V850E2/MN4

Timer Array Unit Control

R01AN0922EJ0100
Rev.1.00
Jan 13, 2012

Introduction

This application note explains how to set up the 16-bit timer array unit A (TAUA) and 32-bit timer array unit J (TAUJ) and also gives an outline of the operation and describes the procedure for using a sample program. The sample program makes the TAUA generate the PWM signal and output the signal to the TAUJ and makes the TAUJ measure the width of the signal input from the TAUA.

Target Device

V850E2/MN4 Microcontrollers

Contents

1. Overview

This application note illustrates the usage examples of the 16-bit TAUA (timer array unit A) and 32-bit TAUJ (timer array unit J).

In the TAUA, channel 0 is set as a master channel, channel 1 is set as a slave channel, and the PWM signal is output from the TAUA1TTOUT1 pin in synchronous channel operation. In the TAUJ, the TAUJ0 is set to independent channel operation, the PWM signal output from the TAUA is input to the TAUJ0TTIN0 pin, and the signal width is measured.

See section 4 for the details of the individual operations.

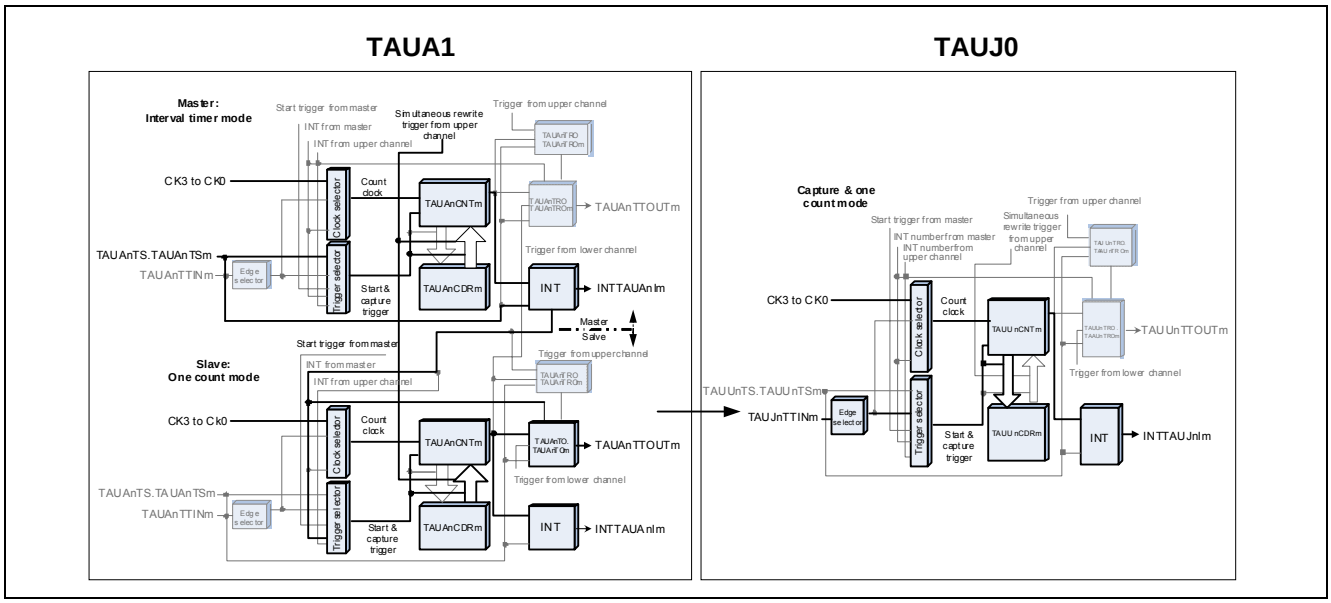

Figure 1.1 TAU Block Diagram

1.1 Initialization

The general registers and functional pins are initialized.

<Port setup>

- Port n function control expansion registers (PFCEn)
- Port n function control registers (PFCn)
- Port n mode control registers (PMCn)
- Port n mode registers (PMn)

1.2 TAUA Setup

The TAUA is set to synchronous channel operation and PWM signal output function. Channel 0 in the TAUA1 is set as a master channel and set to interval timer mode. Channel 1 in the TAUA1 is set as a slave channel and set to one count mode. See section 4.2 for details.

1.3 TAUJ Setup

The TAUJ is set to independent channel operation and signal width measurement function. Channel 0 in the TAUJ0 is set to capture & one count mode. See section 4.3 for details.

2. Usage Environment

This section explains the circuit diagram and development environment to run this sample program.

2.1 Circuit Diagram

See “V850E2/MN4 Target Board User Manual: QB-V850E2MN4DUAL-TB (R20UT0683XJ)” for the details of the circuit diagram.
The P3_1 pin functions as the TAUA1TTOUT1 pin to output the PWM signal and the P2_1 pin functions as the TAUJ0TTIN0 pin to input the PWM signal. The P3_1 pin must be connected to the P2_1 pin.

2.2 Development Environment

It is necessary to install the tools that are listed below to run the sample program.

- CubeSuite+

The integrated development environment CubeSuite+ from Renesas Electronics provides various software development tools that are necessary for the user to develop applications. The user can use these tools seamlessly and easily in various development stages including coding, assembly, compilation, debugging using an emulator or simulator, and flash programming.

- MINICUBE

MINICUBE is a general-purpose in-circuit emulator from Renesas Electronics which adopts the JTAG interface system. It allows the user to debug an onboard real processor and provides highly transparent and stable emulation functionalities. An adapter is required to connect a TB board to MINICUBE.

3. Software

This section describes the file organization of the sample program.

3.1 File Organization

The sample program consists of the files that are listed below.

File Name (Tool Structure)	Description	Common Source File	CubeSuite+ File
crtE.s	Hardware initialization processing		●
V850E2MN4.dir	Link/directive file		●
taua.h	Variable and function declarations	●	
main.c	Main processing	●	
initial.c	Software initialization processing	●	
taua_control.c	Timer array unit A setup	●	
tauj_control.c	Timer array unit J setup	●	
interrupt.c	Interrupt processing	●	

4. Sample Application

This section explains how to set up the TAUA and the TAUJ.

4.1 Flow Charts

The flow charts of this sample program are shown below.

4.1.1 Main Processing

The main processing sets up the ports and the timer array units. The timers are started, the TAUA is made to generate the PWM pulse, and then, the TAUJ is made to measure the pulse width.

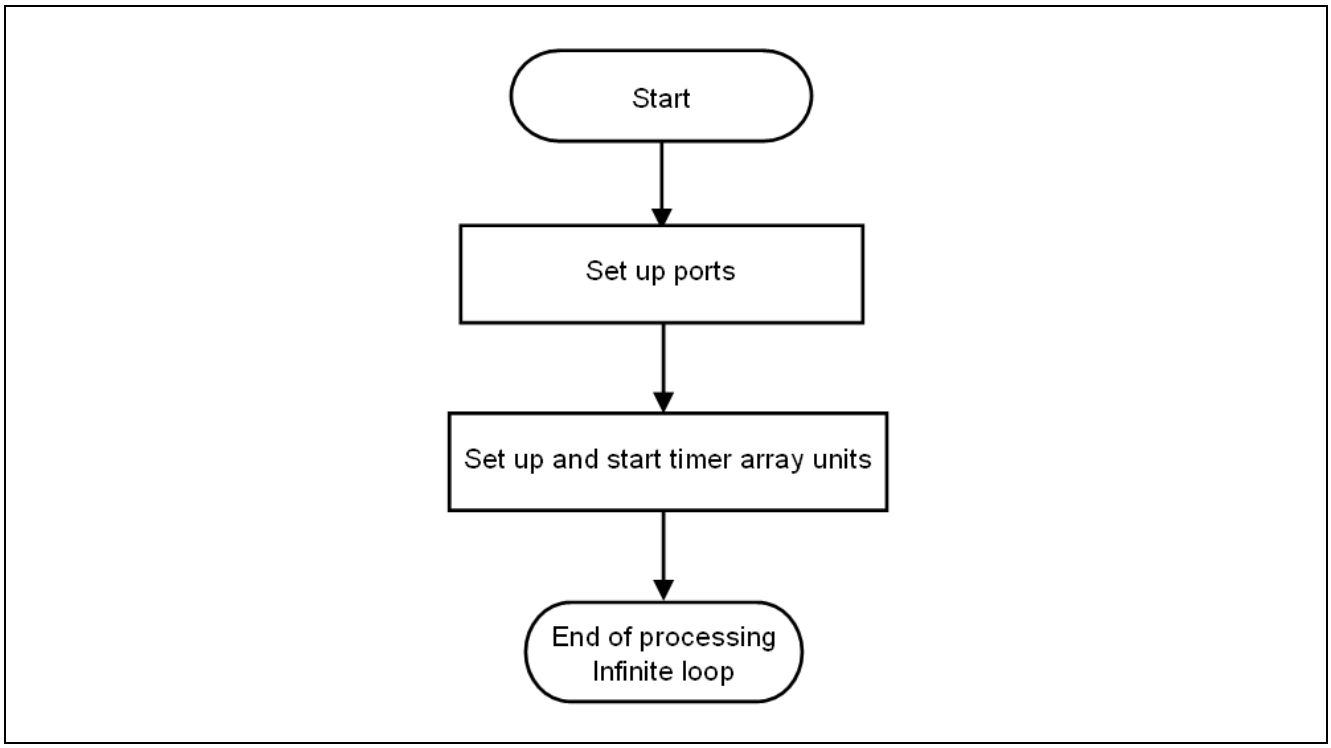

Figure 4.1 Main Processing

4.1.2 Interrupt Processing Flow

The INTTUAJ0I0 interrupt function processing checks for an overflow to calculate the input PWM signal width (HIGH).

