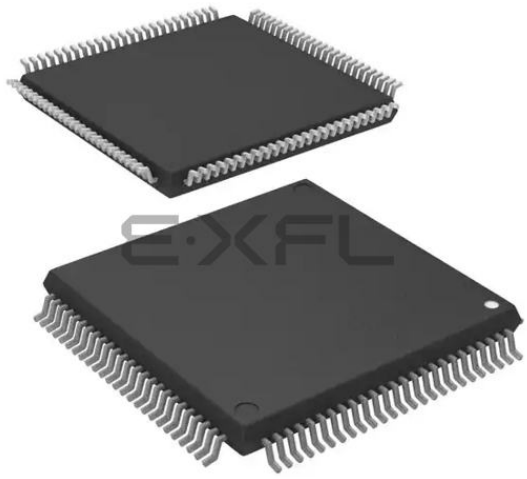
Renesas Electronics America Inc - UPD70F3842GC-UEU-AX Datasheet



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	Not For New Designs
Core Processor	V850ES
Core Size	32-Bit Single-Core
Speed	20MHz
Connectivity	CSI, EBI/EMI, I²C, UART/USART
Peripherals	DMA, LVD, PWM, WDT
Number of I/O	83
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	80K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 12x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd70f3842gc-ueu-ax

4.3 Port Configuration

The ports consist of the following hardware.

Table 4-4. Port Configuration (V850ES/JC3-L (40-pin))

Item	Configuration
Control registers	Port n mode register (PMn: n = 0, 3 to 5, 7, 9, CM, DL) Port n mode control register (PMCn: n = 0, 3 to 5, 9) Port n function control register (PFCn: n = 0, 3 to 5, 9) Port n function control expansion register (PFCEn: n = 0, 5, 9) Port n function register (PFn: n = 0, 3 to 5, 9)
Port pins	I/O: 34

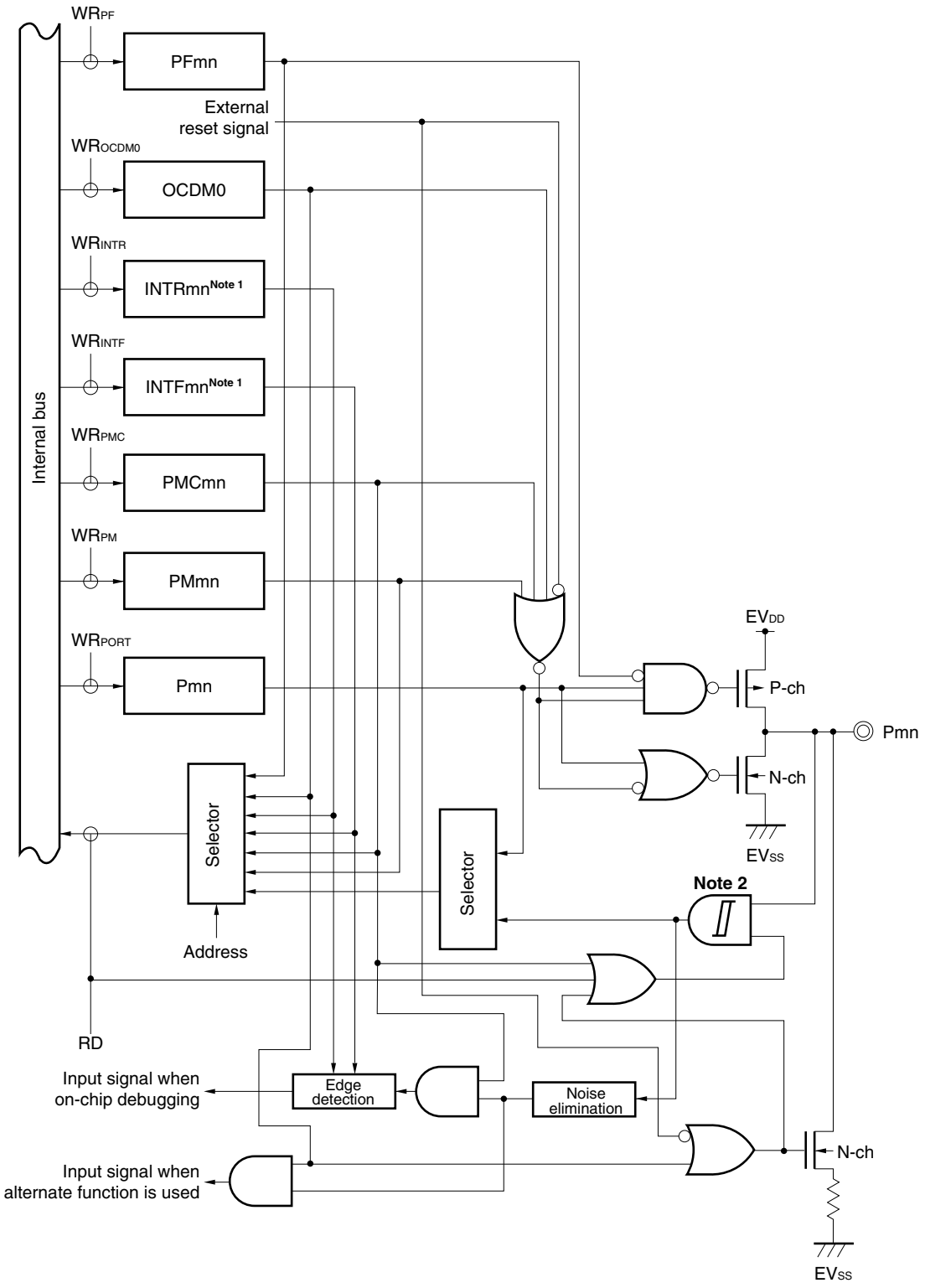
Table 4-5. Port Configuration (V850ES/JC3-L (48-pin))

Item	Configuration
Control registers	Port n mode register (PMn: n = 0, 1, 3 to 5, 7, 9, CM, DL) Port n mode control register (PMCn: n = 0, 3 to 5, 9) Port n function control register (PFCn: n = 0, 3 to 5, 9) Port n function control expansion register (PFCEn: n = 0, 3, 5, 9) Port n function register (PFn: n = 0, 3 to 5, 9)
Port pins	I/O: 34

Table 4-6. Port Configuration (V850ES/JE3-L)

Item	Configuration
Control registers	Port n mode register (PMn: n = 0, 1, 3 to 5, 7, 9, CM, DL) Port n mode control register (PMCn: n = 0, 3 to 5, 9) Port n function control register (PFCn: n = 0, 3 to 5, 9) Port n function control expansion register (PFCEn: n = 0, 3, 5, 9) Port n function register (PFn: n = 0, 3 to 5, 9)
Port pins	I/O: 50

Figure 4-36. Block Diagram of Type AA-1



Notes 1. See **19.7 External Interrupt Request Input Pins (NMI and INTP0 to INTP7)**.

