Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	78K/0
Core Size	8-Bit
Speed	20MHz
Connectivity	3-Wire SIO, I²C, LINbus, UART/USART
Peripherals	LVD, POR, PWM, WDT
Number of I/O	71
Program Memory Size	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f0545agk-gak-ax

3.1.3 Internal data memory space

78K0/Kx2 microcontrollers incorporate the following RAMs.

(1) Internal high-speed RAM

The 32-byte area FEE0H to FEFFH is assigned to four general-purpose register banks consisting of eight 8-bit registers per bank.

This area cannot be used as a program area in which instructions are written and executed.

The internal high-speed RAM can also be used as a stack memory.

Table 3-6. Internal High-Speed RAM Capacity

78K0/KB2	78K0/KC2		78K0/KD2	78K0/KE2	78K0/KF2	Internal High-Speed RAM
30/36 Pins	38/44 Pins	48 Pins	52 Pins	64 Pins	80 Pins	
μPD78F0500, PD78F0500A	–	–	–	–	–	512 × 8 bits (FD00H to FEFFH)
μPD78F0501, 78F0501A	μPD78F0511, 78F0511A	μPD78F0511, 78F0511A	μPD78F0521, 78F0521A	μPD78F0531, 78F0531A	–	768 × 8 bits (FC00H to FEFFH)
μPD78F0502, 78F0502A	μPD78F0512, 78F0512A	μPD78F0512, 78F0512A	μPD78F0522, 78F0522A	μPD78F0532, 78F0532A	–	1024 × 8 bits (FB00H to FEFFH)
μPD78F0503D, 78F0503DA	μPD78F0513D, 78F0513DA	μPD78F0513, 78F0513A	μPD78F0523, 78F0523A	μPD78F0533, 78F0533A	–	
μPD78F0503, 78F0503A	μPD78F0513, 78F0513A					
–	–	μPD78F0514, 78F0514A	μPD78F0524, 78F0524A	μPD78F0534, 78F0534A	μPD78F0544, 78F0544A	
–	–	μPD78F0515D, 78F0515DA	μPD78F0525, 78F0525A	μPD78F0535, 78F0535A	μPD78F0545, 78F0545A	
		μPD78F0515, 78F0515A				
–	–	–	μPD78F0526, 78F0526A	μPD78F0536, 78F0536A	μPD78F0546, 78F0546A	
–	–	–	μPD78F0527D, 78F0527DA	μPD78F0537D, 78F0537DA	μPD78F0547D, 78F0547DA	
			μPD78F0527, 78F0527A	μPD78F0537, 78F0537A	μPD78F0547, 78F0547A	

3.2.2 General-purpose registers

General-purpose registers are mapped at particular addresses (FEE0H to FEFFH) of the data memory. The general-purpose registers consists of 4 banks, each bank consisting of eight 8-bit registers (X, A, C, B, E, D, L, and H).

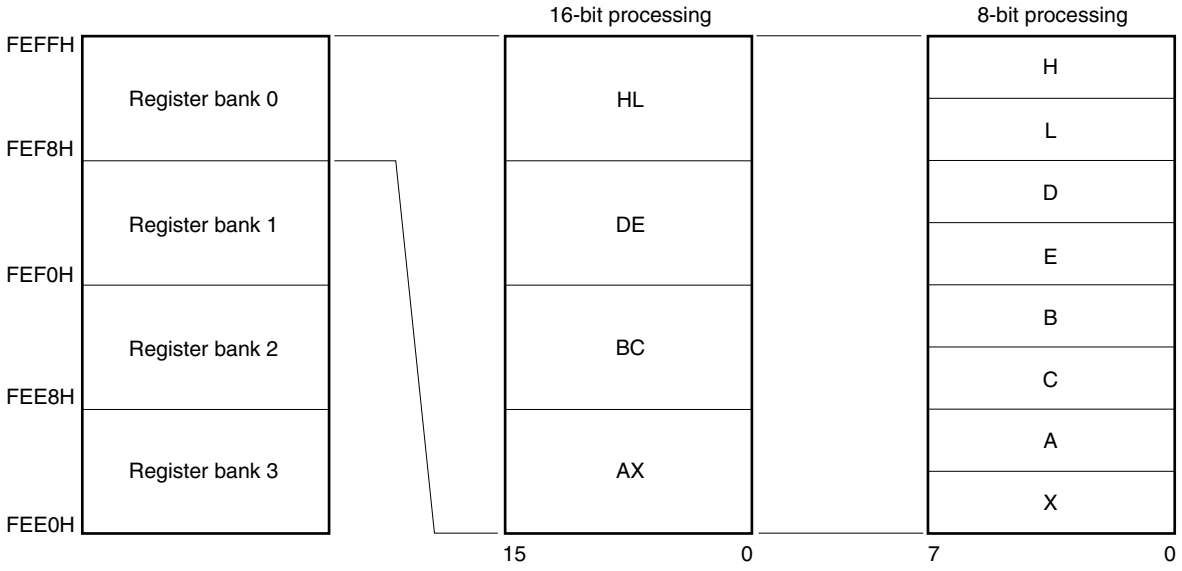
Each register can be used as an 8-bit register, and two 8-bit registers can also be used in a pair as a 16-bit register (AX, BC, DE, and HL).

These registers can be described in terms of function names (X, A, C, B, E, D, L, H, AX, BC, DE, and HL) and absolute names (R0 to R7 and RP0 to RP3).

Register banks to be used for instruction execution are set by the CPU control instruction (SEL RBn). Because of the 4-register bank configuration, an efficient program can be created by switching between a register for normal processing and a register for interrupts for each bank.