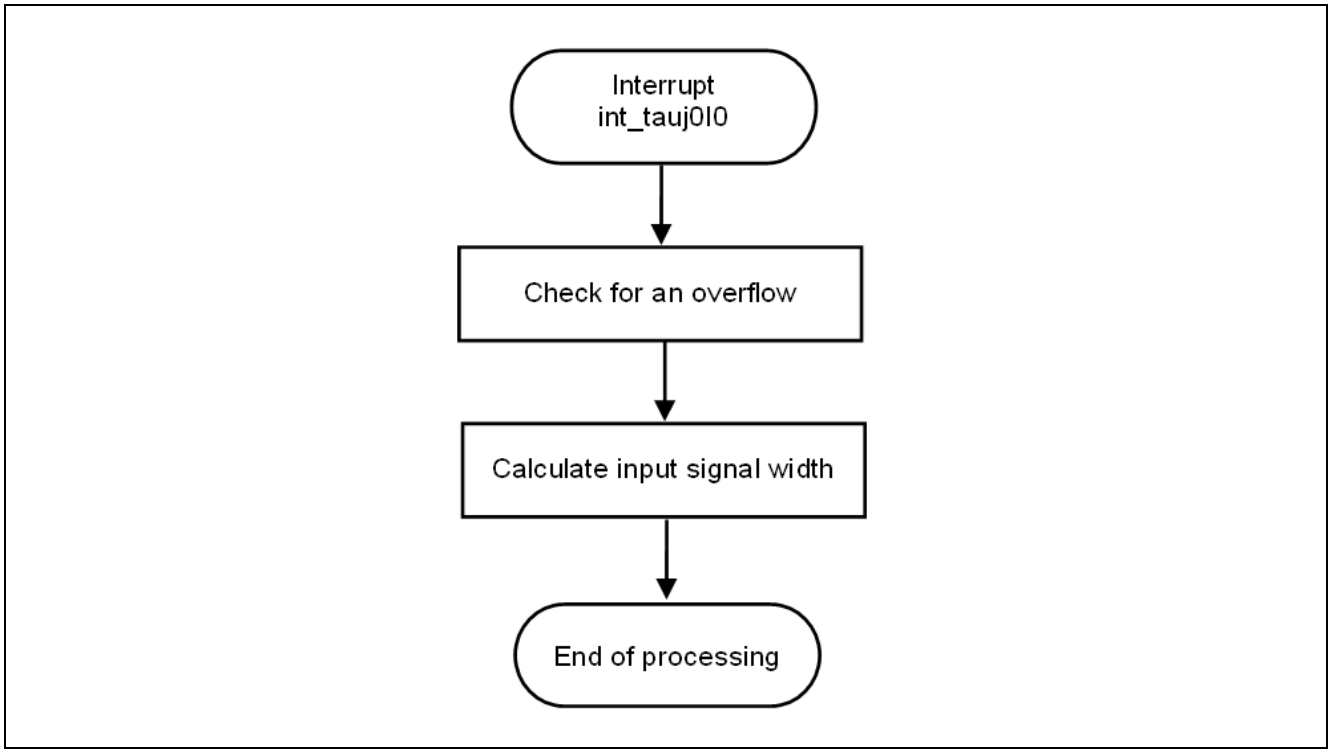

Figure 4.2 Interrupt Processing Flow

4.2 Details of TAUA Setup

In this sample program, the TAUA1 is set to synchronous channel operation and PWM output function. Channel 0 in the TAUA1 is set as a master channel and set to interval timer mode. Channel 1 in the TAUA1 is set as a slave channel and set to synchronous channel output mode 1 in one count mode. The PWM pulse is output from the TAUA1TTOUT1 pin. The pulse cycle is set in the master channel and the duty is set in the slave channel.

The counters are enabled by setting the channel trigger bit TAUA1TS.TAUAnTS[1/0] to 1. Then, bit TAUA1TE.TAUA1TE[1/0] is set to 1 and counting is enabled. The current value of TAUA1CDR0 is loaded into TAUA1CNT0. The counters start counting down at that value of TAUA1CDR0. The current value of TAUA1CDR1 is loaded into TAUA1CNT1 and the counters start to count down from these values. INTTAUA1I0 is generated in the master channel and the PWM signal is output by setting and resetting the TAUA1TTOUT1 (slave).

When the counter of the master channel reaches 0000H and a pulse cycle time has elapsed, INTTAUA1I0 is generated. The value of TAUA1CDR0 is loaded into TAUA1CNT0 and then the counter starts to count down from this value.

The INTTAUA1I0 of the master channel triggers the counter of the slave channel. The current value of TAUA1CDR1 (slave) is loaded into TAUA1CNT1 (slave) and then the counter starts to count down from this value. The TAUA1TTOUT1 signal becomes active. When the counter reaches 0000H (duty time has elapsed), INTTAUA1I1 is generated and the TAUA1TTOUT1 signal becomes inactive. The counter returns to FFFFH and awaits the next INTTAUA1I0 from the master channel (the start of the next pulse cycle).

The counters can be stopped by setting TAUA1TT.TAUA1TT[1/0] to 1 in the master and slave channel. Then, TAUA1TE.TAUA1TE[1:0] are set to 0. TAUA1CNT1, TAUA1CNT0, and TAUA1TTOUT1 of master and slave channel stop but retain their values. The counters can be restarted by setting channel trigger bits TAUA1TS.TAUA1TS[1/0] to 1.

Pulse period = (TAUA1CDR0 (master) 1) × count clock cycle

Duty cycle [%] = (TAUA1CDR1 (slave) / (TAUA1CDR0 (master) 1)) × 100%

In this sample program, the duty cycle is set to 80%.

The general timing diagram for the PWM output function is shown below.

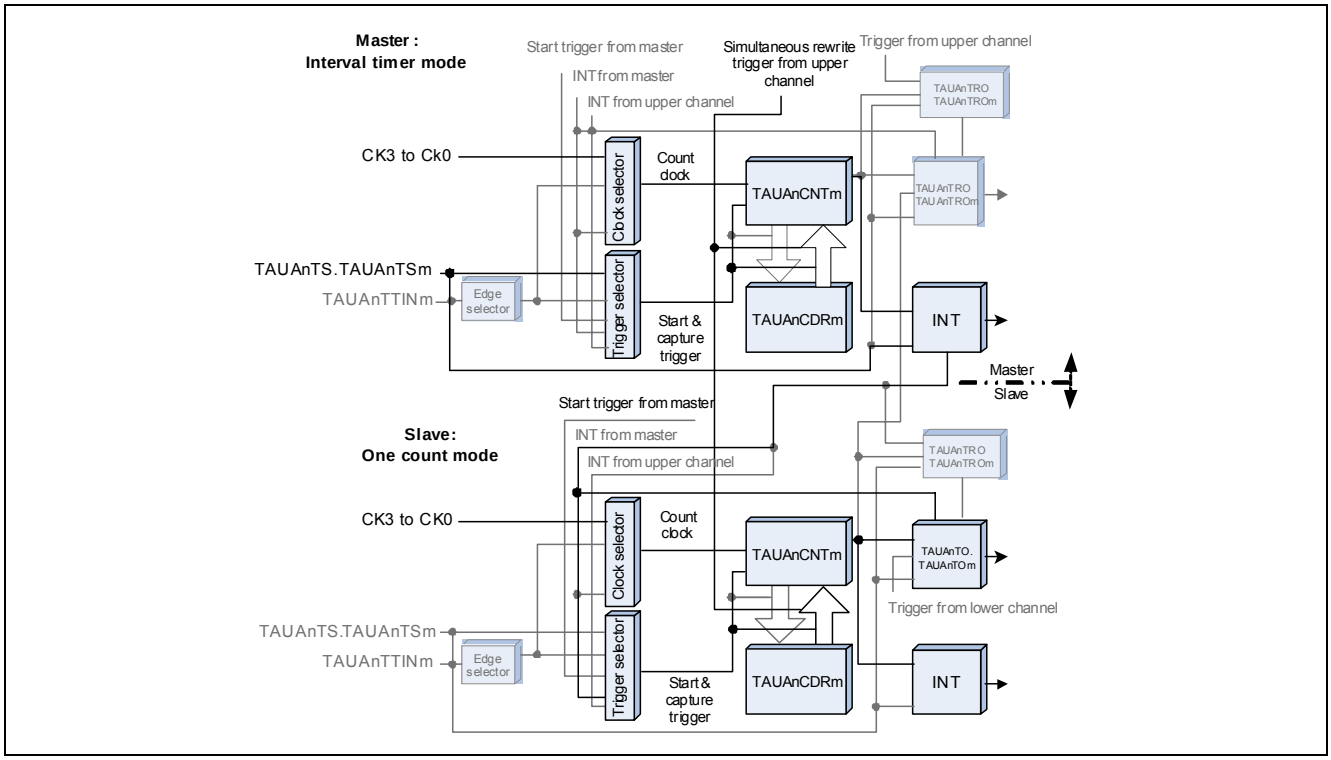

Figure 4.3 Block Diagram for PWM Output Function

Slave channel: Positive logic (TAUAnTOL.TAUAnTOLm = 0)

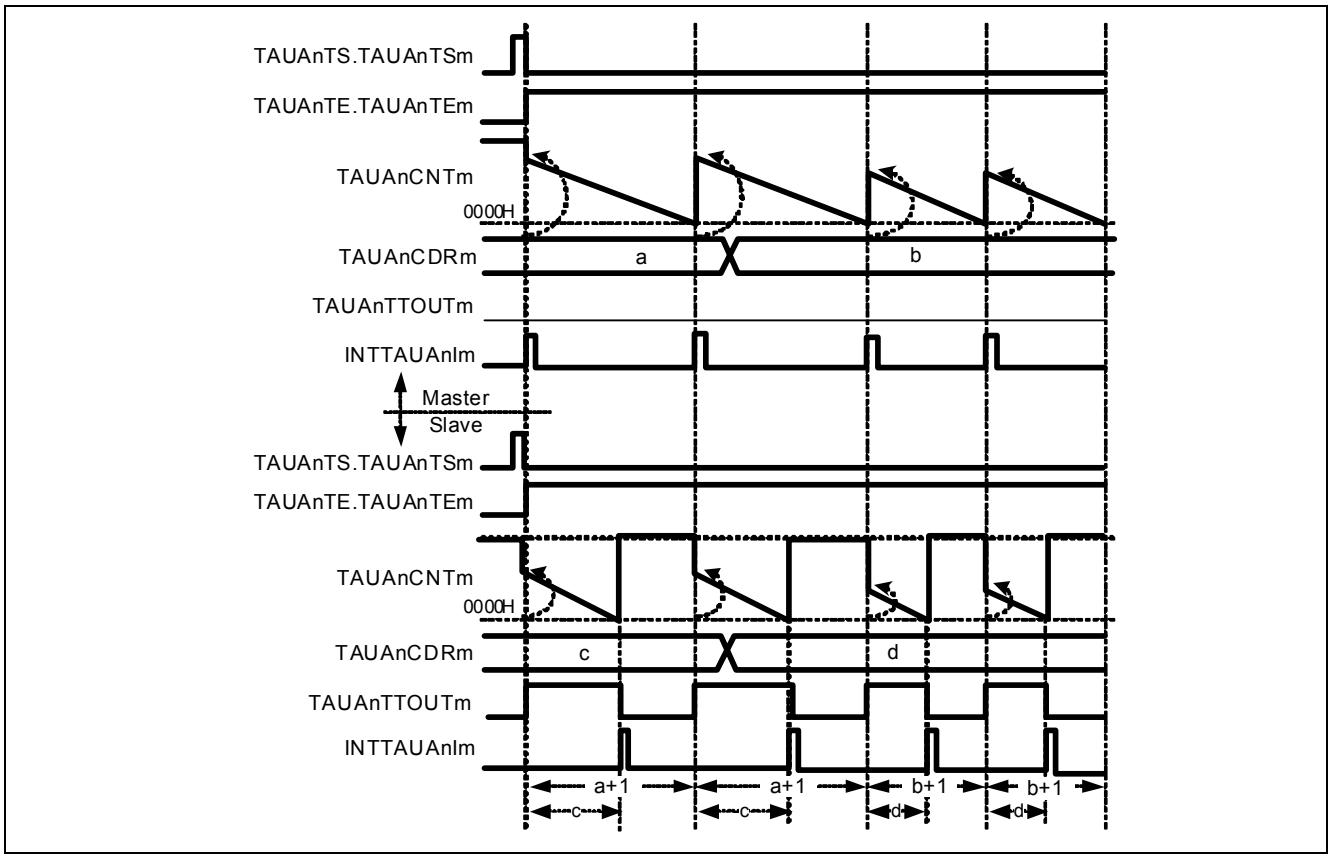

Figure 4.4 General Timing Diagram for PWM Output Function

For specific mode setup, see “V850E2/MN4 Hardware User Manual: Renesas MCU V850E2/Mx4 microcontrollers (R01UH0011EJ).”

