32-Bit Microcontroller FM3 Family
Peripheral Manual Timer Part

Doc. No. 002-04794 Rev. *A

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone (USA): 800.858.1810
Phone (Intnl): 408.943.2600
http://www.cypress.com

How to Use This Manual

Finding a Function

The following methods can be used to search for the explanation of a desired function in this manual:

Search from the table of the contents

The table of the contents lists the manual contents in the order of description.

Search from the register

The address where each register is located is not described in the text. To verify the address of a register, see "A. Register Map" in "Appendixes".

About the Chapters

Basically, this manual explains Timer Part..

Terminology

This manual uses the following terminology.

Term	Explanation
Word	Indicates access in units of 32 bits.
Half word	Indicates access in units of 16 bits.
Byte	Indicates access in units of 8 bits.

Notations

The notations in bit configuration of the register explanation of this manual are written as follows.

bit:	bit number
Field:	bit field name
Attribute:	Attributes for read and write of each bit
R:	Read only
W:	Write only
R/W:	Readable/Writable
-:	Undefined
Initial value:	Initial value of the register after reset
0:	Initial value is 0
1:	Initial value is 1
X:	Initial value is undefined

The multiple bits are written as follows in this manual.

Example: bit7:0 indicates the bits from bit7 to bit0

The values such as for addresses are written as follows in this manual.

Hexadecimal number:	"0x" is attached in the beginning of a value as a prefix (example: 0xFFFF)
Binary number:	"0b" is attached in the beginning of a value as a prefix (example: 0b1111)
Decimal number:	Written using numbers only (example: 1000)

The target products in this manual

In this manual, the products are classified into the following groups and are described as follows. For the descriptions such as "TYPE0", see the relevant items of the target product in the list below.

Table 1 TYPE0 Product list

Description in this manual	Flash memory size			
	512 Kbytes	384 Kbytes	256 Kbytes	128 Kbytes
TYPE0	MB9BF506N MB9BF506R MB9BF506NA MB9BF506RA MB9BF506NB MB9BF506RB	MB9BF505N MB9BF505R MB9BF505NA MB9BF505RA MB9BF505NB MB9BF505RB	MB9BF504N MB9BF504R MB9BF504NA MB9BF504RA MB9BF504NB MB9BF504RB	-
	MB9BF406N MB9BF406R MB9BF406NA MB9BF406RA	MB9BF405N MB9BF405R MB9BF405NA MB9BF405RA	MB9BF404N MB9BF404R MB9BF404NA MB9BF404RA	-
	MB9BF306N MB9BF306R MB9BF306NA MB9BF306RA MB9BF306NB MB9BF306RB	MB9BF305N MB9BF305R MB9BF305NA MB9BF305RA MB9BF305NB MB9BF305RB	MB9BF304N MB9BF304R MB9BF304NA MB9BF304RA MB9BF304NB MB9BF304RB	-
	MB9BF106N MB9BF106R MB9BF106NA MB9BF106RA	MB9BF105N MB9BF105R MB9BF105NA MB9BF105RA	MB9BF104N MB9BF104R MB9BF104NA MB9BF104RA	MB9BF102N MB9BF102R MB9BF102NA MB9BF102RA
	-	MB9AF105N MB9AF105R MB9AF105NA MB9AF105RA	MB9AF104N MB9AF104R MB9AF104NA MB9AF104RA	MB9AF102N MB9AF102R MB9AF102NA MB9AF102RA

Table 2 TYPE1 Product list

Description in this manual	Flash memory size				
	512 Kbytes	384 Kbytes	256 Kbytes	128 Kbytes	64 Kbytes
TYPE1	MB9AF316M MB9AF316N MB9AF316MA MB9AF316NA	MB9AF315M MB9AF315N MB9AF315MA MB9AF315NA	MB9AF314L MB9AF314M MB9AF314N MB9AF314L MB9AF314M MB9AF314N	MB9AF312L MB9AF312M MB9AF312N MB9AF312LA MB9AF312MA MB9AF312NA	MB9AF311L MB9AF311M MB9AF311N MB9AF311LA MB9AF311MA MB9AF311NA
	MB9AF116M MB9AF116N MB9AF116MA MB9AF116NA	MB9AF115M MB9AF115N MB9AF115MA MB9AF115NA	MB9AF114L MB9AF114M MB9AF114N MB9AF114LA MB9AF114MA MB9AF114NA	MB9AF112L MB9AF112M MB9AF112N MB9AF112LA MB9AF112MA MB9AF112NA	MB9AF111L MB9AF111M MB9AF111N MB9AF111LA MB9AF111MA MB9AF111NA

Table 3 TYPE2 Product list

Description in this manual	Flash memory size		
	1 Mbytes	768 Kbytes	512 Kbytes
TYPE2	MB9BFD18S MB9BFD18T	MB9BFD17S MB9BFD17T	MB9BFD16S MB9BFD16T
	MB9BF618S MB9BF618T	MB9BF617S MB9BF617T	MB9BF616S MB9BF616T
	MB9BF518S MB9BF518T	MB9BF517S MB9BF517T	MB9BF516S MB9BF516T
	MB9BF418S MB9BF418T	MB9BF417S MB9BF417T	MB9BF416S MB9BF416T
	MB9BF318S MB9BF318T	MB9BF317S MB9BF317T	MB9BF316S MB9BF316T
	MB9BF218S MB9BF218T	MB9BF217S MB9BF217T	MB9BF216S MB9BF216T
	MB9BF118S MB9BF118T	MB9BF117S MB9BF117T	MB9BF116S MB9BF116T

Table 4 TYPE3 Product list

Description in this manual	Flash memory size	
	128 Kbytes	64 Kbytes
TYPE3	MB9AF132K MB9AF132L	MB9AF131K MB9AF131L
	MB9AF132KA MB9AF132LA	MB9AF131KA MB9AF131LA
	MB9AF132KB MB9AF132LB	MB9AF131KB MB9AF131LB

Table 5 TYPE4 Product list

Description in this manual	Flash memory size			
	512 Kbytes	384 Kbytes	256 Kbytes	128 Kbytes
TYPE4	MB9BF516N MB9BF516R	MB9BF515N MB9BF515R	MB9BF514N MB9BF514R	MB9BF512N MB9BF512R
	MB9BF416N MB9BF416R	MB9BF415N MB9BF415R	MB9BF414N MB9BF414R	MB9BF412N MB9BF412R
	MB9BF316N MB9BF316R	MB9BF315N MB9BF315R	MB9BF314N MB9BF314R	MB9BF312N MB9BF312R
	MB9BF116N MB9BF116R	MB9BF115N MB9BF115R	MB9BF114N MB9BF114R	MB9BF112N MB9BF112R

Table 6 TYPE5 Product list

Description in this manual	Flash memory size	
	128 Kbytes	64 Kbytes
TYPE5	MB9AF312K	MB9AF311K
	MB9AF112K	MB9AF111K

●

Table 7 TYPE6 product list

Description in this manual	Flash memory size		
	256 Kbytes	**128 Kbytes**	**64 Kbytes**
TYPE6	MB9AFB44L	MB9AFB42L	MB9AFB41L
	MB9AFB44M	MB9AFB42M	MB9AFB41M
	MB9AFB44N	MB9AFB42N	MB9AFB41N
	MB9AFB44LA	MB9AFB42LA	MB9AFB41LA
	MB9AFB44MA	MB9AFB42MA	MB9AFB41MA
	MB9AFB44NA	MB9AFB42NA	MB9AFB41NA
	MB9AFB44LB	MB9AFB42LB	MB9AFB41LB
	MB9AFB44MB	MB9AFB42MB	MB9AFB41MB
	MB9AFB44NB	MB9AFB42NB	MB9AFB41NB
	MB9AFA44L	MB9AFA42L	MB9AFA41L
	MB9AFA44M	MB9AFA42M	MB9AFA41M
	MB9AFA44N	MB9AFA42N	MB9AFA41N
	MB9AFA44LA	MB9AFA42LA	MB9AFA41LA
	MB9AFA44MA	MB9AFA42MA	MB9AFA41MA
	MB9AFA44NA	MB9AFA42NA	MB9AFA41NA
	MB9AFA44LB	MB9AFA42LB	MB9AFA41LB
	MB9AFA44MB	MB9AFA42MB	MB9AFA41MB
	MB9AFA44NB	MB9AFA42NB	MB9AFA41NB
	MB9AF344L	MB9AF342L	MB9AF341L
	MB9AF344M	MB9AF342M	MB9AF341M
	MB9AF344N	MB9AF342N	MB9AF341N
	MB9AF344LA	MB9AF342LA	MB9AF341LA
	MB9AF344MA	MB9AF342MA	MB9AF341MA
	MB9AF344NA	MB9AF342NA	MB9AF341NA
	MB9AF344LB	MB9AF342LB	MB9AF341LB
	MB9AF344MB	MB9AF342MB	MB9AF341MB
	MB9AF344NB	MB9AF342NB	MB9AF341NB
	MB9AF144L	MB9AF142L	MB9AF141L
	MB9AF144M	MB9AF142M	MB9AF141M
	MB9AF144N	MB9AF142N	MB9AF141N
	MB9AF144LA	MB9AF142LA	MB9AF141LA
	MB9AF144MA	MB9AF142MA	MB9AF141MA
	MB9AF144NA	MB9AF142NA	MB9AF141NA
	MB9AF144LB	MB9AF142LB	MB9AF141LB
	MB9AF144MB	MB9AF142MB	MB9AF141MB
	MB9AF144NB	MB9AF142NB	MB9AF141NB

Table 8 TYPE7 product list

Description in this manual	Flash memory size	
	128 Kbytes	64 Kbytes
TYPE7	MB9AFA32L MB9AFA32M MB9AFA32N	MB9AFA31L MB9AFA31M MB9AFA31N
	MB9AF132M MB9AF132N	MB9AF131M MB9AF131N
	MB9AFAA2L MB9AFAA2M MB9AFAA2N	MB9AFAA1L MB9AFAA1M MB9AFAA1N
	MB9AF1A2M MB9AF1A2N	MB9AF1A1M MB9AF1A1N

Table 9 TYPE8 product list

Description in this manual	Flash memory size		
	512 Kbytes	384 Kbytes	256 Kbytes
TYPE8	MB9AF156M MB9AF156N MB9AF156R MB9AF156MA MB9AF156NA MB9AF156RA MB9AF156MB MB9AF156NB MB9AF156RB	MB9AF155M MB9AF155N MB9AF155R MB9AF155MA MB9AF155NA MB9AF155RA MB9AF155MB MB9AF155NB MB9AF155RB	MB9AF154M MB9AF154N MB9AF154R MB9AF154MA MB9AF154NA MB9AF154RA MB9AF154MB MB9AF154NB MB9AF154RB

Table 10 TYPE9 product list

Description in this manual	Flash memory size		
	256 Kbytes	128 Kbytes	64 Kbytes
TYPE9	MB9BF524K MB9BF524L MB9BF524M	MB9BF522K MB9BF522L MB9BF522M	MB9BF521K MB9BF521L MB9BF521M
	MB9BF324K MB9BF324L MB9BF324M	MB9BF322K MB9BF322L MB9BF322M	MB9BF321K MB9BF321L MB9BF321M
	MB9BF124K MB9BF124L MB9BF124M	MB9BF122K MB9BF122L MB9BF122M	MB9BF121K MB9BF121L MB9BF121M

Table 11 TYPE10 product list

Description in this manual	Flash memory size
	64 Kbytes
TYPE10	MB9BF121J

Table 12 TYPE11 product list

Description in this manual	Flash memory size
	64 Kbytes
TYPE11	MB9AF421K MB9AF421L
	MB9AF121K MB9AF121L

Table 13 TYPE12 product list

Description in this manual	Flash memory size	
	1.5 Mbytes	1 Mbytes
TYPE12	MB9BF529S MB9BF529T MB9BF529SA MB9BF529TA	MB9BF528S MB9BF528T MB9BF528SA MB9BF528TA
	MB9BF429S MB9BF429T MB9BF429SA MB9BF429TA	MB9BF428S MB9BF428T MB9BF428SA MB9BF428TA
	MB9BF329S MB9BF329T MB9BF329SA MB9BF329TA	MB9BF328S MB9BF328T MB9BF328SA MB9BF328TA
	MB9BF129S MB9BF129T MB9BF129SA MB9BF129TA	MB9BF128S MB9BF128T MB9BF128SA MB9BF128TA

Contents

CHAPTER 1: Watchdog Timer

This chapter explains the watchdog timer.

CODE: 9BFMFS-E02.0

1. Overview

This section explains the overview of the watchdog timer.

The watchdog timer is a function to detect runaway of user program.

If the watchdog timer is not cleared within the specified interval time, it judges that a user program is out of control, and outputs a system reset request or an interrupt request to CPU.

This interrupt request is called a watchdog interrupt request, and a reset request is called a watchdog reset request.

During watchdog timer operation, it is required to clear continually and periodically before the specified interval time has elapsed. If an abnormal operation of user program such as hanging up prevents it from being periodically cleared, it continues down counting, underflows and outputs a watchdog interrupt request or a watchdog reset request.

This MCU has two kinds of watchdog timers as follows.

● Software watchdog timer

- The software watchdog timer is activated by user program.
- A divided clock of APB bus clock is used for a count clock.
- It counts cycles while CPU program is operating, and it stops counting while APB clock is stopped in the standby mode (timer mode, stop mode, and during oscillation stabilization wait time of the source clock). The count value is retained so that it continues counting after returning from the standby mode.
- The software watchdog timer is stopped by all the resets.

● Hardware watchdog timer

- The hardware watchdog timer is activated by tuning on the device, and after releasing all the resets except software resets without an intervention of software.
- The hardware watchdog timer can be stopped by accessing a register by software.
- Low-speed CR clock (CLKLC) is used for a count clock.
- It counts cycles while CLKLC is operated, and it stops counting while CLKLC is stopped in the standby mode (stop mode). The count value is retained so that it continues counting after returning from the standby mode.

● Software/hardware watchdog timer

- Each watchdog timer has a lock register, accessing to all the registers of watchdog timers cannot be done unless accessing and releasing a lock with a certain procedure.
- The watchdog timers can be reloaded by accessing to the watchdog clear register.
- When the first underflow of the watchdog counter is generated, an interrupt request is generated. When the second underflow is generated without clearing the interrupt request, a reset request is generated. This function can be set by the register.

2. Configuration and Block Diagram

This section shows the block diagram of the watchdog timer.

Figure 2-1 Block Diagram of Software Watchdog Timer

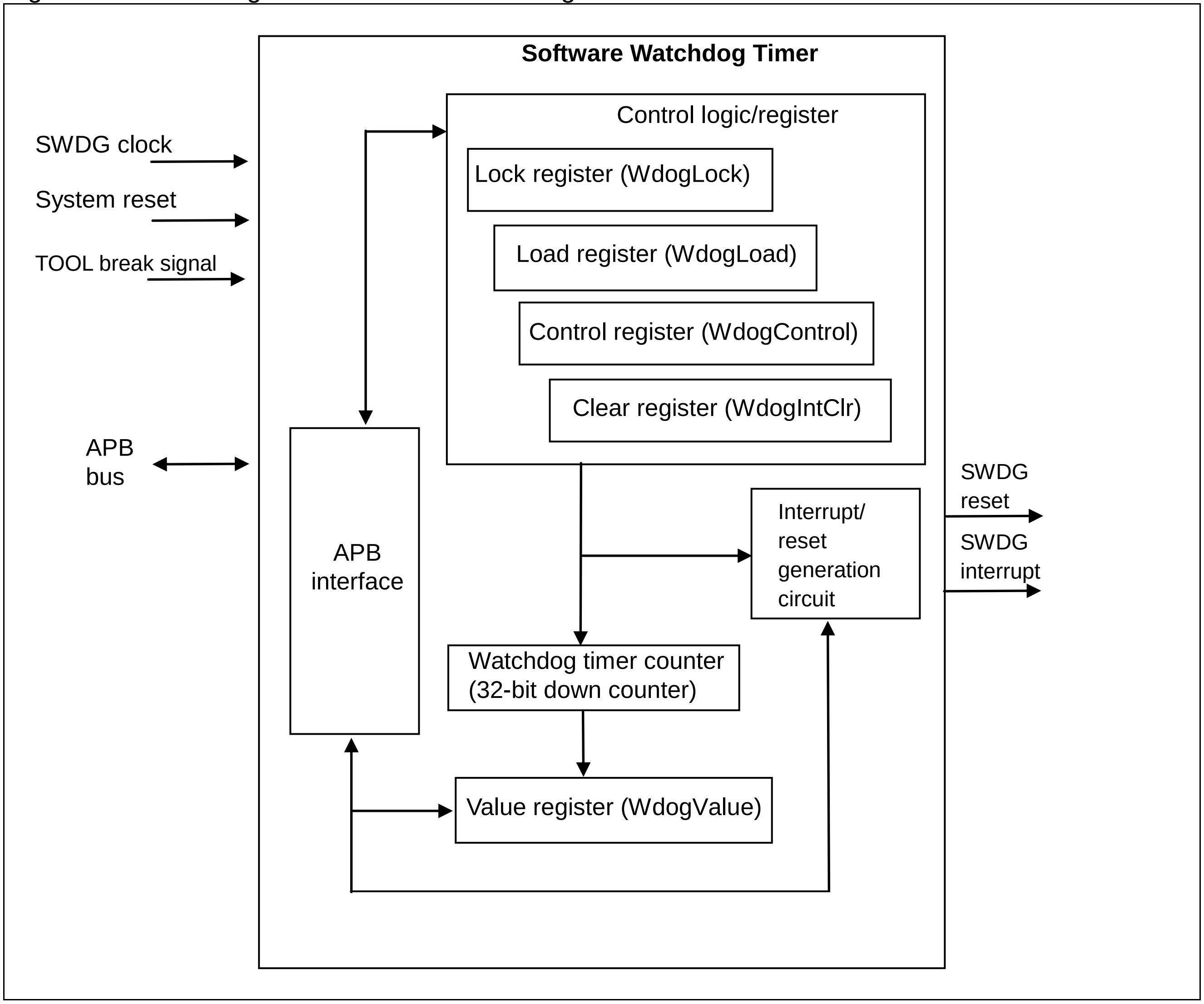
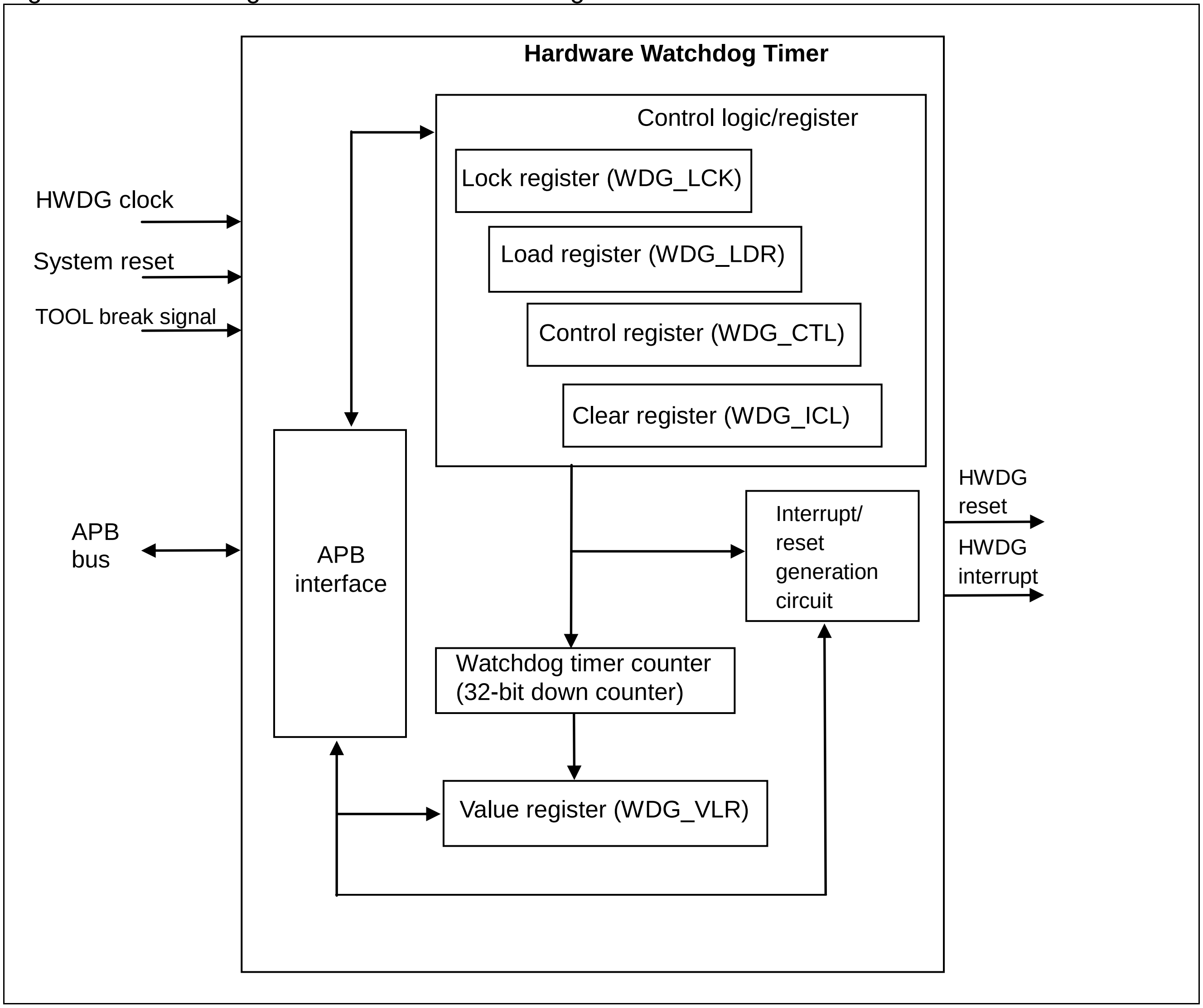


Figure 2-2 Block Diagram of Hardware Watchdog Timer



3. Operations

This section shows the configuration of the watchdog timer.

The watchdog timer consists of the following blocks.

■ Software Watchdog Timer

● Control register / logic

This circuit controls the software watchdog timer.
It consists of the load register, the lock register, the control register, and the clear register.

- Load register (WdogLoad)
  This register is a 32-bit register used to set count interval cycles of the software watchdog timer. The initial value is "0xFFFFFFFF". Table 3-1 shows the examples of interval time setting.

Table 3-1 Examples of Interval Time Setting of Software Watchdog Timer

Count Frequency	Interval Set Value	Interval Time
40 MHz	"0xFFFFFFFF" [initial value]	Approx. 107 s
20 MHz	"0xFFFFFFFF" [initial value]	Approx. 214 s
40 MHz	"0x0000FFFF"	Approx. 1.6 ms
20 MHz	"0x0000FFFF"	Approx. 3.2 ms

- Lock register (WdogLock)
  This register controls accesses of all the registers of the software watchdog timer.

  Writing a value of "0x1ACCE551" to this register enables write access to all the other registers of the software watchdog timer.

- Control register (WdogControl)
  This register sets an interrupt enable of the software watchdog and a reset enable of the software watchdog.

- Clear register (WdogIntClr)
  This is a clear register of the software watchdog timer.
  Writing any value to the clear register reloads the value set in the load register to the counter. After the reloading is completed, the counting is continued.

● Watchdog Timer Counter (32-bit Down Counter)

This is a 32-bit down counter. The count value is reloaded to the set value of the load register (WdogLoad) by accessing to the clear register (WdogIntClr) before the counter value becomes "0" by decrementing.

Table 3-2 shows the down counter reload condition.

Table 3-2 Down Counter Reload Condition of Software Watchdog Timer

Reload Conditions
Accessing to the clear register (WdogIntClr)
When the 32-bit down counter reaches "0"
When the load register (WdogLoad) is modified
When the watchdog is stopped by writing INTEN=0 to the control register (WdogControl), and reactivated by writing INTEN=1

● Value register (WdogValue)

This register can read the current counter value of the watchdog timer.