Figure 3-25. Configuration of General-Purpose Registers

(a) Function name



(b) Absolute name

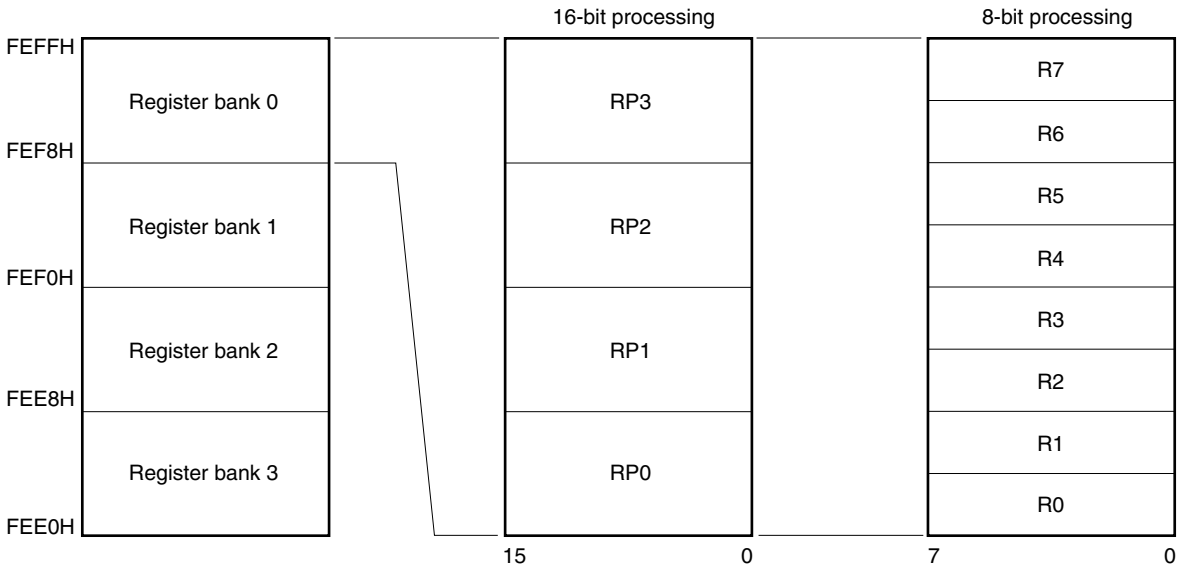
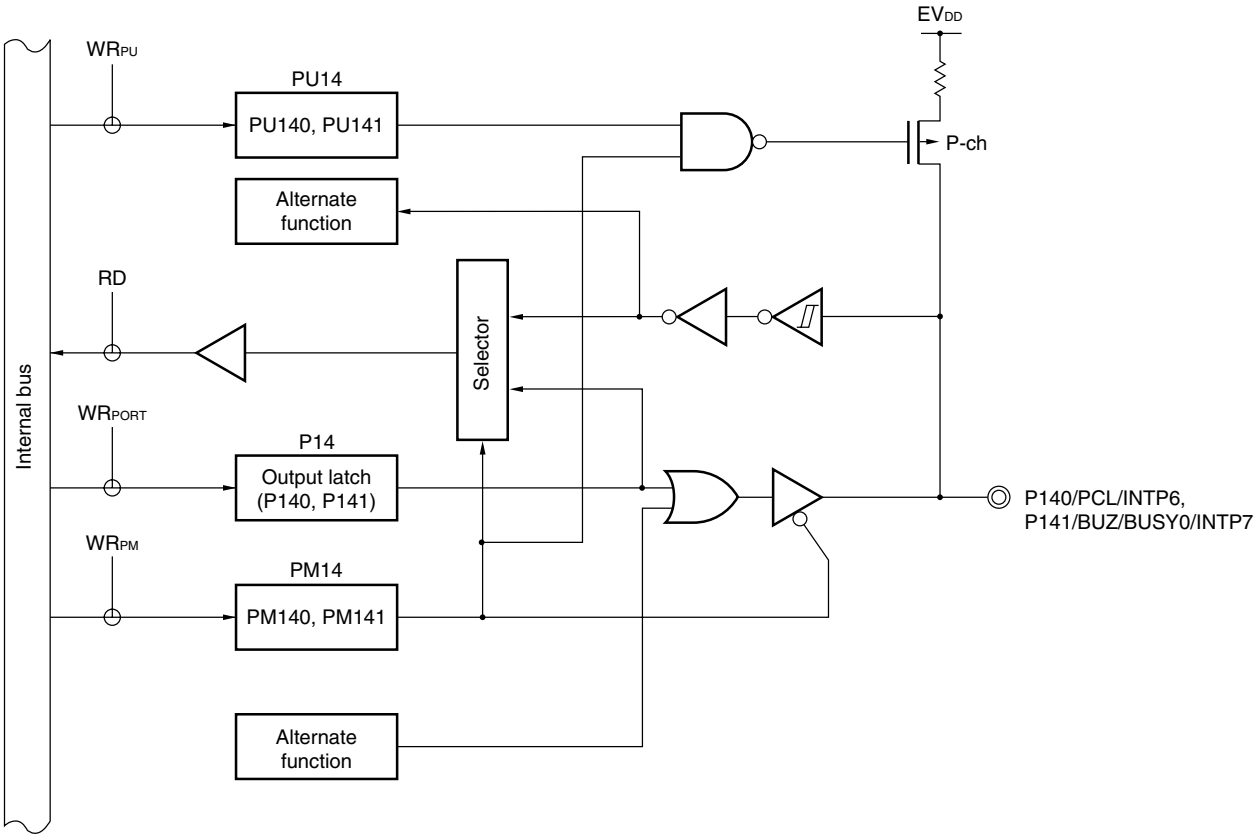


Figure 5-25. Block Diagram of P140 and P141



P14: Port register 14
PU14: Pull-up resistor option register 14
PM14: Port mode register 14
RD: Read signal
WR××: Write signal

Remark With products not provided with an EVDD or EVSS pin, replace EVDD with VDD, or replace EVSS with VSS.

5.4 Port Function Operations

Port operations differ depending on whether the input or output mode is set, as shown below.

5.4.1 Writing to I/O port

(1) Output mode

A value is written to the output latch by a transfer instruction, and the output latch contents are output from the pin.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

(2) Input mode

A value is written to the output latch by a transfer instruction, but since the output buffer is off, the pin status does not change.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

5.4.2 Reading from I/O port

(1) Output mode

The output latch contents are read by a transfer instruction. The output latch contents do not change.

(2) Input mode

The pin status is read by a transfer instruction. The output latch contents do not change.

5.4.3 Operations on I/O port

(1) Output mode

An operation is performed on the output latch contents, and the result is written to the output latch. The output latch contents are output from the pins.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

(2) Input mode

The pin level is read and an operation is performed on its contents. The result of the operation is written to the output latch, but since the output buffer is off, the pin status does not change.
The data of the output latch is cleared when a reset signal is generated.

5.5 Settings of Port Mode Register and Output Latch When Using Alternate Function

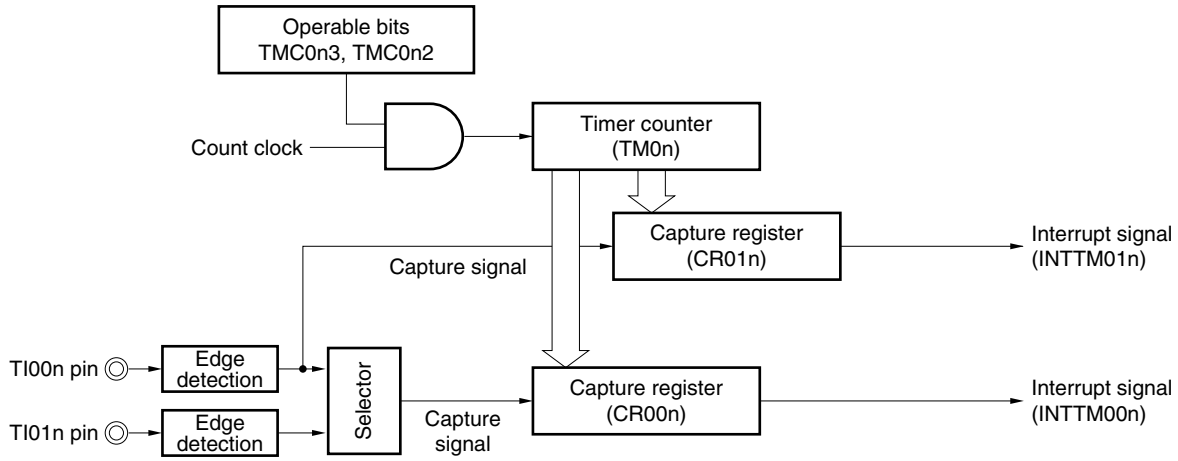
To use the alternate function of a port pin, set the port mode register and output latch as shown in Table 5-6.

