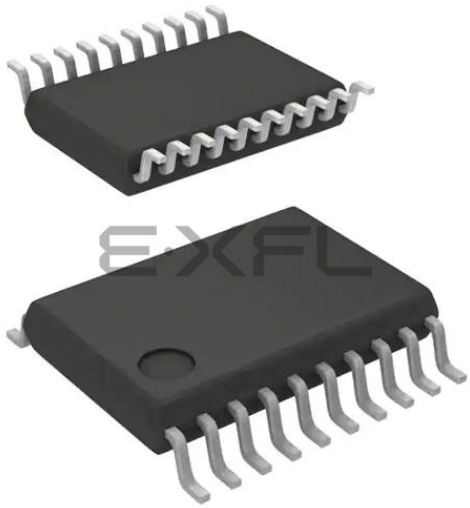


Welcome to E-XFL.COM

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	24MHz
Connectivity	CSI, I²C, UART/USART
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	14
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 11x8/10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-LSSOP (0.173", 4.40mm Width)
Supplier Device Package	20-LSSOP
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r5f1026adsp-v0

24.4.1 Serial programming procedure 695
24.4.2 Flash memory programming mode 695
24.4.3 Selecting communication mode 697
24.4.4 Communication commands 697
24.5 Processing Time for Each Command When PG-FP5 Is in Use (Reference Value) 699
24.6 Self-Programming 700
24.6.1 Self-programming procedure 701
24.6.2 Flash shield window function 702
24.7 Security Settings 703
24.8 Data Flash 705
24.8.1 Data flash overview 705
24.8.2 Register controlling data flash memory 705
24.8.2.1 Data flash control register (DFLCTL) 705
24.8.3 Procedure for accessing data flash memory 706

CHAPTER 25 ON-CHIP DEBUG FUNCTION 707

25.1 Connecting E1 On-chip Debugging Emulator 707
25.2 On-Chip Debug Security ID 709
25.3 Securing of User Resources 709

CHAPTER 26 BCD CORRECTION CIRCUIT 711

26.1 BCD Correction Circuit Function 711
26.2 Registers Used by BCD Correction Circuit 711
26.2.1 BCD correction result register (BCDADJ) 711
26.3 BCD Correction Circuit Operation 712

CHAPTER 27 INSTRUCTION SET 714

27.1 Conventions Used in Operation List 715
27.1.1 Operand identifiers and specification methods 715
27.1.2 Description of operation column 716
27.1.3 Description of flag operation column 717
27.1.4 PREFIX instruction 717
27.2 Operation List 718

CHAPTER 28 ELECTRICAL SPECIFICATIONS (T_A = –40 to +85°C) 735

28.1 Absolute Maximum Ratings 736
28.2 Oscillator Characteristics 737
28.2.1 X1 oscillator characteristics 737
28.2.2 On-chip oscillator characteristics 737
28.3 DC Characteristics 738

3.2.2 General-purpose registers

General-purpose registers are mapped at particular addresses (FFEE0H to FFEFFH) of the data memory. The general-purpose registers consists of 4 banks, each bank consisting of eight 8-bit registers (X, A, C, B, E, D, L, and H).

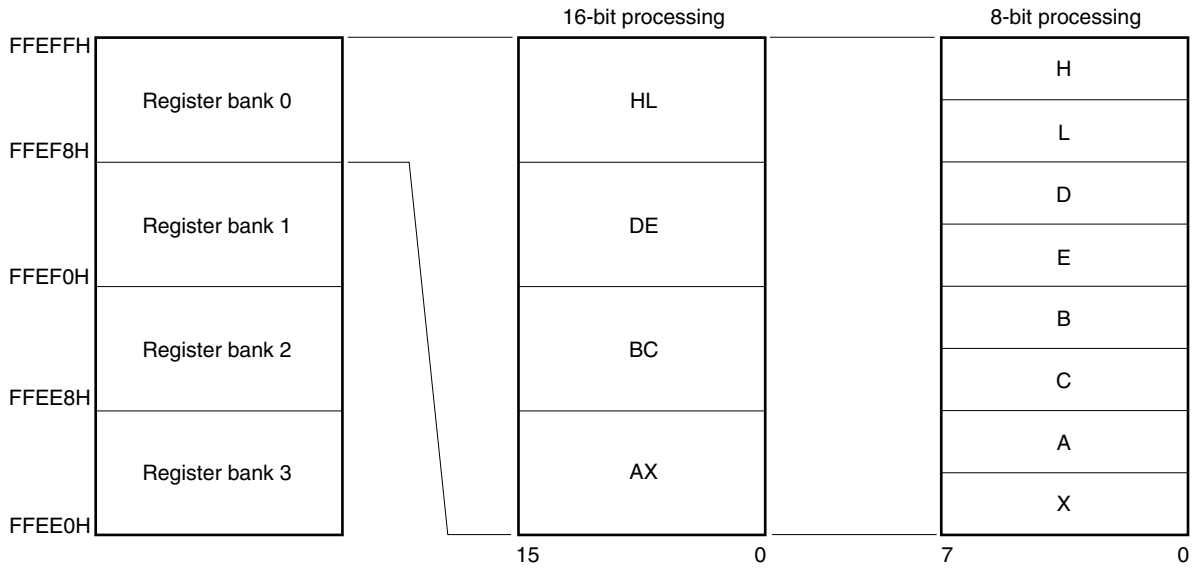
Each register can be used as an 8-bit register, and two 8-bit registers can also be used in a pair as a 16-bit register (AX, BC, DE, and HL).

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

Caution It is prohibited to use the general-purpose register space (FFEE0H to FFEFFH) for fetching instructions or as a stack area.

Figure 3-12. Configuration of General-Purpose Registers

(a) Function name



4.3.3 Pull-up resistor option registers (PUxx)

These registers specify whether the on-chip pull-up resistors are to be used or not. On-chip pull-up resistors can be used in 1-bit units only for the bits set to both normal output mode (POMmn = 0) and input mode (PMmn = 1) for the pins for which the use of an on-chip pull-up resistor is specified in these registers. On-chip pull-up resistors cannot be connected to bits set to output mode and bits used as alternate-function output pins and analog setting (PMC = 1, ADPC = 1), regardless of the settings of these registers.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets PU4 to 01H, PU12 to 20H (20-, 24-pin products), and others to 00H.

Caution When a port with the PIMn register is input from a different-potential device to the TTL buffer, pull up to the power supply of the different-potential device via an external pull-up resistor by setting PUmn = 0.

