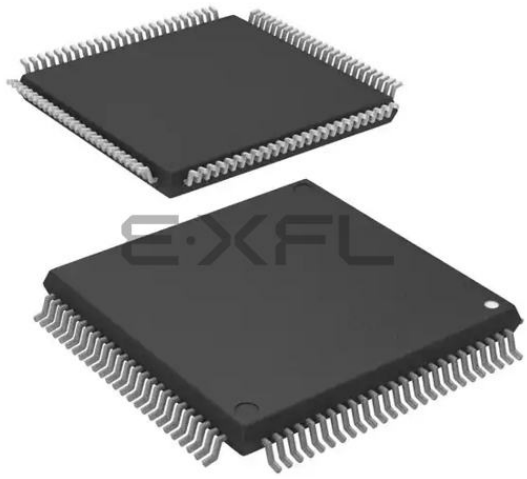
Renesas Electronics America Inc - UPD70F3737GC-UEU-AX Datasheet



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	Active
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	84
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.2V ~ 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/upd70f3737gc-ueu-ax

x Compare operation

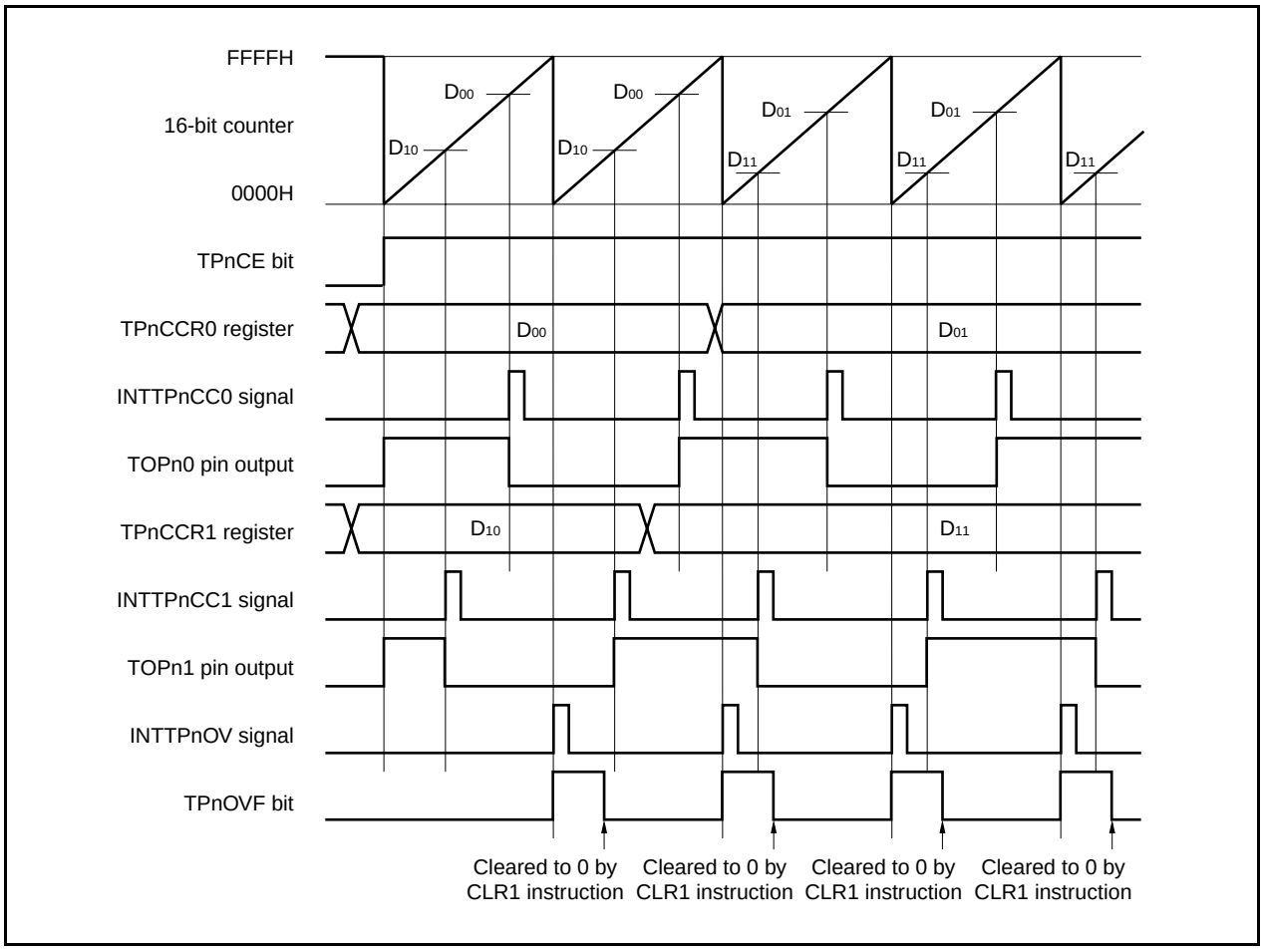
When the TPnCE bit is set to 1, TMPn starts incrementing, and the output signals of the TOPn0 and TOPn1 pins are inverted. When the value of the 16-bit counter later matches the set value of the TPnCCRa register, a compare match interrupt request signal (INTTPnCCa) is generated, and the output signal of the TOPna pin is inverted.

