Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	RL78
Core Size	16-Bit
Speed	24MHz
Connectivity	CSI, I²C, IrDA, LINbus, UART/USART
Peripherals	LCD, LVD, POR, PWM, WDT
Number of I/O	52
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	6K x 8
Voltage - Supply (Vcc/Vdd)	1.7V ~ 5.5V
Data Converters	A/D 4x10b, 3x24b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	80-LQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/r5f10nmedfb-50

4.4 Port Function Operations

Port operations differ depending on whether the input or output mode is set, as shown below.

4.4.1 Writing to I/O port

(1) Output mode

A value is written to the output latch by a transfer instruction, and the output latch contents are output from the pin.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

(2) Input mode

A value is written to the output latch by a transfer instruction, but since the output buffer is off, the pin status does not change. Therefore, byte data can be written to the ports used for both input and output.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

4.4.2 Reading from I/O port

(1) Output mode

The output latch contents are read by a transfer instruction. The output latch contents do not change.

(2) Input mode

The pin status is read by a transfer instruction. The output latch contents do not change.

4.4.3 Operations on I/O port

(1) Output mode

An operation is performed on the output latch contents, and the result is written to the output latch. The output latch contents are output from the pins.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

(2) Input mode

The pin level is read and an operation is performed on its contents. The result of the operation is written to the output latch, but since the output buffer is off, the pin status does not change. Therefore, byte data can be written to the ports used for both input and output.
The data of the output latch is cleared when a reset signal is generated.

Table 4-6. Setting Examples of Registers and Output Latches When Using Alternate Function (8/10)

Pin Name	Used Function		PIOR0×	POM××	PM××	P××	PFSEG×× (ISCVL3, ISCCAP)[Note]	Alternate Function Output		64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU			
P76	P76	Input	_	_	1	×	0	_	_			
		Output	×	_	0	0/1	0	_	_			
	KR6	Input	×	_	1	×	0	_	_	–	√	√
	SEG22	Output	×	_	0	0	1	_	_			
	(INTP6)	Input	PIOR04 = 1	_	1	×	0	_	_			
P77	P77	Input	–	_	1	×	0	_	_			
		Output	×	_	0	0/1	0	_	_			
	KR7	Input	×	_	1	×	0	_	_	–	√	√
	SEG23	Output	×	_	0	0	1	_	_			
	(INTP7)	Input	PIOR04 = 1	_	1	×	0	_	_			
P80	P80	Input	–	×	1	×	0	_	_			
		Output	×	0	0	0/1	0	(SCK10/SCL10) = 1	_			
		N-ch open drain output	×	1	0	0/1	0					
	SEG12	Output	×	×	0	0	1	_	_	–	√	√
	(SCL10)	Output	PIOR02 = 1	0/1	0	1	0	_	_			
	(SCK10)	Input	PIOR02 = 1	×	1	×	0	_	_			
		Output	PIOR02 = 1	0/1	0	1	0	_	_			
P81	P81	Input	–	×	1	×	0	_	_			
		Output	×	0	0	0/1	0	(SDA10) = 1	_			
		N-ch open drain output	×	1	0	0/1	0					
	SEG13	Output	×	×	0	0	1	_	_	–	√	√
	(RxD1)	Input	PIOR02 = 1	×	1	×	0	_	_			
	(SDA10)	I/O	PIOR02 = 1	1	0	1	0	_	_			
	(SI10)	Input	PIOR02 = 1	×	1	×	0	_	_			
P82	P82	Input	–	×	1	×	0	_	_			
		Output	×	0	0	0/1	0	(TxD1/SO10) = 1	_			
		N-ch open drain output	×	1	0	0/1	0					
	SEG14	Output	×	×	0	0	1	_	_	–	√	√
	(TxD1)	Output	PIOR02 = 1	0/1	0	1	0	_	_			
	(SO10)	Output	PIOR02 = 1	0/1	0	1	0	_	_			
P83	P83	Input	_	_	1	×	0	_	_			
		Output	_	_	0	0/1	0	_	_	–	√	√
	SEG15	Output	_	_	0	0	1	_	_			
P84	P84	Input	_	_	1	×	0	_	_			
		Output	_	_	0	0/1	0	SDA30 = 1	_			
		N-ch open drain output	_	1	0	0/1	0					
	SEG40	Output	_	_	0	0	1	_	_	–	–	√
	SI30	Input	_	×	1	×	0	_	_			
	RxD3	Input	_	×	1	×	0	_	_			
	SDA30	I/O	_	1	0	1	0	_	_			
P85	P85	Input	_	_	1	×	0	_	_			
		Output	_	_	0	0/1	0	SO30/TxD3 = 1	_			
		N-ch open drain output	_	1	0	0/1	0					
	SEG41	Output	_	_	0	0	1	_	_	–	–	√
	SO30	Output	_	0/1	0	1	0	_	_			
	TxD3	Output	_	0/1	0	1	0	_	_			

Note ISCVL3 and ISCCAP are registers that correspond to P125, and P126 and P127, respectively.

6.3.10 Subsystem clock supply option control register (OSMC)

This register is used to reduce power consumption by stopping unnecessary clock functions.

If the RTCLPC bit is set to 1, power consumption can be reduced, because clock supply to the peripheral <R> functions, except the independent power supply RTC, 12-bit interval timer, 8-bit interval timer, clock output/buzzer output controller, LCD controller/driver, oscillation stop detection circuit, and frequency measurement circuit is stopped in STOP mode or HALT mode while sub clock (f_{SX}) is selected as CPU clock.

In addition, the OSMC register can be used to select the operating clock of the 12-bit interval timer, 8-bit interval timer, clock output/buzzer output controller, LCD controller/driver, and frequency measurement circuit.

