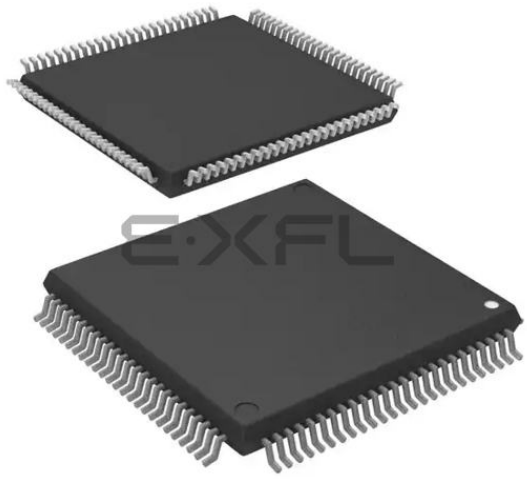
Renesas Electronics America Inc - UPD78F1163AGC-UEU-AX Datasheet



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	96KB (96K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	6K 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/upd78f1163agc-ueu-ax

(j) CLKOUT

This is an internal system clock output pin.

(e) $\overline{\text{WAIT}}$

This is an external wait signal input pin.

Caution To use P02/SO10/TxD1 and P04/$\overline{\text{SCK10}}$/SCL10 as general-purpose ports, set serial communication operation setting register 02 (SCR02) to the default status (0087H). In addition, clear port output mode register 0 (POM0) to 00H.

2.2.2 P10 to P17 (port 1)

P10 to P17 function as an 8-bit I/O port. These pins also function as external interrupt request input, serial interface data I/O, clock I/O, timer I/O, real-time counter clock output, and external expansion output.

The following operation modes can be specified in 1-bit units.

(1) Port mode

P10 to P17 function as an 8-bit I/O port. P10 to P17 can be set to input or output port in 1-bit units using port mode register 1 (PM1). Use of an on-chip pull-up resistor can be specified by pull-up resistor option register 1 (PU1).

(2) Control mode

P10 to P17 function as external interrupt request input, serial interface data I/O, clock I/O, timer I/O, real-time counter clock output, and external expansion output.

(a) SI00

This is a serial data input pin of serial interface CSI00.

(b) SO00

This is a serial data output pin of serial interface CSI00.

(c) $\overline{\text{SCK00}}$

This is a serial clock I/O pin of serial interface CSI00.

(d) RxD0

This is a serial data input pin of serial interface UART0.

(e) RxD3

This is a serial data input pin of serial interface UART3.

(f) TxD0

This is a serial data output pin of serial interface UART0.

(g) TxD3

This is a serial data output pin of serial interface UART3.

(h) TI01, TI02

These are the pins for inputting an external count clock/capture trigger to 16-bit timers 01 and 02.

Table 3-6. Extended SFR (2nd SFR) List (1/5)

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
F0017H	A/D port configuration register	ADPC	R/W	–	√	–	10H
F0030H	Pull-up resistor option register 0	PU0	R/W	√	√	–	00H
F0031H	Pull-up resistor option register 1	PU1	R/W	√	√	–	00H
F0033H	Pull-up resistor option register 3	PU3	R/W	√	√	–	00H
F0034H	Pull-up resistor option register 4	PU4	R/W	√	√	–	00H
F0035H	Pull-up resistor option register 5	PU5	R/W	√	√	–	00H
F0036H	Pull-up resistor option register 6	PU6	R/W	√	√	–	00H
F0037H	Pull-up resistor option register 7	PU7	R/W	√	√	–	00H
F0038H	Pull-up resistor option register 8	PU8	R/W	√	√	–	00H
F003CH	Pull-up resistor option register 12	PU12	R/W	√	√	–	00H
F003DH	Pull-up resistor option register 13	PU13	R/W	√	√	–	00H
F003EH	Pull-up resistor option register 14	PU14	R/W	√	√	–	00H
F0040H	Port input mode register 0	PIM0	R/W	√	√	–	00H
F0044H	Port input mode register 4	PIM4	R/W	√	√	–	00H
F004EH	Port input mode register 14	PIM14	R/W	√	√	–	00H
F0050H	Port output mode register 0	POM0	R/W	√	√	–	00H
F0054H	Port output mode register 4	POM4	R/W	√	√	–	00H
F005EH	Port output mode register 14	POM14	R/W	√	√	–	00H
F0060H	Noise filter enable register 0	NFEN0	R/W	√	√	–	00H
F0061H	Noise filter enable register 1	NFEN1	R/W	√	√	–	00H
F00F0H	Peripheral enable register 0	PER0	R/W	√	√	–	00H
F00F1H	Peripheral enable register 1	PER1	R/W	√	√	–	00H
F00F2H	Internal high-speed oscillator trimming register	HIOTRM	R/W	–	√	–	10H
F00F3H	Operation speed mode control register	OSMC	R/W	–	√	–	00H
F00F4H	Regulator mode control register	RMC	R/W	–	√	–	00H
F00FEH	BCD adjust result register	BCDADJ	R	–	√	–	Undefined
F0100H	Serial status register 00	SSR00L SSR00	R	–	√	√	0000H
F0101H		–		–	–		
F0102H	Serial status register 01	SSR01L SSR01	R	–	√	√	0000H
F0103H		–		–	–		
F0104H	Serial status register 02	SSR02L SSR02	R	–	√	√	0000H
F0105H		–		–	–		
F0106H	Serial status register 03	SSR03L SSR03	R	–	√	√	0000H
F0107H		–		–	–		
F0108H	Serial flag clear trigger register 00	SIR00L SIR00	R/W	–	√	√	0000H
F0109H		–		–	–		
F010AH	Serial flag clear trigger register 01	SIR01L SIR01	R/W	–	√	√	0000H
F010BH		–		–	–		
F010CH	Serial flag clear trigger register 02	SIR02L SIR02	R/W	–	√	√	0000H
F010DH		–		–	–		
F010EH	Serial flag clear trigger register 03	SIR03L SIR03	R/W	–	√	√	0000H
F010FH		–		–	–		

3.4.6 Register indirect addressing

[Function]

Register indirect addressing directly specifies the target addresses using the contents of the register pair specified with the instruction word as an operand address.

