL).

SCL: When I^{2}C is enabled (ICCR0:EN = 1), the SCL pin is automatically set as the shift clock I/O pin to function as the SCL terminal.

To use it as an input pin, enable the I^{2}C operation (ICCR0: EN = 1) and write "0" to the corresponding bit in the port direction register (DDR).

■ Block Diagram of Pins of I2C Bus Interface

Figure 24.4-1 Block Diagram of SCL and SDA of I2C Bus Interface


Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Hysteresis
CMOS
pin
OD
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
ILSR read
ILSR
ILSR write

24.5 Registers of I^{2}C

This section describes the registers of I^{2}C.

■ Registers of I^{2}C

Figure 24.5-1 Registers of I^{2}C

I^{2}C bus control register 0 (IBCR00)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0060_H	AACKX	INTS	ALF	ALE	SPF	SPE	WUF	WUE	00000000_B
	R/W	R/W	R(RM1),W	R/W	R(RM1),W	R/W	R(RM1),W	R/W	

I^{2}C bus control register 1 (IBCR10)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0061_H	BER	BEIE	SCC	MSS	DACKE	GACKE	INTE	INT	00000000_B
	R(RM1),W	R/W	R0,W	R/W	R/W	R/W	R/W	R(RM1),W	

I^{2}C bus status register (IBSR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0062_H	BB	RSC	-	LRB	TRX	AAS	GCA	FBE	00000000_B
	R/WX	R/WX	R0/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

I^{2}C data register (IDDR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0063_H	D7	D6	D5	D4	D3	D2	D1	D0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

I^{2}C address register (IAAR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0064_H	-	A6	A5	A4	A3	A2	A1	A0	00000000_B
	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

I^{2}C clock control register (ICCR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0065_H	DMBP	-	EN	CS4	CS3	CS2	CS1	CS0	00000000_B
	R/W	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	

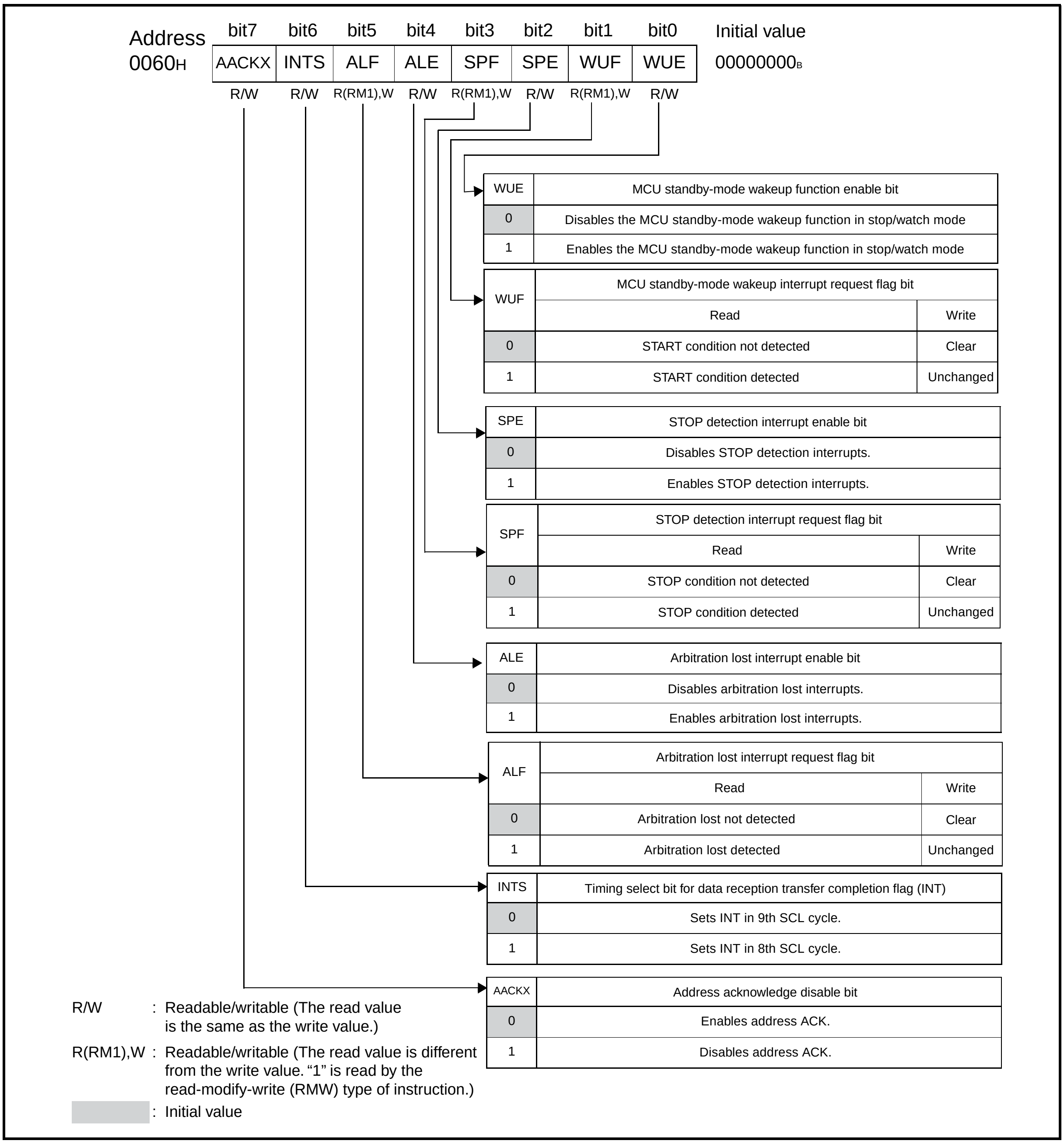
R/W	: Readable/writable (The read value is the same as the write value.)
R(RM1),W	: Readable/writable (The read value is different from write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0,W	: Write only (Writable. The read value is "0".)
R/WX	: Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX	: The read value is "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

24.5.1 I^{2}C Bus Control Registers (IBCR00, IBCR10)

The I^{2}C bus control registers are used to select the operating mode and to enable or disable interrupts, acknowledgment, general call acknowledgment, and MCU standby wakeup function.

■ I^{2}C Bus Control Register 0 (IBCR00)

Figure 24.5-2 I^{2}C Bus Control Register 0 (IBCR00)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0060H	AACKX	INTS	ALF	ALE	SPF	SPE	WUF	WUE	00000000B
	R/W	R/W	R(RM1),W	R/W	R(RM1),W	R/W	R(RM1),W	R/W	

WUE	MCU standby-mode wakeup function enable bit
0	Disables the MCU standby-mode wakeup function in stop/watch mode
1	Enables the MCU standby-mode wakeup function in stop/watch mode

WUF	MCU standby-mode wakeup interrupt request flag bit	
	Read	Write
0	START condition not detected	Clear
1	START condition detected	Unchanged

SPE	STOP detection interrupt enable bit
0	Disables STOP detection interrupts.
1	Enables STOP detection interrupts.

SPF	STOP detection interrupt request flag bit	
	Read	Write
0	STOP condition not detected	Clear
1	STOP condition detected	Unchanged

ALE	Arbitration lost interrupt enable bit
0	Disables arbitration lost interrupts.
1	Enables arbitration lost interrupts.

ALF	Arbitration lost interrupt request flag bit	
	Read	Write
0	Arbitration lost not detected	Clear
1	Arbitration lost detected	Unchanged

INTS	Timing select bit for data reception transfer completion flag (INT)
0	Sets INT in 9th SCL cycle.
1	Sets INT in 8th SCL cycle.

AACKX	Address acknowledge disable bit
0	Enables address ACK.
1	Disables address ACK.

R/W : Readable/writable (The read value is the same as the write value.)

R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)

▨ : Initial value

Table 24.5-1 Functions of Bits in I^{2}C Bus Control Register 0 (IBCR00) (1 / 2)

Bit name		Function
bit7	AACKX: Address acknowledge disable bit	This bit controls the address ACK when the first byte is transmitted. **Writing "0":** Causes the address ACK to be output automatically. (The address ACK is returned automatically if the slave address matches.) **Writing "1":** Prevents the address ACK from being output. • Write "1" to this bit in either of the following ways: - Write "1" to the bit in master mode. - Clear the bit to "0" after making sure that the bus busy bit is "0" (IBSR0:BB = 0). Note: • If AACKX = "1" and IBSR0:FBT = "0" when an IBCR10:INT bit interrupt occurs, no address ACK is output even though the I^{2}C address matches the slave address. Clear the IBCR10:INT bit to "0" as an interrupt is generated upon completion of transfer of each byte of address/data in the same way as during addressing. • If AACKX = "1" and IBSR0:FBT = "1" when an IBCR10:INT bit interrupt occurs, "1" might be written to AACKX after addressing as in slave mode. Either continue normal communication after setting AACKX to "0" again or restart communication after disabling I^{2}C operation (ICCR0:EN = 0).
bit6	INTS: Timing select bit for data reception transfer completion flag (INT)	This bit selects the timing of the transfer completion interrupt (IBCR10:INT) when data is received. Change the bit only when IBSR0:TRX = 0 and IBSR1:FBT = 0. **Writing "0":** Sets the transfer completion interrupt (IBCR10:INT) in the ninth SCL cycle. **Writing "1":** Sets the transfer completion interrupt (IBCR10:INT) in the eighth SCL cycle. Note: • The transfer completion interrupt (IBCR10:INT) is set always in the ninth SCL cycle except during data reception (IBSR1:TRX = 1 or IBSR1:FBT = 1). • If the data ACK depends on the content of the received data (such as packet error checking used by the SM bus), control the data ACK by setting the data ACK enable bit (IBCR10:DACKE) after writing "1" to this bit (for example, using a previous transfer completion interrupt) to read latest received data. • The latest data ACK (IBSR0:LRB) can be read after the ACK has been received (IBSR0:LRB must be read during the transfer completion interrupt in the ninth SCL cycle.) If ACK is read when this bit is "1", therefore, you must write "0" to this bit in the transfer completion interrupt in the eighth SCL cycle so that another transfer completion interrupt will occur in the ninth SCL cycle.
bit5	ALF: Arbitration lost interrupt request flag bit	This bit is used to detect when arbitration is lost. • An arbitration lost interrupt request is generated if this bit and the IBCR00:ALE bit are both "1". • This bit is set to "1" in the following cases: - When arbitration lost is detected during data/address transmission as a master - When "1" is written to the IBCR10:MSS bit with the bus being used by another system. However, the bit is not set when "1" is written to the MSS bit after the system returns AACK or GACK as a slave. • This bit is set to "0" in the following cases: - When "0" is written to the IBCR00:ALF bit with IBSR0:BB = 0. - When "0" is written to the IBCR10:INT bit to clear the transmission completion flag. • Writing "1" to this bit leaves its value unchanged and has no effect on the operation. • The bit returns "1" when read by a read-modify-write (RMW) type of instruction.
bit4	ALE: Arbitration lost interrupt enable bit	This bit enables or disables arbitration lost interrupts. • An arbitration lost interrupt request is generated if this bit and the IBCR00:ALF bit are both "1". **Writing "0":** Disables arbitration lost interrupts. **Writing "1":** Enables arbitration lost interrupts.
bit3	SPF: STOP detection interrupt request flag bit	This bit is used to detect a STOP condition. • A STOP detection interrupt request is generated if this bit and the IBCR00:SPE bit are both "1". • This bit is set to "1" if a valid STOP condition is detected when the bus is busy. **Writing "0":** Clears itself (changes the value to "0"). **Writing "1":** Leaves its value unchanged without affecting the operation. • The bit returns "1" when read by a read-modify-write (RMW) type of instruction.

Table 24.5-1 Functions of Bits in I2C Bus Control Register 0 (IBCR00) (2 / 2)

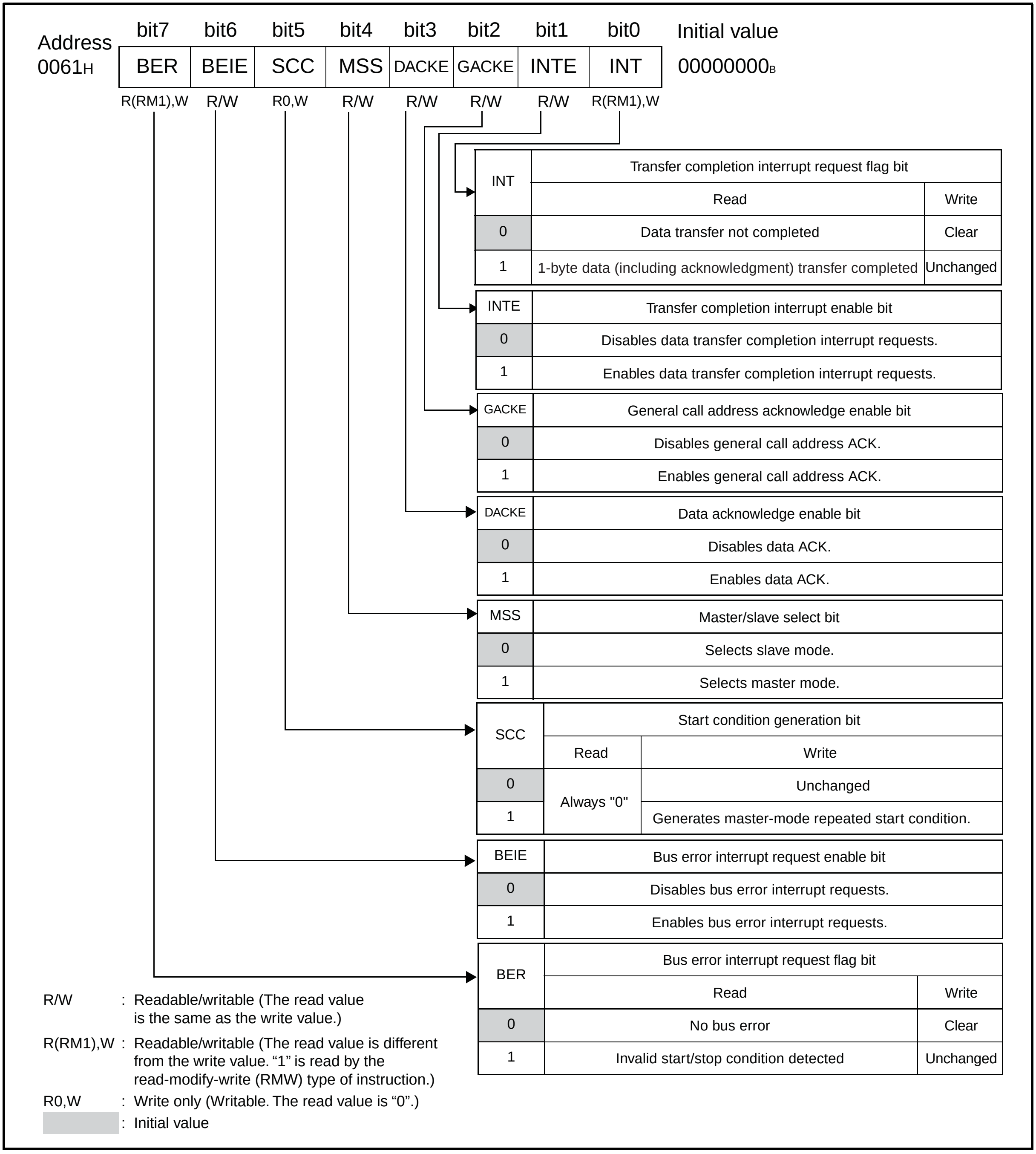
Bit name		Function
bit2	SPE: STOP detection interrupt enable bit	This bit enables or disables STOP detection interrupts. • A STOP detection interrupt request is generated if this bit and the IBCR00:SPF bit are both "1". **Writing "0":** Disables STOP detection interrupts. **Writing "1":** Enables STOP detection interrupts.
bit1	WUF: MCU standby-mode wakeup interrupt request flag bit	This bit is used to detect MCU wakeup from a standby mode (stop or watch mode). • A wakeup interrupt request is generated if this bit and the IBCR00:WUE bit are both "1". • This bit is set to "1" if a START condition is detected with the wakeup function enabled (IBCR00:WUE = 1). **Writing "0":** Clears itself (changes the value to "0"). **Writing "1":** Leaves its value unchanged without affecting the operation. • The bit returns "1" when read by a read-modify-write (RMW) type of instruction.
bit0	WUE: MCU standby-mode wakeup function enable bit	This bit enables or disables the function to wake up the MCU from standby mode (stop or watch mode). **Writing "0":** Disables the wakeup function. **Writing "1":** Enables the wakeup function. • If a start condition is detected in stop or watch mode when this bit is "1", a wakeup interrupt request is generated to start I2C operation. Note: • Write "1" to this bit immediately before the MCU enters the stop or watch mode. To ensure that I2C operation can restart immediately after the MCU wakes up from stop or watch mode, clear (write "0" to) this bit as soon as possible. • When a wakeup interrupt request occurs, the MCU wakes up after the oscillation stabilization wait time elapses. To prevent the data loss immediately after wakeup, therefore, the SCL must rise as the first cycle and the first bit must be received as data after 100 µs (assuming that the minimum oscillation stabilization wait time is 100 µs) from the wakeup due to the start of I2C transmission (upon detection of the falling edge of SDA). • During a MCU standby mode, the status flags, state machine, and I2C bus outputs for the I2C function retain the states they had prior to entering the standby mode. To prevent a hang-up of the entire I2C bus system, make sure that IBSR0:BB = 0 before entering standby mode. • The wakeup function does not support the transition of the MCU to stop or watch mode with IBSR0:BB = 1. If the MCU enters stop or watch mode with IBSR0:BB = 1, a bus error will occur upon detection of a start condition. • The wakeup function is useful only when the MCU remains in stop/watch mode. (In PLL stop mode, for example, the time from wakeup to the start of communication becomes longer than in stop/watch mode as the PLL oscillation stabilization wait time is required in addition to the oscillation stabilization wait time.)

Note:

The AACKX, INTS, and WUE bits in the IBCR00 register are set to "0" and cannot be written to either when I2C operation is disabled (ICCR0:EN = 0) or when a bus error occurs (IBCR10:BER = 1).

■ I^{2}C Bus Control Register 1 (IBCR10)

Figure 24.5-3 I^{2}C Bus Control Register 1 (IBCR10)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0061$_H$	BER	BEIE	SCC	MSS	DACKE	GACKE	INTE	INT	00000000$_B$
	R(RM1),W	R/W	R0,W	R/W	R/W	R/W	R/W	R(RM1),W	

INT	Transfer completion interrupt request flag bit	
	Read	Write
0	Data transfer not completed	Clear
1	1-byte data (including acknowledgment) transfer completed	Unchanged

INTE	Transfer completion interrupt enable bit
0	Disables data transfer completion interrupt requests.
1	Enables data transfer completion interrupt requests.

GACKE	General call address acknowledge enable bit
0	Disables general call address ACK.
1	Enables general call address ACK.

DACKE	Data acknowledge enable bit
0	Disables data ACK.
1	Enables data ACK.

MSS	Master/slave select bit
0	Selects slave mode.
1	Selects master mode.

SCC	Start condition generation bit	
	Read	Write
0	Always "0"	Unchanged
1		Generates master-mode repeated start condition.

BEIE	Bus error interrupt request enable bit
0	Disables bus error interrupt requests.
1	Enables bus error interrupt requests.

BER	Bus error interrupt request flag bit	
	Read	Write
0	No bus error	Clear
1	Invalid start/stop condition detected	Unchanged

R/W : Readable/writable (The read value is the same as the write value.)

R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)

R0,W : Write only (Writable. The read value is "0".)

▨ : Initial value

Table 24.5-2 Functions of Bits in I^2C Bus Control Register 1 (IBCR10) (1 / 2)

Bit name		Function
bit7	BER: Bus error interrupt request flag bit	This bit is used to detect bus errors. • A bus error interrupt request is generated if this bit and the IBCR10:BEIE bit are both "1". • This bit is set to "1" when an invalid start or stop condition is detected. **Writing "0":** Clears itself (changes the value to "0"). **Writing "1":** Leaves its value unchanged without affecting the operation. • The bit returns "1" when read by a read-modify-write operation. • When this bit is set to "1", ICCR0:EN is set to "0", and the I^2C interface enters halt mode to terminate data transfer.
bit6	BEIE: Bus error interrupt request enable bit	This bit enables or disables bus error interrupts. • A bus error interrupt request is generated if this bit and the IBCR10:BER bit are both "1". **Writing "0":** Disables bus error interrupts. **Writing "1":** Enables bus error interrupts.
bit5	SCC: Start condition generation bit	This bit can be used to generate a start condition repeatedly to restart communications in master mode. • Writing "1" to the bit in master mode generates a start condition repeatedly. • Writing "0" to the bit is meaningless. • When read, the bit returns "0". Note: • Do not set IBCR10:SCC = 1 and IBCR10:MSS = 0 at the same time. • An attempt to write "1" to this bit is ignored when IBCR10:INT = 0 (no start condition is generated). If you write "1" to this bit and "0" to the IBCR10:INT bit at the same time when the IBCR10:INT = 1, this bit takes priority and generates a start condition.
bit4	MSS: Master/slave select bit	This bit selects master mode or slave mode. • Writing "1" to this bit while the I^2C bus is in the idle state (IBSR0:BB = 0) selects master mode, generates a start condition, and then starts address transfer. • Writing "0" to the bit while the I^2C bus is in the busy state (IBSR0:BB = 1) selects slave mode, generates a stop condition, and then ends data transfer. • If arbitration lost occurs during data or address transfer in master mode, this bit is cleared to "0" and the mode changes to slave mode. Note: • Do not set IBCR10:SCC = 1 and IBCR10:MSS = 0 at the same time. • An attempt to write "0" to this bit is ignored when IBCR10:INT = 0. If you write "0" to this bit and "0" to the IBCR10:INT bit at the same time when the IBCR10:INT = 1, this bit takes priority and generates a stop condition. • The IBCR00:ALF bit is not set even though you write "1" to the MSS bit during transmission or reception in slave mode. Do not write "1" to the MSS bit during transmission or reception in slave mode.
bit3	DACKE: Data acknowledge enable bit	This bit controls data acknowledgment during data reception. **Writing "0":** Disables data acknowledge output. **Writing "1":** Enables data acknowledge output. In this case, data acknowledgment is output in the ninth SCL cycle during data reception in master mode. In slave mode, data acknowledgment is output in the ninth SCL cycle only if address acknowledgment has already been output.
bit2	GACKE: General call address acknowledge enable bit	This bit controls general call address acknowledgment. **Writing "0":** Disables output of general call address acknowledge. **Writing "1":** Causes a general call address acknowledgment to be output if a general call address (00_H) is received in master or slave mode.
bit1	INTE: Transfer completion interrupt enable bit	This bit enables or disables transfer completion interrupts. **Writing "0":** Disables transfer completion interrupts. **Writing "1":** Enables transfer completion interrupts. • A transfer completion interrupt request is generated if this bit and the IBCR10:INT bit are both "1".

Table 24.5-2 Functions of Bits in I^{2}C Bus Control Register 1 (IBCR10) (2 / 2)

Bit name		Function
bit0	INT: Transfer completion interrupt request flag bit	This bit is used to detect transfer completion. • A transfer completion interrupt request is generated if this bit and the IBCR10:INTE bit are both "1". • This bit is set to "1" upon completion of transfer of 1-byte address or data (whether or not this includes an acknowledgment depends on the IBCR00:INTS setting) if any of the following four conditions is satisfied. - In bus master mode - Addressed as slave - General call address received - Arbitration lost detected • This bit is set to "0" in the following cases: - "0" written to the bit - Repeated start condition (IBCR10:SCC = 1) or stop condition (IBCR10:MSS = 0) occurred in master mode. • An attempt to write "1" to this bit leaves its value unchanged and has no effect on the operation. • The bit returns "1" when read by a read-modify-write (RMW) type of instruction. • The SCL line remains at "L" while this bit is "1". • Writing "0" to clear the bit (change the value to "0") releases the SCL line to enable transmission for the next byte of data. Note: • If "1" is written to IBCR10:SCC when this bit is "0", the IBCR10:SCC bit has priority and the start condition is generated. • If "0" is written to IBCR10:MSS when this bit is "0", the IBCR10:MSS bit has priority and the stop condition is generated. • If IBCR00:INTS = 1 when data is received, this bit is set to "1" upon completion of transfer of one-byte data (including no acknowledgment). In other cases, this bit is set to "1" upon completion of transmission or reception of one-byte data/address including an acknowledgment.

Notes:

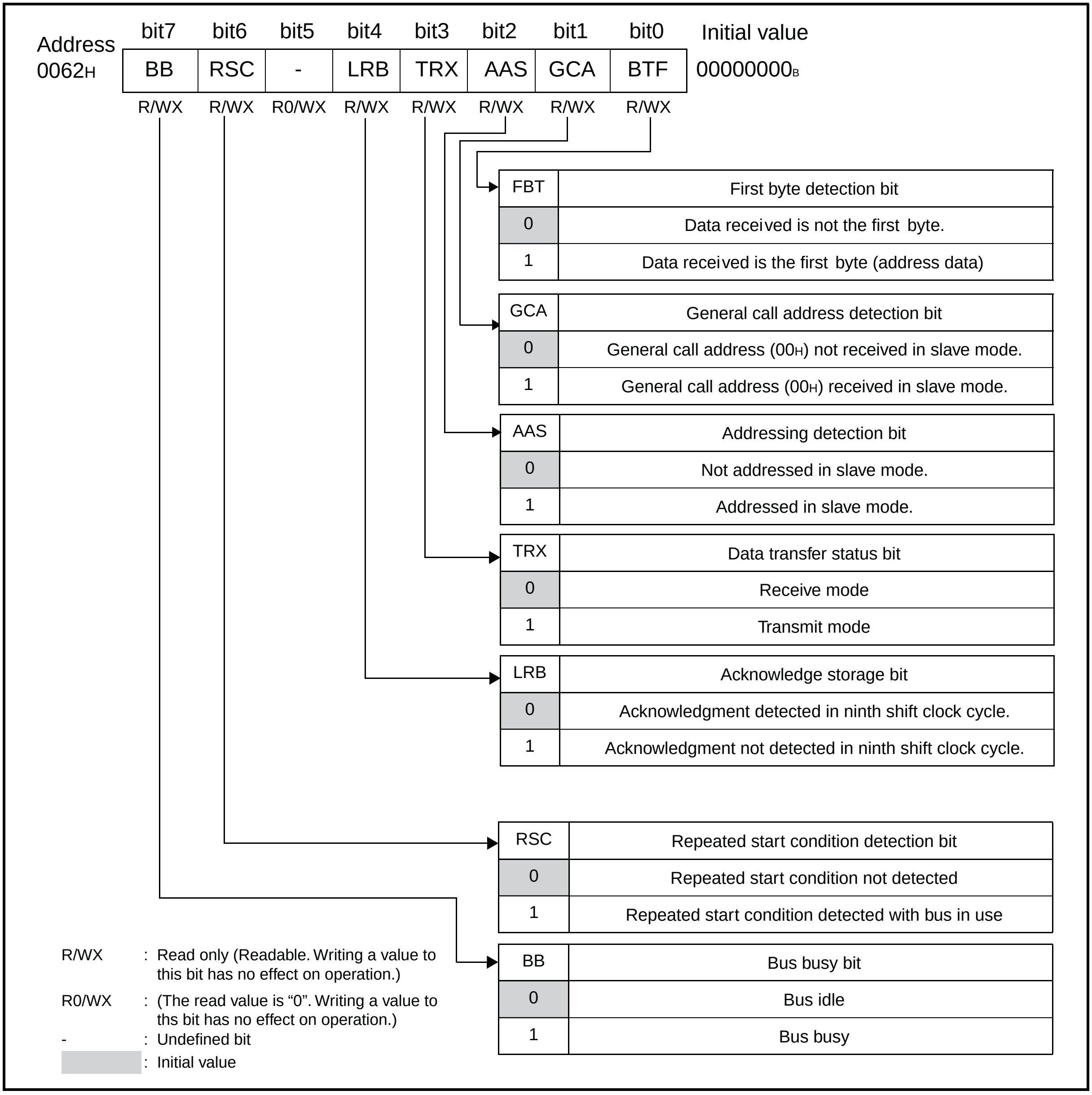
- When clearing the interrupt request flag (IBCR10:BER) by writing "0", do not update the interrupt request enable bit (IBCR10:BEIE) at the same time.
- All the bits in IBCR10 except the BER and BEIE bits are cleared to "0" either when operation is disabled (ICCR0:EN = 0) or when a bus error occurs (IBCR10:BER = 1).

24.5.2 I2C Bus Status Register (IBSR0)

The IBSR0 register contains the status of the I2C interface.

■ I2C Bus Status Register (IBSR0)

Figure 24.5-4 I2C Bus Status Register (IBSR0)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0062H	BB	RSC	-	LRB	TRX	AAS	GCA	BTF	00000000B
	R/WX	R/WX	R0/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

FBT	First byte detection bit
0	Data received is not the first byte.
1	Data received is the first byte (address data)

GCA	General call address detection bit
0	General call address (00H) not received in slave mode.
1	General call address (00H) received in slave mode.

AAS	Addressing detection bit
0	Not addressed in slave mode.
1	Addressed in slave mode.

TRX	Data transfer status bit
0	Receive mode
1	Transmit mode

LRB	Acknowledge storage bit
0	Acknowledgment detected in ninth shift clock cycle.
1	Acknowledgment not detected in ninth shift clock cycle.

RSC	Repeated start condition detection bit
0	Repeated start condition not detected
1	Repeated start condition detected with bus in use

BB	Bus busy bit
0	Bus idle
1	Bus busy

R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX : (The read value is “0”. Writing a value to ths bit has no effect on operation.)
- : Undefined bit
▨ : Initial value

Table 24.5-3 Functions of Bits in I^{2}C Bus Status Register (IBSR0)

Bit name		Function
bit7	BB: Bus busy bit	This bit indicates the bus status. • This bit is set to "1" when a start condition is detected. • This bit is set to "0" when a stop condition is detected.
bit6	RSC: Repeated start condition detection bit	This bit is used to detect repeated start conditions. • This bit is set to "1" when a repeated start condition is detected. • This bit is set to "0" in the following cases: - When "0" is written to IBCR10:INT. - When the slave address does not match the address set in IAAR0 in slave mode. - When the slave address matches the address set in IAAR0 but IBCR00:AACKX = 1 in slave mode. - When the general call address is received but IBCR10:GACKE = 0 in slave mode. - When a stop condition is detected.
bit5	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.
bit4	LRB: Acknowledge storage bit	This bit saves the value of the SDA line in the ninth shift clock cycle during data byte transfer. • This bit is set to "1" when no acknowledgment is detected (SDA = H). • This bit is set to "0" in the following cases: - When acknowledgment is detected (SDA = L) - When a start or stop condition is detected. Note: It follows from the above that this bit must be read after ACK. (Read the value in response to the transfer completion interrupt in the ninth SCL cycle.) Accordingly, if ACK is read when the IBCR00:INTS bit is "1", you must write "0" to the IBCR00:INTS bit in the transfer completion interrupt triggered by the eighth SCL cycle so that another transfer completion interrupt will be triggered by the ninth SCL cycle.
bit3	TRX: Data transfer status bit	This bit indicates the data transfer mode. • This bit is set to "1" when data transfer is performed in transfer mode. • This bit is set to "0" in the following cases: - Data is transferred in receive mode. - NACK is received in slave transmit mode.
bit2	AAS: Addressing detection bit	This bit indicates that the MCU has been addressed in slave mode. • This bit is set to "1" if the MCU is addressed in slave mode. • This bit is set to "0" when a start or stop condition is detected.
bit1	GCA: General call address detection bit	This bit is used to detect a general call address. • This bit is set to "1" in the following cases: - When the general call address (00_H) is received in slave mode. - When the general call address (00_H) is received in master mode with IBCR10:GACKE = 1. - When arbitration lost is detected during transmission of the second byte of the general call address in master mode. • This bit is set to "0" in the following cases: - When a start or stop condition is detected. - When arbitration lost is not detected during transmission of the second byte of the general call address in master mode.
bit0	FBT: First byte detection bit	This bit is used to detect first byte. • This bit is set to "1" when a start condition is detected. • This bit is set to "0" in the following cases: - When "0" is written to the IBCR10:INT bit. - When the slave address does not match the address set in IAAR0 in slave mode. - When the slave address matches the address set in IAAR0 but IBCR00:AACKX = 1 in slave mode. - When the general call address is received with IBCR10:GACKE = 0 in slave mode.

24.5.3 I^2C Data Register (IDDR0)

The IDDR0 register is used to set the data or address to send and to hold the data or address received.

■ I^2C Data Register (IDDR0)

Figure 24.5-5 I^2C Data Register (IDDR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0063_H	D7	D6	D5	D4	D3	D2	D1	D0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

In transmit mode, each bit of the data or address value written to the register is shifted to the SDA line, starting with the MSB. The write side of this register is double-buffered, where if the bus is in use (IBSR0:BB=1), the write data is loaded to the 8-bit shift register either when the current data transfer completion interrupt is cleared (writing "0" to the IBCR10:INT bit) or when a repeated start condition is generated (writing "1" to the IBCR10:SCC bit). Each bit of the shift register data is output (shifted) to the SDA line. Note that writing to this register has no effect on the current data transfer. In slave mode, however, data is transferred to the shift register after the address is determined.

The received data or address can be read from this register during the transfer completion interrupt (IBCR10:INT = 1). When it is read, however, the serial transfer register is directly read from, the receive data is valid only while IBCR10:INT = 1.

24.5.4 I2C Address Register (IAAR0)

The IAAR0 register is used to set the slave address.

■ I2C Address Register (IAAR0)

Figure 24.5-6 I2C Address Register (IAAR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0064_H	-	A6	A5	A4	A3	A2	A1	A0	00000000_B
	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

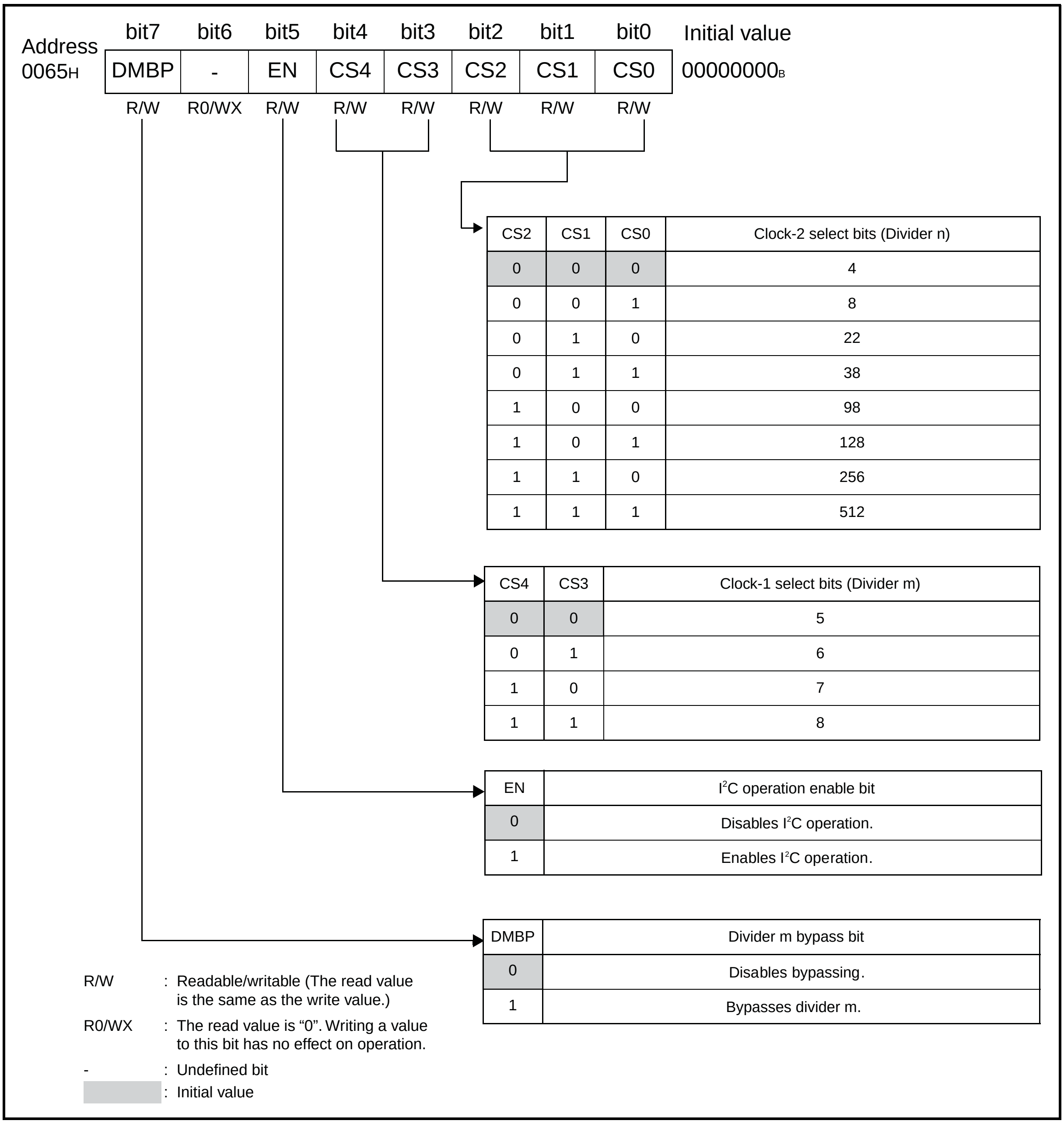
The I2C address register (IAAR0) is used to set the slave address. In slave mode, address data from the master is recieved and then compared with the value of the IAAR0 register.

24.5.5 I2C Clock Control Register (ICCR0)

The ICCR0 register is used to enable I2C operation and select the shift clock frequency.

■ I2C Clock Control Register (ICCR0)

Figure 24.5-7 I2C Clock Control Register (ICCR0)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0065H	DMBP	-	EN	CS4	CS3	CS2	CS1	CS0	00000000B
	R/W	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	

CS2	CS1	CS0	Clock-2 select bits (Divider n)
0	0	0	4
0	0	1	8
0	1	0	22
0	1	1	38
1	0	0	98
1	0	1	128
1	1	0	256
1	1	1	512

CS4	CS3	Clock-1 select bits (Divider m)
0	0	5
0	1	6
1	0	7
1	1	8

EN	I2C operation enable bit
0	Disables I2C operation.
1	Enables I2C operation.

DMBP	Divider m bypass bit
0	Disables bypassing.
1	Bypasses divider m.

R/W	:	Readable/writable (The read value is the same as the write value.)
R0/WX	:	The read value is “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

Table 24.5-4 Functions of Bits in I^{2}C Clock Control Register (ICCR0)

Bit name		Function
bit7	DMBP: Divider m bypass bit	This bit is used to bypass the divider m to generate the shift clock frequency. **Writing "0":** Sets the value set in CS3 and CS4 as the divider m value. (m = ICCR0:CS4, CS3) **Writing "1":** Bypasses the divider m. Note: Do not set this bit to "1" when divider n = 4 (ICCR0:CS2 to CS0 = 000_B).
bit6	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.
bit5	EN: I^{2}C operation enable bit	• This bit enables I^{2}C interface operation. **Writing "0":** Disables operation of the I^{2}C interface and clears the following bits to "0". - AACKX, INTS, and WUE bits in the IBCR00 register - All the bits in the IBCR10 register except the BER and BEIE bits - All bits in the IBSR0 register **Writing "1":** Enables operation of the I^{2}C interface. • This bit is set to "0" in the following cases: - When "0" is written to this bit. - When IBCR10:BER is "1".
bit4, bit3	CS4, CS3: Clock-1 select bits (Divider m)	These bits set the shift clock frequency. • Shift clock frequency (Fsck) is set as shown by the following equation: $\text{Fsck} = \dfrac{\phi}{(m \times n + 2)}$ ϕ represents the machine clock frequency (MCLK).
bit2 to bit0	CS2, CS1, CS0: Clock-2 select bits (Divider n)	

Note:

If the standby mode wakeup function is not used, disable I^{2}C operation before switching the MCU to stop or watch mode.

24.6 I^2C Interrupts

The I^2C interface has a transfer interrupt and a stop interrupt which are triggered by the following events.

- **Transfer interrupt**
  A transfer interrupt occurs either upon completion of data transfer or when a bus error occurs.
- **Stop interrupt**
  A stop interrupt occurs upon detection of a stop condition or arbitration lost or upon access to the I^2C interface in stop/watch mode.

■ Transfer Interrupt

Table 24.6-1 shows the transfer interrupt control bits and I^2C interrupt sources.

Table 24.6-1 Transfer Interrupt Control Bits and I^2C Interrupt Sources

	End of transfer	Bus error
Interrupt request flag bit	IBCR10:INT = "1"	IBCR10:BER = "1"
Interrupt request enable bit	IBCR10:INTE = "1"	IBCR10:BEIE = "1"
Interrupt source	Data transfer complete	Bus error occurred

- Interrupt upon completion of transfer

  An interrupt request is output to the CPU upon completion of data transfer if the transfer completion interrupt request enable bit has been set to enable (IBCR10:INTE = 1). In the interrupt service routine, write "0" to the transfer completion interrupt request flag bit (IBCR10:INT) to clear the interrupt request. When data transfer is completed, the IBCR10:INT bit is set to "1" regardless of the value of the IBCR10:INTE bit.

- Interrupt in response to a bus error

  When the following conditions are met, a bus error is deemed to have occurred, and the I^2C interface will be stopped.

  - When a stop condition is detected in master mode.
  - When a start or stop condition is detected during transmission or reception of the first byte.
  - When a start or stop condition is detected during transmission or reception of data (excluding the start, first data, and stop bits).

In these cases, an interrupt request is output to the CPU if the bus error interrupt request enable bit has been set to enable (IBCR10:BEIE = 1). In the interrupt service routine, write "0" to the bus error interrupt request flag bit (IBCR10:BER) to clear the interrupt request. When a bus error occurs, the IBCR10:BER bit is set to "1" regardless of the value of the IBCR10:BEIE bit.

■ Stop Interrupt

Table 24.6-2 shows the stop interrupt control bits and I²C interrupt sources (trigger events).

Table 24.6-2 Stop Interrupt Control Bits and I²C Interrupt Sources

	Detection of stop condition	Detection of arbitration lost	MCU wakeup from stop/watch mode
Interrupt request flag bit	IBCR00:SPF = "1"	IBCR00:ALF = "1"	IBCH00:WUF = "1"
Interrupt request enable bit	IBCR00:SPE = "1"	IBCR00:ALE = "1"	IBCR00:WUE = "1"
Interrupt source	Stop condition detected	Arbitration lost detected	Start condition detected

- Interrupt upon detection of a stop condition

  A stop condition is considered to be valid if all of the following conditions are satisfied when the stop condition is detected.

  - The bus is busy (state which the start condition is detected).
  - IBCR10:MSS = 0
  - After transfer of one byte of data completes, including the acknowledgment.

In this case, an interrupt request is output to the CPU if the stop condition detection interrupt request enable bit has been set to enable (IBCR00:SPE = 1). In the interrupt service routine, write "0" to the IBCR00:SPF bit to clear the interrupt request.

The IBCR00:SPF bit is set to "1" when a valid stop condition occurs regardless of the value of the IBCR00:SPE bit.

- Interrupt upon detection of arbitration lost

  When arbitration lost is detected, an interrupt request is output to the CPU if the arbitration lost detection interrupt request enable bit has been set to enable (IBCR00:ALE = 1). Either write "0" to the arbitration lost interrupt request flag bit (IBCR00:ALF) while the bus is idle or write "0" to the IBCR10:INT bit from the interrupt service routine while the bus is busy to clear the interrupt request.

  When arbitration lost occurs, the IBCR00:ALF bit is set to "1" regardless of the value for the IBCR00:ALE bit.

- Interrupt for MCU wakeup from stop/watch mode

  When a start condition is detected, an interrupt request is output to the CPU if the function to wake up the MCU from stop or watch mode has been enabled (IBCR00:WUE = 1).

  In the interrupt service routine, write "0" to the MCU standby mode wakeup interrupt request flag bit (IBCR00:WUF) to clear the interrupt request.

■ Register and Vector Table Addresses Related to I^{2}C Interrupts

Table 24.6-3 Register and Vector Table Addresses Related to I^{2}C Interrupts

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
I^{2}C	IRQ16	ILR4	L16	$FFDA_H$	$FFDB_H$

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

24.7 Operations of I^{2}C and Setting Procedure Example

This section describes the operations of I^{2}C.

■ Operations of I^{2}C

● I^{2}C interface

The I^{2}C interface is an eight-bit serial interface synchronized with a shift clock.

● MCU standby mode wakeup function

The wakeup function wakes up the MCU upon detection of a start condition, from low power consumption mode such as stop or watch mode.

■ Setting Procedure Example

Below is an example of procedure for setting I^{2}C.

● Initial settings

1) Set the port for input (DDR2).
2) Set the interrupt level (ILR4).
3) Set the slave address (IAAR0).
4) Select the clock and enable I^{2}C operation (ICCR0).
5) Enable bus error interrupt requests (IBCR10:BEIE = 1).

● Interrupt processing

1) Arbitrary processing
2) Clear the bus error interrupt request flag (IBCR10:BER = 0).

24.7.1 I^{2}C Interface

The I^{2}C interface is an eight-bit serial interface synchronized with the shift clock.

■ I^{2}C System

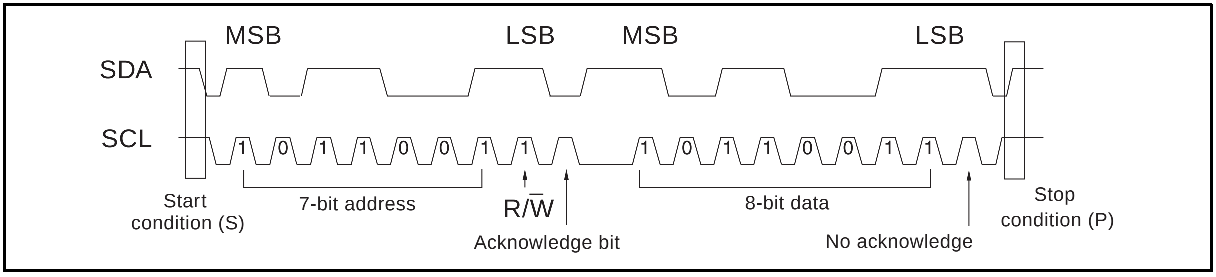
The I^{2}C bus system uses the serial data line (SDA) and serial clock line (SCL) for data transfers. All the devices connected to the bus require open drain or open collector outputs which must be connected with a pull-up resistor.

Each of the devices connected to the bus has a unique address which can be set up using software. The devices always operate in a simple master/slave relationship, where the master functions as the master transmitter or master receiver. The I^{2}C interface is a true multi-master bus with a collision detection function and arbitration function to prevent data from being lost if more than one master attempts to start data transfer at the same time.

■ I^{2}C Protocol

Figure 24.7-1 shows the format required for data transfer.

Figure 24.7-1 Data Transfer Example



The slave address is transmitted after a start condition (S) is generated. This address is seven bits followed by the data direction bit (R/$\overline{W}$) in the eighth bit position. Data is transmitted after the address. The data is eight bits followed by an acknowledgment.

Data can be transmitted continuously to the same slave address in consecutive units of eight bits plus acknowledgment.

Data transfer is always ended in the master stop condition (P). However, the repeated start condition (S) can be used to transmit the address which indicates a different slave without generating a stop condition.

■ Start Conditions

While the bus is idle (SCL and SDA are both at the logical "H" level), the master generates a start condition to start transmission. As shown in Figure 24.7-1, a start condition is triggered when the SDA line is changed from "H" to "L" while SCL = "H". This starts a new data transfer and commences master/slave operation.

A start condition can be generated in either of the following two ways.

- By writing "1" to the IBCR10:MSS bit while the I^{2}C bus is not in use (IBCR10:MSS = 0, IBSR0:BB = 0, IBCR10:INT = 0, and IBCR00:ALF = 0). (Next, IBSR0:BB is set to "1" to indicate that the bus is busy.)
- By writing "1" to the IBCR10:SCC bit during an interrupt while in bus master mode (IBCR10:MSS = 1, IBSR0:BB = 1, IBCR10:INT = 1, and IBCR00:ALF = 0). (This generates a repeated START condition.)

Writing "1" to the IBCR10:MSS or IBCR10:SCC bit is ignored in other than the above cases. If another system is using the bus when "1" is written to the IBCR10:MSS bit, the IBCR00:ALF bit is set to "1".

■ Addressing

● Slave addressing in master mode

In master mode, IBSR0:BB and IBSR0:TRX are set to "1" after the start condition is generated, and the slave address in the IDDR0 register is output to the bus starting with the MSB. The address data consists of eight bits: the 7-bit slave address and the data transfer direction R/$\overline{\text{W}}$ bit (bit 0 in the IDDR0 register).

The acknowledgment from the slave is received after the address data is sent. SDA goes to "L" in the ninth clock cycle and the acknowledge bit from the receiving device is received (see Figure 24.7-1). In this case, the R/W bit (IDDR0:bit0) is inverted logically and stored in the IBSR0:TRX bit as "1" if the SDA level is "L".

● Addressing in slave mode

In slave mode, after the start condition is detected, IBSR0:BB is set to "1" and IBSR0:TRX is set to "0", and the data received from the master is stored in the IDDR0 register. After the address data is received, the IDDR0 and IAAR0 registers are compared. If the addresses match, IBSR0:AAS is set to "1" and an acknowledgment is sent to the master. Next, bit 0 of the receive data (bit 0 in the IDDR0 register) is saved in the IBSR0:TRX bit.

■ Data Transfer

If the MCU is addressed as a slave, data can be sent or received byte by byte with the direction determined by the R/$\overline{\text{W}}$ bit sent by the master.

Each byte to be output on the SDA line is fixed at eight bits. As shown in Figure 24.7-1, the receiver sends an acknowledgment to the sender by forcing the SDA line to the stable "L" level while the acknowledge clock pulse is "H". Data is transferred at one clock pulse per bit with MSB at the head. Sending and receiving an acknowledgment is required after each byte is transferred. Accordingly, nine clock pulses are required to transfer one complete data byte.

■ Acknowledgment

An acknowledgment is sent by the receiver in the ninth clock cycle for data byte transfer by the sender based on the following conditions.

An address acknowledgment is generated in the following cases.

- The received address matches the address set in IAAR0, and the address acknowledgment is output automatically (IBCR00:AACKX = 0).
- A general call address (00H) is received and the general call address acknowledgment output is enabled (IBCR10:GACKE = 1).

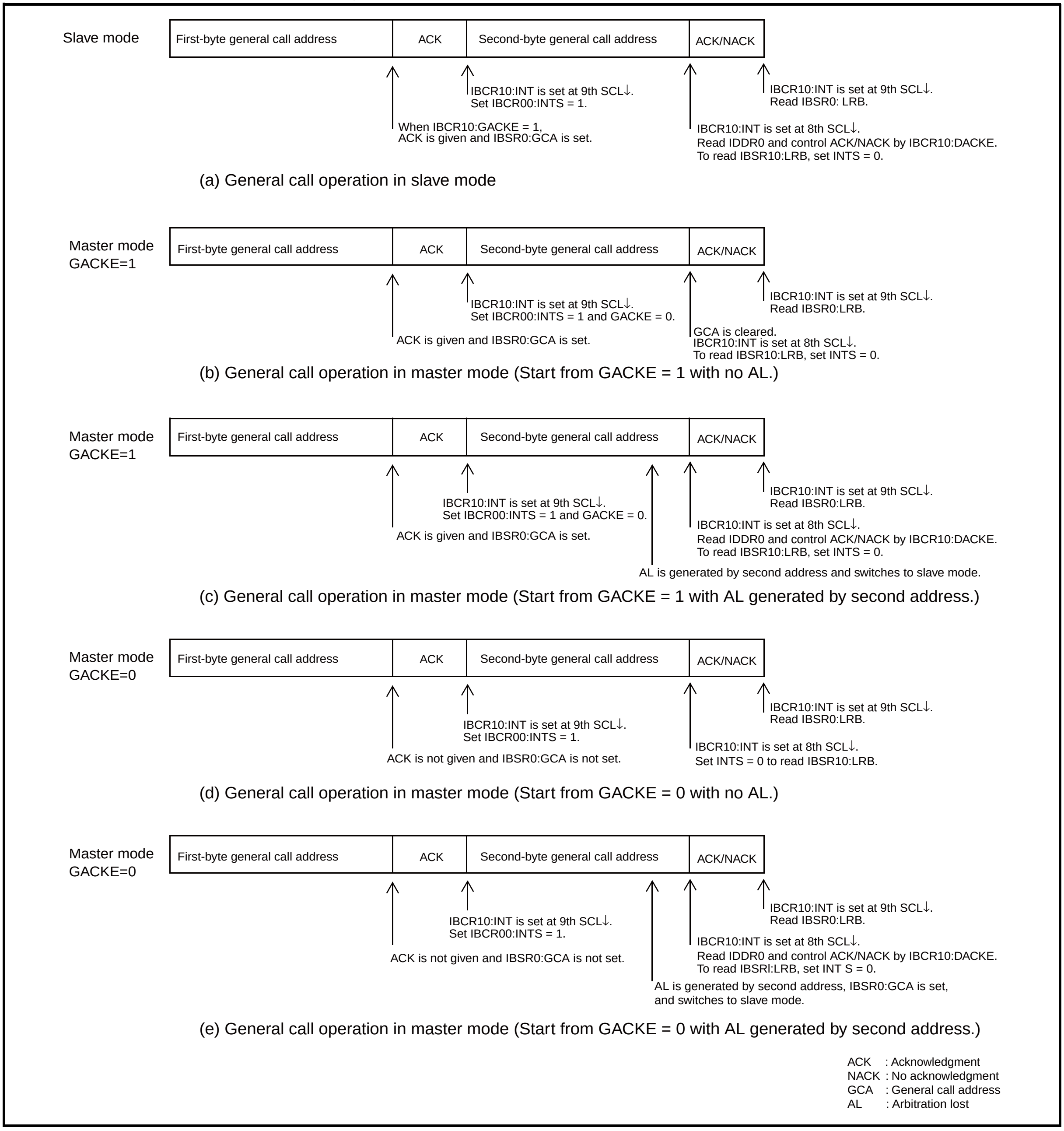
A data acknowledge bit used when data is received can be enabled or disabled by the IBCR10:DACKE bit. In master mode, a data acknowledgment is generated if IBCR10:DACKE = 1. In slave mode, a data acknowledgment is generated if an address acknowledgment has already been generated and IBCR10:DACKE = 1. The received acknowledgment is saved in IBSR0:LRB in the ninth SCL cycle.

- If the data ACK depends on the content of received data (such as packet error checking used by the SM bus), control the data ACK by setting the data ACK enable bit (IBCR10:DACKE) after writing "1" to the IBCR00:INTS bit (for example, by a previous transfer completion interrupt) so that the latest received data can be read.
- The latest data ACK (IBSR0:LRB) can be read after the ACK has been received (IBSR0:LRB must be read during the transfer completion interrupt triggered by the ninth SCL cycle). Accordingly, if ACK is read when the IBCR00:INTS bit is "1", you must write "0" to this bit in the transfer completion interrupt triggered by the eighth SCL cycle so that another transfer completion interrupt will be triggered by the ninth SCL cycle.

■ General Call Address

A general call address consists of the start address byte (00_H) and the second address byte that follows. To use a general call address, you must set IBCR10:GACKE=1 before the acknowledge of the first byte general call address. Also, the acknowledgment for the second address byte can be controlled as shown below.

Figure 24.7-2 General Call Operation


Slave mode
First-byte general call address
ACK
Second-byte general call address
ACK/NACK
IBCR10:INT is set at 9th SCL↓.
Set IBCR00:INTS = 1.
IBCR10:INT is set at 9th SCL↓.
Read IBSR0: LRB.
When IBCR10:GACKE = 1,
ACK is given and IBSR0:GCA is set.
IBCR10:INT is set at 8th SCL↓.
Read IDDR0 and control ACK/NACK by IBCR10:DACKE.
To read IBSR10:LRB, set INTS = 0.
Master mode
GACKE=1
First-byte general call address
ACK
Second-byte general call address
ACK/NACK
IBCR10:INT is set at 9th SCL↓.
Set IBCR00:INTS = 1 and GACKE = 0.
IBCR10:INT is set at 9th SCL↓.
Read IBSR0:LRB.
ACK is given and IBSR0:GCA is set.
GCA is cleared.
IBCR10:INT is set at 8th SCL↓.
To read IBSR10:LRB, set INTS = 0.
Master mode
GACKE=1
First-byte general call address
ACK
Second-byte general call address
ACK/NACK
IBCR10:INT is set at 9th SCL↓.
Set IBCR00:INTS = 1 and GACKE = 0.
IBCR10:INT is set at 9th SCL↓.
Read IBSR0:LRB.
ACK is given and IBSR0:GCA is set.
IBCR10:INT is set at 8th SCL↓.
Read IDDR0 and control ACK/NACK by IBCR10:DACKE.
To read IBSR10:LRB, set INTS = 0.
AL is generated by second address and switches to slave mode.
Master mode
GACKE=0
First-byte general call address
ACK
Second-byte general call address
ACK/NACK
IBCR10:INT is set at 9th SCL↓.
Set IBCR00:INTS = 1.
IBCR10:INT is set at 9th SCL↓.
Read IBSR0:LRB.
ACK is not given and IBSR0:GCA is not set.
IBCR10:INT is set at 8th SCL↓.
Set INTS = 0 to read IBSR10:LRB.
Master mode
GACKE=0
First-byte general call address
ACK
Second-byte general call address
ACK/NACK
IBCR10:INT is set at 9th SCL↓.
Set IBCR00:INTS = 1.
IBCR10:INT is set at 9th SCL↓.
Read IBSR0:LRB.
ACK is not given and IBSR0:GCA is not set.
IBCR10:INT is set at 8th SCL↓.
Read IDDR0 and control ACK/NACK by IBCR10:DACKE.
To read IBSRl:LRB, set INT S = 0.
AL is generated by second address, IBSR0:GCA is set,
and switches to slave mode.
ACK : Acknowledgment
NACK : No acknowledgment
GCA : General call address
AL : Arbitration lost

(a) General call operation in slave mode

(b) General call operation in master mode (Start from GACKE = 1 with no AL.)

(c) General call operation in master mode (Start from GACKE = 1 with AL generated by second address.)

(d) General call operation in master mode (Start from GACKE = 0 with no AL.)

(e) General call operation in master mode (Start from GACKE = 0 with AL generated by second address.)

If this module sends a general call address at the same time as another device, you can determine whether the module successfully seized control of the bus by checking whether arbitration lost was detected when the second address byte was transferred. If arbitration lost was detected, the module goes to slave mode and continues to receive data from the master.

■ Stop Condition

The master can release the bus and end communications by generating a stop condition. Changing the SDA line from "L" to "H" while SCL is "H" generates a stop condition. This signals to the other devices on the bus that the master has finished communications (referred to below as "bus free"). However, the master can continue to generate start conditions without generating a stop condition. This is called a repeated start condition.

Writing "0" to the IBCR10:MSS bit during an interrupt while in bus master mode (IBCR10:MSS = 1, IBSR0:BB = 1, IBCR10:INT = 1, and IBCR00:ALF = 0) generates a stop condition and changes to slave mode. In other cases, writing "0" to the IBCR10:MSS bit is ignored.

■ Arbitration

The interface circuit is a true multi-master bus able to connect multiple master devices. Arbitration occurs when another master within the system simultaneously transfers data during a master transfer.

Arbitration occurs on the SDA line while the SCL line is at the "H" level. When the send data is "1" and the data on the SDA line is "L" at the master, this is treated as arbitration lost. In this case, data output is halted and IBCR00:ALF is set to "1". If this occurs, an interrupt is generated if arbitration lost interrupts have been enabled (IBCR00:ALE = 1). If IBCR00:ALF is set to "1", the module sets IBCR10:MSS = 0 and IBSR0:TRX = 0, clears TRX, and goes to slave receive mode.

If IBCR00:ALF is set to "1" when IBSR0:BB = 0, IBCR00:ALF is cleared only by writing "0". If IBCR00:ALF is set to "1" when IBSR0:BB = 1, IBCR00:ALF is cleared only by clearing IBCR10:INT to "0".

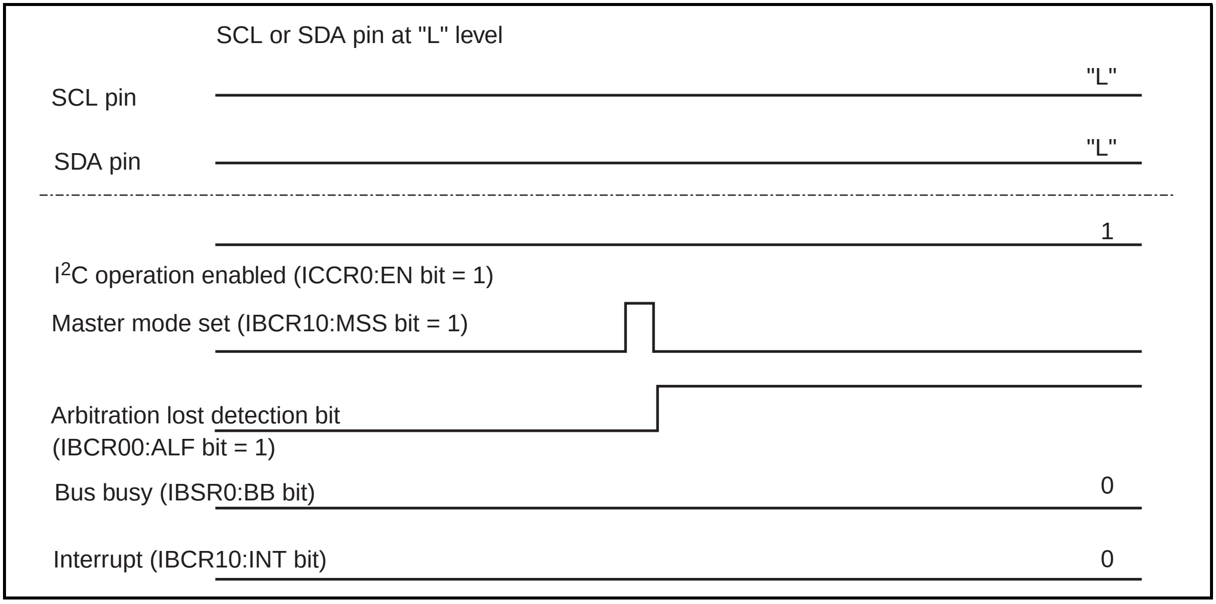
● Conditions for generating an arbitration lost interrupt when IBSR0:BB = "0"

When a start condition is generated by the program (by setting the IBCR10:MSS bit to "1") at the timing shown in Figure 24.7-3 or Figure 24.7-4, interrupt generation (IBCR10:INT bit = 1) is prohibited by arbitration lost detection (IBCR00:ALF = 1).

- Conditions (1) in which no interrupt is generated due to arbitration lost

If the program triggers a start condition (by setting the IBCR10:MSS bit to "1") when no start condition has been detected (IBSR0:BB bit = 0) and the SDA and SCL line pins are at the "L" level.

Figure 24.7-3 Timing Diagram with No Interrupt Generated with IBCR00:ALF = 1

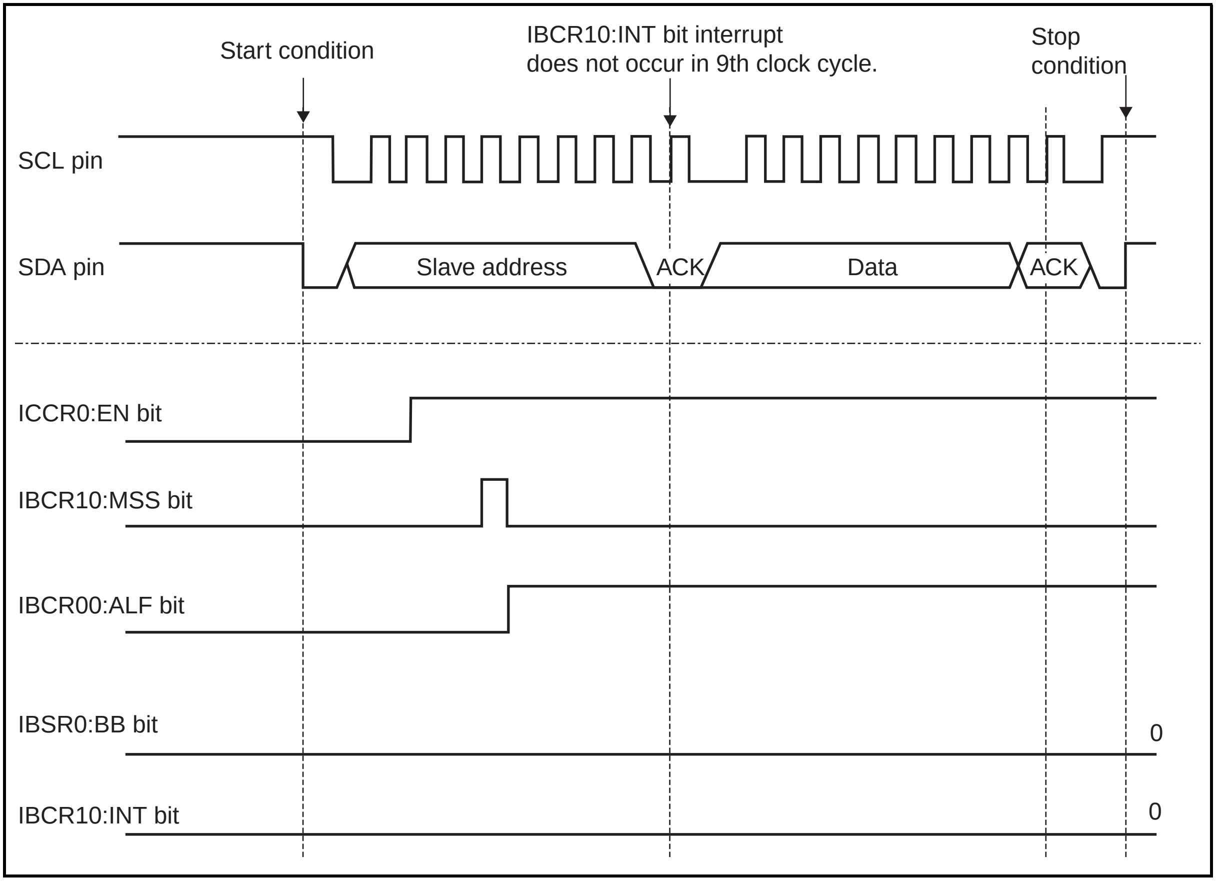

SCL or SDA pin at "L" level
SCL pin
"L"
SDA pin
"L"
1
I2C operation enabled (ICCR0:EN bit = 1)
Master mode set (IBCR10:MSS bit = 1)
Arbitration lost detection bit
(IBCR00:ALF bit = 1)
Bus busy (IBSR0:BB bit)
0
Interrupt (IBCR10:INT bit)
0

- Conditions (2) in which no interrupt is generated due to arbitration lost

If the program enables I2C operation (by setting the ICCR0:EN bit to "1") and triggers a start condition (by setting the IBCR10:MSS bit to "1") when the I2C bus is in use by another master.

This is because, as shown in Figure 24.7-4, this I2C module cannot detect the start condition (IBSR0:BB bit= 0) if another master starts communications on the I2C bus when the operation of this I2C module has been disabled (ICCR0:EN bit = 0).

Figure 24.7-4 Timing Diagram with No Interrupt Generated with IBCR0:ALF = 1



If this situation can occur, use the following procedure to set up the module from the software.

1) Trigger a start condition from the program (by setting the IBCR10:MSS bit to "1").

2) Check the IBCR00:ALF and IBSR0:BB bits in the arbitration lost interrupt.

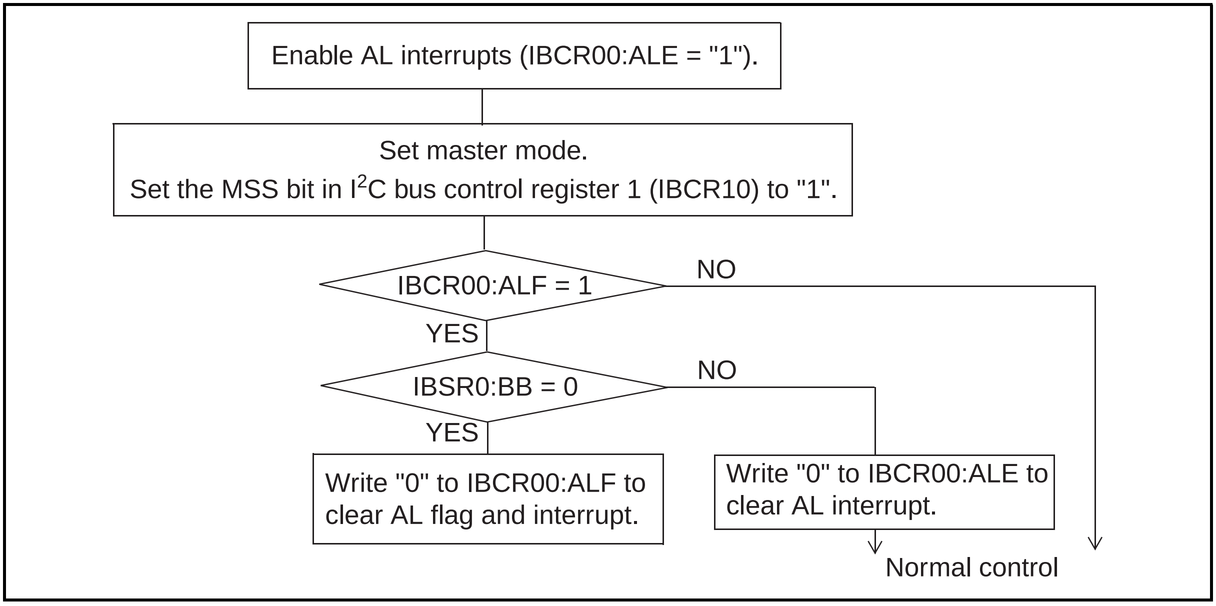
If IBCR00:ALF = 1 and IBSR0:BB = 0, clear the IBCR00:ALF bit to "0".

If IBCR00:ALF = 1 and IBSR0:BB = 1, clear the IBCR00:ALE bit to "0" and perform control as normal. (Normal control means writing "0" to the IBCR00:INT bit in the INT interrupt to clear IBCR00:ALF.)

In other cases, perform control as normal (Normal control means writing "0" to the IBCR00:INT bit in the INT interrupt to clear IBCR00:ALF.)

The following sample flow chart illustrates the procedure:

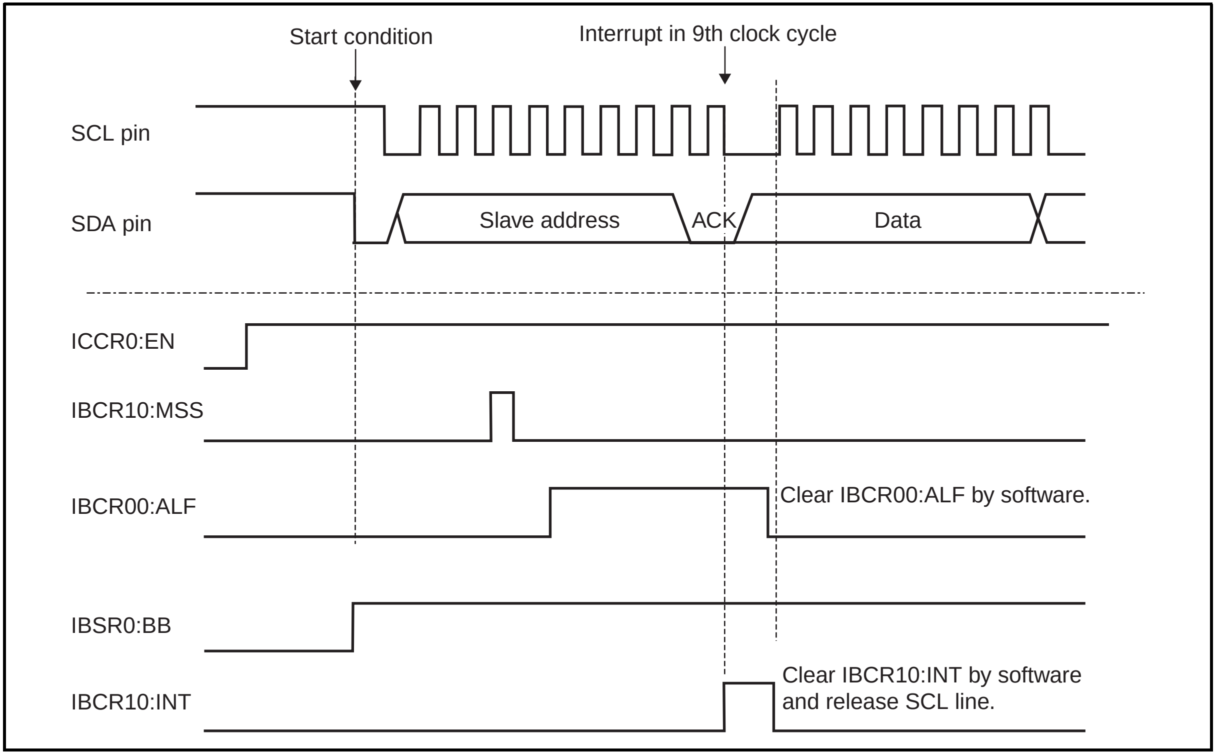
Figure 24.7-5 Sample Flow Chart of Setting


Enable AL interrupts (IBCR00:ALE = "1").
Set master mode.
Set the MSS bit in I²C bus control register 1 (IBCR10) to "1".
IBCR00:ALF = 1
NO
YES
IBSR0:BB = 0
NO
YES
Write "0" to IBCR00:ALF to clear AL flag and interrupt.
Write "0" to IBCR00:ALE to clear AL interrupt.
Normal control

● Example of generating an interrupt (IBCR10:INT = 1) with "IBCR00:ALF = 1" detected

If a start condition is generated by the program (by setting the IBCR10:MSS bit to "1") with the bus busy (IBSR0:BB = 1) and arbitration lost detected, a IBCR10:INT bit interrupt occurs upon detection of "IBCR00:ALF = 1".

Figure 24.7-6 Timing Diagram with Interrupt Generated with "IBCR00:ALF = 1" Detected


Start condition
Interrupt in 9th clock cycle
SCL pin
SDA pin
Slave address
ACK
Data
ICCR0:EN
IBCR10:MSS
IBCR00:ALF
Clear IBCR00:ALF by software.
IBSR0:BB
IBCR10:INT
Clear IBCR10:INT by software and release SCL line.

24.7.2 Function to Wake-up MCU from Standby Mode

The wakeup function enables the I^{2}C macro to be accessed while the MCU is in stop or watch mode.

■ Function to Wake Up the MCU from Standby Mode

The I^{2}C macro includes a function to wake up the MCU from standby mode. The function is enabled by writing "1" to the IBCR00:WUE bit.

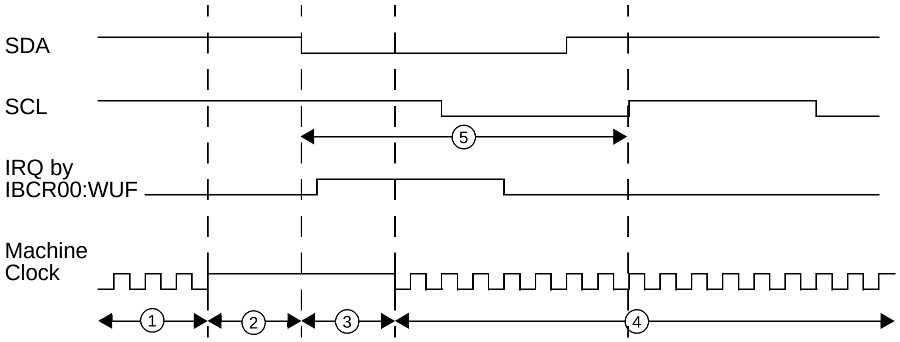
When the MCU is in stop/watch mode with the IBCR00:WUE bit containing "1", if a start condition is detected on the I^{2}C bus, the wakeup interrupt request flag bit (IBCR00:WUF) is set to "1" and the wakeup interrupt request is generated to wake up the MCU from stop/watch mode.

- Set IBCR00:WUE to "1" immediately prior to setting the MCU to stop or watch mode. Similarly, clear IBCR00:WUE (by writing "0") after the MCU wakes up from stop or watch mode so that I^{2}C operation can restart as soon as possible.
- The wakeup function only applies to the MCU stop and watch modes.

Note:

In PLL stop mode, a PLL oscillation stabilization wait time is required in addition to the oscillation stabilization wait time. This causes a very long delay between the MCU waking up and communications restarting.

Figure 24.7-7 Comparison of Normal I^{2}C Operation and Wakeup Operation



① Set the IBCR00:WUE bit to "1" immediately before entering stop/watch mode and make sure that IBSR0:BB = 0.

② Set the MCU to stop/watch mode and the machine clock stops.

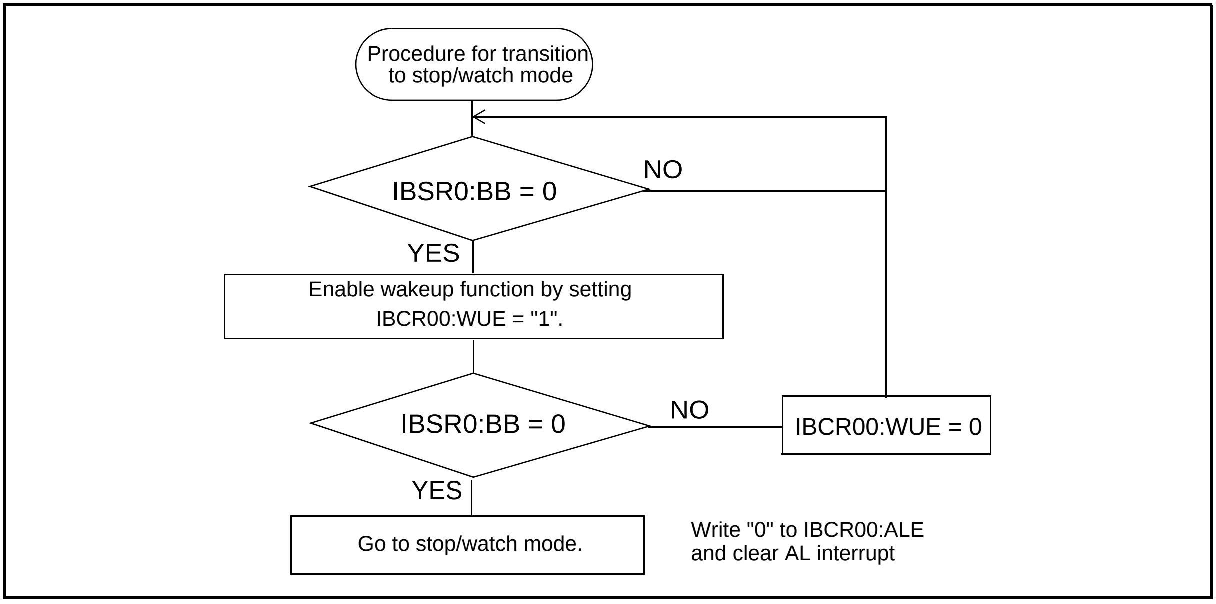
③ Detect a start condition in stop/watch mode. IBCR00:WUF is set to 1 and a wakeup IRQ is generated. After the oscillation stabilization wait time, the MCU wakes up and enters main clock mode.

④ Clear the IBCR00:WUE bit to "0" so that I^{2}C can restart the normal operation, and clear the IBCR00:WUF bit to "0" to clear the wakeup interrupt.

⑤ To receive the data byte correctly, the SCL must be released in the first cycle after 100 µs (assuming a minimum oscillation stabilization wait time of 100 µs) from the start of I^{2}C transmission (falling edge detection of SDA).

The following sample flow chart illustrates the wakeup function.

Figure 24.7-8 Sample Flow Chart of Wakeup Function


Procedure for transition to stop/watch mode
IBSR0:BB = 0
NO
YES
Enable wakeup function by setting IBCR00:WUE = "1".
IBSR0:BB = 0
NO
IBCR00:WUE = 0
YES
Go to stop/watch mode.
Write "0" to IBCR00:ALE and clear AL interrupt

24.8 Notes on Using I^{2}C Interface

This section provides notes on using the I^{2}C interface.

■ Notes on Using I^{2}C Interface

● Notes on setting I^{2}C interface registers

- Operation of the I^{2}C interface must be enabled (ICCR0:EN) before setting the I^{2}C bus control registers (IBCR00 and IBCR10).
- Setting the master/slave select bit (IBCR10:MSS) (by writing "1") starts data transfer.

● Notes on setting the shift clock frequency

- The shift clock frequency can be calculated by determining the m, n, and DMBP values using the Fsck equation in Table 24.5-4.
- "DMBP=1" may not be selected if the value of n is 4 (ICCR0:CS2 = CS1 = CS = 0).

● Notes on priority for simultaneous writes

- Contention between next byte transfer and stop condition
  When "0" is written to IBCR10:MSS with IBCR10:INT cleared, the MSS bit takes priority and a stop condition develops.
- Contention between next byte transfer and start condition
  When "1" is written to IBCR10:SCC with IBCR10:INT cleared, the SCC bit takes priority and a start condition develops.

● Notes on setup using software

- Do not select a repeated start condition (IBCR10:SCC=1) and slave mode (IBCR10:MSS=0) simultaneously.
- Execution cannot return from interrupt processing if the interrupt request enable bit is enabled (IBCR10:BEIE=1/IBCR10:INTE=1) with the interrupt request flag bit (IBCR10:BER/IBCR10:INT) containing "1". Be sure to clear the IBCR10:BER/ IBCR10:INT bit.
- The following bits are cleared to "0" when I^{2}C operation is disabled (ICCR0:EN=0):
  - AACKX, INTS, and WUE bits in the IBCR00 register
  - All the bits in the IBCR10 register except the BER and BEIE bits
  - All bits in the IBSR0 register

● Notes on data acknowledgment

In slave mode, a data acknowledgment is generated in either of the following cases:

- When the received address matches the value in the address register (IAAR0) and IBCR00:AACKX = 0.
- When a general call address (00_H) is received and IBCR10:GACKE = 1.

● Notes on selecting the transfer complete timing

- The transfer complete timing select bit (IBCR00:INTS) is valid only during data reception (IBSR10:TRX = 0 and IBSR10:FBT = 0).
- In cases other than data reception (IBSR10:TRX = 1 or IBSR10:FBT = 1), the transfer completion interrupt (IBCR10:INT) is always generated in the ninth SCL cycle.
- If the data ACK depends on the content of the received data (such as packet error checking used by the SM bus), control the data ACK by setting the data ACK enable bit (IBCR10:DACKE) after writing "1" to the IBCR00:INTS bit (for example, using a previous transfer completion interrupt) to read latest received data.
- The latest data ACK (IBSR0:LRB) can be read after the ACK has been received (IBSR0:LRB must be read during the transfer completion interrupt in the ninth SCL cycle.) If ACK is read when the IBCR0:INTS bit is "1", therefore, you must write "0" to the IBCR00:INTS bit in the transfer completion interrupt in the eighth SCL cycle so that another transfer completion interrupt will occur in the ninth SCL cycle.

● Notes on using the MCU standby mode wakeup function

- Set IBCR00:WUE to "1" immediately prior to setting the MCU to stop or watch mode. Similarly, clear IBCR00:WUE (by writing "0") after the MCU wakes up from stop or watch mode so that I^{2}C operation can restart as soon as possible.
- When a wakeup interrupt request occurs, the MCU wakes up after the oscillation stabilization wait time elapses. To prevent the data loss immediately after wakeup, design the system so that the SCL rises as the first cycle and the first bit must be transmitted as data after 100 µs (assuming a minimum oscillation stabilization wait time of 100 µs) from the wakeup due to start of I^{2}C transmission (upon detection of the falling edge of SDA).
- During a MCU standby mode, the status flags, state machine, and I^{2}C bus outputs for the I^{2}C function retain the states they had prior to entering the standby mode. To prevent a hang-up of the entire I^{2}C bus system, make sure that IBSR0:BB = 0 before entering standby mode.
- The wakeup function does not support the transition of the MCU to stop or watch mode with IBSR0:BB = 1. If the MCU enters stop or watch mode with IBSR0:BB = 1, a bus error will occur upon detection of a start condition.
- In PLL stop mode, for example, the time from wakeup to the start of communication becomes longer than in stop/watch mode by the PLL oscillation stabilization wait time as the PLL oscillation stabilization wait time is required in addition to the oscillation stabilization wait time.
- To ensure correct operation of the I^{2}C interface, always clear IBCR00:WUE to "0" after the MCU wakes up from stop or watch mode, regardless of whether this occurs due to the I^{2}C wakeup function or the wakeup function for some other resource (such as an external interrupt).

24.9 Sample Settings for I^2C

This section provides sample settings for the I^2C interface.

■ Sample Settings

- Enabling/disabling I^2C operation

Use the I^2C operation enable bit (ICCR0:EN).

Operation	I^2C operation enable bit (EN)
To disable I^2C operation	Set the bit to "0".
To enable I^2C operation	Set the bit to "1".

- Selecting the I^2C master or slave mode

Use the master/slave select bit (IBCR10:MSS).

Operation	Master/slave select bit (MSS)
To select master mode	Set the bit to "1".
To select slave mode	Set the bit to "0".

- Selecting the shift clock

Use the clock select bits (ICCR0:CS4/CS3/CS2/CS1/CS0).

- Bypassing the m divider when the shift clock frequency is generated

Use the divider-m bypass bit (ICCR0:DMBP).

Operation	Divider m bypass bit (DMBP)
To bypass divider m	Set the bit to "1".

- Controlling I^2C address acknowledgment

Use the address acknowledge disable bit (IBCR00:AACKX).

Operation	Address acknowledge disable bit (AACKX)
To enable address acknowledge output	Set the bit to "0".
To disable address acknowledge output	Set the bit to "1".

- Controlling I^{2}C data acknowledgment

Use the data acknowledge enable bit (IBCR10:DACKE).

Operation	Data acknowledge enable bit (DACKE)
To enable data acknowledge output	Set the bit to "1".
To disable data acknowledge output	Set the bit to "0".

- Controlling I^{2}C general call address acknowledgment

Use the general call address acknowledge enable bit (IBCR10:GACKE).

Operation	General call address acknowledge enable bit (GACKE)
To enable general call address acknowledge output	Set the bit to "1".
To disable general call address acknowledge output	Set the bit to "0".

- Restarting I^{2}C communication

Use the start condition generation bit (IBCR10:SCC).

Operation	Start condition generation bit (SCC)
To restart communication	Set the bit to "1".

- Selecting the I^{2}C data reception transfer completion flag (INT)

Use the timing select bit (IBCR00:INTS) for the data reception transfer completion flag (INT).

Operation	Timing select bit (INTS) for data reception transfer completion flag (INT)
To cause a transfer interrupt in the 9th SCL cycle	Set the bit to "0".
To cause a transfer interrupt in the 8th SCL cycle	Set the bit to "1".

- Interrupt related register

To set the interrupt level, use the following interrupt level setting register.

Interrupt source	Interrupt level setting register	Interrupt vector
ch. 0	Interrupt level setting register (ILR4) Address: 0007D$_H$	#16 Address: 0FFDA$_H$

- Enabling, disabling, and clearing interrupts

Interrupt request enable flag and interrupt request flag

- Transfer interrupt

  (Data transfer completion interrupt)

  To enable interrupts, use the transfer completion interrupt enable bit (IBCR10:INTE).

Operation	Transfer completion interrupt enable bit (INTE)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

To clear interrupt requests, use the transfer completion interrupt request flag bit (IBCR10:INT).

Operation	Transfer completion interrupt request flag bit (INT)
To clear an interrupt request	Set the bit to "0".

(Bus error generation interrupt)

To enable interrupts, use the bus error interrupt request enable bit (IBCR10:BEIE).

Operation	Bus error interrupt request enable bit (BEIE)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

To clear interrupt requests, use the bus error interrupt request flag bit (IBCR10:BER).

Operation	Bus error interrupt request flag bit (BER)
To clear an interrupt request	Set the bit to "0".

- Stop interrupt

  (Stop condition detection interrupt)

  To enable interrupts, use the STOP detection interrupt enable bit (IBCR00:SPE).

Operation	STOP detection interrupt enable bit (SPE)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

To clear interrupt requests, use the STOP detection interrupt request flag bit (IBCR00:SPF).

Operation	STOP detection interrupt request flag bit (SPF)
To clear an interrupt request	Set the bit to "0".

(Arbitration lost detection interrupt)

To enable interrupts, use the arbitration lost interrupt enable bit (IBCR00:ALE).

Operation	Arbitration lost interrupt enable bit (ALE)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

To clear interrupt requests, use the arbitration lost interrupt request flag bit (IBCR00:ALF).

Operation	Arbitration lost interrupt request flag bit (ALF)
To clear an interrupt request	Write "0" to the flag.

(Start condition detection interrupt)

To enable interrupts, use the MCU standby-mode wakeup function enable bit (IBCR00:WUE).

Operation	MCU standby-mode wakeup function enable bit (WUE)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

To clear interrupt requests, use the MCU standby-mode wakeup interrupt request flag bit (IBCR00:WUF).

Operation	MCU standby-mode wakeup interrupt request flag bit (WUF)
To clear an interrupt request	Set the bit to "0".

CHAPTER 25

8/10-BIT A/D CONVERTER

This chapter describes the functions and operations of the 8/10-bit A/D converter.

25.1 Overview of 8/10-bit A/D Converter

The 8/10-bit A/D converter is a 10-bit successive approximation type of 8/10-bit A/D converter. It can be started by the software and internal clock, with one input signal selected from multiple analog input pins.