● Interrupt and reset generation circuit

When an underflow of the watchdog timer counter is detected, a watchdog interrupt and a watchdog reset are generated due to the register setting.

- Interrupt status register (WdogRIS)
  This register shows the status of a software watchdog interrupt.

● Activation of software watchdog timer

- Access to the control register (WdogControl), and enable the watchdog interrupt and watchdog reset.
- Table 3-3 shows the combination of watchdog interrupt and watchdog reset settings.

Table 3-3 Combination of software watchdog interrupt and reset

Interrupt	Reset	Operation
Disable	Disable	The watchdog timer is not operated
Enable	Disable	An interrupt is generated at underflow
Disable	Enable	The watchdog timer is not operated
Enable	Enable	An interrupt is generated at the first underflow A reset is generated at the second underflow [Initial setting]

Enabling an interrupt of the control register (WdogControl) becomes an activation trigger of the watchdog timer.

● Reload and lock of the register of the software watchdog timer

- The register has not been locked with initial condition after reset. When you wish to enable locking, write any values other than "0x1ACCE551" to the WdogLock register with software.
- When you access the clear register, write "0x1ACCE551" to the WdogLock register to release the lock.
- The value set to the load register (WdogLoad) is reloaded by writing an arbitrary value to the clear register (WdogIntClr).
- After accessed the clear register, it will not be automatically locked. Lock it again with software.

● Halting the software watchdog timer

- The software watchdog timer is stopped by accessing to the control register (WdogControl), and writing "0" to the watchdog interrupt enable bit.
- The software watchdog timer is stopped by generating a reset.

■ Hardware Watchdog Timer

● Control register / logic

This is a circuit to control the hardware watchdog timer.
It consists of the load register, the lock register, the control register, and the clear register.

- Load register (WDG_LDR)
  This register is a 32-bit register used to set count interval cycles of the hardware watchdog timer. The initial value is "0x0000FFFF" (down counter for 16 bits=> approx. 655 ms @ 100 kHz (TYP)). For the frequency of CLKLC which is a count clock, see "Data Sheet" for the product that you are using.

- Lock register (WDG_LCK)
  This register controls the accesses of all the registers of the hardware watchdog timer. Writing a value of "0x1ACCE551" to this register enables write access to all the registers except the control register (WDG_CTL).

- Control register (WDG_CTL)
  This register sets watchdog interrupt enable and watchdog reset enable. To access this register, write "0x1ACCE551" to the lock register, and also write "0xE5331AAE" to the lock register. In case of not writing the correct value after writing "0x1ACCE551", repeat the process from the beginning.

- Clear register (WDG_ICL)
  This is a clear register of the hardware watchdog timer.
  By writing an arbitrary 8-bit value and its reversed value in series, the timer counter is reloaded to the value stored in the load register and its counting is continued.

● Watchdog Timer Counter (32-bit down counter)

This is a 32-bit down counter. The count value is reloaded to the set value of the load register (WDG_LDR) by accessing to the clear register (WDG_ICL) before the counter value becomes "0" by decrementing.

Table 3-4 shows the down counter reload condition.

Table 3-4 Down Counter Reload Condition of Hardware Watchdog Timer

Reload Conditions
Accessing to the clear register (WDG_ICL)
When the 32-bit down counter reaches "0"
When the load register (WDG_LDR) is modified
When the watchdog is stopped by writing INTEN=0 to the control register (WDG_CTL), and reactivated by writing INTEN=1

● Value register (WDG_VLR)

This register can read the current counter value of the watchdog timer. However, during tool break, a correct value can be read when the watchdog timer is stopped. Except during tool break, an inaccurate value may be read due to asynchronous reading. In this case, a countermeasure is necessary such as comparing read values after reading it twice.

● Interrupt and reset generation circuit

When an underflow of the watchdog timer counter is detected, a watchdog interrupt and a watchdog reset are generated due to the register setting.

- Interrupt status register (WDG_RIS)
  This register shows the status of a hardware watchdog interrupt.

● Activation of hardware watchdog timer

- Writing "0x1ACCE551" to the lock register (WDG_LCK) and then writing a reversal value "0xE5331AAE" to it enables to access to the control register (WDG_CTL) also.
- Access to the control register (WDG_CTL), and enable the watchdog interrupt and the watchdog reset. Table 3-5 shows the combination of watchdog interrupt and watchdog reset settings.

Table 3-5 Combination of hardware watchdog interrupt and reset

Interrupt	Reset	Operation
Disable	Disable	The watchdog timer is not operated
Enable	Disable	An interrupt is generated at underflow
Disable	Enable	The watchdog timer is not operated
Enable	Enable	An interrupt is generated at the first underflow A reset is generated at the second underflow [Initial setting]

Enabling an interrupt of the control register (WDG_CTL) becomes an activation trigger of the hardware watchdog timer.

● Reload and lock of the register of the hardware watchdog timer

The set value is reloaded from the load register to the 32-bit down counter by writing a value to the clear register (WDG_ICL). After reloading, the register is locked again.
Unlock is required each time accessing to the clear register hereafter.

● Stopping the hardware watchdog timer

- Writing "0x1ACCE551" to the lock register (WDG_LCK) and then writing a reversal value "0xE5331AAE" to it enables to access to the control register (WDG_CTL).
- The hardware watchdog timer is stopped by accessing to the control register (WDG_CTL), and writing "0" to the watchdog interrupt enable bit.

■ Differences between software watchdog timer and hardware watchdog timer

Table 3-6 shows the major differences between software watchdog timer and hardware watchdog timer.

Table 3-6 Differences between software watchdog timer and hardware watchdog timer

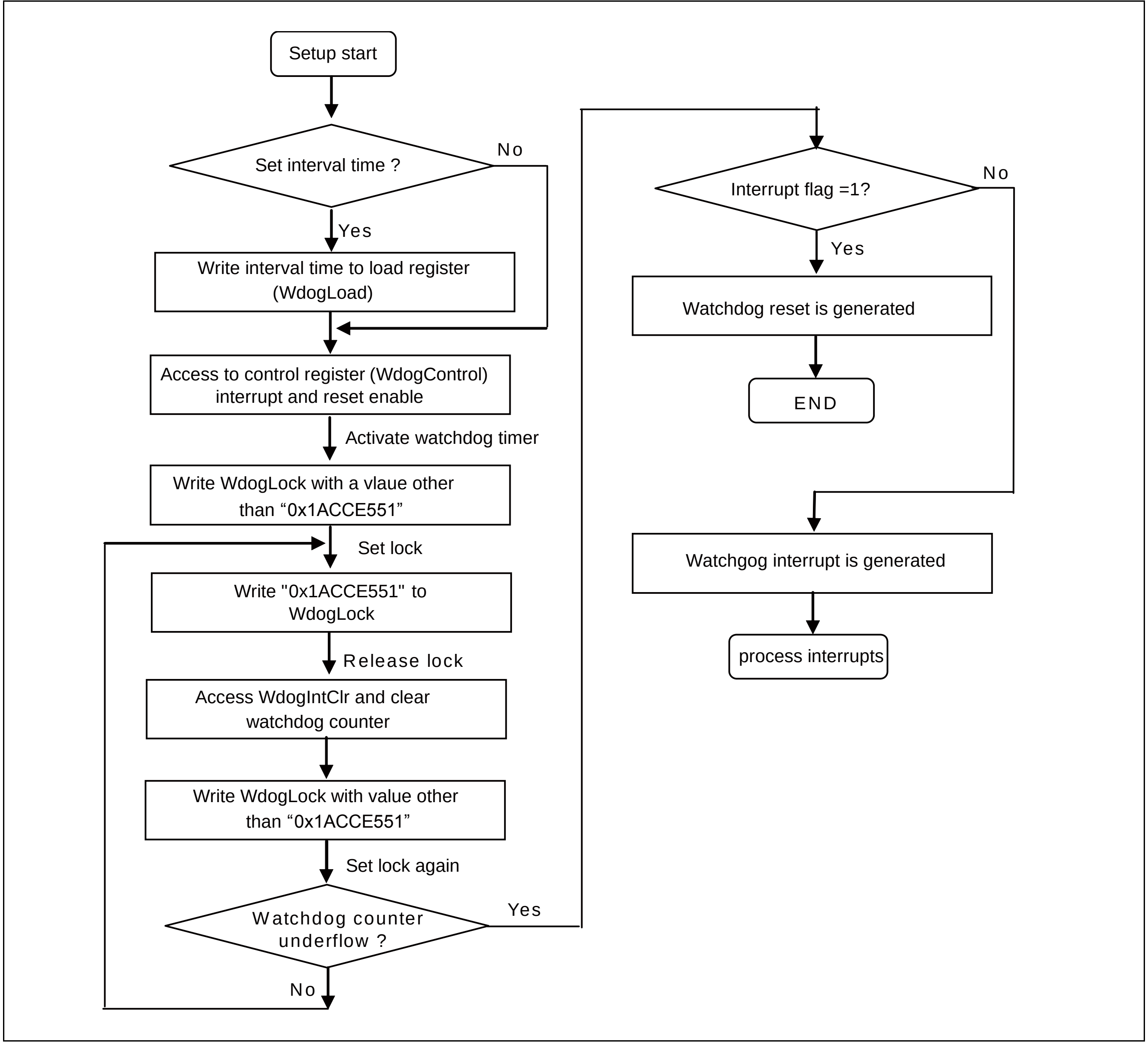
	Software Watchdog	Hardware Watchdog
Count clock	Divided clock of APB	CLKLC
Read value of the value register	Synchronous reading Reading possible	Asynchronous reading Only during tool break, a correct value can be read. Except during tool break, an inaccurate value may be read.
Initial value of watchdog interrupt setting and reset setting	Disable (No watchdog operation)	Enable (With a watchdog operation)
Register lock function initial state	No lock (Software locks after activation)	Lock (Hardware locks from the activation)
Releasing lock	Writing "0x1ACCE551" to lock register to release all lock for the registers	Writing "0x1ACCE551" to lock register to release all lock for the registers except WDG_CTL
WdogControl/ WDG_CTL register Releasing separate lock	None	Writing "0xE5331AAE" to lock register to release lock of WDG_CTL register
Relock conditions	Writing a value other than "0x1ACCE551" to the lock register locks all the registers again.	After releasing lock for the registers except WDG_CTL, the lock is resumed under any of the following conditions: · Writing a value other than "0x1ACCE551" or "0xE5331AAE" to WDG_LCK · Writing to WDG_LDR · Writing to WDG_CTL · Writing to WDG_ICL again
		After releasing lock for the registers including WDG_CTL, lock is resumed under any of the following conditions: · Writing a value other than "0x1ACCE551" to WDG_LCK · Writing to WDG_LDR · Writing to WDG_ICL · Writing to WDG_CTL
Initial value of load register	0xFFFFFFFF	0x0000FFFF
Bit number of clear register	32 bits	8 bits
Clear register access	Clear by writing an arbitrary value	Clear by writing an arbitrary value, and then writing a reversal value of the arbitrary value

4. Setting Procedure Example

This section explains a setting procedure example of the watchdog timer.

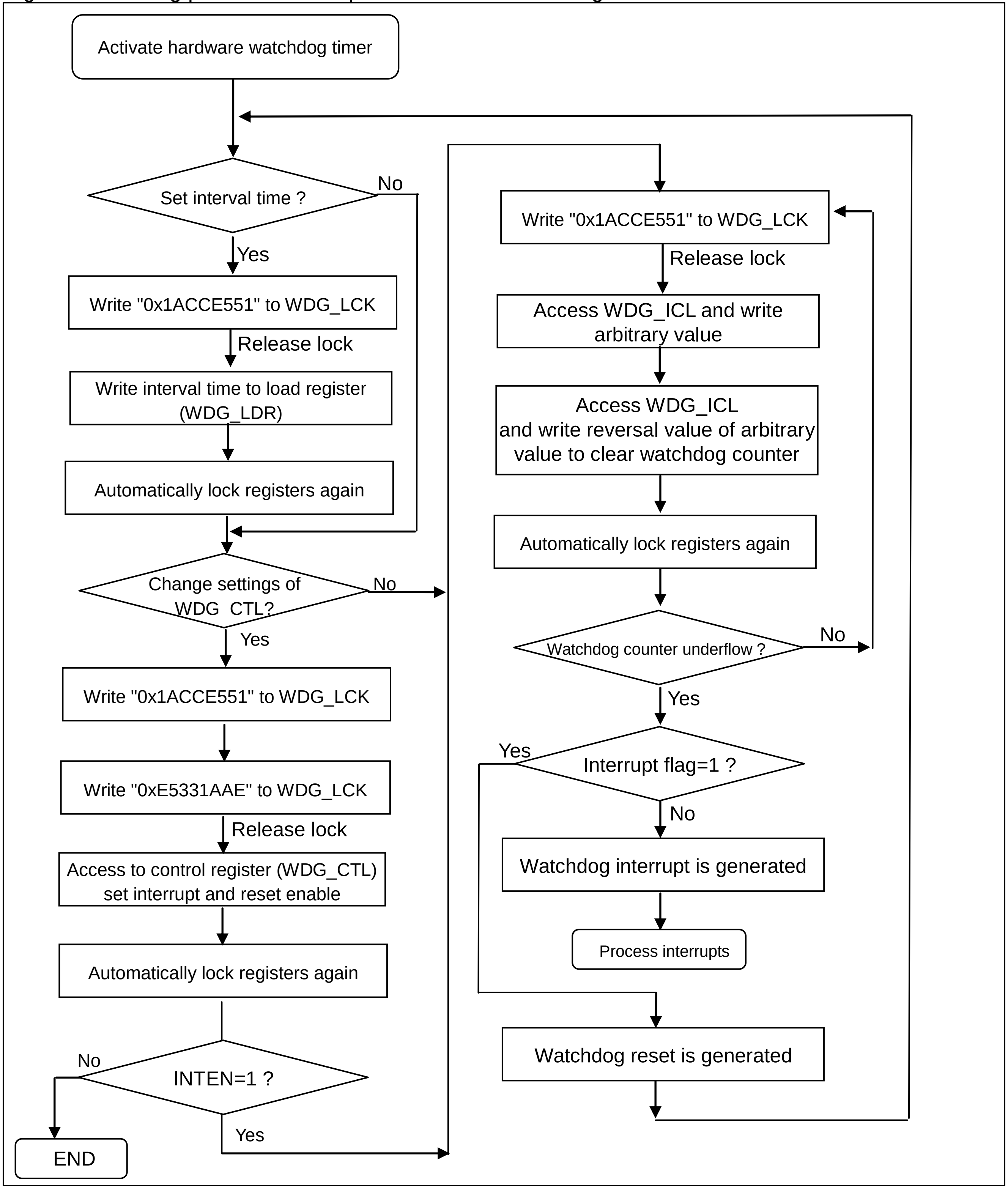
■ Software watchdog timer

Figure 4-1 Setting procedure example of software watchdog timer


Setup start
Set interval time ?
No
Yes
Write interval time to load register (WdogLoad)
Access to control register (WdogControl) interrupt and reset enable
Activate watchdog timer
Write WdogLock with a vlaue other than “0x1ACCE551”
Set lock
Write "0x1ACCE551" to WdogLock
Release lock
Access WdogIntClr and clear watchdog counter
Write WdogLock with value other than “0x1ACCE551”
Set lock again
Watchdog counter underflow ?
Yes
No
Interrupt flag =1?
No
Yes
Watchdog reset is generated
END
Watchgog interrupt is generated
process interrupts

■ Hardware watchdog timer

Figure 4-2 Setting procedure example of hardware watchdog timer

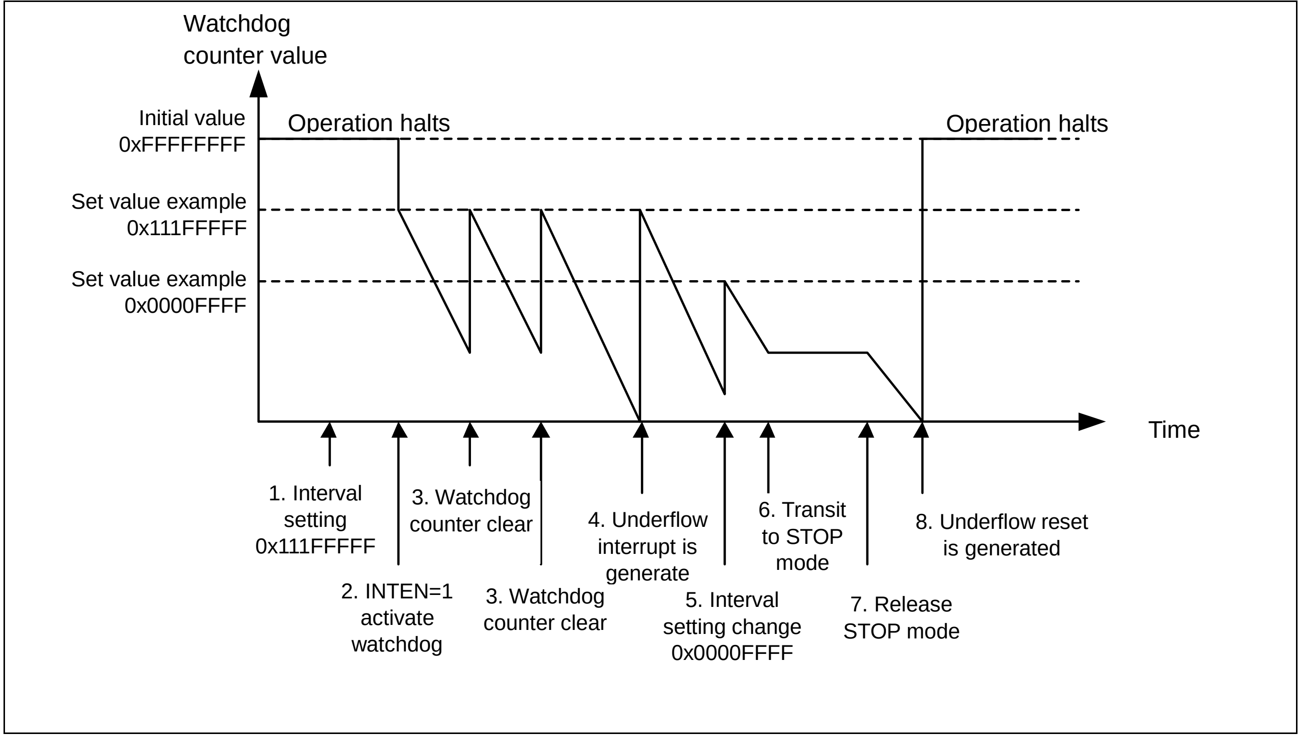

Activate hardware watchdog timer
Set interval time ?
No
Yes
Write "0x1ACCE551" to WDG_LCK
Release lock
Write interval time to load register (WDG_LDR)
Automatically lock registers again
Change settings of WDG_CTL?
No
Yes
Write "0x1ACCE551" to WDG_LCK
Write "0xE5331AAE" to WDG_LCK
Release lock
Access to control register (WDG_CTL) set interrupt and reset enable
Automatically lock registers again
No
INTEN=1 ?
Yes
END
Write "0x1ACCE551" to WDG_LCK
Release lock
Access WDG_ICL and write arbitrary value
Access WDG_ICL and write reversal value of arbitrary value to clear watchdog counter
Automatically lock registers again
Watchdog counter underflow ?
No
Yes
Yes
Interrupt flag=1 ?
No
Watchdog interrupt is generated
Process interrupts
Watchdog reset is generated

5. Operation Example

This section shows an operation example of the watchdog timers.

■ Software watchdog timer

Figure 5-1 Operation example of software watchdog timer



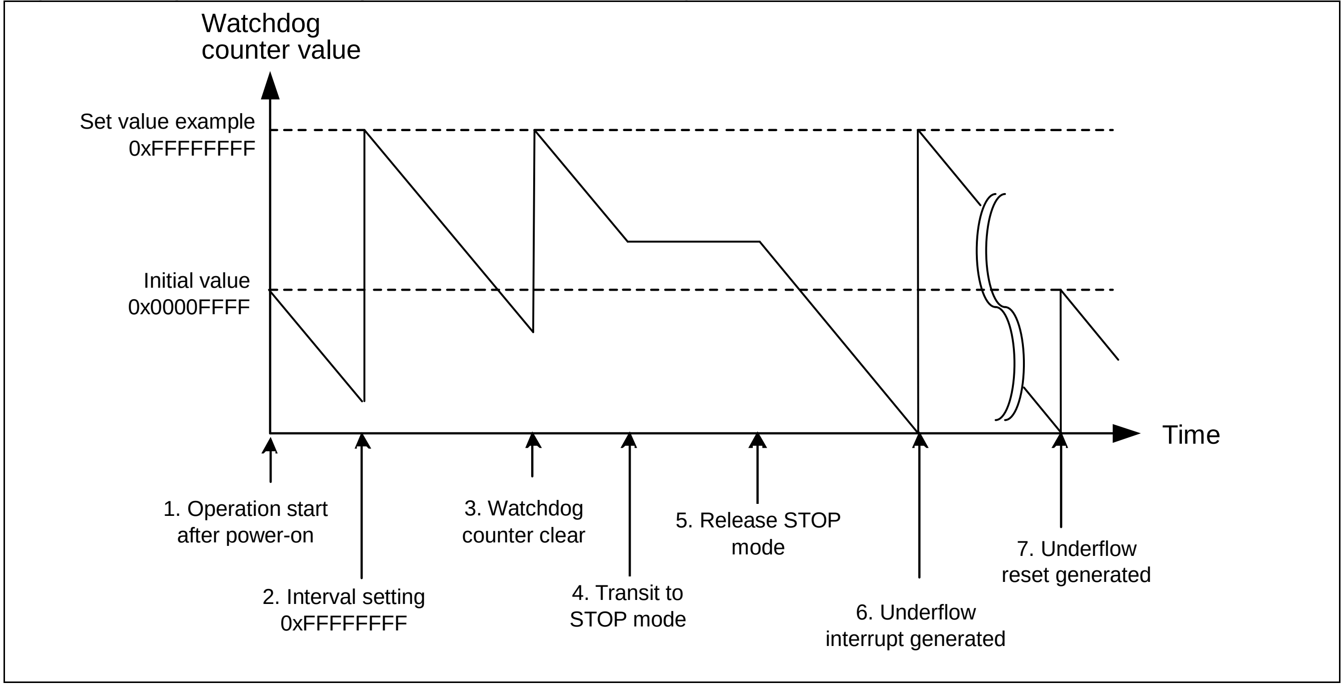
1. Set SWC_PSR, DBWDT_CTL before activation.
   Write to WdogLoad register to set interval time.
   The interval time is not reflected because the watchdog is not activated. The count value is the initial value.
2. Access to WdogControl register, and write INTEN=1 to activate the watchdog.
   At this time, the interval time is reflected and decrementing will be started from the value set in 1.
3. Access to WdogIntClr register, and write an arbitrary value to clear the watchdog counter.
   At this time, the set value will be the value set in 2.
4. Without clearing the counter, an interrupt will be generated at underflow.
   At this time, the set value will be the value set in 2.
5. Access to WdogLoad register to change interval time.
   At this time, the down count value will be cleared to the set value.
6. Transit to STOP mode. The software watchdog will be stopped by this.
7. Release STOP mode. The down counter is restarted. The count value is not cleared.
   (Note) Decrementing will be restarted after oscillation wait stabilization is completed, and the base clock starts its operation.
8. A software watchdog reset will be generated when the second underflow is generated without clearing the interrupt flag by accessing the WdogIntClr register.
   The software watchdog timer stops its operation by generating a reset.

<Note>

Release of the lock register is required to access each register. It is omitted in the operation example.

