Renesas Electronics America Inc - UPD78F1508AGF-GAT-AX Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

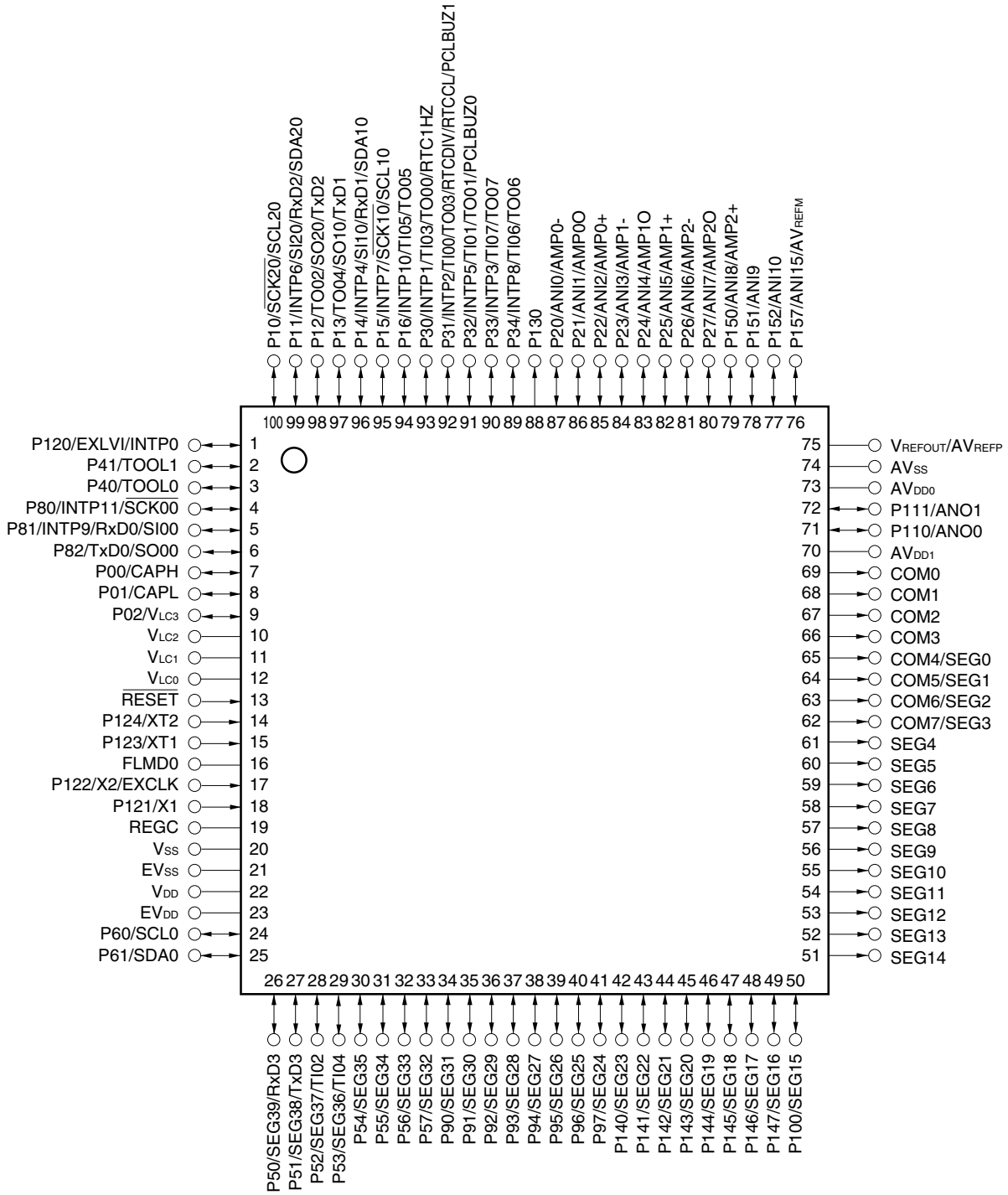
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	78K/0R
Core Size	16-Bit
Speed	20MHz
Connectivity	CSI, I²C, LINbus, UART/USART
Peripherals	DMA, LCD, LVD, POR, PWM, WDT
Number of I/O	76
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	7K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 12x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	128-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f1508agf-gat-ax

1.3.2 78K0R/LG3

<R> (1) μPD78F150xA

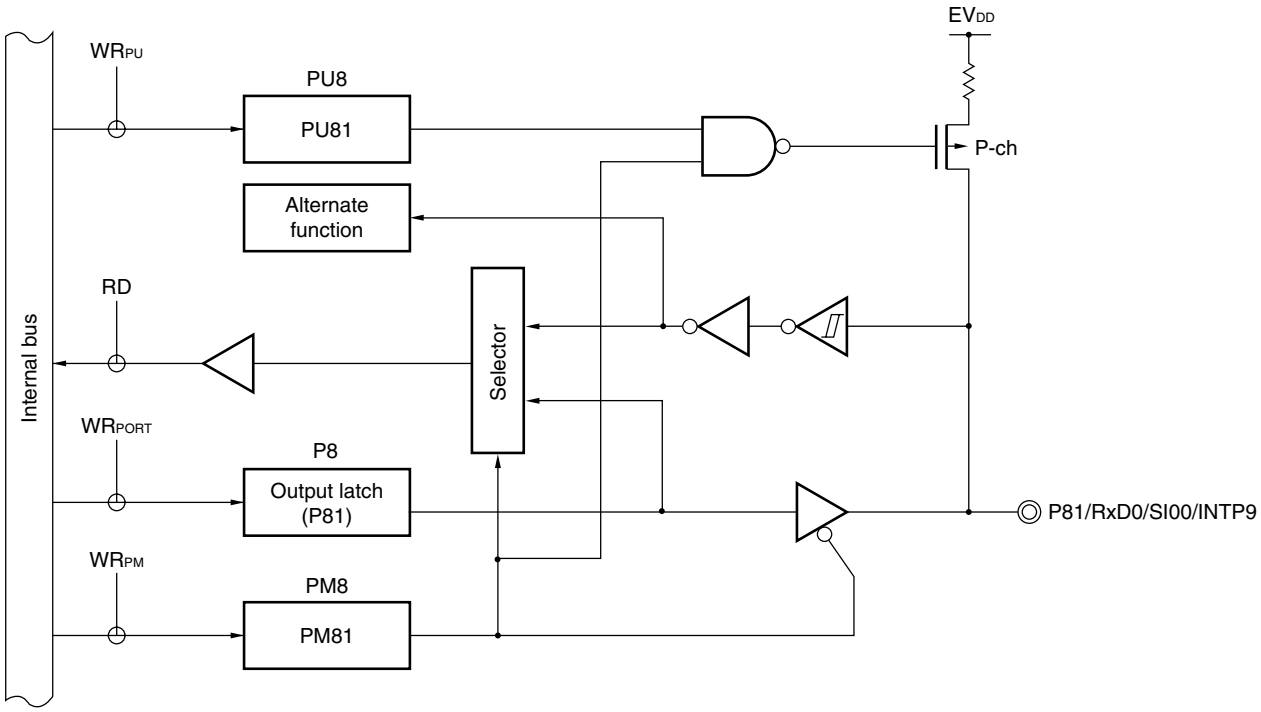
- 100-pin plastic LQFP (fine pitch) (14×14)



Cautions 1. Make AVSS the same potential as VSS.

