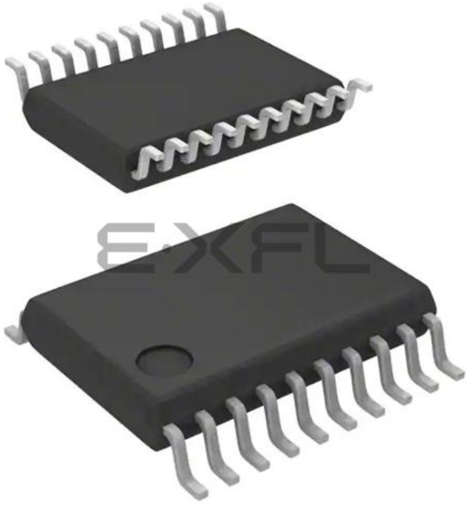
Renesas Electronics America Inc - R5F10267GSP#V0 Datasheet



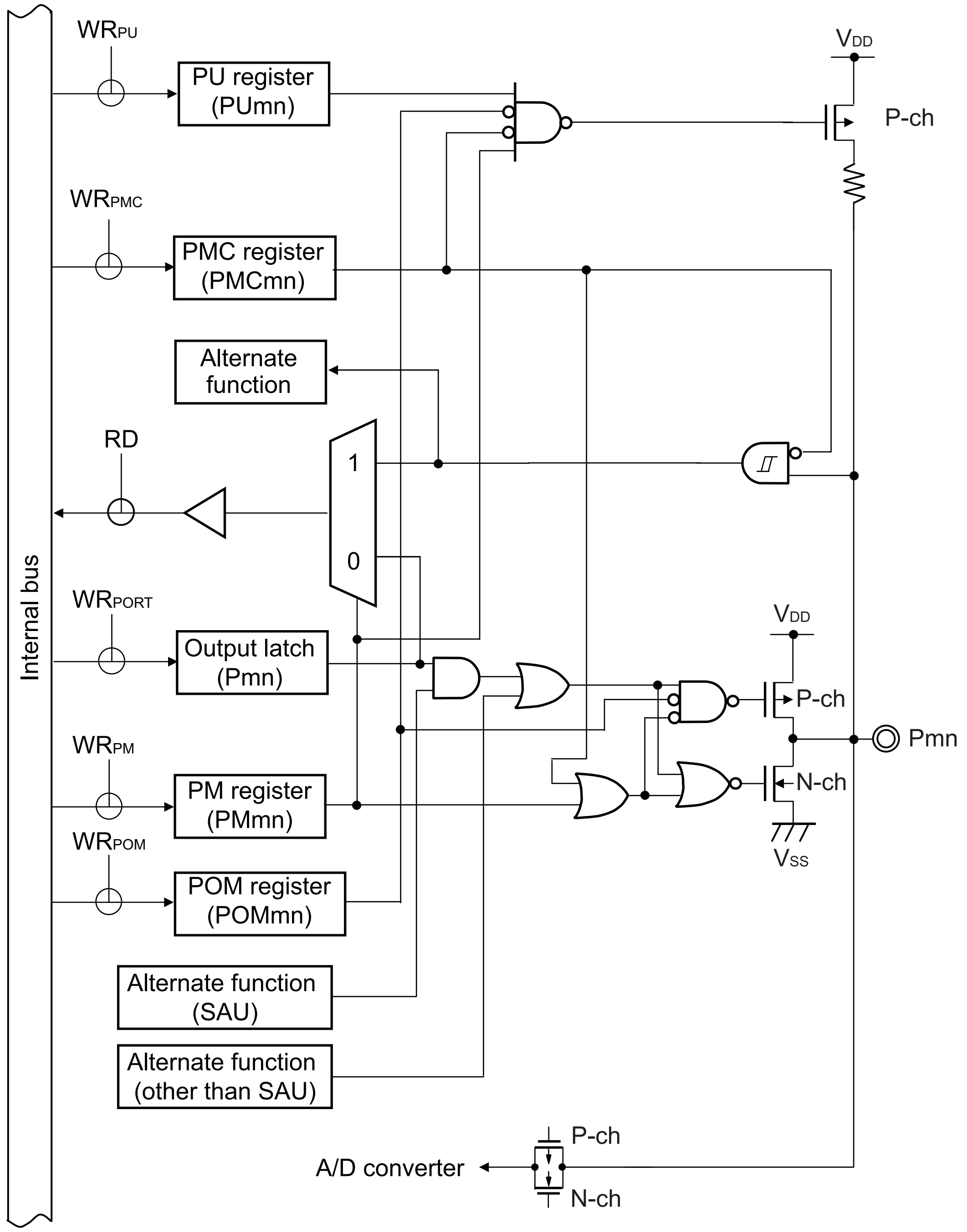
What is "Embedded - Microcontrollers"?

"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	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 11x8/10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°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/r5f10267gsp-v0