<R> The OSMC register can be set by an 8-bit memory manipulation instruction or a 1-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

6.6 Controlling Clock

6.6.1 Example of setting high-speed on-chip oscillator

After a reset release, the CPU/peripheral hardware clock (f_{CLK}) always starts operating with the high-speed on-chip oscillator clock. The frequency of the high-speed on-chip oscillator can be selected from 24, 16, 12, 8, 6, 4, 3, 2, 1.5, and 1 MHz by using FRQSEL0 to FRQSEL3 of the option byte (000C2H). In addition, Oscillation can be changed by the high-speed on-chip oscillator frequency select register (HOCODIV).

[Option byte setting]

Address: 000C2H

Option byte (000C2H)	7	6	5	4	3	2	1	0
	CMODE1 0/1	CMODE0 0/1	1	1	FRQSEL3 0/1	FRQSEL2 0/1	FRQSEL1 0/1	FRQSEL0 0/1

CMODE1	CMODE0	Setting of flash operation mode	
0	0	LV (low-voltage main) mode	V_{DD} = 1.7 V to 5.5 V @ 1 MHz to 4 MHz
1	0	LS (low-speed main) mode	V_{DD} = 1.9 V to 5.5 V @ 1 MHz to 8 MHz
1	1	HS (high-speed main) mode	V_{DD} = 2.1 V to 5.5 V @ 1 MHz to 6 MHz V_{DD} = 2.4 V to 5.5 V @ 1 MHz to 12 MHz V_{DD} = 2.5 V to 5.5 V @ 1 MHz to 16 MHz V_{DD} = 2.7 V to 5.5 V @ 1 MHz to 24 MHz V_{DD} = 2.8 V to 5.5 V @ 1 MHz to 32 MHz
Other than above		Setting prohibited	

FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	Frequency of the high-speed on-chip oscillator
				f_{IH}
0	0	0	0	24 MHz
1	0	0	1	16 MHz
0	0	0	1	12 MHz
1	0	1	0	8 MHz
0	0	1	0	6 MHz
1	0	1	1	4 MHz
0	0	1	1	3 MHz
1	1	0	0	2 MHz
0	1	0	0	1.5 MHz
1	1	0	1	1 MHz
Other than above				Setting prohibited

6.7 Resonator and Oscillator Constants

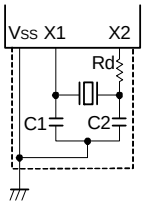
For the resonators for which operation has been verified and their oscillator constants, see the target product page on the Renesas Web site.

Caution 1. The constants for these oscillator circuits are reference values based on specific environments set up for evaluation by the manufacturers. For actual applications, request evaluation by the manufacturer of the oscillator circuit mounted on a board. Furthermore, if you are switching from a different product to this microcontroller, and whenever you change the board, again request evaluation by the manufacturer of the oscillator circuit mounted on the new board.

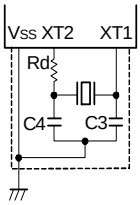
Caution 2. The oscillation voltage and oscillation frequency only indicate the oscillator characteristic. Use the RL78 microcontroller so that the internal operation conditions are within the specifications of the DC and AC characteristics.

Figure 6 - 25 Example of External Circuit

(a) X1 oscillation



(b) XT1 oscillation



8.3.6 Timer channel enable status register m (TEm)

The TEm register is used to enable or stop the timer operation of each channel.

Each bit of the TEm register corresponds to each bit of the timer channel start register m (TSm) and the timer channel stop register m (TTm). When a bit of the TSm register is set to 1, the corresponding bit of this register is set to 1. When a bit of the TTm register is set to 1, the corresponding bit of this register is cleared to 0.

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

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

Reset signal generation clears this register to 0000H.

Figure 8-17. Format of Timer Channel Enable Status register m (TEm)

Address: F01B0H, F01B1H After reset: 0000H R

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEm	0	0	0	0	TEHm3	0	TEHm1	0	TEm7	TEm6	TEm5	TEm4	TEm3	TEm2	TEm1	TEm0

TEH03	Indication of whether operation of the higher 8-bit timer is enabled or stopped when channel 3 is in the 8-bit timer mode
0	Operation is stopped.
1	Operation is enabled.

TEH01	Indication of whether operation of the higher 8-bit timer is enabled or stopped when channel 1 is in the 8-bit timer mode
0	Operation is stopped.
1	Operation is enabled.

TEmn	Indication of operation enable/stop status of channel n
0	Operation is stopped.
1	Operation is enabled.
This bit displays whether operation of the lower 8-bit timer for TEm1 and TEm3 is enabled or stopped when channel 1 or 3 is in the 8-bit timer mode.	

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

Figure 8-52. Operation Procedure When External Event Counter Function Is Used

	Software Operation	Hardware Status
TAU default setting		Power-off status (Clock supply is stopped and writing to each register is disabled.)
	Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1. →	Power-on status. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.	
Channel default setting	Sets the corresponding bit of the noise filter enable register 1 (NFEN1) to 0 (off) or 1 (on). Sets timer mode register mn (TMRmn) (determines operation mode of channel). Sets number of counts to timer data register mn (TDRmn). Clears the TOEmn bit of timer output enable register m (TOEm) to 0.	Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	Sets the TSmn bit to 1. → The TSmn bit automatically returns to 0 because it is a trigger bit.	TEmn = 1, and count operation starts. Value of the TDRmn register is loaded to timer count register mn (TCRmn) and detection of the TImn pin input edge is awaited.
During operation	Set value of the TDRmn register can be changed. The TCRmn register can always be read. The TSRmn register is not used. Set values of the TMRmn register, TOMmn, TOLmn, TOmn, and TOEmn bits cannot be changed.	Counter (TCRmn) counts down each time input edge of the TImn pin has been detected. When count value reaches 0000H, the value of the TDRmn register is loaded to the TCRmn register again, and the count operation is continued. By detecting TCRmn = 0000H, the INTTMmn output is generated. After that, the above operation is repeated.
Operation stop	The TTmn bit is set to 1. → The TTmn bit automatically returns to 0 because it is a trigger bit.	TEmn = 0, and count operation stops. The TCRmn register holds count value and stops.
TAU stop	The TAUmEN bit of the PER0 register is cleared to 0. →	Power-off status All circuits are initialized and SFR of each channel is also initialized.