2. There are no hysteresis characteristics in port mode.

Table 4-17. Settings When Pins Are Used for Alternate Functions (1/5)

Function Name	Alternate Function		Pnx Bit of Pn Register	PMnx Bit of PMn Register	PMCnx Bit of PMCn Register	PFCEnx Bit of PFCEn Register	PFCnx Bit of PFCn Register	Other Bits (Registers)
	Name	I/O						
P02	NMI	Input	P02 = Setting not required	PM02 = Setting not required	PMC02 = 1	–	–	
P03	INTP0	Input	P03 = Setting not required	PM03 = Setting not required	PMC03 = 1	PFCE03 = 0	PFC03 = 0	
	ADTRG	Input	P03 = Setting not required	PM03 = Setting not required	PMC03 = 1	PFCE03 = 0	PFC03 = 1	
	RTC1HZ	Input	P03 = Setting not required	PM03 = Setting not required	PMC03 = 1	PFCE03 = 1	PFC03 = 1	
P04	INTP1	Input	P04 = Setting not required	PM04 = Setting not required	PMC04 = 1	PFCE04 = 0	PFC04 = 0	
	RTCDIV	Output	P04 = Setting not required	PM04 = Setting not required	PMC04 = 1	PFCE04 = 0	PFC04 = 1	
	RTCCL	Output	P04 = Setting not required	PM04 = Setting not required	PMC04 = 1	PFCE04 = 1	PFC04 = 0	
P05	INTP2	Input	P05 = Setting not required	PM05 = Setting not required	PMC05 = 1	–	–	
	$\overline{\text{DRST}}$	Input	P05 = Setting not required	PM05 = Setting not required	PMC05 = Setting not required	–	–	OCDM0 (OCDM) = 1
P06	INTP3	Input	P06 = Setting not required	PM06 = Setting not required	PMC06 = 1	–	–	
P10	ANO0	Output	P10 = Setting not required	PM10 = 1	–	–	–	
P30	TXDA0	Output	P30 = Setting not required	PM30 = Setting not required	PMC30 = 1	–	PFC30 = 0	
	SOB4	Output	P30 = Setting not required	PM30 = Setting not required	PMC30 = 1	–	PFC30 = 1	
P31	RXDA0	Input	P31 = Setting not required	PM31 = Setting not required	PMC31 = 1	–	**Note**, PFC31 = 0	
	INTP7	Input	P31 = Setting not required	PM31 = Setting not required	PMC31 = 1	–	**Note**, PFC31 = 0	
	SIB4	Input	P31 = Setting not required	PM31 = Setting not required	PMC31 = 1	–	PFC31 = 1	
P32	ASCKA0	Input	P32 = Setting not required	PM32 = Setting not required	PMC32 = 1	PFCE32 = 0	PFC32 = 0	
	$\overline{\text{SCKB4}}$	I/O	P32 = Setting not required	PM32 = Setting not required	PMC32 = 1	PFCE32 = 0	PFC32 = 1	
	TIP00	Input	P32 = Setting not required	PM32 = Setting not required	PMC32 = 1	PFCE32 = 1	PFC32 = 0	
	TOP00	Output	P32 = Setting not required	PM32 = Setting not required	PMC32 = 1	PFCE32 = 1	PFC32 = 1	
P33	TIP01	Input	P33 = Setting not required	PM33 = Setting not required	PMC33 = 1	–	PFC33 = 0	
	TOP01	Output	P33 = Setting not required	PM33 = Setting not required	PMC33 = 1	–	PFC33 = 1	
P34	TIP10	Input	P34 = Setting not required	PM34 = Setting not required	PMC34 = 1	–	PFC34 = 0	
	TOP10	Output	P34 = Setting not required	PM34 = Setting not required	PMC34 = 1	–	PFC34 = 1	
P35	TIP11	Input	P35 = Setting not required	PM35 = Setting not required	PMC35 = 1	–	PFC35 = 0	
	TOP11	Output	P35 = Setting not required	PM35 = Setting not required	PMC35 = 1	–	PFC35 = 1	

Note INTP7 and RXDA0 are alternate functions. When using the pin for RXDA0, disable edge detection for INTP7 (clear the INTF3.INTF31 bit and the INTR3.INTR31 bit to 0). When using the pin for INTP7, stop UARTA0 reception (clear the UA0CTL0.UA0RXE bit to 0).

(2) Clock control register (CKC)

The CKC register is a special register. Data can be written to this register only in a combination of specific sequences (see **3.4.7 Special registers**).

The CKC register controls the internal system clock in the PLL mode.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 0AH.

After reset: 0AH R/W Address: FFFFF822H

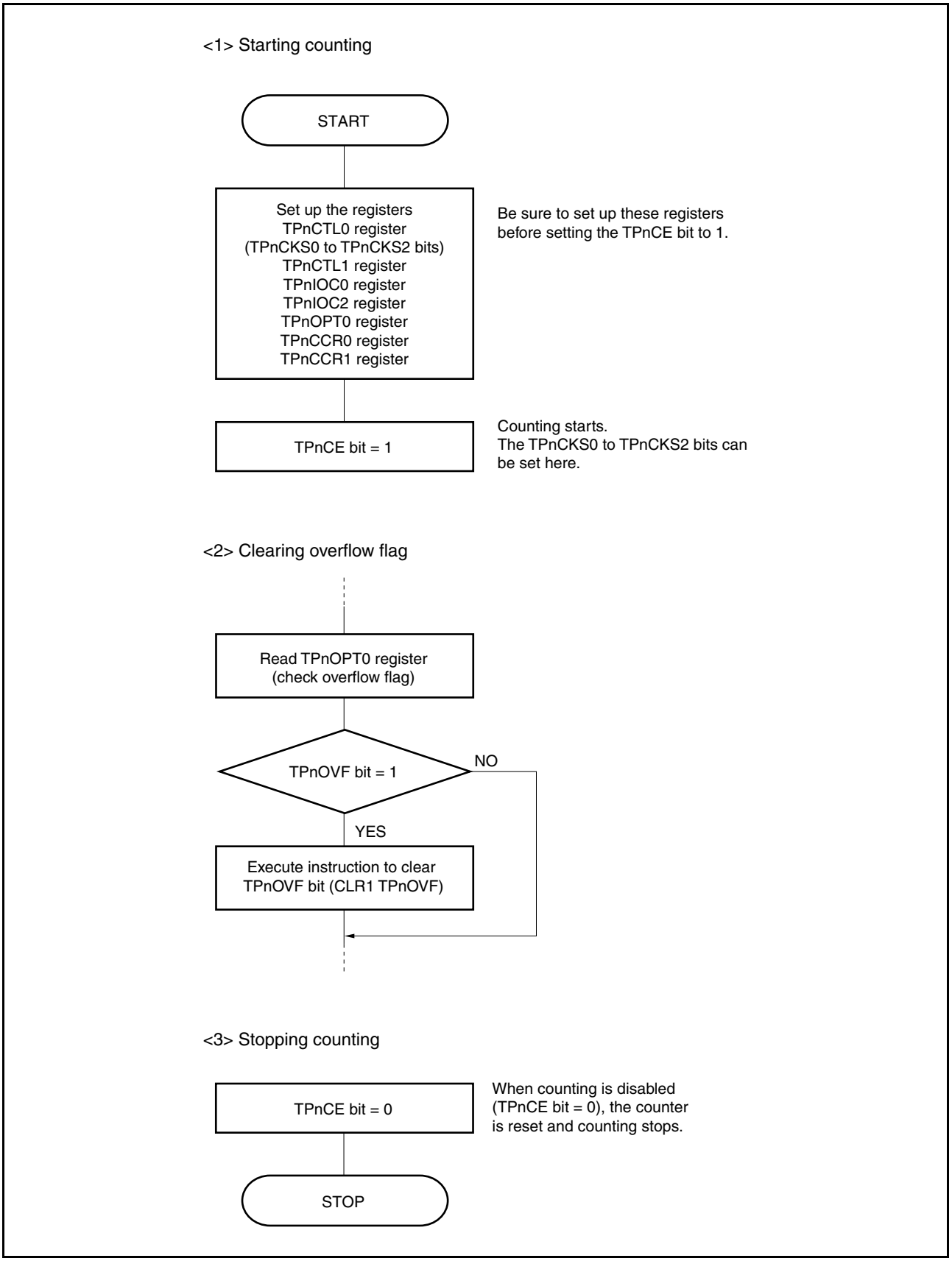
	7	6	5	4	3	2	1	0
CKC	0	0	0	0	1	0	1	CKDIV0

