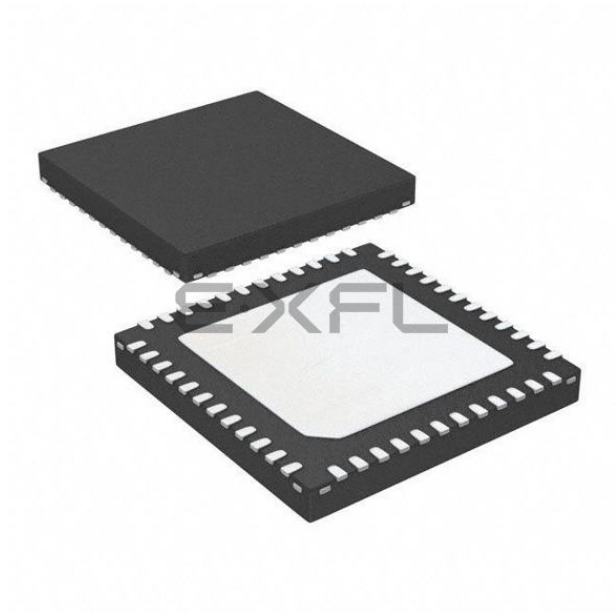


Welcome to E-XFL.COM

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	Obsolete
Core Processor	RL78
Core Size	16-Bit
Speed	32MHz
Connectivity	CSI, I²C, LINbus, UART/USART
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	34
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	8K x 8
RAM Size	12K x 8
Voltage - Supply (Vcc/Vdd)	1.6V ~ 5.5V
Data Converters	A/D 10x8/10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-WFQFN Exposed Pad
Supplier Device Package	48-HWQFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r5f100ggana-w0

(3/4)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P70	7-1-1	I/O	Input port	KR0/SCK21/SCL21	Port 7. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output (EV_{DD} tolerance).
P71	7-1-2			KR1/SI21/SDA21	
P72	7-1-1			KR2/SO21	
P73				KR3	
P74	7-1-2			KR4/INTP8	
P75	7-1-1			KR5/INTP9	
P76				KR6/INTP10/(RxD2)	
P77				KR7/INTP11/(TxD2)	
P80	8-1-2	I/O	Input port	(SCK10)/(SCL10)	Port 8. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P80 and P81 can be set to TTL input buffer. Output of P80 to P82 can be set to N-ch open-drain output (EV_{DD} tolerance).
P81				(SI10)/(RxD1)/(SDA10)	
P82	7-1-2			(SO10)/(TxD1)	
P83	7-1-1			—	
P84				(INTP6)	
P85				(INTP7)	
P86				(INTP8)	
P87				(INTP9)	
P90	7-1-1	I/O	Input port	—	Port 9. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P96 can be set to N-ch open-drain output (EV_{DD} tolerance).
P91				—	
P92				—	
P93				—	
P94				—	
P95				SCK11/SCL11	
P96	7-1-2			SI11/SDA11	
P97	7-1-1			SO11	
P100	7-3-1	I/O	Analog input port	ANI20	Port 10. 7-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P100 can be set to analog input **Note**.
P101	7-1-1		Input port	—	
P102				TI06/TO06	
P103				TI14/TO14	
P104				TI15/TO15	
P105				TI16/TO16	
P106				TI17/TO17	

Note Digital or analog for each pin can be selected with the port mode control register x (PMCx) (can be set in 1-bit unit).

Remark Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register (PIOR). Refer to **Figure 4-8 Format of Peripheral I/O Redirection Register (PIOR)**.

6.3.9 Timer output enable register m (TOEm)

The TOEm register is used to enable or disable timer output of each channel.

Channel n for which timer output has been enabled becomes unable to rewrite the value of the TOmn bit of timer output register m (TOm) described later by software, and the value reflecting the setting of the timer output function through the count operation is output from the timer output pin (TOmn).

The TOEm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the TOEm register can be set with a 1-bit or 8-bit memory manipulation instruction with TOEmL.

Reset signal generation clears this register to 0000H.

Figure 6-17. Format of Timer Output Enable register m (TOEm)

Address: F01BAH, F01BBH (TOE0), F01FAH, F01FBH (TOE1) After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOEm	0	0	0	0	0	0	0	0	TOEm7	TOEm6	TOEm5	TOEm4	TOEm3	TOEm2	TOEm1	TOEm0

TOEmn	Timer output enable/disable of channel n
0	Disable output of timer. Without reflecting on TOmn bit timer operation, to fixed the output. Writing to the TOmn bit is enabled and the level set in the TOmn bit is output from the TOmn pin.
1	Enable output of timer. Reflected in the TOmn bit timer operation, to generate the output waveform. Writing to the TOmn bit is disabled (writing is ignored).

Caution Be sure to clear bits 15 to 8 to 0.

Remark m: Unit number (m = 0, 1), n: Channel number (n = 0 to 7)

The window open period can be set is as follows.

Table 10-4. Setting Window Open Period of Watchdog Timer

WINDOW1	WINDOW0	Window Open Period of Watchdog Timer
0	0	Setting prohibited
0	1	50%
1	0	75% Note
1	1	100%

<R> **Note** When the window open period is set to 75%, clearing the counter of the watchdog timer (writing ACH to WDTE) must proceed outside the corresponding period from among those listed below, over which clearing of the counter is prohibited (for example, confirming that the interval timer interrupt request flag (WDTIIF) of the watchdog timer is set).