The 16-bit counter continues incrementing in synchronization with the count clock. Once the counter reaches FFFFH, it generates an overflow interrupt request signal (INTTPnOV) at the next clock, is cleared to 0000H, and continues incrementing. At this time, the overflow flag (the TPnOPT0.TPnOVF bit) is also set to 1. The overflow flag must be cleared to 0 by executing a CLR1 software instruction.

The TPnCCRa register can be rewritten while the counter is incrementing. If it is rewritten, the new value is immediately applied, and compared with the count value.

Remark a = 0, 1
n = 0 to 5

Figure 7-53. Basic Timing of Operations in Free-Running Timer Mode (Compare Function)



When the TQ0CE bit is set to 1, the 16-bit counter is cleared from FFFFH to 0000H, starts incrementing, and outputs a PWM waveform from the TOQ0k pin.

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

Active level width = (Set value of TQ0CCRk register) u Count clock cycle
Cycle = (Set value of TQ0CCR0 register + 1) u Count clock cycle
Duty factor = (Set value of TQ0CCRk register)/(Set value of TQ0CCR0 register + 1)

The PWM waveform can be changed by rewriting the TQ0CCRm register while the counter is incrementing. The newly written value is reflected when the value of the 16-bit counter matches the value of the CCR0 buffer register and the 16-bit counter is cleared to 0000H.

The INTTQ0CC0 compare match interrupt request signal is generated when the 16-bit counter increments next time after its count value matches the value of the CCR0 buffer register, and the 16-bit counter is cleared to 0000H. The INTTQ0CCk compare match interrupt request signal is generated when the value of the 16-bit counter matches the value of the CCRk buffer register.

Remark k = 1 to 3
m = 0 to 3

Figure 8-46. Register Settings in PWM Output Mode (1/3)

(a) TMQ0 control register 0 (TQ0CTL0)

	TQ0CE					TQ0CKS2	TQ0CKS1	TQ0CKS0
TQ0CTL0	0/1	0	0	0	0	0/1	0/1	0/1

These bits select the count clock[Note].

0: Stop counting.
1: Enable counting.

(b) TMQ0 control register 1 (TQ0CTL1)