2. Connect the REGC pin to Vss via a capacitor (0.47 to 1 μF).

Figure 4-21. Block Diagram of P81



P8: Port register 8
PU8: Pull-up resistor option register 8
PM8: Port mode register 8
RD: Read signal
WR××: Write signal

Cautions 2. To start X1 oscillation as set by MSTOP, check the oscillation stabilization time of the X1 clock by using the oscillation stabilization time counter status register (OSTC).

3. Do not stop the clock selected for the CPU peripheral hardware clock (f_{CLK}) with the OSC register.

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

Table 5-2. Condition Before Stopping Clock Oscillation and Flag Setting

Clock	Condition Before Stopping Clock (Invalidating External Clock Input)	Setting of CSC Register Flags
X1 clock External main system clock	CPU and peripheral hardware clocks operate with a clock other than the high-speed system clock. (• CLS = 0 and MCS = 0 • CLS = 1)	MSTOP = 1
Subsystem clock	CPU and peripheral hardware clocks operate with a clock other than the subsystem clock. (CLS = 0)	XTSTOP = 1
Internal high-speed oscillation clock	CPU and peripheral hardware clocks operate with a clock other than the internal high-speed oscillator clock and 20 MHz internal high-speed oscillation clock. (• CLS = 0 and MCS = 1 • CLS = 1)	HIOSTOP = 1

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

This is the register that indicates the count status of the X1 clock oscillation stabilization time counter.
The X1 clock oscillation stabilization time can be checked in the following case,

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

OSTC can be read by a 1-bit or 8-bit memory manipulation instruction.
When reset signal is generated, the STOP instruction and MSTOP (bit 7 of CSC register) = 1 clear OSTC to 00H.

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

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

Figure 5-4. Format of Oscillation Stabilization Time Counter Status Register (OSTC)

Address: FFFA2H After reset: 00H R

Symbol	7	6	5	4	3	2	1	0
OSTC	MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18

MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18	Oscillation stabilization time status		
									fx = 10 MHz	fx = 20 MHz
0	0	0	0	0	0	0	0	2^8/fx max.	25.6 μs max.	12.8 μs max.
1	0	0	0	0	0	0	0	2^8/fx min.	25.6 μs min.	12.8 μs min.
1	1	0	0	0	0	0	0	2^9/fx min.	51.2 μs min.	25.6 μs min.
1	1	1	0	0	0	0	0	2^{10}/fx min.	102.4 μs min.	51.2 μs min.
1	1	1	1	0	0	0	0	2^{11}/fx min.	204.8 μs min.	102.4 μs min.
1	1	1	1	1	0	0	0	2^{13}/fx min.	819.2 μs min.	409.6 μs min.
1	1	1	1	1	1	0	0	2^{15}/fx min.	3.27 ms min.	1.64 ms min.
1	1	1	1	1	1	1	0	2^{17}/fx min.	13.11 ms min.	6.55 ms min.
1	1	1	1	1	1	1	1	2^{18}/fx min.	26.21 ms min.	13.11 ms min.

Cautions 1. After the above time has elapsed, the bits are set to 1 in order from MOST8 and remain 1.

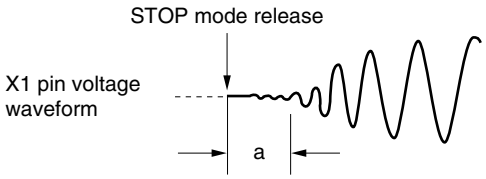
2. The oscillation stabilization time counter counts up to the oscillation stabilization time set by OSTS.

In the following cases, set the oscillation stabilization time of OSTS to the value greater than the count value which is to be checked by the OSTC register after the oscillation starts.

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

(Note, therefore, that only the status up to the oscillation stabilization time set by OSTS is set to OSTC after the STOP mode is released.)

3. The X1 clock oscillation stabilization wait time does not include the time until clock oscillation starts ("a" below).



Remark fx: X1 clock oscillation frequency

(1) Peripheral enable register 0 (PER0)

PER0 is used to enable or disable use of each peripheral hardware macro. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When the timer array unit 0 is used, be sure to set bit 0 (TAU0EN) of this register to 1.

When the timer array unit 1 is used, be sure to set bit 1 (TAU1EN) of this register to 1.

PER0 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

Caution When setting the timer array unit, be sure to set TAUmEN to 1 first. If TAUmEN = 0, writing to a control register of the timer array unit is ignored, and all read values are default values.

Figure 6-5. Format of Peripheral Enable Register 0 (PER0)

Address: F00F0H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PER0	RTCEN	DACEN	ADCEN	IICAEN[Note]	SAU1EN	SAU0EN	TAU1EN	TAU0EN

TAUmEN	Control of timer array unit m input clock
0	Stops supply of input clock. • SFR used by the timer array unit m cannot be written. • The timer array unit m is in the reset status.
1	Supplies input clock. • SFR used by the timer array unit m can be read/written.

Note 78K0R/LG3, 78K0R/LH3 only

(2) Timer clock select register m (TPSm)

TPSm 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 TPSm, and CKm0 is selected by bits 3 to 0.

Rewriting of TPSm during timer operation is possible only in the following cases.

Rewriting of PRSm00 to PRSm03 bits: Possible only when all the channels set to CKSmn = 0 are in the operation stopped state (TEmn = 0)

Rewriting of PRSm10 to PRSm13 bits: Possible only when all the channels set to CKSmn = 1 are in the operation stopped state (TEmn = 0)

TPSm can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of TPSm can be set with an 8-bit memory manipulation instruction with TPSmL.

Reset signal generation clears this register to 0000H.