■ Hardware watchdog timer

Figure 5-2 Operation example of hardware watchdog timer


Watchdog counter value
Set value example 0xFFFFFFFF
Initial value 0x0000FFFF
Time
1. Operation start after power-on
2. Interval setting 0xFFFFFFFF
3. Watchdog counter clear
4. Transit to STOP mode
5. Release STOP mode
6. Underflow interrupt generated
7. Underflow reset generated

1. Hardware watchdog timer starts operation after turning on the power.
   The initial value of the count value is "0x0000FFFF".
2. Access to WDG_LDR register to change interval time.
   At this time, the decremented value will be cleared to the set value.
3. Access to WDG_ICL register, and write an arbitrary value and then write a reversal value of arbitrary value to clear the watchdog counter.
   At this time, the set value will be the value set in 2.
4. Transit to STOP mode. The hardware watchdog will be stopped by this.
5. Release STOP mode. The down counter is restarted. The count value is not cleared.
   (Note) After the CLKLC oscillation starts, HWDG clock is input and the dawn counting restarts.
6. Without clearing the counter, an interrupt will be generated at underflow.
   At this time, the set value will be the value set in 2.
7. A hardware watchdog reset will be generated when the second underflow is generated without clearing the interrupt flag by accessing the WDG_ICL register.
   The count value returns to the initial value and decrementing is started.

<Note>

Release of the lock register is required to access each register. It is omitted in the operation example.

6. Registers

This section explains the registers of clock generation.

Table 6-1 List of registers for the watchdog timer

Abbreviated register name	Register Name	Reference
WdogLoad	Software watchdog timer load register	6.1
WdogValue	Software watchdog timer value register	6.2
WdogControl	Software watchdog timer control register	6.3
WdogIntClr	Software watchdog timer clear register	6.4
WdogRIS	Software watchdog timer interrupt status register	6.5
WdogLock	Software watchdog timer lock register	6.6
WDG_LDR	Hardware watchdog timer load register	6.7
WDG_VLR	Hardware watchdog timer value register	6.8
WDG_CTL	Hardware watchdog timer control register	6.9
WDG_ICL	Hardware watchdog timer clear register	6.10
WDG_RIS	Hardware watchdog timer interrupt status register	6.11
WDG_LCK	Hardware watchdog timer lock register	6.12

6.1. Software Watchdog Timer Load Register (WdogLoad)

The WdogLoad register sets the cycle of the software watchdog timer.

■ Register configuration

bit	31 … 0
Field	WdogLoad
Attribute	R/W
Initial value	0xFFFFFFFF

■ Register function

[bit31:0] WdogLoad : Interval cycle setting bits

Process	Explanation
In case of writing	Sets the cycle of the software watchdog. The initial value is "0xFFFFFFFF". The minimum value for writing is "0x00000001". When "0x00000000" is written, an interrupt will be generated. (A reset may be generated by setting.)
In case of reading	A set value can be read. The initial value "0xFFFFFFFF" is read.

<Notes>

- During watchdog timer operation, if the value of WdogLoad is modified, the value of WdogLoad will be reflected to the timer counter, and counting is continued.
- During the watchdog timer is halting, if the value of WdogLoad is modified, the value of WdogLoad will be reflected to the timer counter at activation of the watchdog timer.

6.2. Software Watchdog Timer Value Register (WdogValue)

The WdogValue register can read the current counter value of the software watchdog timer.

■ Register configuration

bit	31 ... 0
Field	WdogValue
Attribute	R
Initial value	0xFFFFFFFF

■ Register function

[bit31:0] WdogValue : Counter value bits

Process	Explanation
In case of writing	No effect on the operation. .
In case of reading	The count value of the current watchdog counter is read. The initial value "0xFFFFFFFF" is read if reading before activation.

<Note>

See "5.13 Debug Break Watchdog Timer Control Register (DBWDT_CTL)" in the chapter of "Clock" in "Peripheral Manual" for the setting of watchdog timer at tool break.

6.3. Software Watchdog Timer Control Register (WdogControl)

The WdogControl register sets enable/disable of the software watchdog timer.

■ Register configuration

bit	7 2	1	0
Field	Reserved	RESEN	INTEN
Attribute	-	R/W	R/W
Initial value	-	0	0

■ Register function

[bit7:2] Reserved : Reserved bits

"0b000000" is read from these bits.
In case of writing, set "0b000000".

[bit1] RESEN : Reset enable bit of the software watchdog

Process		Explanation
Read		Register value is read.
Write	0	A watchdog reset is disabled.
Write	1	A watchdog reset is enabled.

[bit0] INTEN : Interrupt and counter enable bit of the software watchdog

Process		Explanation
Read		Register value is read.
Write	0	A watchdog interrupt is disabled. A watchdog counter is disabled.
Write	1	A watchdog interrupt is enabled. A watchdog counter is enabled.

<Notes>

- By writing "1" to INTEN bit, the watchdog counter loads an interval cycle value from WdogLoad and the software watchdog timer is activated.
- Writing "0" to INTEN bit stops the watchdog counter. The watchdog counter reloads the cycle value from WdogLoad when "1" is written again and reactivated.
- The watchdog timer can be activated by enabling INTEN bit only. The watchdog timer is not activated by enabling RESEN bit only. To activate the watchdog timer, INTEN bit should be enabled. See "3. Operations" for details.
- Writing "0" to INTEN bit clears the interrupt flag in software watchdog timer interrupt status register (WdogRIS).

6.4. Software Watchdog Timer Clear Register (WdogIntClr)

The WdogIntClr register clears the software watchdog timer.

■ Register configuration

bit	31	0
Field	WdogIntClr	
Attribute	R/W	
Initial value	0xXXXXXXXX	

■ Register function

[bit31:0] WdogIntClr : clear bits

Process	Explanation
Read	An undefined value is read.
Write	Writing an arbitrary value · Clears an interrupt of the watchdog timer, if an interrupt of the watchdog timer is generated. · Reloads the set value from the WdogLoad register to the watchdog timer counter.

6.5. Software Watchdog Timer Interrupt Status Register (WdogRIS)

The WdogRIS register shows the interrupt status of the software watchdog timer.

■ Register configuration

bit	7	1	0
Field	Reserved		RIS
Attribute	-		R
Initial value	-		0

■ Register function

[bit7:1] Reserved : Reserved bits

"0b0000000" is read from these bits.
In case of writing, set "0b0000000".

[bit0] RIS : Software watchdog interrupt status bit

Process		Explanation
Write		No effect on the operation.
Read	0	A watchdog interrupt is not generated.
Read	1	A watchdog interrupt is generated.

6.6. Software Watchdog Timer Lock Register (WdogLock)

The WdogLock register controls accesses of all the registers of software watchdog timer.

■ Register configuration

bit	31	0
Field	WdogLock	
Attribute	R/W	
Initial value	0x00000000	

■ Register function

[bit31:0] WdogLock : Software watchdog lock register

Process	Explanation
Write	Writing "0x1ACCE551": Releases locks of all the registers of software watchdog timer. Writing other than "0x1ACCE551": Lock function for all of the software watchdog timer registers will be enabled.
Read	"0x00000000" : The locks are released. "0x00000001" : The locks are not released.

<Notes>

- Lock for initial values are not enabled. Enable lock function after the software watchdog timer is started.
- After lock is released, the software watchdog timer clear register (WdogIntClr) will become accessible.
- After accessed the clear register (WdogIntClr), lock will not be automatically enabled. Incorporate "lock release -> clear -> lock enable" for any clear sequence.
- In case of accessing to each register of the hardware watchdog when the locks are not released, reading is enabled and the values of each register can be read. Writing is ignored.

6.7. Hardware Watchdog Timer Load Register (WDG_LDR)

The WDG_LDR register sets the cycle of hardware watchdog timer.

■ Register configuration

bit	31	0
Field	WDG_LDR	
Attribute	R/W	
Initial value	0x0000FFFF	

■ Register function

[bit31:0] WDG_LDR : Interval cycle setting bits

Process	Explanation
Write	Sets cycle of the hardware watchdog. The initial value is "0x0000FFFF". The minimum value of writing is "0x00000001". An interrupt is generated when "0x00000000" is written.
Read	A set value can be read. The initial value "0x0000FFFF" is read.

<Notes>

- During watchdog timer operation, if the value of WDG_LDR is modified, the value of WDG_LDR will be reflected to the timer counter and counting is continued.
- During the watchdog timer is halting, if the value of WDG_LDR is modified, the value of WDG_LDR will be reflected to the timer counter at activation of the watchdog timer.
- The case of modifying the WDG_LDR register when the watchdog timer interrupt was occurred, the watchdog timer interrupt is cleared.
- This register cannot be cleared by a software reset or a software watchdog reset.

6.8. Hardware Watchdog Timer Value Register (WDG_VLR)

The WDG_VLR register can read the current counter value of the hardware watchdog timer.

■ Register configuration

bit	31	0
Field	WDG_VLR	
Attribute	R	
Initial value	0xXXXXXXXX	

■ Register function

[bit31:0] WDG_VLR : Counter value bits

Process	Explanation
Read	The count value of the current watchdog counter can be read. By turning on the power, the hardware watchdog automatically activates, therefore decrementing is already started at the time of reading. The value after power on or the value decremented from the initial value "0x0000FFFF" is read.
Write	No effect on the operation.

<Notes>

- This register cannot be cleared by software reset or software watchdog reset.
- Reading a correct value of this register is possible only if the watchdog timer stops at tool break. See "5.13 Debug Break Watchdog Timer Control Register (DBWDT_CTL)" in the chapter of "Clock" in "Peripheral Manual" for the setting of watchdog timer at tool break. Except during tool break, an inaccurate value may be read due to asynchronous reading for the bus clock. In this case, a countermeasure is necessary such as comparing read values after reading it twice.

6.9. Hardware Watchdog Timer Control Register (WDG_CTL)

The WDG_CTL register sets enable/disable of the hardware watchdog timer.

■ Register configuration

bit	7 … 2	1	0
Field	Reserved	RESEN	INTEN
Attribute	-	R/W	R/W
Initial value	-	1	1

■ Register function

[bit7:2] Reserved : Reserved bits

"0b000000" is read from these bits.
In case of writing, set these bits to "0b000000".

[bit1] RESEN : Hardware watchdog reset enable bit

Process		Explanation
Read		A value of register is read.
Write	0	A watchdog reset is disabled.
Write	1	A watchdog reset is enabled.

[bit0] INTEN : Hardware watchdog interrupt and counter enable bit

Process		Explanation
Read		The value of the register is read.
Write	0	A watchdog interrupt is disabled. A watchdog counter is disabled.
Write	1	A watchdog interrupt is enabled. A watchdog counter is enabled.

<Notes>

- Writing "0" to INTEN bit stops the watchdog counter. When writing "1" again, the watchdog counter reloads the cycle value from WDG_LDR register to activate the counter.
- The watchdog timer can be activated by enabling INTEN bit only. The watchdog timer is not activated by enabling RESEN bit only. To activate the watchdog timer, INTEN bit should be enabled.
- To access this register, write "0x1ACCE551" to the hardware watchdog timer lock register (WDG_LCK), and also write the reversal value "0xE5331AAE" to release lock.
- This register cannot be cleared by a software reset or a software watchdog reset
- Writing "0" to INTEN bit clears the interrupt flag in hardware watchdog timer interrupt status register (WDG_RIS).

<Note>

The differences between each series are explained in "1. List of Limitations for TYPE0 Products" of "C. List of Limitations" in "Appendixes". Check the list.

6.10. Hardware Watchdog Timer Clear Register (WDG_ICL)

The WDG_ICL register clears the hardware watchdog timer.

■ Register configuration

bit	7 … 0
Field	WDG_ICL
Attribute	R/W
Initial value	0xXX

■ Register function

[bit7:0] WDG_ICL : clear bits

Process	Explanation
Read	Undefined value is read.
Write	Writing an arbitrary 8-bit value, and then write a reversal value of the arbitrary value, · Clears an interrupt of watchdog timer, if an interrupt of watchdog timer is generated. · Reloads the set value from the WDG_LDR register to the watchdog timer counter.

<Note>

This register cannot be cleared by a software reset or a software watchdog reset.

6.11. Hardware Watchdog Timer Interrupt Status Register (WDG_RIS)

The WDG_RIS register shows the interrupt status of the hardware watchdog timer.

■ Register configuration

bit	7	1	0
Field	Reserved		RIS
Attribute	-		R
Initial value	-		0

■ Register function

[bit7:1] Reserved : Reserved bits

"0b0000000" is read from these bits.

In case of writing, set "0b0000000".

[bit0] RIS : Hardware watchdog interrupt status bit

Process		Explanation
Write		No effect on the operation.
Read	0	Hardware watchdog interrupt is not generated.
Read	1	Hardware watchdog interrupt is generated.

<Note>

This register cannot be cleared by a software reset or a software watchdog reset.

6.12. Hardware Watchdog Timer Lock Register (WDG_LCK)

The WDG_LCK register controls accesses of all the registers of the hardware watchdog timer.

■ Register configuration

bit	31	0
Field	WDG_LCK	
Attribute	R/W	
Initial value	0x00000001	

■ Register function

[bit31:0] WDG_LCK : Hardware watchdog lock register

Process	Explanation
Write	Writing "0x1ACCE551": The locks of all the registers other than the control register are released. Later, in case of writing the reversal value, "0xE5331AAE": The locks of all the registers are released. In case of other procedure is performed or writing any value other than above: The locks of all the registers will be enabled.
Read	"0x00000000" : The locks are released. "0x00000001" : The locks are not released.

<Notes>

- This register cannot be cleared by a software reset or a software watchdog reset.
- In case of accessing to each register of the hardware watchdog when the locks are not released, reading is enabled and the values of each register can be read. Writing is ignored.

7. Usage Precautions

The section explains the precautions when using the watchdog timer.

- Hardware watchdog timer clear register
  To clear the hardware watchdog, write an arbitrary 8-bit value, and then write a reversal value of the arbitrary value. Clearing cannot be performed unless the correct reversal value is written. Even if clearing is not performed, the register is locked again.

- Cooperation with a debug tool
  When a tool break is applied by a debug tool, to continue or stop of the counter of the watchdog timer can be set by setting of the register. See the chapter, "Clock" in "Peripheral Manual" for details about the behavior of the watchdog timers during debugging.

- Operation at standby mode
  Writing to a key register is required at setting of the standby mode not to stop the watchdog timer for the case of the mode is transited to the standby mode because of an unintended program operation.
  See the chapter, "Low Power Consumption Mode" in "Peripheral Manual" for more details.

- Generation of a watchdog reset can be confirmed by the reset source register. See the section "4.1 Reset Cause Register (RST_STR: ReSeT STatus Register)" in the chapter "Reset" in "Peripheral Manual" for more details.

- For an interrupt source, see the section "4.4 EXC02 Batch Read Register (EXC02MON)" and "4.6 IRQ01 Batch Read Register (IRQ01MON)" in the chapter "Interrupts (A)" in "Peripheral Manual".

- For an interrupt source, see the section "4.2 EXC02 Batch Read Register (EXC02MON)" and "4.4 IRQ01 Batch Read Register (IRQ01MON)" in the chapter "Interrupts (B)" in "Peripheral Manual".

- For an interrupt source, see the section "4.1 EXC02 Batch Read Register (EXC02MON)" and "4.3 IRQ01 Batch Read Register (IRQ01MON)" in the chapter "Interrupts(C)" in "Peripheral Manual".

- Use a divided clock of APB clock for the count clock of the software watchdog.
  See the chapter, "Clock" in "PERIPHERAL MANUAL" for divided clock setting of the count clock.

- Hardware watchdog and interrupt handler
  Before releasing the Lock for WDG_CTL (after releasing the Lock for the register other than WDG_CTL), if another interrupt becomes effective by the hardware watchdog and the interrupt handler begins its processing, the Lock releasing count could not be detected by hardware.
  So, at the beginning of the interrupt handler, write values in the WDG_LCK register to lock the register.

CHAPTER 2: Dual Timer

This chapter explains the Dual Timer functions and operations.

CODE: 9BFDT-E02.0_SP804-E01.0

1. Overview

Dual Timer consists of two programmable 32/16-bit down counters. An interrupt is generated when the counter reaches zero.

■ Dual Timer Overview

Dual Timer consists of two programmable Free Run Counters. Each timer block operates identically. The Free Run Counters can be programmed for 32-bit or 16-bit counter size by Control Register. Also, any one of the following three timer modes can be selected:

- Free-running mode
  The counter operates continuously and wraps around to its maximum value each time that it reaches zero.

- Periodic mode
  The counter is reloaded from Load Register and operates continuously each time that it reaches zero.

- One-shot mode
  Writing to the Load Register (TimerXLoad) loads the counter with a new value. The counter halts until it is reprogrammed when the counter reaches zero.

Two Free Run Counters operate in common timer clock (TIMCLK). APB bus clock (PCLK) is used as the timer clock. Also, each Free Run Counter has a prescaler that can divide by 1, 16, or 256. Therefore, the count rate of each Free Run Counter can be controlled by each prescaler.

Writing to the Load Register (TimerXLoad) loads the counter with the timer count value. If the timer counter is enabled, the timer decrements at the rate determined in the timer clock and in the prescaler setting. When the timer counter has been running, writing to the Load Register restarts the counter immediately with a new value.

An alternative way of loading the timer count is to write to Background Load Register (TimerXBGLoad). In this way, the current count value is not affected immediately after the writing, and the counter continues to decrement. Then, in the case where the counter reaches zero, the timer counter is reloaded with a new load value if it is in Periodic Mode.

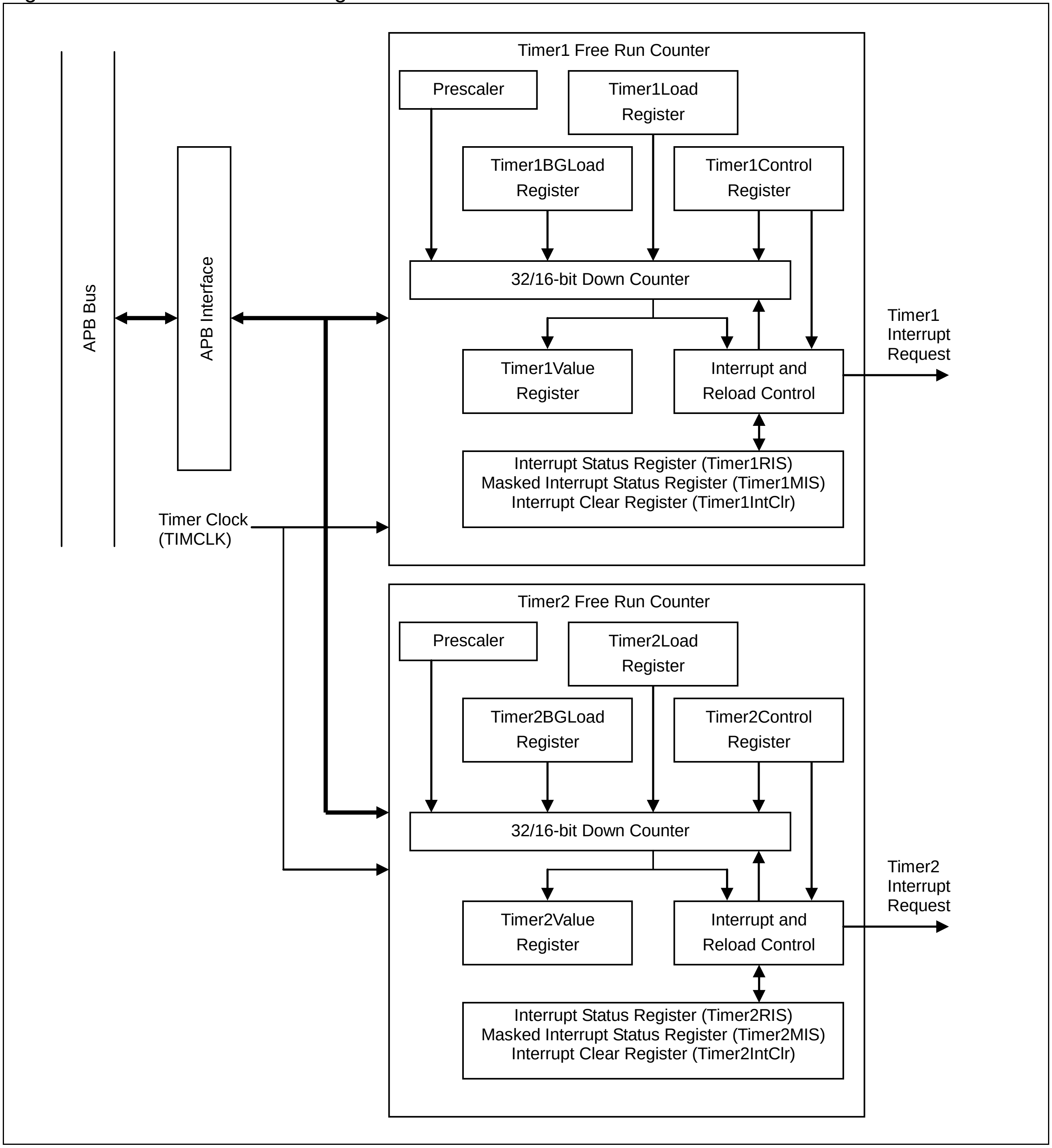
When the timer count reaches zero, an interrupt is generated. Writing to Interrupt Clear Register (TimerXIntClr) clears the interrupt. Also, the interrupt output signal can be masked by Interrupt Mask Register.

The current count value can be read from Value Register at any time.

2. Configuration

This section illustrates the Dual Timer configuration.

Figure 2-1 Dual Timer Block Diagram



3. Operations

This section explains Dual Timer operations.

3.1. Timer Operating Mode

Operating modes are selected from three timer modes based on the settings of the Control Register (TimerXControl)'s mode bit (TimerMode) and one-shot mode bit (OneShot).

Table 3-1 Mode Selection Table

TimerMode	OneShot	Selective Mode
0	0	Free-running Mode
1	0	Periodic Mode
-	1	One-shot Mode

Timer size bit (TimerSize) of the Control Register is used to appropriately configure 32-bit or 16-bit counter operation.

<Note>

The character "X" in a register name in this chapter indicates either register of Free Run Counter 1 or 2.

■ Free-running Mode

When a reset is performed, the timer value is initialized to 0xFFFFFFFF. Then, if the counter is enabled, the count decrements by one at the timer clock (TIMCLK) rising edge. Alternatively, writing to the Load Register (TimerXLoad) loads a new initial counter value. Then, if the counter is enabled, the counter starts to decrement from this loaded value.

