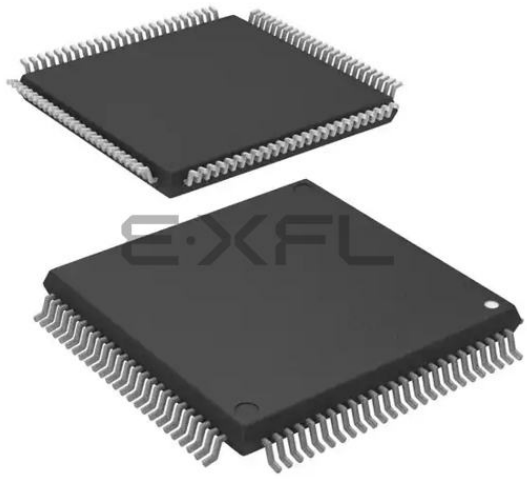
Renesas Electronics America Inc - UPD78F1164AGC-UEU-AX Datasheet



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/0R
Core Size	16-Bit
Speed	20MHz
Connectivity	3-Wire SIO, I²C, LINbus, UART/USART
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	83
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 16x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f1164agc-ueu-ax

3.2 Processor Registers

The 78K0R/KG3 products incorporate the following processor registers.

3.2.1 Control registers

The control registers control the program sequence, statuses and stack memory. The control registers consist of a program counter (PC), a program status word (PSW) and a stack pointer (SP).

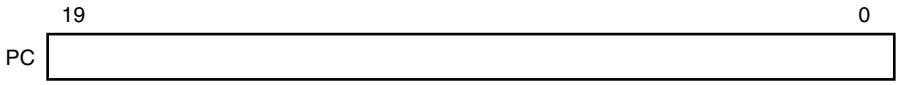
(1) Program counter (PC)

The program counter is a 20-bit register that holds the address information of the next program to be executed.

In normal operation, PC is automatically incremented according to the number of bytes of the instruction to be fetched. When a branch instruction is executed, immediate data and register contents are set.

Reset signal generation sets the reset vector table values at addresses 0000H and 0001H to the program counter.

Figure 3-16. Format of Program Counter



(2) Program status word (PSW)

The program status word is an 8-bit register consisting of various flags set/reset by instruction execution.

Program status word contents are stored in the stack area upon vector interrupt request acknowledgment or PUSH PSW instruction execution and are restored upon execution of the RETB, RETI and POP PSW instructions.

Reset signal generation sets PSW to 06H.

Figure 3-17. Format of Program Status Word

	7							0
PSW	IE	Z	RBS1	AC	RBS0	ISP1	ISP0	CY

(a) Interrupt enable flag (IE)

This flag controls the interrupt request acknowledge operations of the CPU.

When 0, the IE flag is set to the interrupt disabled (DI) state, and all maskable interrupt requests are disabled.

When 1, the IE flag is set to the interrupt enabled (EI) state and interrupt request acknowledgment is controlled with an in-service priority flag (ISP1, ISP0), an interrupt mask flag for various interrupt sources, and a priority specification flag.

The IE flag is reset (0) upon DI instruction execution or interrupt acknowledgment and is set (1) upon EI instruction execution.

(b) Zero flag (Z)

When the operation result is zero, this flag is set (1). It is reset (0) in all other cases.

(c) Register bank select flags (RBS0, RBS1)

These are 2-bit flags to select one of the four register banks.

In these flags, the 2-bit information that indicates the register bank selected by SEL RBn instruction execution is stored.

4.2.4 Port 3

Port 3 is a 2-bit I/O port with an output latch. Port 3 can be set to the input mode or output mode in 1-bit units using port mode register 3 (PM3). When the P30 and P31 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 3 (PU3).

This port can also be used for external interrupt request input, timer I/O, and real-time counter correction clock output.

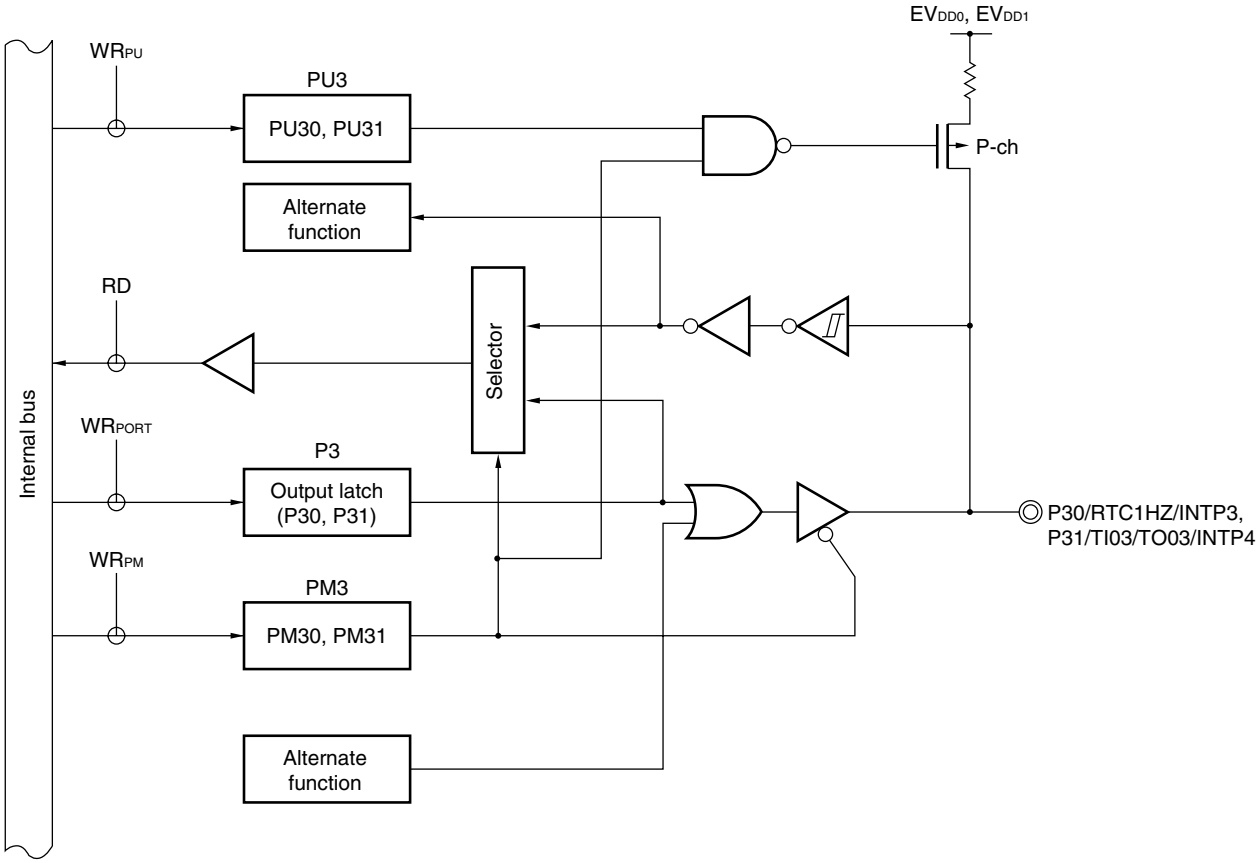
Reset signal generation sets port 3 to input mode.