■ A/D Conversion Function

The A/D converter converts analog voltage (input voltage) input through an analog input pin to an 8-bit or 10-bit digital value.

- The input signal can be selected from multiple analog input pins.
- The conversion speed can be set in a program. (can be selected according to operating voltage and frequency).
- An interrupt is generated when A/D conversion is completed.
- The completion of conversion can be determined according to the ADI bit in the ADC1 register.

To activate the A/D conversion function, use one of the following methods.

- Activation using the AD bit in the ADC1 register
- Continuous activation using the external pin (ADTG)
- Continuous activation using the 8/16-bit composite timer output TO00

25.2 Configuration of 8/10-bit A/D Converter

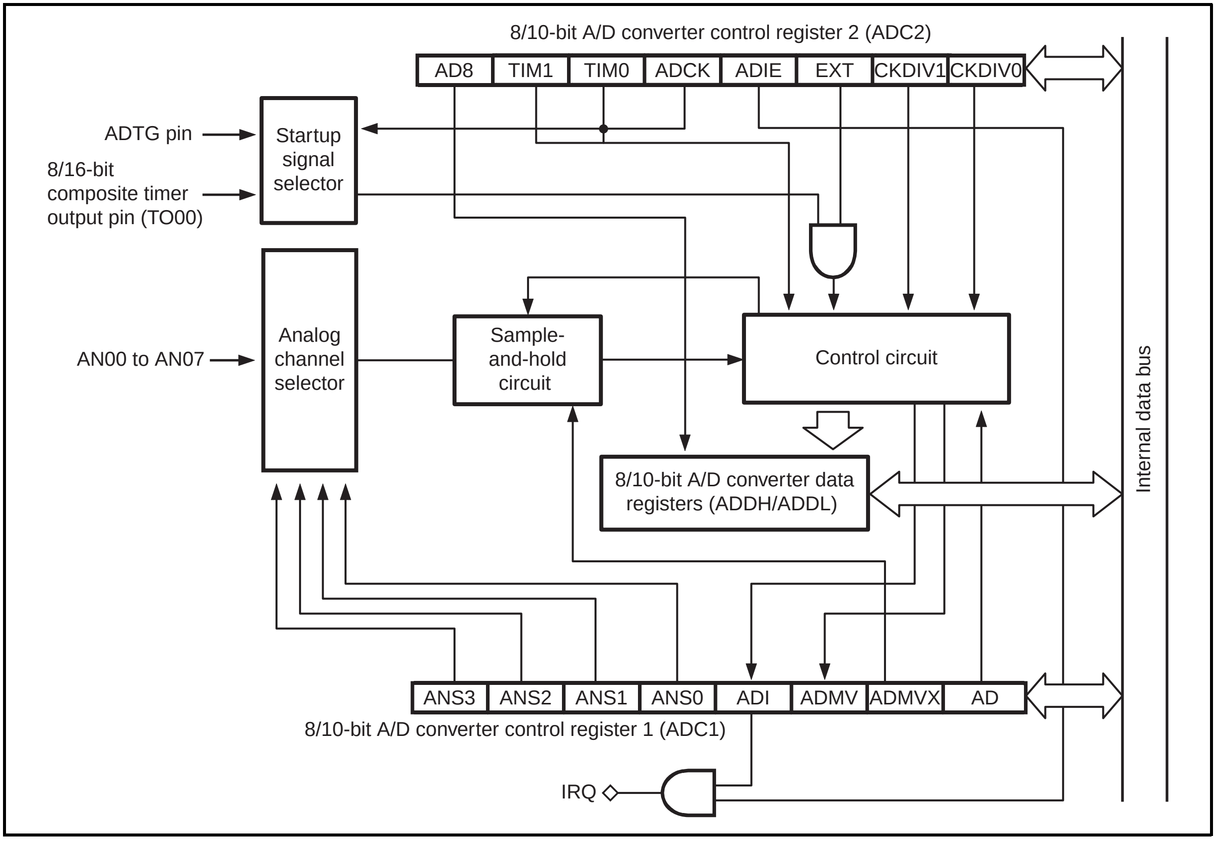
The 8/10-bit A/D converter consists of the following blocks:

- **Clock selector (input clock selector for starting A/D conversion)**
- **Analog channel selector**
- **Sample-and-hold circuit**
- **Control circuit**
- **8/10-bit A/D converter data registers (ADDH/ADDL)**
- **8/10-bit A/D converter control register 1 (ADC1)**
- **8/10-bit A/D converter control register 2 (ADC2)**

■ Block Diagram of 8/10-bit A/D Converter

Figure 25.2-1 is the block diagram of the 8/10-bit A/D converter.

Figure 25.2-1 Block Diagram of 8/10-bit A/D Converter



● Clock selector

This selects the A/D conversion clock with continuous activation having been enabled (ADC2:EXT = 1).

● Analog channel selector

This is the circuit selecting an input channel from several analog input pins.

● Sample-and-hold circuit

This circuit holds input voltage selected by the analog channel selector. By sampling the input voltage and holding it immediately after A/D conversion starts, this circuit prevents A/D conversion from being affected by the fluctuation in input voltage during the conversion (comparison).

● Control circuit

The A/D conversion function determines the values in the 10-bit A/D data register sequentially from MSB to LSB based on the voltage compare signal from the comparator. When A/D conversion is completed, the A/D conversion function sets the interrupt request flag bit (ADC1: ADI) to "1".

● 8/10-bit A/D converter data registers (ADDH/ADDL)

The upper two bits of 10-bit A/D data are stored in the ADDH register; the lower eight bits in the ADDL register.

If the A/D conversion precision bit (ADC2:AD8) is set to "1", the A/D conversion precision becomes 8-bit precision, and the upper eight bits of 10-bit A/D data are to be stored in the ADDL register.

● 8/10-bit A/D converter control register 1 (ADC1)

This register is used to enable and disable different functions, select an analog input pin, and check the status of the A/D converter.

● 8/10-bit A/D converter control register 2 (ADC2)

This register is used to select an input clock, enable and disable interrupts and control different A/D conversion functions.

■ Input Clock

The 8/10-bit A/D converter uses an output clock from the prescaler as the input clock (operating clock).

25.3 Pins of 8/10-bit A/D Converter

This section describes the pins of the 8/10-bit A/D converter.

■ Pins of 8/10-bit A/D Converter

The MB95410H/470H Series has 8 channels of analog input pin.

The analog input pins are also used as general-purpose I/O ports.

● AN07 pin to AN00 pin

AN07 to AN00: When using the A/D conversion function, input to one of these pins the analog voltage to be converted. A pin of AN07 to AN00 functions as an analog input pin if the bit in the port direction register (DDR) corresponding to that pin is set to "0" and the analog input pin select bits (ADC1:ANS0 to ANS3) are set to the values representing that pin. A pin not used as an analog input pin can be used as a general-purpose I/O port also when the 8/10-bit A/D converter is used.

● ADTG pin

ADTG: This is a pin used to activate the A/D conversion function with an external trigger. Before using the ADTG pin for activating the A/D conversion with an external trigger, set the pin as an input port using the corresponding port direction register (DDR).

● AV$_{CC}$ pin

AV$_{CC}$: This is an 8/10-bit A/D converter power supply pin. Use this at the same potential as V$_{CC}$. If A/D conversion precision is required, ensure that V$_{CC}$ noise does not enter AV$_{CC}$, or use a separate power source. Connect this pin to a power source even when the 8/10-bit A/D converter is not in use.

● AV$_{SS}$ pin

AV$_{SS}$: This is a ground pin of the 8/10-bit A/D converter. Use this at the same potential as V$_{SS}$. If A/D conversion precision is required, ensure that V$_{SS}$ noise does not enter AV$_{SS}$. Connect this pin to a ground (GND) even when the 8/10-bit A/D converter is not in use.

■ Block Diagram of Pins of 8/10-bit A/D Converter

Figure 25.3-1 Block Diagram of AN01 and AN04 of 8/10-bit A/D Converter

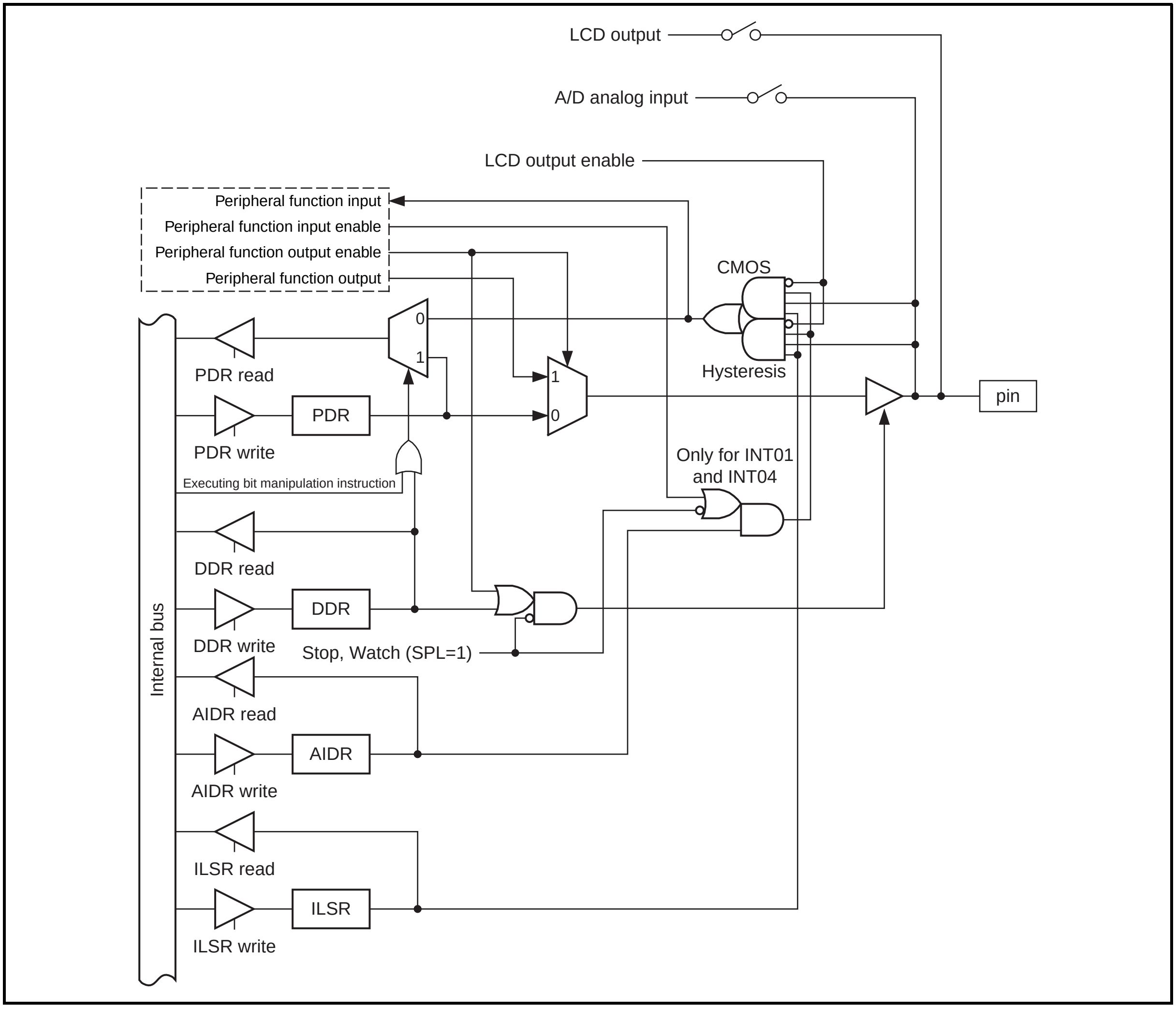
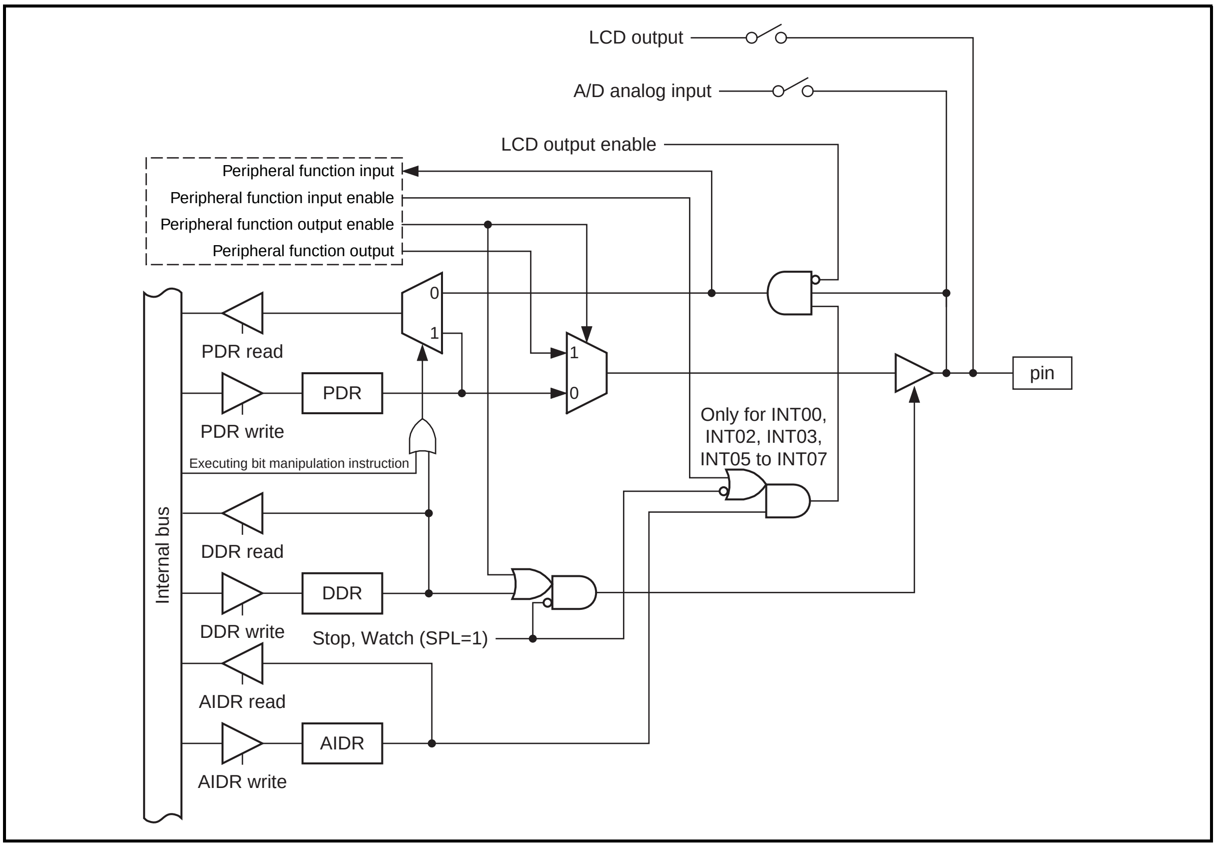


Figure 25.3-2 Block Diagram of AN00, AN02, AN03, AN05, AN06 and AN07 of 8/10-bit A/D Converter


LCD output
A/D analog input
LCD output enable
Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
Only for INT00, INT02, INT03, INT05 to INT07
pin
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
AIDR read
AIDR
AIDR write
Internal bus

25.4 Registers of 8/10-bit A/D Converter

The 8/10-bit A/D converter has four registers: A/D converter control register 1 (ADC1), A/D converter control register 2 (ADC2), A/D converter data register upper (ADDH) and A/D converter data register lower (ADDL).

■ Registers of 8/10-bit A/D Converter

Figure 25.4-1 lists the registers of the 8/10-bit A/D converter.

Figure 25.4-1 Registers of 8/10-bit A/D Converter.

8/10-bit A/D converter control register 1 (ADC1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
006C$_H$	ANS3	ANS2	ANS1	ANS0	ADI	ADMV	ADMVX	AD	00000000$_B$
	R/W	R/W	R/W	R/W	R(RM1),W	R/WX	R/W	R0,W	

8/10-bit A/D converter control register 2 (ADC2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
006D$_H$	AD8	TIM1	TIM0	ADCK	ADIE	EXT	CKDIV1	CKDIV0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

8/10-bit A/D converter data register upper (ADDH)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
006E$_H$	-	-	-	-	-	-	SAR9	SAR8	00000000$_B$
	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R/WX	R/WX	

8/10-bit A/D converter data register lower (ADDL)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
006F$_H$	SAR7	SAR6	SAR5	SAR4	SAR3	SAR2	SAR1	SAR0	00000000$_B$
	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

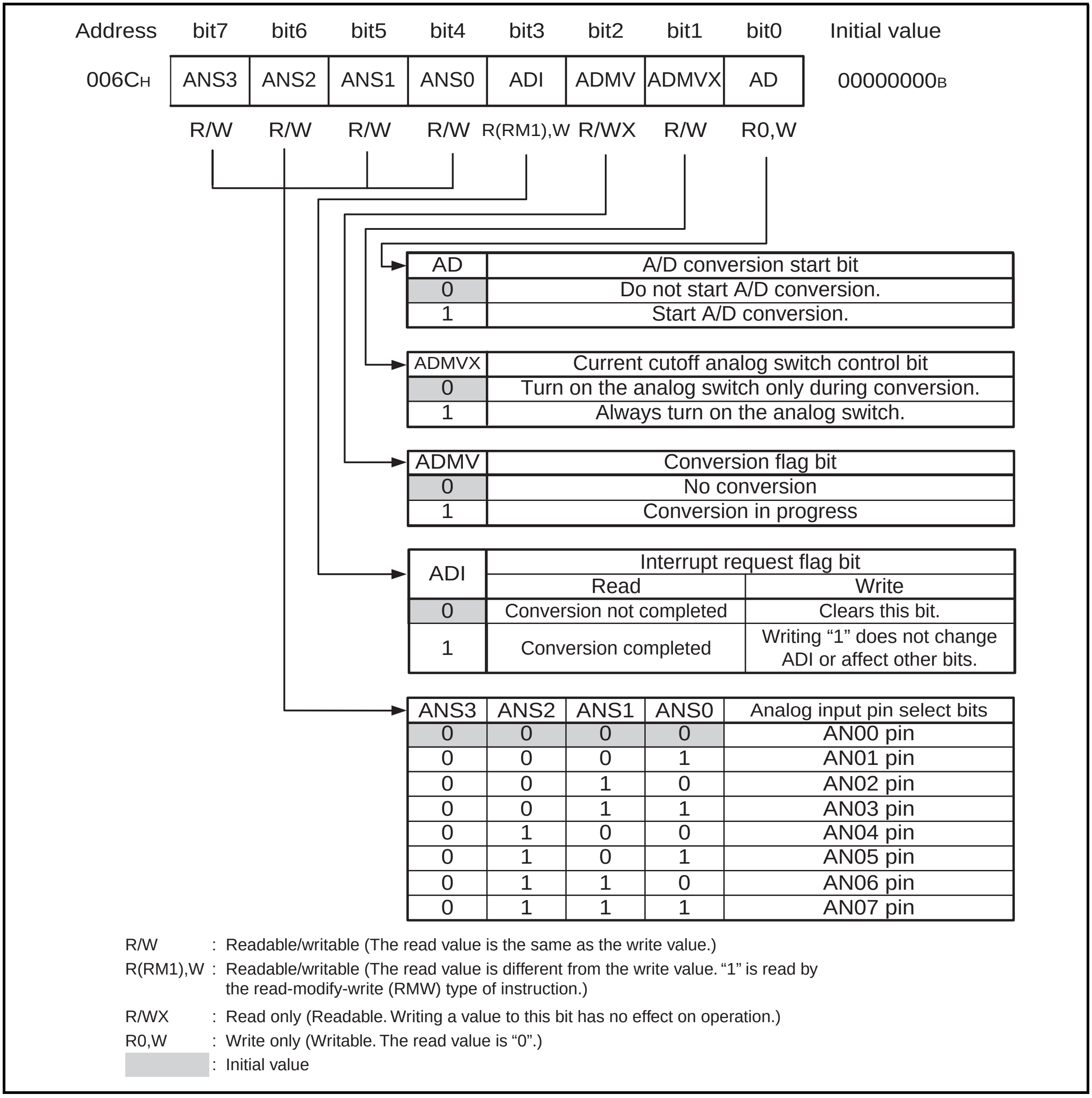
R/W : Readable/writable (The read value is the same as the write value.)
R(RM1), W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)
R0,W : Write only (Writable. The read value is "0".)
R0/WX : The read value is "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

25.4.1 8/10-bit A/D Converter Control Register 1 (ADC1)

The 8/10-bit A/D converter control register 1 (ADC1) is used to enable and disable individual functions of the 8/10-bit A/D converter, select an analog input pin and check the status of the converter.

■ 8/10-bit A/D Converter Control Register 1 (ADC1)

Figure 25.4-2 8/10-bit A/D Converter Control Register 1 (ADC1)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$006C_H$	ANS3	ANS2	ANS1	ANS0	ADI	ADMV	ADMVX	AD	00000000_B
	R/W	R/W	R/W	R/W	R(RM1),W	R/WX	R/W	R0,W	

AD	A/D conversion start bit
0	Do not start A/D conversion.
1	Start A/D conversion.

ADMVX	Current cutoff analog switch control bit
0	Turn on the analog switch only during conversion.
1	Always turn on the analog switch.

ADMV	Conversion flag bit
0	No conversion
1	Conversion in progress

ADI	Interrupt request flag bit	
	Read	Write
0	Conversion not completed	Clears this bit.
1	Conversion completed	Writing “1” does not change ADI or affect other bits.

ANS3	ANS2	ANS1	ANS0	Analog input pin select bits
0	0	0	0	AN00 pin
0	0	0	1	AN01 pin
0	0	1	0	AN02 pin
0	0	1	1	AN03 pin
0	1	0	0	AN04 pin
0	1	0	1	AN05 pin
0	1	1	0	AN06 pin
0	1	1	1	AN07 pin

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)
R0,W : Write only (Writable. The read value is “0”.)
▨ : Initial value

Table 25.4-1 Functions of Bits in 8/10-bit A/D Converter Control Register 1 (ADC1)

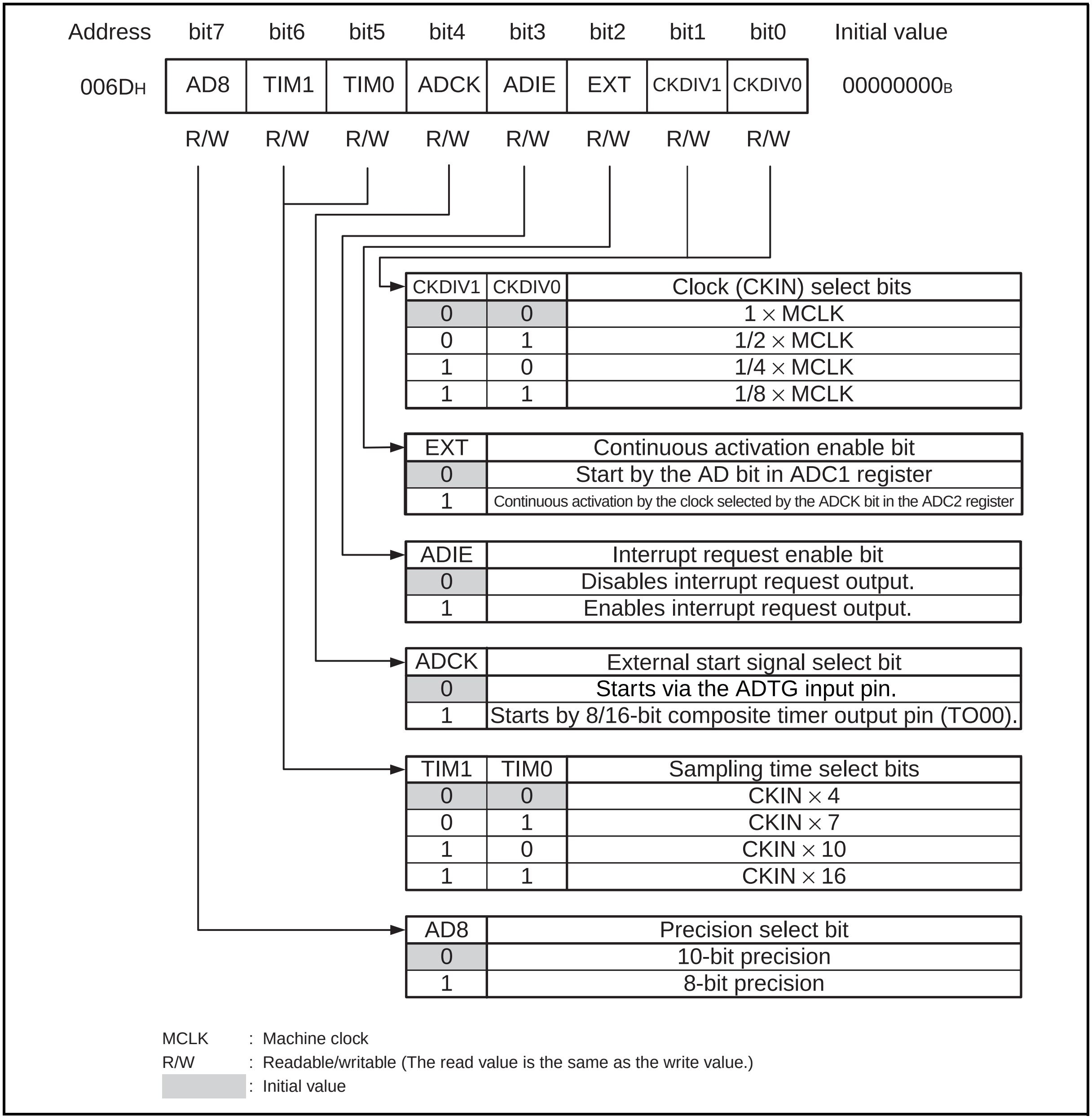
Bit name		Function
bit7 to bit4	ANS3, ANS2, ANS1, ANS0: Analog input pin select bits	These bits select an analog input pin to be used from AN00 to AN07. When A/D conversion is started (AD = 1) by the software (ADC2:EXT = 0), these bits can be modified simultaneously. Note: When the ADMV bit is "1", do not modify these bits. Pins not used as analog input pins can be used as general-purpose ports.
bit3	ADI: Interrupt request flag bit	This bit detects the completion of A/D conversion. • When the A/D conversion function is used, the bit is set to "1" immediately after A/D conversion is complete. • Interrupt requests are output when this bit and the interrupt request enable bit (ADC2:ADIE) are both set to "1". • When "0" is written to this bit, it is cleared. Writing "1" to this bit does not change it or affect other bits. • When read by the read-modify-write (RMW) type of instruction, this bit returns "1".
bit2	ADMV: Conversion flag bit	This bit indicates that A/D conversion is in progress. The bit is set to "1" during A/D conversion. This bit is read-only. A value written to this bit is meaningless and has no effect on operation.
bit1	ADMVX: Current cutoff analog switch control bit	This bit controls the analog switch for cutting off the internal reference power supply. Since rush current flows immediately after A/D conversion starts, when the external impedance of Vcc pin is high, A/D conversion precision may be affected. This can be avoided by setting this bit to "1" before A/D conversion starts. In addition, in order to reduce current consumption, set the bit to "0" before transiting to standby mode.
bit0	AD: A/D conversion start bit	This bit activates A/D conversion function with the software. Writing "1" to the bit activates the A/D conversion function. When EXT = 1, starting the A/D conversion with this bit is disabled. With EXT = 0, when "1" is written to this bit while A/D conversion is in progress, A/D conversion restarts. Note: Writing "0" to this bit cannot stop the operation of the A/D conversion function. The read value of this bit is always "0".

25.4.2 8/10-bit A/D Converter Control Register 2 (ADC2)

The 8/10-bit A/D converter control register 2 (ADC2) is used to control different functions of the 8/10-bit A/D converter, select the input clock, enable and disable interrupts.

■ 8/10-bit A/D Converter Control Register 2 (ADC2)

Figure 25.4-3 8/10-bit A/D Converter Control Register 2 (ADC2)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
006DH	AD8	TIM1	TIM0	ADCK	ADIE	EXT	CKDIV1	CKDIV0	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

CKDIV1	CKDIV0	Clock (CKIN) select bits
0	0	1 × MCLK
0	1	1/2 × MCLK
1	0	1/4 × MCLK
1	1	1/8 × MCLK

EXT	Continuous activation enable bit
0	Start by the AD bit in ADC1 register
1	Continuous activation by the clock selected by the ADCK bit in the ADC2 register

ADIE	Interrupt request enable bit
0	Disables interrupt request output.
1	Enables interrupt request output.

ADCK	External start signal select bit
0	Starts via the ADTG input pin.
1	Starts by 8/16-bit composite timer output pin (TO00).

TIM1	TIM0	Sampling time select bits
0	0	CKIN × 4
0	1	CKIN × 7
1	0	CKIN × 10
1	1	CKIN × 16

AD8	Precision select bit
0	10-bit precision
1	8-bit precision

MCLK : Machine clock
R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value

Table 25.4-2 Functions of Bits in 8/10-bit A/D Converter Control Register 2 (ADC2)

Bit name		Function
bit7	AD8: Precision select bit	This bit selects the resolution of A/D conversion. **Writing "0":** Selects 10-bit precision. **Writing "1":** Selects 8-bit precision. Reading the ADDL register can obtain 8-bit data. Note: The data bits to be used are different depending on the resolution selected. Modify this bit only when the A/D converter has stopped operating.
bit6, bit5	TIM1, TIM0: Sampling time select bits	These bits set the sampling time. • Modify the sampling time according to operating conditions (voltage and frequency). • The CKIN value is determined by the clock select bits (ADC2:CKDIV1, CKDIV0). Note: Modify these bits only when the A/D converter has stopped operating.
bit4	ADCK: External start signal select bit	This bit selects the start signal for external start (ADC2:EXT = 1).
bit3	ADIE: Interrupt request enable bit	This bit enables or disables outputting interrupts to the interrupt controller. • Interrupt requests are output when both this bit and the interrupt request flag bit (ADC1: ADI) have been set to "1".
bit2	EXT: Continuous activation enable bit	This bit selects whether to activate the A/D conversion function with the software, or to continuously activate the A/D conversion function whenever a rising edge of the input clock is detected.
bit1, bit0	CKDIV1, CKDIV0: Clock select bits	These bits select the clock to be used for A/D conversion. The input clock is generated by the prescaler. See "CHAPTER 6 CLOCK CONTROLLER" for details. • The sampling time varies according to the clock selected by these bits. • Modify these bits according to operating conditions (voltage and frequency). Note: Modify these bits only when the A/D converter has stopped operating.

25.4.3 8/10-bit A/D Converter Data Registers Upper/ Lower (ADDH/ADDL)

The 8/10-bit A/D converter data registers upper/lower (ADDH, ADDL) store the results of 10-bit A/D conversion during 10-bit A/D conversion.
The upper two bits of 10-bit data are stored in the ADDH register and the lower eight bits the ADDL register.

■ 8/10-bit A/D Converter Data Registers Upper/Lower (ADDH, ADDL)

Figure 25.4-4 8/10-bit A/D Converter Data Registers Upper/Lower (ADDH, ADDL)

ADDH	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address	-	-	-	-	-	-	SAR9	SAR8	00000000_B
$006E_H$	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R0/WX	R/WX	R/WX	

ADDL	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address	SAR7	SAR6	SAR5	SAR4	SAR3	SAR2	SAR1	SAR0	00000000_B
$006F_H$	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

R/WX	: Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX	: The read value is "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

The upper two bits of 10-bit A/D data correspond to bit1 and bit0 in the ADDH register and the lower eight bits bit7 to bit0 in the ADDL register.

If the AD8 bit in ADC2 register is set to "1", 8-bit precision is selected. Reading the ADDL register can obtain 8-bit data.

These two registers are read-only registers. Writing data to them has no effect on operation.

In A/D conversion in which 8-bit precision is selected, SAR8 and SAR9 in the ADDH register become "0".

● A/D conversion function

When A/D conversion is started, the results of conversion are finalized and stored in the ADDH and ADDL registers after the conversion time according to the register settings elapses. After A/D conversion is completed and before the next A/D conversion is completed, read A/D data registers (conversion results), and clear the interrupt request flag bit (ADI) in the ADC1 register. During A/D conversion, the values of the ADDH and ADDL registers are results of the last A/D conversion.

25.5 Interrupts of 8/10-bit A/D Converter

The completion of conversion during the operation of the A/D converter is an interrupt source of the 8/10-bit A/D converter.

■ Interrupts During 8/10-bit A/D Converter Operation

When A/D conversion is completed, the interrupt request flag bit (ADC1:ADI) is set to "1". Then if the interrupt request enable bit has been enabled (ADC2:ADIE = 1), an interrupt request is made to the interrupt controller. Write "0" to the ADI bit using the interrupt service routine to clear the interrupt request.

The ADI bit is set to "1" when A/D conversion is completed, irrespective of the value of the ADIE bit.

The CPU cannot return from interrupt processing if the interrupt request flag bit (ADC1: ADI) is "1" with interrupt requests having been enabled (ADC2:ADIE = 1). Always clear the ADI bit in the interrupt service routine.

■ Register and Vector Table Addresses Related to 8/10-bit A/D Converter Interrupts

Table 25.5-1 Register and Vector Table Addresses Related to 8/10-bit A/D Converter Interrupts

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
8/10-bit A/D converter	IRQ18	ILR4	L18	FFD6$_H$	FFD7$_H$

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

25.6 Operations of 8/10-bit A/D Converter and Setting Procedure Example

The 8/10-bit A/D converter can activate A/D conversion with the software or activate A/D conversion continuously according to the setting of the EXT bit in the ADC2 register.

■ Operations of 8/10-bit A/D Converter Conversion Function

● Software activation

To activate the A/D conversion function with the software, do the settings shown in Figure 25.6-1.

Figure 25.6-1 Settings for A/D Conversion Function (Software Activation)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ADC1	ANS3	ANS2	ANS1	ANS0	ADI	ADMV	ADMVX	AD
	◎	◎	◎	◎	◎	◎	◎	1
ADC2	AD8	TIM1	TIM0	ADCK	ADIE	EXT	CKDIV1	CKDIV0
	◎	◎	◎	×	◎	0	◎	◎
ADDH	-	-	-	-	-	-	A/D converted value retained	
ADDL	A/D converted value retained							

◎: Bit to be used
× : Unused bit
1 : Set to "1"
0 : Set to "0"

When the A/D conversion function is activated, A/D conversion starts. In addition, the A/D conversion function can be re-activated even during conversion.

● Continuous activation

To execute continuous activation of the A/D conversion function, do the settings shown in Figure 25.6-2.

Figure 25.6-2 Settings for A/D Conversion Function (Continuous Activation)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ADC1	ANS3	ANS2	ANS1	ANS0	ADI	ADMV	ADMVX	AD
	◎	◎	◎	◎	◎	◎	◎	×
ADC2	AD8	TIM1	TIM0	ADCK	ADIE	EXT	CKDIV1	CKDIV0
	◎	◎	◎	◎	◎	1	◎	◎
ADDH	-	-	-	-	-	-	A/D converted value retained	
ADDL	A/D converted value retained							

◎: Bit to be used
× : Unused bit
1 : Set to "1"

When continuous activation is enabled, the A/D conversion function is activated at the rising edge of the input clock selected to start A/D conversion. Continuous activation is stopped when disabled (ADC2:EXT = 0).

■ Operations of A/D Conversion Function

This section explains the operations of 8/10-bit A/D converter.

1) When A/D conversion is started, the conversion flag bit is set (ADC1:ADMV = 1) and the selected analog input pin is connected to the sample-and-hold circuit.
2) The voltage in the analog input pin is loaded into a sample-and-hold capacitor in the sample-and-hold circuit during the sampling cycle. This voltage is held until A/D conversion is completed.
3) The comparator in the control circuit compares the voltage loaded into sample-and-hold capacitor with the A/D conversion reference voltage, from the most significant bit (MSB) to the least significant bit (LSB), and then transfers the results to the ADDH and ADDL registers.
   After the results have been transferred to the two registers, the conversion flag bit is cleared (ADC1:ADMV = 0) and the interrupt request flag bit is set to "1" (ADC1:ADI = 1).

Notes:

- The contents of the ADDH and ADDL registers are retained until the end of A/D conversion. Therefore, during A/D conversion, the values resulting from last conversion will be returned if the two registers are read.
- Do not change the analog input pin (ADC1:ANS3 to ANS0) while AD conversion function is being used. During continuous activation in particular, disable continuous activation (ADC2:EXT = 0) before changing the analog input pin.
- The start of the reset mode, the stop mode or the watch mode causes the A/D converter to stop and the ADMV bit to be cleared to "0".

■ Setting Procedure Example

Below is an example of procedure for setting the 8/10-bit A/D converter:

● Initial settings

1) Set the input port (DDR0).
2) Set the interrupt level (ILR4).
3) Enable A/D input (ADC1:ANS0 to ANS3).
4) Set the sampling time (ADC2:TIM1, TIM0).
5) Select the clock (ADC2:CKDIV1, CKDIV0).
6) Set A/D conversion precision (ADC2:AD8).
7) Select the operating mode (ADC2:EXT).
8) Select the start trigger (ADC2:ADCK).
9) Enable interrupts (ADC2:ADIE = 1).
10)Activate the A/D conversion function (ADC1:AD = 1).

● Interrupt processing

1) Clear the interrupt request flag (ADC1:ADI = 0).
2) Read converted values (ADDH, ADDL).
3) Activate the A/D conversion function (ADC1:AD = 1).

25.7 Notes on Using 8/10-bit A/D Converter

This section provides notes on using the 8/10-bit A/D converter.

■ Notes on Using 8/10-bit A/D Converter

● Notes on setting the 8/10-bit A/D converter with a program

- The contents of the ADDH and ADDL registers are retained until the end of A/D conversion. Therefore, during A/D conversion, the values resulting from last conversion will be returned if the two registers are read.
- Do not change the analog input pin (ADC1:ANS3 to ANS0) while AD conversion function is being used. During continuous activation in particular, disable continuous activation (ADC2:EXT = 0) before changing the analog input pin.
- A reset, or the start of the stop mode or watch mode causes the A/D converter to stop and the ADMV bit to be cleared to "0".
- The CPU cannot return from interrupt processing if the interrupt request flag bit (ADC1:ADI) is "1" with interrupt requests having been enabled (ADC2:ADIE = 1). Always clear the ADI bit in the interrupt service routine.

● Note on interrupt requests

If the restart of A/D conversion (ADC1:AD = 1) and the completion of A/D conversion occur simultaneously, the interrupt request flag bit (ADC1:ADI) is set.

● A/D conversion error

As | Vcc - Vss | decreases, the A/D conversion error increases proportionately.

● 8/10-bit A/D converter analog input sequences

Apply the analog input (AN00 to AN07) and the digital power supply (V_{CC}) simultaneously, or apply the analog input after applying the digital power supply.

Disconnect the digital power supply (V_{CC}) at the same time as the analog input (AN00 to AN07), or after disconnecting analog input (AN00 to AN07).

Ensure that the analog input voltage does not exceed the voltage of digital power supply when turning on or off the power of the 8/10-bit A/D converter.

● Conversion time

The conversion speed of A/D conversion function is affected by clock mode, main clock oscillation frequency and main clock speed switching (gear function).

Example: Sampling time = CKIN × (ADC2:TIM1/TIM0 setting)

Compare time = CKIN × 10 (fixed value) + MCLK

A/D converter startup time: minimum = MCLK + MCLK

maximum = MCLK + CKIN

Conversion time = A/D converter startup time + sampling time + compare time

- The conversion time may have an error of up to (1 CKIN – 1 MCLK), depending on the time at which A/D conversion starts.

- When setting the A/D converter in software, ensure that the settings satisfy the specifications of "sampling time" and "compare time" of the A/D converter mentioned in the data sheet of the MB95410H/470H Series.

25.8 Sample Settings for 8/10-bit A/D Converter

This section provides sample settings for the 8/10-bit A/D converter.

■ Sample Settings

● Method of selecting an operating clock for the 8/10-bit A/D converter

Use the clock select bits (ADC2:CKDIV1, CKDIV0) to select an operating clock.

● Method of selecting the sampling time of the 8/10-bit A/D converter

Use the sampling time select bits (ADC2:TIM1, TIM0) to select sampling time.

● Method of controlling the analog switch for cutting off the internal reference power supply of the 8/10-bit A/D converter

Use the current cutoff analog switch control bit (ADC1:ADMVX) to control the analog switch for cutting off internal reference power supply.

Operation	Current cutoff analog switch control bit (ADMVX)
To switch off internal reference power supply	Set the bit to "0".
To switch on internal reference power supply	Set the bit to "1".

● Method of selecting the method of activating the 8/10-bit A/D conversion function

Use the continuous activation enable bit (ADC2:EXT) to select an activation trigger.

A/D conversion activation source	Continuous activation enable bit (EXT)
To select the software trigger	Set the bit to "0".
To select the input clock rising signal	Set the bit to "1".

- Method of generating a software trigger
  Use the A/D conversion start bit (ADC1:AD) to generate a software trigger.

Operation	A/D conversion start bit (AD)
To generate a software trigger	Set the bit to "1".

- Method of activating the A/D conversion function using the input clock
  An activation trigger is generated at the rising edge of the input clock.
  To select the input clock, use external start signal select bit (ADC2:ADCK).

Input clock	External start signal select bit (ADCK)
Do not use any external start signal	Set the bit to "0".
To select the 8/16-bit composite timer output pin (TO00)	Set the bit to "1".

● Method of selecting A/D conversion precision

Use the precision select bit (ADC2:AD8) to select the precision of conversion results.

Operation	Precision select bit (AD8)
To select 10-bit precision	Set the bit to "0".
To select 8-bit precision	Set the bit to "1".

● Method of using analog input pins

Use the analog input pin select bits (ADC1:ANS3 to ANS0) to select an analog input pin.

Operation	Analog input pin select bits (ANS3 to ANS0)
To use the AN00 pin	Set the bits to "0000_B".
To use the AN01 pin	Set the bits to "0001_B".
To use the AN02 pin	Set the bits to "0010_B".
To use the AN03 pin	Set the bits to "0011_B".
To use the AN04 pin	Set the bits to "0100_B".
To use the AN05 pin	Set the bits to "0101_B".
To use the AN06 pin	Set the bits to "0110_B".
To use the AN07 pin	Set the bits to "0111_B".

● Method of checking the completion of conversion

There are two methods of checking whether conversion has been completed or not.

- Checking with the interrupt request flag bit (ADC1:ADI)

Interrupt request flag bit (ADI)	Meaning
The read value is "0".	No A/D conversion completion interrupt request
The read value is "1".	A/D conversion completion interrupt request made

- Checking with the conversion flag bit (ADC1:ADMV)

Conversion flag bit (ADMV)	Meaning
The read value is "0".	A/D conversion completed (stopped)
The read value is "1".	A/D conversion in progress

● Interrupted-related register

Use the following interrupt level setting register to set the interrupt level.

Interrupt source	Interrupt level setting register	Interrupt vector
8/10-bit AD converter	Interrupt level setting register (ILR4) Address: $0007D_H$	#18 Address: $0FFD6_H$

● Method of enabling, disabling, and clearing interrupts

Use the interrupt request enable bit (ADC2:ADIE) to enable interrupts.

Operation	Interrupt request enable bit (ADIE)
To disable interrupt requests	Set the bit to "0".
To enable interrupt requests	Set the bit to "1".

Use the interrupt request bit (ADC1:ADI) to clear an interrupt request.

Operation	Interrupt request bit (ADI)
To clear an interrupt request	Set the bit to "0".

CHAPTER 26

LOW-VOLTAGE DETECTION RESET CIRCUIT

This chapter describes the function and operation of the low-voltage detection reset circuit (only available on MB95F414K/F416K/F418K/F474K/F476K/F478K).

26.1 Overview of Low-voltage Detection Reset Circuit

**The low-voltage detection reset circuit monitors power supply voltage and generates a reset signal if the power supply voltage drops below the low-voltage detection voltage level.
This circuit is only available on MB95F414K/F416K/F418K/F474K/F476K/F478K.**

■ Low-voltage Detection Reset Circuit

The low-voltage detection reset circuit monitors power supply voltage and generates a reset signal if the power supply voltage drops below the low-voltage detection voltage level.

This circuit is only available on MB95F414K/F416K/F418K/F474K/F476K/F478K.

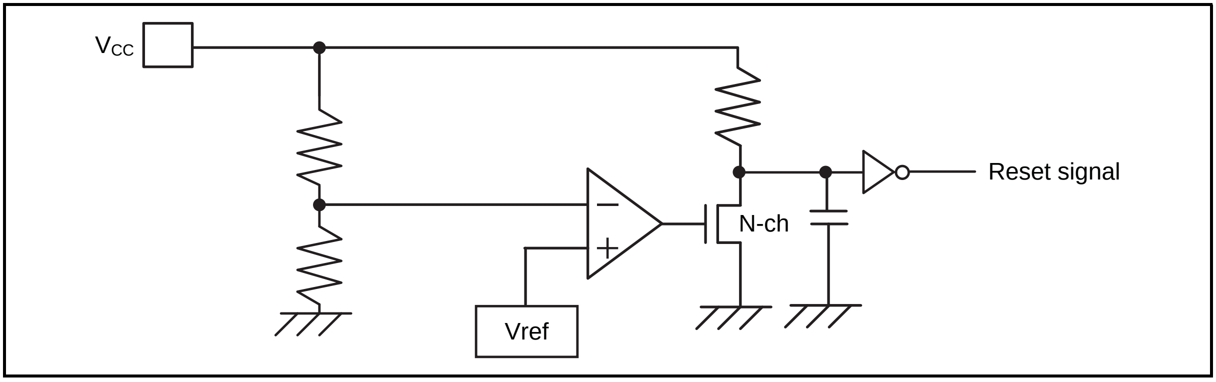
Refer to the data sheet of the MB95410H/470H Series for details of the electrical characteristics of this circuit.

26.2 Configuration of Low-voltage Detection Reset Circuit

Figure 26.2-1 is the block diagram of the low-voltage detection reset circuit.

■ Block Diagram of Low-voltage Detection Reset Circuit

Figure 26.2-1 Block Diagram of Low-voltage Detection Reset Circuit



26.3 Pins of Low-voltage Detection Reset Circuit

This section describes the pins of the low-voltage detection reset circuit.

■ Pins of Low-voltage Detection Reset Circuit

● V_{CC} pin

The low-voltage detection reset circuit monitors the voltage of this pin.

● V_{SS} pin

This is the GND pin serving as the reference for voltage detection.

● $\overline{\text{RST}}$ pin

The low-voltage detection reset signal is output within the microcontroller and to this pin.

26.4 Operation of Low-voltage Detection Reset Circuit

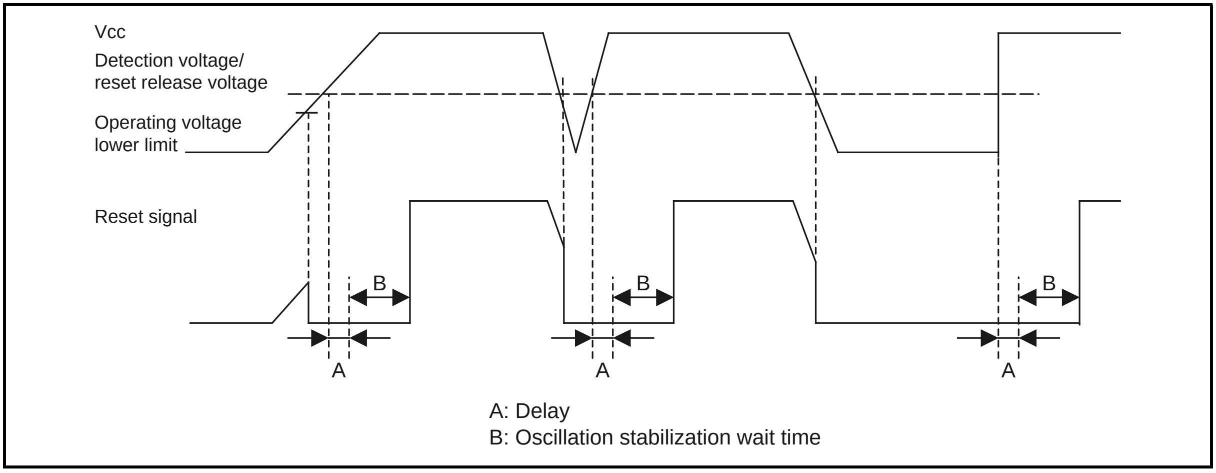
The low-voltage detection reset circuit can generate a reset signal if the power supply voltage drops below the low-voltage detection voltage.

■ Operation of Low-voltage Detection Reset Circuit

The low-voltage detection reset circuit generates a reset signal if the power supply voltage falls below the low-voltage detection voltage. Afterward, if the low-voltage detection reset circuit detects the low-voltage detection reset release voltage, it outputs a reset signal lasting for the oscillation stabilization wait time and then releases the reset.

For details of the electrical characteristics mentioned above, refer to the data sheet of the MB95410H/470H Series.

Figure 26.4-1 Operation of Low-voltage Detection Reset



■ Operation in Standby Mode

The low-voltage detection reset circuit keeps operating even in standby mode (stop mode, sleep mode, subclock mode and watch mode).

CHAPTER 27

CLOCK SUPERVISOR COUNTER

This chapter describes the functions and operations of the clock supervisor counter.

27.1 Overview of Clock Supervisor Counter

The clock supervisor counter can check the external clock frequency to detect the abnormal state of the external clock.

■ Overview of Clock Supervisor Counter

The clock supervisor counter can check the external clock frequency to detect the abnormal state of the external clock.

It counts up a built-in 8-bit counter according to the external clock input within a time-base timer interval selected from eight options.

The count clock of this module can be selected from the main oscillation clock, the main PLL clock and the suboscillation clock.

Note:

The clock supervisor counter must operate in main CR clock mode with the hardware watchdog timer (running in standby mode).

Otherwise, it cannot detect the abnormal state of the external clock correctly and will hang up if the external clock stops.

See "CHAPTER 12 HARDWARE/SOFTWARE WATCHDOG TIMER" for the hardware watchdog timer (running in standby mode).

27.2 Configuration of Clock Supervisor Counter

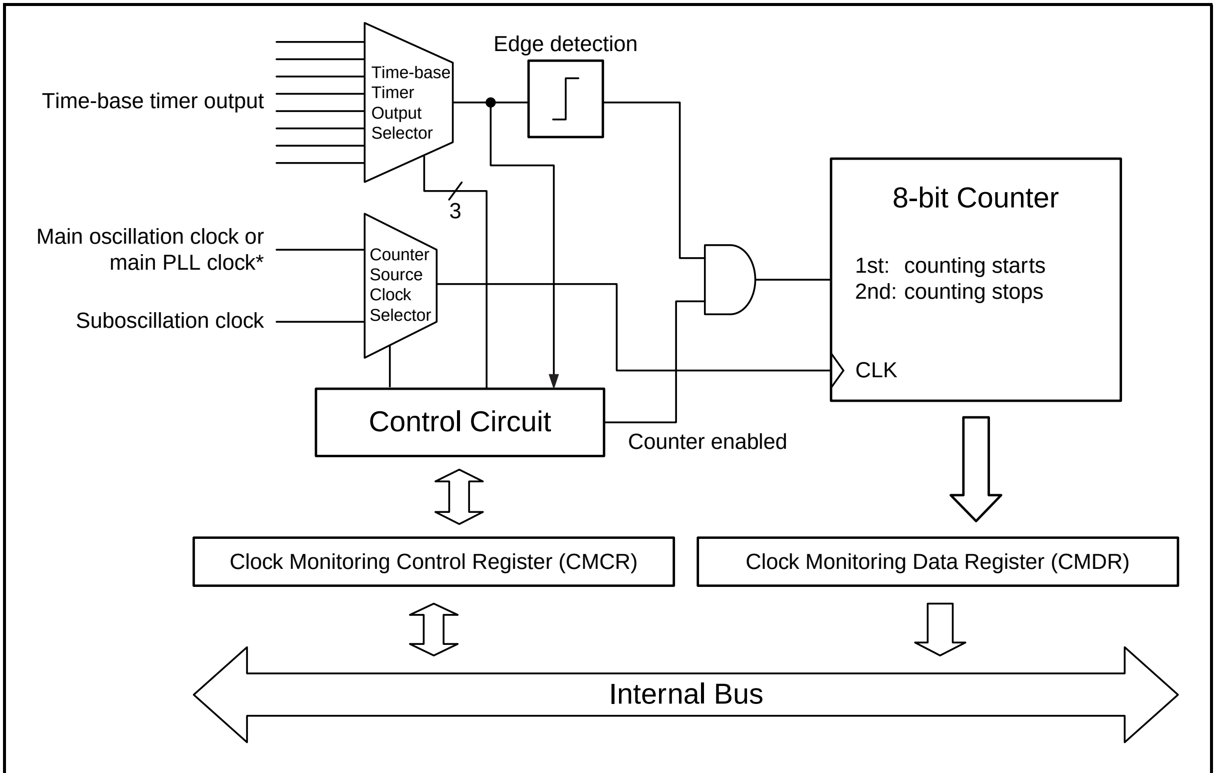
The clock supervisor counter consists of the following blocks:

- **Control circuit**
- **Clock Monitoring Control Register (CMCR)**
- **Clock Monitoring Data Register (CMDR)**
- **Time-base timer output selector**
- **Counter source clock selector**

■ Block Diagram of Clock Supervisor Counter

Figure 27.2-1 is the block diagram of the clock supervisor counter.

Figure 27.2-1 Block Diagram of Clock Supervisor Counter


Time-base timer output
Time-base Timer Output Selector
Edge detection
3
Main oscillation clock or main PLL clock*
Suboscillation clock
Counter Source Clock Selector
8-bit Counter
1st: counting starts
2nd: counting stops
CLK
Control Circuit
Counter enabled
Clock Monitoring Control Register (CMCR)
Clock Monitoring Data Register (CMDR)
Internal Bus

*: When the PLLC:PCS[1:0] bits are set to “00”, the main clock divided by 2 (FCH/2) is used as the count clock.
When the PLLC:PCS[1:0] bits are set to “01”, “10” or “11”, the main PLL clock is used as the count clock.

- Control circuit

  This block controls the start and stop of the counter, the counter clock source, and the counter enable period based on the settings of the clock monitoring control register (CMCR).

- Clock Monitoring Control Register (CMCR)

  This register is used to select the counter source clock, select the counter enable period from the eight different time-base timer intervals, start the counter and check whether the counter is operating or not.

- Clock Monitoring Data Register (CMDR)

  This register block is used to read the counter value after the counter stops. The software can determine whether the external clock frequency is correct or not according to the contents of this register.

- Time-base timer interval selector

  This block is used to select the counter enable period from eight different time-base timer intervals.

- Counter source clock selector

  This block is used to select the counter source clock from the main oscillation clock, the main PLL clock and the suboscillation clock.

27.3 Registers of Clock Supervisor Counter

This section describes the registers of the clock supervisor counter.

■ Registers of Clock Supervisor Counter

Figure 27.3-1 shows the registers of the clock supervisor counter.

Figure 27.3-1 Clock Supervisor Counter Registers

Clock monitoring data register (CMDR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FEA$_H$	CMDR7	CMDR6	CMDR5	CMDR4	CMDR3	CMDR2	CMDR1	CMDR0	00000000$_B$
	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

Clock monitoring control register (CMCR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FE9$_H$	-	-	Reserved	CMCSEL	TBTSEL2	TBTSEL1	TBTSEL0	CMCEN	00000000$_B$
	R0/WX	R0/WX	R/W0	R/W	R/W	R/W	R/W	R/W	

R/W	: Readable/writable (The read value is the same as the write value.)
R/WX	: Read only (Readable. Writing a value to this bit has no effect on operation.)
R/W0	: The write value is "0". The read value is the same as the write value.
R0/WX	: The read value is "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

27.3.1 Clock Monitoring Data Register (CMDR)

The clock monitoring data register (CMDR) is used to read the count value after the clock supervisor counter stops. The software can determine whether the external clock frequency is correct or not according to the content of this register.

■ Clock Monitoring Data register (CMDR)

Figure 27.3-2 Clock Monitoring Data Register (CMDR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FEA_H$	CMDR7	CMDR6	CMDR5	CMDR4	CMDR3	CMDR2	CMDR1	CMDR0	00000000_B
	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)

The clock monitoring data register (CMDR) is used to read the counter value after the clock supervisor counter stops.

- The counter value can be read from this clock monitoring data register (CMDR). The software can check whether the external clock frequency is correct or not according to the counter value read and the time-base timer interval selected.

Table 27.3-1 Functions of Bits in Clock Monitoring Data Register (CMDR)

Bit name		Function
bit7 to bit0	CMDR7 to CMDR0	The CMDR register is a data register indicating the clock supervisor counter value after the counter stops. This register is cleared if one of the following events occurs: • Reset • The CMCEN bit is modified from "0" to "1" by the software. • The CMCEN bit is modified from "1" to "0" by the software while the counter is running. • After the external clock stops, the falling edge of the selected time-base timer clock is detected twice (See Figure 27.5-2).

Note:

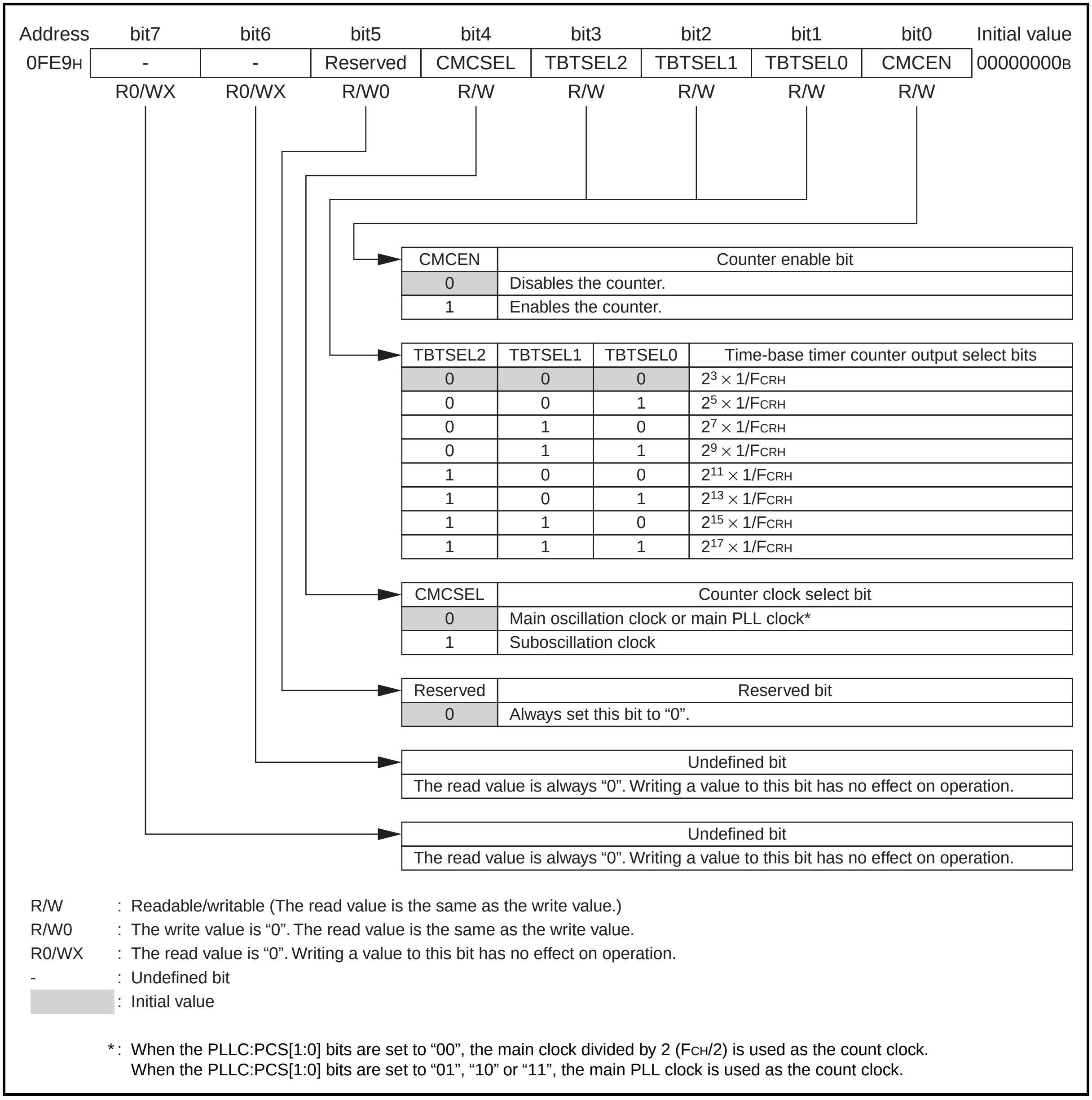
The value of this register is "0" as long as the counter is operating (CMCEN = 1).

27.3.2 Clock Monitoring Control Register (CMCR)

The clock monitoring control register (CMCR) is used to select the counter source clock, select the time-base timer interval as the counter enable period, start the counter and check whether the counter is running or not.

■ Clock Monitoring Control register (CMCR)

Figure 27.3-3 Clock Monitoring Control Register (CMCR)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FE9H	-	-	Reserved	CMCSEL	TBTSEL2	TBTSEL1	TBTSEL0	CMCEN	00000000B
	R0/WX	R0/WX	R/W0	R/W	R/W	R/W	R/W	R/W	

CMCEN	Counter enable bit
0	Disables the counter.
1	Enables the counter.

TBTSEL2	TBTSEL1	TBTSEL0	Time-base timer counter output select bits
0	0	0	$2^3 \times 1/F_{CRH}$
0	0	1	$2^5 \times 1/F_{CRH}$
0	1	0	$2^7 \times 1/F_{CRH}$
0	1	1	$2^9 \times 1/F_{CRH}$
1	0	0	$2^{11} \times 1/F_{CRH}$
1	0	1	$2^{13} \times 1/F_{CRH}$
1	1	0	$2^{15} \times 1/F_{CRH}$
1	1	1	$2^{17} \times 1/F_{CRH}$

CMCSEL	Counter clock select bit
0	Main oscillation clock or main PLL clock*
1	Suboscillation clock

Reserved	Reserved bit
0	Always set this bit to “0”.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W	:	Readable/writable (The read value is the same as the write value.)
R/W0	:	The write value is “0”. The read value is the same as the write value.
R0/WX	:	The read value is “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

*: When the PLLC:PCS[1:0] bits are set to “00”, the main clock divided by 2 ($F_{CH}/2$) is used as the count clock.
When the PLLC:PCS[1:0] bits are set to “01”, “10” or “11”, the main PLL clock is used as the count clock.

Table 27.3-2 Functions of Bits in Clock Monitoring Control Register (CMCR)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7,
bit6</td><td>Undefined bits</td><td>Their read values are always "0". Writing values to these bits has no effect on operation.</td></tr>
<tr><td>bit5</td><td>Reserved bit</td><td>Always set this bit to "0".</td></tr>
<tr><td>bit4</td><td>CMCSEL:
Counter clock select bit</td><td>This bit selects the counter clock source.
<b>Writing "0":</b> Selects the main oscillation clock (PLLC:PCS[1:0] = 00) or the main PLL clock (PLLC:PCS[1:0] = 01, 10 or 11) as the source clock of the counter.
<b>Writing "1":</b> Selects the suboscillation clock as the source clock of the counter.</td></tr>
<tr><td>bit3
to
bit1</td><td>TBTSEL2, TBTSEL1, TBTSEL0:
Time-base timer counter output select bits</td><td>These bits select the time-base timer interval.
The operation of the clock supervisor counter is enabled and disabled according to the time-base timer counter output selected by these bits.
The first rising edge of the interval selected enables the counter operation and the second rising edge of the same output disables the counter operation.
<table>
<tr><th>TBTSEL2</th><th>TBTSEL1</th><th>TBTSEL0</th><th>Time-base timer counter output select bits</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>$2^{3} \times 1/F_{CRH}$</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>$2^{5} \times 1/F_{CRH}$</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>$2^{7} \times 1/F_{CRH}$</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>$2^{9} \times 1/F_{CRH}$</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>$2^{11} \times 1/F_{CRH}$</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>$2^{13} \times 1/F_{CRH}$</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>$2^{15} \times 1/F_{CRH}$</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>$2^{17} \times 1/F_{CRH}$</td></tr>
</table></td></tr>
<tr><td>bit0</td><td>CMCEN:
Counter enable bit</td><td>This bit enables and disables the clock supervisor counter.
<b>Writing "0":</b> Stops the counter and clears the CMDR register.
<b>Writing "1":</b> Enables the counter. The counter starts counting when detecting the rising edge of the time-base timer interval. It stops counting when detecting the second rising edge of the same interval.
This bit is automatically set to "0" when the counter stops.</td></tr>
</table>

Notes:

- Do not modify the CMCSEL bit when CMCEN = 1.
- Do not modify the TBTSEL[2:0] bits when CMCEN = 1.

27.4 Operations of Clock Supervisor Counter

This section describes the operations of the clock supervisor counter.

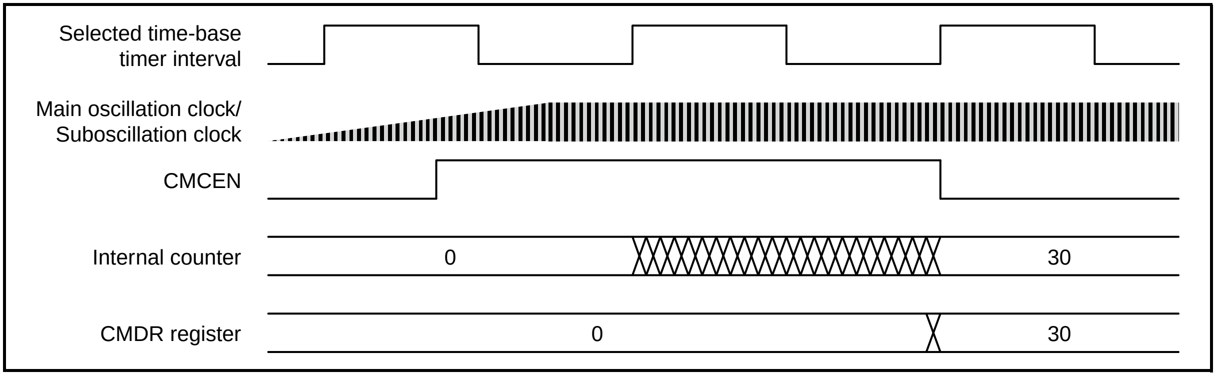
■ Clock Supervisor Counter

● Clock Supervisor Counter Operation 1

The clock supervisor counter is first enabled by the software (CMCEN = 1), and then the clock supervisor counter operates with the time-base timer interval selected from eight options by the TBTSEL[2:0] bits. Between two rising edges of the time-base timer interval selected, the internal counter is clocked by the external clock.

The count clock of this module can be selected from the main oscillation clock, the main PLL clock and the suboscillation clock.

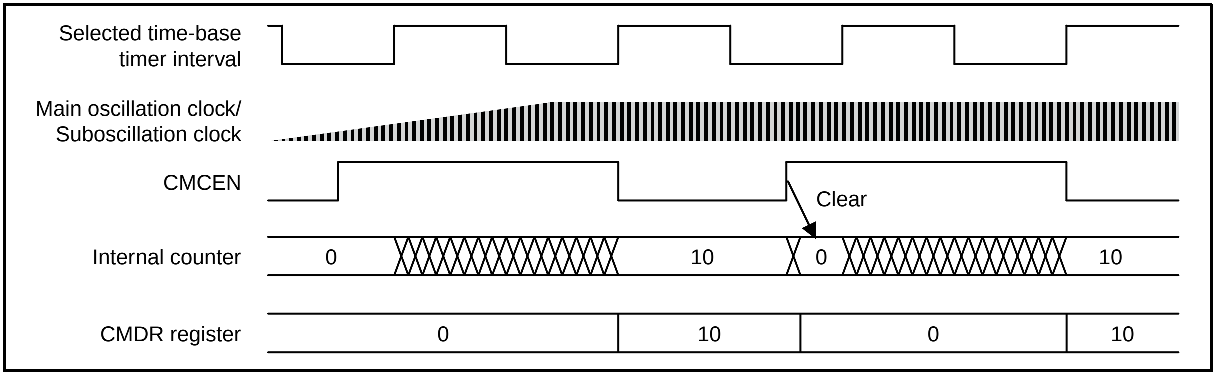
Figure 27.4-1 Clock Supervisor Counter Operation


Selected time-base timer interval
Main oscillation clock/ Suboscillation clock
CMCEN
Internal counter
0
30
CMDR register
0
30

● Clock Supervisor Counter Operation 2

The CMDR register is cleared when the CMCEN bit changes from "0" to "1".

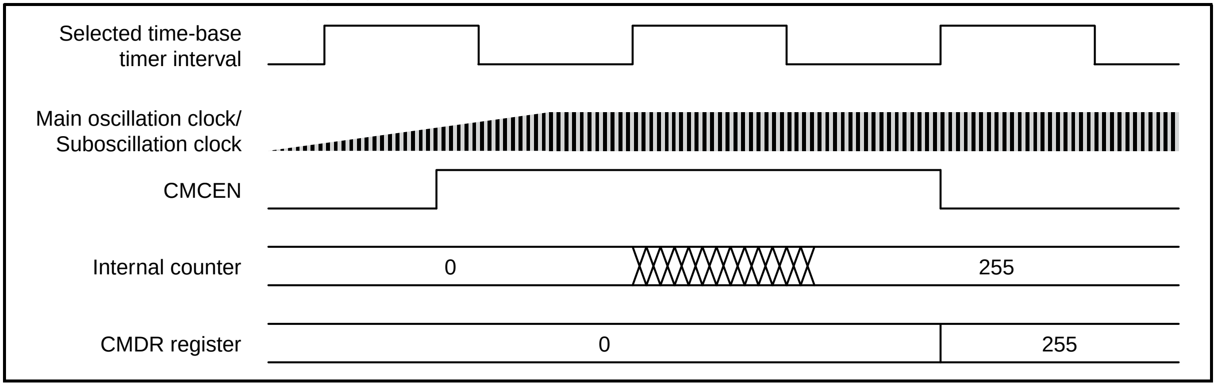
Figure 27.4-2 Clock Supervisor Counter Operation 2


Selected time-base timer interval
Main oscillation clock/ Suboscillation clock
CMCEN
Clear
Internal counter
0
10
0
10
CMDR register
0
10
0
10

● Clock Supervisor Counter Operation 3

The counter stops counting if it reaches "255". It cannot count further.

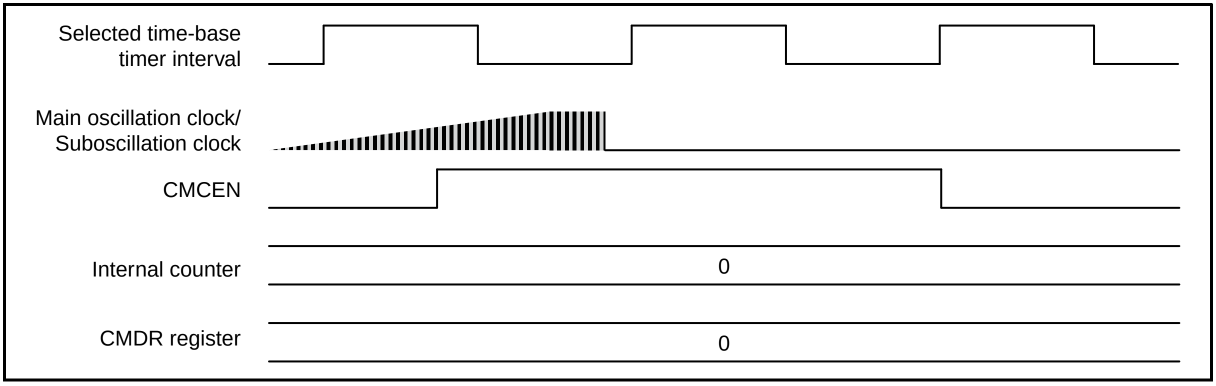
Figure 27.4-3 Clock Supervisor Counter Operation 3


Selected time-base timer interval
Main oscillation clock/ Suboscillation clock
CMCEN
Internal counter
0
255
CMDR register
0
255

● Clock Supervisor Counter Operation 4

If the external clock selected stops, the counter stops counting. The software can then identify that the external clock selected is in the abnormal state.

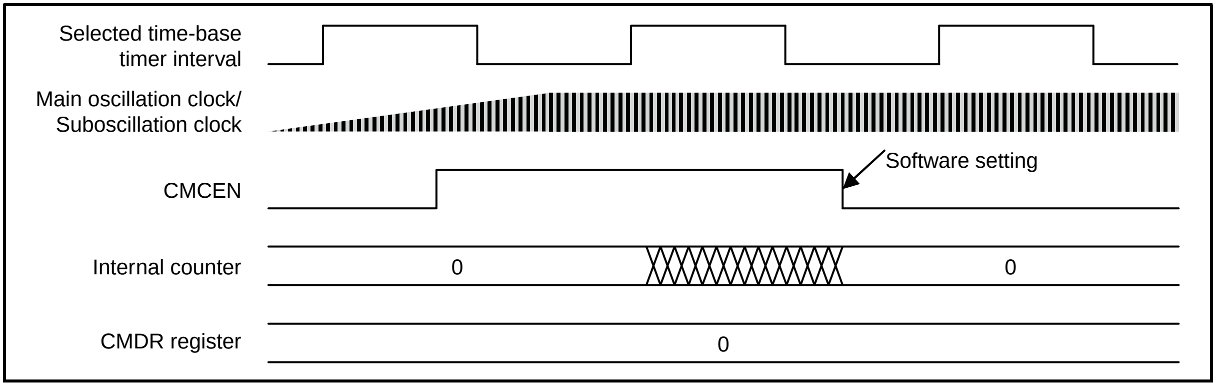
Figure 27.4-4 Clock Supervisor Counter Operation 4


Selected time-base timer interval
Main oscillation clock/ Suboscillation clock
CMCEN
Internal counter
0
CMDR register
0

● Clock Supervisor Counter Operation 5

The counter is cleared to "0" by the software if the CMCEN is set to "0" while the counter is operating.

Figure 27.4-5 Clock Supervisor Counter Operation 5


Selected time-base timer interval
Main oscillation clock/ Suboscillation clock
Software setting
CMCEN
Internal counter
0
0
CMDR register
0

■ Table of Time-base Timer Intervals & Clock Supervisor Counter Values

Table 27.4-1 shows time-base timer intervals suitable for using different main CR clock frequency to measure different external clocks.

Table 27.4-1 Table of Counter Values in Relation to TBTSEL Settings (1 / 2)

Main CR (F_{CRH}) [MHz]	Main/Sub-crystal oscillation [MHz]	Main CR error	Measur-ement error	TBTSEL2 - TBTSEL0							
				000_B	001_B	010_B	011_B	100_B	101_B	110_B	111_B
				(2^3×1/F_{CRH})	(2^5×1/F_{CRH})	(2^7×1/F_{CRH})	(2^9×1/F_{CRH})	(2^{11}×1/F_{CRH})	(2^{13}×1/F_{CRH})	(2^{15}×1/F_{CRH})	(2^{17}×1/F_{CRH})
1	0.03277	+5%	-1	0	0	0	6	30	126	510	2044
		-5%	+1	1	1	3	9	36	142	566	2261
	0.5	+5%	-1	0	6	29	120	486	1949	7800	31206
		-5%	+1	3	9	34	135	539	2156	8624	34493
	1	+5%	-1	2	14	59	242	974	3899	15602	62414
		-5%	+1	5	17	68	270	1078	4312	17247	68986
	4	+5%	-1	14	59	242	974	3899	15602	62414	249659
		-5%	+1	17	68	270	1078	4312	17247	68986	275942
	6	+5%	-1	21	90	364	1461	5850	23404	93621	374490
		-5%	+1	26	102	405	1617	6468	25870	103478	413912
	10	+5%	-1	37	151	608	2437	9751	39008	156037	624151
		-5%	+1	43	169	674	2695	10779	43116	172464	689853
	20	+5%	-1	75	303	1218	4875	19503	78018	312075	1248303
		-5%	+1	85	337	1348	5390	21558	86232	344927	1379706
	32.5	+5%	-1	122	494	1979	7922	31694	126779	507122	2028494
		-5%	+1	137	548	2190	8758	35032	140127	560506	2242022
8	0.03277	+5%	-1	0	0	0	0	2	14	62	254
		-5%	+1	1	1	1	2	5	18	71	283
	0.5	+5%	-1	0	0	2	14	59	242	974	3899
		-5%	+1	1	2	5	17	68	270	1078	4312
	1	+5%	-1	0	0	6	29	120	486	1949	7800
		-5%	+1	1	3	9	34	135	539	2156	8624
	4	+5%	-1	0	6	29	120	486	1949	7800	31206
		-5%	+1	3	9	34	135	539	2156	8624	34493
	6	+5%	-1	1	10	44	181	730	2924	11701	46810
		-5%	+1	4	13	51	203	809	3234	12935	51739
	10	+5%	-1	3	18	75	303	1218	4875	19503	78018
		-5%	+1	6	22	85	337	1348	5390	21558	86232
	20	+5%	-1	8	37	151	608	2437	9751	39008	156037
		-5%	+1	11	43	169	674	2695	10779	43116	172464
	32.5	+5%	-1	14	60	246	989	3960	15846	63389	253560
		-5%	+1	18	69	274	1095	4379	17516	70064	280253

Table 27.4-1 Table of Counter Values in Relation to TBTSEL Settings (2 / 2)

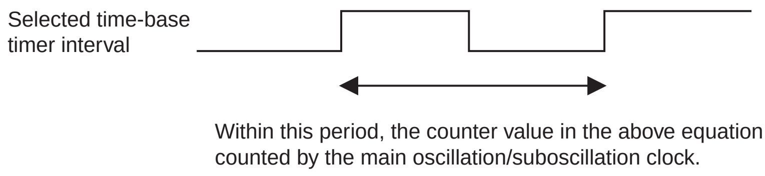
Main CR (FCRH) [MHz]	Main/Sub-crystal oscillation [MHz]	Main CR error	Measur-ement error	TBTSEL2 - TBTSEL0							
				000_B	001_B	010_B	011_B	100_B	101_B	110_B	111_B
				(2^3×1/FCRH)	(2^5×1/FCRH)	(2^7×1/FCRH)	(2^9×1/FCRH)	(2^{11}×1/FCRH)	(2^{13}×1/FCRH)	(2^{15}×1/FCRH)	(2^{17}×1/FCRH)
10	0.03277	+5%	-1	0	0	0	0	2	11	50	203
		-5%	+1	1	1	1	1	4	15	57	227
	0.5	+5%	-1	0	0	2	11	47	194	779	3119
		-5%	+1	1	1	4	14	54	216	863	3450
	1	+5%	-1	0	0	5	23	96	389	1559	6240
		-5%	+1	1	2	7	27	108	432	1725	6899
	4	+5%	-1	0	5	23	96	389	1559	6240	24965
		-5%	+1	2	7	27	108	432	1725	6899	27595
	6	+5%	-1	1	8	35	145	584	2339	9361	37448
		-5%	+1	3	11	41	162	647	2587	10348	41392
	10	+5%	-1	2	14	59	242	974	3899	15602	62414
		-5%	+1	5	17	68	270	1078	4312	17247	68986
	20	+5%	-1	6	29	120	486	1949	7800	31206	124829
		-5%	+1	9	34	135	539	2156	8624	34493	137971
	32.5	+5%	-1	11	48	197	791	3168	12677	50711	202848
		-5%	+1	14	55	219	876	3504	14013	56051	224203
12.5	0.03277	+5%	-1	0	0	0	0	1	9	39	162
		-5%	+1	1	1	1	1	3	12	46	181
	0.5	+5%	-1	0	0	1	8	38	155	623	2495
		-5%	+1	1	1	3	11	44	173	690	2760
	1	+5%	-1	0	0	3	18	77	311	1247	4992
		-5%	+1	1	2	6	22	87	345	1380	5519
	4	+5%	-1	0	3	18	77	311	1247	4992	19971
		-5%	+1	2	6	22	87	345	1380	5519	22076
	6	+5%	-1	0	6	28	116	467	1871	7488	29958
		-5%	+1	3	9	33	130	518	2070	8279	33113
	10	+5%	-1	2	11	47	194	779	3119	12482	49931
		-5%	+1	4	14	54	216	863	3450	13798	55189
	20	+5%	-1	5	23	96	389	1559	6240	24965	99863
		-5%	+1	7	27	108	432	1725	6899	27595	110377
	32.5	+5%	-1	8	38	157	632	2534	10141	40568	162278
		-5%	+1	11	44	176	701	2803	11211	44841	179362

░ : Recommended setting

▓ : The counter value becomes "0" or "255".

Table 27.4-1 is calculated by the following equation:

$$\text{Counter value}^{*1} = \frac{\left\{\begin{array}{l} 2^{3} \times 1/F_{CRH}(\text{TBTSEL}=000) \\ 2^{5} \times 1/F_{CRH}(\text{TBTSEL}=001) \\ 2^{7} \times 1/F_{CRH}(\text{TBTSEL}=010) \\ 2^{9} \times 1/F_{CRH}(\text{TBTSEL}=011) \\ 2^{11} \times 1/F_{CRH}(\text{TBTSEL}=100) \\ 2^{13} \times 1/F_{CRH}(\text{TBTSEL}=101) \\ 2^{15} \times 1/F_{CRH}(\text{TBTSEL}=110) \\ 2^{17} \times 1/F_{CRH}(\text{TBTSEL}=111) \end{array}\right\} \times \text{Main oscillation/Suboscillation clock frequency}}{2^{*2}} \pm 1 \text{ (Measurement error)}$$



Within this period, the counter value in the above equation is counted by the main oscillation/suboscillation clock.

*1 : Omit the decimal places of the counter value.

*2 : If the main PLL clock is selected as the count clock, this value becomes “1” (no division).

If the time-base timer interrupt is used to make the clock supervisor counter wait for the oscillation stabilization time, please satisfy the following condition:

Time-base Timer Interval > Main oscillation/Suboscillation Stabilization Time × 1.05

e.g. F_{CH} = 4 MHz, F_{CRH} = 1 MHz, MWT[3:0] = 1111 (in WATR register)

$$\text{Time-base Timer Interval} > \frac{(2^{14} - 2)}{4 \times 10^{6}} \times 1.05 \approx 4.3 \text{ ms}$$

↓

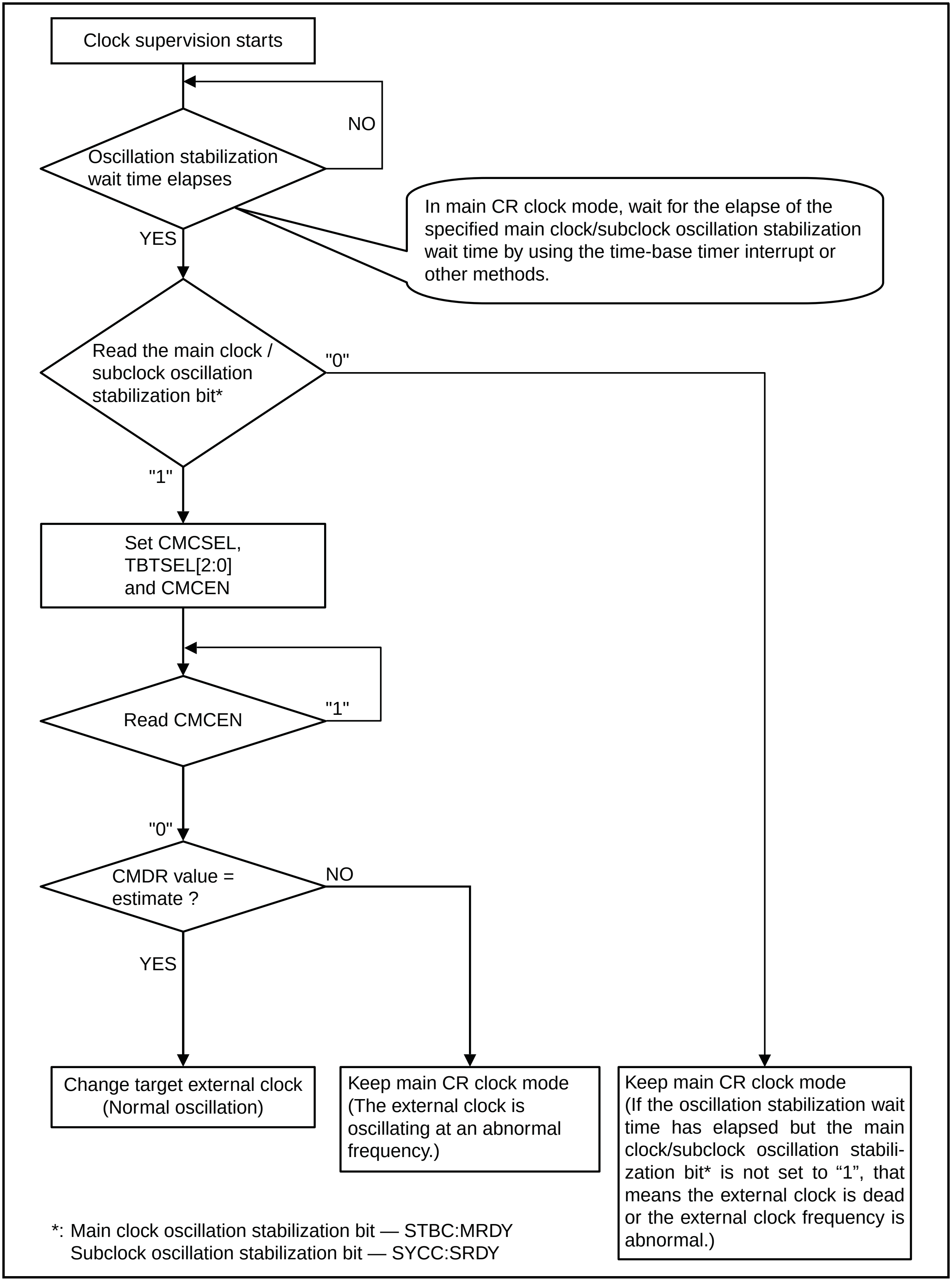
TBC[3:0] = 0110 ($2^{13} \times 1/F_{CRH}$)

Notes:

- See "11.1 Overview of Time-base Timer" for time-base timer interval settings.
- See "6.5 Oscillation Stabilization Wait Time Setting Register (WATR)" for main/ sub-oscillation stabilization time settings.

■ Sample Operation Flow Chart of Clock Supervisor

Figure 27.4-6 Sample Operation Flow Chart of Clock Supervisor


Clock supervision starts
Oscillation stabilization wait time elapses
NO
YES
In main CR clock mode, wait for the elapse of the specified main clock/subclock oscillation stabilization wait time by using the time-base timer interrupt or other methods.
Read the main clock / subclock oscillation stabilization bit*
"0"
"1"
Set CMCSEL, TBTSEL[2:0] and CMCEN
Read CMCEN
"1"
"0"
CMDR value = estimate ?
NO
YES
Change target external clock (Normal oscillation)
Keep main CR clock mode (The external clock is oscillating at an abnormal frequency.)
Keep main CR clock mode (If the oscillation stabilization wait time has elapsed but the main clock/subclock oscillation stabilization bit* is not set to “1”, that means the external clock is dead or the external clock frequency is abnormal.)
*: Main clock oscillation stabilization bit — STBC:MRDY
Subclock oscillation stabilization bit — SYCC:SRDY

27.5 Notes on Using Clock Supervisor Counter

This section provides notes on using the clock supervisor counter.

■ Notes on Using Clock Supervisor Counter

● Restrictions

- The clock supervisor counter must operate in main CR clock mode with the hardware watchdog timer (running in standby mode). Otherwise, it cannot detect the abnormal state of the external clock correctly and will hang up if the external clock stops. See "CHAPTER 12 HARDWARE/SOFTWARE WATCHDOG TIMER" for the hardware watchdog timer (running in standby mode).
- Use main CR clock mode only. DO NOT use any other clock mode.
- If the time-base timer stops, the internal counter stops working. DO NOT clear the time-base timer while the clock supervisor counter is counting with the external clock.
- Select a time-base timer interval that is sufficiently long for the clock supervisor counter to operate. See Table 27.4-1 for time-base timer intervals.
- Read the CMDR register when CMCEN = 0. (The value of CMDR remains "0" while the clock supervisor counter is operating (CMCEN = 1).)
- When using the clock supervisor counter, ensure that the machine clock cycle is shorter than half the time-base timer interval selected. If the machine clock cycle is longer than half the time-base timer interval selected, CMCEN may remain "1" even after the clock supervisor counter stops.

Table 27.5-1 below shows the appropriate clock gear setting for each TBTSEL setting.