Remark The port pins mounted depend on the product. See **Table 5-3. Port Functions**.

(3) Free-running timer mode operation

(CR00n: capture register, CR01n: capture register)

Figure 7-41. Block Diagram of Free-Running Timer Mode
(CR00n: Capture Register, CR01n: Capture Register)



Remarks 1. If both CR00n and CR01n are used as capture registers in the free-running timer mode, the TO0n output level is not inverted.

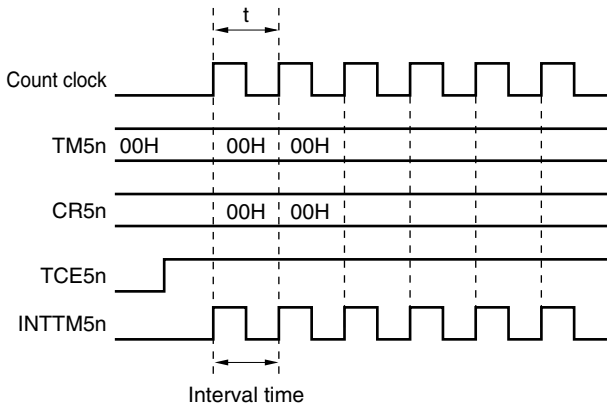
However, it can be inverted each time the valid edge of the TI00n pin is detected if bit 1 (TMC0n1) of 16-bit timer mode control register 0n (TMC0n) is set to 1.

2. n = 0: 78K0/KE2 products whose flash memory is less than 32 KB, and 78K0/KB2, 78K0/KC2, 78K0/KD2 products

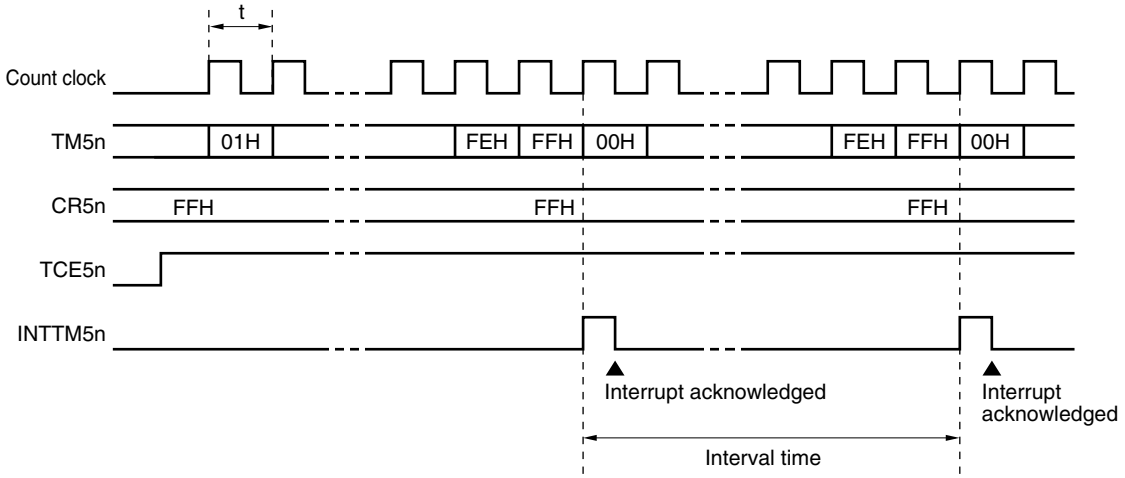
n = 0, 1: 78K0/KE2 products whose flash memory is at least 48 KB, and 78K0/KF2 products

Figure 8-11. Interval Timer Operation Timing (2/2)

(b) When CR5n = 00H



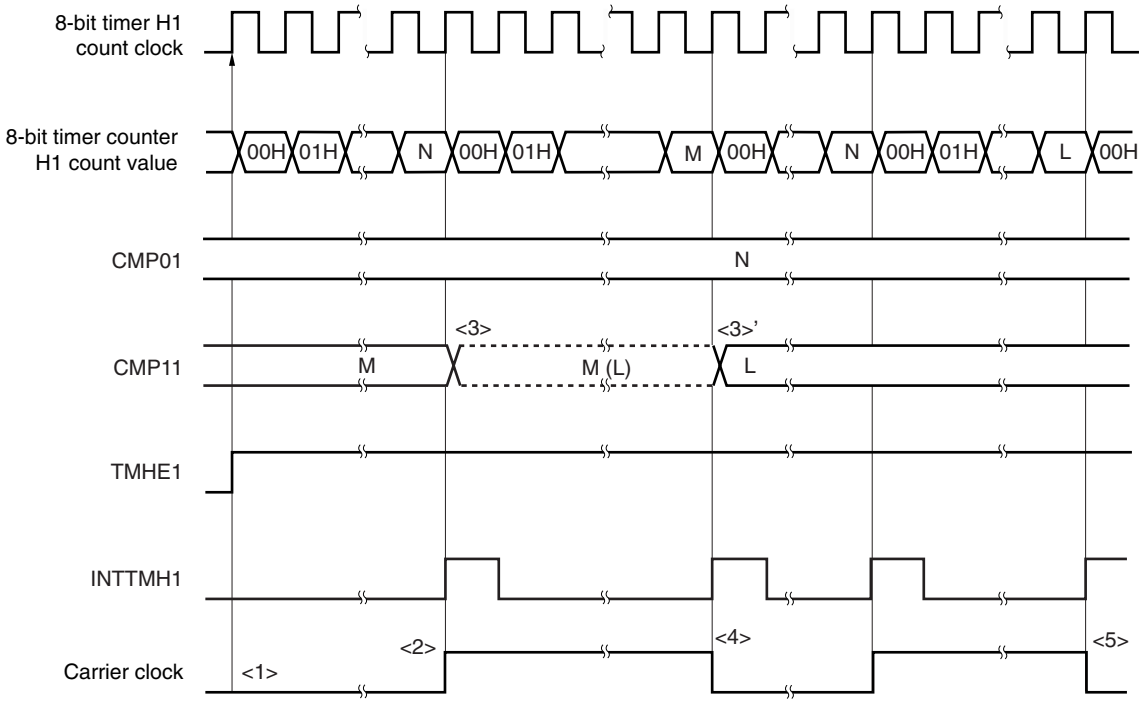
(c) When CR5n = FFH



Remark n = 0, 1

Figure 9-15. Carrier Generator Mode Operation Timing (3/3)

(c) Operation when CMP11 is changed



<1> When TMHE1 = 1 is set, the 8-bit timer H1 starts a count operation. At that time, the carrier clock remains default.