		TQ0EST	TQ0EEE			TQ0MD2	TQ0MD1	TQ0MD0
TQ0CTL1	0	0	0/1	0	0	1	0	0

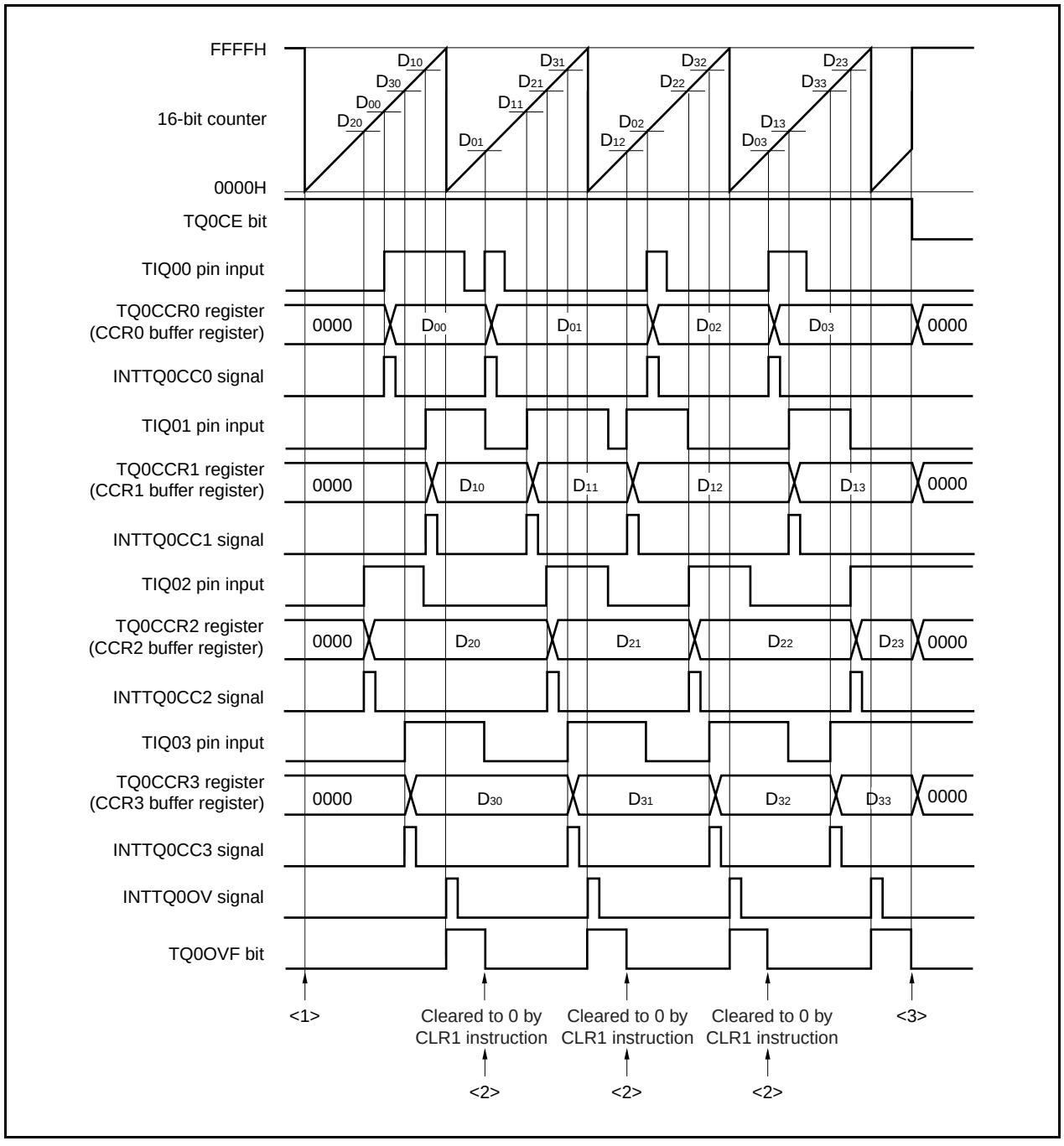
1, 0, 0:
PWM output mode

0: Operate on count clock selected by using TQ0CKS0 to TQ0CKS2 bits.
1: Increment based on external event count input signal.

Note The setting of these bits is invalid when the TQ0CTL1.TQ0EEE bit is 1.

(b) Using a capture/compare register as a capture register

Figure 8-57. Timing and Processing of Operations in Free-Running Timer Mode (Capture Function) (1/2)



CHAPTER 13 REAL-TIME OUTPUT FUNCTION (RTO)

13.1 Function

The real-time output function transfers preset data to the real-time output buffer registers (RTBL0 and RTBH0), and then transfers this data by hardware to an external device via the output latches, upon occurrence of a timer interrupt. The pins through which the data is output to an external device constitute a port called the real-time output function (RTO).