Remark mn: Unit number + Channel number
m = 0, 1, mn = 00 to 07, 10 to 13

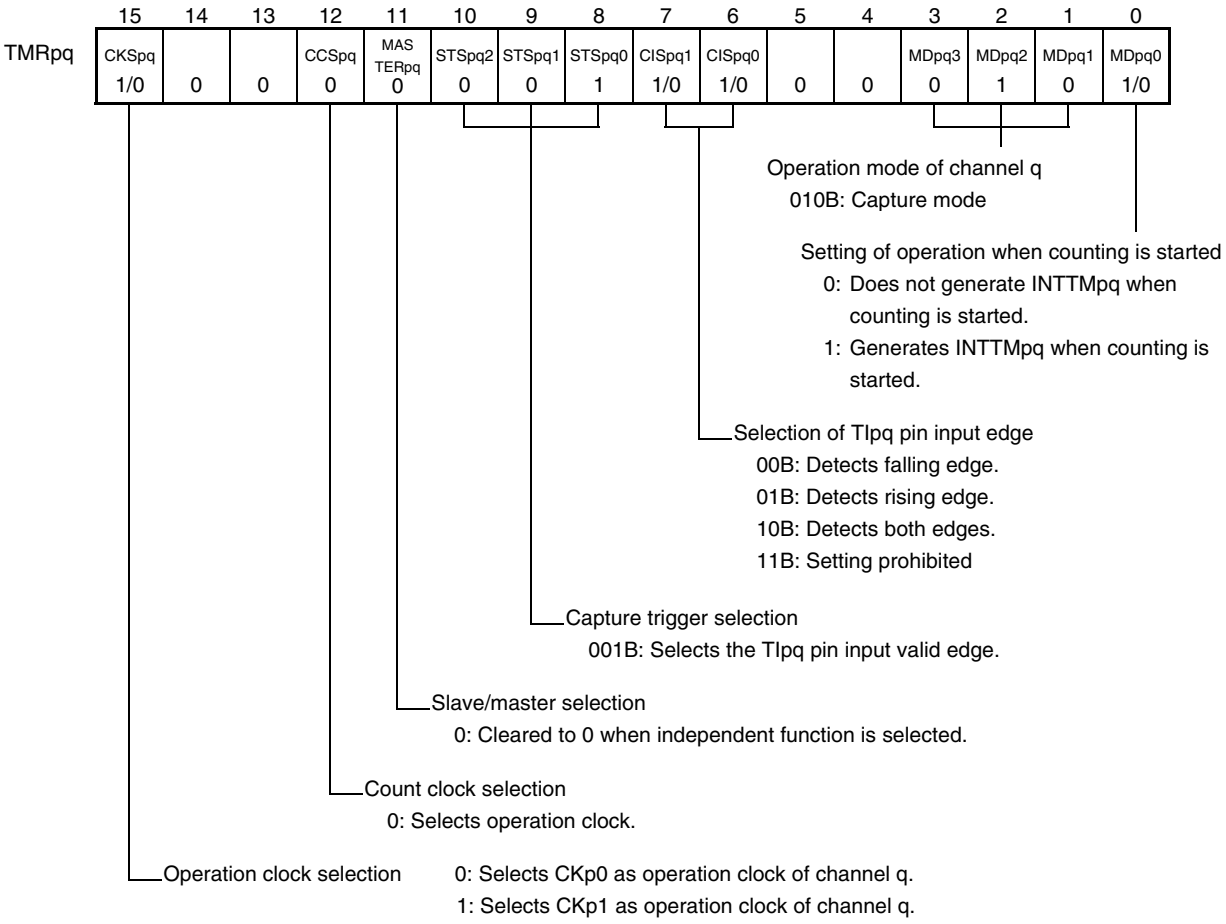
Figure 6-40. Operation Procedure of Interval Timer/Square Wave Output Function (2/2)

	Software Operation	Hardware Status
TAU stop	To hold the TOpq pin output level Clears TOpq bit to 0 after the value to be held is set to the port register. →	The TOpq pin output level is held by port function.
	When holding the TOpq pin output level is not necessary Switches the port mode register to input mode. →	The TOpq pin output level goes into Hi-Z output state.
	The TAU0EN or TAU1EN bits of the PER0 register is cleared to 0. →	Power-off status All circuits are initialized and SFR of each channel is also initialized. (The TOpq bit is cleared to 0 and the TOpq pin is set to port mode.)

Remark mn: Unit number + Channel number, pq: Unit number + Channel number (only for channels provided with timer I/O pins)
78K0R/LF3: mn = 00 to 07, 10 to 13, pq = 00 to 04, 07
78K0R/LG3: mn = 00 to 07, 10 to 13, pq = 00 to 07
78K0R/LH3: mn = 00 to 07, 10 to 13, pq = 00 to 07, 10 to 13

Figure 6-51. Example of Set Contents of Registers to Measure Input Pulse Interval

(a) Timer mode register pq (TMRpq)



(b) Timer output register p (TOp)

	Bit q	
TOp	TOpq	0: Outputs 0 from TOpq.
	0	

(c) Timer output enable register p (TOEp)

	Bit q	
TOEp	TOEpq	0: Stops TOpq output operation by counting operation.
	0	

(d) Timer output level register p (TOLp)

	Bit q	
TOLp	TOLpq	0: Cleared to 0 when TOMpq = 0 (toggle mode).
	0	

(e) Timer output mode register p (TOMp)

	Bit q	
TOMp	TOMpq	0: Sets toggle mode.
	0	

Remark pq: Unit number + Channel number (only for channels provided with timer I/O pins)
78K0R/LF3: p = 0, pq = 00 to 04, 07
78K0R/LG3: p = 0, pq = 00 to 07
78K0R/LH3: p = 0, 1, pq = 00 to 07, 10 to 13

14.4.2 Stopping the operation by channels

The stopping of the operation by channels is set using each of the following registers.

Figure 14-23. Each Register Setting When Stopping the Operation by Channels (1/2)

- **Serial Channel Enable Status Register m (SEm) … This register indicates whether data transmission/reception operation of each channel is enabled or stopped.**

Sem

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	Sem3 0/1	Sem2 0/1	Sem1 0/1	Sem0 0/1

0: Operation stops

- The Sem register is a read-only status register, whose operation is stopped by using the STm register.
  With a channel whose operation is stopped, the value of CKOmn of the Som register can be set by software.

- **Serial channel stop register m (STm) … This register is a trigger register that is used to enable stopping communication/count by each channel.**

STm

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	STm3 0/1	STm2 0/1	STm1 0/1	STm0 0/1

1: Clears Semn to 0 and stops the communication operation

* Because STmn is a trigger bit, it is cleared immediately when SEmn = 0.