Table 27.5-1 Appropriate Clock Gear Setting for Respective TBTSEL

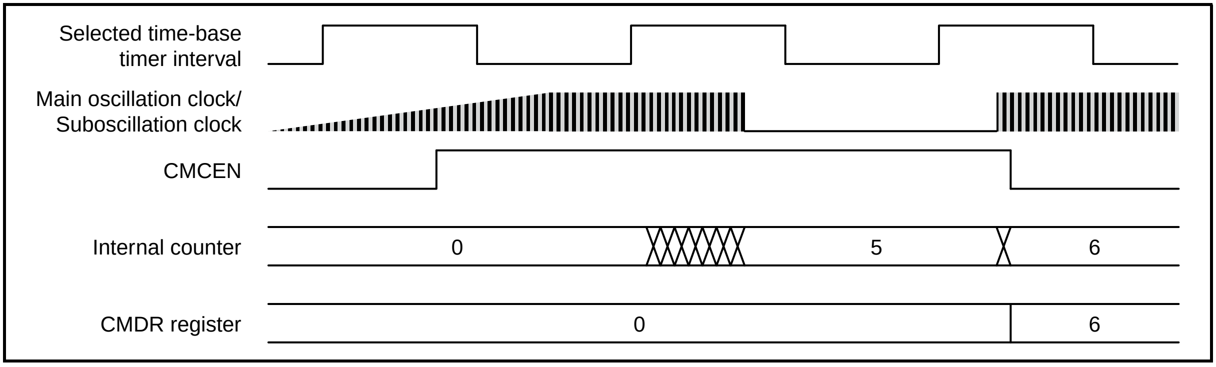
DIV (clock gear setting)	TBTSEL2 to TBTSEL0		
	000_B	001_B	010_B to 111_B
	$2^3 \times 1/F_{CRH}$	$2^5 \times 1/F_{CRH}$	$2^7 \times 1/F_{CRH}$ to $2^{17} \times 1/F_{CRH}$
00 ($1 \times 1/F_{CRH}$)	❍	❍	❍
01 ($4 \times 1/F_{CRH}$)	x	❍	❍
10 ($8 \times 1/F_{CRH}$)	x	❍	❍
11 ($16 \times 1/F_{CRH}$)	x	x	❍

❍: Recommended

x: Prohibited

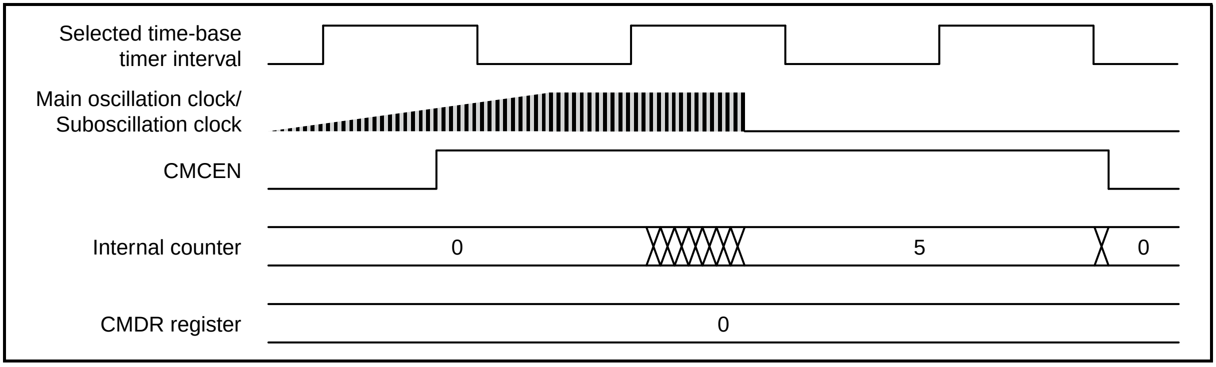
- If the external clock stops while the clock supervisor counter is operating, and it restarts after the second rising edge of the time-base timer interval selected, CMCEN is set to "0" after the external clock restarts.

Figure 27.5-1 Clock Supervisor Counter Operation 1


Selected time-base timer interval
Main oscillation clock/ Suboscillation clock
CMCEN
Internal counter
0
5
6
CMDR register
0
6

- With the clock supervisor counter running, if the external clock stops, CMCEN is set to "0" when a falling edge of the time-base timer interval selected is detected after the second rising edge of the same interval. The counter is cleared at the same falling edge.

Figure 27.5-2 Clock Supervisor Counter Operation 2


Selected time-base timer interval
Main oscillation clock/ Suboscillation clock
CMCEN
Internal counter
0
5
0
CMDR register
0

CHAPTER 28

LCD CONTROLLER (MB95410H SERIES)

This chapter describes the functions and operations of the LCD controller.

28.1 Overview of LCD Controller

The LCD controller has 2 modes: 8 COM mode and 4 COM mode.
In 8 COM mode, the LCD controller can use 36 bytes of display data memory and controls an LCD display via 8 common outputs and 36 segment outputs. It also has 2 different bias output options for driving an LCD panel.
In 4 COM mode, the LCD controller can use 20 bytes of display data memory and controls an LCD display via 4 common outputs and 40 segment outputs. It also has 3 different duty output options for driving an LCD panel.

■ Functions of LCD Controller

The LCD controller uses its segment and common outputs to display the contents of display data memory (display RAM) directly on the LCD panel.

- It selects the 8 COM mode and the 4 COM mode through software.
- It has an LCD drive voltage divider resistor whose resistance value can be selected from 10 kΩ to 100 kΩ through software. An external divider resistor can also be used instead.
- In 8 COM mode, 8 common outputs (COM0 to COM7) and 36 segment outputs (SEG00 to SEG35) are available
- In 4 COM mode, 4 common outputs (COM0 to COM3) and 40 segment outputs (SEG00 to SEG39) are available.
- The display RAM size is 36 bytes (36 × 8 bits) in 8 COM mode and 20 bytes (40 × 4 bits) in 4 COM mode.
- It can use the main clock or the subclock as its operating clock.
- It has a blinking function, which is only available to certain pins.
- It can directly drive an LCD panel.
- In 8 COM mode, the bias can be selected from 1/3 or 1/4.
- In 4 COM mode, the duty can be selected from 1/2, 1/3 or 1/4 (governed by the bias setting).
- The interrupt is in sync with the LCD module frame frequency.

Table 28.1-1 lists the bias-duty combinations available.

Table 28.1-1 Bias-duty Combinations

Duty	1/2 bias	1/3 bias	1/4 bias
1/2	❍	X	X
1/3	X	❍	X
1/4	X	❍	X
1/8, BLS8 = 0	X	❍	X
1/8, BLS8 = 1	X	X	❍

❍ : Recommended combination
X : Prohibited combination

28.2 Configuration of LCD Controller

The LCD controller consists of the following blocks, which are divided functionally into a controller section that generates the segment and common signals based on the content of display RAM and a driver section that drives the LCD.

Controller section

- **LCDC control registers (LCDCC1, LCDCC2)**
- **LCDC enable registers (LCDCE1 to LCDCE7)**
- **LCDC blinking setting registers (LCDCB1, LCDCB2)**
- **Display RAM**
- **Clock selection**
- **Timing control**

Driver section

- **AC waveform generator circuit**
- **Common driver**
- **Segment driver**
- **Divider resistor**

■ LCD Controller Block Diagrams

Figure 28.2-1 LCD Controller Block Diagram (8 COM Mode)

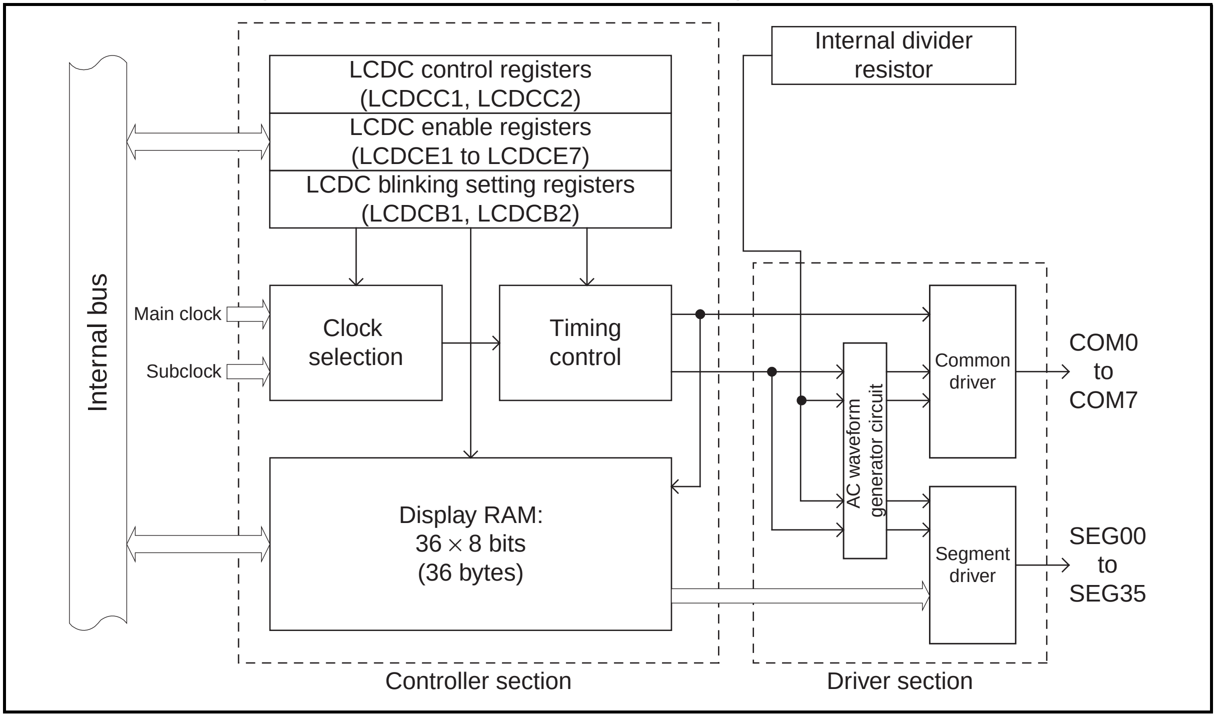
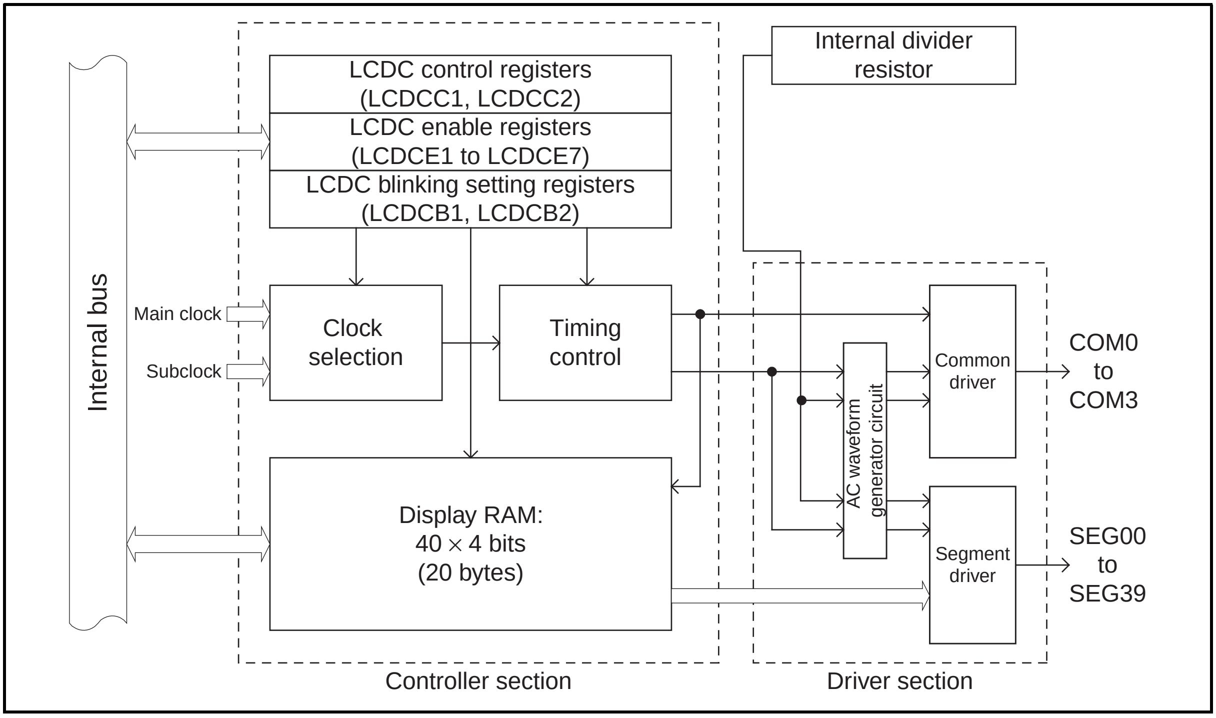

Internal bus
LCDC control registers (LCDCC1, LCDCC2)
LCDC enable registers (LCDCE1 to LCDCE7)
LCDC blinking setting registers (LCDCB1, LCDCB2)
Internal divider resistor
Main clock
Subclock
Clock selection
Timing control
Display RAM: 36 × 8 bits (36 bytes)
AC waveform generator circuit
Common driver
Segment driver
COM0 to COM7
SEG00 to SEG35
Controller section
Driver section

Figure 28.2-2 LCD Controller Block Diagram (4 COM Mode)


Internal bus
LCDC control registers (LCDCC1, LCDCC2)
LCDC enable registers (LCDCE1 to LCDCE7)
LCDC blinking setting registers (LCDCB1, LCDCB2)
Main clock
Subclock
Clock selection
Timing control
Display RAM: 40 × 4 bits (20 bytes)
Internal divider resistor
AC waveform generator circuit
Common driver
Segment driver
COM0 to COM3
SEG00 to SEG39
Controller section
Driver section

- LCDC control register 1 (LCDCC1)

This register is used to select the clock for generating the frame period, select the display mode, select the frame period clock, and control the LCD driving power supply.

- LCDC control register 2 (LCDCC2)

This register is used to enable and disable interrupts, indicate interrupt status and set the following parameters:

- Internal resistance value (10 kΩ or 100 kΩ)
- Bias to be used in 8 COM mode (1/3 or 1/4)
- Displaying data or a blank screen
- Inverted display

- LCDC enable registers 1 to 7 (LCDCE1 to LCDCE7)

These registers are used to control port inputs, blink interval, and pins.

- LCDC blinking setting register 1 (LCDCB1), LCDC blinking setting register 2 (LCDCB2)

These registers are used to turn on or off blinking.

- Display RAM

In 8 COM mode, 36 × 8 bits of RAM is available for generating segment output signals.

In 4 COM mode, 40 × 4 bits of RAM is available for generating segment output signals.

The content of the display RAM are read automatically in sync with the common signal selection timing and are output from segment output pins.

When the display RAM is modified, the content of the VRAM will be output from segment output pins.

● Clock selection

The frame frequency is generated based on the selection from the eight frequencies generated from the two clocks.

● Timing control

The COM and SEG signals are controlled based on the frame frequency and register settings.

● AC waveform generator circuit

This block generates AC waveforms for driving the LCD from timing control signals.

● Common driver

This block is the driver of the LCD COM pins.

● Segment driver

This block is the driver of the LCD SEG pins.

● Divider resistor

This block is a resistor used to generate the LCD drive voltage. A divider resistor can be connected to as an external component when a LCDC drive power supply pin (V0 to V4) serves as a divider resistor connection pin.

■ LCD Controller Power Supply Voltage

The power supply voltage for the LCD driver is generated by internal divider resistors or by connecting external divider resistors to the V0 to V4 pins.

■ Input Clock

The LCD controller uses the output clock of time-base timer or watch prescaler as the input clock (operation clock).

28.2.1 Internal Divider Resistors for LCD Controller

The internal divider resistors generate power supply voltage for the LCD driver.

■ Internal Divider Resistors

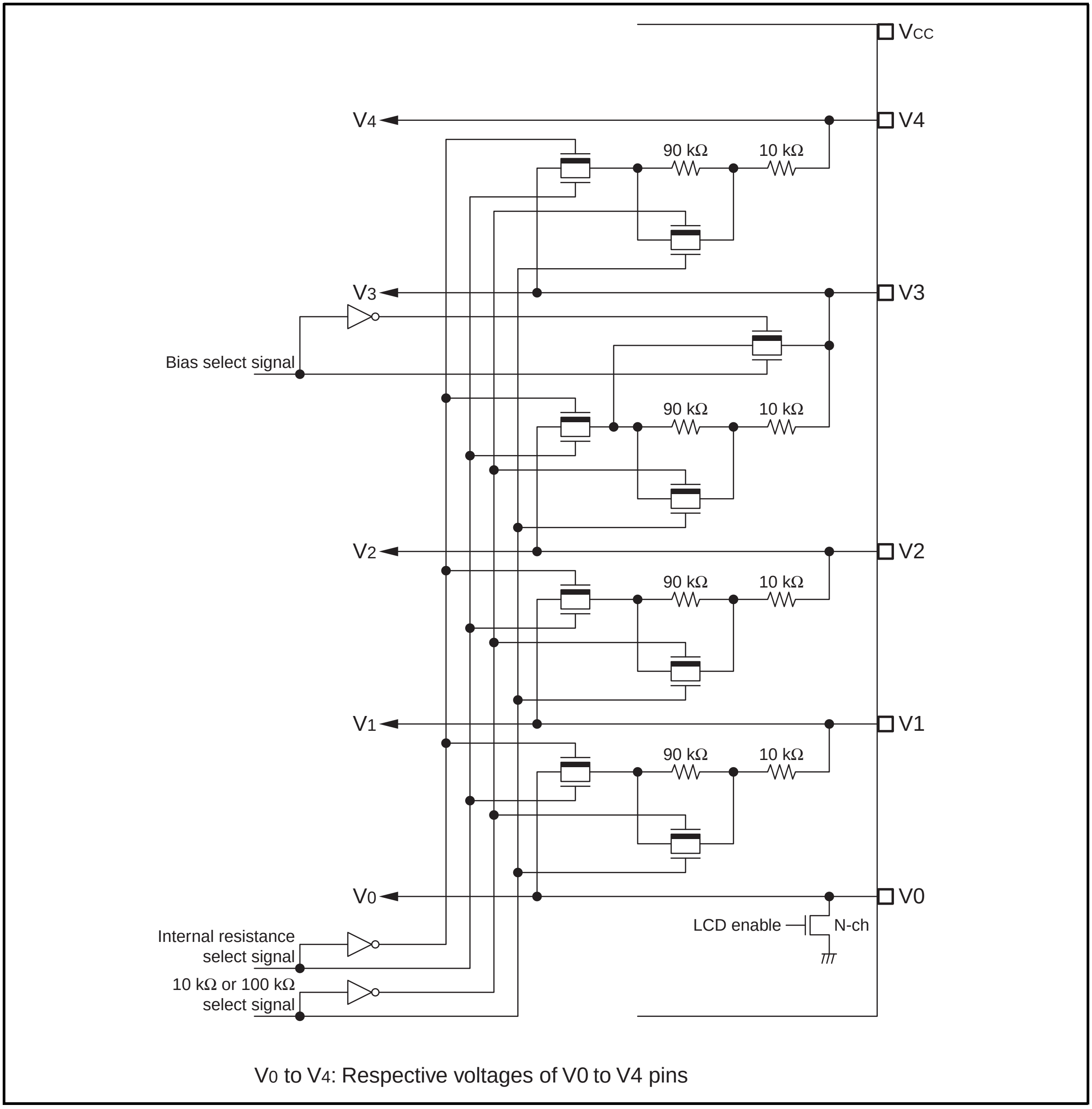
Internal divider resistors are included. In addition, external divider resistors can be connected to the LCDC drive power pins (V0 to V4).

The internal and external divider resistors are selected by the driving power control bit in the LCDC control register 1 (LCDCC1:VSEL). Setting the VSEL bit to "1" energizes the internal divider resistors. To use only the internal divider resistors without any external divider resistor, set the VE3 bit in the LCDC enable register 1 (LCDCE1) to "1". (When internal split resistors are used, the V4 pin cannot be used as general-purpose I/O ports.)

The LCD controller stops upon transition to main stop or watch mode (STBC:TMD = 1) while operation in main stop and watch modes is disabled (LCDCC1:LCDEN = 0) with LCD operation halted (LCDCC1:MS[2:0] = 000_B).

Figure 28.2-3 shows an equivalent circuit with internal divider resistors used.

Figure 28.2-3 Equivalent Circuit with Internal Divider Resistors Used



V0 to V4: Respective voltages of V0 to V4 pins

■ Use of Internal Divider Resistors and Brightness Control

There are two types of internal divider resistors: 10 kΩ and 100 kΩ. Figure 28.2-4 shows examples of using the internal divider resistors.

If sufficient brightness cannot be achieved with the internal divider resistors in use, connect a variable resistor (VR) externally (between the Vcc pin and the V4 pin) to adjust the V4 voltage. Figure 28.2-5 illustrates connecting a VR to the V4 pin to control brightness.

Figure 28.2-4 States with Internal Divider Resistors Used

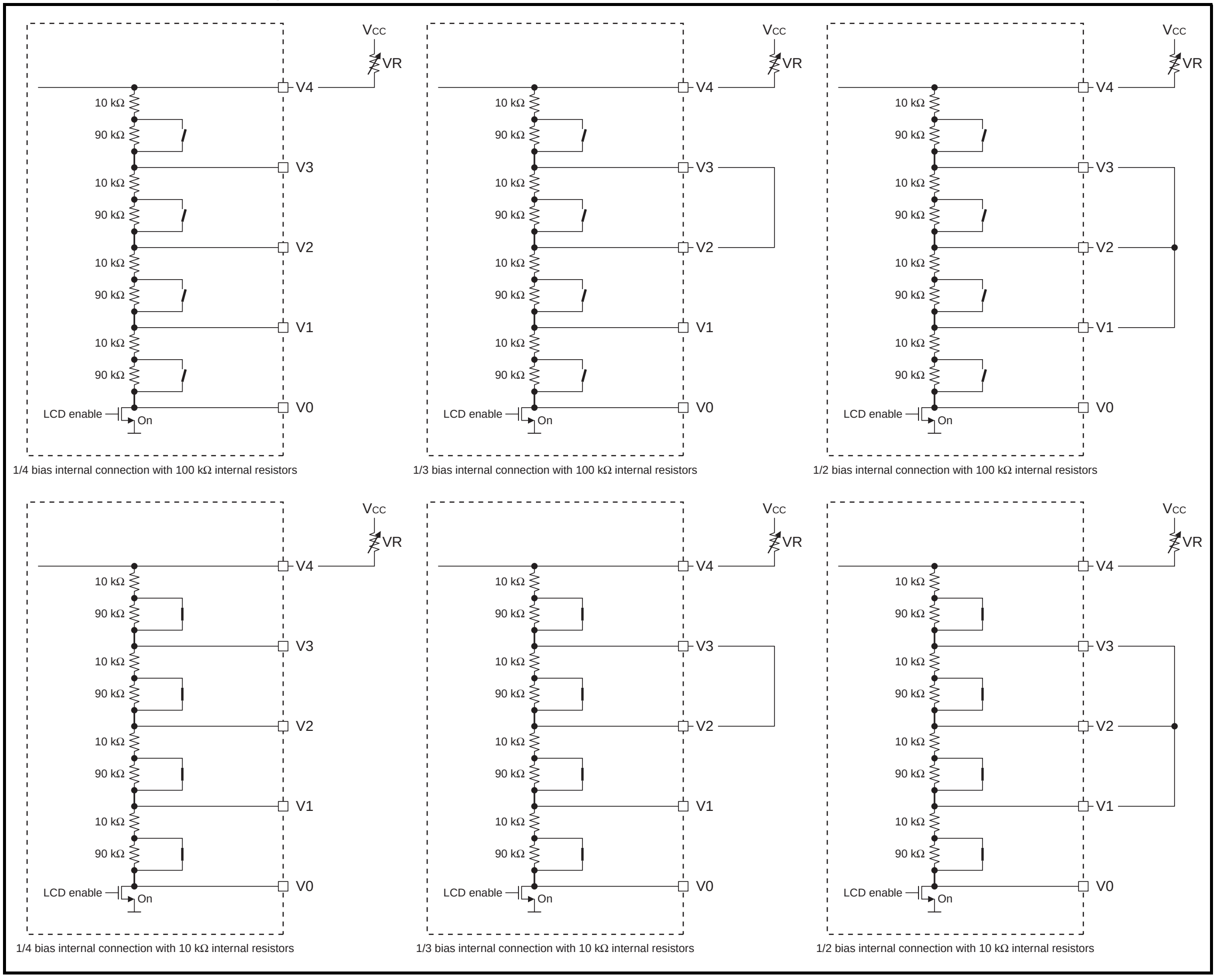
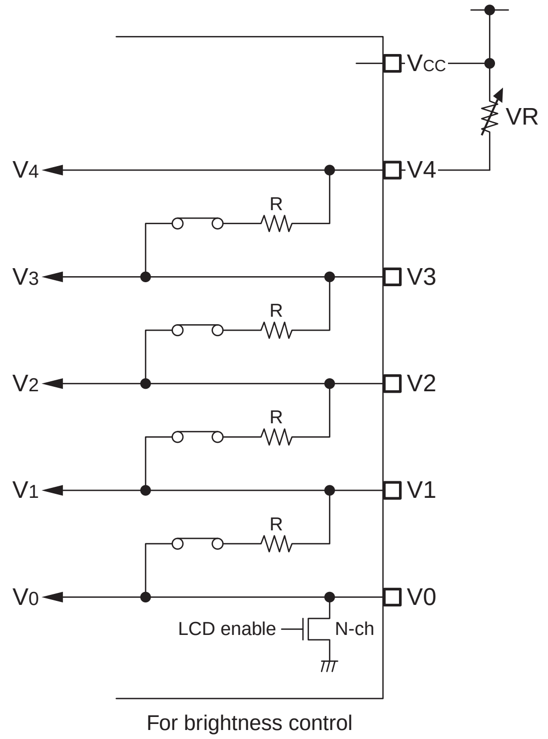

Vcc
VR
V4
V3
V2
V1
V0
10 kΩ
90 kΩ
LCD enable
On
1/4 bias internal connection with 100 kΩ internal resistors
1/3 bias internal connection with 100 kΩ internal resistors
1/2 bias internal connection with 100 kΩ internal resistors
1/4 bias internal connection with 10 kΩ internal resistors
1/3 bias internal connection with 10 kΩ internal resistors
1/2 bias internal connection with 10 kΩ internal resistors

Figure 28.2-5 Brightness Control by Connecting VR to V4 Pin



V_0 to V_4: Respective voltages of V0 to V4 pins

28.2.2 External Divider Resistors for LCD Controller

The V0 to V4 pins of this series can be connected to external divider resistors. Connecting a variable resistor between the V_{CC} pin and the V4 pin can control brightness.

■ External Divider Resistors

If not using the internal divider resistors, you can connect external divider resistors to the LCD drive power supply pins (V0 to V4) instead. Figure 28.2-6 shows an example of connecting external divider resistors, and Table 28.2-1 lists the LCD drive voltage settings for the bias method.

Figure 28.2-6 Example of Connecting External Divider Resistors

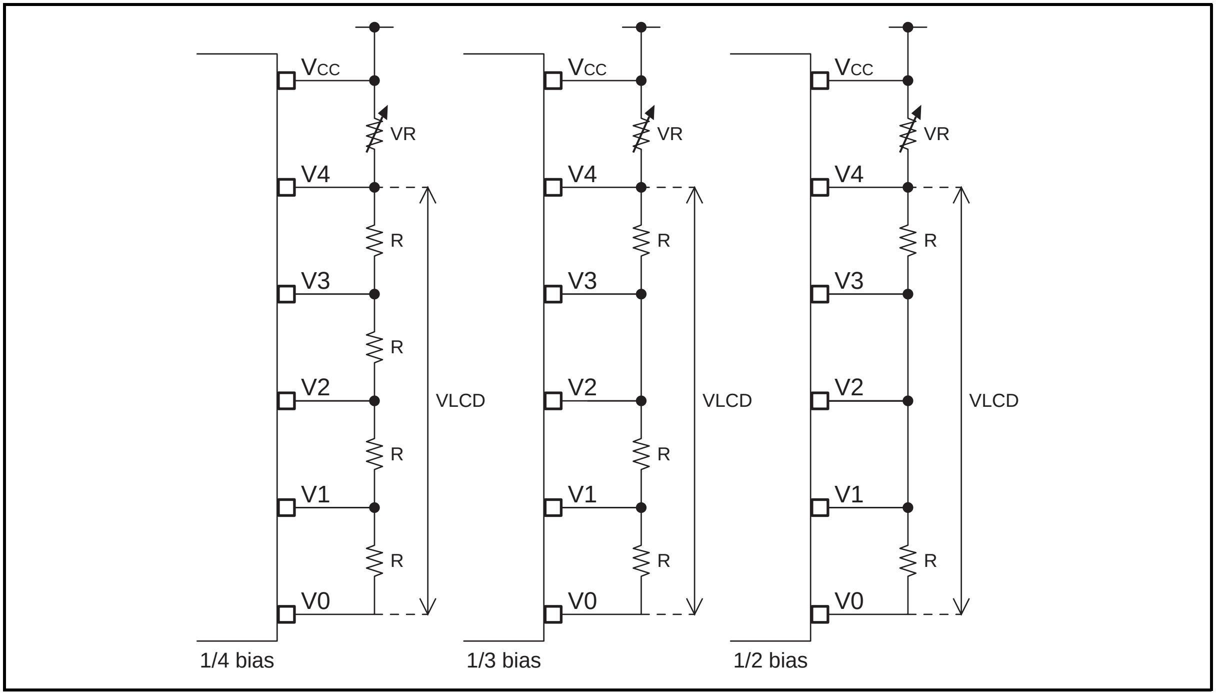


Table 28.2-1 LCD Drive Voltage Settings

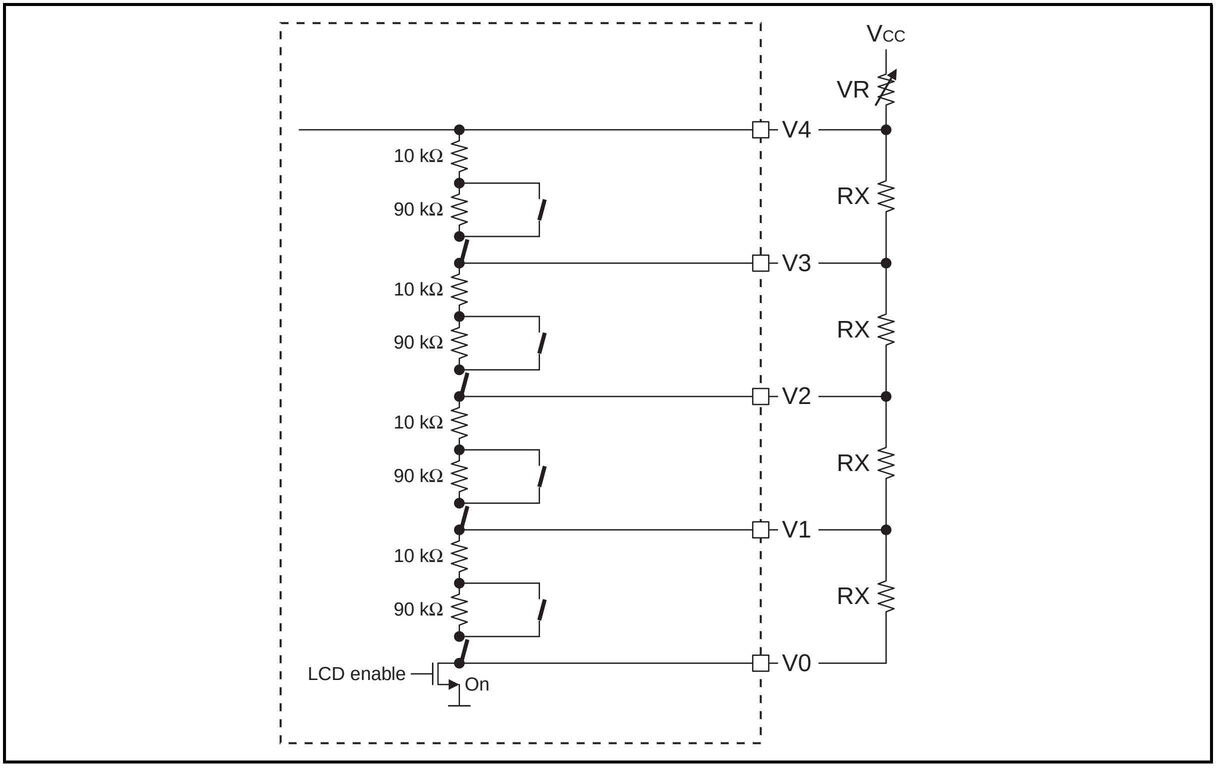
	V4	V3	V2	V1	V0
1/2 bias	VLCD	X	1/2 VLCD	X	GND
1/3 bias	VLCD	2/3 VLCD	2/3 VLCD	1/3 VLCD	GND
1/4 bias	VLCD	3/4 VLCD	1/2 VLCD	1/4 VLCD	GND

VLCD	: LCD operating voltage
X	: No external divider resistor
GND	: Ground

■ Use of External Divider Resistors

As the V0 pin is connected to V_{SS} (GND) internally via a transistor, when using external divider resistors, you can shut off the current flowing to the resistors when the LCD controller is halted by connecting the V_{SS} end of the divider resistors to the V0 pin. Figure 28.2-7 shows the state with external divider resistors used.

Figure 28.2-7 States with External Divider Resistors Used



1. To connect the external divider resistors without being affected by the internal divider resistors, write "0" to the drive voltage control bit in the LCDC control register 1 (LCDCC1:VSEL) to disconnect all internal divider resistors. Write "1" to the V4 to V0 select bits in the LCDC enable register 1 (LCDCE1:VE[4:0]) to use a pin as an LCD drive power supply pin.
2. When the internal divider resistors are disconnected, writing a value other than "000_B" to the display mode select bits (MS[2:0]) in the LCDCC1 register turns on the LCDC enable transistor (Q1) and, in turn, current flows to the external divider resistors.
3. Writing "000_B" to the MS[2:0] bits turns off the LCDC enable transistor (Q1) and, in turn, no current flows to the external divider resistor

Note:

The appropriate resistance of an external RX resistor depends on the LCD used. Use an external RX resistor whose resistance is suitable for the LCD used.

28.3 Pins of LCD Controller

This section describes the pins of the LCD controller.

■ Pins of LCD Controller

The pins of the LCD controller are: 8 common output pins (COM0 to COM7), 40 segment output pins (SEG00 to SEG39), and 5 LCD drive power supply pins (V0 to V4).

To use these pins for the LCD, set the corresponding bits in the LCDC enable registers (LCDCE1 to LCDCE7) to "1".

To use an LCD pin as a general-purpose I/O port, set its corresponding bit in an LCDC enable register (LCDCE1 to LCDCE7) for selecting the pin function to "0", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● COM0 to COM7 pins

In 8 COM mode, COM0 to COM7 function as LCD common output pins.

In 4 COM mode, COM0 to COM3 function as LCD common output pins, and COM4 to COM7 are defaulted as I/O ports regardless of the settings of the LCDCE1 to LCDCE7 registers.

In addition, COM0 to COM7 pins can also function as general-purpose I/O ports.

● SEG00 to SEG39 pins

In 8 COM mode, SEG00 to SEG35 function as LCD segment output pins, and SEG36 to SEG39 are defaulted as general-purpose I/O ports regardless of the settings of the LCDCE1 to LCDCE7 registers.

In 4 COM mode, SEG00 to SEG39 function as LCD segment output pins.

In addition, SEG00 to SEG39 can also function as general-purpose I/O ports.

● V0 to V4 pins

These pins function as the power supply pins for driving the LCD.

In addition, they can also function as general-purpose I/O ports.

■ Block Diagrams of Pins of LCD Controller

Figure 28.3-1 Block Diagram of V0 to V4 of LCD Controller

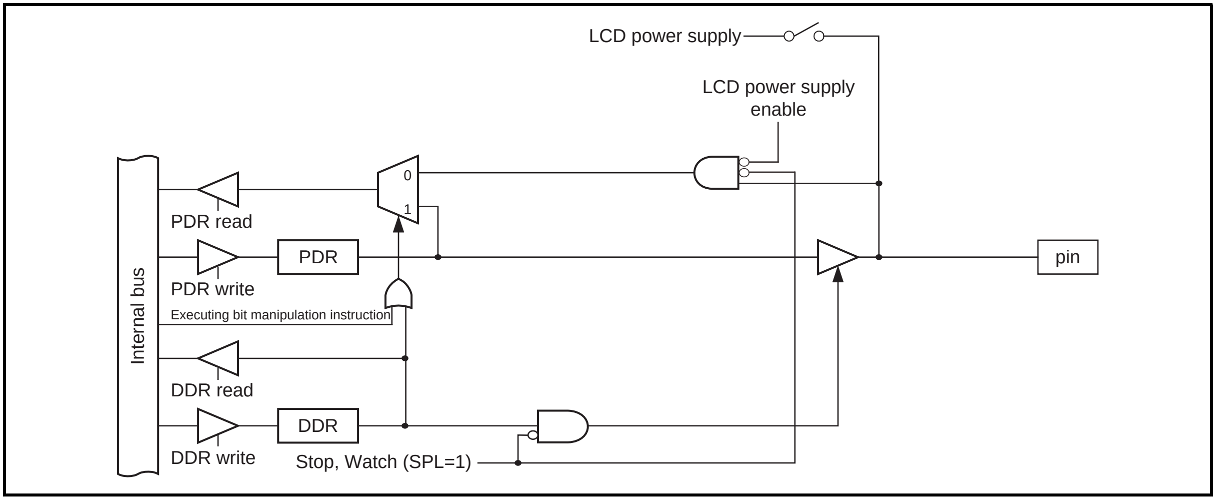


Figure 28.3-2 Block Diagram of COM0 to COM7, SEG00 to SEG26 and SEG37 to SEG39 of LCD Controller

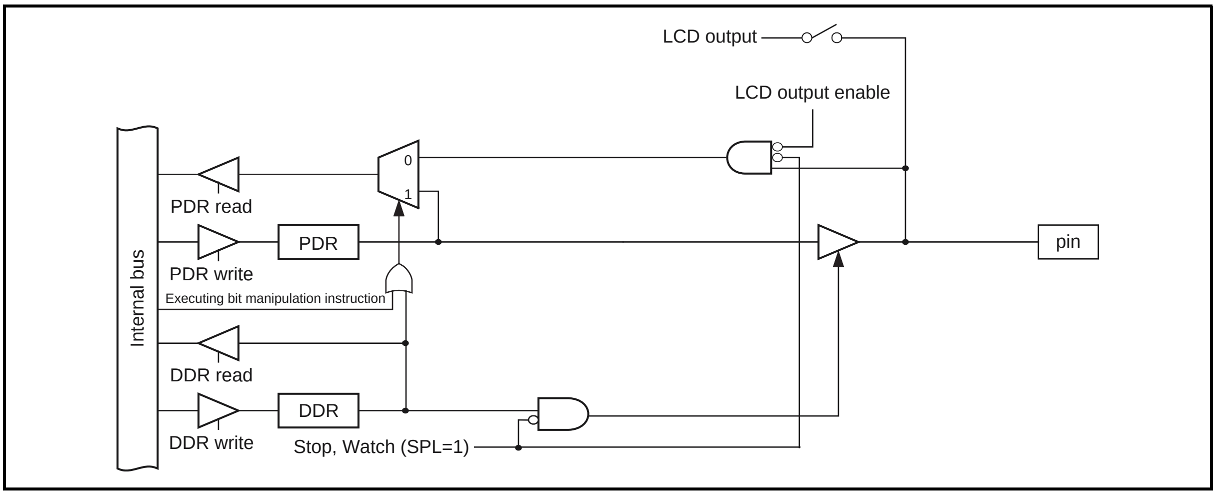


Figure 28.3-3 Block Diagram of SEG27 to SEG29 of LCD Controller

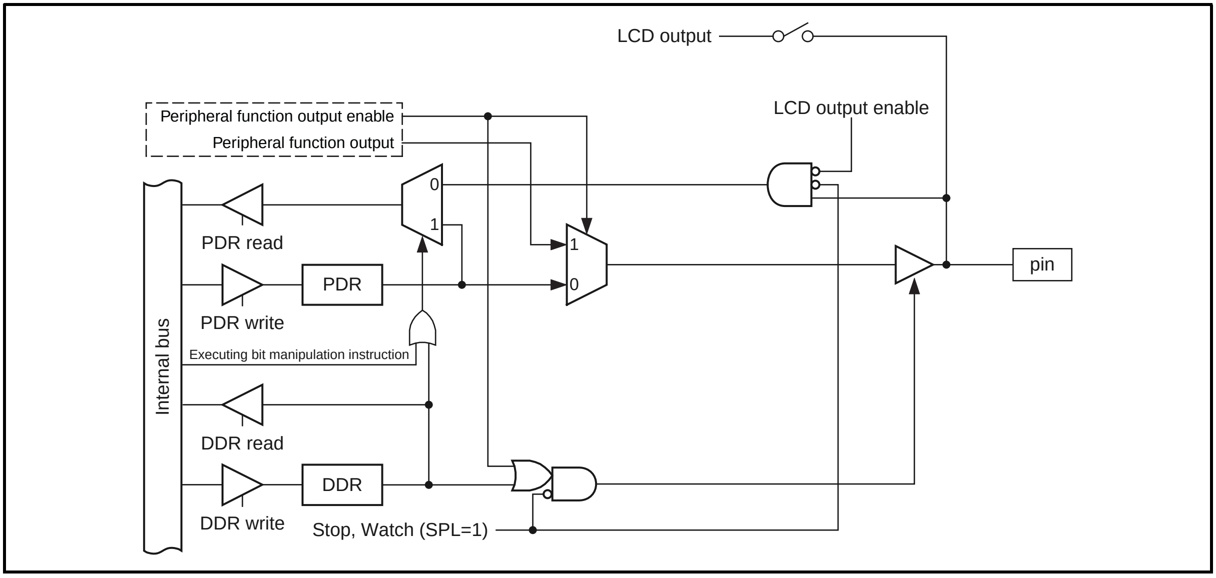


Figure 28.3-4 Block Diagram of SEG33 and SEG36 of LCD Controller

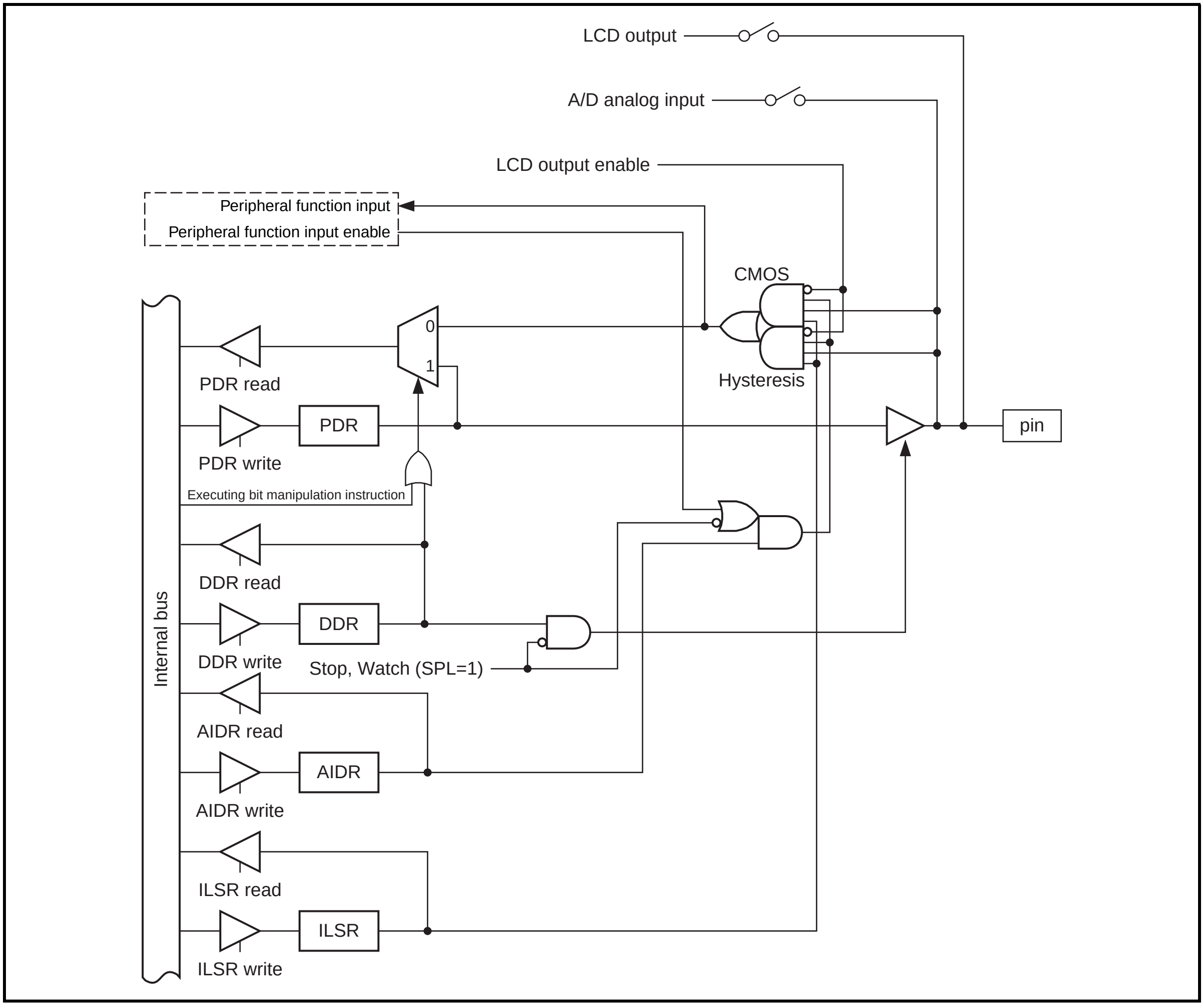


Figure 28.3-5 Block Diagram of SEG32, SEG34 and SEG35 of LCD Controller

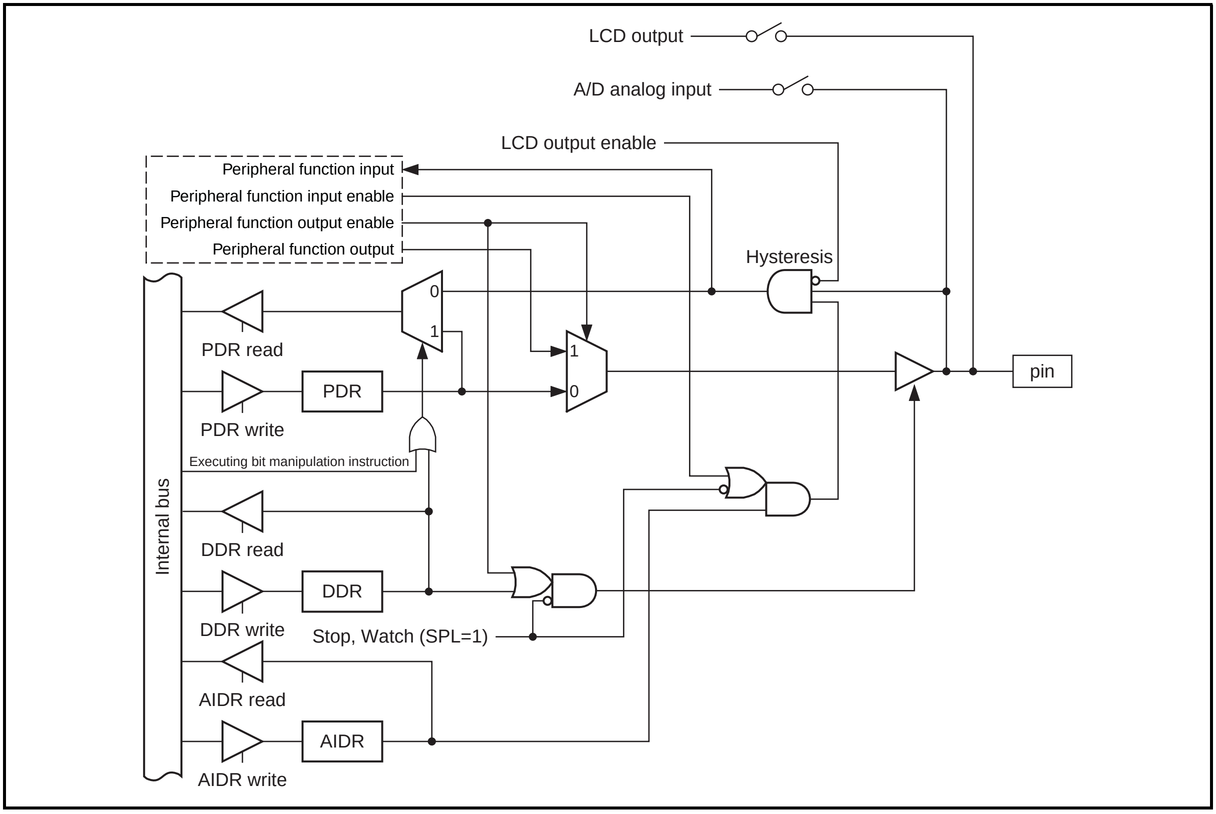
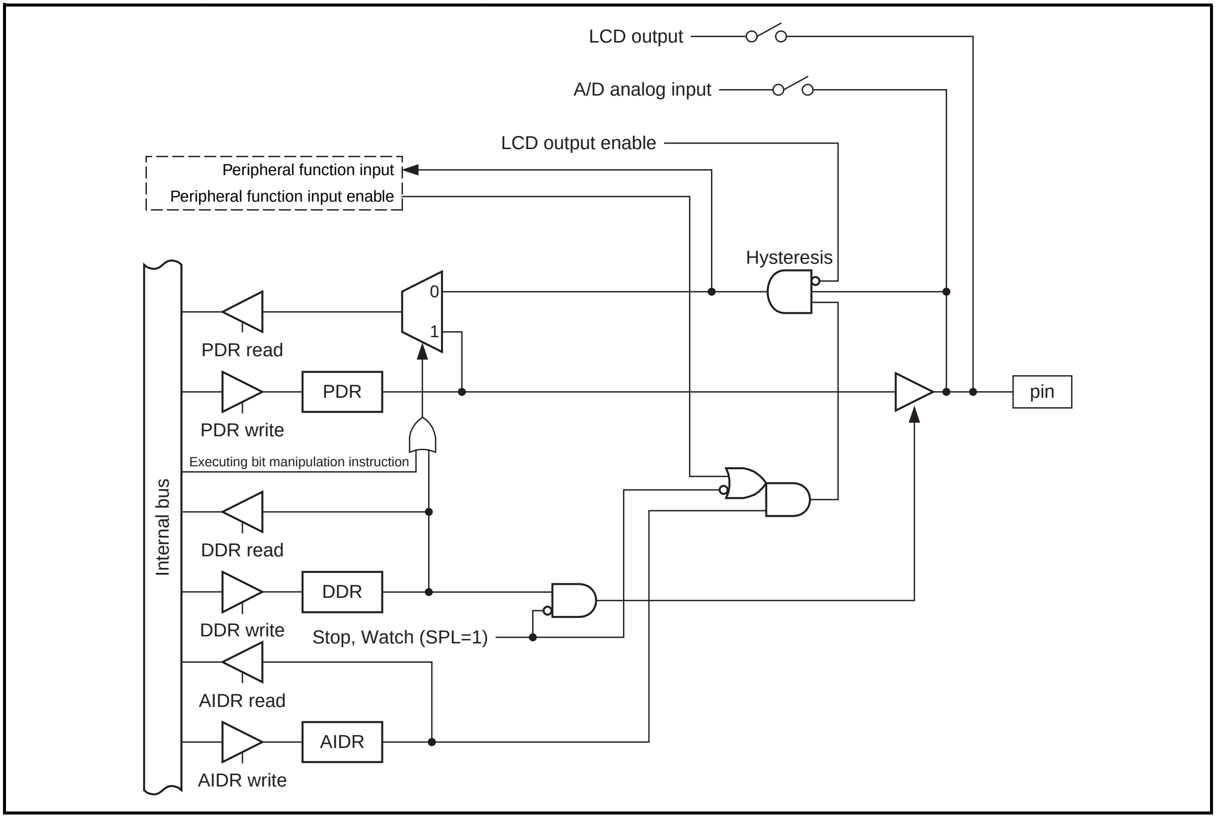

LCD output
A/D analog input
LCD output enable
Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Hysteresis
0
1
1
0
pin
PDR read
PDR
PDR write
Executing bit manipulation instruction
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
AIDR read
AIDR
AIDR write
Internal bus

Figure 28.3-6 Block Diagram of SEG30 and SEG31 of LCD Controller



28.4 Registers of LCD Controller

This section describes the registers of the LCD controller.

■ Registers of LCD Controller

Figure 28.4-1 LCD Controller Registers (1/2)

LCDC control register 1 (LCDCC1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB0_H$	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC control register 2 (LCDCC2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$004F_H$	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF	00010100_B
	R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R(RM1),W	

LCDC enable register 1 (LCDCE1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB2_H$	PICTL	BLSEL	VE4	VE3	VE2	VE1	VE0	-	00111110_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0/WX	

LCDC enable register 2 (LCDCE2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB3_H$	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 3 (LCDCE3)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB4_H$	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 4 (LCDCE4)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB5_H$	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W	: Readable/writable (The read value is the same as the write value.)
R(RM1),W	: Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0/WX	: The read value is always "0". Writing a value to this bit has no effect on operation.
-	: Undefined bit

Figure 28.4-1 LCD Controller Registers (2/2)

LCDC enable register 5 (LCDCE5)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB6$_H$	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 6 (LCDCE6)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB7$_H$	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 7 (LCDCE7)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB8$_H$	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC blinking setting register 1 (LCDCB1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB9$_H$	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC blinking setting register 2 (LCDCB2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FBA$_H$	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

28.4.1 LCDC Control Register 1 (LCDCC1)

The LCDC control register 1 (LCDCC1) is used to set the clock, display mode, and power supply control.

■ LCDC Control Register 1 (LCDCC1)

Figure 28.4-2 LCDC Control Register 1 (LCDCC1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB0H	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

FP1	FP0	Frame period select bits	
		Frame period generated by main clock (CSS = 0)	Frame period generated by subclock (CSS = 1)
0	0	$2^{14} \times N/F_{CH}$ *1 $2^{13} \times N/F_{CH}$ *2 $2^{13} \times N/F_{CRH}$	$2^{6} \times N/F_{CL}$ $2^{6} \times N/F_{CRL}$
0	1	$2^{15} \times N/F_{CH}$ *1 $2^{14} \times N/F_{CH}$ *2 $2^{14} \times N/F_{CRH}$	$2^{7} \times N/F_{CL}$ $2^{7} \times N/F_{CRL}$
1	0	$2^{16} \times N/F_{CH}$ *1 $2^{15} \times N/F_{CH}$ *2 $2^{15} \times N/F_{CRH}$	$2^{8} \times N/F_{CL}$ $2^{8} \times N/F_{CRL}$
1	1	$2^{17} \times N/F_{CH}$ *1 $2^{16} \times N/F_{CH}$ *2 $2^{16} \times N/F_{CRH}$	$2^{9} \times N/F_{CL}$ $2^{9} \times N/F_{CRL}$

MS2	MS1	MS0	Display mode select bits
0	0	0	Stops LCD operation.
0	0	1	4 COM mode 1/2 duty output mode (Time division number N = 2)
0	1	0	4 COM mode 1/3 duty output mode (Time division number N = 3)
0	1	1	4 COM mode 1/4 duty output mode (Time division number N = 4)
1	0/1	0/1	8 COM mode 1/8 duty output mode (Time division number N = 8)

VSEL	LCD drive power supply control bit
0	Uses external divider resistors.
1	Uses internal divider resistors.

LCDEN	Main stop/watch mode operation enable bit
0	Disables the operation.
1	Enables the operation.

CSS	Frame period generation clock select bit
0	Main clcok
1	Subclock

*1	:	Main clock mode
*2	:	Main PLL clock mode
F_{CH}	:	Main clock
F_{CL}	:	Subclock
F_{CRH}	:	Main CR clock
F_{CRL}	:	Sub-CR clock
R/W	:	Readable/writable (The read value is the same as the write value.)
▨	:	Initial value

Table 28.4-1 Functions of Bits in LCDC Control Register 1 (LCDCC1)

Bit name		Function
bit7	CSS: Frame period generation clock select bit	This bits selects the clock for generating the frame period for LCD display. • When this bit is "0", the LCD controller operates with the output of the time-base timer driven by the main clock. When the bit is "1", the LCD controller operates with the output of the watch prescaler driven by the subclock. Note: As the main clock stops oscillation in main stop mode and subclock mode, the LCD controller cannot operate with the output of the time-base timer in these modes. Shifting the main clock speed (using the gear function) during operation with the time-base timer output does not affect the frame period. LCD display may flicker when the clock speed is being shifted. Before shifting it, therefore, temporarily halt the display, for example, by using blanking (LCDCC2:BK = 1).
bit6	LCDEN: Main stop/watch mode operation enable bit	This bit specifies whether the LCD controller is to continue to operate in main stop mode and watch mode. **Writing "0":** Stops the LCD controller. **Writing "1":** Makes the LCD controller continue to operate even after the clock mode transits to main stop mode or watch mode. Note: In the case of making the LCD controller continue to operate in main stop mode or watch mode, select the subclock as the clock for generating the frame period for the LCD display (CSS = 1).
bit5	VSEL: LCD driving power control bit	This bit selects whether to energize the internal divider resistors. **Writing "0":** Disconnects the internal divider resistors. **Writing "1":** Energizes the internal divider resistors. Note: Write "0" to this bit when connecting to the external divider resistor.
bit4 to bit2	MS2, MS1, MS0: Display mode select bits	These bits select the display mode from 4 COM mode and 8 COM mode and also select an output waveform duty from four options. • The common output pin to be used is determined by the duty output mode selected. • When these bits are "000_B", the LCD controller driver stops the LCD display operation. Note: If the selected frame period generation clock can be halted, for example, upon transition to stop mode, halt the LCD display operation (MS2, MS1, MS0 = 000_B) in advance. As the LCD display may flicker when the display mode changes, halt the display temporarily, for example, by using blanking (LCDCC2:BK = 1) before changing the display mode.
bit1, bit0	FP1, FP0: Frame period select bits	This bit selects an LCD display frame period from four options. Note: Set these bits according to the optimum frame period for the LCD module to be used. The frame period is affected by the source oscillation frequency. As the LCD display may flicker when the frame period changes, halt the display temporarily, for example, by using blanking (LCDCC2:BK = 1) before changing the frame period.

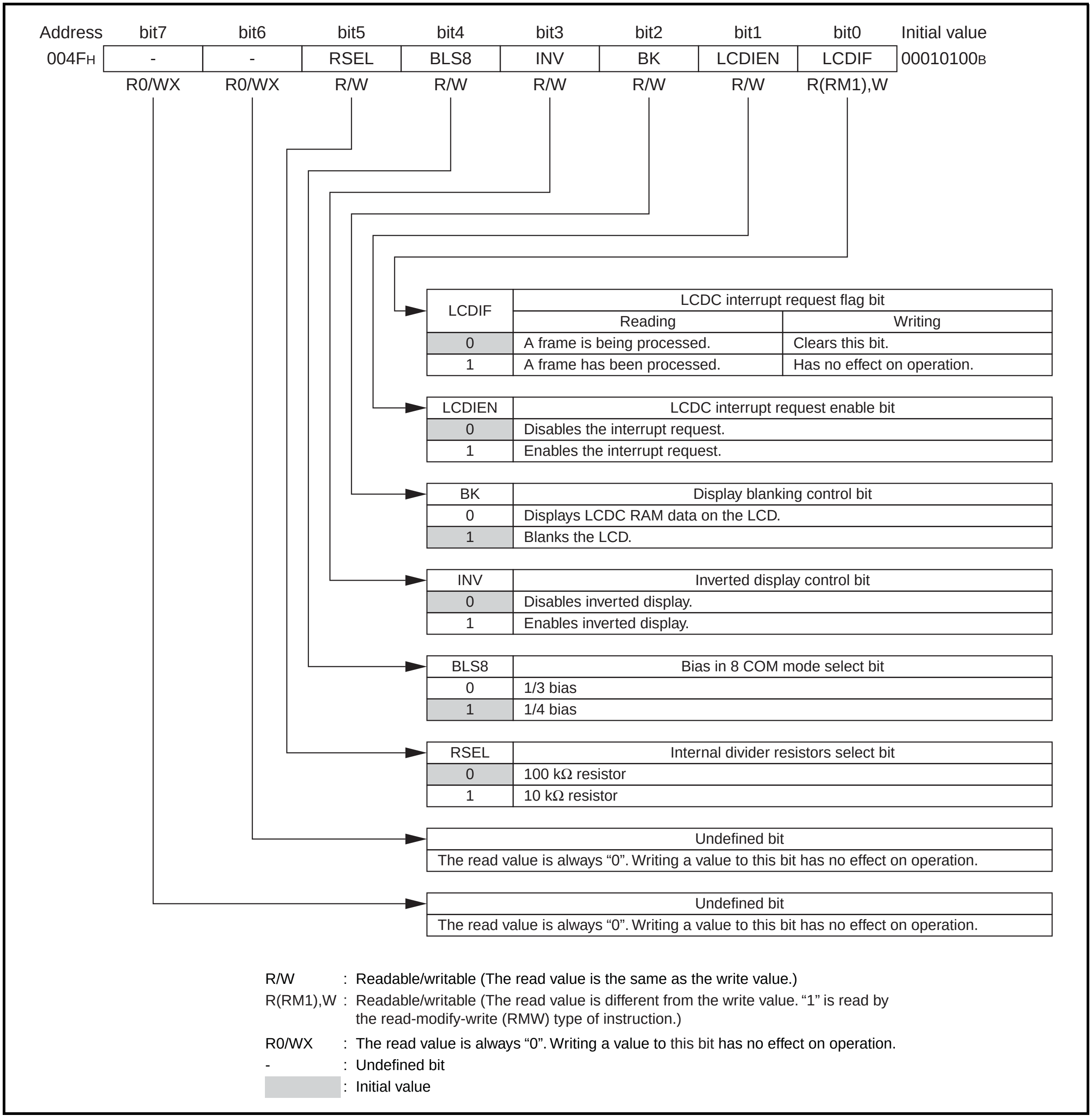
28.4.2 LCDC Control Register 2 (LCDCC2)

The LCDC control register 2 (LCDCC2) is used to enable and disable interrupts, indicate interrupt status and set the following parameters:

- **Internal resistance value from 10 kΩ or 100 kΩ**
- **Bias to be used in 8 COM mode from 1/3 or 1/4**
- **Displaying data or a blank screen**
- **Inverted display**

■ LCDC Control Register 2 (LCDC2)

Figure 28.4-3 LCDC Control Register 2 (LCDC2)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
004FH	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF	00010100B
	R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R(RM1),W	

LCDIF	LCDC interrupt request flag bit	
	Reading	Writing
0	A frame is being processed.	Clears this bit.
1	A frame has been processed.	Has no effect on operation.

LCDIEN	LCDC interrupt request enable bit
0	Disables the interrupt request.
1	Enables the interrupt request.

BK	Display blanking control bit
0	Displays LCDC RAM data on the LCD.
1	Blanks the LCD.

INV	Inverted display control bit
0	Disables inverted display.
1	Enables inverted display.

BLS8	Bias in 8 COM mode select bit
0	1/3 bias
1	1/4 bias

RSEL	Internal divider resistors select bit
0	100 kΩ resistor
1	10 kΩ resistor

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W	:	Readable/writable (The read value is the same as the write value.)
R(RM1),W	:	Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R0/WX	:	The read value is always “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

Table 28.4-2 Functions of Bits in LCDC Control Register 2 (LCDCC2)

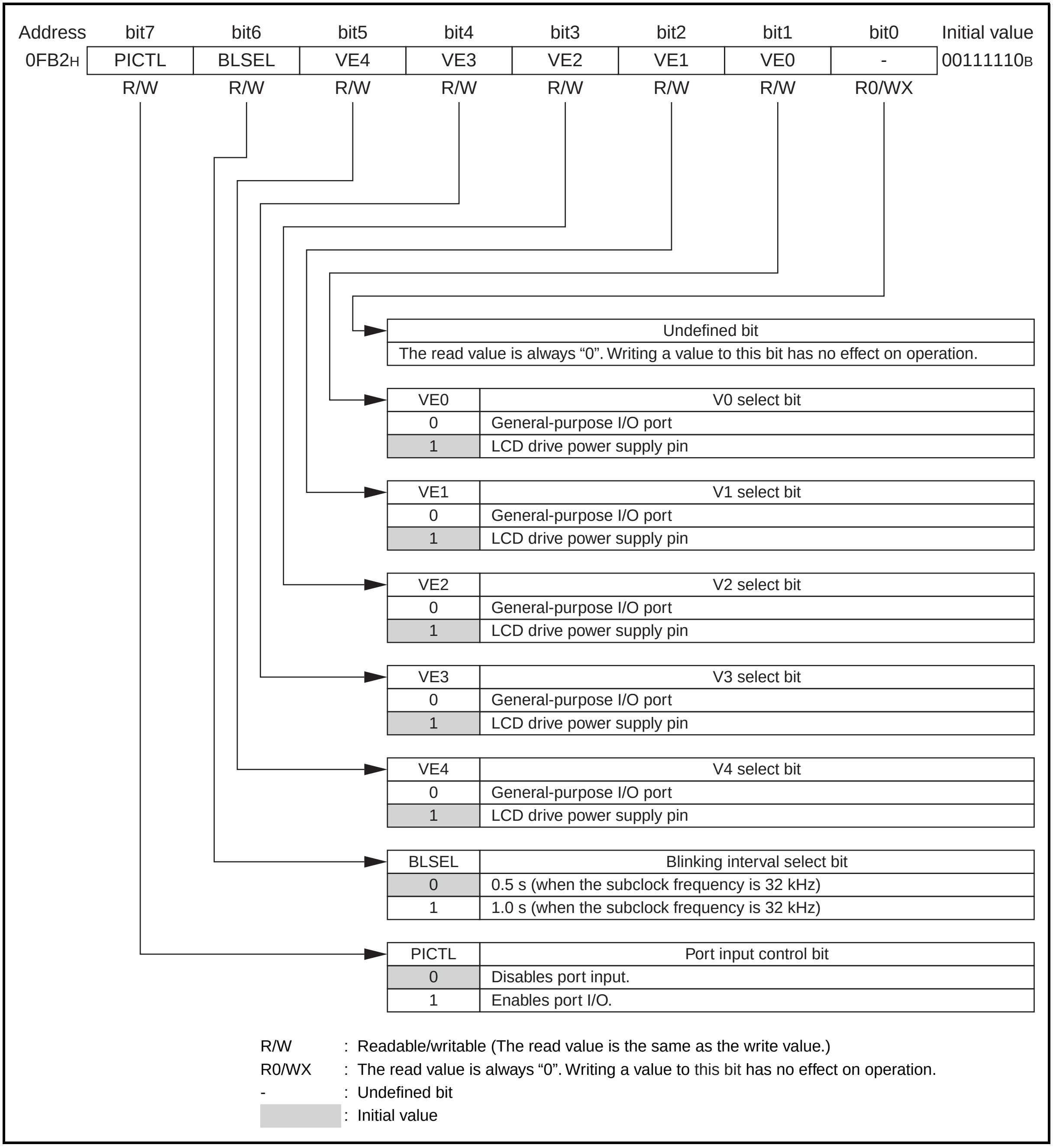
Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5	RSEL: Internal divider resistor select bit	This bit selects which type of resistors is to be used as internal divider resistors. **Writing "0":** Selects the 100 kΩ resistor. **Writing "1":** Selects the 10 kΩ resistor.
bit4	BLS8: Bias in 8 COM mode select bit	This bit selects which type of bias is to be used by software in 8 COM mode. **Writing "0":** Selects 1/3 bias. **Writing "1":** Selects 1/4 bias. Note: Although this bit can be accessed in both 8 COM mode and 4 COM mode, writing a value to this bit in 4 COM mode has no effect on operation.
bit3	INV: Inverted display control bit	This bit controls the inverted display on the LCD. **Writing "0":** Disables inverted display. **Writing "1":** Enables inverted display.
bit2	BK: Display blanking control bit	This bit controls display blanking of the LCD. **Writing "0":** Displays LCDC RAM data on the LCD. **Writing "1":** Blanks the LCD. When display blanking is selected (BK = 1), a segment output pin outputs a waveform not selected for displaying data on the LCD.
bit1	LCDIEN: LCDC interrupt request enable bit	This bit enables or disables the generation of an interrupt in sync with the LCD module frame frequency. **Writing "0":** Disables the interrupt request. **Writing "1":** Enables the interrupt request.
bit0	LCDIF: LCDC interrupt request flag bit	This bit indicates whether the LCD controller has finished processing a frame. **Reading "0":** Indicates that the LCD controller is processing a frame. **Reading "1":** Indicates that the LCD controller has finished processing a frame. **Writing "0":** Clears this bit. **Writing "1":** Has no effect on operation. This bit always returns "1" when read by a read-modify-write (RMW) type of instruction.

28.4.3 LCDC Enable Register 1 (LCDCE1)

The LCDC enable register 1 (LCDCE1) is used to control port input, set the blink cycle, and enable LCD pins.

■ LCDC Enable Register 1 (LCDCE1)

Figure 28.4-4 LCDC Enable Register 1 (LCDCE1)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB2H	PICTL	BLSEL	VE4	VE3	VE2	VE1	VE0	-	00111110B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0/WX	

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

VE0	V0 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

VE1	V1 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

VE2	V2 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

VE3	V3 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

VE4	V4 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

BLSEL	Blinking interval select bit
0	0.5 s (when the subclock frequency is 32 kHz)
1	1.0 s (when the subclock frequency is 32 kHz)

PICTL	Port input control bit
0	Disables port input.
1	Enables port I/O.

R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is always “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
▨ : Initial value

Table 28.4-3 Functions of Bits in LCDC Enable Register 1 (LCDCE1)

Bit name		Function
bit7	PICTL: Port input control bit	This bit controls general-purpose I/O ports that also function as segment or common output pins. **Writing "0":** Disables the input function of such general-purpose I/O ports and suppresses shoot-through current during LCD output. In addition, writing "0" to PICTL also disables the output function of such general-purpose I/O ports. **Writing "1":** Enables the I/O function of such general-purpose I/O ports. To use a segment or common output pin as a general-purpose I/O port, write "1" to PICTL. Note: As the input function of such general-purpose I/O ports will be disabled on a reset, in order to use their input function, write "1" to PICTL. When they are used as segment or common output pins, their input function will be disabled regardless of the setting of this bit.
bit6	BLSEL: Blinking interval select bit	This bit selects the blinking interval to be used when blinking is enabled. Blinking is to be enabled by the LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2). A blinking interval of 1.0 s makes the LCD stay on for 0.5 s and off for 0.5 s; a blinking interval of 0.5 s makes the LCD stay on for 0.25 s and off for 0.25 s.
bit5	VE4: V4 select bit	This bit selects the function of the V4 pin. **Writing "0":** Makes the V4 pin function as a general-purpose I/O port. **Writing "1":** Makes the V4 pin function as an LCD drive power supply pin.
bit4	VE3: V3 select bit	This bit selects the function of the V3 pin. **Writing "0":** Makes the V3 pin function as a general-purpose I/O port. **Writing "1":** Makes the V3 pin function as an LCD drive power supply pin.
bit3	VE2: V2 select bit	This bit selects the function of the V2 pin. **Writing "0":** Makes the V2 pin function as a general-purpose I/O port. **Writing "1":** Makes the V2 pin function as an LCD drive power supply pin.
bit2	VE1: V1 select bit	This bit selects the function of the V1 pin. **Writing "0":** Makes the V1 pin function as a general-purpose I/O port. **Writing "1":** Makes the V1 pin function as an LCD drive power supply pin.
bit1	VE0: V0 select bit	This bit selects the function of the V0 pin. **Writing "0":** Makes the V0 pin function as a general-purpose I/O port. **Writing "1":** Makes the V0 pin function as an LCD drive power supply pin.
bit0	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.

Note:

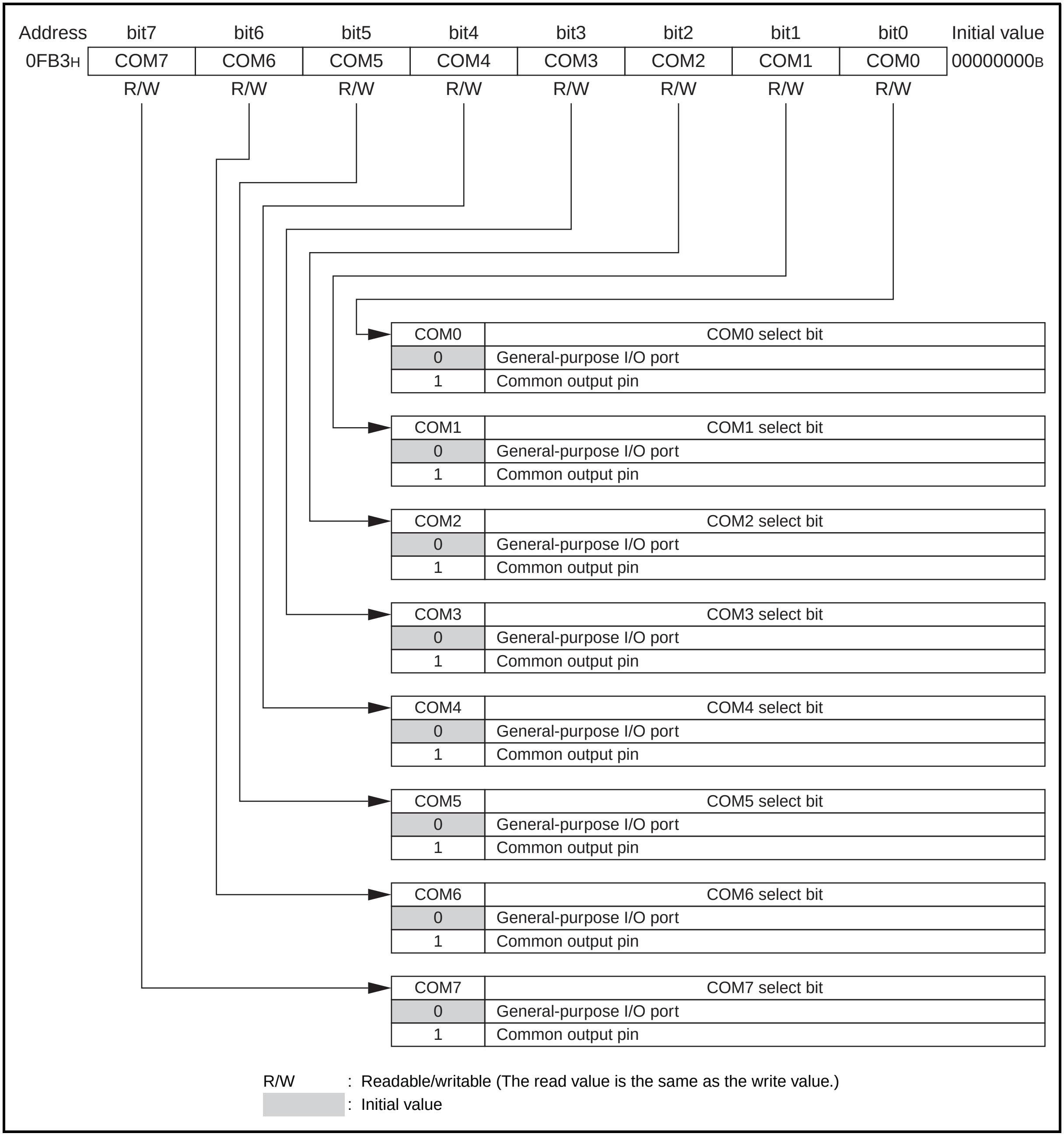
In the case of using the internal divider resistor, since the V4 pin cannot be used as a general-purpose I/O port, write "1" to the VE4 bit to make the V4 pin function as an LCD controller drive power supply pin.

28.4.4 LCDC Enable Register 2 (LCDCE2)

The LCDC enable register 2 (LCDCE2) is used to control the output of COM0 to COM7.

■ LCDC Enable Register 2 (LCDCE2)

Figure 28.4-5 LCDC Enable Register 2 (LCDCE2)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB3H	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

COM0	COM0 select bit
0	General-purpose I/O port
1	Common output pin

COM1	COM1 select bit
0	General-purpose I/O port
1	Common output pin

COM2	COM2 select bit
0	General-purpose I/O port
1	Common output pin

COM3	COM3 select bit
0	General-purpose I/O port
1	Common output pin

COM4	COM4 select bit
0	General-purpose I/O port
1	Common output pin

COM5	COM5 select bit
0	General-purpose I/O port
1	Common output pin

COM6	COM6 select bit
0	General-purpose I/O port
1	Common output pin

COM7	COM7 select bit
0	General-purpose I/O port
1	Common output pin

R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value

Table 28.4-4 Functions of Bits in LCDC Enable Register 2 (LCDCE2)

Bit name		Function
bit7	COM7: COM7 select bit	This bit selects the function of the COM7 pin. In 8 COM mode: **Writing "0":** Makes the COM7 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM7 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit6	COM6: COM6 select bit	This bit selects the function of the COM6 pin. In 8 COM mode: **Writing "0":** Makes the COM6 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM6 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit5	COM5: COM5 select bit	This bit selects the function of the COM5 pin. In 8 COM mode: **Writing "0":** Makes the COM5 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM5 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit4	COM4: COM4 select bit	This bit selects the function of the COM4 pin. In 8 COM mode: **Writing "0":** Makes the COM4 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM4 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit3	COM3: COM3 select bit	This bit selects the function of the COM3 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM3 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM3 pin function as a common output pin.
bit2	COM2: COM2 select bit	This bit selects the function of the COM2 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM2 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM2 pin function as a common output pin.
bit1	COM1: COM1 select bit	This bit selects the function of the COM1 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM1 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM1 pin function as a common output pin.
bit0	COM0: COM0 select bit	This bit selects the function of the COM0 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM0 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM0 pin function as a common output pin.

28.4.5 LCDC Enable Register 3 to LCDC Enable Register 6 (LCDCE3 to LCDCE6)

The LCDC enable register 3 to the LCDC enable register 6 (LCDCE3 to LCDCE6) are used to control segment output pins SEG00 to SEG31.

■ LCDC Enable Register 3 to LCDCE Enable Register 6 (LCDCE3 to LCDCE6)

Figure 28.4-6 LCDC Enable Register 3 to LCDCE Register 6 (LCDCE3 to LCDCE6)

LCDC enable register 3 (LCDCE3)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB4_H$	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 4 (LCDCE4)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB5_H$	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 5 (LCDCE5)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB6_H$	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 6 (LCDCE6)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB7_H$	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

SEGn*	SEGn* select bit
0	General-purpose I/O port
1	Segment output pin

▨ : Initial value

* : The letter “n” after SEG represents the number appearing in the bit name.

Note:

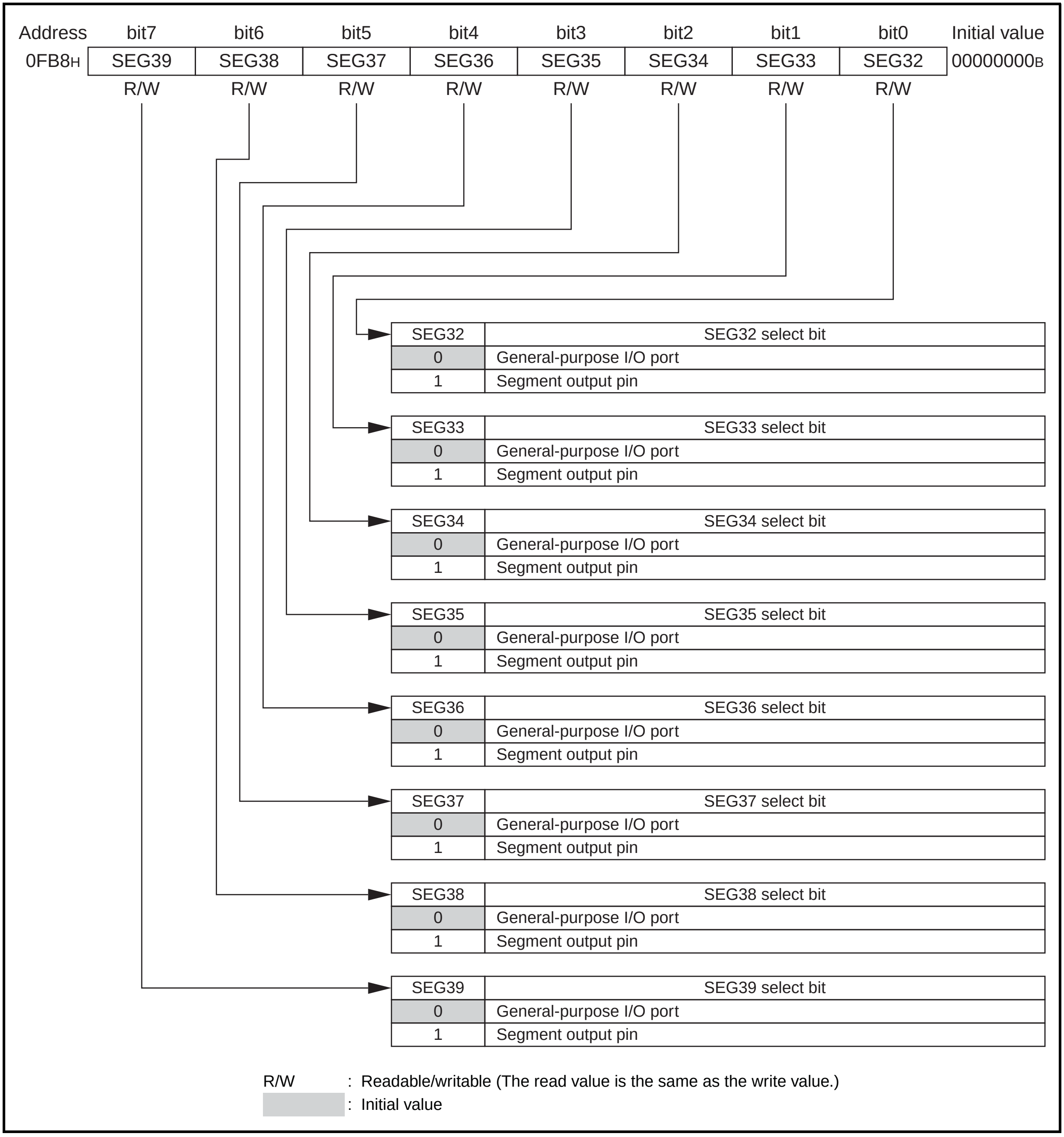
Only when PICTL is set to "1" are LCDCE3 to LCDCE6 enabled to control their corresponding segment output pins.

28.4.6 LCDCE Enable Register 7 (LCDCE7)

The LCDC enable register 7 (LCDCE7) is used to control segment output pins SEG32 to SEG39.

■ LCDC Enable Register 7 (LCDCE7)

Figure 28.4-7 LCDC Enable Register 7 (LCDCE7)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB8H	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

SEG32	SEG32 select bit
0	General-purpose I/O port
1	Segment output pin

SEG33	SEG33 select bit
0	General-purpose I/O port
1	Segment output pin

SEG34	SEG34 select bit
0	General-purpose I/O port
1	Segment output pin

SEG35	SEG35 select bit
0	General-purpose I/O port
1	Segment output pin

SEG36	SEG36 select bit
0	General-purpose I/O port
1	Segment output pin

SEG37	SEG37 select bit
0	General-purpose I/O port
1	Segment output pin

SEG38	SEG38 select bit
0	General-purpose I/O port
1	Segment output pin

SEG39	SEG39 select bit
0	General-purpose I/O port
1	Segment output pin

R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value

Table 28.4-5 Functions of Bits in LCDC Enable Register 7 (LCDCE7)

Bit name		Function
bit7	SEG39: SEG39 select bit	This bit selects the function of the SEG39 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG39 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG39 pin function as a segment output pin.
bit6	SEG38: SEG38 select bit	This bit selects the function of the SEG38 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG38 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG38 pin function as a segment output pin.
bit5	SEG37: SEG37 select bit	This bit selects the function of the SEG37 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG37 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG37 pin function as a segment output pin.
bit4	SEG36: SEG36 select bit	This bit selects the function of the SEG36 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG36 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG36 pin function as a segment output pin.
bit3	SEG35: SEG35 select bit	This bit selects the function of the SEG35 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG35 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG35 pin function as a segment output pin.
bit2	SEG34: SEG34 select bit	This bit selects the function of the SEG34 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG34 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG34 pin function as a segment output pin.
bit1	SEG33: SEG33 select bit	This bit selects the function of the SEG33 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG33 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG33 pin function as a segment output pin.
bit0	SEG32: SEG32 select bit	This bit selects the function of the SEG32 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG32 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG32 pin function as a segment output pin.

Note:

Only when PICTL is set to "1" is LCDCE7 enabled to control its corresponding segment output pins.

28.4.7 LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2)

The LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2) are used to turn on or off blinking.

■ LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2)

Figure 28.4-8 LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2)

LCDC blinking setting register 1 (LCDCB1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB9_H$	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC blinking setting register 2 (LCDCB2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FBA_H$	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

BLDx*1	SnCm*2 blinking setting bit
0	Turns off the blinking of SnCm*2.
1	Turns on the blinking of SnCm*2.

▨ : Initial value

*1 : The letter “x” after BLD represents the number (0 to 15) appearing in the bit name.

*2 : Sn = SEGn (“n” represents one of the numbers from 00 to 03.)
Cm = COMm (“m” represents one of the numbers from 0 to 7.)

In 8 COM mode, the blinking function is applied to the dots specified in the combinations of SEG00 to SEG01 and COM0 to COM7.

In 4 COM mode, the blinking function is applied to the dots specified in the combinations of SEG00 to SEG03 and COM0 to COM3.

Select a blinking interval using the BLSEL bit in the LCDC enable register 1 (LCDCE1).

All segments for which blinking has been turned on will blink synchronously.

The setting of each blinking select bit remains in effect even when its corresponding bit in the display RAM holds "1".

28.5 LCD Controller Display RAM

**The display RAM size varies between 8 COM mode and 4 COM mode.
In 8 COM mode, the display RAM has 36 × 8 bits (36 bytes) of display data memory for generating segment output signals.
In 4 COM mode, the display RAM has 40 × 4 bits (20 bytes) of display data memory for generating segment output signals.**

■ Display RAM and Output Pins

The contents of display RAM are read automatically in sync with the common signal selection timing and output from the segment output pins.

Each bit containing "1" is converted to the selected voltage (displayed on the LCD); the one containing "0" is converted to the unselected voltage (undisplayed on the LCD).

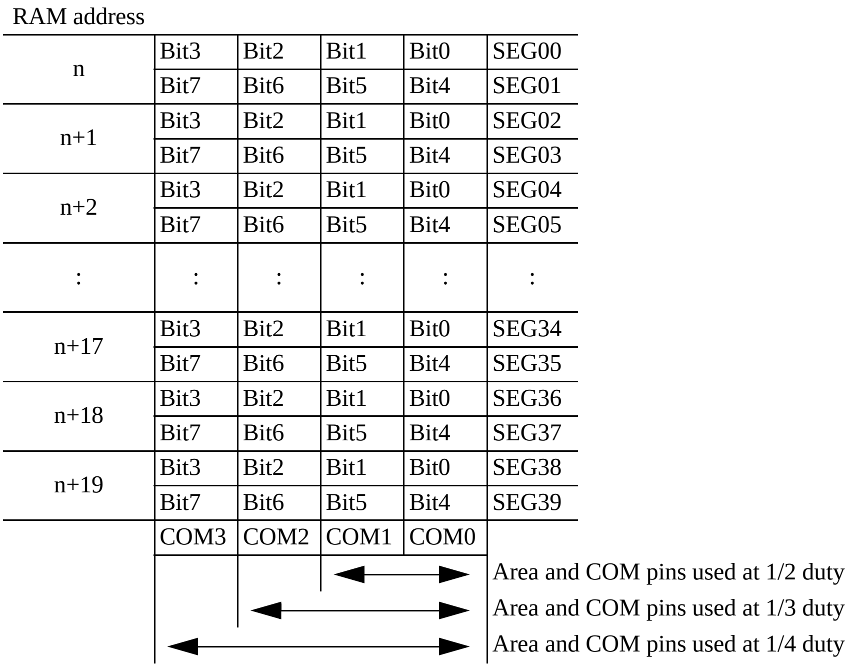
As the LCD display operation is performed asynchronously with the CPU operation, data can be read from or written to the display RAM at any timing. When a pin shared between a segment output pin and a general-purpose I/O port is not used as a segment output pin, the pin can be used as a general-purpose I/O port, and the display RAM corresponding to such pin can be used as normal RAM. Table 28.5-1 shows the relationship between duty setting/common outputs and bits used in the display RAM.

Figure 28.5-1 and Figure 28.5-2 shows how display RAM addresses are allocated for common output pins and segment output pins in 8 COM mode and in 4 COM mode respectively.

Figure 28.5-1 Display RAM and Common/Segment Output Pins in 8 COM Mode

RAM address									
n	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG00
n+1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG01
n+2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG02
n+3	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG03
n+4	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG04
:	:	:	:	:	:	:	:	:	:
n+30	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG30
n+31	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG31
n+32	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG32
n+33	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG33
n+34	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG34
n+35	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG35
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	
	◄──	──	──	──	──	──	──	──►	Area and COM pins used at 1/8 duty

Figure 28.5-2 Display RAM and Common/Segment output Pins in 4 COM Mode



RAM address					
n	Bit3	Bit2	Bit1	Bit0	SEG00
	Bit7	Bit6	Bit5	Bit4	SEG01
n+1	Bit3	Bit2	Bit1	Bit0	SEG02
	Bit7	Bit6	Bit5	Bit4	SEG03
n+2	Bit3	Bit2	Bit1	Bit0	SEG04
	Bit7	Bit6	Bit5	Bit4	SEG05
:	:	:	:	:	:
n+17	Bit3	Bit2	Bit1	Bit0	SEG34
	Bit7	Bit6	Bit5	Bit4	SEG35
n+18	Bit3	Bit2	Bit1	Bit0	SEG36
	Bit7	Bit6	Bit5	Bit4	SEG37
n+19	Bit3	Bit2	Bit1	Bit0	SEG38
	Bit7	Bit6	Bit5	Bit4	SEG39
	COM3	COM2	COM1	COM0	

Area and COM pins used at 1/2 duty

Area and COM pins used at 1/3 duty

Area and COM pins used at 1/4 duty

Note:

"n" in the address column represents "0FBD$_H$".

