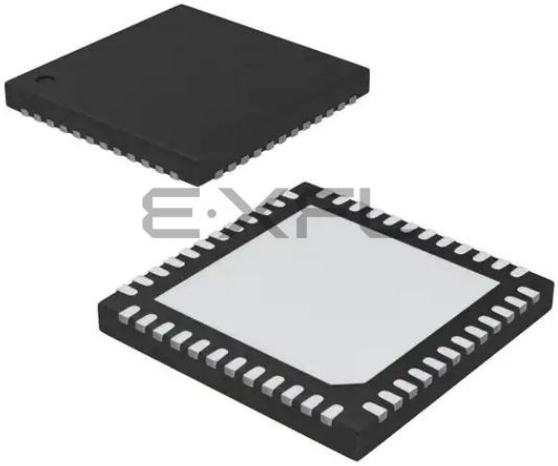
Renesas Electronics America Inc - UPD70F3820GB-GAH-AX Datasheet



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	V850ES
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	CANbus, CSI, EBI/EMI, I²C, UART/USART, USB
Peripherals	DMA, LVD, PWM, WDT
Number of I/O	45
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.85V ~ 3.6V
Data Converters	A/D 10x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-WFQFN Exposed Pad
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd70f3820gb-gah-ax

Table 1-1. V850ES/JC3-H (40 pin) Product List

Generic Name			V850ES/JC3-H (40 pin)				
Part Number			μPD70F3809	μPD70F3810	μPD70F3811	μPD70F3812	μPD70F3813
Internal memory	Flash memory		16 KB	32 KB	64 KB	128 KB	256 KB
	RAM		8 KB	16 KB	24 KB	24 KB	24 KB
Memory space	Logical space		64 MB				
General-purpose register			32 bits × 32 registers				
Clock	Main clock		(PLL mode: f_X = 3 to 6 MHz, f_{XX} = 24 to 48 MHz (multiplied by 8) Clock through mode: f_X = 3 to 6 MHz (internal: f_{XX} = 3 to 6 MHz)				
	Subclock		f_{XT} = 32.768 kHz				
	Internal oscillator		f_R = 220 kHz (TYP.)				
	Minimum instruction execution time		20.8 ns (main clock (f_{XX}) = 48 MHz)				
I/O port (5 V tolerant)			I/O: 25 (15)				
Timer	16-bit TAA		4 channels (including 1 channel used only for interval function)				
	16-bit TAB		1 channel (only for interval function)				
	16-bit TMM		4 channels				
	16-bit TMT		1 channel (only for interval function)				
	Watch timer		1 channel (RTC)				
	WDT		1 channel				
Real-time output function			4 bits × 1 channel				
Key interrupt function			4 bits × 1 channel				
10-bit A/D converter			5 channels				
USB controller			USB2.0 Full-speed function: 1channel				
Serial interface	CSIF/UARTC		2 channels				
	CSIF		1 channel				
	CSIF/UARTC/I^2C		1 channel				
DMA controller			4 channels (transfer target: on-chip peripheral I/O, internal RAM)				
Interrupt source	External[Note]		10				
	Internal		52				
Power save function			HALT/IDLE1/IDLE2/STOP/subclock/sub-IDLE mode				
Reset source			$\overline{\text{RESET}}$ pin input, watchdog timer 2 (WDT2), clock monitor (CLM), low-voltage detector (LVI)				
On-chip debugging			MINICUBE®, MINICUBE2 supported				
Operating power supply voltage			2.85 to 3.6 V				
Operating ambient temperature			−40 to +85°C				
Package			40-pin plastic WQFN (6 × 6 mm)				

Note The figures in parentheses indicate the number of external interrupts that can release STOP mode.

4.4 Port Register Settings When Alternate Function Is Used

Table 4-17 shows the port register settings when each port is used for an alternate function. When using a port pin as an alternate-function pin, refer to the description of each pin.

(4) PLL lockup time specification register (PLLS)

The PLLS register is an 8-bit register used to select the PLL lockup time when the PLLCTL.PLLON bit is changed from 0 to 1.

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

Reset sets this register to 03H.

After reset: 03H R/W Address: FFFFF6C1H

	7	6	5	4	3	2	1	0
PLLS	0	0	0	0	0	0	PLLS1	PLLS0

PLLS1	PLLS0	Selection of PLL lockup time
0	0	$2^{10}/f_X$
0	1	$2^{11}/f_X$
1	0	$2^{12}/f_X$
1	1	$2^{13}/f_X$ (default value)

Cautions 1. Set so that the lockup time is 800 μs or longer.

2. Do not change the PLLS register setting during the lockup period.

5.5.3 Usage

(1) When PLL is used

- After the reset signal has been released, the PLL operates (PLLCTL.PLLON bit = 1), but because the default mode is the clock-through mode (PLLCTL.SELPLL bit = 0), select the PLL mode (SELPLL bit = 1).
- To enable PLL operation, first set the PLLON bit to 1, and then set the SELPLL bit to 1 after the LOCKR.LOCK bit = 0. To stop the PLL, first select the clock-through mode (SELPLL bit = 0), wait for 8 clocks or more, and then stop the PLL (PLLON bit = 0).
- The PLL stops during transition to the IDLE2 or STOP mode regardless of the setting and is restored from the IDLE2 or STOP mode to the status before transition. The time required for restoration is as follows.

(a) When transiting to the IDLE2 or STOP mode from the clock through mode

- STOP mode: Set the OSTS register so that the oscillation stabilization time is 1 ms (min.) or longer.
- IDLE2 mode: Set the OSTS register so that the setup time is 350 μs (min.) or longer.

(b) When transiting to the IDLE 2 or STOP mode while remaining in the PLL operation mode

- STOP mode: Set the OSTS register so that the oscillation stabilization time is 1 ms (min.) or longer.
- IDLE2 mode: Set the OSTS register so that the setup time is 800 μs (min.) or longer.

When transiting to the IDLE1 mode, the PLL does not stop. Stop the PLL if necessary.

(2) When PLL is not used

- The clock-through mode (SELPLL bit = 0) is selected after the reset signal has been released, but the PLL is operating (PLLON bit = 1) and must therefore be stopped (PLLON bit = 0).

TAB1 waits for a trigger when the TAB1CE bit is set to 1. When the trigger is generated, the 16-bit counter is cleared from FFFFH to 0000H, starts counting at the same time, and outputs a PWM waveform from the TOAB1k pin. If the trigger is generated again while the counter is operating, the counter is cleared to 0000H and restarted. (The output of the TOAB10 pin is inverted. The TOAB1k pin outputs a high level regardless of the status (high/low) when a trigger is generated.)