[Operand format]

Identifier	Description
–	[DE], [HL] (only the space from F0000H to FFFFFH is specifiable)
–	ES:[DE], ES:[HL] (higher 4-bit addresses are specified by the ES register)

Figure 3-33. Example of [DE], [HL]

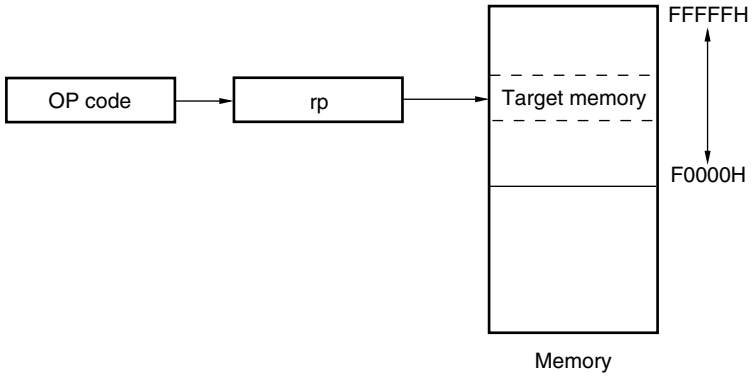
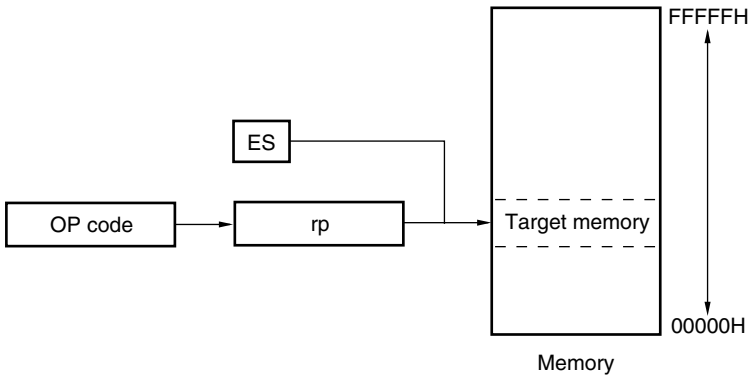


Figure 3-34. Example of ES:[DE], ES:[HL]



3.4.9 Stack addressing

[Function]

The stack area is indirectly addressed with the stack pointer (SP) contents. This addressing is automatically employed when the PUSH, POP, subroutine call, and return instructions are executed or the register is saved/restored upon generation of an interrupt request.

Stack addressing is applied only to the internal RAM area.

[Operand format]

Identifier	Description
–	PUSH AX/BC/DE/HL POP AX/BC/DE/HL CALL/CALLT RET BRK RETB (Interrupt request generated) RETI

Figure 7-41. Example of Set Contents of Registers in External Event Counter Mode

(a) Timer mode register 0n (TMR0n)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n	CKS0n			CCS0n	MASTER0n	STS0n2	STS0n1	STS0n0	CIS0n1	CIS0n0			MD0n3	MD0n2	MD0n1	MD0n0
	1/0	0	0	1	0	0	0	0	1/0	1/0	0	0	0	1	1	0

Operation mode of channel n
011B: Event count mode

Setting of operation when counting is started
0: Neither generates INTTM0n nor inverts timer output when counting is started.

Selection of TI0n pin input edge
00B: Detects falling edge.
01B: Detects rising edge.
10B: Detects both edges.
11B: Setting prohibited

Start trigger selection
000B: Selects only software start.

Slave/master selection
0: Cleared to 0 when single-operation function is selected.

Count clock selection
1: Selects the TI0n pin input valid edge.

Operation clock selection
0: Selects CK00 as operation clock of channel n.
1: Selects CK01 as operation clock of channel n.

(b) Timer output register 0 (TO0)

	Bit n
TO0	TO0n
	0

0: Outputs 0 from TO0n.

(c) Timer output enable register 0 (TOE0)

	Bit n
TOE0	TOE0n
	0

0: Stops the TO0n output operation by counting operation.

(d) Timer output level register 0 (TOL0)

	Bit n
TOL0	TOL0n
	0

0: Cleared to 0 when TOM0n = 0 (toggle mode).