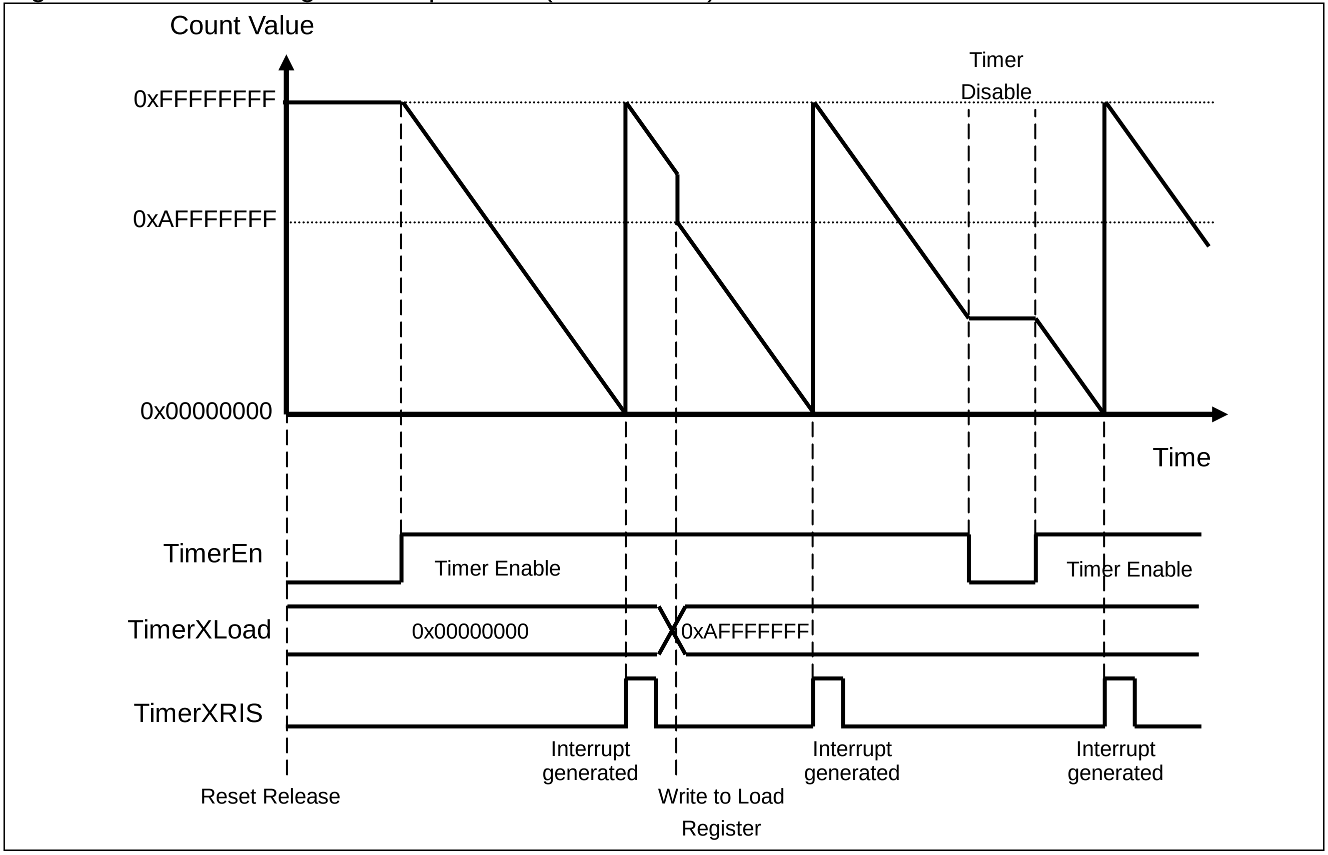
In 32-bit mode, when the count reaches zero (0x00000000), an interrupt is generated. Then, regardless of the Load Register's value, the counter wraps around to 0xFFFFFFFF. The counter starts to decrement again, and as long as the counter is enabled, this whole cycle is repeated.

In 16-bit mode, only the lower 16 bits of the counter are decremented. When the count reaches 0x0000, an interrupt is generated. Then, regardless of the Load Register's value, the counter wraps around to 0xFFFF.

If the Enable bit (TimerEn) of the Control Register (TimerXControl) is cleared and that the counter is disabled, the counter halts and holds the current value. If the counter is enabled again, the counter continues to decrement from the current value.

The counter value can be read from the Value Register (TimerXValue) at any time.

Figure 3-1 Free-running Mode Operation (32-bit Mode)



■ Periodic Mode

Writing to the Load Register (TimerXLoad) loads an initial counter value. Then, the counter starts to decrement from this value if the counter is enabled.

In 32-bit mode, all 32 bits of the counter are decremented. Then, when the count reaches zero (0x00000000), an interrupt is generated. The counter reloads the Load Register value. The counter starts to decrement again. As long as the counter is enabled, this whole cycle is repeated.

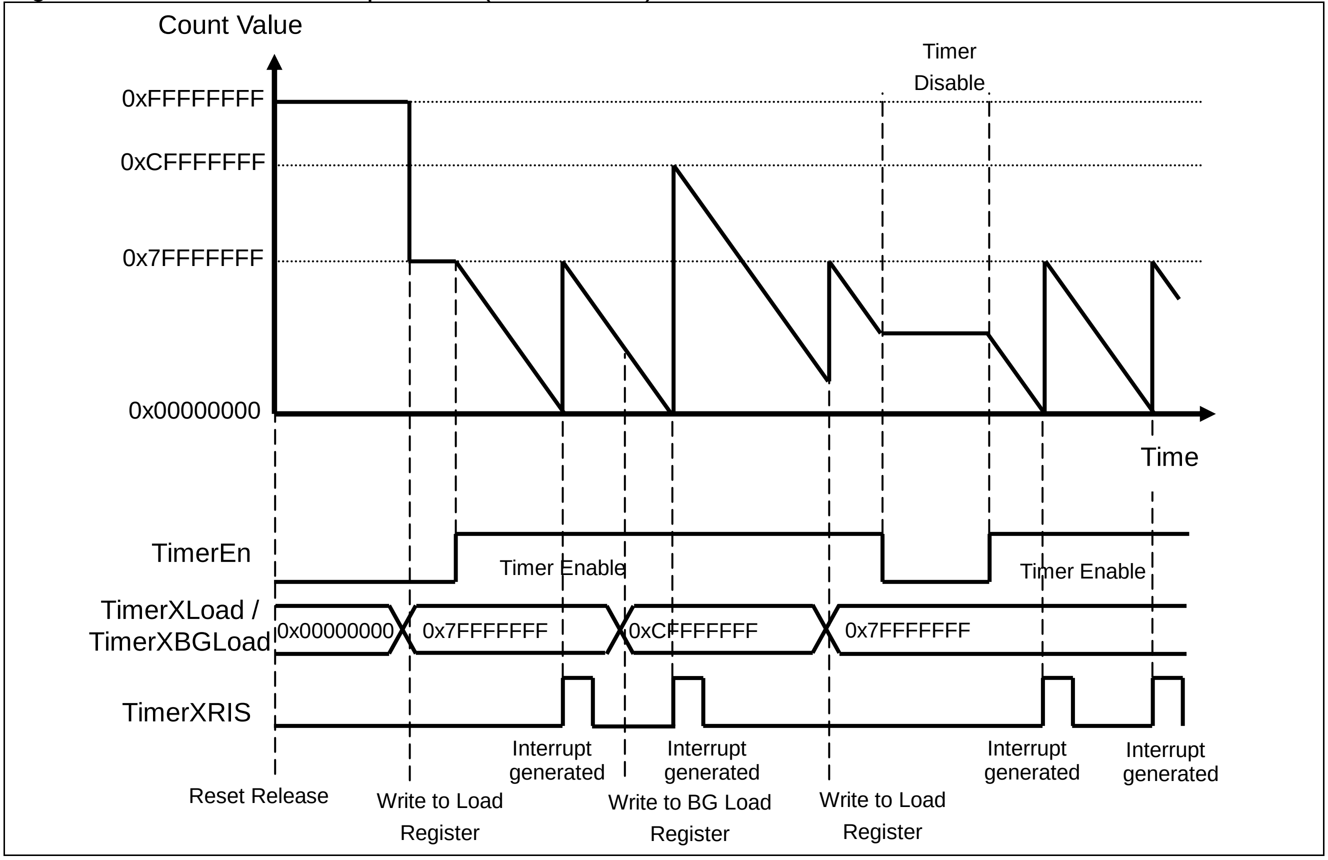
In 16-bit mode, only the lower 16 bits of the counter are decremented. When the count reaches 0x0000, an interrupt is generated. Then, the counter reloads the Load Register value. The counter starts to decrement again. As long as the counter is enabled, this whole cycle is repeated.

When a new value is written to the Background Load Register (TimerXBGLoad) while the counter is running, the value of the Load Register is also updated to the same value. However, the counter continues to decrement to zero. When the counter reaches zero, it reloads the new value. As long as the Timer is set to Periodic Mode, this new load value is used for each subsequent reload.

When a new value is written to the Load Register for loading the value to the counter while the counter is running, the counter value is changed to the new load value at the next timer clock.

If the Enable bit (TimerEn) of the Control Register (TimerXControl) is cleared and that the counter is disabled, the counter halts and holds the current value. If the counter is enabled again, the counter continues to decrement from the current value.

Figure 3-2 Periodic Mode Operation (32-bit Mode)



■ One-shot Mode

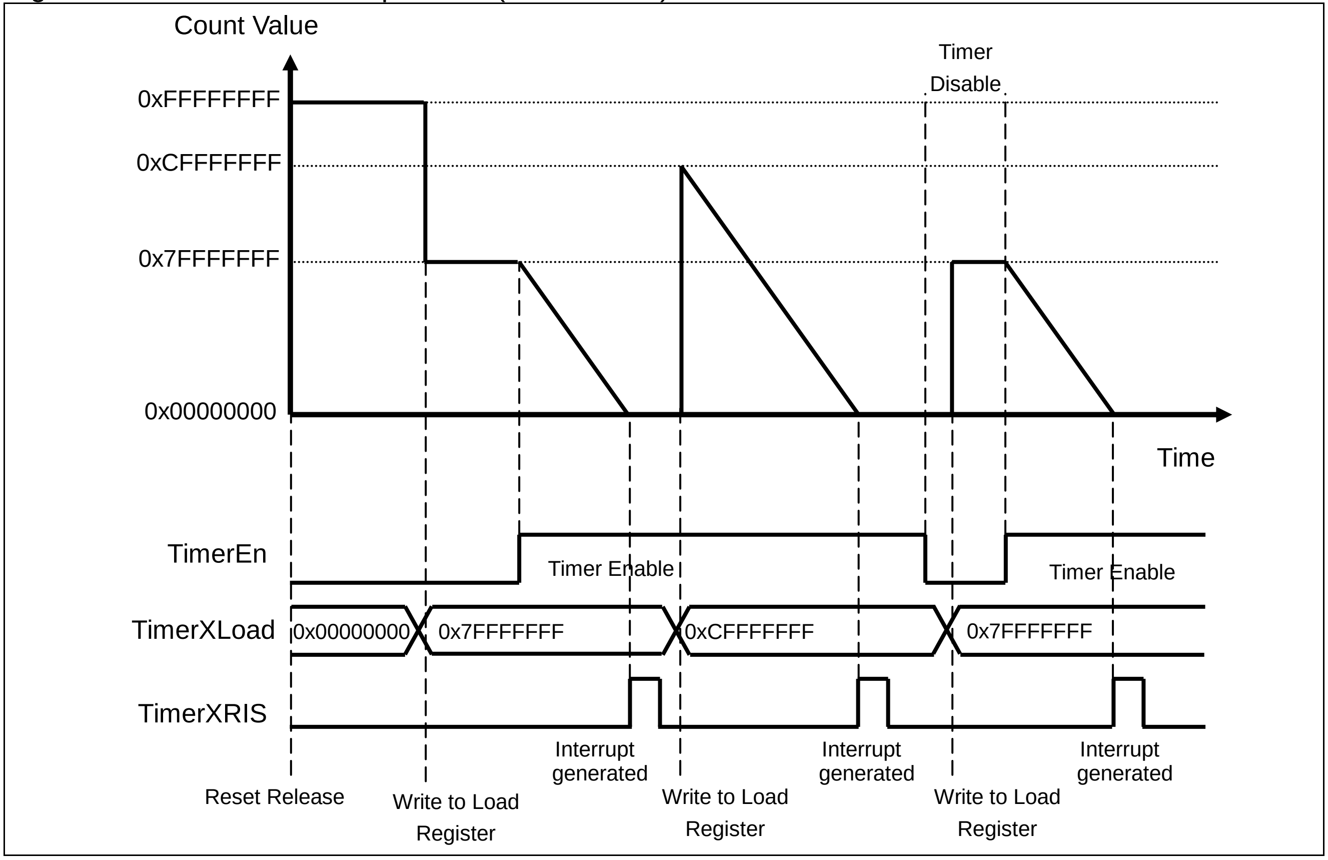
To start the count down sequence in One-shot Mode, a new load value is written to the Load Register (TimerXLoad). If the counter is enabled, it starts to decrement from this value.

In 32-bit mode, all 32 bits of the counter are decremented. Then, when the count reaches zero (0x00000000), an interrupt is generated. Then, the counter halts.

In 16-bit mode, only the lower 16 bits of the counter are decremented. When the count reaches 0x0000, an interrupt is generated. Then, the counter halts.

In One-shot Mode, writing a new value to the Load Register starts the counter again. Then, the counter value is changed to the new load value at the next timer clock.

Figure 3-3 One-shot Mode Operation (32-bit Mode)



3.2. Initial state

After the reset, the timer is initialized as shown below:

- Timer counter disabled
- Free-running mode selected
- 16-bit counter mode selected
- Prescaler in the setting of dividing by 1
- Interrupt clear and interrupt enable states
- Load Register set to zero
- Counter value set to 0xFFFFFFFF

3.3. Interrupt Operation

This section explains interrupt operation.

An interrupt is generated when the counter reaches 0x00000000 (in 32-bit mode) or 0xXXXX0000 (in 16-bit mode) in the setting of interrupt enable (IntEnable=1). In 16-bit mode, the upper 16 bits of the counter are ignored.

Writing to Interrupt Clear Register (TimerXIntClr) clears an interrupt.

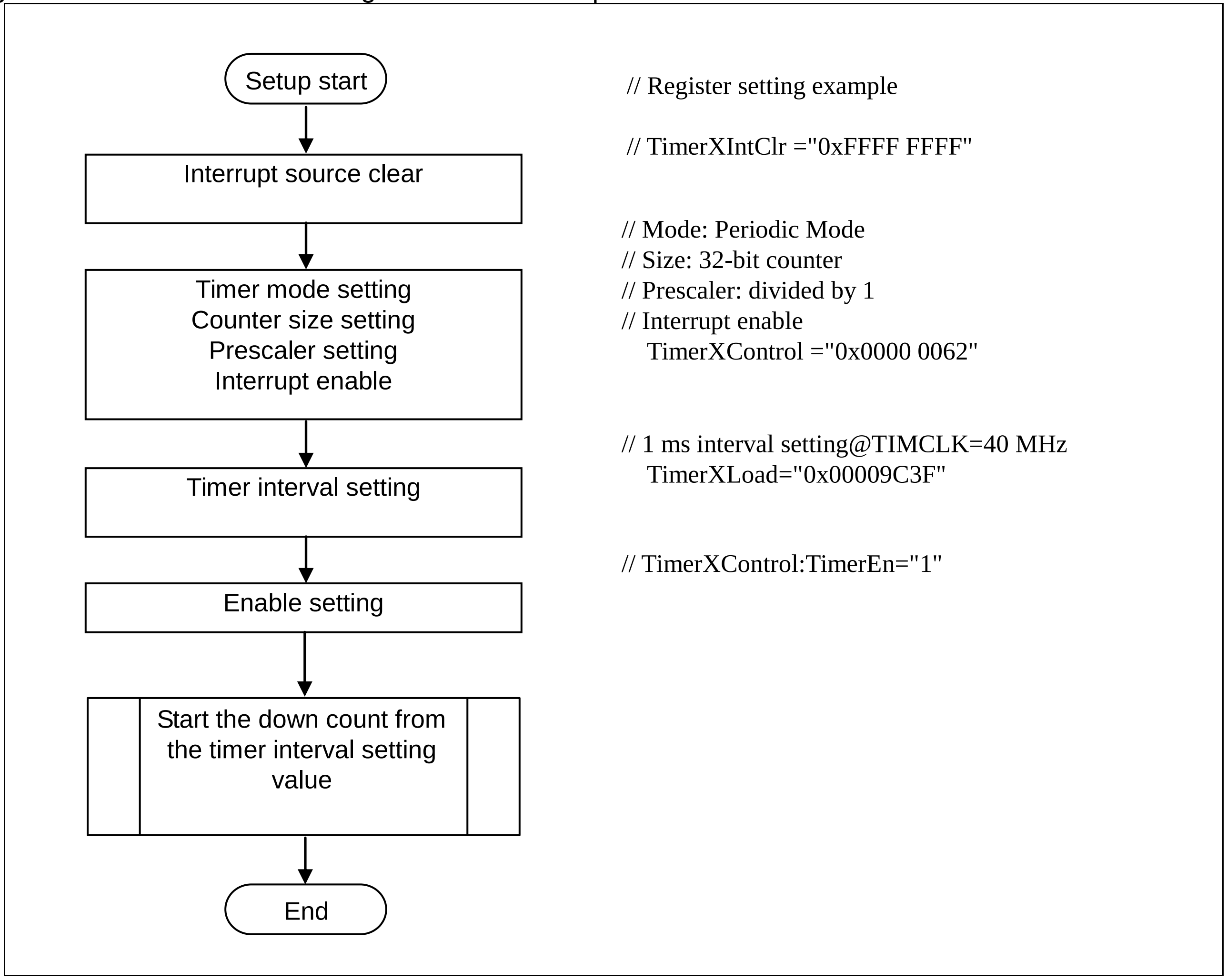
The interrupt signals generated in the Timer module can be masked when Interrupt Enable bit (IntEnable) of the Control Register (TimerXControl) is set to "0". The raw interrupt state before being masked can be read from Interrupt Status Register (TimerXRIS). Also, the masked interrupt state can be read from Masked Interrupt Status Register (TimerXMIS).

4. Setting Procedure Example

This section describes an example of the Dual Timer setting procedure.

■ Dual Timer Setting Procedure Flow

Figure 4-1 Periodic Mode Setting Procedure Example


Setup start
Interrupt source clear
Timer mode setting
Counter size setting
Prescaler setting
Interrupt enable
Timer interval setting
Enable setting
Start the down count from the timer interval setting value
End
// Register setting example
// TimerXIntClr ="0xFFFF FFFF"
// Mode: Periodic Mode
// Size: 32-bit counter
// Prescaler: divided by 1
// Interrupt enable
TimerXControl ="0x0000 0062"
// 1 ms interval setting@TIMCLK=40 MHz
TimerXLoad="0x00009C3F"
// TimerXControl:TimerEn="1"

■ Timer Interval Setting

Expressions of the timer interval calculations in respective modes are shown in Table 4-1:

Table 4-1 Expression for Timer Interval Calculation

Mode	Timer Interval
32-bit Free-running	$(\text{PRESCALE}_{\text{DIV}} / \text{TIMCLK}_{\text{FREQ}}) \times 2^{32}$
16-bit Free-running	$(\text{PRESCALE}_{\text{DIV}} / \text{TIMCLK}_{\text{FREQ}}) \times 2^{16}$
Periodic & One-shot	$(\text{PRESCALE}_{\text{DIV}} / \text{TIMCLK}_{\text{FREQ}}) \times (\text{TimerXLoad} + 1)$

- $\text{TIMCLK}_{\text{FREQ}}$ is the timer clock (TIMCLK) frequency.
- $\text{PRESCALE}_{\text{DIV}}$ is the prescaler division factor of 1, 16, or 256 configured by bit3:2 of the Control Register (TimerXControl).
- TimerXLoad is the value of the Load Register (TimerXLoad).

For example, in the case of $\text{TIMCLK}_{\text{FREQ}}$=40 MHz and $\text{PRESCALE}_{\text{DIV}}$=1, the value of the Load Register (TimerXLoad) to configure 1ms timer interval can be calculated as follows:

$$\begin{aligned}\text{TimerXLoad} &= \text{Timer interval} \times \text{TIMCLK}_{\text{FREQ}} / \text{PRESCALE}_{\text{DIV}} - 1\\ &= 1\text{ ms} \times 40\text{ MHz} / 1 - 1 = 4 \times 10^4 - 1 = \text{0x00009C3F}\end{aligned}$$

<Note>

The minimum valid value of the Load Register (TimerXLoad) is "0x00000001". If the Load Register is set to "0x00000000", an interrupt will be immediately generated.

5. Registers

This section explains the structures and functions of the registers used in Dual Timer.

■ Dual Timer Register List

Abbreviation	Register Name	Reference
Timer1Load	Timer1 Load Register	5.1
Timer1Value	Timer1 Value Register	5.2
Timer1Control	Timer1 Control Register	5.3
Timer1IntClr	Timer1 Interrupt Clear Register	5.4
Timer1RIS	Timer1 Interrupt Status Register	5.5
Timer1MIS	Timer1 Masked Interrupt Status Register	5.6
Timer1BGLoad	Timer1 Background Load Register	5.7
Timer2Load	Timer2 Load Register	5.1
Timer2Value	Timer2 Value Register	5.2
Timer2Control	Timer2 Control Register	5.3
Timer2IntClr	Timer2 Interrupt Clear Register	5.4
Timer2RIS	Timer2 Interrupt Status Register	5.5
Timer2MIS	Timer2 Masked Interrupt Status Register	5.6
Timer2BGLoad	Timer2 Background Load Register	5.7

5.1. Load Register (TimerXLoad) X=1 or 2

The Load Register (TimerXLoad) set a start value to decrement the counter in 32-bit Register.

bit	31	16
Field	TimerXLoad[31:16]	
Attribute	R/W	
Initial value	0x0000	

bit	15	0
Field	TimerXLoad[15:0]	
Attribute	R/W	
Initial value	0x0000	

[bit31:0] TimerXLoad : Timer X Load bits

When a value is directly written to this register, the current count is immediately set to a new value at the next timer clock. Also, in Periodic Mode setting, this value is used for reloading the counter when the current count reaches zero.

In addition, the value in this register is also overwritten when the Background Load Register (TimerXBGLoad) is written. However, in this case, the current count is not immediately affected.

After either the Load Register (TimerXLoad) or the Background Load Register (TimerXBGLoad) is written, the register value written last is returned at any reading. In other words, the same value is read from both of the Load Register and the Background Load Register, and the value is always reloaded after the counter reaches zero in Periodic Mode.

<Note>

The minimum valid value of the Load Register (TimerXLoad) is "0x00000001". If the Load Register is set to "0x00000000", an interrupt will be immediately generated.

5.2. Value Register (TimerXValue) X=1 or 2

The Value Register (TimerXValue) indicates the current value of the decrement counter in 32-bit Read Only Register.

bit	31	16
Field	TimerXValue[31:16]	
Attribute	R	
Initial value	0xFFFF	

bit	15	0
Field	TimerXValue[15:0]	
Attribute	R	
Initial value	0xFFFF	

[bit31:0] TimerXValue : Timer X Value bits

After a load operation which a new load value is written to the Load Register (TimerXLoad), the new load value is reflected immediately to this Value Register (TimerXValue).

<Note>

In 16-bit timer mode, the upper 16 bits of 32-bit Value Register (TimerXValue) are not automatically set to "0x0000". For example, when no writing to the Load Register (TimerXLoad) has occurred yet since the change in the Timer from 32-bit mode to 16-bit mode, the upper 16 bits of the Value Register have non-zero values.

5.3. Control Register (TimerXControl) X=1 or 2

The Control Register (TimerXControl) controls the Timer.

bit	31	16
Field	Reserved	
Attribute	-	
Initial value	0xXXXX	

bit	15	8	7	6	5	4	3	2	1	0
Field	Reserved		Timer En	Timer Mode	Int Enable	Reserved	TimerPre		Timer Size	One Shot
Attribute	-		R/W							
Initial value	0xXX		0	0	1	0	00		0	0

[bit31:8] Reserved : Reserved bits

These bits have no effect in write mode.
The read value is undefined.

[bit7] TimerEn : Enable bit

Value	Description
0	Timer disabled [Initial value]
1	Timer enabled

[bit6] TimerMode : Mode bit

Value	Description
0	Free-running Mode [Initial value]
1	Periodic Mode

[bit5] IntEnable : Interrupt enable bit

Value	Description
0	Interrupt disabled
1	Interrupt enabled [Initial value]

[bit4] Reserved : Reserved bit

This bit have no effect in write mode.
The read value is undefined.

[bit3:2] TimerPre : Prescale bits

bit3	bit2	Description
0	0	Clock divided by 1 [Initial value]
0	1	Clock divided by 16
1	0	Clock divided by 256
1	1	Setting is prohibited.

[bit1] TimerSize : Counter size bit

Select 16/32-bit counter operation.