Figure 4-3. Format of Pull-up Resistor Option Register

20-, 24-pin products

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PU0	0	0	0	0	PU03	PU02	PU01	PU00	F0030H	00H	R/W
PU1	0	0	0	PU14	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU4	0	0	0	0	0	PU42	PU41	PU40	F0034H	01H	R/W
PU12	0	0	PU125[Note]	0	0	0	0	0	F003CH	20H	R/W

30-pin products

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PU0	0	0	0	0	0	0	PU01	PU00	F0030H	00H	R/W
PU1	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU3	0	0	0	0	0	0	PU31	PU30	F0033H	00H	R/W
PU4	0	0	0	0	0	0	0	PU40	F0034H	01H	R/W
PU5	0	0	0	0	0	0	PU51	PU50	F0035H	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	F003CH	00H	R/W
PU14	PU147	0	0	0	0	0	0	0	F003EH	00H	R/W

PUmn	Pmn pin on-chip pull-up resistor selection (m = 0, 1, 3 to 5, 12, 14; n = 0 to 7)
0	On-chip pull-up resistor not connected.
1	On-chip pull-up resistor connected.

Note PU125 can be selected only when P125/KR1/SI01 (PORTSELB = 0) is selected.
If the $\overline{\text{RESET}}$ pin (PORTSELB = 1) is selected, the on-chip pull-up resistor is always valid.

Caution Be sure to set bits that are not mounted to their initial values.

4.3.6 Port mode control registers (PMCxx)

These registers set the digital I/O or analog input in 1-bit units.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

Figure 4-6. Format of Port Mode Control Register

20-, 24-pin products

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PMC1	1	1	1	PMC14	PMC13	PMC12	PMC11	PMC10	F0061H	FFH	R/W
PMC4	1	1	1	1	1	PMC42	PMC41	1	F0064H	FFH	R/W

30-pin products

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PMC0	1	1	1	1	1	1	PMC01	PMC00	F0060H	FFH	R/W
PMC12	1	1	1	1	1	1	1	PMC120	F006CH	FFH	R/W
PMC14	PMC147	1	1	1	1	1	1	1	F006EH	FFH	R/W

PMCmn	Pmn pin digital I/O/analog input selection (m = 1, 4, 12, 14; n = 0 to 4, 7)
0	Digital I/O (alternate function other than analog input)
1	Analog input

Cautions 1. Use the port mode register m (PMm) to select the input mode for the ports that are set to analog input by using the PMCxx register.

2. Do not use the analog input channel specification register (ADS) to set the pins that will be set to digital I/O by using the PMCxx register.

3. Be sure to set bits that are not mounted to their initial values.

Table 4-6. Concept of Basic Settings

Output Function of Used Pin	Output Function of Unused Alternate Function		
	Port output function	SAU output function	Output function for other than SAU
Port output function	–	Output is high (1)	Output is low (0)
Output function for SAU	High (1)	–	Output is low (0)
Output function for other than SAU	Low (0)	Output is high (1)	Output is low (0)Note

Note Because more than one output function other than SAU might be assigned to a single pin, the output of unused alternate functions must be set to low level (0). For details about the setting method, see **4.5.2 Register settings for alternate function whose output function is not used**.

4.5.2 Register settings for alternate function whose output function is not used

If the output of an alternate function is not used, specify the settings as described below. If the output of the peripheral function is still the target of the peripheral I/O redirection function, the output can be switched to another pin by setting the peripheral I/O redirection register (PIOR). This allows usage of the port function or other alternate functions assigned to the target pin.

(1) SOp = 1 and TxDq = 1 (settings when the serial output (SOp/TxDq) of SAU is not used)

If the serial output (SOp/TxDq) is not used, such as a case in which only the serial input of SAU is used, set the bit in the serial output enable register m (SOEm) that corresponds to the unused output to 0 (output disabled) and set the SOmn bit in the serial output register m (SOm) to 1 (high). These settings are the same as the initial state.

(2) SCKp = 1, SDAr = 1, and SCLr = 1 (settings when channel n in SAU is not used)

If SAU is not used, set the bit n (SEmn) in the serial channel enable status register m (SEm) to 0 (operation stopped state), set the bit in the serial output enable register m (SOEm) that corresponds to the unused output to 0 (output disabled), and set the SOmn bit and CKOmn bit in the serial output register m (SOm) to 1 (high). These settings are the same as the initial state.

(3) TOmn = 0 (settings when the output of channel n in TAU is not used)

If the TOmn output of TAU is not used, set the bit in the timer output enable register 0 (TOE0) that corresponds to the unused output to 0 (output disabled) and set the bit in the timer output register 0 (TO0) to 0 (low). These settings are the same as the initial state.

(4) SDAAn = 0 and SCLAn = 0 (setting when IICA is not used)

If IICA is not used, set the IICEn bit in the IICA control register n0 (IICCTLn0) to 0 (operation stopped). This setting is the same as the initial state.

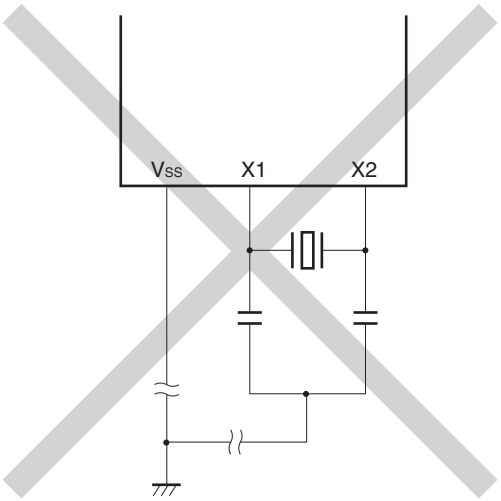
(5) PCLBUZn = 0 (setting when clock output or buzzer output is not used)

If the clock output or buzzer output is not used, set the PCLOEn bit in the clock output selection register n (CKSn) to 0 (output disabled). This setting is the same as the initial state.

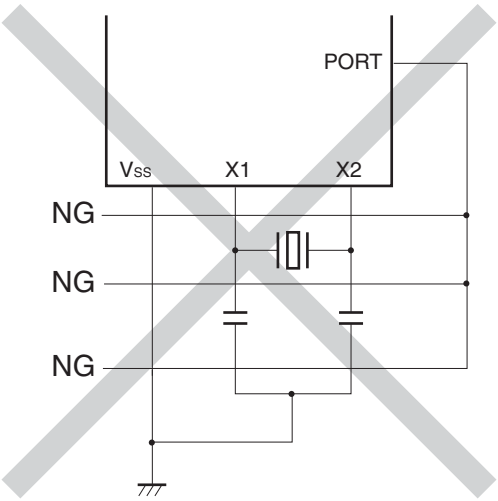
Figure 5-12 shows examples of incorrect resonator connection.