CKDIV0	Internal system clock (f_{XX}) in PLL mode
0	$f_{XX} = 4 \times f_X$ (f_X = 2.5 to 5.0 MHz)
1	Setting prohibited

Cautions 1. The PLL mode cannot be used when f_X = 5.0 to 10.0 MHz.

2. Be sure to set the CKC register to 0AH. If a value other than 0AH is set, the operation is not guaranteed.

Figure 6-56. Timing and Processing of Operations in Free-Running Timer Mode (Compare Function) (2/2)



(7) TMQ0 capture/compare register 0 (TQ0CCR0)

The TQ0CCR0 register can be used as a capture register or a compare register depending on the mode.

This register can be selected as a capture register or a compare register only in the free-running timer mode, according to the setting of the TQ0OPT0.TQ0CCS0 bit. In any other mode, this register can be used only as a compare register.

The TQ0CCR0 register can be read or written during operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

Caution Accessing the TQ0CCR0 register is prohibited in the following statuses. Moreover, if the system is in the wait status, the only way to cancel the wait status is to execute a reset. For details, see 3.4.9 (1) Accessing specific on-chip peripheral I/O registers.

- **When the CPU operates on the subclock and main clock oscillation is stopped**
- **When the CPU operates on the internal oscillator clock**

After reset: 0000H R/W Address: FFFFF546H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TQ0CCR0																

(a) Function as compare register

The TQ0CCR0 register can be rewritten even when the TQ0CTL0.TQ0CE bit = 1.

The set value of the TQ0CCR0 register is transferred to the CCR0 buffer register. When the value of the 16-bit counter matches the value of the CCR0 buffer register, a compare match interrupt request signal (INTTQ0CC0) is generated. If TOQ00 pin output is enabled at this time, the output of the TOQ00 pin is inverted (For details, see the descriptions of each operating mode.).

When the TQ0CCR0 register is used as a cycle register in the interval timer mode, external event count mode, external trigger pulse output mode, one-shot pulse output mode, or PWM output mode, the value of the 16-bit counter is cleared (0000H) if its count value matches the value of the CCR0 buffer register.

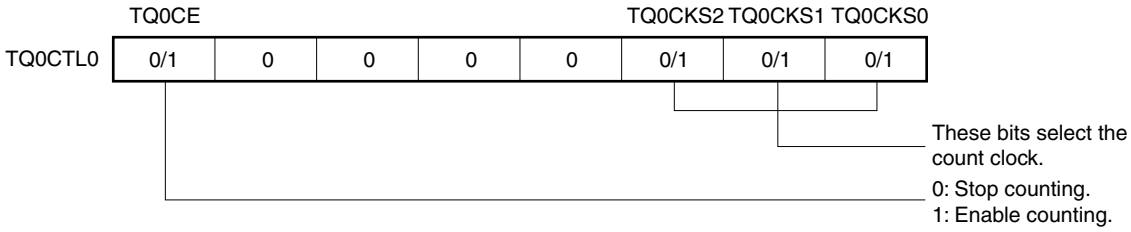
(b) Function as capture register

When the TQ0CCR0 register is used as a capture register in the free-running timer mode, the count value of the 16-bit counter is stored in the TQ0CCR0 register if the valid edge of the capture trigger input pin (TIQ00 pin) is detected. In the pulse-width measurement mode, the count value of the 16-bit counter is stored in the TQ0CCR0 register and the 16-bit counter is cleared (0000H) if the valid edge of the capture trigger input pin (TIQ00 pin) is detected.

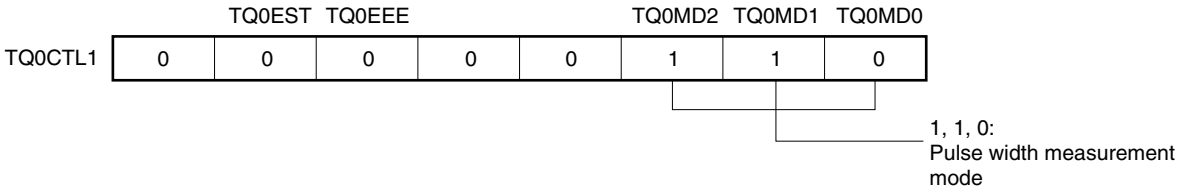
Even if the capture operation and reading the TQ0CCR0 register conflict, the correct value of the TQ0CCR0 register can be read.