WDCS2	WDCS1	WDCS0	Watchdog timer overflow time (f_{IL} = 17.25 kHz (MAX.))	Period over which clearing the counter is prohibited when the window open period is set to 75%
0	0	0	$2^6/f_{IL}$ (3.71 ms)	1.85 ms to 2.51 ms
0	0	1	$2^7/f_{IL}$ (7.42 ms)	3.71 ms to 5.02 ms
0	1	0	$2^8/f_{IL}$ (14.84 ms)	7.42 ms to 10.04 ms
0	1	1	$2^9/f_{IL}$ (29.68 ms)	14.84 ms to 20.08 ms
1	0	0	$2^{11}/f_{IL}$ (118.72 ms)	56.36 ms to 80.32 ms
1	0	1	$2^{13}/f_{IL}$ (474.90 ms)	237.44 ms to 321.26 ms
1	1	0	$2^{14}/f_{IL}$ (949.80 ms)	474.89 ms to 642.51 ms
1	1	1	$2^{16}/f_{IL}$ (3799.19 ms)	1899.59 ms to 2570.04 ms

Caution When bit 0 (WDSTBYON) of the option byte (000C0H) = 0, the window open period is 100% regardless of the values of the WINDOW1 and WINDOW0 bits.

Remark If the overflow time is set to $2^9/f_{IL}$, the window close time and open time are as follows.

	Setting of Window Open Period		
	50%	75%	100%
Window close time	0 to 20.08 ms	0 to 10.04 ms	None
Window open time	20.08 to 29.68 ms	10.04 to 29.68 ms	0 to 29.68 ms

<When window open period is 50%>

- Overflow time:
  $2^9/f_{IL}$ (MAX.) = 2^9/17.25 kHz = 29.68 ms
- Window close time:
  0 to $2^9/f_{IL}$ (MIN.) × (1 – 0.5) = 0 to 2^9/12.75 kHz × 0.5 = 0 to 20.08 ms
- Window open time:
  $2^9/f_{IL}$ (MIN.) × (1 – 0.5) to $2^9/f_{IL}$ (MAX.) = 2^9/12.75 kHz × 0.5 to 2^9/17.25 kHz = 20.08 to 29.68 ms

Table 11-1. Settings of ADCS and ADCE Bits

ADCS	ADCE	A/D Conversion Operation
0	0	Conversion stopped state
0	1	Conversion standby state
1	0	Setting prohibited
1	1	Conversion-in-progress state

Table 11-2. Setting and Clearing Conditions for ADCS Bit

A/D Conversion Mode			Set Conditions	Clear Conditions
Software trigger	Select mode	Sequential conversion mode	When 1 is written to ADCS	When 0 is written to ADCS
		One-shot conversion mode		● When 0 is written to ADCS ● The bit is automatically cleared to 0 when A/D conversion ends.
	Scan mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		● When 0 is written to ADCS ● The bit is automatically cleared to 0 when conversion ends on the specified four channels.
Hardware trigger no-wait mode	Select mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		When 0 is written to ADCS
	Scan mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		When 0 is written to ADCS
Hardware trigger wait mode	Select mode	Sequential conversion mode	When a hardware trigger is input	When 0 is written to ADCS
		One-shot conversion mode		● When 0 is written to ADCS ● The bit is automatically cleared to 0 when A/D conversion ends.
	Scan mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		● When 0 is written to ADCS ● The bit is automatically cleared to 0 when conversion ends on the specified four channels.

11.6.12 Hardware trigger wait mode (scan mode, one-shot conversion mode)

<1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.

<2> If a hardware trigger is input while in the hardware trigger standby status, A/D conversion is performed on the four analog input channels specified by scan 0 to scan 3, which are specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input. A/D conversion is performed on the analog input channels in order, starting with that specified by scan 0.

<3> A/D conversion is sequentially performed on the four analog input channels, the conversion results are stored in the A/D conversion result register (ADCR, ADCRH) each time conversion ends, and the A/D conversion end interrupt request signal (INTAD) is generated.

<4> After A/D conversion ends, the ADCS bit is automatically cleared to 0, and the A/D converter enters the stop status.

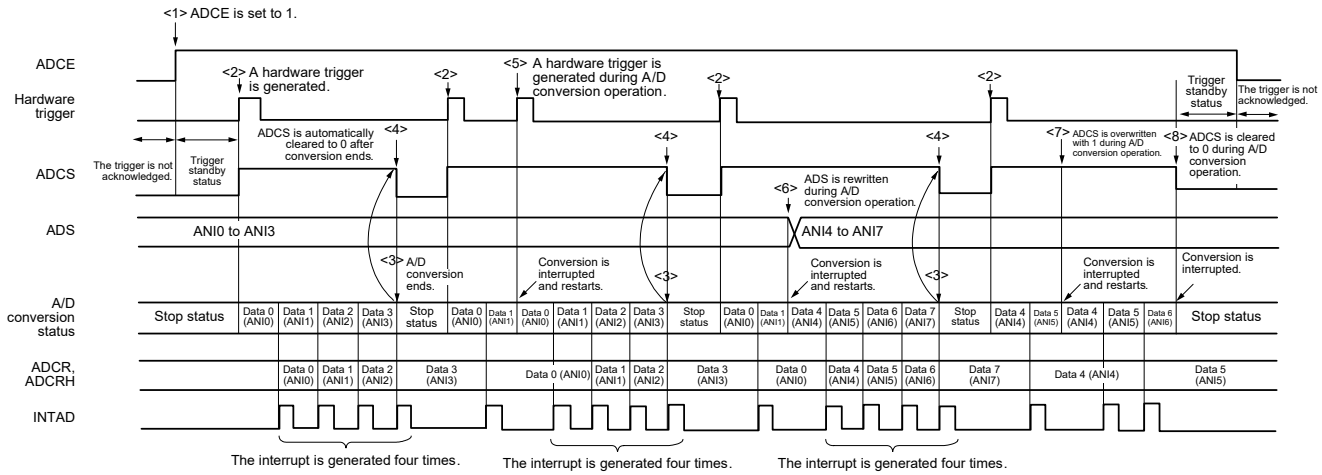
<5> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.