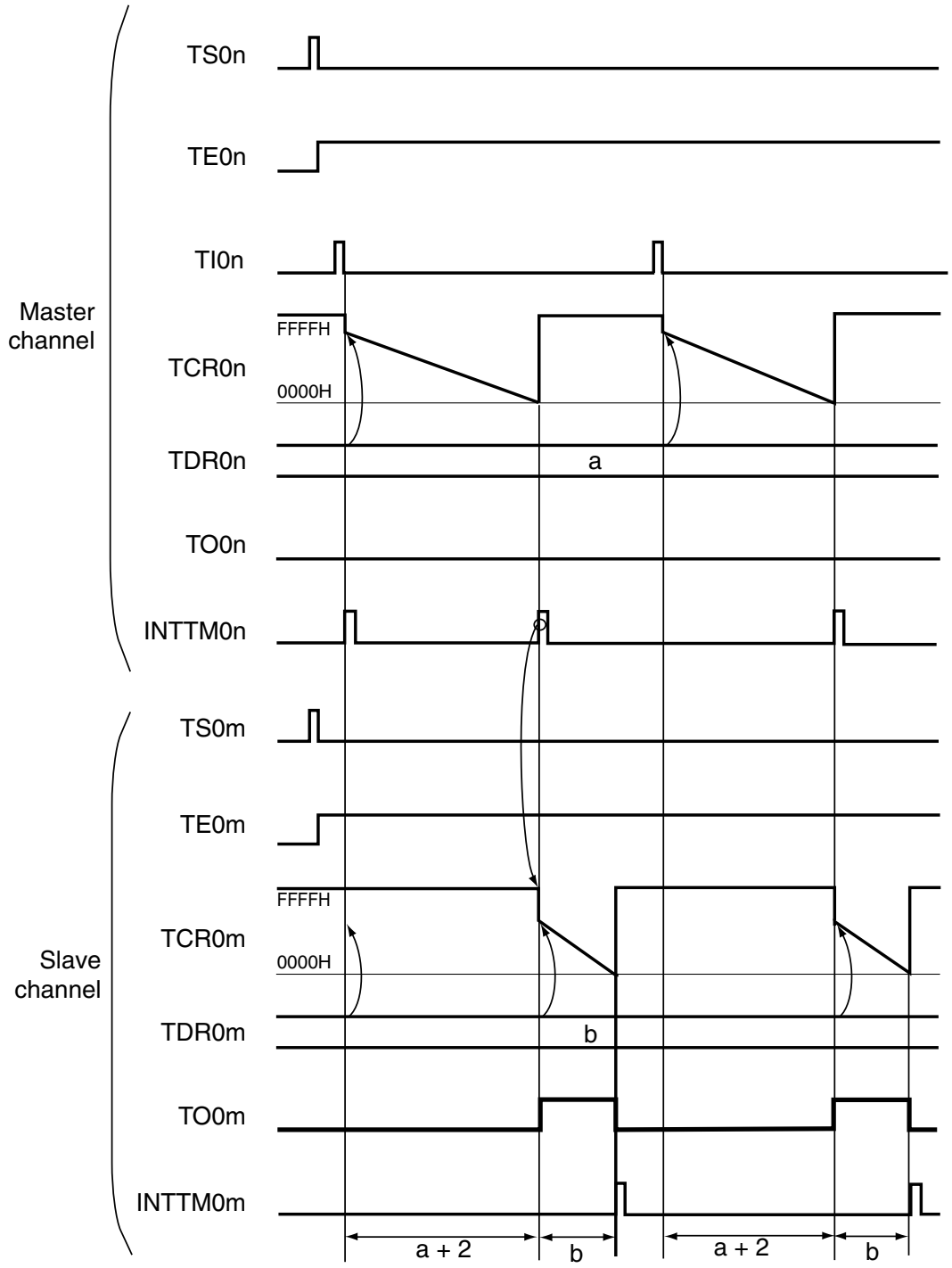
(e) Timer output mode register 0 (TOM0)

	Bit n
TOM0	TOM0n
	0

0: Sets toggle mode.

Remark n = 0 to 7

Figure 7-61. Example of Basic Timing of Operation as One-Shot Pulse Output Function



Remark n = 0, 2, 4, 6
m = n + 1

9.2 Configuration of Watchdog Timer

The watchdog timer includes the following hardware.

Table 9-1. Configuration of Watchdog Timer

Item	Configuration
Control register	Watchdog timer enable register (WDTE)

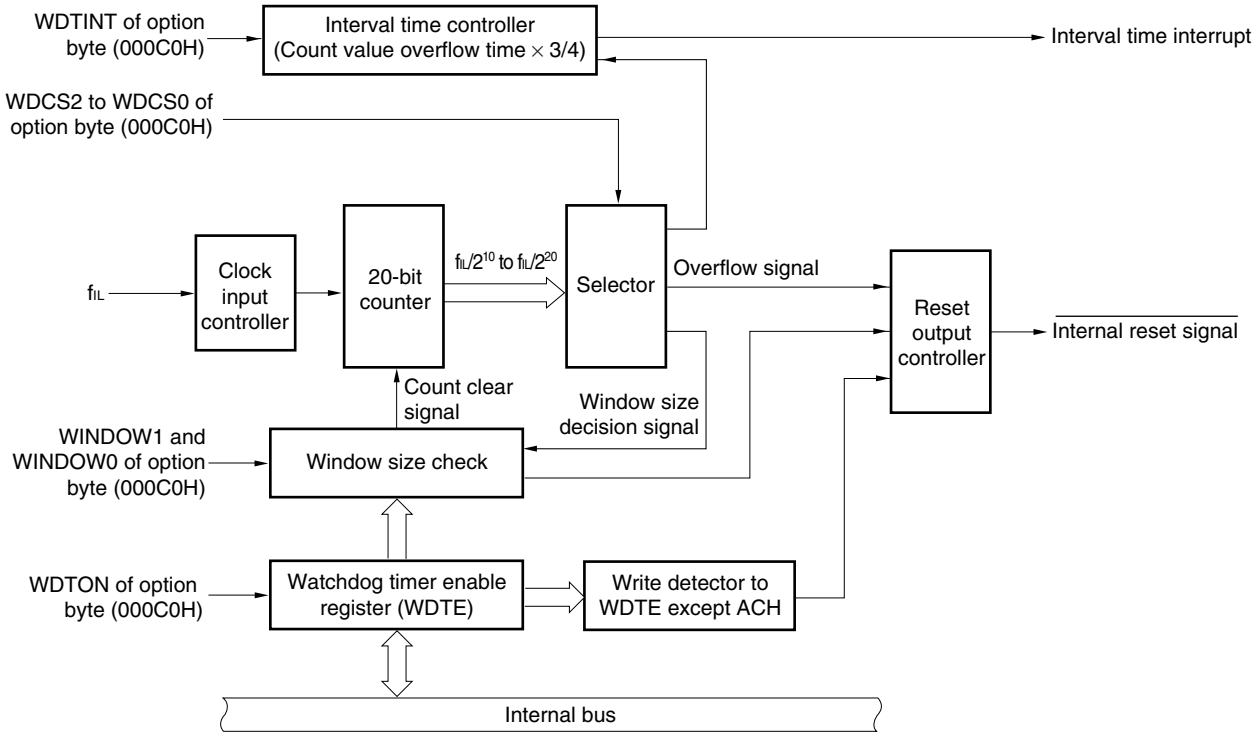
How the counter operation is controlled, overflow time, window open period, and interval interrupt are set by the option byte.