Figure 7-67. Register Settings in Pulse Width Measurement Mode (1/2)

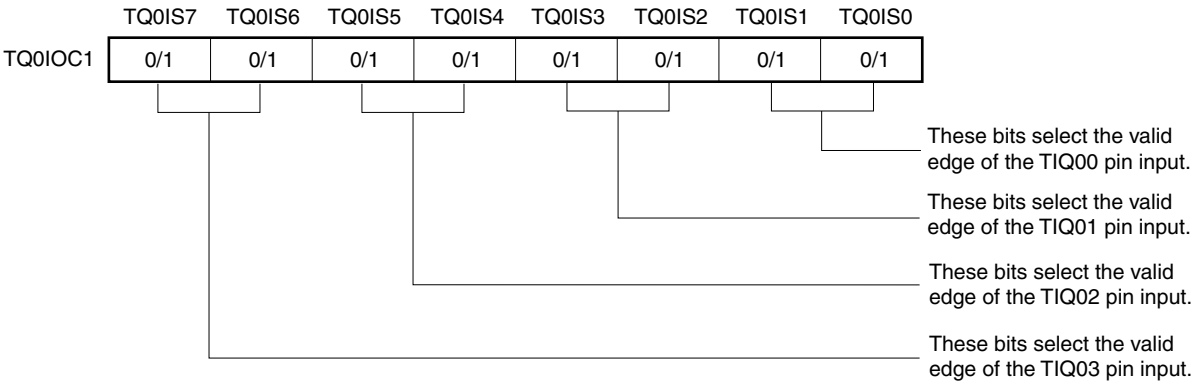
(a) TMQ0 control register 0 (TQ0CTL0)



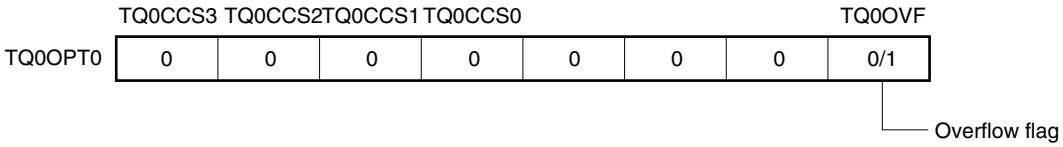
(b) TMQ0 control register 1 (TQ0CTL1)



(c) TMQ0 I/O control register 1 (TQ0IOC1)



(d) TMQ0 option register 0 (TQ0OPT0)



(a) Alarm interrupt setting examples (RC1ALM, RC1ALH, and RC1ALW setting examples)

Tables 10-4 and 10-5 show setting examples if Sunday is RC1WEEK = 00, Monday is RC1WEEK = 01, Tuesday is RC1WEEK = 02, ···, and Saturday is RC1WEEK = 06.

Table 10-4. Alarm Setting Example if AMPM = 0 (RC1HOUR Register 12-Hour Display)

Register / Alarm Setting Time	RC1ALW	RC1ALH	RC1ALM
Sunday, 7:00 a.m.	01H	07H	00H
Sunday/Monday, 00:15 p.m.	03H	32H	15H
Monday/Tuesday/Friday, 5:30 p.m.	26H	25H	30H
Everyday, 10:45 p.m.	7FH	30H	45H

Table 10-5. Alarm Setting Example if AMPM = 1 (RC1HOUR Register 24-Hour Display)

Register / Alarm Setting Time	RC1ALW	RC1ALH	RC1ALM
Sunday, 7:00	01H	07H	00H
Sunday/Monday, 12:15	03H	12H	15H
Monday/Tuesday/Friday, 17:30	26H	17H	30H
Everyday, 22:45	7FH	22H	45H

(17) Prescaler mode register 0 (PRSM0)

The PRSM0 register is an 8-bit register that controls the generation of the real time counter count clock (f_{BRG}).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset : 00H R/W Address : FFFFF8B0H

	7	6	5	<4>	3	2	1	0
PRSM0	0	0	0	BGCE0	0	0	BGCS01	BGCS00

BGCE0	Main clock operation enable
0	Disabled
1	Enabled

BGCS01	BGCS00	Selection of real time counter source clock(f_{BGCS})		
			5 MHz	4 MHz
0	0	fx	200 ns	250 ns
0	1	fx/2	400 ns	500 ns
1	0	fx/4	800 ns	1 μs
1	1	fx/8	1.6 μs	2 μs

Cautions 1. Do not change the values of the BGCS00 and BGCS01 bits during real time counteroperation.

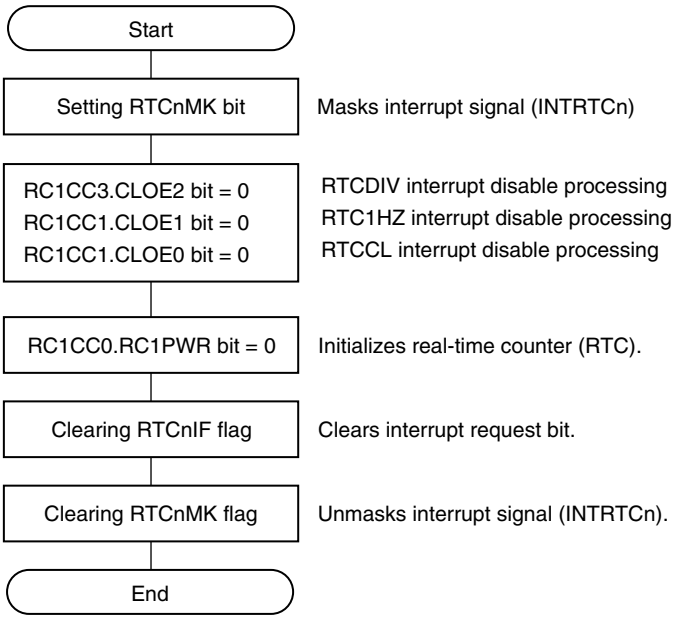
2. Set the PRSM0 register before setting the BGCE0 bit to 1.

3. Set the PRSM0 and PRSCM0 registers according to the main clock frequency that is used so as to obtain an f_{BRG} frequency of 32.768 kHz.

10.4.8 Initializing real-time counter

The procedure for initializing the real-time counter is shown below.

Figure 10-9. Initializing Real-Time Counter



Remarks 1. See **19.3.4 Interrupt control register (xxICn)** for details of the RTCnIF and RTCnMK bits.

2. n = 0 to 2

As shown in Figure 10-10, the watch can be accurately counted by incrementing the RC1SUBC count value, if a positive error faster than 32.768 kHz occurs at the resonator. Similarly, if a negative error slower than 32.768 kHz occurs at the resonator, the watch can be accurately counted by decrementing the RC1SUBC count value.

The RC1SUBC correction value is determined by using the RC1SUBU.F6 to RC1SUBU.F0 bits.

The F6 bit is used to determine whether to increment or decrement RC1SUBC and the F5 to F0 bits to determine the RC1SUBC value.

(1) Incrementing the RC1SUBC count value

The RC1SUBC count value is incremented by the value set using the F5 to F0 bits, by setting the F6 bit to 0.

Expression for calculating the increment value: (F5 to F0 bit value − 1) × 2

[Example of incrementing the RC1SUBC count value: F6 bit = 0]

If 15H (010101B) is set to the F5 to F0 bits

(15H − 1) × 2 = 40 (increments the RC1SUBC count value by 40)

RC1SUBC count value = 32,768 + 40 = 32,808

(2) Decrementing the RC1SUBC count value

The RC1SUBC count value is decremented by an inverted value of the value set using the F5 to F0 bits, by setting the F6 bit to 1.