<2> When the count value of the 8-bit timer counter H1 matches the value of the CMP01 register, the INTTMH1 signal is output, the carrier signal is inverted, and the timer counter is cleared to 00H. At the same time, the compare register whose value is to be compared with that of the 8-bit timer counter H1 is changed from the CMP01 register to the CMP11 register.

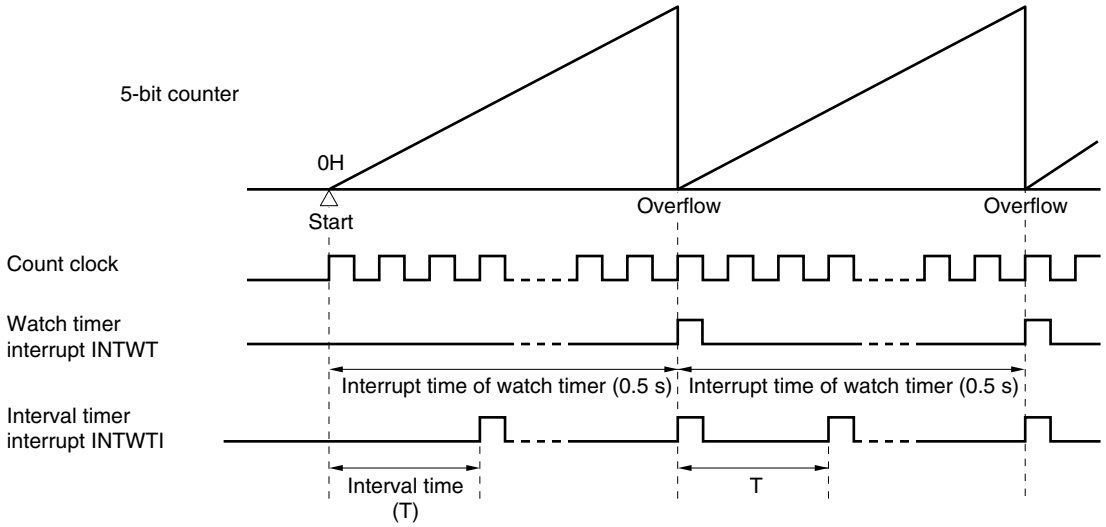
<3> The CMP11 register is asynchronous to the count clock, and its value can be changed while the 8-bit timer H1 is operating. The new value (L) to which the value of the register is to be changed is latched. When the count value of the 8-bit timer counter H1 matches the value (M) of the CMP11 register before the change, the CMP11 register is changed (<3>').

However, it takes three count clocks or more since the value of the CMP11 register has been changed until the value is transferred to the register. Even if a match signal is generated before the duration of three count clocks elapses, the new value is not transferred to the register.

<4> When the count value of 8-bit timer counter H1 matches the value (M) of the CMP1 register before the change, the INTTMH1 signal is output, the carrier signal is inverted, and the timer counter is cleared to 00H. At the same time, the compare register whose value is to be compared with that of the 8-bit timer counter H1 is changed from the CMP11 register to the CMP01 register.

<5> The timing at which the count value of the 8-bit timer counter H1 and the CMP11 register value match again is indicated by the value after the change (L).

Figure 10-3. Operation Timing of Watch Timer/Interval Timer



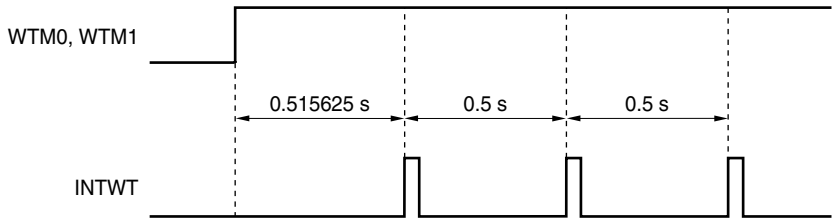
Remark fw: Watch timer clock frequency
Figures in parentheses are for operation with fw = 32.768 kHz (WTM7 = 1, WTM3, WTM2 = 0, 0)

10.5 Cautions for Watch Timer

When operation of the watch timer and 5-bit counter is enabled by the watch timer mode control register (WTM) (by setting bits 0 (WTM0) and 1 (WTM1) of WTM to 1), the interval until the first interrupt request signal (INTWT) is generated after the register is set does not exactly match the specification made with bits 2 and 3 (WTM2, WTM3) of WTM. Subsequently, however, the INTWT signal is generated at the specified intervals.

Figure 10-4. Example of Generation of Watch Timer Interrupt Request Signal (INTWT)
(When Interrupt Period = 0.5 s)

It takes 0.515625 seconds for the first INTWT to be generated ($2^9 \times 1/32768 = 0.015625$ s longer).
INTWT is then generated every 0.5 seconds.



11.2 Configuration of Watchdog Timer

The watchdog timer includes the following hardware.

Table 11-1. Configuration of Watchdog Timer

Item	Configuration
Control register	Watchdog timer enable register (WDTE)