Value	Description
0	16-bit counter [Initial value]
1	32-bit counter

[bit0] OneShot : One-shot mode bit

Select One-shot Mode or Counter Wrapping Mode (Free-running Mode/Periodic Mode). Based on Mode bit (TimerMode) settings, Free-running Mode or Periodic Mode is selected.

Value	Description
0	Wrapping Mode (Free-running Mode/Periodic Mode) [Initial value]
1	One-shot Mode

<Note>

The counter mode, size, or prescale settings must not be changed while the Timer is running. To configure a new setting, the Timer needs to be disabled first and that a new setting value needs to be written to respective registers. Then, after the setting is changed, the Timer needs to be enabled again. Failure to follow this setting procedure can result in unpredictable behaviors of the device.

5.4. Interrupt Clear Register (TimerXIntClr) X=1 or 2

The Interrupt Clear Register (TimerXIntClr) clears an interrupt.

bit	31	16
Field	TimerXIntClr[31:16]	
Attribute	W	
Initial value	0xXXXX	

bit	15	0
Field	TimerXIntClr[15:0]	
Attribute	W	
Initial value	0xXXXX	

[bit31:0] TimerXIntClr : Interrupt clear bits

Writing any value to this register clears an interrupt output from the counter.

5.5. Interrupt Status Register (TimerXRIS) X=1 or 2

The Interrupt Status Register (TimerXRIS) indicates an unmasked and raw interrupt status.

bit	31	16
Field	Reserved	
Attribute	-	
Initial value	0xXXXX	

bit	15	1	0
Field	Reserved		Timer XRIS
Attribute	-		R
Initial value	XXXXXXXXXXXXXXX		0

[bit31:1] Reserved : Reserved bits

These bits have no effect in write mode.
The read value is undefined.

[bit0] TimerXRIS :Interrupt Status Register bit

Value	Description
0	No interrupt generated from the counter [Initial value]
1	Interrupt generated from the counter

5.6. Masked Interrupt Status Register (TimerXMIS) X=1 or 2

The Masked Interrupt Status Register (TimerXMIS) indicates the masked interrupt status.

bit	31	16
Field	Reserved	
Attribute	-	
Initial value	0xXXXX	

bit	15	1	0
Field	Reserved		Timer XMIS
Attribute	-		R
Initial value	XXXXXXXXXXXXXXX		0

[bit31:1] Reserved : Reserved bits

These bits have no effect in write mode.
The read value is undefined.

[bit0] TimerXMIS : Masked Interrupt Status bit

This bit is a logical AND value of the Raw Interrupt Status and the Timer Interrupt Enable bit (IntEnable) of the Control Register (TimerXControl). The same value as this bit is connected to the interrupt output signal.

Value	Description
0	No interrupt generated from the counter [Initial value]
1	Interrupt generated from the counter

5.7. Background Load Register (TimerXBGLoad) X=1 or 2

The Background Load Register (TimerXBGLoad) is a 32-bit register having a value which the counter starts to decrement.

bit	31	16
Field	TimerXBGLoad[31:16]	
Attribute	R/W	
Initial value	0x0000	

bit	15	0
Field	TimerXBGLoad[15:0]	
Attribute	R/W	
Initial value	0x0000	

[bit31:0] TimerXBGLoad : Background Load bits

This register is used to reload the counter when the current count reaches zero in Periodic Mode setting. This is not used in Free-running Mode or One-shot Mode.

Writing to this register reloads the counter differently from the writing to the Load Register (TimerXLoad). The difference is as follows. Writing to the Load Register immediately starts the counter with the new value; however, writing to this register does not immediately restart the counter with the new value.

After a value is written to either of the Load Register or the Background Load Register (TimerXBGLoad), the register value written last is returned at any reading. In other words, the same value is read from the Load Register (TimerXLoad) and the Background Load Register (TimerXBGLoad), and the value is always reloaded after the counter reaches zero in Periodic Mode.

CHAPTER 3-1: Watch Counter Configuration

This chapter explains the configuration of the watch counter.

CODE :9xFWCTOP-E01.0

1. Configuration

For the configuration of watch counter, see the following relevant chapters.

■ Referred Watch Counter Chapter of each product

Table 1-1 Watch Counter Prescaler Correspondence Table

Product TYPE	Referred Chapter
TYPE0 to TYPE11	Chapter "Watch Counter Prescaler (A)"
TYPE12	Chapter "Watch Counter Prescaler (B)"

Table 1-2 Watch Counter Correspondence Table

Product TYPE	Referred Chapter
TYPE0 to TYPE12	Chapter "Watch Counter"

CHAPTER 3-2: Watch Counter Prescaler (A)

This chapter explains the functions and operations of the watch counter prescaler.

CODE: 9BFWCPRE-E01.3

1. Overview of Watch Counter Prescaler

The watch counter prescaler is a prescaler which generates a counter clock used for a watch counter.

■ Watch Counter Prescaler

This is a prescaler which generates a count clock of the watch counter.

The watch counter prescaler can select a main clock or a sub clock as an input clock (F_{CL}). The watch counter prescaler outputs the division clocks (WCCK0 to WCCK3) shown in Table 1-1 by setting the SEL_OUT bit of the clock selection register (CLK_SEL).

Table 1-1 Division clocks generated by the watch counter prescaler

SEL_OUT	WCCK3	WCCK2	WCCK1	WCCK0
0	$2^{15}/F_{CL}$	$2^{14}/F_{CL}$	$2^{13}/F_{CL}$	$2^{12}/F_{CL}$
1	$2^{25}/F_{CL}$	$2^{24}/F_{CL}$	$2^{23}/F_{CL}$	$2^{22}/F_{CL}$

SEL_OUT: Output clock selection bit of clock selection register (CLK_SEL)
F_{CL}: Frequency of input clock

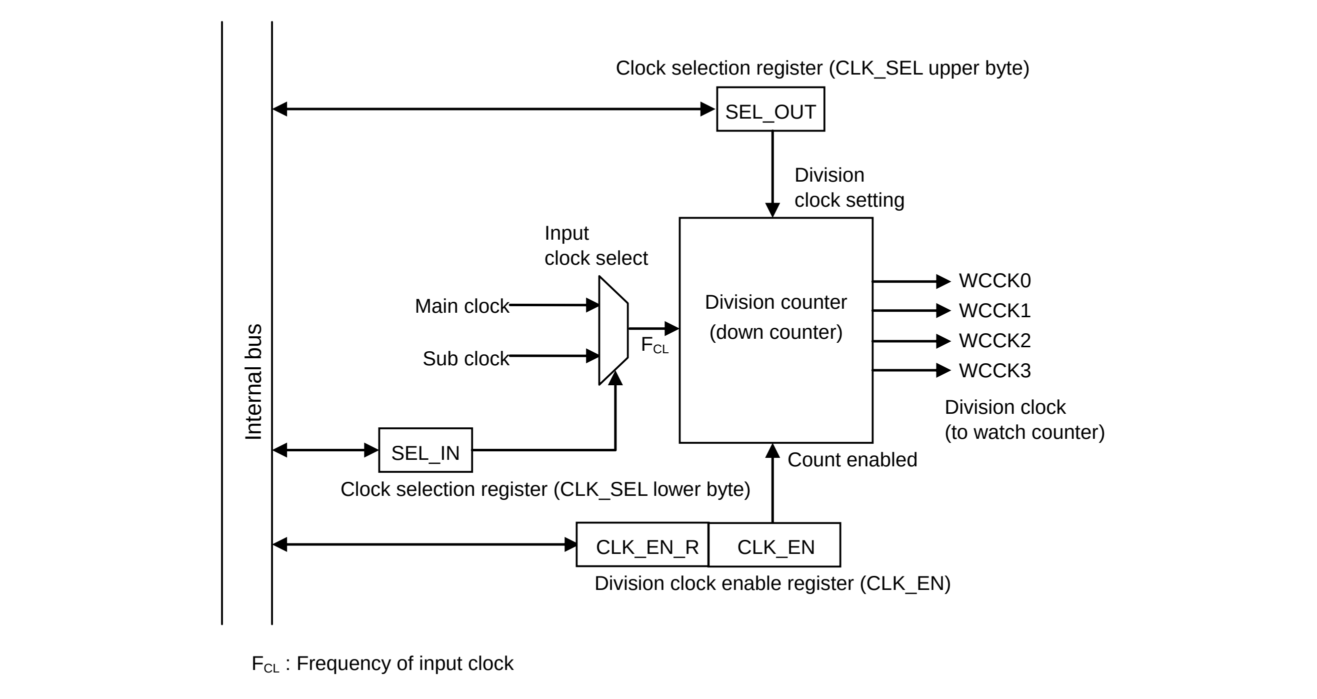
2. Configuration of Watch Counter Prescaler

This section shows the block diagram of watch counter prescaler.

■ Block diagram of watch counter prescaler

Figure 2-1 shows the block diagram of watch counter prescaler.

Figure 2-1 Block diagram of watch counter prescaler



● Clock selection register (CLK_SEL)

This register selects the input clock (F_{CL}) which inputs the division counter, and sets the division clocks (WCCK0 to WCCK3) that output.

● Division clock enable register (CLK_EN)

This register enables counting down of the division counter.

There is a delay for 2 cycles of the clock selected by the Input selection bit (SEL_IN) of the clock selection register (CLK_SEL) during a period of time from a value is written to this register until the division counter starts to operate.

● Division counter

This is a down counter which generates the division clocks (WCCK0 to WCCK3) of the input clock (F_{CL}).

3. Explanation of Operations and Setting Procedure Examples of Watch Counter Prescaler

This section explains the operations of the watch counter prescaler. Also, procedures for setting the operating state are shown.

■ Procedures for setting the watch counter prescaler

The procedures for setting the watch counter prescaler are shown below.

● To start output of the division clock

1. Select the input clock (F_{CL}) of the division counter with the Input Clock Selection Bit (SEL_IN) of the clock selection register (CLK_SEL). Also, set the division clock that outputs with the Output Clock Selection Bit(SEL_OUT) of the clock selection register (CLK_SEL).
   At this time, the division clock to be output is fixed to "L" since the division counter is not operated.
2. Set "1" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to enable output of the division clock.

● To stop output of the division clock

Set "0" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to disable output of the division clock.

● To restart after stopping output of the division clock

1. Set "1" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to enable output of the division clock.
2. Write "0" to the watch counter operation enable bit (WCEN) of the watch counter control register (WCCR) of the watch counter, and clear the value of the 6-bit down counter in the watch counter to "0b000000".
3. Write "1" to the watch counter operation enable bit (WCEN) of the watch counter control register (WCCR) of the watch counter to restart the operation of the watch counter.

● To switch while the division clock is operating

1. Set "0" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to disable output of the division clock.
2. Read the CLK_EN_R bit of the division clock enable register (CLK_EN), and confirm whether output of the division clock is stopped (CLK_EN_R=0).
3. Select the input clock (F_{CL}) of the division counter by the SEL_IN bit of the clock selection register (CLK_SEL). Also, set the division clock to be output with the Output Clock Selection Bit(SEL_OUT) of the clock selection register (CLK_SEL).
4. Set "1" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to enable output of the division clock.

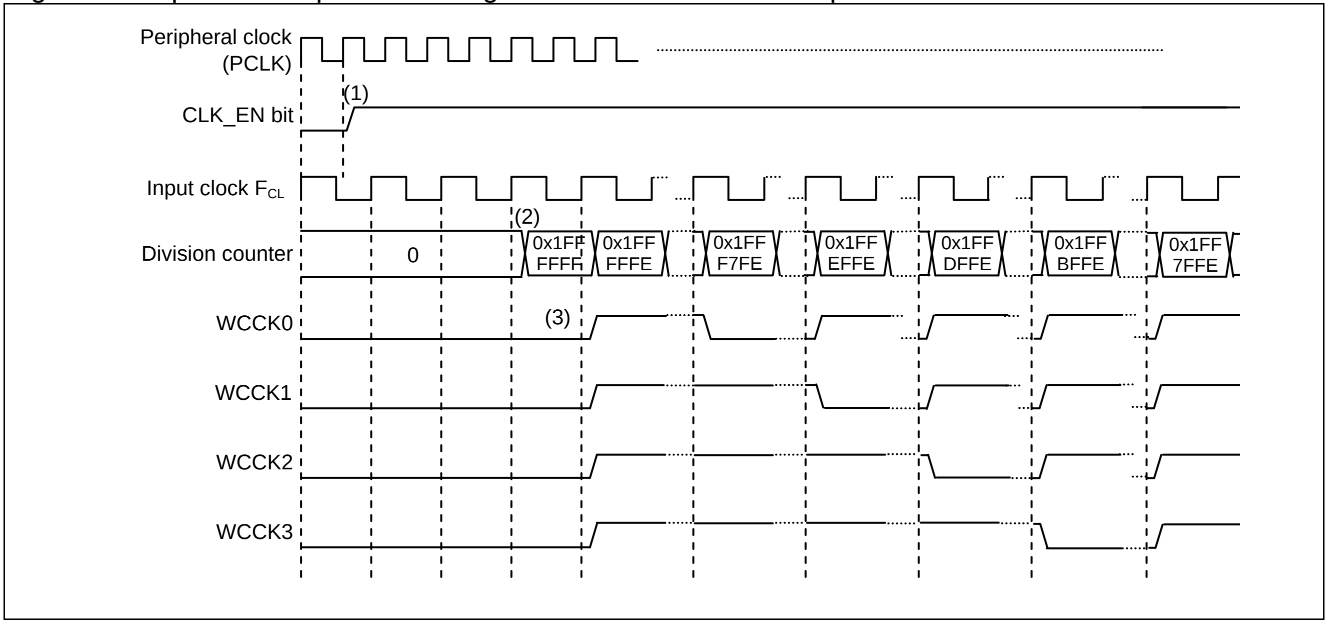
<Notes>

- The peripheral clock (PCLK) is used to set each register of the watch counter prescaler. The input clock (F_{CL}) of the division clock and the peripheral clock (PCLK) are not synchronized. Since the input clock (F_{CL}) of the division counter and peripheral clock (PCLK) are not synchronized, a delay for 3 clocks of the input clock (F_{CL}) is occurred to WCCK0 to WCCK3 after a value is set to each register.
- Regarding 2. of "●To switch while the division clock is operating", a glitch may be occurred when the division clock is switched while the division counter is operating. Confirm whether output of the division counter is stopped.
- The watch counter uses output of the watch counter prescaler as a count clock. Therefore, the settings of the watch counter prescaler should not be changed while the watch counter is operating.

■ Operation of the watch counter prescaler

Figure 3-1 shows an operation of the watch counter prescaler when SEL_OUT bit is set to "0" as an example.

Figure 3-1 Operation explanation diagram of the watch counter prescaler



(1) Set the CLK_EN bit at rising of the peripheral clock (PCLK).
(2) The division counter is operated synchronizing with the input clock (F_{CL}).
(3) The clocks are output to WCCK0 to WCCK3 from the counter according to the settings of the SEL_OUT bit.

<Note>

The peripheral clock (PCLK) is used for the settings of each register of the watch counter prescaler. Since the input clock (F_{CL}) of the division counter and peripheral clock (PCLK) are not synchronized, a delay for 4 clocks of the input clock (F_{CL}) is occurred to WCCK0 to WCCK3 after a value is set to each register.

■ Relationship between the frequency of the input clock (F_{CL}) and the cycle of the division clock

Table 3-1 shows the setting example of the frequency of the input clock (F_{CL}) and the cycle of the division clock.

Table 3-1 Setting example of the watch counter prescaler

SEL_IN	SEL_OUT	Input clock frequency (F_{CL})	Cycle of division clock			
			WCCK3	WCCK2	WCCK1	WCCK0
0 (sub clock)	0	32.768 kHz	1s	500 ms	250 ms	125 ms
1 (main clock)	1	33.554 MHz	1s	500 ms	250 ms	125 ms

4. Registers of Watch Counter Prescaler

This section explains the registers for the watch counter prescaler.

■ List of registers for the watch counter prescaler

Table 4-1 List of registers for the watch counter prescaler

Abbreviated Register Name	Register Name	Reference
CLK_SEL	Clock selection register	4.1
CLK_EN	Division clock enable register	4.2

4.1. Clock Selection Register (CLK_SEL)

The clock selection register (CLK_SEL) selects the input clock (F_{CL}) and sets the division clocks (WCCK0 to WCCK3) to be output.

bit	15	to	11	10	9	8
Field	Reserved			Reserved		SEL_OUT
Attribute	-			-		R/W
Initial value	00000			00		0

bit	7		1	0
Field	Reserved			SEL_IN
Attribute	-			R/W
Initial value	0000000			0

[bit15:11, bit7:1] Reserved : Reserved bits

"0" is always read.
Writing is ignored.

[bit10:9] Reserved : Reserved bits

Always write "0" to these bits.

[bit8] SEL_OUT : Output clock selection bit

This bit selects the division clocks (WCCK0 to WCCK3) to be output from the division counter.

Value	Explanation			
	WCCK3	WCCK2	WCCK1	WCCK0
0	$2^{15}/F_{CL}$	$2^{14}/F_{CL}$	$2^{13}/F_{CL}$	$2^{12}/F_{CL}$
1	$2^{25}/F_{CL}$	$2^{24}/F_{CL}$	$2^{23}/F_{CL}$	$2^{22}/F_{CL}$

[bit0] SEL_IN : Input clock selection bit

This bit selects the input clock (F_{CL}) to be used.

Value	Explanation
0	Generates a division clock using the sub clock.
1	Generates a division clock using the main clock.

4.2. Division Clock Enable Register (CLK_EN)

The division clock enable register (CLK_EN) is a register to enable a count down of the division counter.

bit	7	2	1	0
Field	Reserved		CLK_EN_R	CLK_EN
Attribute	-		R/W	R/W
Initial value	000000		0	0

[bit7:2] Reserved : Reserved bits

"0" is always read.
Writing is ignored.

[bit1] CLK_EN_R : Division clock enable read bit

This bit can read the value of CLK_EN bit used for controlling the division. Writing to this bit does not affect the operations and the reading value.

Value	Explanation
0	The counter for the clock division stops counting, and oscillation of the division clock is not performed.
1	The counter for the clock division starts counting, and oscillation of the division clock is performed.

[bit0] CLK_EN : Division clock enable bit

There is a delay for 2 cycles in the clock selected by the CLK_SEL register during a period of time from a value is written to the CLK_EN bit until the value is reflected.

Value	Explanation
0	The division counter stops counting, and disables oscillation of the division clock. Clears the value of the division counter to "0".
1	The division counter starts counting, and enables oscillation of the division clock.

CHAPTER 3-3: Watch Counter Prescaler (B)

This chapter explains the functions and operations of the watch counter prescaler.

CODE: 9BFWCPRE_FM4-E01.0

1. Overview of Watch Counter Prescaler

The watch counter prescaler is a prescaler which generates a counter clock used for a watch counter.

■ Watch Counter Prescaler

This is a prescaler which generates a count clock of the watch counter.

The watch counter prescaler can select a main clock, a sub clock, sub clock, high-speed CR, or CLKLC as an input clock (F_{CL}). The watch counter prescaler outputs the division clocks (WCCK0 to WCCK3) shown in Table 1-1 by setting the output clock selection bit (SEL_OUT[2:0]) of the clock selection register (CLK_SEL).

Table 1-1 Division clocks generated by the watch counter prescaler

SEL_OUT[2:0]	WCCK3	WCCK2	WCCK1	WCCK0
000	$2^{15}/F_{CL}$	$2^{14}/F_{CL}$	$2^{13}/F_{CL}$	$2^{12}/F_{CL}$
001	$2^{25}/F_{CL}$	$2^{24}/F_{CL}$	$2^{23}/F_{CL}$	$2^{22}/F_{CL}$
010	$2^{4}/F_{CL}$	$2^{3}/F_{CL}$	$2^{2}/F_{CL}$	$2/F_{CL}$
011	$2^{8}/F_{CL}$	$2^{7}/F_{CL}$	$2^{6}/F_{CL}$	$2^{5}/F_{CL}$
100	$2^{12}/F_{CL}$	$2^{11}/F_{CL}$	$2^{10}/F_{CL}$	$2^{9}/F_{CL}$
101	$2^{19}/F_{CL}$	$2^{18}/F_{CL}$	$2^{17}/F_{CL}$	$2^{16}/F_{CL}$
110	$2^{23}/F_{CL}$	$2^{22}/F_{CL}$	$2^{21}/F_{CL}$	$2^{20}/F_{CL}$

SEL_OUT[2:0]:	Output clock selection bit of clock selection register (CLK_SEL)
F_{CL}:	Frequency of input clock

<Note>

· CLKLC is a clock that is obtained by dividing low-speed CR by CR prescaler.

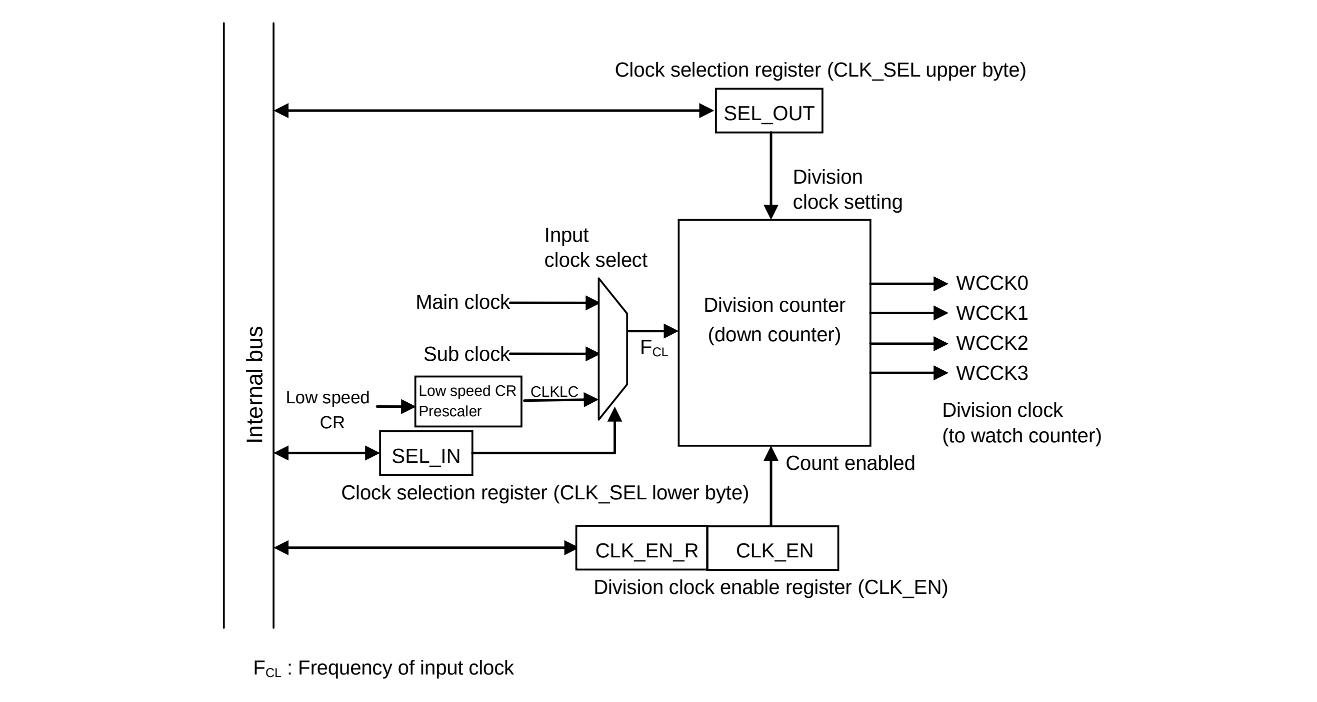
2. Configuration of Watch Counter Prescaler

This section shows the block diagram of watch counter prescaler.

■ Block diagram of watch counter prescaler

Figure 2-1 shows the block diagram of watch counter prescaler.