Figure 5-12. Examples of Incorrect Resonator Connection (1/2)

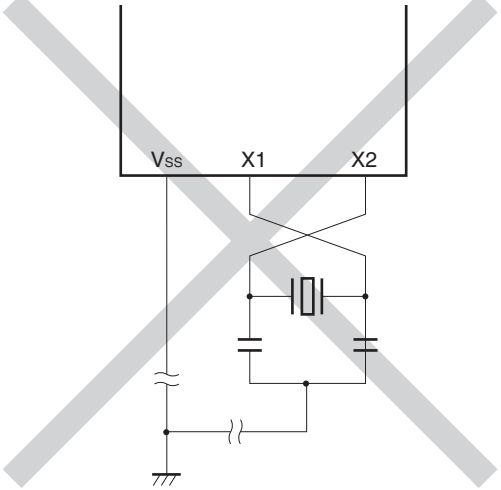
(a) Too long wiring



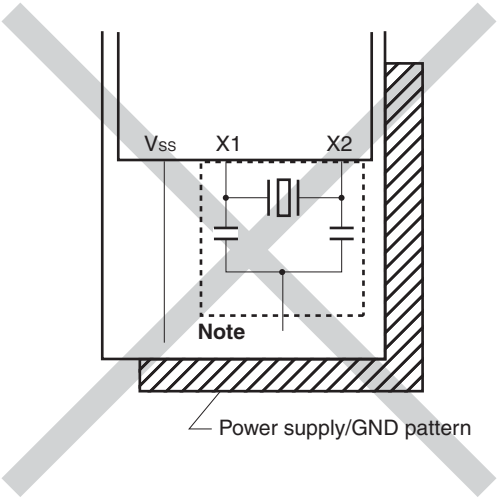
(b) Crossed signal line



(c) The X1 and X2 signal line wires cross.



(d) A power supply/GND pattern exists under the X1 and X2 wires.



Note Do not place a power supply/GND pattern under the wiring section (section indicated by a broken line in the figure) of the X1 and X2 pins and the resonators in a multi-layer board or double-sided board.
Do not configure a layout that will cause capacitance elements and affect the oscillation characteristics.

CHAPTER 6 TIMER ARRAY UNIT

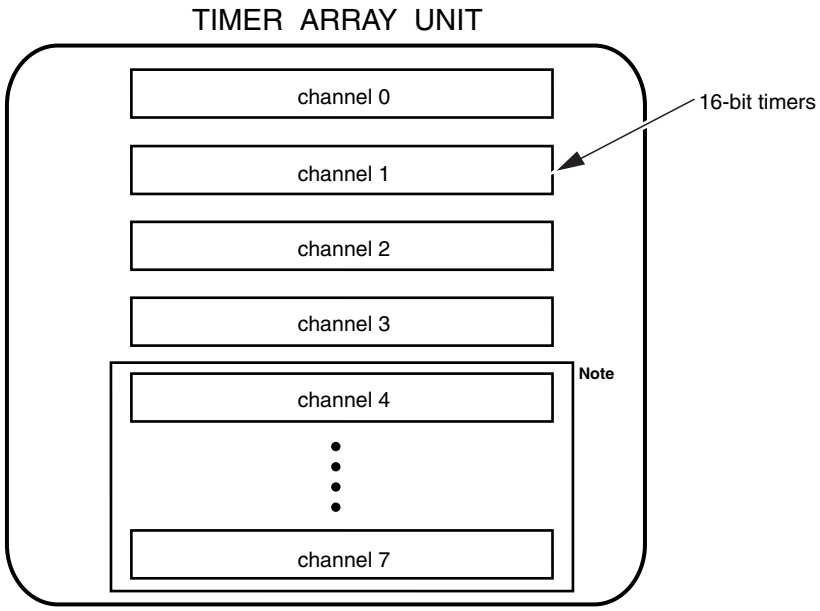
The number of channels of the timer array unit differs depending on the product.

Channels	20-, 24-pin	30-pin
Channel 0	√	√
Channel 1	√	√
Channel 2	√	√
Channel 3	√	√
Channel 4	–	√
Channel 5	–	√
Channel 6	–	√
Channel 7	–	√

Caution The presence or absence of timer I/O pins depends on the product. For details, see Table 6-2 Timer I/O Pins in the Product.

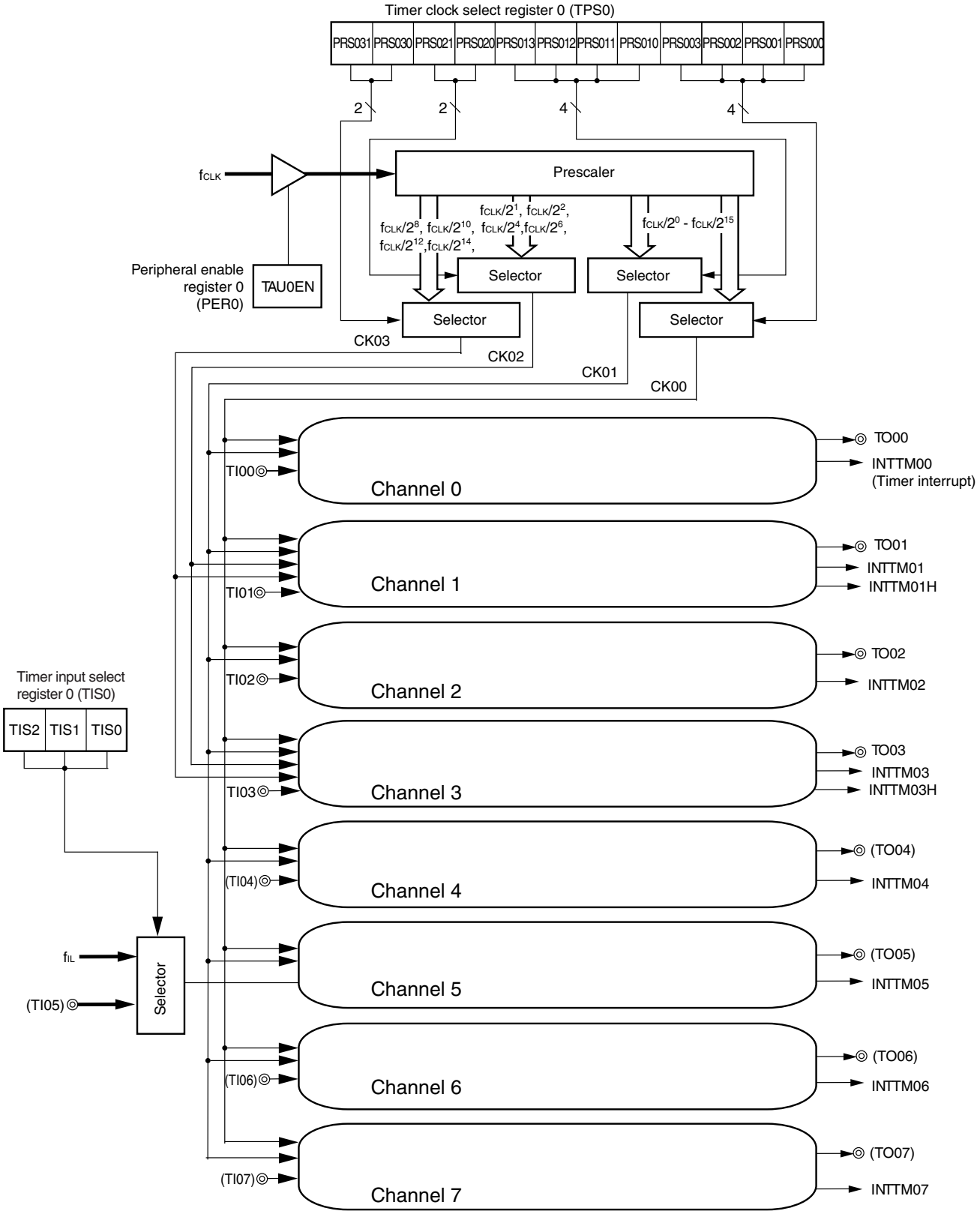
The timer array unit has four/eight 16-bit timers.