The active level width, cycle, and duty factor of the PWM waveform can be calculated as follows.

Active level width = (Set value of TAB1CCRk register) × Count clock cycle
Cycle = (Set value of TAB1CCR0 register + 1) × Count clock cycle
Duty factor = (Set value of TAB1CCRk register)/(Set value of TAB1CCR0 register + 1)

The compare match request signal (INTTAB1CC0) is generated when the 16-bit counter counts up next time after its count value matches the value of the CCR0 buffer register, and the 16-bit counter is cleared to 0000H. The compare match interrupt request signal (INTTAB1CCk) is generated when the count value of the 16-bit counter matches the value of the CCRk buffer register.

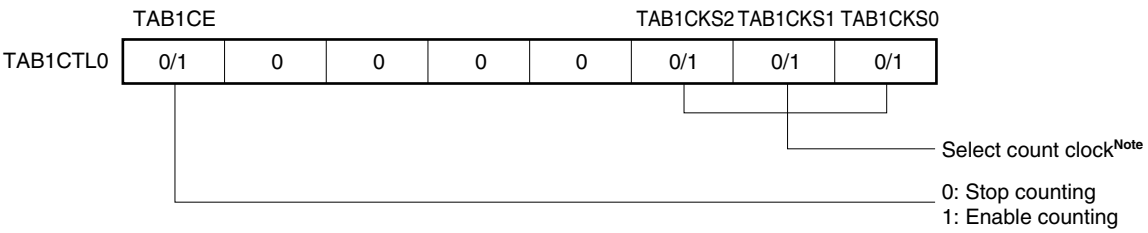
The value set to the TAB1CCRm register is transferred to the CCRm buffer register when the count value of the 16-bit counter matches the value of the CCR0 buffer register and the 16-bit counter is cleared to 0000H.

The valid edge of the external trigger input signal or setting the software trigger (TAB1CTL1.TAB1EST bit) to 1 is used as the trigger.

Remark k = 1 to 3,
m = 0 to 3,

Figure 7-18. Setting of Registers in External Trigger Pulse Output Mode (1/3)

(a) TAB1 control register 0 (TAB1CTL0)



Note The setting is invalid when the TAB1CTL1.TAB1EEE bit = 1.

To transfer data from the TAB1CCRm register to the CCRm buffer register, the TAB1CCR1 register must be written.

To change both the cycle and active level of the PWM waveform at this time, first set the cycle to the TAB1CCR0 register, set the active level width to the TAB1CCR2 and TAB1CCR3 registers, and then set the active level width to the TAB1CCR1 register.

To change only the cycle of the PWM waveform, first set the cycle to the TAB1CCR0 register, and then write the same value to the TAB1CCR1 register.

To change only the active level width (duty factor) of the PWM wave, first set the active level to the TAB1CCR2 and TAB1CCR3 registers, and then set the active level to the TAB1CCR1 register.

To change only the active level width (duty factor) of the PWM waveform output by the TOAB11 pin, only the TAB1CCR1 register has to be set.

To change only the active level width (duty factor) of the PWM waveform output by the TOAB12 and TOAB13 pins, first set the active level width to the TAB1CCR2 and TAB1CCR3 registers, and then write the same value to the TAB1CCR1 register.

After the TAB1CCR1 register is written, the value written to the TAB1CCRm register is transferred to the CCRm buffer register in synchronization with the timing of clearing the 16-bit counter, and is used as the value to be compared with the value of the 16-bit counter.

To write the TAB1CCR0 to TAB1CCR3 registers again after writing the TAB1CCR1 register once, do so after the INTTAB1CC0 signal is generated. Otherwise, the value of the CCRm buffer register may become undefined because the timing of transferring data from the TAB1CCRm register to the CCRm buffer register conflicts with writing the TAB1CCRm register.

Remark m = 0 to 3,

7.5.7 Pulse width measurement mode (TAB1MD2 to TAB1MD0 bits = 110)

In the pulse width measurement mode, TAB1 starts counting when the TAB1CTL0.TAB1CE bit is set to 1. Each time the valid edge input to the TIAB1m pin has been detected, the count value of the 16-bit counter is stored in the TAB1CCRm register, and the 16-bit counter is cleared to 0000H.

The interval of the valid edge can be measured by reading the TAB1CCRm register after a capture interrupt request signal (INTTAB1CCm) occurs.

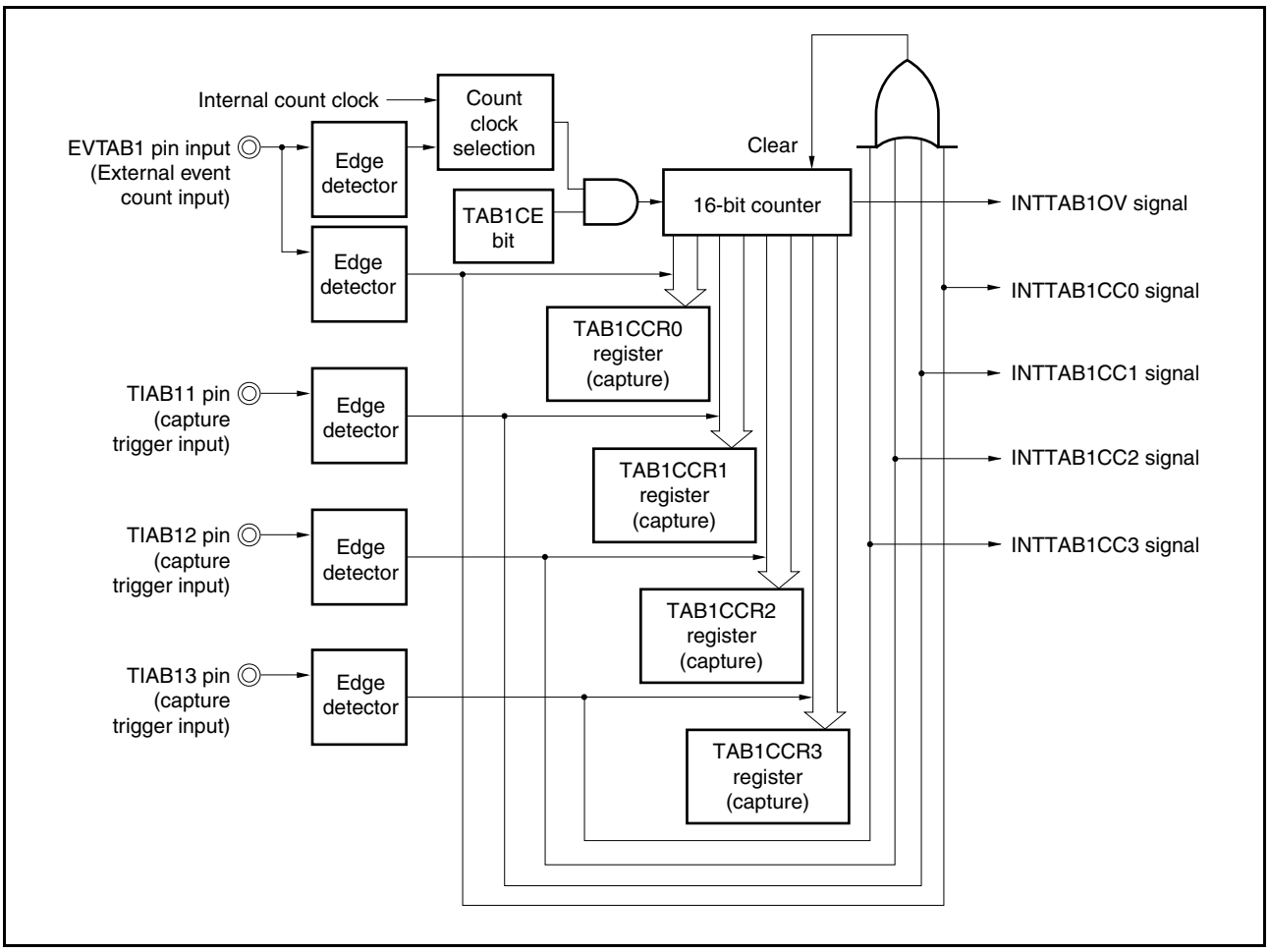
Select one of the TIAB10 to TIAB13 pins as the capture trigger input pin. Specify “No edge detected” for the unused pins by using the TAB1IOC1 register.