Operation is resumed.

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

15.6.9 Hardware trigger wait mode (select mode, sequential conversion mode)

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

<2> If a hardware trigger is input while in the hardware trigger standby status, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input.

<3> When A/D conversion ends, the conversion result is stored in the A/D conversion result register (ADCR, ADCRH), and the A/D conversion end interrupt request signal (INTAD) is generated. After A/D conversion ends, the next A/D conversion immediately starts. (At this time, no hardware trigger is necessary.)

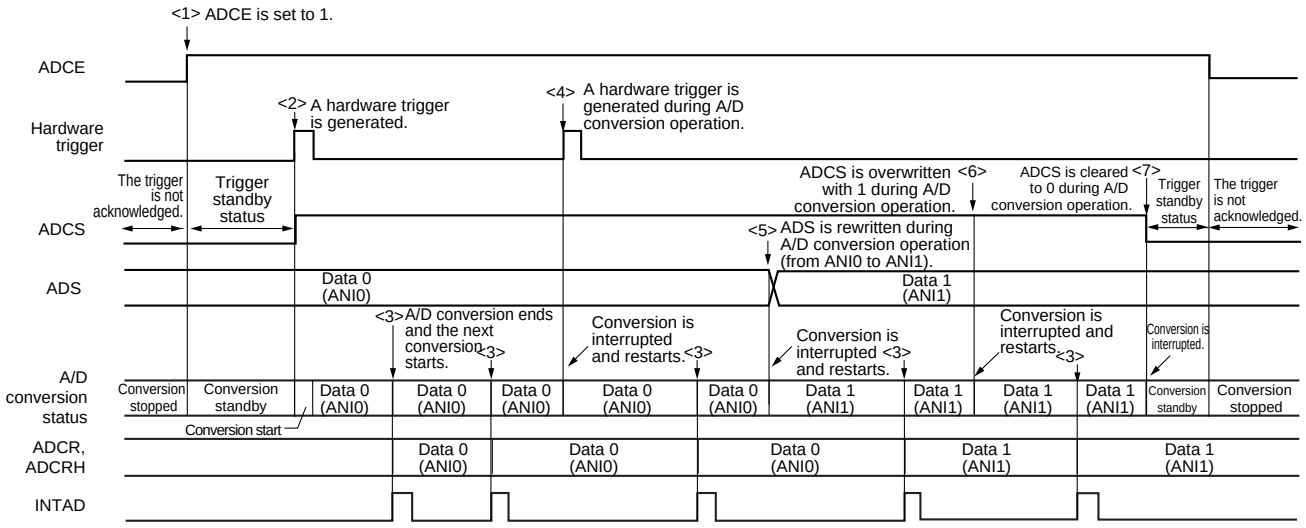
<4> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<5> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.

<6> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the system enters the hardware trigger standby status, and the A/D converter enters the stop status. When ADCE = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

Figure 15-26. Example of Hardware Trigger Wait Mode (Select Mode, Sequential Conversion Mode) Operation Timing



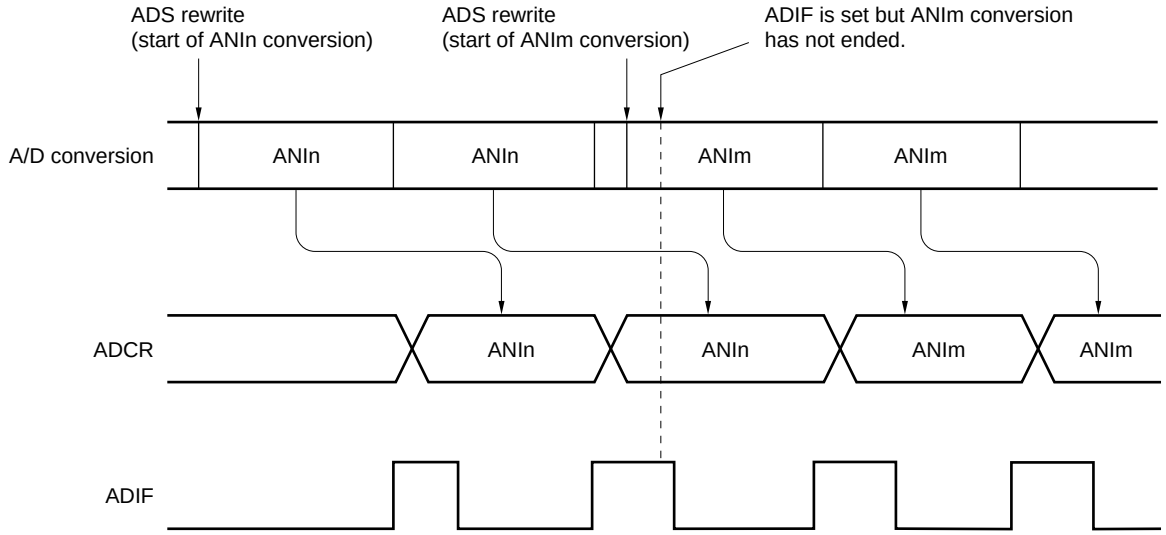
(7) Interrupt request flag (ADIF)

The interrupt request flag (ADIF) is not cleared even if the analog input channel specification register (ADS) is changed.

Therefore, if an analog input pin is changed during A/D conversion, the A/D conversion result and ADIF flag for the pre-change analog input may be set just before the ADS register rewrite. Caution is therefore required since, at this time, when ADIF flag is read immediately after the ADS register rewrite, ADIF flag is set despite the fact A/D conversion for the post-change analog input has not ended.

When A/D conversion is stopped and then resumed, clear ADIF flag before the A/D conversion operation is resumed.