Figure 2-9. Pin Block Diagram for Pin Type 7-3-2

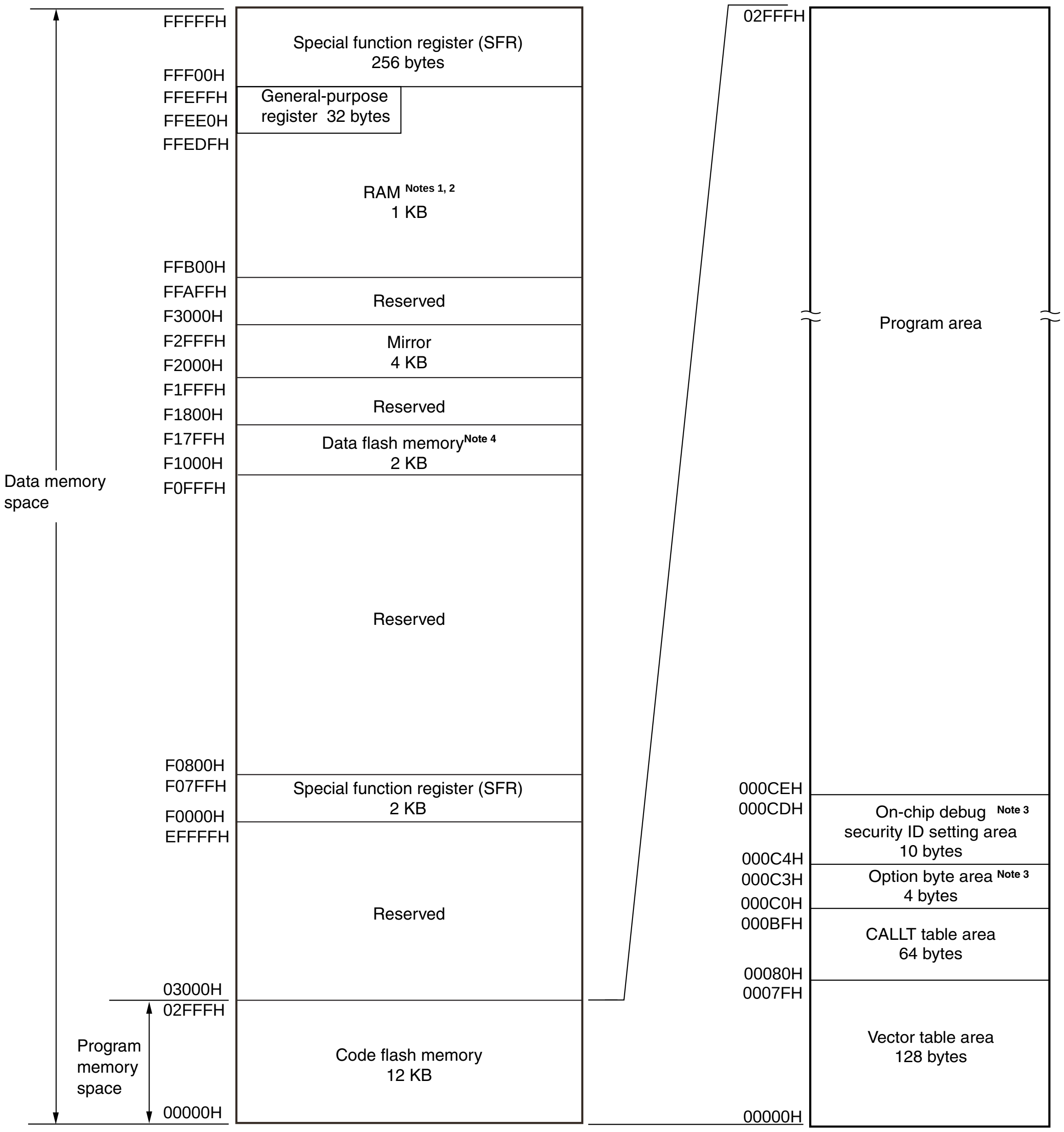


Remarks 1. For alternate functions, see **2.1 Port Functions**.

2. SAU: Serial array unit

<R> **Caution** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

Figure 3-4. Memory Map for the R5F10x69, R5F10x79, and R5F10xA9 (x = 2 or 3)



<R> **Notes 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DMA transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory. For R5F10x69 and R5F10x79, the RAM area used by the flash library starts at FFB00H. For the RAM areas used by the flash library, see Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944).

2. Instructions can be executed from the RAM area excluding the general-purpose register area.

3. Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

4. The areas are reserved in the R5F10369, R5F10379, and R5F103A9.

Caution While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area +10 bytes when instructions are fetched from RAM areas, respectively. Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see 21.3.2 RAM parity error detection.

6.9.2 Operation as PWM function

Two channels can be used as a set to generate a pulse of any period and duty factor.

The period and duty factor of the output pulse can be calculated by the following expressions.

Pulse period = {Set value of TDR0n (master) + 1} × Count clock period
Duty factor [%] = {Set value of TDR0p (slave)}/{Set value of TDR0n (master) + 1} × 100
0% output: Set value of TDR0p (slave) = 0000H
100% output: Set value of TDR0p (slave) ≥ {Set value of TDR0n (master) + 1}

Remark Although the duty factor exceeds 100% if the set value of TDR0p (slave) > (set value of TDR0n (master) + 1), it summarizes to 100% output.

The master channel operates in the interval timer mode. If the channel start trigger bit (TS0n) of timer channel start register 0 (TS0) is set to 1, an interrupt (INTTM0n) is output, the value set to timer data register 0n (TDR0n) is loaded to timer count register 0n (TCR0n), and the counter counts down in synchronization with the count clock. When the counter reaches 0000H, INTTM0n is output, the value of the TDR0n register is loaded again to the TCR0n register, and the counter counts down. This operation is repeated until the channel stop trigger bit (TT0n) of timer channel stop register 0 (TT0) is set to 1.

If two channels are used to output a PWM waveform, the period until the master channel counts down to 0000H is the PWM output (TO0p) cycle.

The slave channel operates in one-count mode. By using INTTM0n from the master channel as a start trigger, the TCR0p register loads the value of the TDR0p register and the counter counts down to 0000H. When the counter reaches 0000H, it outputs INTTM0p and waits until the next start trigger (INTTM0n from the master channel) is generated.

If two channels are used to output a PWM waveform, the period until the slave channel counts down to 0000H is the PWM output (TO0p) duty.

PWM output (TO0p) goes to the active level one clock after the master channel generates INTTM0n and goes to the inactive level when the TCR0p register of the slave channel becomes 0000H.