When an external clock is used as the count clock, measure the pulse width of the TIAB0k pin because the external clock is fixed to the TIAB00 pin. At this time, clear the TAB0IOC1.TAB0IS1 and TAB0IOC1.TAB0IS0 bits to 00 (capture trigger input (TIAB00 pin): No edge detected).

For TAB1, the external clock is input from the EVTAB1 pin, and the pulse width can be measured by using the TIAB10 to TIAB13 pins.

Remark m = 0 to 3,
k = 1 to 3

Figure 7-34. Configuration in Pulse Width Measurement Mode



(8) TMT0 option register 0 (TT0OPT0)

The TT0OPT0 register is an 8-bit register that sets the capture/compare operation and detects overflows.

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: FFFFF607H

	7	6	5	4	3	2	1	<0>
TT0OPT0	0	0	TT0CCS1	TT0CCS0	0	0	0	TT0OVF

TT0CCS1	TT0CCR1 register capture/compare selection
0	Selected as compare register
1	Selected as capture register (cleared by the TT0CTL0.TT0CE bit = 0)
The TT0CCS1 bit setting is valid only in the free-running timer mode.	

TT0CCS0	TT0CCR0 register capture/compare selection
0	Selected as compare register
1	Selected as capture register (cleared by the TT0CTL0.TT0CE bit = 0)
The TT0CCS0 bit setting is valid only in the free-running timer mode.	

TT0OVF	TMT0 overflow detection flag
Set (1)	Overflow occurred
Reset (0)	0 written to TT0OVF bit or TT0CTL0.TT0CE bit = 0

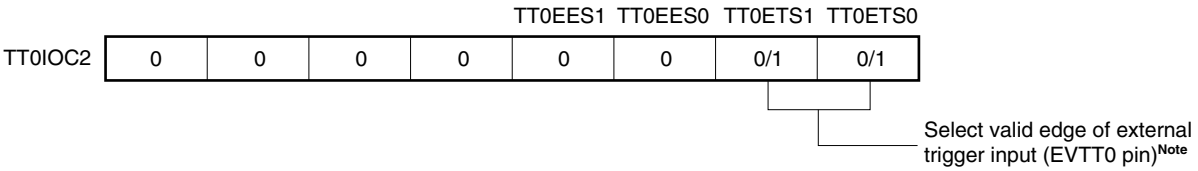
- The TT0OVF bit is set to 1 when the 16-bit counter value overflows from FFFFH to 0000H in the free-running timer mode or the pulse width measurement mode.
- An overflow interrupt request signal (INTTT0OV) is generated when the TT0OVF bit is set to 1. The INTTT0OV signal is not generated in modes other than the free-running timer mode and the pulse width measurement mode.
- The TT0OVF bit is not cleared to 0 even when the TT0OVF bit or the TT0OPT0 register are read when the TT0OVF bit = 1.
- Before clearing the TT0OVF bit to 0 after generation of the INTTT0OV signal, be sure to confirm (by reading) that the TT0OVF bit is set to 1.
- The TT0OVF bit can be both read and written, but the TT0OVF bit cannot be set to 1 by software. Writing 1 has no effect on the operation of TMT0.

Cautions 1. Rewrite the TT0CCS1 and TT0CCS0 bits when the TT0CE bit = 0. (The same value can be written when the TT0CE bit = 1.) If rewriting was mistakenly performed, clear the TT0CE bit to 0 and then set these bits again.

2. Be sure to set bits 1 to 3, 6, and 7 to "0".

Figure 8-23. Setting of Registers in External Trigger Pulse Output Mode (2/2)

(d) TMT0 I/O control register 2 (TT0IOC2)



Note Set the valid edge selection of the unused alternate external input signals to "No edge detection".

(e) TMT0 counter read buffer register (TT0CNT)

The value of the 16-bit counter can be read by reading the TT0CNT register.