Each 16-bit timer is called a channel and can be used as an independent timer. In addition, two or more “channels” can be used to create a high-accuracy timer.



Note Provided only in 30-pin products

Figure 6-2. Entire Configuration of Timer Array Unit (30-pin products)



Remark Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register (PIOR).

6.3.13 Noise filter enable register 1 (NFEN1)

The NFEN1 register is used to set whether the noise filter can be used for the timer input signal to each channel.

Enable the noise filter by setting the corresponding bits to 1 on the pins in need of noise removal.

When the noise filter is enabled, after synchronization with the operating clock (f_{MCK}) for the target channel, whether the signal keeps the same value for two clock cycles is detected.

When the noise filter is disabled, the input signal is only synchronized with the operating clock (f_{MCK}) for the target channel[Note].

The NFEN1 registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

Note For details, see **6.5.1 (2) When valid edge of input signal via the TI0n pin is selected (CCS0n = 1)**, **6.5.2 Start timing of counter**, and **6.7 Timer Input (TI0n) Control**.

Figure 6-19. Format of Noise Filter Enable Register 1 (NFEN1)

Address: F0071H After reset: 00H R/W

20- and 24-pin products

Symbol	7	6	5	4	3	2	1	0
NFEN1	0	0	0	0	TNFEN03	TNFEN02	TNFEN01	TNFEN00

30-pin products

Symbol	7	6	5	4	3	2	1	0
NFEN1	TNFEN07	TNFEN06	TNFEN05	TNFEN04	TNFEN03	TNFEN02	TNFEN01	TNFEN00

TNFEN0n	Enable/disable using noise filter of TI0n pin input signal
0	Noise filter OFF
1	Noise filter ON

Remark The presence or absence of timer I/O pins of channel 0 to 7 depends on the product. See **Table 6-2 Timer I/O Pins in the Products** for details.

11.5.8 Calculating transfer clock frequency

The transfer clock frequency for 3-wire serial I/O (CSI00, CSI01, CSI11, CSI20) communication can be calculated by the following expressions.

(1) Master

(Transfer clock frequency) = {Operation clock (f_{MCK}) frequency of target channel} ÷ (SDRmn[15:9] + 1) ÷ 2 [Hz]

(2) Slave

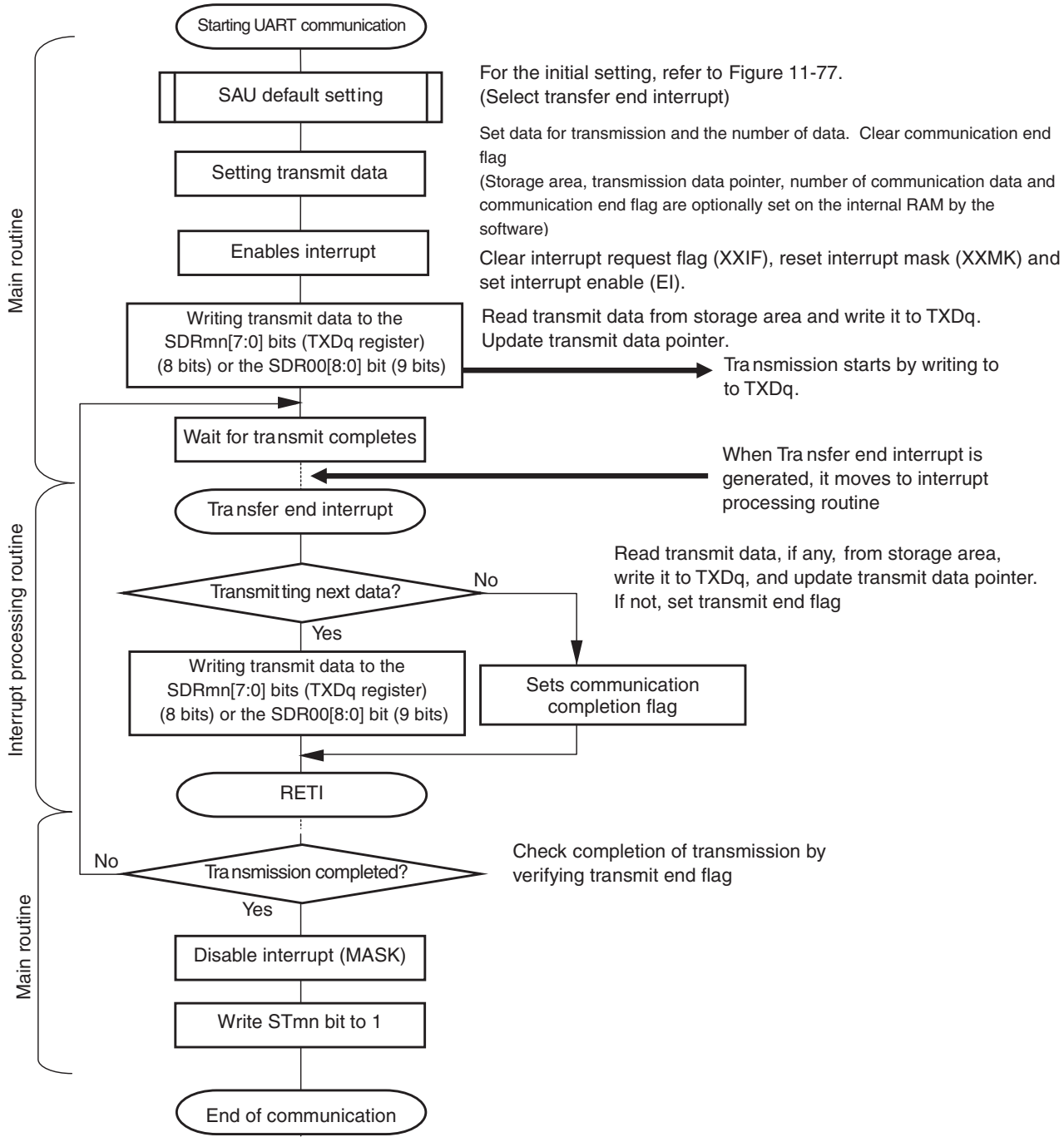
(Transfer clock frequency) = {Frequency of serial clock (SCK) supplied by master}[Note] [Hz]