(c) Serial output enable register m (SOEm) … This register is a register that is used to enable or stop output of the serial communication operation of each channel.

SOE0

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	0	SOE02 0/1	SOE01 0/1	SOE00 0/1

0: Stops output by serial communication operation

* For channel n, whose serial output is stopped, the SO0n value of the SO0 register can be set by software.

SOE1

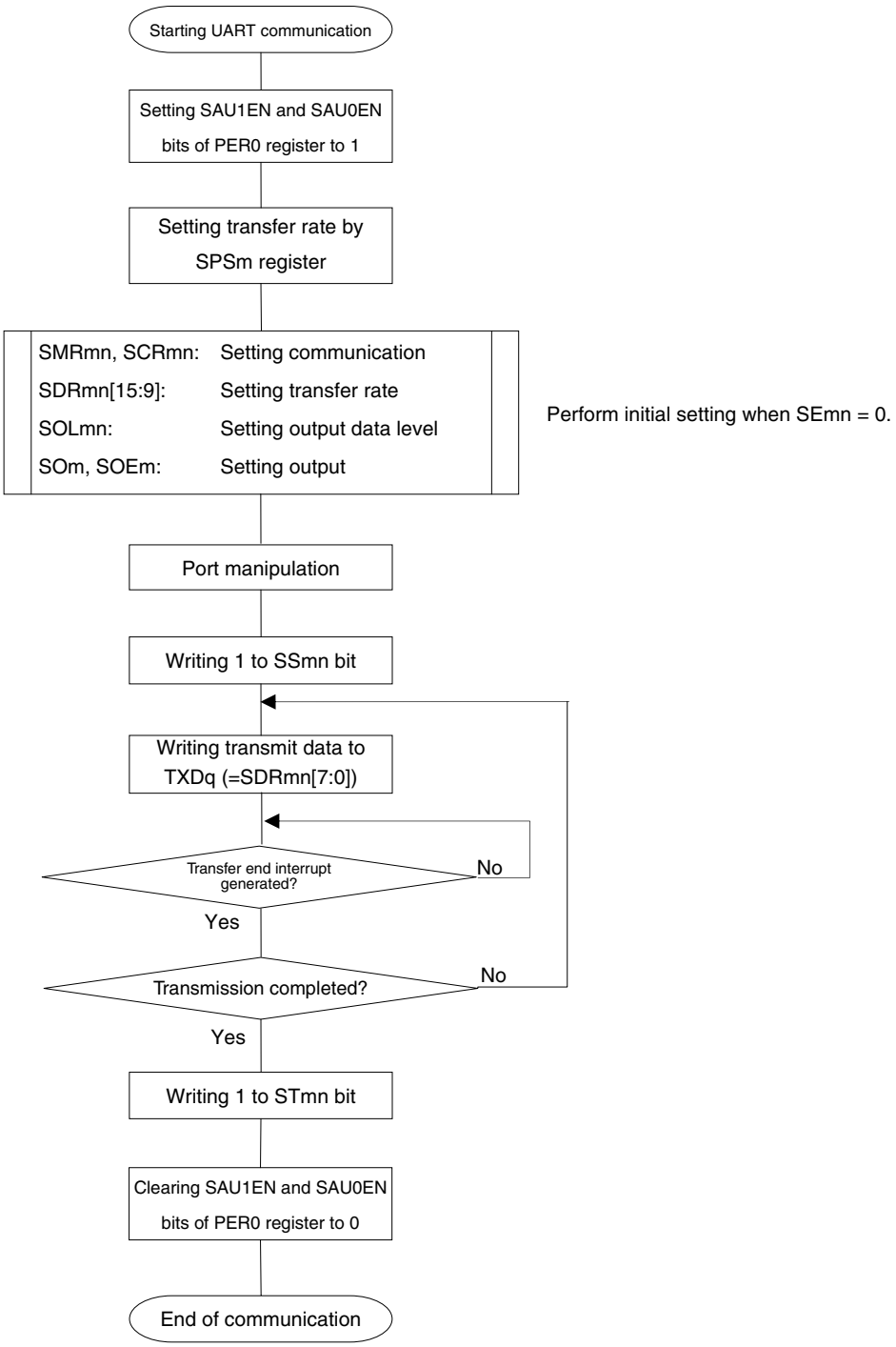
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	SOE12 0/1	0	SOE10 0/1

0: Stops output by serial communication operation

* For channel n, whose serial output is stopped, the SO1n value of the SO1 register can be set by software.

Remark m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)
□ : Setting disabled (fixed by hardware), 0/1: Set to 0 or 1 depending on the usage of the user

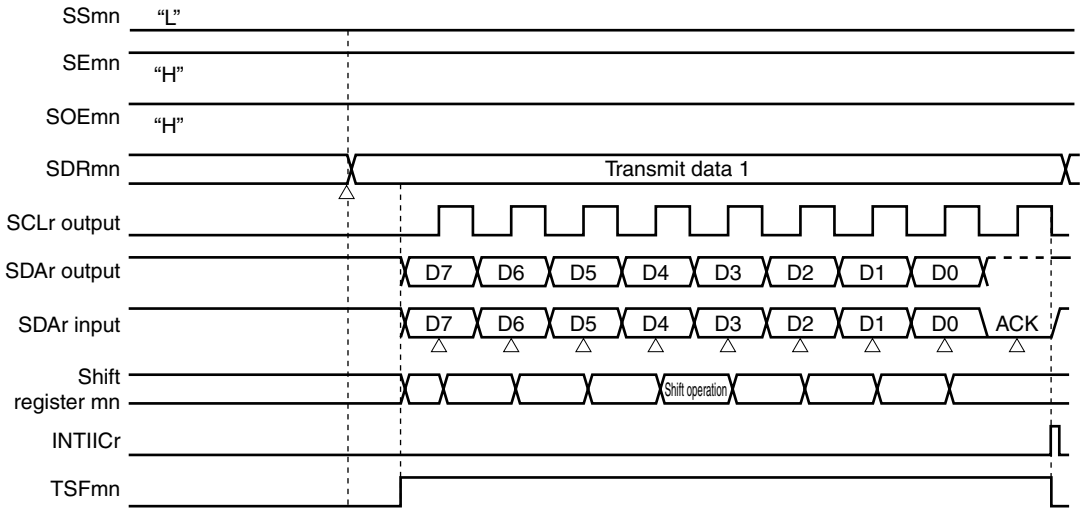
Figure 14-73. Flowchart of UART Transmission (in Single-Transmission Mode)



Caution After setting the SAUmEN to 1, be sure to set the SPSm register after 4 or more clocks have elapsed.