Figure 4-14 shows block a diagram of port 3.

Cautions 1. To use P31/TI03/TO03/INTP4 as a general-purpose port, set bit 3 (TO03) of timer output register 0 (TO0) and bit 3 (TOE03) of timer output enable register 0 (TOE0) to "0", which is the same as their default status setting.

2. To use P30/RTC1HZ/INTP3 as a general-purpose port, set bit 5 (RCLOE1) of real-time counter control register 0 (RTCC0) to "0", which is the same as its default status setting.

Figure 4-14. Block Diagram of P30 and P31



P3: Port register 3
PU3: Pull-up resistor option register 3
PM3: Port mode register 3
RD: Read signal
WR××: Write signal

Figure 4-40. Format of Port Mode Register

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PM0	1	PM06	PM05	PM04	PM03	PM02	PM01	PM00	FFF20H	FFH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FFF22H	FFH	R/W
PM3	1	1	1	1	1	1	PM31	PM30	FFF23H	FFH	R/W
PM4	PM47	PM46	PM45	PM44	PM43	PM42	PM41	PM40	FFF24H	FFH	R/W
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50	FFF25H	FFH	R/W
PM6	PM67	PM66	PM65	PM64	PM63	PM62	PM61	PM60	FFF26H	FFH	R/W
PM7	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70	FFF27H	FFH	R/W
PM8	PM87	PM86	PM85	PM84	PM83	PM82	PM81	PM80	FFF28H	FFH	R/W
PM11	1	1	1	1	1	1	PM111	PM110	FFF2BH	FFH	R/W
PM12	1	1	1	1	1	1	1	PM120	FFF2CH	FFH	R/W
PM13	1	1	1	1	1	1	PM131	0	FFF2DH	FEH	R/W
PM14	1	1	PM145	PM144	PM143	PM142	PM141	PM140	FFF2EH	FFH	R/W
PM15	PM157	PM156	PM155	PM154	PM153	PM152	PM151	PM150	FFF2FH	FFH	R/W

PMmn	Pmn pin I/O mode selection (m = 0 to 8, 11 to 15; n = 0 to 7)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

Caution Be sure to set bit 7 of PM0, bits 2 to 7 of PM3, bits 2 to 7 of PM11, bits 1 to 7 of PM12, bits 2 to 7 of PM13, and bits 6 and 7 of PM14 to "1". And be sure to set bit 0 of PM13 to "0".

The function of the external bus interface pins differs depending on the setting of the memory extension mode control register (MEM).

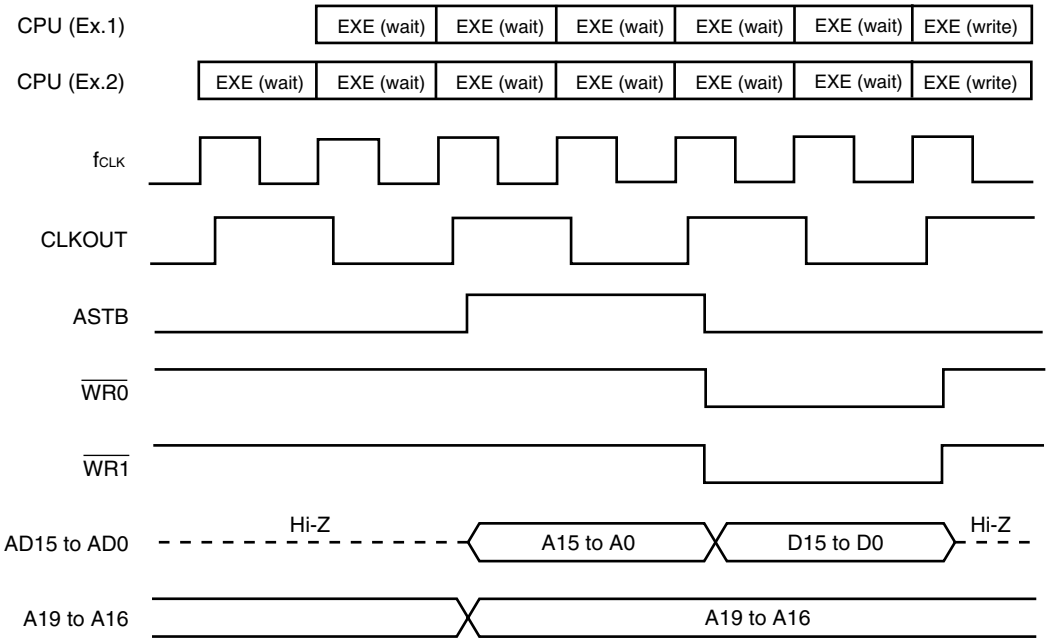
MM3	MM2	MM1	MM0	EX31 to EX28	EX27 to EX24	EX23 to EX20	EX19 to EX16	EX15 to EX12	EX11 to EX8	EX7 to EX0
0	0	0	0	–	–	–	–	–	–	AD7 to AD0
0	0	0	1	–	–	–	–	–	A11 to A8	AD7 to AD0
0	0	1	0	–	–	–	–	A15 to A12	A11 to A8	AD7 to AD0
0	0	1	1	–	–	–	A19 to A16	A15 to A12	A11 to A8	AD7 to AD0
0	1	0	0	–	–	–	–	D15 to D12	D11 to D8	AD7 to AD0
0	1	0	1	–	–	–	–	D15 to D12	AD11 to AD8	AD7 to AD0
0	1	1	0	–	–	–	–	AD15 to AD12	AD11 to AD8	AD7 to AD0
0	1	1	1	–	–	–	A19 to A16	AD15 to AD12	AD11 to AD8	AD7 to AD0
1	0	0	0	–	–	–	–	A7 to A4	A3 to A0	D7 to D0
1	0	0	1	–	–	–	A11 to A8	A7 to A4	A3 to A0	D7 to D0
1	0	1	0	–	–	A15 to A12	A11 to A8	A7 to A4	A3 to A0	D7 to D0
1	0	1	1	–	A19 to A16	A15 to A12	A11 to A8	A7 to A4	A3 to A0	D7 to D0
1	1	0	0	–	–	A7 to A4	A3 to A0	D15 to D12	D11 to D8	D7 to D0
1	1	0	1	–	A11 to A8	A7 to A4	A3 to A0	D15 to D12	D11 to D8	D7 to D0
1	1	1	0	A15 to A12	A11 to A8	A7 to A4	A3 to A0	D15 to D12	D11 to D8	D7 to D0
1	1	1	1	Setting prohibited						