Table 9-2. Setting of Option Bytes and Watchdog Timer

Setting of Watchdog Timer	Option Byte (000C0H)
Watchdog timer interval interrupt	Bit 7 (WDTINT)
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)
Controlling counter operation of watchdog timer (in HALT/STOP mode)	Bit 0 (WDSTBYON)

Remark For the option byte, see **CHAPTER 24 OPTION BYTE**.

Figure 9-1. Block Diagram of Watchdog Timer



Remark f_{IL}: Internal low-speed oscillation clock frequency

(12) Internal equivalent circuit

The equivalent circuit of the analog input block is shown below.

Figure 11-28. Internal Equivalent Circuit of ANIn Pin

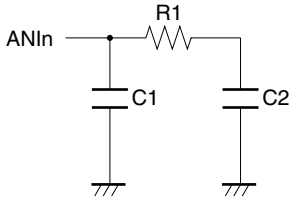


Table 11-6. Resistance and Capacitance Values of Equivalent Circuit (Reference Values)

AV_{REF0}	R1	C1	C2
4.0 V ≤ V_{DD} ≤ 5.5 V	8.1 kΩ	8 pF	5 pF
2.7 V ≤ V_{DD} < 4.0 V	31 kΩ	8 pF	5 pF
2.3 V ≤ V_{DD} < 2.7 V	381 kΩ	8 pF	5 pF

Remarks 1. The resistance and capacitance values shown in Table 11-6 are not guaranteed values.

2. n = 0 to 15

<R> **(13) Starting the A/D converter**

Start the A/D converter after the AV_{REF0} and AV_{REF1} voltages (the reference voltages for the D/A converter) stabilize.

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).

16.6 Cautions on Using DMA Controller

<R> **(1) Priority of DMA**

During DMA transfer, a request from the other DMA channel is held pending even if generated. The pending DMA transfer is started after the ongoing DMA transfer is completed. If two DMA requests are generated at the same time, however, DMA channel 0 takes priority over DMA channel 1.

If a DMA request and an interrupt request are generated at the same time, the DMA transfer takes precedence, and then interrupt servicing is executed.

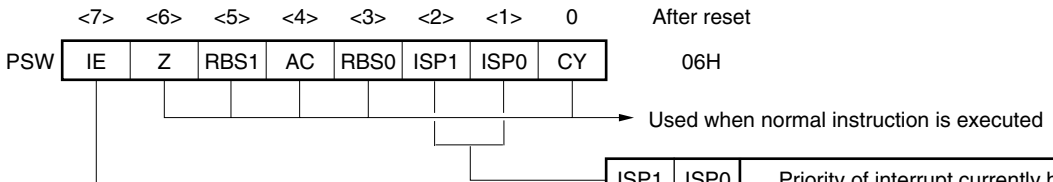
(5) Program status word (PSW)

The program status word is a register used to hold the instruction execution result and the current status for an interrupt request. The IE flag that sets maskable interrupt enable/disable and the ISP0 and ISP1 flags that controls multiple interrupt servicing are mapped to the PSW.

Besides 8-bit read/write, this register can carry out operations using bit manipulation instructions and dedicated instructions (EI and DI). When a vectored interrupt request is acknowledged, if the BRK instruction is executed, the contents of the PSW are automatically saved into a stack and the IE flag is reset to 0. If a maskable interrupt request is acknowledged, the contents of the priority specification flag of the acknowledged interrupt are transferred to the ISP0 and ISP1 flags. The PSW contents are also saved into the stack with the PUSH PSW instruction. They are restored from the stack with the RETI, RETB, and POP PSW instructions.

Reset signal generation sets PSW to 06H.

Figure 17-6. Configuration of Program Status Word



ISP1	ISP0	Priority of interrupt currently being serviced
0	0	Enables interrupt of level 0 (while interrupt of level 1 or 0 is being serviced).
0	1	Enables interrupt of level 0 and 1 (while interrupt of level 2 is being serviced).
1	0	Enables interrupt of level 0 to 2 (while interrupt of level 3 is being serviced).
1	1	Enables all interrupts (waits for acknowledgment of an interrupt).

IE	Interrupt request acknowledgment enable/disable
0	Disabled
1	Enabled

19.2 Standby Function Operation

19.2.1 HALT mode

(1) HALT mode

The HALT mode is set by executing the HALT instruction. HALT mode can be set regardless of whether the CPU clock before the setting was the high-speed system clock, internal high-speed oscillation clock, or subsystem clock.