Figure 15-46. Timing of A/D Conversion End Interrupt Request Generation



(8) Conversion results just after A/D conversion start

While in the software trigger mode or hardware trigger no-wait mode, the first A/D conversion value immediately after A/D conversion starts may not fall within the rating range if the ADCS bit is set to 1 within 1 μs after the ADCE bit was set to 1. Take measures such as polling the A/D conversion end interrupt request (INTAD) and removing the first conversion result.

(9) A/D conversion result register (ADCR, ADCRH) read operation

When a write operation is performed to A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), and A/D port configuration register (ADPC), the contents of the ADCR and ADCRH registers may become undefined. Read the conversion result following conversion completion before writing to the ADM0, ADS, or ADPC register. Using a timing other than the above may cause an incorrect conversion result to be read.

Figure 18-8. Format of Serial Communication Operation Setting Register mn (SCRmn) (2/2)

Address: F0118H, F0119H (SCR00) to F011EH, F011FH (SCR03), After reset: 0087H R/W
F0158H, F0159H (SCR10) to F015EH, F015FH (SCR13)

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

PTCmn1	PTCmn0	Setting of parity bit in UART mode	
		Transmission	Reception
0	0	Does not output the parity bit.	Receives without parity
0	1	Outputs 0 parity[Note 3].	No parity judgment
1	0	Outputs even parity.	Judged as even parity.
1	1	Outputs odd parity.	Judges as odd parity.
Be sure to set PTCmn1, PTCmn0 = 0, 0 in the CSI mode and simplified I^2C mode.			

DIRmn	Selection of data transfer sequence in CSI and UART modes
0	Inputs/outputs data with MSB first.
1	Inputs/outputs data with LSB first.
Be sure to clear DIRmn = 0 in the simplified I^2C mode.	

SLCmn1[Note 1]	SLCmn0	Setting of stop bit in UART mode
0	0	No stop bit
0	1	Stop bit length = 1 bit
1	0	Stop bit length = 2 bits (mn = 00, 02, 10 , 12 only)
1	1	Setting prohibited
When the transfer end interrupt is selected, the interrupt is generated when all stop bits have been completely transferred. Set 1 bit (SLCmn1, SLCmn0 = 0, 1) during UART reception and in the simplified I^2C mode. Set no stop bit (SLCmn1, SLCmn0 = 0, 0) in the CSI mode. Set 1 bit (SLCmn1, SLCmn0 = 0, 1) or 2 bits (SLCmn1, SLCmn0 = 1, 0) during UART transmission.		

DLSmn1[Note 2]	DLSmn0	Setting of data length in CSI and UART modes
0	1	9-bit data length (stored in bits 0 to 8 of the SDRmn register) (settable in UART mode only)
1	0	7-bit data length (stored in bits 0 to 6 of the SDRmn register)
1	1	8-bit data length (stored in bits 0 to 7 of the SDRmn register)
Other than above		Setting prohibited
Be sure to set DLSmn1, DLSmn0 = 1, 1 in the simplified I^2C mode.		

Notes 1. The SCR00, SCR02, SCR10, and SCR12 registers only.

2. The SCR00, SCR01, and SCR10 registers, and the SCR11 register of 100-pin products only. Others are fixed to 1.

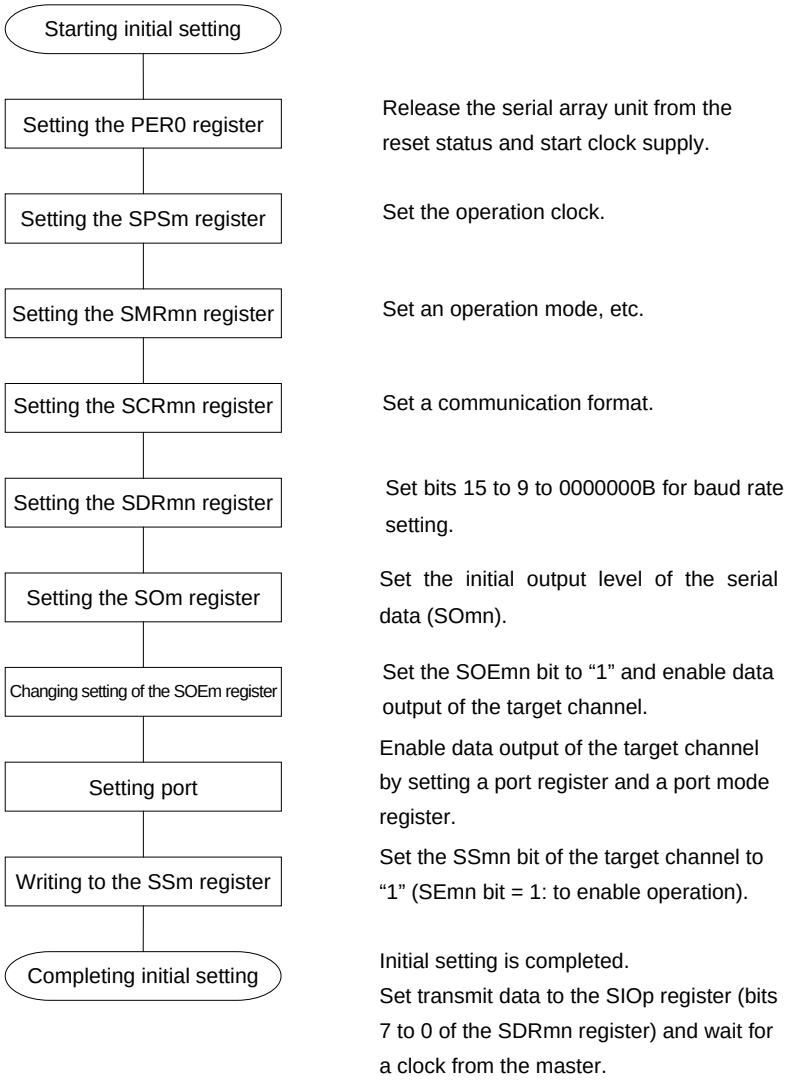
3. 0 is always added regardless of the data contents.