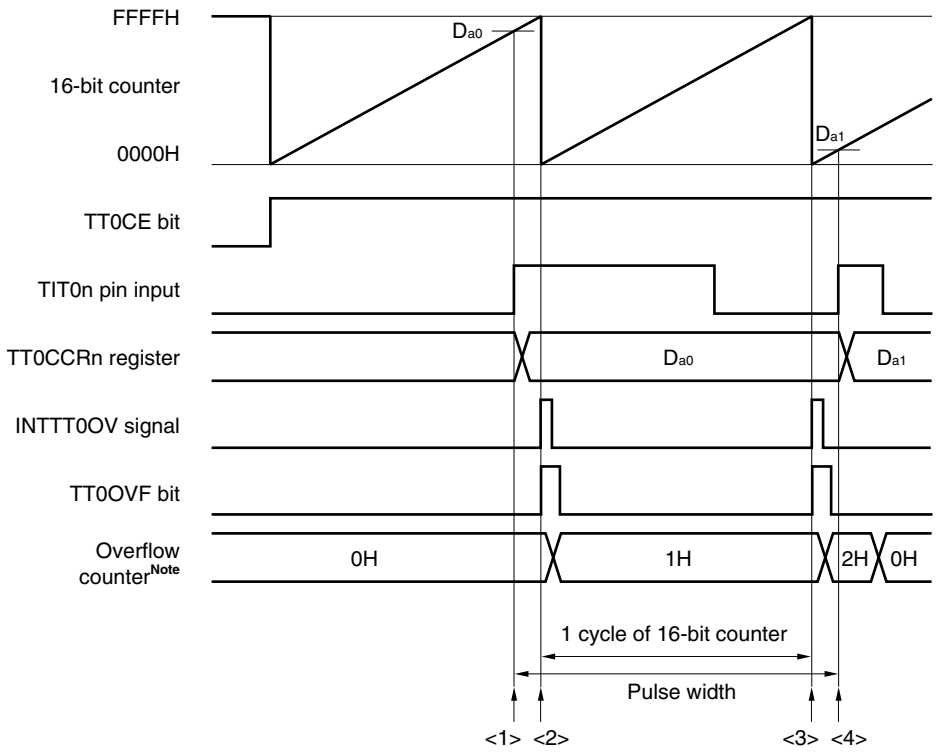
(f) TMT0 capture/compare registers 0 and 1 (TT0CCR0 and TT0CCR1)

If D_0 is set to the TT0CCR0 register and D_1 to the TT0CCR1 register, the cycle and active level of the PWM waveform are as follows.

Cycle = $(D_0 + 1) \times$ Count clock cycle
Active level width = $D_1 \times$ Count clock cycle

Remark TMT0 control register 2 (TT0CTL2), TMT0 I/O control register 1 (TT0IOC1), TMT0 I/O control register 3 (TT0IOC3), TMT0 option register 0 (TT0OPT0), TMT0 option register 1 (TT0OPT1), and TMT0 counter write register (TT0TCW) are not used in the external trigger pulse output mode.

Example when capture trigger interval is long



Note The overflow counter is set arbitrarily by software on the internal RAM.

<1> Read the TT0CCRn register (setting of the default value of the TIT0n pin input).

<2> An overflow occurs. Increment the overflow counter and clear the overflow flag to 0 in the overflow interrupt servicing.

<3> An overflow occurs a second time. Increment the overflow counter and clear the overflow flag to 0 in the overflow interrupt servicing.

<4> Read the TT0CCRn register.

Read the overflow counter.

→ When the overflow counter is "N", the pulse width can be calculated by (N × 10000H + D_{a1} – D_{a0}).

In this example, the pulse width is (20000H + D_{a1} – D_{a0}) because an overflow occurs twice.

Clear the overflow counter (0H).

Remark n = 0, 1

(e) Clearing overflow flag

The overflow flag can be cleared to 0 by clearing the TT0OVF bit to 0 with the CLR instruction after reading the TT0OVF bit when it is 1 and by writing 8-bit data (bit 0 is 0) to the TT0OPT0 register after reading the TT0OVF bit when it is 1.

(1) 16-bit counter

This is a 16-bit counter that counts the internal clock.

The 16-bit counter cannot be read or written.

(2) TMMn compare register 0 (TMnCMP0)

The TMnCMP0 register is a 16-bit compare register.

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

Reset sets this register to 0000H.

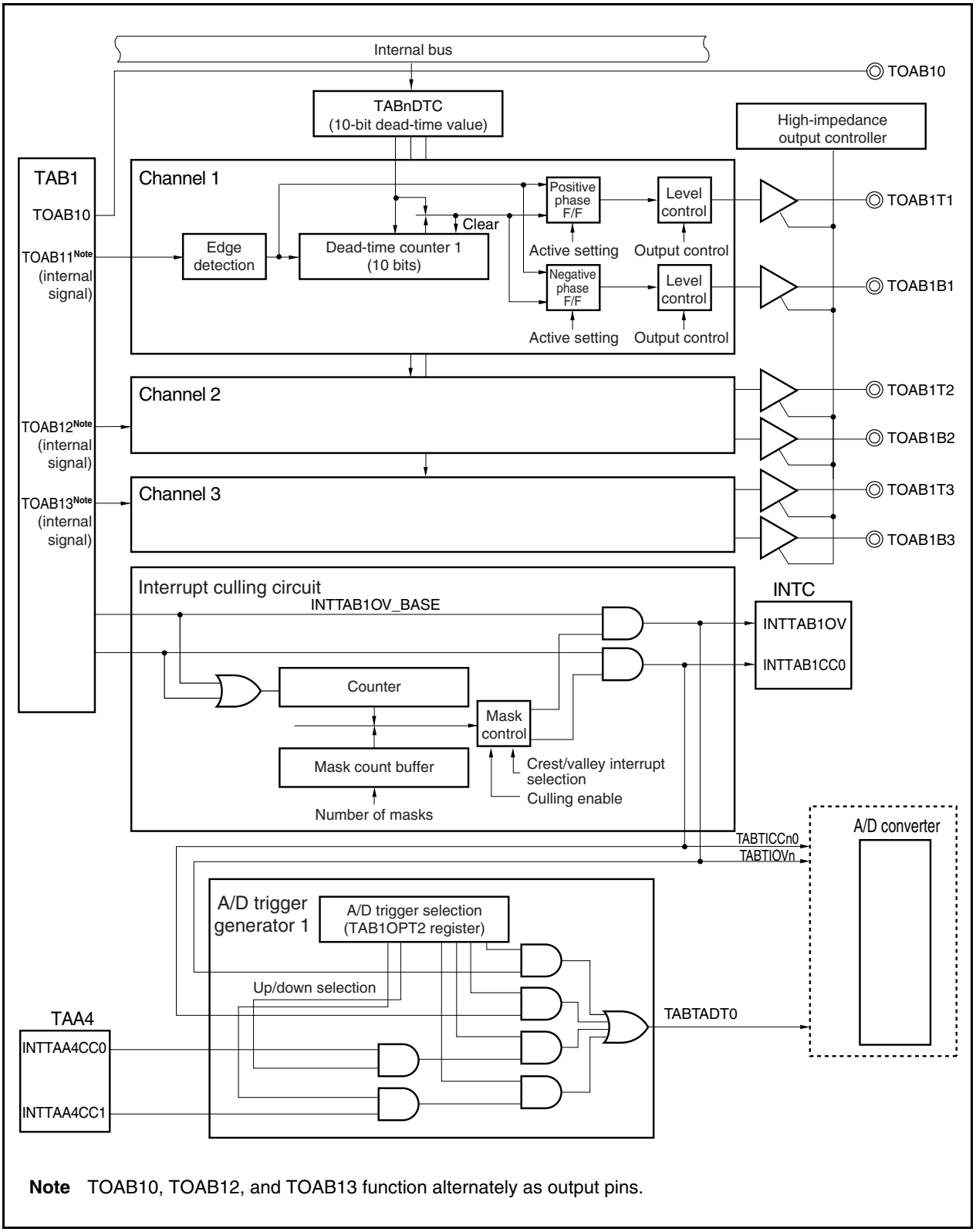
Software can be used to always write the same value to the TMnCMP0 register.