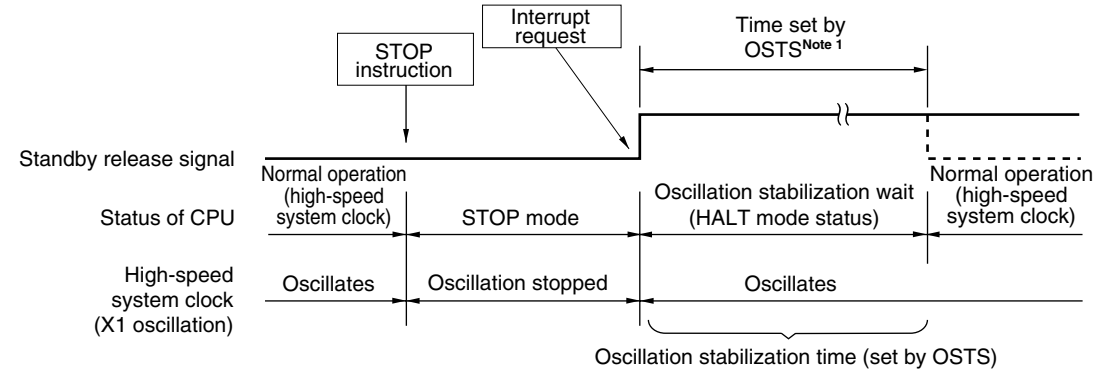
The operating statuses in the HALT mode are shown below.

(a) Release by unmasked interrupt request

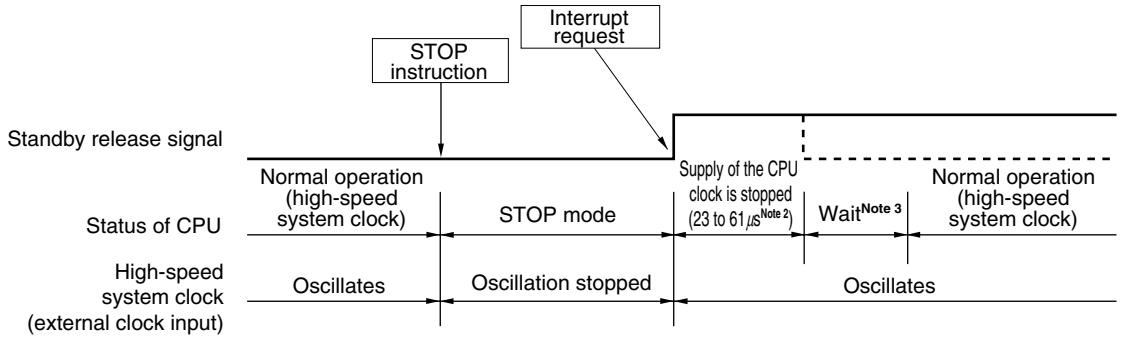
When an unmasked interrupt request is generated, the STOP mode is released. After the oscillation stabilization time has elapsed, if interrupt acknowledgment is enabled, vectored interrupt servicing is carried out. If interrupt acknowledgment is disabled, the next address instruction is executed.

Figure 19-6. STOP Mode Release by Interrupt Request Generation (1/2)

(1) When high-speed system clock (X1 oscillation) is used as CPU clock



(2) When high-speed system clock (external clock input) is used as CPU clock



Notes 1. When the oscillation stabilization time set by OSTS is equal to or shorter than 61 μs, the HALT status is retained to a maximum of "61 μs + wait time".

2. When $f_{CLK} = f_{EX}$

3. The wait time is as follows:

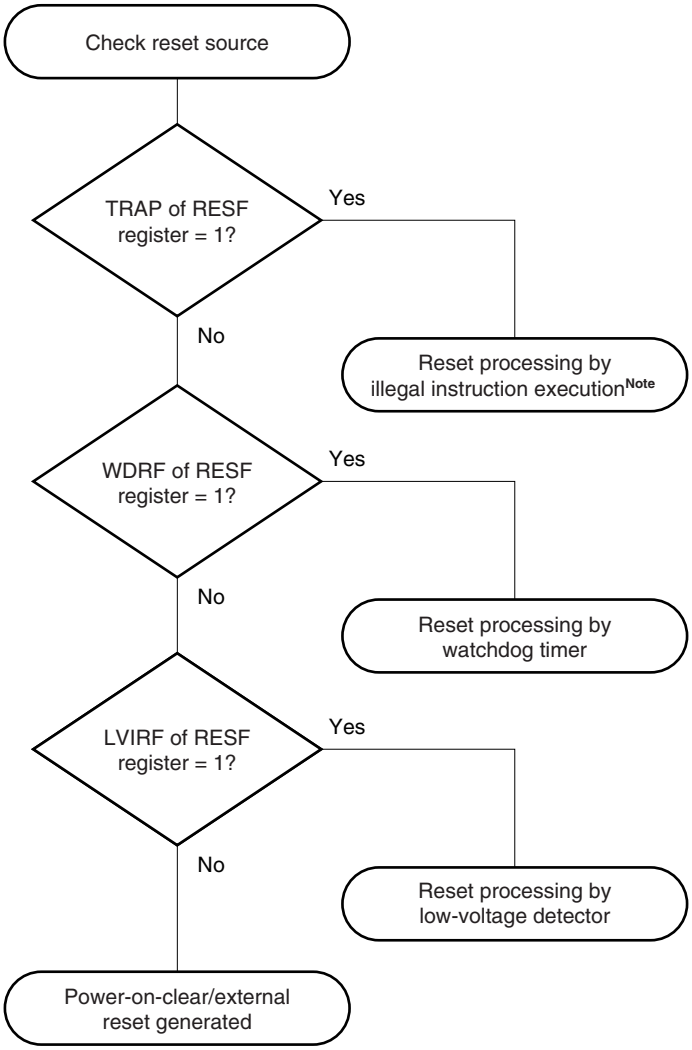
- When vectored interrupt servicing is carried out: 10 to 12 clocks
- When vectored interrupt servicing is not carried out: 5 or 6 clocks

Remarks 1. The broken lines indicate the case when the interrupt request that has released the standby mode is acknowledged.