Caution Be sure to clear bits 3, 6, and 11 to "0" (Also clear bit 5 of the SCR01, SCR03, SCR11, or SCR13 register to 0). Be sure to set bit 2 to "1".

Remark m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 10, 30), mn = 00 to 03, 10 to 13

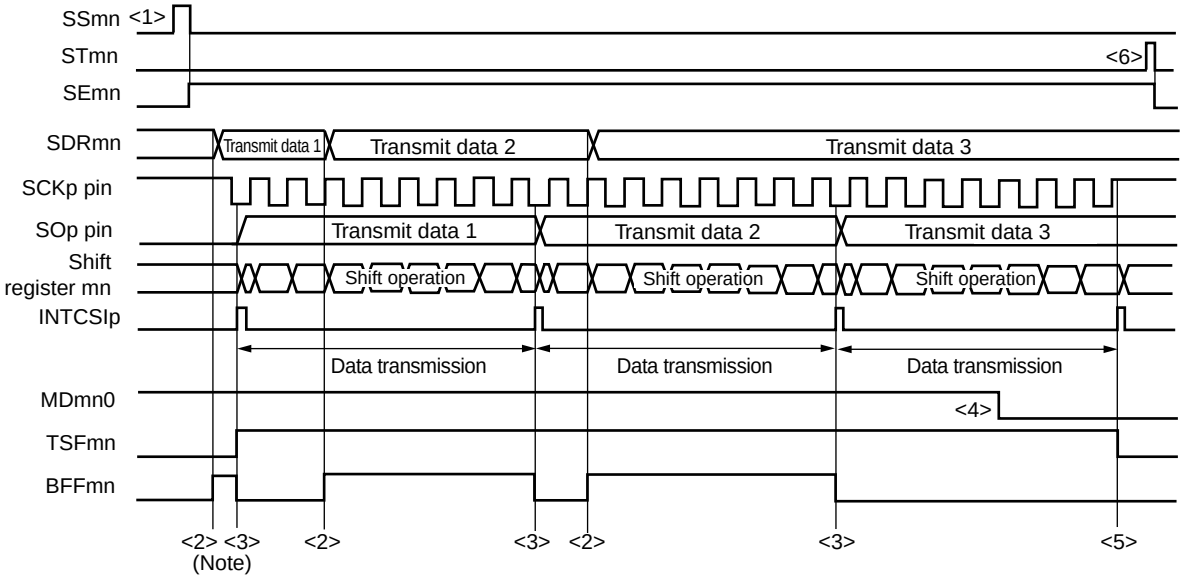
(2) Operation procedure

Figure 18-51. Initial Setting Procedure for Slave Transmission



(4) Processing flow (in continuous transmission mode)

Figure 18-56. Timing Chart of Slave Transmission (in Continuous Transmission Mode)
(Type 1: DAPmn = 0, CKPmn = 0)



Note If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

Caution The MDmn0 bit of serial mode register mn (SMRmn) can be rewritten even during operation. However, rewrite it before transfer of the last bit is started.

Remark m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), p: CSI number (p = 00, 10, 30), mn = 00, 02, 12

18.6.5 Procedure for processing errors that occurred during UART (UART0 to UART3) communication

The procedure for processing errors that occurred during UART (UART0 to UART3) communication is described in Figures 18-97 and 18-98.

Figure 18-97. Processing Procedure in Case of Parity Error or Overrun Error

Software Manipulation	Hardware Status	Remark
Reads serial data register mn (SDRmn). →	The BFFmn bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads serial status register mn (SSRmn).		Error type is identified and the read value is used to clear error flag.
Writes 1 to serial flag clear trigger register mn (SIRmn). →	Error flag is cleared.	Error can be cleared only during reading, by writing the value read from the SSRmn register to the SIRmn register without modification.

Figure 18-98. Processing Procedure in Case of Framing Error

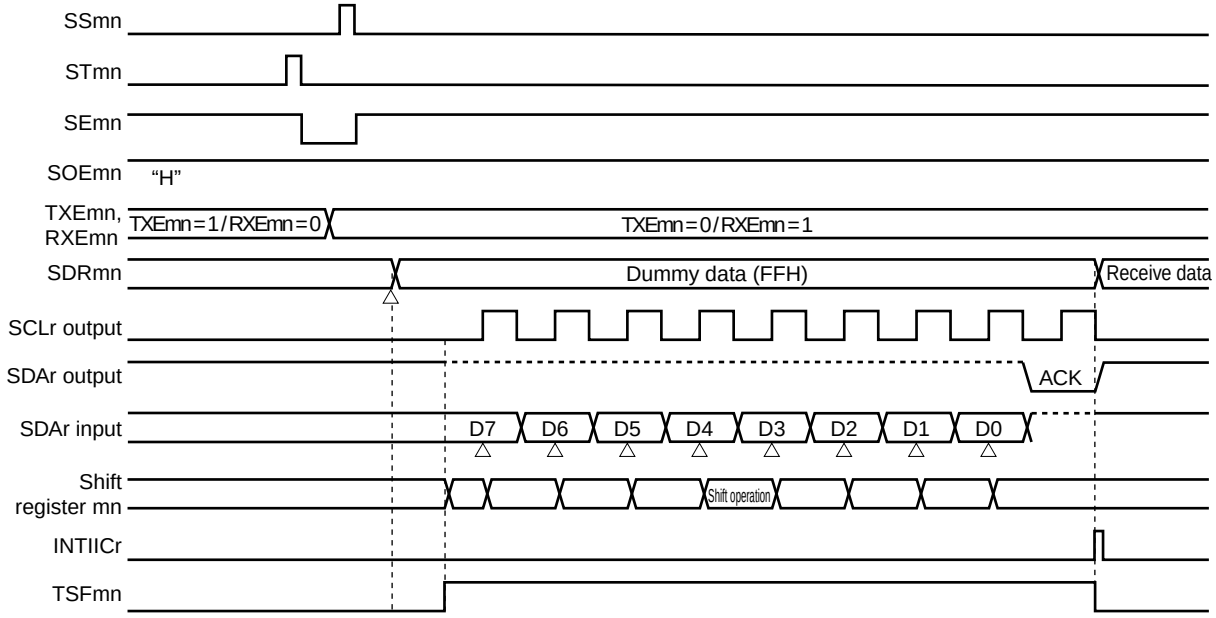
Software Manipulation	Hardware Status	Remark
Reads serial data register mn (SDRmn). →	The BFFmn bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads serial status register mn (SSRmn).		Error type is identified and the read value is used to clear error flag.
Writes serial flag clear trigger register mn (SIRmn). →	Error flag is cleared.	Error can be cleared only during reading, by writing the value read from the SSRmn register to the SIRmn register without modification.
Sets the STmn bit of serial channel stop register m (STm) to 1. →	The SEmn bit of serial channel enable status register m (SEm) is set to 0 and channel n stops operating.	
Synchronization with other party of communication		Synchronization with the other party of communication is re-established and communication is resumed because it is considered that a framing error has occurred because the start bit has been shifted.
Sets the SSmn bit of serial channel start register m (SSm) to 1. →	The SEmn bit of serial channel enable status register m (SEm) is set to 1 and channel n is enabled to operate.	