(2) Processing flow

Figure 14-93. Timing Chart of Data Transmission



Remark m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), r: IIC number (r = 10, 20)

Figure 14-94. Flowchart of Data Transmission

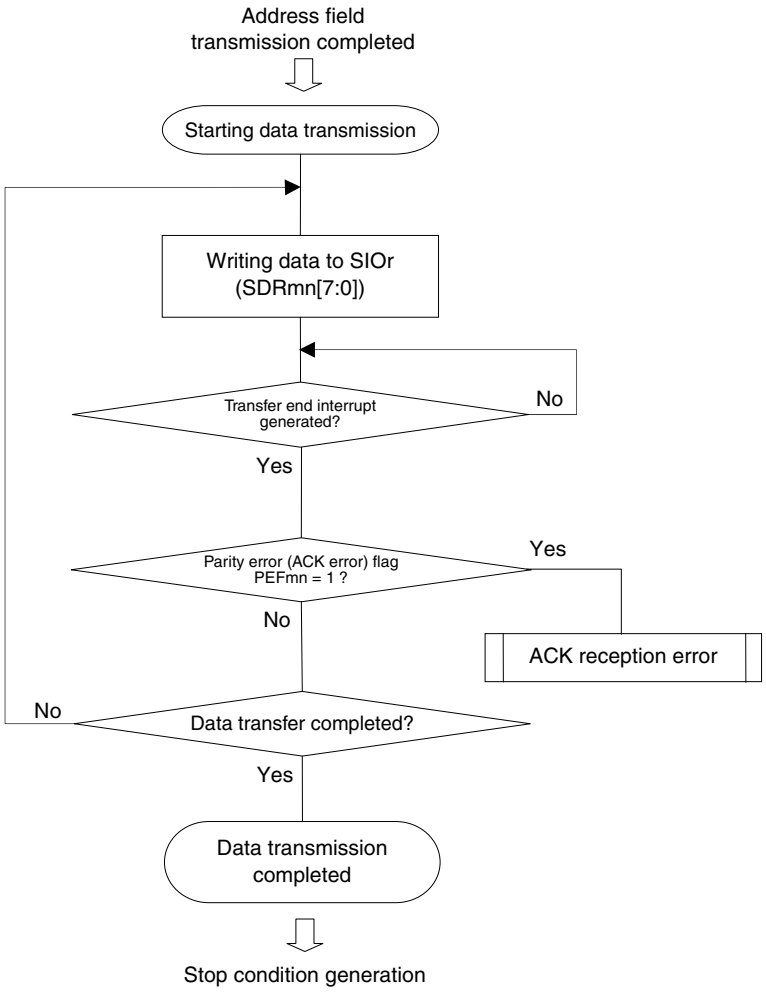


Table 14-9. Relationship between register settings and pins
(Channel 0 of unit 1: CSI20, UART2 transmission, IIC20)

SE10 Note1	MD102	MD101	SOE10	SO10	CKO10	TXE10	RXE10	PM10	P10	PM11 Note2	P11 Note2	PM12	P12	Operation mode	Pin Function		
															$\overline{SCK20}$/SCL20/P10	SI20/SDA20/RxD2/INTP6/P11 Note2	SO20/TxD2/TO02/P12
0	0	0	0	1	1	0	0	× Note3	× Note3	× Note3	× Note3	× Note3	× Note3	Operation stop mode	P10	INTP6/P11	TO02/P12
	0	1														RxD2/INTP6/P11	
	1	0														INTP6/P11	
1	0	0	0	1	1	0	1	1	×	1	×	× Note3	× Note3	Slave CSI20 reception	$\overline{SCK20}$ (input)	SI20	TO02/P12
			1	0/1 Note4	1	1	0	1	×	× Note3	× Note3	0	1	Slave CSI20 transmission	$\overline{SCK20}$ (input)	INTP6/P11	SO20
			1	0/1 Note4	1	1	1	1	×	1	×	0	1	Slave CSI20 transmission/reception	$\overline{SCK20}$ (input)	SI20	SO20
			0	1	0/1 Note4	0	1	0	1	1	×	× Note3	× Note3	Master CSI20 reception	$\overline{SCK20}$ (output)	SI20	TO02/P12
			1	0/1 Note4	0/1 Note4	1	0	0	1	× Note3	× Note3	0	1	Master CSI20 transmission	$\overline{SCK20}$ (output)	INTP6/P11	SO20
			1	0/1 Note4	0/1 Note4	1	1	0	1	1	×	0	1	Master CSI20 transmission/reception	$\overline{SCK20}$ (output)	SI20	SO20
	0	1	1	0/1 Note4	1	1	0	× Note3	× Note3	× Note3	× Note3	0	1	UART2 transmission Note5	P10	RxD2/INTP6/P11	TxD2
0	1	0	0	0/1 Note6	0/1 Note6	0	0	0	1	0	1	× Note3	× Note3	IIC20 start condition	SCL20	SDA20	TO02/P12
						1	0										
						0	1										
1			1	0/1 Note4	0/1 Note4	1	0	0	1	0	1	× Note3	× Note3	IIC20 address field transmission	SCL20	SDA20	TO02/P12
			1	0/1 Note4	0/1 Note4	1	0	0	1	0	1	× Note3	× Note3	IIC20 data transmission	SCL20	SDA20	TO02/P12
			1	0/1 Note4	0/1 Note4	0	1	0	1	0	1	× Note3	× Note3	IIC20 data reception	SCL20	SDA20	TO02/P12
0			0	0/1 Note7	0/1 Note7	0	0	0	1	0	1	× Note3	× Note3	IIC20 stop condition	SCL20	SDA20	TO02/P12
						1	0										
						0	1										

Notes 1. The SE1 register is a read-only status register which is set using the SS1 and ST1 registers.

2. When channel 1 of unit 1 is set to UART2 reception, this pin becomes an RxD2 function pin (refer to **Table 14-10**). In this case, operation stop mode or UART2 transmission must be selected for channel 0 of unit 1.

3. This pin can be set as a port function pin.

4. This is 0 or 1, depending on the communication operation. For details, refer to **14.3 (12) Serial output register m (SOm)**.

5. When using UART2 transmission and reception in a pair, set channel 1 of unit 1 to UART2 reception (refer to **Table 14-10**).