Rewriting the TMnCMP0 register is prohibited when the TMnCTL0.TMnCE bit = 1.

After reset: 0000H R/W Address: TM0CMP0 FFFFFA84H, TM1CMP0 FFFFFA94H, TM2CMP0 FFFFFAA4H, TM3CMP0 FFFFFAB4H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMnCMP0 (n = 0 to 3)																

Figure 10-2. TMQ1 Option

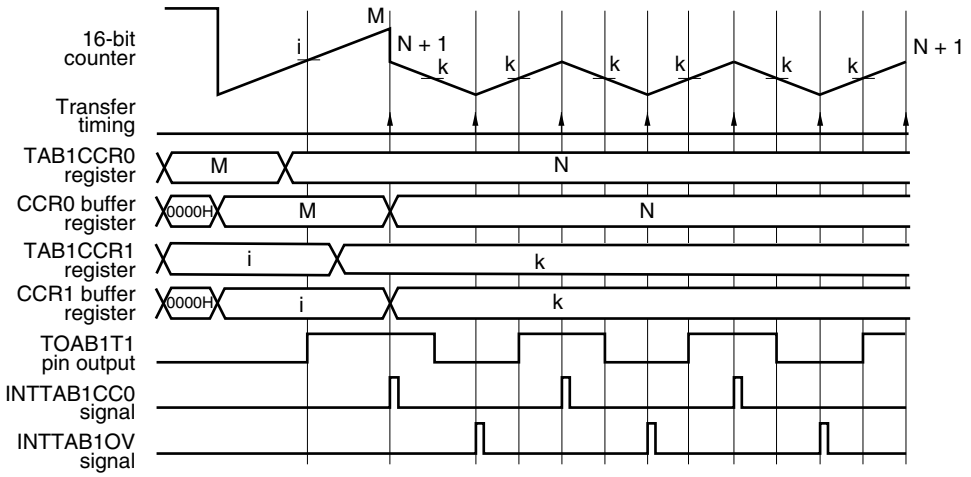


Note TOAB10, TOAB12, and TOAB13 function alternately as output pins.

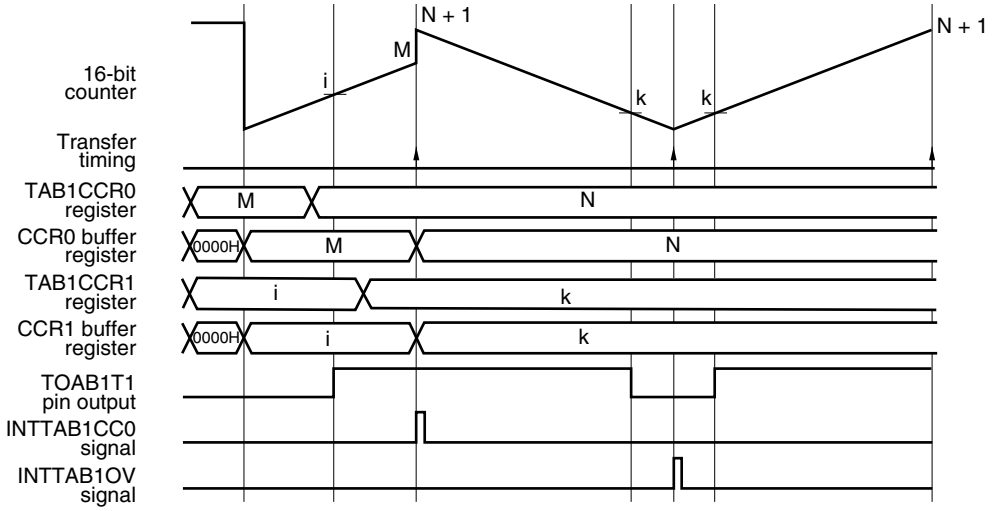
The transfer timing in Figure 10-28 is at the point where the crest timing occurs. While the 16-bit counter is counting down, the cycle changes and an asymmetrical triangular wave is output. Because the cycle changes, rewrite the duty factor (voltage data value).

Figure 10-28. Example of Rewriting TAB1CCR0 Register (During Counting Up)

(a) M > N



(b) M < N



Remarks 1. If transfer (match between the value of the 16-bit counter and the value of the CCR0 buffer register) occurs in the 6-phase PWM output mode, the value of the TAB1CCR0 register plus 1 is loaded to the 16-bit counter. In this way, the expected wave can be output even if the cycle value is changed at the transfer timing of the crest (match between the 16-bit counter value and the TAB1CCR0 register value) timing.

2. M: Value of CCR0 buffer register before rewriting
N: Value of CCR0 buffer register after rewriting

17.6.9 Continuous transfer mode (master mode, transmission/reception mode)

MSB first (CFnCTL0.CFnDIR bit = 0), communication type 1 (CFnCTL1.CFnCKP and CFnCTL1.CFnDAP bits = 00), communication clock (f_{CCLK}) = f_{XX}/2 or f_{XX}/3 (CFnCTL1.CFnCKS2 to CFnCTL1.CFnCKS0 bits = 000), transfer data length = 8 bits (CFnCTL2.CFnCL3 to CFnCTL2.CFnCL0 bits = 0000)