<6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the first channel respecified by the ADS register. The partially converted data is discarded.

<7> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<8> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the system enters the hardware trigger standby status, and the A/D converter enters the stop status. When ADCE = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

Figure 11-28. Example of Hardware Trigger Wait Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing



12.3.2 Serial clock select register m (SPSm)

The SPSm register is a 16-bit register that is used to select two types of operation clocks (CKm0, CKm1) that are commonly supplied to each channel. CKm1 is selected by bits 7 to 4 of the SPSm register , and CKm0 is selected by bits 3 to 0.

Rewriting the SPSm register is prohibited when the register is in operation (when SEmn = 1).

The SPSm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SPSm register can be set with an 8-bit memory manipulation instruction with SPSmL.

Reset signal generation clears the SPSm register to 0000H.

Figure 12-6. Format of Serial Clock Select Register m (SPSm)

Address: F0126H, F0127H (SPS0), F0166H, F0167H (SPS1) Note 1 After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPSm	0	0	0	0	0	0	0	0	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00

PRS mk3	PRS mk2	PRS mk1	PRS mk0	Section of operation clock (CKmk) Note 2					
					f_{CLK} = 2 MHz	f_{CLK} = 5 MHz	f_{CLK} = 10 MHz	f_{CLK} = 20 MHz	f_{CLK} = 32 MHz
0	0	0	0	f_{CLK}	2 MHz	5 MHz	10 MHz	20 MHz	32 MHz
0	0	0	1	$f_{CLK}/2$	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0	0	1	0	$f_{CLK}/2^2$	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0	0	1	1	$f_{CLK}/2^3$	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0	1	0	0	$f_{CLK}/2^4$	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
0	1	0	1	$f_{CLK}/2^5$	62.5 kHz	156 kHz	313 kHz	625 kHz	1 MHz
0	1	1	0	$f_{CLK}/2^6$	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz
0	1	1	1	$f_{CLK}/2^7$	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz	250 kHz
1	0	0	0	$f_{CLK}/2^8$	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
1	0	0	1	$f_{CLK}/2^9$	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz	62.5 kHz
1	0	1	0	$f_{CLK}/2^{10}$	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
1	0	1	1	$f_{CLK}/2^{11}$	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz	15.6 kHz
1	1	0	0	$f_{CLK}/2^{12}$	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
1	1	0	1	$f_{CLK}/2^{13}$	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
1	1	1	0	$f_{CLK}/2^{14}$	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
1	1	1	1	$f_{CLK}/2^{15}$	61 Hz	153 kHz	305 Hz	610 Hz	977 Hz

Notes 1. 30 to 128-pin products only.

2. When changing the clock selected for f_{CLK} (by changing the system clock control register (CKC) value), do so after having stopped (serial channel stop register m (STm) = 000FH) the operation of the serial array unit (SAU).

Caution Be sure to clear bits 15 to 8 to "0".

Remarks 1. f_{CLK}: CPU/peripheral hardware clock frequency

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

3. k = 0, 1

12.3.6 Serial flag clear trigger register mn (SIRmn)

The SIRmn register is a trigger register that is used to clear each error flag of channel n.

When each bit (FECTmn, PECTmn, OVCTmn) of this register is set to 1, the corresponding bit (FEFmn, PEFmn, OVFmn) of serial status register mn is cleared to 0. Because the SIRmn register is a trigger register, it is cleared immediately when the corresponding bit of the SSRmn register is cleared.

The SIRmn register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SIRmn register can be set with an 8-bit memory manipulation instruction with SIRmnL.

Reset signal generation clears the SIRmn register to 0000H.

Figure 12-10. Format of Serial Flag Clear Trigger Register mn (SIRmn)

Address: F0108H, F0109H (SIR00) to F010EH, F010FH (SIR03), After reset: 0000H R/W
F0148H, F0149H (SIR10) to F014EH, F014FH (SIR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SIRmn	0	0	0	0	0	0	0	0	0	0	0	0	0	FECTmn Note 2	PECTmn	OVCTmn

FECTmn	Clear trigger of framing error of channel n
0	Not cleared
1	Clears the FEFmn bit of the SSRmn register to 0.

PECTmn	Clear trigger of parity error flag of channel n
0	Not cleared
1	Clears the PEFmn bit of the SSRmn register to 0.

OVCTmn	Clear trigger of overrun error flag of channel n
0	Not cleared
1	Clears the OVFmn bit of the SSRmn register to 0.