Because RTO can output signals without jitter, it is suitable for controlling a stepper motor.

In the V850ES/JG3-L, one 6-bit real-time output port channel is provided.

The real-time output port can be set to the port mode or real-time output port mode in 1-bit units.

(2) A/D converter mode register 1 (ADA0M1)

The ADA0M1 register is an 8-bit register that specifies the conversion time.

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: FFFFF201H

	7	6	5	4	3	2	1	0
ADA0M1	ADA0HS1	0	0	0	0	ADA0FR2	ADA0FR1	ADA0FR0

ADA0HS1	Normal conversion mode/high-speed mode (A/D conversion time) selection
0	Normal conversion mode
1	High-speed conversion mode

Cautions 1. Changing the ADA0M1 register is prohibited while A/D conversion is enabled (ADA0M0.ADA0CE bit = 1).

2. When selecting the external trigger mode/timer trigger mode (ADA0M0.ADA0TMD bit = 1), set the high-speed conversion mode (ADA0HS1 bit = 1). Do not input a trigger during the stabilization time that is inserted once after A/D conversion is enabled (ADA0CE bit = 1).

3. Be sure to clear bits 6 to 3 to "0".

Remark For A/D conversion time setting examples, see **Tables 14-2** and **14-3**.

The relationship between the analog voltage input to the analog input pins (ANI0 to ANI11) and the A/D conversion result (ADA0CRn register) is as follows.

$$SAR = INT\left(\frac{V_{IN}}{AV_{REF0}} \times 1{,}024 + 0.5\right)$$

$$ADA0CR^{Note} = SAR \times 64$$

Or,

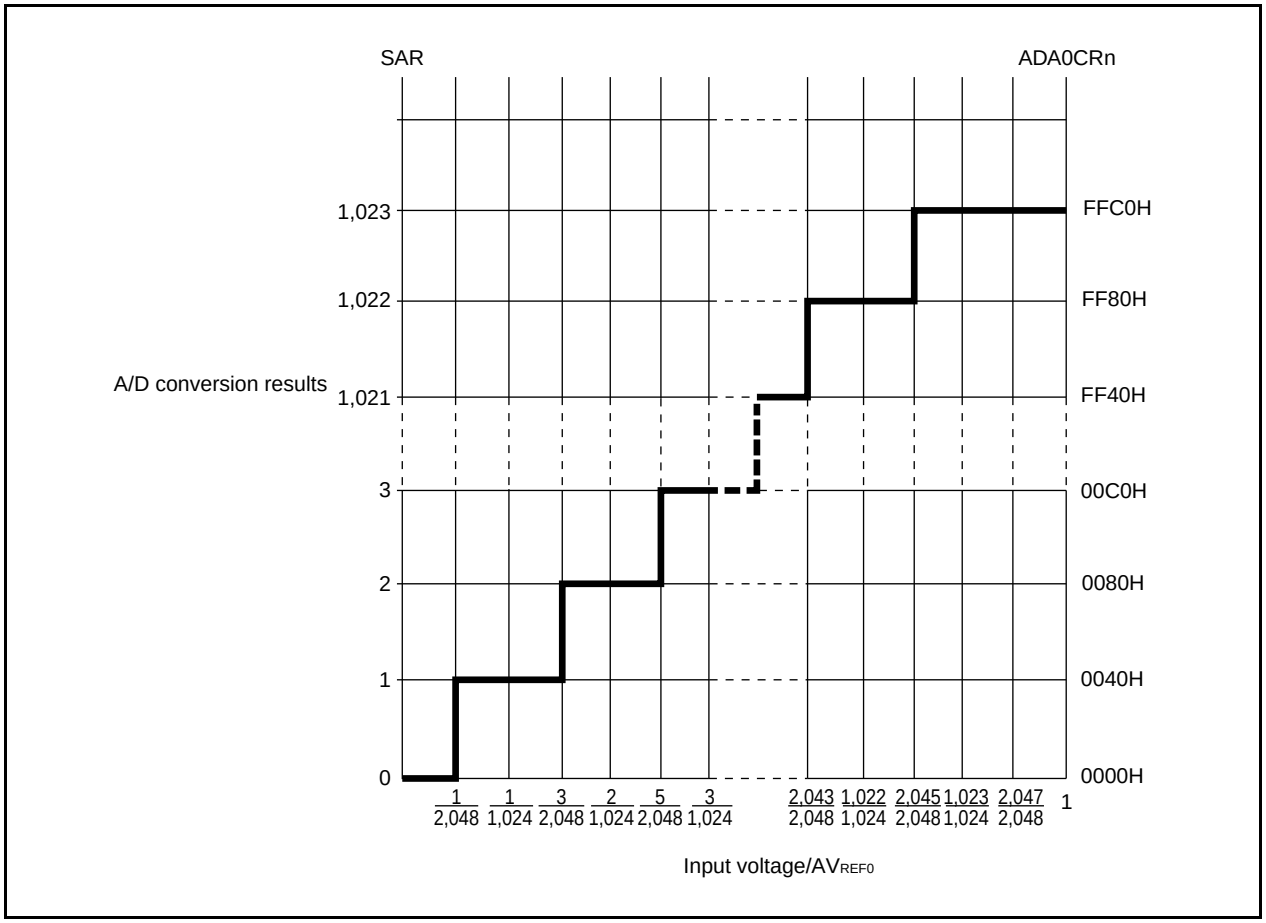
$$(SAR - 0.5) \times \frac{AV_{REF0}}{1{,}024} \le V_{IN} < (SAR + 0.5) \times \frac{AV_{REF0}}{1{,}024}$$