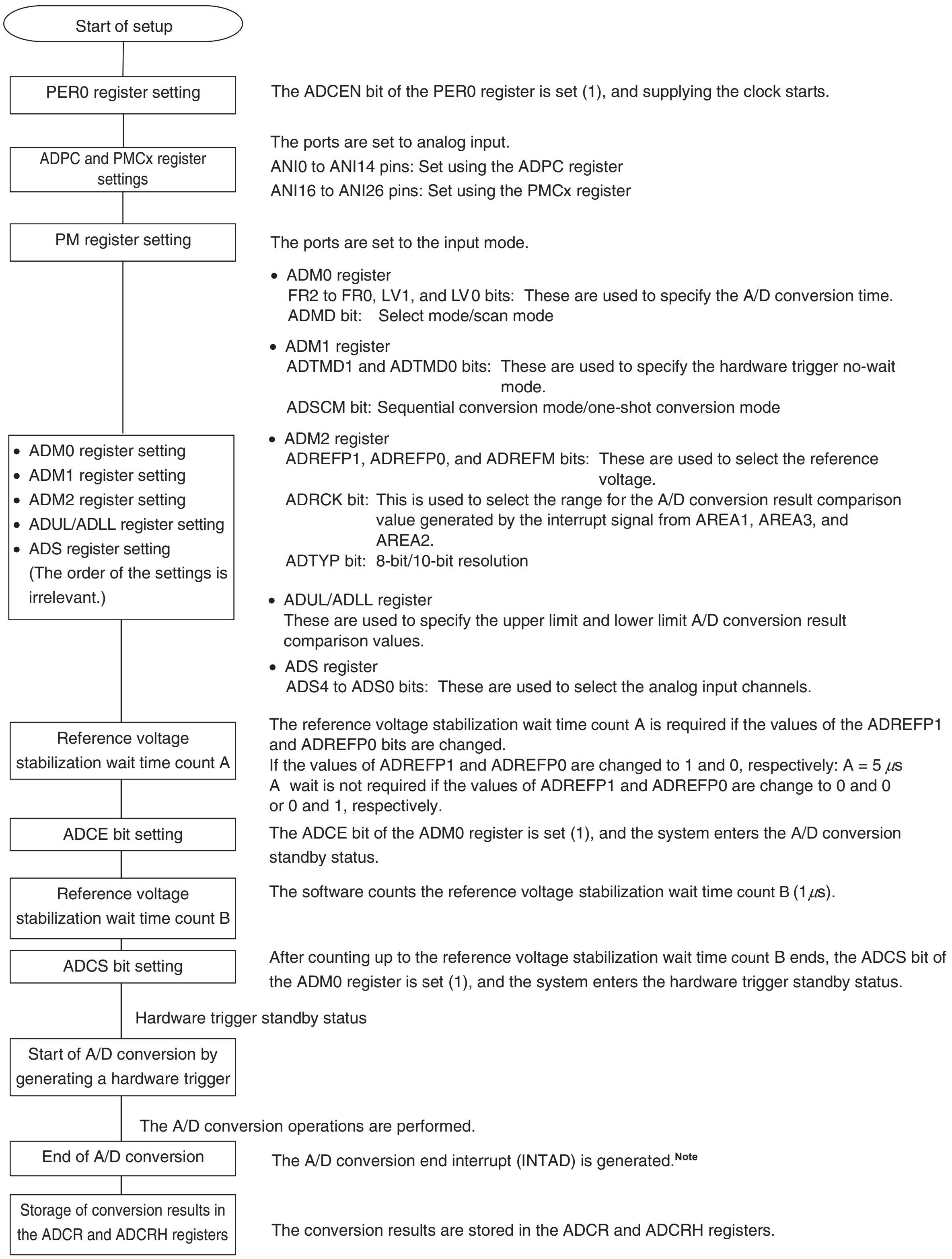
Caution To rewrite both timer data register 0n (TDR0n) of the master channel and the TDR0p register of the slave channel, a write access is necessary two times. The timing at which the values of the TDR0n and TDR0p registers are loaded to the TCR0n and TCR0p registers is upon occurrence of INTTM0n of the master channel. Thus, when rewriting is performed split before and after occurrence of INTTM0n of the master channel, the TO0p pin cannot output the expected waveform. To rewrite both the TDR0n register of the master and the TDR0p register of the slave, therefore, be sure to rewrite both the registers immediately after INTTM0n is generated from the master channel.

Remark n: Channel number (n = 0, 2, 4, 6)
p: Slave channel number (n < p ≤ 7)

10.7.2 Setting up hardware trigger no-wait mode

Figure 10-30. Setting up Hardware Trigger No-Wait Mode

<R>


Start of setup
PER0 register setting
ADPC and PMCx register settings
PM register setting
• ADM0 register setting
• ADM1 register setting
• ADM2 register setting
• ADUL/ADLL register setting
• ADS register setting
(The order of the settings is irrelevant.)
Reference voltage stabilization wait time count A
ADCE bit setting
Reference voltage stabilization wait time count B
ADCS bit setting
Hardware trigger standby status
Start of A/D conversion by generating a hardware trigger
The A/D conversion operations are performed.
End of A/D conversion
Storage of conversion results in the ADCR and ADCRH registers

The ADCEN bit of the PER0 register is set (1), and supplying the clock starts.

The ports are set to analog input.
ANI0 to ANI14 pins: Set using the ADPC register
ANI16 to ANI26 pins: Set using the PMCx register

The ports are set to the input mode.

- ADM0 register
  FR2 to FR0, LV1, and LV0 bits: These are used to specify the A/D conversion time.
  ADMD bit: Select mode/scan mode
- ADM1 register
  ADTMD1 and ADTMD0 bits: These are used to specify the hardware trigger no-wait mode.
  ADSCM bit: Sequential conversion mode/one-shot conversion mode
- ADM2 register
  ADREFP1, ADREFP0, and ADREFM bits: These are used to select the reference voltage.
  ADRCK bit: This is used to select the range for the A/D conversion result comparison value generated by the interrupt signal from AREA1, AREA3, and AREA2.
  ADTYP bit: 8-bit/10-bit resolution
- ADUL/ADLL register
  These are used to specify the upper limit and lower limit A/D conversion result comparison values.
- ADS register
  ADS4 to ADS0 bits: These are used to select the analog input channels.

The reference voltage stabilization wait time count A is required if the values of the ADREFP1 and ADREFP0 bits are changed.
If the values of ADREFP1 and ADREFP0 are changed to 1 and 0, respectively: A = 5 μs
A wait is not required if the values of ADREFP1 and ADREFP0 are change to 0 and 0 or 0 and 1, respectively.

The ADCE bit of the ADM0 register is set (1), and the system enters the A/D conversion standby status.

The software counts the reference voltage stabilization wait time count B (1 μs).

After counting up to the reference voltage stabilization wait time count B ends, the ADCS bit of the ADM0 register is set (1), and the system enters the hardware trigger standby status.

The A/D conversion end interrupt (INTAD) is generated.[Note]

The conversion results are stored in the ADCR and ADCRH registers.

Note Depending on the settings of the ADRCK bit and ADUL/ADLL register, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR and ADCRH registers.

CHAPTER 11 SERIAL ARRAY UNIT

Serial array unit 0 has two serial channels in 20- and 24-pinproducts and four serial channels in 30-pin products, and serial array unit 1 mounted 30-pin products, has two serial channels.

Each channel can achieve 3-wire serial (CSI), UART, and simplified I^2C communication.

Function assignment of each channel supported by the RL78/G12 is as shown below.

20- or 24-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I^2C
0	0	CSI00	UART0	IIC00[Note]
	1	CSI01[Note]		IIC01[Note]

30-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I^2C
0	0	CSI00	UART0	IIC00[Note]
	1	–		–
	2	–	UART1[Note]	–
	3	CSI11[Note]		IIC11[Note]
1	0	CSI20[Note]	UART2[Note]	IIC20[Note]
	1	–		–

Note Provided in the R5F102 products only.

A single channel cannot be used under multiple communication methods. When a different communication method is to be configured, use another channel.

When using CSI00, CSI20, IIC00, IIC20, UART0, UART1, or UART2, communication between devices with different voltages (1.8, 2.5, or 3 V) is possible, except when the serial I/O for UART0 of a 20- or 24-pin product is assigned to P6 by the setting of the I/O redirection register (PIOR1 = 1). For details about the settings, see **4.4.4 Handling different potentials (1.8 V, 2.5 V, and 3 V) by using I/O buffers.**

11.3.10 Serial channel enable status register m (SEm)

The SEm register indicates whether the data transmission/reception operation of each channel is enabled or disabled.