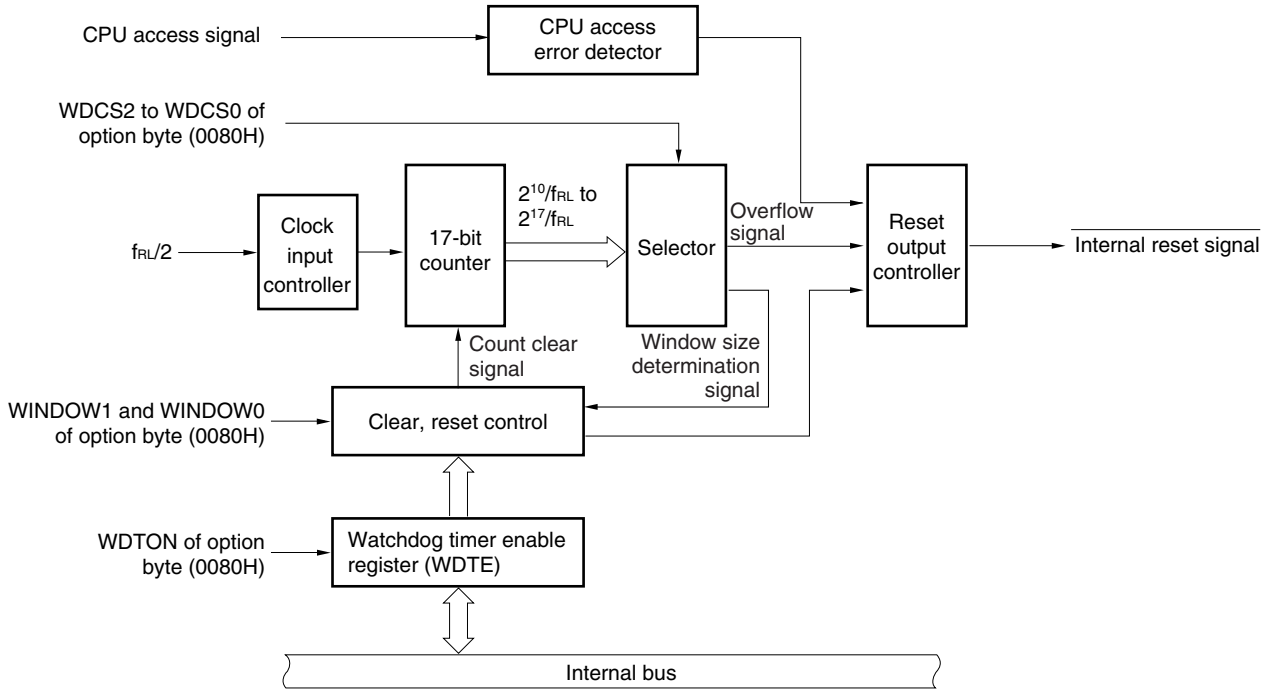
How the counter operation is controlled, overflow time, and window open period are set by the option byte.

Table 11-2. Setting of Option Bytes and Watchdog Timer

Setting of Watchdog Timer	Option Byte (0080H)
Window open period	Bits 6 and 5 (WINDOW1, WINDOW0)
Controlling counter operation of watchdog timer	Bit 4 (WDTON)
Overflow time of watchdog timer	Bits 3 to 1 (WDCS2 to WDCS0)

Remark For the option byte, see **CHAPTER 26 OPTION BYTE**.

Figure 11-1. Block Diagram of Watchdog Timer



(5) Full-scale error

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (Full-scale – 3/2LSB) when the digital output changes from 1.....110 to 1.....111.

(6) Integral linearity error

This shows the degree to which the conversion characteristics deviate from the ideal linear relationship. It expresses the maximum value of the difference between the actual measurement value and the ideal straight line when the zero-scale error and full-scale error are 0.

(7) Differential linearity error

While the ideal width of code output is 1LSB, this indicates the difference between the actual measurement value and the ideal value.

Figure 13-16. Zero-Scale Error

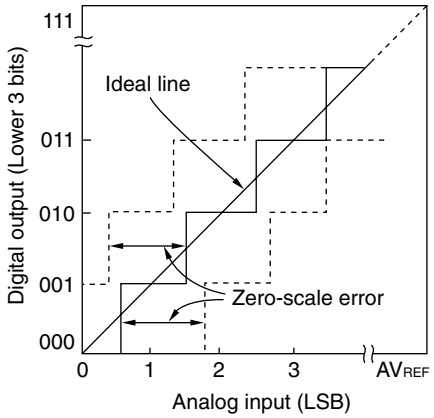


Figure 13-17. Full-Scale Error

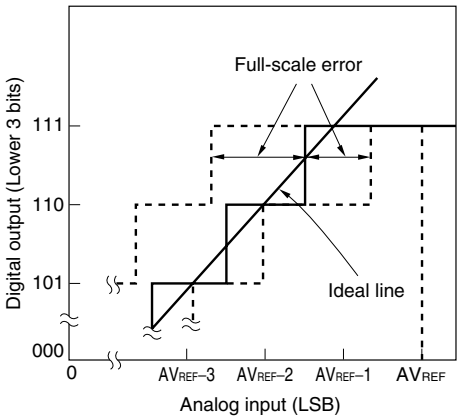


Figure 13-18. Integral Linearity Error

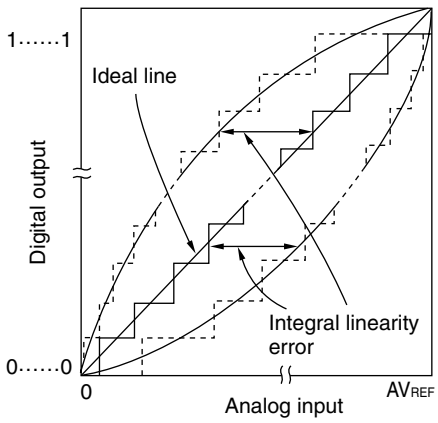
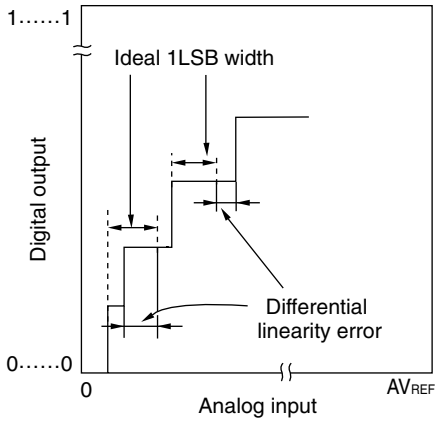


Figure 13-19. Differential Linearity Error



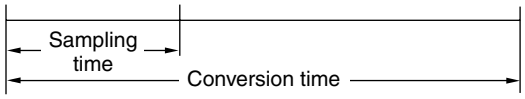
(8) Conversion time

This expresses the time from the start of sampling to when the digital output is obtained.

The sampling time is included in the conversion time in the characteristics table.

(9) Sampling time

This is the time the analog switch is turned on for the analog voltage to be sampled by the sample & hold circuit.



(4) Divisor selection register 0 (BRGCA0)

This is an 8-bit register used to select the base clock divisor of CSIA0.

This register can be set by an 8-bit memory manipulation instruction. However, when bit 0 (TSF0) of serial status register 0 (CSIS0) is 1, rewriting BRGCA0 is prohibited.

Reset signal generation sets this register to 03H.