Expression for calculating the decrement value: (Inverted value of F5 to F0 bit value + 1) × 2

[Example of decrementing the RC1SUBC count value: F6 bit = 1]

If 15H (010101B) is set to the F5 to F0 bits

Inverted data of 15H (010101B) = 2AH (101010B)

(2AH + 1) × 2 = 86 (decrements the RC1SUBC count value by 86)

RC1SUBC count value = 32,768 − 86 = 32,682

(2/2)

UAnDIR	Data transfer order
0	MSB first
1	LSB first

- This register can be rewritten only when the UAnPWR bit is 0 or the UAnTXE bit and the UAnRXE bit are 0.
- When transmission and reception are performed in the LIN format, set the UAnDIR bit to 1.

UAnPS1	UAnPS0	Parity selection during transmission	Parity selection during reception
0	0	No parity output	Reception with no parity
0	1	0 parity output	Reception with 0 parity
1	0	Odd parity output	Odd parity check
1	1	Even parity output	Even parity check

- This register is rewritten only when the UAnPWR bit is 0 or the UAnTXE bit and the UAnRXE bit are 0.
- If “Reception with 0 parity” is selected during reception, a parity check is not performed. Therefore, the UAnSTR.UAnPE bit is not set.
- When transmission and reception are performed in the LIN format, clear the UAnPS1 and UAnPS0 bits to 00.

UAnCL	Specification of data character length of 1 frame of transmit/receive data
0	7 bits
1	8 bits

- This register can be rewritten only when the UAnPWR bit is 0 or the UAnTXE bit and the UAnRXE bit are 0.
- When transmission and reception are performed in the LIN format, set the UAnCL bit to 1.

UAnSL	Specification of length of stop bit for transmit data
0	1 bit
1	2 bits

This register can be rewritten only when the UAnPWR bit is 0 or the UAnTXE bit and the UAnRXE bit are 0.

Remark For details of parity, see **15.6.6 Parity types and operations**.

(2) UARTAn control register 1 (UAnCTL1)

For details, see **15.7 (2) UARTAn control register 1 (UAnCTL1)**.

(3) UARTAn control register 2 (UAnCTL2)

For details, see **15.7 (3) UARTAn control register 2 (UAnCTL2)**.

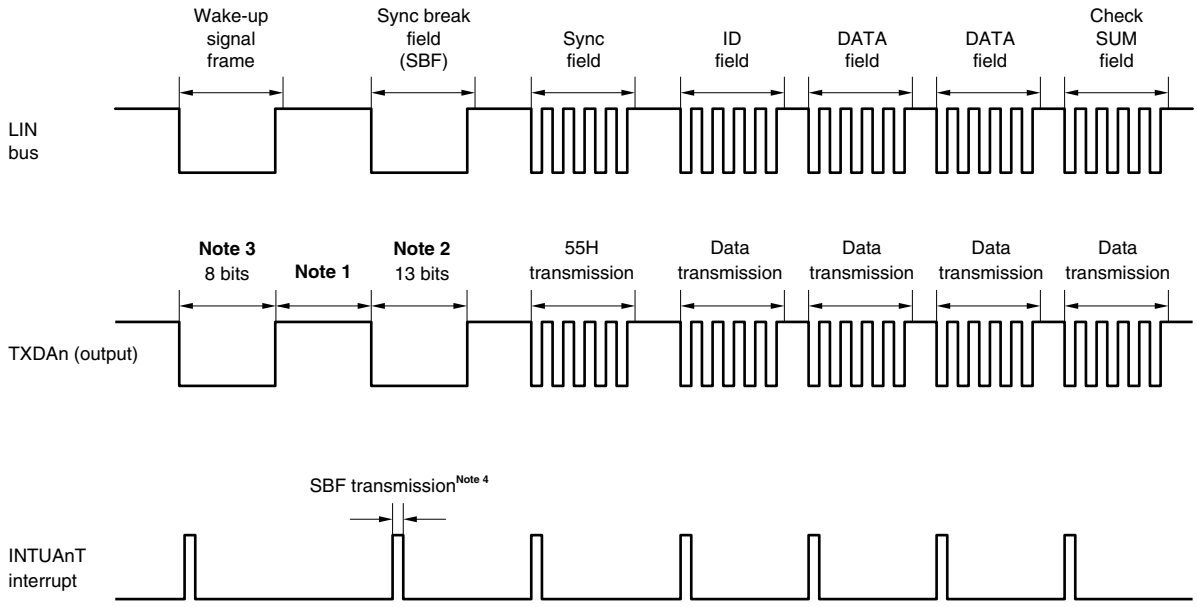
15.6.7 LIN transmission/reception format

The V850ES/JC3-L, V850ES/JE3-L have an SBF (Sync Break Field) transmission/reception control function to enable use of the LIN function.

Remark LIN stands for Local Interconnect Network and is a low-speed (1 to 20 kbps) serial communication protocol intended to reduce costs of automotive networks.
LIN communication is single-master communication, and up to 15 slaves can be connected to the master.
The LIN slaves are used to control the switches, actuators, and sensors, and these are connected to the LIN master via the LIN network.
Normally, the LIN master is connected to a network such as CAN (Controller Area Network).
In addition, the LIN bus uses a single-wire method and is connected to the nodes via a transceiver that complies with ISO9141.
In the LIN protocol, the master transmits a frame with baud rate information and the slave receives it and corrects the baud rate error. Therefore, communication is possible when the baud rate error in the slave is ±15% or less.

Figures 15-11 and 15-12 outline the transmission and reception manipulations of LIN.

Figure 15-11. LIN Transmission Format



Notes 1. The interval between each field is controlled by software.

2. SBF output is performed by hardware. The output width is the bit length set by the UAnOPT0.UAnSBL2 to UAnOPT0.UAnSBL0 bits. If even finer output width adjustments are required, such adjustments can be performed using the UAnCTLn.UAnBRS7 to UAnCTLn.UAnBRS0 bits.

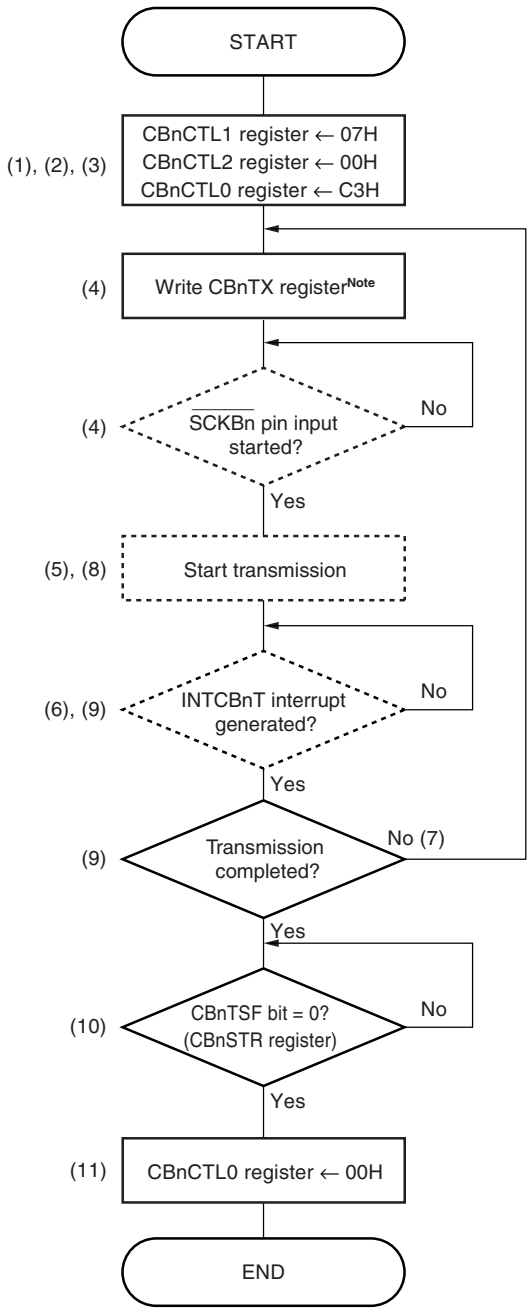
3. 80H transfer in the 8-bit mode is substituted for the wakeup signal frame.