EXEN	EXWEN	MM3	MM2	CLKOUT	ASTB	$\overline{RD}$	$\overline{WR0}$	$\overline{WR1}$	$\overline{WAIT}$
0	X	X	X	–	–	–	–	–	–
1	0	0	0	CLKOUT	ASTB	RD	Write strobe	–	–
1	0	0	1	CLKOUT	ASTB	RD	Low bytes write strobe	High bytes write strobe	–
1	0	1	0	CLKOUT	–	RD	Write strobe	–	–
1	0	1	1	CLKOUT	–	RD	Low bytes write strobe	High bytes write strobe	–
1	1	0	0	CLKOUT	ASTB	RD	Write strobe	–	WAIT
1	1	0	1	CLKOUT	ASTB	RD	Low bytes write strobe	High bytes write strobe	WAIT
1	1	1	0	CLKOUT	–	RD	Write strobe	–	WAIT
1	1	1	1	CLKOUT	–	RD	Low bytes write strobe	High bytes write strobe	WAIT

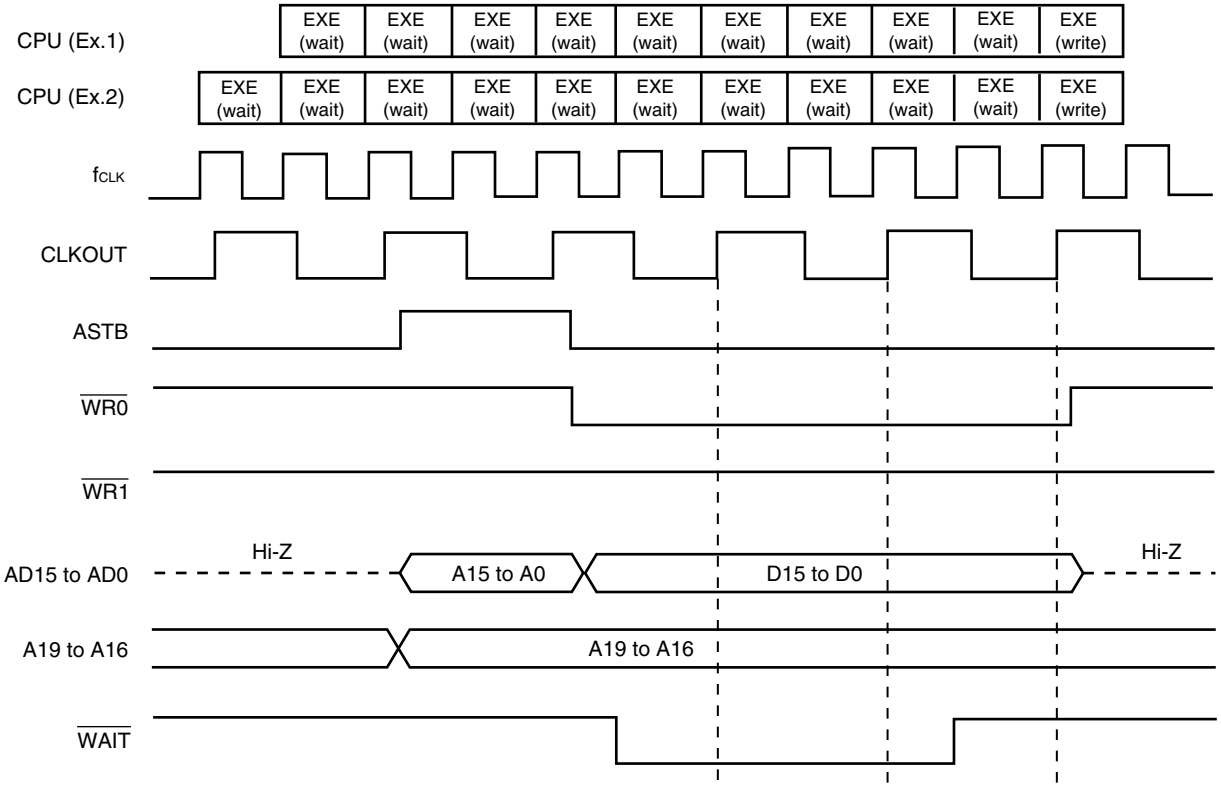
Remark EXxx: Pin name
Axx: Address bus
Dxx: Data bus
ADxx: Multiplexed address/data bus
–: External bus interface is not used. These pins can be used as port pins.

Figure 5-6. Timing to Write to External Memory (2/2)

(c) No wait, 16-bit bus CLKOUT = f_{CLK}/2 (EXWEN = 0, MM3 = 0, MM2 = 1)



(d) With wait, 16-bit bus CLKOUT = f_{CLK}/2 (EXWEN = 1, MM3 = 0, MM2 = 1), lower 8-bit writing



Caution 4. The setting of the flags of the register to stop clock oscillation (invalidate the external clock input) and the condition before clock oscillation is to be stopped are as follows.

Table 6-2. Condition Before Stopping Clock Oscillation and Flag Setting

Clock	Condition Before Stopping Clock (Invalidating External Clock Input)	Setting of CSC Register Flags
X1 clock External main system clock	• CLS = 0 and MCS = 0 • CLS = 1 (CPU and peripheral hardware clocks operate with a clock other than the high-speed system clock.)	MSTOP = 1
Subsystem clock	• CLS = 0 (CPU and peripheral hardware clocks operate with a clock other than the subsystem clock.)	XTSTOP = 1
Internal high-speed oscillation clock	• CLS = 0 and MCS = 1 • CLS = 1 (CPU and peripheral hardware clocks operate with a clock other than the internal high-speed oscillator clock.)	HIOSTOP = 1

(3) Oscillation stabilization time counter status register (OSTC)

This is the register that indicates the count status of the X1 clock oscillation stabilization time counter.