Figure 17-5. Format of Divisor Selection Register 0 (BRGCA0)

Address: FF93H After reset: 03H R/W

Symbol	7	6	5	4	3	2	1	0
BRGCA0	0	0	0	0	0	0	BRGCA01	BRGCA00

BRGCA01	BRGCA00	Selection of base clock (f_W) divisor of CSIA0[Note]						
			f_W = 1 MHz	f_W = 2 MHz	f_W = 2.5 MHz	f_W = 5 MHz	f_W = 10 MHz	f_W = 20 MHz
0	0	$f_W/6$	166.67 kHz	333.3 kHz	416.67 kHz	833.33 kHz	1.67 MHz	Setting prohibited
0	1	$f_W/2^3$	125 kHz	250 kHz	312.5 kHz	625 kHz	1.25 MHz	Setting prohibited
1	0	$f_W/2^4$	62.5 kHz	125 kHz	156.25 kHz	312.5 kHz	625 kHz	1.25 MHz
1	1	$f_W/2^5$	31.25 kHz	62.5 kHz	78.125 kHz	156.25 kHz	312.5 kHz	625 kHz

Note Set the transfer clock so as to satisfy the following conditions.

- When 4.0 V ≤ V_{DD} ≤ 5.5 V: transfer clock ≤ 1.67 MHz
- When 2.7 V ≤ V_{DD} < 4.0 V: transfer clock ≤ 833.33 kHz
- When 1.8 V ≤ V_{DD} < 2.7 V: transfer clock ≤ 555.56 kHz (Standard products and (A) grade products only)

Remark f_W: Base clock frequency selected by CKS00 bit of CSIS0 register (f_{PRS} or $f_{PRS}/2$)

f_{PRS}: Peripheral hardware clock frequency

Figure 18-5. Format of IIC Control Register 0 (IICC0) (2/4)

SPIE0[Note 1]	Enable/disable generation of interrupt request when stop condition is detected
0	Disable
1	Enable

Condition for clearing (SPIE0 = 0)	Condition for setting (SPIE0 = 1)
• Cleared by instruction • Reset	• Set by instruction

WTIM0[Note 1]	Control of wait and interrupt request generation
0	Interrupt request is generated at the eighth clock's falling edge. Master mode: After output of eight clocks, clock output is set to low level and wait is set. Slave mode: After input of eight clocks, the clock is set to low level and wait is set for master device.
1	Interrupt request is generated at the ninth clock's falling edge. Master mode: After output of nine clocks, clock output is set to low level and wait is set. Slave mode: After input of nine clocks, the clock is set to low level and wait is set for master device.
An interrupt is generated at the falling edge of the ninth clock during address transfer independently of the setting of this bit. The setting of this bit is valid when the address transfer is completed. When in master mode, a wait is inserted at the falling edge of the ninth clock during address transfers. For a slave device that has received a local address, a wait is inserted at the falling edge of the ninth clock after an acknowledge ($\overline{ACK}$) is issued. However, when the slave device has received an extension code, a wait is inserted at the falling edge of the eighth clock.	

Condition for clearing (WTIM0 = 0)	Condition for setting (WTIM0 = 1)
• Cleared by instruction • Reset	• Set by instruction

ACKE0[Notes 1, 2]	Acknowledgment control
0	Disable acknowledgment.
1	Enable acknowledgment. During the ninth clock period, the SDA0 line is set to low level.

Condition for clearing (ACKE0 = 0)	Condition for setting (ACKE0 = 1)
• Cleared by instruction • Reset	• Set by instruction

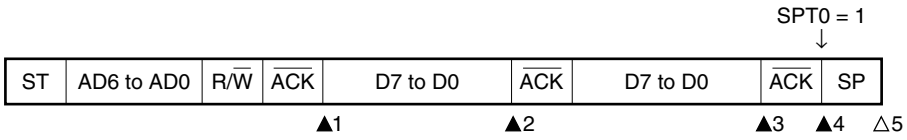
Notes 1. This flag's signal is invalid when IICE0 = 0.

2. The set value is invalid during address transfer and if the code is not an extension code.
When the device serves as a slave and the addresses match, an acknowledge is generated regardless of the set value.

(1) Master device operation

(a) Start ~ Address ~ Data ~ Data ~ Stop (transmission/reception)

(i) When WTIM0 = 0

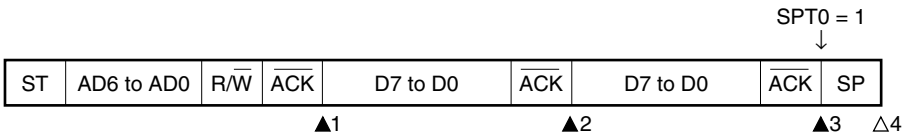


▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000×000B
▲3: IICS0 = 1000×000B (Sets WTIM0 to 1[Note])
▲4: IICS0 = 1000××00B (Sets SPT0 to 1)
△5: IICS0 = 00000001B

Note To generate a stop condition, set WTIM0 to 1 and change the timing for generating the INTIIC0 interrupt request signal.

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

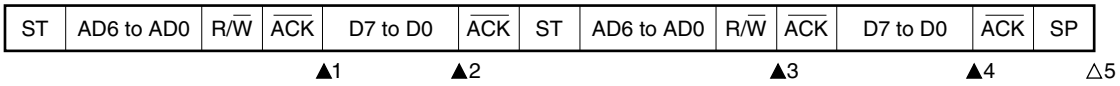


▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000×100B
▲3: IICS0 = 1000××00B (Sets SPT0 to 1)
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(c) Start ~ Address ~ Data ~ Start ~ Code ~ Data ~ Stop

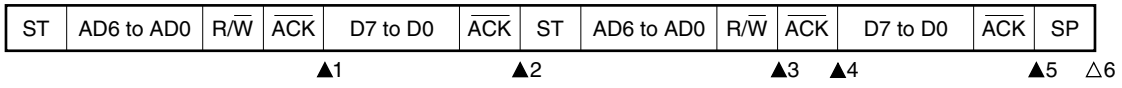
(i) When WTIM0 = 0 (after restart, does not match address (= extension code))



▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001×000B
▲3: IICS0 = 0010×010B
▲4: IICS0 = 0010×000B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1 (after restart, does not match address (= extension code))