4. A transmission enable interrupt request signal (INTUAnT) is output at the start of each transmission. The INTUAnT signal is also output at the start of each SBF transmission.

16.6.10 Continuous transfer mode (slave mode, transmission mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$) (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 111), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

Figure 16-23. Continuous Transfer Mode Operation (Slave Mode, Transmission Mode)



Note If the serial clock is input via the $\overline{SCKBn}$ pin of the master before the CBnTX register is written, data cannot be transmitted normally. In this case, initialize both the master and the slave.

Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in **Figure 16-24**.

17.14.2 When communication reservation function is disabled (IICFn.IICRSVn bit = 1)

If the IICCn.STTn bit is set when the bus is not being used by the V850ES/JC3-L and V850ES/JE3-L in a bus communication, this request is rejected and a start condition is not generated. There are two modes in which the bus is not used by the V850ES/JC3-L and V850ES/JE3-L.

- When arbitration results in the V850ES/JC3-L and V850ES/JE3-L being neither the master nor a slave
- When an extension code is received and slave operation is disabled ($\overline{ACK}$ is not returned and the bus was released when the IICCn.LRELn bit was set to 1).

To confirm whether the start condition was generated or request was rejected, check the IICFn.STCFn flag. The time shown in Table 17-7 is required until the STCFn flag is set after setting the STTn bit to 1. Therefore, secure the time by software.

Table 17-7. Wait Periods

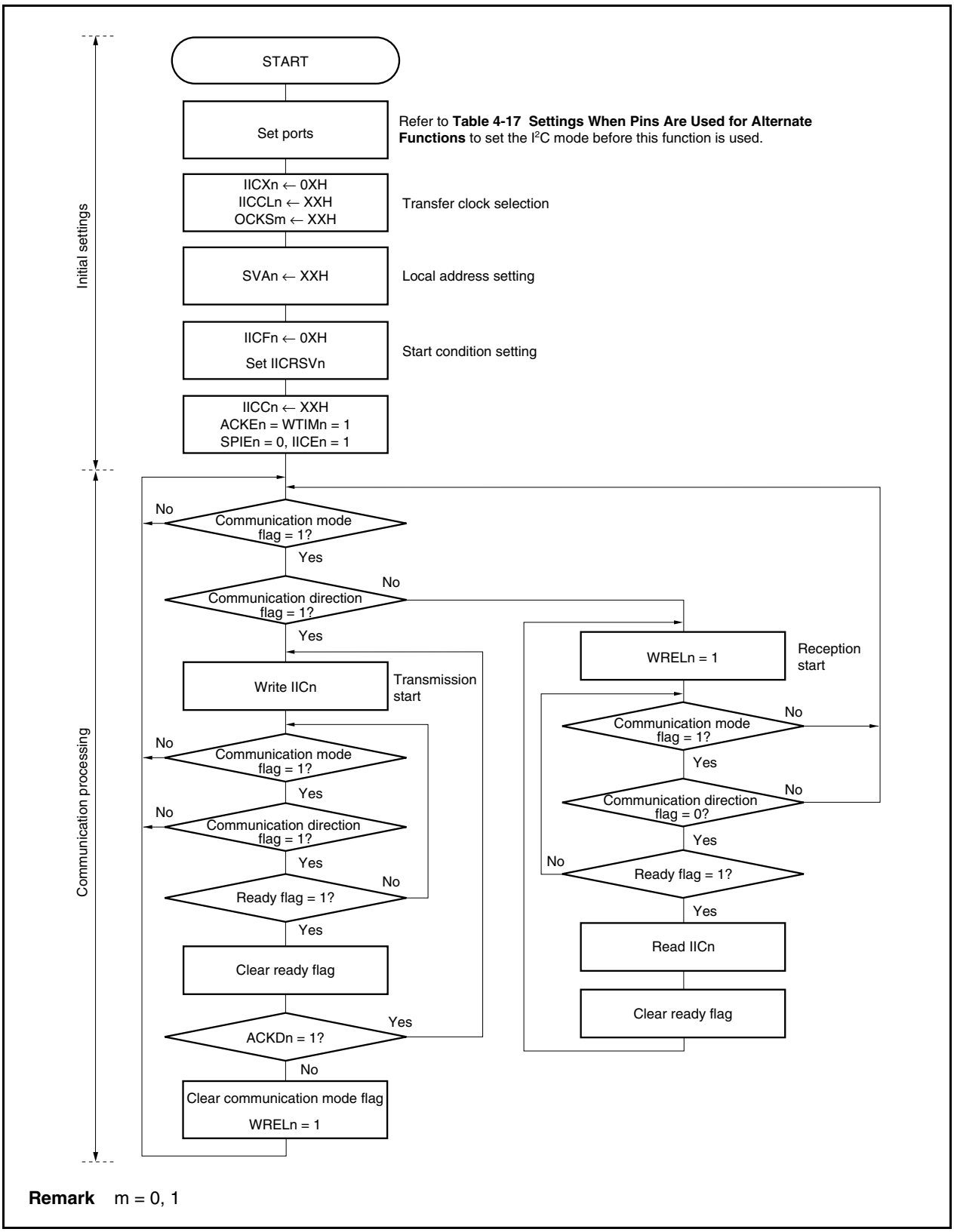
OCKSENm	OCKSm1	OCKSm0	CLn1	CLn0	Wait Period
1	0	0	0	×	10 clocks
1	0	1	0	×	15 clocks
1	1	0	0	×	20 clocks
1	1	1	0	×	25 clocks
0	0	0	1	0	5 clocks

Remarks 1. ×: don't care

2. m = 0, 1

For reception, the required number of data items are received and $\overline{ACK}$ is not returned for the next data immediately after transfer is complete. After that, the master device generates the stop condition or restart condition. This causes exit from communications.