4.3 Details of TAUJ Setup

In this sample program, the TAUJ0 is set to independent channel operation and signal width measurement function. Channel 0 is set to capture & one count mode. The TAUJ0TTIN0 pin is used to measure the input signal width (HIGH).

The counter is enabled by setting the channel trigger bit TAUJ0TS.TAUJ0TS0 to 1. Then, bit TAUJ0TE.TAUJ0TE0 is set to 1 and counting is enabled. When a valid start edge of TAUJ0TTIN0 is detected, the TAUJ0CNT0 counter starts counting up from 00000000H. When a valid stop edge of TAUJ0TTIN0 is detected, the value of TAUJ0CNT0 is captured and transferred to TAUJ0CDR0 and then interrupt INTTAUJ0I0 is generated. The counter retains its value and await the next valid start edge of TAUJ0TTIN0.

If the counter reaches FFFFFFFFH before a valid stop edge of TAUJ0TTIN0 is detected, it overflows. The counter is reset to 00000000H and then continues to operate. The values transferred to TAUJ0CDR0 and TAUJ0CSR0.TAUJ0OVF respectively depend on the values of bits TAUJ0CMOR0.TAUJ0COS[1:0]. See "V850E2/MN4 Hardware User Manual: Renesas MCU V850E2/Mx4 microcontrollers (R01UH0011EJ)" for details.

TAUJ0TTIN0 input signal width = count clock cycle × [(TAUJ0CSR0.TAUJ0OVF ×(FFFFFFFFH + 1)) + TAUJ0CDR0 capture value + 1]

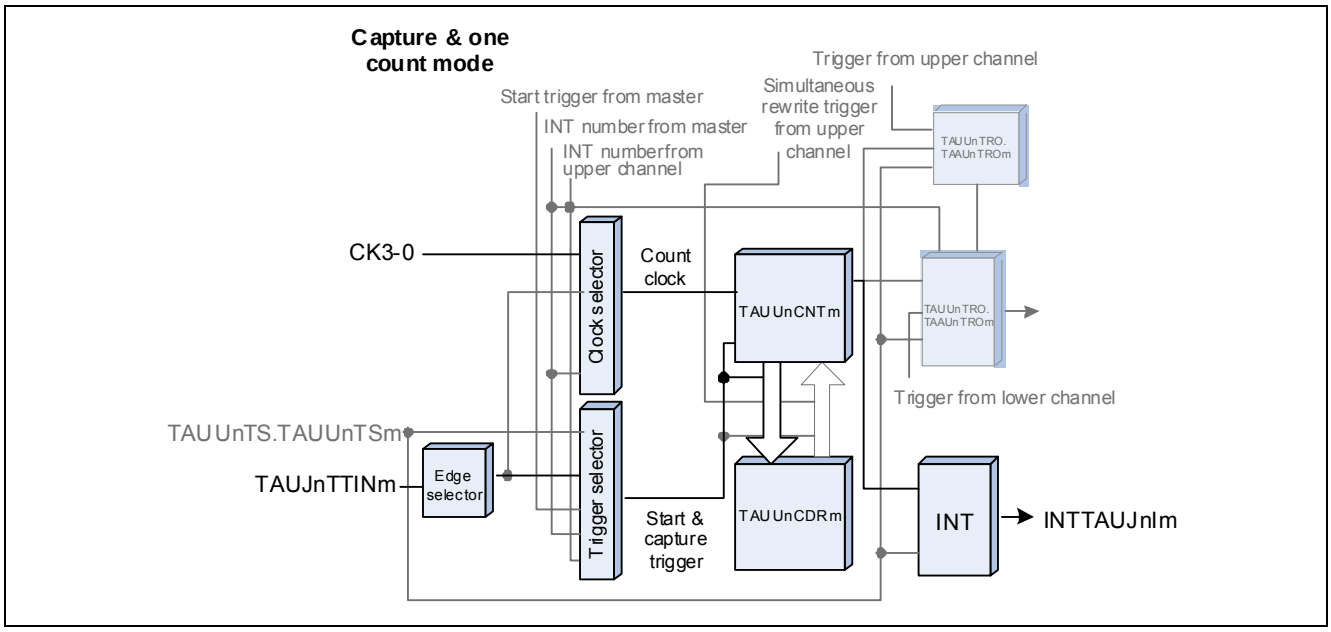

Figure 4.5 Block Diagram for Input Signal Width Measurement Function

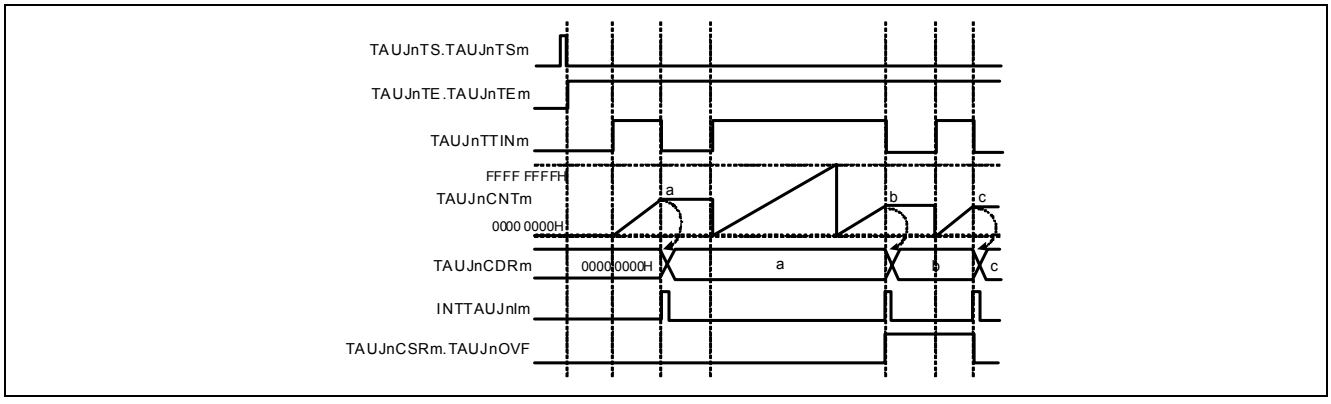

Figure 4.6 General Timing Diagram for Input Signal Width Measurement Function

For specific mode setup, see "V850E2/MN4 Hardware User Manual: Renesas MCU V850E2/Mx4 microcontrollers (R01UH0011EJ)."

4.4 Register Setup

This section explains how to set up the relevant registers according to the flow charts shown in section 4.1. The registers described below must be configured to control the timer I/O.

4.4.1 Port Setup

In this sample program, the pin P3_1, which is used to generate the PWM signal in the TAUA, is used as TAUA1TTOUT1. The pin P2_1, which is used to measure the input signal width in the TAUJ, is used as TAUJ0TTIN0. The pertinent control registers must be set up as shown in the table below.

Macro	Pin	PMC	PFCE	PFC	PM	Corresponding Function
TAUA1TTOUT1	P3_1	1	0	1	0	Alternative mode 2, output
TAUJ0TTIN0	P2_1	1	1	0	1	Alternative mode 3, input

Setting examples

```
/* P3_1:  TAUA1TTOUT1; alternative mode 2; output */
PFCE3 |= 0x0000;
PFC3  |= 0x0002;
PMC3  |= 0x0002;
PM3   &= 0xfffd;

/* P2_1:  TAUJ0TTIN1; alternative mode 3; input */
PFCE2 |= 0x0002;
PFC2  |= 0x0000;
PMC2  |= 0x0002;
PM2   |= 0x0002;
```

4.4.2 TAUAn Prescaler Registers

- TAUAn prescaler clock select register (TAUAnTPS)

This register specifies the CK0, CK1, CK2, and CK3_PRE prescaler clocks for all channels. CK3 is generated by dividing CK3_PRE by the factor specified in TAUAnBRS.

In this sample program, CK0 is specified.

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

Address <TAUAn_base_OS> + 240_H

Initial Value $FFFF_H$. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAnPRS3[3:0]				TAUAnPRS2[3:0]				TAUAnPRS1[3:0]				TAUAnPRS0[3:0]			
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-224 TAUAnTPS register contents (1/4)

Bit position	Bit name	Function
15 to 12	TAUAn PRS3[3:0]	Specifies the CK3_PRE clock. Clock CK3_PRE is the input clock of the BRG unit. The BRG unit supplies the CK3 operation clock for all channels.

PRS3[3:0]	CK3_PRE clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

These bits can only be rewritten when all counters using CK3 are stopped (TAUAnTE.TAUAnTEm = 0).

Figure 4.7 TAUAnTPS Register Format (1/4)

Table 12-224 TAUAnTPS register contents (2/4)

Bit position	Bit name	Function
11 to 8	TAUAn PRS2[3:0]	Specifies the CK2 clock.

PRS2[3:0]	CK2 clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

These bits can only be rewritten when all counters using CK2 are stopped (TAUAnTE.TAUAnTEm = 0).

Figure 4.8 TAUAnTPS Register Format (2/4)

Table 12-224 TAUAnTPS register contents (3/4)

Bit position	Bit name	Function
7 to 4	TAUAn PRS1[3:0]	Specifies the CK1 clock.

PRS1[3:0]	CK1 clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

These bits can only be rewritten when all counters using CK1 are stopped (TAUAnTE.TAUAnTEm = 0).

Figure 4.9 TAUAnTPS Register Format (3/4)

Table 12-224 TAUAnTPS register contents (4/4)

Bit position	Bit name	Function
3 to 0	TAUAn PRS0[3:0]	Specifies the CK0 clock.