Note The permissible maximum transfer clock frequency is f_{MCK}/6.

Remark The value of SDRmn[15:9] is the value of bits 15 to 9 of serial data register mn (SDRmn) (0000000B to 1111111B) and therefore is 0 to 127.

The operation clock (f_{MCK}) is determined by serial clock select register m (SPSm) and bit 15 (CKSmn) of serial mode register mn (SMRmn).

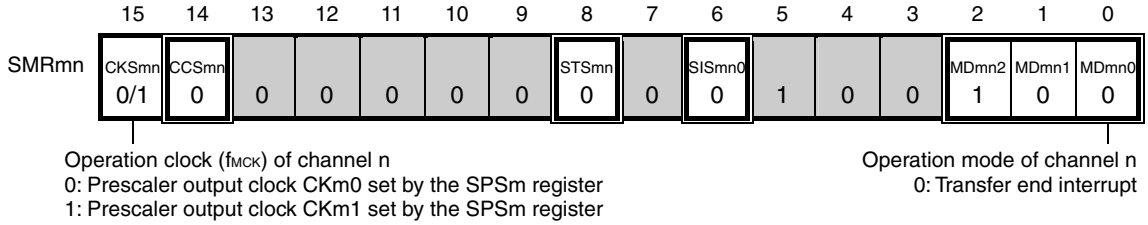
Figure 11-81. Flowchart of UART Transmission (in Single-Transmission Mode)



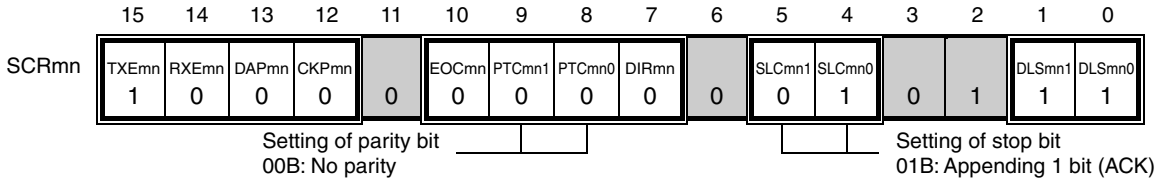
(1) Register setting

Figure 11-98. Example of Contents of Registers for Address Field Transmission of Simplified I²C (IIC00, IIC01, IIC11, IIC20)

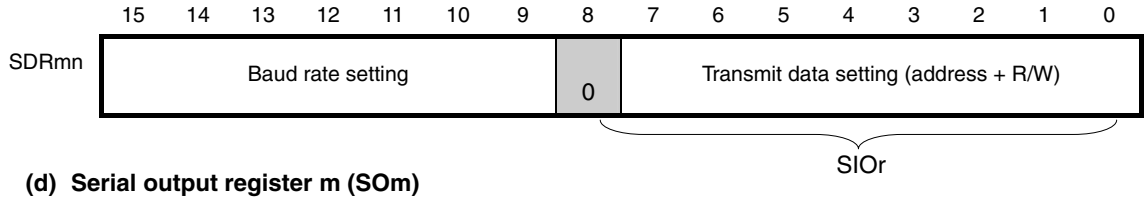
(a) Serial mode register 0n (SMR0n)



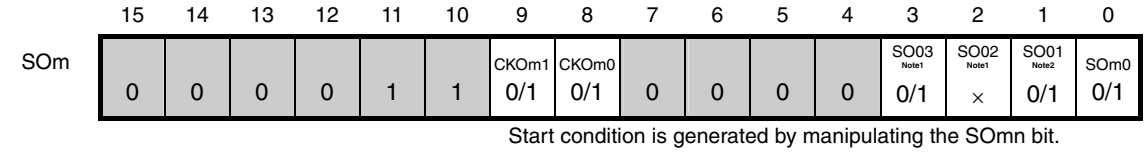
(b) Serial communication operation setting register mn (SCRmn)



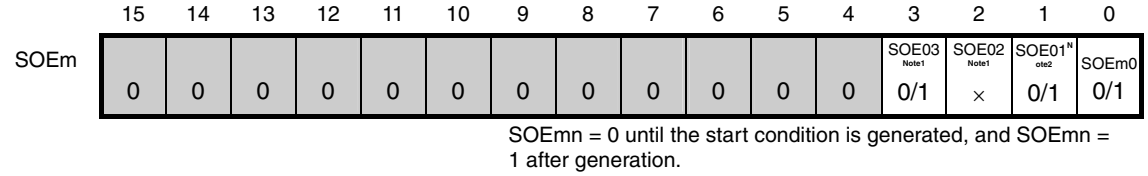
(c) Serial data register mn (SDRmn) (lower 8 bits: SIOr)



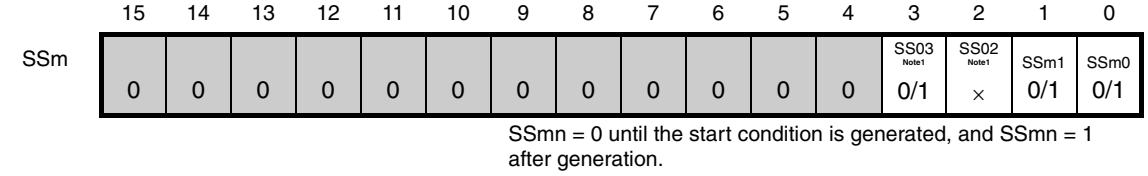
(d) Serial output register m (SOm)



(e) Serial output enable register m (SOEm)



(f) Serial channel start register m (SSm) ... Sets only the bits of the target channel is 1.