6. Set the CKO10 bit to 1 before a start condition is generated. Clear the SO10 bit from 1 to 0 when the start condition is generated.

7. Set the CKO10 bit to 1 before a stop condition is generated. Clear the SO10 bit from 0 to 1 when the stop condition is generated.

Remark X: Don't care

Table 14-11. Relationship between register settings and pins (Channel 2 of unit 1: UART3 transmission)

SE12 Note1	MD122	MD121	SOE12	SO12	TXE12	RXE12	PM51	P51	Operation mode	Pin Function
										TxD3/SEG52/P51
0	0	1	0	1	0	0	× Note2	× Note2	Operation stop mode	SEG52/P51
1	0	1	1	0/1 Note3	1	0	0	1	UART3 transmission Note4	TxD3

Notes 1. The SE1 register is a read-only status register which is set using the SS1 and ST1 registers.

2. This pin can be set as a port function pin.

3. This is 0 or 1, depending on the communication operation. For details, refer to **14.3 (12) Serial output register m (SOm)**.

4. When using UART3 transmission and reception in a pair, set channel 3 of unit 1 to UART3 reception (refer to **Table 14-12**).

Remark X: Don't care

Table 14-12. Relationship between register settings and pins (Channel 3 of unit 1: UART3 reception)

SE13 Note1	MD132	MD131	TXE13	RXE13	PM50	P50	Operation mode	Pin Function
								RxD3/SEG53/P50
0	0	1	0	0	× Note2	× Note2	Operation stop mode	SEG53/P50
1	0	1	0	1	1	×	UART3 reception Note3, 4	RxD3

Notes 1. The SE1 register is a read-only status register which is set using the SS1 and ST1 registers.

2. This pin can be set as a port function pin.

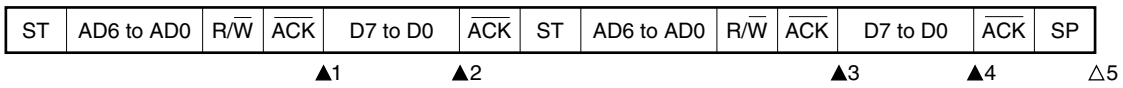
3. When using UART3 transmission and reception in a pair, set channel 2 of unit 1 to UART3 transmission (refer to **Table 14-11**).

4. The SMR12 register of channel 2 of unit 1 must also be set during UART3 reception. For details, refer to **14.5.2 (1) Register setting**.

Remark X: Don't care

(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop

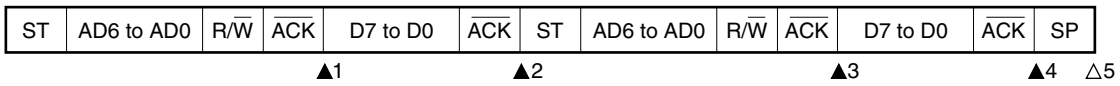
(i) When WTIM = 0 (after restart, matches with SVA)



▲1: IICS = 0001×110B
▲2: IICS = 0001×000B
▲3: IICS = 0001×110B
▲4: IICS = 0001×000B
△5: IICS = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE = 1
×: Don't care

(ii) When WTIM = 1 (after restart, matches with SVA)

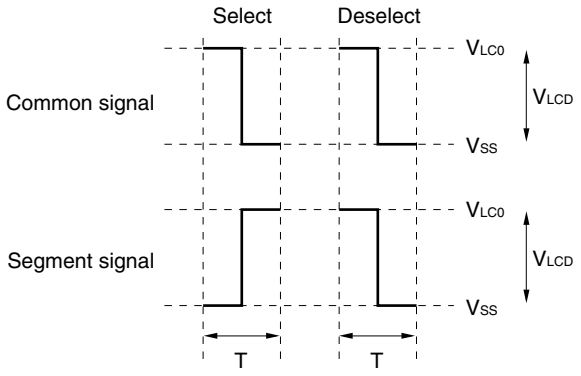


▲1: IICS = 0001×110B
▲2: IICS = 0001××00B
▲3: IICS = 0001×110B
▲4: IICS = 0001××00B
△5: IICS = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE = 1
×: Don't care

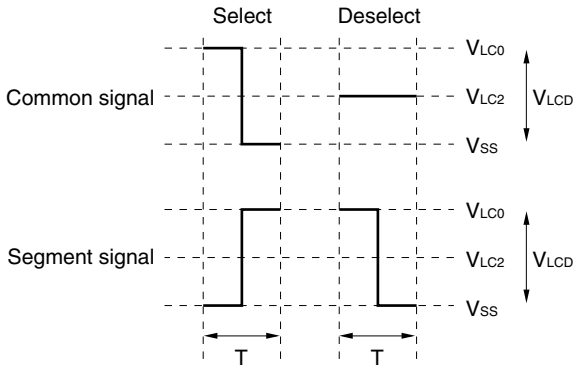
Figure 16-14. Voltages and Phases of Common and Segment Signals (1/2)

(a) Static display mode



T: One LCD clock period

(b) 1/2 bias method



T: One LCD clock period

16.7 Display Modes

16.7.1 Static display example

Figure 16-16 shows how the three-digit LCD panel having the display pattern shown in Figure 16-15 is connected to the segment signals (SEG0 to SEG23) and the common signal (COM0). This example displays data "12.3" in the LCD panel. The contents of the display data memory (F0400H to F0417H) correspond to this display.

The following description focuses on numeral "2." (2.) displayed in the second digit. To display "2." in the LCD panel, it is necessary to apply the select or deselect voltage to the SEG8 to SEG15 pins according to Table 16-5 at the timing of the common signal COM0; see Figure 16-15 for the relationship between the segment signals and LCD segments.

Table 16-5. Select and Deselect Voltages (COM0)

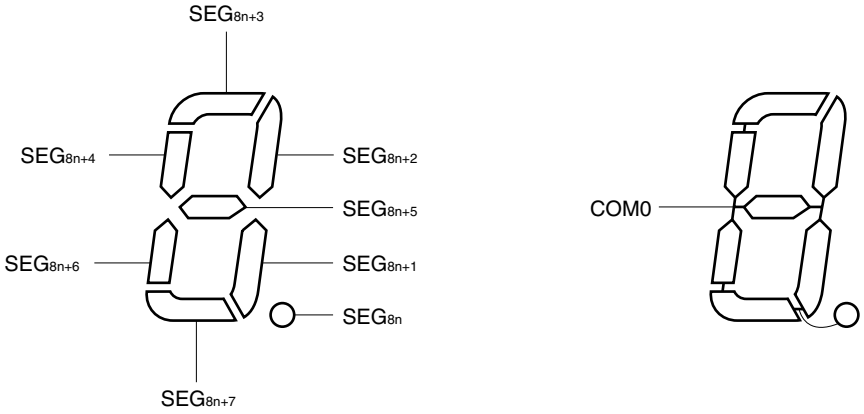
Segment / Common	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15
COM0	Select	Deselect	Select	Select	Deselect	Select	Select	Select

According to Table 16-5, it is determined that the bit-0 pattern of the display data memory locations (F0408H to F040FH) must be 10110111.