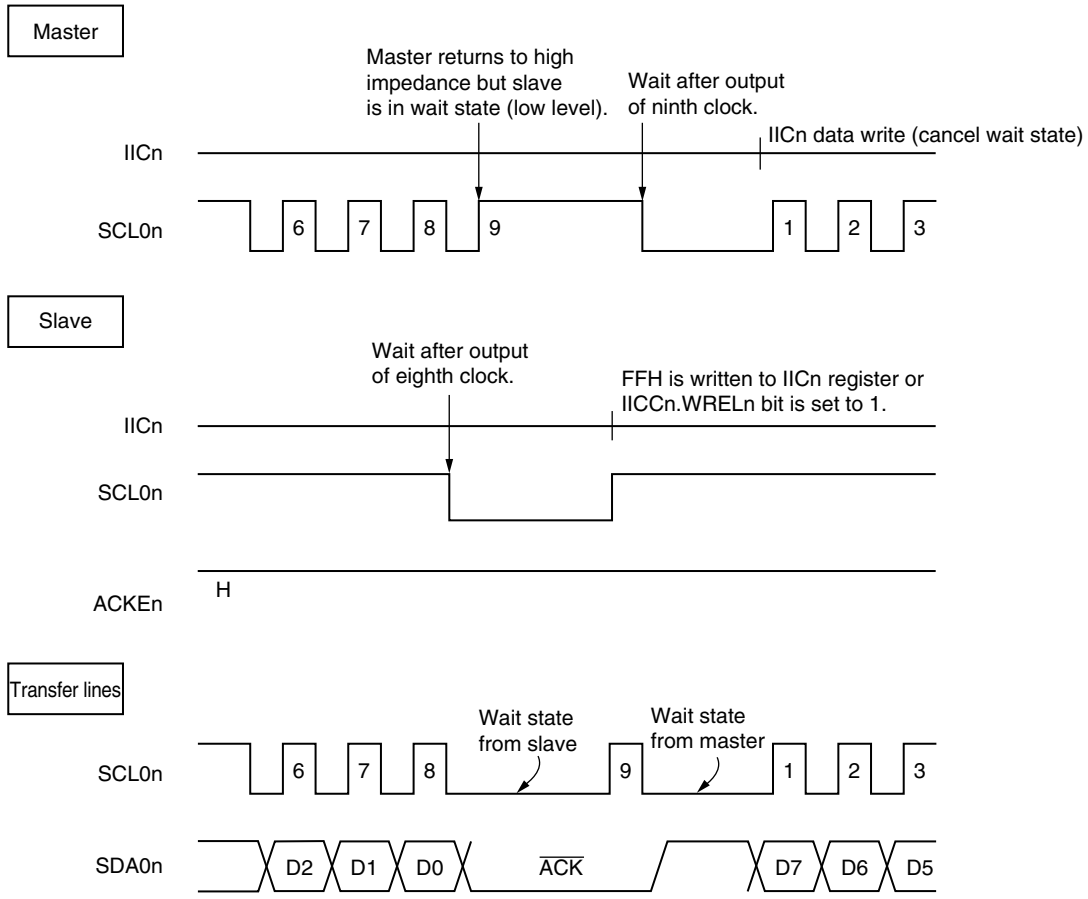
18.6.6 Wait state

A wait state is used to notify the communication partner that a device (master or slave) is preparing to transmit or receive data (i.e., is in a wait state).

Setting the SCL0n pin to low level notifies the communication partner of the wait state. When the wait state has been canceled for both the master and slave devices, the next data transfer can begin.

Figure 18-13. Wait State (1/2)

(a) When master device is in a nine-clock wait state and slave device is in an eight-clock wait state (master: transmission, slave: reception, and IICCn.ACKEn bit = 1)



19.1.3 Configuration

The CAN controller is composed of the following four blocks.

(1) Internal bus interface

This functional block provides an Internal bus interface and means of transmitting and receiving signals between the CAN module and the host CPU.

(2) MCM (Memory Control Module)

This functional block controls access to the CAN protocol layer and to the CAN RAM within the CAN module.

(3) CAN protocol layer

This functional block is involved in the operation of the CAN protocol and its related settings.

(4) CAN RAM

This is the CAN memory functional block, which is used to store message IDs, message data, etc.