Table 28.5-1 Relationship Between Duty Settings/Common Outputs and Display RAM Bits Used

Duty setting	Common output pins used	Display data bits used							
		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
1/2	COM0, COM1 (2 pins)	-	-	❍	❍	-	-	❍	❍
1/3	COM0 to COM2 (3 pins)	-	❍	❍	❍	-	❍	❍	❍
1/4	COM0 to COM3 (4 pins)	❍	❍	❍	❍	❍	❍	❍	❍
1/8	COM0 to COM7 (8 pins)	❍	❍	❍	❍	❍	❍	❍	❍

❍ : Bit used
- : Bit not used

28.6 Interrupts of LCD Controller

The LCD controller generates interrupts in sync with the LCD module frame frequency.

■ Interrupt during LCD Controller Operation

Upon completing a frame, the LCD controller sets the LCDC interrupt request flag bit (LCDCC2:LCDIF) to "1". If the interrupt request has already been enabled (LCDCC2:LCDIEN = 1) when the LCDIF bit is set to "1", the LCD controller will make an interrupt request to the interrupt controller. To clear an interrupt request, write "0" to the LCDIF bit in the interrupt service routine.

The LCD controller always sets the LCDIF bit to "1" upon completing a frame, regardless of the value of the LCDIEN bit. If both the LCDIF bit and the LCDIEN bit remain "1" after an LCDC interrupt request is made, the CPU cannot return from interrupt processing. To enable the CPU to return from interrupt processing, always clear the LCDIF bit to "0" after an LCDC interrupt request is made.

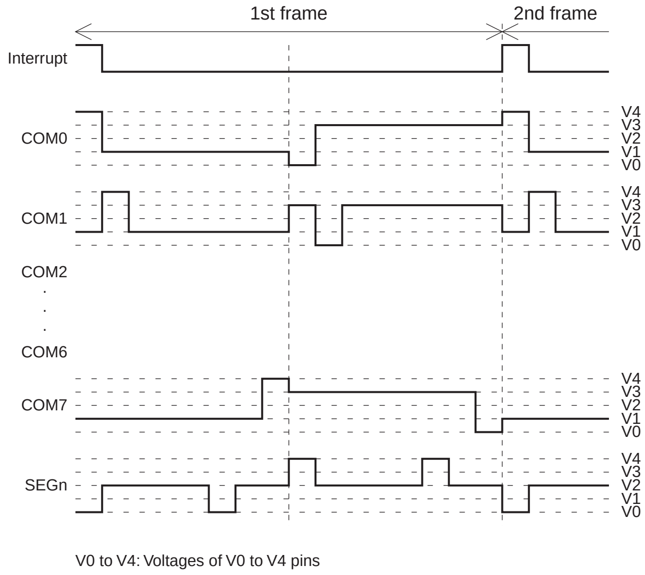
■ Register and Vector Table Addresses Related to LCD Controller Interrupts

Table 28.6-1 Register and Vector Table Addresses Related to LCD Controller Interrupts

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
LCD controller	IRQ08	ILR2	L08	FFEA_H	FFEB_H

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

Figure 28.6-1 Interrupt Timing Chart



28.7 Operations of LCD Controller

This section describes the operations of the LCD controller.

■ Operations of LCD Controller

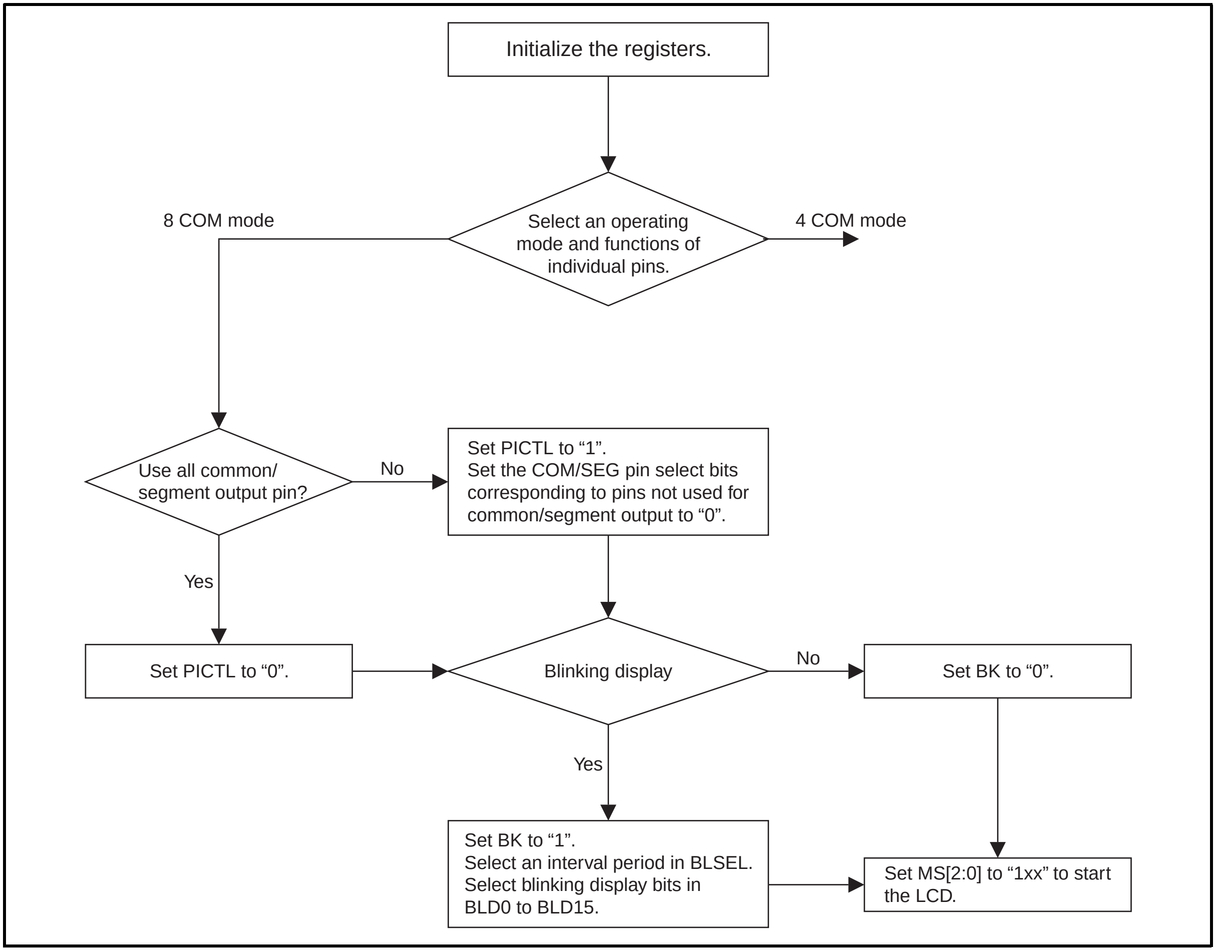
Figure 28.7-1 shows the settings required for LCD display in 8 COM mode.

Figure 28.7-1 LCD Controller Settings in 8 COM Mode

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
LCDCC1	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0
	❍	❍	❍	1	0/1	0/1	❍	❍
LCDCC2	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF
	-	-	❍	❍	❍	❍	❍	❍
LCDCE1	PICTL	BLSEL	VE4	VE3	VE2	VE1	VE0	-
	❍	❍	❍	❍	❍	❍	❍	-
LCDCE2	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE3	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE4	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE5	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE6	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE7	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
	-	-	-	-	❍	❍	❍	❍
LCDCB1	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCB2	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8
	❍	❍	❍	❍	❍	❍	❍	❍
Display RAM	Display data							

❍ : Bit used
- : Bit not used
1 : Write "1".
0/1 : Write "0" or "1".

Figure 28.7-2 LCD Controller Setting Flow in 8 COM Mode


Initialize the registers.
8 COM mode
Select an operating mode and functions of individual pins.
4 COM mode
Use all common/ segment output pin?
No
Set PICTL to “1”. Set the COM/SEG pin select bits corresponding to pins not used for common/segment output to “0”.
Yes
Set PICTL to “0”.
Blinking display
No
Set BK to “0”.
Yes
Set BK to “1”. Select an interval period in BLSEL. Select blinking display bits in BLD0 to BLD15.
Set MS[2:0] to “1xx” to start the LCD.

- When the selected frame period generation clock is oscillating with the settings made as shown in Figure 28.7-1, the LCD controller outputs the LCD panel drive waveform to the common and segment output pins (COM0 to COM7, SEG00 to SEG35) according to the contents of the display RAM and the settings of different LCDC registers.
- The LCD output pins are selected according to LCDCE1 to LCDCE7. Pins not selected as common/segment output pins are used as general-purpose I/O ports.
- The frame period generation clock can be changed even during LCD display operation. As the LCD display may flicker when the frame period generation clock is changed, always turn off the LCD display temporarily, for example, using the display blanking (LCDCC2:BK = 1) function, before changing the frame period generation clock to prevent the LCD display from flickering.
- The display drive output is a 2-frame alternating waveform selected according to bias and duty settings.
- To use the blink function, set the corresponding bits in the LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2) to "1" to turn on the blinking function. The blinking interval can be selected from two options in the BLSEL bit in the LCDC enable register 1 (LCDCE1).

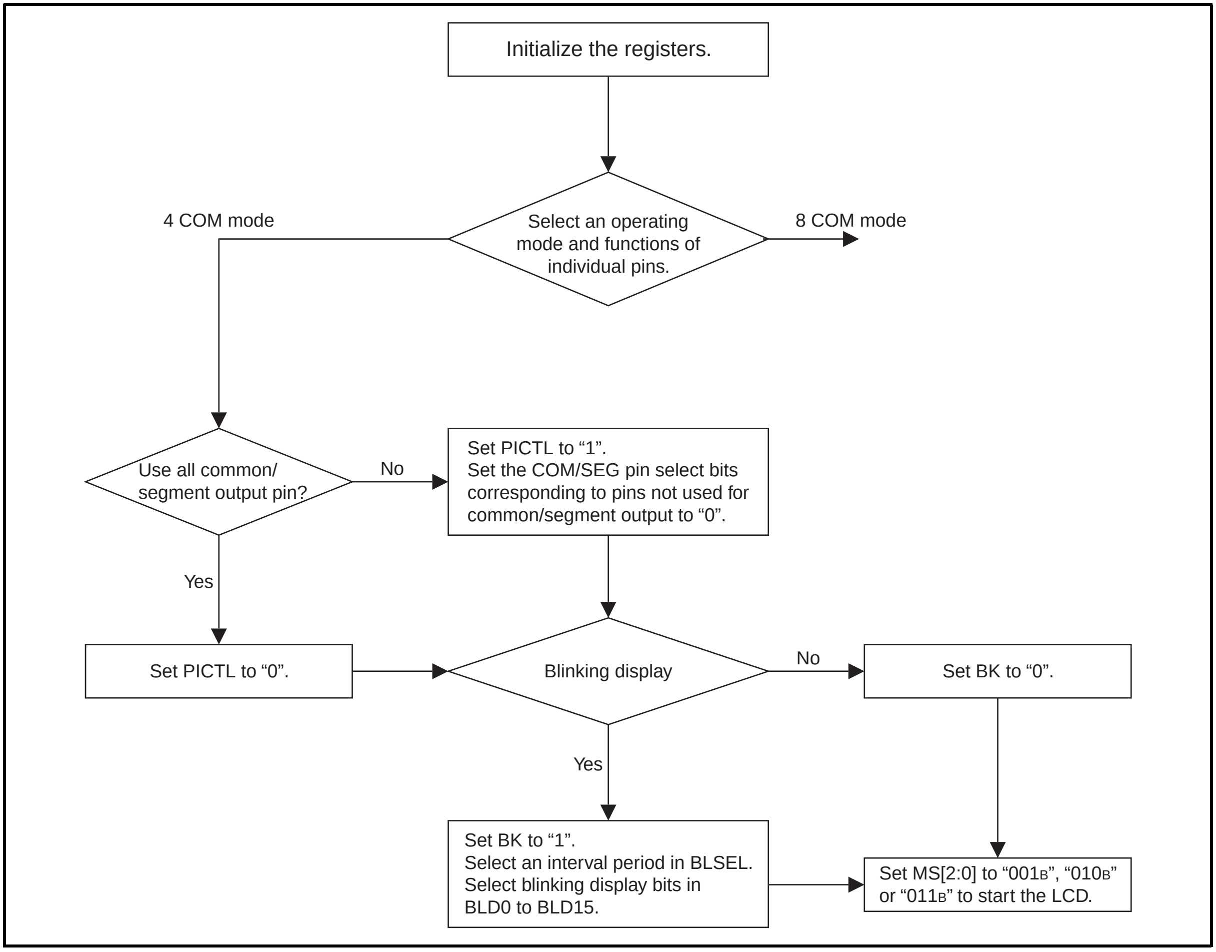
Figure 28.7-3 shows the settings required for LCD display in 4 COM mode.

Figure 28.7-3 LCD Controller Settings in 4 COM Mode

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
LCDCC1	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0
	❍	❍	❍	$001_B/010_B/011_B$			❍	❍
LCDCC2	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF
	-	-	❍	❍	❍	❍	❍	❍
LCDCE1	PICTL	BLSEL	VE4	VE3	VE2	VE1	VE0	-
	❍	❍	❍	❍	❍	❍	❍	-
LCDCE2	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	-	-	-	-	❍	❍	❍	❍
LCDCE3	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE4	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE5	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE6	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE7	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCB1	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCB2	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8
	❍	❍	❍	❍	❍	❍	❍	❍
Display RAM	Display data							

❍ : Bit used
- : Bit not used.

Figure 28.7-4 LCD Controller Setting Flow in 4 COM Mode


Initialize the registers.
4 COM mode
Select an operating mode and functions of individual pins.
8 COM mode
Use all common/ segment output pin?
No
Set PICTL to “1”.
Set the COM/SEG pin select bits corresponding to pins not used for common/segment output to “0”.
Yes
Set PICTL to “0”.
Blinking display
No
Set BK to “0”.
Yes
Set BK to “1”.
Select an interval period in BLSEL.
Select blinking display bits in BLD0 to BLD15.
Set MS[2:0] to “001B”, “010B” or “011B” to start the LCD.

- When the selected frame period generation clock is oscillating with the settings made as shown in Figure 28.7-3, the LCD controller outputs the LCD panel drive waveform to the common and segment output pins (COM0 to COM3, SEG00 to SEG39) according to the contents of the display RAM and the settings of different LCDC registers.
- The LCD output pins are selected according to LCDCE1 to LCDCE7. Pins not selected as common/segment output pins are used as general-purpose I/O ports.
- The frame period generation clock can be changed even during LCD display operation. As the LCD display may flicker when the frame period generation clock is changed, always turn off the LCD display temporarily, for example, using the display blanking (LCDCC2:BK = 1) function, before changing the frame period generation clock to prevent the LCD display from flickering.
- The display drive output is a 2-frame alternating waveform selected according to bias and duty settings.
- The COM2 and COM3 pin outputs in 1/2 duty mode and the COM3 pin output in 1/3 duty mode can be used to output the deselected level waveform or as I/O ports.
- To use the blink function, set the corresponding bits in the LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2) to "1" to turn on the blinking function. The blinking interval can be selected from two options in the BLSEL bit in the LCDC enable register 1 (LCDCE1).

Note:

If the selected frame period generation clock halts during LCD display operation, the AC waveform generator circuit also halts and therefore a DC voltage is applied to the liquid crystal elements. In this case, the LCD display operation must be stopped in advance. The conditions under which the main clock (time-base timer) or subclock (watch prescaler) halts depend on the selected clock mode and standby mode. The frame period is also affected if the time-base timer or watch prescaler is cleared depending on the setting of the frame period generation clock select bit (LCDCC1:CSS).

■ LCD Drive Waveform

Due to the characteristics of the LCD, DC driving of the LCD chemically changes and degrades the liquid crystal display elements. Therefore, the LCD controller driver contains an AC waveform generator circuit to drive the LCD using a 2-frame alternating waveform. There are five types of output waveform as follows:

In 8 COM mode:

- 1/4 bias, 1/8 duty output waveform
- 1/3 bias, 1/8 duty output waveform

In 4 COM mode:

- 1/2 bias, 1/2 duty output waveform
- 1/3 bias, 1/3 duty output waveform
- 1/3 bias, 1/4 duty output waveform

28.7.1 Output Waveform in LCD Controller Operation in 4 COM Mode (1/2 Bias, 1/2 Duty)

The display drive output is a multiplex drive type of 2-frame alternating waveform.
In 4 COM mode with 1/2 bias and 1/2 duty, only COM0 and COM1 are used for display; neither COM2 nor COM3 is used.

■ 4 COM Mode, 1/2 Bias, 1/2 Duty Output Waveform Example

Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

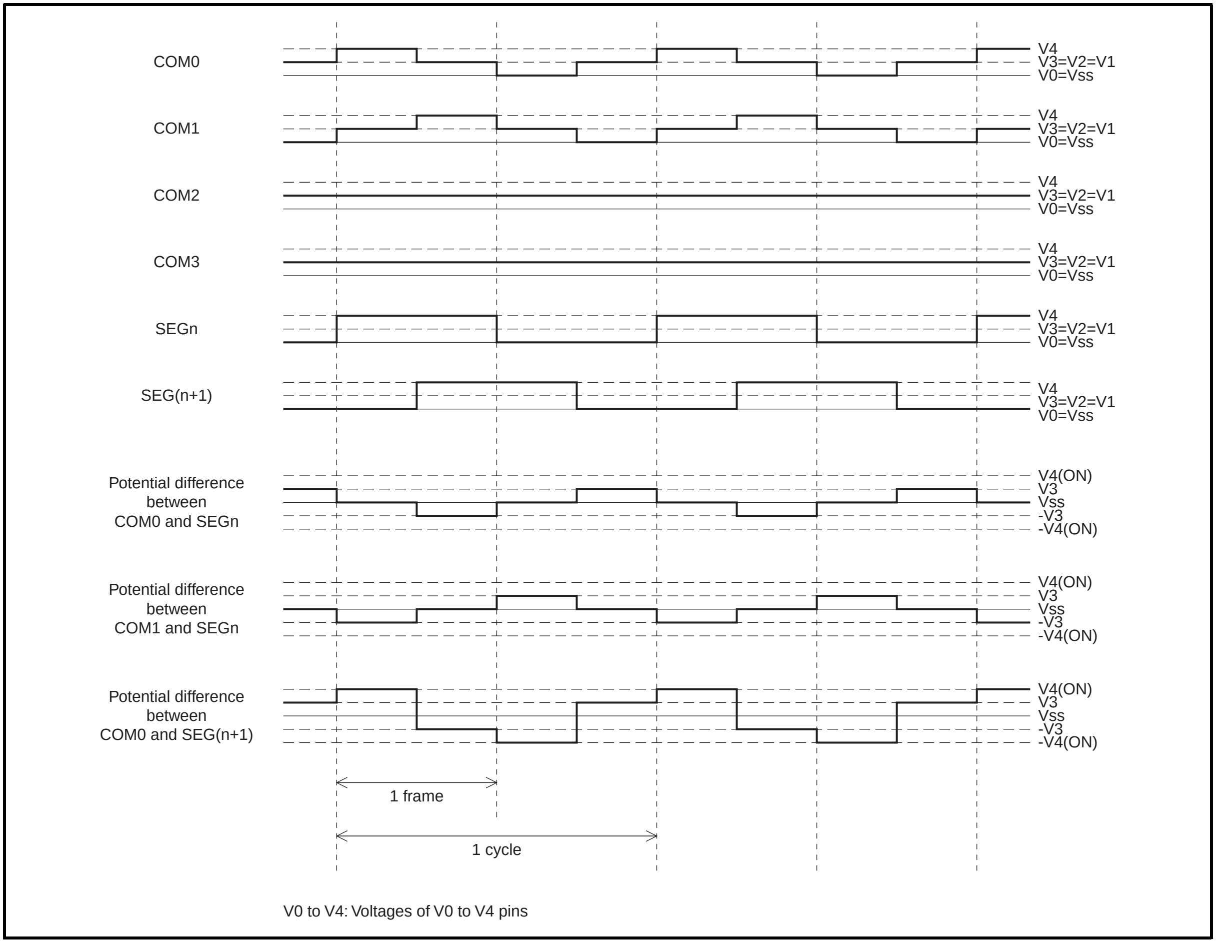
Figure 28.7-5 shows the output waveform when the contents of display RAM are those shown in Table 28.7-1.

Table 28.7-1 Sample Contents of Display RAM

Segment	Contents of Display RAM			
	COM3	COM2	COM1	COM0
SEGn	-	-	0	0
SEG(n+1)	-	-	0	1

-: Unused

Figure 28.7-5 4 COM Mode, 1/2 Bias, 1/2 Duty Output Waveform Example


COM0
COM1
COM2
COM3
SEGn
SEG(n+1)
Potential difference between COM0 and SEGn
Potential difference between COM1 and SEGn
Potential difference between COM0 and SEG(n+1)
V4
V3=V2=V1
V0=Vss
V4(ON)
V3
Vss
-V3
-V4(ON)
1 frame
1 cycle
V0 to V4: Voltages of V0 to V4 pins

28.7.2 Output Waveform in LCD Controller Operation in 4 COM Mode (1/3 Bias, 1/3 Duty)

In 4 COM mode with 1/3 bias and 1/3 duty, COM0, COM1, and COM2 are used for display; COM3 is not used.

■ 4 COM Mode, 1/3 Bias, 1/3 Duty Output Waveform Example

Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

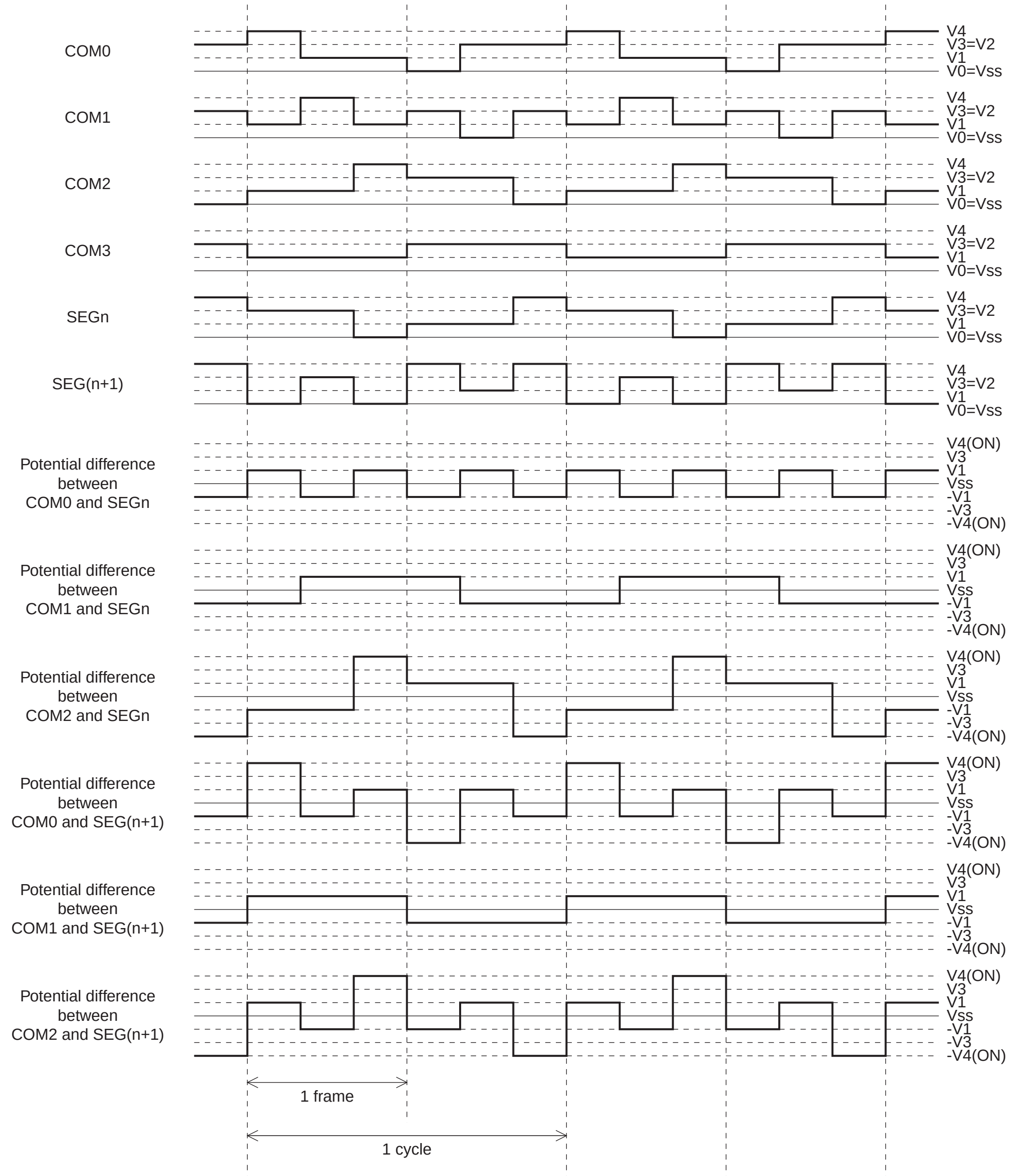
Figure 28.7-6 shows the output waveform when the contents of display RAM are those shown in Table 28.7-2.

Table 28.7-2 Sample Contents of Display RAM

Segment	Contents of Display RAM			
	COM3	COM2	COM1	COM0
SEGn	-	1	0	0
SEG(n+1)	-	1	0	1

-: Unused

Figure 28.7-6 4 COM Mode, 1/3 Bias, 1/3 Duty Output Waveform Example


COM0
COM1
COM2
COM3
SEGn
SEG(n+1)
Potential difference between COM0 and SEGn
Potential difference between COM1 and SEGn
Potential difference between COM2 and SEGn
Potential difference between COM0 and SEG(n+1)
Potential difference between COM1 and SEG(n+1)
Potential difference between COM2 and SEG(n+1)
V4
V3=V2
V1
V0=Vss
V4(ON)
V3
V1
Vss
-V1
-V3
-V4(ON)
1 frame
1 cycle

V0 to V4: Voltages of V0 to V4 pins

28.7.3 Output Waveform in LCD Controller Operation in 4 COM Mode (1/3 Bias, 1/4 Duty)

In 4 COM Mode with 1/3 bias and 1/4 duty, COM0 to COM3 are used for display.

■ 4 COM Mode, 1/3 Bias, 1/4 Duty Output Waveform Example

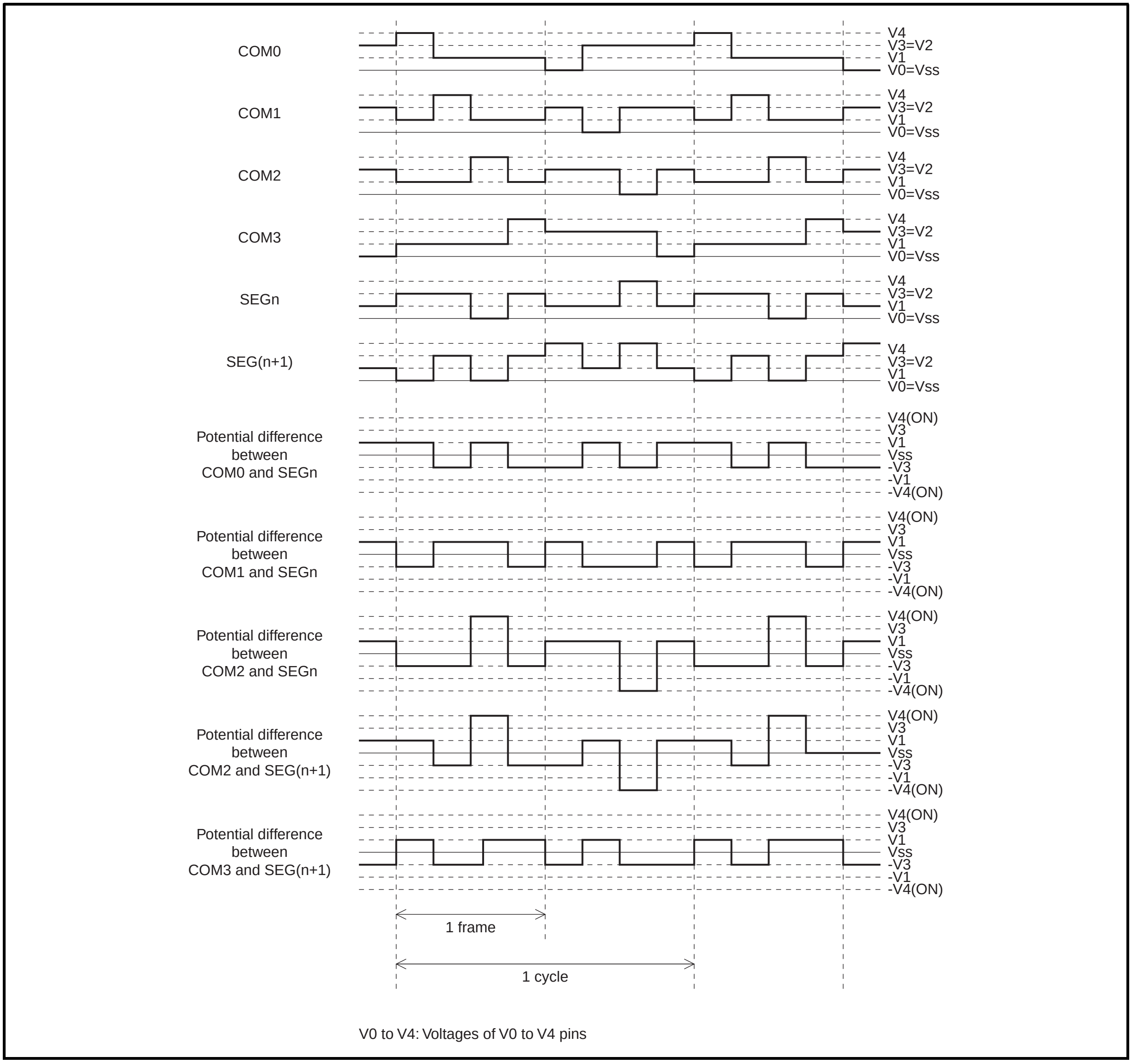
Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

Figure 28.7-7 shows the output waveform when the contents of display RAM are those shown in Table 28.7-3.

Table 28.7-3 Sample Contents of Display RAM

Segment	Contents of Display RAM			
	COM3	COM2	COM1	COM0
SEGn	0	1	0	0
SEG(n+1)	0	1	0	1

Figure 28.7-7 4 COM Mode, 1/3 Bias, 1/4 Duty Output Waveform Example


COM0
COM1
COM2
COM3
SEGn
SEG(n+1)
V4
V3=V2
V1
V0=Vss
Potential difference between COM0 and SEGn
Potential difference between COM1 and SEGn
Potential difference between COM2 and SEGn
Potential difference between COM2 and SEG(n+1)
Potential difference between COM3 and SEG(n+1)
V4(ON)
V3
V1
Vss
-V3
-V1
-V4(ON)
1 frame
1 cycle

V0 to V4: Voltages of V0 to V4 pins

28.7.4 Output Waveform in LCD Controller Operation in 8 COM Mode (1/4 Bias, 1/8 Duty)

In 8 COM Mode with 1/4 bias and 1/8 duty, COM0 to COM7 are used for display.

■ 8 COM Mode, 1/4 Bias, 1/8 Duty Output Waveform Example

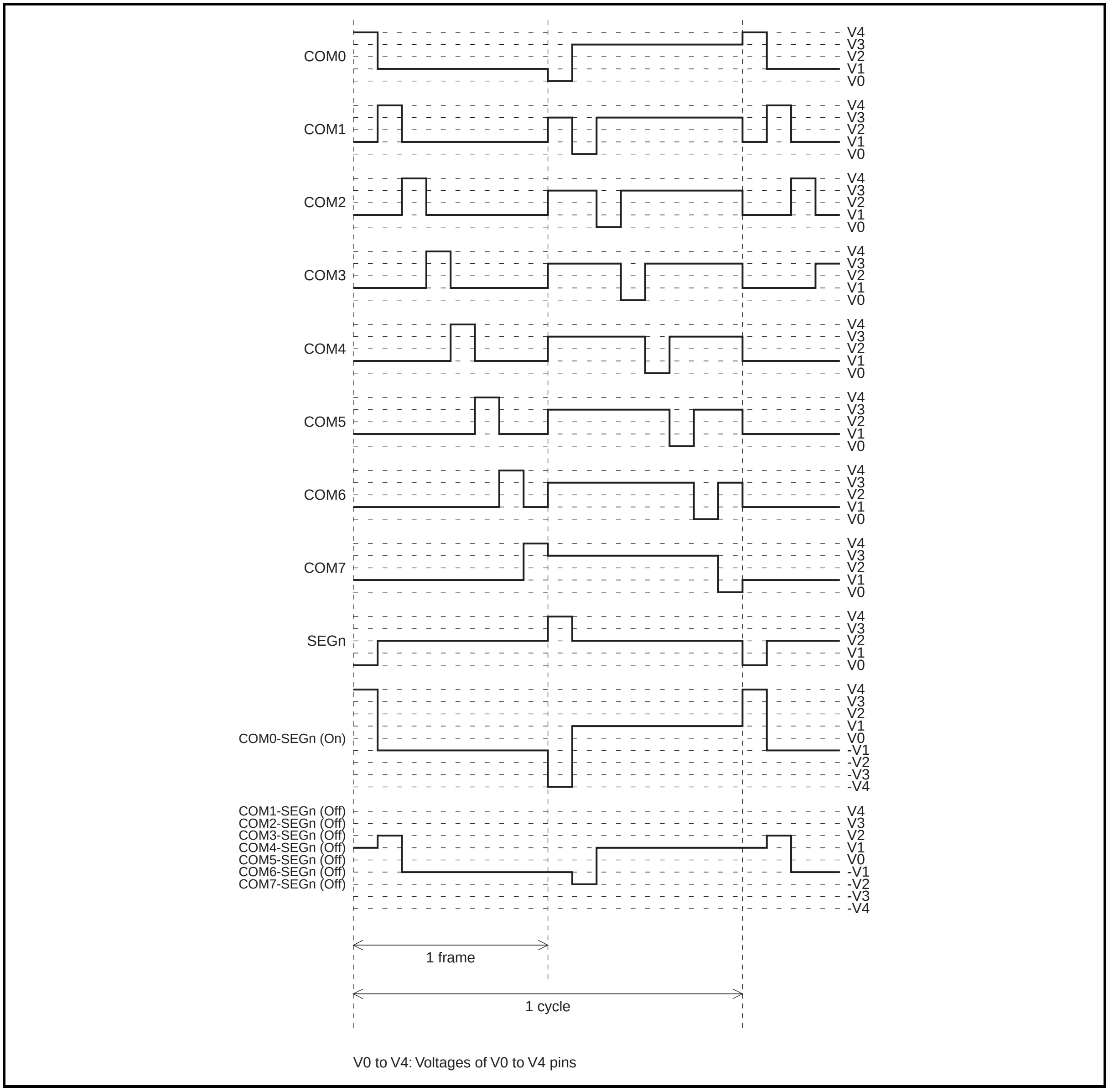
Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

Figure 28.7-8 shows the output waveform when the contents of display RAM are those shown in Table 28.7-4.

Table 28.7-4 Sample Contents of Display RAM

Segment	Contents of Display RAM							
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEGn	0	0	0	0	0	0	0	1

Figure 28.7-8 8 COM Mode, 1/4 Bias, 1/8 Duty Output Waveform Example



V0 to V4: Voltages of V0 to V4 pins

28.7.5 Output Waveform in LCD Controller Operation in 8 COM Mode (1/3 Bias, 1/8 Duty)

In 8 COM Mode with 1/3 bias and 1/8 duty, COM0 to COM7 are used for display.

■ 8 COM Mode, 1/3 Bias, 1/8 Duty Output Waveform Example

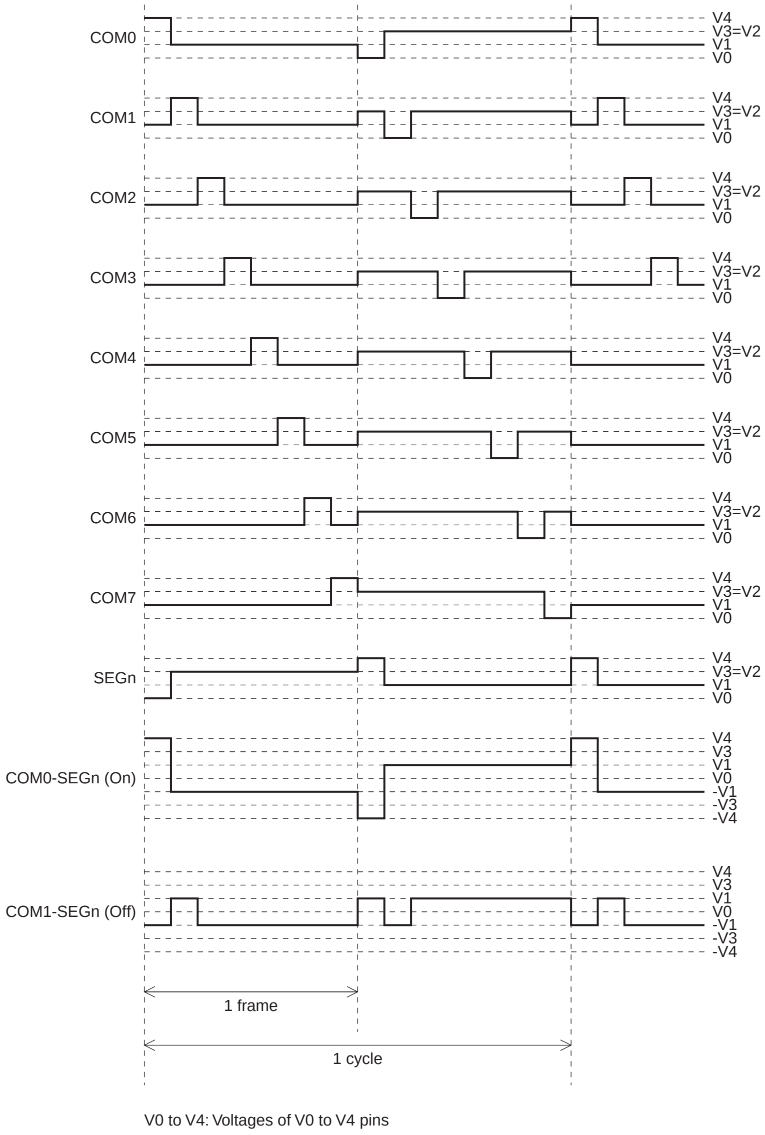
Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

Figure 28.7-9 shows the output waveform when the contents of display RAM are those shown in Table 28.7-5.

Table 28.7-5 Sample Contents of Display RAM

Segment	Contents of Display RAM							
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEGn	0	0	0	0	0	0	0	1

Figure 28.7-9 8 COM Mode, 1/3 Bias, 1/8 Duty Output Waveform Example



28.8 Notes on Using LCD Controller

This section provides notes on using the LCD controller.

■ Notes on Using LCD Controller

- To use an LCD pin as a general-purpose I/O port, set a corresponding common/segment select bit in an LCDC enable register (LCDCE1 to LCDCE7) to "0", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".
- If the selected frame period generation clock halts during LCD display operation, the AC waveform generator circuit also halts and therefore a DC voltage is applied to the liquid crystal elements. In this case, the LCD display operation must be stopped in advance. The conditions under which the main clock (time-base timer) or subclock (watch prescaler) halts depend on the selected clock mode and standby mode. The frame period is also affected if the time-base timer or watch prescaler is cleared according to the setting of the frame period generation clock select bit (LCDCC1:CSS).
- The operation of outputting display RAM data to the LCD is not in sync with the CPU accessing to the display RAM. When the interval for rewriting the display RAM is shorter than the LCD cycle, flickers may occur that are caused by different display patterns between frames.

CHAPTER 29

LCD CONTROLLER (MB95470H SERIES)

This chapter describes the functions and operations of the LCD controller.

29.1 Overview of LCD Controller

**The LCD controller has 2 modes: 8 COM mode and 4 COM mode.
In 8 COM mode, the LCD controller can use 28 bytes of display data memory and controls an LCD display via 8 common outputs and 28 segment outputs. It also has 2 different bias output options for driving an LCD panel.
In 4 COM mode, the LCD controller can use 16 bytes of display data memory and controls an LCD display via 4 common outputs and 32 segment outputs. It also has 3 different duty output options for driving an LCD panel.**

■ Functions of LCD Controller

The LCD controller uses its segment and common outputs to display the contents of display data memory (display RAM) directly on the LCD panel.

- It selects the 8 COM mode and the 4 COM mode through software.
- It has an LCD drive voltage divider resistor whose resistance value can be selected from 10 kΩ to 100 kΩ through software. An external divider resistor can also be used instead.
- In 8 COM mode, 8 common outputs (COM0 to COM7) and 28 segment outputs (SEG00 to SEG27) are available
- In 4 COM mode, 4 common outputs (COM0 to COM3) and 32 segment outputs (SEG00 to SEG31) are available.
- The display RAM size is 28 bytes (28 × 8 bits) in 8 COM mode and 16 bytes (32 × 4 bits) in 4 COM mode.
- It can use the main clock or the subclock as its operating clock.
- It has a blinking function, which is only available to certain pins.
- It can directly drive an LCD panel.
- In 8 COM mode, the bias can be selected from 1/3 or 1/4.
- In 4 COM mode, the duty can be selected from 1/2, 1/3 or 1/4 (governed by the bias setting).
- The interrupt event is in sync with the LCD module frame frequency.

Table 29.1-1 lists the bias-duty combinations available.

Table 29.1-1 Bias-duty Combinations

Duty	1/2 bias	1/3 bias	1/4 bias
1/2	❍	X	X
1/3	X	❍	X
1/4	X	❍	X
1/8, BLS8 = 0	X	❍	X
1/8, BLS8 = 1	X	X	❍

❍ : Recommended combination
X : Prohibited combination

29.2 Configuration of LCD Controller

The LCD controller consists of the following blocks, which are divided functionally into a controller section that generates the segment and common signals based on the content of display RAM and a driver section that drives the LCD.

Controller section

- **LCDC control registers (LCDCC1, LCDCC2)**
- **LCDC enable registers (LCDCE1 to LCDCE6)**
- **LCDC blinking setting registers (LCDCB1, LCDCB2)**
- **Display RAM**
- **Clock selection**
- **Timing control**

Driver section

- **AC waveform generator circuit**
- **Common driver**
- **Segment driver**
- **Divider resistor**

■ LCD Controller Block Diagrams

Figure 29.2-1 LCD Controller Block Diagram (8 COM Mode)

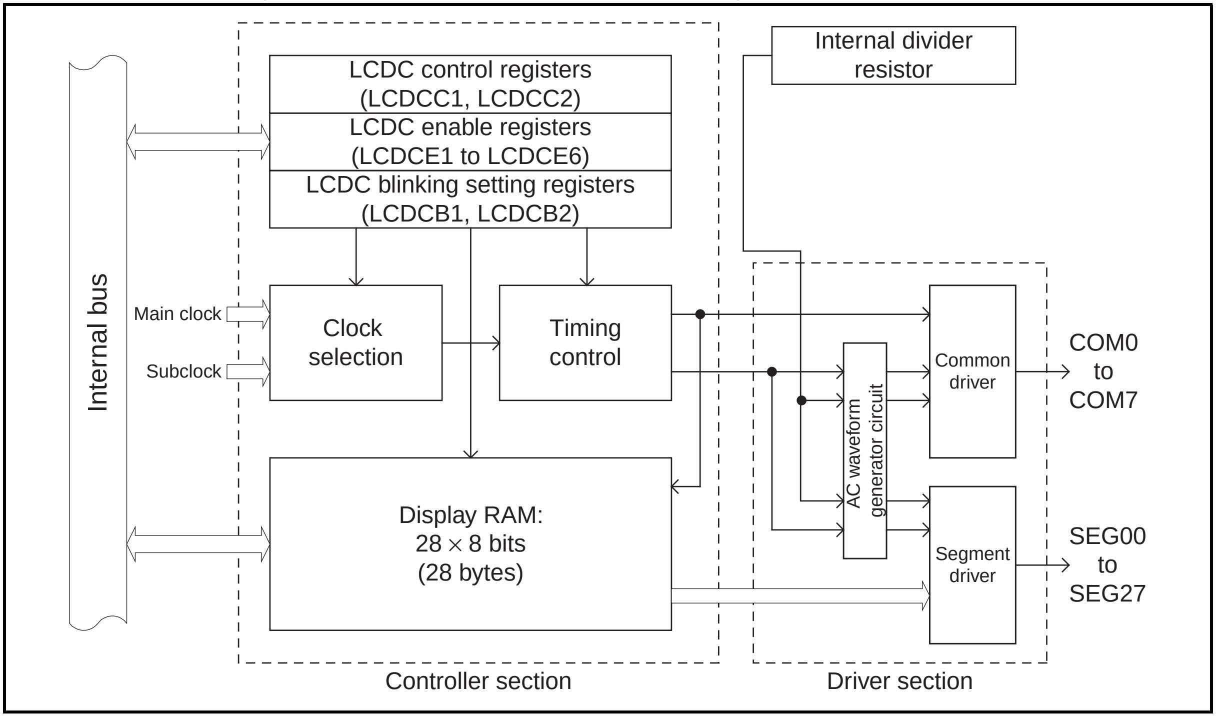
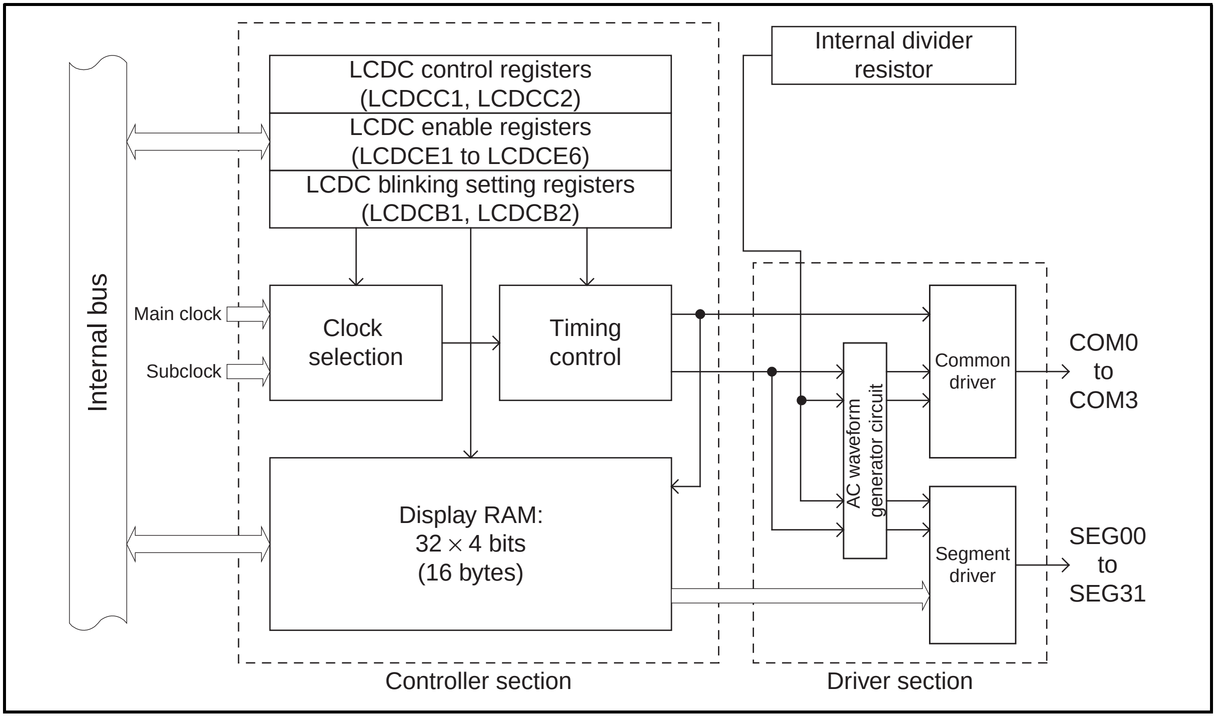

Internal bus
LCDC control registers (LCDCC1, LCDCC2)
LCDC enable registers (LCDCE1 to LCDCE6)
LCDC blinking setting registers (LCDCB1, LCDCB2)
Internal divider resistor
Main clock
Subclock
Clock selection
Timing control
Display RAM: 28 × 8 bits (28 bytes)
AC waveform generator circuit
Common driver
Segment driver
COM0 to COM7
SEG00 to SEG27
Controller section
Driver section

Figure 29.2-2 LCD Controller Block Diagram (4 COM Mode)


Internal bus
LCDC control registers (LCDCC1, LCDCC2)
LCDC enable registers (LCDCE1 to LCDCE6)
LCDC blinking setting registers (LCDCB1, LCDCB2)
Main clock
Subclock
Clock selection
Timing control
Display RAM: 32 × 4 bits (16 bytes)
Internal divider resistor
AC waveform generator circuit
Common driver
Segment driver
COM0 to COM3
SEG00 to SEG31
Controller section
Driver section

● LCDC control register 1 (LCDCC1)

This register is used to select the clock for generating the frame period, select the display mode, select the frame period clock, and control the LCD driving power supply.

● LCDC control register 2 (LCDCC2)

This register is used to enable and disable interrupts, indicate interrupt status and set the following parameters:

- Internal resistance value (10 kΩ or 100 kΩ)
- Bias to be used in 8 COM mode (1/3 or 1/4)
- Displaying data or a blank screen
- Inverted display

● LCDC enable registers 1 to 6 (LCDCE1 to LCDCE6)

These registers are used to control port inputs, blink interval, and pins.

● LCDC blinking setting register 1 (LCDCB1), LCDC blinking setting register 2 (LCDCB2)

These registers are used to turn on or off blinking.

● Display RAM

In 8 COM mode, 28 × 8 bits of RAM is available for generating segment output signals.

In 4 COM mode, 32 × 4 bits of RAM is available for generating segment output signals.

The content of the display RAM are read automatically in sync with the common signal selection timing and are output from segment output pins.

When the display RAM is modified, the content of the VRAM will be output from segment output pins.

● Clock selection

The frame frequency is generated based on the selection from the eight frequencies generated from the two clocks.

● Timing control

The COM and SEG signals are controlled based on the frame frequency and register settings.

● AC waveform generator circuit

This block generates AC waveforms for driving the LCD from timing control signals.

● Common driver

This block is the driver of the LCD COM pins.

● Segment driver

This block is the driver of the LCD SEG pins.

● Divider resistor

This block is a resistor used to generate the LCD drive voltage. A divider resistor can be connected to as an external component when a LCDC drive power supply pin (V1 to V4) serves as a divider resistor connection pin.

■ LCD Controller Power Supply Voltage

The power supply voltage for the LCD driver is generated by internal divider resistors or by connecting external divider resistors to the V1 to V4 pins.

■ Input Clock

The LCD controller uses the output clock of time-base timer or watch prescaler as the input clock (operation clock).

29.2.1 Internal Divider Resistors for LCD Controller

The internal divider resistors generate power supply voltage for the LCD driver.

■ Internal Divider Resistors

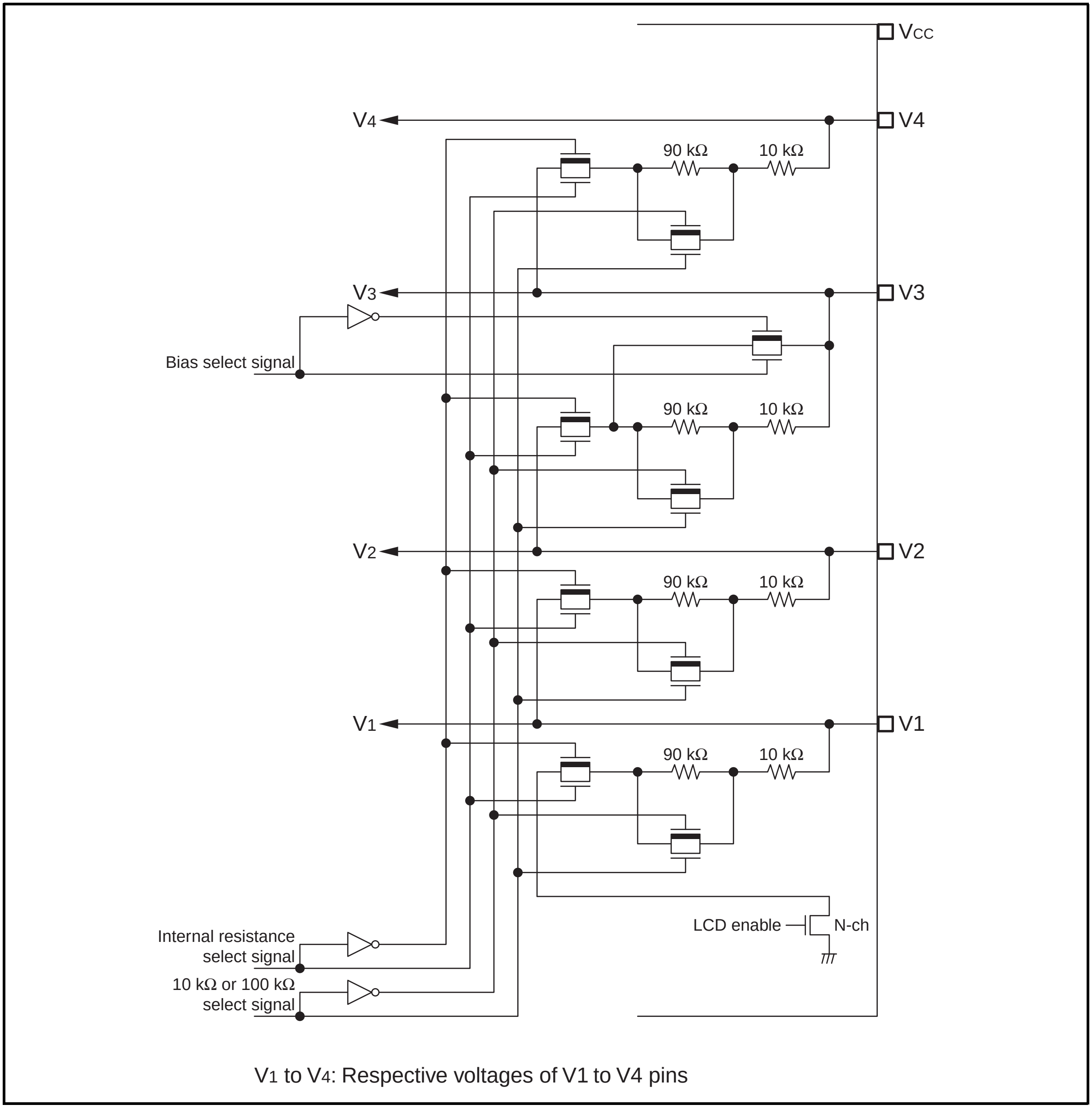
Internal divider resistors are included. In addition, external divider resistors can be connected to the LCD driving power pins (V1 to V4).

The internal and external divider resistors are selected by the driving power control bit in the LCDC control register 1 (LCDCC1:VSEL). Setting the VSEL bit to "1" energizes the internal divider resistors. To use only the internal divider resistors without any external divider resistor, set the VE3 bit in the LCDC enable register 1 (LCDCE1) to "1". (When internal split resistors are used, the V4 pin cannot be used as general-purpose I/O ports.)

The LCD controller stops upon transition to main stop or watch mode (STBC:TMD = 1) while operation in main stop and watch modes is disabled (LCDCC1:LCDEN = 0) with LCD operation halted (LCDCC1:MS[2:0] = 000_B).

Figure 29.2-3 shows an equivalent circuit with internal divider resistors used.

Figure 29.2-3 Equivalent Circuit with Internal Divider Resistors Used



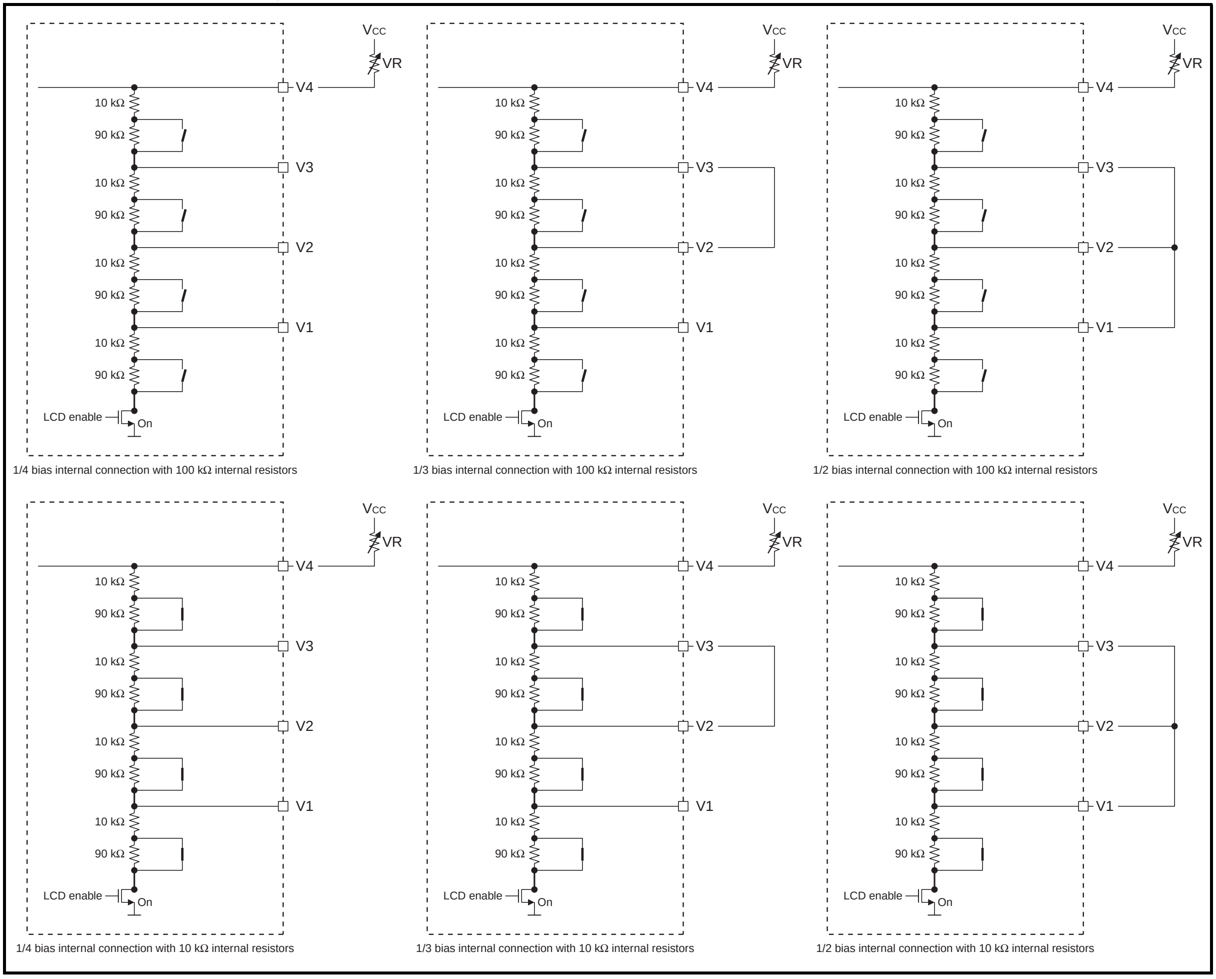
V1 to V4: Respective voltages of V1 to V4 pins

■ Use of Internal Divider Resistors and Brightness Control

There are two types of internal divider resistors: 10 kΩ and 100 kΩ. Figure 29.2-4 shows examples of using the internal divider resistors.

If sufficient brightness cannot be achieved with the internal divider resistors in use, connect a variable resistor (VR) externally (between the Vcc pin and the V4 pin) to adjust the V4 voltage. Figure 29.2-5 illustrates connecting a VR to the V4 pin to control brightness.

Figure 29.2-4 States with Internal Divider Resistors Used



1/4 bias internal connection with 100 kΩ internal resistors

1/3 bias internal connection with 100 kΩ internal resistors

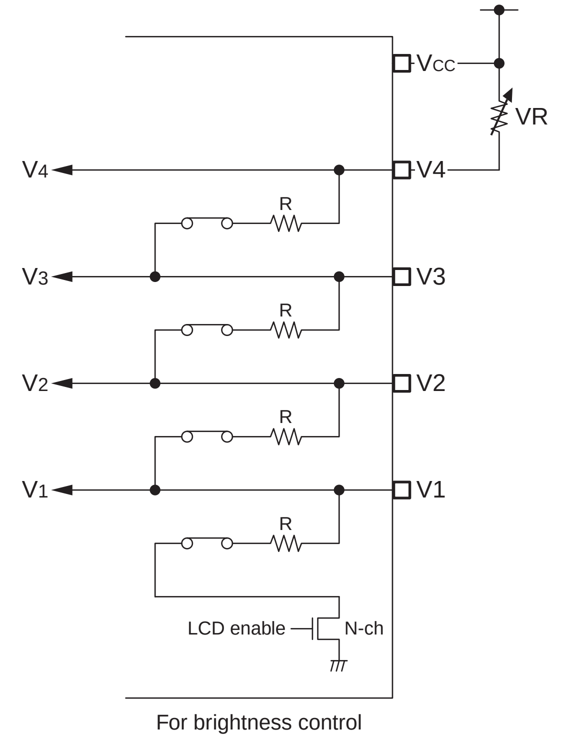
1/2 bias internal connection with 100 kΩ internal resistors

1/4 bias internal connection with 10 kΩ internal resistors

1/3 bias internal connection with 10 kΩ internal resistors

1/2 bias internal connection with 10 kΩ internal resistors

Figure 29.2-5 Brightness Control by Connecting VR to V4 Pin



V1 to V4: Respective voltages of V1 to V4 pins

29.2.2 External Divider Resistors for LCD Controller

The V1 to V4 pins of this series can be connected to external divider resistors. Connecting a variable resistor between the V_{CC} pin and the V4 pin can control brightness.

■ External Divider Resistors

If not using the internal divider resistors, you can connect external divider resistors to the LCD drive power supply pins (V1 to V4) instead. Figure 29.2-6 shows an example of connecting external divider resistors, and Table 29.2-1 lists the LCD drive voltage settings for the bias method.

Figure 29.2-6 Example of Connecting External Divider Resistors

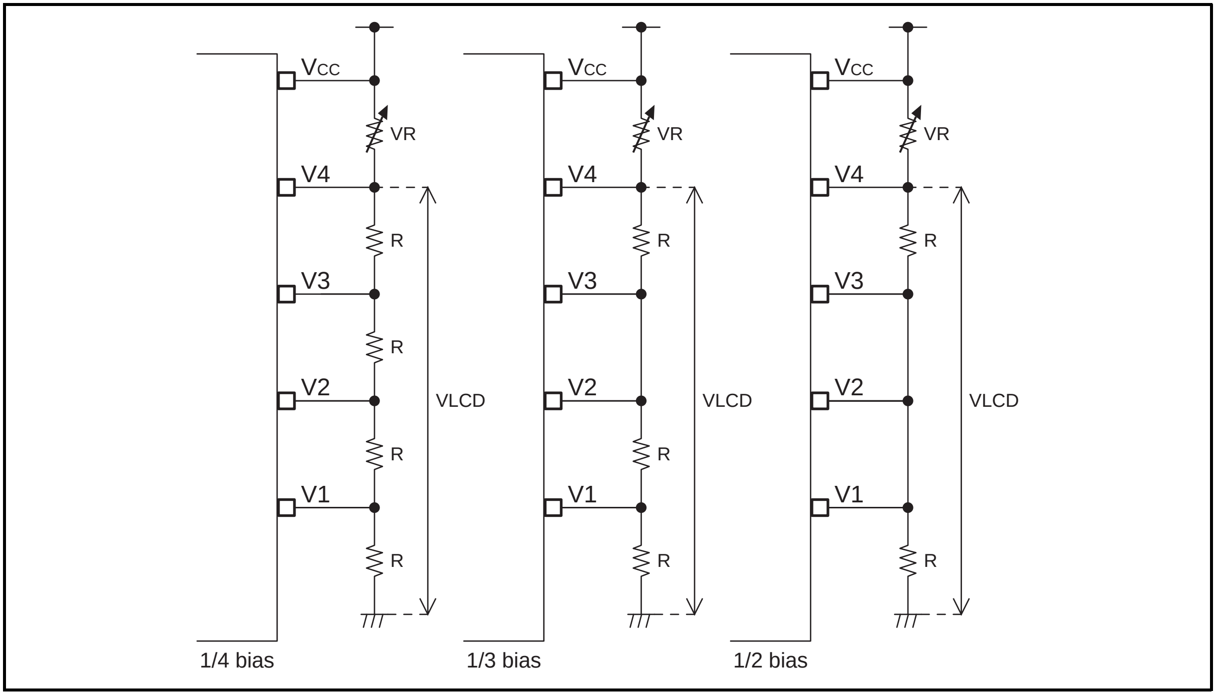


Table 29.2-1 LCD Driving Voltage Settings

	V4	V3	V2	V1
1/2 bias	VLCD	X	1/2 VLCD	X
1/3 bias	VLCD	2/3 VLCD	2/3 VLCD	1/3 VLCD
1/4 bias	VLCD	3/4 VLCD	1/2 VLCD	1/4 VLCD

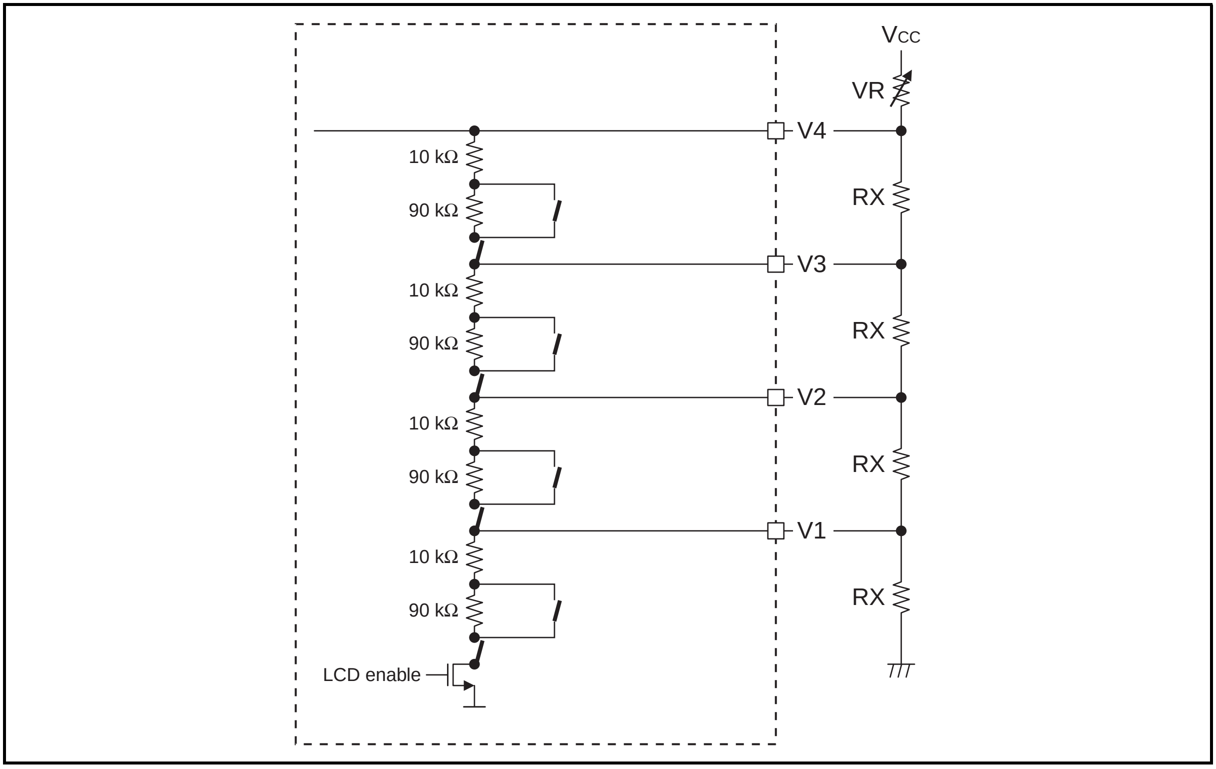
VLCD : LCD operating voltage

X : No external divider resistor

■ Use of External Divider Resistors

As the V1 pin is connected to V_{SS} (GND) internally via a transistor, when using external divider resistors, you can shut off the current flowing to the resistors when the LCD controller is halted by connecting the V_{SS} end of the divider resistors to the V1 pin. Figure 29.2-7 shows the state with external divider resistors used.

Figure 29.2-7 States with External Divider Resistors Used



1) To connect the external divider resistors without being affected by the internal divider resistors, you need to write "0" to the drive voltage control bit in the LCDC control register 1 (LCDCC1:VSEL) to disconnect all internal divider resistors. Write "1" to the V4 to V1 select bits in the LCDC enable register 1 (LCDCE1:VE[4:1]) so that the target ports can be used as power supply pins to drive the LCD.

2) When the internal divider resistors are disconnected, writing a value other than "000_B" to the display mode select bits (MS[2:0]) in LCDCC1 turns on the LCD controller.

Note:

The appropriate resistance of an external RX resistor depends on the LCD used. Use an external RX resistor whose resistance is suitable to the LCD used.

29.3 Pins of LCD Controller

This section describes the pins of the LCD controller.

■ Pins of LCD Controller

The pins of the LCD controller are: 8 common output pins (COM0 to COM7), 32 segment output pins (SEG00 to SEG31), and 4 LCD drive power supply pins (V1 to V4).

To use these pins for the LCD, set the corresponding bits in the LCDC enable registers (LCDCE1 to LCDCE6) to "1".

To use an LCD pin as a general-purpose I/O port, set its corresponding bit in an LCDC enable register (LCDCE1 to LCDCE6) for selecting the pin function to "0", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".

● COM0 to COM7 pins

In 8 COM mode, COM0 to COM7 function as LCD common output pins.

In 4 COM mode, COM0 to COM3 function as LCD common output pins, and COM4 to COM7 are defaulted as I/O ports regardless of the settings of the LCDCE1 to LCDCE6 registers.

In addition, COM0 to COM7 pins can also function as general-purpose I/O ports.

● SEG00 to SEG31 pins

In 8 COM mode, SEG00 to SEG27 function as LCD segment output pins, and SEG28 to SEG31 are defaulted as general-purpose I/O ports regardless of the settings of the LCDCE1 to LCDCE6 registers.

In 4 COM mode, SEG00 to SEG31 function as LCD segment output pins.

In addition, SEG00 to SEG31 can also function as general-purpose I/O ports.

● V1 to V4 pins

These pins function as the power supply pins for driving the LCD.

In addition, they can also function as general-purpose I/O ports.

■ Block Diagrams of Pins of LCD Controller

Figure 29.3-1 Block Diagram of V1 to V4 of LCD Controller

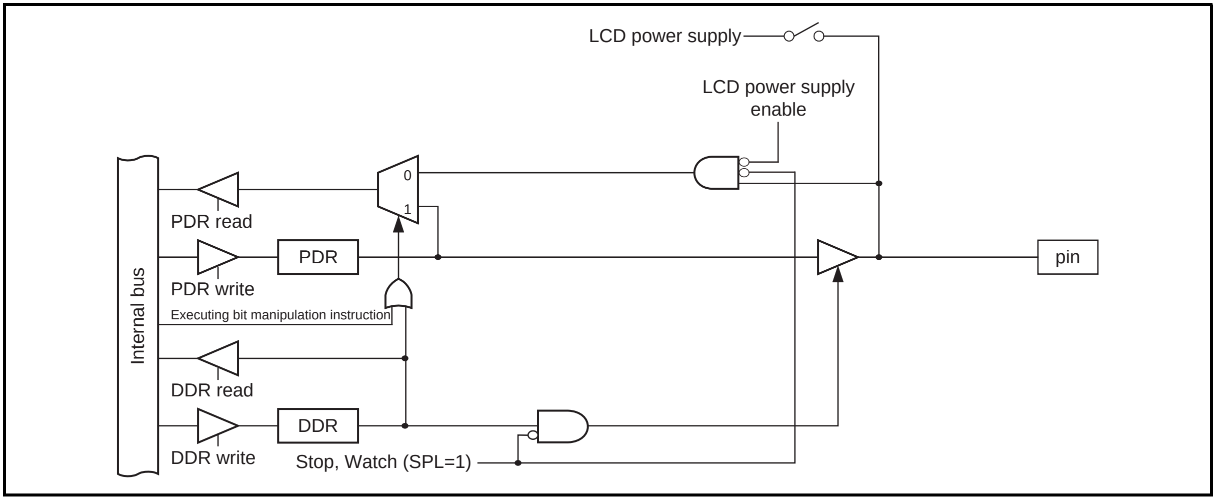


Figure 29.3-2 Block Diagram of COM0 to COM7 and SEG00 to SEG18 of LCD Controller

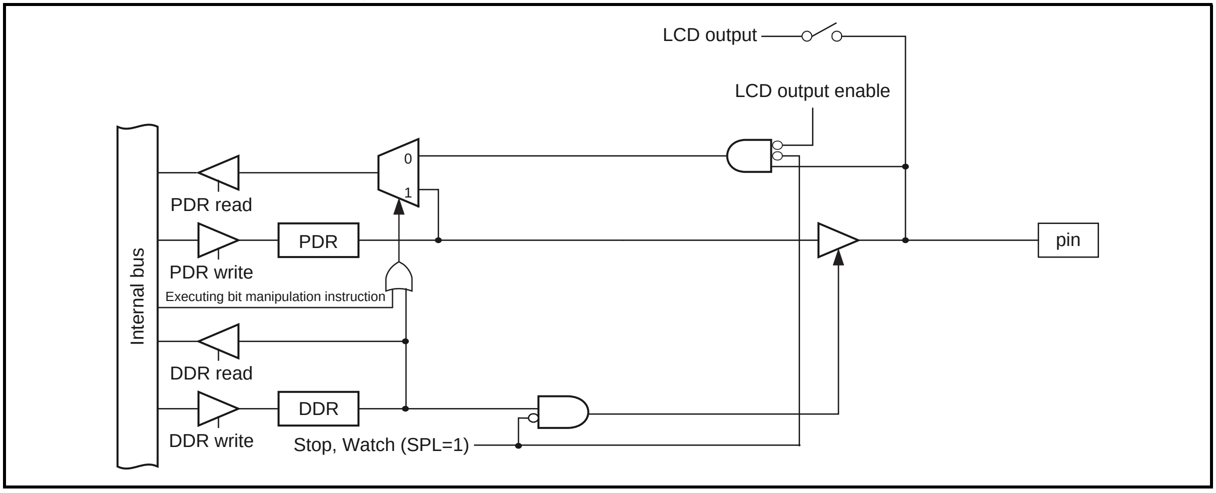


Figure 29.3-3 Block Diagram of SEG19, SEG20, SEG30 and SEG31 of LCD Controller

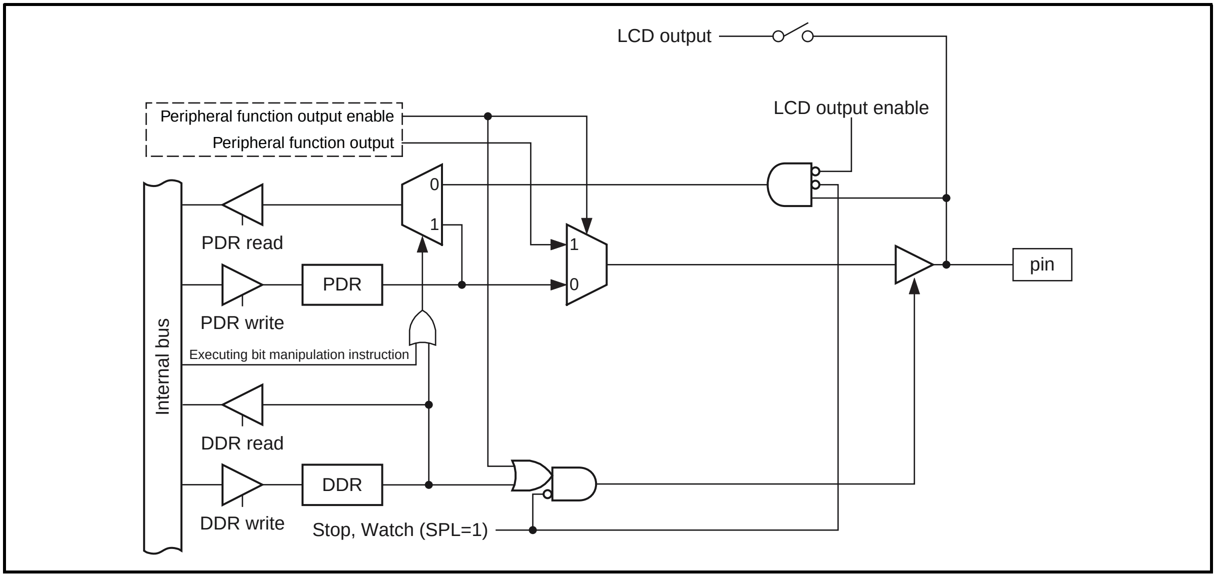


Figure 29.3-4 Block Diagram of SEG21 of LCD Controller

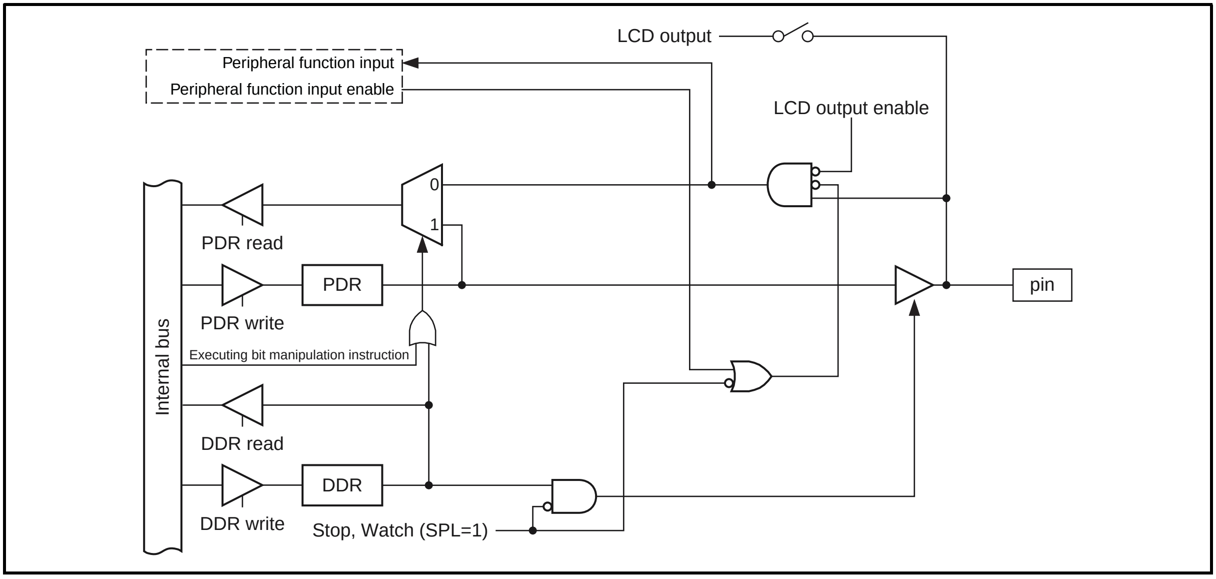


Figure 29.3-5 Block Diagram of SEG22 and SEG23 of LCD Controller

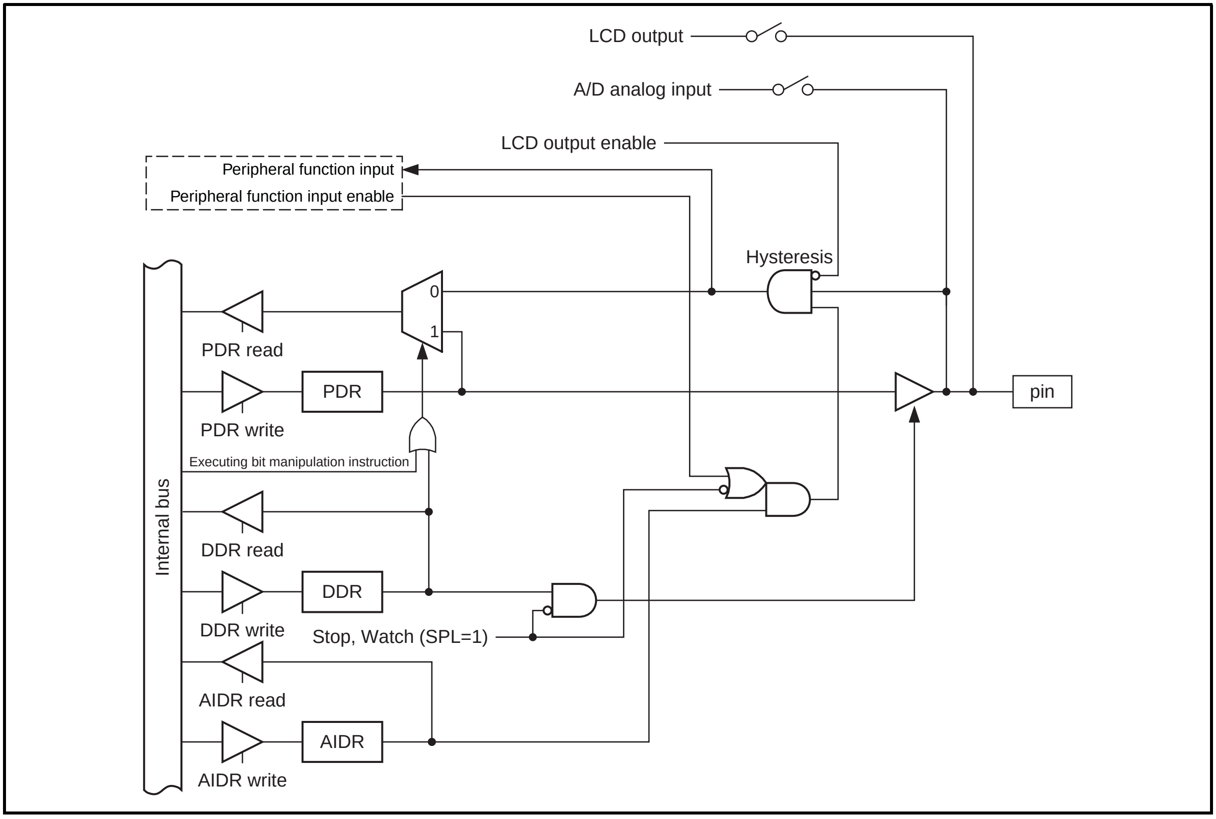


Figure 29.3-6 Block Diagram of SEG24, SEG26, SEG27 and SEG29 of LCD Controller

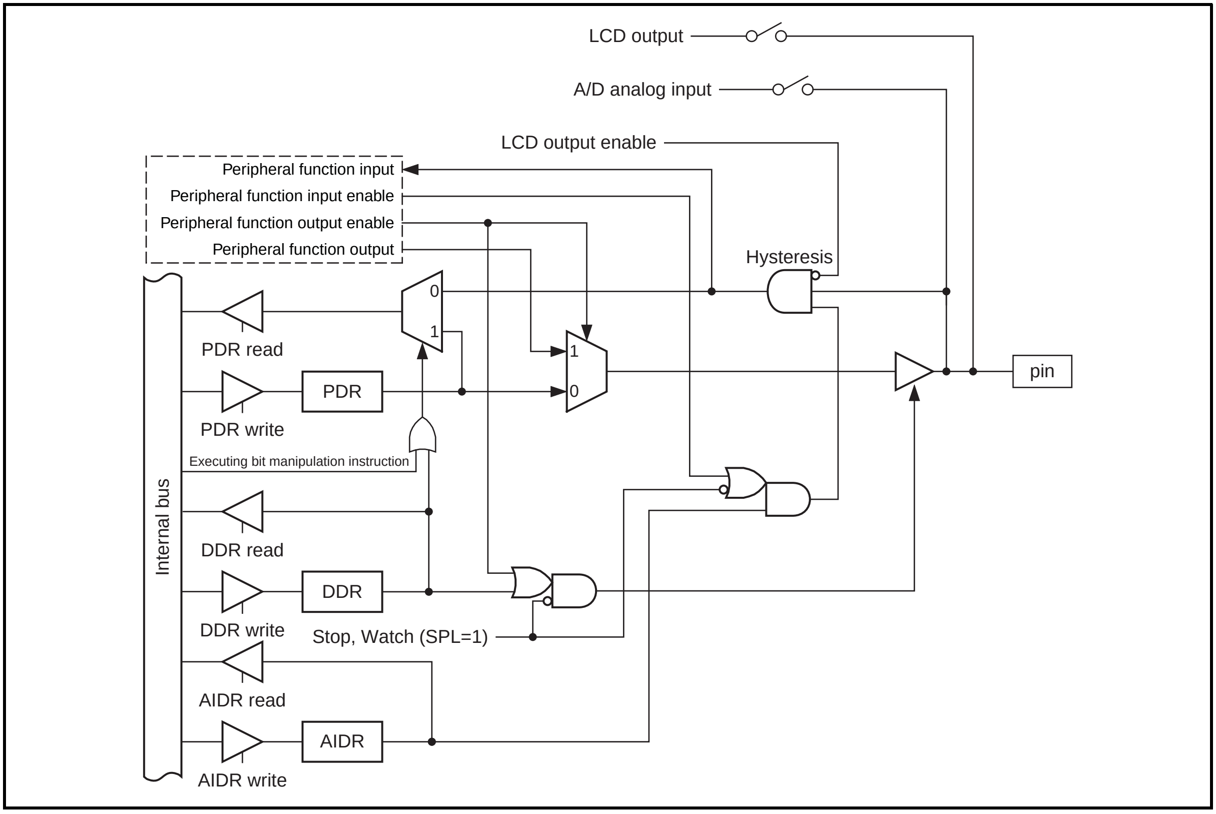

LCD output
A/D analog input
LCD output enable
Peripheral function input
Peripheral function input enable
Peripheral function output enable
Peripheral function output
Hysteresis
0
1
1
0
PDR read
PDR
PDR write
pin
Executing bit manipulation instruction
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
AIDR read
AIDR
AIDR write

Figure 29.3-7 Block Diagram of SEG25 of LCD Controller

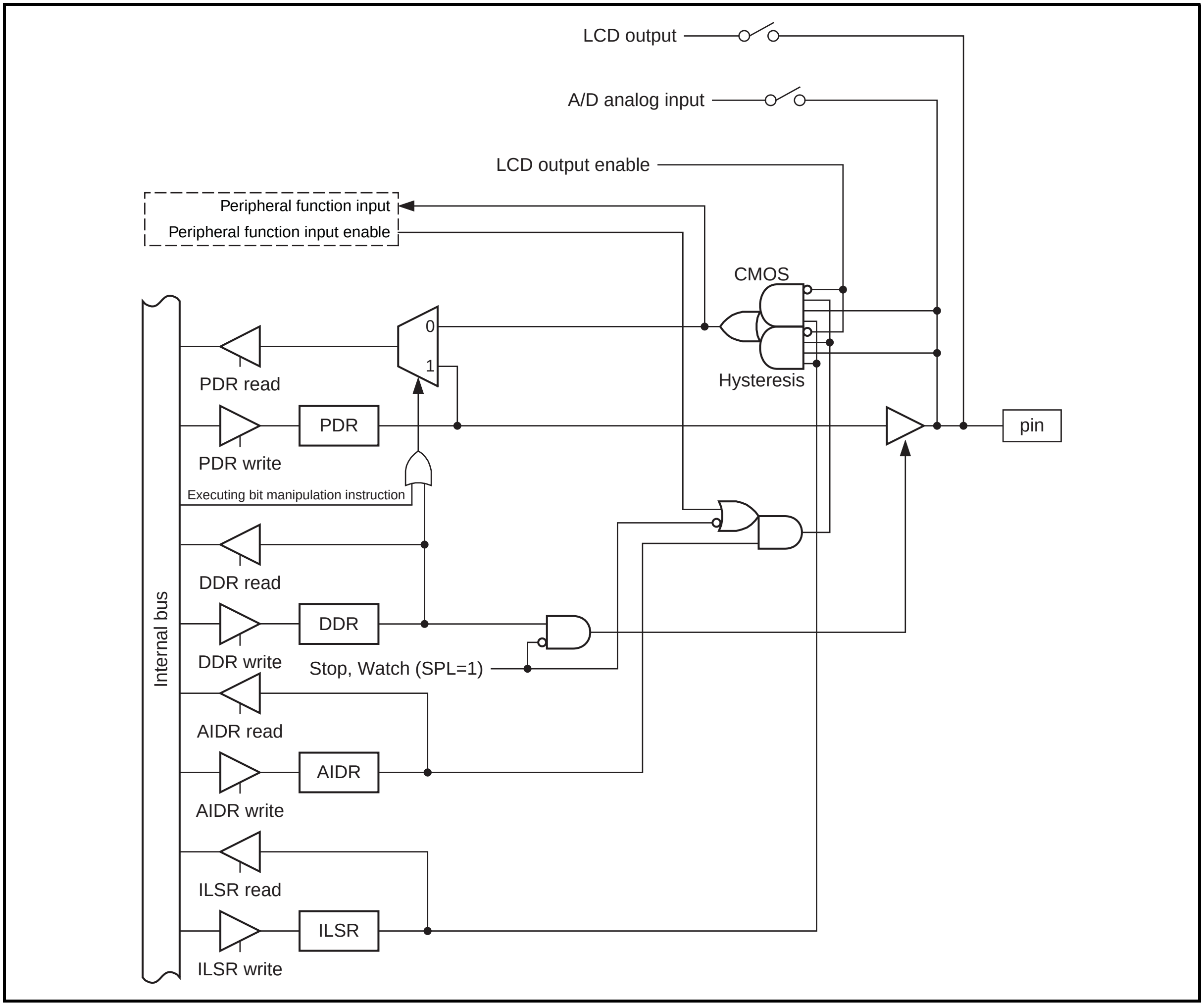
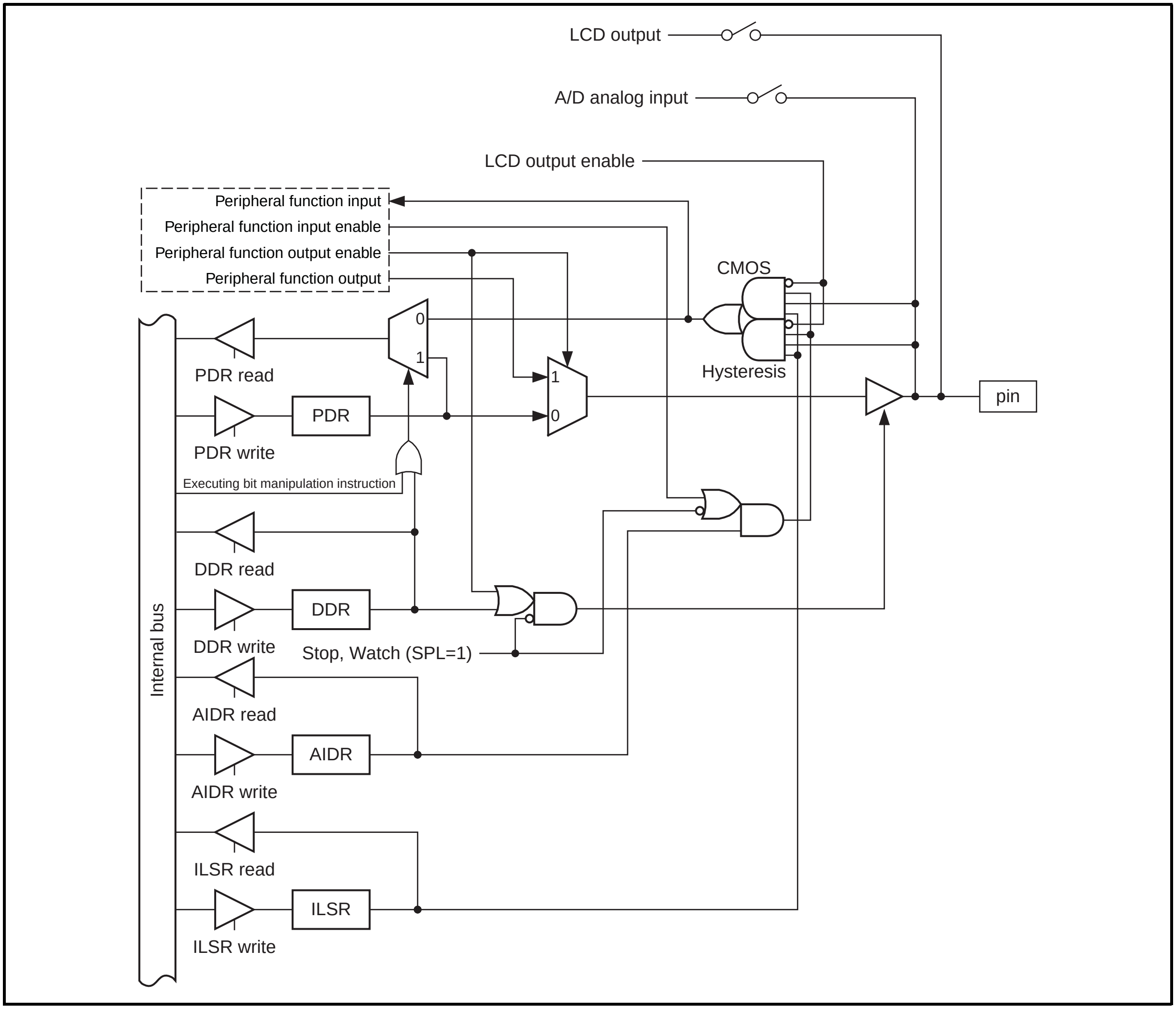


Figure 29.3-8 Block Diagram of SEG28 of LCD Controller



29.4 Registers of LCD Controller

This section describes the registers of the LCD controller.