The X1 clock oscillation stabilization time can be checked in the following case,

- If the X1 clock starts oscillation while the internal high-speed oscillation clock or subsystem clock is being used as the CPU clock.
- If the STOP mode is entered and then released while the internal high-speed oscillation clock is being used as the CPU clock with the X1 clock oscillating.

OSTC can be read by a 1-bit or 8-bit memory manipulation instruction.

When reset signal is generated, the STOP instruction and MSTOP (bit 7 of CSC register) = 1 clear OSTC to 00H.

Remark The oscillation stabilization time counter starts counting in the following cases.

- When oscillation of the X1 clock starts (EXCLK, OSCSEL = 0, 1 → MSTOP = 0)
- When the STOP mode is released

Caution When using the X1 oscillator and XT1 oscillator, wire as follows in the area enclosed by the broken lines in the Figures 6-10 and 6-11 to avoid an adverse effect from wiring capacitance.

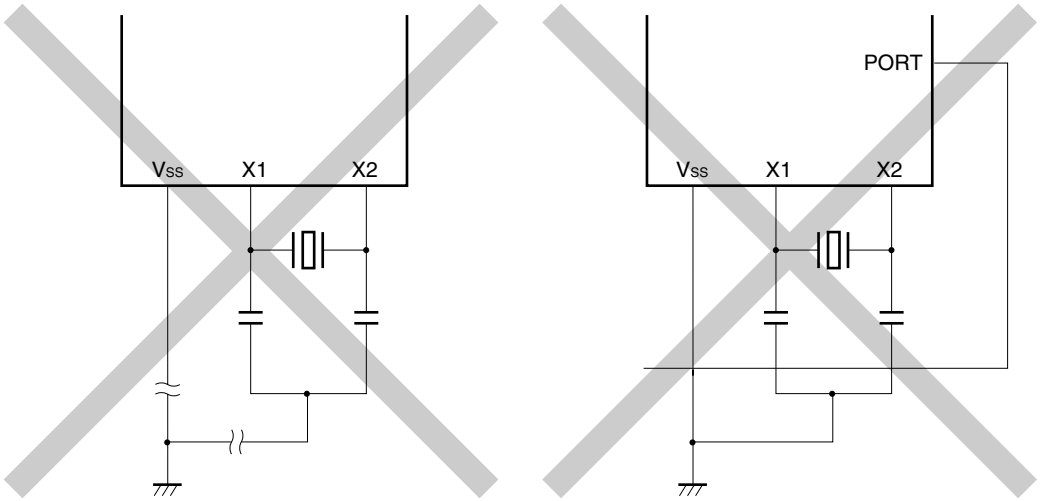
- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as V_{SS}. Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

Note that the XT1 oscillator is designed as a low-amplitude circuit for reducing power consumption.

Figure 6-12 shows examples of incorrect resonator connection.

Figure 6-12. Examples of Incorrect Resonator Connection (1/2)

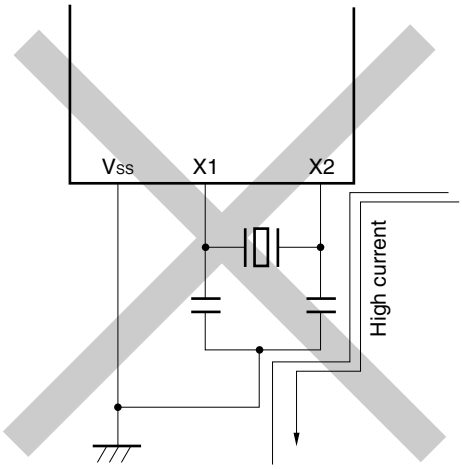
(a) Too long wiring **(b) Crossed signal line**



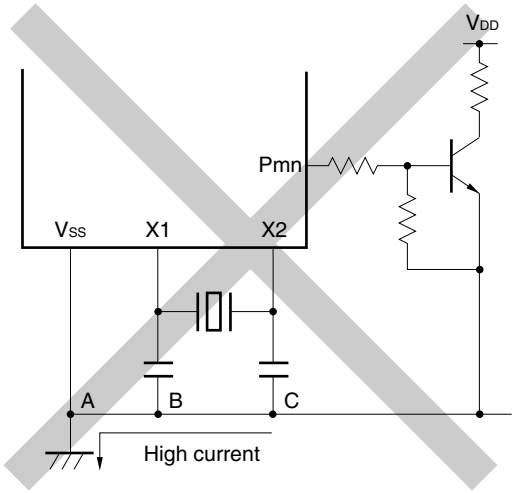
Remark When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

Figure 6-12. Examples of Incorrect Resonator Connection (2/2)

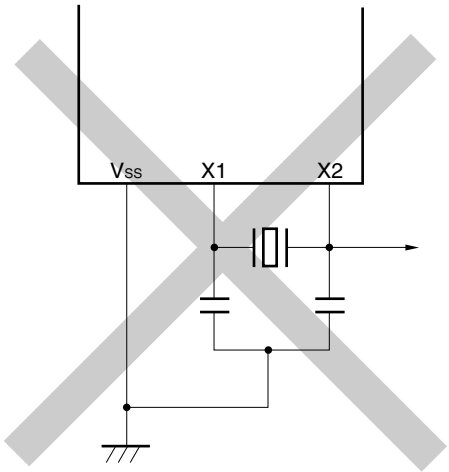
(c) Wiring near high alternating current



(d) Current flowing through ground line of oscillator (potential at points A, B, and C fluctuates)



(e) Signals are fetched



Remark When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

Caution When X2 and XT1 are wired in parallel, the crosstalk noise of X2 may increase with XT1, resulting in malfunctioning.

Table 6-4. CPU Clock Transition and SFR Register Setting Examples (3/4)

(6) CPU clock changing from high-speed system clock (C) to internal high-speed oscillation clock (B)

(Setting sequence of SFR registers) →

Setting Flag of SFR Register / Status Transition	CSC Register	Oscillation accuracy stabilization time	CKC Register
	HIOSTOP		MCM0
(C) → (B)	0	10 μs	0

Unnecessary if these registers are already set

(7) CPU clock changing from high-speed system clock (C) to subsystem clock (D)

(Setting sequence of SFR registers) →

Setting Flag of SFR Register / Status Transition	CMC Register[Note]	CSC Register	Waiting for Oscillation Stabilization	CKC Register
	OSCSELS	XTSTOP		CSS
(C) → (D)	1	0	Necessary	1

Unnecessary if the CPU is operating with the internal high-speed oscillation clock

Note The CMC register can be written only once by an 8-bit memory manipulation instruction after reset release.