When 1 is written to a bit of serial channel start register m (SSm), the corresponding bit of this register is set to 1. When 1 is written to a bit of serial channel stop register m (STm), the corresponding bit of this register is cleared to 0.

If the operation of channel n is enabled, the value of the CKOmn bit (serial clock output of channel n) of serial output register m (SOm) cannot be rewritten by software, and a value is output from the serial clock pin according to the communication operation.

If the operation of channel n is disabled, the value of the CKOmn bit of the SOm register can be set by software and its value is output from the serial clock pin. In this way, any waveform, such as that of a start condition/stop condition, can be created by software.

The SEm register can be read by a 16-bit memory manipulation instruction.

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

Reset signal generation clears the SEm register to 0000H.

Figure 11-15. Format of Serial Channel Enable Status Register m (SEm)

Address: F0120H, F0121H (SE0) After reset: 0000H R

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SE0	0	0	0	0	0	0	0	0	0	0	0	0	SE03 Note	SE02 Note	SE01	SE00

Address: F0160H, F0161H (SE1) After reset: 0000H R

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SE1[Note 1]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SE11	SE11

SEmn	Indication of operation enable/disable status of channel n
0	Operation is disabled (stopped)
1	Operation is enabled.

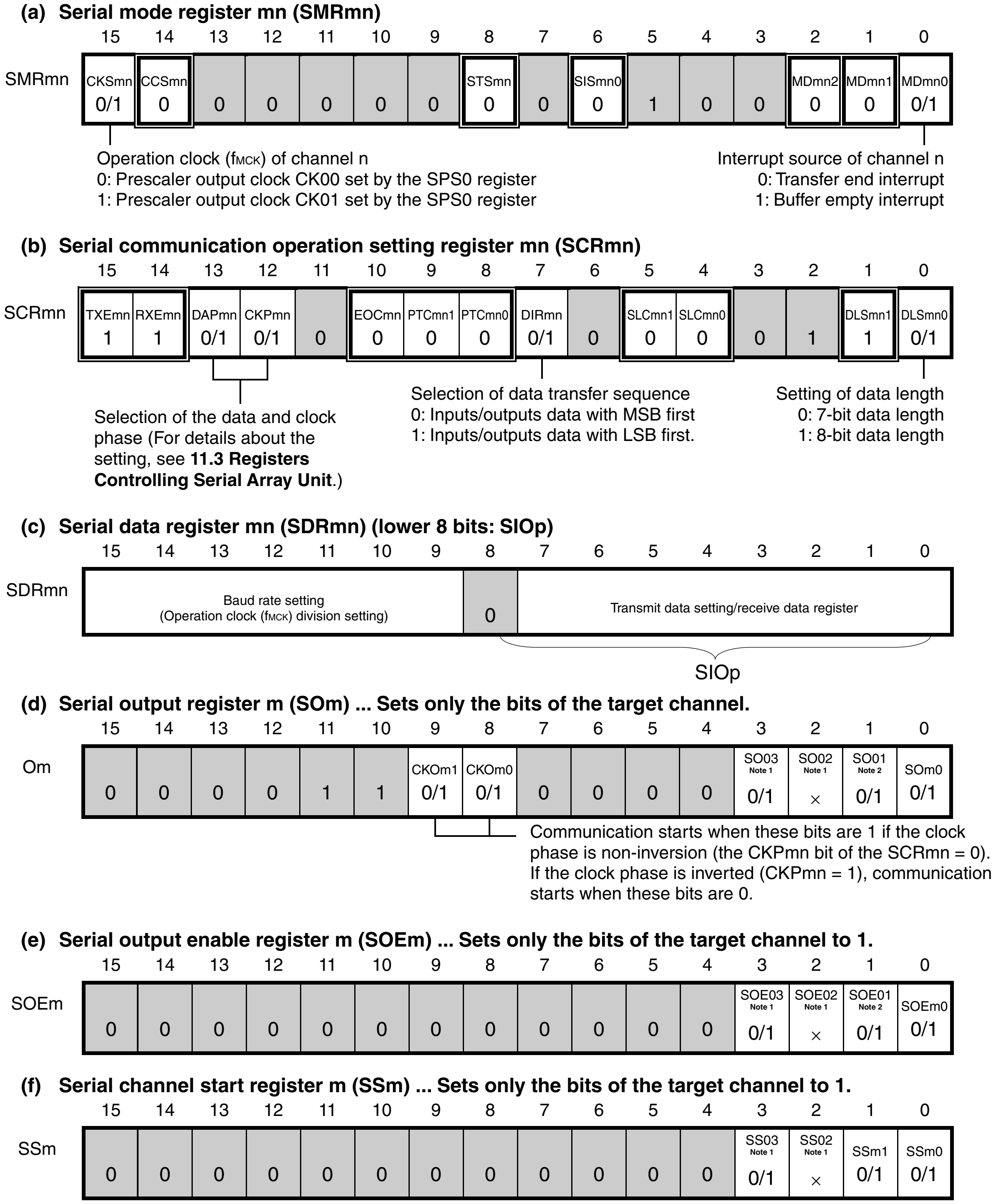
Note 30-pin product only.

Caution Be sure to clear bits 15 to 2 of the SE0 register for 20- or 24-pin products, bits 15 to 4 of the SE0 register for 30-pin products, and bits 15 to 2 of the SE1 register to "0".

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

(1) Register setting

Figure 11-41. Example of Contents of Registers for Master Transmission/Reception of 3-Wire Serial I/O (CSI00, CSI01, CSI11, CSI20) (1/2)



(a) Serial mode register mn (SMRmn)

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

Operation clock (fMCK) of channel n
0: Prescaler output clock CK00 set by the SPS0 register
1: Prescaler output clock CK01 set by the SPS0 register

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

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

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

Selection of the data and clock phase (For details about the setting, see **11.3 Registers Controlling Serial Array Unit**.)

Selection of data transfer sequence
0: Inputs/outputs data with MSB first
1: Inputs/outputs data with LSB first.

Setting of data length
0: 7-bit data length
1: 8-bit data length

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

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

SIOp

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

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Om	0	0	0	0	1	1	CKOm1 0/1	CKOm0 0/1	0	0	0	0	SO03 Note 1 0/1	SO02 Note 1 ×	SO01 Note 2 0/1	SOm0 0/1

Communication starts when these bits are 1 if the clock phase is non-inversion (the CKPmn bit of the SCRmn = 0). If the clock phase is inverted (CKPmn = 1), communication starts when these bits are 0.

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

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	SOE03 Note 1 0/1	SOE02 Note 1 ×	SOE01 Note 2 0/1	SOEm0 0/1

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

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SS03 Note 1 0/1	SS02 Note 1 ×	SSm1 0/1	SSm0 0/1

Notes 1. Provided in the serial array unit 0 of 30-pin products.