■ Registers of LCD Controller

Figure 29.4-1 LCD Controller Registers (1/2)

LCDC control register 1 (LCDCC1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB0$_H$	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC control register 2 (LCDCC2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
004F$_H$	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF	00010100$_B$
	R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R(RM1),W	

LCDC enable register 1 (LCDCE1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB2$_H$	PICTL	BLSEL	VE4	VE3	VE2	VE1	-	-	00111100$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R0/WX	R0/WX	

LCDC enable register 2 (LCDCE2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB3$_H$	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 3 (LCDCE3)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB4$_H$	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 4 (LCDCE4)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB5$_H$	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R0/WX : The read value is always "0". Writing a value to this bit has no effect on operation.
- : Undefined bit

Figure 29.4-1 LCD Controller Registers (2/2)

LCDC enable register 5 (LCDCE5)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB6$_H$	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 6 (LCDCE6)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB7$_H$	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC blinking setting register 1 (LCDCB1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB9$_H$	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC blinking setting register 2 (LCDCB2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FBA$_H$	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

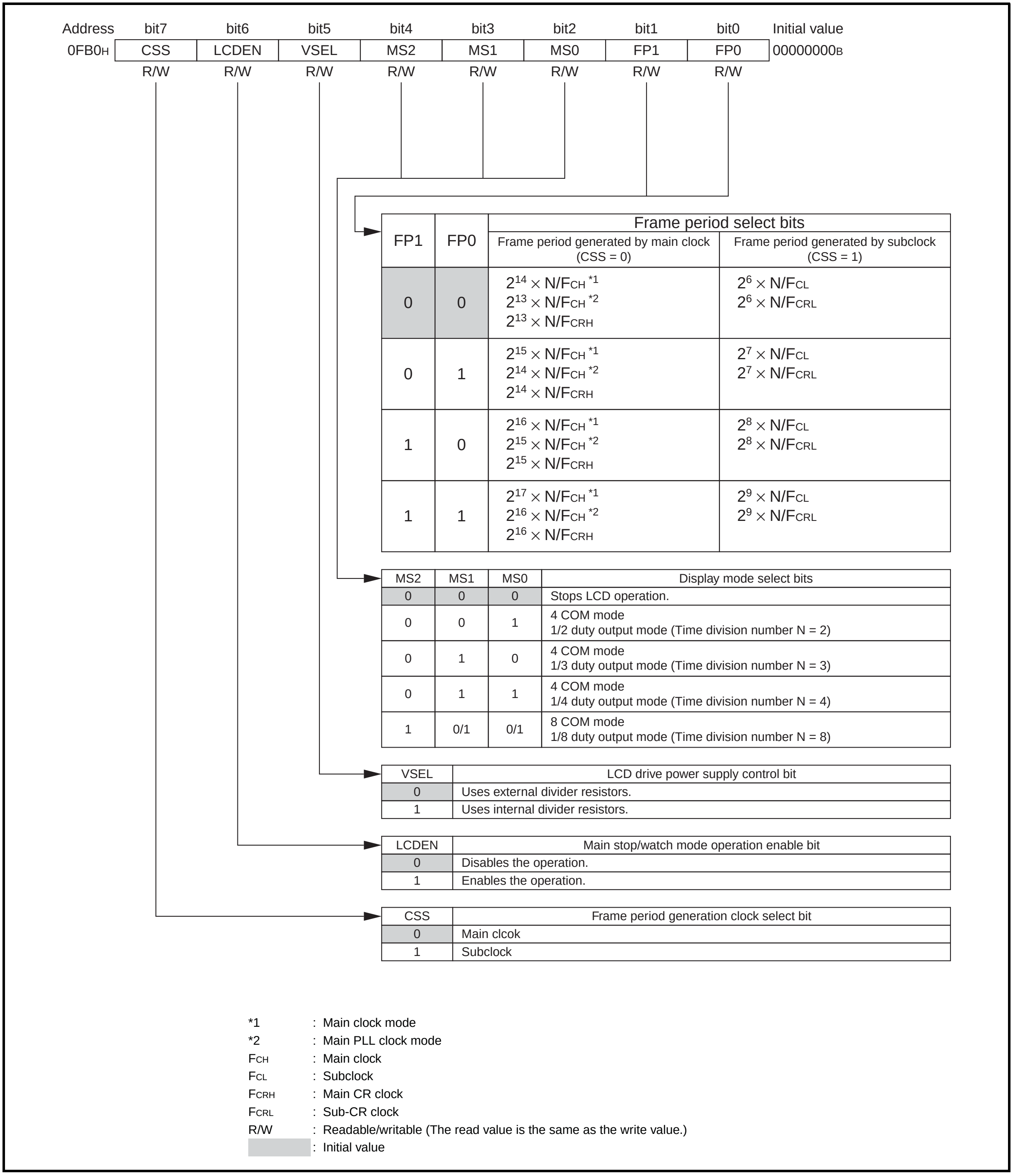
R/W : Readable/writable (The read value is the same as the write value.)

29.4.1 LCDC Control Register 1 (LCDCC1)

The LCDC control register 1 (LCDCC1) is used to set the clock, display mode, and power supply control.

■ LCDC Control Register 1 (LCDCC1)

Figure 29.4-2 LCDC Control Register 1 (LCDCC1)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB0$_H$	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0	00000000$_B$
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

FP1	FP0	Frame period select bits	
		Frame period generated by main clock (CSS = 0)	Frame period generated by subclock (CSS = 1)
0	0	$2^{14} \times N/F_{CH}$ *1 $2^{13} \times N/F_{CH}$ *2 $2^{13} \times N/F_{CRH}$	$2^{6} \times N/F_{CL}$ $2^{6} \times N/F_{CRL}$
0	1	$2^{15} \times N/F_{CH}$ *1 $2^{14} \times N/F_{CH}$ *2 $2^{14} \times N/F_{CRH}$	$2^{7} \times N/F_{CL}$ $2^{7} \times N/F_{CRL}$
1	0	$2^{16} \times N/F_{CH}$ *1 $2^{15} \times N/F_{CH}$ *2 $2^{15} \times N/F_{CRH}$	$2^{8} \times N/F_{CL}$ $2^{8} \times N/F_{CRL}$
1	1	$2^{17} \times N/F_{CH}$ *1 $2^{16} \times N/F_{CH}$ *2 $2^{16} \times N/F_{CRH}$	$2^{9} \times N/F_{CL}$ $2^{9} \times N/F_{CRL}$

MS2	MS1	MS0	Display mode select bits
0	0	0	Stops LCD operation.
0	0	1	4 COM mode 1/2 duty output mode (Time division number N = 2)
0	1	0	4 COM mode 1/3 duty output mode (Time division number N = 3)
0	1	1	4 COM mode 1/4 duty output mode (Time division number N = 4)
1	0/1	0/1	8 COM mode 1/8 duty output mode (Time division number N = 8)

VSEL	LCD drive power supply control bit
0	Uses external divider resistors.
1	Uses internal divider resistors.

LCDEN	Main stop/watch mode operation enable bit
0	Disables the operation.
1	Enables the operation.

CSS	Frame period generation clock select bit
0	Main clcok
1	Subclock

*1	:	Main clock mode
*2	:	Main PLL clock mode
F_{CH}	:	Main clock
F_{CL}	:	Subclock
F_{CRH}	:	Main CR clock
F_{CRL}	:	Sub-CR clock
R/W	:	Readable/writable (The read value is the same as the write value.)
▨	:	Initial value

Table 29.4-1 Functions of Bits in LCDC Control Register 1 (LCDCC1)

Bit name		Function
bit7	CSS: Frame period generation clock select bit	This bits selects the clock for generating the frame period for LCD display. • When this bit is "0", the LCD controller operates with the output of the time-base timer driven by the main clock. When the bit is "1", the LCD controller operates with the output of the watch prescaler driven by the subclock. Note: As the main clock stops oscillation in main stop mode and subclock mode, the LCD controller cannot operate with the output of the time-base timer in these modes. Shifting the main clock speed (using the gear function) during operation with the time-base timer output does not affect the frame period. LCD display may flicker when the clock speed is being shifted. Before shifting it, therefore, temporarily halt the display, for example, by using blanking (LCDCC2:BK = 1).
bit6	LCDEN: Main stop/watch mode operation enable bit	This bit specifies whether the LCD controller is to continue to operate in main stop mode and watch mode. **Writing "0":** Stops the LCD controller. **Writing "1":** Makes the LCD controller continue to operate even after the clock mode transits to main stop mode or watch mode. Note: In the case of making the LCD controller continue to operate in main stop mode or watch mode, select the subclock as the clock for generating the frame period for the LCD display (CSS = 1).
bit5	VSEL: LCD driving power control bit	This bit selects whether to energize the internal divider resistors. **Writing "0":** Shuts off the internal divider resistors. **Writing "1":** Energizes the internal divider resistors. Note: Write "0" to this bit when connecting to the external divider resistor.
bit4 to bit2	MS2, MS1, MS0: Display mode select bits	These bits select the display mode from 4 COM mode and 8 COM mode and also select an output waveform duty from four options. • The common output pin to be used is determined by the duty output mode selected. • When these bits are "000_B", the LCD controller driver stops the LCD display operation. Note: If the selected frame period generation clock can be halted, for example, upon transition to stop mode, halt the LCD display operation (MS2, MS1, MS0 = 000_B) in advance. As the LCD display may flicker when the display mode changes, halt the display temporarily, for example, by using blanking (LCDCC2:BK = 1) before changing the display mode.
bit1, bit0	FP1, FP0: Frame period select bits	This bit selects an LCD display frame period from four options. Note: Set these bits according to the optimum frame period for the LCD module to be used. The frame period is affected by the source oscillation frequency. As the LCD display may flicker when the frame period changes, halt the display temporarily, for example, by using blanking (LCDCC2:BK = 1) before changing the frame period.

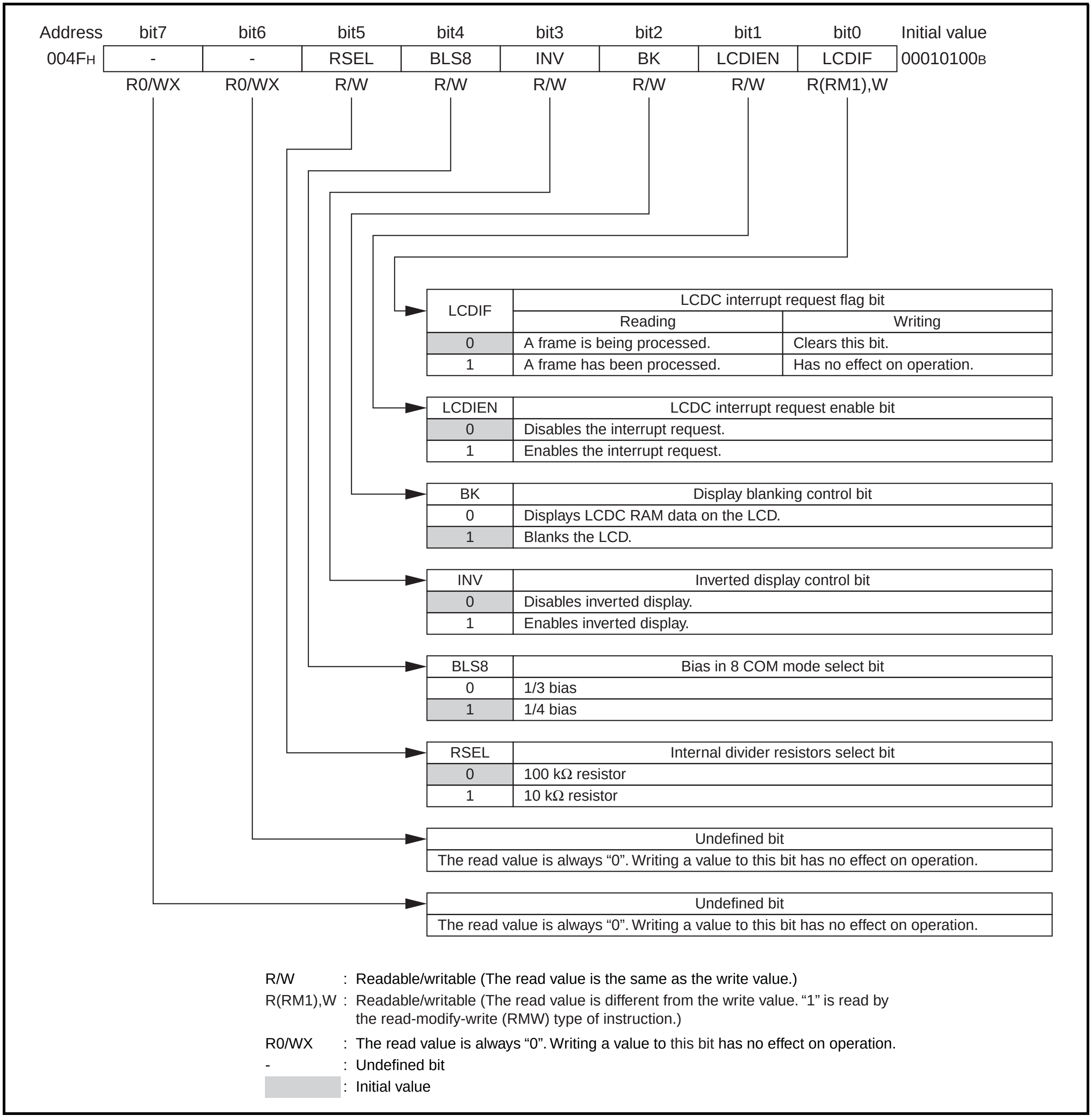
29.4.2 LCDC Control Register 2 (LCDCC2)

The LCDC control register 2 (LCDCC2) is used to enable and disable interrupts, indicate interrupt status and set the following parameters:

- **Internal resistance value from 10 kΩ or 100 kΩ**
- **Bias to be used in 8 COM mode from 1/3 or 1/4**
- **Displaying data or a blank screen**
- **Inverted display**

■ LCDC Control Register 2 (LCDC2)

Figure 29.4-3 LCDC Control Register 2 (LCDC2)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
004FH	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF	00010100B
	R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R(RM1),W	

LCDIF	LCDC interrupt request flag bit	
	Reading	Writing
0	A frame is being processed.	Clears this bit.
1	A frame has been processed.	Has no effect on operation.

LCDIEN	LCDC interrupt request enable bit
0	Disables the interrupt request.
1	Enables the interrupt request.

BK	Display blanking control bit
0	Displays LCDC RAM data on the LCD.
1	Blanks the LCD.

INV	Inverted display control bit
0	Disables inverted display.
1	Enables inverted display.

BLS8	Bias in 8 COM mode select bit
0	1/3 bias
1	1/4 bias

RSEL	Internal divider resistors select bit
0	100 kΩ resistor
1	10 kΩ resistor

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R0/WX : The read value is always “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
▨ : Initial value

Table 29.4-2 Functions of Bits in LCDC Control Register 2 (LCDCC2)

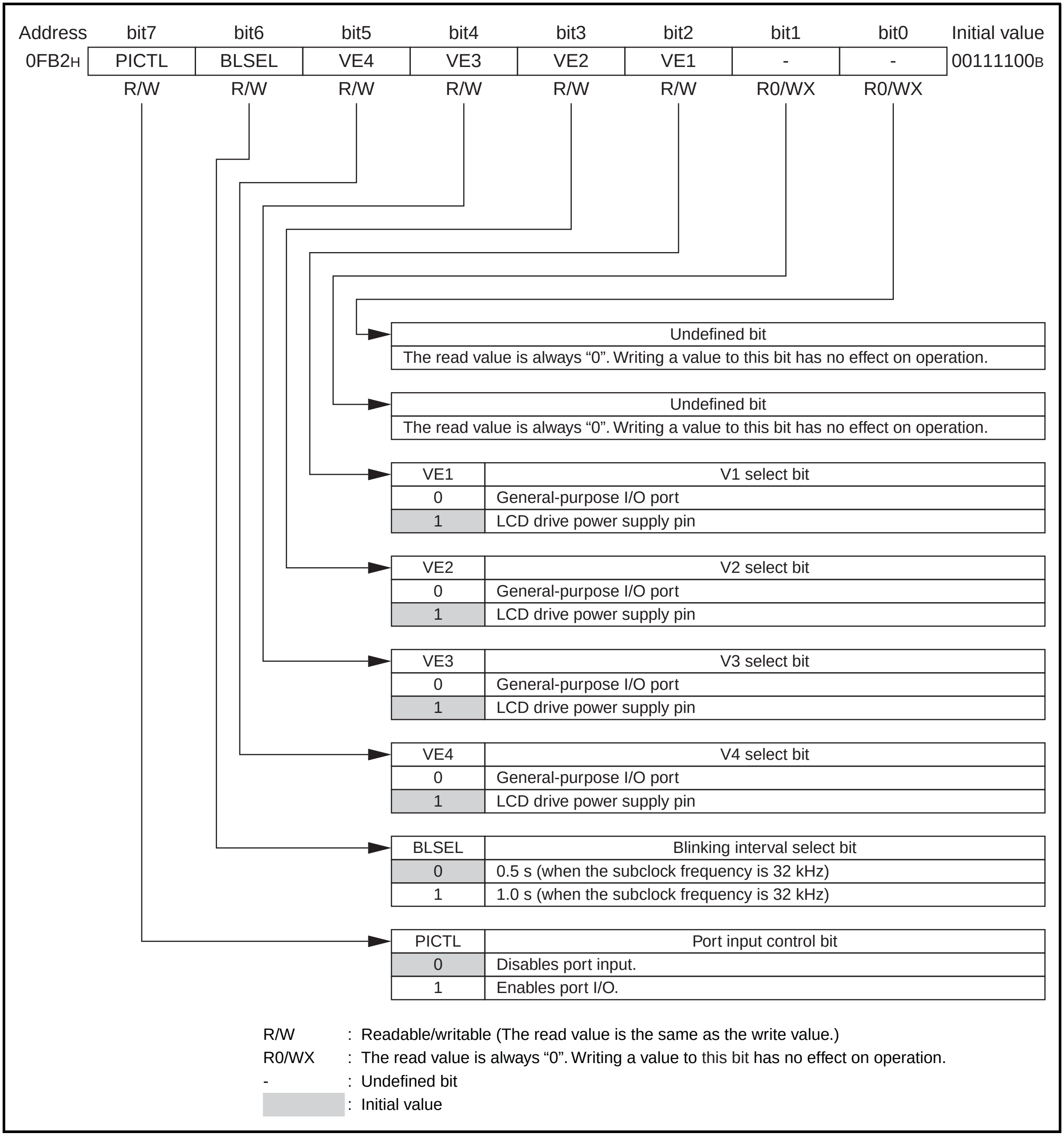
Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5	RSEL: Internal divider resistor select bit	This bit selects which type of resistors is to be used as internal divider resistors. **Writing "0":** Selects the 100 kΩ resistor. **Writing "1":** Selects the 10 kΩ resistor.
bit4	BLS8: Bias in 8 COM mode select bit	This bit selects which type of bias is to be used by software in 8 COM mode. **Writing "0":** Selects 1/3 bias. **Writing "1":** Selects 1/4 bias. Note: Although this bit can be accessed in both 8 COM mode and 4 COM mode, writing a value to this bit in 4 COM mode has no effect on operation.
bit3	INV: Inverted display control bit	This bit controls the inverted display on the LCD. **Writing "0":** Disables inverted display. **Writing "1":** Enables inverted display.
bit2	BK: Display blanking control bit	This bit controls display blanking of the LCD. **Writing "0":** Displays LCDC RAM data on the LCD. **Writing "1":** Blanks the LCD. When display blanking is selected (BK = 1), a segment output pin outputs a waveform not selected for displaying data on the LCD.
bit1	LCDIEN: LCDC interrupt request enable bit	This bit enables or disables the generation of an interrupt in sync with the LCD module frame frequency. **Writing "0":** Disables the interrupt request. **Writing "1":** Enables the interrupt request.
bit0	LCDIF: LCDC interrupt request flag bit	This bit indicates whether the LCD controller has finished processing a frame. **Reading "0":** Indicates that the LCD controller is processing a frame. **Reading "1":** Indicates that the LCD controller has finished processing a frame. **Writing "0":** Clears this bit. **Writing "1":** Has no effect on operation. This bit always returns "1" when read by a read-modify-write (RMW) type of instruction.

29.4.3 LCDC Enable Register 1 (LCDCE1)

The LCDC enable register 1 (LCDCE1) is used to control port input, set the blink cycle, and enable LCD pins.

■ LCDC Enable Register 1 (LCDCE1)

Figure 29.4-4 LCDC Enable Register 1 (LCDCE1)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB2H	PICTL	BLSEL	VE4	VE3	VE2	VE1	-	-	00111100B
	R/W	R/W	R/W	R/W	R/W	R/W	R0/WX	R0/WX	

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

VE1	V1 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

VE2	V2 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

VE3	V3 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

VE4	V4 select bit
0	General-purpose I/O port
1	LCD drive power supply pin

BLSEL	Blinking interval select bit
0	0.5 s (when the subclock frequency is 32 kHz)
1	1.0 s (when the subclock frequency is 32 kHz)

PICTL	Port input control bit
0	Disables port input.
1	Enables port I/O.

R/W	:	Readable/writable (The read value is the same as the write value.)
R0/WX	:	The read value is always “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

Table 29.4-3 Functions of Bits in LCDC Enable Register 1 (LCDCE1)

Bit name		Function
bit7	PICTL: Port input control bit	This bit controls general-purpose I/O ports that also function as segment or common output pins. **Writing "0":** Disables the input function of such general-purpose I/O ports and suppresses shoot-through current during LCD output. In addition, writing "0" to PICTL also disables the output function of such general-purpose I/O ports. **Writing "1":** Enables the I/O function of such general-purpose I/O ports. To use a segment or common output pin as a general-purpose I/O port, write "1" to PICTL. Note: As the input function of such general-purpose I/O ports will be disabled on a reset, in order to use their input function, write "1" to PICTL. When they are used as segment or common output pins, their input function will be disabled regardless of the setting of this bit.
bit6	BLSEL: Blinking interval select bit	This bit selects the blinking interval to be used when blinking is enabled. Blinking is to be enabled by the LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2). A blinking interval of 1.0 s makes the LCD stay on for 0.5 s and off for 0.5 s; a blinking interval of 0.5 s makes the LCD stay on for 0.25 s and off for 0.25 s.
bit5	VE4: V4 select bit	This bit selects the function of the V4 pin. **Writing "0":** Makes the V4 pin function as a general-purpose I/O port. **Writing "1":** Makes the V4 pin function as an LCD drive power supply pin.
bit4	VE3: V3 select bit	This bit selects the function of the V3 pin. **Writing "0":** Makes the V3 pin function as a general-purpose I/O port. **Writing "1":** Makes the V3 pin function as an LCD drive power supply pin.
bit3	VE2: V2 select bit	This bit selects the function of the V2 pin. **Writing "0":** Makes the V2 pin function as a general-purpose I/O port. **Writing "1":** Makes the V2 pin function as an LCD drive power supply pin.
bit2	VE1: V1 select bit	This bit selects the function of the V1 pin. **Writing "0":** Makes the V1 pin function as a general-purpose I/O port. **Writing "1":** Makes the V1 pin function as an LCD drive power supply pin.
bit1, bit0	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.

Note:

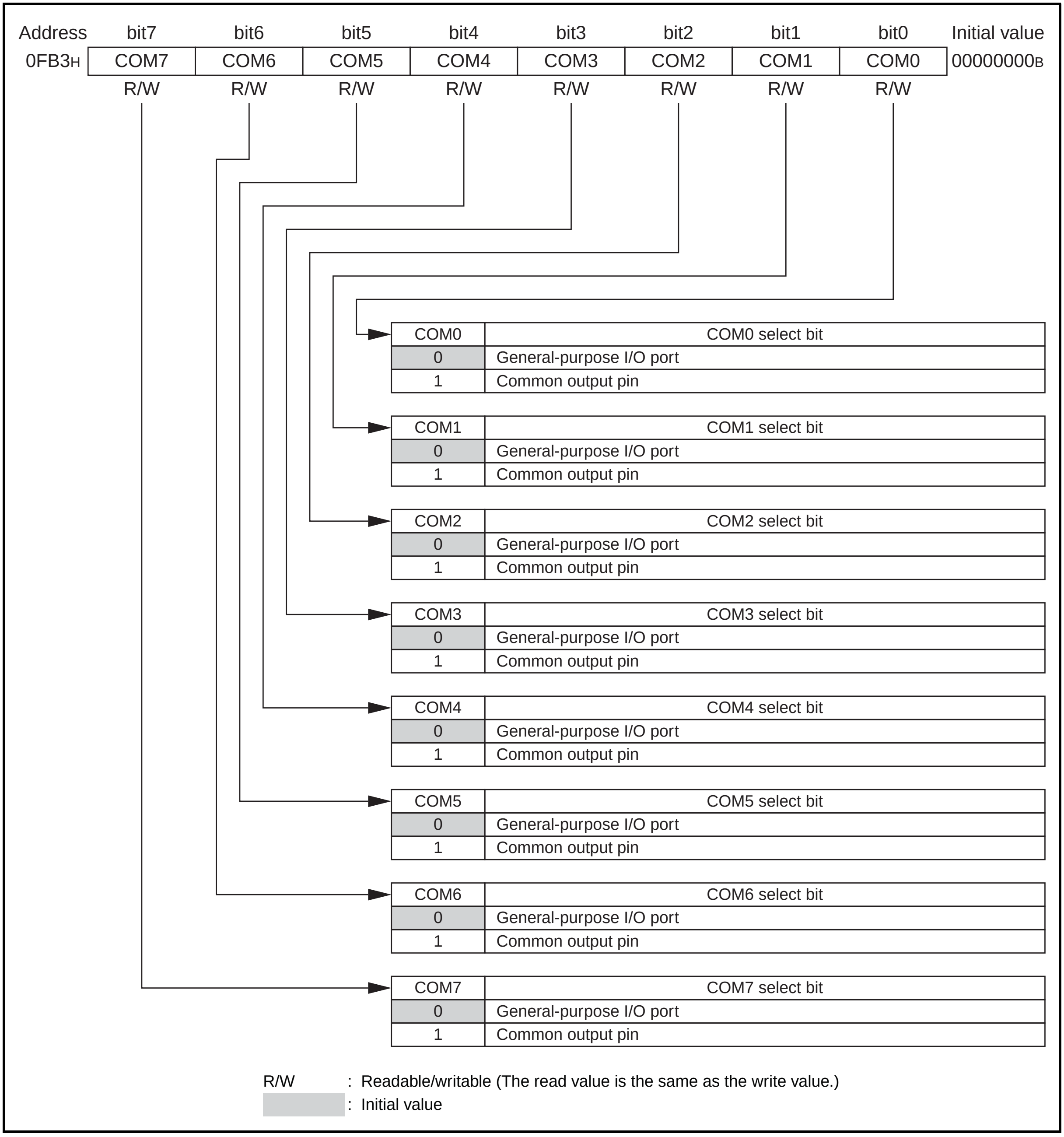
In the case of using the internal divider resistor, since the V4 pin cannot be used as a general-purpose I/O port, write "1" to the VE4 bit to make the V4 pin function as an LCD controller drive power supply pin.

29.4.4 LCDC Enable Register 2 (LCDCE2)

The LCDC enable register 2 (LCDCE2) is used to control the output of COM0 to COM7.

■ LCDC Enable Register 2 (LCDCE2)

Figure 29.4-5 LCDC Enable Register 2 (LCDCE2)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB3H	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

COM0	COM0 select bit
0	General-purpose I/O port
1	Common output pin

COM1	COM1 select bit
0	General-purpose I/O port
1	Common output pin

COM2	COM2 select bit
0	General-purpose I/O port
1	Common output pin

COM3	COM3 select bit
0	General-purpose I/O port
1	Common output pin

COM4	COM4 select bit
0	General-purpose I/O port
1	Common output pin

COM5	COM5 select bit
0	General-purpose I/O port
1	Common output pin

COM6	COM6 select bit
0	General-purpose I/O port
1	Common output pin

COM7	COM7 select bit
0	General-purpose I/O port
1	Common output pin

R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value

Table 29.4-4 Functions of Bits in LCDC Enable Register 2 (LCDCE2)

Bit name		Function
bit7	COM7: COM7 select bit	This bit selects the function of the COM7 pin. In 8 COM mode: **Writing "0":** Makes the COM7 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM7 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit6	COM6: COM6 select bit	This bit selects the function of the COM6 pin. In 8 COM mode: **Writing "0":** Makes the COM6 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM6 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit5	COM5: COM5 select bit	This bit selects the function of the COM5 pin. In 8 COM mode: **Writing "0":** Makes the COM5 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM5 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit4	COM4: COM4 select bit	This bit selects the function of the COM4 pin. In 8 COM mode: **Writing "0":** Makes the COM4 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM4 pin function as a common output pin. In 4 COM mode, writing a value to this bit has no effect on operation.
bit3	COM3: COM3 select bit	This bit selects the function of the COM3 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM3 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM3 pin function as a common output pin.
bit2	COM2: COM2 select bit	This bit selects the function of the COM2 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM2 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM2 pin function as a common output pin.
bit1	COM1: COM1 select bit	This bit selects the function of the COM1 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM1 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM1 pin function as a common output pin.
bit0	COM0: COM0 select bit	This bit selects the function of the COM0 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the COM0 pin function as a general-purpose I/O port. **Writing "1":** Makes the COM0 pin function as a common output pin.

29.4.5 LCDC Enable Register 3 to LCDC Enable Register 5 (LCDCE3 to LCDCE5)

The LCDC enable register 3 to the LCDC enable register 5 (LCDCE3 to LCDCE5) are used to control segment output pins SEG00 to SEG23.

■ LCDC Enable Register 3 to LCDCE Enable Register 5 (LCDCE3 to LCDCE5)

Figure 29.4-6 LCDC Enable Register 3 to LCDCE Register 5 (LCDCE3 to LCDCE5)

LCDC enable register 3 (LCDCE3)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB4_H$	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 4 (LCDCE4)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB5_H$	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC enable register 5 (LCDCE5)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB6_H$	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

SEGn*	SEGn* select bit
0	General-purpose I/O port
1	Segment output pin

▨ : Initial value
* : The letter “n” after SEG represents the number appearing in the bit name.

Note:

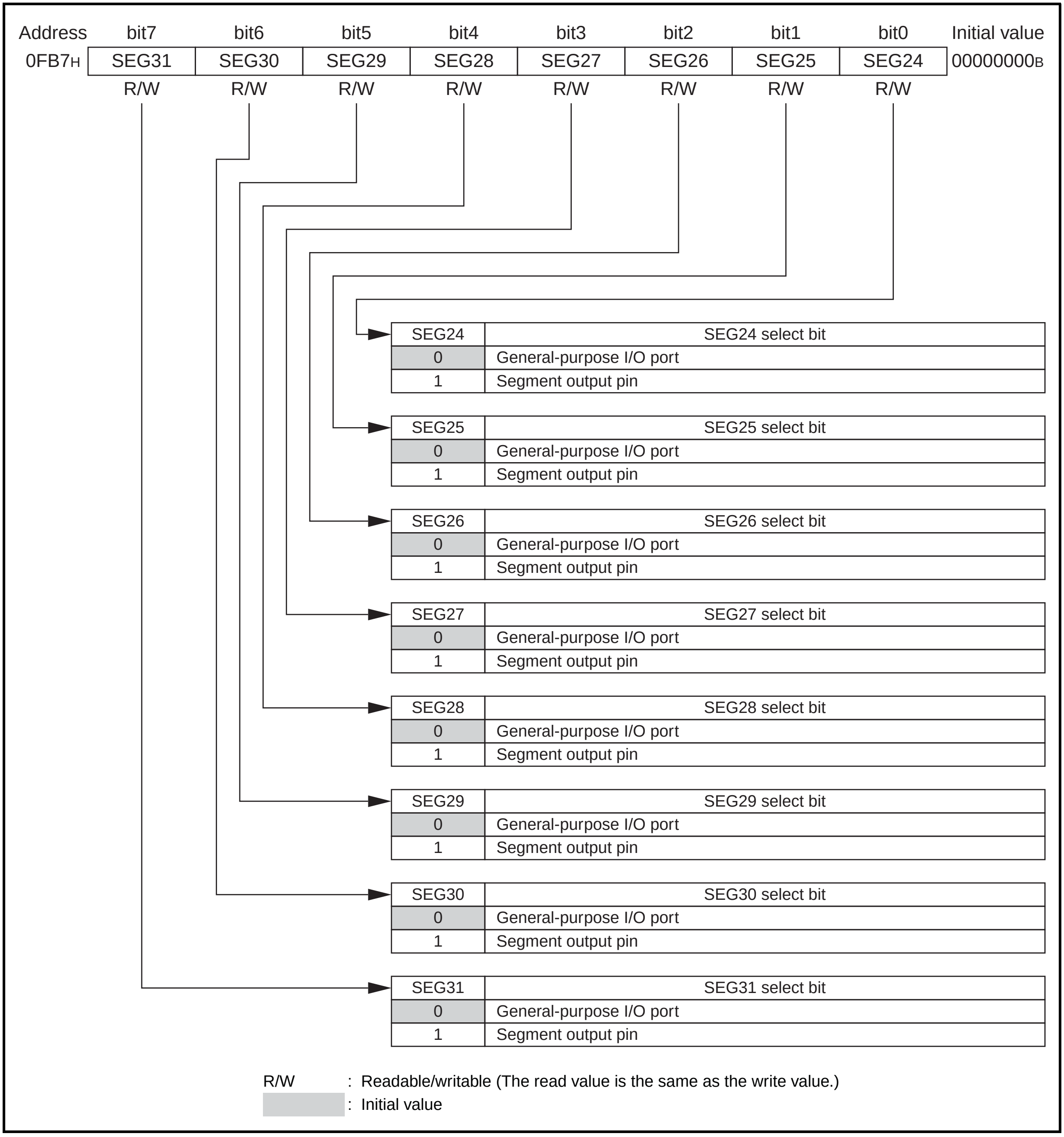
Only when PICTL is set to "1" are LCDCE3 to LCDCE5 enabled to control their corresponding segment output pins.

29.4.6 LCDCE Enable Register 6 (LCDCE6)

The LCDC enable register 6 (LCDCE6) is used to control segment output pins SEG24 to SEG31.

■ LCDC Enable Register 6 (LCDCE6)

Figure 29.4-7 LCDC Enable Register 7 (LCDCE6)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FB7H	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

SEG24	SEG24 select bit
0	General-purpose I/O port
1	Segment output pin

SEG25	SEG25 select bit
0	General-purpose I/O port
1	Segment output pin

SEG26	SEG26 select bit
0	General-purpose I/O port
1	Segment output pin

SEG27	SEG27 select bit
0	General-purpose I/O port
1	Segment output pin

SEG28	SEG28 select bit
0	General-purpose I/O port
1	Segment output pin

SEG29	SEG29 select bit
0	General-purpose I/O port
1	Segment output pin

SEG30	SEG30 select bit
0	General-purpose I/O port
1	Segment output pin

SEG31	SEG31 select bit
0	General-purpose I/O port
1	Segment output pin

R/W : Readable/writable (The read value is the same as the write value.)
▨ : Initial value

Table 29.4-5 Functions of Bits in LCDC Enable Register 6 (LCDCE6)

Bit name		Function
bit7	SEG31: SEG31 select bit	This bit selects the function of the SEG31 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG31 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG31 pin function as a segment output pin.
bit6	SEG30: SEG30 select bit	This bit selects the function of the SEG30 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG30 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG30 pin function as a segment output pin.
bit5	SEG29: SEG29 select bit	This bit selects the function of the SEG29 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG29 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG29 pin function as a segment output pin.
bit4	SEG28: SEG28 select bit	This bit selects the function of the SEG28 pin. In 8 COM mode, writing a value to this bit has no effect on operation. In 4 COM mode: **Writing "0":** Makes the SEG28 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG28 pin function as a segment output pin.
bit3	SEG27: SEG27 select bit	This bit selects the function of the SEG27 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG27 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG27 pin function as a segment output pin.
bit2	SEG26: SEG26 select bit	This bit selects the function of the SEG26 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG26 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG26 pin function as a segment output pin.
bit1	SEG25: SEG25 select bit	This bit selects the function of the SEG25 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG25 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG25 pin function as a segment output pin.
bit0	SEG24: SEG24 select bit	This bit selects the function of the SEG24 pin. In both 8 COM mode and 4 COM mode: **Writing "0":** Makes the SEG24 pin function as a general-purpose I/O port. **Writing "1":** Makes the SEG24 pin function as a segment output pin.

Note:

Only when PICTL is set to "1" is LCDCE6 enabled to control its corresponding segment output pins.

29.4.7 LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2)

The LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2) are used to turn on or off blinking.

■ LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2)

Figure 29.4-8 LCDC Blinking Setting Register 1, LCDC Blinking Setting Register 2 (LCDCB1, LCDCB2)

LCDC blinking setting register 1 (LCDCB1)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FB9_H$	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

LCDC blinking setting register 2 (LCDCB2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
$0FBA_H$	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value.)

BLDx*1	SnCm*2 blinking setting bit
▨ 0	Turns off the blinking of SnCm*2.
1	Turns on the blinking of SnCm*2.

▨ : Initial value
*1 : The letter “x” after BLD represents the number (0 to 15) appearing in the bit name.
*2 : Sn = SEGn (n represents one of the numbers from 00 to 03.)
Cm = COMm (m represents one of the numbers from 0 to 7.)

In 8 COM mode, the blinking function is applied to the dots specified in the combinations of SEG00 to SEG01 and COM0 to COM7.

In 4 COM mode, the blinking function is applied to the dots specified in the combinations of SEG00 to SEG03 and COM0 to COM3.

Select a blinking interval using the BLSEL bit in the LCDC enable register 1 (LCDCE1).

All segments for which blinking has been turned on will blink synchronously.

The setting of each blinking select bit remains in effect even when its corresponding bit in the display RAM holds "1".

29.5 LCD Controller Display RAM

**The display RAM size varies between 8 COM mode and 4 COM mode.
In 8 COM mode, the display RAM has 28 × 8 bits (28 bytes) of display data memory for generating segment output signals.
In 4 COM mode, the display RAM has 32 × 4 bits (16 bytes) of display data memory for generating segment output signals.**

■ Display RAM and Output Pins

The contents of display RAM are read automatically in sync with the common signal selection timing and output from the segment output pins.

Each bit containing "1" is converted to the selected voltage (displayed on the LCD); the one containing "0" is converted to the unselected voltage (undisplayed on the LCD).

As the LCD display operation is performed asynchronously with the CPU operation, display RAM can be read from or written to at any timing. When a pin shared between a segment output pin and a general-purpose I/O port is not used as a segment output pin, the pin can be used as a general-purpose I/O port, and the display RAM corresponding to such pin can be used as normal RAM. Table 29.5-1 shows the relationship between duty setting/common outputs and bits used in the display RAM.

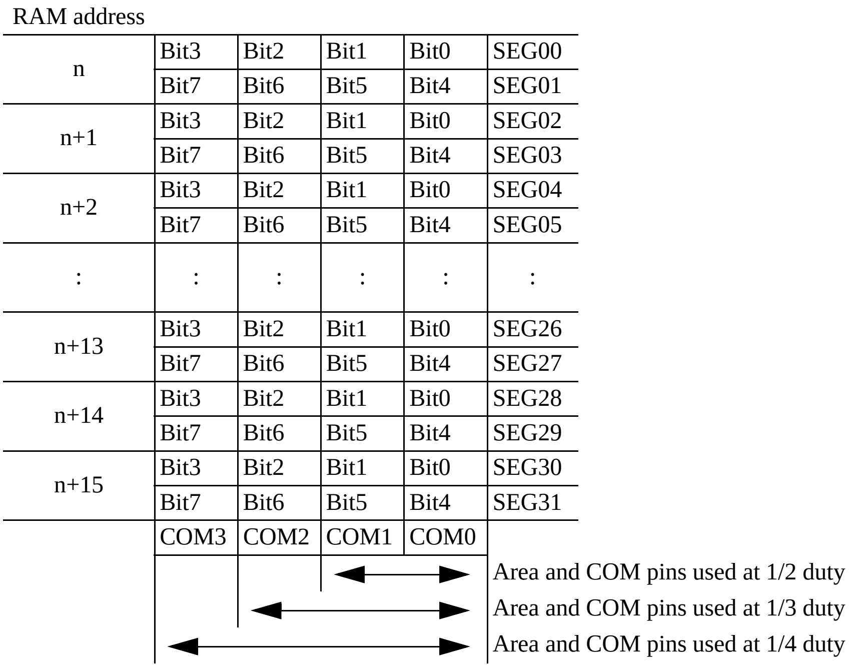
Figure 29.5-1 and Figure 29.5-2 shows how display RAM addresses are allocated for common output pins and segment output pins in 8 COM mode and in 4 COM mode respectively.

Figure 29.5-1 Display RAM and Common/Segment Output Pins in 8 COM Mode

RAM address									
n	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG00
n+1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG01
n+2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG02
n+3	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG03
n+4	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG04
:	:	:	:	:	:	:	:	:	:
n+22	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG22
n+23	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG23
n+24	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG24
n+25	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG25
n+26	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG26
n+27	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SEG27
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	

Area and COM pins used at 1/8 duty

Figure 29.5-2 Display RAM and Common/Segment output Pins in 4 COM Mode



RAM address					
n	Bit3	Bit2	Bit1	Bit0	SEG00
	Bit7	Bit6	Bit5	Bit4	SEG01
n+1	Bit3	Bit2	Bit1	Bit0	SEG02
	Bit7	Bit6	Bit5	Bit4	SEG03
n+2	Bit3	Bit2	Bit1	Bit0	SEG04
	Bit7	Bit6	Bit5	Bit4	SEG05
:	:	:	:	:	:
n+13	Bit3	Bit2	Bit1	Bit0	SEG26
	Bit7	Bit6	Bit5	Bit4	SEG27
n+14	Bit3	Bit2	Bit1	Bit0	SEG28
	Bit7	Bit6	Bit5	Bit4	SEG29
n+15	Bit3	Bit2	Bit1	Bit0	SEG30
	Bit7	Bit6	Bit5	Bit4	SEG31
	COM3	COM2	COM1	COM0	

Area and COM pins used at 1/2 duty
Area and COM pins used at 1/3 duty
Area and COM pins used at 1/4 duty

Note:

"n" in the address column stands for "0FBD$_H$".

Table 29.5-1 Relationship Between Duty Settings/Common Outputs and Display RAM Bits Used

Duty setting	Common output pins used	Display data bits used							
		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
1/2	COM0, COM1 (2 pins)	-	-	○	○	-	-	○	○
1/3	COM0 to COM2 (3 pins)	-	○	○	○	-	○	○	○
1/4	COM0 to COM3 (4 pins)	○	○	○	○	○	○	○	○
1/8	COM0 to COM7 (8 pins)	○	○	○	○	○	○	○	○

○ : Bit used
- : Bit not used

29.6 Interrupts of LCD Controller

The LCD controller generates interrupts in sync with the LCD module frame frequency.

■ Interrupt during LCD Controller Operation

Upon completing a frame, the LCD controller sets the LCDC interrupt request flag bit (LCDCC2:LCDIF) to "1". If the interrupt request has already been enabled (LCDCC2:LCDIEN = 1) when the LCDIF bit is set to "1", the LCD controller will make an interrupt request to the interrupt controller. To clear an interrupt request, write "0" to the LCDIF bit in the interrupt service routine.

The LCD controller always sets the LCDIF bit to "1" upon completing a frame, regardless of the value of the LCDIEN bit. If both the LCDIF bit and the LCDIEN bit remain "1" after an LCDC interrupt request is made, the CPU cannot return from interrupt processing. To enable the CPU to return from interrupt processing, always clear the LCDIF bit to "0" after an LCDC interrupt request is made.

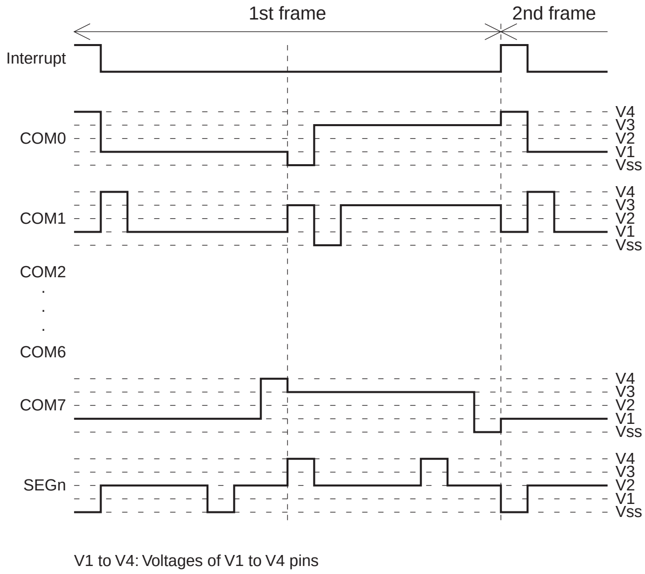
■ Register and Vector Table Addresses Related to LCD Controller Interrupts

Table 29.6-1 Register and Vector Table Addresses Related to LCD Controller Interrupts

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
LCD controller	IRQ08	ILR2	L08	$FFEA_H$	$FFEB_H$

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

Figure 29.6-1 Interrupt Timing Chart



29.7 Operations of LCD Controller

This section describes the operations of the LCD controller.

■ Operations of LCD Controller

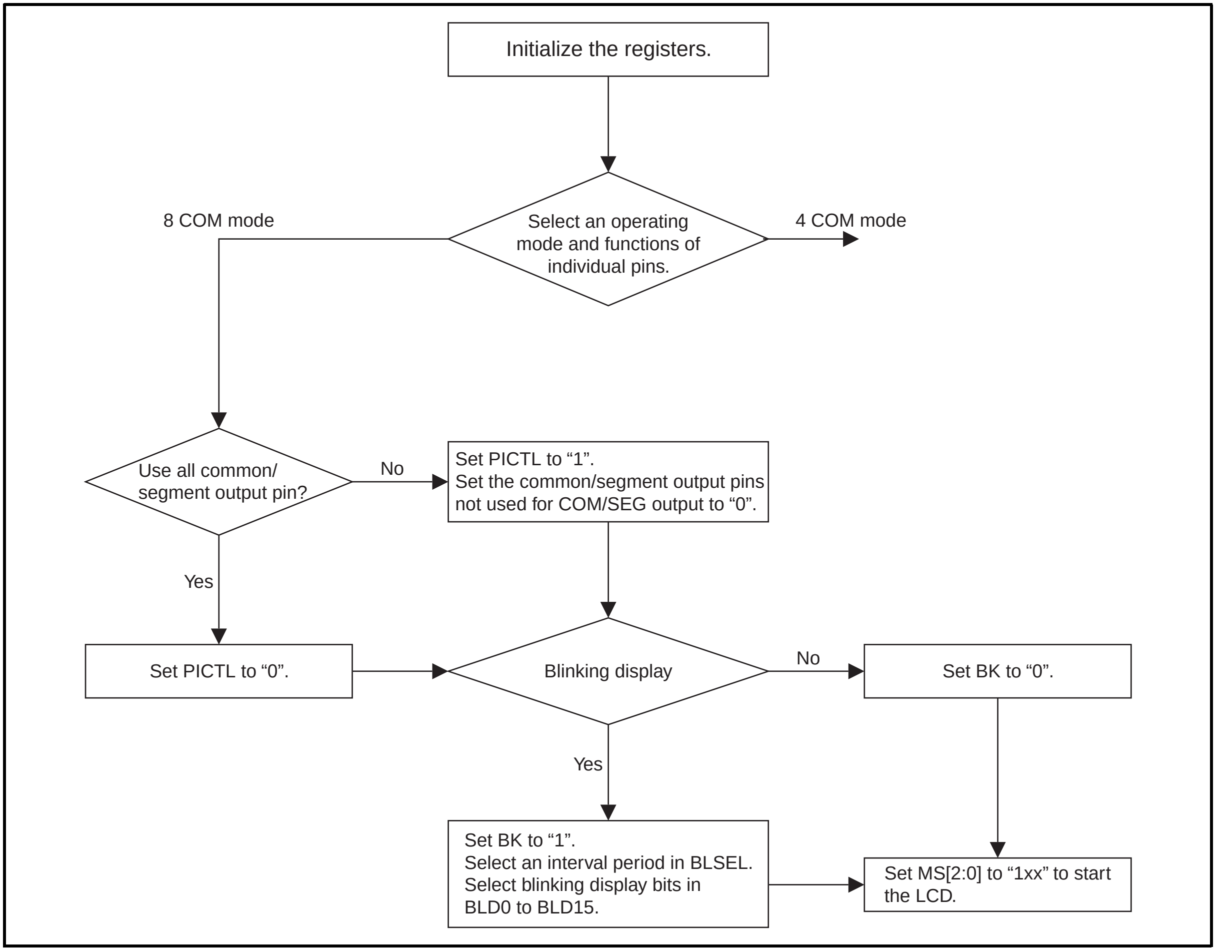
Figure 29.7-1 shows the settings required for LCD display.

Figure 29.7-1 LCD Controller Settings in 8 COM Mode

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
LCDCC1	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0
	❍	❍	❍	1	0/1	0/1	❍	❍
LCDCC2	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF
	-	-	❍	❍	❍	❍	❍	❍
LCDCE1	PICTL	BLSEL	VE4	VE3	VE2	VE1	-	-
	❍	❍	❍	❍	❍	❍	-	-
LCDCE2	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE3	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE4	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE5	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE6	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24
	-	-	-	-	❍	❍	❍	❍
LCDCB1	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCB2	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8
	❍	❍	❍	❍	❍	❍	❍	❍
Display RAM	Display data							

❍ : Bit used
- : Bit not used
1 : Write "1".
0/1 : Write "0" or "1".

Figure 29.7-2 LCD Controller Setting Flow in 8 COM Mode


Initialize the registers.
8 COM mode
Select an operating mode and functions of individual pins.
4 COM mode
Use all common/ segment output pin?
No
Set PICTL to "1". Set the common/segment output pins not used for COM/SEG output to "0".
Yes
Set PICTL to "0".
Blinking display
No
Set BK to "0".
Yes
Set BK to "1". Select an interval period in BLSEL. Select blinking display bits in BLD0 to BLD15.
Set MS[2:0] to "1xx" to start the LCD.

- When the selected frame period generation clock is oscillating with the settings made as shown in Figure 29.7-1, the LCD controller outputs the LCD panel drive waveform to the common and segment output pins (COM0 to COM7, SEG00 to SEG27) according to the contents of the display RAM and the settings of different LCDC registers.
- The LCD output pins are selected according to LCDCE1 to LCDCE6. Pins not selected as common/segment output pins are used as general-purpose I/O ports.
- The frame period generation clock can be changed even during LCD display operation. As the LCD display may flicker when the frame period generation clock is changed, always turn off the LCD display temporarily, for example, using the display blanking (LCDCC2:BK = 1) function, before changing the frame period generation clock to prevent the LCD display from flickering.
- The display drive output is a 2-frame alternating waveform selected according to bias and duty settings.
- To use the blink function, set the corresponding bits in the LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2) to "1" to turn on the blinking function. The blinking interval can be selected from two options in the BLSEL bit in the LCDC enable register 1 (LCDCE1).

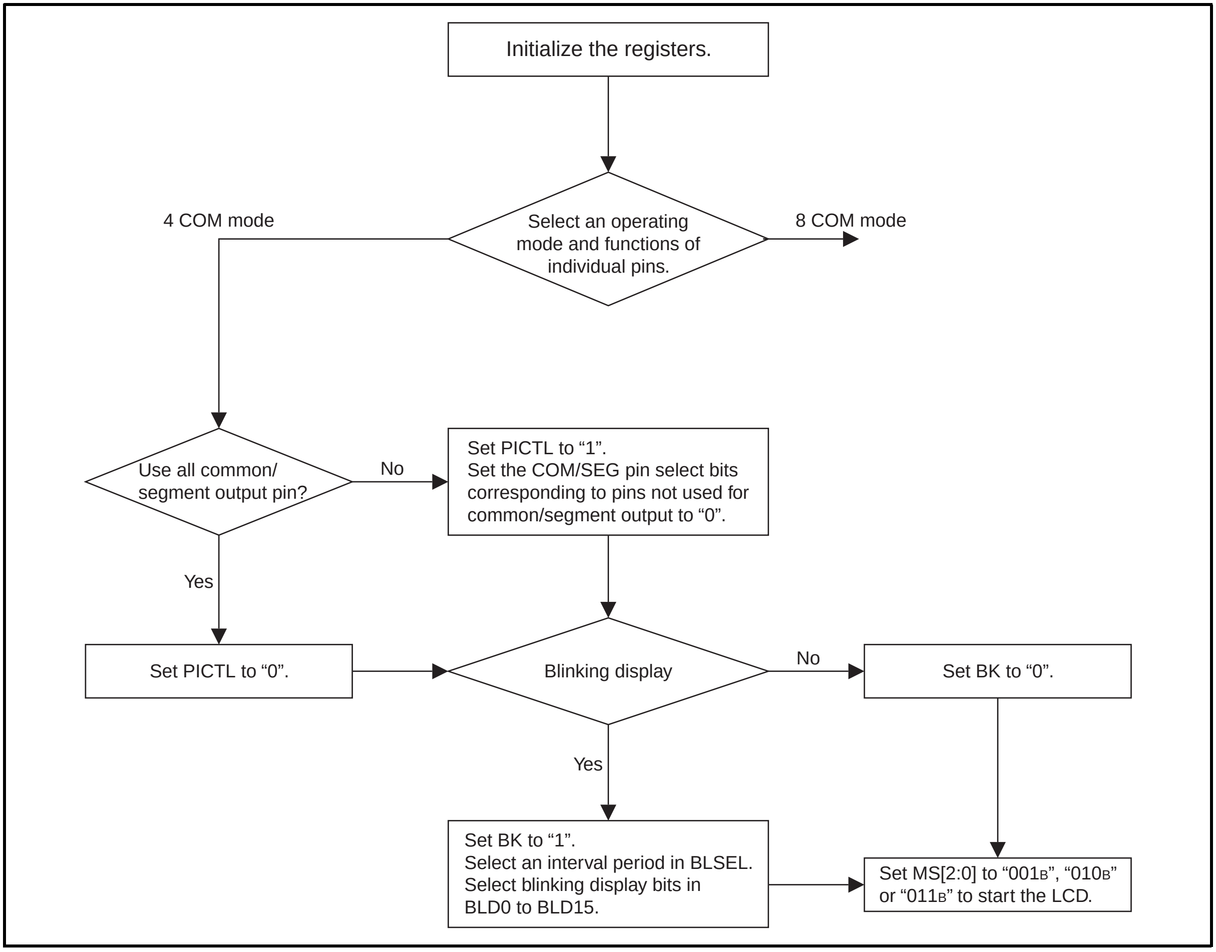
Figure 29.7-3 shows the settings required for LCD display in 4 COM mode.

Figure 29.7-3 LCD Controller Settings in 4 COM Mode

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
LCDCC1	CSS	LCDEN	VSEL	MS2	MS1	MS0	FP1	FP0
	❍	❍	❍	$001_B/010_B/011_B$			❍	❍
LCDCC2	-	-	RSEL	BLS8	INV	BK	LCDIEN	LCDIF
	-	-	❍	❍	❍	❍	❍	❍
LCDCE1	PICTL	BLSEL	VE4	VE3	VE2	VE1	-	-
	❍	❍	❍	❍	❍	❍	-	-
LCDCE2	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	-	-	-	-	❍	❍	❍	❍
LCDCE3	SEG07	SEG06	SEG05	SEG04	SEG03	SEG02	SEG01	SEG00
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE4	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG09	SEG08
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE5	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCE6	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCB1	BLD7	BLD6	BLD5	BLD4	BLD3	BLD2	BLD1	BLD0
	❍	❍	❍	❍	❍	❍	❍	❍
LCDCB2	BLD15	BLD14	BLD13	BLD12	BLD11	BLD10	BLD9	BLD8
	❍	❍	❍	❍	❍	❍	❍	❍
Display RAM	Display data							

❍ : Bit used
- : Bit not used.

Figure 29.7-4 LCD Controller Setting Flow in 4 COM Mode


Initialize the registers.
Select an operating mode and functions of individual pins.
4 COM mode
8 COM mode
Use all common/ segment output pin?
No
Set PICTL to “1”. Set the COM/SEG pin select bits corresponding to pins not used for common/segment output to “0”.
Yes
Set PICTL to “0”.
Blinking display
No
Set BK to “0”.
Yes
Set BK to “1”. Select an interval period in BLSEL. Select blinking display bits in BLD0 to BLD15.
Set MS[2:0] to “001_B”, “010_B” or “011_B” to start the LCD.

- When the selected frame period generation clock is oscillating with the settings made as shown in Figure 29.7-3, the LCD controller outputs the LCD panel drive waveform to the common and segment output pins (COM0 to COM3, SEG00 to SEG31) according to the contents of the display RAM and the settings of different LCDC registers.
- The LCD output pins are selected according to LCDCE1 to LCDCE6. Pins not selected as common/segment output pins are used as general-purpose I/O ports.
- The frame period generation clock can be changed even during LCD display operation. As the LCD display may flicker when the frame period generation clock is changed, always turn off the LCD display temporarily, for example, using the display blanking (LCDCC2:BK = 1) function, before changing the frame period generation clock to prevent the LCD display from flickering.
- The display drive output is a 2-frame alternating waveform selected according to bias and duty settings.
- The COM2 and COM3 pin outputs in 1/2 duty mode and the COM3 pin output in 1/3 duty mode can be used to output the deselected level waveform or as I/O ports.
- To use the blink function, set the corresponding bits in the LCDC blinking setting register 1 (LCDCB1) and the LCDC blinking setting register 2 (LCDCB2) to "1" to turn on the blinking function. The blinking interval can be selected from two options in the BLSEL bit in the LCDC enable register 1 (LCDCE1).

Note:

If the selected frame period generation clock halts during LCD display operation, the AC waveform generator circuit also halts and therefore a DC voltage is applied to the liquid crystal elements. In this case, the LCD display operation must be stopped in advance. The conditions under which the main clock (time-base timer) or subclock (watch prescaler) halts depend on the selected clock mode and standby mode. The frame period is also affected if the time-base timer or watch prescaler is cleared depending on the setting of the frame period generation clock select bit (LCDCC1:CSS).

■ LCD Drive Waveform

Due to the characteristics of the LCD, DC driving of the LCD chemically changes and degrades the liquid crystal display elements. Therefore, the LCD controller driver contains an AC waveform generator circuit to drive the LCD using a 2-frame alternating waveform. There are five types of output waveform as follows:

In 8 COM mode:

- 1/4 bias, 1/8 duty output waveform
- 1/3 bias, 1/8 duty output waveform

In 4 COM mode:

- 1/2 bias, 1/2 duty output waveform
- 1/3 bias, 1/3 duty output waveform
- 1/3 bias, 1/4 duty output waveform

29.7.1 Output Waveform in LCD Controller Operation in 4 COM Mode (1/2 Bias, 1/2 Duty)

The display drive output is a multiplex drive type of 2-frame alternating waveform.
In 4 COM mode with 1/2 bias and 1/2 duty, only COM0 and COM1 are used for display; neither COM2 nor COM3 is used.

■ 4 COM Mode, 1/2 Bias, 1/2 Duty Output Waveform Example

Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

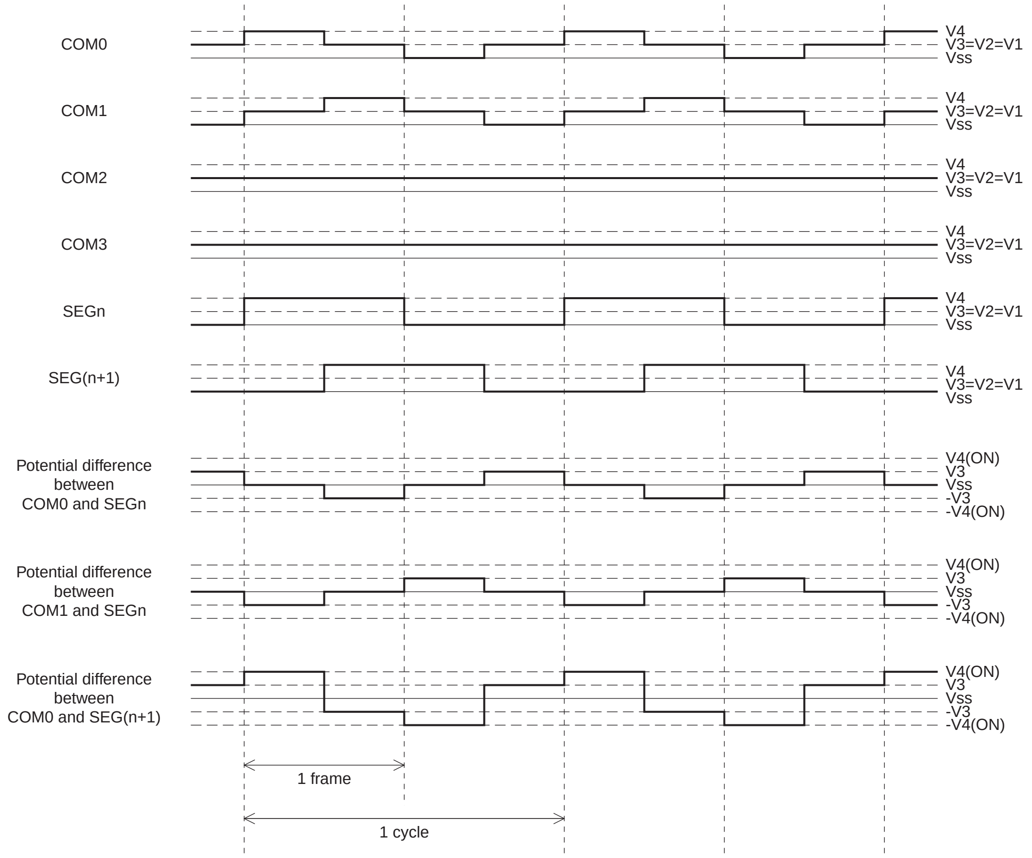
Figure 29.7-5 shows the output waveform when the contents of display RAM are those shown in Table 29.7-1.

Table 29.7-1 Sample Contents of Display RAM

Segment	Contents of Display RAM			
	COM3	COM2	COM1	COM0
SEGn	-	-	0	0
SEG(n+1)	-	-	0	1

-: Unused

Figure 29.7-5 4 COM Mode, 1/2 Bias, 1/2 Duty Output Waveform Example



V1 to V4: Respective voltages of V1 to V4 pins

29.7.2 Output Waveform in LCD Controller Operation in 4 COM Mode (1/3 Bias, 1/3 Duty)

In 4 COM mode with 1/3 bias and 1/3 duty, COM0, COM1, and COM2 are used for display; COM3 is not used.

■ 4 COM Mode, 1/3 Bias, 1/3 Duty Output Waveform Example

Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

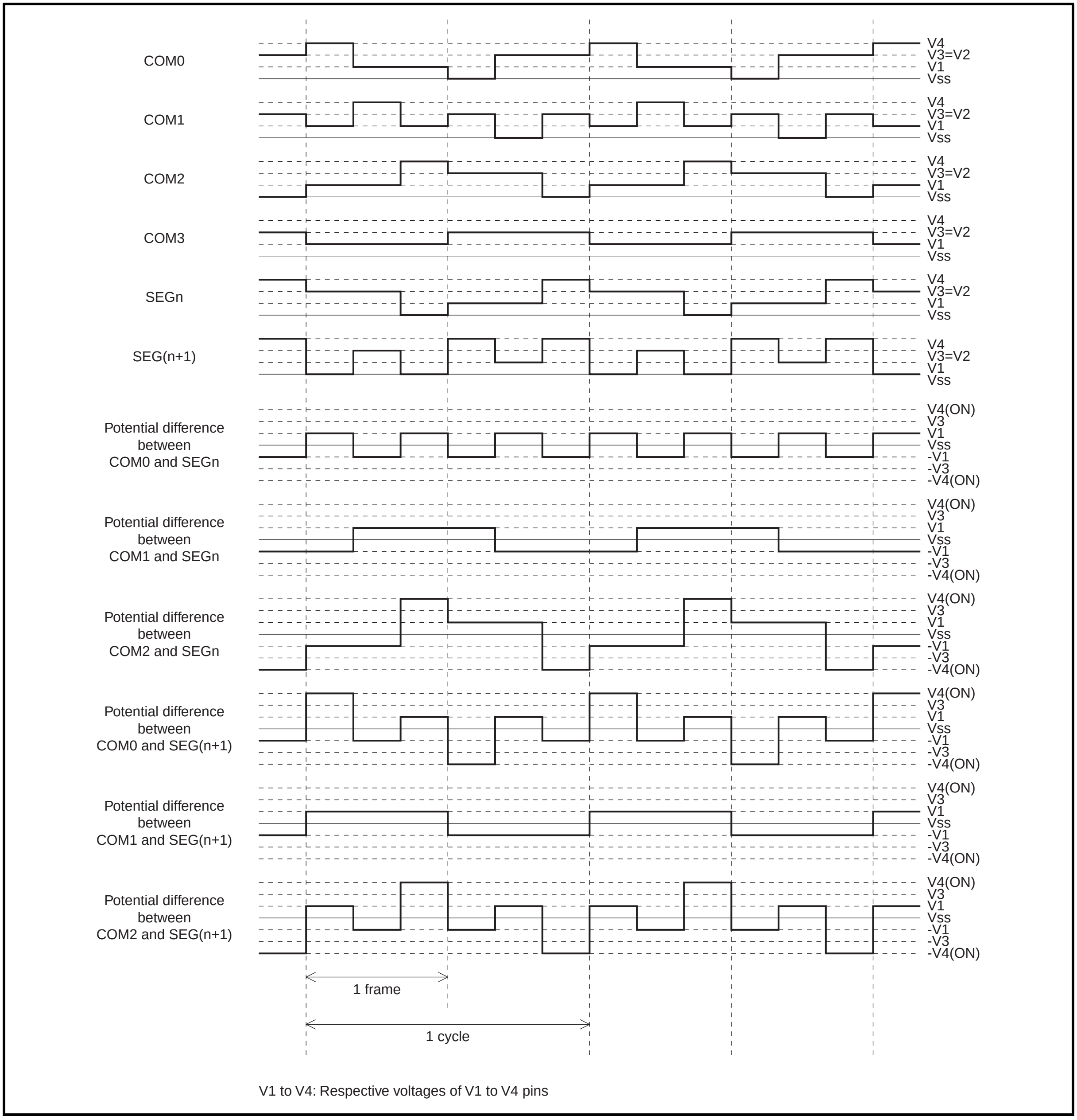
Figure 29.7-6 shows the output waveform when the contents of display RAM are those shown in Table 29.7-2.

Table 29.7-2 Sample Contents of Display RAM

Segment	Contents of Display RAM			
	COM3	COM2	COM1	COM0
SEGn	-	1	0	0
SEG(n+1)	-	1	0	1

-: Unused

Figure 29.7-6 4 COM Mode, 1/3 Bias, 1/3 Duty Output Waveform Example


COM0
COM1
COM2
COM3
SEGn
SEG(n+1)
Potential difference between COM0 and SEGn
Potential difference between COM1 and SEGn
Potential difference between COM2 and SEGn
Potential difference between COM0 and SEG(n+1)
Potential difference between COM1 and SEG(n+1)
Potential difference between COM2 and SEG(n+1)
V4
V3=V2
V1
Vss
V4(ON)
V3
V1
Vss
-V1
-V3
-V4(ON)
1 frame
1 cycle
V1 to V4: Respective voltages of V1 to V4 pins

29.7.3 Output Waveform in LCD Controller Operation in 4 COM Mode (1/3 Bias, 1/4 Duty)

In 4 COM Mode with 1/3 bias and 1/4 duty, COM0 to COM3 are used for display.

■ 4 COM Mode, 1/3 Bias, 1/4 Duty Output Waveform Example

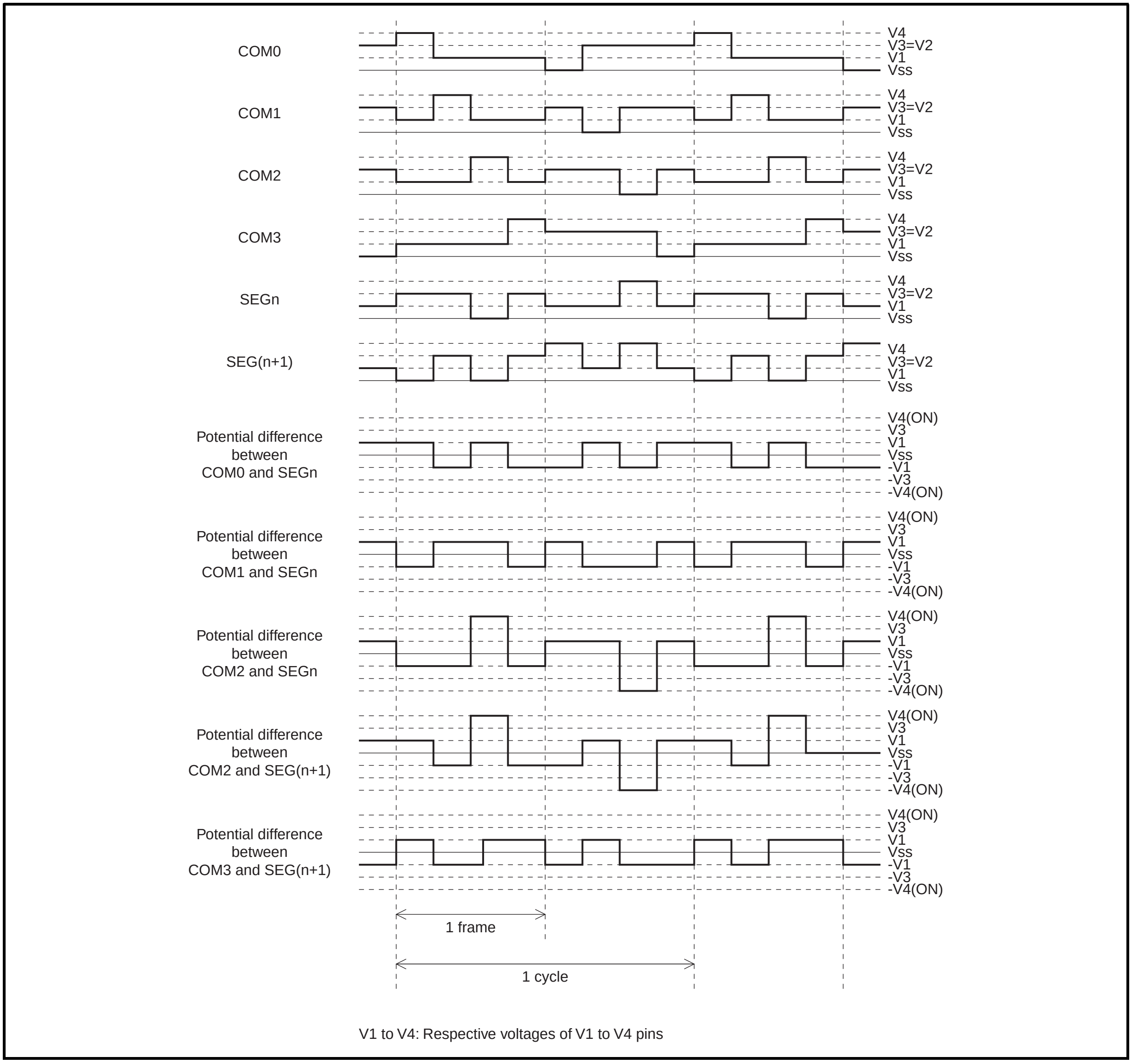
Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

Figure 29.7-7 shows the output waveform when the contents of display RAM are those shown in Table 29.7-3.

Table 29.7-3 Sample Contents of Display RAM

Segment	Contents of Display RAM			
	COM3	COM2	COM1	COM0
SEGn	0	1	0	0
SEG(n+1)	0	1	0	1

Figure 29.7-7 4 COM Mode, 1/3 Bias, 1/4 Duty Output Waveform Example


COM0
COM1
COM2
COM3
SEGn
SEG(n+1)
V4
V3=V2
V1
Vss
Potential difference between COM0 and SEGn
Potential difference between COM1 and SEGn
Potential difference between COM2 and SEGn
Potential difference between COM2 and SEG(n+1)
Potential difference between COM3 and SEG(n+1)
V4(ON)
V3
V1
Vss
-V1
-V3
-V4(ON)
1 frame
1 cycle

V1 to V4: Respective voltages of V1 to V4 pins

29.7.4 Output Waveform in LCD Controller Operation in 8 COM Mode (1/4 Bias, 1/8 Duty)

In 8 COM Mode with 1/4 bias and 1/8 duty, COM0 to COM7 are used for display.

■ 8 COM Mode, 1/4 Bias, 1/8 Duty Output Waveform Example

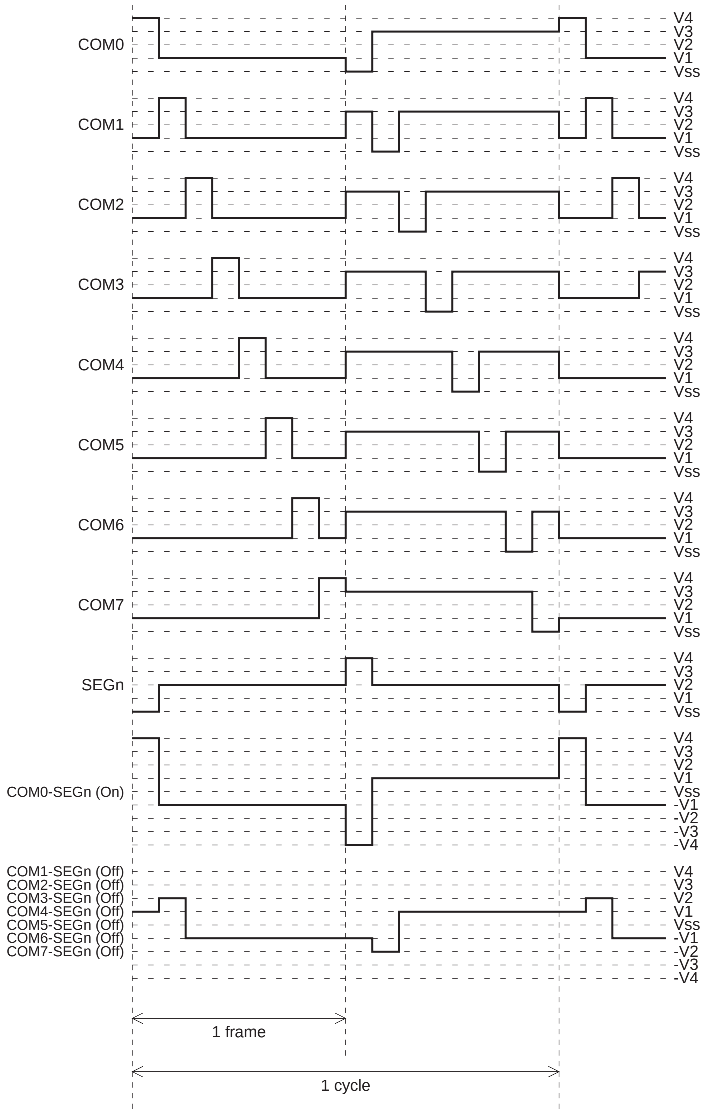
Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

Figure 29.7-8 shows the output waveform when the contents of display RAM are those shown in Table 29.7-4.

Table 29.7-4 Sample Contents of Display RAM

Segment	Contents of Display RAM							
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEGn	0	0	0	0	0	0	0	1

Figure 29.7-8 8 COM Mode, 1/4 Bias, 1/8 Duty Output Waveform Example



V0 to V4: Voltages of V0 to V4 pins

29.7.5 Output Waveform in LCD Controller Operation in 8 COM Mode (1/3 Bias, 1/8 Duty)

In 8 COM Mode with 1/3 bias and 1/8 duty, COM0 to COM7 are used for display.

■ 8 COM Mode, 1/3 Bias, 1/8 Duty Output Waveform Example

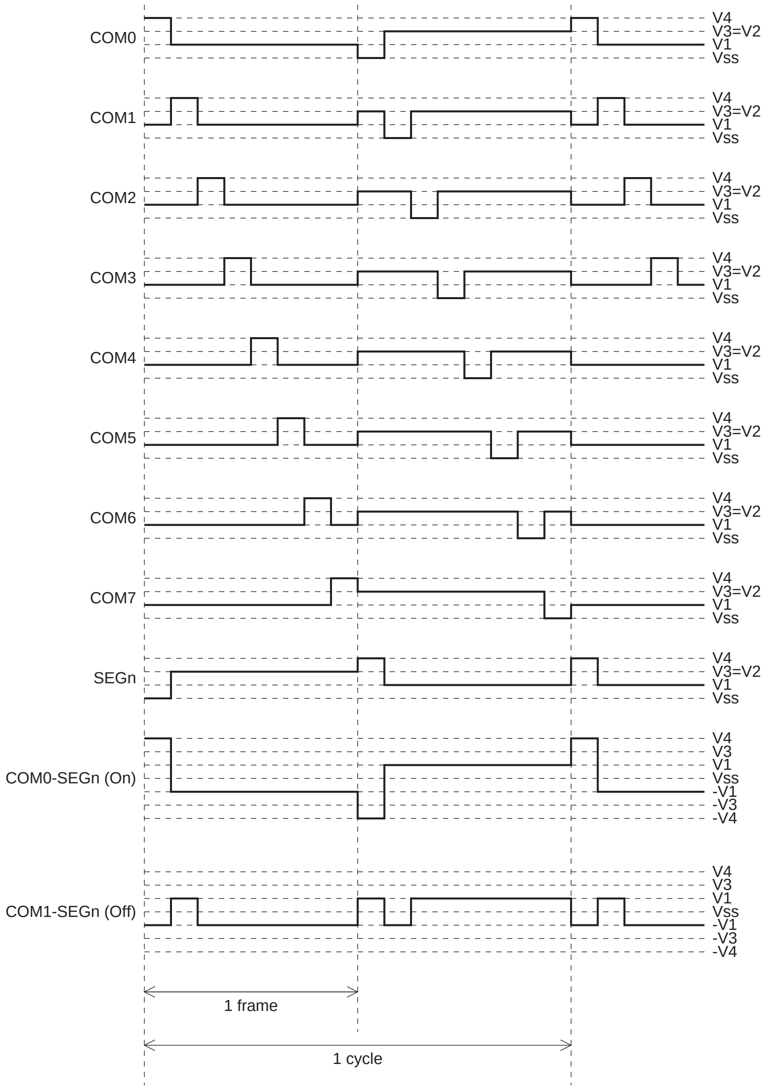
Liquid crystal elements are turned "ON" for display that have the maximum potential difference between the common and segment outputs.

Figure 29.7-9 shows the output waveform when the contents of display RAM are those shown in Table 29.7-5.

Table 29.7-5 Sample Contents of Display RAM

Segment	Contents of Display RAM							
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEGn	0	0	0	0	0	0	0	1

Figure 29.7-9 8 COM Mode, 1/3 Bias, 1/8 Duty Output Waveform Example



V1 to V4: Voltages of V1 to V4 pins

29.8 Notes on Using LCD Controller

This section provides notes on using the LCD controller.

■ Notes on Using LCD Controller

- To use an LCD pin as a general-purpose I/O port, set a corresponding common/segment select bit in an LCDC enable register (LCDCE1 to LCDCE6) to "0", and then set the port input control bit (PICTL) in the LCDC enable register 1 (LCDCE1) to "1".
- If the selected frame period generation clock halts during LCD display operation, the AC waveform generator circuit also halts and therefore a DC voltage is applied to the liquid crystal elements. In this case, the LCD display operation must be stopped in advance. The conditions under which the main clock (time-base timer) or subclock (watch prescaler) halts depend on the selected clock mode and standby mode. The frame period is also affected if the time-base timer or watch prescaler is cleared according to the setting of the frame period generation clock select bit (LCDCC1:CSS).
- The operation of outputting display RAM data to the LCD is not in sync with the CPU accessing to the display RAM. When the interval for rewriting the display RAM is shorter than the LCD cycle, flickers may occur that are caused by different display patterns between frames.

CHAPTER 30

DUAL OPERATION FLASH MEMORY

This chapter describes the function and operations of the 160/288/480 kbit dual operation Flash memory.

30.1 Overview of Dual Operation Flash Memory

The dual operation Flash memory is located at 1000_H to $1FFF_H$ and $C000_H$ to $FFFF_H$ for 160 kbit Flash memory, at 1000_H to $1FFF_H$ and 8000_H to $FFFF_H$ for 288 kbit Flash memory or at 1000_H to $FFFF_H$ for 480 kbit Flash memory on the CPU memory map.

The dual operation Flash consists of an upper bank and a lower bank*. Unlike conventional Flash products, writing/erasing data to/from one bank and reading data from another bank can be executed simultaneously.

***: MB95F418H/F418K/F478H/F478K:**
upper bank: 16 Kbyte × 3 + 8 Kbyte × 1; lower bank: 2 Kbyte × 2
MB95F416H/F416K/F476H/F476K:
upper bank: 16 Kbyte × 2; lower bank: 2 Kbyte × 2
MB95F414H/F414K/F474H/F474K:
upper bank: 16 Kbyte × 1; lower bank: 2 Kbyte × 2

■ Overview of Dual Operation Flash Memory

The following methods can be used to write data into and erase data from the Flash memory:

- Writing/erasing using a dedicated serial programmer
- Writing/erasing by program execution

Since data can be written into and erased from the dual operation Flash memory by instructions from the CPU via the Flash memory interface circuit, program code and data can be efficiently updated with the device mounted on a circuit board. The minimum sector size of the dual operation Flash is 2 Kbyte, which is a type of sector configuration facilitating the management of the program/data area.

Data can be updated by executing a program in RAM or by executing a program in the Flash memory in dual operation mode. The erase/write operation and the read operation can be executed in different banks (upper bank/lower bank) simultaneously.

The dual operation Flash can use the following combinations:

Upper bank	Lower bank
Read	
Read	Write/sector erase
Write/sector erase	Read
Chip erase	

■ Features of Dual Operation Flash Memory

- Sector configuration:
  - 20 Kbyte (16 Kbyte + 2 Kbyte × 2)
  - 36 Kbyte (16 Kbyte × 2 + 2 Kbyte × 2)
  - 60 Kbyte (16 Kbyte × 3 + 8 Kbyte + 2 Kbyte × 2)
- Two-bank configuration, enabling simultaneous execution of an erase/program and a read
- Automatic program algorithm (Embedded Algorithm)
- Erase suspend/resume function integrated
- Detection of completion of writing/erasing using the data polling or toggle bit function
- Detection of completion of writing/erasing by CPU interrupts
- Capable of erasing data from specific sectors (any combination of sectors)
- Writing/erase count: 100000 times
- Flash read cycle time (minimum): 1 machine cycle

■ Writing and Erasing Flash Memory

- Writing data to and reading data from the same bank of the Flash memory cannot be executed simultaneously.
- To write data to or erase data from a bank in the Flash memory, execute either the program for writing/erasing stored in another bank, or copy the program on the Flash memory to the RAM first and then execute it.
- The dual operation Flash memory enables program execution in the Flash memory and write control using interrupts. In addition, it is not necessary to download a program to RAM in order to write data to a bank, thereby reducing the time of program download and eliminating the need to protect RAM data against power interruption.

30.2 Sector/Bank Configuration of Flash Memory

This section explains the registers and the sector/bank configuration of Flash memory.

■ Sector/Bank Configuration of Dual Operation Flash Memory

Figure 30.2-1 shows the sector configuration of the dual operation Flash memory. The upper and lower addresses of each sector are shown in the figure.

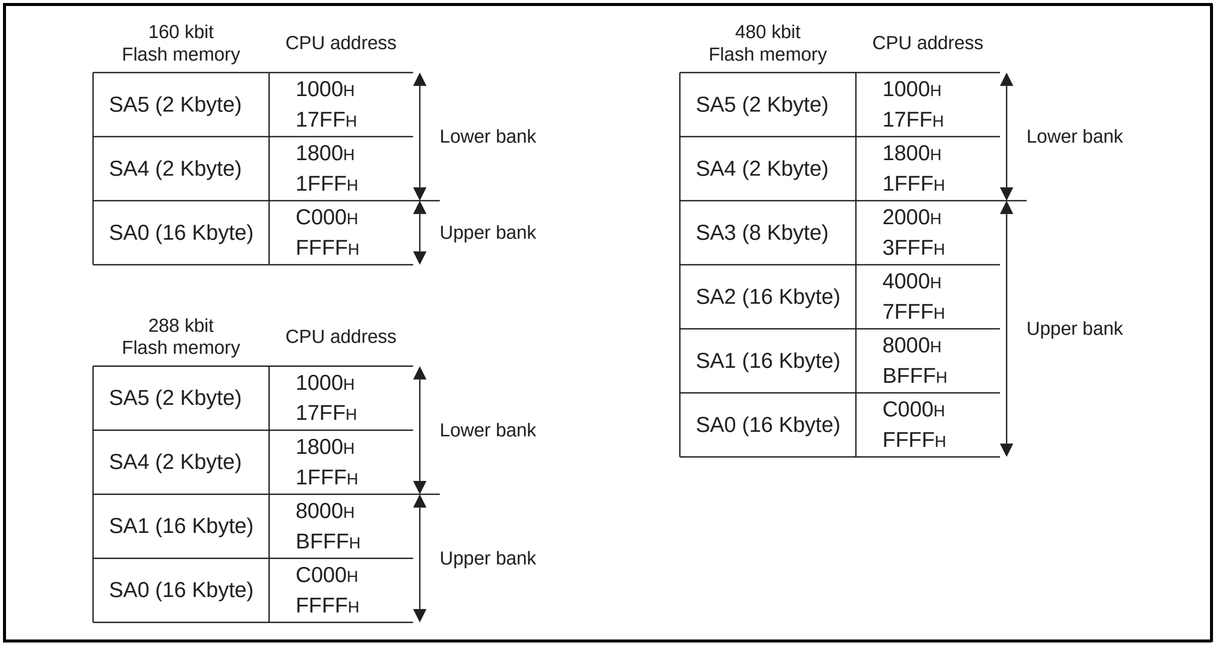
● Sector configuration

In the case of accessing the Flash memory from the CPU, SA0 is located $C000_H$ to $FFFF_H$, SA1 at 8000_H to $BFFF_H$, SA2 at 4000_H to $7FFF_H$, SA3 at 2000_H to $3FFF_H$, SA4 at 1800_H to $1FFF_H$ and SA5 at 1000_H to $17FF_H$. SA1, SA2 and SA3 cannot be accessed in the 160 kbit Flash memory; SA2 and SA3 cannot be accessed in the 288 kbit Flash memory.