Figure 19-1. Block Diagram of CAN Module

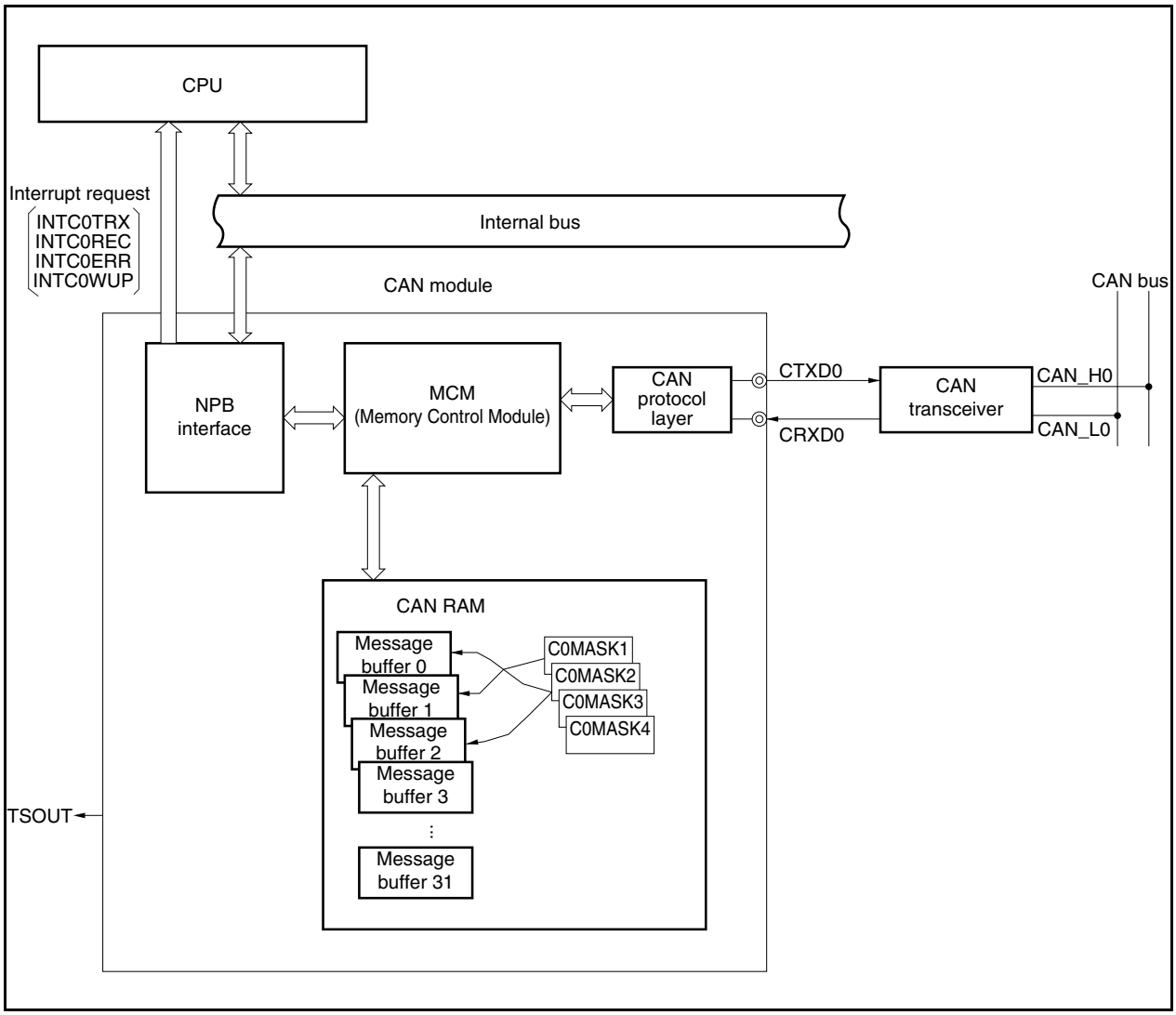
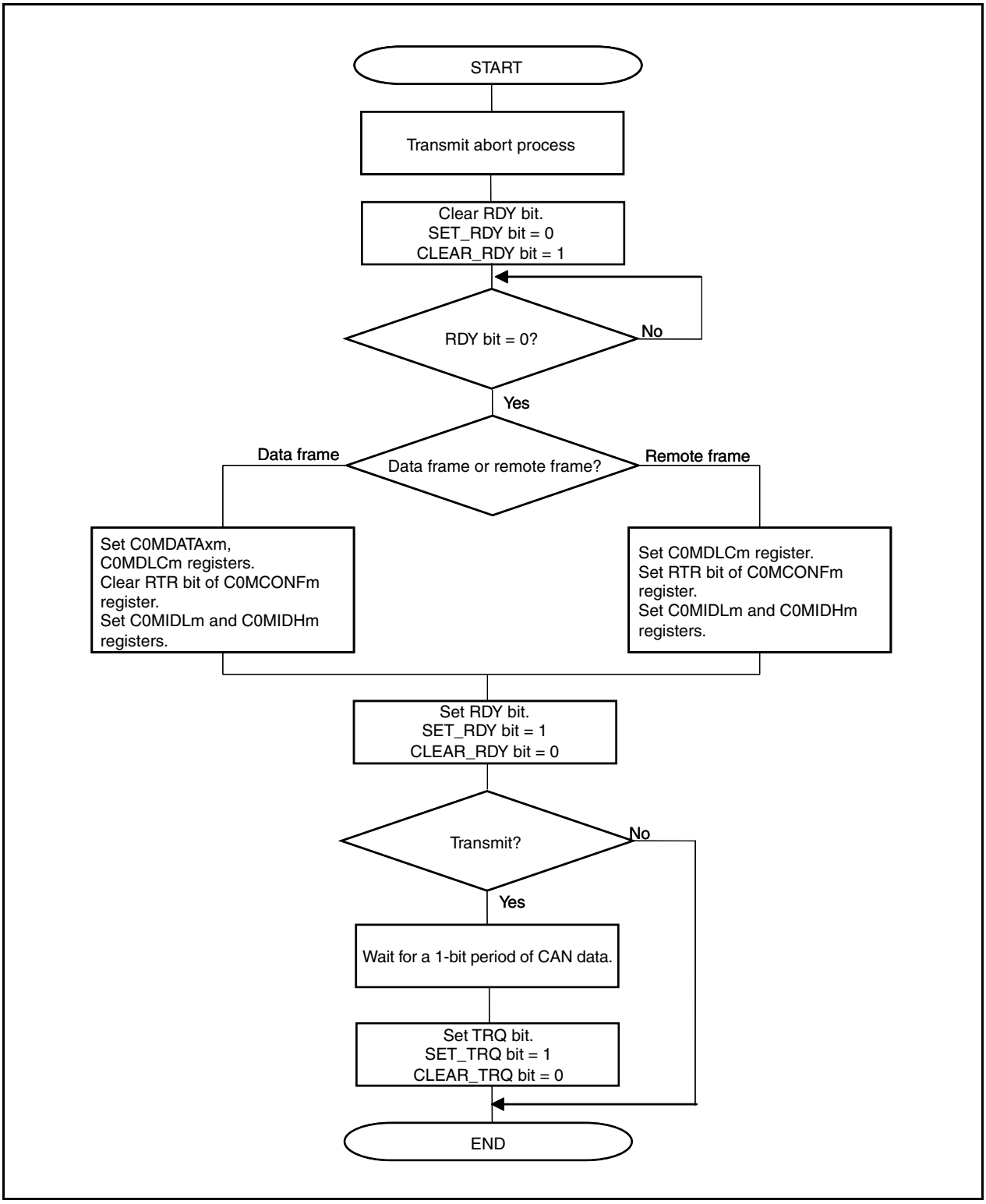


Figure 19-40 shows the processing for a transmit message buffer during transmission (MT2 to MT0 bits of C0MCONFm register = 000B).

Figure 19-40. Message Buffer Redefinition during Transmission



(4/13)

Address	Function Register Name	Symbol	R/W	Manipulatable Bits			Default Value
				1	8	16	
00200226H	UF0 configuration/interface/endpoint descriptor register 48	UF0CIE48	R/W		√		Undefined
00200228H	UF0 configuration/interface/endpoint descriptor register 49	UF0CIE49	R/W		√		Undefined
0020022AH	UF0 configuration/interface/endpoint descriptor register 50	UF0CIE50	R/W		√		Undefined
0020022CH	UF0 configuration/interface/endpoint descriptor register 51	UF0CIE51	R/W		√		Undefined
0020022EH	UF0 configuration/interface/endpoint descriptor register 52	UF0CIE52	R/W		√		Undefined
00200230H	UF0 configuration/interface/endpoint descriptor register 53	UF0CIE53	R/W		√		Undefined
00200232H	UF0 configuration/interface/endpoint descriptor register 54	UF0CIE54	R/W		√		Undefined
00200234H	UF0 configuration/interface/endpoint descriptor register 55	UF0CIE55	R/W		√		Undefined
00200236H	UF0 configuration/interface/endpoint descriptor register 56	UF0CIE56	R/W		√		Undefined
00200238H	UF0 configuration/interface/endpoint descriptor register 57	UF0CIE57	R/W		√		Undefined
0020023AH	UF0 configuration/interface/endpoint descriptor register 58	UF0CIE58	R/W		√		Undefined
0020023CH	UF0 configuration/interface/endpoint descriptor register 59	UF0CIE59	R/W		√		Undefined
0020023EH	UF0 configuration/interface/endpoint descriptor register 60	UF0CIE60	R/W		√		Undefined
00200240H	UF0 configuration/interface/endpoint descriptor register 61	UF0CIE61	R/W		√		Undefined
00200242H	UF0 configuration/interface/endpoint descriptor register 62	UF0CIE62	R/W		√		Undefined
00200244H	UF0 configuration/interface/endpoint descriptor register 63	UF0CIE63	R/W		√		Undefined
00200246H	UF0 configuration/interface/endpoint descriptor register 64	UF0CIE64	R/W		√		Undefined
00200248H	UF0 configuration/interface/endpoint descriptor register 65	UF0CIE65	R/W		√		Undefined
0020024AH	UF0 configuration/interface/endpoint descriptor register 66	UF0CIE66	R/W		√		Undefined
0020024CH	UF0 configuration/interface/endpoint descriptor register 67	UF0CIE67	R/W		√		Undefined
0020024EH	UF0 configuration/interface/endpoint descriptor register 68	UF0CIE68	R/W		√		Undefined
00200250H	UF0 configuration/interface/endpoint descriptor register 69	UF0CIE69	R/W		√		Undefined