PRS0[3:0]	CK0 clock
0000_B	$PCLK/2^0$
0001_B	$PCLK/2^1$
0010_B	$PCLK/2^2$
0011_B	$PCLK/2^3$
0100_B	$PCLK/2^4$
0101_B	$PCLK/2^5$
0110_B	$PCLK/2^6$
0111_B	$PCLK/2^7$
1000_B	$PCLK/2^8$
1001_B	$PCLK/2^9$
1010_B	$PCLK/2^{10}$
1011_B	$PCLK/2^{11}$
1100_B	$PCLK/2^{12}$
1101_B	$PCLK/2^{13}$
1110_B	$PCLK/2^{14}$
1111_B	$PCLK/2^{15}$

These bits can only be rewritten when all counters using CK0 are stopped (TAUAnTE.TAUAnTEm = 0).

Note The TAUAn clock input PCLK is specified in the first section of this chapter under the keyword “Clock supply”.

Figure 4.10 TAUAnTPS Register Format (4/4)

Setting example

```
TAUA1TPS = 0x000a;                    /* CK0:PCLK / 2^10 */
```

- TAUAn prescaler baudrate value register (TAUAnBRS)

This register specifies the division factor of prescaler clock CK3.

CK3 is generated by dividing CK3_PRE by the factor specified in this register plus one. The PCLK prescaler for CK3_PRE is specified in TAUAnTPS.TAUAnPRS3[3:0].

This register does not use CK3. Setting this register is unnecessary.

4.4.3 TAUAn Control Registers

- TAUAn channel data register (TAUAnCDRm)

This register functions either as a compare register or a capture register, depending on the operation mode specified in TAUAnCMORm.TAUAnMD[4:1].

In this sample program, the cycle of the PWM pulse is set in TAUA1CDR0 and the duty of the pulse is set in TAUA1CDR1.

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

- In capture mode, only reading is possible. Write operation is ignored.
- In compare mode, reading and writing is possible.

Address <TAUAn_base_USER> + m x 4_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAnCDR[15:0]															
R/W															

Table 12-226 TAUAnCDRm register contents

Bit position	Bit name	Function
15 to 0	TAUAn CDR[15:0]	Data register for the capture/compare value.

Figure 4.11 TAUAnCDRm Register Format

Setting example

```
TAUA1CDR0 = 4999;          /* cycle of PWM */
TAUA1CDR1 = 4000;          /* duty of PWM is 80%*/
```

- TAUAn channel counter register (TAUAnCNTm)

This register is the channel m counter register.

Access This register can be read in 16-bit units.

Address <TAUAn_base_USER> + 80_H + m x 4_H

Initial Value 0000_H or $FFFF_H$ The initial value depends on the operation mode, see *Table 12-228 "TAUAnCNTm read values after the counter is re-enabled" on page 867*. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAnCNT[15:0]															
R															

Table 12-227 TAUAnCNTm register contents

Bit position	Bit name	Function
15 to 0	TAUAn CNT[15:0]	16-bit counter value.

The read value depends on the counter, the operation mode change, and the values of the TAUAnTS.TAUAnTSm and TAUAnTT.TAUAnTTm bits.

The *initial* counter read value depends on the operation mode and how the counter was stopped:

- by a reset
- by a counter stop trigger (TAUAnTT.TAUAnTTm = 1)

The following table lists the initial counter read values after the counter has stopped (TAUAnTE.TAUAnTEm = 0) and re-enabled (TAUAnTS.TAUAnTSm = 1).

The table also contains the counter read value one count after the counter is enabled (TAUAnTS.TAUAnTSm = 1) for modes where the counter waits for a start trigger.

Table 12-228 TAUAnCNTm read values after the counter is re-enabled (1/2)

Mode name	Count method (up/down)	TAUAnCNTm value		
		After reset	After stop trigger	After one count
Interval Timer mode	Count down	$FFFF_H$	Stop value	-
Judge mode	Count down	$FFFF_H$	Stop value	-
Capture mode	Count up	0000_H	Stop value	-
Event Count mode	Count down	$FFFF_H$	Stop value	-
One Count mode	Count down	$FFFF_H$	Stop value	$FFFF_H$
Capture & One Count mode	Count up	0000_H	Stop value	Captured value + 1 (TAUAnCDRm)
Judge & One Count mode	Count down	$FFFF_H$	Stop value	TAUAnCNTm value - 1
Up Down Count mode	Count up/down	$FFFF_H$	Stop value	-
Pulse One Count mode	Count down	$FFFF_H$	Stop value	0000_H

Figure 4.12 TAUAnCNTm Register Format (1/2)

Table 12-228 TAUAnCNTm read values after the counter is re-enabled (2/2)

Mode name	Count method (up/down)	TAUAnCNTm value		
		After reset	After stop trigger	After one count
Count Capture Mode	Count up	0000_H	Stop value	-
Gate Count Mode	Count down	$FFFF_H$	Stop value	Stop value
Capture & Gate Count Mode	Count up	0000_H	Stop value	Stop value

Note If the operation mode is changed while the counter is stopped, the initial counter value after counter restart is undefined. The operation mode is changed by register TAUAnCMORm.TAUAnMD[4:1].

Figure 4.13 TAUAnCNTm Register Format (2/2)

- TAUAn channel mode OS register (TAUAnCMORm)

This register controls channel m operation.

In this sample program, channel 0 in the TAUA1 is set as a master channel and set to interval timer mode. When the counter is triggered by software trigger, INTTAUA1I0 is generated at the start of operation. Channel 1 is set as a slave channel, is set to one count mode, and enables the start trigger during operation by using INTTAUA1I0 of the master channel as a start trigger.

Access This register can be read or written in 16-bit units. Writing is only possible while the counter is stopped (TAUAnTE.TAUAnTEm = 0).

Address <TAUAn_base_OS> + 200_H + m x 4_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn CKS[1:0]		TAUAn CCS[1:0]		TAUAn MAS	TAUAnSTS[2:0]			TAUAn COS[1:0]		-	TAUAnMD[4:0]				
R/W		R/W		R/W	R/W			R/W		R	R/W				

Table 12-229 TAUAnCMORm register contents (1/4)

Bit position	Bit name	Function
15,14	TAUAn CKS[1:0]	Selects the operation clock. The operation clock is used for the TAUAnTTINm input edge detection circuit. It can also be used as the count clock depending on bits TAUAnCMORm.TAUAnCCS[1:0]. **TAUAn CKS1 / TAUAn CKS0 / Selected operation clock** 0 / 0 / CK0 0 / 1 / CK1 1 / 0 / CK2 1 / 1 / CK3
13,12	TAUAn CCS[1:0]	Selects the count clock for TAUAnCNTm counter: **TAUAn CCS1 / TAUAn CCS0 / Selected count clock** 0 / 0 / Operation clock as specified by TAUAnCMORm.TAUAnCKS[1:0]. 0 / 1 / Valid edge of TAUAnTTINm input signal 1 / 0 / Setting prohibited 1 / 1 / INTTAUAnIm signal of the master channel
11	TAUAnMAS	Specifies the channel as master or slave channel during synchronous channel operation: 0: Slave 1: Master This bit is only valid for even channels (CHm_even). For odd channels (CHm_odd), it is fixed to 0.

Figure 4.14 TAUAnCMORm Register Format (1/4)

Table 12-229 TAUAnCMORm register contents (2/4)

Bit position	Bit name	Function
10 to 8	TAUAn STS[2:0]	Selects the external start trigger:

TAUAn STS2	TAUAn STS1	TAUAn STS0	Description
0	0	0	Software trigger
0	0	1	Valid edge of the TAUAnTTINm input signal. TAUAnCMURm.TAUAnTIS[1:0] specifies the valid edge.
0	1	0	Valid edge of the TAUAnTTINm input signal is the start trigger and the reverse edge is the stop (capture) trigger
0	1	1	Setting prohibited
1	0	0	INTTAUAnI of the master channel
1	0	1	INTTAUAnI of the upper channel (m-1), regardless of the master setting
1	1	0	Dead-time output signal of the TAUAnTTOUTm generation unit
1	1	1	Up/down output trigger signal TAUAnTUDSm of the master channel.

Figure 4.15 TAUAnCMORm Register Format (2/4)

Table 12-229 TAUAnCMORm register contents (3/4)

<table>
<tr><th>Bit position</th><th>Bit name</th><th>Function</th></tr>
<tr><td>7, 6</td><td>TAUAn COS[1:0]</td><td>Specifies when the capture register TAUAnCDRm and the overflow flag TAUAnCSRm.TAUAnOVF of channel m are updated.
These bits are only valid if channel m is in capture mode.
<table>
<tr><th>TAUAn COS1</th><th>TAUAn COS0</th><th>TAUAnCDRm</th><th>TAUAnCSRm.TAUAnOVF</th></tr>
<tr><td>0</td><td>0</td><td rowspan="2">Updated upon detection of a TAUAnTTINm input valid edge.</td><td>Updated (cleared or set) upon detection of a TAUAnTTINm input valid edge:
• If a counter overflow has occurred since the last valid edge detection, TAUAnCSRm.TAUAnOVF is set.
• If no counter overflow has occurred since the last valid edge detection, TAUAnCSR.OVF is cleared.</td></tr>
<tr><td>0</td><td>1</td><td>Set upon counter overflow and cleared by setting TAUAnCSCm.TAUAnCLOV.</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">Updated upon detection of a TAUAnTTINm input valid edge and upon counter overflow:
• TAUAnTTINm input valid edge: Counter value is written to TAUAnCDRm
• Overflow: $FFFF_H$ is written to TAUAnCDRm. The next TAUAnTTINm input valid edge detection is ignored.</td><td>Not set.</td></tr>
<tr><td>1</td><td>1</td><td>Set upon counter overflow and cleared by setting TAUAnCSCm.TAUAnCLOV.</td></tr>
</table>
</td></tr>
</table>

Figure 4.16 TAUAnCMORm Register Format (3/4)

Table 12-229 TAUAnCMORm register contents (4/4)

Bit position	Bit name	Function
4 to 0	TAUAn MD[4:0]	Specifies the operation mode.