(8) CPU clock changing from subsystem clock (D) to internal high-speed oscillation clock (B)

(Setting sequence of SFR registers) →

Setting Flag of SFR Register / Status Transition	CSC Register	CKC Register	
	HIOSTOP	MCM0	CSS
(D) → (B)	0	0	0

Unnecessary if the CPU is operating with the internal high-speed oscillation clock

Unnecessary if this register is already set

Remark (A) to (I) in Table 6-4 correspond to (A) to (I) in Figure 6-15.

Figure 10-2. Format of Clock Output Select Register n (CKSn)

Address: FFFA5H After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
CKSn	PCLOEn	0	0	0	CSELn	CCSn2	CCSn1	CCSn0

PCLOEn	PCLBUZn output enable/disable specification
0	Output disable (default)
1	Output enable

CSELn	CCSn2	CCSn1	CCSn0	PCLBUZn output clock selection			
					f_{MAIN} = 5 MHz	f_{MAIN} = 10 MHz	f_{MAIN} = 20 MHz
0	0	0	0	f_{MAIN}	5 MHz	10 MHz[Note]	Setting prohibited[Note]
0	0	0	1	$f_{MAIN}/2$	2.5 MHz	5 MHz	10 MHz[Note]
0	0	1	0	$f_{MAIN}/2^2$	1.25 MHz	2.5 MHz	5 MHz
0	0	1	1	$f_{MAIN}/2^3$	625 kHz	1.25 MHz	2.5 MHz
0	1	0	0	$f_{MAIN}/2^4$	312.5 kHz	625 kHz	1.25 MHz
0	1	0	1	$f_{MAIN}/2^{11}$	2.44 kHz	4.88 kHz	9.76 kHz
0	1	1	0	$f_{MAIN}/2^{12}$	1.22 kHz	2.44 kHz	4.88 kHz
0	1	1	1	$f_{MAIN}/2^{13}$	610 Hz	1.22 kHz	2.44 kHz
1	0	0	0	f_{SUB}	32.768 kHz		
1	0	0	1	$f_{SUB}/2$	16.384 kHz		
1	0	1	0	$f_{SUB}/2^2$	8.192 kHz		
1	0	1	1	$f_{SUB}/2^3$	4.096 kHz		
1	1	0	0	$f_{SUB}/2^4$	2.048 kHz		
1	1	0	1	$f_{SUB}/2^5$	1.024 kHz		
1	1	1	0	$f_{SUB}/2^6$	512 Hz		
1	1	1	1	$f_{SUB}/2^7$	256 Hz		

Note Setting an output clock exceeding 10 MHz is prohibited when 2.7 V ≤ V_{DD}. Setting a clock exceeding 5 MHz at V_{DD} < 2.7 V is also prohibited.

Cautions 1. Change the output clock after disabling clock output (PCLOEn = 0).

2. If the selected clock (f_{MAIN} or f_{SUB}) stops during clock output (PCLOEn = 1), the output becomes undefined.

Remarks 1. n = 0, 1

2. f_{MAIN}: Main system clock frequency

3. f_{SUB}: Subsystem clock frequency

(1) Register setting

Figure 13-40. Example of Contents of Registers for Master Transmission/Reception of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI20)

(a) Serial output register m (SOm) … Sets only the bits of the target channel.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOm	0	0	0	0	1	CKOm2 0/1	CKOm1 0/1	CKOm0 0/1	0	0	0	0	1	SOm2 0/1	SOm1 0/1	SOm0 0/1

Communication starts when these bits are 1 if the data phase is forward (CKPmn = 0). If the phase is reversed (CKPmn = 1), communication starts when these bits are 0.

(b) Serial output enable register m (SOEm) … Sets only the bits of the target channel to 1.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 0/1	SOEm1 0/1	SOEm0 0/1

(c) Serial channel start register m (SSm) … Sets only the bits of the target channel to 1.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 ×	SSm2 0/1	SSm1 0/1	SSm0 0/1

(d) Serial mode register mn (SMRmn)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMRmn	CKSmn 0/1	CCSmn 0	0	0	0	0	0	STSmn 0	0	SISmn0 0	1	0	0	MDmn2 0	MDmn1 0	MDmn0 0/1

Interrupt sources of channel n
0: Transfer end interrupt
1: Buffer empty interrupt

(e) Serial communication operation setting register mn (SCRmn)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCRmn	TXEmn 1	RXEmn 1	DAPmn 0/1	CKPmn 0/1	0	EOCmn 0	PTCmn1 0	PTCmn0 0	DIRmn 0/1	0	SLCmn1 0	SLCmn0 0	0	DLSmn2 1	DLSmn1 1	DLSmn0 0/1

(f) Serial data register mn (SDRmn) (lower 8 bits: SIOp)

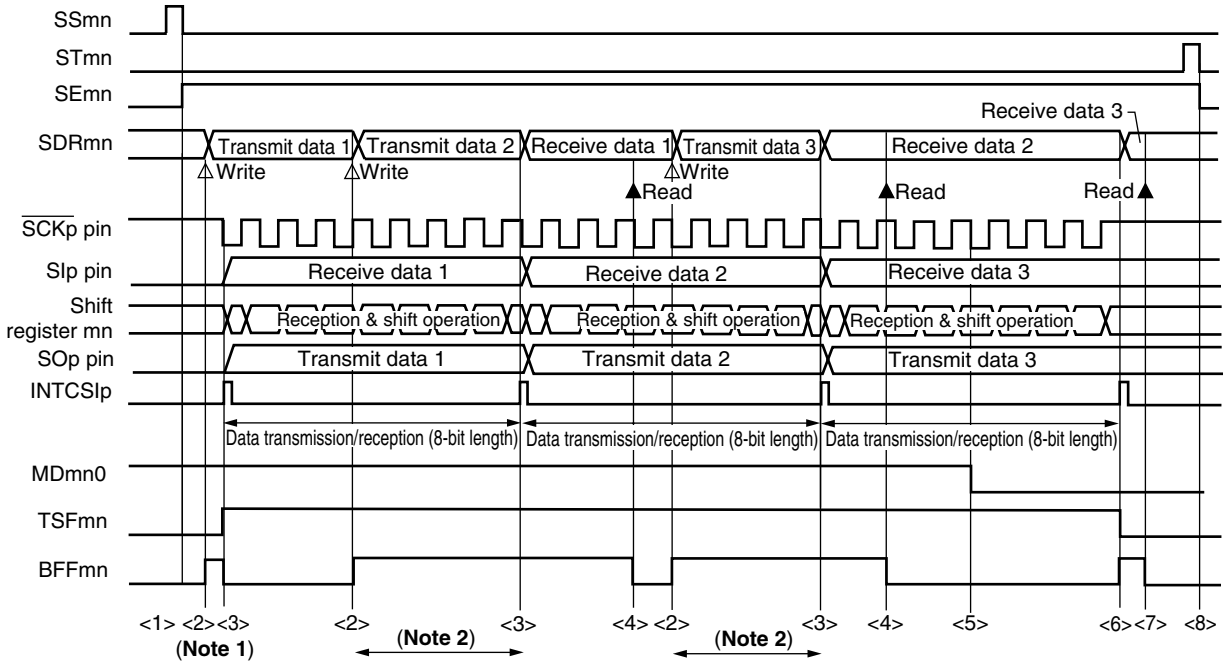
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDRmn	Baud rate setting							0	Transmit data setting/receive data register							