Notes 1. Provided only in 30-pin product serial array unit 0.

2. Only for 20, 24-pin product

Remarks 1. m: Unit number, n: Channel number (mn = 00, 01, 03, 10), r: IIC number (r = 00, 01, 11, 20)

2. □: Setting is fixed in the CSI master transmission 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

13.4.2 Multiplication (signed) operation

- Initial setting
  - <1> Set the multiplication/division control register (MDUC) to 08H.
  - <2> Set the multiplicand to multiplication/division data register A (L) (MDAL).
  - <3> Set the multiplier to multiplication/division data register A (H) (MDAH).
    (There is no preference in the order of executing steps <2> and <3>. Multiplication operation is automatically started when the multiplier and multiplicand are set to the MDAH and MDAL registers, respectively.)
- During operation processing
  - <4> Wait for at least one clock. The operation will end when one clock has been issued.
- Operation end
  - <5> Read the product (lower 16 bits) from multiplication/division data register B (L) (MDBL).
  - <6> Read the product (higher 16 bits) from multiplication/division data register B (H) (MDBH).
    (There is no preference in the order of executing steps <5> and <6>.)
- Next operation
  - <7> Start with the initial settings of each step to change the operation mode. When the same operation mode is used sequentially, settings <1> and <2> can be omitted.

Caution The data is in the two's complement format in multiplication mode (signed).

Remark Steps <1> to <7> correspond to <1> to <7> in Figure 13-7.

Figure 13-7. Timing Diagram of Multiplication (Signed) Operation (–2 × 32767 = –65534)

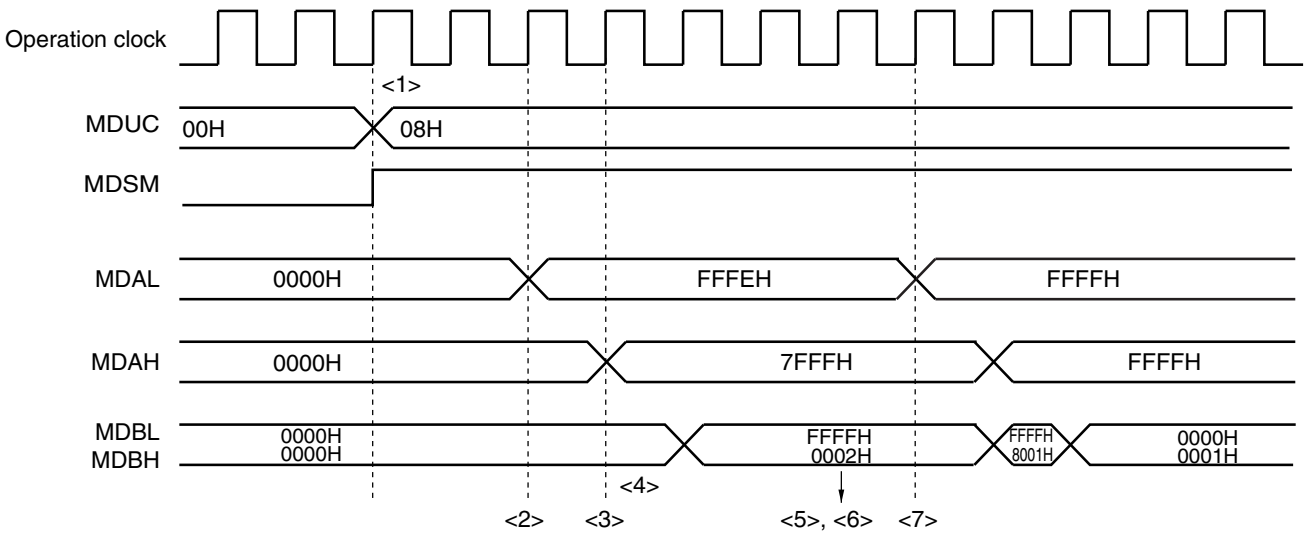
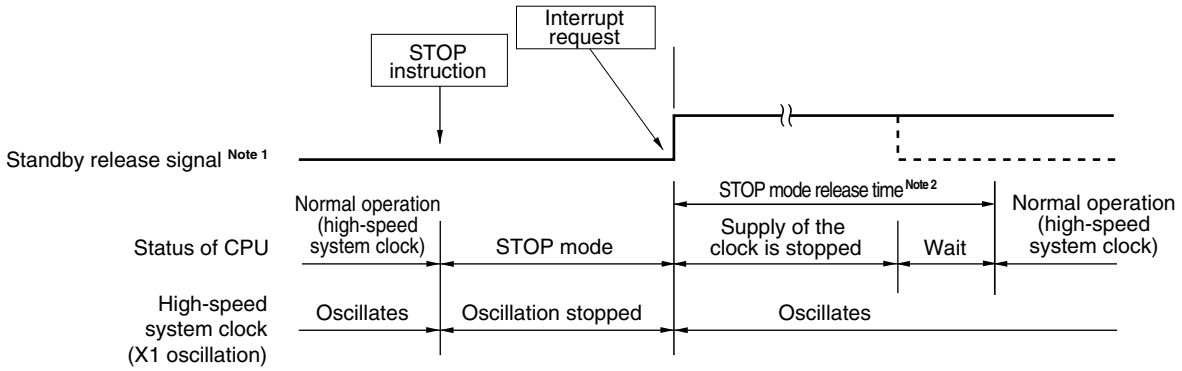


Figure 17-3. STOP Mode Release by Interrupt Request Generation (2/2)

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



Notes 1. For details of the standby release signal, see **Figure 15-1 Basic Configuration of Interrupt Function**.