TAUAn MD4	TAUAn MD3	TAUAn MD2	TAUAn MD1	TAUAn MD0	Description
0	0	0	0	1/0	Interval Timer mode
0	0	0	1	1/0	Judge mode
0	0	1	0	1/0	Capture mode
0	0	1	1	0	Event Count mode
0	1	0	0	1/0	One Count mode
0	1	0	1	1/0	Setting prohibited
0	1	1	0	0	Capture & One Count mode
0	1	1	1	1/0	Judge & One Count mode
1	0	0	0	0	Setting prohibited
1	0	0	1	0	Up Down Count mode
1	0	1	0	1/0	Pulse One Count mode
1	0	1	1	1/0	Count Capture mode
1	1	0	0	0	Gate Count mode
1	1	0	1	0	Capture & Gate Count mode

Mode	Role of the TAUAnMD0 bit
Interval Timer mode Capture mode Count Capture mode	Specifies whether the INTTAUAnIm signal is output when the counter starts counting (when the start trigger is input). 0: No INTTAUAnIm generated 1: INTTAUAnIm generated
Event Count mode Up Down Count mode	This bit must be set to 0.
One Count mode Gate Count mode Pulse One Count mode	Enables/disables start trigger detection during counting. 0: Disabled 1: Enabled
Capture & One Count mode Capture & Gate Count mode	This bit must be set to 0.
Judge mode Judge One Count mode	Specifies when INTTAUAnIm is generated. 0: When TAUAnCNTm ≤ TAUAnCDRm 1: When TAUAnCNTm > TAUAnCDRm

Figure 4.17 TAUAnCMORm Register Format (4/4)

Setting examples

```
TAUA1CMOR0 = 0x0801;
             /* CK0, master, software trigger count, interval mode, int at start */
TAUA1CMOR1 = 0x0409;
             /* CK0, slave, INT of master as trigger, one count mode, start trigger effective */
```

- TAUAn channel mode user register (TAUAnCMURm)

This register specifies the type of valid edge detection used for the TAUAnTTINm input. In this sample program, the TAUA does not use edge detection function.

- TAUAn channel status register (TAUAnCSRm)

This register indicates the count direction and the overflow status of channel m counter. In this sample program, the TAUA does not use edge detection function.

- TAUAn channel status clear register (TAUAnCSCm)

This registers is a trigger register for clearing the overflow flag TAUAnCSRm.TAUAnOVF of channel m. In this sample program, the TAUA does not use edge detection function.

- TAUAn channel start trigger register (TAUAnTS)

This register enables the counter for each channel.

Access This register can be written in 16-bit units. It is always read as 0000_H.

Address <TAUAn_base_USER> + $1C4_H$

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TS15	TAUAn TS14	TAUAn TS13	TAUAn TS12	TAUAn TS11	TAUAn TS10	TAUAn TS09	TAUAn TS08	TAUAn TS07	TAUAn TS06	TAUAn TS05	TAUAn TS04	TAUAn T03	TAUAn TS02	TAUAn TS01	TAUAn TS00
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 12-233 TAUAnTS register contents

Bit position	Bit name	Function
15 to 0	TAUAnTSm	Enables the counter for channel m: 0: No function 1: Enables the counter and sets TAUAnTE.TAUAnTEm = 1. TAUAnTE.TAUAnTEm = 1 only *enables* counter. Whether the counter *starts* depends on the selected operation mode.

Figure 4.18 TAUAnTS Register Format

Setting example

```
TAUA1TS = 0x0003;                    /* ch 0 & ch 1 count start */
```

- TAUAn channel enable status register (TAUAnTE)

This register indicates whether the counter is enabled/disabled.

Access This register can be read in 16-bit units.

Address <TAUAn_base_USER> + $1C0_H$

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TE15	TAUAn TE14	TAUAn TE13	TAUAn TE12	TAUAn TE11	TAUAn TE10	TAUAn TE09	TAUAn TE08	TAUAn TE07	TAUAn TE06	TAUAn TE05	TAUAn TE04	TAUAn TE03	TAUAn TE02	TAUAn TE01	TAUAn TE00
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 12-234 TAUAnTE register contents

Bit position	Bit name	Function
15 to 0	TAUAnTEm	Indicates whether counter for channel m is enabled or disabled: 0: Counter disabled 1: Counter enabled This bit is set when TAUAnTSSTm (the synchronous channel start trigger signal) trigger input is detected or when TAUAnTS.TAUAnTSm is set. Setting TAUAnTT.TAUAnTTm to 1 resets this bit to 0.

Figure 4.19 TAUAnTE Register Format

- TAUAn channel stop trigger register (TAUAnTT)

This register stops the counter for each channel.

Access This register can be written in 16-bit units. It is always read as 0000_H.

Address <TAUAn_base_USER> + $1C8_H$

Initial Value 0000_H.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TT15	TAUAn TT14	TAUAn TT13	TAUAn TT12	TAUAn TT11	TAUAn TT10	TAUAn TT09	TAUAn TT08	TAUAn TT07	TAUAn TT06	TAUAn TT05	TAUAn TT04	TAUAn TT03	TAUAn TT02	TAUAn TT01	TAUAn TT00
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 12-235 TAUAnTT register contents

Bit position	Bit name	Function
15 to 0	TAUAnTTm	Stops the counter of channel m: 0: No function 1: Stops the counter and reset TAUAnTE.TAUAnTEm. TAUAnCNTm, TAUAnTO.TAUAnTOm, and TAUAnTTOUTm retain their values from before the counter stopped.

Figure 4.20 TAUAnTT Register Format

4.4.4 TAUAn Output Registers

- TAUAn channel output enable register (TAUAnTOE)

This register enables/disables independent channel output mode controlled by software. In this sample program, the TAUA1 is set to PWM output function. Channel 1 in the TAUA1 enables the independent macro output function.

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

Address <TAUAn_base_USER> + $5C_H$

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TOE15	TAUAn TOE14	TAUAn TOE13	TAUAn TOE12	TAUAn TOE11	TAUAn TOE10	TAUAn TOE09	TAUAn TOE08	TAUAn TOE07	TAUAn TOE06	TAUAn TOE05	TAUAn TOE04	TAUAn TOE03	TAUAn TOE02	TAUAn TOE01	TAUAn TOE00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-236 TAUAnTOE register contents

Bit position	Bit name	Function
15 to 0	TAUAnTOEm	Enables/disables Independent Channel Output Mode Controlled by Software: 0: Enables Independent Channel Output Mode Controlled by Software 1: Disables Independent Channel Output Mode Controlled by Software

Figure 4.21 TAUAnTOE Register Format

Setting example

```
TAUA1TOE = 0x0002;          /* ch1 single output permit */
```

- TAUAn channel output mode register (TAUAnTOM)

This register specifies the output mode of each channel. In this sample program, the TAUA1 is set to PWM output function. Channel 1 in the TAUA1 is set to synchronous channel operation mode.

Access This register can be read/written in 16-bit units. Writing is only possible while the counter is stopped (TAUAnTE.TAUAnTEm = 0).

Address <TAUAn_base_USER> + 248_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TOM15	TAUAn TOM14	TAUAn TOM13	TAUAn TOM12	TAUAn TOM11	TAUAn TOM10	TAUAn TOM09	TAUAn TOM08	TAUAn TOM07	TAUAn TOM06	TAUAn TOM05	TAUAn TOM04	TAUAn TOM03	TAUAn TOM02	TAUAn TOM01	TAUAn TOM00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-237 TAUAnTOM register contents

Bit position	Bit name	Function
15 to 0	TAUAnTOMm	Specifies the channel output mode: 0: Independent channel output mode 1: Synchronous channel output mode The output mode depends on several channel output control bits, as can be seen in *Table 12-9 "Channel output modes" on page 578*.

Figure 4.22 TAUAnTOM Register Format

Setting example

```
TAUA1TOM = 0x0002;          /* channel co-work mode */
```

- TAUAn channel output configuration register (TAUAnTOC)

This register specifies the output mode of each channel in combination with TAUAnTOMm. In this sample program, the TAUA1 is set to PWM output function. Channel 1 in the TAUA1 is set to synchronous channel operation mode 1.

Access This register can be read/written in 16-bit units. Writing is only possible while the counter is stopped (TAUAnTE.TAUAnTEm = 0).

Address <TAUAn_base_OSbase_USER>> + $24C_H$

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TOC15	TAUAn TOC14	TAUAn TOC13	TAUAn TOC12	TAUAn TOC11	TAUAn TOC10	TAUAn TOC09	TAUAn TOC08	TAUAn TOC07	TAUAn TOC06	TAUAn TOC05	TAUAn TOC04	TAUAn TOC03	TAUAn TOC02	TAUAn TOC01	TAUAn TOC00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-238 TAUAnTOC register contents

Bit position	Bit name	Function
15 to 0	TAUAn TOCm	Specifies the output mode: 0: Operation mode 1 1: Operation mode 2 The output mode also depends on TAUAnTOM.TAUAnTOMm, as can be seen in the following table.

TOMm	TOCm	Description
0	0	Toggle mode: TAUAnTTOUTm toggles when INTTAUAnIm occurs.
	1	Set/reset mode: TAUAnTTOUTm set when INTTAUAnIm occurs upon count start and reset when INTTAUAnIm occurs due to detection of a match between TAUAnCNTm and TAUAnCDRm.
1	0	Synchronous Channel Operation Mode 1: TAUAnTTOUTm set when INTTAUAnI occurs on the master channel and reset when INTTAUAnI occurs on the slave channel.
	1	Synchronous Channel Operation Mode 2: TAUAnTTOUTm set when INTTAUAnIm occurs while the slave channel is counting down and reset when INTTAUAnIm occurs while the slave channel is counting up.

Figure 4.23 TAUAnTOC Register Format

Setting example

```
TAUA1TOC = 0x0000;          /* ch1 co-word output mode 1 */
```

- TAUAn channel dead time output enable register (TAUAnTDE)

This register enables/disables the dead time operation for each channel. In this sample program, the TAUA disables dead time operation.

Access This register can be read/written in 16-bit units. Writing is only possible while the counter is stopped (TAUAnTE.TAUAnTEm = 0).

Address <TAUAn_base_OS> + 250_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TDE15	TAUAn TDE14	TAUAn TDE13	TAUAn TDE12	TAUAn TDE11	TAUAn TDE10	TAUAn TDE09	TAUAn TDE08	TAUAn TDE07	TAUAn TDE06	TAUAn TDE05	TAUAn TDE04	TAUAn TDE03	TAUAn TDE02	TAUAn TDE01	TAUAn TDE00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-239 TAUAnTDE register contents

Bit position	Bit name	Function
15 to 0	TAUAnTDEm	Enables/disables dead time control operation of channel m: 0: Disables dead time operation 1: Enables dead time operation The same settings must be set for the even and the odd slave channel that comprise a set. These bits only apply when: • TAUAnTOE.TAUAnTOEm, TAUAnTOM.TAUAnTOMm, and TAUAnTOC.TAUAnTOCm = 1.

Figure 4.24 TAUAnTDE Register Format

Setting example

```
TAUA1TDE = 0x0000;          /* dead time prohibit */
```

- TAUAn channel dead time output mode register (TAUAnTDM)

This register specifies when dead time is added during dead time output. This sample program does not use the dead time function.

- TAUAn channel dead time output level register (TAUAnTDL)

This register selects the phase period to which the dead time is added. This sample program does not use the dead time function.

- TAUAn channel real-time output enable register (TAUAnTRE)

This register enables/disables real-time output. This program disables real-time output.