INT( ): Function that returns the integer of the value in ( )
V_{IN}: Analog input voltage
AV_{REF0}: AV_{REF0} pin voltage
ADA0CR: Value of ADA0CRn register

Note The lower 6 bits of the ADA0CRn register are fixed to 0.

The following shows the relationship between the analog input voltage and the A/D conversion results.

Figure 14-2. Relationship Between Analog Input Voltage and A/D Conversion Results



(8) Reading ADA0CRn register

When the ADA0M0 to ADA0M2, ADA0S, ADA0PFM, or ADA0PFT register is written, the contents of the ADA0CRn register may be undefined. Read the conversion result after completion of conversion and before writing to the ADA0M0 to ADA0M2, ADA0S, ADA0PFM, or ADA0PFT register. Also, when an external/timer trigger is acknowledged, the contents of the ADA0CRn register may be undefined. Read the conversion result after completion of conversion and before the next external/timer trigger is acknowledged. The correct conversion result may not be read at a timing different from the above.

(9) Standby mode

Because the A/D converter stops operating in the STOP mode, power consumption can be reduced, but the conversion results will be invalid. Operations are resumed after the STOP mode is released, but the A/D conversion results after the STOP mode is released are invalid. When using the A/D converter after the STOP mode is released, before setting the STOP mode or releasing the STOP mode, clear the ADA0M0.ADA0CE bit to 0 then set the ADA0CE bit to 1 after releasing the STOP mode.

In the IDLE1, IDLE2, or subclock operation mode, operation continues. In the IDLE1 and IDLE2 modes, since the analog input voltage value cannot be retained, the A/D conversion results after the IDLE1 and IDLE2 modes are released are invalid. The results of conversions before the IDLE1 and IDLE2 modes were set are valid. To lower the power consumption, therefore, clear the ADA0M0.ADA0CE bit to 0.

(10) High-speed conversion mode

In the high-speed conversion mode, rewriting the ADA0M0, ADA0M2, ADA0S, ADA0PFM, and ADA0PFT registers and inputting a trigger during the stabilization time are prohibited.

(11) A/D conversion time

A/D conversion time is the total time of stabilization time, conversion time, wait time, and trigger response time (for details of these times, refer to **Table 14-2 Conversion Time Selection in Normal Conversion Mode (ADA0HS1 Bit = 0)** and **Table 14-3 Conversion Time Selection in High-Speed Conversion Mode (ADA0HS1 Bit = 1)**).