SIOp

Remark m: Unit number (m = 0, 1), n: Channel number (n = 0 to 2), p: CSI number (p = 00, 01, 10, 20)
□: Setting is fixed in the CSI master transmission/reception mode, ■: Setting disabled (set to the initial value)
×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

(4) Processing flow (in continuous transmission/reception mode)

Figure 13-46. Timing Chart of Master Transmission/Reception (in Continuous Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)



Notes 1. When transmit data is written to the SDRmn register while BFFmn = 1, the transmit data is overwritten.

2. The transmit data can be read by reading the SDRmn register during this period. At this time, the transfer operation is not affected.

Caution The MDmn0 bit can be rewritten even during operation.
However, rewrite it before transfer of the last bit is started, so that it has been rewritten before the transfer end interrupt of the last transmit data.

Remarks 1. <1> to <8> in the figure correspond to <1> to <8> in **Figure 13-47 Flowchart of Master Transmission/Reception (in Continuous Transmission/Reception Mode)**.

2. m: Unit number (m = 0, 1), n: Channel number (n = 0 to 2), p: CSI number (p = 00, 01, 10, 20)

13.7.6 Procedure for processing errors that occurred during simplified I²C (IIC10, IIC20) communication

The procedure for processing errors that occurred during simplified I²C (IIC10, IIC20) communication is described in Figures 13-105 and 13-106.

<R> **Figure 13-105. Processing Procedure in Case of Parity Error or Overrun Error**

Software Manipulation	Hardware Status	Remark
Reads SDRmn register. →	▶ BFF = 0, and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads SSRmn register.		Error type is identified and the read value is used to clear error flag.
Writes SIRmn register. →	▶ Error flag is cleared.	Only error generated at the point of reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.

Figure 13-106. Processing Procedure in Case of Parity Error (ACK error) in Simplified I²C Mode

Software Manipulation	Hardware Status	Remark
Reads SDRmn register. →	▶ BFF = 0, and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads SSRmn register.		Error type is identified and the read value is used to clear error flag.
Writes SIRmn register. →	▶ Error flag is cleared.	Only error generated at the point of reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.
Sets STmn bit to 1. →	▶ SEmn = 0, and channel n stops operation.	Slave is not ready for reception because ACK is not returned. Therefore, a stop condition is created, the bus is released, and communication is started again from the start condition. Or, a restart condition is generated and transmission can be redone from address transmission.
Creates stop condition.		
Creates start condition.		
Sets SSmn bit to 1. →	▶ SEmn = 1, and channel n is enabled to operate.	

Remark m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), mn = 02, 10

Figure 14-7. Format of IIC Status Register 0 (IICS0) (2/3)

COI0	Detection of matching addresses
0	Addresses do not match.
1	Addresses match.

Condition for clearing (COI0 = 0)	Condition for setting (COI0 = 1)
• When a start condition is detected • When a stop condition is detected • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • Reset	• When the received address matches the local address (slave address register 0 (SVA0)) (set at the rising edge of the eighth clock).

TRC0	Detection of transmit/receive status
0	Receive status (other than transmit status). The SDA0 line is set for high impedance.
1	Transmit status. The value in the SO0 latch is enabled for output to the SDA0 line (valid starting at the falling edge of the first byte's ninth clock).

Condition for clearing (TRC0 = 0)	Condition for setting (TRC0 = 1)
<Both master and slave> • When a stop condition is detected • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • Cleared by WREL0 = 1[Note] (wait cancel) • When ALD0 changes from 0 to 1 (arbitration loss) • Reset <Master> • When "1" is output to the first byte's LSB (transfer direction specification bit) <Slave> • When a start condition is detected • When "0" is input to the first byte's LSB (transfer direction specification bit) <When not used for communication>	<Master> • When a start condition is generated • When "0" is output to the first byte's LSB (transfer direction specification bit) <Slave> • When "1" is input to the first byte's LSB (transfer direction specification bit)

Note If the wait state is canceled by setting bit 5 (WREL0) of IIC control register 0 (IICC0) to 1 at the ninth clock when bit 3 (TRC0) of IIC status register 0 (IICS0) is 1, TRC0 is cleared, and the SDA0 line goes into a high-impedance state.

Remark LREL0: Bit 6 of IIC control register 0 (IICC0)
IICE0: Bit 7 of IIC control register 0 (IICC0)

Table 14-3. Selection Clock Setting

IICX0	IICCL0			Transfer Clock (f_{CLK}/m)	Settable Selection Clock (f_{CLK}) Range	Operation Mode
Bit 0	Bit 3	Bit 1	Bit 0			
CLX0	SMC0	CL01	CL00			
0	0	0	0	f_{CLK}/88	4.00 MHz to 8.4 MHz	Normal mode (SMC0 bit = 0)
0	0	0	1	f_{CLK}/172	8.38 MHz to 16.76 MHz	
0	0	1	0	f_{CLK}/344	16.76 MHz to 20 MHz	
0	0	1	1	f_{CLK}/44	2.00 MHz to 4.2 MHz	
0	1	0	×	f_{CLK}/48	7.60 MHz to 16.76 MHz	Fast mode (SMC0 bit = 1)
0	1	1	0	f_{CLK}/96	16.00 MHz to 20 MHz	
0	1	1	1	f_{CLK}/24	4.00 MHz to 8.4 MHz	
1	0	×	×	Setting prohibited		
1	1	0	×	f_{CLK}/48	8.00 MHz to 8.38 MHz	Fast mode (SMC0 bit = 1)
1	1	1	0	Setting prohibited	16.00 MHz to 16.76 MHz	
1	1	1	1	f_{CLK}/24	4.00 MHz to 4.19 MHz	

Caution Determine the transfer clock frequency of I^2C by using CLX0, SMC0, CL01, and CL00 before enabling the operation (by setting bit 7 (IICE0) of IIC control register 0 (IICC0) to 1). To change the transfer clock frequency, clear IICE0 once to 0.