Notes 1. SIR00 to SIR03: All products
SIR10, SIR11: 30 to 128-pin products
SIR12, SIR13: 80 to 128-pin products

2. The SIR01, SIR03, SIR11, and SIR13 registers only.

Caution Be sure to clear bits 15 to 3 (or bits 15 to 2 for the SIR00, SIR02, SIR10, or SIR12 register) to "0".

Remarks 1. m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

2. When the SIRmn register is read, 0000H is always read.

Figure 12-20. Interrupt in UART Reception Operation in SNOOZE Mode

EOCmn Bit	SSECm Bit	Reception Ended Successfully	Reception Ended in an Error
0	0	INTSRx is generated.	INTSRx is generated.
0	1	INTSRx is generated.	INTSRx is generated.
1	0	INTSRx is generated.	INTSREx is generated.
1	1	INTSRx is generated.	No interrupt is generated.

12.3.15 Input switch control register (ISC)

The ISC1 and ISC0 bits of the ISC register are used to realize a LIN-bus communication operation by UART2 in coordination with an external interrupt and the timer array unit.

When bit 0 is set to 1, the input signal of the serial data input (RxD2) pin is selected as an external interrupt (INTP0) that can be used to detect a wakeup signal.

When bit 1 is set to 1, the input signal of the serial data input (RxD2) pin is selected as a timer input, so that wake up signal can be detected, the low width of the break field, and the pulse width of the sync field can be measured by the timer.

The ISC register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears the ISC register to 00H.

Figure 12-21. Format of Input Switch Control Register (ISC)

Address: F0073H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ISC	0	0	0	0	0	0	ISC1	ISC0

ISC1	Switching channel 7 input of timer array unit
0	30, 32, 36, 40, 44, 48, 52, 64, 80, 100, and 128-pin products: Uses the input signal of the TI07 pin as a timer input (normal operation). 20, 24, and 25-pin products: Do not use a timer input signal for channel 7.
1	Input signal of the RxD2 pin is used as timer input (detects the wakeup signal and measures the low width of the break field and the pulse width of the sync field). Setting is prohibited in the 20, 24, and 25-pin products.

ISC0	Switching external interrupt (INTP0) input
0	Uses the input signal of the INTP0 pin as an external interrupt (normal operation).
1	Uses the input signal of the RxD2 pin as an external interrupt (wakeup signal detection).

Caution Be sure to clear bits 7 to 2 to "0".

(1) Register setting

Figure 12-111. Example of Contents of Registers for Data Reception of Simplified I²C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31)

(a) Serial mode register mn (SMRmn) … Do not manipulate this register during data transmission/reception.

SMRmn

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

(b) Serial communication operation setting register mn (SCRmn) … Do not manipulate the bits of this register, except the TXEmn and RXEmn bits, during data transmission/reception.

SCRmn

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

(c) Serial data register mn (SDRmn) (lower 8 bits: SIOr)

SDRmn

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Baud rate setting Note 4							0	Dummy transmit data setting (FFH)							

SIOr

(d) Serial output register m (SOm) … Do not manipulate this register during data transmission/reception.

SOm

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	CKOm3	CKOm2	CKOm1	CKOm0	0	0	0	0	SOm3	SOm2	SOm1	SOm0
				0/1 Note 5	0/1 Note 5	0/1 Note 5	0/1 Note 5					0/1 Note 5	0/1 Note 5	0/1 Note 5	0/1 Note 5

(e) Serial output enable register m (SOEm) … Do not manipulate this register during data transmission/reception.

SOEm

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	SOEm3	SOEm2	SOEm1	SOEm0
												0/1	0/1	0/1	0/1

(f) Serial channel start register m (SSm) … Do not manipulate this register during data transmission/reception.

SSm

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

(**Notes** and **Remarks** are listed on the next page.)

Figure 13-6. Format of IICA Control Register n0 (IICCTLn0) (1/4)

Address: F0230H (IICCTL00), F0238H (IICCTL10) After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICCTLn0	IICEn	LRELn	WRELn	SPIEn	WTIMn	ACKEn	STTn	SPTn

IICEn	I^2C operation enable
0	Stop operation. Reset the IICA status register n (IICSn)[Note 1]. Stop internal operation.
1	Enable operation.
Be sure to set this bit (1) while the SCLAn and SDAAn lines are at high level.	
Condition for clearing (IICEn = 0)	Condition for setting (IICEn = 1)
● Cleared by instruction ● Reset	● Set by instruction

LRELn[Notes 2, 3]	Exit from communications
0	Normal operation
1	This exits from the current communications and sets standby mode. This setting is automatically cleared to 0 after being executed. Its uses include cases in which a locally irrelevant extension code has been received. The SCLAn and SDAAn lines are set to high impedance. The following flags of IICA control register n0 (IICCTLn0) and the IICA status register n (IICSn) are cleared to 0. ● STTn ● SPTn ● MSTSn ● EXCn ● COIn ● TRCn ● ACKDn ● STDn
The standby mode following exit from communications remains in effect until the following communications entry conditions are met. ● After a stop condition is detected, restart is in master mode. ● An address match or extension code reception occurs after the start condition.	
Condition for clearing (LRELn = 0)	Condition for setting (LRELn = 1)
● Automatically cleared after execution ● Reset	● Set by instruction

WRELn[Notes 2, 3]	Wait cancellation
0	Do not cancel wait
1	Cancel wait. This setting is automatically cleared after wait is canceled.
When the WRELn bit is set (wait canceled) during the wait period at the ninth clock pulse in the transmission status (TRCn = 1), the SDAAn line goes into the high impedance state (TRCn = 0).	
Condition for clearing (WRELn = 0)	Condition for setting (WRELn = 1)
● Automatically cleared after execution ● Reset	● Set by instruction

Notes 1. The IICA status register n (IICSn), the STCFn and IICBSYn bits of the IICA flag register n (IICFn), and the CLDn and DADn bits of IICA control register n1 (IICCTLn1) are reset.