Remark m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

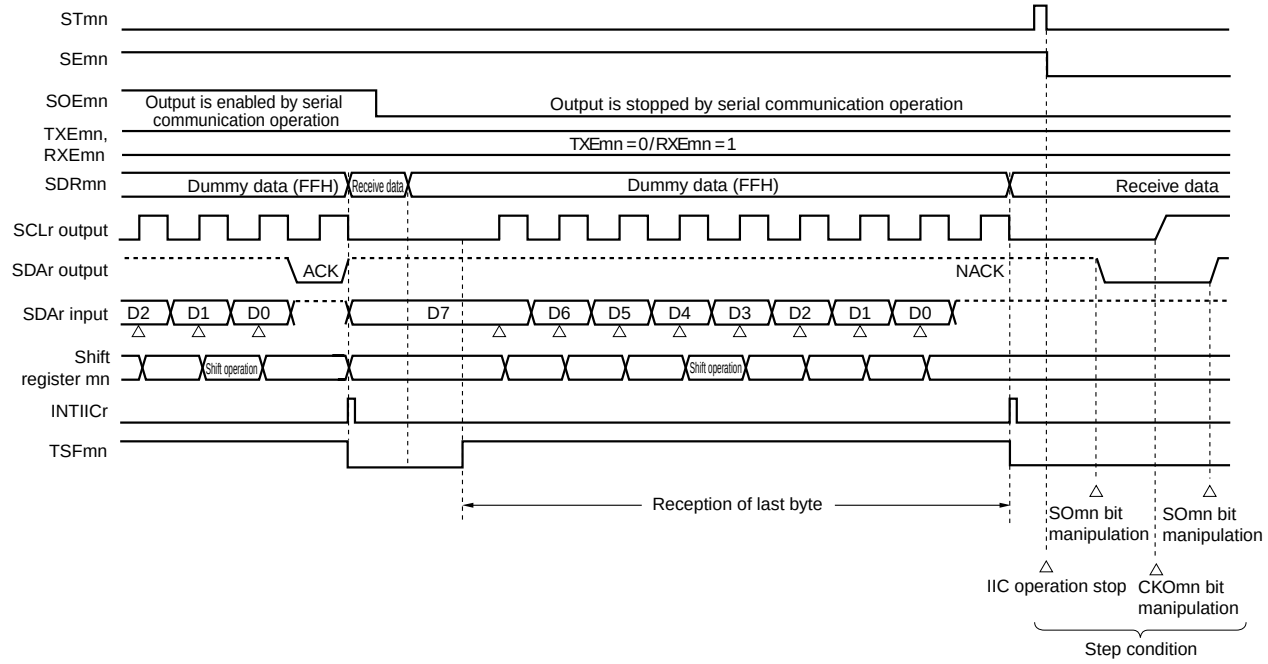
(2) Processing flow

Figure 18-112. Timing Chart of Data Reception

(a) When starting data reception



(b) When receiving last data



Remark m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), r: IIC number (r = 00, 10, 30), mn = 00, 02, 12

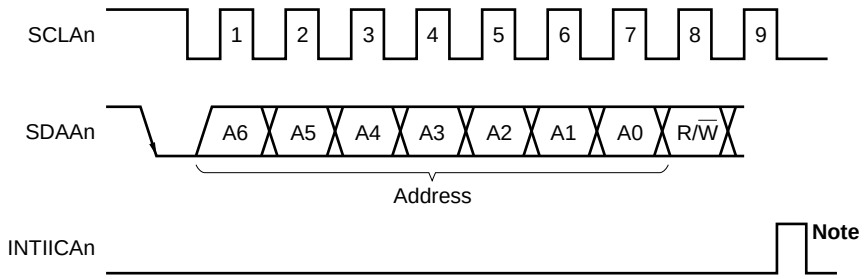
19.5.2 Addresses

The address is defined by the 7 bits of data that follow the start condition.

An address is a 7-bit data segment that is output in order to select one of the slave devices that are connected to the master device via the bus lines. Therefore, each slave device connected via the bus lines must have a unique address.

The slave devices include hardware that detects the start condition and checks whether or not the 7-bit address data matches the data values stored in the slave address register n (SVAn). If the address data matches the SVAn register values, the slave device is selected and communicates with the master device until the master device generates a start condition or stop condition.

Figure 19-17. Address



Note INTIICAn is not issued if data other than a local address or extension code is received during slave device operation.

Addresses are output when a total of 8 bits consisting of the slave address and the transfer direction described in **19.5.3 Transfer direction specification** are written to the IICA shift register n (IICAn). The received addresses are written to the IICAn register.