2. 20-, 24-pin products only.

Remarks 1. m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 11, 20), mn = 00, 01, 03, 10

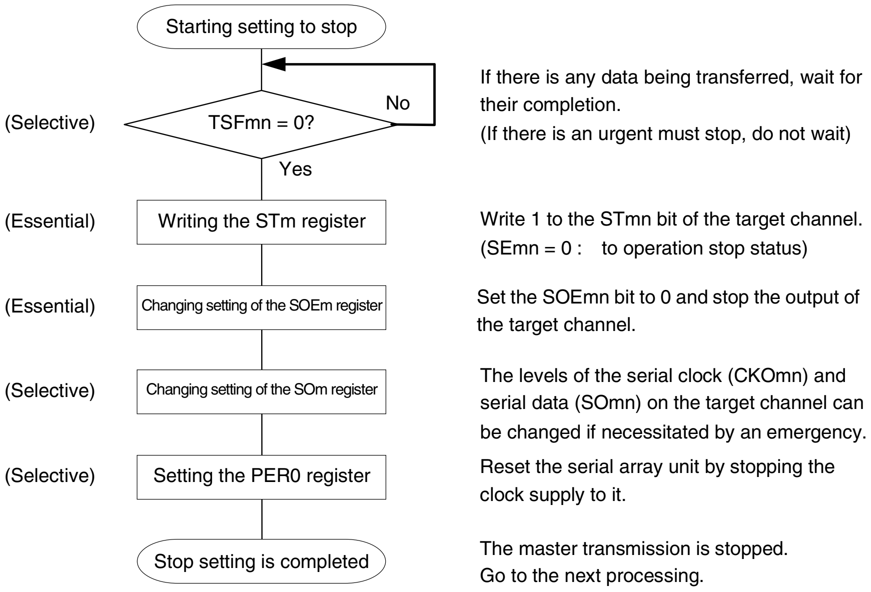
2. □: Setting is fixed in the CSI master transmission/reception mode

▨ : Setting disabled (set to the initial value)

×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

Figure 11-51. Procedure for Stopping Slave Transmission


Starting setting to stop
(Selective)
TSFmn = 0?
No
Yes
(Essential)
Writing the STm register
(Essential)
Changing setting of the SOEm register
(Selective)
Changing setting of the SOm register
(Selective)
Setting the PER0 register
Stop setting is completed

If there is any data being transferred, wait for their completion.
(If there is an urgent must stop, do not wait)

Write 1 to the STmn bit of the target channel.
(SEmn = 0 : to operation stop status)

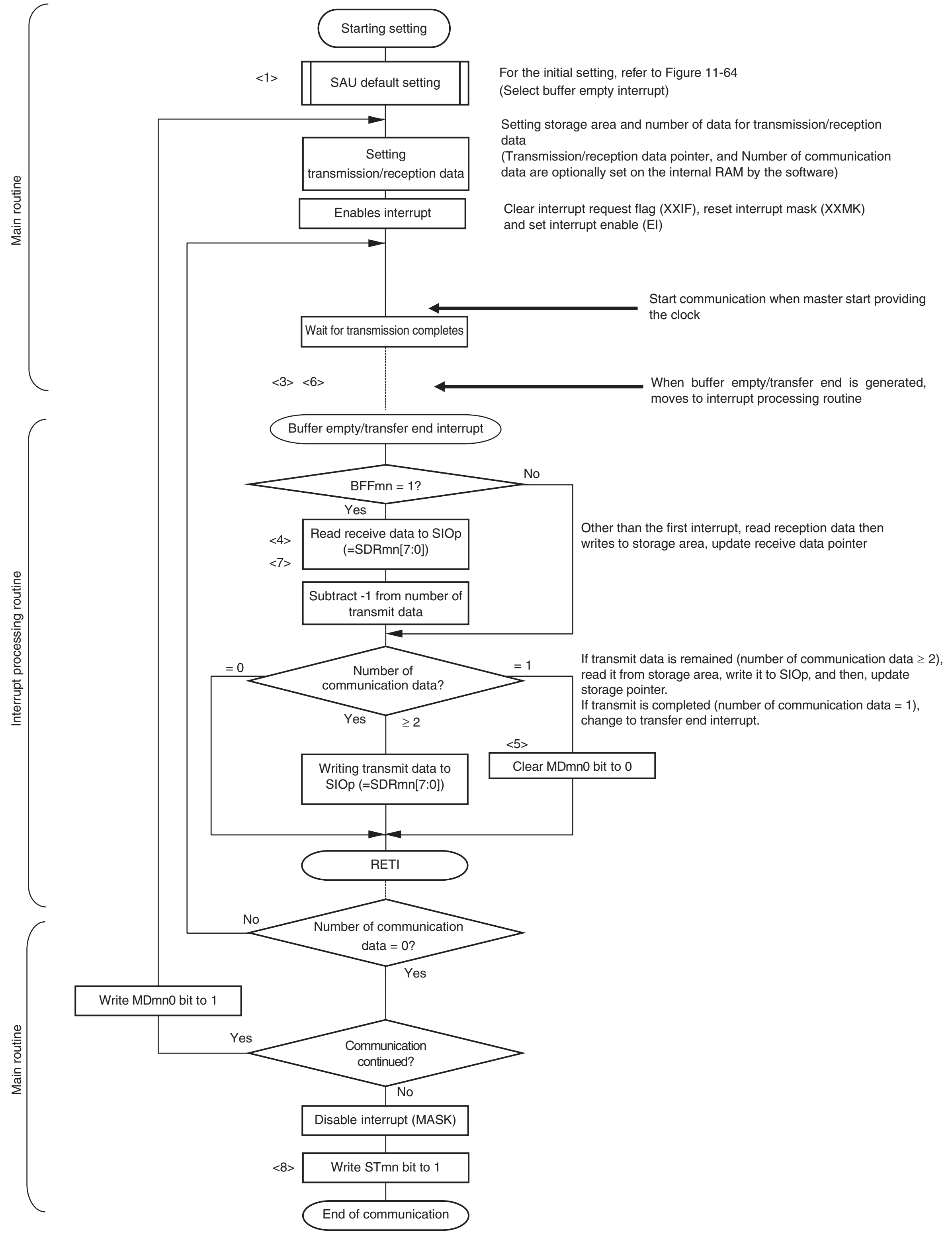
Set the SOEmn bit to 0 and stop the output of the target channel.

The levels of the serial clock (CKOmn) and serial data (SOmn) on the target channel can be changed if necessitated by an emergency.

Reset the serial array unit by stopping the clock supply to it.