Figure 17-21. Slave Operation Flowchart (1)



Remark m = 0, 1

(4) DMA addressing control registers 0 to 3 (DADC0 to DADC3)

The DADC0 to DADC3 registers are 16-bit registers that control the DMA transfer mode for DMA channel n (n = 0 to 3).

These registers can be read or written in 16-bit units.

Reset sets these registers to 0000H.

After reset: 0000H R/W Address: DADC0 FFFFF0D0H, DADC1 FFFFF0D2H, DADC2 FFFFF0D4H, DADC3 FFFFF0D6H

DADCn (n = 0 to 3)

15	14	13	12	11	10	9	8
0	DS0	0	0	0	0	0	0

7	6	5	4	3	2	1	0
SAD1	SAD0	DAD1	DAD0	0	0	0	0

DS0	Setting of transfer data size
0	8 bits
1	16 bits

SAD1	SAD0	Setting of count direction of the transfer source address
0	0	Increment
0	1	Decrement
1	0	Fixed
1	1	Setting prohibited

DAD1	DAD0	Setting of count direction of the destination address
0	0	Increment
0	1	Decrement
1	0	Fixed
1	1	Setting prohibited

Cautions 1. Be sure to clear bits 15, 13 to 8, and 3 to 0 of the DADCn register to 0.

2. Set the DADCn register during one of the following periods in which DMA transfer is disabled (DCHCn.Enn bit = 0).

- **Period from after reset to start of first DMA transfer**
- **Period from after channel initialization by DCHCn.INITn bit to start of DMA transfer**
- **Period from after completion of DMA transfer (DCHCn.TCn bit = 1) to start of the next DMA transfer**

3. The DSn0 bit specifies the size of the transfer data, and does not control bus sizing.

4. If the transfer data size is set to 16 bits (DS0 bit = 1), transfer cannot be started from an odd address. Transfer is always started from an address with the first bit of the lower address aligned to 0.

5. If DMA transfer is executed on an on-chip peripheral I/O register (as the transfer source or destination), be sure to specify the same transfer size as the register size. For example, to execute DMA transfer on an 8-bit register, be sure to specify 8-bit transfer.

(b) Repeatedly setting the INITn bit until transfer is forcibly terminated correctly

<1> Before starting DMA, copy the initial number of transfers of the channel to be forcibly terminated to a general-purpose register.
<2> Suppress a request from the DMA request source for the channel to be forcibly terminated (stop operation of the on-chip peripheral I/O).
<3> Check that the DMA transfer request for the channel to be forcibly terminated is not held pending, by using the DTFRn.DFn bit. If a DMA transfer request is held pending, wait until execution of the pending DMA transfer request is completed.
<4> When it has been confirmed that the DMA request for the channel to be forcibly terminated is not held pending, clear the Enn bit to 0.
<5> Again, clear the Enn bit for the channel to be forcibly terminated to 0.
If the internal RAM is the transfer source or destination of the channel to be forcibly terminated, execute this operation again.
<6> Set the INITn bit of the channel to be forcibly terminated to 1.
<7> Read the value of the DBCn register corresponding to the channel to be forcibly terminated, and compare it with the value copied in <1>. If the two values do not match, repeat operations <6> and <7>.

Remarks 1. When the value of the DBCn register is read in <7>, the initial number of transfers is read if forced termination has been correctly completed. If not, the remaining number of transfers is read.
2. Note that method (b) may take a long time if the application frequently uses DMA transfer for a channel other than the DMA channel to be forcibly terminated.

(5) Procedure for temporarily stopping DMA transfer (clearing Enn bit)

Stop and resume the DMA transfer under execution using the following procedure.

<1> Suppress a transfer request from the DMA request source (stop operation of the on-chip peripheral I/O).
<2> Check the DMA transfer request is not held pending, by using the DFn bit (check if the DFn bit = 0).
If a request is held pending, wait until execution of the pending DMA transfer request is completed.
<3> Check the TCn bit to confirm that DMA transfer is not complete (confirm that the TCn bit is 0). If the TCn bit is 1, execute the DMA transfer completion processing.
<4> If it has been confirmed that no DMA transfer request is held pending, clear the Enn bit to 0 (this operation suspends DMA transfer).
<5> Set the Enn bit to 1 to resume DMA transfer.
<6> Resume the operation of the DMA request source that has been stopped (start operation of the on-chip peripheral I/O).

(6) Memory boundary

The operation is not guaranteed if the address of the transfer source or destination exceeds the area of the DMA source or destination (internal RAM or on-chip peripheral I/O) during DMA transfer. (For details about the addresses of each area, see **Figure 3-2**.)

(7) Transferring misaligned data

DMA transfer of misaligned 16-bit data is not supported.
If an odd address is specified as the transfer source or destination, the least significant bit of the address is forcibly handled as 0.

23.3 Registers

The clock monitor is controlled by the clock monitor mode register (CLM).

(1) Clock monitor mode register (CLM)

The CLM register is a special register that can only be written in a combination of specific sequences (see **3.4.7 Special registers**).

This register is used to set the operation mode of the clock monitor.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF870H

	7	6	5	4	3	2	1	<0>
CLM	0	0	0	0	0	0	0	CLME

CLME	Clock monitor operation enable or disable
0	Disable clock monitor operation.
1	Enable clock monitor operation.

Cautions 1. Once the CLME bit has been set to 1, it cannot be cleared to 0 by any means other than a reset.

2. When a reset by the clock monitor occurs, the CLME bit is cleared to 0 and the RESF.CLMRF bit is set to 1.

31.8.8 D/A converter

(T_A = –40 to +85°C, V_{DD} = EV_{DD} = AV_{REF0} = AV_{REF1}, V_{SS} = EV_{SS} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution					8	bit
Overall error		R = 2 MΩ			±1.2	%FSR
Settling time		C = 20 pF			3	μs
Output resistor	R_O	Output data 55H		6.42		kΩ
Reference voltage	AV_{REF1}		2.7		3.6	V
AV_{REF1} current	AI_{REF1}	D/A conversion operating		1	2.5	mA
		D/A conversion stopped			5	μA

Remark R is the output pin load resistance and C is the output pin load capacitance.

31.8.9 LVI circuit characteristics

(T_A = –40 to +85°C, V_{DD} = EV_{DD} = AV_{REF0} = AV_{REF1} = 2.2 to 3.6 V, V_{SS} = EV_{SS} = AV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Detection voltage	V_{LVI0}		2.7	2.8	2.9	V
	V_{LVI1}		2.2	2.3	2.4	V
Response time[Note]	t_{LD}	At rising edge: After V_{DD} reaches V_{LVI0}/V_{LVI1} (MAX.) At falling edge: After V_{DD} has dropped to V_{LVI0}/V_{LVI1}(MIN.)		0.2	2.0	ms
Minimum pulse width	t_{LW}	V_{DD} = V_{LVI0}/V_{LVI1} (MIN.)	0.2			ms
Reference voltage stabilization wait time	t_{LWAIT}	After V_{DD} reaches V_{LVI0}/V_{LVI1} (MAX.)		0.1	0.2	ms

Note Time required to detect the detection voltage and output an interrupt or reset signal.