2. f_{EX}: External main system clock frequency
f_{CLK}: CPU/peripheral hardware clock frequency

Figure 21-3. Example of Software Processing After Reset Release (2/2)

• Checking reset source



Note When instruction code FFH is executed.
Reset by the illegal instruction execution not issued by emulation with the in-circuit emulator or on-chip debug emulator.

Table 25-7. Relationship Between Enabling Security Function and Command

(1) During on-board/off-board programming

Valid Security	Executed Command		
	Batch Erase (Chip Erase)	Block Erase	Write
Prohibition of batch erase (chip erase)	Cannot be erased in batch	Blocks cannot be erased.	Can be performed[Note].
Prohibition of block erase	Can be erased in batch.		Can be performed.
Prohibition of writing			Cannot be performed.
Prohibition of rewriting boot cluster 0	Cannot be erased in batch	Boot cluster 0 cannot be erased.	Boot cluster 0 cannot be written.

Note Confirm that no data has been written to the write area. Because data cannot be erased after batch erase (chip erase) is prohibited, do not write data if the data has not been erased.

(2) During self programming

Valid Security	Executed Command	
	Block Erase	Write
Prohibition of batch erase (chip erase)	Blocks can be erased.	Can be performed.
Prohibition of block erase		
Prohibition of writing		
Prohibition of rewriting boot cluster 0	Boot cluster 0 cannot be erased.	Boot cluster 0 cannot be written.

Remark To prohibit writing and erasing during self-programming, use the flash shield window function (see **25.9.2** for details).

Table 25-8. Setting Security in Each Programming Mode

(1) On-board/off-board programming

Security	Security Setting	How to Disable Security Setting
Prohibition of batch erase (chip erase)	Set via GUI of dedicated flash memory programmer, etc.	Cannot be disabled after set.
Prohibition of block erase		Execute batch erase (chip erase) command
Prohibition of writing		
Prohibition of rewriting boot cluster 0		Cannot be disabled after set.

(2) Self programming

Security	Security Setting	How to Disable Security Setting
Prohibition of batch erase (chip erase)	Set by using information library.	Cannot be disabled after set.
Prohibition of block erase		Execute batch erase (chip erase) command during on-board/off-board programming (cannot be disabled during self programming)
Prohibition of writing		
Prohibition of rewriting boot cluster 0		

DC Characteristics (13/16)

μPD78F1167A(A), 78F1168A(A)

(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
Supply current	I_{DD1} Note 1	Operating mode	f_{SUB} = 32.768 kHz Note 2, T_A = –40 to +70°C	V_{DD} = 5.0 V		6.4	36.0	μA
				V_{DD} = 3.0 V		6.4	36.0	μA
				V_{DD} = 2.0 V		6.3	32.8	μA
			f_{SUB} = 32.768 kHz Note 2, T_A = –40 to +85°C	V_{DD} = 5.0 V		6.4	51.0	μA
				V_{DD} = 3.0 V		6.4	51.0	μA
				V_{DD} = 2.0 V		6.3	47.8	μA

Notes 1. Total current flowing into V_{DD}, EV_{DD0}, EV_{DD1}, AV_{REF0}, and AV_{REF1}, including the input leakage current flowing when the level of the input pin is fixed to V_{DD} or V_{SS}. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, LVI circuit, I/O port, and on-chip pull-up/pull-down resistors.

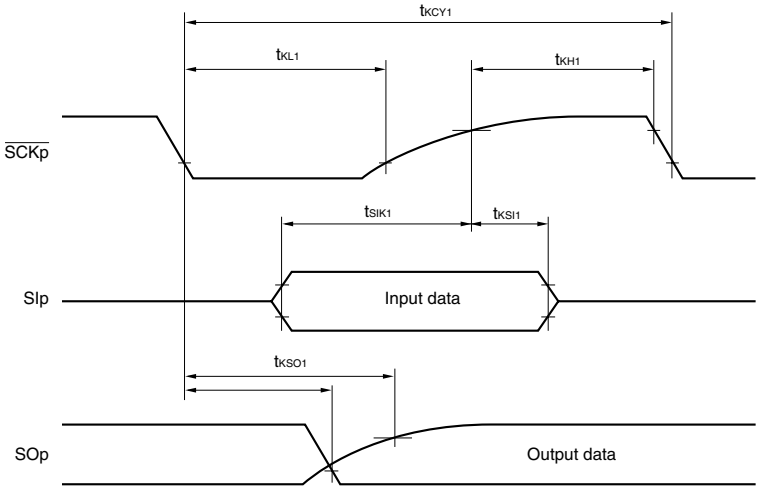
2. When internal high-speed oscillator and high-speed system clock are stopped. When watchdog timer is stopped.