Access This register can be read/written in 16-bit units. Writing is only possible while TAUAnTE.TAUAnTEm is 0.

Address <TAUAn_base_OS> + 258_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TRE15	TAUAn TRE14	TAUAn TRE13	TAUAn TRE12	TAUAn TRE11	TAUAn TRE10	TAUAn TRE09	TAUAn TRE08	TAUAn TRE07	TAUAn TRE06	TAUAn TRE05	TAUAn TRE04	TAUAn TRE03	TAUAn TRE02	TAUAn TRE01	TAUAn TRE00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-242 TAUAnTRE register contents

Bit position	Bit name	Function
15 to 0	TAUAnTREm	Enables or disables real-time output of channel m. 0: Disables real-time output 1: Enables real-time output These bits only apply when TAUTnTOE.TAUAnTOEm = 1. When TAUAnTRE.TAUAnTREm = 0, TAUAnTTOUTm is not affected by real-time output. When TAUAnTRE.TAUAnTREm = 1, TAUAnTTOUTm outputs the value of the real-time output bit TAUAnTRO.TAUAnTROm, dependent on the timer operation.

Figure 4.25 TAUAnTRE Register Format

Setting example

```
TAUA1TRE = 0x0000;          /* real time output prohibit */
```

- TAUAn channel real-time output control register (TAUAnTRC)

This register controls the real-time output trigger for each channel. This sample program does not use the real-time output function.

- TAUAn channel real-time output register (TAUAnTRO)

This register specifies the value output to TAUAnTTOUTm. This sample program does not use the real-time output function.

- TAUAn channel modulation output enable register (TAUAnTME)

This register enables/disables modulation output for the timer output and real-time output. This sample program does not use the real-time output function.

4.4.5 TAUAn Channel Output Level Registers

- TAUAn channel output register (TAUAnTO)

This register specifies and reads the level of TAUAnTTOUTm. In this sample program, the functions of specifying and reading the level of TAUAnTTOUTm are not used.

- TAUAn channel output level register (TAUAnTOL)

This register specifies the output logic of the channel output bit (TAUAnTO.TAUAnTOm). In this sample program, the level of TAUAnTTOUTm is set to positive logic.

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

Address <TAUAn_base_USER> + 40_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn TOL15	TAUAn TOL14	TAUAn TOL13	TAUAn TOL12	TAUAn TOL11	TAUAn TOL10	TAUAn TOL09	TAUAn TOL08	TAUAn TOL07	TAUAn TOL06	TAUAn TOL05	TAUAn TOL04	TAUAn TOL03	TAUAn TOL02	TAUAn TOL01	TAUAn TOL00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-247 TAUAnTOL register contents

Bit position	Bit name	Function
15 to 0	TAUAnTOLm	Specifies the output logic of the channel m output bit (TAUAnTO.TAUAnTOm): 0: Positive logic (active high) 1: Inverted logic (active low)

Figure 4.26 TAUAnTOL Register Format

Setting example

```
TAUA1TOL = 0x0000;          /* positive logic */
```

4.4.6 TAUAn Simultaneous Rewrite Registers

- TAUAn channel reload data enable register (TAUAnRDE)

This register enables/disables simultaneous rewrite of data registers TAUAnCDRm and TAUAnTOLm. In this sample program, the TAUA generates the PWM signal in synchronous channel operation. Simultaneous rewrite is enabled in channels 0 and 1 in the TAUA1.

Access This register can be read/written in 16-bit units. Writing is only possible while TAUAnTE.TAUAnTEm is 0.

Address <TAUAn_base_OS> + 260_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn RDE15	TAUAn RDE14	TAUAn RDE13	TAUAn RDE12	TAUAn RDE11	TAUAn RDE10	TAUAn RDE09	TAUAn RDE08	TAUAn RDE07	TAUAn RDE06	TAUAn RDE05	TAUAn RDE04	TAUAn RDE03	TAUAn RDE02	TAUAn RDE01	TAUAn RDE00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-248 TAUAnRDE register contents

Bit position	Bit name	Function
15 to 0	TAUAnRDEm	Enables/disables simultaneous rewrite of the data register of channel m: 0: Disables simultaneous rewrite 1: Enabled simultaneous rewrite

Figure 4.27 TAUAnRDE Register Format

Setting example

```
TAUA1RDE = 0x0003;          /* cocurrent rewrite permit */
```

- TAUAn channel reload data control channel select register (TAUAnRDS)

This register selects the channel that controls simultaneous rewrite. In this sample program, the master channel is set as a channel that monitors the simultaneous rewrite trigger in the TAUA1.

Access This register can be read/written in 16-bit or 1-bit units. Writing is only possible while TAUAnTE.TAUAnTEm is 0.

Address <TAUAn_base_OS> + 268_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn RDS15	TAUAn RDS14	TAUAn RDS13	TAUAn RDS12	TAUAn RDS11	TAUAn RDS10	TAUAn RDS09	TAUAn RDS08	TAUAn RDS07	TAUAn RDS06	TAUAn RDS05	TAUAn RDS04	TAUAn RDS03	TAUAn RDS02	TAUAn RDS01	TAUAn RDS00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-249 TAUAnRDS register contents

Bit position	Bit name	Function
15 to 0	TAUAnRDSm	Specifies which channel is monitored for the simultaneous rewrite trigger: 0: Master channel 1: Another upper channel

Figure 4.28 TAUAnRDS Register Format0

Setting example

```
TAUA1RDS = 0x0000;          /* master trigger monitor */
```

- TAUAn channel reload data mode register (TAUAnRDM)

This register determines when the simultaneous rewrite control signal is generated. In this sample program, the simultaneous rewrite control signal is set as the signal that is generated when the counter in the master channel starts counting.

Access	This register can be read/written in 16-bit units. Writing is only possible while TAUAnTE.TAUAnTEm is 0.
Address	<TAUAn_base_OS> + 264_H
Initial Value	0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUAn RDM15	TAUAn RDM14	TAUAn RDM13	TAUAn RDM12	TAUAn RDM11	TAUAn RDM10	TAUAn RDM09	TAUAn RDM08	TAUAn RDM07	TAUAn RDM06	TAUAn RDM05	TAUAn RDM04	TAUAn RDM03	TAUAn RDM02	TAUAn RDM01	TAUAn RDM00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12-250 TAUAnRDM register contents

Bit position	Bit name	Function
15 to 0	TAUAnRDMm	Selects when the signal that triggers simultaneous is generated: 0: When the master channel counter starts counting 1: At the top of a triangle wave cycle These bits only apply when TAUAnRDE.TAUAnRDEm = 1 and TAUAnRDS.RDSm = 0.

Figure 4.29 TAUAnRDM Register Format

Setting example

```
TAUA1RDM = 0x0000;          /* cocurrent at beginning of master count */
```

- TAUAn channel reload data control register (TAUAnRDC)

This register specifies the channel in which the INTTAUAnIm signal that triggers simultaneous rewrite is generated. This sample program does not use the TAUAnRDC register because TAUAnRDS.TAUAnRDSm is set to 0.

- TAUAn channel reload data trigger register (TAUAnRDT)

This register triggers the simultaneous rewrite pending state. This sample program does not use this function.

- TAUAn channel reload status register (TAUAnRSF)

This flag register indicates the simultaneous rewrite status. This sample program does not use this function.

4.4.7 TAUJn Prescaler Registers

- TAUJn prescaler clock select register (TAUJnTPS)

This register specifies the CK0, CK1, CK2, and CK3_PRE prescaler clocks for all channels. CK3 is generated by the factor specified in TAUJnBRS.

In this sample program, CK0 is specified.

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

Address <TAUJn_base> + 90_H

Initial Value $FFFF_H$. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUJnPRS3[3:0]				TAUJnPRS2[3:0]				TAUJnPRS1[3:0]				TAUJnPRS0[3:0]			
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13-56 TAUJnTPS register contents (1/3)

Bit position	Bit name	Function
15 to 12	TAUJnPRS3[3:0]	Specifies the CK3_PRE clock. Clock CK3_PRE is the input clock of the BRG unit. The BRG unit supplies the CK3 operation clock for all channels.

TAUJnPRS3[3:0]	CK3_PRE clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

These bits can only be rewritten when all counters using CK3 are stopped (TAUJnTE.TAUJnTEm = 0).

Figure 4.30 TAUJnTPS Register Format (1/3)

Table 13-56 TAUJnTPS register contents (2/3)

Bit position	Bit name	Function
11 to 8	TAUJnPRS2[3:0]	Specifies the CK2 clock. PRS2[3:0] / CK2 clock 0000_B / $PCLK/2^0$ 0001_B / $PCLK/2^1$ 0010_B / $PCLK/2^2$ 0011_B / $PCLK/2^3$ 0100_B / $PCLK/2^4$ 0101_B / $PCLK/2^5$ 0110_B / $PCLK/2^6$ 0111_B / $PCLK/2^7$ 1000_B / $PCLK/2^8$ 1001_B / $PCLK/2^9$ 1010_B / $PCLK/2^{10}$ 1011_B / $PCLK/2^{11}$ 1100_B / $PCLK/2^{12}$ 1101_B / $PCLK/2^{13}$ 1110_B / $PCLK/2^{14}$ 1111_B / $PCLK/2^{15}$ These bits can only be rewritten when all counters using CK2 are stopped (TAUJnTE.TAUJnTEm = 0).
7 to 4	TAUJnPRS1[3:0]	Specifies the CK1 clock. PRS1[3:0] / CK1 clock 0000_B / $PCLK/2^0$ 0001_B / $PCLK/2^1$ 0010_B / $PCLK/2^2$ 0011_B / $PCLK/2^3$ 0100_B / $PCLK/2^4$ 0101_B / $PCLK/2^5$ 0110_B / $PCLK/2^6$ 0111_B / $PCLK/2^7$ 1000_B / $PCLK/2^8$ 1001_B / $PCLK/2^9$ 1010_B / $PCLK/2^{10}$ 1011_B / $PCLK/2^{11}$ 1100_B / $PCLK/2^{12}$ 1101_B / $PCLK/2^{13}$ 1110_B / $PCLK/2^{14}$ 1111_B / $PCLK/2^{15}$ These bits can only be rewritten when all counters using CK1 are stopped (TAUJnTE.TAUJnTEm = 0).

Figure 4.31 TAUJnTPS Register Format (2/3)

Table 13-56 TAUJnTPS register contents (3/3)

Bit position	Bit name	Function
3 to 0	TAUJnPRS0[3:0]	Specifies the CK0 clock.