During A/D conversion in the normal conversion mode, if the ADA0M0, ADA0M2, ADA0S, ADA0PFM, and ADA0PFT registers are written or a trigger is input, reconversion is carried out. However, if the stabilization time end timing conflicts with writing to these registers, or if the stabilization time end timing conflicts with the trigger input, the stabilization time of 64 clocks is reinserted.

If a conflict occurs again with the reinserted stabilization time end timing, the stabilization time is reinserted. Therefore do not set the trigger input interval and control register write interval to 64 clocks or lower.

(12) Variation of A/D conversion results

The results of the A/D conversion may vary depending on the fluctuation of the supply voltage, or may be affected by noise. To reduce the variation, take countermeasures in the program such as averaging the A/D conversion results.

15.2 Configuration

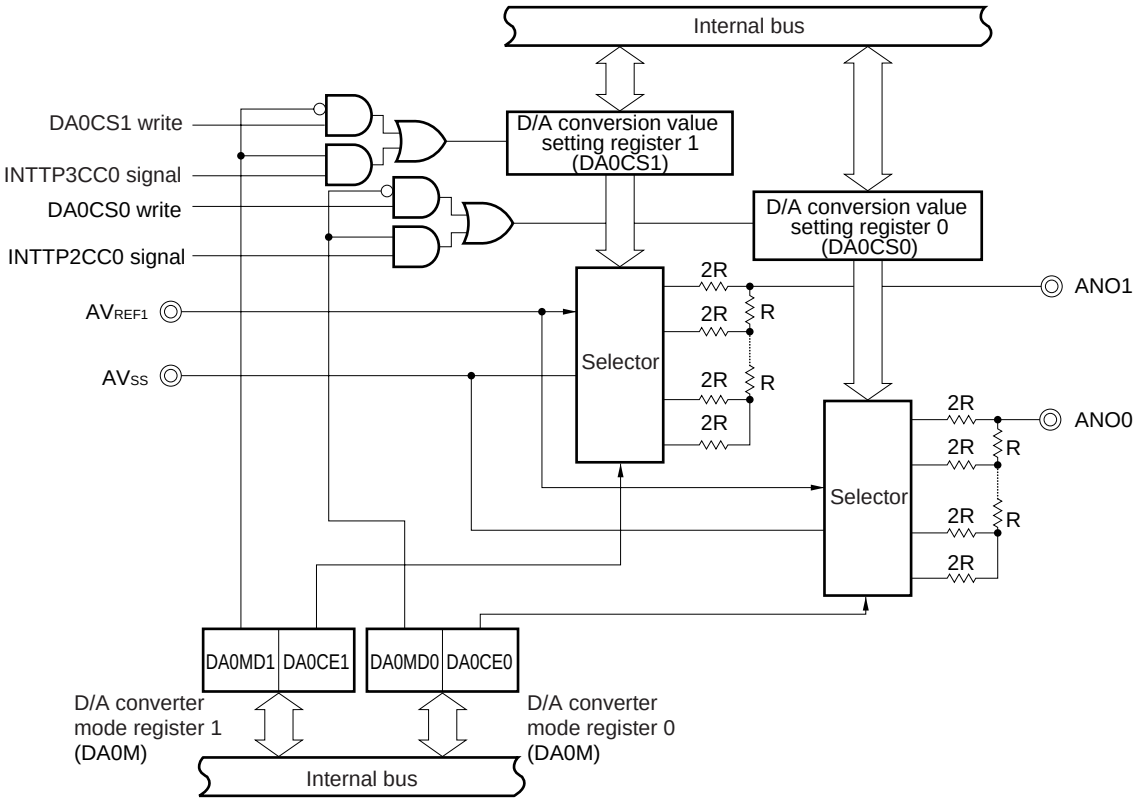
The D/A converter includes the following hardware.