Remarks 1. ×: don't care

2. f_{CLK}: CPU/peripheral hardware clock frequency

14.5.5 Acknowledge ($\overline{ACK}$)

$\overline{ACK}$ is used to check the status of serial data at the transmission and reception sides.

The reception side returns $\overline{ACK}$ each time it has received 8-bit data.

The transmission side usually receives $\overline{ACK}$ after transmitting 8-bit data. When $\overline{ACK}$ is returned from the reception side, it is assumed that reception has been correctly performed and processing is continued. Whether $\overline{ACK}$ has been detected can be checked by using bit 2 (ACKD0) of IIC status register 0 (IICS0).

When the master receives the last data item, it does not return $\overline{ACK}$ and instead generates a stop condition. If a slave does not return $\overline{ACK}$ after receiving data, the master outputs a stop condition or restart condition and stops transmission. If $\overline{ACK}$ is not returned, the possible causes are as follows.

<1> Reception was not performed normally.

<2> The final data item was received.

<3> The reception side specified by the address does not exist.

To generate $\overline{ACK}$, the reception side makes the SDA0 line low at the ninth clock (indicating normal reception).

Automatic generation of $\overline{ACK}$ is enabled by setting bit 2 (ACKE0) of IIC control register 0 (IICC0) to 1. Bit 3 (TRC0) of the IICS0 register is set by the data of the eighth bit that follows 7-bit address information. Usually, set ACKE0 to 1 for reception (TRC0 = 0).

If a slave can receive no more data during reception (TRC0 = 0) or does not require the next data item, then the slave must inform the master, by clearing ACKE0 to 0, that it will not receive any more data.

When the master does not require the next data item during reception (TRC0 = 0), it must clear ACKE0 to 0 so that $\overline{ACK}$ is not generated. In this way, the master informs a slave at the transmission side that it does not require any more data (transmission will be stopped).

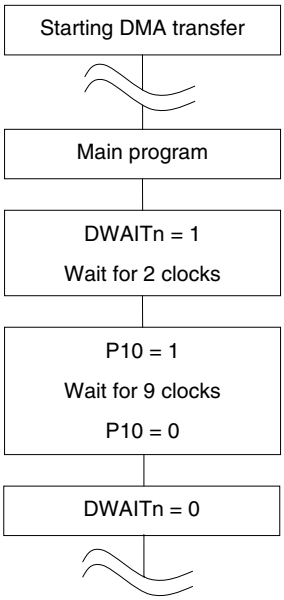
16.5.6 Holding DMA transfer pending by DWAITn

<R> When DMA transfer is started, transfer is performed while an instruction is executed. At this time, the operation of the CPU is stopped and delayed for the duration of 2 clocks. If this poses a problem to the operation of the set system, a DMA transfer can be held pending by setting DWAITn to 1. The DMA transfer for a transfer trigger that occurred while DMA transfer was held pending is executed after the pending status is canceled. However, because only one transfer trigger can be held pending for each channel, even if multiple transfer triggers occur for one channel during the pending status, only one DMA transfer is executed after the pending status is canceled.

To output a pulse with a width of 10 clocks of the operating frequency from the P00 pin, for example, the clock width increases to 12 if a DMA transfer is started midway. In this case, the DMA transfer can be held pending by setting DWAITn to 1.

After setting DWAITn to 1, it takes two clocks until a DMA transfer is held pending.

Figure 16-12. Example of Setting for Holding DMA Transfer Pending by DWAITn



<R> **Caution When DMA transfer is held pending while using both DMA channels, be sure to hold the DMA transfer pending for both channels (by setting DWAIT0 and DWAIT1 to 1). If the DMA transfer of one channel is executed while that of the other channel is held pending, DMA transfer might not be held pending for the latter channel.**

Remarks 1. n: DMA channel number (n = 0, 1)

2. 1 clock: $1/f_{CLK}$ (f_{CLK}: CPU clock)

(4) External interrupt rising edge enable registers (EGP0, EGP1), external interrupt falling edge enable registers (EGN0, EGN1)

These registers specify the valid edge for INTP0 to INTP11.

EGP0, EGP1, EGN0, and EGN1 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

Figure 17-5. Format of External Interrupt Rising Edge Enable Registers (EGP0, EGP1) and External Interrupt Falling Edge Enable Registers (EGN0, EGN1)

Address: FFF38H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGP0	EGP7	EGP6	EGP5	EGP4	EGP3	EGP2	EGP1	EGP0

Address: FFF39H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGN0	EGN7	EGN6	EGN5	EGN4	EGN3	EGN2	EGN1	EGN0

Address: FFF3AH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGP1	0	0	0	0	EGP11	EGP10	EGP9	EGP8

Address: FFF3BH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGN1	0	0	0	0	EGN11	EGN10	EGN9	EGN8

EGPn	EGNn	INTPn pin valid edge selection (n = 0 to 11)
0	0	Edge detection disabled
0	1	Falling edge
1	0	Rising edge
1	1	Both rising and falling edges

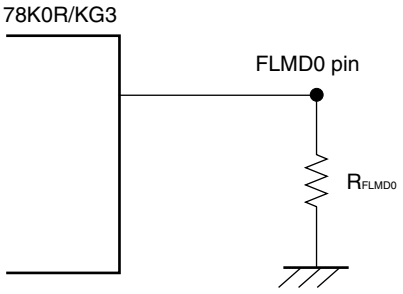
Table 17-3 shows the ports corresponding to EGPn and EGNn.

DC Characteristics (7/16)

(T_A = –40 to +85°C, 1.8 V ≤ V_{DD} = EV_{DD0} = EV_{DD1} ≤ 5.5 V, 1.8 V ≤ AV_{REF0} ≤ V_{DD}, 1.8 V ≤ AV_{REF1} ≤ V_{DD}, V_{SS} = EV_{SS0} = EV_{SS1} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
On-chip pull-up resistance	R_U	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P120, P131, P140 to P145	V_I = V_{SS}, In input port	10	20	100	kΩ
FLMD0 pin external pull-down resistance[Note]	R_{FLMD0}	When enabling the self-programming mode setting with software		100			kΩ

Note It is recommended to leave the FLMD0 pin open. If the pin is required to be pulled down externally, set R_{FLMD0} to 100 kΩ or more.