2. The signal of this bit is invalid while IICEn is 0.

3. When the LRELn and WRELn bits are read, 0 is always read.

Caution If the operation of I^2C is enabled (IICEn = 1) when the SCLAn line is high level, the SDAAn line is low level, and the digital filter is turned on (DFCn bit of IICCTLn1 register = 1), a start condition will be inadvertently detected immediately. In this case, set (1) the LRELn bit by using a 1-bit memory manipulation instruction immediately after enabling operation of I^2C (IICEn = 1).

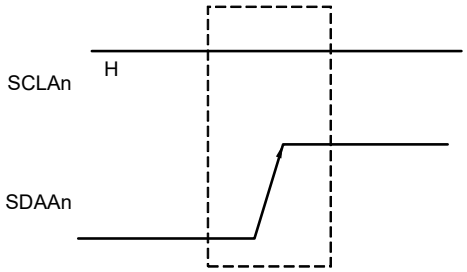
Remark n = 0, 1

13.5.5 Stop condition

When the SCLAn pin is at high level, changing the SDAAn pin from low level to high level generates a stop condition.

A stop condition is a signal that the master device generates to the slave device when serial transfer has been completed. When the device is used as a slave, stop conditions can be detected.

Figure 13-19. Stop Condition



A stop condition is generated when bit 0 (SPTn) of IICA control register n0 (IICCTLn0) is set to 1. When the stop condition is detected, bit 0 (SPDn) of the IICA status register n (IICSn) is set to 1 and INTIICAn is generated when bit 4 (SPIEn) of the IICCTLn0 register is set to 1.

Remark n = 0, 1

(2) When communication reservation function is disabled (bit 0 (IICRSVn) of IICA flag register n (IICFn) = 1)

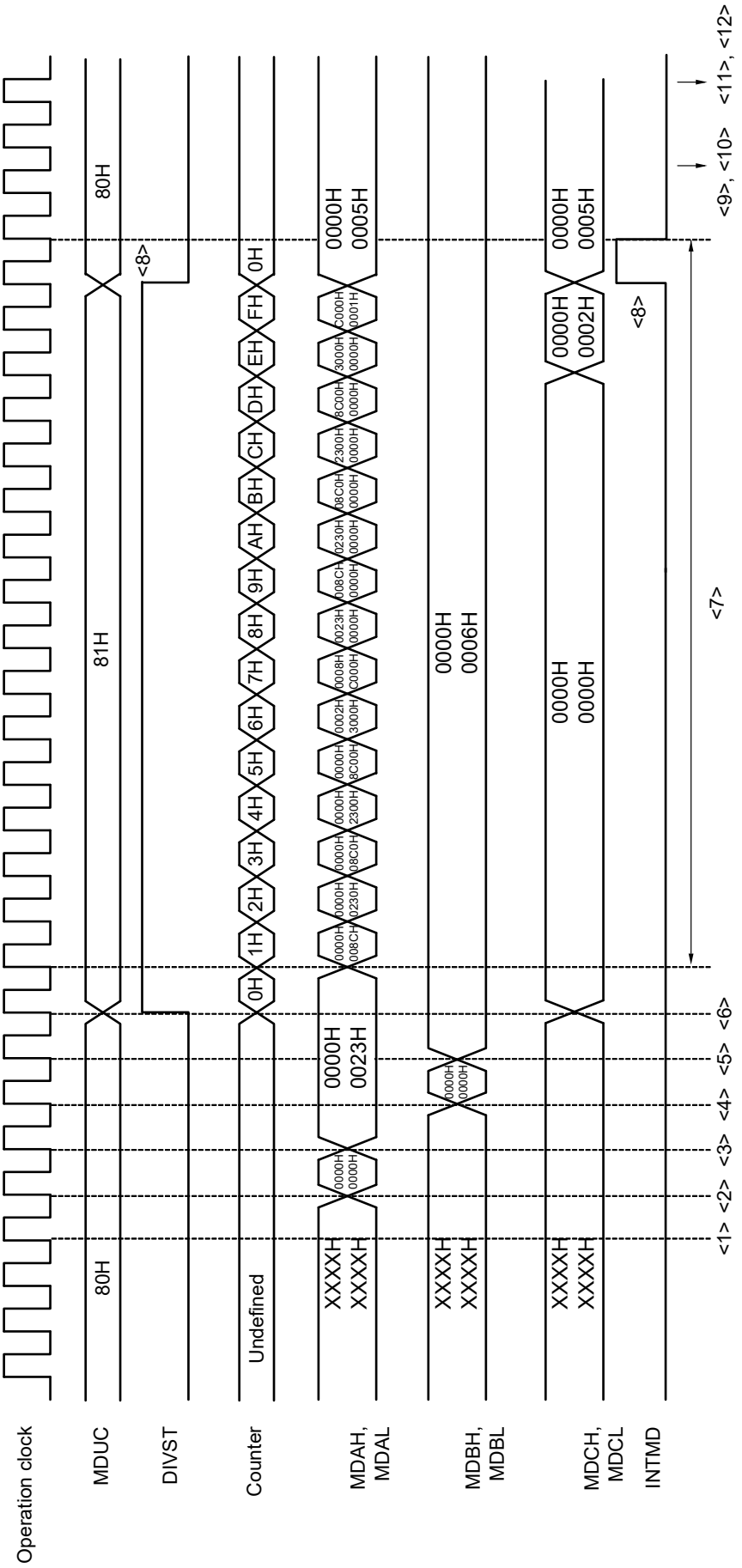
When bit 1 (STTn) of IICA control register n0 (IICCTLn0) is set to 1 when the bus is not used in a communication during bus communication, this request is rejected and a start condition is not generated. The following two statuses are included in the status where bus is not used.