Remarks 1. f_{SUB} : Subsystem clock frequency (XT1 clock oscillation frequency)

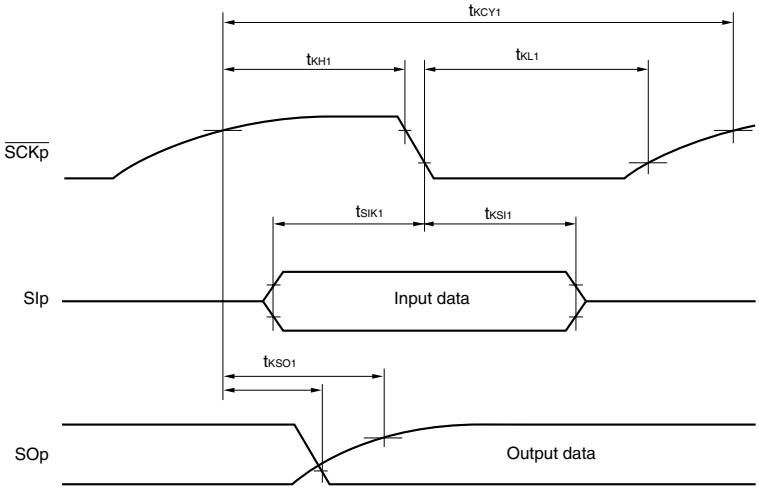
<R> **2.** Temperature condition of the TYP. value is T_A = 25°C

(3) Serial interface: Serial array unit (13/18)

CSI mode serial transfer timing (during communication at different potential)
(When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1)



CSI mode serial transfer timing (during communication at different potential)
(When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0)



Caution Select the TTL input buffer for SIp and the N-ch open-drain output (V_{DD} tolerance) mode for SOp and $\overline{\text{SCKp}}$ by using the PIMg and POMg registers.

Remarks 1. p: CSI number (p = 01, 10, 20), g: PIM and POM number (g = 0, 4, 14)

2. m: Unit number (m = 0, 1), n: Channel number (n = 0 to 2)

3. CSI00 cannot communicate at different potential. Use CSI01, CSI10, and CSI20 for communication at different potential.

(32/35)

Chapter	Classification	Function	Details of Function	Cautions	Page
Chapter 29	Soft	Electrical specifications (standard products)	During communication at different potential (2.5 V, 3 V) (CSI mode) (master mode, $\overline{SCKp}$... internal clock output)	Select the TTL input buffer for SIp and the N-ch open-drain output (V_{DD} tolerance) mode for SOp and $\overline{SCKp}$ by using the PIMg and POMg registers.	pp.810, 811, 812 ☐
			During communication at different potential (2.5 V, 3 V) (CSI mode) (slave mode, $\overline{SCKp}$... external clock input)	Select the TTL input buffer for SIp and $\overline{SCKp}$ and the N-ch open-drain output (V_{DD} tolerance) mode for SOp by using the PIMg and POMg registers.	pp.814, 815 ☐
			During communication at different potential (2.5 V, 3 V) (simplified I^2C mode)	Select the TTL input buffer and the N-ch open-drain output (V_{DD} tolerance) mode for SDAr and the N-ch open-drain output (V_{DD} tolerance) mode for SCLr by using the PIMg and POMg registers.	pp.816, 817 ☐
Chapter 30	Hard	Electrical specifications ((A) grade products)	–	The 78K0R/KG3 has an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. NEC Electronics is not liable for problems occurring when the on-chip debug function is used.	p.827 ☐
			Absolute maximum ratings	Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.	pp.827, 828 ☐
			X1 oscillator characteristics	When using the X1 oscillator, wire as follows in the area enclosed by the broken lines in the above figures 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.	p.829 ☐
				Since the CPU is started by the internal high-speed oscillation clock after a reset release, check the X1 clock oscillation stabilization time using the oscillation stabilization time counter status register (OSTC) by the user. Determine the oscillation stabilization time of the OSTC register and oscillation stabilization time select register (OSTS) after sufficiently evaluating the oscillation stabilization time with the resonator to be used.	p.829 ☐

(3/5)

Page	Description	Classification
CHAPTER 13 SERIAL ARRAY UNIT (continuation)		
p.436	Change of **Figure 13-37. Flowchart of Master Reception (in Single-Reception Mode)**	(c)
p.437	Addition of **Figure 13-38. Timing Chart of Master Reception (in Continuous Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0)**	(c)
p.438	Addition of **Figure 13-39. Flowchart of Master Reception (in Continuous Reception Mode)**	(c)
p.451	Change of **Figure 13-51. Procedure for Resuming Slave Transmission**	(b)
p.453	Change of **Figure 13-53. Flowchart of Slave Transmission (in Single-Transmission Mode)**	(c)
p.455	Change of **Figure 13-55. Flowchart of Slave Transmission (in Continuous Transmission Mode)**	(c)
p.457	Change of **Figure 13-56. Example of Contents of Registers for Slave Reception of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI20)**	(c)
p.459	Change of **Figure 13-59. Procedure for Resuming Slave Reception**	(c)
p.461	Change of **Figure 13-61. Flowchart of Slave Reception (in Single-Reception Mode)**	(c)
p.463	Addition of **Caution** to **Figure 13-62. Example of Contents of Registers for Slave Transmission/Reception of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI20)**	(c)
p.464	Addition of **Caution** to **Figure 13-63. Initial Setting Procedure for Slave Transmission/Reception**	(c)
p.466	Change of **Figure 13-65. Procedure for Resuming Slave Transmission/Reception**	(c)
p.468	Change of **Figure 13-67. Flowchart of Slave Transmission/Reception (in Single-Transmission/Reception Mode)**	(c)
p.470	Change of **Figure 13-69. Flowchart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode)**	(c)
p.486	Change of **Figure 13-79. Example of Contents of Registers for UART Reception of UART (UART0, UART1, UART2, UART3) (1/2)**	(c)
p.489	Change of **Figure 13-82. Procedure for Resuming UART Reception**	(c)
p.491	Change of **Figure 13-84. Flowchart of UART Reception**	(c)
p.505	Change of **13.7 Operation of Simplified I^2C (IIC10, IIC20) Communication**	(c)
p.506	Change of transfer rate in **13.7.1 Address field transmission**	(b)
p.511	Change of transfer rate in **13.7.2 Data transmission**	(b)
p.514	Change of error detection flag and transfer rate in **13.7.3 Data reception**	(b)
p.519	Addition of **Caution** to **13.7.5 Calculating transfer rate**	(c)
p.519	Change of **Remark** in **13.7.5 Calculating transfer rate**	(c)
p.522	Addition of **Figure 13-105. Processing Procedure in Case of Parity Error or Overrun Error**	(c)
CHAPTER 14 SERIAL INTERFACE IIC0		
p.539	Change of description of STT0 bit in **Figure 14-6. Format of IIC Control Register 0 (IICC0) (3/4)**	(c)
CHAPTER 16 DMA CONTROLLER		
p.611	Addition of **Note** to **Figure 16-4. Format of DMA Mode Control Register n (DMCn) (1/2)**	(c)
p.617	Change of description in **16.5.1 CSI consecutive transmission**	(c)
p.618	Change of description in **Figure 16-7. Setting Example of CSI Consecutive Transmission**	(c)
pp.619, 620	Addition of **16.5.2 CSI master reception**	(c)
pp.621, 622	Addition of **16.5.3 CSI transmission/reception**	(c)
p.627	Change of description in **16.5.6 Holding DMA transfer pending by DWAITn**	(c)