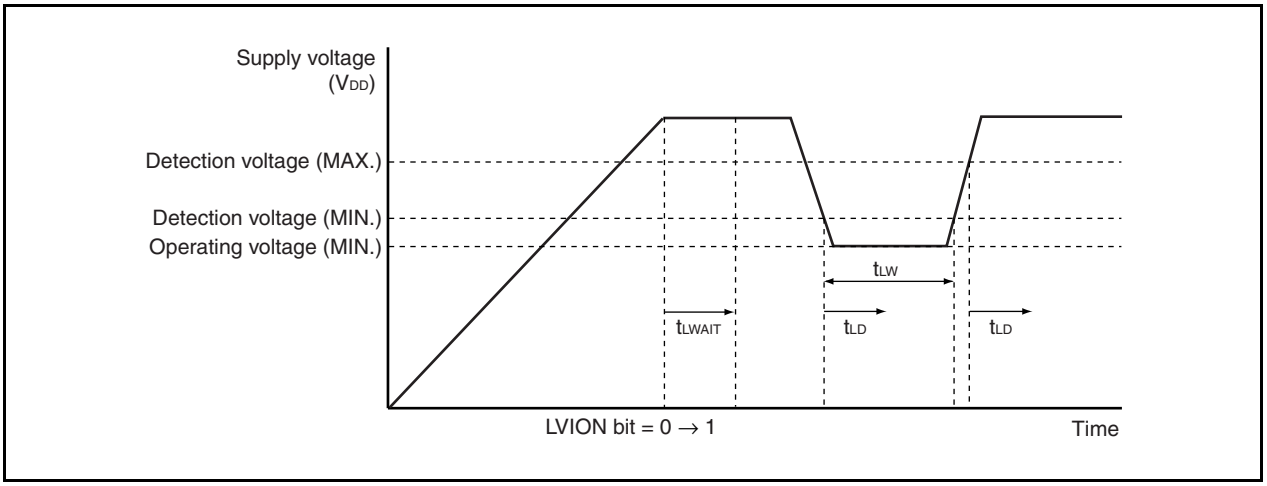
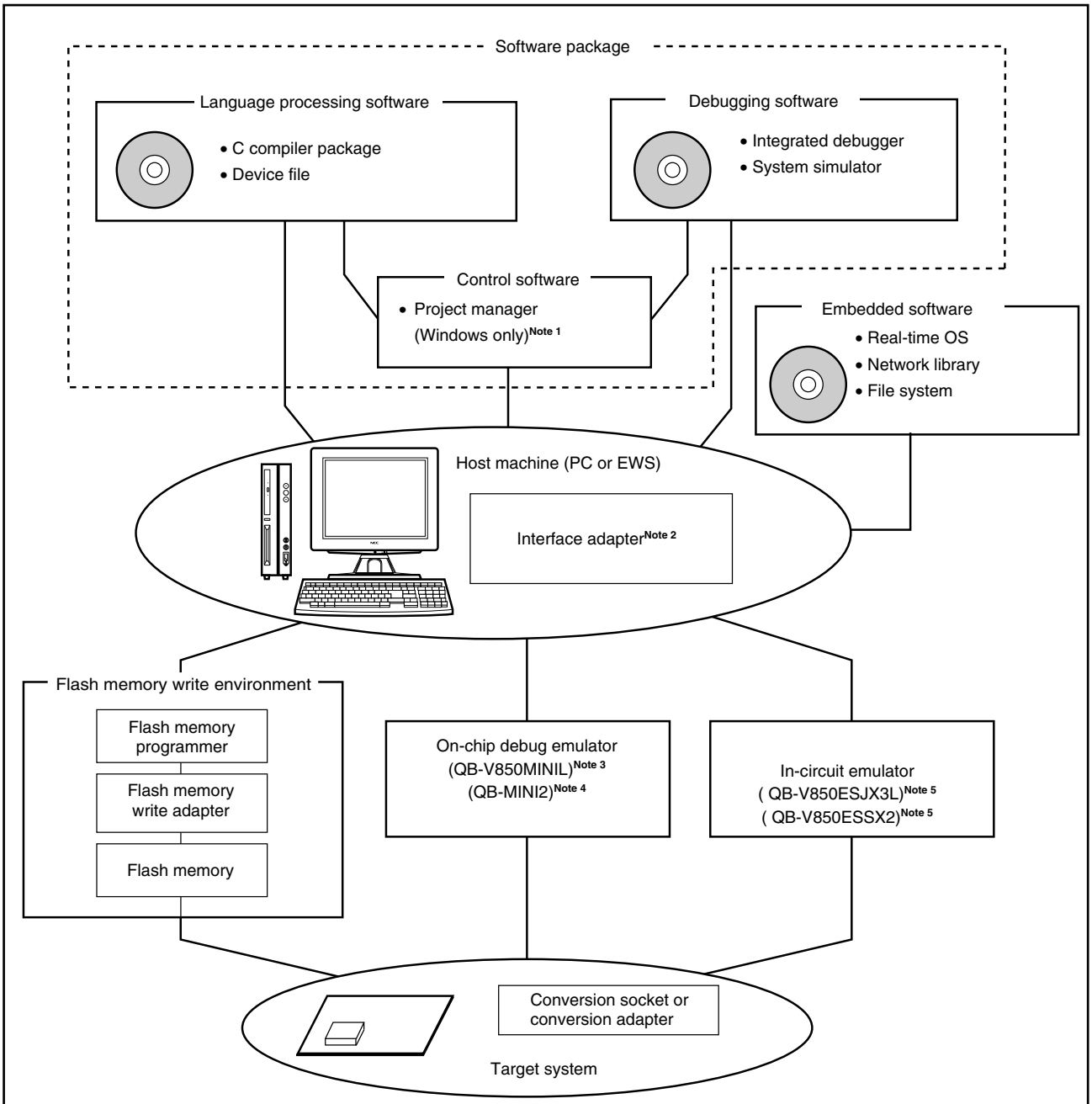


Figure A-1. Development Tool Configuration



Notes 1. Project manager PM+ is included in the C compiler package.
PM+ is only used in Windows.

2. The QB-V850MINIL, QB-MINI2, QB-V850ESJX3L and QB-V850ESSX2 support the USB interface only.

3. The QB-V850MINIL is supplied with the USB interface cable, OCD cable, and self-check board. All other products are optional (integrated debugger is not supplied.)

4. The QB-MINI2 is supplied with USB interface cable, 16-pin target cable, 10-pin target cable, and 78K0-OCD board (integrated debugger is not supplied.) All other products are optional.

5. The QB-V850ESJX3L and QB-V850ESSX2 are supplied with the ID850QB, flash memory programmer (MINICUBE2), power supply unit, and USB interface adapter. All other products are optional.