Figure 2-1 Block diagram of watch counter prescaler



● Clock selection register (CLK_SEL)

This register selects the input clock (F_{CL}) which inputs the division counter, and sets the division clocks (WCCK0 to WCCK3) that output.

● Division clock enable register (CLK_EN)

This register enables counting down of the division counter.

There is a delay for 2 cycles of the clock selected by the Input selection bit (SEL_IN) of the clock selection register (CLK_SEL) during a period of time from a value is written to this register until the division counter starts to operate.

● Division counter

This is a down counter which generates the division clocks (WCCK0 to WCCK3) of the input clock (F_{CL}).

3. Explanation of Operations and Setting Procedure Examples of Watch Counter Prescaler

This section explains the operations of the watch counter prescaler. Also, procedures for setting the operating state are shown.

■ Procedures for setting the watch counter prescaler

The procedures for setting the watch counter prescaler are shown below.

● To start output of the division clock

1. Select the input clock (F_{CL}) of the division counter with the Input Clock Selection Bit (SEL_IN) of the clock selection register (CLK_SEL). Also, set the division clock that outputs with the Output Clock Selection Bit(SEL_OUT) of the clock selection register (CLK_SEL).
   At this time, the division clock to be output is fixed to "L" since the division counter is not operated.
2. Set "1" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to enable output of the division clock.

● To stop output of the division clock

Set "0" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to disable output of the division clock.

● To restart after stopping output of the division clock

1. Set "1" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to enable output of the division clock.
2. Write "0" to the watch counter operation enable bit (WCEN) of the watch counter control register (WCCR) of the watch counter, and clear the value of the 6-bit down counter in the watch counter to "0b000000".
3. Write "1" to the watch counter operation enable bit (WCEN) of the watch counter control register (WCCR) of the watch counter to restart the operation of the watch counter.

● To switch while the division clock is operating

1. Set "0" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to disable output of the division clock.
2. Read the CLK_EN_R bit of the division clock enable register (CLK_EN), and confirm whether output of the division clock is stopped (CLK_EN_R=0).
3. Select the input clock (F_{CL}) of the division counter by the SEL_IN bit of the clock selection register (CLK_SEL). Also, set the division clock to be output with the Output Clock Selection Bit(SEL_OUT) of the clock selection register (CLK_SEL).
4. Set "1" to the division clock enable bit (CLK_EN) of the division clock enable register (CLK_EN) to enable output of the division clock.

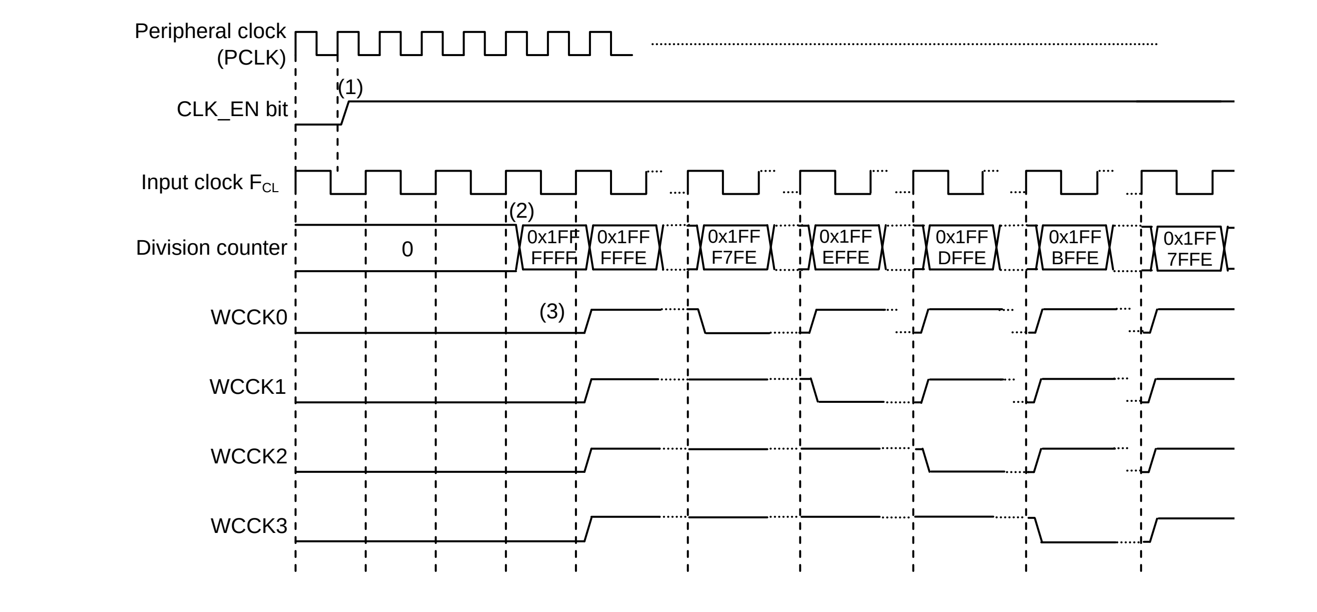
<Notes>

- The peripheral clock (PCLK) is used to set each register of the watch counter prescaler. The input clock (F_{CL}) of the division clock and the peripheral clock (PCLK) are not synchronized. Since the input clock (F_{CL}) of the division counter and peripheral clock (PCLK) are not synchronized, a delay for 3 clocks of the input clock (F_{CL}) is occurred to WCCK0 to WCCK3 after a value is set to each register.
- Regarding 2. of "●To switch while the division clock is operating", a glitch may occur when the division clock is switched while the division counter is operating. Confirm whether output of the division counter is stopped.
- The watch counter uses output of the watch counter prescaler as a count clock. Therefore, the settings of the watch counter prescaler should not be changed while the watch counter is operating.

■ Operation of the watch counter prescaler

Figure 3-1 shows an operation of the watch counter prescaler when SEL_OUT bit is set to "0" as an example.

Figure 3-1 Operation explanation diagram of the watch counter prescaler



(1) Set the CLK_EN bit at rising of the peripheral clock (PCLK).
(2) The division counter is operated synchronizing with the input clock (F_{CL}).
(3) The clocks are output to WCCK0 to WCCK3 from the counter according to the settings of the SEL_OUT bit.

<Note>

The peripheral clock (PCLK) is used for the settings of each register of the watch counter prescaler. Since the input clock (F_{CL}) of the division counter and peripheral clock (PCLK) are not synchronized, a delay for 4 clocks of the input clock (F_{CL}) is occurred to WCCK0 to WCCK3 after a value is set to each register.

■ Relationship between the frequency of the input clock (F_{CL}) and the cycle of the division clock

Table 3-1 shows the setting example of the frequency of the input clock (F_{CL}) and the cycle of the division clock.

Table 3-1 Setting example of the watch counter prescaler

SEL_IN	SEL_OUT	Input clock frequency (F_{CL})	Cycle of division clock			
			WCCK3	WCCK2	WCCK1	WCCK0
00 (sub clock)	000	32.768 kHz	1 s	500 ms	250 ms	125 ms
01 (main clock)	001	33.554 MHz	1 s	500 ms	250 ms	125 ms
10 (High Speed CR)	110	4 MHz	2.097 s	1.049 s	524 ms	262 ms
11 (CLKLC)	100	4 kHz	1.024 s	512 ms	256 ms	128 ms

4. Registers of Watch Counter Prescaler

This section explains the registers for the watch counter prescaler.

■ List of registers for the watch counter prescaler

Table 4-1 List of registers for the watch counter prescaler

Abbreviated Register Name	Register Name	Reference
CLK_SEL	Clock selection register	4.1
CLK_EN	Division clock enable register	4.2

4.1. Clock Selection Register (CLK_SEL)

The clock selection register (CLK_SEL) selects the input clock (F_{CL}) and sets the division clocks (WCCK0 to WCCK3) to be output.

bit	15	to	11	10	9	8
Field		Reserved			SEL_OUT[2:0]	
Attribute		-			R/W	
Initial value		00000			000	

bit	7		2	1	0
Field		Reserved		SEL_IN[1:0]	
Attribute		-		R/W	
Initial value		000000		00	

[bit15:11, bit7:2] Reserved : Reserved bits

"0" is always read.
Writing is ignored.

[bit10:8] SEL_OUT : Output clock selection bit

This bit selects the division clocks (WCCK0 to WCCK3) to be output from the division counter.

Value	Explanation			
	WCCK3	WCCK2	WCCK1	WCCK0
000	$2^{15}/F_{CL}$	$2^{14}/F_{CL}$	$2^{13}/F_{CL}$	$2^{12}/F_{CL}$
010	$2^{25}/F_{CL}$	$2^{24}/F_{CL}$	$2^{23}/F_{CL}$	$2^{22}/F_{CL}$
010	$2^{4}/F_{CL}$	$2^{3}/F_{CL}$	$2^{2}/F_{CL}$	$2/F_{CL}$
011	$2^{8}/F_{CL}$	$2^{7}/F_{CL}$	$2^{6}/F_{CL}$	$2^{5}/F_{CL}$
100	$2^{12}/F_{CL}$	$2^{11}/F_{CL}$	$2^{10}/F_{CL}$	$2^{9}/F_{CL}$
101	$2^{19}/F_{CL}$	$2^{18}/F_{CL}$	$2^{17}/F_{CL}$	$2^{16}/F_{CL}$
110	$2^{23}/F_{CL}$	$2^{22}/F_{CL}$	$2^{21}/F_{CL}$	$2^{20}/F_{CL}$

[bit1:0] SEL_IN : Input clock selection bit

This bit selects the input clock (F_{CL}) to be used.

Value	Explanation
00	Generates a division clock using the sub clock.
01	Generates a division clock using the main clock.
10	Generates a division clock using the high-speed CR
11	Generates a division clock using CLKLC.

4.2. Division Clock Enable Register (CLK_EN)

The division clock enable register (CLK_EN) is a register to enable a count down of the division counter.

bit	7 … 2	1	0
Field	Reserved	CLK_EN_R	CLK_EN
Attribute	-	R/W	R/W
Initial value	000000	0	0

[bit7:2] Reserved: Reserved bits

"0" is always read.
Writing is ignored.

[bit1] CLK_EN_R : Division clock enable read bit

This bit can read the value of CLK_EN bit used for controlling the division. Writing to this bit does not affect the operations and the reading value.

Value	Explanation
0	The counter for the clock division stops counting, and oscillation of the division clock is not performed.
1	The counter for the clock division starts counting, and oscillation of the division clock is performed.

[bit0] CLK_EN : Division clock enable bit

There is a delay for 2 cycles in the clock selected by the CLK_SEL register during a period of time from a value is written to the CLK_EN bit until the value is reflected.

Value	Explanation
0	The division counter stops counting, and disables oscillation of the division clock. Clears the value of the division counter to "0".
1	The division counter starts counting, and enables oscillation of the division clock.

CHAPTER 3-4: Watch Counter

This chapter explains the functions and operations of the watch counter.

CODE: 9BFWC-E01.4_FW09-E00.5

1. Overview of the Watch Counter

The watch counter is a timer that counts down starting from the specified value, and it generates an interrupt request at the time that the 6-bit down counter enters an underflow condition.

■ Watch counter

- For the watch counter, one of the four types of clock (WCCK0, WCCK1, WCCK2, and WCCK3) selected by the count clock select bits (CS[1:0]) of the watch counter control register (WCCR) is used as a count clock of the 6-bit down counter.
- A number between 0 and 63 can be set as the value used for counting by the 6-bit down counter. If "60" is the count value used for a counting period of 1 second, an interrupt request is generated at an interval of 1 minute. If "0" is the count value used for a counting period of 1 second, an interrupt request is generated at an interval of 64 seconds
- An interrupt request can be generated at the time that the 6-bit down counter enters an underflow condition.

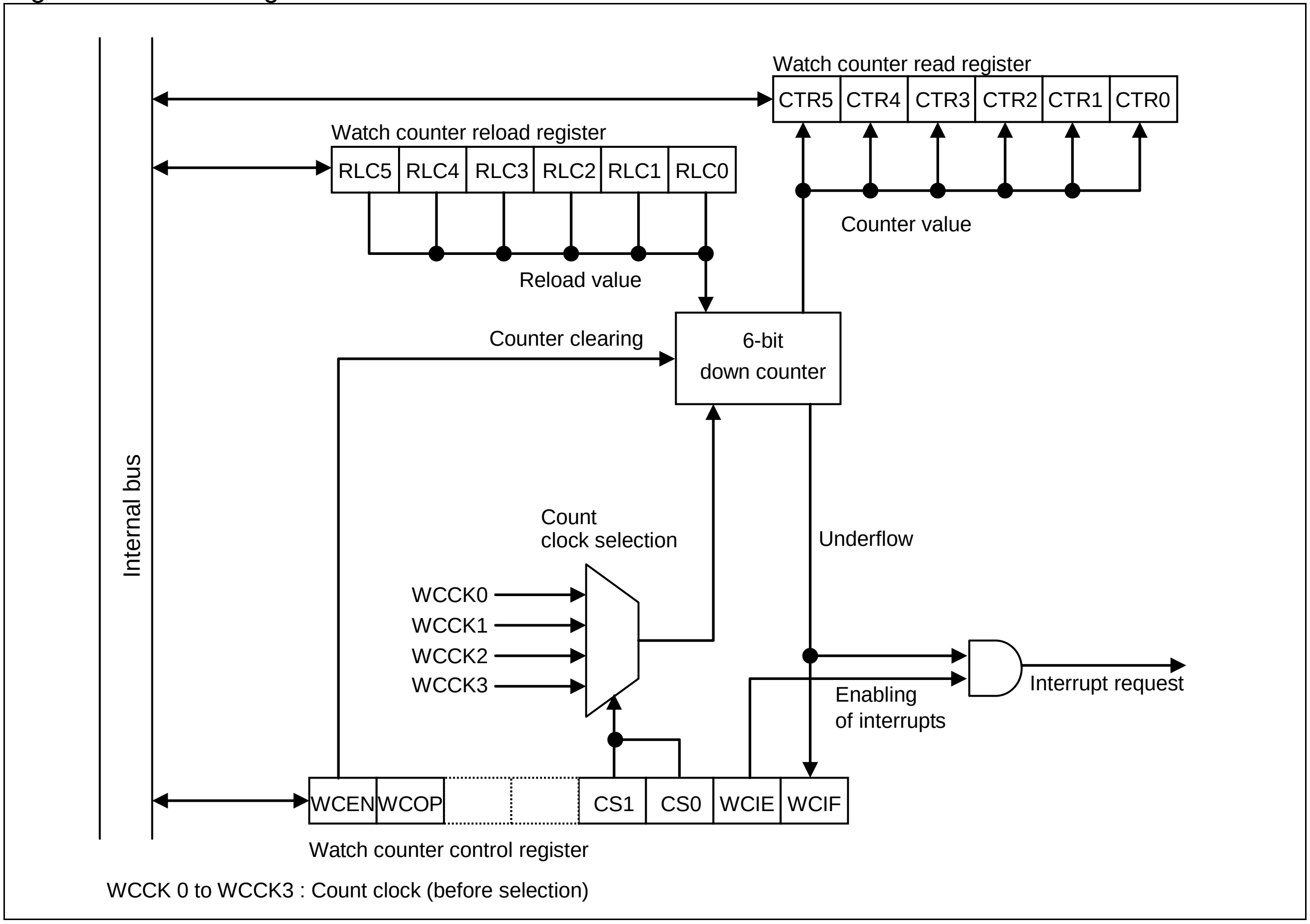
2. Configuration of the Watch Counter

This section shows the block diagram of the watch counter.

■ Block diagram of the watch counter

Figure 2-1 shows a block diagram of the watch counter.

Figure 2-1 Block diagram of the watch counter



● 6-bit down counter

This is the 6-bit down counter of the watch counter. It reloads the value set in the watch counter reload register (WCRL) and starts counting down.

● Watch counter reload register (WCRL)

This register specifies the value used by the watch counter to start counting. The 6-bit down counter counts down starting from the value set in this register.

● Watch counter read register (WCRD)

This register reads the value in the 6-bit down counter. Also, the register can be read to check the count value.

● Watch counter control register (WCCR)

This register controls the operation of the watch counter.

3. Interrupts of the Watch Counter

The 6-bit down counter enters an underflow condition when the value in the 6-bit down counter becomes "0b000001", and an underflow interrupt request is then generated.

■ Interrupts of the watch counter

Table 3-1 shows the interrupts that can be used with the watch counter.

Table 3-1 Interrupts of the watch counter

Interrupt request	Interrupt request flag	Interrupt request enabled	Clearing an interrupt request
Underflow interrupt request	WCCR:WCIF=1	WCCR:WCIE=1	Write "0" to the WCIF bit for WCCR

WCCR: Watch counter control register

<Note>

If generation of interrupt requests is enabled while the interrupt request flag is "1", an interrupt request is generated at the same time. To enable generation of the interrupt request, do either of the following.

- Enable interrupt requests before enabling the generation of interrupt requests.
- Clear interrupt requests simultaneously with interrupts enabled.

4. Explanation of Operations and Setting Procedure Examples of the Watch Counter

This section explains operations of the watch counter. Also, examples of procedures for setting the operating state are shown.

■ Setting procedure examples of the watch counter

To operate the watch counter, follow the procedure below.

(1) Select a count clock by using the count clock select bits (CS[1:0]) of the watch counter control register (WCCR).

(2) Set a count value to the counter reload value setting bits (RLC[5:0]) in the watch counter reload register (WCRL).

(3) Enable the operation of the watch counter by using the watch counter operation enable bit (WCEN) (WCEN = 1) of the watch counter control register (WCCR).
Start a countdown. Counting is performed at the rising edge of the count clock.

(4) If the 6 -bit down counter enters an underflow condition, the value of the interrupt request flag bit (WCIF in the watch counter control register (WCCR) is changed to "1".
At this time, if generation of underflow interrupt requests has been enabled by the WCIE bit (WCIE = 1) in the watch counter control register (WCCR), an underflow interrupt request is generated.
Also, the value that is set in the counter reload value setting bits (RLC[5:0]) in the watch counter reload register (WCRL) is reloaded in the 6-bit down counter and the countdown is restarted.

(5) If the value of the counter reload value setting bits (RLC[5:0]) in the watch counter reload register (WCRL) is changed to another value while the watch counter is active, the watch counter is updated with the new value at the next reload time.

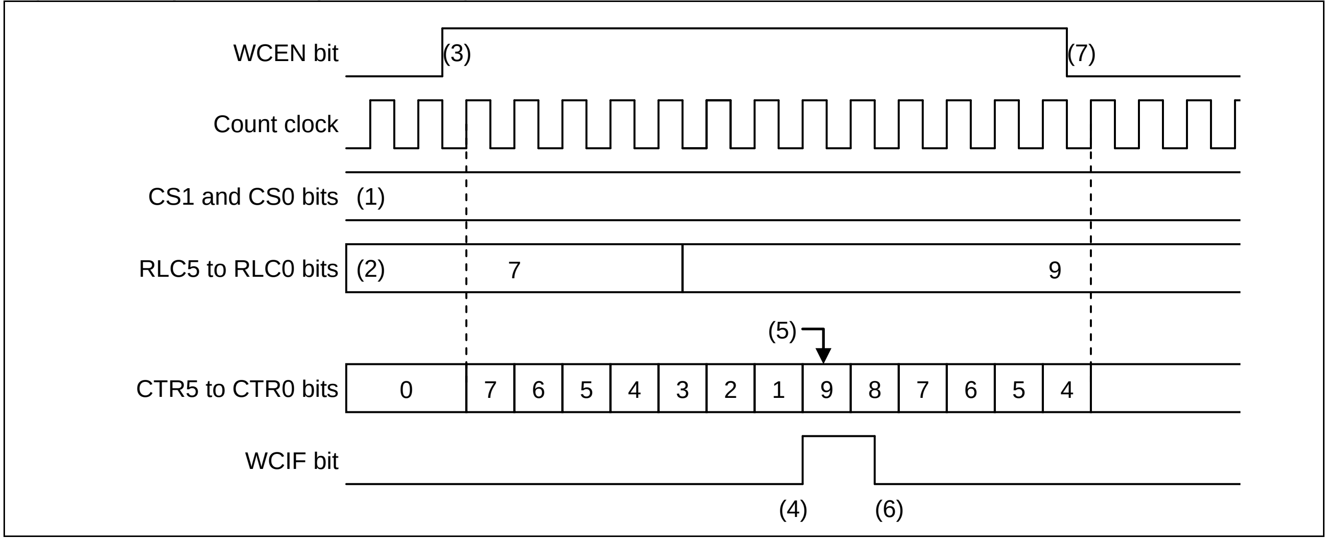
(6) The underflow interrupt request is cleared when "0" is written to the interrupt request flag bit (WCIF) in the watch counter control register (WCCR).

(7) The 6-bit down counter is cleared to "0b000000" and the counting operation is stopped when "0" is written to the watch counter operation enable bit (WCEN) in the watch counter control register (WCCR).

■ Operation of the watch counter

Figure 4-1 shows the operation of the watch counter.

Figure 4-1 Operation explanation figure of the watch counter



<Notes>

- The peripheral clock (PCLK) is used for the settings of each register of the watch counter. Since the count clock and peripheral clock (PCLK) are not synchronized, an error of up to 1T (T: Count clock period) may occur at the count start time, depending on the time at which "1" is written to the WCEN bit in the watch counter control register (WCCR).
- Even at transition of the timer mode, the watch counter continues operating as long as the main clock or sub clock is operating. The timer mode can be canceled with the watch counter interrupt processing routine.
- Under the following condition, verify that the watch counter is stopped by checking the watch counter operating state flag (WCOP) (WCOP=0) in the watch counter control register (WCCR) before reactivating the watch counter.

  Condition: In case of reactivating the watch counter after the watch counter is stopped by writing "0" to the WCEN in the watch counter control register (WCCR) by using the WCEN bit (WCEN = 1).

5. Registers of Watch Counter

This section explains the registers for the watch counter.

■ Registers for the watch counter

Table 5-1 List of registers for the watch counter

Abbreviated Register Name	Register Name	Reference
WCRD	Watch counter read register	5.1
WCRL	Watch counter reload register	5.2
WCCR	Watch counter control register	5.3

5.1. Watch Counter Read Register (WCRD)

This register reads the value in the 6-bit down counter.

bit	7	6	5	4	3	2	1	0
Field	Reserved		CTR[5:0]					
Attribute	-		R					
Initial value	00		000000					

[bit7:6] Reserved : Reserved bits

"0" is always read.

Writing is ignored.

[bit5:0] CTR[5:0] : Counter read bits

These bits can read the counter value.

Writing is ignored.

<Note>

If the 6-bit down counter is operating when its value is read, the register value must be read twice and verified to be the same value.

5.2. Watch Counter Reload Register (WCRL)

This register specifies the value used by the watch counter to start counting. The 6-bit down counter counts down starting from the value set in the register.

The register specifies the reload value for the 6-bit down counter. If the 6-bit down counter enters an underflow condition, the value in this register is reloaded in the 6-bit down counter, and the countdown is restarted.

bit	15	14	13	12	11	10	9	8
Field	Reserved		RLC[5:0]					
Attribute	-		R/W					
Initial value	00		000000					

[bit15:14] Reserved : Reserved bits

"0" is always read.
Writing is ignored.

[bit13:8] RLC[5:0] : Counter reload value setting bits

These bits set the reload value for the 6-bit down counter.

The 6-bit counter counts downwards from the reload value and enters an underflow condition when its value reaches "1". If "0b000000" is set in these bits, it performs 64 countdowns from "63" to "0".
If this bit is modified during counting, the modified value is valid at reloading after underflow.

<Notes>

- If the value of RLC[5:0] bits is changed to another value while the 6-bit down counter is active, an underflow occurs and the new value is then reloaded.
- If the value of RLC[5:0] bits is changed to another value at the same time that an underflow interrupt request is generated, the correct value is not reloaded. Be sure to rewrite the value of RLC[5:0] bits either when the watch counter is stopped or in the interrupt processing routine before an interrupt request is generated.
- To verify whether the reload value is correctly set, read this register.