Remark "Classification" in the above table classifies revisions as follows.

(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(10/24)

Edition	Description	Chapter
4th edition	Addition of **Note** to **Figure 5-1 (e) Memory map of μPD78F1166** and **(g) Memory map of μPD78F1168**	CHAPTER 5 EXTERNAL BUS INTERFACE
	Change of **(c) No wait, 16-bit bus CLKOUT = f_{CLK}/2 (EXWEN = 0, MM3 = 1, MM2 = 1)** and **(d) With wait, 16-bit bus CLKOUT = f_{CLK}/2 (EXWEN = 1, MM3 = 1, MM2 = 1)** in **Figure 5-6. Timing to Read External Memory**	
	Change of **(a) No wait, 8-bit bus CLKOUT = f_{CLK} (EXWEN = 0, MM3 = 1, MM2 = 0)** and **(b) With wait, 8-bit bus CLKOUT = f_{CLK} (EXWEN = 1, MM3 = 1, MM2 = 0)** in **Figure 5-7. Timing to Write to External Memory**	
	Changes of **Figure 5-9 Example of Synchronous Memory Connection** and **Figure 5-10 Example of Asynchronous Memory Connection**	
	Change of **Figure 6-1 Block Diagram of Clock Generator**	CHAPTER 6 CLOCK GENERATOR
	Addition of **Caution** to **Figure 6-7 Format of Peripheral Enable Register**	
	Addition of **Note 4** to **6.3 (7) Operation speed mode control register (OSMC)**	
	Change of description of **6.3 (8) Internal high-speed oscillator trimming register (HIOTRM)**	
	Addition of time until CPU operation start in **Figure 6-13 Clock Generator Operation When Power Supply Voltage Is Turned On (When LVI Default Start Function Stopped Is Set (Option Byte: LVIOFF = 1))**	
	Change of **Figure 6-14 Clock Generator Operation When Power Supply Voltage Is Turned On (When LVI Default Start Function Enabled Is Set (Option Byte: LVIOFF = 0))**	
	Addition of **Caution** to **6.6.1 (3) <3>**	
	Change of **Table 6-4 (6) CPU clock changing from high-speed system clock (C) to internal high-speed oscillation clock (B)**	
	Change of CSC register bit name in **Table 6-4 (9) CPU clock changing from subsystem clock (D) to high-speed system clock (C)**	
	Addition of **Caution 2** to **7.3 (1) Peripheral enable register 0 (PER0)**	CHAPTER 7 TIMER ARRAY UNIT
	Change of **Figure 7-6 Format of Timer Mode Register 0n (TMR0n)**	
	Addition of description to **7.3 (4) Timer status register 0n (TSR0n)**	
	Addition of **Table 7-3 OVF Bit Operation and Set/Clear Conditions in Each Operation Mode**	
	Addition of **Table 7-4 Operations from Count Operation Enabled State to TCR0n Count Start**, and (a) through (e)	
	Addition of description to **7.3 (11) Timer output level register 0 (TOL0)**	
	Change of description of **7. 3 (12) Timer output mode register 0 (TOM0)**	
	Change of **Figure 7-20 Format of Timer Output Mode Register 0 (TOM0)** and **Remark**	
	Change of description of bit 7 and addition of **Note** in **Figure 7-22 Format of Noise Filter Enable Register 1 (NFEN1)**	
	Addition of **7.4 Channel Output (TO0n pin) Control**	
	Addition of **7.5 Channel Input (TI0n Pin) Control**	
	Addition of MD0n0 bit condition to titles in the following figures • **Figure 7-37 Example of Basic Timing of Operation as Interval Timer/Square Wave Output** **(MD0n0 = 1)** • **Figure 7-45 Example of Basic Timing of Operation as Frequency Divider (MD0n0 = 1)** • **Figure 7-49 Example of Block Diagram of Operation as Input Pulse Interval Measurement** **(MD0n0 = 0)**	