Table 15-1. D/A Converter Registers Used by Software

Item	Configuration
Control registers	D/A converter mode register (DA0M) D/A conversion value setting registers 0 and 1 (DA0CS0, DA0CS1)

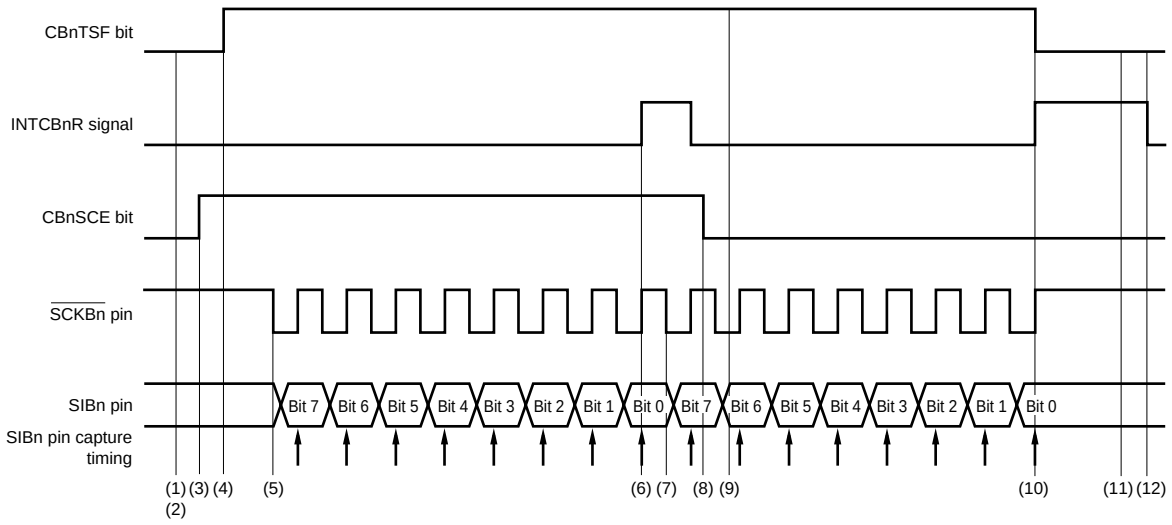
The block diagram of the D/A converter is shown below.

Figure 15-1. Block Diagram of D/A Converter



Caution DA0CS0 and D0ACS1 are also used as the AV_{REF1} and AV_{SS} pins. The AV_{SS} pin is also used by the A/D converter.

Figure 18-26. Continuous Transfer Mode Operation Timing (Slave Mode, Reception Mode)



(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write A3H to the CBnCTL0 register, and select the reception mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by performing a dummy read of the CBnRX register, and the device waits for serial clock input.

(5) When the serial clock is input, capture the receive data of the SIBn pin in synchronization with the serial clock.

(6) When reception is completed, the reception complete interrupt request signal (INTCBnR) is generated, and reading receive data from the CBnRX register is enabled.

(7) When the serial clock is input with the CBnCTL0.CBnSCE bit set to 1, continuous reception is started.

(8) To end continuous reception with the current reception, clear the CBnSCE bit to 0.

(9) Read the CBnRX register.

(10) When reception is completed, the INTCBnR signal is generated, and reading receive data from the CBnRX register is enabled. If the CBnSCE bit is set to 0 before communication is complete, clear the CBnTSF bit to 0 to end the receive operation.

(11) Read the CBnRX register.

(12) To disable reception, clear the CBnCTL0.CBnPWR and CBnCTL0.CBnRXE bits to 0 after confirming that the CBnTSF bit is 0.

Remark n = 0 to 4

(3) Start ~ Code ~ Data ~ Start ~ Code ~ Data ~ Stop

<1> When WTIMn bit = 0 (after restart, extension code reception)

ST	AD6 to AD0	R/$\overline{W}$	$\overline{ACK}$	D7 to D0	$\overline{ACK}$	ST	AD6 to AD0	R/$\overline{W}$	$\overline{ACK}$	D7 to D0	$\overline{ACK}$	SP
			1		2				3		4	' 5

1: IICSn register = 0010X010B

2: IICSn register = 0010X000B

3: IICSn register = 0010X010B

4: IICSn register = 0010X000B

' 5: IICSn register = 00000001B

Remarks 1. : Always generated

' : Generated only when SPIEn bit = 1

X: don't care

2. n = 0 to 2

<2> When WTIMn bit = 1 (after restart, extension code reception)