5.3. Watch Counter Control Register (WCCR)

This register selects a count clock for the watch counter or enables/disables generation of interrupt requests. The register also enables/disables the operation of the watch counter.

bit	23	22	21	20	19	18	17	16
Field	WCEN	WCOP	Reserved		CS1	CS0	WCIE	WCIF
Attribute	R/W	R	-		R/W	R/W	R/W	R/W
Initial value	0	0	00		0	0	0	0

[bit23] WCEN : Watch counter operation enable bit

This bit enables the operation of the watch counter.

- The peripheral clock (PCLK) is used for the settings of each register of the watch counter. Since the count clock and the peripheral clock (PCLK) are not synchronized, an error of up to 1T (T: count clock period) may occur at the count start time, depending on the time at which "1" is written to WCEN bit of watch counter control register (WCCR).
- Before writing "1" to this bit to start the operation of the watch counter, verify that the watch counter is stopped by checking the WCOP bit (WCOP=0).

Value	Explanation
0	The watch counter is disabled/stopped. The value in the 6-bit down counter is cleared to "0b000000".
1	The watch counter is enabled/started.

[bit22] WCOP : Watch counter operating state flag

This bit indicates the operating state of the watch counter.

Value	Explanation
0	The watch counter is stopped.
1	The watch counter is active.

[bit21:20] Reserved : Reserved bits

"0" is always read.

Writing is ignored.

[bit19:18] CS1, CS0 : Count clock select bits

These bits select a clock for the watch counter.
Change these bits when WCCR:WCEN=0 (watch counter operation disabled) and WCOP=0 (watch counter stopped).

bit19	bit18	Explanation
0	0	Selects WCCK0 as a count clock.
0	1	Selects WCCK1 as a count clock.
1	0	Selects WCCK2 as a count clock.
1	1	Selects WCCK3 as a count clock.

[bit17] WCIE : Interrupt request enable bit

This bit specifies whether to generate an underflow interrupt request when the 6-bit down counter underflows (WCIF=1).

Value	Explanation
0	Disables generation of underflow interrupt requests.
1	Enables generation of underflow interrupt requests.

[bit16] WCIF : Interrupt request flag bit

This bit becomes "1" when the counter underflows.

- When this bit and the WCIE bit are "1", a watch counter interrupt is generated.
- "1" can be read when reading by the read modify write access.

Value		Explanation
Write	0	Clear this bit.
	1	No effect on operation.
Read	0	Indicate that an underflow does not occur.
	1	Indicate that an underflow occurs.

CHAPTER 4-1: Real-Time Clock

The real-time clock is composed of the RTC clock control block and RTC count block.

CODE: 9xFRTCTOP-E02.0

1. Configuration of Real-Time Clock

This section shows the configuration of the real-time clock.

■ Reference chapter of the Real-Time Clock

Table 1-1 Correspondence table for Real-time Clock

Product TYPE	Reference
TYPE3, TYPE4, TYPT5	Chapter "RTC Count Block" Chapter "RTC Clock Control Block (A)"
Others	Chapter "RTC Count Block" Chapter "RTC Clock Control Block (B)"

■ Configuration of the Real-time Clock

Figure 1-1 Configuration of the RTC Clock Control Block (A) and RTC count Block

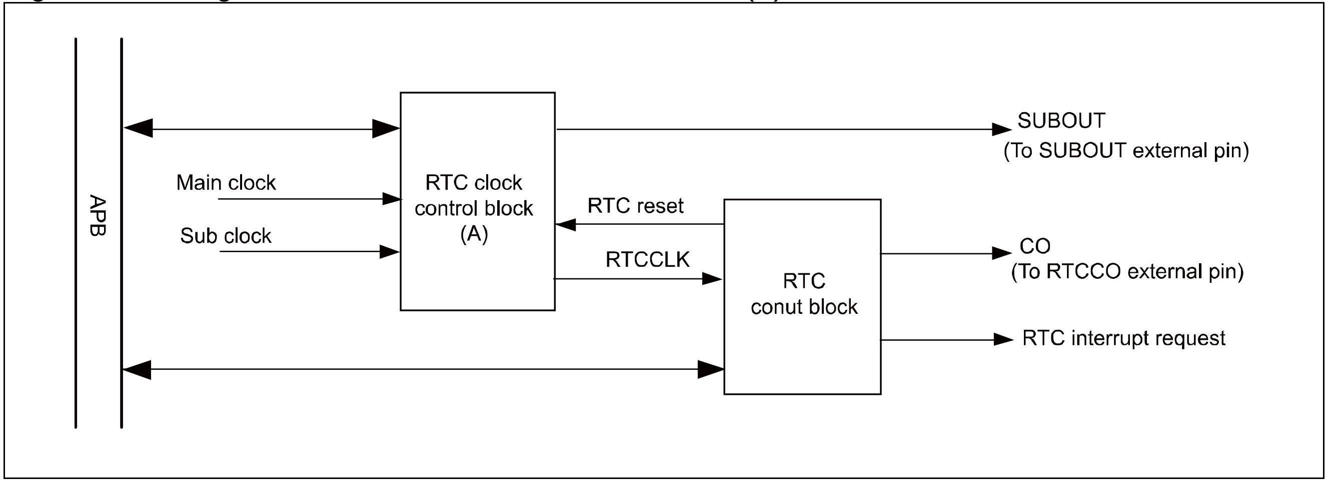
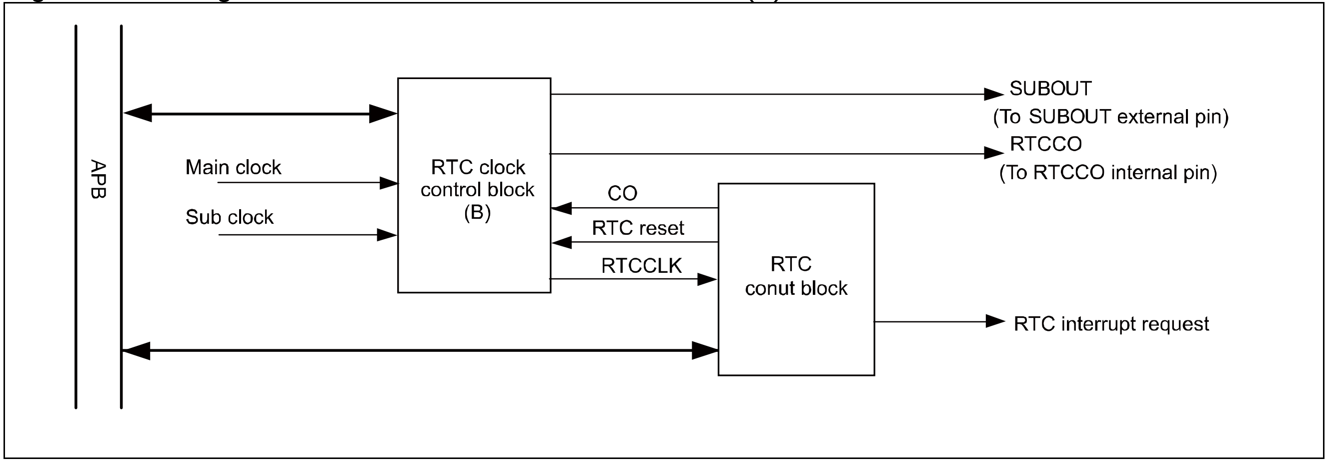


Figure 1-2 Configuration of the RTC Clock Control Block (B) and RTC count Block



2. Real-Time Clock Abbreviations/Acronyms

This section explains the abbreviations and acronyms of the real-time clock.

■ Abbreviations/Acronyms

RTC: Real-time Clock

3. Resetting the Real-Time Clock

This section explains the resetting of the real-time clock.

■ Resetting Real-time Clock

Four types of resets are available to the real-time clock, and they are each initialized with a different register.

1. Low-voltage Detection Reset/Power-on Reset
   All the registers of the real-time clock are initialized.
2. System Reset
   A system reset occurs with a reset factor (i.e., INITX pin input, a software watchdog reset, a hardware watchdog reset, a clock failure detection reset, or an abnormal frequency detection reset).
   All the registers of the RTC clock control block are initialized.
   For the registers that are initialized in the RTC count block, see "4. Resetting of the RTC Count Block" in Chapter "RTC Count Block".
3. RTC Reset
   An RTC reset occurs by writing "1" to the SRST (RTC reset bit) in the RTC count block.
   For the initialization of the registers in the RTC clock control block, see the precautions for the respective resisters in "7. Registers of the RTC Clock Control Block" in Chapter "RTC Clock Control Block".
   For the registers that are initialized in the RTC count block, see "4. Resetting of the RTC Count Block" in Chapter "RTC Count Block".

CHAPTER 4-2: RTC Count Block

This chapter explains the functions and operation of the RTC count block.

CODE: FS13-E1.2

1. Overview of the RTC Count Block

The RTC count block counts years, months, dates, hours, minutes, seconds, and days of the week from 00 to 99 years. Alarm and timer settings are possible as well. An alarm can be set to a specific year, month, date, hour, and minute. It can also be set to a specific year, month, date, hours, or minutes independently. A timer can be set to a period up to one day. It can be set to a desired period (with the hours, minutes, and seconds specified) or in desired intervals (with hours, minutes, and seconds specified). An overview of the RTC count block is shown below.

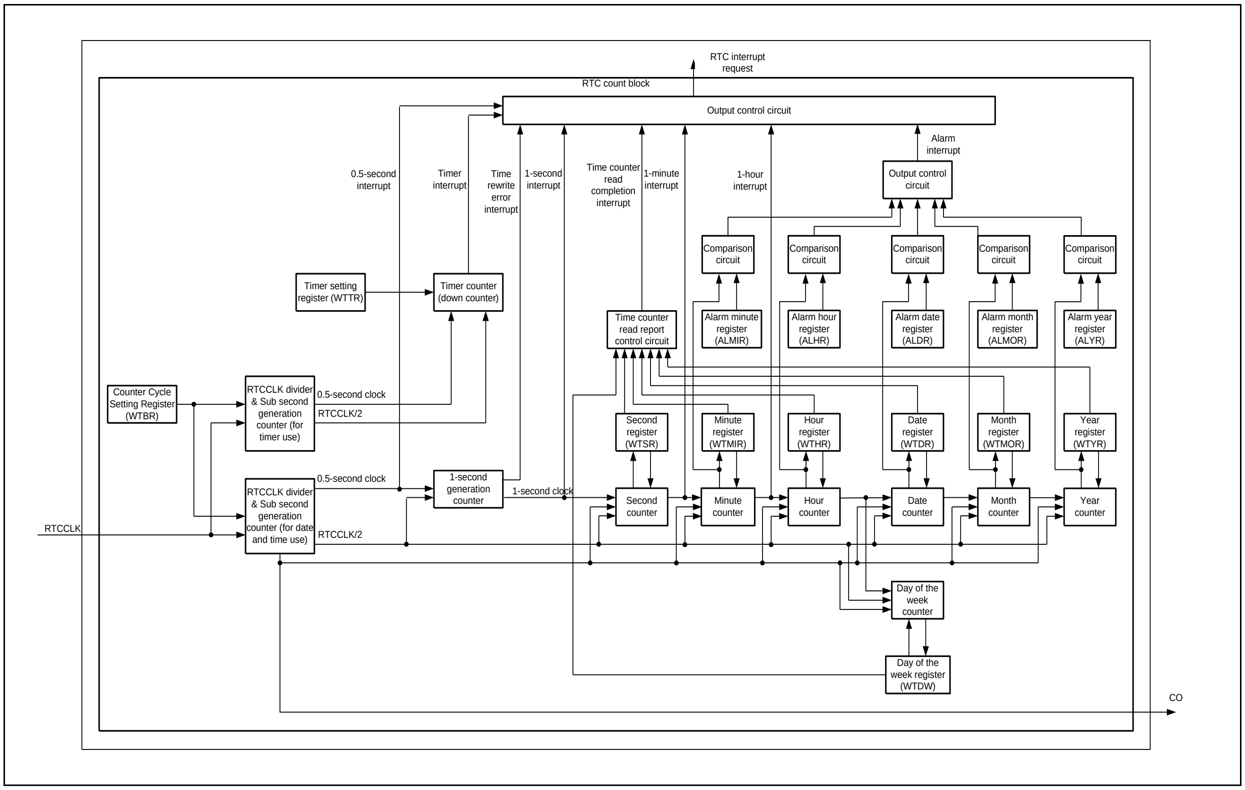
■ Functional Overview of the RTC Count Block

- Date and time (year/month/date/hour/minute/second/day of the week) settings
- Counts dates and time (years, months, days, hours, minutes, seconds, and days of the week)
- Leap year compliant (00 year is operated as leap year)
- Alarm settings with a specific date and time (year/month/date/hour/minute)
- A specific year, month, date, hour, or minute can be set individually as well
- A timer can be set to a period of up to one day. It can be set to a desired period (with the hours, minutes, and seconds specified) or in desired intervals (with hours, minutes, and seconds specified)
- It is possible to rewrite the time by resetting the watch count of the RTC count block to make time-signal-based settings
- It is possible to rewrite the time while the watch count of the RTC count block continues for time zone changes. In that case, the clock is guaranteed to continue time counting if the value is rewritten within a second
- The following interrupts can be output:
  - Alarm (with an interrupt generated at a set date and time)
  - Every hour
  - Every minute
  - Every second
  - Every 0.5 seconds
  - Timer
  - Time rewrite error
  - Timer counter read completion
  - Pulse output in 0.5-second intervals

2. Block Diagram of RTC Count Block

Figure 2-1 shows the block diagram of RTC count block.

Figure 2-1 Block Diagram of RTC Count Block



■ Counter Cycle Setting Register (WTBR)

This register stores a value to be loaded to a sub-second generation counter (for date, time, and timer use).

Set this register to a 0.5-second count value. The value in this register will be loaded to the sub-second generation counter when the RTC starts operating or the sub-second generation counter is set to 0.

■ RTCCLK Divider and Sub-second Generation Counter (for Timer Use)

The RTCCLK divider (for timer use) generates a clock from the RTCCLK divided by 2. The sub-second generation counter (for timer use) operates at the generated clock in the RTCCLK divider (for timer use) to count a sub-second (0.5-second) cycle.

■ RTCCLK Divider and Sub-second Generation Counter (for Date and Time Use)

The RTCCLK divider (for date and time use) generates a clock from the RTCCLK divided by 2. The sub-second generation counter (for date and time use) operates at the generated clock in the RTCCLK divider (for date and time use) to count a sub-second (0.5-second) cycle.

■ Timer Setting Register (WTTR)

This register stores a timer set value that will be up with an elapse of the set value (hours, minutes, and seconds) or in intervals of the set value (hours, minutes, and seconds).

■ Timer Counter (Down Counter)

The timer counter is loaded with a value set in the timer setting register, and counts down in a cycle of 0.5-second clock cycles output from the sub-second generation counter (for timer use).

■ 1-second Generation Counter

This counter counts 0.5-second clock cycles output from the sub-second generation counter (for date and time use) and generates 1-second clock pulses.

■ Second counter, minute counter, hour counter, date counter, month counter, year counter, and day of the week counter.

The second counter, minute counter, hour counter, date counter, month counter, year counter, and day of the week counter count seconds, minutes, hours, dates, months, years, and days of the week.

■ Second register (WTSR), minute register (WTMIR), hour register (WTHR), date register (WTDR), month register (WTMOR), and year register (WTYR)

This register indicates the second, minute, hour, date, month, and year information in the RTC count block.

■ Time Counter Read Report Control Circuit

This circuit reports the completion of the reading of the time counter.

■ Alarm minute register (ALMIR), alarm hour register (ALHR), alarm date register (ALDR), alarm month register (ALMOR), alarm year register (ALYR)

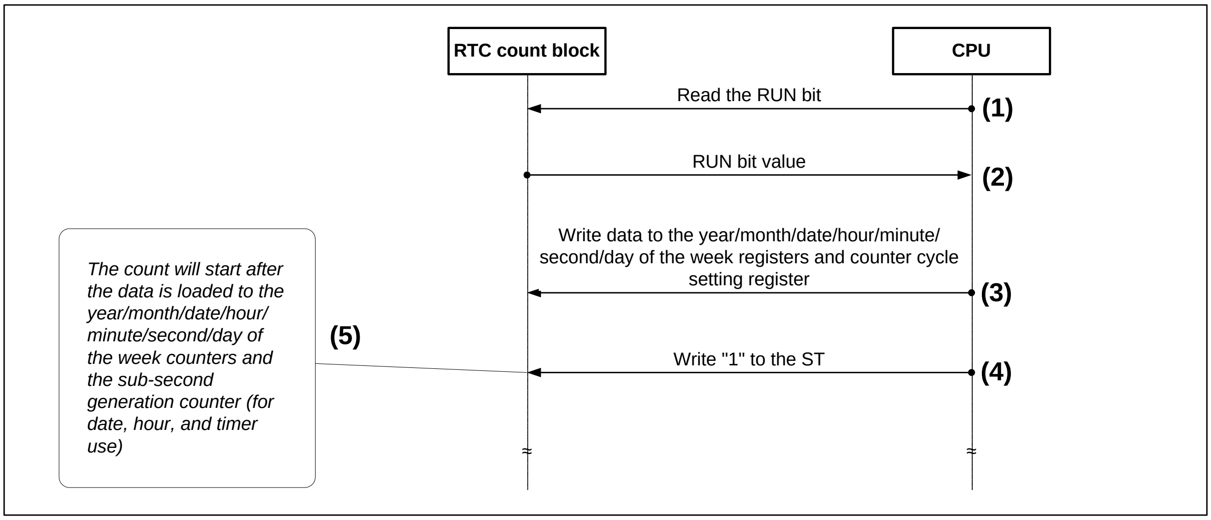
This register stores minute, hour, date, month, and year alarm set values. When an alarm is ON, the comparison circuit will compare the stored value in the register and the minute/hour/date/month/year counter value, and when the values coincides, an alarm interrupt will be generated.

3. Operation of RTC Count Block and Setting Procedures Example

This section explains the operation of the RTC count block and a setting procedures example.

■ Example of the Initial Time Setting Procedures

Figure 3-1 Setting Procedure of the Initial Time Settings


RTC count block
CPU
Read the RUN bit
(1)
RUN bit value
(2)
Write data to the year/month/date/hour/minute/ second/day of the week registers and counter cycle setting register
(3)
Write "1" to the ST
(4)
(5)
The count will start after the data is loaded to the year/month/date/hour/ minute/second/day of the week counters and the sub-second generation counter (for date, hour, and timer use)

(1) Read the RUN bit.

(2) The following initial time settings are possible in the following flow while the RTC count block is not in operation under the following condition: RUN = "0".
See the examples of the time rewrite setting procedures (with time count continued) and time rewrite setting procedures (with time count reset) under the following condition: RUN = "1".

(3) Write the desired time to the year/month/date/hour/minute/second/day of the week registers (WTYR, WTMOR, WTDR, WTHR, WTMIR, WTSR, and WTDW). Write a desired 0.5-second count value to the counter cycle setting register (WTBR).

(4) Write the following value: ST = "1".

(5) When ST=”1” is set, the following operations are executed to start counting:

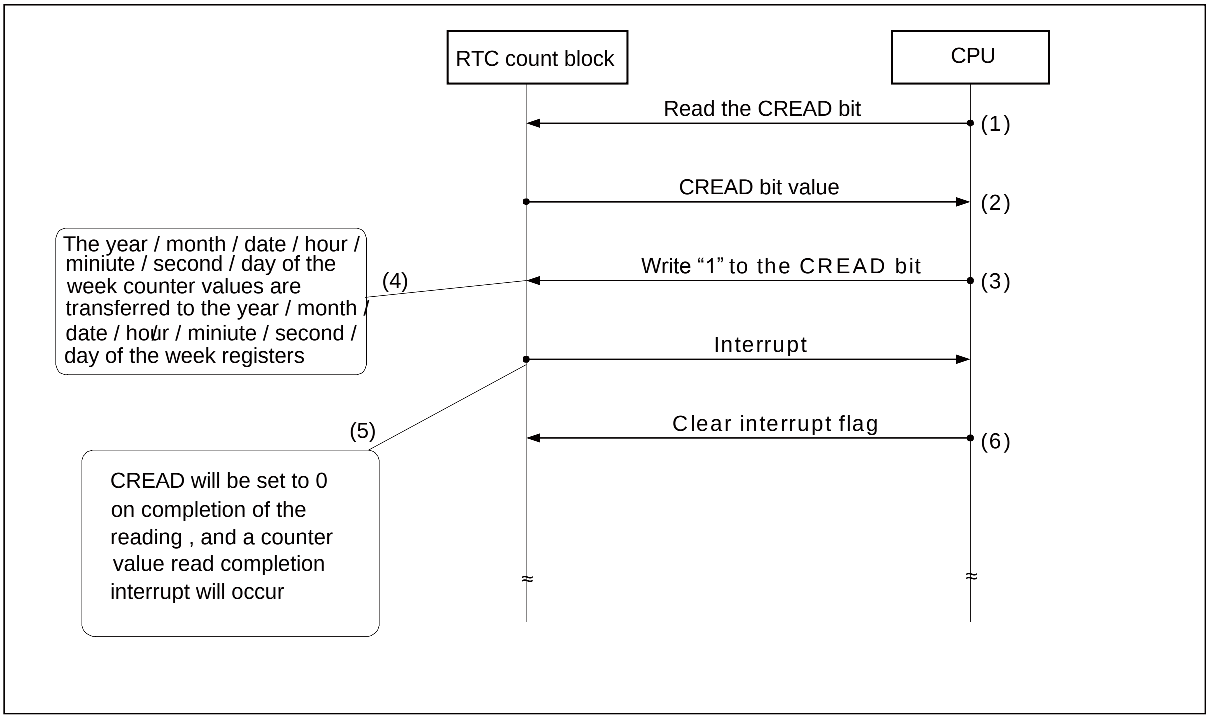
- The year/month/date/hour/minute/second/day of the week register values are loaded to the year/month/date/hour/minute/second/day of the week counters.
- The value in the counter cycle setting register (WTBR) are loaded to the sub-second generation counters (for date, time, and timer use), and start counting.

<Note>

If continuing the time after system reset or RTC reset, write READ=1 and wait READ=0 before writing ST=1. The time before these reset is transferred from the year/month/date/hour/minute/second/day of the week counters to the year, month, date, hour, minute, second, and day of the week registers. After writing ST=1, the value of the year, month, date, hour, minute, second, and day of the week registers is transferred to the year/month/date/hour/minute/second/day of the week counters, and it is possible to continue the time.

■ Example of the Time Read Setting Procedures

Figure 3-2 Setting Procedure of the Time Read


RTC count block
CPU
Read the CREAD bit
(1)
CREAD bit value
(2)
The year / month / date / hour / miniute / second / day of the week counter values are transferred to the year / month / date / hour / miniute / second / day of the week registers
(4)
Write "1" to the CREAD bit
(3)
Interrupt
(5)
Clear interrupt flag
(6)
CREAD will be set to 0 on completion of the reading , and a counter value read completion interrupt will occur

(1) Read the CREAD bit.

(2) When the CREAD is "1", wait until that the CREAD value becomes "0".

(3) Write the following value: CREAD = "1".

(4) The year/month/date/hour/minute/second/day of the week counter values will be transferred to the year, month, date, hour, minute, second, and day of the week registers (WTYR, WTMOR, WTDR, WTHR, WTMIR, WTSR, WTDW).

(5) When the above operation is completed, the following condition will be set :CREAD="0" and a year/month/date/hour/minute/second/day of the week counter value read completion interrupt will occur.