The master transmission is stopped.
Go to the next processing.

Figure 11-70. Flowchart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode)


Main routine
Interrupt processing routine
Main routine
Starting setting
<1>
SAU default setting
For the initial setting, refer to Figure 11-64
(Select buffer empty interrupt)
Setting transmission/reception data
Setting storage area and number of data for transmission/reception data
(Transmission/reception data pointer, and Number of communication data are optionally set on the internal RAM by the software)
Enables interrupt
Clear interrupt request flag (XXIF), reset interrupt mask (XXMK) and set interrupt enable (EI)
Start communication when master start providing the clock
Wait for transmission completes
<3> <6>
When buffer empty/transfer end is generated, moves to interrupt processing routine
Buffer empty/transfer end interrupt
BFFmn = 1?
No
Yes
<4>
<7>
Read receive data to SIOp (=SDRmn[7:0])
Other than the first interrupt, read reception data then writes to storage area, update receive data pointer
Subtract -1 from number of transmit data
= 0
Number of communication data?
= 1
If transmit data is remained (number of communication data ≥ 2), read it from storage area, write it to SIOp, and then, update storage pointer.
If transmit is completed (number of communication data = 1), change to transfer end interrupt.
Yes
≥ 2
<5>
Writing transmit data to SIOp (=SDRmn[7:0])
Clear MDmn0 bit to 0
RETI
No
Number of communication data = 0?
Yes
Write MDmn0 bit to 1
Yes
Communication continued?
No
Disable interrupt (MASK)
<8>
Write STmn bit to 1
End of communication

Caution Be sure to set transmit data to the SIOp register before the clock from the master is started.

Remark <1> to <8> in the figure correspond to <1> to <8> in **Figure 11-69 Timing Chart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode)**.

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).

The channels supporting simplified I^{2}C (IIC00, IIC01, IIC11, IIC20) are channels 0, 1, 3 of SAU0 and channel 0 of SAU1.

20- or 24-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I^{2}C
0	0	CSI00	UART0	IIC00[Note]
	1	CSI01[Note]		IIC01[Note]

30-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I^{2}C
0	0	CSI00	UART0	IIC00[Note]
	1	–		–
	2	–	UART1[Note]	–
	3	CSI11[Note]		IIC11[Note]
1	0	CSI20[Note]	UART 2Note	IIC20[Note]
	1	–		–

Note Provided in the R5F102 products only.

Simplified I^{2}C (IIC00, IIC01, IIC11, IIC20) performs the following four types of communication operations.

- Address field transmission (See **11.7.1**.)
- Data transmission (See **11.7.2**.)
- Data reception (See **11.7.3**.)
- Stop condition generation (See **11.7.4**.)

(1) Register setting

Figure 11-105. Example of Contents of Registers for Data Reception of Simplified I²C (IIC00, IIC01, IIC11, IIC20)

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

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

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

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

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDRmn	Baud rate setting Note4							0	Dummy transmit data setting (FFH)							
									SIOr							

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

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

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

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

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

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

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

2. Only for 20, 24-pin products.

3. The values may change during operation, depending on the communication data.

4. Because the setting is completed by address field transmission, setting is not required.

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 IIC 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

12.6 Timing Charts

When using the I^{2}C bus mode, the master device outputs an address via the serial bus to select one of several slave devices as its communication partner.

After outputting the slave address, the master device transmits the TRC0 bit (bit 3 of the IICA status register 0 (IICS0)), which specifies the data transfer direction, and then starts serial communication with the slave device.

Figures 12-32 and 12-33 show timing charts of the data communication.

The IICA shift register 0 (IICA0)'s shift operation is synchronized with the falling edge of the serial clock (SCLA0). The transmit data is transferred to the SO latch and is output (MSB first) via the SDAA0 pin.

Data input via the SDAA0 pin is captured into IICA0 at the rising edge of SCLA0.

13.2.1 Multiplication/division data register A (MDAH, MDAL)

The MDAH and MDAL registers set the values that are used for a multiplication or division operation and store the operation result. They set the multiplier and multiplicand data in the multiplication mode or multiply-accumulator mode, and set the dividend data in the division mode. Furthermore, the operation result (quotient) is stored in the MDAH and MDAL registers in the division mode.

The MDAH and MDAL registers can be set by a 16-bit manipulation instruction.

Reset signal generation clears these registers to 0000H.

Figure 13-2. Format of Multiplication/Division Data Register A (MDAH, MDAL)

Address: FFFF0H, FFFF1H, FFFF2H, FFFF3H After reset: 0000H, 0000H R/W

Symbol	FFFF3H								FFFF2H							
MDAH	MDAH 15	MDAH 14	MDAH 13	MDAH 12	MDAH 11	MDAH 10	MDAH 9	MDAH 8	MDAH 7	MDAH 6	MDAH 5	MDAH 4	MDAH 3	MDAH 2	MDAH 1	MDAH 0

Symbol	FFFF1H								FFFF0H							
MDAL	MDAL 15	MDAL 14	MDAL 13	MDAL 12	MDAL 11	MDAL 10	MDAL 9	MDAL 8	MDAL 7	MDAL 6	MDAL 5	MDAL 4	MDAL 3	MDAL 2	MDAL 1	MDAL 0

Cautions 1. Do not rewrite the MDAH and MDAL registers values during division operation processing (when the multiplication/division control register (MDUC) value is 81H or C1H). The operation will be executed in this case, but the operation result will be an undefined value.

2. The MDAH and MDAL registers values read during division operation processing (when the MDUC register value is 81H or C1H) will not be guaranteed.

3. The data is in the two's complement format in either the multiplication mode (signed) or multiply-accumulator mode (signed).

The following table shows the functions of the MDAH and MDAL registers during operation execution.