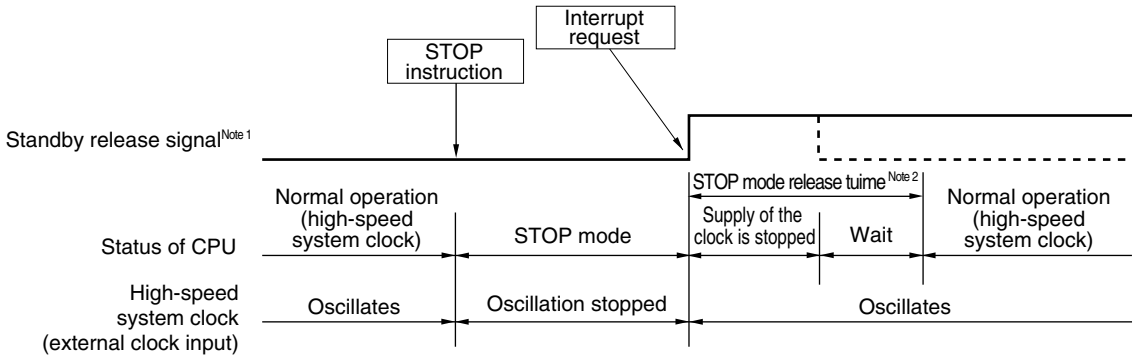
2. STOP mode release time

Supply of the clock is stopped: 18 μs to "whichever is longer 65 μs and the oscillation stabilization time (set by OSTS)"

Wait

- When vectored interrupt servicing is carried out: 10 to 11 clocks
- When vectored interrupt servicing is not carried out: 4 to 5 clocks

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



Notes 1. For details of the standby release signal, see **Figure 15-1**.

2. STOP mode release time

Supply of the clock is stopped: 18 μs to 65 μs

Wait

- When vectored interrupt servicing is carried out: 7 clocks
- When vectored interrupt servicing is not carried out: 1 clock

Caution To reduce the oscillation stabilization time after release from the STOP mode while CPU operates based on the high-speed system clock (X1 oscillation), switch the clock to the high-speed on-chip oscillator clock temporarily before executing the STOP instruction.

Remarks 1. The clock supply stop time varies depending on the temperature conditions and STOP mode period.

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

Table 18-2. State of Hardware After Receiving a Reset Signal

<R>

Hardware		After Reset Acknowledgment[Note]
Program counter (PC)		The contents of the reset vector table (00000H, 00001H) are set.
Stack pointer (SP)		Undefined
Program status word (PSW)		06H
RAM	Data memory	Undefined
	General-purpose registers	Undefined

Note During reset signal generation or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.

Remark For the state of the special function register (SFR) after receiving a reset signal, see **3.1.4 Special function register (SFR) area** and **3.1.5 Extended special function register (2nd SFR: 2nd Special Function Register) area**.

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

2. After an interrupt is generated, perform the processing according to **Figure 20-8 Processing Procedure After an Interrupt Is Generated**.

3. After a reset is released, perform the processing according to **Figure 20-9 Initial Setting of Interrupt and Reset Mode**.

Remark V_{POR}: POR power supply rise detection voltage

V_{PDR}: POR power supply fall detection voltage

The following methods for programming the flash memory are available.

The code flash memory can be rewritten to through serial programming using a flash memory programmer or an external device (UART communication), or through self-programming.

- Serial programming using flash memory programmer (see **24.4**)
  Data can be written to the flash memory on-board or off-board by using a dedicated flash memory programmer.

- Serial programming using external device (UART communication) (see **24.2**)
  Data can be written to the flash memory on-board through UART communication with an external device (microcontroller or ASIC).

- Self-programming (see **24.6**)
  The user application can execute self-programming of the code flash memory by using the flash self-programming library. [Note]

Note The self-programming function is not available in R5F10266 and R5F10366.

The data flash memory can be rewritten to by using the data flash library during user program execution (background operation). For access and writing to the data flash memory, see **24.8 Data Flash**.

Table 27-5. Operation List (4/17)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit data transfer	XCH	A, [HL+B]	2	2	–	A ←→ (HL+B)			
		A, ES:[HL+B]	3	3	–	A ←→ ((ES, HL)+B)			
		A, [HL+C]	2	2	–	A ←→ (HL+C)			
		A, ES:[HL+C]	3	3	–	A ←→ ((ES, HL)+C)			
	ONEB	A	1	1	–	A ← 01H			
		X	1	1	–	X ← 01H			
		B	1	1	–	B ← 01H			
		C	1	1	–	C ← 01H			
		!addr16	3	1	–	(addr16) ← 01H			
		ES:!addr16	4	2	–	(ES, addr16) ← 01H			
		saddr	2	1	–	(saddr) ← 01H			
	CLRB	A	1	1	–	A ← 00H			
		X	1	1	–	X ← 00H			
		B	1	1	–	B ← 00H			
		C	1	1	–	C ← 00H			
		!addr16	3	1	–	(addr16) ← 00H			
		ES:!addr16	4	2	–	(ES,addr16) ← 00H			
		saddr	2	1	–	(saddr) ← 00H			
	MOVS	[HL+byte], X	3	1	–	(HL+byte) ← X	×		×
		ES:[HL+byte], X	4	2	–	(ES, HL+byte) ← X	×		×
16-bit data transfer	MOVW	rp, #word	3	1	–	rp ← word			
		saddrp, #word	4	1	–	(saddrp) ← word			
		sfrp, #word	4	1	–	sfrp ← word			
		AX, rp Note 3	1	1	–	AX ← rp			
		rp, AX Note 3	1	1	–	rp ← AX			
		AX, !addr16	3	1	4	AX ← (addr16)			
		!addr16, AX	3	1	–	(addr16) ← AX			
		AX, ES:!addr16	4	2	5	AX ← (ES, addr16)			
		ES:!addr16, AX	4	2	–	(ES, addr16) ← AX			
		AX, saddrp	2	1	–	AX ← (saddrp)			
		saddrp, AX	2	1	–	(saddrp) ← AX			
		AX, sfrp	2	1	–	AX ← sfrp			
		sfrp, AX	2	1	–	sfrp ← AX			

Notes 1. Number of CPU clocks (f_{CLK}) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

2. Number of CPU clocks (f_{CLK}) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

3. Except rp = AX

Remark Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

(9/17)