PRS0[3:0]	CK0 clock
0000_B	$PCLK/2^0$
0001_B	$PCLK/2^1$
0010_B	$PCLK/2^2$
0011_B	$PCLK/2^3$
0100_B	$PCLK/2^4$
0101_B	$PCLK/2^5$
0110_B	$PCLK/2^6$
0111_B	$PCLK/2^7$
1000_B	$PCLK/2^8$
1001_B	$PCLK/2^9$
1010_B	$PCLK/2^{10}$
1011_B	$PCLK/2^{11}$
1100_B	$PCLK/2^{12}$
1101_B	$PCLK/2^{13}$
1110_B	$PCLK/2^{14}$
1111_B	$PCLK/2^{15}$

These bits can only be rewritten when all counters using CK0 are stopped (TAUJnTE.TAUJnTEm = 0).

Note The TAUJn clock input PCLK is specified in the first section of this chapter under the keyword "Clock supply".

Figure 4.32 TAUJnTPS Register Format (3/3)

Setting example

```
TAUJ0TPS = 0x0000;                    /* CK0:PCLK / 2^0 */
```

- TAUJn prescaler baudrate value register (TAUJnBRS)

This register specifies the division factor of prescaler clock CK3.

CK3 is generated by dividing CK3_PRE by the factor specified in this register plus one. The PCLK prescaler for CK3_PRE is specified in TAUJnTPS.TAUJnPRS3[3:0].

This register does not use CK3. Setting this register is unnecessary.

4.4.8 TAUJn Control Registers

- TAUJn channel data register (TAUJnCDRm)

This register functions either as a compare register or as a capture register, depending on the operation mode specified in TAUJnCMORm.TAUJnMD[4:1].

In this sample program, the TAUJ0TTIN0 signal width is estimated by a combination of the values of TAUJ0CDR0 and TAUJ0CSR0.TAUJnOVF.

Access This register can be read/written in 32-bit units.

- In capture mode, only reading is possible. Write operation is ignored.
- In compare mode, reading and writing is possible.

Address <TAUJn_base> + m x 4_H

Initial Value $0000\ 0000_H$. This register is initialized by any reset.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TAUJnCDR[31:16]															
R/W															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUJnCDR[15:0]															
R/W															

Table 13-58 TAUJnCDRm register contents

Bit position	Bit name	Function
31 to 0	TAUJnCDR[31:0]	Data register for the capture/compare value

Figure 4.33 TAUJnCDRm Register Format

Setting example

```
pulse_width = (overflow_flag * (0xffffffff + 1)) + TAUJ0CDR0 + 1;
                                                    /* calculate the width of input pulse */
```

- TAUJn channel counter register (TAUJnCNTm)

This register is the channel m counter register.

Access This register can be read in 32-bit units.

Address <TAUJn_base> + 10_H + m x 4_H

Initial Value 0000 0000_H or FFFF $FFFF_H$. The initial value depends on the operation mode, see *Table 13-60 "TAUJnCNTm read values after the counter is re-enabled" on page 991*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TAUJnCNT[31:16]															
R															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUJnCNT[15:0]															
R															

Table 13-59 TAUJnCNTm register contents

Bit position	Bit name	Function
31 to 0	TAUJnCNT[31:0]	32-bit counter value

The read value depends on the counter, the operation mode change, and the values of the TAUJnTS.TAUJnTSm and TAUJnTT.TAUJnTTm bits.

The *initial* counter read value depends on the operation mode and how the counter was stopped:

- by a reset
- by a counter stop trigger (TAUJnTT.TAUJnTTm = 1)

The following table lists the initial counter read values after the counter has stopped (TAUJnTE.TAUJnTEm= 0) and re-enabled (TAUJnTS.TAUJnTSm = 1).

The table also contains the counter read value one count after the counter is enabled (TAUJnTS.TAUJnTSm = 1) for modes where the counter waits for a start trigger.

Figure 4.34 TAUJnCNTm Register Format

Table 13-60 TAUJnCNTm read values after the counter is re-enabled

Mode name	Count method (up/down)	TAUJnCNTm value		
		When operation mode is changed after reset	After stop trigger	After one count
Interval Timer mode	Count down	FFFF FFFF$_H$	Stop value	-
Capture mode	Count up	0000 0000$_H$	Stop value	-
One Count mode	Count down	FFFF FFFF$_H$	Stop value	FFFF FFFF$_H$
Capture & One Count mode	Count up	0000 0000$_H$	Stop value	Captured value + 1 (TAUJnCDRm)
Count Capture Mode	Count up	0000 0000$_H$	Stop value	-
Gate Count Mode	Count down	FFFF FFFF$_H$	Stop value	Stop value
Capture & Gate Count Mode	Count up	0000 0000$_H$	Stop value	Stop value

Note If the operation mode is changed while the counter is stopped, the initial counter value after counter restart is undefined. The operation mode is changed by the TAUJnCMORm.TAUJnMD[4:1] bits.

Figure 4.35 TAUJnCNTm read values

- TAUJn channel mode OS register (TAUJnCMORm)

This register controls channel m operation.

In this sample program, channel 0 in the TAUJ0 is set to capture & one count mode and disables the start trigger during operation by using the valid edge of the TAUJ0TTIN0 input signal as an external start trigger and the reverse edge as a stop trigger.

Access This register can be read or written in 16-bit units. Writing is only possible while the counter is stopped (TAUJnTE.TAUJnTEm = 0).

Address <TAUJn_base> + 80_H + m x 4_H

Initial Value 0000_H. This register is initialized by any reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAUJn CKS[1:0]		TAUJn CCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJn COS[1:0]		-	TAUJnMD[4:0]				
R/W		R/W		R/W	R/W			R/W		R	R/W				

Table 13-61 TAUJnCMORm register contents (1/3)

Bit position	Bit name	Function
15,14	TAUJn CKS[1:0]	Selects the prescaler output. The prescaler output is used for the TAUJnTTINm input edge detection circuit. It can also be used as the count clock depending on bits TAUJnCMORm.CCS[1:0]. TAUJn CKS1 / TAUJn CKS0 / Selected prescaler output 0 / 0 / CK0 0 / 1 / CK1 1 / 0 / CK2 1 / 1 / CK3
13,12	TAUJn CCS[1:0]	Selects the count clock for TAUJnCNTm counter. TAUJn CCS1 / TAUJn CCS0 / Selected count clock 0 / 0 / Prescaler output specified by TAUJnCMORm.TAUJnCKS[1:0] 0 / 1 / Valid edge of TAUJnTTINm input signal 1 / 0 / Setting prohibited 1 / 1
11	TAUJnMAS	Specifies the channel as master or slave channel during synchronous channel operation. 0: Slave 1: Master This bit is only valid for even channels (CHm_even). For odd channels (CHm_odd), it is fixed to 0.

Figure 4.36 TAUJnCMORm Register Format (1/3)

Table 13-61 TAUJnCMORm register contents (2/3)

<table>
<tr><th>Bit position</th><th>Bit name</th><th>Function</th></tr>
<tr><td>10 to 8</td><td>TAUJn STS[2:0]</td><td>Selects the external start trigger.
<table>
<tr><th>TAUJn STS2</th><th>TAUJn STS1</th><th>TAUJn STS0</th><th>Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Software trigger</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Valid edge of the TAUJnTTINm input signal. TAUJnCMURm.TAUJnTIS[1:0] specifies the valid edge.</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Valid edge of the TAUJnTTINm input signal is used as the start trigger, and the reverse edge is used as the stop trigger.</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Setting prohibited</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>INT of the master channel</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Setting prohibited</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td></td></tr>
<tr><td>1</td><td>1</td><td>1</td><td></td></tr>
</table></td></tr>
<tr><td>7, 6</td><td>TAUJn COS[1:0]</td><td>Specifies when the capture register TAUJnCDRm and the overflow flag TAUJnCSRm.TAUJnOVF of channel m are updated.
These bits are only valid if channel m is in capture mode.
<table>
<tr><th>TAUJn COS1</th><th>TAUJn COS0</th><th>TAUJnCDRm</th><th>TAUJnCSRm.TAUJnOVF</th></tr>
<tr><td>0</td><td>0</td><td rowspan="2">Updated upon detection of a TAUJnTTINm input valid edge.</td><td>Updated (cleared or set) upon detection of a TAUJnTTINm input valid edge:
• If a counter overflow has occurred since the last valid edge detection, TAUJnCSRm.TAUJnOVF is set.
• If no counter overflow has occurred since the last valid edge detection, TAUJnCSRm.TAUJnOVF is cleared.</td></tr>
<tr><td>0</td><td>1</td><td>Set upon counter overflow and cleared by setting TAUJnCSCm.TAUJnCLOV.</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">Updated upon detection of a TAUJnTTINm input valid edge and upon counter overflow:
• TAUJnTTINm input valid edge: Counter value is written to TAUJnCDRm.
• Overflow: FFFF FFFF$_H$ is loaded to TAUJnCDRm.
The next TAUJnTTINm input valid edge detection is ignored.</td><td>Not set.</td></tr>
<tr><td>1</td><td>1</td><td>Set upon counter overflow and cleared by setting TAUJnCSCm.TAUJnCLOV.</td></tr>
</table></td></tr>
</table>

Figure 4.37 TAUJnCMORm Register Format (2/3)

Table 13-61 TAUJnCMORm register contents (3/3)

Bit position	Bit name	Function
4 to 0	TAUJn MD[4:0]	Specifies the operation mode.

TAUJnMD4	TAUJnMD3	TAUJnMD2	TAUJnMD1	TAUJnMD0	Description
0	0	0	0	1/0	Interval Timer mode
0	0	0	1	1/0	Setting prohibited
0	0	1	0	1/0	Capture mode
0	0	1	1	1/0	Setting prohibited
0	1	0	0	1/0	One Count mode
0	1	0	1	1/0	Setting prohibited
0	1	1	0	0	Capture & One Count mode
0	1	1	1	1/0	Setting prohibited
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1	1/0	Count Capture mode
1	1	0	0	0	Gate Count mode
1	1	0	1	0	Capture & Gate Count mode

Mode	Role of the MD0 bit
Interval Timer mode Capture mode Count Capture mode	Specifies whether the INTTAUJnIm signal is output when the counter starts counting (when the start trigger is input). 0: Does not output INTTAUJnIm. 1: Outputs INTTAUJnIm
One Count mode Gate Count mode	Enables/disables start trigger detection during counting: 0: Disables 1: Enables
Capture & One Count mode Capture & Gate Count mode	This bit must be set to 0.

Figure 4.38 TAUJnCMORm Register Format (3/3)

Setting example

```
TAUJ0CMOR0 = 0x020c;
                /* CK0, TTIN trigger edge count, capture and one count mode, no int at start */
```

- TAUJn channel mode user register (TAUJnCMURm)

This register specifies the type of valid edge detection used for the TAUJnTTINm input.

In this sample program, the TAUJ is set to single operation signal width measurement and measures the width of the pulse input from TAUJ0TTIN0. When both edges are detected, the high level width is measured by regarding the start trigger as the rising edge and the stop trigger as the falling edge.