- When arbitration results in neither master nor slave operation
- When an extension code is received and slave operation is disabled (ACK is not returned and the bus was released by setting bit 6 (LRELn) of the IICCTLn0 register to 1 and saving communication)

To confirm whether the start condition was generated or request was rejected, check STCFn (bit 7 of the IICFn register). It takes up to 5 cycles of f_{MCK} until the STCFn bit is set to 1 after setting STTn = 1. Therefore, secure the time by software.

Remark n = 0, 1

Figure 14-10. Timing Diagram of Division Operation (Example: 35 ÷ 6 = 5, Remainder 5)



16.3.1 Interrupt request flag registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H, IF3L)

The interrupt request flags are set to 1 when the corresponding interrupt request is generated or an instruction is executed. They are cleared to 0 when an instruction is executed upon acknowledgment of an interrupt request or upon reset signal generation.

When an interrupt is acknowledged, the interrupt request flag is automatically cleared and then the interrupt routine is entered.

The IF0L, IF0H, IF1L, IF1H, IF2L, IF2H, and IF3L registers can be set by a 1-bit or 8-bit memory manipulation instruction. When the IF0L and IF0H registers, the IF1L and IF1H registers, and the IF2L and IF2H registers are combined to form 16-bit registers IF0, IF1, and IF2, they can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

Remark If an instruction that writes data to this register is executed, the number of instruction execution clocks increases by 2 clocks.

Figure 16-2. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H, IF3L) (1/2)

Address: FFFE0H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0L	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0	LVIIF	WDTIIF

Address: FFFE1H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0H	SREIF0 TMIF01H	SRIF0 CSIIF01 IICIF01	STIF0 CSIIF00 IICIF00	DMAIF1	DMAIF0	SREIF2 TMIF11H	SRIF2 CSIIF21 IICIF21	STIF2 CSIIF20 IICIF20

Address: FFFE2H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF1L	TMIF03	TMIF02	TMIF01	TMIF00	IICAIF0	SREIF1 TMIF03H	SRIF1 CSIIF11 IICIF11	STIF1 CSIIF10 IICIF10

Address: FFFE3H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF1H	TMIF04	TMIF13	SRIF3 CSIIF31 IICIF31	STIF3 CSIIF30 IICIF30	KRIF	ITIF	RTCIF	ADIF

Address: FFFD0H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF2L	PIF10	PIF9	PIF8	PIF7	PIF6	TMIF07	TMIF06	TMIF05

16.4 Interrupt Servicing Operations

16.4.1 Maskable interrupt request acknowledgment

A maskable interrupt request becomes acknowledgeable when the interrupt request flag is set to 1 and the mask (MK) flag corresponding to that interrupt request is cleared to 0. A vectored interrupt request is acknowledged if interrupts are in the interrupt enabled state (when the IE flag is set to 1). However, a low-priority interrupt request is not acknowledged during servicing of a higher priority interrupt request.

The times from generation of a maskable interrupt request until vectored interrupt servicing is performed are listed in Table 16-4 below.

For the interrupt request acknowledgment timing, see **Figures 16-8** and **16-9**.

Table 16-4. Time from Generation of Maskable Interrupt Until Servicing

	Minimum Time	Maximum Time[Note]
Servicing time	9 clocks	16 clocks

Note Maximum time does not apply when an instruction from the internal RAM area is executed.

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

If two or more maskable interrupt requests are generated simultaneously, the request with a higher priority level specified in the priority specification flag is acknowledged first. If two or more interrupts requests have the same priority level, the request with the highest default priority is acknowledged first.

An interrupt request that is held pending is acknowledged when it becomes acknowledgeable.

Figure 16-7 shows the interrupt request acknowledgment algorithm.

If a maskable interrupt request is acknowledged, the contents are saved into the stacks in the order of PSW, then PC, the IE flag is reset (0), and the contents of the priority specification flag corresponding to the acknowledged interrupt are transferred to the ISP1 and ISP0 flags. The vector table data determined for each interrupt request is the loaded into the PC and branched.

Restoring from an interrupt is possible by using the RETI instruction.