Figure 16-17 shows the LCD drive waveforms of SEG11 and SEG12, and COM0. When the select voltage is applied to SEG11 at the timing of COM0, an alternate rectangle waveform, $+V_{LCD}/-V_{LCD}$, is generated to turn on the corresponding LCD segment.

COM1 to COM3 are supplied with the same waveform as for COM0. So, COM0 to COM3 may be connected together to increase the driving capacity.

Figure 16-15. Static LCD Display Pattern and Electrode Connections



Remark 78K0R/LF3: n = 0 to 3
78K0R/LG3: n = 0 to 4
78K0R/LH3: n = 0 to 5

18.5.4 Consecutive capturing of A/D conversion results

A flowchart of an example of setting for consecutively capturing A/D conversion results is shown below.

- Consecutive capturing of A/D conversion results.
- DMA channel 1 is used for DMA transfer.
- DMA start source: INTAD
- Interrupt of A/D is specified by IFC13 to IFC10 (bits 3 to 0 of the DMC1 register) = 1100B.
- Transfers FFF1EH and FFF1FH (2 bytes) of the 12-bit A/D conversion result register to 512 bytes of FFCE0H to FFEDFH of RAM.

Table 19-2. Flags Corresponding to Interrupt Request Sources (2/2)

LF3	LG3	LH3	Interrupt Source	Interrupt Request Flag		Interrupt Mask Flag		Priority Specification Flag	
					Register		Register		Register
√	√	√	INTST1[Note 1]	STIF1[Note 1]	IF1L	STMK1[Note 1]	MK1L	STPR01, STPR11[Note 1]	PR01L, PR11L
√	√	√	INTCSI10[Note 1]	CSIIF10[Note 1]		CSIMK10[Note 1]		CSIPR010, CSIPR110[Note 1]	
√	√	√	INTIIC10[Note 1]	IICIF10[Note 1]		IICMK10[Note 1]		IICPR010, IICPR110[Note 1]	
√	√	√	INTSR1	SRIF1		SRMK1		SRPR01, SRPR11	
√	√	√	INTSRE1	SREIF1		SREMK1		SREPR01, SREPR11	
–	√	√	INTIICA	IICAIF		IICAMK		IICAPR0, IICAPR1	
√	√	√	INTTM00	TMIF00		TMMK00		TMPR000, TMPR100	
√	√	√	INTTM01	TMIF01		TMMK01		TMPR001, TMPR101	
√	√	√	INTTM02	TMIF02		TMMK02		TMPR002, TMPR102	
√	√	√	INTTM03	TMIF03		TMMK03		TMPR003, TMPR103	
√	√	√	INTAD	ADIF	IF1H	ADMK	MK1H	ADPR0, ADPR1	PR01H, PR11H
√	√	√	INTRTC	RTCIF		RTCMK		RTCPR0, RTCPR1	
√	√	√	INTRTCI	RTCIIF		RTCIMK		RTCIPR0, RTCIPR1	
–	–	√	INTKR	KRIF		KRMK		KRPR0, KRPR1	
√	√	√	INTST2[Note 2]	STIF2[Note 2]		STMK2[Note 2]		STPR02, STPR12[Note 2]	
√	√	√	INTCSI20[Note 2]	CSIIF20[Note 2]		CSIMK20[Note 2]		CSIPR020, CSIPR120[Note 2]	
√	√	√	INTIIC20[Note 2]	IICIF20[Note 2]		IICMK20[Note 2]		IICPR020, IICPR120[Note 2]	
√	√	√	INTSR2	SRIF2		SRMK2		SRPR02, SRPR12	
√	√	√	INTSRE2	SREIF2		SREMK2		SREPR02, SREPR12	
√	√	√	INTTM04	TMIF04		TMMK04		TMPR004, TMPR104	
√	√	√	INTTM05	TMIF05	IF2L	TMMK05	MK2L	TMPR005, TMPR105	PR02L, PR12L
√	√	√	INTTM06	TMIF06		TMMK06		TMPR006, TMPR106	
√	√	√	INTTM07	TMIF07		TMMK07		TMPR007, TMPR107	
√	√	√	INTP6	PIF6		PMK6		PPR06, PPR16	
√	√	√	INTP7	PIF7		PMK7		PPR07, PPR17	
–	√	√	INTP8	PIF8		PMK8		PPR08, PPR18	
–	√	√	INTP9	PIF9		PMK9		PPR09, PPR19	
–	√	√	INTP10	PIF10		PMK10		PPR010, PPR110	
–	√	√	INTP11	PIF11	IF2H	PMK11	MK2H	PPR011, PPR111	PR02H, PR12H
√	√	√	INTTM10	TMIF10		TMMK10		TMPR010, TMPR110	
√	√	√	INTTM11	TMIF11		TMMK11		TMPR011, TMPR111	
√	√	√	INTTM12	TMIF12		TMMK12		TMPR012, TMPR112	
√	√	√	INTTM13	TMIF13		TMMK13		TMPR013, TMPR113	
√	√	√	INTMD	MDIF		MDMK		MDPR0, MDPR1	

Notes 1. Do not use UART1, CSI10, and IIC10 at the same time because they share flags for the interrupt request sources. If one of the interrupt sources INTST1, INTCSI10, and INTIIC10 is generated, bit 0 of IF1L is set to 1. Bit 0 of MK1L, PR01L, and PR11L supports these three interrupt sources.

2. Do not use UART2, CSI20, and IIC20 at the same time because they share flags for the interrupt request sources. If one of the interrupt sources INTST2, INTCSI20, and INTIIC20 is generated, bit 4 of IF1H is set to 1. Bit 4 of MK1H, PR01H, and PR11H supports these three interrupt sources.

27.4.7 Power supply

To use the supply voltage output of the flash memory programmer, connect the V_{DD} pin to V_{DD} of the flash memory programmer, and the V_{SS} pin to GND of the flash memory programmer.

To use the on-board supply voltage, connect in compliance with the normal operation mode.