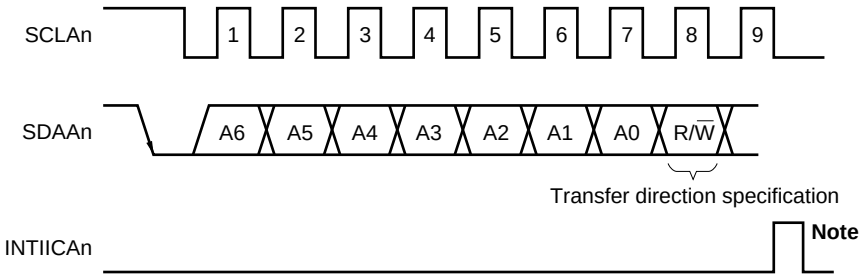
The slave address is assigned to the higher 7 bits of the IICAn register.

19.5.3 Transfer direction specification

In addition to the 7-bit address data, the master device sends 1 bit that specifies the transfer direction.

When this transfer direction specification bit has a value of “0”, it indicates that the master device is transmitting data to a slave device. When the transfer direction specification bit has a value of “1”, it indicates that the master device is receiving data from a slave device.

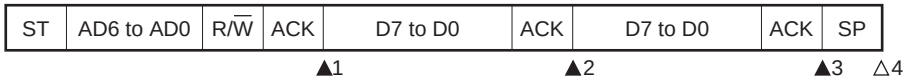
Figure 19-18. Transfer Direction Specification



Note INTIICAn is not issued if data other than a local address or extension code is received during slave device operation.

Remark n = 0

(ii) When WTIMn = 1



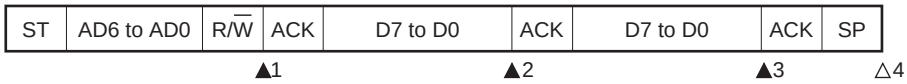
▲1 ▲2 ▲3 △4

▲1: IICSn = 0101×110B
▲2: IICSn = 0001×100B
▲3: IICSn = 0001××00B
△4: IICSn = 00000001B

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

(b) When arbitration loss occurs during transmission of extension code

(i) When WTIMn = 0



▲1 ▲2 ▲3 △4

▲1: IICSn = 0110×010B
▲2: IICSn = 0010×000B
▲3: IICSn = 0010×000B
△4: IICSn = 00000001B

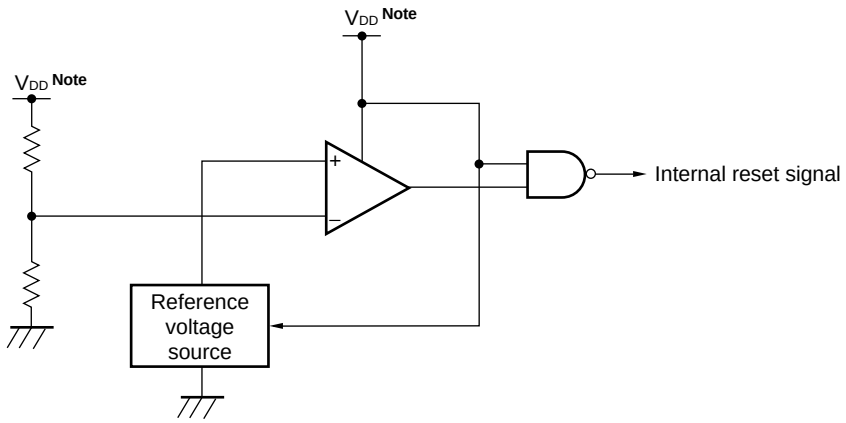
Remark ▲: Always generated
△: Generated only when SPIEn = 1
×: Don't care

Remark n = 0

28.2 Configuration of Power-on-reset Circuit

The block diagram of the power-on-reset circuit is shown in Figure 28-1.