Table 13-2. Functions of MDAH and MDAL Registers During Operation Execution

Operation Mode	Setting	Operation Result
Multiplication mode (unsigned) Multiply-accumulator mode (unsigned)	MDAH: Multiplier (unsigned) MDAL: Multiplicand (unsigned)	–
Multiplication mode (signed) Multiply-accumulator mode (signed)	MDAH: Multiplier (signed) MDAL: Multiplicand (signed)	–
Division mode (unsigned)	MDAH: Dividend (unsigned) (higher 16 bits) MDAL: Dividend (unsigned) (lower 16 bits)	MDAH: Division result (quotient) (unsigned) Higher 16 bits MDAL: Division result (quotient) (unsigned) Lower 16 bits

Figure 13-5. Format of Multiplication/Division Control Register (MDUC)

Address: F00E8H After reset: 00H R/W[Note 1]

Symbol	<7>	<6>	5	4	<3>	<2>	<1>	<0>
MDUC	DIVMODE	MACMODE	0	0	MDSM	MACOF	MACSF	DIVST

DIVMODE	MACMODE	MDSM	Operation mode selection
0	0	0	Multiplication mode (unsigned) (default)
0	0	1	Multiplication mode (signed)
0	1	0	Multiply-accumulator mode (unsigned)
0	1	1	Multiply-accumulator mode (signed)
1	0	0	Division mode (unsigned), generation of a division completion interrupt (INTMD)
1	1	0	Division mode (unsigned), not generation of a division completion interrupt (INTMD)
Other than above			Setting prohibited

MACOF	Overflow flag of multiply-accumulation result (accumulated value)
0	No overflow
1	With over flow
<Set condition> • For the multiply-accumulator mode (unsigned) The bit is set when the accumulated value goes outside the range from 00000000h to FFFFFFFFh. • For the multiply-accumulator mode (signed) The bit is set when the result of adding a positive product to a positive accumulated value exceeds 7FFFFFFFh and is negative, or when the result of adding a negative product to a negative accumulated value exceeds 80000000h and is positive.	

MACSF	Sign flag of multiply-accumulation result (accumulated value)
0	The accumulated value is positive.
1	The accumulated value is negative.
Multiply-accumulator mode (unsigned):	The bit is always 0.
Multiply-accumulator mode (signed):	The bit indicates the sign bit of the accumulated value.

DIVST[Note 2]	Division operation start/stop
0	Division operation processing complete
1	Starts division operation/division operation processing in progress

Notes 1. Bits 1 and 2 are read-only bits.

2. The DIVST bit can only be set (1) in the division mode. In the division mode, division operation is started by setting (1) the DIVST bit. The DIVST bit is automatically cleared (0) when the operation ends. In the multiplication mode, operation is automatically started by setting the multiplier and multiplicand to multiplication/division data register A (MDAH, MDAL), respectively.

Cautions 1. Do not rewrite the DIVMODE, MDSM bits during operation processing (while the DIVST bit is 1). If it is rewritten, the operation result will be an undefined value.

2. The DIVST bit cannot be cleared (0) by using software during division operation processing (while the DIVST bit is 1).

13.4.4 Multiply-accumulation (signed) operation

- Initial setting

  <1> Set the multiplication/division control register (MDUC) to 48H.

  <2> Set the initial accumulated value of higher 16 bits to multiplication/division data register C (H) (MDCH).

  (<3> If the accumulated value in the MDCH register is negative, the MACSF bit is set to 1.)

  <4> Set the initial accumulated value of lower 16 bits to multiplication/division data register C (L) (MDCL).

  <5> Set the multiplicand to multiplication/division data register A (L) (MDAL).

  <6> Set the multiplier to multiplication/division data register A (H) (MDAH).

  (There is no preference in the order of executing steps <2>, <4>, and <5>. Multiplication operation is automatically started when the multiplier is set to the MDAH register of <6>, respectively.)

- During operation processing

  <7> The multiplication operation finishes in one clock cycle.

  (The multiplication result is stored in multiplication/division data register B (L) (MDBL) and multiplication/division data register B (H) (MDBH).)

  <8> After <7>, the multiply-accumulation operation finishes in one additional clock cycle. (There is a wait of at least two clock cycles after specifying the initial settings is finished (<6>).)

- Operation end

  <9> If the accumulated value stored in the MDCL and MDCH registers is positive, the MACSF bit is cleared to 0.

  <10> Read the accumulated value (lower 16 bits) from the MDCL register.

  <11> Read the accumulated value (higher 16 bits) from the MDCH register.

  (There is no preference in the order of executing steps <10> and <11>.)

  (<12> If the result of the multiply-accumulation operation causes an overflow, the MACOF bit is set to 1, INTMD signal is occurred.)

- Next operation

  <13> Start with the initial settings of each step to change the operation mode. When the same operation mode is used sequentially, settings <1> and <5> can be omitted.

Caution The data is in the two's complement format in multiply-accumulation (signed) operation.

Remark Steps <1> to <12> correspond to <1> to <12> in Figure 13-9.

Table 27-5. Operation List (8/17)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	SUBC	A, #byte	2	1	–	A, CY ← A – byte – CY	×	×	×
		saddr, #byte	3	2	–	(saddr), CY ← (saddr) – byte – CY	×	×	×
		A, r Note 3	2	1	–	A, CY ← A – r – CY	×	×	×
		r, A	2	1	–	r, CY ← r – A – CY	×	×	×
		A, !addr16	3	1	4	A, CY ← A – (addr16) – CY	×	×	×
		A, ES:!addr16	4	2	5	A, CY ← A – (ES, addr16) – CY	×	×	×
		A, saddr	2	1	–	A, CY ← A – (saddr) – CY	×	×	×
		A, [HL]	1	1	4	A, CY ← A – (HL) – CY	×	×	×
		A, ES:[HL]	2	2	5	A,CY ← A – (ES, HL) – CY	×	×	×
		A, [HL+byte]	2	1	4	A, CY ← A – (HL+byte) – CY	×	×	×
		A, ES:[HL+byte]	3	2	5	A,CY ← A – ((ES, HL)+byte) – CY	×	×	×
		A, [HL+B]	2	1	4	A, CY ← A – (HL+B) – CY	×	×	×
		A, ES:[HL+B]	3	2	5	A,CY ← A – ((ES, HL)+B) – CY	×	×	×
		A, [HL+C]	2	1	4	A, CY ← A – (HL+C) – CY	×	×	×
		A, ES:[HL+C]	3	2	5	A, CY ← A – ((ES:HL)+C) – CY	×	×	×
	AND	A, #byte	2	1	–	A ← A ∧ byte	×		
		saddr, #byte	3	2	–	(saddr) ← (saddr) ∧ byte	×		
		A, r Note 3	2	1	–	A ← A ∧ r	×		
		r, A	2	1	–	R ← r ∧ A	×		
		A, !addr16	3	1	4	A ← A ∧ (addr16)	×		
		A, ES:!addr16	4	2	5	A ← A ∧ (ES:addr16)	×		
		A, saddr	2	1	–	A ← A ∧ (saddr)	×		
		A, [HL]	1	1	4	A ← A ∧ (HL)	×		
		A, ES:[HL]	2	2	5	A ← A ∧ (ES:HL)	×		
		A, [HL+byte]	2	1	4	A ← A ∧ (HL+byte)	×		
		A, ES:[HL+byte]	3	2	5	A ← A ∧ ((ES:HL)+byte)	×		
		A, [HL+B]	2	1	4	A ← A ∧ (HL+B)	×		
		A, ES:[HL+B]	3	2	5	A ← A ∧ ((ES:HL)+B)	×		
		A, [HL+C]	2	1	4	A ← A ∧ (HL+C)	×		
		A, ES:[HL+C]	3	2	5	A ← A ∧ ((ES:HL)+C)	×		

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 r = A

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.