Table 18-2. Operating Statuses in STOP Mode

STOP Mode Setting / Item			When STOP Instruction Is Executed While CPU Is Operating on Main System Clock		
			When CPU Is Operating on High-speed on-chip oscillator clock (f_{IH})	When CPU Is Operating on X1 Clock (f_X)	When CPU Is Operating on External Main System Clock (f_{EX})
System clock			Clock supply to the CPU is stopped		
	Main system clock	f_{IH}	Stopped		
		f_X			
		f_{EX}			
	Subsystem clock	f_{XT}	Status before STOP mode was set is retained		
		f_{EXS}			
	f_{IL}		Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of subsystem clock supply mode control register (OSMC) ● WUTMMCK0 = 1: Oscillates ● WUTMMCK0 = 0 and WDTON = 0: Stops ● WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates ● WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops		
CPU			Operation stopped		
Code flash memory					
Data flash memory					
RAM					
Port (latch)			Status before STOP mode was set is retained		
Timer array unit			Operation disabled		
Real-time clock (RTC)			Operable		
12-bit interval timer					
Watchdog timer			See **CHAPTER 10 WATCHDOG TIMER**.		
Clock output/buzzer output			Operates when the subsystem clock is selected as the clock source for counting and the RTCLPC bit is 0 (operation is disabled when a clock other than the subsystem clock is selected and the RTCLPC bit is not 0).		
A/D converter			Wakeup operation is enabled (switching to the SNOOZE mode)		
Serial array unit (SAU)			Wakeup operation is enabled only for CSIp and UARTq (switching to the SNOOZE mode) Operation is disabled for anything other than CSIp and UARTq		
Serial interface (IICA)			Wakeup by address match operable		
Multiplier and divider/multiply-accumulator			Operation disabled		
DMA controller					
Power-on-reset function			Operable		
Voltage detection function					
External interrupt					
Key interrupt function					
CRC operation function	High-speed CRC		Operation stopped		
	General-purpose CRC				
RAM parity error detection function					
RAM guard function					
SFR guard function					
Illegal-memory access detection function					

Remarks 1. Operation stopped: Operation is automatically stopped before switching to the STOP mode.
Operation disabled: Operation is stopped before switching to the STOP mode.
f_{IH}: High-speed on-chip oscillator clock f_{IL}: Low-speed on-chip oscillator clock
f_X: X1 clock f_{EX}: External main system clock
f_{XT}: XT1 clock f_{EXS}: External subsystem clock

2. 20 to 64-pin products: p = 00; q = 0
80 to 128-pin products: p = 00, 20; q = 0, 2

CHAPTER 22 SAFETY FUNCTIONS

22.1 Overview of Safety Functions

The following safety functions are provided in the RL78/G13 to comply with the IEC60730 and IEC61508 safety standards.

These functions enable the microcontroller to self-diagnose abnormalities and stop operating if an abnormality is detected.

(1) Flash memory CRC operation function (high-speed CRC, general-purpose CRC)

This detects data errors in the flash memory by performing CRC operations.

Two CRC functions are provided in the RL78/G13 that can be used according to the application or purpose of use.

- High-speed CRC: The CPU can be stopped and a high-speed check executed on its entire code flash memory area during the initialization routine.
- General CRC: This can be used for checking various data in addition to the code flash memory area while the CPU is running.

(2) RAM parity error detection function

This detects parity errors when reading RAM data.

(3) RAM guard function

This prevents RAM data from being rewritten when the CPU freezes.

(4) SFR guard function

This prevents SFRs from being rewritten when the CPU freezes.

(5) Invalid memory access detection function

This detects illegal accesses to invalid memory areas (such as areas where no memory is allocated and areas to which access is restricted).

(6) Frequency detection function

This function allows a self-check of the CPU/peripheral hardware clock frequencies using the timer array unit.

(7) A/D test function

This is used to perform a self-check of the A/D converter by performing A/D conversion of the A/D converter's positive and negative reference voltages, analog input channel (ANI), temperature sensor output voltage, and internal reference voltage.

Remark For usage examples of the safety functions complying with the IEC60730 safety standards, refer to **RL78 Family IEC60730/60335 self test library application notes (R01AN1062, R01AN1296)**.

22.2 Registers Used by Safety Functions

The safety functions use the following registers for each function.

Register	Each Function of Safety Function
• Flash memory CRC control register (CRC0CTL) • Flash memory CRC operation result register (PGCRCL)	Flash memory CRC operation function (high-speed CRC)
• CRC input register (CRCIN) • CRC data register (CRCD)	CRC operation function (general-purpose CRC)
• RAM parity error control register (RPECTL)	RAM parity error detection function
• Invalid memory access detection control register (IAWCTL)	RAM guard function
	SFR guard function
	Invalid memory access detection function
• Timer input select register 0 (TIS0)	Frequency detection function
• A/D test register (ADTES)	A/D test function

The content of each register is described in **22.3 Operation of Safety Functions**.

22.3 Operation of Safety Functions

22.3.1 Flash memory CRC operation function (high-speed CRC)

The IEC60730 standard mandates the checking of data in the flash memory, and recommends using CRC to do it. The high-speed CRC provided in the RL78/G13 can be used to check the entire code flash memory area during the initialization routine. The high-speed CRC can be executed only when the program is allocated on the RAM and in the HALT mode of the main system clock.

The high-speed CRC performs an operation by reading 32-bit data per clock from the flash memory while stopping the CPU. This function therefore can finish a check in a shorter time (for example, 512 µs@32 MHz with 64-KB flash memory).

The CRC generator polynomial used complies with "$X^{16} + X^{12} + X^5 + 1$" of CRC-16-CCITT.

The high-speed CRC operates in MSB first order from bit 31 to bit 0.

Caution The CRC operation result might differ during on-chip debugging because the monitor program is allocated.