Edition	Description	Chapter
1.10	Modification of description in 28.8 Flash Memory Programming Characteristics	CHAPTER 28 ELECTRICAL SPECIFICATIONS
1.00	Addition of products of industrial application	Throughout
	Renamed interval timer (unit) to 12-bit interval timer	
	Addition of pin name of the peripheral I/O redirection function	
	Modification of reset processing time	
	Deletion of LIN communication function	
	Renamed V_{LVI}, V_{LVIH}, V_{LVIL} to V_{LVD}, V_{LVDH}, V_{LVDL} (LVD detection voltage)	
	Renamed RAMTOP to RPE, renamed ITIF, ITMK, ITKAPR0, ITKAPR1 to TMKAIF, TMKAMK, TMKAPR0, TMKAPR1 (interrupt source, flag)	
	Addition of description to 1.1 Features	CHAPTER 1 OUTLINE
	Modification of description in 1.2 Ordering Information	
	Addition of Figure 1-1. Part Number, Memory Size, and Package of RL78/G12	
	Addition and Modification of description in 1.6 Outline of Functions	
	Modification of description in 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)	
	Modification of description in 2.3 Pin I/O Circuits and Recommended Connection of Unused Pins	
	Addition of remark to Table 3-1. Correspondence Between Address Values and Block Numbers in Flash Memory	CHAPTER 3 CPU ARCHITECTURE
	Addition of product in Table 3-2. Internal ROM Capacity	
	Addition of INTFL to Table 3-3. Vector Table (20-, 24-pin products)	
	Modification of description in 3.1.2 Mirror area	
	Addition of description to Caution in Table 3-5. Internal RAM Capacity	
	Modification of Figure 3-23. Outline of Table Indirect Addressing	
	Addition of setting of registers when using port xx to Table 4-2 to 4-4, 4-6 to 4-12, 4-14 to 4-20	CHAPTER 4 PORT FUNCTIONS
	Modification of block diagrams for Pxxx	
	Addition of description to (3) Port 2	
	Addition of description to (4) Port 4	
	Addition of Note to (6) Port 12	
	Addition of description to (3) Port 2	
	Addition of description to (2) Port register (Pxx)	
	Addition of description to (3) Pull-up resistor option registers (PUxx)	
	Modification of description in Figure 4-37. Format of Port Output Mode Register	
	Addition of Caution to Figure 4-38. Format of Port Mode Control Register	
	Addition of description to (8) Peripheral I/O redirection register (PIOR)	
	Addition of description to 4.4.1 Writing to I/O port, and 4.4.3 Operations on I/O port	
	Addition of description to 4.4.4 Connecting to external device with different potential (1.8 V, 2.5 V, 3 V)	
	Addition of description to 4.5 Settings of Port Mode Register, and Output Latch When Using Alternate Function	
	Addition of 4.6.2 Notes on specifying the pin settings	
	Addition of description to 5.1 (1) <2> High-speed on-chip oscillator	CHAPTER 5 CLOCK GENERATOR
	Addition of Caution to Figure 5-2. Format of Clock Operation Mode Control Register (CMC)	
	Modification of Figure 5-1. Block Diagram of Clock Generator	
	Modification of description in Table 5-2. Condition Before Stopping Clock Oscillation and Flag Setting	
	Deletion of Note to Figure 5-7. Format of Peripheral Enable Register 0 (PER0) (1/2)	
	Addition of description to (7) Operation speed mode control register (OSMC)	
	Modification of Caution 3 to Figure 5-9 Format of High-Speed On-Chip Oscillator Frequency Selection Register (HOCODIV)	

(12/17)

Edition	Description	Chapter
1.00	Modification of description to Figure 12-6. Format of IICA Control Register 00 (IICCTL00)	CHAPTER 12 SERIAL INTERFACE IICA
	Addition of description to Figure 12-7. Format of IICA Status Register 0 (IICS0)	
	Modification of Figure 12-28, 29, 30	
	Modification of Figure 13-1. Block Diagram of Multiplier and Divider/Multiply-Accumulator	CHAPTER 13 MULTIPLIER AND DIVIDER/MULTIPLY ACCUMULATOR
	Modification of value to Figure 13-6. Timing Diagram of Multiplication (Unsigned) Operation (2 × 3 = 6)	
	Addition of description to 13.4.5 Division operation	
	Addition of description	CHAPTER 15 INTERRUPT FUNCTION
	Addition of description to Table 15-1 and 15-2. Interrupt Source List	
	Addition of INTFL to Table 15-3 and 15-4. Flags Corresponding to Interrupt Request Sources	
	Modification of description to Table 15-5. Time from Generation of Maskable Interrupt Until Servicing	
	Modification of Figure 15-12. Interrupt Request Acknowledgment Timing (Maximum Time)	
	Modification of Table 15-6. Relationship Between Interrupt Requests Enabled for Multiple Interrupt Servicing During Interrupt Servicing	
	Addition and modification of description to Though out	CHAPTER 16 KEY INTERRUPT FUNCTION
	Modification of Caution to (3) SNOOZE mode	CHAPTER 17 STANDBY FUNCTION
	Modification of description to Table 17-1. Operating Statuses in HALT Mode	
	Addition and modification of release to standby function, wait time for SNOOZE status	
	Addition of description to Table 18-1. Operation Statuses During Reset Period	CHAPTER 18 RESET FUNCTION
	Addition of description to Table 18-2. Hardware Statuses After Reset Acknowledgment	
	Addition of Note to Table 18-2. Hardware Statuses After Reset Acknowledgment	
	Addition and modification of 19.1 Functions of Power-on-reset Circuit, 19.3 Operation of Power-on-reset Circuit	CHAPTER 19 POWERON-RESET CIRCUIT
	Modification of Figure 20-1. Block Diagram of Voltage Detector	CHAPTER 20 VOLTAGE DETECTOR
	Modification of description to Figure 20-2. Format of Voltage Detection Register (LVIM)	
	Addition of description to Figure 20-3. Format of Voltage Detection Level Select Register (LVIS)	
	Addition of Caution to Table 20-1. LVD Operation Mode and Detection Voltage Settings for User Option Byte (000C1H)	
	Modification of Figure 20-4, 20-5, 20-6	
	Addition of description to Figure 20-7, 20-8	
	Addition of description to 21.3.1 CRC operation function (general-purpose CRC)	CHAPTER 21 SAFETY FUNCTIONS
	Addition of description to Figure 21-3. CRC Operation Function (General-Purpose CRC)	
	Addition of description to Figure 21-5. Format of Invalid Memory Access Detection Control Register (IAWCTL)	
	Modification of Figure 21-7. Invalid access detection area	
	Addition of description to 21.3.7 A/D test function	
	Addition of Figure (moved from description of CHAPTER 2 PIN FUNCTIONS of the previous edition)	CHAPTER 22 REGULATOR
	Addition of description to (2) 000C1H	CHAPTER 23 OPTION BYTE
	Addition of description to Figure 23-1. Format of User Option Byte (000C0H)	
	Modification and addition of Caution to Figure 23-2. Format of User Option Byte (000C1H)	
	Deletion of description to 24.1.1 Programming environment	CHAPTER 24 FLASH MEMORY
	Addition of description to 24.2 Writing to Flash Memory by Using External Device (that Incorporates UART)	
	Addition of description to Figure 24-8. Setting of Flash Memory Programming Mode	
	Addition of description to Table 24-5, Table 24-13. Programming Modes and Voltages at Which Data Can Be Written, Erased, or Verified	