● Bank configuration

The 160 kbit Flash memory consists of the lower bank from SA5 to SA4 and the upper bank of SA0. The 288 kbit Flash memory consists of the lower bank from SA5 to SA4 and the upper bank from SA1 to SA0. The 480 kbit Flash memory consists of the lower bank from SA5 to SA4 and the upper bank from SA3 to SA0.

Figure 30.2-1 Sector/Bank Configuration of Dual Operation Flash Memory



160 kbit Flash memory	CPU address	
SA5 (2 Kbyte)	1000_H $17FF_H$	Lower bank
SA4 (2 Kbyte)	1800_H $1FFF_H$	
SA0 (16 Kbyte)	$C000_H$ $FFFF_H$	Upper bank

288 kbit Flash memory	CPU address	
SA5 (2 Kbyte)	1000_H $17FF_H$	Lower bank
SA4 (2 Kbyte)	1800_H $1FFF_H$	
SA1 (16 Kbyte)	8000_H $BFFF_H$	Upper bank
SA0 (16 Kbyte)	$C000_H$ $FFFF_H$	

480 kbit Flash memory	CPU address	
SA5 (2 Kbyte)	1000_H $17FF_H$	Lower bank
SA4 (2 Kbyte)	1800_H $1FFF_H$	
SA3 (8 Kbyte)	2000_H $3FFF_H$	Upper bank
SA2 (16 Kbyte)	4000_H $7FFF_H$	
SA1 (16 Kbyte)	8000_H $BFFF_H$	
SA0 (16 Kbyte)	$C000_H$ $FFFF_H$	

30.3 Registers of Flash Memory

This section shows the registers of the Flash memory.

■ Registers of Flash Memory

Figure 30.3-1 Registers of Flash Memory

Flash memory status register 2 (FSR2)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0071_H	PEIEN	PGMEND	PTIEN	PGMTO	EEIEN	ERSEND	ETIEN	ERSTO	00000000_B
	R/W	R(RM1),W	R/W	R(RM1),W	R/W	R(RM1),W	R/W	R(RM1),W	

Flash memory status register (FSR)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0072_H	-	-	RDYIRQ	RDY	Reserved	IRQEN	WRE	SSEN	$000X0000_B$
	R0/WX	R0/WX	R(RM1),W	R/WX	R/W0	R/W	R/W	R/W	

Flash memory sector write control register 0 (SWRE0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0073_H	Reserved	Reserved	SA5E	SA4E	SA3E	SA2E	SA1E	SA0E	00000000_B
	R/W0	R/W0	R/W	R/W	R/W	R/W	R/W	R/W	

Flash memory status register 3 (FSR3)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0074_H	Reserved	-	-	ERIP	ESPS	SERS	PGMS	HANG	00000000_B
	R/W0	R0/WX	R0/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

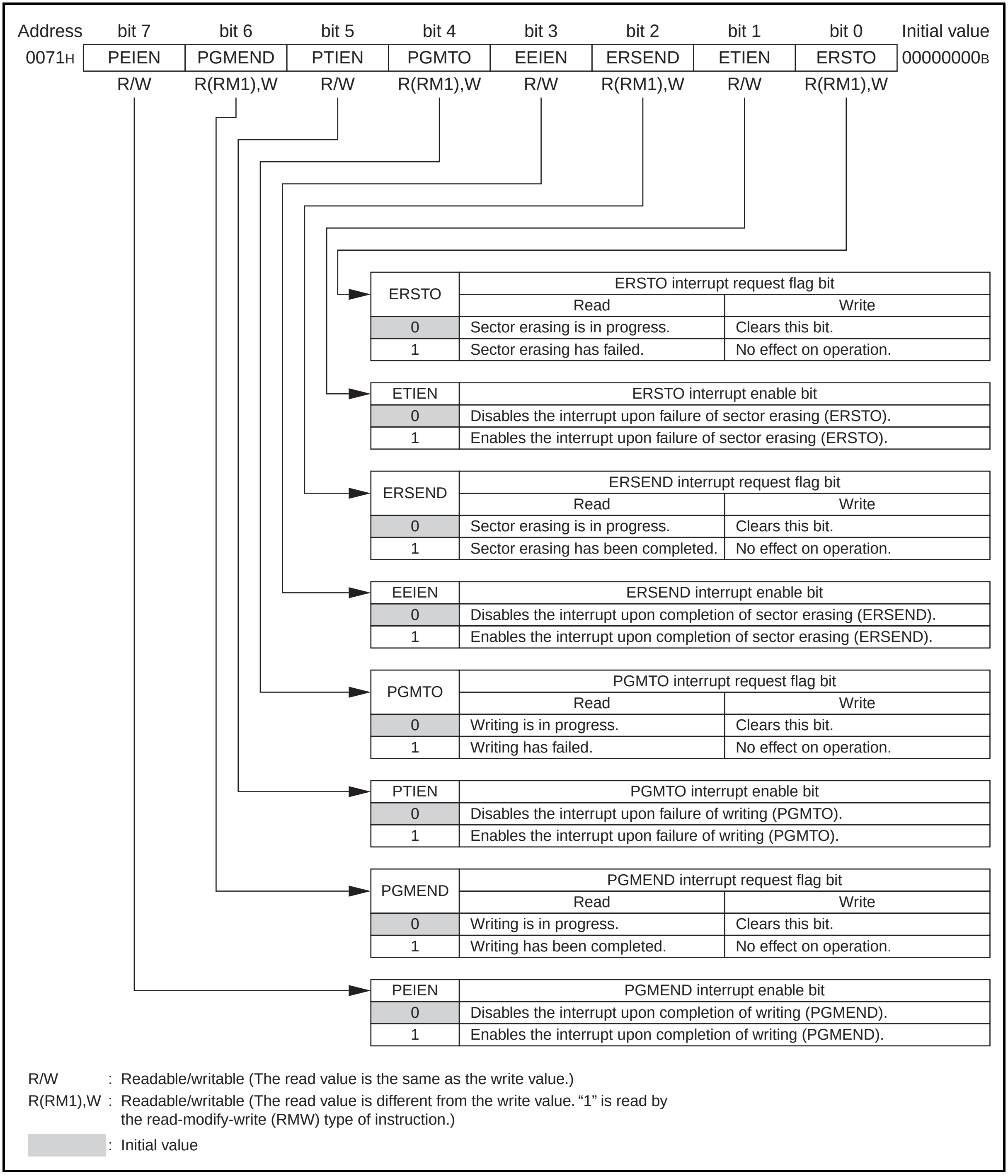
R/W	:	Readable/writable (The read value is the same as the write value.)
R(RM1),W	:	Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R/WX	:	Read only (Readable. Writing a value to this bit has no effect on operation.)
R/W0	:	The write value is "0". The read value is the same as the write value.
R0/WX	:	The read value is "0". Writing a value to this bit has no effect on operation.
-	:	Undefined bit
X	:	Indeterminate

30.3.1 Flash Memory Status Register 2 (FSR2)

Figure 30.3-2 lists the functions of the flash memory status register 2 (FSR2).

■ Flash Memory Status Register 2 (FSR2)

Figure 30.3-2 Flash Memory Status Register 2 (FSR2)



Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
0071H	PEIEN	PGMEND	PTIEN	PGMTO	EEIEN	ERSEND	ETIEN	ERSTO	00000000B
	R/W	R(RM1),W	R/W	R(RM1),W	R/W	R(RM1),W	R/W	R(RM1),W	

ERSTO	ERSTO interrupt request flag bit	
	Read	Write
0	Sector erasing is in progress.	Clears this bit.
1	Sector erasing has failed.	No effect on operation.

ETIEN	ERSTO interrupt enable bit
0	Disables the interrupt upon failure of sector erasing (ERSTO).
1	Enables the interrupt upon failure of sector erasing (ERSTO).

ERSEND	ERSEND interrupt request flag bit	
	Read	Write
0	Sector erasing is in progress.	Clears this bit.
1	Sector erasing has been completed.	No effect on operation.

EEIEN	ERSEND interrupt enable bit
0	Disables the interrupt upon completion of sector erasing (ERSEND).
1	Enables the interrupt upon completion of sector erasing (ERSEND).

PGMTO	PGMTO interrupt request flag bit	
	Read	Write
0	Writing is in progress.	Clears this bit.
1	Writing has failed.	No effect on operation.

PTIEN	PGMTO interrupt enable bit
0	Disables the interrupt upon failure of writing (PGMTO).
1	Enables the interrupt upon failure of writing (PGMTO).

PGMEND	PGMEND interrupt request flag bit	
	Read	Write
0	Writing is in progress.	Clears this bit.
1	Writing has been completed.	No effect on operation.

PEIEN	PGMEND interrupt enable bit
0	Disables the interrupt upon completion of writing (PGMEND).
1	Enables the interrupt upon completion of writing (PGMEND).

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
▨ : Initial value

Table 30.3-1 Functions of Bits in Flash Memory Status Register 2 (FSR2) (1 / 2)

Bit name		Function
bit7	PEIEN: PGMEND interrupt enable bit	This bit enables or disables the generation of interrupt requests triggered by the completion of Flash memory writing. **Writing "0":** Clears the bit.prevents an interrupt request from occurring even when Flash memory writing is completed (FSR2:PGMEND = 1). **Writing "1":** Causes an interrupt request to occur when Flash memory writing is completed (FSR2:PGMEND = 1).
bit6	PGMEND: PGMEND interrupt request flag bit	This bit indicates the completion of Flash memory writing. The PGMEND bit is set to "1" upon completion of the Flash memory automatic algorithm when Flash memory writing is completed. • An interrupt request occurs when the PGMEND bit is set to "1", provided that generating an interrupt request upon completion of Flash memory writing has been enabled (FSR2:PEIEN = 1). • When the PGMEND bit is set to "0" after Flash memory writing is completed, further Flash memory writing is disabled. • When Flash memory writing fails (FSR3:HANG = 1), this bit is cleared to "0". **Writing "0":** Clears the bit. **Writing "1":** Has no effect on the operation. When read by the read-modify-write (RMW) instruction, this bit always returns "1".
bit5	PTIEN: PGMTO interrupt enable bit	This bit enables or disables the generation of interrupt requests triggered by the failure of Flash memory writing. **Writing "0":** Prevents an interrupt requests from occurring even when Flash memory writing fails (FSR2:PGMTO = 1). **Writing "1":** Causes an interrupt requests to occur when Flash memory writing fails (FSR2:PGMTO = 1).
bit4	PGMTO: PGMTO interrupt request flag bit	This bit indicates Flash memory writing has failed. The PGMTO bit is set to "1" upon failure of the Flash memory automatic algorithm when Flash memory writing fails. • An interrupt request occurs when the PGMTO bit is set to "1", provided that generating an interrupt request upon failure of Flash memory writing has been enabled (FSR2:PEIEN = 1). • When the PGMTO bit is set to "1" after Flash memory writing is completed, Flash memory writing is disabled. **Writing "0":** Clears the bit. **Writing "1":** Has no effect on the operation. When read by the read-modify-write (RMW) instruction, this bit always returns "1".
bit3	EEIEN: ERSEND interrupt enable bit	This bit enables or disables the generation of interrupt requests triggered by the completion of Flash memory sector erasing. **Writing "0":** Prevents an interrupt request from occurring even when Flash memory sector erasing is completed (FSR2:ERSEND=1). **Writing "0":** Causes an interrupt request to occur when Flash memory writing is completed (FSR2:ERSEND=1).
bit2	ERSEND: ERSEND interrupt request flag bit	This bit indicates the completion of Flash memory sector erasing. The ERSEND bit is set to "1" upon completion of the Flash memory automatic algorithm when Flash memory writing is completed. • An interrupt request occurs when the ERSEND bit is set to "1", provided that generating an interrupt request upon completion of Flash memory sector erasing has been enabled (FSR2:EEIEN = 1). • When the ERSEND bit is set to "0" after Flash memory writing is completed, further Flash memory sector erasing is disabled. • When Flash memory sector erasing fails (FSR3:HANG = 1), this bit is cleared to "0". **Writing "0":** Clears the bit. **Writing "1":** Has no effect on the operation. When read by the read-modify-write (RMW) instruction, this bit always returns "1".

Table 30.3-1 Functions of Bits in Flash Memory Status Register 2 (FSR2) (2 / 2)

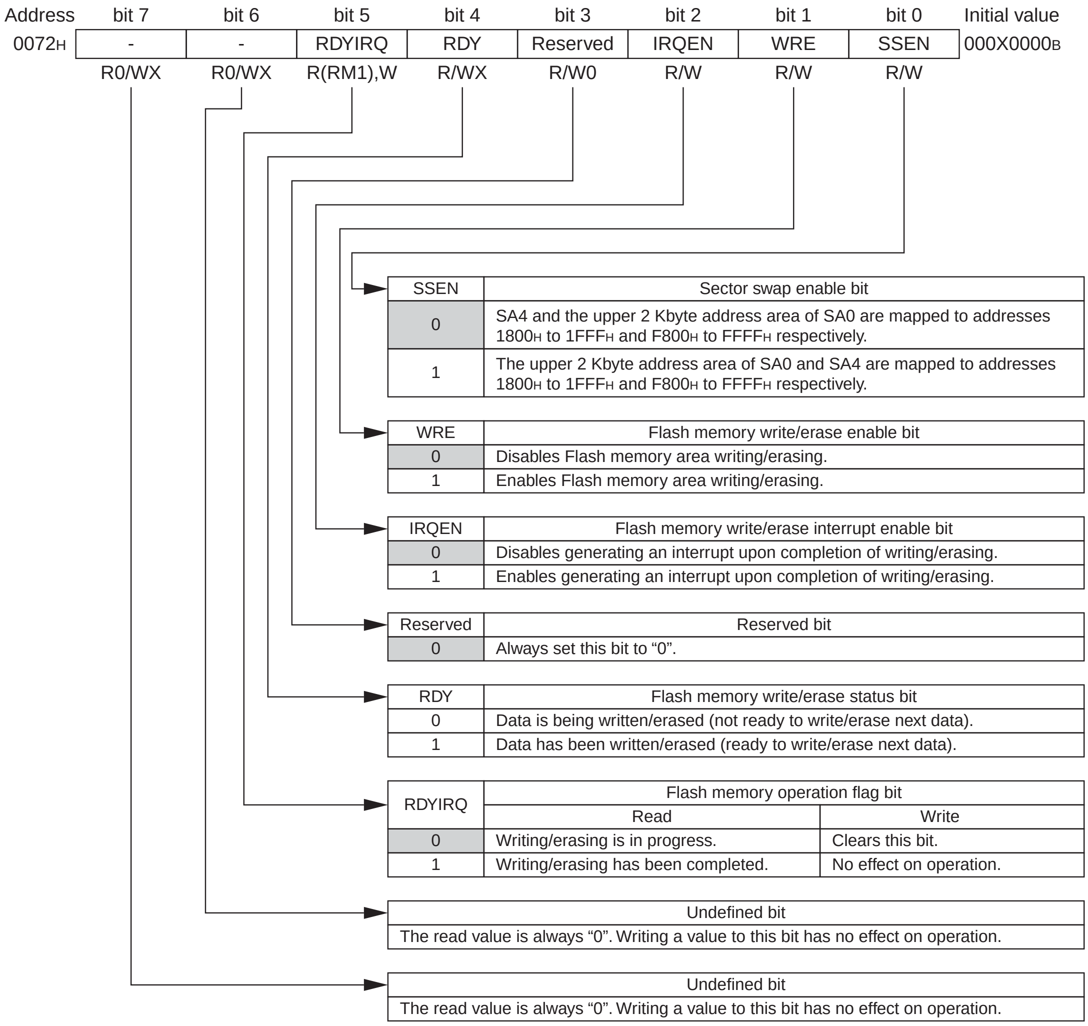
Bit name		Function
bit1	ETIEN: ERSTO interrupt enable bit	This bit enables or disables the writing to or erasing data from the Flash memory area. Set the WRE bit before invoking a Flash memory write/erase command. **Writing "0":** Prevents an interrupt request from occurring even when Flash memory sector erasing fails (FSR2:ERSTO = 1). **Writing "1":** Causes an interrupt requests to occur when Flash memory sector erasing fails (FSR2:ERSTO = 1).
bit0	ERSTO: ERSTO interrupt request flag bit	This bit indicates that Flash memory sector erasing has failed. The ERSTO bit is set to "1" upon failure of the Flash memory automatic algorithm when Flash memory sector erasing fails. • An interrupt request occurs when the ERSTO bit is set to "1", provided that generating an interrupt request upon failure of Flash memory sector erasing has been enabled (FSR2:ETIEN = 1). • When the ERSTO bit is set to "1" after Flash memory writing is completed, further Flash memory sector erasing is disabled. **Writing "0":** Clears the bit. **Writing "1":** Has no effect on the operation. When read by the read-modify-write (RMW) instruction, this bit always returns "1".

30.3.2 Flash Memory Status Register (FSR)

Figure 30.3-3 lists the functions of the flash memory status register (FSR).

■ Flash Memory Status Register (FSR)

Figure 30.3-3 Flash Memory Status Register (FSR)



Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
0072$_H$	-	-	RDYIRQ	RDY	Reserved	IRQEN	WRE	SSEN	000X0000$_B$
	R0/WX	R0/WX	R(RM1),W	R/WX	R/W0	R/W	R/W	R/W	

SSEN	Sector swap enable bit
0	SA4 and the upper 2 Kbyte address area of SA0 are mapped to addresses 1800$_H$ to 1FFF$_H$ and F800$_H$ to FFFF$_H$ respectively.
1	The upper 2 Kbyte address area of SA0 and SA4 are mapped to addresses 1800$_H$ to 1FFF$_H$ and F800$_H$ to FFFF$_H$ respectively.

WRE	Flash memory write/erase enable bit
0	Disables Flash memory area writing/erasing.
1	Enables Flash memory area writing/erasing.

IRQEN	Flash memory write/erase interrupt enable bit
0	Disables generating an interrupt upon completion of writing/erasing.
1	Enables generating an interrupt upon completion of writing/erasing.

Reserved	Reserved bit
0	Always set this bit to “0”.

RDY	Flash memory write/erase status bit
0	Data is being written/erased (not ready to write/erase next data).
1	Data has been written/erased (ready to write/erase next data).

RDYIRQ	Flash memory operation flag bit	
	Read	Write
0	Writing/erasing is in progress.	Clears this bit.
1	Writing/erasing has been completed.	No effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

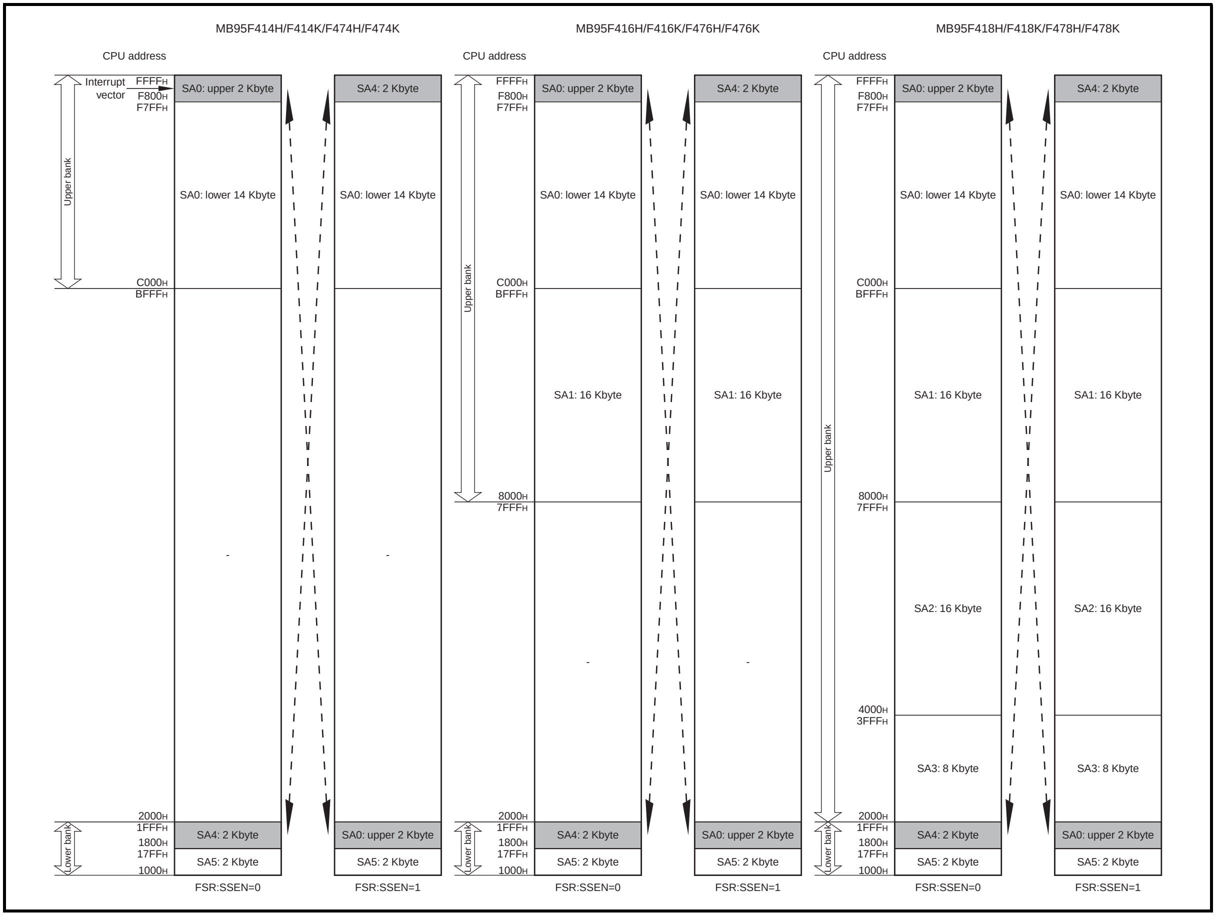
Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)
R/W0 : The write value is “0”. The read value is the same as the write value.
R0/WX : The read value is “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
X : Indeterminate
▨ : Initial value

Table 30.3-2 Functions of Bits in Flash Memory Status Register (FSR)

Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5	RDYIRQ: Flash memory operation flag bit	This bit shows the operating state of the Flash memory. The RDYIRQ bit is set to "1" upon completion of the Flash memory automatic algorithm when Flash memory writing/erasing is completed. • An interrupt request occurs when the RDYIRQ bit is set to "1" if interrupts triggered by the completion of Flash memory writing/erasing have been enabled (FSR:IRQEN = 1). • If the RDYIRQ bit is set to "0" when Flash memory writing/erasing is completed, further Flash memory writing/erasing is disabled. **Writing "0":** Clears the bit. **Writing "1":** Has no effect on the operation. "1" is read from the bit whenever a read-modify-write (RMW) instruction is used.
bit4	RDY: Flash memory program/erase status bit	This bit shows the writing/erasing status of the Flash memory. • Flash memory writing/erasing cannot be performed with the RDY bit set to "0". • A read/reset command can be accepted even when the RDY bit contains "0". The RDY bit is set to "1" upon completion of writing/erasing. • It takes a delay of two machine clock (MCLK) cycles after the issuance of a program/ erase command for the RDY bit to be set to "0". Read this bit after, for example, inserting NOP twice after issuing the program/erase command.
bit3	Reserved bit	Always set this bit to "0".
bit2	IRQEN: Flash memory program/erase interrupt enable bit	This bit enables or disables the generation of interrupt requests in response to the completion of Flash memory writing/erasing. **Writing "0":** Prevents an interrupt request from occurring even when the flash memory operation flag bit is set to "1" (FSR:RDYIRQ = 1). **Writing "1":** Causes an interrupt request to occur when the Flash memory operation flag bit is set to "1" (FSR:RDYIRQ = 1).
bit1	WRE: Flash memory program/erase enable bit	This bit enables or disables the writing/erasing of data into/from the Flash memory area. Set the WRE bit before invoking a Flash memory program/erase command. **Writing "0":** Prevents a program/erase signal from being generated even when a program/ erase command is input. **Writing "1":** Allows Flash memory writing/erasing to be performed after a program/erase command is input. • When no data is to be written to or erased from the Flash memory, set the WRE bit to "0" to prevent it from being accidentally programmed or erased. • To write data to the Flash memory, set FSR:WRE to "1" to write-enable the Flash memory and set the flash memory sector write control register 0 (SWRE0). When FSR:WRE disables writing (contains "0"), write access to the Flash memory does not take place even though it is enabled by the flash memory write control register 0 (SWRE0).
bit0	SSEN: Sector swap enable bit	This bit is used to replace the upper 2 Kbyte address area of sector SA0 in the upper bank, which contains an interrupt vector, with sector SA4 in the lower bank in dual operation mode. **Writing "0":** Maps SA4 at addresses 1800_H to $1FFF_H$ and the upper 2 Kbyte address area of SA0 at SA0 at addresses $F800_H$ to $FFFF_H$. **Writing "1":** Maps the upper 2 Kbyte address area of SA0 at addresses 1800_H to $1FFF_H$ and SA4 at addresses $F800_H$ to $FFFF_H$.

Figure 30.3-4 Access Sector Map by FSR:SSEN Value



30.3.3 Flash Memory Sector Write Control Register 0 (SWRE0)

The flash memory sector write control register 0 (SWRE0) exists in the Flash memory interface to be used to set the Flash memory write-protect feature.

■ Flash Memory Sector Write Control Register 0 (SWRE0)

The flash memory sector write control register 0 (SWRE0) has bits for enabling/disabling writing data into individual sectors (SA5 to SA0). The initial value of each bit is "0", meaning writing data is disabled. Writing "1" to an SAxE bit in SWRE0 enables writing data into the sector corresponding to that bit. Writing "0" to an SAxE bit in SWRE0 prevents data from being accidentally written into the sector corresponding to that bit. When "0" is written to a bit in SWRE0, even though "1" is written to that bit afterward, data cannot be written into the sector corresponding to that bit. To re-write the data, execute a reset operation.

Figure 30.3-5 Flash Memory Sector Write Control Register 0 (SWRE0)

SWRE0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address	Reserved	Reserved	SA5E	SA4E	SA3E	SA2E	SA1E	SA0E	00000000_B
0073_H	R/W0	R/W0	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable/writable (The read value is the same as the write value)
R/W0 : The write value is "0". The read value is the same as the write value.

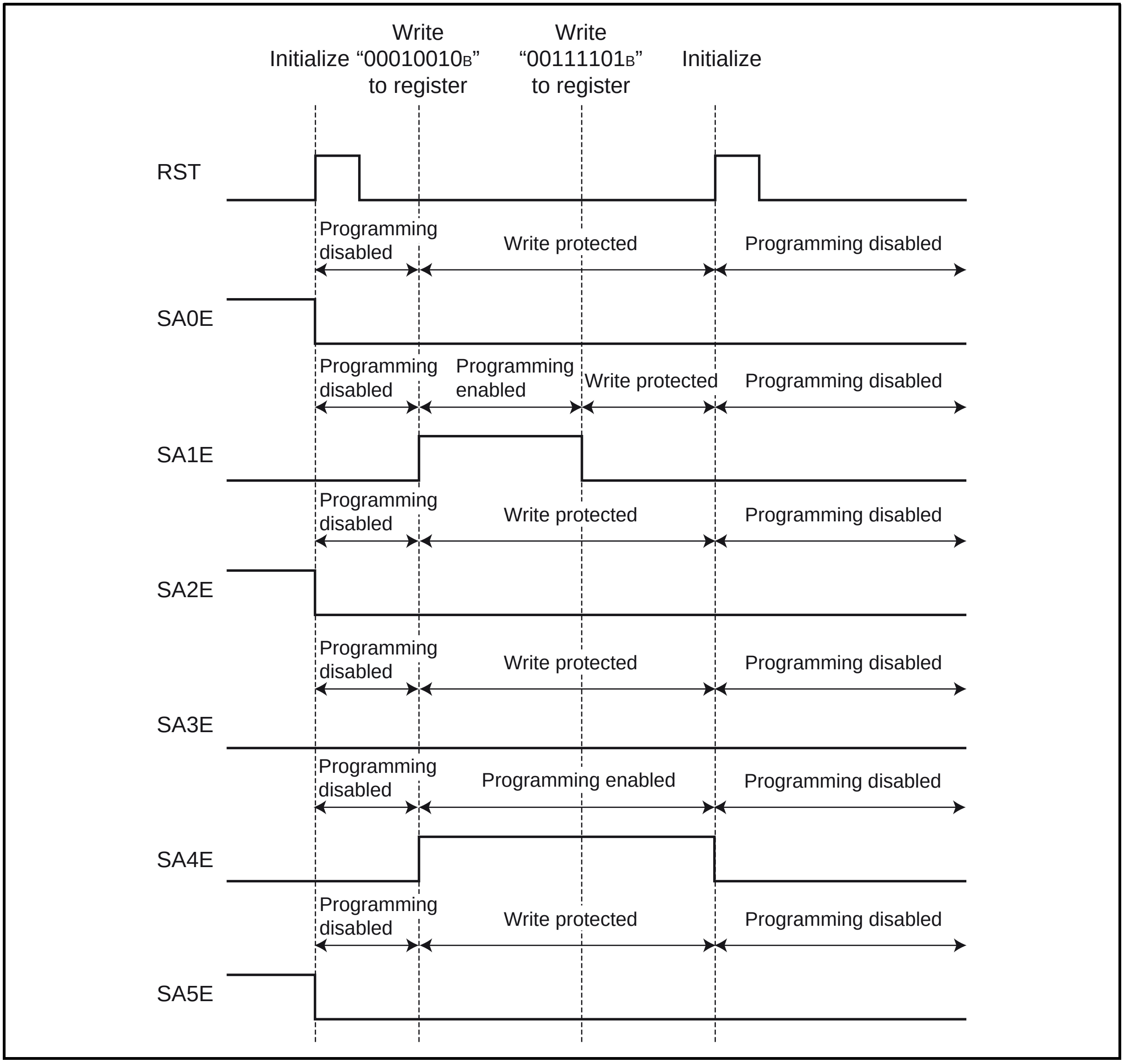
Only write data to SWRE0 by the byte. Setting the bits in SWRE0 using a bit manipulation instruction is prohibited.

Table 30.3-3 Functions of Bits in Flash Memory Sector Write Control Register 0 (SWRE0)

<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit7,
bit6</td><td>Reserved bits</td><td>Always set these bits to "0".</td></tr>
<tr><td>bit5
to
bit0</td><td>SA5E to SA0E:
Writing function setup bits</td><td>
These bits are used to set the function of preventing data from being accidentally written into a sector of the Flash memory. Writing "1" to a bit in SWRE0 enables writing data into the sector corresponding to that bit. Writing "0" to a bit in SWRE0 prevents data from being accidentally written into the sector corresponding to that bit. In addition, resetting this bit initializes it to "0" (writing disabled).

Table of writing function setup bits and their corresponding Flash memory sectors
<table>
<tr><th>Bit Name</th><th>Corresponding Sector in Flash Memory</th></tr>
<tr><td>SA5E</td><td>SA5</td></tr>
<tr><td>SA4E</td><td>SA4</td></tr>
<tr><td>SA3E</td><td>SA3</td></tr>
<tr><td>SA2E</td><td>SA2</td></tr>
<tr><td>SA1E</td><td>SA1</td></tr>
<tr><td>SA0E</td><td>SA0</td></tr>
</table>
<b>Writing disabled</b> : SAxE is "0". With no "0" written to the SAxE bit in the flash memory sector write control register 0 (SWRE0), writing data into a sector can be enabled by setting the SAxE bit corresponding to that sector to "1". (This is the state after SAxE is reset).
<b>Writing enabled</b> : SAxE is "1". Data can be written into a sector corresponding to the SAxE bit.
<b>Write protect</b> : SAxE is "0 ". With "0" written to the SAxE bit in the flash memory sector write control register 0 (SWRE0), writing data into a sector cannot be enabled even though the SAxE bit corresponding to that sector is set to "1".
</td></tr>
</table>

Figure 30.3-6 Example of Flash Memory Writing-disabled, Writing-enabled, and Write-protected States Depending on Flash Memory Sector Write Control Register 0 (SWRE0)



Writing disabled:

SAxE is "0". With no "0" written to the SAxE bit in the flash memory sector write control register 0 (SWRE0), writing data into a sector can be enabled by setting the SAxE bit corresponding to that sector to "1". (This is the state after SAxE is reset).

Writing enabled:

SAxE is "1". Data can be written to a sector corresponding to the SAxE bit.

Write protected:

SAxE is "0 ". With "0" written to the SAxE bit in the flash memory sector write control register 0 (SWRE0), writing data to a sector cannot be enabled even though the SAxE bit corresponding to that sector is set to "1".

■ Note on Setting SWRE0 Register

To write data to or erase data from SA5 (1000_H to $17FF_H$) or SA4 (1800_H to $1FFF_H$) of the Flash memory when FSR:SSEN is "0", set both SA5E and SA4E in the SWRE0 register to "1" first.

To write data to or erase data when FSR:SSEN is "1", set SA5E, SA4E and SA0E in the SWRE0 register to "1" first.

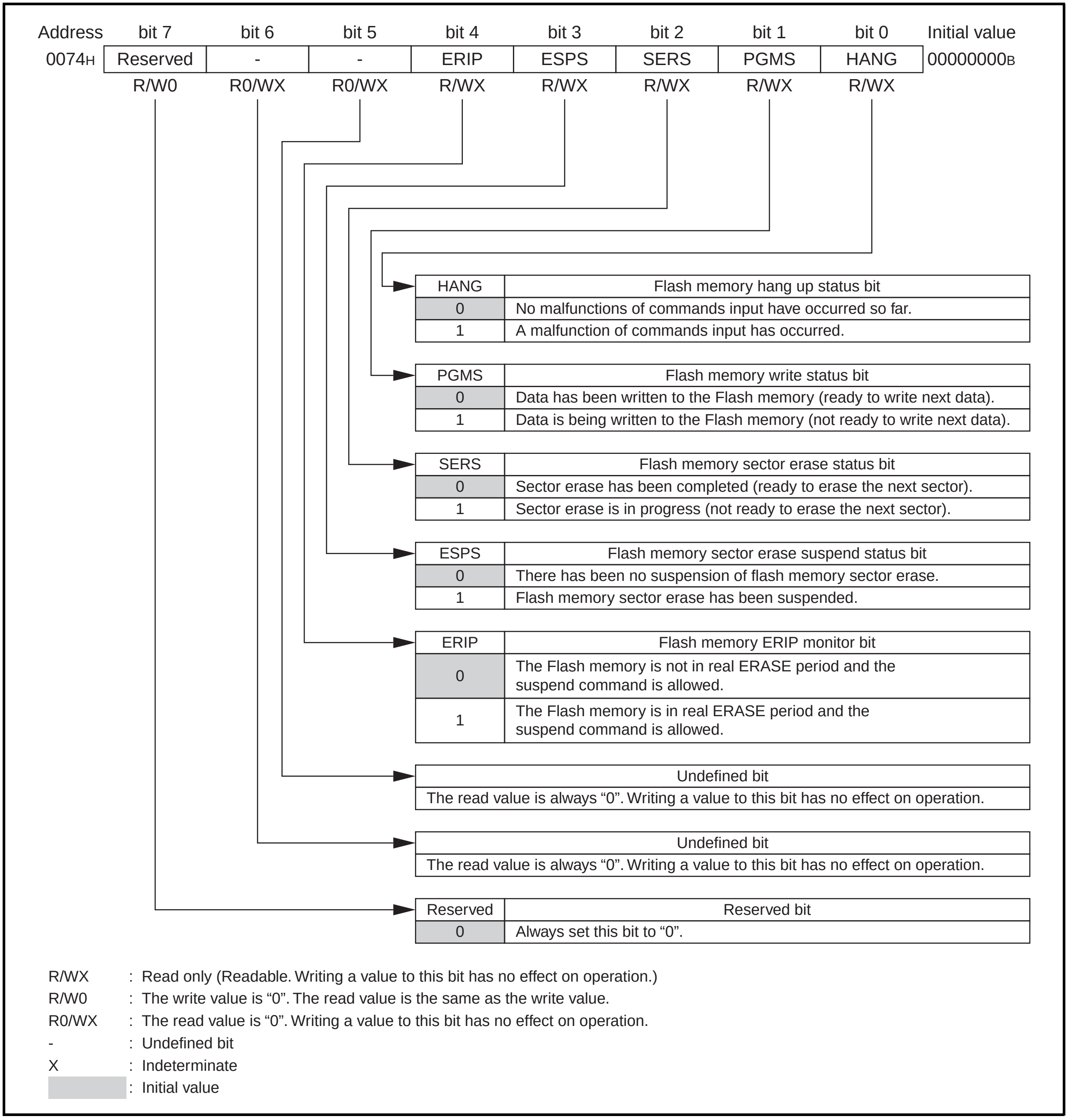
For details of the sector map of the Flash memory, see Figure 30.3-4.

30.3.4 Flash Memory Status Register 3 (FSR3)

Figure 30.3-7 lists the functions of the flash memory status register 3 (FSR3).

■ Flash Memory Status Register 3 (FSR3)

Figure 30.3-7 Flash Memory Status Register 3 (FSR3)



Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
0074H	Reserved	-	-	ERIP	ESPS	SERS	PGMS	HANG	00000000B
	R/W0	R0/WX	R0/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

HANG	Flash memory hang up status bit
0	No malfunctions of commands input have occurred so far.
1	A malfunction of commands input has occurred.

PGMS	Flash memory write status bit
0	Data has been written to the Flash memory (ready to write next data).
1	Data is being written to the Flash memory (not ready to write next data).

SERS	Flash memory sector erase status bit
0	Sector erase has been completed (ready to erase the next sector).
1	Sector erase is in progress (not ready to erase the next sector).

ESPS	Flash memory sector erase suspend status bit
0	There has been no suspension of flash memory sector erase.
1	Flash memory sector erase has been suspended.

ERIP	Flash memory ERIP monitor bit
0	The Flash memory is not in real ERASE period and the suspend command is allowed.
1	The Flash memory is in real ERASE period and the suspend command is allowed.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Reserved	Reserved bit
0	Always set this bit to “0”.

R/WX	:	Read only (Readable. Writing a value to this bit has no effect on operation.)
R/W0	:	The write value is “0”. The read value is the same as the write value.
R0/WX	:	The read value is “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
X	:	Indeterminate
▨	:	Initial value

Table 30.3-4 Functions of Bits in Flash Memory Status Register 3 (FSR3)

Bit name		Function
bit7	Reserved bit	Always set this bit to "0".
bit6, bit5	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit4	ERIP: Flash memory ERIP monitor bit	This bit monitors ERIP signal from the Flash memory. • When the ERIP bit is set to "1", that indicates the Flash memory is on read ERASE period and suspend command is not allowed. • When the ERIP bit is set to "0", that indicates the Flash memory is not on read ERASE period and suspend command is allowed. • There is a delay of two machine clock (MCLK) cycles between the issue of a sector erase suspend command and the ERIP bit being set to "1". After issuing a sector erase suspend command, wait for those two machine clock cycles to elapse (e.g. inserting NOP twice) before reading this bit.
bit3	ESPS: Flash memory sector erase suspend status bit	This bit shows the sector erase suspend status of the Flash memory. • When the ESPS bit is set to "1", that indicates Flash memory sector erase has been suspended. • When the ESPS bit is set to "0", that indicates there has been no suspension of Flash memory sector erase. • There is a delay of two machine clock (MCLK) cycles between the issue of a sector erase suspend command and the ESPS bit being set to "1". After issuing a sector erase suspend command, wait for those two machine clock cycles to elapse (e.g. inserting NOP twice) before reading this bit.
bit2	SERS: Flash memory sector erase status bit	This bit shows the sector erase status of the Flash memory. • When the SERS bit is set to "1", that indicates Flash memory sector erase is in progress. • When the SERS bit is set to "0", that indicates Flash memory sector erase has been completed. • There is a delay of two machine clock (MCLK) cycles between the issue of a sector erase command and the SERS bit being set to "1". After issuing a sector erase suspend command, wait for those two machine clock cycles to elapse (e.g. inserting NOP twice) before reading this bit.
bit1	PGMS: Flash memory write status bit	This bit shows the writing status of the Flash memory. • When the PGMS bit is set to "1", that indicates data is being written to the Flash memory. • When the PGMS bit is set to "0", that indicates data has been written to the Flash memory. • There is a delay of two machine clock (MCLK) cycles between the issue of a write command and the PGMS bit being set to "1". After issuing a sector erase suspend command, wait for those two machine clock cycles to elapse (e.g. inserting NOP twice) before reading this bit. • The PGMS bit will never be asserted under the condition that the machine clock (MCLK) cycle is longer than 1 µs. Use this bit with the machine clock (MCLK) cycle shorter than 1 µs.
bit0	HANG: Flash memory hang up status bit	This bit shows whether the Flash memory has malfunctioned or not. • When the HANG bit is set to "1", that indicates a malfunction of commands input has occurred. • When the HANG bit is set to "0", that indicates no malfunctions of commands input have occurred so far. • There is a delay of two machine clock (MCLK) cycles between the issue of a reset command and the HANG bit being cleared to "0". After issuing a sector erase suspend command, wait for those two machine clock cycles to elapse (e.g. inserting NOP twice) before reading this bit.

■ Examples of Status of Flash Memory Status Register 2, Flash Memory Status Register 3 and RDY Bit (FSR:bit4)

Figure 30.3-8 FSR2:PGMEND during Flash Memory Writing

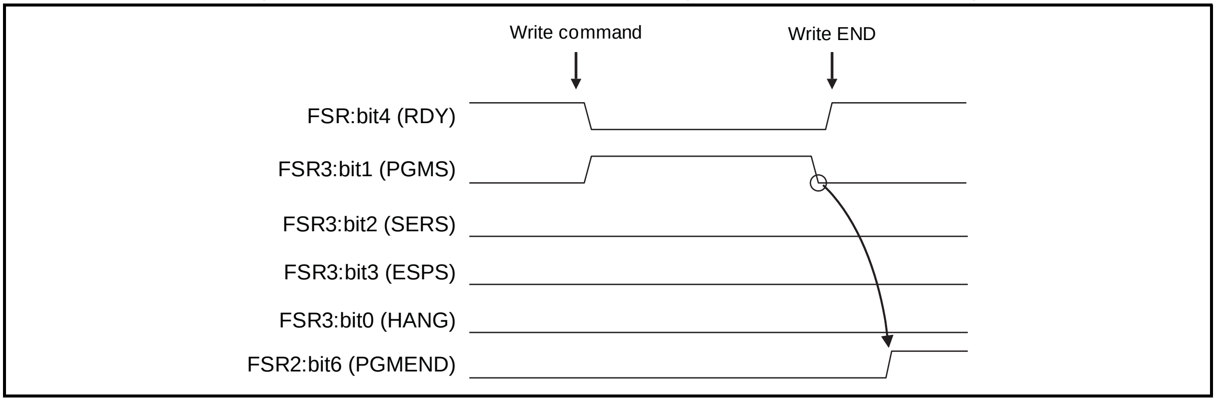


Figure 30.3-9 FSR2:PGMTO when Flash Memory Writing Failed

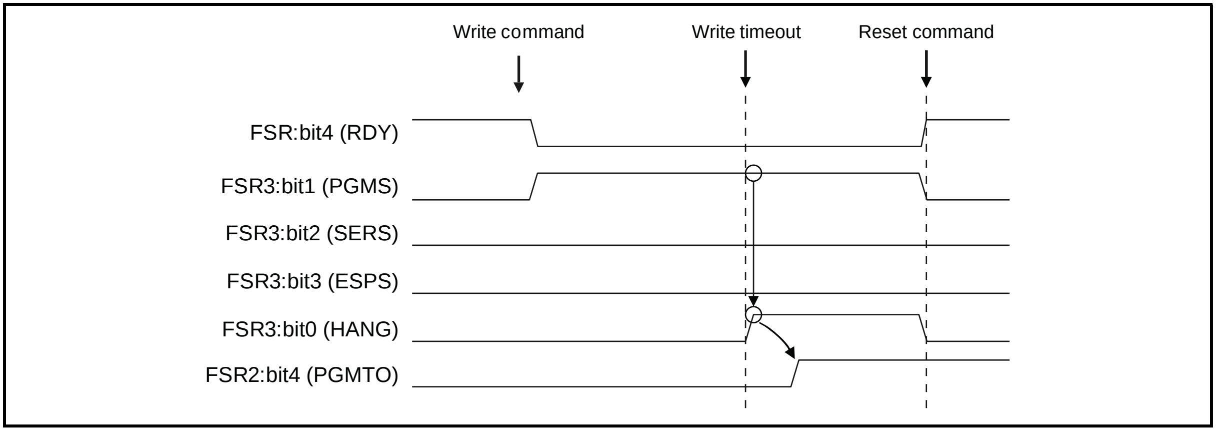


Figure 30.3-10 FSR2:ERSEND during Flash Memory Sector Erase

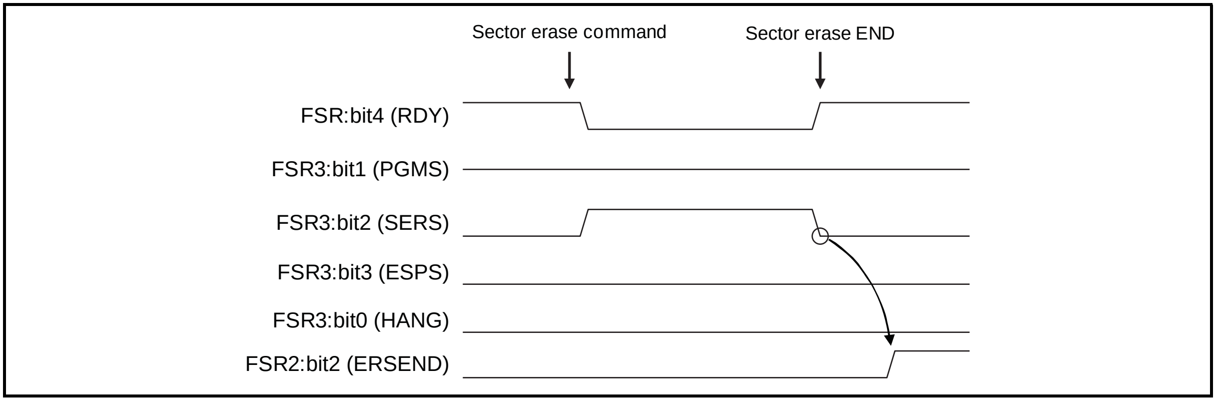


Figure 30.3-11 FSR2:ERSTO when Flash Memory Sector Erase Failed

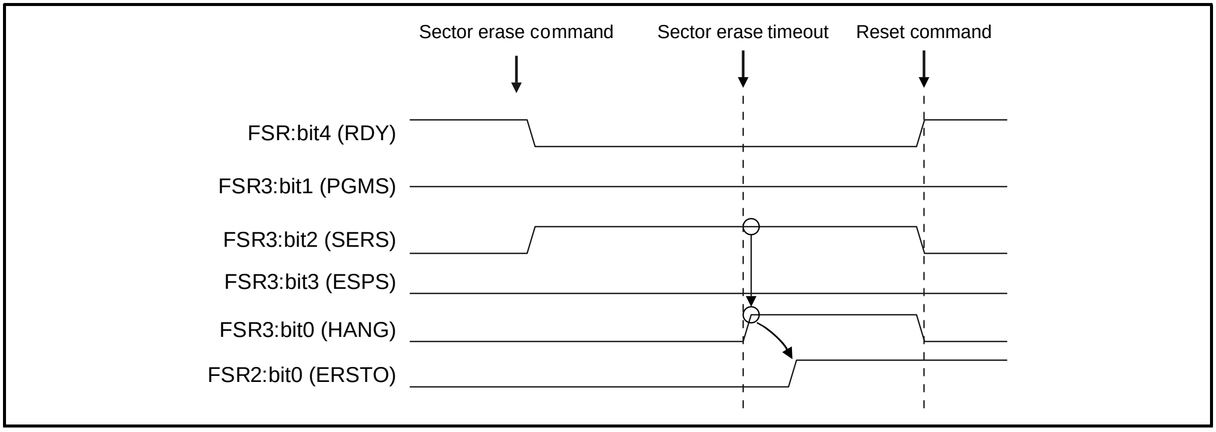


Figure 30.3-12 FSR2:PGMEND and FSR2:ERSEND when Flash Memory Writing Is in Progress with Flash Memory Sector Erase Suspended

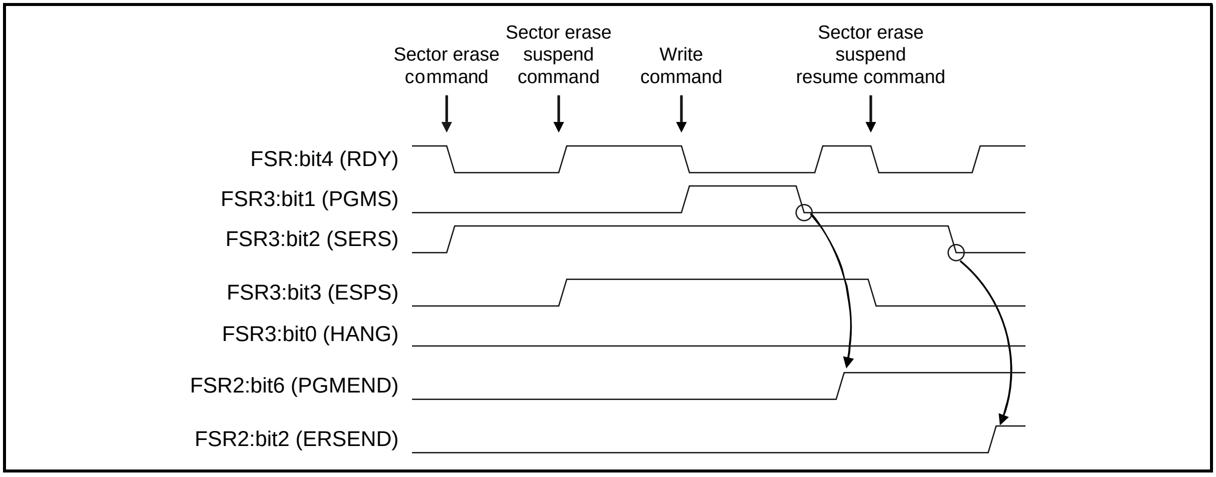


Figure 30.3-13 FSR2:PGMTO and FSR2:ERSEND when Flash Memory Writing Failed with Flash Memory Sector Erase Suspended

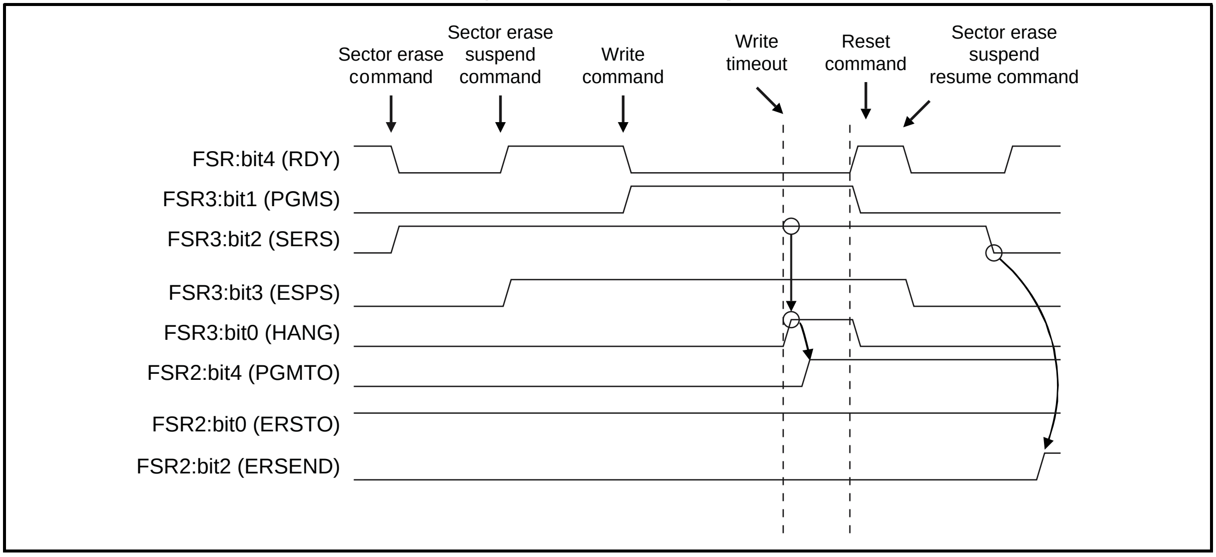


Figure 30.3-14 FSR2:ERSEND during Flash Memory Sector Erase

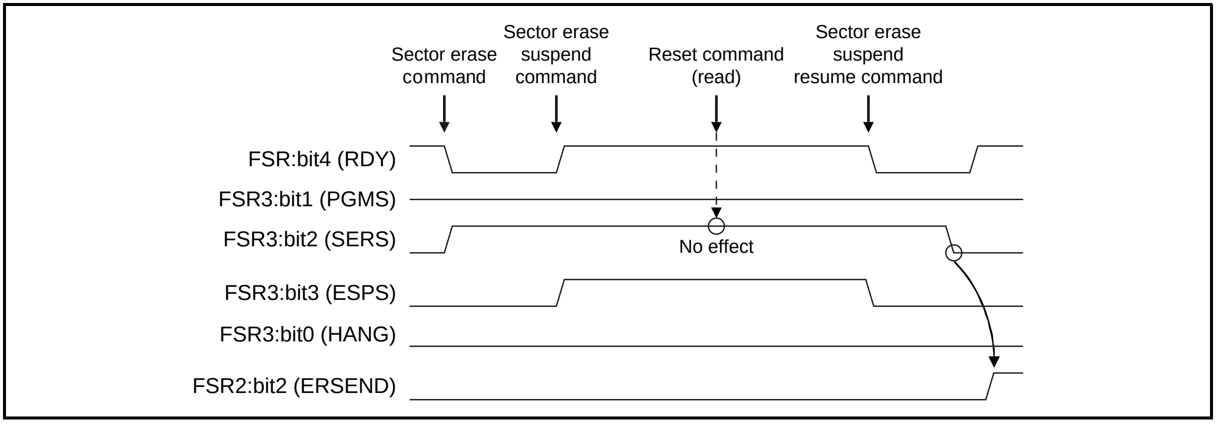
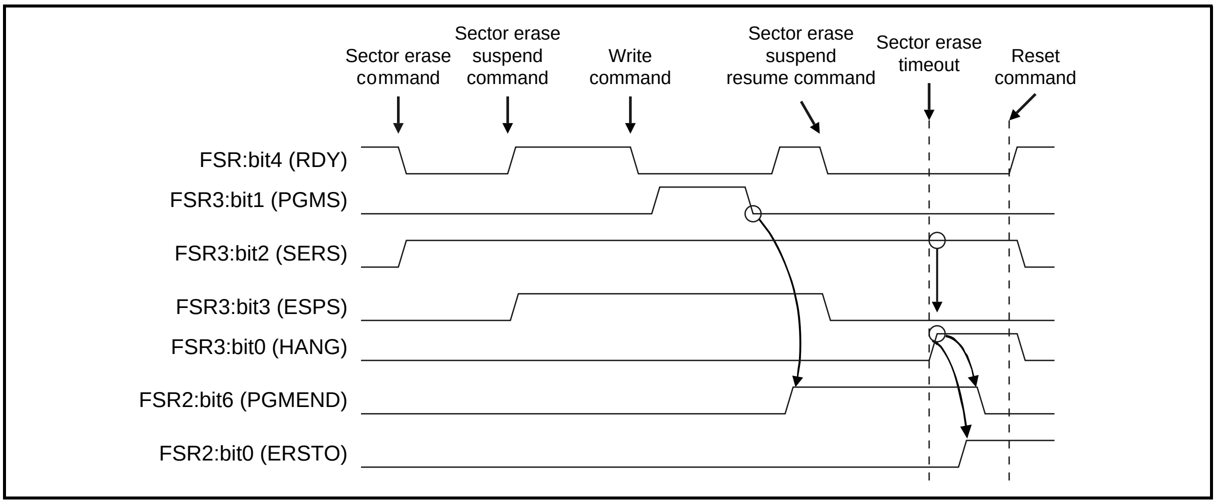


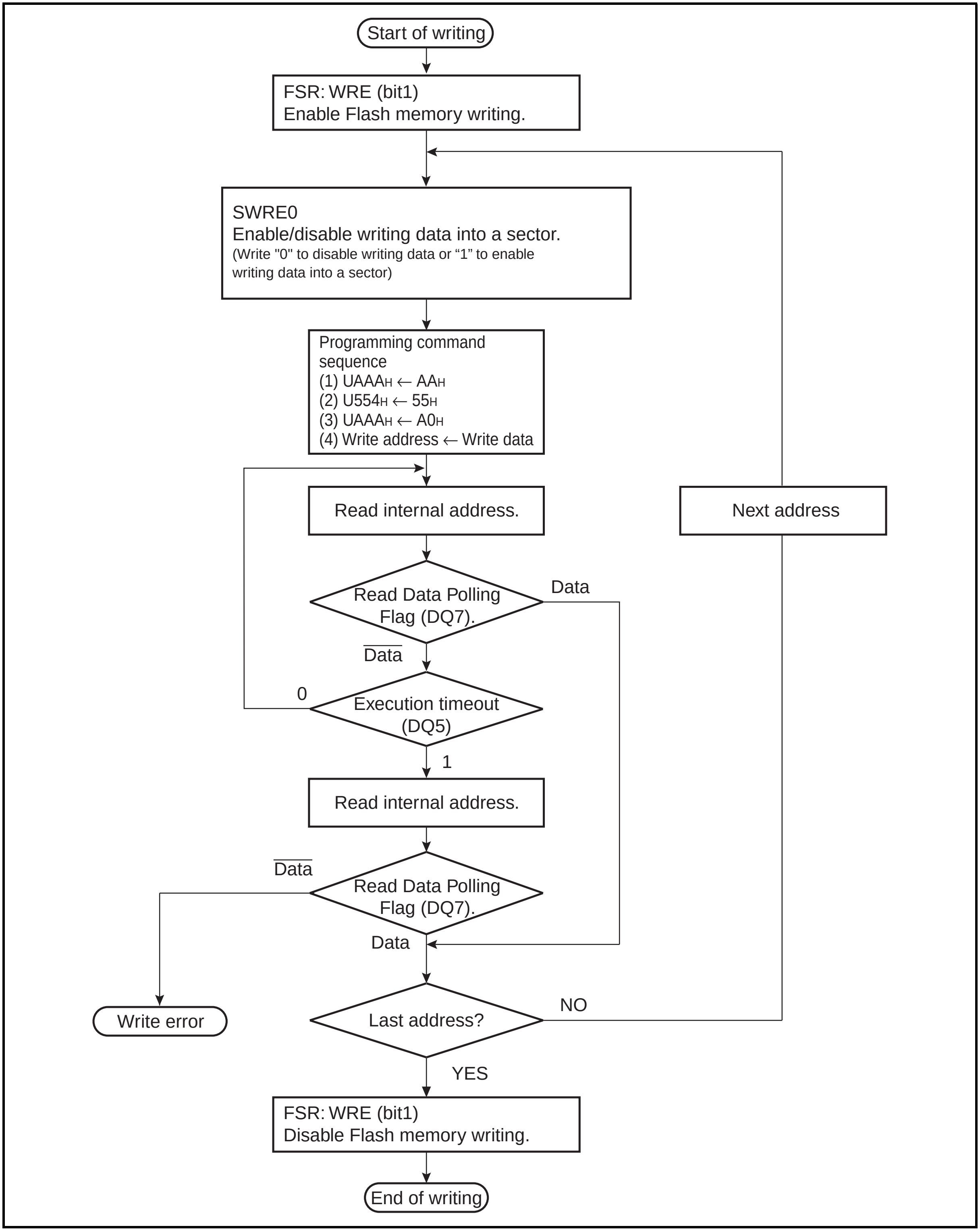
Figure 30.3-15 FSR2:PGMEND and FSR2:ERSTO when Flash Memory Sector Erase Failed after Sector Erase Has Resumed



■ Flash Memory Sector Write Control Register 0 (SWRE0) Setup Flow Chart

Set the FSR:WRE bit and write-enable or write-protect each sector by setting the corresponding bit in the flash memory sector write control register 0 (SWRE0) to "1" or "0", respectively.

Figure 30.3-16 Sample Procedure for Write-protecting/Write-enabling Flash Memory


Start of writing
FSR: WRE (bit1)
Enable Flash memory writing.
SWRE0
Enable/disable writing data into a sector.
(Write "0" to disable writing data or "1" to enable writing data into a sector)
Programming command sequence
(1) UAAAH ← AAH
(2) U554H ← 55H
(3) UAAAH ← A0H
(4) Write address ← Write data
Read internal address.
Next address
Read Data Polling Flag (DQ7).
Data
$\overline{\text{Data}}$
0
Execution timeout (DQ5)
1
Read internal address.
$\overline{\text{Data}}$
Read Data Polling Flag (DQ7).
Data
Write error
Last address?
NO
YES
FSR: WRE (bit1)
Disable Flash memory writing.
End of writing

■ Note on Setting the FSR:WRE Bit

To write data to the Flash memory, set FSR:WRE to "1" to enable Flash memory writing, and then set the bit in the SWRE0 register corresponding to a sector to which data is to be written. When Flash memory writing is disabled by setting FSR:WRE to "0", no write access to a sector in the Flash memory can be executed even though it has been enabled by setting a bit corresponding to that sector in the SWRE0 register to "1".

30.4 Starting the Flash Memory Automatic Algorithm

There are four commands that invoke the Flash memory automatic algorithm: read/reset, write (program), chip-erase, and sector-erase. The sector erase command is capable of suspending and resuming.

■ Command Sequence Table

Table 30.4-1 lists the commands used in writing/erasing the Flash memory.

Table 30.4-1 Command Sequence

Command sequence	Bus write cycle	1st bus write cycle		2nd bus write cycle		3rd bus write cycle		4th bus write cycle		5th bus write cycle		6th bus write cycle	
		Address	Data	Address	Data	Address	Data	Address	Data	Address	Data	Address	Data
Read/reset*	1	F$_X$XX$_H$	F0$_H$	-	-	-	-	-	-	-	-	-	-
	4	UAAA$_H$	AA$_H$	U554$_H$	55$_H$	UAAA$_H$	F0$_H$	RA	RD	-	-	-	-
Write	4	UAAA$_H$	AA$_H$	U554$_H$	55$_H$	UAAA$_H$	A0$_H$	PA	PD	-	-	-	-
Chip erase	6	XAAA$_H$	AA$_H$	X554$_H$	55$_H$	XAAA$_H$	80$_H$	XAAA$_H$	AA$_H$	X554$_H$	55$_H$	XAAA$_H$	10$_H$
Sector erase	6	UAAA$_H$	AA$_H$	U554$_H$	55$_H$	UAAA$_H$	80$_H$	UAAA$_H$	AA$_H$	U554$_H$	55$_H$	SA	30$_H$
Sector erase suspend			Writing data "B0$_H$" to address "BA" suspends sector erasing.										
Sector erase resume			Writing data "30$_H$" to address "BA" resumes suspended sector erasing.										

RA : Read address
PA : Write (program) address
SA : Sector address (specify arbitrary one address in sector)
RD : Read data
PD : Write (program) data
U : Upper 4 bits same as RA, PA, and SA
BA : Upper 3 bits of an address
F$_X$: FF/FE
X : Arbitrary address
*: Both types of read/reset command can reset the Flash memory to read mode.

Notes:

- Addresses in the table are the values in the CPU memory map. All addresses and data are hexadecimal values. However, "X" is an arbitrary value.
- Address "U" in the table is not arbitrary, whose four bits (bit15 to bit12) must have the same value as RA, PA and SA.

  Example: If RA = C48E$_H$, U = C; If PA = 1024$_H$, U = 1;

  If SA = 3000$_H$, U = 3
- The chip erase command is accepted only when all sectors have been program-enabled. The chip erase command is ignored if the bit for any sector in the Flash memory sector write control register 0 (SWRE0) has been set to "0" (to program-disabled or write-protect the sector).

■ Notes on Issuing Commands

Pay attention to the following points when issuing commands in the command sequence table:

- Program-enable each required sector before issuing the first command.
- The upper address U bits (bit15 to bit12) used when commands are issued must have the same value as RA, PA, and SA, from the first command on.

If the above measures are not followed, commands are not recognized normally. When commands are not recognized normally, execute a reset to initialize the command sequencer in the Flash memory.

30.5 Checking Automatic Algorithm Execution Status

Since the Flash memory uses the automatic algorithm to execute the write/erase flow, its internal operating status can be checked through the hardware sequence flags.

■ Hardware Sequence Flag

● Overview of hardware sequence flag

The hardware sequence flag consists of the following 4-bit outputs:

- Data polling flag (DQ7)
- Toggle bit flag (DQ6)
- Execution timeout flag (DQ5)
- Sector erase timer flag (DQ3)

The hardware sequence flags can tell whether a write command, a chip-erase command or a sector-erase command has been terminated and whether an erase code can be written.

The value of a hardware sequence flag can be checked by a read access to the address of a target sector in the Flash memory after a command sequence is set. Note that a hardware sequence flag is output only to the bank from which a command has been issued.
Table 30.5-1 shows the bit allocation of the hardware sequence flags.

Table 30.5-1 Bit Allocation of Hardware Sequence Flags

Bit no.	7	6	5	4	3	2	1	0
Hardware sequence flag	DQ7	DQ6	DQ5	-	DQ3	-	-	-

- To decide whether an automatic write command, a chip-erase command or a sector-erase command is being executed or has been terminated, check the respective hardware sequence flags or the flash memory write/erase status bit in the flash memory status register (FSR:RDY). After writing/erasing is terminated, the Flash memory returns to the read/reset state.
- When creating a write/erase program, read data after confirming the termination of automatic writing/erasing using the DQ3, DQ5, DQ6 and DQ7 flags.
- The hardware sequence flags can also be used to check whether the second sector erase code write and those to be executed afterward are valid or not.

- Explanation of hardware sequence flag

Table 30.5-2 lists the functions of the hardware sequence flag.

Table 30.5-2 List of Hardware Sequence Flag Functions

State		DQ7	DQ6	DQ5	DQ3
State transition during normal operation	Writing → Writing completed (when write address has been specified)	$\overline{\text{DQ7}}$ → DATA: 7	Toggle → DATA: 6	0 → DATA: 5	0 → DATA: 3
	Chip/sector erasing → Erasing completed	0 → 1	Toggle → Stop	0 → 1	1
	Sector erasing wait → Erasing started	0	Toggle	0	0 → 1
	Erasing → Sector erasing suspended (Sector being erased)	0 → 1	Toggle → Stop	0	1 → 0
	Sector erasing suspended → Erasing resumed (Sector being erased)	1 → 0	Stop → Toggle	0	0 → 1
	Sector erasing being suspended (Sector not being erased)	DATA: 7	DATA: 6	DATA: 5	DATA: 3
Abnormal operation	Writing	$\overline{\text{DQ7}}$	Toggle	1	0
	Chip/sector erasing	0	Toggle	1	1

30.5.1 Data Polling Flag (DQ7)

The data polling flag (DQ7) is a hardware sequence flag used to indicate that the automatic algorithm is being executing or has been completed using the data polling function.

■ Data Polling Flag (DQ7)

Table 30.5-3 and Table 30.5-4 show the state transition of the data polling flag.

Table 30.5-3 State Transition of Data Polling Flag (During Normal Operation)

Operating state	Writing → Writing completed	Chip/sector erasing → Erasing completed	Sector erasing wait → Erasing started	Sector erasing → Sector erasing suspended (Sector being erased)	Sector erasing suspended → Erasing resumed (Sector being erased)	Sector erasing being suspended (Sector not being erased)
DQ7	$\overline{\text{DQ7}}$ → DATA: 7	0 → 1	0	0 → 1	1 → 0	DATA: 7

Table 30.5-4 State Transition of Data Polling Flag (During Abnormal Operation)

Operating state	Writing	Chip/sector erasing
DQ7	$\overline{\text{DQ7}}$	0

● At writing

When read access takes place during execution of the automatic write algorithm, the Flash memory outputs the inverted value of bit7 in the last data written to DQ7.

If read access takes place on completion of the automatic write algorithm, the Flash memory outputs bit7 of the value read from the read-accessed address to DQ7.

● At chip/sector erasing

When read access is made to the sector currently being erased during execution of the chip/ sector erase algorithm, bit7 of the Flash memory outputs "0". Bit7 of the Flash memory outputs "1" upon completion of chip/sector erasing.

● At sector erasing suspension

- When read access takes place with a sector-erase operation suspended, the Flash memory outputs "0" to DQ7 if the read address is the sector being erased. If not, the Flash memory outputs bit 7 (DATA:7) of the value read from the read address to DQ7.
- Referring the data polling flag (DQ7) together with the toggle bit flag (DQ6) permits a decision on whether the Flash memory is in the sector erase suspended state or which sector is being erased.

Note:

Once the automatic algorithm has been started, read access to the specified address is ignored. Data reading is allowed after the data polling flag (DQ7) is set to "1". Data reading after the end of the automatic algorithm should be performed following read access made to confirm the completion of data polling.

30.5.2 Toggle Bit Flag (DQ6)

The toggle bit flag (DQ6) is a hardware sequence flag used to indicate that the automatic algorithm is being executed or has been completed using the toggle bit function.

■ Toggle Bit Flag (DQ6)

Table 30.5-5 and Table 30.5-6 show the state transition of the toggle bit flag.

Table 30.5-5 State Transition of Toggle Bit Flag (During Normal Operation)

Operating state	Writing → Writing completed	Chip/sector erasing → Erasing completed	Sector erasing wait → Erasing started	Sector erasing → Sector erasing suspended (Sector being erased)	Sector erasing suspended → Erasing resumed (Sector being erased)	Sector erasing being suspended (Sector not being erased)
DQ6	Toggle → DATA: 6	Toggle → Stop	Toggle	Toggle → Stop	Stop → Toggle	DATA: 6

Table 30.5-6 State Transition of Toggle Bit Flag (During Abnormal Operation)

Operating state	Writing	Chip/sector erasing
DQ6	Toggle	Toggle

● At writing and chip/sector erasing

- When read access is made continuously during execution of the automatic write algorithm or chip-erase/sector-erase algorithm, the Flash memory toggles the output between "1" and "0" at each read access.
- When read access is made continuously after the automatic write algorithm or chip-erase/sector-erase algorithm is terminated, the Flash memory outputs bit6 (DATA:6) of the value read from the read address at each read access.

● At sector erasing suspension

When read access takes place with a sector-erase operation suspended, the Flash memory outputs "1" if the read address is the sector being erased. If not, the Flash memory outputs bit6 (DATA: 6) of the value read from the read address.

30.5.3 Execution Timeout Flag (DQ5)

The execution timeout flag (DQ5) is a hardware sequence flag indicating that the execution time of the automatic algorithm exceeds a specified time (required for writing/erasing) in the Flash memory

■ Execution Timeout Flag (DQ5)

Table 30.5-7 and Table 30.5-8 show the state transition of the execution timeout flag.

Table 30.5-7 State Transition of Execution Timeout Flag (During Normal Operation)

Operating state	Writing → Writing completed	Chip/sector erasing → Erasing completed	Sector erasing wait → Erasing started	Sector erasing → Sector erasing suspended (Sector being erased)	Sector erasing suspended → Erasing resumed (Sector being erased)	Sector erasing being suspended (Sector not being erased)
DQ5	0 → DATA: 5	0 → 1	0	0	0	DATA: 5

Table 30.5-8 State Transition of Execution Timeout Flag (During Normal Operation)

Operating state	Writing	Chip/sector erasing
DQ5	1	1

- At writing and chip erasing

When read access is made with the write or chip-erase/sector-erase automatic algorithm invoked, the flag outputs "0" when the algorithm execution time is within the specified time (required for writing/erasing) or "1" when it exceeds that time.

The execution timeout flag (DQ5) can be used to check whether writing/erasing has succeeded or failed regardless of whether the automatic algorithm has been running or terminated. When the execution timeout flag (DQ5) outputs "1", it indicates that writing has failed if the automatic algorithm is still running for the data polling or toggle bit function.

If an attempt is made to write "1" to a Flash memory address holding "0", for example, the Flash memory is locked, preventing the automatic algorithm from being terminated and valid data from being output from the data polling flag (DQ7). As the toggle bit flag (DQ6) does not stop toggling, the time limit is exceeded and the execution timeout flag (DQ5) outputs "1". The state in which the execution timeout flag (DQ5) outputs "1" means that the Flash memory has not been used correctly; but not that the Flash memory is defective. When this state occurs, execute the reset command.

30.5.4 Sector Erase Timer Flag (DQ3)

The sector erase timer flag (DQ3) is a hardware sequence flag used to indicate whether the Flash memory is waiting for sector erasing after the sector erase command has started.

■ Sector Erase Timer Flag (DQ3)

Table 30.5-9 and Table 30.5-10 show the state transition of the sector erase timer flag.

Table 30.5-9 State Transition of Sector Erase Timer Flag (During Normal Operation)

Operating state	Writing → Writing completed	Chip/sector erasing → Erasing completed	Sector erasing wait → Erasing started	Sector erasing → Sector erasing suspended (Sector being erased)	Sector erasing suspended → Erasing resumed (Sector being erased)	Sector erasing being suspended (Sector not being erased)
DQ3	0 → DATA: 3	1	0 → 1	1 → 0	0 → 1	DATA: 3

Table 30.5-10 State Transition of Sector Erase Timer Flag (During Abnormal Operation)

Operating state	Writing	Chip/sector erasing
DQ3	0	1

● At sector erasing

- When read access is made after the sector erase command has been started, the sector erase timer flag (DQ3) outputs "0" within the sector erasing wait period. The flag outputs "1" if the sector erase wait period has been exceeded.
- When the sector erase timer flag (DQ3) is "1", sector erasing is being performed if the sector erase algorithm shows running for the data polling or toggle bit function (with DQ7 holding 0 and DQ6 toggling the output). If any command other than the sector erase suspend command is set subsequently, it is ignored until sector erasing is terminated.
- If the sector erase timer flag (DQ3) is "0", the Flash memory can accept the sector erase command. Before writing the sector erase command into the Flash memory, make sure that the sector erase timer flag (DQ3) is "0". If the flag is "1", the Flash memory may not accept the sector erase command suspended.

● At sector erasing suspension

When read access is made with a sector erase operation suspended, the Flash memory outputs "1" if the read address is the sector being erased. If not, the Flash memory outputs bit3 (DATA: 3) of the value read from the read address.

30.6 Writing/Erasing Flash Memory

This section describes the individual procedures for Flash memory reading/resetting, writing, chip-erasing, sector-erasing, sector-erase suspending, and sector-erase resuming by entering their respective commands to invoke the automatic algorithm.

■ Writing/Erasing Flash Memory

The automatic algorithm can be invoked by writing the read/reset, program, chip-erase, sector-erase, sector-erase suspend, and sector-erase resume command sequence to the Flash memory from the CPU. Writing command sequence to the Flash memory from the CPU must always be performed continuously. The termination of the automatic algorithm can be checked by the data polling function. After the automatic algorithm terminates normally, the Flash memory returns to the read/reset state.

The individual operations are explained in the following order:

- Enter read/reset state.
- Write data.
- Erase all data (chip-erase).
- Erase arbitrary data (sector erase).
- Suspend sector erasing.
- Resume sector erasing.

30.6.1 Placing Flash Memory in the Read/Reset State

This section explains the procedure for entering the read/reset command to place the Flash memory in the read/reset state.

■ Placing Flash Memory in the Read/Reset State

- To place the Flash memory in the read/reset state, send the read/reset command in the command sequence table continuously from the CPU to the Flash memory.
- The read/reset command is available in two different command sequences: one involves a single bus operation and the other involves four bus operations, which are essentially the same.
- Since the read/reset state is the initial state of the Flash memory, the Flash memory always enters this state after the power is turned on and at the normal termination of a command. The read/reset state is also described as the wait state for command input.
- In the read/reset state, data in the Flash memory can be read by a read access to the Flash memory.
- Read access to the Flash memory does not require the read/reset command. If a command is not terminated normally, use the read/reset command to initialize the automatic algorithm.

30.6.2 Writing Data to Flash Memory

This section explains the procedure for entering the write (program) command to write data to the Flash memory.

■ Writing Data to Flash Memory

- To start the automatic algorithm for writing data into the Flash memory, send the program command in the command sequence table continuously from the CPU to the Flash memory.
- Upon completion of data writing to a target address in the fourth cycle, the automatic algorithm starts automatic writing.

● How to specify addresses

Writing can be performed even in any order of addresses or across a sector boundary. Data written by a single program command is only one byte.

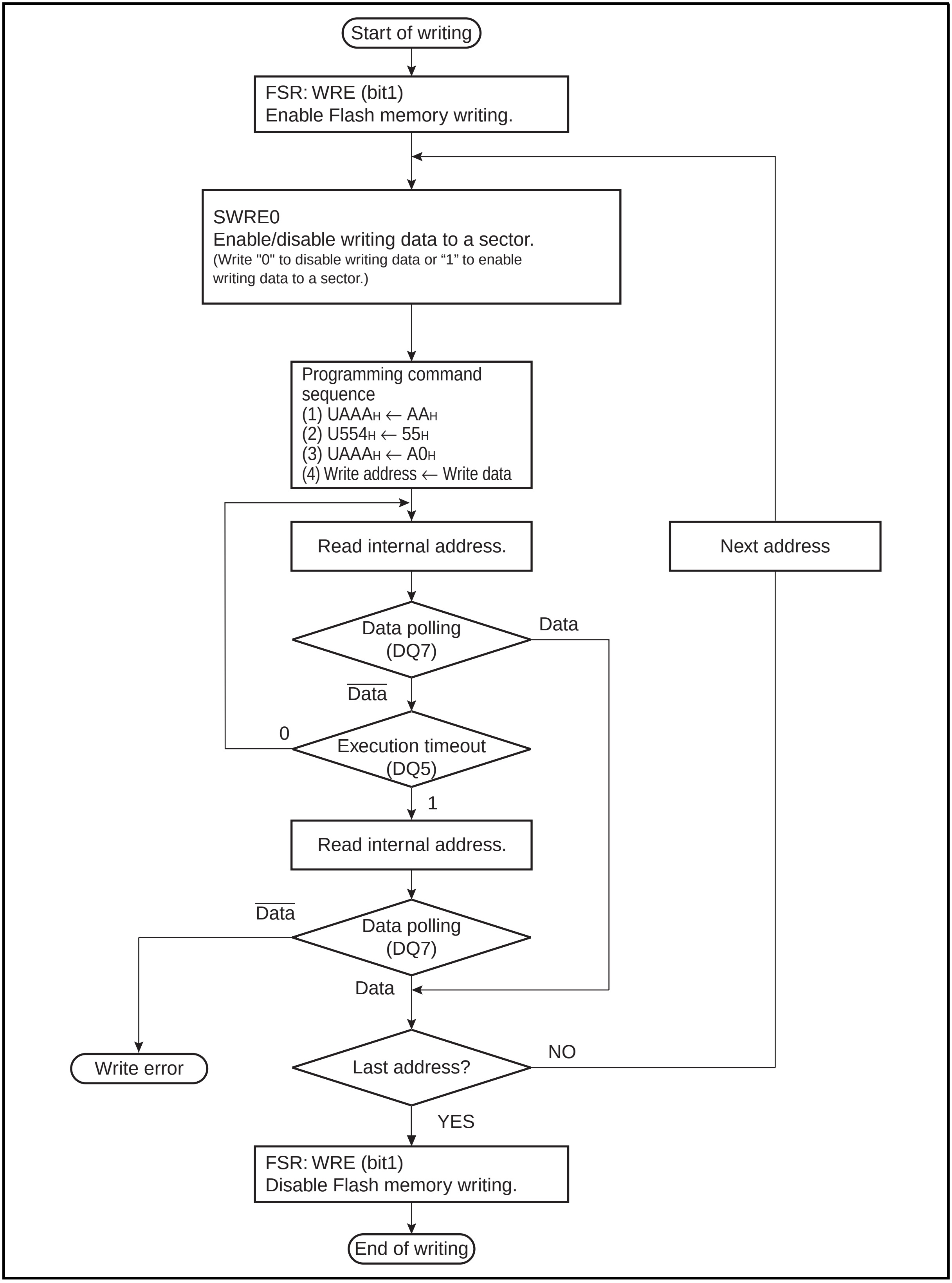
● Notes on writing data

- Bit data cannot be returned from "0" to "1" by writing. When bit data "1" is programmed to bit data "0", the data polling function (DQ7) or toggle operation (DQ6) is not terminated, the Flash memory element is determined to be defective, and the execution timeout flag (DQ5) detects an error to indicate that the specified writing time has been exceeded.
  When data is read in the read/reset state, the bit data remains "0". To return the bit data from "0" to "1", erase the Flash memory.
- All commands are ignored during automatic writing.
- If a hardware reset occurs during writing, the data being programmed to the current address is not guaranteed. Retry from the chip-erase or sector-erase command.

■ Flash Memory Writing Procedure

- Figure 30.6-1 gives an example of the procedure for writing data into the Flash memory. The hardware sequence flags can be used to check the operating state of the automatic algorithm in the Flash memory. The data polling flag (DQ7) is used for checking the completion of writing into the Flash memory in this example.
- Flag check data should be read from the address where data was last written.
- Because the data polling flag (DQ7) and execution timeout flag (DQ5) are updated at the same time, the data polling flag (DQ7) must be checked even when the execution timeout flag (DQ5) is "1".
- Similarly, the toggle bit flag (DQ6) must be checked as it stops toggling at the same time as when the execution timeout flag (DQ5) changes to "1".

Figure 30.6-1 Sample Procedure for Writing into Flash Memory



30.6.3 Erasing All Data from Flash Memory (Chip Erase)

This section describes the procedure for issuing the chip erase command to erase all data from the Flash memory.

■ Erasing Data from Flash Memory (Chip Erase)

- To erase all data from the Flash memory, send the chip erase command in the command sequence table continuously from the CPU to the Flash memory.
- The chip erase command is executed in six bus operations. Chip erasing is started upon completion of the sixth writing cycle.
- Before chip erasing, the user need not perform writing into the Flash memory. During execution of the automatic erase algorithm, the Flash memory automatically programs "0" before erasing all cells automatically.

■ Notes on Chip Erasing

- The chip erase command is accepted only when all sectors have been program-enabled. The chip erase command is ignored if the bit for any sector in the Flash memory sector write control register 0 (SWRE0) has been set to "0" (to program-disable or write-protect the sector).
- If a hardware reset occurs during erasure, the data being erased from the Flash memory is not guaranteed.

30.6.4 Erasing Arbitrary Data from Flash Memory (Sector Erase)

This section explains the procedure for entering the sector erase command to erase any sector in the Flash memory. Sector-by-sector erasing is enabled and multiple sectors can be specified at a time.

■ Erasing Arbitrary Data from Flash Memory (Sector Erase)

To erase data from an arbitrary sector in the Flash memory, send the sector erase command in the command sequence table continuously from the CPU to the Flash memory.

● Specifying a sector

- The sector erase command is executed in six bus operations. A minimum of 50 µs sector erase wait time is started by specifying the address for the sixth cycle as the address in the target sector and writing the sector erase code (30_H) as data.
- To erase data from more than one sector, program the erase code (30_H) to the sector address to be erased, following the above.

● Notes on specifying two or more sectors

- Sector erasing is started after a 50 µs period waiting for sector erasing is completed after the last sector erase code has been programmed.
- To erase data from two or more sectors simultaneously, input the sector addresses and the erase code (the sixth cycle of the command sequence) within a minimum of 50 µs sector erase wait time. If the erase code is input after 50 µs or later, it cannot be accepted.
- The sector erase timer flag (DQ3) can be used to check whether it is valid to write consecutive sector erase codes.
- When reading the sector erase timer flag (DQ3), specify the address of the sector to be erased.

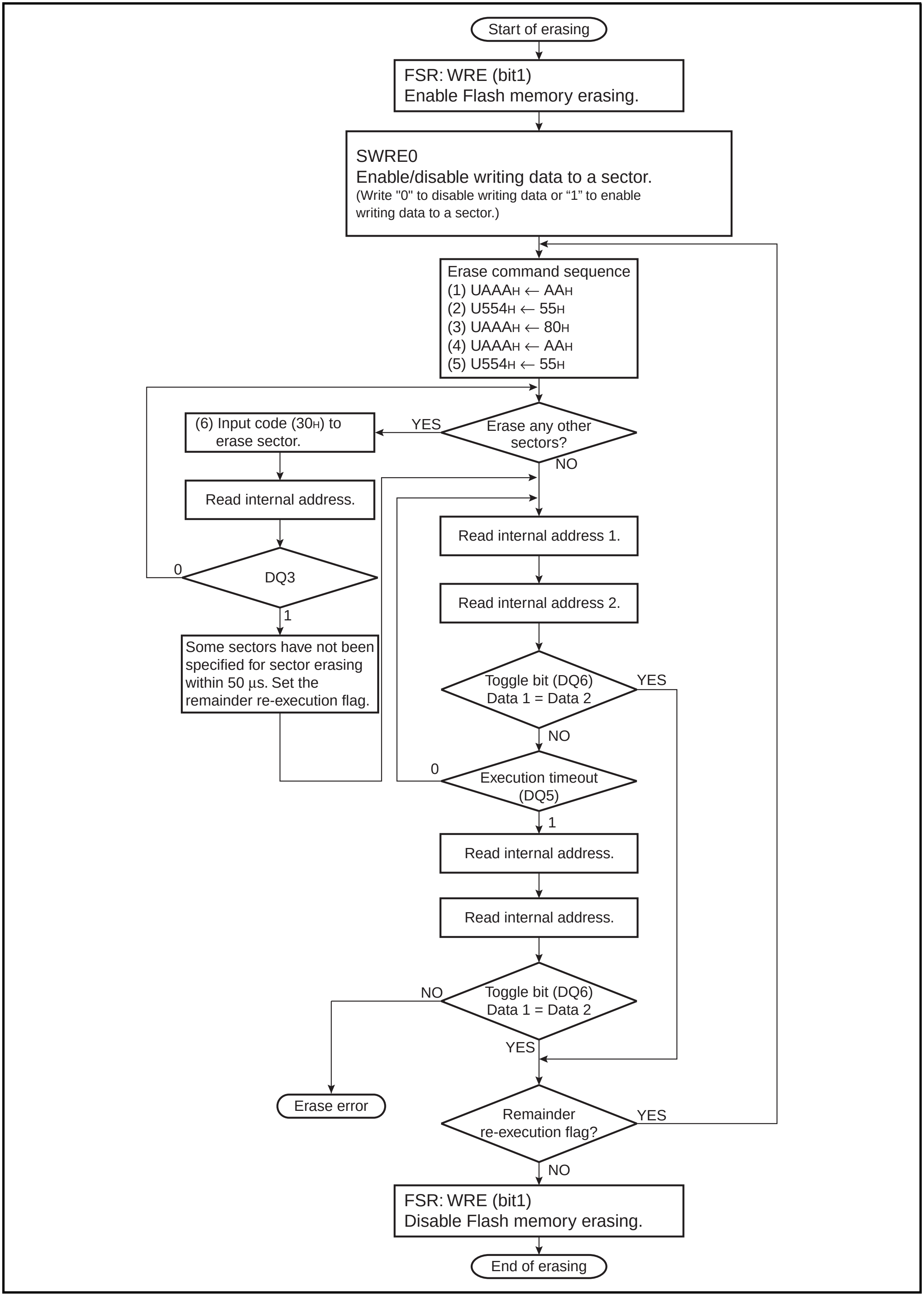
■ Flash Memory Sector Erasing Procedure

- Hardware sequence flags can be used to check the state of the automatic algorithm in the Flash memory. Figure 30.6-2 gives an example of the Flash memory sector erase procedure. In this example, the toggle bit flag (DQ6) is used to check that sector erasing is completed.
- The toggle bit flag (DQ6) stops toggling the output concurrently with the change of the execution timeout flag (DQ5) to "1". So the toggle bit flag (DQ6) must be checked even when the execution timeout flag (DQ5) is "1".
- Similarly, the data polling flag (DQ7) changes concurrently with the transition of the execution timeout flag (DQ5), so the data polling flag (DQ7) must be checked.

■ Notes on Erasing Data from Sectors

If a hardware reset occurs during erasing data from a sector, the data being erased is not guaranteed. Retry erasing the same sector.

Figure 30.6-2 Sample Procedure for Erasing Data from Sectors in Flash Memory


Start of erasing
FSR: WRE (bit1)
Enable Flash memory erasing.
SWRE0
Enable/disable writing data to a sector.
(Write "0" to disable writing data or "1" to enable writing data to a sector.)
Erase command sequence
(1) UAAAH ← AAH
(2) U554H ← 55H
(3) UAAAH ← 80H
(4) UAAAH ← AAH
(5) U554H ← 55H
Erase any other sectors?
YES
NO
(6) Input code (30H) to erase sector.
Read internal address.
DQ3
0
1
Some sectors have not been specified for sector erasing within 50 µs. Set the remainder re-execution flag.
Read internal address 1.
Read internal address 2.
Toggle bit (DQ6) Data 1 = Data 2
YES
NO
Execution timeout (DQ5)
0
1
Read internal address.
Read internal address.
Toggle bit (DQ6) Data 1 = Data 2
NO
YES
Erase error
Remainder re-execution flag?
YES
NO
FSR: WRE (bit1)
Disable Flash memory erasing.
End of erasing

30.6.5 Suspending Sector Erasing from Flash Memory

The section explains the procedure for entering the sector erase suspend command to suspend sector erasing from the Flash memory. Data can be read from sectors not being erased.