(2/2)

Bit position	Bit name	Function
3	DMAED	This bit indicates that the DMA end (TC) signal for Endpoint n (n = 1 to 4, 7) is active. 1: DMA end signal for Endpoint n has been input (interrupt request is generated). 0: DMA end signal for Endpoint n has not been input (default value). When this bit is set to 1, the DMA request signal for Endpoint n becomes inactive. The DMA request signal for Endpoint n does not become active unless FW enables DMA transfer. Use the UF0DMS0 register to confirm on which endpoint the operation is actually performed. However, this bit is not automatically cleared to 0 even if the UF0DMS0 register is read by FW.
2	SETRQ	This bit indicates that the SET_XXXX request to be automatically processed has been received and automatically processed (XXXX = CONFIGURATION or FEATURE). 1: SET_XXXX request to be automatically processed has been received (interrupt request is generated). 0: SET_XXXX request to be automatically processed has not been received (default value). This bit is set to 1 after completion of the status stage. Reference the UF0SET register to identify what is the target of the request. This bit is not automatically cleared to 0 even if the UF0SET register is read by FW. The EPHALT bit is also set to 1 when the SET_FEATURE Endpoint request has been received.
1	CLRRQ	This bit indicates that the CLEAR_FEATURE request has been received and automatically processed. 1: CLEAR_FEATURE request has been received (interrupt request is generated). 0: CLEAR_FEATURE request has not been received (default value). This bit is set to 1 after completion of the status stage. Reference the UF0CLR register to identify what is the target of the request. This bit is not automatically cleared to 0 even if the UF0CLR register is read by FW.
0	EPHALT	This bit indicates that an endpoint has stalled. 1: Endpoint has stalled (interrupt request is generated). 0: Endpoint has not stalled (default value). This bit is also set to 1 when an endpoint has stalled by setting FW. Identify the endpoint that has stalled, by referencing the UF0EPS2 register. This bit is not automatically cleared to 0 even when the CLEAR_FEATURE Endpoint, SET_INTERFACE, or SET_CONFIGURATION request is received. It is not automatically cleared to 0, either, if the next SETUP token is received in case of overrun of Endpoint0. **Caution Even if Halt Feature of Endpoint0 is set and this interrupt request is generated, bit 0 of the UF0EPS2 register is masked and cleared to 0 between when a SET_FEATURE Endpoint0, CLEAR_FEATURE Endpoint0, or GET_STATUS Endpoint0 request, or FW-processed request is received and when a SETUP token other than the above is received.**

(8) UF0 address register (UF0ADRS)

This register stores the device address.

This register is read-only, in 8-bit units.

The device address sent by the SET_ADDRESS request is analyzed and the resultant value is automatically written to this register. If the SET_ADDRESS request is processed by FW, the value of this register is reflected as the device address when the SUCCESS signal is received in the status stage.

Caution Do not execute a write access to this register. If written, the operation is not guaranteed.

	7	6	5	4	3	2	1	0	Address	After reset
UF0ADRS	0	ADRS6	ADRS5	ADRS4	ADRS3	ADRS2	ADRS1	ADRS0	00200180H	00H

Bit position	Bit name	Function
6 to 0	ADRS6 to ADRS0	These bits hold the device address of SIE.

22.4.2 Restore

Restoration from software exception processing is carried out by the RETI instruction.

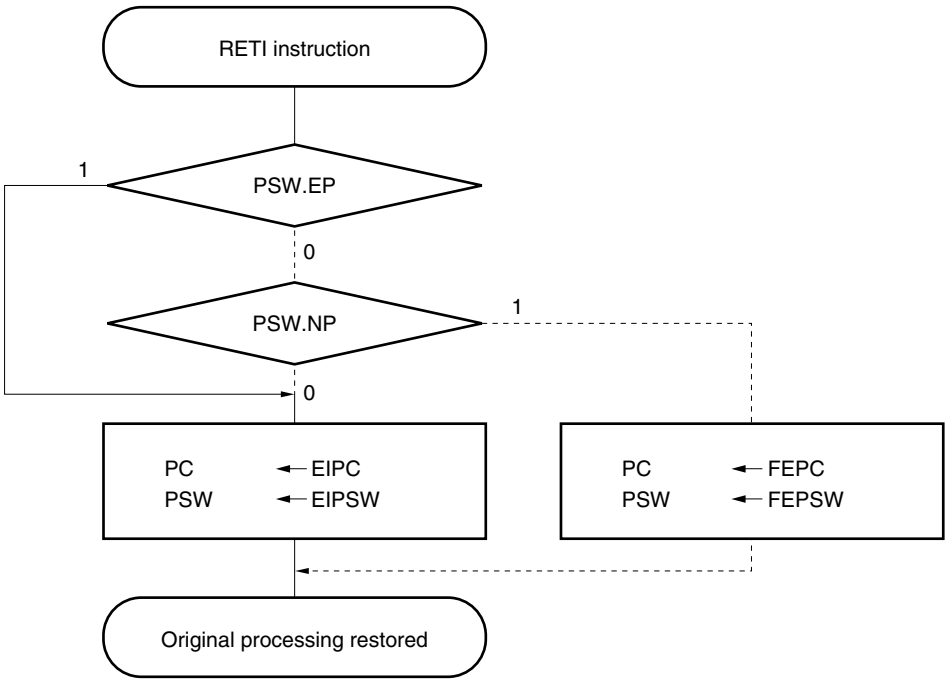
By executing the RETI instruction, the CPU carries out the following processing and shifts control to the restored PC's address.

<1> Loads the restored PC and PSW from EIPC and EIPSW because the PSW.EP bit is 1.

<2> Transfers control to the address of the restored PC and PSW.

The processing of the RETI instruction is shown below.

Figure 22-10. RETI Instruction Processing



Caution When the EP and NP bits are changed by the LDSR instruction during the software exception processing, in order to restore the PC and PSW correctly during recovery by the RETI instruction, it is necessary to set the EP bit back to 1 and the NP bit back to 0 using the LDSR instruction immediately before the RETI instruction.

Remark The solid line shows the CPU processing flow.