Figure 28-1. Block Diagram of Power-on-reset Circuit

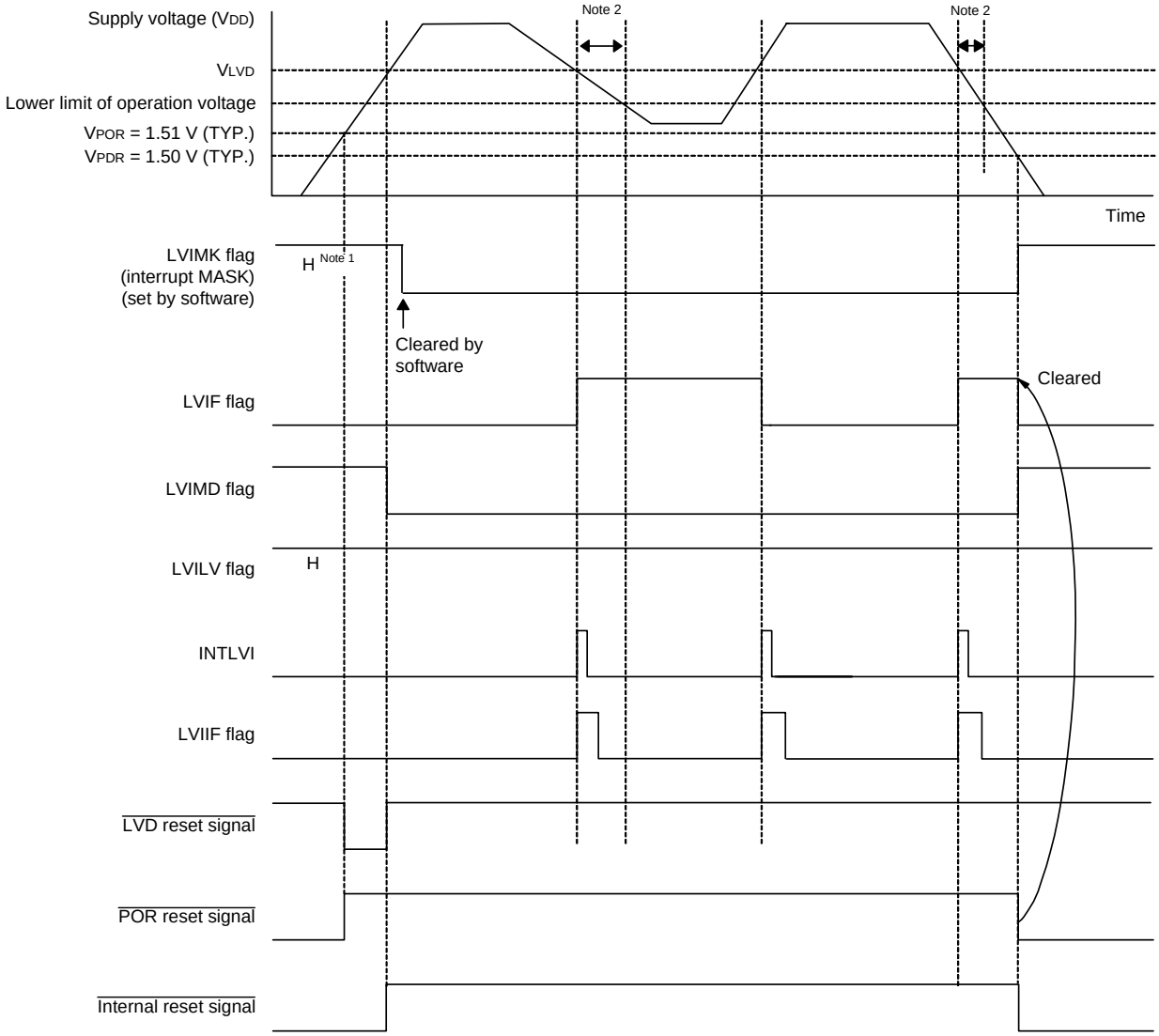


Note Internal power supply voltage (internal V_{DD}) when using the battery backup function.

28.3 Operation of Power-on-reset Circuit

The timing of generation of the internal reset signal by the power-on-reset circuit and voltage detector is shown below.

Figure 29 - 18 Timing of Voltage Detector Internal Interrupt Signal Generation (Option Byte LVIMDS1, LVIMDS0 = 0, 1)



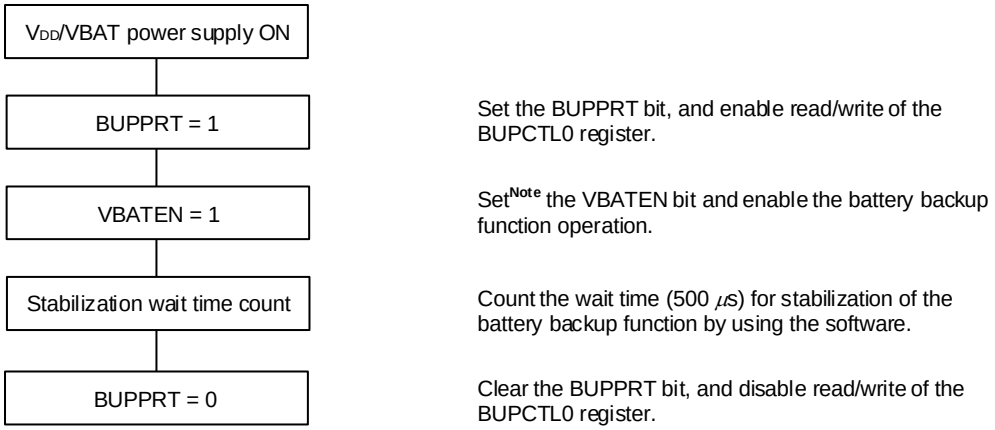
Note 1. The LVIMK flag is set to “1” by reset signal generation.

Note 2. When the voltage falls, this LSI should be placed in the STOP mode, or placed in the reset state by controlling the externally input reset signal, before the voltage falls below the operating voltage range defined in **41.4 AC Characteristics**. When restarting the operation, make sure that the operation voltage has returned within the range of operation.

Remark VPOR: POR power supply rise detection voltage
VPDR: POR power supply fall detection voltage

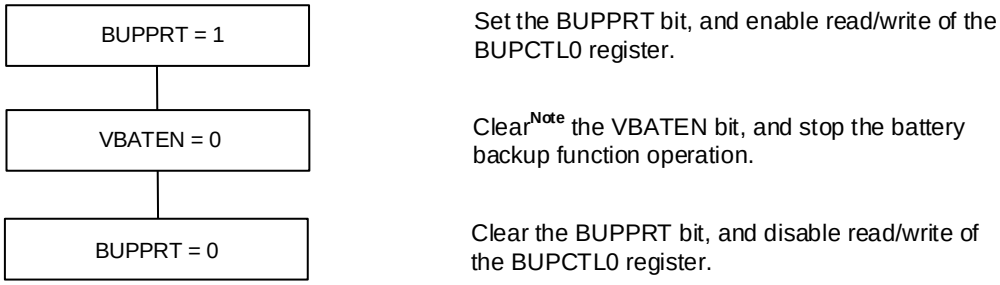
Figure 30-7 shows the procedure for setting the battery backup function operation, and Figure 30-8 shows the procedure for setting the battery backup function stop.

<R> **Figure 30-7. Procedure for Setting Battery Backup Function Operation**



Note Write 1 after writing 0 to set the VBATEN bit to 1.

<R> **Figure 30-8. Procedure for Setting Battery Backup Function Stop**



Note Write 0 after writing 1 to clear the VBATEN bit to 0.

CHAPTER 37 ON-CHIP DEBUG FUNCTION

37.1 Connecting E1 On-chip Debugging Emulator

The RL78 microcontroller uses the V_{DD}, $\overline{RESET}$, TOOL0, and V_{SS} pins to communicate with the host machine via an E1 on-chip debugging emulator. Serial communication is performed by using a single-line UART that uses the TOOL0 pin.

Caution The RL78 microcontroller has an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. Renesas Electronics is not liable for problems occurring when the on-chip debug function is used.