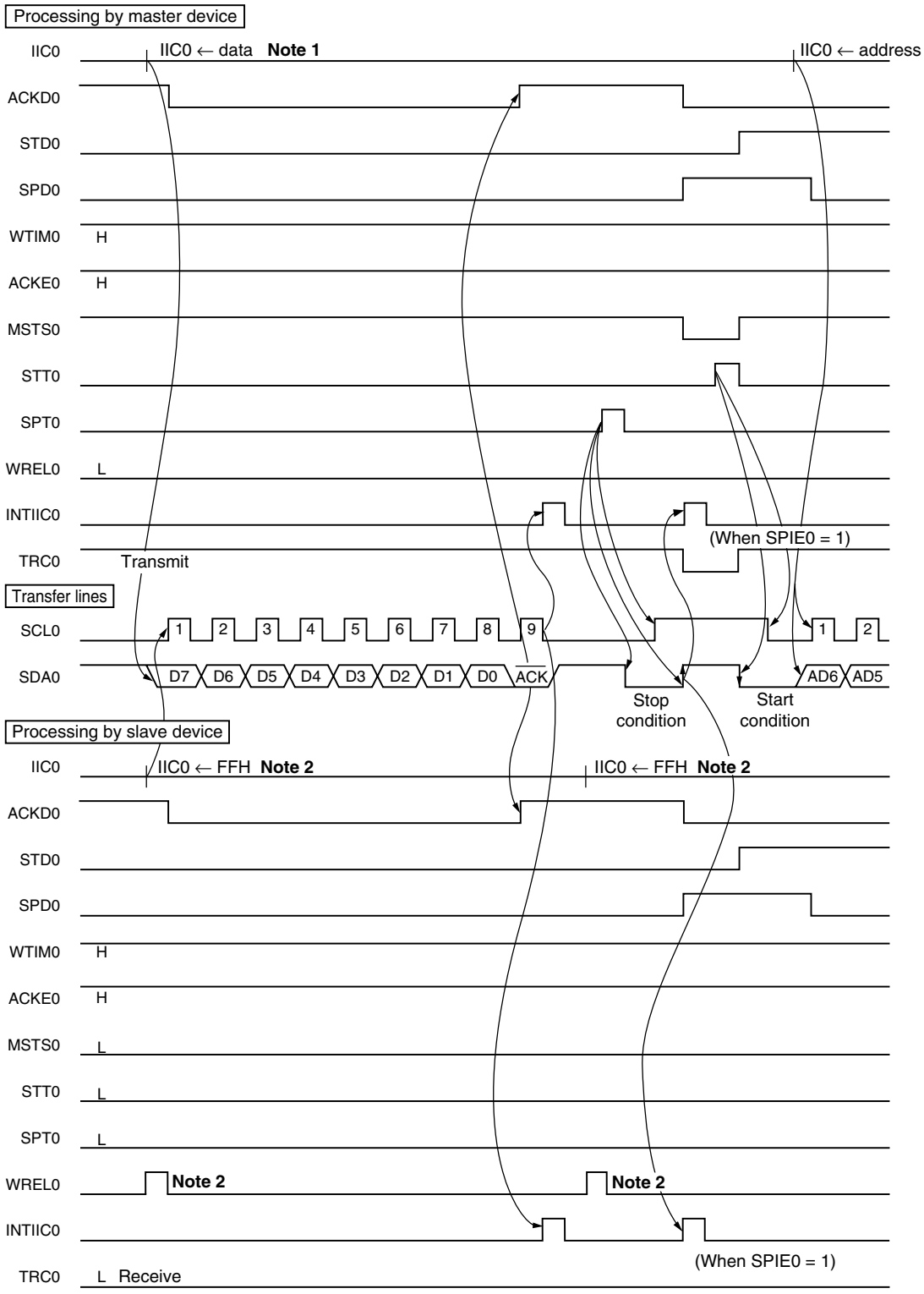


▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001××00B
▲3: IICS0 = 0010×010B
▲4: IICS0 = 0010×110B
▲5: IICS0 = 0010××00B
△6: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

Figure 18-27. Example of Master to Slave Communication
(When 9-Clock Wait Is Selected for Both Master and Slave) (3/3)

(3) Stop condition

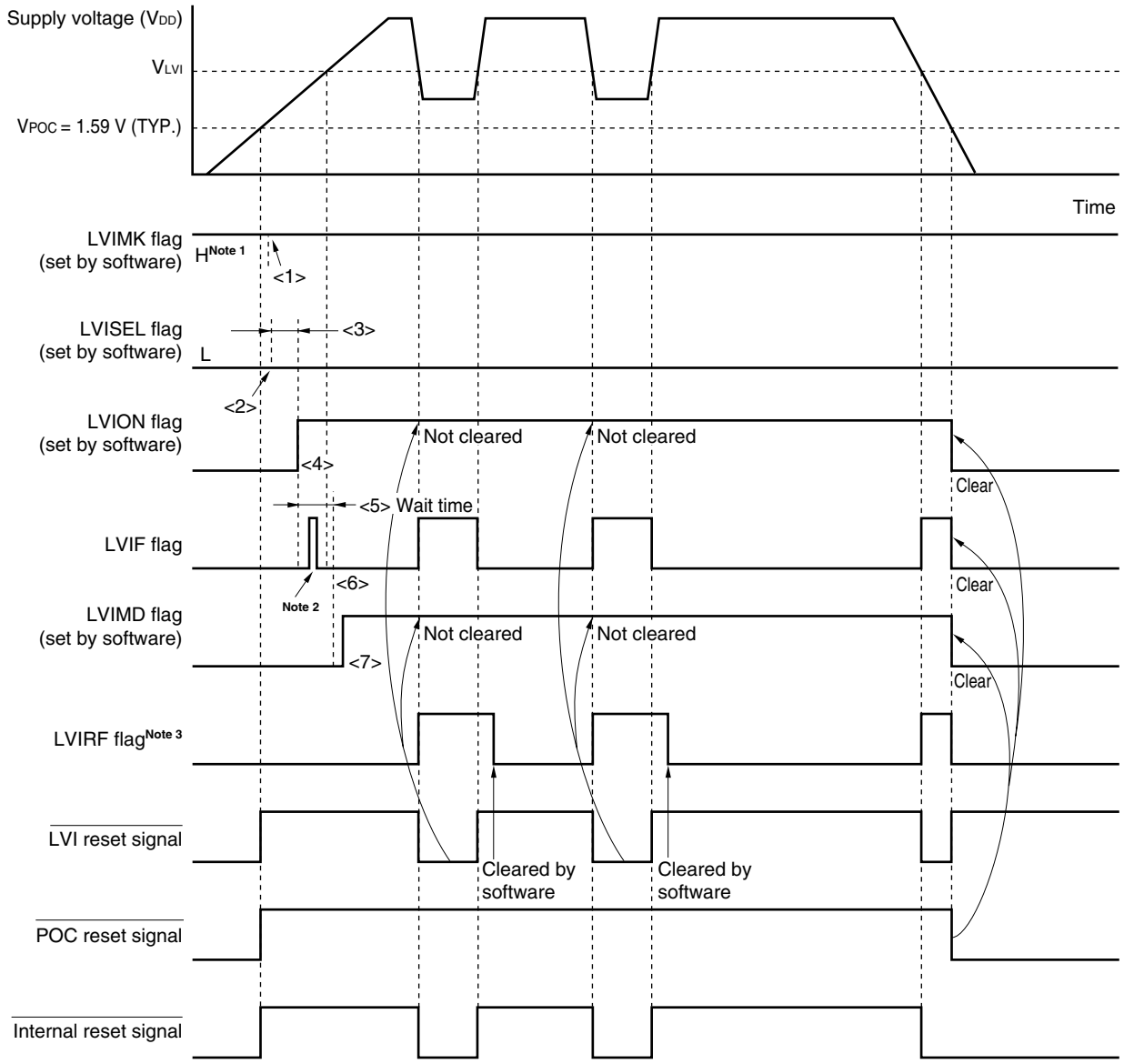


Notes 1. Write data to IIC0, not setting WREL0, in order to cancel a wait state during master transmission.