ST	AD6 to AD0	R/$\overline{W}$	$\overline{ACK}$	D7 to D0	$\overline{ACK}$	ST	AD6 to AD0	R/$\overline{W}$	$\overline{ACK}$	D7 to D0	$\overline{ACK}$	SP
			1	2		3			4	5		6 ' 7

1: IICSn register = 0010X010B

2: IICSn register = 0010X110B

3: IICSn register = 0010XX00B

4: IICSn register = 0010X010B

5: IICSn register = 0010X110B

6: IICSn register = 0010XX00B

' 7: IICSn register = 00000001B

Remarks 1. : Always generated

' : Generated only when SPIEn bit = 1

X: don't care

2. n = 0 to 2

21.5.2 Debug trap

A debug trap is an exception that occurs when the DBTRAP instruction is executed and can always be acknowledged.

(1) Operation

If a debug trap occurs, the CPU performs the following processing.

<1> Saves the current PC to DBPC.
<2> Saves the current PSW to DBPSW.
<3> Sets the PSW.NP, PSW.EP, and PSW.ID bits to 1.
<4> Sets the handler address (00000060H) for the debug trap to the PC and transfers control.

(4) Regulator protection register (REGPR)

The REGPR register is used to protect the regulator output voltage level control register 0 (REGOVL0) so that

23.7 Subclock Operation Mode/Low-Voltage Subclock Operation Mode

23.7.1 Setting and operation status

The subclock operation mode is set by setting the PCC.CK3 bit to 1 in the normal operation mode. The low-voltage subclock operation mode is set by setting the REGOVL0 register to 02H in the subclock operation mode.

When the subclock operation mode is set, the internal system clock is changed from the main clock to the subclock. Check whether the clock has been switched by using the PCC.CLS bit.

When the PCC.MCK bit is set to 1, the operation of the main clock oscillator is stopped. As a result, the system operates only on the subclock.

In the subclock operation mode, power consumption can be reduced to a level lower than in the normal operation mode because the subclock is used as the internal system clock. In addition, power consumption can be further reduced to the level of the STOP mode by stopping the operation of the main clock oscillator. Power consumption decreases further in the low-voltage subclock operation mode because the voltage of the regulator is lowered.

When the main clock oscillator is stopped in the subclock operation mode, CSIBn and UARTA0, which can operate on the external clock, also continue operating. In the low-voltage subclock operation mode, however, stop supplying the external clock input to CSIBn and UARTA0 because these blocks cannot continue operating (n = 0 to 4).

Cautions 1. When manipulating the CK3 bit, do not change the set values of the PCC.CK2 to PCC.CK0 bits (using a bit manipulation instruction to manipulate the bit is recommended). For details of the PCC register, see 6.3 (1) Processor clock control register (PCC).

2. If the following conditions are not satisfied, change the CK2 to CK0 bits so that the conditions are satisfied and set the subclock operation mode.

Internal system clock (f_{CLK}) > Subclock (f_{XT} = 32.768 kHz) u 4

Remark Internal system clock (f_{CLK}): Clock generated from main clock (f_{XX}) in accordance with the settings of the CK2 to CK0 bits

Be sure to set the low-voltage subclock operation mode using the following procedure.

(1) Procedure for switching from subclock operation mode to low-voltage subclock operation mode

Make the following settings in the subclock operation mode.

<1> Stop the main clock and PLL.

<2> Stop the functions whose operation is specified as stopped in **Table 23-14 Operating Status in Low-Voltage Sub-IDLE Mode**.

Be especially sure to stop the following, because they are signals from external sources.

- Stop the SCKBn input clock when the SCKBn input clock to CSIBn is selected (n = 0 to 4).
- Stop the ASCKA0 input clock when the ASCKA0 input clock to UARTA0 is selected.

<3> Disable DMA (if DMA is enabled).

<4>
- Disable maskable interrupts by using the DI instruction.
- Disable the NMI interrupt (INTF02 = 0, INTR02 = 0).
- Create a status in which the INTWDT2 signal is not generated (create a status in which the INTWDT2