■ Suspending Sector Erasing from Flash Memory

- To suspend the Flash memory sector erasing, send the sector erase suspend command mentioned in the command sequence table from the CPU to the Flash memory.
- The sector erase suspend command suspends the current sector erase operation, allowing data to be read from sectors that are not being erased.
- The sector erase suspend command is only enabled during the sector erase period including the erase wait time; it is ignored during chip erasing or writing.
- The sector erase suspend command is executed when the sector erase suspend code ($B0_H$) is written. Specify an address in the sector selected to be erased. If an attempt is made to execute the sector erase suspend command again when sector erasing has been suspended, the new sector erase suspend command input is ignored.
- When a sector erase suspend command is input during the sector erase wait period, the sector erase wait time ends immediately, the sector erase operation is stopped, and the Flash memory enters the erase stop state.
- When a sector erase suspend command is input during sector erasing after the sector erase wait period, the erase suspend state occurs after a maximum of 20 µs has elapsed since the issue of the sector erase suspend command.

Note:

To suspend sector erasing by issuing a sector erase suspend command, issue the command after 20 ms or longer has elapsed since the issue of a sector erase command or a sector erase resume command.

30.6.6 Resuming Sector Erasing from Flash Memory

This section explains the procedure for entering the sector erase resume command to resume suspended erasing of a sector in the Flash memory.

■ Resuming Sector Erasing from Flash Memory

- To resume suspended sector erasing, send the sector erase resume command mentioned in the command sequence table from the CPU to the Flash memory.
- The sector erase resume command resumes a sector erase operation suspended by the sector erase suspend command. The sector erase resume command is executed by writing erase resume code (30_H). Specify an address in the sector selected to be erased.
- A sector erase resume command input during sector erasing is ignored.

30.7 Operations of Dual Operation Flash Memory

Pay attention in particular to the following points when using the dual operation Flash memory:

- **Interrupt generated when the upper bank is updated**
- **Procedure for setting the sector conversion enable bit in the flash memory status register (FSR:SSEN)**

■ Interrupt Generated When the Upper Bank Is Updated

The dual operation Flash memory consists of two banks. Like conventional Flash products, however, it cannot be erased/written and read at the same time in banks on the same side.

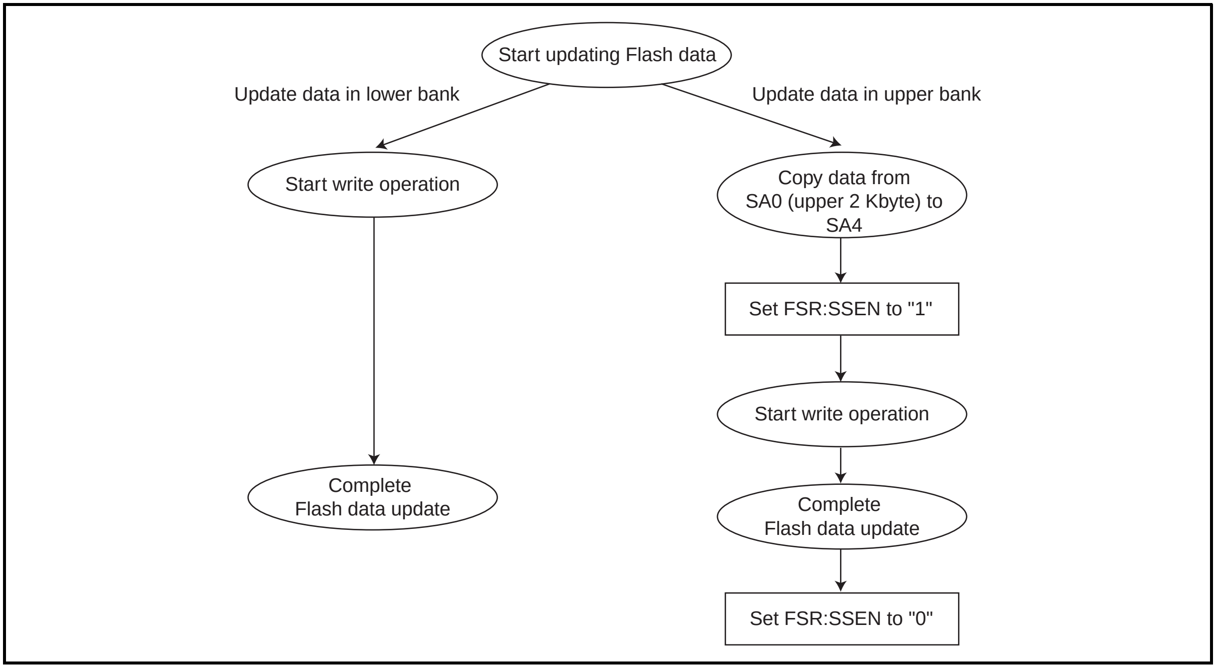
As SA0 contains an interrupt vector, an interrupt vector from the CPU cannot be read normally when an interrupt occurs during a write to the upper bank. Before the upper bank can be updated, set the sector swap enable bit (FSR:SSEN) to "1". When an interrupt occurs, therefore, SA4 is accessed to read interrupt vector data. The data of SA0 must be copied to SA4 before the sector swap enable bit (FSR:SSEN) is set.

■ Procedure of Setting the Sector Conversion Enable Bit (FSR:SSEN)

Figure 30.7-1 shows a sample procedure of setting the sector swap enable bit (FSR:SSEN).

To modify data in the upper bank, set FSR:SSEN to "1". While data is being written to the Flash memory, modifying the setting of FSR:SSEN is prohibited. The setting of FSR:SSEN can only be modified before the start of writing data to the Flash memory or after the completion of writing data to the Flash memory. In addition, control the Flash memory interrupts while setting FSR:SSEN as follows: before setting FSR:SSEN, disable the Flash memory interrupts; after setting FSR:SSEN, enable the interrupts.

Figure 30.7-1 Sample Procedure for Setting the Sector Conversion Enable Bit (FSR:SSEN)


Start updating Flash data
Update data in lower bank
Update data in upper bank
Start write operation
Copy data from SA0 (upper 2 Kbyte) to SA4
Set FSR:SSEN to "1"
Start write operation
Complete Flash data update
Complete Flash data update
Set FSR:SSEN to "0"

■ Operation During Programming/Erasing

It is prohibited to write data to the Flash memory within an interrupt routine when an interrupt occurs during Flash memory writing/erasing.

When two or more write/erase routines exist, wait for one write/erase routine to finish before executing another write/erase routine.

While data is being written to or erased from the Flash memory, making state transition in the current mode (clock mode or standby mode) is prohibited. Ensure that writing data to or erasing data from the Flash memory ends before making state transition.

■ Register and Vector Table Addresses Related to Dual Operation Flash Memory Interrupts

Table 30.7-1 Register and Vector Table Addresses Related to Dual Operation Flash Memory Interrupts

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
Flash memory	IRQ23	ILR5	L23	FFCC_H	FFCD_H

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

30.8 Flash Security

The Flash security controller function prevents the contents of the Flash memory from being read through external pins.

■ Flash Security

Writing protection code "01_H" to the Flash memory address ($FFFC_H$) restricts access to the Flash memory, disabling any read/write access to the Flash memory from any external pin. Once the protection of the Flash memory is enabled, the function cannot be unlocked until a chip erase command operation is executed.

It is advisable to write the protection code at the end of Flash writing to avoid enabling unnecessary protection during writing.

Once Flash security is enabled, a chip erase operation must be executed before data can be written to the Flash memory again.

30.9 Notes on Using Dual Operation Flash Memory

This section provides notes on using the dual operation Flash memory.

■ Restriction on Using Toggle Bit Flag (DQ6)

When using the dual operation Flash memory (The Flash memory write control program is executed on the Flash memory), the toggle bit flag (DQ6) cannot be used to check the operating state of the Flash memory during writing or erasing. Therefore, use the data polling flag (DQ7) to check the internal operating state of the Flash memory after writing data to the Flash memory or erasing data from the Flash memory as shown in the examples in Figure 30.6-1 and Figure 30.6-2.

The restriction above does not apply if the Flash memory write control program is executed on the RAM.

CHAPTER 31

EXAMPLE OF SERIAL PROGRAMMING CONNECTION

This chapter describes the example of serial programming connection.

31.1 Basic Configuration of Serial Programming Connection

The MB95410H/470H Series support Flash memory serial on-board programming. This section describes the configuration.

■ Basic Configuration of Serial Programming Connection

The BGM adaptor MB2146-08-E, manufactured by Fujitsu Semiconductor Limited, is used for serial onboard programming.

Figure 31.1-1 shows the basic configuration of serial programming connection.

Figure 31.1-1 Basic Configuration of Serial Programming Connection

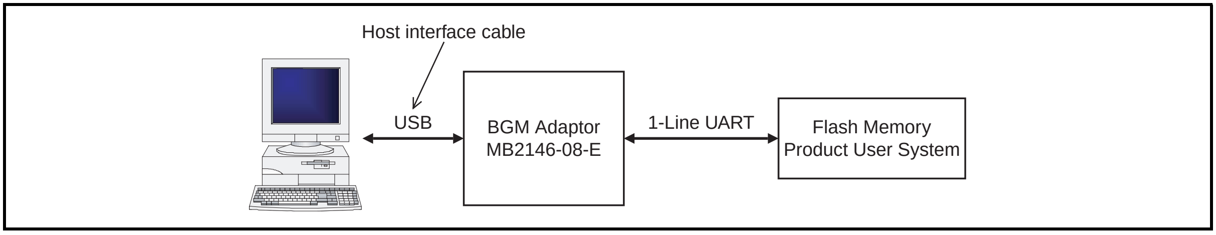


Table 31.1-1 Pins Used for Fujitsu Semiconductor Standard Serial Onboard Programming

Pin	Function	Description
V_{CC}	Power supply voltage supply pin	The write voltage (4.5 V to 5.5 V) is supplied from the user system.
V_{SS}	GND pin	It is shared with the GND of the Flash microcontroller programmer.
$\overline{\text{RST}}$	Reset	The $\overline{\text{RST}}$ pin is pulled up to V_{CC}.
DBG	1-line UART setting serial write mode	The DBG pin provides 1-line UART communication with the programmer. Serial write mode is set if voltage is supplied to the DBG pin and the V_{CC} pin at specific timings. (For the timings, see Figure 31.2-1.)

- Oscillation Clock Frequency

The UART clock is provided by the main CR clock. The UART baud rate needs to be set to 31250 bps or 62500 bps depending on the Flash memory operation to be executed.

31.2 Example of Serial Programming Connection

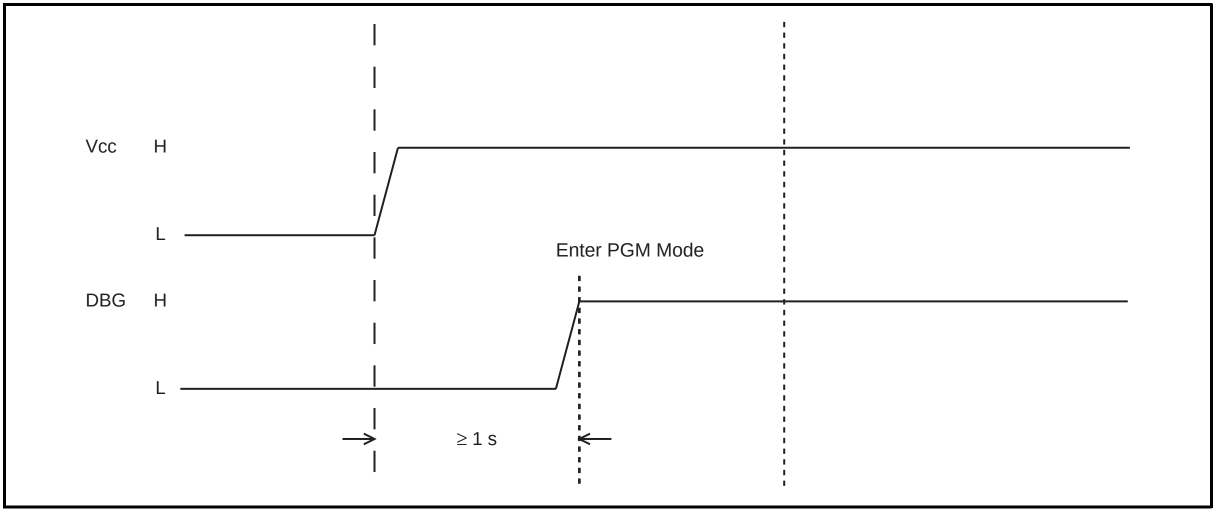
The microcontroller enters the PGM mode at the following timing.

■ MCU Entering PGM mode

The microcontroller enters the PGM mode at the following timing.

The serial programmer controls the DBG pin according to V_{CC} input.

Figure 31.2-1 Timing Diagram

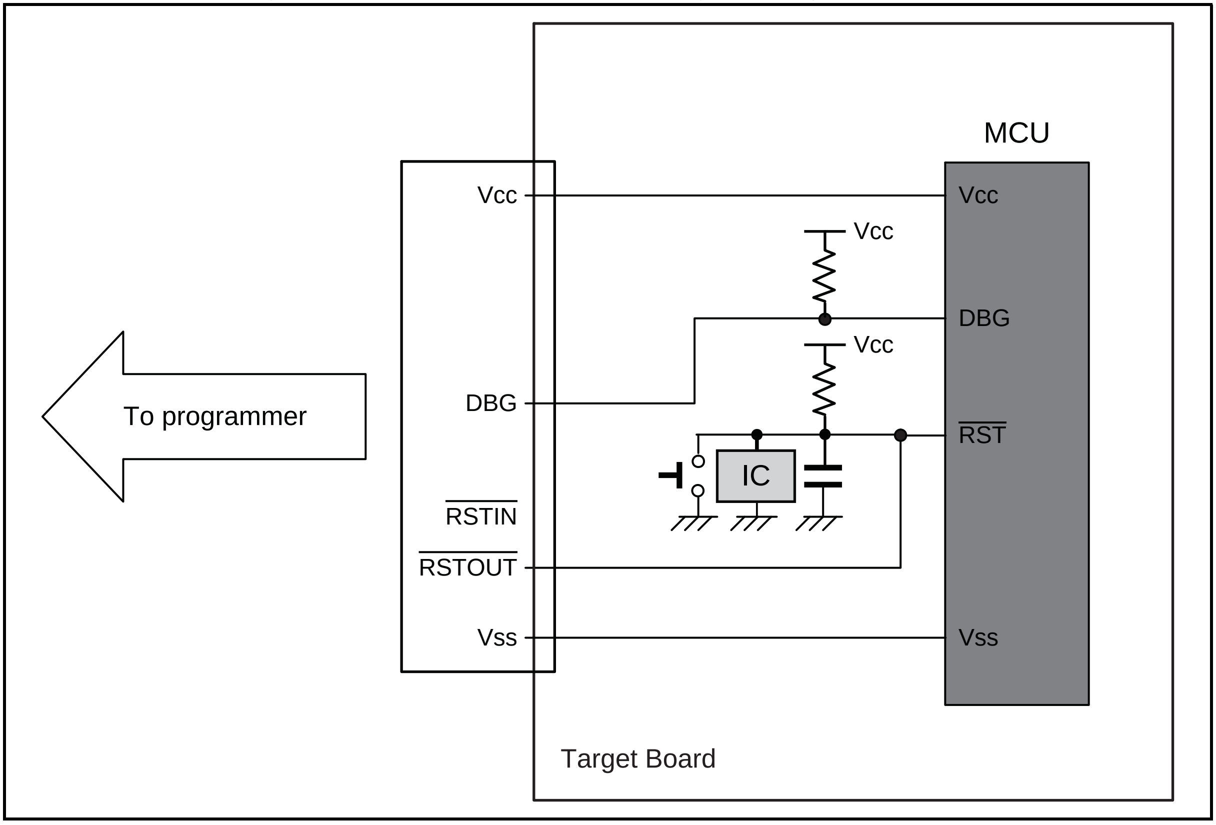


■ Example of Serial Programming Connection

Figure 31.2-2 shows an example of connection for serial writing.

The power is supplied from the programmer through the V_{CC} pin to the adaptor.

Figure 31.2-2 Example of Serial Programming Connection



Since the pull-up resistance depends on the tool used and the interconnection length, refer to the tool document when selecting a pull-up resistor.

In the case of using the BGM adaptor MB2146-08-E of Fujitsu Semiconductor Limited, it is recommended to use a pull-up resistor of approximately 2 kΩ to 10 kΩ.

CHAPTER 32

NON-VOLATILE REGISTER (NVR) FUNCTION

This chapter describes the functions and operations of the NVR interface.

32.1 Overview of NVR Interface

The NVR (Non-Volatile Register) area is a reserved area in the Flash that stores system information and option settings. After a reset, data in the NVR Flash area will be fetched and stored in registers in the NVR IO area. In the MB95410H/470H Series, the NVR interface is used to store the following data:

- **Frequency selection for main CR Clock (2 bits)**
- **Coarse trimming value for main CR Clock (5 bits)**
- **Fine trimming value for main CR Clock (6 bits)**
- **Watchdog Timer Selection ID (16 bits)**

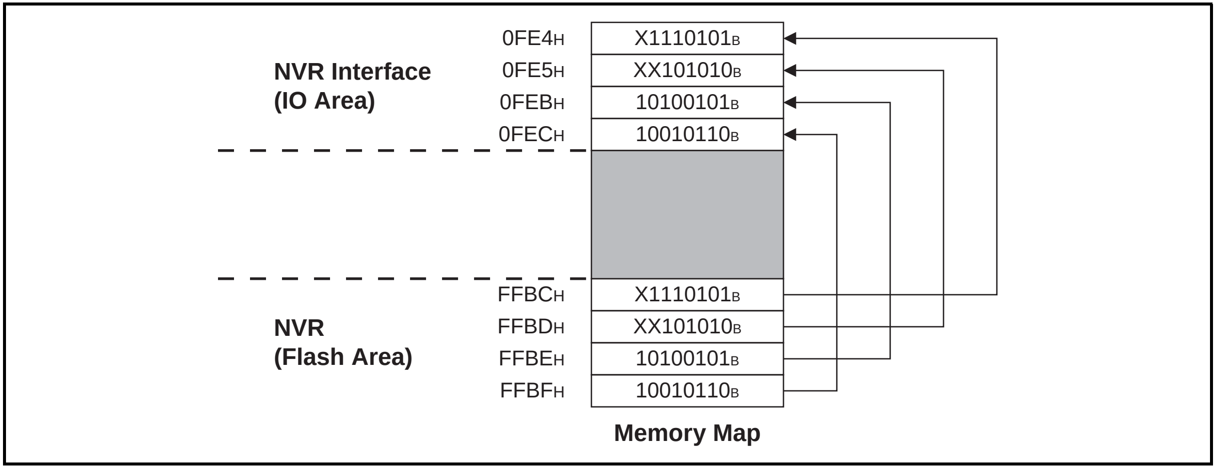
■ Functions of NVR Interface

Functions of the NVR interface are as follows:

1. The NVR interface retrieves all data from the NVR Flash area and stores it in the registers in the NVR IO area after a reset. (See Figure 32.1-1 and Figure 32.2-1.)
2. The NVR interface enables the user to choose the frequency of the main CR clock (1 MHz/ 8 MHz/10 MHz/12.5 MHz) by setting the frequency selection bits.
3. The NVR interface enables the user to know the value of the initial CR trimming setting.
4. The NVR interface enables the user to select the hardware watchdog timer or software watchdog timer by modifying the 16-bit watchdog timer selection ID (The watchdog timer selection ID cannot be modified while the CPU is running.)

Figure 32.1-1 shows the retrieval of the NVR interface during a reset.

Figure 32.1-1 Retrieval of NVR during Reset



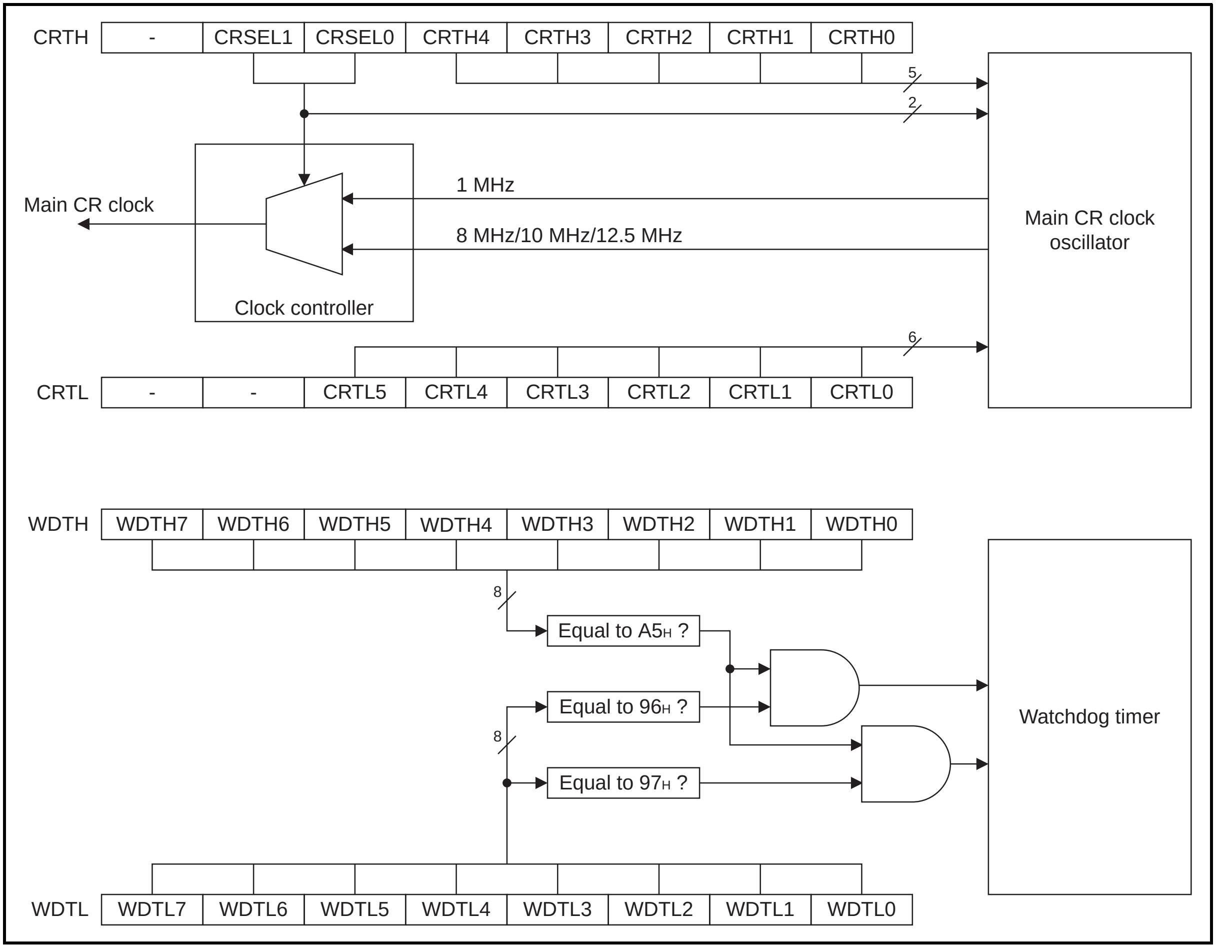
32.2 Configuration of NVR Interface

The NVR interface consists of the following blocks:

- **Main CR Clock Frequency Selection (CRSEL)**
- **Trimming of Main CR Clock (CRTH and CRTL)**
- **Watchdog Timer Selection ID (WDTH and WDTL)**

■ Block Diagram of NVR Interface

Figure 32.2-1 Block Diagram of NVR Interface



32.3 Registers of NVR Interface

This section lists the registers of the NVR interface.

■ Registers of NVR Interface

Figure 32.3-1 Registers of NVR Interface

	Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
CRTH	0FE4$_H$	–	CRSEL1	CRSEL0	CRTH4	CRTH3	CRTH2	CRTH1	CRTH0	0XXXXXXX$_B$
		R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
CRTL	0FE5$_H$	–	–	CRTL5	CRTL4	CRTL3	CRTL2	CRTL1	CRTL0	00XXXXXX$_B$
		R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	
WDTH	0FEB$_H$	WDTH7	WDTH6	WDTH5	WDTH4	WDTH3	WDTH2	WDTH1	WDTH0	XXXXXXXX$_B$
		R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	
WDTL	0FEC$_H$	WDTL7	WDTL6	WDTL5	WDTL4	WDTL3	WDTL2	WDTL1	WDTL0	XXXXXXXX$_B$
		R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	R/WX	

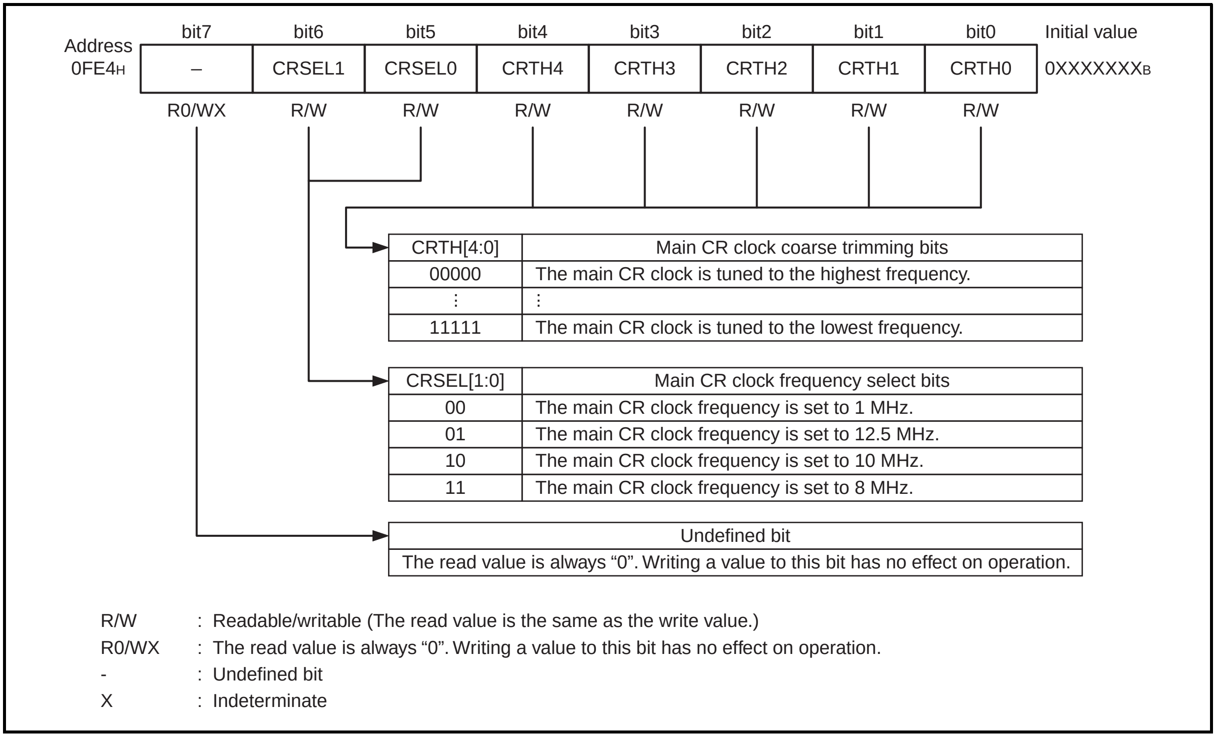
R/W	:	Readable/writable (The read value is the same as the write value.)
R/WX	:	Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX	:	The read value is “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
X	:	Indeterminate

32.3.1 Main CR Clock Trimming Register (Upper) (CRTH)

Figure 32.3-2 shows the main CR clock trimming register (upper) (CRTH).

■ Main CR Clock Trimming Register (Upper) (CRTH)

Figure 32.3-2 Main CR Clock Trimming Register (Upper) (CRTH)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FE4H	–	CRSEL1	CRSEL0	CRTH4	CRTH3	CRTH2	CRTH1	CRTH0	0XXXXXXXB
	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

CRTH[4:0]	Main CR clock coarse trimming bits
00000	The main CR clock is tuned to the highest frequency.
⋮	⋮
11111	The main CR clock is tuned to the lowest frequency.

CRSEL[1:0]	Main CR clock frequency select bits
00	The main CR clock frequency is set to 1 MHz.
01	The main CR clock frequency is set to 12.5 MHz.
10	The main CR clock frequency is set to 10 MHz.
11	The main CR clock frequency is set to 8 MHz.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is always “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
X : Indeterminate

Table 32.3-1 Functions of Bits in Main CR Clock Trimming Register (Upper) (CRTH) (1 / 2)

Bit name		Function
bit7	Undefined bit	The read value is always "0". Writing a value to this bit has no effect on operation.
bit6, bit5	CRSEL1, CRSEL0: Main CR clock frequency select bits	These bits are loaded from the Flash address FFBC$_H$ (bit6 and bit5) after a reset. Their initial values are determined by the pre-loaded values in the NVR Flash area. The frequency of the main CR clock can be selected by modifying the values of CRSEL. CRSEL[1:0] — Main CR clock frequency 00$_B$ — 1 MHz 01$_B$ — 12.5 MHz 10$_B$ — 10 MHz 11$_B$ — 8 MHz See "32.5 Notes on Using NVR" for notes on changing the main CR frequency selection.

Table 32.3-1 Functions of Bits in Main CR Clock Trimming Register (Upper) (CRTH) (2 / 2)

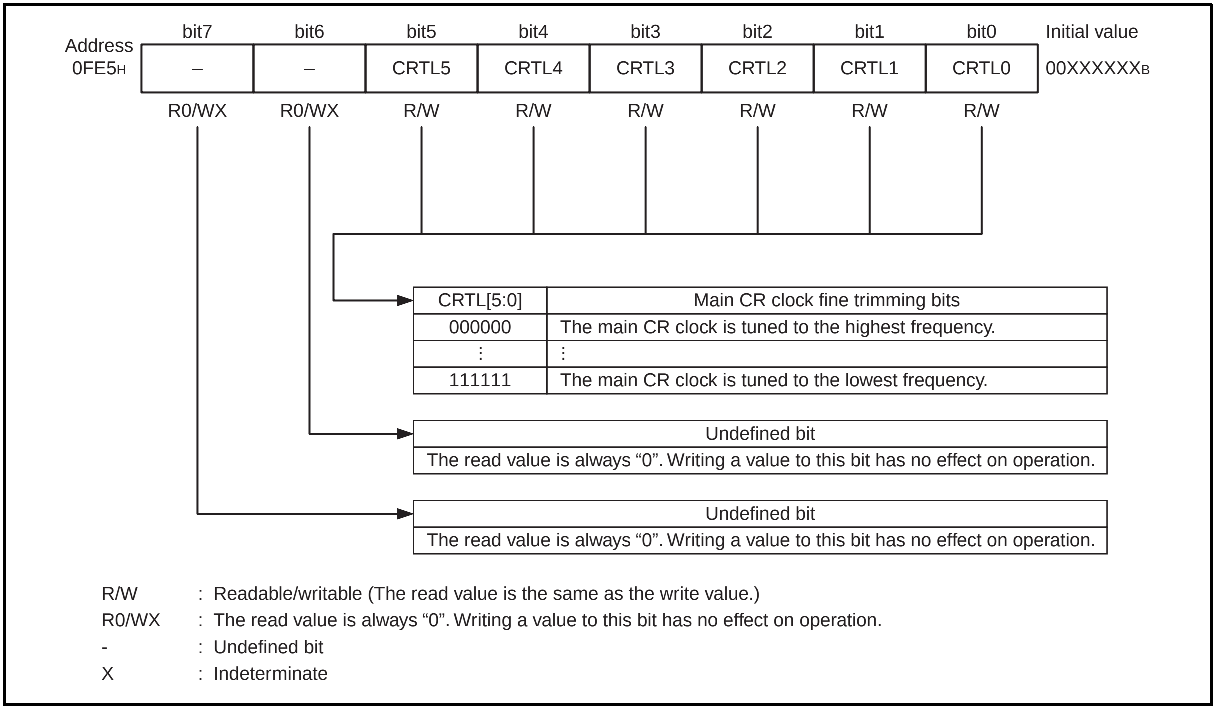
<table>
<tr><th colspan="2">Bit name</th><th>Function</th></tr>
<tr><td>bit4
to
bit0</td><td>CRTH4 to CRTH0:
Main CR coarse trimming bits</td><td>These bits are loaded from the Flash address FFBC$_H$ (bit4 to bit0) after a reset. Their initial values are determined by the pre-loaded values in the NVR Flash area.
Coarse trimming modifies the main CR clock frequency with a bigger step. Increasing the coarse trimming value can decrease the main CR clock frequency. See the table below.
<table>
<tr><th>CRTH [4:0]</th><th>Main CR clock frequency</th></tr>
<tr><td>00000$_B$</td><td>Highest</td></tr>
<tr><td>:</td><td>:</td></tr>
<tr><td>11111$_B$</td><td>Lowest</td></tr>
</table>
See "32.4 Notes on Main CR Clock Trimming" and "32.5 Notes on Using NVR" for details of main CR clock trimming and notes on changing the main CR clock values respectively.</td></tr>
</table>

32.3.2 Main CR Clock Trimming Register (Lower) (CRTL)

Figure 32.3-3 shows the main CR clock trimming register (lower) (CRTL).

■ Main CR Clock Trimming Register (Lower) (CRTL)

Figure 32.3-3 Main CR Clock Trimming Register (Lower) (CRTL)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0FE5H	–	–	CRTL5	CRTL4	CRTL3	CRTL2	CRTL1	CRTL0	00XXXXXXB
	R0/WX	R0/WX	R/W	R/W	R/W	R/W	R/W	R/W	

CRTL[5:0]	Main CR clock fine trimming bits
000000	The main CR clock is tuned to the highest frequency.
⋮	⋮
111111	The main CR clock is tuned to the lowest frequency.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W : Readable/writable (The read value is the same as the write value.)
R0/WX : The read value is always “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
X : Indeterminate

Table 32.3-2 Functions of Bits in CR Trimming Register (Lower) (CRTL)

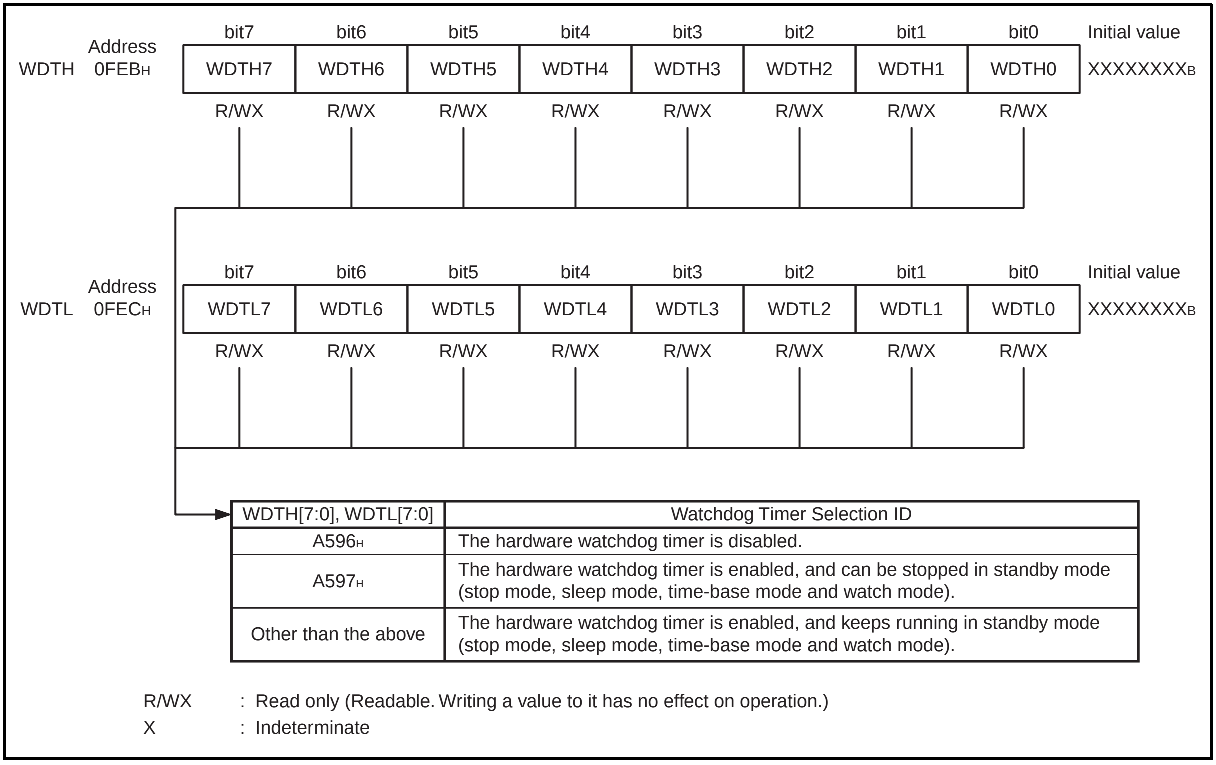
<table>
<tr><th colspan="2">Bit Name</th><th>Function</th></tr>
<tr><td>bit7,
bit6</td><td>Undefined bits</td><td>Their read values are always "0". Writing values to these bits has no effect on operation.</td></tr>
<tr><td>bit5
to
bit0</td><td>CRTL5 to CRTL0:
Main CR fine trimming bits</td><td>These bits are loaded from the Flash address FFBD_H (bit5 to bit0) after a reset. Their initial values are determined by the pre-load values in the NVR Flash area.
Fine trimming modifies the main CR clock frequency with a smaller step.
Increasing the fine trimming value can decrease the main CR clock frequency.
<table>
<tr><th>CRTL [5:0]</th><th>Main CR clock frequency</th></tr>
<tr><td>000000_B</td><td>Highest</td></tr>
<tr><td>:</td><td>:</td></tr>
<tr><td>111111_B</td><td>Lowest</td></tr>
</table>
See "32.4 Notes on Main CR Clock Trimming" and "32.5 Notes on Using NVR" for details of main CR clock trimming and notes on changing the main CR clock values respectively.</td></tr>
</table>

32.3.3 Watchdog Timer Selection ID Registers (WDTH,WDTL)

Figure 32.3-4 shows watchdog timer selection ID registers (WDTH, WDTL).

■ Watchdog Timer Selection ID Registers (WDTH, WDTL)

Figure 32.3-4 Watchdog Timer Selection ID Registers (WDTH,WDTL)


WDTH
Address
0FEBH
bit7 bit6 bit5 bit4 bit3 bit2 bit1 bit0
WDTH7 WDTH6 WDTH5 WDTH4 WDTH3 WDTH2 WDTH1 WDTH0
R/WX R/WX R/WX R/WX R/WX R/WX R/WX R/WX
Initial value
XXXXXXXXB
WDTL
Address
0FECH
bit7 bit6 bit5 bit4 bit3 bit2 bit1 bit0
WDTL7 WDTL6 WDTL5 WDTL4 WDTL3 WDTL2 WDTL1 WDTL0
R/WX R/WX R/WX R/WX R/WX R/WX R/WX R/WX
Initial value
XXXXXXXXB

WDTH[7:0], WDTL[7:0]	Watchdog Timer Selection ID
A596H	The hardware watchdog timer is disabled.
A597H	The hardware watchdog timer is enabled, and can be stopped in standby mode (stop mode, sleep mode, time-base mode and watch mode).
Other than the above	The hardware watchdog timer is enabled, and keeps running in standby mode (stop mode, sleep mode, time-base mode and watch mode).

R/WX : Read only (Readable. Writing a value to it has no effect on operation.)
X : Indeterminate

Table 32.3-3 Functions of Bits in Watchdog Timer ID Register (Upper) (WDTH)

Bit name		Function
bit7 to bit0	WDTH7 to WDTH0: Watchdog timer selection ID (upper)	These bits are loaded from the Flash address FFBE$_H$ (bit7 to bit0) after a reset. The initial values are determined by the pre-loaded values in the NVR Flash area. This register cannot be modified while the CPU is running. See Table 32.3-5 for watchdog timer selection. See "32.5 Notes on Using NVR" for notes on writing NVR values.

Table 32.3-4 Functions of Bits in Watchdog Timer ID Register (Lower) (WDTL)

Bit name		Function
bit7 to bit0	WDTL7 to WDTL0: Watchdog timer selection ID (lower)	These bits are loaded from the Flash address FFBF$_H$ (bit7 to bit0) after a reset. The initial values are determined by the pre-loaded values in the NVR Flash area. This register cannot be modified while the CPU is running. See Table 32.3-5 for watchdog timer selection. See "32.5 Notes on Using NVR" for notes on writing NVR values.

Table 32.3-5 Watchdog Timer Selection ID

WDTH[7:0], WDTL[7:0]	Function
$A596_H$	The hardware watchdog timer is disabled; the software watchdog timer is enabled.
$A597_H$	The hardware watchdog timer is enabled; the software watchdog timer is disabled. The hardware watchdog timer can be stopped in all standby modes (stop mode, sleep mode, time-base timer mode and watch mode).
Other than the above	The hardware watchdog timer is enabled; the software watchdog timer is disabled. The hardware watchdog timer keeps operating in all standby modes (stop mode, sleep mode, time-base timer mode and watch mode).

32.4 Notes on Main CR Clock Trimming

This section provides notes on main CR clock trimming.

After a hardware reset, the 11-bit CR clock trimming value will be loaded from the NVR Flash area to registers in the NVR IO area.

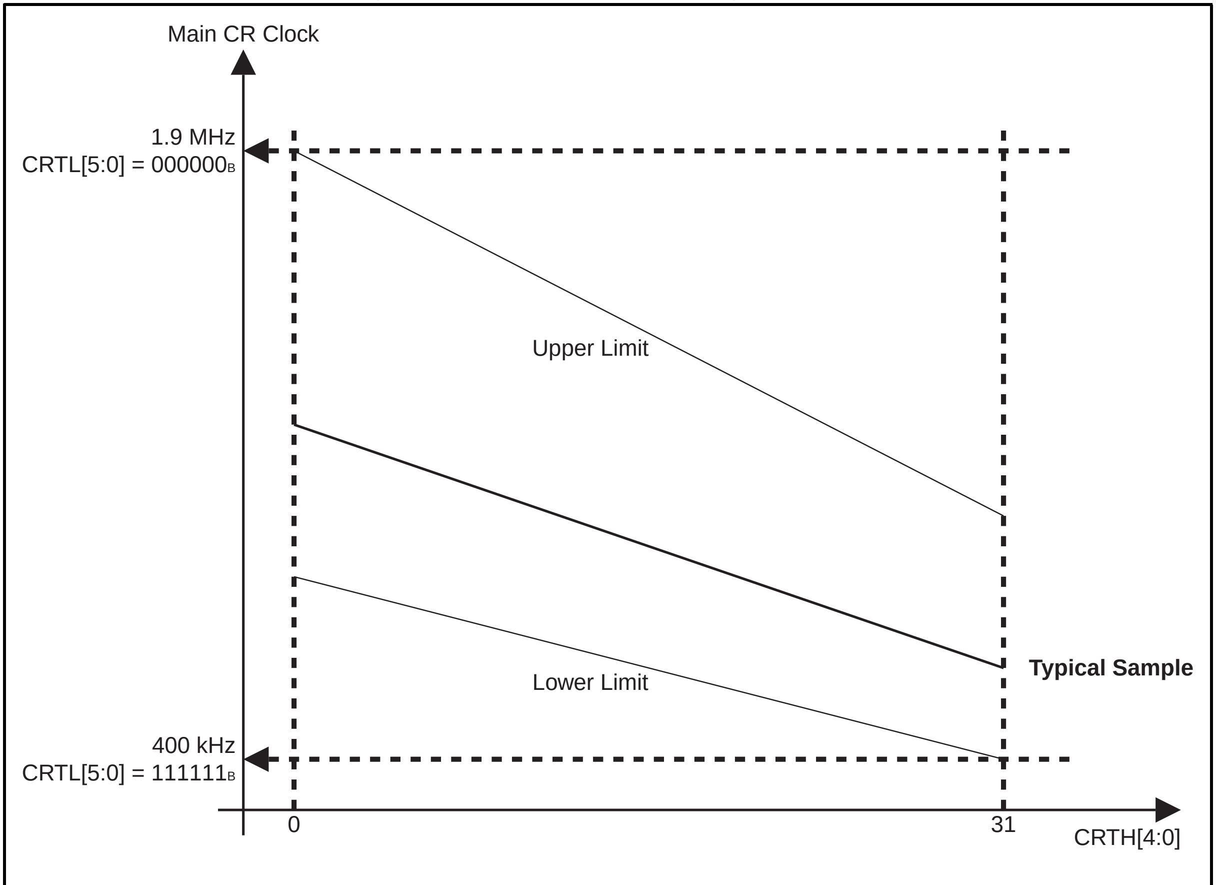
Table 32.4-1 shows the step size of CR Trimming.

Table 32.4-1 Step Size of CR Trimming

Function	Coarse trimming value CRTH[4:0]	Fine trimming value CRTL[5:0]
To achieve the minimum frequency	11111_B	111111_B
To achieve the maximum frequency	00000_B	000000_B
Step Size	20 kHz to 50 kHz	1.6 kHz to 8 kHz

The relationship between coarse trimming step size and CR frequency is shown in the diagram below.

Figure 32.4-1 Coarse Trimming Step Size



Note: The size of fine trimming ranges from -6.5% (CRTL[5:0] = 111111B) to +6.5% (CRTL[5:0] = 000000B).

32.5 Notes on Using NVR

This section provides notes on using NVR.

■ Notes on Changing Main CR Frequency

1. The frequency of the main CR clock can be selected by writing different values to the CRTH:CRSEL[1:0] bits. However, unstable oscillation occurs for a certain period of time after the modification of clock frequency has been initiated. To prevent such oscillation, it is strongly recommended that the following actions should be taken. Firstly, switch the CPU clock source from the main CR clock to another clock (main clock / subclock / sub-CR clock), then modify the main CR parameters, and switch back to the main CR clock.
2. Please note that the NVR interface does not program a modified value to the NVR Flash area. If the CRTH and CRTL registers are modified, the modified value is programmed to the NVR Flash area by the Flash writer.

■ Notes on Flash Erase and Trimming Value

1. A Flash erase operation will erase all NVR data.

   The Flash writer carries out the following procedure to keep original system settings.

   (1) Make a backup of data in CRTH:CRTH[4:0] and CRTL:CRTL[5:0].

   (2) Erase the Flash.

   (3) Restore all data in CRTH:CRTH[4:0] and CRTL:CRTL[5:0] to the NVR Flash area.

   If there is new data in CRTH:CRTH[4:0] and CRTL:CRTL[5:0], the Flash writer will write the new data to the NVR Flash area.
2. The trimming value has been preset before this device is shipped. If the preset trimming value is modified after the device has been shipped, Fujitsu Semiconductor does not warrant proper operation of the device with respect to use based on the modified trimming value.
3. If the Flash operation is performed by the user program code, the original trimming data should also be restored to the NVR Flash area by the user program code. Otherwise, the trimming value, which has been preset before this device is shipped, is erased by the Flash erase operation.

CHAPTER 33

VOLTAGE COMPARATOR

This chapter describes the functions and operations of the voltage comparator.

33.1 Overview of Voltage Comparator

The voltage comparator is used to monitor the voltages of two analog inputs, which can be either one internal input and one external input or two external inputs, and can automatically generate an interrupt upon detecting a change in the edge of voltage comparator output.

■ Functions of Voltage Comparator

The function of the voltage comparator is to monitor and compare the voltages of two analog inputs and compare them. Using the internal voltage or external voltage of the positive analog input as a reference voltage, the voltage comparator will output "H" if the voltage of the negative analog input is lower than the reference voltage; otherwise, it will output "L". In addition, upon detecting a rising edge or falling edge of the voltage comparator output, the voltage comparator generates a corresponding interrupt.

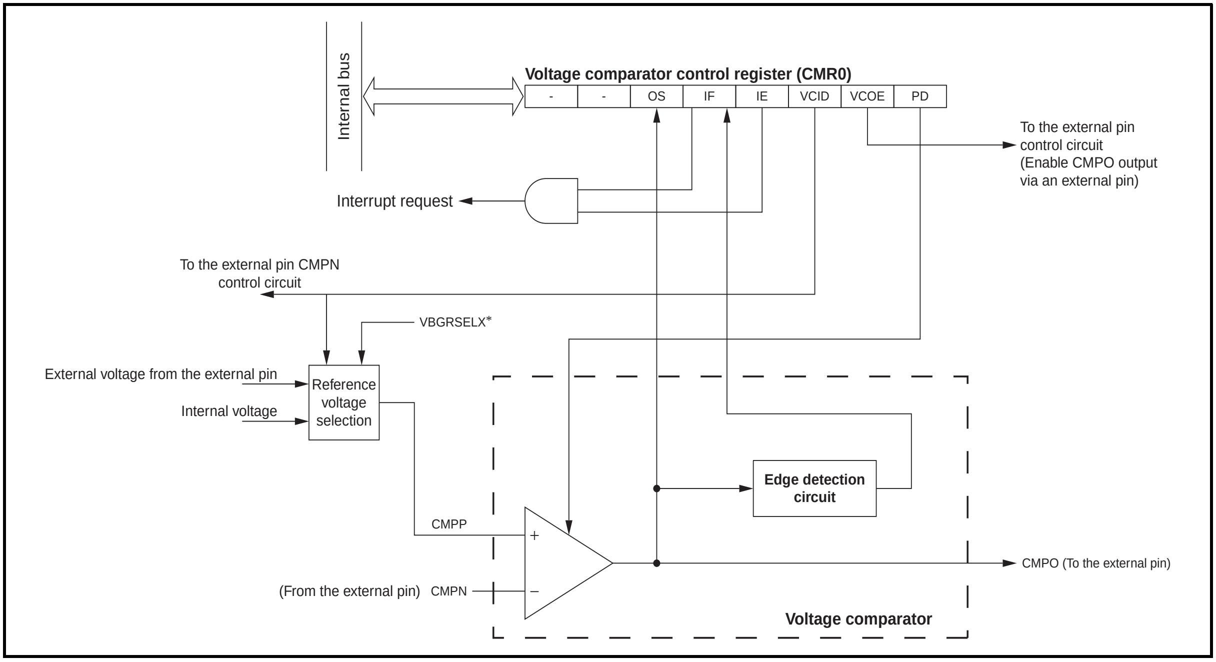
33.2 Configuration of Voltage Comparator

The entire voltage comparator module consists of the following blocks:

- **Voltage comparator × 1 channel**
- **Edge detection circuit × 1 channel**
- **Voltage comparator control register × 1 channel (CMR0)**

■ Block Diagram of Voltage Comparator

Figure 33.2-1 Block Diagram of Voltage Comparator


Internal bus
Voltage comparator control register (CMR0)
-
-
OS
IF
IE
VCID
VCOE
PD
To the external pin control circuit (Enable CMPO output via an external pin)
Interrupt request
To the external pin CMPN control circuit
VBGRSELX*
External voltage from the external pin
Internal voltage
Reference voltage selection
Edge detection circuit
CMPP
+
(From the external pin) CMPN
−
CMPO (To the external pin)
Voltage comparator

*: See "34.2 System Configuration Register (SYSC)" for details of the VBGRSELX bit and of selecting internal reference voltage and external reference voltage.

● Voltage comparator

The voltage comparator monitors and compares the voltages of two analog inputs and compare them. Using the internal voltage or external voltage of the positive analog input as a reference voltage, the voltage comparator will output "H" if the voltage of the negative analog input is lower than the reference voltage; otherwise, it will output "L".

● Edge detection circuit

Except in stop mode, watch mode or time-base timer mode, upon detection of a rising edge or falling edge of voltage comparator output, the edge detection circuit automatically raises an interrupt flag an interrupt flag (CMR0:IF).

● Voltage comparator control register (CMR0)

This register is used to turn on and off the voltage comparator (CMR0:PD), to enable and disable voltage comparator output (CMR0:VCOE), and to enable and disable voltage comparator analog inputs (CMR0:VCID).

Except in stop mode, watch mode or time-base timer mode, if the interrupt request enable bit (CMR0:IE) has been set to "1", upon detection of a rising edge or falling edge of voltage comparator output, the voltage comparator generates an interrupt request and the interrupt flag bit (CMR0:IF) is automatically set to "1" at the same time.

The output status can be read through the output status bit (CMR0:OS).

33.3 Pins of Voltage Comparator

This section describes the pins of the voltage comparator.

■ Pins of Voltage Comparator

Table 33.3-1 Pins of Voltage Comparator

Pin Name	Pin Function	I/O Type	Pull-up Option	Standby Control	Settings Required for Using The Pin	Default Status
P21/PPG01/ CMPP	GPIO/ 8/16-bit PPG ch. 0 output/Voltage comparator positive analog input	CMOS input/ CMOS output/ Analog input	Unavailable	Available	CMR0:VCID = 0 (Enables voltage comparator analog input)	GPIO input disabled/ GPIO output disabled/ 8/16-bit PPG ch. 0 output disabled/ Voltage comparator analog input enabled
P20/PPG00/ CMPN	GPIO/ 8/16-bit PPG ch. 0 output/Voltage comparator negative analog input	CMOS input/ CMOS output/ Analog input				GPIO input disabled/ GPIO output disabled/ 8/16-bit PPG ch. 0 output disabled/ Voltage comparator analog input enabled
P17/CMPO	GPIO/ Voltage comparator output	CMOS input/ CMOS output			CMR0:VCOE = 1 (Enables voltage comparator output)	GPIO input enabled/ GPIO output disabled/ Voltage comparator output disabled

Note:

When the voltage comparator analog input function of a pin is enabled, the GPIO function, and the input and output functions of other peripheral resources of the same pin will be disabled.

■ Block Diagrams of Pins of Voltage Comparator

Figure 33.3-1 Block Diagram of CMPP and CMPN of Voltage Comparator

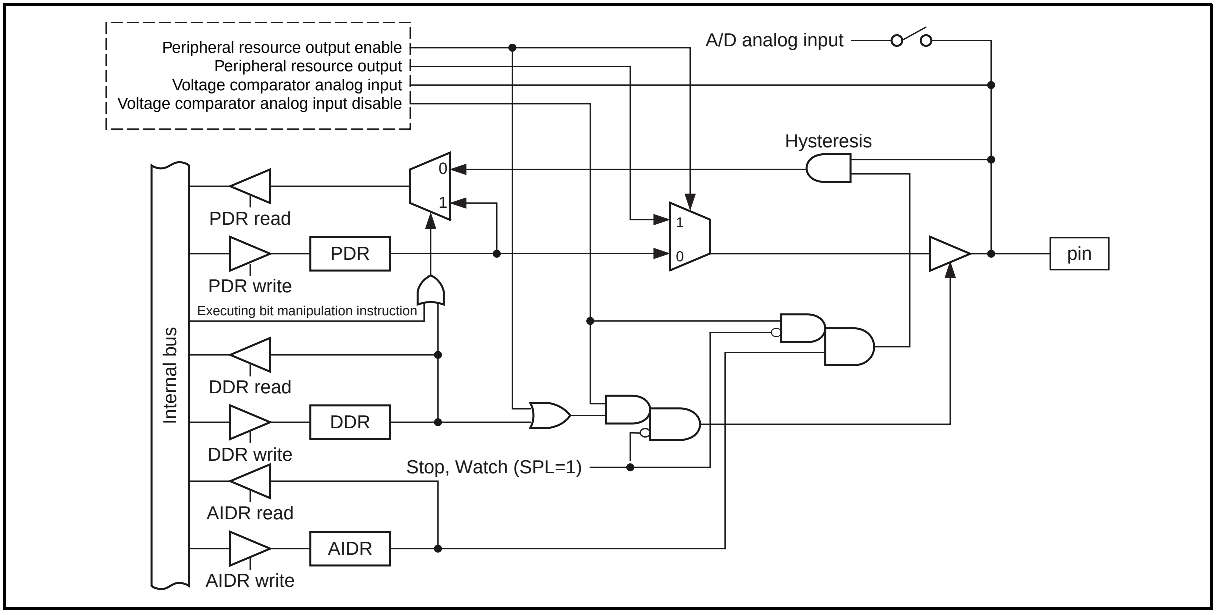
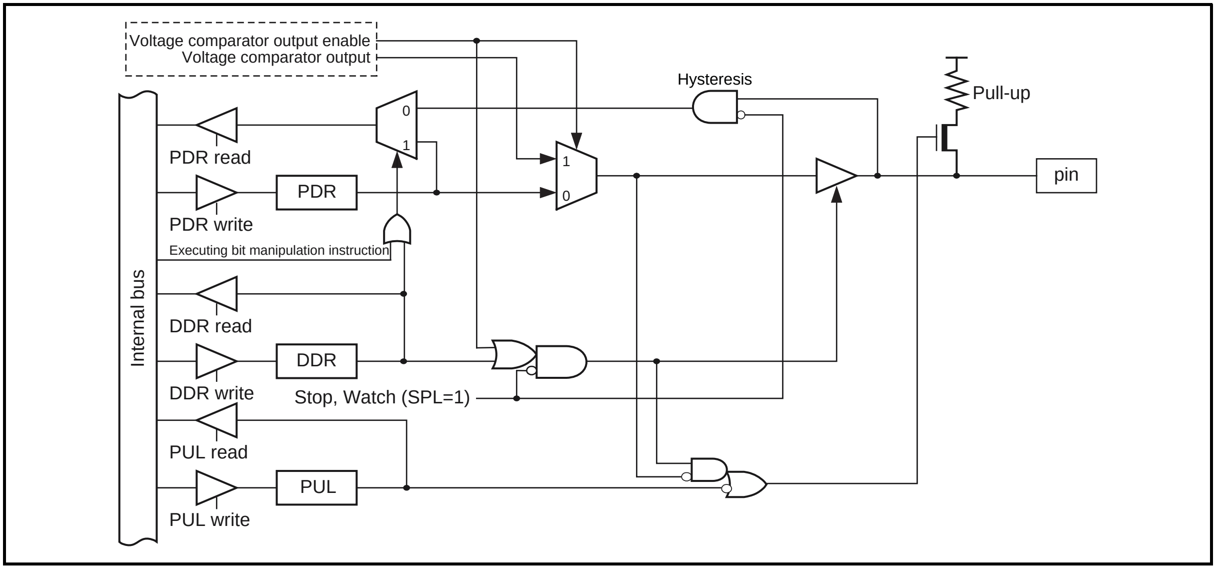

Peripheral resource output enable
Peripheral resource output
Voltage comparator analog input
Voltage comparator analog input disable
A/D analog input
Hysteresis
0
1
1
0
PDR read
PDR
PDR write
Executing bit manipulation instruction
pin
Internal bus
DDR read
DDR
DDR write
Stop, Watch (SPL=1)
AIDR read
AIDR
AIDR write

Figure 33.3-2 Block Diagram of CMPO of Voltage Comparator



33.4 Register of Voltage Comparator

This section describes the register of the voltage comparator.

■ Register of Voltage Comparator

Figure 33.4-1 Register of Voltage Comparator

Voltage comparator control register (CMR0)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0050_H	-	-	OS	IF	IE	VCID	VCOE	PD	$000X0001_B$
	R0/WX	R0/WX	R/WX	R(RM1),W	R/W	R/W	R/W	R/W	

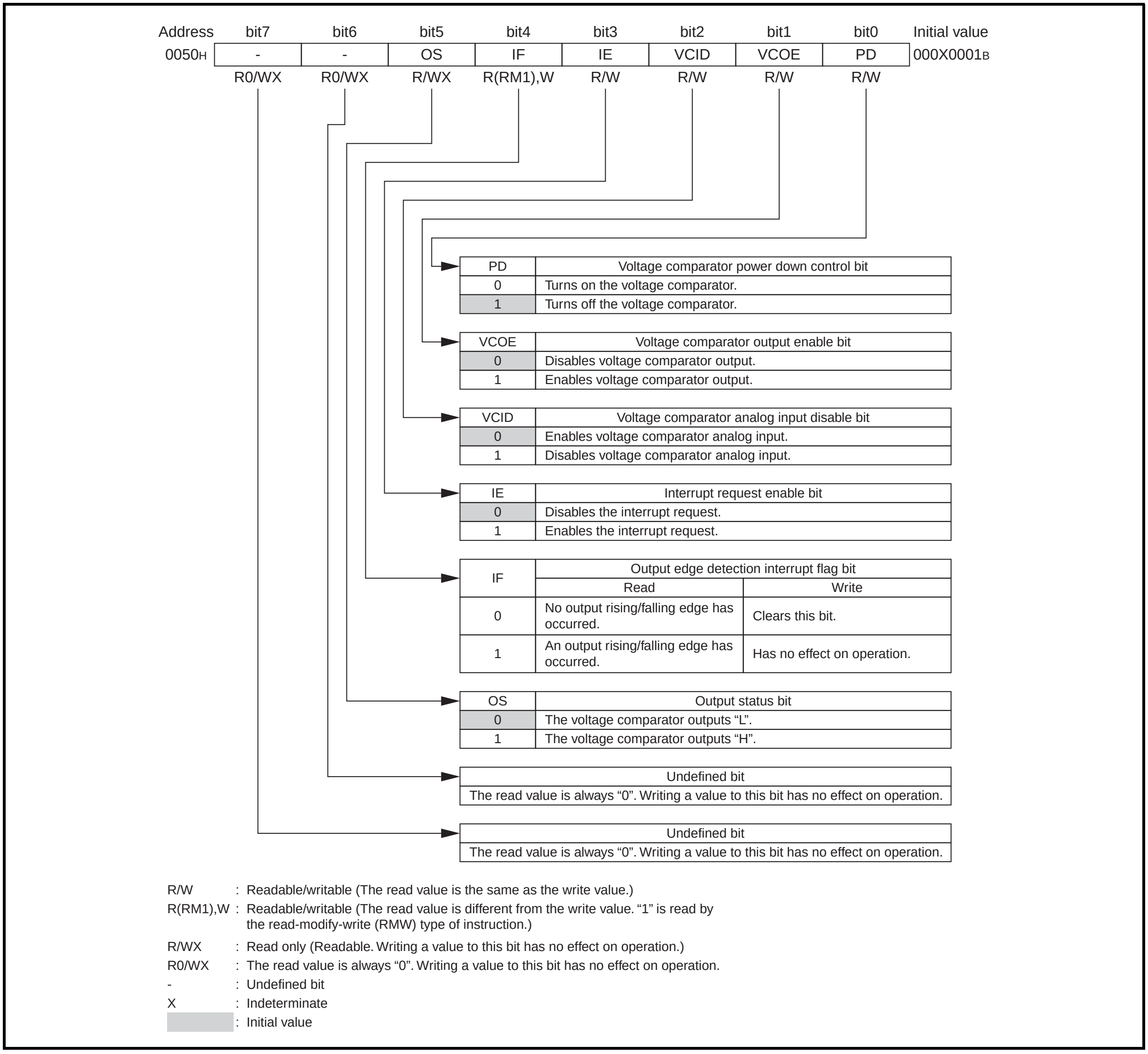
R/W : Readable/writable (The read value is the same as the write value.)
R(RM1), W : Readable/writable (The read value is different from the write value. "1" is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX : The read value is always "0". Writing a value to this bit has no effect on operation.
- : Undefined bit
X : Indeterminatet

33.4.1 Voltage Comparator Control Register (CMR0)

The voltage comparator control register is used to turn on and off the voltage comparator (CMR0:PD), to enable and disable voltage comparator output (CMR0:VCOE), and to enable and disable voltage comparator analog inputs (CMR0:VCID).
Except in stop mode, watch mode or time-base timer mode, if the interrupt request enable bit (CMR0:IE) has been set to "1", upon detection of a rising edge or falling edge of voltage comparator output, the voltage comparator generates an interrupt request and the interrupt flag bit (CMR0:IF) is automatically set to "1" at the same time.
The output status can be read through the output status bit (CMR0:OS).

■ Voltage Comparator Control Register (CMR0)

Figure 33.4-2 Voltage Comparator Control Register (CMR0)



Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0050H	-	-	OS	IF	IE	VCID	VCOE	PD	000X0001B
	R0/WX	R0/WX	R/WX	R(RM1),W	R/W	R/W	R/W	R/W	

PD	Voltage comparator power down control bit
0	Turns on the voltage comparator.
1 ▨	Turns off the voltage comparator.

VCOE	Voltage comparator output enable bit
0 ▨	Disables voltage comparator output.
1	Enables voltage comparator output.

VCID	Voltage comparator analog input disable bit
0 ▨	Enables voltage comparator analog input.
1	Disables voltage comparator analog input.

IE	Interrupt request enable bit
0 ▨	Disables the interrupt request.
1	Enables the interrupt request.

IF	Output edge detection interrupt flag bit	
	Read	Write
0	No output rising/falling edge has occurred.	Clears this bit.
1	An output rising/falling edge has occurred.	Has no effect on operation.

OS	Output status bit
0 ▨	The voltage comparator outputs “L”.
1	The voltage comparator outputs “H”.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

R/W : Readable/writable (The read value is the same as the write value.)
R(RM1),W : Readable/writable (The read value is different from the write value. “1” is read by the read-modify-write (RMW) type of instruction.)
R/WX : Read only (Readable. Writing a value to this bit has no effect on operation.)
R0/WX : The read value is always “0”. Writing a value to this bit has no effect on operation.
- : Undefined bit
X : Indeterminate
▨ : Initial value

Table 33.4-1 Functions of Bits in Voltage Comparator Control Register (CMR0)

Bit name		Function
bit7, bit6	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit5	OS: Output status bit	This bit indicates the output status of the voltage comparator. **Reading "0":** Indicates the voltage comparator outputs "H". **Reading "1":** Indicates the voltage comparator outputs "L". Note: This bit will not be updated in stop mode, watch mode or time-base timer mode. When the PD bit is set to "1" (to turn off the voltage comparator), the OS bit will become "0".
bit4	IF: Output edge detection interrupt flag bit	This bit detects the output rising edge and the output falling edge of the voltage comparator. • With the voltage comparator in operation, this bit will be set to "1" if an output rising edge or an output falling edge occurs. • When read by a read-modify-write (RMW) instruction, this bit always returns "1". **Writing "0":** Clears this bit. **Writing "1":** Has no effect on operation. Note: This bit will not be updated in stop mode, watch mode or time-base timer mode.
bit3	IE: Interrupt request enable bit	This bit enables or disables the interrupt request of the voltage comparator. **Writing "0":** Disables the interrupt request of the voltage comparator. **Writing "1":** Enables the interrupt request of the voltage comparator. With the interrupt request enabled, the voltage comparator will generate an interrupt request when detecting an output rising edge or an output falling edge.
bit2	VCID: Voltage comparator analog input disable bit	This bit is used to enable or disable voltage comparator analog input. **Writing "0":** Enables voltage comparator analog input. **Writing "1":** Disables voltage comparator analog input.
bit1	VCOE: Voltage comparator output enable bit	This bit is used to enable or disable voltage comparator output. **Writing "0":** Disables voltage comparator output. The output pins of the voltage comparator will be used as general-purpose I/O ports. **Writing "1":** Enables voltage comparator output.
bit0	PD: Voltage comparator power down control bit	This bit is used to turn on or off the voltage comparator. **Writing "0":** Turns on the voltage comparator. **Writing "1":** Turns off the voltage comparator.

33.5 Interrupts of Voltage Comparator

The voltage comparator generates an interrupt called output edge detection interrupt. An interrupt request number and an interrupt vector are assigned to the interrupt.

■ Output Edge Detection Interrupt

Table 33.5-1 shows details of the output edge detection interrupt.

Table 33.5-1 Details of Output Edge Detection Interrupt

Item	Details
Interrupt generating condition	An output rising edge or output falling edge occurs.
Interrupt flag	CMR0:IF
Interrupt enable bit	CMR0:IE

Note:

In stop mode, watch mode or time-base timer mode, the edge detection circuit stops operating, and the output edge detection interrupt flag bit (IF) in the voltage comparator control register (CMR0) is not updated even if the voltage comparator has been turned on.

■ Register and Vector Table Addresses Related to Interrupts of Voltage Comparator

Table 33.5-2 Register and Vector Table Addresses Related to Interrupts of Voltage Comparator

Interrupt source	Interrupt request no.	Interrupt level setting register		Vector table address	
		Register	Setting bit	Upper	Lower
Voltage comparator	IRQ15	ILR3	L15	$FFDC_H$	$FFDD_H$

See "APPENDIX B Table of Interrupt Sources" for the respective interrupt request numbers and vector table addresses of different peripheral functions.

33.6 Operations of Voltage Comparator

The voltage comparator can be activated by the software according to the settings of the PD bit in the CMR0 register.

■ Software Activation of Voltage Comparator

To activate the voltage comparator using the software, do the settings shown in Figure 33.6-1.

Figure 33.6-1 Settings for Activating Voltage Comparator

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
CMR0	-	-	OS	IF	IE	VCID	VCOE	PD
	×	×	❍	❍	0	0	0	0

❍ : Bit to be used
× : Unused bit
0 : Set to "0"

After the voltage comparator is activated as shown above, it has to wait for the stabilization time to elapse before starting to operate. For details of the stabilization wait time, refer to the data sheet of the MB95410H/470H Series.

Note:

Before activating the voltage comparator, set the IE bit in the CMR0 register to "0" in advance in order to avoid any unexpected interrupt generated due to the voltage comparator being unstable at its startup.

■ Setting Procedure Example

Below is an example of procedure for setting the voltage comparator:

● Initial settings

1) Disable the voltage comparator interrupt request. (CMR0:IE = 0)
2) Activate the voltage comparator according to the settings shown in Figure 33.6-1.
3) Wait until the voltage comparator stabilizes.
4) Clear the interrupt flag bit. (CMR0:IF = 0)
5) Enable the voltage comparator interrupt request. (CMR0:IE = 1), and enable the voltage comparator output (CMR0:VCOE = 1) if necessary.

CHAPTER 34
SYSTEM CONFIGURATION CONTROLLER

This chapter describes the functions and operations of the system configuration controller (called the "controller" in this chapter).

34.1 Overview of System Configuration Register (SYSC)

The controller consists of the SYSC register, which is an 8-bit register used to configure the clock and reset system, and select the reference voltage for voltage comparator.

■ Functions of SYSC

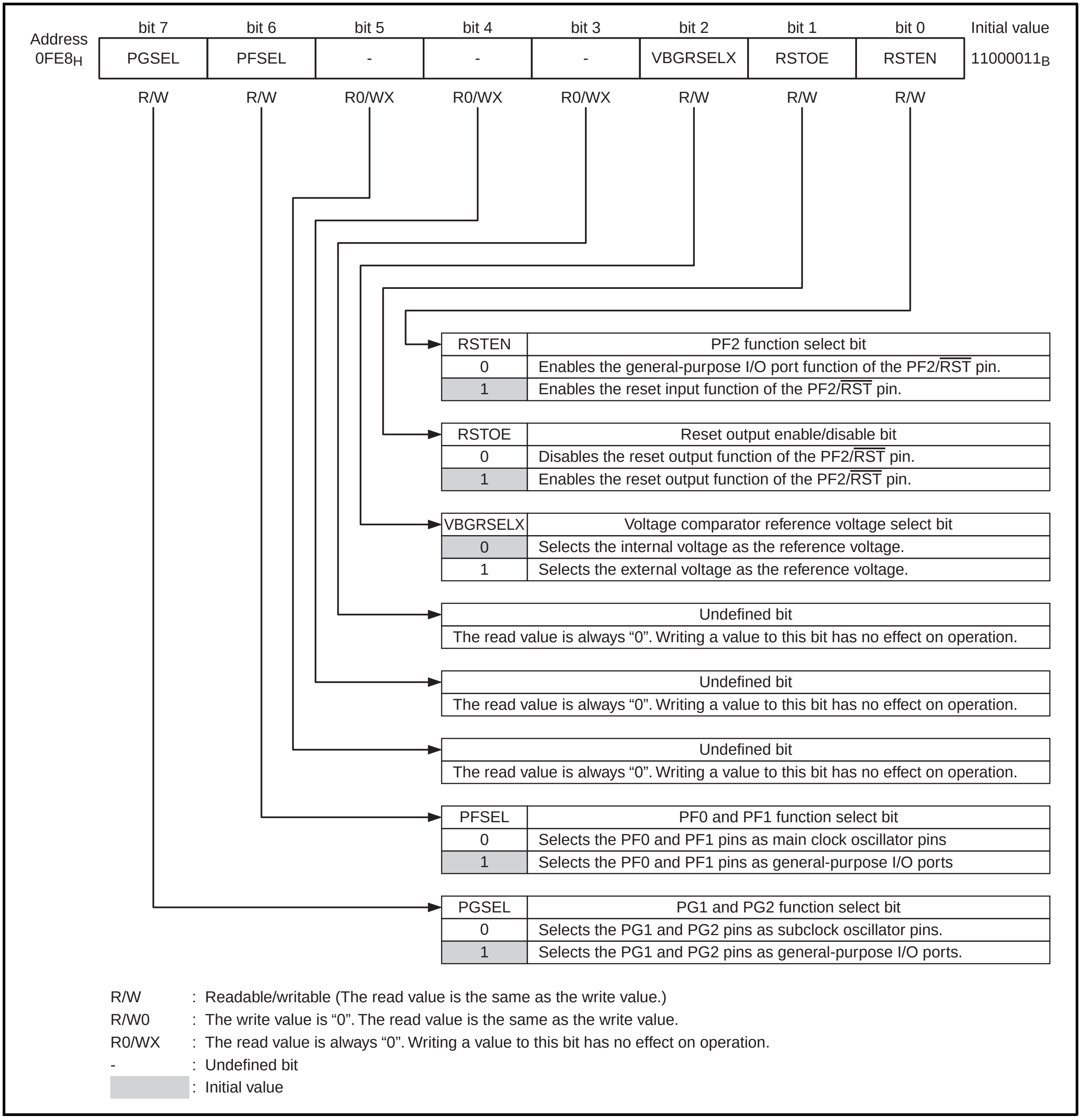
- Selection of the general-purpose I/O port/reset function for the PF2/$\overline{\text{RST}}$ pin
- Enabling/disabling reset output for the $\overline{\text{RST}}$ pin
- Selection of the general-purpose I/O port/oscillation function for the PG1/X0A pin and that for the PG2/X1A pin
- Selection of the general-purpose I/O port/oscillation function for the PF0/X0 pin and that for the PF1/X1 pin
- Selection of the internal or external reference voltage for the voltage comparator

34.2 System Configuration Register (SYSC)

This section provides details of the system configuration register (SYSC).

■ System Configuration Register (SYSC)

Figure 34.2-1 System Configuration Register (SYSC)



Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
0FE8$_H$	PGSEL	PFSEL	-	-	-	VBGRSELX	RSTOE	RSTEN	11000011$_B$
	R/W	R/W	R0/WX	R0/WX	R0/WX	R/W	R/W	R/W	

RSTEN	PF2 function select bit
0	Enables the general-purpose I/O port function of the PF2/$\overline{\text{RST}}$ pin.
▨ 1	Enables the reset input function of the PF2/$\overline{\text{RST}}$ pin.

RSTOE	Reset output enable/disable bit
0	Disables the reset output function of the PF2/$\overline{\text{RST}}$ pin.
▨ 1	Enables the reset output function of the PF2/$\overline{\text{RST}}$ pin.

VBGRSELX	Voltage comparator reference voltage select bit
▨ 0	Selects the internal voltage as the reference voltage.
1	Selects the external voltage as the reference voltage.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

Undefined bit
The read value is always “0”. Writing a value to this bit has no effect on operation.

PFSEL	PF0 and PF1 function select bit
0	Selects the PF0 and PF1 pins as main clock oscillator pins
▨ 1	Selects the PF0 and PF1 pins as general-purpose I/O ports

PGSEL	PG1 and PG2 function select bit
0	Selects the PG1 and PG2 pins as subclock oscillator pins.
▨ 1	Selects the PG1 and PG2 pins as general-purpose I/O ports.

R/W	:	Readable/writable (The read value is the same as the write value.)
R/W0	:	The write value is “0”. The read value is the same as the write value.
R0/WX	:	The read value is always “0”. Writing a value to this bit has no effect on operation.
-	:	Undefined bit
▨	:	Initial value

Table 34.2-1 Functions of Bits in SYSC Register

Bit name		Function
bit7	PGSEL: PG1 and PG2 function select bit	This bit is used to select the function of the PG1 and PG2 pins. If this bit is set to "0", the PG1 and PG2 pins are selected as subclock oscillator pins, and the subclock oscillation is enabled or disabled by the subclock oscillation enable bit (SYCC2:SOSCE). If this bit is set to "1", the PG1 and PG2 pins are selected as general-purpose I/O ports.
bit6	PFSEL: PF0 and PF1 function select bit	This bit is used to select the function of the PF0 and PF1 pins. If this bit is set to "0", the PF0 and PF1 pins are selected as the main clock oscillator pins, and the main clock oscillation is enabled or disabled by the main clock oscillation enable bit (SYCC2:MOSCE). If this bit is set to "1", the PF0 and PF1 pins are selected as the general-purpose I/O port.
bit5 to bit3	Undefined bits	Their read values are always "0". Writing values to these bits has no effect on operation.
bit2	VBGRSELX: Voltage comparator reference voltage select bit	This bit is used to select the reference voltage for the voltage comparator. **Writing "0"**: Selects the internal voltage (bandgap reference voltage) as the reference voltage for the voltage comparator. For details of the bandgap reference voltage, refer to the data sheet of the MB95410H/470H Series. **Writing "1"**: Selects the external voltage from the CMPP pin as the reference voltage for the voltage comparator. For details, see "CHAPTER 33 VOLTAGE COMPARATOR".
bit1	RSTOE: Reset output enable/ disable bit	This bit is used to enable and disable the reset output function of the PF2/$\overline{\text{RST}}$ pin with the reset input function enabled. If the reset input function is disabled according to the setting of SYSC:RSTEN, the reset output function is disabled regardless of the setting of this bit. See the reset input enable/disable bit (SYSC:RSTEN) of this register. If this bit is set to "0", the reset output function of the PF2/$\overline{\text{RST}}$ pin is disabled. If this bit is set to "1", the reset output function of the PF2/$\overline{\text{RST}}$ pin is enabled.
bit0	RSTEN: PF2 function select bit	This bit is used to enable and disable the reset input function of the PF2/$\overline{\text{RST}}$ pin. The reset input function is always enabled in MB95F414H/F416H/F418H/F474H/F476H/F478H regardless of the setting of this bit. If this bit is set to "0", the reset input function of the PF2/$\overline{\text{RST}}$ pin is disabled, and the general-purpose I/O port function is enabled. If this bit is set to "1", the reset input function of the PF2/$\overline{\text{RST}}$ pin is enabled, and the general-purpose I/O port function is disabled. Set bit2 in the PDRF register to "1" before modifying this bit.

Note:

To keep the reset input/output function after the reset, SYSC:RSTEN and SYSC:RSTOE are initialized to "1" after the power is switched on. They will not be initialized by any other type of reset.

If the reset input/output functions have to be used in the system, it is strongly recommended that SYSC:RSTEN be initialized to "1" in the initialize program routine after a reset for stable operation. With the reset input/output functions having been enabled, all types of reset, including the watchdog reset, can be used.

34.3 Notes on Using Controller

This section provides notes on using the controller.

■ Notes on Using Controller

● Selecting the reference voltage for the voltage comparator

Set SYSC:VBGRSELX to "1" when signal input from the CMPP pin is needed. If an internal reference voltage is needed, set SYSC:VBGRSELX to "0" to enable using the internal reference voltage. With SYSC:VBGRSELX set to "0", the external reference voltage input from the CMPP pin cannot be used.

APPENDIX

This section shows the I/O maps, interrupt list, memory map, pin states and mask options.

APPENDIX A I/O Maps

This section shows the I/O maps used in the MB95410H/470H Series.

■ I/O Maps

Table A-1 I/O Map (MB95410H Series) (1 / 6)

Address	Register abbreviation	Register name	R/W	Initial value
0000_H	PDR0	Port 0 data register	R/W	00000000_B
0001_H	DDR0	Port 0 direction register	R/W	00000000_B
0002_H	PDR1	Port 1 data register	R/W	00000000_B
0003_H	DDR1	Port 1 direction register	R/W	00000000_B
0004_H	—	(Disabled)	—	—
0005_H	WATR	Oscillation stabilization wait time setting register	R/W	11111111_B
0006_H	PLLC	PLL control register	R/W	00000000_B
0007_H	SYCC	System clock control register	R/W	$XXXXXX11_B$
0008_H	STBC	Standby control register	R/W	$00000XXX_B$
0009_H	RSRR	Reset source register	R/W	$000XXXXX_B$
$000A_H$	TBTC	Time-base timer control register	R/W	00000000_B
$000B_H$	WPCR	Watch prescaler control register	R/W	00000000_B
$000C_H$	WDTC	Watchdog timer control register	R/W	00000000_B
$000D_H$	SYCC2	System clock control register 2	R/W	$XX100011_B$
$000E_H$	PDR2	Port 2 data register	R/W	00000000_B
$000F_H$	DDR2	Port 2 direction register	R/W	00000000_B
0010_H, 0011_H	—	(Disabled)	—	—
0012_H	PDR4	Port 4 data register	R/W	00000000_B
0013_H	DDR4	Port 4 direction register	R/W	00000000_B
0014_H	PDR5	Port 5 data register	R/W	00000000_B
0015_H	DDR5	Port 5 direction register	R/W	00000000_B
0016_H	PDR6	Port 6 data register	R/W	00000000_B
0017_H	DDR6	Port 6 direction register	R/W	00000000_B
0018_H to $001B_H$	—	(Disabled)	—	—
$001C_H$	PDR9	Port 9 data register	R/W	00000000_B
$001D_H$	DDR9	Port 9 direction register	R/W	00000000_B
$001E_H$	PDRA	Port A data register	R/W	00000000_B
$001F_H$	DDRA	Port A direction register	R/W	00000000_B
0020_H	PDRB	Port B data register	R/W	00000000_B
0021_H	DDRB	Port B direction register	R/W	00000000_B
0022_H	PDRC	Port C data register	R/W	00000000_B

MB95410H/470H Series

Table A-1 I/O Map (MB95410H Series) (2 / 6)

Address	Register abbreviation	Register name	R/W	Initial value
0023_H	DDRC	Port C direction register	R/W	00000000_B
0024_H, 0025_H	—	(Disabled)	—	—
0026_H	PDRE	Port E data register	R/W	00000000_B
0027_H	DDRE	Port E direction register	R/W	00000000_B
0028_H	PDRF	Port F data register	R/W	00000000_B
0029_H	DDRF	Port F direction register	R/W	00000000_B
$002A_H$	PDRG	Port G data register	R/W	00000000_B
$002B_H$	DDRG	Port G direction register	R/W	00000000_B
$002C_H$	—	(Disabled)	—	—
$002D_H$	PUL1	Port 1 pull-up register	R/W	00000000_B
$002E_H$	PUL2	Port 2 pull-up register	R/W	00000000_B
$002F_H$, 0030_H	—	(Disabled)	—	—
0031_H	PUL5	Port 5 pull-up register	R/W	00000000_B
0032_H to 0034_H	—	(Disabled)	—	—
0035_H	PULG	Port G pull-up register	R/W	00000000_B
0036_H	T01CR1	8/16-bit composite timer 01 status control register 1	R/W	00000000_B
0037_H	T00CR1	8/16-bit composite timer 00 status control register 1	R/W	00000000_B
0038_H	T11CR1	8/16-bit composite timer 11 status control register 1	R/W	00000000_B
0039_H	T10CR1	8/16-bit composite timer 10 status control register 1	R/W	00000000_B
$003A_H$	PC01	8/16-bit PPG timer 01 control register	R/W	00000000_B
$003B_H$	PC00	8/16-bit PPG timer 00 control register	R/W	00000000_B
$003C_H$	PC11	8/16-bit PPG timer 11 control register	R/W	00000000_B
$003D_H$	PC10	8/16-bit PPG timer 10 control register	R/W	00000000_B
$003E_H$	TMCSRH0	16-bit reload timer control status register upper	R/W	00000000_B
$003F_H$	TMCSRL0	16-bit reload timer control status register lower	R/W	00000000_B
0040_H to 0047_H	—	(Disabled)	—	—
0048_H	EIC00	External interrupt circuit control register ch. 0/ch. 1	R/W	00000000_B
0049_H	EIC10	External interrupt circuit control register ch. 2/ch. 3	R/W	00000000_B
$004A_H$	EIC20	External interrupt circuit control register ch. 4/ch. 5	R/W	00000000_B
$004B_H$	EIC30	External interrupt circuit control register ch. 6/ch. 7	R/W	00000000_B
$004C_H$ to $004E_H$	—	(Disabled)	—	—
$004F_H$	LCDCC2	LCDC control register 2	R/W	00010100_B
0050_H	CMR0	Voltage comparator control register	R/W	$000X0001_B$

Table A-1 I/O Map (MB95410H Series) (3 / 6)

Address	Register abbreviation	Register name	R/W	Initial value
0051$_H$ to 0055$_H$	—	(Disabled)	—	—
0056$_H$	SMC10	UART/SIO serial mode control register 1 ch. 0	R/W	00000000$_B$
0057$_H$	SMC20	UART/SIO serial mode control register 2 ch. 0	R/W	00100000$_B$
0058$_H$	SSR0	UART/SIO serial status register ch. 0	R/W	00000001$_B$
0059$_H$	TDR0	UART/SIO serial output data register ch. 0	R/W	00000000$_B$
005A$_H$	RDR0	UART/SIO serial input data register ch. 0	R	00000000$_B$
005B$_H$	SMC11	UART/SIO serial mode control register 1 ch. 1	R/W	00000000$_B$
005C$_H$	SMC21	UART/SIO serial mode control register 2 ch. 1	R/W	00100000$_B$
005D$_H$	SSR1	UART/SIO serial status register ch. 1	R/W	00000001$_B$
005E$_H$	TDR1	UART/SIO serial output data register ch. 1	R/W	00000000$_B$
005F$_H$	RDR1	UART/SIO serial input data register ch. 1	R	00000000$_B$
0060$_H$	IBCR00	I^{2}C bus control register 0	R/W	00000001$_B$
0061$_H$	IBCR10	I^{2}C bus control register 1	R/W	00000000$_B$
0062$_H$	IBCR0	I^{2}C bus status register	R	00000000$_B$
0063$_H$	IDDR0	I^{2}C data register	R/W	00000000$_B$
0064$_H$	IAAR0	I^{2}C address register	R/W	00000000$_B$
0065$_H$	ICCR0	I^{2}C clock control register	R/W	00000000$_B$
0066$_H$	SMC12	UART/SIO serial mode control register 1 ch. 2	R/W	00000000$_B$
0067$_H$	SMC22	UART/SIO serial mode control register 2 ch. 2	R/W	00100000$_B$
0068$_H$	SSR2	UART/SIO serial status register ch. 2	R/W	00000001$_B$
0069$_H$	TDR2	UART/SIO serial output data register ch. 2	R/W	00000000$_B$
006A$_H$	RDR2	UART/SIO serial input data register ch. 2	R	00000000$_B$
006B$_H$	—	(Disabled)	—	—
006C$_H$	ADC1	8/10-bit A/D converter control register 1	R/W	00000000$_B$
006D$_H$	ADC2	8/10-bit A/D converter control register 2	R/W	00000000$_B$
006E$_H$	ADDH	8/10-bit A/D converter data register upper	R/W	00000000$_B$
006F$_H$	ADDL	8/10-bit A/D converter data register lower	R/W	00000000$_B$
0070$_H$	WCSR	Watch counter status register	R/W	00000000$_B$
0071$_H$	FSR2	Flash memory status register 2	R/W	00000000$_B$
0072$_H$	FSR	Flash memory status register	R/W	000X0000$_B$
0073$_H$	SWRE0	Flash memory sector write control register 0	R/W	00000000$_B$
0074$_H$	FSR3	Flash memory status register 3	R	00000000$_B$
0075$_H$	—	(Disabled)	—	—
0076$_H$	WREN	Wild register address compare enable register	R/W	00000000$_B$
0077$_H$	WROR	Wild register data test setting register	R/W	00000000$_B$
0078$_H$	—	Mirror of register bank pointer (RP) and direct bank pointer (DP)	—	—
0079$_H$	ILR0	Interrupt level setting register 0	R/W	11111111$_B$

Table A-1 I/O Map (MB95410H Series) (4 / 6)

Address	Register abbreviation	Register name	R/W	Initial value
$007A_H$	ILR1	Interrupt level setting register 1	R/W	11111111_B
$007B_H$	ILR2	Interrupt level setting register 2	R/W	11111111_B
$007C_H$	ILR3	Interrupt level setting register 3	R/W	11111111_B
$007D_H$	ILR4	Interrupt level setting register 4	R/W	11111111_B
$007E_H$	ILR5	Interrupt level setting register 5	R/W	11111111_B
$007F_H$	—	(Disabled)	—	—
$0F80_H$	WRARH0	Wild register address setting register (upper) ch. 0	R/W	00000000_B
$0F81_H$	WRARL0	Wild register address setting register (lower) ch. 0	R/W	00000000_B
$0F82_H$	WRDR0	Wild register data setting register ch. 0	R/W	00000000_B
$0F83_H$	WRARH1	Wild register address setting register (upper) ch. 1	R/W	00000000_B
$0F84_H$	WRARL1	Wild register address setting register (lower) ch. 1	R/W	00000000_B
$0F85_H$	WRDR1	Wild register data setting register ch. 1	R/W	00000000_B
$0F86_H$	WRARH2	Wild register address setting register (upper) ch. 2	R/W	00000000_B
$0F87_H$	WRARL2	Wild register address setting register (lower) ch. 2	R/W	00000000_B
$0F88_H$	WRDR2	Wild register data setting register ch. 2	R/W	00000000_B
$0F89_H$ to $0F91_H$	—	(Disabled)	—	—
$0F92_H$	T01CR0	8/16-bit composite timer 01 status control register 0	R/W	00000000_B
$0F93_H$	T00CR0	8/16-bit composite timer 00 status control register 0	R/W	00000000_B
$0F94_H$	T01DR	8/16-bit composite timer 01 data register	R/W	00000000_B
$0F95_H$	T00DR	8/16-bit composite timer 00 data register	R/W	00000000_B
$0F96_H$	TMCR0	8/16-bit composite timer 00/01 timer mode control register	R/W	00000000_B
$0F97_H$	T11CR0	8/16-bit composite timer 11 status control register 0	R/W	00000000_B
$0F98_H$	T10CR0	8/16-bit composite timer 10 status control register 0	R/W	00000000_B
$0F99_H$	T11DR	8/16-bit composite timer 11 data register	R/W	00000000_B
$0F9A_H$	T10DR	8/16-bit composite timer 10 data register	R/W	00000000_B
$0F9B_H$	TMCR1	8/16-bit composite timer 10/11 timer mode control register	R/W	00000000_B
$0F9C_H$	PPS01	8/16-bit PPG01 cycle setting buffer register	R/W	11111111_B
$0F9D_H$	PPS00	8/16-bit PPG00 cycle setting buffer register	R/W	11111111_B
$0F9E_H$	PDS01	8/16-bit PPG01 duty setting buffer register	R/W	11111111_B
$0F9F_H$	PDS00	8/16-bit PPG00 duty setting buffer register	R/W	11111111_B
$0FA0_H$	PPS11	8/16-bit PPG11 cycle setting buffer register	R/W	11111111_B
$0FA1_H$	PPS10	8/16-bit PPG10 cycle setting buffer register	R/W	11111111_B
$0FA2_H$	PDS11	8/16-bit PPG11 duty setting buffer register	R/W	11111111_B
$0FA3_H$	PDS10	8/16-bit PPG10 duty setting buffer register	R/W	11111111_B
$0FA4_H$	PPGS	8/16-bit PPG start register	R/W	00000000_B
$0FA5_H$	REVC	8/16-bit PPG output inversion register	R/W	00000000_B
$0FA6_H$	TMRH0	16-bit reload timer timer register upper	R/W	00000000_B
	TMRLRH0	16-bit reload timer reload register upper	R/W	00000000_B

Table A-1 I/O Map (MB95410H Series) (5 / 6)

Address	Register abbreviation	Register name	R/W	Initial value
0FA7$_H$	TMRL0	16-bit reload timer timer register lower	R/W	00000000$_B$
	TMRLRL0	16-bit reload timer reload register lower	R/W	00000000$_B$
0FA8$_H$	PSSR0	UART/SIO dedicated baud rate generator prescaler select register ch. 0	R/W	00000000$_B$
0FA9$_H$	BRSR0	UART/SIO dedicated baud rate generator baud rate setting register ch. 0	R/W	00000000$_B$
0FAA$_H$	PSSR1	UART/SIO dedicated baud rate generator prescaler select register ch. 1	R/W	00000000$_B$
0FAB$_H$	BRSR1	UART/SIO dedicated baud rate generator baud rate setting register ch. 1	R/W	00000000$_B$
0FAC$_H$	PSSR2	UART/SIO dedicated baud rate generator prescaler select register ch. 2	R/W	00000000$_B$
0FAD$_H$	BRSR2	UART/SIO dedicated baud rate generator baud rate setting register ch. 2	R/W	00000000$_B$
0FAE$_H$	—	(Disabled)	—	—
0FAF$_H$	AIDRL	A/D input disable register (lower)	R/W	00000000$_B$
0FB0$_H$	LCDCC1	LCDC control register 1	R/W	00000000$_B$
0FB1$_H$	—	(Disabled)	—	—
0FB2$_H$	LCDCE1	LCDC enable register 1	R/W	00111110$_B$
0FB3$_H$	LCDCE2	LCDC enable register 2	R/W	00000000$_B$
0FB4$_H$	LCDCE3	LCDC enable register 3	R/W	00000000$_B$
0FB5$_H$	LCDCE4	LCDC enable register 4	R/W	00000000$_B$
0FB6$_H$	LCDCE5	LCDC enable register 5	R/W	00000000$_B$
0FB7$_H$	LCDCE6	LCDC enable register 6	R/W	00000000$_B$
0FB8$_H$	LCDCE7	LCDC enable register 7	R/W	00000000$_B$
0FB9$_H$	LCDCB1	LCDC blinking setting register 1	R/W	00000000$_B$
0FBA$_H$	LCDCB2	LCDC blinking setting register 2	R/W	00000000$_B$
0FBB$_H$, 0FBC$_H$	—	(Disabled)	—	—
0FBD$_H$ to 0FE0$_H$	LCDRAM	LCDC display RAM (36 bytes)	R/W	00000000$_B$
0FE1$_H$	—	(Disabled)	—	—
0FE2$_H$	EVCR	Event counter control register	R/W	00000000$_B$
0FE3$_H$	WCDR	Watch counter data register	R/W	00111111$_B$
0FE4$_H$	CRTH	Main CR clock trimming register (upper)	R/W	0XXXXXXX$_B$
0FE5$_H$	CRTL	Main CR clock trimming register (lower)	R/W	00XXXXXX$_B$
0FE6$_H$, 0FE7$_H$	—	(Disabled)	—	—
0FE8$_H$	SYSC	System configuration register	R/W	11000011$_B$
0FE9$_H$	CMCR	Clock monitoring control register	R/W	XX000000$_B$
0FEA$_H$	CMDR	Clock monitoring data register	R	00000000$_B$
0FEB$_H$	WDTH	Watchdog timer selection ID register (upper)	R	XXXXXXXX$_B$

MB95410H/470H Series

Table A-1 I/O Map (MB95410H Series) (6 / 6)

Address	Register abbreviation	Register name	R/W	Initial value
0FEC$_H$	WDTL	Watchdog timer selection ID register (lower)	R	XXXXXXXX$_B$
0FED$_H$	—	(Disabled)	—	—
0FEE$_H$	ILSR	Input level select register	R/W	00000000$_B$
0FEF$_H$	WICR	Interrupt pin selection circuit control register	R/W	01000000$_B$
0FF0$_H$ to 0FFF$_H$	—	(Disabled)	—	—

- **R/W access symbols**

R/W	: Readable / Writable
R	: Read only
W	: Write only

- **Initial value symbols**

0	: The initial value of this bit is "0".
1	: The initial value of this bit is "1".
X	: The initial value of this bit is indeterminate.