However, when using the on-board supply voltage, be sure to connect the V_{DD} and V_{SS} pins to V_{DD} and GND of the flash memory programmer to use the power monitor function with the flash memory programmer.

Supply the same other power supplies (EV_{DD}, EV_{SS}, AV_{DD0}, AV_{DD1}, and AV_{SS}) as those in the normal operation mode.

27.5 Registers Controlling Flash Memory

(1) Background event control register (BECTL)

Even if the FLMD0 pin is not controlled externally, it can be controlled by software with the BECTL register to set the self-programming mode.

However, depending on the processing of the FLMD0 pin, it may not be possible to set the self-programming mode by software. When using BECTL, leaving the FLMD0 pin open is recommended. When pulling it down externally, use a resistor with a resistance of 100 kΩ or more. In addition, in the normal operation mode, use BECTL with the pull down selection. In the self-programming mode, the setting is switched to pull up in the self- programming library.

The BECTL register is set by a 1-bit or 8-bit memory manipulation instruction.

Reset input sets this register to 00H.

Figure 27-6. Format of Background Event Control Register (BECTL)

Address: FFFBEH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
BECTL	FLMDPUP	0	0	0	0	0	0	0

FLMDPUP	Software control of FLMD0 pin
0	Selects pull-down
1	Selects pull-up

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

(T_A = –40 to +85°C, 1.8 V ≤ V_{DD} = EV_{DD} ≤ 5.5 V, V_{SS} = EV_{SS} = AVss = 0 V)

(c) During communication at same potential (CSI mode) (slave mode, $\overline{SCKp}$... external clock input)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{SCKp}$ cycle time	t_{KCY2}	4.0 V ≤ V_{DD} ≤ 5.5 V		6/f_{MCK}			ns
		2.7 V ≤ V_{DD} < 4.0 V	16 MHz < f_{MCK}	8/f_{MCK}			ns
			f_{MCK} ≤ 16 MHz	6/f_{MCK}			ns
		1.8 V ≤ V_{DD} < 2.7 V	16 MHz < f_{MCK}	8/f_{MCK}			ns
			f_{MCK} ≤ 16 MHz	6/f_{MCK}			ns
$\overline{SCKp}$ high-/low-level width	t_{KH2}, t_{KL2}			t_{KCY2}/2			ns
SIp setup time (to $\overline{SCKp}$↑) Note 1	t_{SIK2}			80			ns
SIp hold time (from $\overline{SCKp}$↑) Note 2	t_{KSI2}			1/f_{MCK}+50			ns
Delay time from $\overline{SCKp}$↓ to SOp output Note 3	t_{KSO2}	C = 30 pF Note 4	4.0 V ≤ V_{DD} = EV_{DD} ≤ 5.5 V			2/f_{MCK}+45	ns
			2.7 V ≤ V_{DD} = EV_{DD} < 4.0 V			2/f_{MCK}+57	ns
			1.8 V ≤ V_{DD} = EV_{DD} < 2.7 V			2/f_{MCK}+125	ns

Notes 1. When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The SIp setup time becomes "to $\overline{SCKp}$↓" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

2. When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The SIp hold time becomes "from $\overline{SCKp}$↓" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

3. When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The delay time to SOp output becomes "from $\overline{SCKp}$↑" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

4. C is the load capacitance of the $\overline{SCKp}$ and SOp output lines.

Caution Select the normal input buffer for SIp and $\overline{SCKp}$ and the normal output mode for SOp by using the PIMg and POMx registers.

Remarks 1. p: CSI number (p = 00, 01, 10, 20), g: PIM number (g = 1, 7), x: POM number (x = 1, 7, 8)

2. f_{MCK}: Serial array unit operation clock frequency
(Operation clock to be set by the CKSmn bit of the SMRmn register. m: Unit number (m = 0, 1), n: Channel number (n = 0 to 2))

(2) Serial interface: Serial array unit (8/18)

(T_A = –40 to +85°C, 2.7 V ≤ V_{DD} = EV_{DD} ≤ 5.5 V, V_{SS} = EV_{SS} = AVss = 0 V)

(e) Communication at different potential (2.5 V, 3 V) (UART mode) (dedicated baud rate generator output) (2/2)

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Transfer rate		transmission	4.0 V ≤ V_{DD} = EV_{DD} ≤ 5.5 V, 2.7 V ≤ V_b ≤ 4.0 V				**Note 1**	bps
				f_{CLK} = 16.8 MHz, f_{MCK} = f_{CLK}, C_b = 50 pF, R_b = 1.4 kΩ, V_b = 2.7 V			2.8 Note 2	Mbps
			2.7 V ≤ V_{DD} = EV_{DD} < 4.0 V, 2.3 V ≤ V_b < 2.7 V				**Note 3**	bps
				f_{CLK} = 19.2 MHz, f_{MCK} = f_{CLK}, C_b = 50 pF, R_b = 2.7 kΩ, V_b = 2.3 V			1.2 Note 4	Mbps

Notes 1. The smaller maximum transfer rate derived by using f_{MCK}/6 or the following expression is the valid maximum transfer rate.

Expression for calculating the transfer rate when 4.0 V ≤ V_{DD} = EV_{DD} ≤ 5.5 V and 2.7 V ≤ V_b ≤ 4.0 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1-\frac{2.2}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1-\frac{2.2}{V_b})\}}{(\frac{1}{\text{Transfer rate}}) \times \text{Number of transferred bits}} \times 100 \text{ [\%]}$$

* This value is the theoretical value of the relative difference between the transmission and reception sides.

2. This value as an example is calculated when the conditions described in the “Conditions” column are met. Refer to Note 1 above to calculate the maximum transfer rate under conditions of the customer.

3. The smaller maximum transfer rate derived by using f_{MCK}/6 or the following expression is the valid maximum transfer rate.

Expression for calculating the transfer rate when 2.7 V ≤ V_{DD} = EV_{DD} < 4.0 V and 2.3 V ≤ V_b < 2.7 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1-\frac{2.0}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1-\frac{2.0}{V_b})\}}{(\frac{1}{\text{Transfer rate}}) \times \text{Number of transferred bits}} \times 100 \text{ [\%]}$$

* This value is the theoretical value of the relative difference between the transmission and reception sides.

4. This value as an example is calculated when the conditions described in the “Conditions” column are met. Refer to Note 3 above to calculate the maximum transfer rate under conditions of the customer.

Caution Select the TTL input buffer for RxDq and the N-ch open drain output (V_{DD} tolerance) mode for TxDq by using the PIMg and POMx registers.

(Remarks are given on the next page.)