Access	This register can be read/written in 8-bit units.
Address	<TAUJn_base> + 20_H + m x 4_H
Initial Value	00_H. This bit is initialized by any reset.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	TAUJnTIS[1:0]	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13-62 TAUJnCMURm register contents

Bit position	Bit name	Function
1, 0	TAUJnTIS[1:0]	Specifies the valid edge of the TAUJnTTINm input.

TAUJn TIS1	TAUJn TIS0	Description
0	0	Falling edge
0	1	Rising edge
1	0	Rising and falling edges (low-width measurement selection). Start trigger: falling edge Stop trigger (capture): rising edge
1	1	Rising and falling edges (high-width measurement selection). Start trigger: rising edge Stop trigger (capture): falling edge

- Edge detection for TAUJnTTINm input signals is performed based on the prescaler output selected by TAUJnCMORm.TAUJnCKS[1:0].

Figure 4.39 TAUJnCMURm Register Format

Setting example

```
TAUJ0CMUR0 = 0x0003;                /* detect high width*/
```

- TAUJn channel status register (TAUJnCSRm)

This register indicates the count direction and the overflow status of channel m counter.

Access This register can be read in 8-bit units.

Address <TAUJn_base> + 30_H + m x 4_H

Initial Value 00_H. This bit is initialized by any reset.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TAUJnOVF
R	R	R	R	R	R	R	R

Table 13-63 TAUJnCSRm register contents

Bit position	Bit name	Function
0	TAUJnOVF	Indicates the counter overflow status: 0: No overflow occurred 1: Overflow occurred This bit is only used in the following modes: • Capture mode • Capture & One Count mode • Count Capture mode • Capture & Gate Count mode The function of this bit depends on the setting of control bits TAUJnCMORm.TAUJnCOS[1:0].

Figure 4.40 TAUJnCSRm Register Format

- TAUJn channel status clear register (TAUJnCSCm)

This registers is a trigger register for clearing the overflow flag TAUJnCSRm.TAUJnOVF of channel m.

Access	This register can be written in 8-bit units. It is always read as 00_H.
Address	<TAUJn_base> + 40_H + m x 4_H
Initial Value	00_H. This bit is initialized by any reset.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	0	TAUJnCLOV
R	R	R	R	R	R	R	W

Table 13-64 TAUJnCSCm register contents

Bit position	Bit name	Function
0	TAUJnCLOV	0: No function 1: Clears the overflow flag TAUJnCSRm.TAUJnOVF

Figure 4.41 TAUJnCSCm Register Format

- TAUJn channel start trigger register (TAUJnTS)

This register enables the counter for each channel.

Access	This register can be written in 8-bit units. It is always read as 00_H.
Address	<TAUJn_base> + 54_H
Initial Value	00_H. This bit is initialized by any reset.

7	6	5	4	3	2	1	0
-	-	-	-	TAUJnTS03	TAUJnTS02	TAUJnTS01	TAUJnTS00
W	W	W	W	W	W	W	W

Table 13-65 TAUJnTS register contents

Bit position	Bit name	Function
3 to 0	TAUJnTSm	Enables the counter for channel m. 0: No function 1: Enables the counter and sets TAUJnTE.TAUJnTEm = 1. TAUJnTE.TAUJnTEm = 1 only *enables* counter. Whether the counter *starts* depends on the selected operation mode.

Figure 4.42 TAUJnTS Register Format

Setting example

```
TAUJ0TS = 0x0001;                    /* ch 0 count start */
```

- TAUJn channel enable status register (TAUJnTE)

This register indicates whether the counter is enabled/disabled.

Access This register can be read in 8-bit units.

Address <TAUJn_base> + 50_H

Initial Value 00_H. This bit is initialized by any reset.

7	6	5	4	3	2	1	0
-	-	-	-	TAUJnTE03	TAUJnTE02	TAUJnTE01	TAUJnTE00
R	R	R	R	R	R	R	R

Table 13-66 TAUJnTE register contents

Bit position	Bit name	Function
3 to 0	TAUJnTEm	Indicates whether counter for channel m is enabled or disabled. 0: Counter disabled 1: Counter enabled This bit is set when TAUJnTSSTm (the synchronous channel start trigger signal) trigger input is detected or when TAUJnTS.TAUJnTSm is set. Setting TAUJnTT.TAUJnTTm to 1 resets this bit to 0.

Figure 4.43 TAUJnTE Register Format

- TAUJn channel stop trigger register (TAUJnTT)

This register stops the counter for each channel.

Access This register can be written in 8-bit units. It is always read as 00_H.

Address <TAUJn_base> + 58_H

Initial Value 00_H.

7	6	5	4	3	2	1	0
-	-	-	-	TAUJnTT03	TAUJnTT02	TAUJnTT01	TAUJnTT00
W	W	W	W	W	W	W	W

Table 13-67 TAUJnTT register contents

Bit position	Bit name	Function
3 to 0	TAUJnTTm	Stops the counter of channel m. 0: No function 1: Stop the counter and reset TAUJnTE.TAUJnTEm. TAUJnCNTm, TAUJnTO.TAUJnTOm, and TAUJnTTOUTm retain their values from before the counter stopped.

Figure 4.44 TAUJnTT Register Format

4.4.9 TAUJn Output Registers

- TAUJn channel output enable register (TAUJnTOE)

This register enables or disables independent channel output mode controlled by software. In this sample program, the TAUJ is set to signal width measurement function. The TAUJ disables the independent macro output function.

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

Address <TAUJn_base> + 60_H

Initial Value 00_H. This bit is initialized by any reset.

7	6	5	4	3	2	1	0
-	-	-	-	TAUJnTOE03	TAUJnTOE02	TAUJnTOE01	TAUJnTOE00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13-68 TAUJnTOE register contents

Bit position	Bit name	Function
3 to 0	TAUJnTOEm	Enables or disables the independent timer output function. 0: Disables the independent timer output function. 1: Enables the independent timer output function.

Figure 4.45 TAUJnTOE Register Format

Setting example

```
TAUJ0TOE = 0x0000;                    /* channel single mode */
```

- TAUJn channel output mode register (TAUJnTOM)

This register specifies the output mode of each channel. In this sample program, the TAUJ does not use the output function.

- TAUJn channel output configuration register (TAUJnTOC)

This register specifies the output mode of each channel in combination with TAUJnTOMm. In this sample program, the TAUJ does not use the output function.

4.4.10 TAUJn Channel Output Level Registers

- TAUJn channel output register (TAUJnTO)

This register specifies and reads the level of TAUJnTTOUTm. In this sample program, the TAUJ does not use the output function.

- TAUJn channel output level register (TAUJnTOL)

This register specifies the output logic of the channel output bit (TAUJnTO.TAUJnTOm). In this sample program, the TAUJ does not use the output function.

4.4.11 TAUJn Simultaneous Rewrite Registers

- TAUJn channel reload data enable register (TAUJnRDE)

This register enables/disables simultaneous rewrite of data registers TAUJnCDRm and TAUJnTOLm. In this sample program, the TAUJ does not use the simultaneous rewrite function.

- TAUJn channel reload data mode register (TAUJnRDM)

This register determines when the simultaneous rewrite control signal is generated. In this sample program, the TAUJ does not use the simultaneous rewrite function.

- TAUJn channel reload data trigger register (TAUJnRDT)

This register triggers the simultaneous rewrite pending state. In this sample program, the TAUJ does not use the simultaneous rewrite function.

- TAUJn channel reload status register (TAUJnRSF)

This flag register indicates the simultaneous rewrite status. In this sample program, the TAUJ does not use the simultaneous rewrite function.

4.5 Function Specifications

This section describes the specifications for the functions that are used by the sample program.

4.5.1 Main (main.c)

[Function Name]	main ()
[Function]	Calls necessary initialization functions before entering an infinite loop.
[Arguments]	None
[Return Value]	None
[Startup Method]	Enters the main function after hardware initialization.
[SFRs Used]	None
[Calling Function]	None
[Variables]	None
[File Name]	main.c
[Notes]	None

4.5.2 Initialization Processing (initial.c)

[Function Name]	port_initial()
[Function]	Sets up ports and their mode.
[Arguments]	None
[Return Value]	None
[Startup Method]	Call
[SFRs Used]	PFCE2, PFC2, PMC2, PM2, PFCE3, PFC3, PMC3, PM3
[Calling Function]	main()
[Variables]	None
[File Name]	initial.c
[Notes]	None

4.5.3 Timer Array Unit A Control (taua_control.c)

[Function Name]	taua1_initial()
[Function]	Sets up the synchronous PWM output mode.
[Arguments]	None
[Return Value]	None
[Startup Method]	Call
[SFRs Used]	TAUA1TPS, TAUA1BRS, TAUA1CMOR0, TAUA1CMUR0, TAUA1CMOR1, TAUA1CMUR1,TAUA1CDR0, TAUA1CDR1, TAUA1TOE, TAUA1TOM, TAUA1TOC, TAUA1TOL, TAUA1TDE, TAUA1TDM, TAUA1TDL, TAUA1TRE, TAUA1TRO, TAUA1TRC, TAUA1TME, TAUA1RDE, TAUA1RDS, TAUA1RDM, TAUA1RDC, TAUA1RDT, TAUA1TS
[Calling Function]	main()
[Variables]	None
[File Name]	taua_control.c
[Notes]	None

4.5.4 Timer Array Unit J Control (tauj_control.c)

[Function Name]	tauj0_initial()
[Function]	Sets up the independent signal width measurement mode.
[Arguments]	None
[Return Value]	None
[Startup Method]	Call
[SFRs Used]	TAUJ0TPS, TAUJ0BRS, TAUJ0CMOR0, TAUJ0CMUR0, TAUJ0TOE, TAUJ0TOM, TAUJ0TOC, TAUJ0TOL, TAUJ0RDE, TAUJ0RDM, TAUJ0RDT, ICTAUJ0I0, TAUJ0TS
[Calling Function]	main()
[Variables]	None
[File Name]	tauj_control.c
[Notes]	None

4.5.5 Interrupt Processing (interrupt.c)

[Function Name]	int_tauj0i0()
[Function]	End of input signal width measurement
[Arguments]	None
[Return Value]	None
[Startup Method]	Request INTTAUJ0I0 is present in an unmasked state.
[SFRs Used]	TAUJ0CSR0, TAUJ0CDR0
[Calling Function]	None
[Variables]	overflow_flag, pulse_width
[File Name]	interrupt.c
[Notes]	None

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/inquiry

Revision Record

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Jan 13, 2012	—	First edition issued

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins
   Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.
   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment when power is supplied.
   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.
   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.
5. Differences between Products
   Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.
   — The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
   - "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
   - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
   - "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.1