Note:

Do not write to an address that is "(Disabled)". If a "(Disabled)" address is read, an undefined value is returned.

Table A-2 I/O Map (MB95470H Series) (1 / 5)

Address	Register abbreviation	Register name	R/W	Initial value
0000_H	PDR0	Port 0 data register	R/W	00000000_B
0001_H	DDR0	Port 0 direction register	R/W	00000000_B
0002_H	PDR1	Port 1 data register	R/W	00000000_B
0003_H	DDR1	Port 1 direction register	R/W	00000000_B
0004_H	—	(Disabled)	—	—
0005_H	WATR	Oscillation stabilization wait time setting register	R/W	11111111_B
0006_H	PLLC	PLL control register	R/W	00000000_B
0007_H	SYCC	System clock control register	R/W	$XXXXXX11_B$
0008_H	STBC	Standby control register	R/W	$00000XXX_B$
0009_H	RSRR	Reset source register	R/W	$000XXXXX_B$
$000A_H$	TBTC	Time-base timer control register	R/W	00000000_B
$000B_H$	WPCR	Watch prescaler control register	R/W	00000000_B
$000C_H$	WDTC	Watchdog timer control register	R/W	00000000_B
$000D_H$	SYCC2	System clock control register 2	R/W	$XX100011_B$
$000E_H$	PDR2	Port 2 data register	R/W	00000000_B
$000F_H$	DDR2	Port 2 direction register	R/W	00000000_B
0010_H to 0015_H	—	(Disabled)	—	—
0016_H	PDR6	Port 6 data register	R/W	00000000_B
0017_H	DDR6	Port 6 direction register	R/W	00000000_B
0018_H to $001B_H$	—	(Disabled)	—	—
$001C_H$	PDR9	Port 9 data register	R/W	00000000_B
$001D_H$	DDR9	Port 9 direction register	R/W	00000000_B
$001E_H$	PDRA	Port A data register	R/W	00000000_B
$001F_H$	DDRA	Port A direction register	R/W	00000000_B
0020_H	PDRB	Port B data register	R/W	00000000_B
0021_H	DDRB	Port B direction register	R/W	00000000_B
0022_H	PDRC	Port C data register	R/W	00000000_B
0023_H	DDRC	Port C direction register	R/W	00000000_B
0024_H, 0025_H	—	(Disabled)	—	—
0026_H	PDRE	Port E data register	R/W	00000000_B
0027_H	DDRE	Port E direction register	R/W	00000000_B
0028_H	PDRF	Port F data register	R/W	00000000_B
0029_H	DDRF	Port F direction register	R/W	00000000_B
$002A_H$	PDRG	Port G data register	R/W	00000000_B
$002B_H$	DDRG	Port G direction register	R/W	00000000_B
$002C_H$	—	(Disabled)	—	—

Table A-2 I/O Map (MB95470H Series) (2 / 5)

Address	Register abbreviation	Register name	R/W	Initial value
$002D_H$	PUL1	Port 1 pull-up register	R/W	00000000_B
$002E_H$	PUL2	Port 2 pull-up register	R/W	00000000_B
$002F_H$ to 0034_H	—	(Disabled)	—	—
0035_H	PULG	Port G pull-up register	R/W	00000000_B
0036_H	T01CR1	8/16-bit composite timer 01 status control register 1	R/W	00000000_B
0037_H	T00CR1	8/16-bit composite timer 00 status control register 1	R/W	00000000_B
0038_H	T11CR1	8/16-bit composite timer 11 status control register 1	R/W	00000000_B
0039_H	T10CR1	8/16-bit composite timer 10 status control register 1	R/W	00000000_B
$003A_H$	PC01	8/16-bit PPG01 control register	R/W	00000000_B
$003B_H$	PC00	8/16-bit PPG00 control register	R/W	00000000_B
$003C_H$	PC11	8/16-bit PPG11 control register	R/W	00000000_B
$003D_H$	PC10	8/16-bit PPG10 control register	R/W	00000000_B
$003E_H$	TMCSRH0	16-bit reload timer control status register upper	R/W	00000000_B
$003F_H$	TMCSRL0	16-bit reload timer control status register lower	R/W	00000000_B
0040_H to 0047_H	—	(Disabled)	—	—
0048_H	EIC00	External interrupt circuit control register ch. 0/ch. 1	R/W	00000000_B
0049_H	EIC10	External interrupt circuit control register ch. 2/ch. 3	R/W	00000000_B
$004A_H$	EIC20	External interrupt circuit control register ch. 4/ch. 5	R/W	00000000_B
$004B_H$	EIC30	External interrupt circuit control register ch. 6/ch. 7	R/W	00000000_B
$004C_H$ to $004E_H$	—	(Disabled)	—	—
$004F_H$	LCDCC2	LCDC control register 2	R/W	00010100_B
0050_H	CMR0	Voltage comparator control register	R/W	$000X0001_B$
0051_H to 0055_H	—	(Disabled)	—	—
0056_H	SMC10	UART/SIO serial mode control register 1 ch. 0	R/W	00000000_B
0057_H	SMC20	UART/SIO serial mode control register 2 ch. 0	R/W	00100000_B
0058_H	SSR0	UART/SIO serial status register ch. 0	R/W	00000001_B
0059_H	TDR0	UART/SIO serial output data register ch. 0	R/W	00000000_B
$005A_H$	RDR0	UART/SIO serial input data register ch. 0	R	00000000_B
$005B_H$	SMC11	UART/SIO serial mode control register 1 ch. 1	R/W	00000000_B
$005C_H$	SMC21	UART/SIO serial mode control register 2 ch. 1	R/W	00100000_B
$005D_H$	SSR1	UART/SIO serial status register ch. 1	R/W	00000001_B
$005E_H$	TDR1	UART/SIO serial output data register ch. 1	R/W	00000000_B
$005F_H$	RDR1	UART/SIO serial input data register ch. 1	R	00000000_B
0060_H	IBCR00	I^2C bus control register 0	R/W	00000001_B

Table A-2 I/O Map (MB95470H Series) (3 / 5)

Address	Register abbreviation	Register name	R/W	Initial value
0061_H	IBCR10	I^2C bus control register 1	R/W	00000000_B
0062_H	IBCR0	I^2C bus status register	R	00000000_B
0063_H	IDDR0	I^2C data register	R/W	00000000_B
0064_H	IAAR0	I^2C address register	R/W	00000000_B
0065_H	ICCR0	I^2C clock control register	R/W	00000000_B
0066_H	SMC12	UART/SIO serial mode control register 1 ch. 2	R/W	00000000_B
0067_H	SMC22	UART/SIO serial mode control register 2 ch. 2	R/W	00100000_B
0068_H	SSR2	UART/SIO serial status register ch. 2	R/W	00000001_B
0069_H	TDR2	UART/SIO serial output data register ch. 2	R/W	00000000_B
$006A_H$	RDR2	UART/SIO serial input data register ch. 2	R	00000000_B
$006B_H$	—	(Disabled)	—	—
$006C_H$	ADC1	8/10-bit A/D converter control register 1	R/W	00000000_B
$006D_H$	ADC2	8/10-bit A/D converter control register 2	R/W	00000000_B
$006E_H$	ADDH	8/10-bit A/D converter data register upper	R/W	00000000_B
$006F_H$	ADDL	8/10-bit A/D converter data register lower	R/W	00000000_B
0070_H	WCSR	Watch counter status register	R/W	00000000_B
0071_H	FSR2	Flash memory status register 2	R/W	00000000_B
0072_H	FSR	Flash memory status register	R/W	$000X0000_B$
0073_H	SWRE0	Flash memory sector write control register 0	R/W	00000000_B
0074_H	FSR3	Flash memory status register 3	R	00000000_B
0075_H	—	(Disabled)	—	—
0076_H	WREN	Wild register address compare enable register	R/W	00000000_B
0077_H	WROR	Wild register data test setting register	R/W	00000000_B
0078_H	—	Mirror of register bank pointer (RP) and direct bank pointer (DP)	—	—
0079_H	ILR0	Interrupt level setting register 0	R/W	11111111_B
$007A_H$	ILR1	Interrupt level setting register 1	R/W	11111111_B
$007B_H$	ILR2	Interrupt level setting register 2	R/W	11111111_B
$007C_H$	ILR3	Interrupt level setting register 3	R/W	11111111_B
$007D_H$	ILR4	Interrupt level setting register 4	R/W	11111111_B
$007E_H$	ILR5	Interrupt level setting register 5	R/W	11111111_B
$007F_H$	—	(Disabled)	—	—
$0F80_H$	WRARH0	Wild register address setting register (upper) ch. 0	R/W	00000000_B
$0F81_H$	WRARL0	Wild register address setting register (lower) ch. 0	R/W	00000000_B
$0F82_H$	WRDR0	Wild register data setting register ch. 0	R/W	00000000_B
$0F83_H$	WRARH1	Wild register address setting register (upper) ch. 1	R/W	00000000_B
$0F84_H$	WRARL1	Wild register address setting register (lower) ch. 1	R/W	00000000_B
$0F85_H$	WRDR1	Wild register data setting register ch. 1	R/W	00000000_B
$0F86_H$	WRARH2	Wild register address setting register (upper) ch. 2	R/W	00000000_B
$0F87_H$	WRARL2	Wild register address setting register (lower) ch. 2	R/W	00000000_B

Table A-2 I/O Map (MB95470H Series) (4 / 5)

Address	Register abbreviation	Register name	R/W	Initial value
$0F88_H$	WRDR2	Wild register data setting register ch. 2	R/W	00000000_B
$0F89_H$ to $0F91_H$	—	(Disabled)	—	—
$0F92_H$	T01CR0	8/16-bit composite timer 01 status control register 0	R/W	00000000_B
$0F93_H$	T00CR0	8/16-bit composite timer 00 status control register 0	R/W	00000000_B
$0F94_H$	T01DR	8/16-bit composite timer 01 data register	R/W	00000000_B
$0F95_H$	T00DR	8/16-bit composite timer 00 data register	R/W	00000000_B
$0F96_H$	TMCR0	8/16-bit composite timer 00/01 timer mode control register	R/W	00000000_B
$0F97_H$	T11CR0	8/16-bit composite timer 11 status control register 0	R/W	00000000_B
$0F98_H$	T10CR0	8/16-bit composite timer 10 status control register 0	R/W	00000000_B
$0F99_H$	T11DR	8/16-bit composite timer 11 data register	R/W	00000000_B
$0F9A_H$	T10DR	8/16-bit composite timer 10 data register	R/W	00000000_B
$0F9B_H$	TMCR1	8/16-bit composite timer 10/11 timer mode control register	R/W	00000000_B
$0F9C_H$	PPS01	8/16-bit PPG01 cycle setting buffer register	R/W	11111111_B
$0F9D_H$	PPS00	8/16-bit PPG00 cycle setting buffer register	R/W	11111111_B
$0F9E_H$	PDS01	8/16-bit PPG01 duty setting buffer register	R/W	11111111_B
$0F9F_H$	PDS00	8/16-bit PPG00 duty setting buffer register	R/W	11111111_B
$0FA0_H$	PPS11	8/16-bit PPG11 cycle setting buffer register	R/W	11111111_B
$0FA1_H$	PPS10	8/16-bit PPG10 cycle setting buffer register	R/W	11111111_B
$0FA2_H$	PDS11	8/16-bit PPG11 duty setting buffer register	R/W	11111111_B
$0FA3_H$	PDS10	8/16-bit PPG10 duty setting buffer register	R/W	11111111_B
$0FA4_H$	PPGS	8/16-bit PPG start register	R/W	00000000_B
$0FA5_H$	REVC	8/16-bit PPG output inversion register	R/W	00000000_B
$0FA6_H$	TMRH0	16-bit reload timer timer register upper	R/W	00000000_B
	TMRLRH0	16-bit reload timer reload register upper	R/W	00000000_B
$0FA7_H$	TMRL0	16-bit reload timer timer register lower	R/W	00000000_B
	TMRLRL0	16-bit reload timer reload register lower	R/W	00000000_B
$0FA8_H$	PSSR0	UART/SIO dedicated baud rate generator prescaler select register ch. 0	R/W	00000000_B
$0FA9_H$	BRSR0	UART/SIO dedicated baud rate generator baud rate setting register ch. 0	R/W	00000000_B
$0FAA_H$	PSSR1	UART/SIO dedicated baud rate generator prescaler select register ch. 1	R/W	00000000_B
$0FAB_H$	BRSR1	UART/SIO dedicated baud rate generator baud rate setting register ch. 1	R/W	00000000_B
$0FAC_H$	PSSR2	UART/SIO dedicated baud rate generator prescaler select register ch. 2	R/W	00000000_B
$0FAD_H$	BRSR2	UART/SIO dedicated baud rate generator baud rate setting register ch. 2	R/W	00000000_B
$0FAE_H$	—	(Disabled)	—	—
$0FAF_H$	AIDRL	A/D input disable register (lower)	R/W	00000000_B

Table A-2 I/O Map (MB95470H Series) (5 / 5)

Address	Register abbreviation	Register name	R/W	Initial value
$0FB0_H$	LCDCC1	LCDC control register 1	R/W	00000000_B
$0FB1_H$	—	(Disabled)	—	—
$0FB2_H$	LCDCE1	LCDC enable register 1	R/W	00111100_B
$0FB3_H$	LCDCE2	LCDC enable register 2	R/W	00000000_B
$0FB4_H$	LCDCE3	LCDC enable register 3	R/W	00000000_B
$0FB5_H$	LCDCE4	LCDC enable register 4	R/W	00000000_B
$0FB6_H$	LCDCE5	LCDC enable register 5	R/W	00000000_B
$0FB7_H$	LCDCE6	LCDC enable register 6	R/W	00000000_B
$0FB8_H$	—	(Disabled)	—	—
$0FB9_H$	LCDCB1	LCDC blinking setting register 1	R/W	00000000_B
$0FBA_H$	LCDCB2	LCDC blinking setting register 2	R/W	00000000_B
$0FBB_H$, $0FBC_H$	—	(Disabled)	—	—
$0FBD_H$ to $0FD8_H$	LCDRAM	LCDC display RAM (28 bytes)	R/W	00000000_B
$0FD9_H$ to $0FE1_H$	—	(Disabled)	—	—
$0FE2_H$	EVCR	Event counter control register	R/W	00000000_B
$0FE3_H$	WCDR	Watch counter data register	R/W	00111111_B
$0FE4_H$	CRTH	Main CR clock trimming register (upper)	R/W	$0XXXXXXX_B$
$0FE5_H$	CRTL	Main CR clock trimming register (lower)	R/W	$00XXXXXX_B$
$0FE6_H$, $0FE7_H$	—	(Disabled)	—	—
$0FE8_H$	SYSC	System configuration register	R/W	11000011_B
$0FE9_H$	CMCR	Clock monitoring control register	R/W	$XX000000_B$
$0FEA_H$	CMDR	Clock monitoring data register	R	00000000_B
$0FEB_H$	WDTH	Watchdog timer selection ID register (upper)	R	$XXXXXXXX_B$
$0FEC_H$	WDTL	Watchdog timer selection ID register (lower)	R	$XXXXXXXX_B$
$0FED_H$	—	(Disabled)	—	—
$0FEE_H$	ILSR	Input level select register	R/W	00000000_B
$0FEF_H$	WICR	Interrupt pin selection circuit control register	R/W	01000000_B
$0FF0_H$ to $0FFF_H$	—	(Disabled)	—	—

- **R/W access symbols**

R/W	: Readable / Writable
R	: Read only
W	: Write only

- **Initial value symbols**

0	: The initial value of this bit is "0".
1	: The initial value of this bit is "1".
X	: The initial value of this bit is indeterminate.

Note:

Do not write to an address that is "(Disabled)". If a "(Disabled)" address is read, an undefined value is returned.

APPENDIX B Table of Interrupt Sources

This section shows the table of interrupt sources used in the MB95410H/470H Series.

■ Table of Interrupt Sources

See "CHAPTER 5 CPU" for interrupt operation.

Table B-1 Table of Interrupt Sources

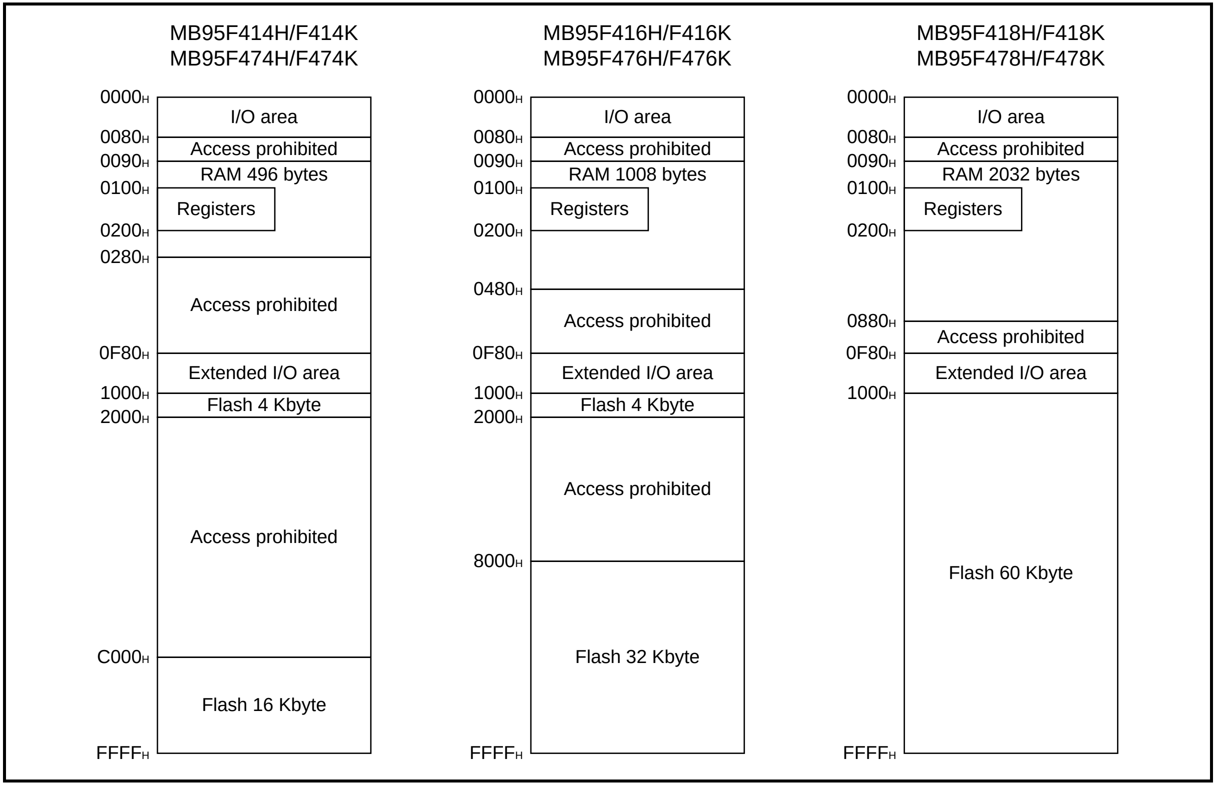
Interrupt source	Interrupt request number	Vector table address		Bit name of interrupt level setting register	Priority order of interrupt sources of the same level (occurring simultaneously)
		Upper	Lower		
External interrupt ch. 0	IRQ00	FFFA$_H$	FFFB$_H$	L00[1:0]	High
External interrupt ch. 4					
External interrupt ch. 1	IRQ01	FFF8$_H$	FFF9$_H$	L01[1:0]	
External interrupt ch. 5					
External interrupt ch. 2	IRQ02	FFF6$_H$	FFF7$_H$	L02[1:0]	
External interrupt ch. 6					
External interrupt ch. 3	IRQ03	FFF4$_H$	FFF5$_H$	L03[1:0]	
External interrupt ch. 7					
UART/SIO ch. 0	IRQ04	FFF2$_H$	FFF3$_H$	L04[1:0]	
8/16-bit composite timer ch. 0 (lower)	IRQ05	FFF0$_H$	FFF1$_H$	L05[1:0]	
8/16-bit composite timer ch. 0 (upper)	IRQ06	FFEE$_H$	FFEF$_H$	L06[1:0]	
UART/SIO ch. 2	IRQ07	FFEC$_H$	FFED$_H$	L07[1:0]	
LCD controller	IRQ08	FFEA$_H$	FFEB$_H$	L08[1:0]	
8/16-bit PPG ch. 1 (lower)	IRQ09	FFE8$_H$	FFE9$_H$	L09[1:0]	
UART/SIO ch. 1					
8/16-bit PPG ch. 1 (upper)	IRQ10	FFE6$_H$	FFE7$_H$	L10[1:0]	
16-bit reload timer ch. 0	IRQ11	FFE4$_H$	FFE5$_H$	L11[1:0]	
8/16-bit PPG ch. 0 (upper)	IRQ12	FFE2$_H$	FFE3$_H$	L12[1:0]	
8/16-bit PPG ch. 0 (lower)	IRQ13	FFE0$_H$	FFE1$_H$	L13[1:0]	
8/16-bit composite timer ch. 1 (upper)	IRQ14	FFDE$_H$	FFDF$_H$	L14[1:0]	
Voltage comparator	IRQ15	FFDC$_H$	FFDD$_H$	L15[1:0]	
I^{2}C	IRQ16	FFDA$_H$	FFDB$_H$	L16[1:0]	
—	IRQ17	FFD8$_H$	FFD9$_H$	L17[1:0]	
8/10-bit A/D converter	IRQ18	FFD6$_H$	FFD7$_H$	L18[1:0]	
Time-base timer	IRQ19	FFD4$_H$	FFD5$_H$	L19[1:0]	
Watch prescaler	IRQ20	FFD2$_H$	FFD3$_H$	L20[1:0]	
Watch counter					
—	IRQ21	FFD0$_H$	FFD1$_H$	L21[1:0]	
8/16-bit composite timer ch. 1 (lower)	IRQ22	FFCE$_H$	FFCF$_H$	L22[1:0]	
Flash memory	IRQ23	FFCC$_H$	FFCD$_H$	L23[1:0]	Low

APPENDIX C Memory Maps

This section shows the memory maps of the MB95410H/470H Series.

■ Memory Maps

Figure C-1 Memory Maps of Different Products



Parameter / Part number	Flash memory	RAM
MB95F414H/F414K/F474H/F474K	20 Kbyte	496 bytes
MB95F416H/F416K/F476H/F476K	36 Kbyte	1008 bytes
MB95F418H/F418K/F478H/F478K	60 Kbyte	2032 bytes

APPENDIX D Pin States of MB95410H/470H Series

Table D-1 below shows the pin states of the MB95410H/470H Series in each mode.

■ Pin States in Each Mode

Table D-1 Pin States in Each Mode (1 / 4)

Pin name	Normal operation	Sleep mode	Stop mode		Watch mode		In reset
			SPL=0	SPL=1	SPL=0	SPL=1	
PF0/X0	Oscillation circuit input	Oscillation circuit input	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Oscillation circuit input[*3]
PF1X1	Oscillation circuit output	Oscillation circuit input	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Oscillation circuit output[*3]
PG1/X0A	Oscillation circuit input	Oscillation circuit input	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Oscillation circuit input[*5]
PG2/X1A	Oscillation circuit output	Oscillation circuit input	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Oscillation circuit output[*5]
PF2/$\overline{\text{RST}}$	Reset input	Reset input	Reset input	Reset input	Reset input	Reset input	Reset input[*4]
P00/INT00/ AN00/UO2/ SEG29[*1]	I/O port/ peripheral function I/O/ analog input	I/O port/ peripheral function I/O/ analog input	- Retain - Input interception[*8]	- Hi-Z - Input interception[*8] (However, an external interrupt can be input when the external interrupt is enabled.)	- Retain - Input interception[*8]	- Hi-Z - Input interception[*8] (However, an external interrupt can be input when the external interrupt is enabled.)	- Hi-Z - Input disabled[*10]
P01/INT01/ AN01/UI2/ SEG36[*1]/ SEG28[*1]/ TO00[*2]							
P02/INT02/ AN02/ UCK2/ SEG35[*1]/ SEG27[*1]							
P03/INT03/ AN03/UO1/ SEG34[*1]/ SEG26[*1]							
P04/INT04/ AN04/UI1/ SEG33[*1]/ SEG25[*1]							
P05/INT05/ AN05/ UCK1/ SEG32[*1]/ SEG24[*1]							

MB95410H/470H Series

Table D-1   Pin States in Each Mode (2 / 4)

Pin name	Normal operation	Sleep mode	Stop mode		Watch mode		In reset
			SPL=0	SPL=1	SPL=0	SPL=1	
P06/INT06/ AN06/ SEG31*1/ SEG23*1	I/O port/ peripheral function I/O/ analog input	I/O port/ peripheral function I/O/ analog input	- Retain - Input interception*8	- Hi-Z - Input interception*8 (However, an external interrupt can be input when the external interrupt is enabled.)	- Retain - Input interception*8	- Hi-Z - Input interception*8 (However, an external interrupt can be input when the external interrupt is enabled.)	- Hi-Z - Input disabled*10
P07/INT07/ AN07/ SEG30*1/ SEG22*1							
P10/UI0/ TO0*6	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Hi-Z - Input enabled*9 (However, it does not function.)
P11/UO0							
P12/DBG	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	"H"	- Retain - Input interception*8	"H"	"H"
P13/ADTG	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Hi-Z - Input enabled*9 (However, it does not function.)
P14/UCK0/ EC0/TI0*6							
P15/PPG11/ SEG31*1	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
P16/PPG10/ SEG30*1							
P17/CMPO	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input enabled*9 (However, it does not function.)
P20/PPG00/ CMPN	I/O port/ peripheral function I/O/ analog input	I/O port/ peripheral function I/O/ analog input	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Hi-Z - Input enabled*9
P21/PPG01/ CMPP							
P22/SCL	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception *8, *11	- Hi-Z - Input interception *8, *11	- Retain - Input interception *8, *11	- Hi-Z - Input interception *8, *11	- Hi-Z - Input enabled*9
P23/SDA							

Table D-1   Pin States in Each Mode (3 / 4)

Pin name	Normal operation	Sleep mode	Stop mode		Watch mode		In reset
			SPL=0	SPL=1	SPL=0	SPL=1	
P40/ SEG21*12	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
P41/ SEG20*12							
P42/ SEG19*12							
P43/ SEG18*12							
P50/ TO01*6, *12	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Retain - Input interception*8	- Hi-Z (However, the setting of the pull-up is effective.) - Input interception*8	- Hi-Z - Input enabled*9 (However, it does not function.)
P51/EC0*12							
P52/TI0/ TO00*12							
P53/TO0*12							
P60/ SEG10*1/ SEG06*1	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
P61/ SEG11*1/ SEG07*1							
P62/ SEG12*1/ SEG08*1							
P63/ SEG13*1/ SEG09*1							
P64/ SEG14*1/ SEG10*1							
P65/ SEG15*1/ SEG11*1							
P66/ SEG16*1/ SEG12*1							
P67/ SEG17*1/ SEG13*1							
P90/V4	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
P91/V3							
P92/V2							
P93/V1							
P94/V0*7							

MB95410H/470H Series

Table D-1 Pin States in Each Mode (4 / 4)

Pin name	Normal operation	Sleep mode	Stop mode		Watch mode		In reset
			SPL=0	SPL=1	SPL=0	SPL=1	
PA0/COM0	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
PA1/COM1							
PA2/COM2							
PA3/COM3							
PA4/COM4							
PA5/COM5							
PA6/COM6							
PA7/COM7							
PB0/SEG00	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
PB1/SEG01							
PB2/ SEG37*12							
PB3/ SEG38*12							
PB4/ SEG39*12							
PC0/SEG02	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
PC1/SEG03							
PC2/SEG04							
PC3/SEG05							
PC4/ SEG06*12							
PC5/ SEG07*12							
PC6/ SEG08*12							
PC7/ SEG09*12							
PE0/SEG22/ SEG14*1	I/O port/ peripheral function I/O	I/O port/ peripheral function I/O	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Retain - Input interception*8	- Hi-Z - Input interception*8	- Hi-Z - Input disabled*10
PE1/SEG23/ SEG15*1							
PE2/SEG24/ SEG16*1							
PE3/SEG25/ SEG17*1							
PE4/SEG26/ SEG18*1							
PE5/SEG27/ SEG19*1							
PE6/SEG28/ SEG20*1							
PE7/SEG29/ SEG21*1							

SPL: Pin state setting bit in the standby control register (STBC:SPL)

Hi-Z: High impedance

*1: The MB95410H Series and the MB95470H Series have different SEG output assignment as shown below.

SEG Output	Pin on MB95410H Series	Pin on MB95470H Series
SEG06	PC4	P60
SEG07	PC5	P61
SEG08	PC6	P62
SEG09	PC7	P63
SEG10	P60	P64
SEG11	P61	P65
SEG12	P62	P66
SEG13	P63	P67
SEG14	P64	PE0
SEG15	P65	PE1
SEG16	P66	PE2
SEG17	P67	PE3
SEG18	P43	PE4
SEG19	P42	PE5
SEG20	P41	PE6
SEG21	P40	PE7
SEG22	PE0	P07
SEG23	PE1	P06
SEG24	PE2	P05
SEG25	PE3	P04
SEG26	PE4	P03
SEG27	PE5	P02
SEG28	PE6	P01
SEG29	PE7	P00
SEG30	P07	P16
SEG31	P06	P15
SEG32	P05	—
SEG33	P04	—
SEG34	P03	—
SEG35	P02	—
SEG36	P01	—

*2: TO00 is assigned to P01 on the MB95470H Series.

*3: PF0/X0 and PF1/X1 will transit to this state on a reset when configured as main oscillation pins.

*4: PF2/$\overline{\text{RST}}$ will transit to this state on a reset when configured as an external reset pin.

*5: PG1/X0A and PG2/X1A will transit to this state on a reset when configured as sub-oscillation pins.

*6: On the MB95470H Series, TO0 is assigned to P10, EC0 and TI0 to P14, and TO01 to P13.

*7: P94/V0 is only available on the MB95410H Series.

*8: "Input interception" means direct input gate operation from the pin is disabled.

*9: "Input enabled" means that the input function is enabled. While the input function is enabled, pull-up or pull-down operation has to be performed in order to prevent leaks due to external input. If a pin is used as an output port, the pin state is the same as that of other ports.

*10: "Input disabled" means direct input gate operation from the pin is disabled.

*11: The I^{2}C interface can wake up the MCU in stop mode or watch mode when its MCU standby mode wakeup function is enabled. For details of the MCU standby mode wakeup function, see "CHAPTER 24 I^{2}C".

*12: P40/SEG21, P41/SEG20, P42/SEG19, P43/SEG18, P50/TO01, P51/EC0, P52/TI0/TO00, P53/TO0, PB2/SEG37, PB3/SEG38, PB4/SEG39, PC4/SEG06, PC5/SEG07, PC6/SEG08 and PC7/SEG09 are only available on the MB95410H Series.

MB95410H/470H Series

APPENDIX E Instruction Overview

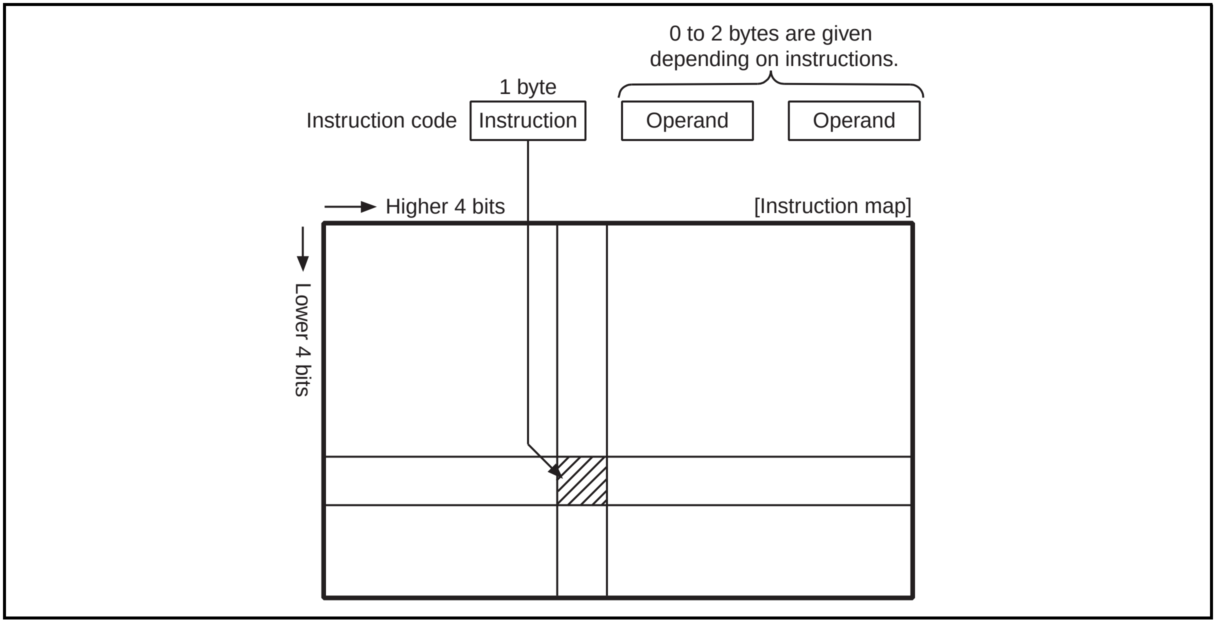
This section explains the instructions used in F^{2}MC-8FX.

■ Instruction Overview of F^{2}MC-8FX

In F^{2}MC-8FX, there are 140 kinds of one byte instructions (as the map, 256 bytes), and the instruction code is composed of the instruction and the operand following it.

Figure E-1 shows the correspondence of the instruction code and the instruction map.

Figure E-1 Instruction Code and Instruction Map



- The instruction is classified into following four types; forwarding system, operation system, branch system and others.
- There are various methods of addressing, and ten kinds of addressing can be selected by the selection and the operand specification of the instruction.
- This provides with the bit operation instruction, and can execute the read-modify-write (RMW) type of instruction.
- There is an instruction that directs special operation.

■ Explanation of Display Sign of Instruction

Table E-1 shows the explanation of the sign used by explaining the instruction code of this APPENDIX E.

Table E-1 Explanation of Sign in Instruction Table

Sign	Signification
dir	Direct address (8-bit length)
off	Offset (8-bit length)
ext	Extended address (16-bit length)
#vct	Vector table number (3-bit length)
#d8	Immediate data (8-bit length)
#d16	Immediate data (16-bit length)
dir:b	Bit direct address (8-bit length: 3-bit length)
rel	Branch relative address (8-bit length)
@	Register indirect (Example: @A, @IX, @EP)
A	Accumulator (Whether 8- bit length or 16- bit length is decided by the instruction used.)
AH	Upper 8-bit of accumulator (8-bit length)
AL	Lower 8-bit of accumulator (8-bit length)
T	Temporary accumulator (Whether 8- bit length or 16- bit length is decided by the instruction used.)
TH	Upper 8-bit of temporary accumulator (8-bit length)
TL	Lower 8-bit of temporary accumulator (8-bit length)
IX	Index register (16-bit length)
EP	Extra pointer (16-bit length)
PC	Program counter (16-bit length)
SP	Stack pointer (16-bit length)
PS	Program status (16-bit length)
dr	Either of accumulator or index register (16-bit length)
CCR	Condition code register (8-bit length)
RP	Register bank pointer (5-bit length)
DP	Direct bank pointer (3-bit length)
Ri	General-purpose register (8-bit length, i = 0 to 7)
x	This shows that x is immediate data. (Whether 8- bit length or 16- bit length is decided by the instruction used.)
(x)	This shows that contents of x are objects of the access. (Whether 8- bit length or 16- bit length is decided by the instruction used.)
((x))	This shows that the address that contents of x show is an object of the access. (Whether 8- bit length or 16- bit length is decided by the instruction used.)

■ Explanation of Item in Instruction Table

Table E-2 Explanation of Item in Instruction Table

Item	Description
MNEMONIC	It shows the assembly description of the instruction.
~	It shows the number of cycles of the instruction. One instruction cycle is a machine cycle. Note: The number of cycles of the instruction can be delayed by 1 cycle by the immediately preceding instruction. Moreover, the number of cycles of the instruction might be extended in the access to the I/O area.
#	It shows the number of bytes for the instruction.
Operation	It shows the operations for the instruction.
TL, TH, AH	They show the change (auto forwarding from A to T) in the content when each TL, TH, and AH instruction is executed. The sign in the column indicates the followings respectively. • -: No change • dH: upper 8 bits of the data described in operation. • AL and AH: the contents become those of the immediately preceding instruction's AL and AH. • 00: Become 00
N, Z, V, C	They show the instruction into which the corresponding flag is changed respectively. The sign in the column shows the followings respectively. • -: No change • +: Change • R: Become "0" • S: Become "1"
OP CODE	It shows the code of the instruction. When a pertinent instruction occupies two or more codes, it follows the following description rules. [Example] 48 to 4F: This shows 48, 49....4F.

E.1 Addressing

F^2MC-8FX has the following ten types of addressings:

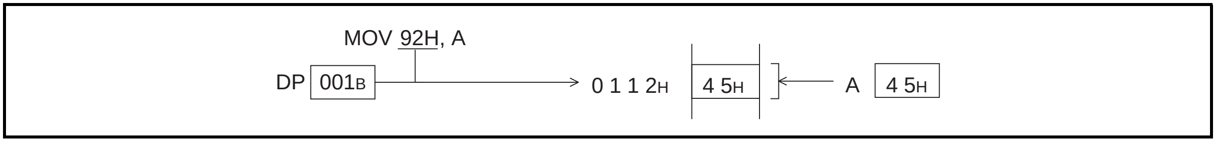
- **Direct addressing**
- **Extended addressing**
- **Bit direct addressing**
- **Index addressing**
- **Pointer addressing**
- **General-purpose register addressing**
- **Immediate addressing**
- **Vector addressing**
- **Relative addressing**
- **Inherent addressing**

■ Explanation of Addressing

● Direct addressing

This is used when accessing the direct area of "0000_H" to "$047F_H$" with addressing indicated "dir" in instruction table. In this addressing, when the operand address is "00_H" to "$7F_H$", it is accessed into "0000_H" to "$007F_H$". Moreover, when the operand address is "80_H" to "FF_H", the access can be mapped in "0080_H" to "$047F_H$" by setting of direct bank pointer DP. Figure E.1-1 shows an example.

Figure E.1-1 Example of Direct Addressing

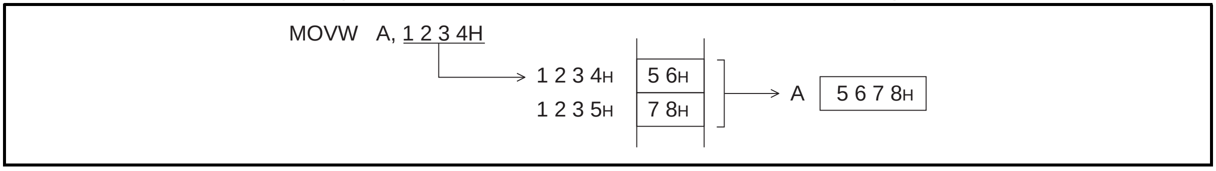


● Extended addressing

This is used when the area of the entire 64 Kbyte is accessed by addressing shown "ext" in the instruction table. In this addressing, the first operand specifies one high rank byte of the address and the second operand specifies one subordinate position byte of the address.

Figure E.1-2 shows an example.

Figure E.1-2 Example of Extended Addressing

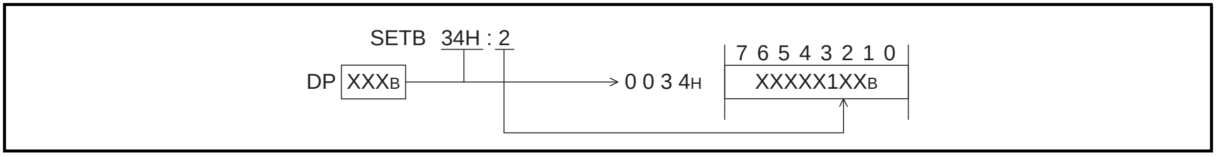


- Bit direct addressing

This is used when accessing the direct area of "0000_H" to "$047F_H$" in bit unit with addressing indicated "dir:b" in instruction table. In this addressing, when the operand address is "00_H" to "$7F_H$", it is accessed into "0000_H" to "$007F_H$". Moreover, when the operand address is "80_H" to "FF_H", the access can be mapped in "0080_H" to "$047F_H$" by setting of direct bank pointer DP. The position of the bit in the specified address is specified by the values of the instruction code of three subordinate position bits.

Figure E.1-3 shows an example.

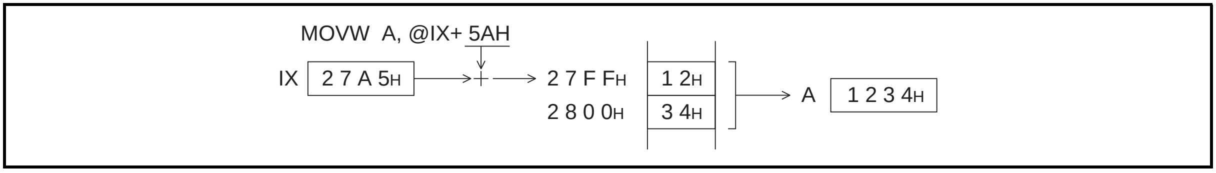
Figure E.1-3 Example of Bit Direct Addressing



- Index addressing

This is used when the area of the entire 64 Kbyte is accessed by addressing shown "@IX+off" in the instruction table. In this addressing, the content of the first operand is sign extended and added to IX (index register) to the resulting address. Figure E.1-4 shows an example.

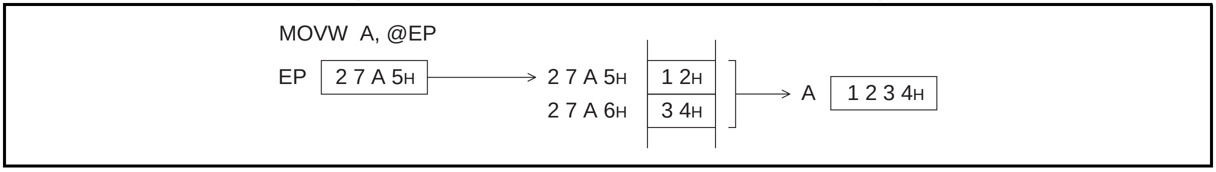
Figure E.1-4 Example of Index Addressing



- Pointer addressing

This is used when the area of the entire 64 Kbyte is accessed by addressing shown "@EP" in the instruction table. In this addressing, the content of EP (extra pointer) is assumed to be an address. Figure E.1-5 shows an example.

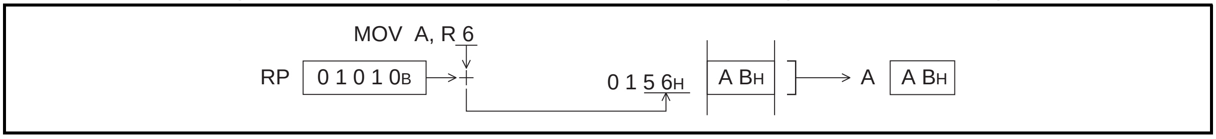
Figure E.1-5 Example of Pointer Addressing



- General-purpose register addressing

This is used when accessing the register bank in general-purpose register area with the addressing shown "Ri" in instruction table. In this addressing, fix one high rank byte of the address to "01" and create one subordinate position byte from the contents of RP (register bank pointer) and three subordinate bits of the operation code to access to this address. Figure E.1-6 shows an example.

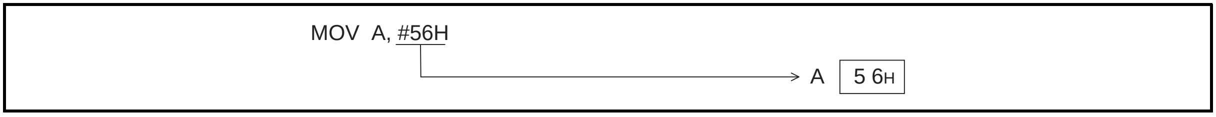
Figure E.1-6 Example of General-purpose Register Addressing



- Immediate addressing

This is used when immediate data is needed in addressing shown "#d8" in the instruction table. In this addressing, the operand becomes immediate data as it is. The specification of byte/word depends on the operation code. Figure E.1-7 shows an example.

Figure E.1-7 Example of Immediate Addressing



- Vector addressing

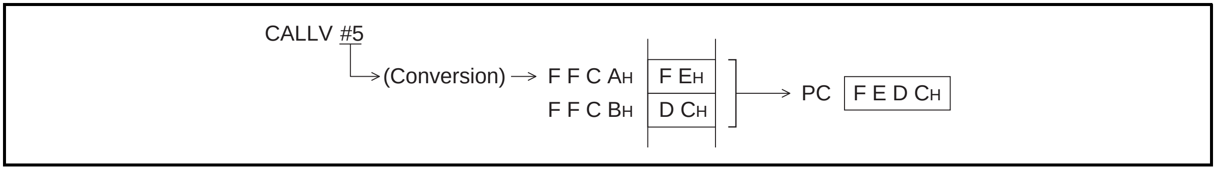
This is used when branching to the subroutine address registered in the table with the addressing shown "#vct" in the instruction table. In this addressing, information on "#vct" is contained in the operation code, and the address of the table is created using the combinations shown in Table E.1-1.

Table E.1-1 Vector Table Address Corresponding to "#vct"

#vct	Vector table address (jump destination high-ranking address: subordinate address)
0	FFC0$_H$: FFC1$_H$
1	FFC2$_H$: FFC3$_H$
2	FFC4$_H$: FFC5$_H$
3	FFC6$_H$: FFC7$_H$
4	FFC8$_H$: FFC9$_H$
5	FFCA$_H$: FFCB$_H$
6	FFCC$_H$: FFCD$_H$
7	FFCE$_H$: FFCF$_H$

Figure E.1-8 shows an example.

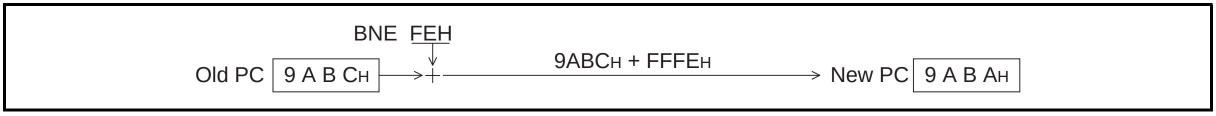
Figure E.1-8 Example of Vector Addressing



- Relative addressing

This is used when branching to the area in 128 bytes before and behind PC (program counter) with the addressing shown "rel" in the instruction table. In this addressing, add the content of the operand to PC with the sign and store the result in PC. Figure E.1-9 shows an example.

Figure E.1-9 Example of Relative Addressing

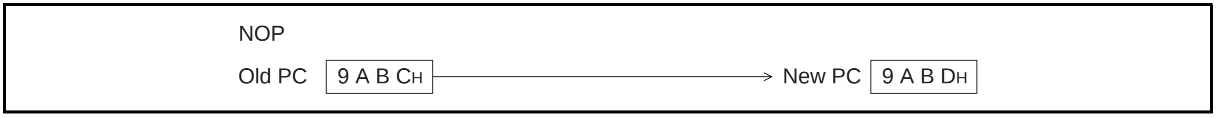


In this example, by jumping to the address where the operation code of BNE is stored, it results in an infinite loop.

- Inherent addressing

This is used when doing the operation decided by the operation code with the addressing that does not have the operand in the instruction table. In this addressing, the operation depends on each instruction. Figure E.1-10 shows an example.

Figure E.1-10 Example of Inherent Addressing



E.2 Special Instruction

This section explains special instructions other than the addressings.

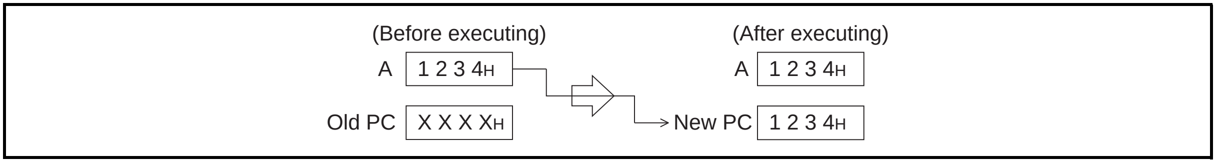
■ Special Instruction

● JMP @A

This instruction is to branch the content of A (accumulator) to PC (program counter) as an address. N pieces of the jump destination is arranged on the table, and one of the contents is selected and transferred to A. N branch processing can be done by executing this instruction.

Figure E.2-1 shows a summary of the instruction.

Figure E.2-1 JMP @A

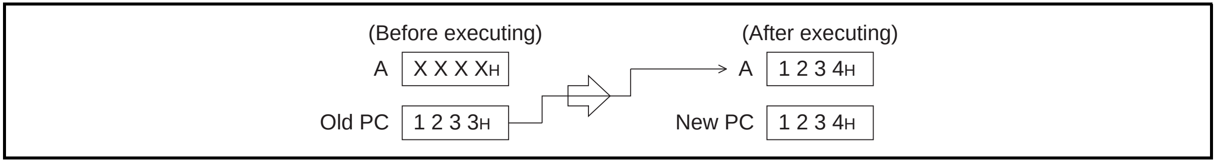


● MOVW A, PC

This instruction works as the opposite of "JMP @A". That is, it stores the content of PC to A. When you have executed this instruction in the main routine and set it to call a specific subroutine, you can make sure that the content of A is the specified value in the subroutine. Also, you can identify that the branch is not from the part that cannot be expected, and use it for the reckless driving judgment.

Figure E.2-2 shows a summary of the instruction.

Figure E.2-2 MOVW A, PC



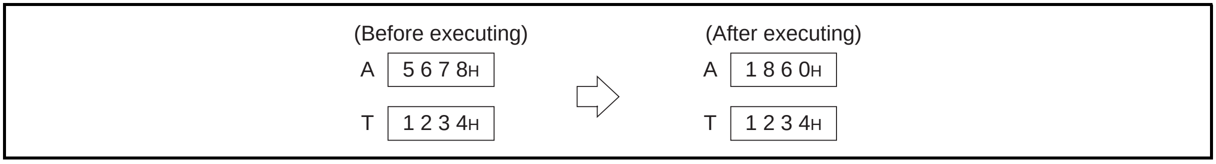
When this instruction is executed, the content of A reaches the same value as the address where the following instruction is stored, rather than the address where operation code of this instruction is stored. Therefore, in Figure E.2-2, the value "1234_H" stored in A corresponds to the address where the following operation code of "MOVW A, PC" is stored.

● MULU A

This instruction performs an unsigned multiplication of AL (lower 8-bit of the accumulator) and TL (lower 8-bit of the temporary accumulator), and stores the 16-bit result in A. The contents of T (temporary accumulator) do not change. The contents of AH (higher 8-bit of the accumulator) and TH (higher 8-bit of the temporary accumulator) before execution of the instruction are not used for the operation. The instruction does not change the flags, and therefore care must be taken when a branch may occur depending on the result of a multiplication.

Figure E.2-3 shows a summary of the instruction.

Figure E.2-3 MULU A

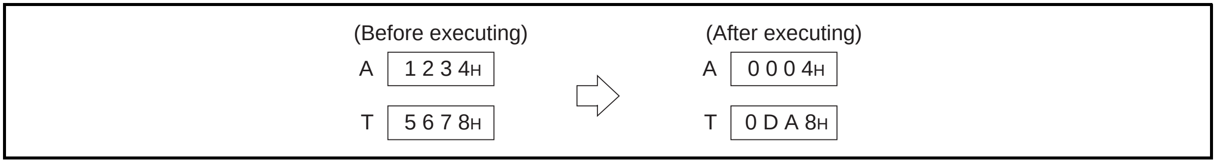


● DIVU A

This instruction divides the 16-bit value in T by the unsigned 16-bit value in A, and stores the 16-bit result and the 16-bit remainder in A and T, respectively. When the value in A before execution of instruction is "0", the Z flag becomes "1" to indicate zero-division is executed. The instruction does not change other flags, and therefore care must be taken when a branch may occur depending on the result of a division.

Figure E.2-4 shows a summary of the instruction.

Figure E.2-4 DIVU A

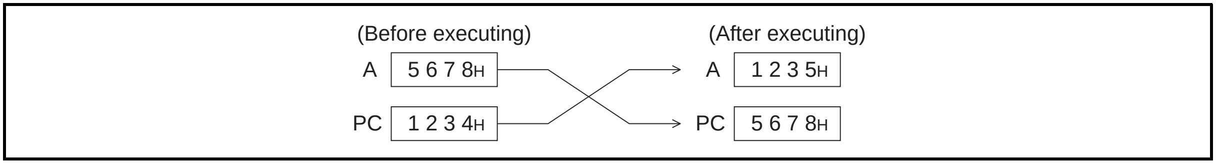


● XCHW A, PC

This instruction swaps the contents of A and PC, resulting in a branch to the address contained in A before execution of the instruction. After the instruction is executed, A becomes the address that follows the address where the operation code of "XCHW A, PC" is stored. This instruction is effective especially when it is used in the main routine to specify a table for use in a subroutine.

Figure E.2-5 shows a summary of the instruction.

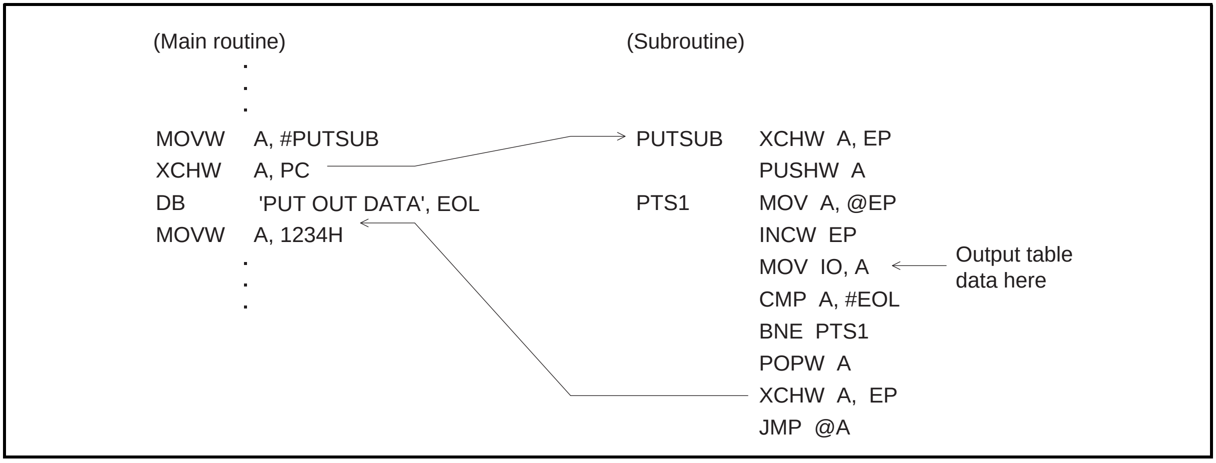
Figure E.2-5 XCHW A, PC



When this instruction is executed, the content of A reaches the same value as the address where the following instruction is stored, rather than the address where operation code of this instruction is stored. Therefore, in Figure E.2-5, the value "1235$_H$" stored in A corresponds to the address where the following operation code of "XCHW A, PC" is stored. This is why "1235$_H$" is stored instead of "1234$_H$".

Figure E.2-6 shows an assembler language example.

Figure E.2-6 Example of Using "XCHW A, PC"



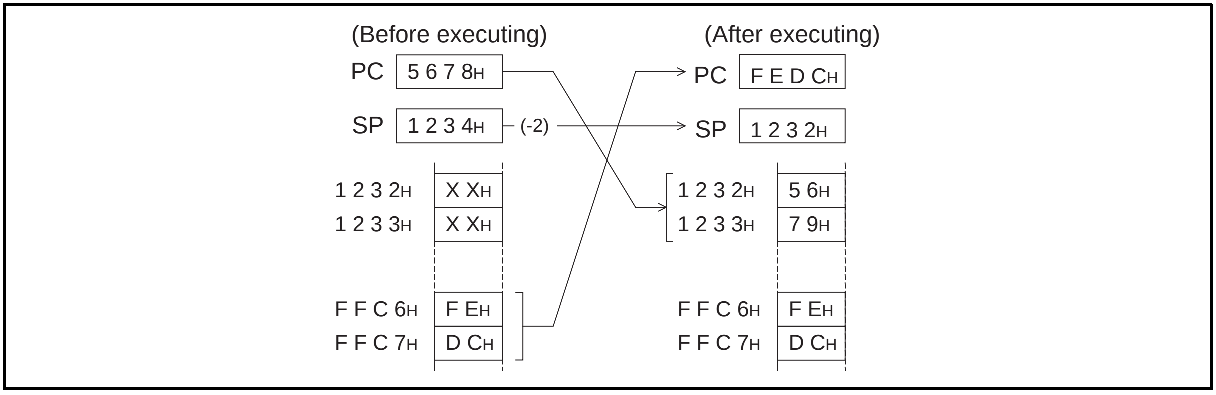
```
(Main routine)                                  (Subroutine)
        .
        .
        .
MOVW    A, #PUTSUB                              PUTSUB   XCHW  A, EP
XCHW    A, PC                                            PUSHW  A
DB       'PUT OUT DATA', EOL                    PTS1     MOV  A, @EP
MOVW    A, 1234H                                         INCW  EP
        .                                                MOV  IO, A     <---- Output table
        .                                                                     data here
        .                                                CMP  A, #EOL
                                                         BNE  PTS1
                                                         POPW  A
                                                         XCHW  A,  EP
                                                         JMP  @A
```

- CALLV #vct

This instruction is used to branch to a subroutine address stored in the vector table. The instruction saves the return address (contents of PC) in the location at the address contained in SP (stack pointer), and uses vector addressing to cause a branch to the address stored in the vector table. Because CALLV #vct is a 1-byte instruction, the use of this instruction for frequently used subroutines can reduce the entire program size.

Figure E.2-7 shows a summary of the instruction.

Figure E.2-7 Example of Executing CALLV #3



After the CALLV #vct instruction is executed, the contents of PC saved on the stack area are the address of the operation code of the next instruction, rather than the address of the operation code of CALLV #vct. Accordingly, Figure E.2-7 shows that the value saved in the stack (1232_H and 1233_H) is 5679_H, which is the address of the operation code of the instruction that follows "CALLV vct" (return address).

Table E.2-1 Vector Table

Vector use (call instruction)	Vector table address	
	Upper	Lower
CALLV #7	FFCE_H	FFCF_H
CALLV #6	FFCC_H	FFCD_H
CALLV #5	FFCA_H	FFCB_H
CALLV #4	FFC8_H	FFC9_H
CALLV #3	FFC6_H	FFC7_H
CALLV #2	FFC4_H	FFC5_H
CALLV #1	FFC2_H	FFC3_H
CALLV #0	FFC0_H	FFC1_H

E.3 Bit Manipulation Instructions (SETB, CLRB)

Some peripheral function registers include bits that are read differently than usual by a bit manipulation instruction.

■ Read-modify-write Operation

By using these bit manipulation instructions, you can set only the specified bit in a register or RAM location to "1" (SETB) or clear to "0" (CLRB). However, as the CPU operates data in 8-bit units, the actual operation (read-modify-write operation) involves a sequence of steps: 8-bit data is read, the specified bit is changed, and the data is written back to the location at the original address.

Table E.3-1 shows bus operation for bit manipulation instructions.

Table E.3-1 Bus Operation for Bit Manipulation Instructions

CODE	MNEMONIC	~	Cycle	Address bus	Data bus	RD	WR	RMW
A0 to A7	CLRB dir:b	4	1	N+2	Next instruction	1	0	1
			2	dir address	Data	1	0	1
A8 to AF	SETB dir:b		3	dir address	Data	0	1	0
			4	N+3	Instruction after next	1	0	0

■ Read Destination on the Execution of Bit Manipulation Instructions

For some I/O ports and the interrupt request flag bits, the read destination differs between a normal read operation and a read-modify-write operation.

● I/O ports (during a bit manipulation)

From some I/O ports, an I/O pin value is read during a normal read operation, while a port data register value is read during a bit manipulation. This prevents the other port data register bits from being changed accidentally, regardless of the I/O directions and states of the pins.

● Interrupt request flag bits (during a bit manipulation)

An interrupt request flag bit functions as a flag bit indicating whether an interrupt request exists during a normal read operation, however, "1" is always read from this bit during a bit manipulation. This prevents the flag from being cleared accidentally by writing the value "0" to the interrupt request flag bit when manipulating another bit.

E.4 F2MC-8FX Instructions

Table E.4-1 to Table E.4-4 show the instructions used by the F2MC-8FX.

■ Transfer Instructions

Table E.4-1 Transfer Instructions

No.	MNEMONIC		~	#	Operation	TL	TH	AH	N	Z	V	C	OPCODE
1	MOV	dir, A	3	2	(dir) ← (A)	-	-	-	-	-	-	-	45
2	MOV	@IX + off, A	3	2	((IX) + off) ← (A)	-	-	-	-	-	-	-	46
3	MOV	ext, A	4	3	(ext) ← (A)	-	-	-	-	-	-	-	61
4	MOV	@EP, A	2	1	((EP)) ← (A)	-	-	-	-	-	-	-	47
5	MOV	Ri, A	2	1	(Ri) ← (A)	-	-	-	-	-	-	-	48 to 4F
6	MOV	A, #d8	2	2	(A) ← d8	AL	-	-	+	+	-	-	04
7	MOV	A, dir	3	2	(A) ← (dir)	AL	-	-	+	+	-	-	05
8	MOV	A, @IX + off	3	2	(A) ← ((IX) + off)	AL	-	-	+	+	-	-	06
9	MOV	A, ext	4	3	(A) ← (ext)	AL	-	-	+	+	-	-	60
10	MOV	A, @A	2	1	(A) ← ((A))	AL	-	-	+	+	-	-	92
11	MOV	A, @EP	2	1	(A) ← ((EP))	AL	-	-	+	+	-	-	07
12	MOV	A, Ri	2	1	(A) ← (Ri)	AL	-	-	+	+	-	-	08 to 0F
13	MOV	dir, #d8	4	3	(dir) ← d8	-	-	-	-	-	-	-	85
14	MOV	@IX + off, #d8	4	3	((IX) + off) ← d8	-	-	-	-	-	-	-	86
15	MOV	@EP, #d8	3	2	((EP)) ← d8	-	-	-	-	-	-	-	87
16	MOV	Ri, #d8	3	2	(Ri) ← d8	-	-	-	-	-	-	-	88 to 8F
17	MOVW	dir, A	4	2	(dir) ← (AH) , (dir + 1) ← (AL)	-	-	-	-	-	-	-	D5
18	MOVW	@IX + off, A	4	2	((IX) + off) ← (AH) , ((IX) + off + 1) ← (AL)	-	-	-	-	-	-	-	D6
19	MOVW	ext, A	5	3	(ext) ← (AH) , (ext + 1) ← (AL)	-	-	-	-	-	-	-	D4
20	MOVW	@EP, A	3	1	((EP)) ← (AH) , ((EP) + 1) ← (AL)	-	-	-	-	-	-	-	D7
21	MOVW	EP, A	1	1	(EP) ← (A)	-	-	-	-	-	-	-	E3
22	MOVW	A, #d16	3	3	(A) ← d16	AL	AH	dH	+	+	-	-	E4
23	MOVW	A, dir	4	2	(AH) ← (dir) , (AL) ← (dir + 1)	AL	AH	dH	+	+	-	-	C5
24	MOVW	A, @IX + off	4	2	(AH) ← ((IX) + off) , (AL) ← ((IX) + off + 1)	AL	AH	dH	+	+	-	-	C6
25	MOVW	A, ext	5	3	(AH) ← (ext) , (AL) ← (ext + 1)	AL	AH	dH	+	+	-	-	C4
26	MOVW	A, @A	3	1	(AH) ← ((A)) , (AL) ← ((A) + 1)	AL	AH	dH	+	+	-	-	93
27	MOVW	A, @EP	3	1	(AH) ← ((EP)) , (AL) ← ((EP) + 1)	AL	AH	dH	+	+	-	-	C7
28	MOVW	A, EP	1	1	(A) ← (EP)	-	-	dH	-	-	-	-	F3
29	MOVW	EP, #d16	3	3	(EP) ← d16	-	-	-	-	-	-	-	E7
30	MOVW	IX, A	1	1	(IX) ← (A)	-	-	-	-	-	-	-	E2
31	MOVW	A, IX	1	1	(A) ← (IX)	-	-	dH	-	-	-	-	F2
32	MOVW	SP, A	1	1	(SP) ← (A)	-	-	-	-	-	-	-	E1
33	MOVW	A, SP	1	1	(A) ← (SP)	-	-	dH	-	-	-	-	F1
34	MOV	@A, T	2	1	((A)) ← (T)	-	-	-	-	-	-	-	82
35	MOVW	@A, T	3	1	((A)) ← (TH) , ((A) + 1) ← (TL)	-	-	-	-	-	-	-	83
36	MOVW	IX, #d16	3	3	(IX) ← d16	-	-	-	-	-	-	-	E6
37	MOVW	A, PS	1	1	(A) ← (PS)	-	-	dH	-	-	-	-	70
38	MOVW	PS, A	1	1	(PS) ← (A)	-	-	-	+	+	+	+	71
39	MOVW	SP, #d16	3	3	(SP) ← d16	-	-	-	-	-	-	-	E5
40	SWAP		1	1	(AH) ←→ (AL)	-	-	AL	-	-	-	-	10
41	SETB	dir:b	4	2	(dir) : b← 1	-	-	-	-	-	-	-	A8 to AF
42	CLRB	dir:b	4	2	(dir) : b← 0	-	-	-	-	-	-	-	A0 to A7
43	XCH	A, T	1	1	(AL) ←→ (TL)	AL	-	-	-	-	-	-	42
44	XCHW	A, T	1	1	(A) ←→ (T)	AL	AH	dH	-	-	-	-	43
45	XCHW	A, EP	1	1	(A) ←→ (EP)	-	-	dH	-	-	-	-	F7
46	XCHW	A, IX	1	1	(A) ←→ (IX)	-	-	dH	-	-	-	-	F6
47	XCHW	A, SP	1	1	(A) ←→ (SP)	-	-	dH	-	-	-	-	F5
48	MOVW	A, PC	2	1	(A) ← (PC)	-	-	dH	-	-	-	-	F0

Note:

In automatic transfer to T during byte transfer to A, AL is transferred to TL.
If an instruction has plural operands, they are saved in the order indicated by MNEMONIC.

■ Arithmetic Operation Instructions

Table E.4-2 Arithmetic Operation Instruction (1 / 2)

No.	MNEMONIC	~	#	Operation	TL	TH	AH	N	Z	V	C	OPCODE
1	ADDC A, Ri	2	1	(A) ← (A) + (Ri) + C	-	-	-	+	+	+	+	28 to 2F
2	ADDC A, #d8	2	2	(A) ← (A) + d8 + C	-	-	-	+	+	+	+	24
3	ADDC A, dir	3	2	(A) ← (A) + (dir) + C	-	-	-	+	+	+	+	25
4	ADDC A, @IX + off	3	2	(A) ← (A) + ((IX) + off) + C	-	-	-	+	+	+	+	26
5	ADDC A, @EP	2	1	(A) ← (A) + ((EP)) + C	-	-	-	+	+	+	+	27
6	ADDCW A	1	1	(A) ← (A) + (T) + C	-	-	dH	+	+	+	+	23
7	ADDC A	1	1	(AL) ← (AL) + (TL) + C	-	-	-	+	+	+	+	22
8	SUBC A, Ri	2	1	(A) ← (A) - (Ri) - C	-	-	-	+	+	+	+	38 to 3F
9	SUBC A, #d8	2	2	(A) ← (A) - d8 - C	-	-	-	+	+	+	+	34
10	SUBC A, dir	3	2	(A) ← (A) - (dir) - C	-	-	-	+	+	+	+	35
11	SUBC A, @IX + off	3	2	(A) ← (A) - ((IX) + off) - C	-	-	-	+	+	+	+	36
12	SUBC A, @EP	2	1	(A) ← (A) - ((EP)) - C	-	-	-	+	+	+	+	37
13	SUBCW A	1	1	(A) ← (T) - (A) - C	-	-	dH	+	+	+	+	33
14	SUBC A	1	1	(AL) ← (TL) - (AL) - C	-	-	-	+	+	+	+	32
15	INC Ri	3	1	(Ri) ← (Ri) + 1	-	-	-	+	+	+	-	C8 to CF
16	INCW EP	1	1	(EP) ← (EP) + 1	-	-	-	-	-	-	-	C3
17	INCW IX	1	1	(IX) ← (IX) + 1	-	-	-	-	-	-	-	C2
18	INCW A	1	1	(A) ← (A) + 1	-	-	dH	+	+	-	-	C0
19	DEC Ri	3	1	(Ri) ← (Ri) - 1	-	-	-	+	+	+	-	D8 to DF
20	DECW EP	1	1	(EP) ← (EP) - 1	-	-	-	-	-	-	-	D3
21	DECW IX	1	1	(IX) ← (IX) - 1	-	-	-	-	-	-	-	D2
22	DECW A	1	1	(A) ← (A) - 1	-	-	dH	+	+	-	-	D0
23	MULU A	8	1	(A) ← (AL) × (TL)	-	-	dH	-	-	-	-	01
24	DIVU A	17	1	(A) ← (T) / (A) , MOD→ (T)	dL	dH	dH	-	+	-	-	11
25	ANDW A	1	1	(A) ← (A) ∧ (T)	-	-	dH	+	+	R	-	63
26	ORW A	1	1	(A) ← (A) ∨ (T)	-	-	dH	+	+	R	-	73
27	XORW A	1	1	(A) ← (A) ∀ (T)	-	-	dH	+	+	R	-	53
28	CMP A	1	1	(TL) - (AL)	-	-	-	+	+	+	+	12
29	CMPW A	1	1	(T) - (A)	-	-	-	+	+	+	+	13
30	RORC A	1	1	↳ C→ A ─┘	-	-	-	+	+	-	+	03
31	ROLC A	1	1	└─ C← A ↲	-	-	-	+	+	-	+	02
32	CMP A, #d8	2	2	(A) - d8	-	-	-	+	+	+	+	14
33	CMP A, dir	3	2	(A) - (dir)	-	-	-	+	+	+	+	15
34	CMP A, @EP	2	1	(A) - ((EP))	-	-	-	+	+	+	+	17
35	CMP A, @IX + off	3	2	(A) - ((IX) + off)	-	-	-	+	+	+	+	16
36	CMP A, Ri	2	1	(A) - (Ri)	-	-	-	+	+	+	+	18 to 1F
37	DAA	1	1	decimal adjust for addition	-	-	-	+	+	+	+	84
38	DAS	1	1	decimal adjust for subtraction	-	-	-	+	+	+	+	94
39	XOR A	1	1	(A) ← (AL) ∀ (TL)	-	-	-	+	+	R	-	52
40	XOR A, #d8	2	2	(A) ← (AL) ∀ d8	-	-	-	+	+	R	-	54
41	XOR A, dir	3	2	(A) ← (AL) ∀ (dir)	-	-	-	+	+	R	-	55
42	XOR A, @EP	2	1	(A) ← (AL) ∀ ((EP))	-	-	-	+	+	R	-	57
43	XOR A, @IX + off	3	2	(A) ← (AL) ∀ ((IX) + off)	-	-	-	+	+	R	-	56
44	XOR A, Ri	2	1	(A) ← (AL) ∀ (Ri)	-	-	-	+	+	R	-	58 to 5F
45	AND A	1	1	(A) ← (AL) ∧ (TL)	-	-	-	+	+	R	-	62

Table E.4-2 Arithmetic Operation Instruction (2 / 2)

No.	MNEMONIC		~	#	Operation	TL	TH	AH	N	Z	V	C	OPCODE
46	AND	A, #d8	2	2	(A) ← (AL) ∧ d8	-	-	-	+	+	R	-	64
47	AND	A, dir	3	2	(A) ← (AL) ∧ (dir)	-	-	-	+	+	R	-	65
48	AND	A, @EP	2	1	(A) ← (AL) ∧ ((EP))	-	-	-	+	+	R	-	67
49	AND	A, @IX + off	3	2	(A) ← (AL) ∧ ((IX) + off)	-	-	-	+	+	R	-	66
50	AND	A, Ri	2	1	(A) ← (AL) ∧ (Ri)	-	-	-	+	+	R	-	68 to 6F
51	OR	A	1	1	(A) ← (AL) ∨ (TL)	-	-	-	+	+	R	-	72
52	OR	A, #d8	2	2	(A) ← (AL) ∨ d8	-	-	-	+	+	R	-	74
53	OR	A, dir	3	2	(A) ← (AL) ∨ (dir)	-	-	-	+	+	R	-	75
54	OR	A, @EP	2	1	(A) ← (AL) ∨ ((EP))	-	-	-	+	+	R	-	77
55	OR	A, @IX + off	3	2	(A) ← (AL) ∨ ((IX) + off)	-	-	-	+	+	R	-	76
56	OR	A, Ri	2	1	(A) ← (AL) ∨ (Ri)	-	-	-	+	+	R	-	78 to 7F
57	CMP	dir, #d8	4	3	(dir) - d8	-	-	-	+	+	+	+	95
58	CMP	@EP, #d8	3	2	((EP)) - d8	-	-	-	+	+	+	+	97
59	CMP	@IX + off, #d8	4	3	((IX) + off) - d8	-	-	-	+	+	+	+	96
60	CMP	Ri, #d8	3	2	(Ri) - d8	-	-	-	+	+	+	+	98 to 9F
61	INCW	SP	1	1	(SP) ← (SP) + 1	-	-	-	-	-	-	-	C1
62	DECW	SP	1	1	(SP) ← (SP) - 1	-	-	-	-	-	-	-	D1

■ Branch Instructions

Table E.4-3 Branch Instructions

No.	MNEMONIC		~	#	Operation	TL	TH	AH	N	Z	V	C	OPCODE
1	BZ/BEQ	rel(at branch)	4	2	if Z = 1 then PC← PC + rel	-	-	-	-	-	-	-	FD
	BZ/BEQ	rel(at no branch)	2										
2	BNZ/BNE	rel(at branch)	4	2	if Z = 0 then PC← PC + rel	-	-	-	-	-	-	-	FC
	BNZ/BNE	rel(at no branch)	2										
3	BC/BLO	rel(at branch)	4	2	if C = 1 then PC← PC + rel	-	-	-	-	-	-	-	F9
	BC/BLO	rel(at no branch)	2										
4	BNC/BHS	rel(at branch)	4	2	if C = 0 then PC← PC + rel	-	-	-	-	-	-	-	F8
	BNC/BHS	rel(at no branch)	2										
5	BN	rel(at branch)	4	2	if N = 1 then PC← PC + rel	-	-	-	-	-	-	-	FB
	BN	rel(at no branch)	2										
6	BP	rel(at branch)	4	2	if N = 0 then PC← PC + rel	-	-	-	-	-	-	-	FA
	BP	rel(at no branch)	2										
7	BLT	rel(at branch)	4	2	if V ∀ N = 1 then PC← PC + rel	-	-	-	-	-	-	-	FF
	BLT	rel(at no branch)	2										
8	BGE	rel(at branch)	4	2	if V ∀ N = 0 then PC← PC + rel	-	-	-	-	-	-	-	FE
	BGE	rel(at no branch)	2										
9	BBC	dir : b, rel	5	3	if (dir : b) = 0 then PC← PC + rel	-	-	-	-	+	-	-	B0 to B7
10	BBS	dir : b, rel	5	3	if (dir : b) = 1 then PC← PC + rel	-	-	-	-	+	-	-	B8 to BF
11	JMP	@A	3	1	(PC) ← (A)	-	-	-	-	-	-	-	E0
12	JMP	ext	4	3	(PC) ← ext	-	-	-	-	-	-	-	21
13	CALLV	#vct	7	1	vector call	-	-	-	-	-	-	-	E8 to EF
14	CALL	ext	6	3	subroutine call	-	-	-	-	-	-	-	31
15	XCHW	A, PC	3	1	(PC) ← (A) , (A) ← (PC) + 1	-	-	dH	-	-	-	-	F4
16	RET		6	1	return from subroutine	-	-	-	-	-	-	-	20
17	RETI		8	1	return from interrupt	-	-	-		restore			30

■ Other Instructions

Table E.4-4 Other Instructions

No.	MNEMONIC		~	#	Operation	TL	TH	AH	N	Z	V	C	OPCODE
1	PUSHW	A	4	1	((SP))← (A), (SP)← (SP) - 2	-	-	-	-	-	-	-	40
2	POPW	A	3	1	(A)← ((SP)), (SP)← (SP) + 2	-	-	dH	-	-	-	-	50
3	PUSHW	IX	4	1	((SP))← (IX), (SP)← (SP) - 2	-	-	-	-	-	-	-	41
4	POPW	IX	3	1	(IX)← ((SP)), (SP)← (SP) + 2	-	-	-	-	-	-	-	51
5	NOP		1	1	No operation	-	-	-	-	-	-	-	00
6	CLRC		1	1	(C)← 0	-	-	-	-	-	-	R	81
7	SETC		1	1	(C)← 1	-	-	-	-	-	-	S	91
8	CLRI		1	1	(I)← 0	-	-	-	-	-	-	-	80
9	SETI		1	1	(I)← 1	-	-	-	-	-	-	-	90

E.5 Instruction Map

Table E.5-1 shows the instruction map of F^{2}MC-8FX.

■ Instruction Map

Table E.5-1 Instruction Map of F^{2}MC-8FX

L \ H	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	SWAP	RET	RETI	PUSHW A	POPW A	MOV A, ext	MOVW A, PS	CLRI	SETI	CLRB dir : 0	BBC dir : 0, rel	INCW A	DECW A	JMP @A	MOVW A, PC
1	MULU A	DIVU A	JMP addr16	CALL addr16	PUSHW IX	POPW IX	MOV ext, A	MOVW PS, A	CLRC	SETC	CLRB dir : 1	BBC dir : 1, rel	INCW SP	DECW SP	MOVW SP, A	MOVW A, SP
2	ROLC A	CMP A	ADDC A	SUBC A	XCH A, T	XOR A	AND A	OR A	MOV @A, T	MOV A, @A	CLRB dir : 2	BBC dir : 2, rel	INCW IX	DECW IX	MOVW IX, A	MOVW A, IX
3	RORC A	CMPW A	ADDCW A	SUBCW A	XCHW A, T	XORW A	ANDW A	ORW A	MOVW @A, T	MOVW A, @A	CLRB dir : 3	BBC dir : 3, rel	INCW EP	DECW EP	MOVW EP, A	MOVW A, EP
4	MOV A, #d8	CMP A, #d8	ADDC A, #d8	SUBC A, #d8		XOR A, #d8	AND A, #d8	OR A, #d8	DAA	DAS	CLRB dir : 4	BBC dir : 4, rel	MOVW A, ext	MOVW ext, A	MOVW A, #d16	XCHW A, PC
5	MOV A, dir	CMP A, dir	ADDC A, dir	SUBC A, dir	MOV dir, A	XOR A, dir	AND A, dir	OR A, dir	MOV dir, #d8	CMP dir, #d8	CLRB dir : 5	BBC dir : 5, rel	MOVW A, dir	MOVW dir, A	MOVW SP, #d16	XCHW A, SP
6	MOV A, @IX+d	CMP A, @IX+d	ADDC A, @IX+d	SUBC A, @IX+d	MOV @IX+d, A	XOR A, @IX+d	AND A, @IX+d	OR A, @IX+d	MOV @IX+d,#d8	CMP @IX+d,#d8	CLRB dir : 6	BBC dir : 6, rel	MOVW A, @IX+d	MOVW @IX+d, A	MOVW IX, #d16	XCHW A, IX
7	MOV A, @EP	CMP A, @EP	ADDC A, @EP	SUBC A, @EP	MOV @EP, A	XOR A, @EP	AND A, @EP	OR A, @EP	MOV @EP, #d8	CMP @EP, #d8	CLRB dir : 7	BBC dir : 7, rel	MOVW A, @EP	MOVW @EP, A	MOVW EP, #d16	XCHW A, EP
8	MOV A, R0	CMP A, R0	ADDC A, R0	SUBC A, R0	MOV R0, A	XOR A, R0	AND A, R0	OR A, R0	MOV R0, #d8	CMP R0, #d8	SETB dir : 0	BBS dir : 0, rel	INC R0	DEC R0	CALLV #0	BNC rel
9	MOV A, R1	CMP A, R1	ADDC A, R1	SUBC A, R1	MOV R1, A	XOR A, R1	AND A, R1	OR A, R1	MOV R1, #d8	CMP R1, #d8	SETB dir : 1	BBS dir : 1, rel	INC R1	DEC R1	CALLV #1	BC rel
A	MOV A, R2	CMP A, R2	ADDC A, R2	SUBC A, R2	MOV R2, A	XOR A, R2	AND A, R2	OR A, R2	MOV R2, #d8	CMP R2, #d8	SETB dir : 2	BBS dir : 2, rel	INC R2	DEC R2	CALLV #2	BP rel
B	MOV A, R3	CMP A, R3	ADDC A, R3	SUBC A, R3	MOV R3, A	XOR A, R3	AND A, R3	OR A, R3	MOV R3, #d8	CMP R3, #d8	SETB dir : 3	BBS dir : 3, rel	INC R3	DEC R3	CALLV #3	BN rel
C	MOV A, R4	CMP A, R4	ADDC A, R4	SUBC A, R4	MOV R4, A	XOR A, R4	AND A, R4	OR A, R4	MOV R4, #d8	CMP R4, #d8	SETB dir : 4	BBS dir : 4, rel	INC R4	DEC R4	CALLV #4	BNZ rel
D	MOV A, R5	CMP A, R5	ADDC A, R5	SUBC A, R5	MOV R5, A	XOR A, R5	AND A, R5	OR A, R5	MOV R5, #d8	CMP R5, #d8	SETB dir : 5	BBS dir : 5, rel	INC R5	DEC R5	CALLV #5	BZ rel
E	MOV A, R6	CMP A, R6	ADDC A, R6	SUBC A, R6	MOV R6, A	XOR A, R6	AND A, R6	OR A, R6	MOV R6, #d8	CMP R6, #d8	SETB dir : 6	BBS dir : 6, rel	INC R6	DEC R6	CALLV #6	BGE rel
F	MOV A, R7	CMP A, R7	ADDC A, R7	SUBC A, R7	MOV R7, A	XOR A, R7	AND A, R7	OR A, R7	MOV R7, #d8	CMP R7, #d8	SETB dir : 7	BBS dir : 7, rel	INC R7	DEC R7	CALLV #7	BLT rel

MB95410H/470H Series
APPENDIX F Mask Options

Table F-1 shows the mask option list of the MB95410H/470H Series.

■ Mask Option List

Table F-1 Mask Option List

No.	Part Number	MB95F414H MB95F416H MB95F418H MB95F474H MB95F476H MB95F478H	MB95F414K MB95F416K MB95F418K MB95F474K MB95F476K MB95F478K
	Selectable/Fixed	Fixed	
1	Low-voltage detection reset	Without low-voltage detection reset	With low-voltage detection reset
2	Reset	With dedicated reset input	Without dedicated reset input

MB95410H/470H Series

Register Index

MB95410H/470H Series
Pin Function Index

MB95410H/470H Series

Interrupt Vector Index

MN702-00005-2v0-E

FUJITSU SEMICONDUCTOR • CONTROLLER MANUAL
New 8FX
8-BIT MICROCONTROLLER
MB95410H/470H Series
HARDWARE MANUAL

June 2013 the second edition

Published **FUJITSU SEMICONDUCTOR LIMITED**
Edited Sales Promotion Department