(3) When reference voltage (+) = V_{DD} (ADREFP1 = 0, ADREFP0 = 0), reference voltage (–) = V_{SS} (ADREFM = 0), target pin: ANI0 to ANI3, ANI16 to ANI22, internal reference voltage, and temperature sensor output voltage

(T_A = –40 to +105°C, 2.4 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V, Reference voltage (+) = V_{DD}, Reference voltage (–) = V_{SS})

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES			8		10	bit
Overall error[Note 1]	AINL	10-bit resolution			1.2	±7.0	LSB
Conversion time	t_{CONV}	10-bit resolution Target pin: ANI0 to ANI3, ANI16 to ANI22	3.6 V ≤ V_{DD} ≤ 5.5 V	2.125		39	μs
			2.7 V ≤ V_{DD} ≤ 5.5 V	3.1875		39	μs
			2.4 V ≤ V_{DD} ≤ 5.5 V	17		39	μs
Conversion time	t_{CONV}	10-bit resolution Target pin: internal reference voltage, and temperature sensor output voltage (HS (high-speed main) mode)	3.6 V ≤ V_{DD} ≤ 5.5 V	2.375		39	μs
			2.7 V ≤ V_{DD} ≤ 5.5 V	3.5625		39	μs
			2.4 V ≤ V_{DD} ≤ 5.5 V	17		39	μs
Zero-scale error[Notes 1, 2]	EZS	10-bit resolution				±0.60	%FSR
Full-scale error[Notes 1, 2]	EFS	10-bit resolution				±0.60	%FSR
Integral linearity error[Note 1]	ILE	10-bit resolution				±4.0	LSB
Differential linearity error[Note 1]	DLE	10-bit resolution				±2.0	LSB
Analog input voltage	V_{AIN}	ANI0 to ANI3, ANI16 to ANI22		0		V_{DD}	V
		Internal reference voltage (HS (high-speed main) mode)		V_{BGR}[Note 3]			V
		Temperature sensor output voltage (HS (high-speed main) mode)		V_{TMPS25}[Note 3]			V

Notes 1. Excludes quantization error (±1/2 LSB).

2. This value is indicated as a ratio (%FSR) to the full-scale value.

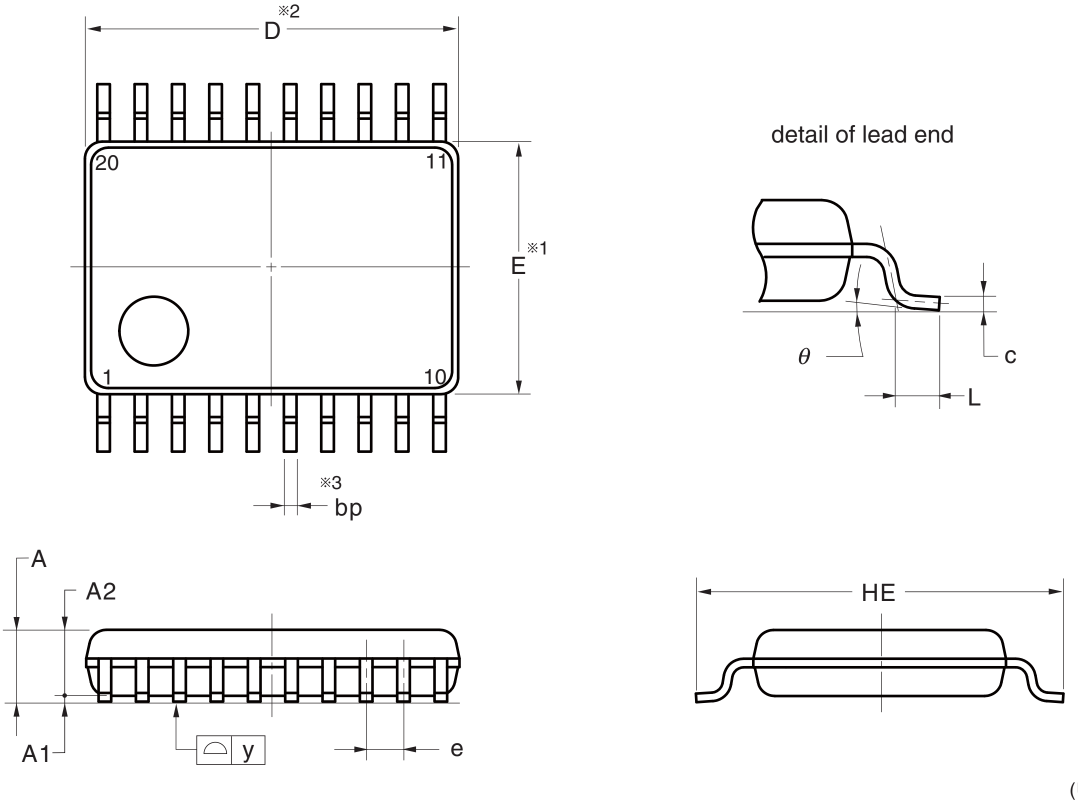
3. Refer to **29.6.2 Temperature sensor/internal reference voltage characteristics**.

CHAPTER 30 PACKAGE DRAWINGS

30.1 20-pin products

R5F1026AASP, R5F10269ASP, R5F10268ASP, R5F10267ASP, R5F10266ASP
R5F1036AASP, R5F10369ASP, R5F10368ASP, R5F10367ASP, R5F10366ASP
R5F1026ADSP, R5F10269DSP, R5F10268DSP, R5F10267DSP, R5F10266DSP
R5F1036ADSP, R5F10369DSP, R5F10368DSP, R5F10367DSP, R5F10366DSP
<R> R5F1026AGSP, R5F10269GSP, R5F10268GSP, R5F10267GSP, R5F10266GSP

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LSSOP20-4.4x6.5-0.65	PLSP0020JB-A	P20MA-65-NAA-1	0.1



(UNIT:mm)

ITEM	DIMENSIONS
D	6.50±0.10
E	4.40±0.10
HE	6.40±0.20
A	1.45 MAX.
A1	0.10±0.10
A2	1.15
e	0.65±0.12
bp	$0.22^{+0.10}_{-0.05}$
c	$0.15^{+0.05}_{-0.02}$
L	0.50±0.20
y	0.10
θ	0° to 10°

NOTE

1.Dimensions “※1” and “※2” do not include mold flash.

2.Dimension “※3” does not include trim offset.

RL78/G12 User’s Manual: Hardware

Publication Date: Rev.2.10 Mar 25, 2016

Published by: Renesas Electronics Corporation