2. To cancel slave wait, write “FFH” to IIC0 or set WREL0.

Figure 25-5. Timing of Low-Voltage Detector Internal Reset Signal Generation (Detects Level of Supply Voltage (V_{DD})) (1/2)

(1) In 1.59 V POC mode (option byte: POCMODE = 0)



Notes 1. The LVIMK flag is set to "1" by reset signal generation.

2. The LVIF flag may be set (1).

3. LVIRF is bit 0 of the reset control flag register (RESF). For details of RESF, see **CHAPTER 23 RESET FUNCTION**.

Remark <1> to <7> in Figure 25-5 above correspond to <1> to <7> in the description of "When starting operation" in **25.4.1 (1) When detecting level of supply voltage (V_{DD})**.

27.8 Security Settings

The 78K0/Kx2 microcontrollers support a security function that prohibits rewriting the user program written to the internal flash memory, so that the program cannot be changed by an unauthorized person.

The operations shown below can be performed using the Security Set command. The security setting is valid when the programming mode is set next.

- Disabling batch erase (chip erase)
  Execution of the block erase and batch erase (chip erase) commands for entire blocks in the flash memory is prohibited by this setting during on-board/off-board programming. Once execution of the batch erase (chip erase) command is prohibited, all of the prohibition settings (including prohibition of batch erase (chip erase)) can no longer be cancelled.

  Caution After the security setting for the batch erase is set, erasure cannot be performed for the device. In addition, even if a write command is executed, data different from that which has already been written to the flash memory cannot be written, because the erase command is disabled.

- Disabling block erase
  Execution of the block erase command for a specific block in the flash memory is prohibited during on-board/off-board programming. However, blocks can be erased by means of self programming.

- Disabling write
  Execution of the write and block erase commands for entire blocks in the flash memory is prohibited during on-board/off-board programming. However, blocks can be written by means of self programming.

- Disabling rewriting boot cluster 0
  Execution of the block erase command and write command on boot cluster 0 (0000H to 0FFFH) in the flash memory is prohibited by this setting. Execution of the batch erase (chip erase) command is also prohibited by this setting.

  Caution If a security setting that rewrites boot cluster 0 has been applied, the rewriting of boot cluster 0 and the batch erase (chip erase) will not be executed for the device.

The batch erase (chip erase), block erase, write commands, and rewriting boot cluster 0 are enabled by the default setting when the flash memory is shipped. Security can be set by on-board/off-board programming and self programming. Each security setting can be used in combination.

Prohibition of erasing blocks and writing is cleared by executing the batch erase (chip erase) command.

Table 27-10 shows the relationship between the erase and write commands when the 78K0/Kx2 microcontroller security function is enabled.

Recommended Oscillator Constants (2/2)

(1) X1 oscillation: Ceramic resonator (T_A = –40 to +85°C) (2/2)

Manufacturer	Part Number	SMD/ Lead	Frequency (MHz)	Recommended Circuit Constants		Oscillation Voltage Range	
				C1 (pF)	C2 (pF)	MIN. (V)	MAX. (V)
TDK Corporation	CCR4.0MUC8	SMD	4.00	Internal (27)	Internal (27)	1.8	5.5
	FCR4.0MC5	Lead		Internal (30)	Internal (30)		
	CCR8.0MXC8	SMD	8.00	Internal (18)	Internal (30)		
	FCR8.0MC5	Lead		Internal (20)	Internal (20)		

Caution The oscillator constants shown above are reference values based on evaluation in a specific environment by the resonator manufacturer. If it is necessary to optimize the oscillator characteristics in the actual application, apply to the resonator manufacturer for evaluation on the implementation circuit. The oscillation voltage and oscillation frequency only indicate the oscillator characteristic. Use the 78K0/Kx2 so that the internal operation conditions are within the specifications of the DC and AC characteristics.

(2) XT1 oscillation: Crystal resonator (T_A = –40 to +85°C)

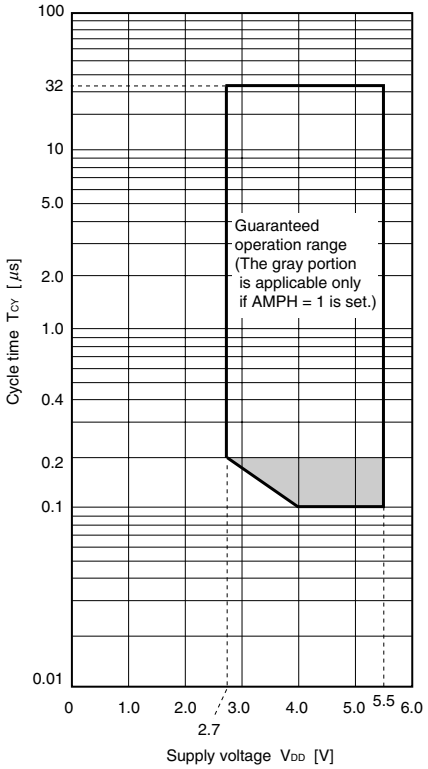
Manufacturer	Part Number	SMD/ Lead	Frequency (MHz)	Load Capacitance CL (pF)	Recommended Circuit Constants						Oscillation Voltage Range	
					V_{DD} = 3.3 V			V_{DD} = 5.0 V			MIN. (V)	MAX. (V)
					C3 (pF)	C4 (pF)	Rd (kΩ)	C3 (pF)	C4 (pF)	Rd (kΩ)		
Seiko Instruments Inc.	VT-200	Lead	32.768	6.0	4	3	100	6	5	100	1.8	5.5
				12.5	15	15	100	18	15	100		

Caution The oscillator constants shown above are reference values based on evaluation in a specific environment by the resonator manufacturer. If it is necessary to optimize the oscillator characteristics in the actual application, apply to the resonator manufacturer for evaluation on the implementation circuit. The oscillation voltage and oscillation frequency only indicate the oscillator characteristic. Use the 78K0/KE2 so that the internal operation conditions are within the specifications of the DC and AC characteristics.

Caution The pins mounted depend on the product. Refer to Caution at the beginning of this chapter.

T_{CY} vs. V_{DD} (Main System Clock Operation)

<1> Conventional-specification Products (μPD78F05xx(A2))



<2> Expanded-specification Products (μPD78F05xxA(A2))