Remark Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

Chapter	Classification	Function	Details of Function	Cautions	Page
Chapter 22	Soft	Low-voltage detector	Used as interrupt (when detecting level of supply voltage (V_{DD})) (LVIOFF = 0)	Even when the LVI default start function is used, if it is set to LVI operation prohibition by the software, it operates as follows: • Does not perform low-voltage detection during LVION = 0. • If a reset is generated while LVION = 0, LVION will be re-set to 1 when the CPU starts after reset release. There is a period when low-voltage detection cannot be performed normally, however, when a reset occurs due to WDT and illegal instruction execution. This is due to the fact that while the pulse width detected by LVI must be 200 μs max., LVION = 1 is set upon reset occurrence, and the CPU starts operating without waiting for the LVI stabilization time.	p.702 □
				When the LVI default start function (bit 0 (LVIOFF) of 000C1H = 0) is used, the LVIRF flag may become 1 from the beginning due to the power-on waveform. For details of RESF, see CHAPTER 20 RESET FUNCTION.	p.702 □
	Hard		Used as interrupt (when detecting level of input voltage from external input pin (EXLVI))	The input voltage from the external input pin (EXLVI) must be EXLVI < V_{DD}.	p.704 □
	Soft		Cautions for low-voltage detector	In a system where the supply voltage (V_{DD}) fluctuates for a certain period in the vicinity of the LVI detection voltage (V_{LVI}), the operation is as follows depending on how the low-voltage detector is used. Operation example 1: When used as reset The system may be repeatedly reset and released from the reset status. The time from reset release through microcontroller operation start can be set arbitrarily by the following action. <Action> After releasing the reset signal, wait for the supply voltage fluctuation period of each system by means of a software counter that uses a timer, and then initialize the ports (see Figure 22-11). Operation example 2: When used as interrupt Interrupt requests may be generated frequently. Take the following action. <Action> Confirm that "supply voltage (V_{DD}) ≥ detection voltage (V_{LVI})" when detecting the falling edge of V_{DD}, or "supply voltage (V_{DD}) < detection voltage (V_{LVI})" when detecting the rising edge of V_{DD}, in the servicing routine of the LVI interrupt by using bit 0 (LVIF) of the low-voltage detection register (LVIM). Clear bit 1 (LVIIF) of interrupt request flag register 0L (IF0L) to 0. For a system with a long supply voltage fluctuation period near the LVI detection voltage, take the above action after waiting for the supply voltage fluctuation time.	pp.706 to 709 □
	Hard			There is some delay from the time supply voltage (V_{DD}) < LVI detection voltage (V_{LVI}) until the time LVI reset has been generated. In the same way, there is also some delay from the time LVI detection voltage (V_{LVI}) ≤ supply voltage (V_{DD}) until the time LVI reset has been released (see Figure 22-12). See the timing in Figure 21-2 (2) When LVI is ON upon power application (option byte: LVIOFF = 0) for the reset processing time until the normal operation is entered after the LVI reset is released.	p.709 □

(15/24)

Edition	Description	Chapter
5th edition	Change of **23.1 Regulator Overview**	CHAPTER 23 REGULATOR
	Addition of **Note 3** to **Figure 23-1 Format of Regulator Mode Control Register (RMC)**	
	Change of description in **24.1.1 (2) 000C1H/010C1H**	CHAPTER 24 OPTION BYTE
	Change of **Figure 24-2 Format of User Option Byte (000C1H/010C1H)** and **Caution 2**	
	Change of description in **25.4.5 REGC pin**	CHAPTER 25 FLASH MEMORY
	Addition of **Caution 4** to **25.8 Flash Memory Programming by Self-Programming**	
	Addition of **26.3 Securing of user resources**	CHAPTER 26 ON-CHIP DEBUGGING
	Throughout modification	CHAPTER 29 ELECTRICAL SPECIFICATIONS (TARGET)
6th edition	Change of status of μPD78F1162, 78F1163, 78F1164, 78F1165, and 78F1166 from under development to mass production	Throughout
	Change of **2.2.22 FLMD0**	CHAPTER 2 PIN FUNCTIONS
	Addition of register and **Note** in **Table 3-5 SFR List**	CHAPTER 3 CPU ARCHITECTURE
	Change of **Caution** on pin used for the serial array unit or the timer array unit	CHAPTER 4 PORT FUNCTIONS
	Addition of PIM register and POM register in block diagram	
	Addition of descriptions to **4.3 (4) Port input mode registers (PIM0, PIM4, PIM14)** and **(5) Port output mode registers (POM0, POM4, POM14)**	
	Addition of description to **5.1 Functions of External Bus Interface**	CHAPTER 5 EXTERNAL BUS INTERFACE
	Change of **(b)** and **(d)** in **Figure 5-6 Timing to Read External Memory**	
	Addition of description to title of **Figure 5-7 Timing to Write to External Memory (c)**	
	Addition of **Caution 5** to **Figure 6-8 Format of Operation Speed Mode Control Register (OSMC)**	CHAPTER 6 CLOCK GENERATOR
	Change of **Table 7-1 Configuration of Timer Array Unit**	CHAPTER 7 TIMER ARRAY UNIT
	Addition of description to **7.3 (10) Timer output register 0 (TO0)**	
	Addition of description to **8.3 (15) Alarm hour register (ALARMWH)**	CHAPTER 8 REAL-TIME COUNTER
	Change of SOm register	CHAPTER 13 SERIAL ARRAY UNIT
	Change of **Figure 13-1 Block Diagram of Serial Array Unit 0**	
	Change of **Figure 13-2 Block Diagram of Serial Array Unit 1**	
	Change of description in **13.3 (12) Serial output register m (SOm)**	
	Change of **Figure 13-15 Format of Serial Output Register m (SOm)**	
	Addition of **13.4 Operation Stop Mode**	
	Change of **Figure 13-50 Timing Chart of Slave Transmission (in Single-Transmission Mode)**	
	Change of **Figure 13-64 Timing Chart of Slave Transmission/Reception (in Single-Transmission/Reception Mode)**	
	Change of setting in **(a) Serial output register m (SOm)** and **(b) Serial output enable register m (SOEm)** in **Figure 13-76 Example of Contents of Registers for UART Reception of UART (UART0, UART1, UART2, UART3)**	