Remark The operation result is different between the high-speed CRC and the general CRC, because the general CRC operates in LSB first order.

22.3.1.1 Flash memory CRC control register (CRC0CTL)

This register is used to control the operation of the high-speed CRC ALU, as well as to specify the operation range.

The CRC0CTL register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

(10/35)

Edition	Description	Chapter
Rev.3.00	Modification of description in 12.5 Operation of 3-Wire Serial I/O	CHAPTER 12 SERIAL ARRAY UNIT
	Modification of transfer rate and note in 12.5.1 Master transmission	
	Modification of Figure 12-28	
	Modification of Figure 12-29	
	Modification of Figure 12-30	
	Modification of Figure 12-31	
	Modification of Figure 12-32	
	Modification of description, transfer rate, clock phase, and note in 12.5.2 Master reception	
	Modification of Figure 12-35	
	Modification of Figure 12-36	
	Modification of Figure 12-37	
	Modification of Figure 12-39	
	Modification of Figure 12-40	
	Modification of description, transfer rate and note in 12.5.3 Master transmission/reception	
	Modification of Figure 12-43	
	Modification of Figure 12-45	
	Modification of Figure 12-47	
	Modification of Figure 12-48	
	Modification of description and notes 1 and 2 in 12.5.4 Slave transmission	
	Modification of Figure 12-51	
	Modification of Figure 12-52	
	Modification of Figure 12-53	
	Modification of Figure 12-54	
	Modification of Figure 12-55	
	Modification of Figure 12-56	
	Modification of description and notes 1 and 2 in 12.5.5 Slave reception	
	Modification of Figure 12-59	
	Modification of Figure 12-61	
	Modification of description and notes 1 and 2 in 12.5.6 Slave transmission/reception	
	Modification of Figure 12-65	
	Modification of Figure 12-67	
	Modification of Figure 12-69	
	Modification of Figure 12-70	
	Modification of Figure 12-71 and caution 2	
	Modification of Figure 12-72	
	Modification of Figure 12-73 and caution 2	
	Modification of Figure 12-74	
	Modification of description in 12.6 Operation of UART (UART0 to UART3) Communication	
	Modification of description and note 2 in 12.6.1 UART transmission	
	Modification of note 1 in Figure 12-76	
	Modification of Figures 12-77 and 12-78	
	Modification of Figure 12-79	

(21/35)

Edition	Description	Chapter
Rev.2.10	Addition of products for industrial applications (T_A = -40 to +105°C)	CHAPTER 30 ELECTRICAL SPECIFICATIONS (G: T_A = -40 to +105°C)
	Addition of products for industrial applications (T_A = -40 to +105°C)	CHAPTER 31 PACKAGE DRAWINGS
Rev.2.01	Modification of formats of register names	Throughout
Rev.2.00	Renamed interval timer (unit) to 12-bit interval timer	Throughout
	Addition of pin name of the peripheral I/O redirection function	
	Renamed V_{LVI}, V_{LVIH}, V_{LVIL} to V_{LVD}, V_{LVDH}, V_{LVDL} (LVD detection voltage)	
	Renamed interrupt source of RAM parity error (RAMTOP) to RPE	
	Renamed f_{EXS} to f_{EXT}	
	Addition of 1.1 Features	CHAPTER 1 OUTLINE
	Modification of 1.2 Ordering Information	
	Addition of Figure 1-1. Part Number, Memory Size, and Package of RL78/G13	
	Modification of 1.3.11 64-pin products	
	Addition and Modification of description in 1.6 Outline of Functions	
	Modification of 2.1 Port Function	CHAPTER 2 PIN FUNCTIONS
	Modification of description in 2.2 Functions other than port pins (Deletion of Description of Port Function)	
	Addition of remark to 2.3 Pin I/O Circuits and Recommended Connection of Unused Pins	
	Change of Table 2-3. Connection of Unused Pins (128-pin products) (2/4)	
	Addition of note 1 to Figures 3-1, 3-2, 3-5 to 3-7, 3-9	CHAPTER 3 CPU ARCHITECTURE
	Addition of caution to Figures 3-1 to 3-10	
	Modification of note in Figures 3-3, 3-4, 3-8, 3-10	
	Addition of remark to Table 3-1. Correspondence Between Address Values and Block Numbers in Flash Memory	
	Modification of description in 3.1.2 Mirror area	
	Modification of caution 2 in 3.1.3 Internal data memory space	
	Addition of note 1 to Figures 3-12, 3-13, 3-16 to 3-18, 3-20	
	Addition of caution to Figures 3-12 to 3-21	
	Addition of note 1 to Figures 3-14, 3-15, 3-19, 3-21	
	Modification of caution 3 in 3.2.1 (3) Stack pointer (SP)	
	Modification of caution 2 in 3.2.2 General-purpose registers	
	Modification of 4.1 Port Functions	CHAPTER 4 PORT FUNCTIONS
	Modification of block diagrams	
	Addition of description to 4.2.3 Port 2	
	Addition of description to 4.2.16 Port 15	
	Addition of caution to 4.3 Registers Controlling Port Function	
	Modification of Figure 4-66. Format of Port Register (128-pin products)	
	Modification of description and addition of caution to 4.3 (3) Pull-up resistor option registers (PUxx)	
	Addition of 4.3 (5) Port output mode registers (POMxx)	
	Addition of cautions 1 and 2 to Figure 4-70. Format of Port Mode Control Register	