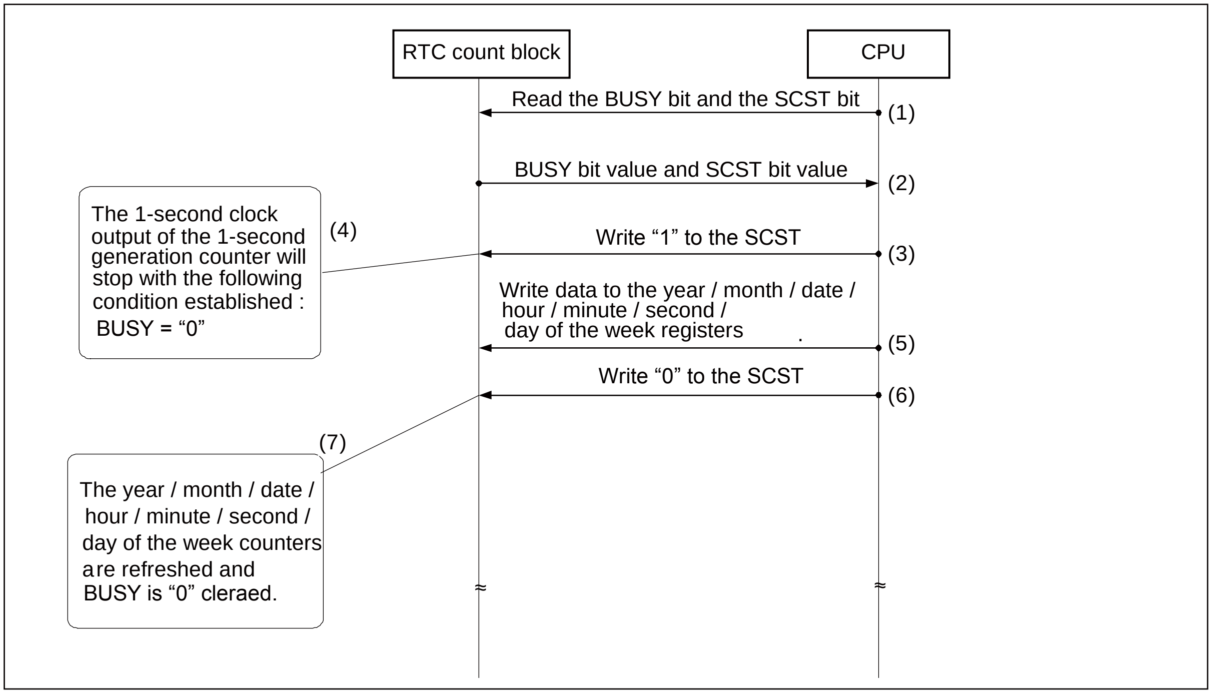
(6) Clear the year/month/date/hour/minute/second/day of the week counter value read completion interrupt flag bit.

<Notes>

- When CREAD="1", do not write "1" to SCST and SRST.
- While CREAD="1", do not set to STOP mode.
- After "1" is written to CREAD, do not stop the RTC count block (with "0" written to ST) until the year/month/date/hour/minute/second/day of the week counter read completion interrupt is generated.

■ Example of the Time Rewrite Setting Procedures (with the Time Count Continued)

Figure 3-3 Setting Procedure of the Time Rewrite Settings (with the Time Count Continued)


RTC count block
CPU
Read the BUSY bit and the SCST bit
(1)
BUSY bit value and SCST bit value
(2)
The 1-second clock output of the 1-second generation counter will stop with the following condition established : BUSY = “0”
(4)
Write “1” to the SCST
(3)
Write data to the year / month / date / hour / minute / second / day of the week registers
(5)
Write “0” to the SCST
(6)
(7)
The year / month / date / hour / minute / second / day of the week counters are refreshed and BUSY is “0” cleraed.
≈
≈

(1) Read the BUSY bit and the SCST bit.

(2) If BUSY="1", wait until the following condition is established: BUSY = "0" and SCST="0".
In other cases, go to step(3).

(3) Write the following value: SCST = "1".

(4) BUSY will be set to "1".
The 1-second clock output of the 1-second generation counter will come to a stop.

(5) Write the desired year, month, date, hour, minute, second, day of the week value to the year/month/date/hour/minute/second/day of the week registers (WTYR, WTMOR, WTDR, WTHR, WTMIR, WTSR, WTDW) while the following condition is maintained: SCST = "1".

(6) Write the following value: SCST = "0".

(7) Only the refreshed set values in the year/month/date/hour/minute/second/day of the week registers will be transferred to the year/month/date/hour/minute/second/day of the week counters and BUSY will be cleared to "0".

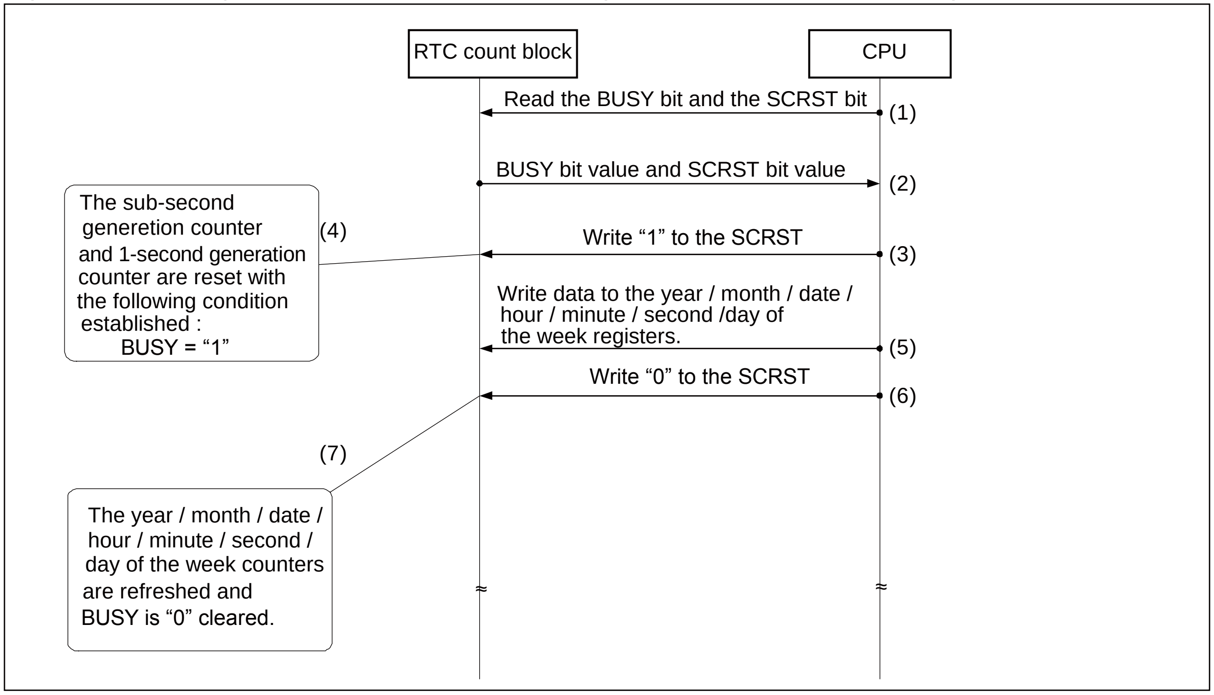
<Notes>

- Writing "1" to the SCST is not allowed under the following condition: BUSY = "1" and SCST="0".
- Writing "1" to SCST is not allowed under the following condition: RUN="0".
- Continuous time counting cannot be guaranteed if the period specified in (3) to (6) exceeds 1 second. In that case, a time rewrite error interrupt will occur. If the time lag occurs, the time rewrite error flag becomes "1". Set SCST=0 to clear the Time rewrite error flag and set the time again according to the above procedure.
- Writing data to the year/month/date/hour/minute/second/day of the week registers is not allowed while data is in transit from the year/month/date/hour/minute/second/day of the week registers to the year/month/date/hour/minute/second/day of the week counters under the following conditions: SCST = "0" and BUSY = "1".
  Keep in mind that the year/month/date/hour/minute/second/day of the week register values will be overwritten with with the year/month/date/hour/minute/second/day of the week counter values if "1" is written to the CREAD before the year/month/date/hour/minute/second/day of the week counters are refreshed after the following condition is set: SCST = "1".

- The data transfer from the year/month/date/hour/minute/second/day of the week registers to the year/month/date/hour/minute/second/day of the week counters will not be performed correctly, and no year/month/date/hour/minute/second/day of the week counter values will be guaranteed if the STOP mode is set while the following condition is maintained: BUSY = "1".
- Writing "0" to ST is not allowed during BUSY = "1".

■ Example of the Time Rewrite Setting Procedures (with the Time Count Reset)

Figure 3-4 Setting Procedure of Time Rewrite (with the Time Count Reset)


RTC count block
CPU
Read the BUSY bit and the SCRST bit
(1)
BUSY bit value and SCRST bit value
(2)
The sub-second generetion counter and 1-second generation counter are reset with the following condition established : BUSY = “1”
(4)
Write “1” to the SCRST
(3)
Write data to the year / month / date / hour / minute / second /day of the week registers.
(5)
Write “0” to the SCRST
(6)
(7)
The year / month / date / hour / minute / second / day of the week counters are refreshed and BUSY is “0” cleared.
≈
≈

(1) Read the BUSY bit and the SCRST bit.
(2) If BUSY = "1", wait until the following condition is established: BUSY = "0" and SCRST="0".
In other cases, go to step(3).
(3) Write the following value: SCRST = "1".
(4) BUSY will be set to "1".
The sub-second generation counter and 1-second generation counter will be reset.
(5) Write the desired new year, month, date, hour, minute, second, day of the week value to the year/month/date/hour/minute/second/day of the week registers (WTYR, WTMOR, WTDR, WTHR, WTMIR, WTSR, WTDW) while the following condition is maintained: SCRST = "1".
(6) Write the following value: SCRST = "0".
(7) Only the refreshed year/month/date/hour/minute/second/day of the week register values will be transferred to the year/month/date/hour/minute/second/day of the week counters and BUSY will be cleared to "0".

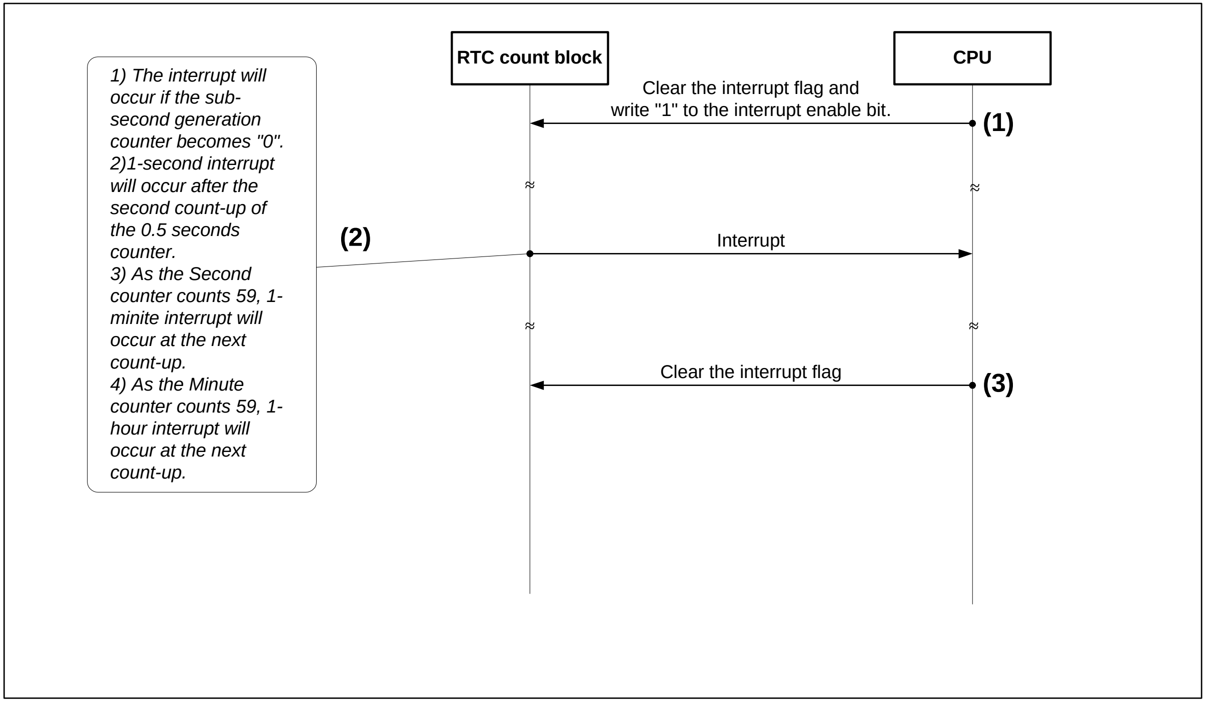
<Notes>

- Writing "1" to the SCRST is disabled under the following condition: BUSY = "1" and SCRST="0".
- Writing "0" to SCRST is not allowed under the following condition: RUN="0".
- Writing data to the year/month/date/hour/minute/second/day of the week registers are disabled while data is in transit from the year/month/date/hour/minute/second/day of the week registers to the year/month/date/hour/minute/second/day of the week counters under the following conditions: SCRST = "0" and BUSY = "1".

- Keep in mind that the year/month/date/hour/minute/second/day of the week register values will be overwritten with the year/month/date/hour/minute/second/day of the week counter values if CREAD is executed before the year/month/date/hour/minute/second/day of the week counters are refreshed after the following condition is set: SCRST = "1".
- The data transfer from the year/month/date/hour/minute/second/day of the week registers to the year/month/date/hour/minute/second/day of the week counters will not be performed correctly, and no year/month/date/hour/minute/second/day of the week counter values will be guaranteed if the STOP mode is set while the following condition is maintained: BUSY = "1".
- Writing "0" to ST is not allowed during BUSY="1".

■ Example of the Setting Procedures of Every 0.5 seconds/Every second/Every minute/Every hour Interrupt

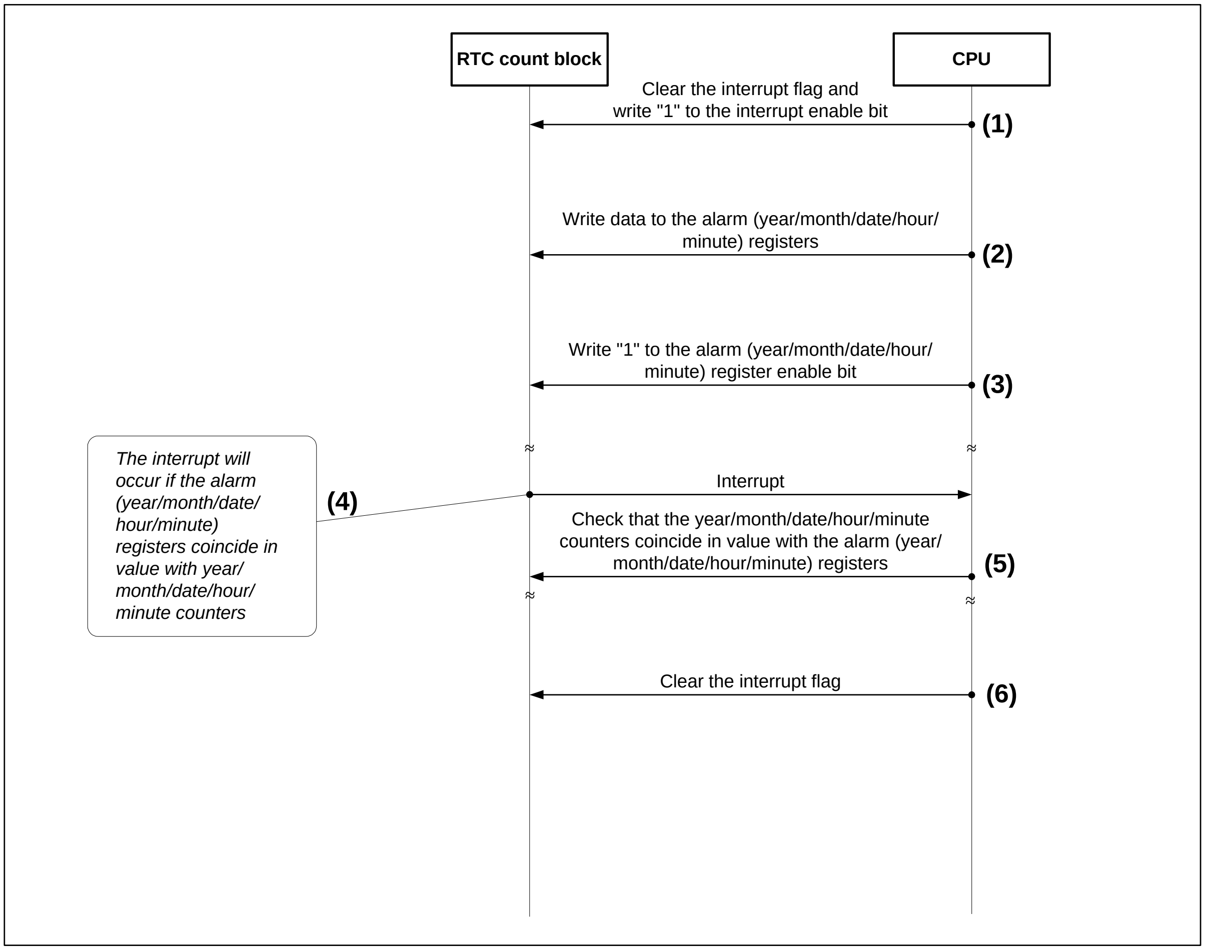
Figure 3-5 Setting Procedure of Every 0.5 seconds/Every second/Every minute/Every hour Interrupt Settings


RTC count block
CPU
Clear the interrupt flag and
write "1" to the interrupt enable bit.
(1)
(2)
Interrupt
Clear the interrupt flag
(3)
1) The interrupt will occur if the sub-second generation counter becomes "0".
2)1-second interrupt will occur after the second count-up of the 0.5 seconds counter.
3) As the Second counter counts 59, 1-minite interrupt will occur at the next count-up.
4) As the Minute counter counts 59, 1-hour interrupt will occur at the next count-up.

(1) Write INTSSI/ INTSI/ INTMI/ INTHI="0" to clear the interrupt flag bit.
Write "1" to the interrupt enable bit of INSSIE, INTSIE, INTMIE, or INHIE to be used to enable the interrupt.
(2) If either of 0.5 seconds/ 1-second/1-minute/1-hour interrupt occurs, an interrupt will be generated.
(3) Write INSSIE/ INTSIE/ INTMIE /INHIE ="0" to clear the interrupt flag bit.

■ Example of the Alarm Interrupt Setting Procedures

Figure 3-6 Setting Procedure of the Alarm Interrupt Settings


RTC count block
CPU
Clear the interrupt flag and write "1" to the interrupt enable bit
(1)
Write data to the alarm (year/month/date/hour/minute) registers
(2)
Write "1" to the alarm (year/month/date/hour/minute) register enable bit
(3)
The interrupt will occur if the alarm (year/month/date/hour/minute) registers coincide in value with year/month/date/hour/minute counters
(4)
Interrupt
Check that the year/month/date/hour/minute counters coincide in value with the alarm (year/month/date/hour/minute) registers
(5)
Clear the interrupt flag
(6)

(1) Clear the alarm interrupt flag bit with the following condition written: INTALI = "0".
The alarm interrupt is enabled with the following condition written: INTALIE = "1".

(2) Write the desired time value to the alarm (year/month/date/hour/minute) registers so that the alarm interrupt will occur at the desired time.

(3) Write "1" to the alarm (year/month/date/hour/minute) register enable bit.

(4) The RTC count block interrupt request will be generated when the alarm (year/month/date/hour/minute) register values coincide with the year/month/date/hour/minute counter values.

(5) Read the time by following the example of the time read setting procedures, and check that the year/month/date/hour/minute counter values coincides with the alarm (year/month/date/hour/minute) register values.

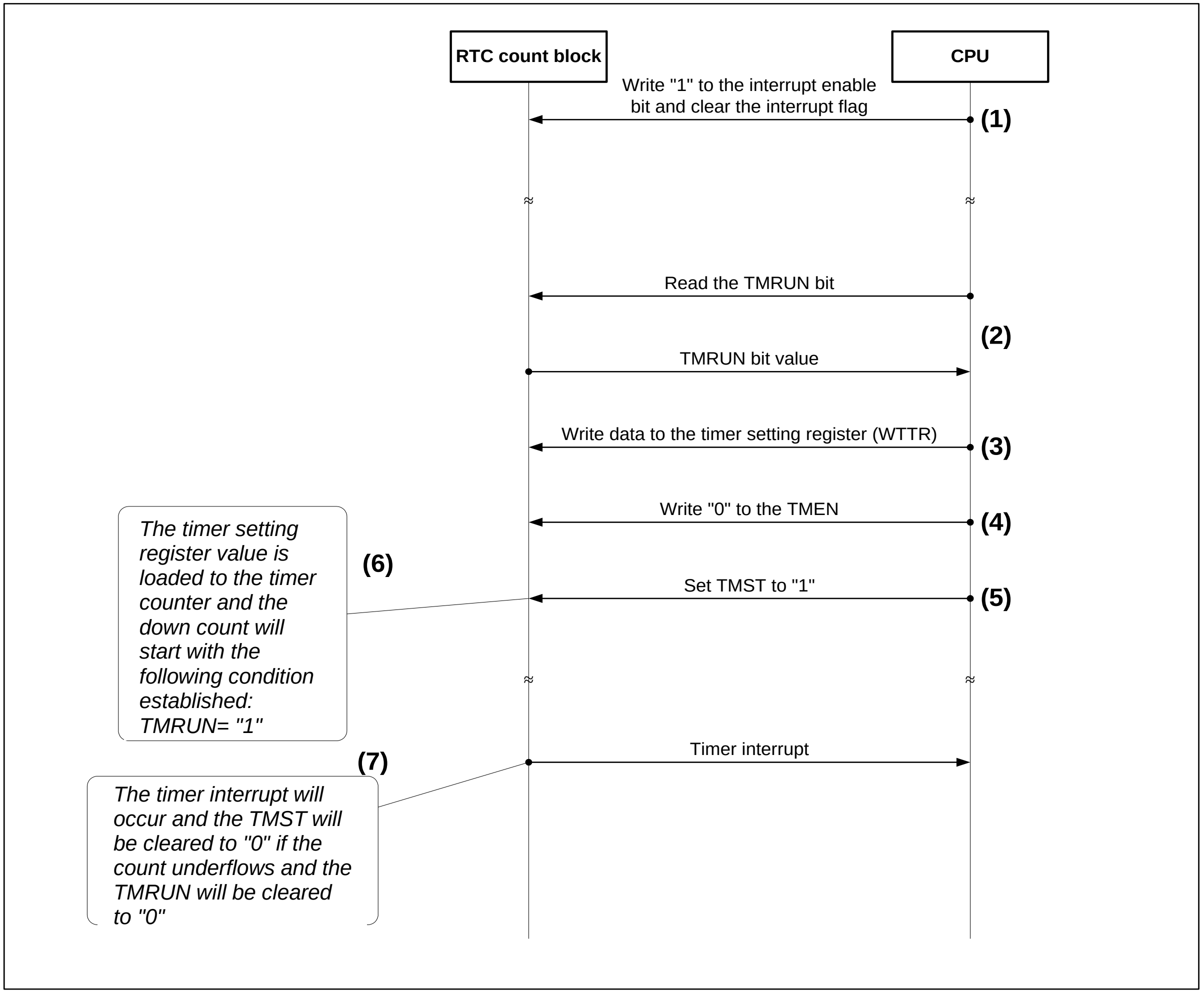
(6) Clear the alarm interrupt flag bit with the following condition written: INTALI = "0".

<Note>

The interrupt may occur immediately after "1" is written to either of the Alarm register enable bit. Read the time after the interrupt to check the value of Year/Month/Date/Hour/Minute counter is the same as that of the Alarm (Year/Month/Date/Hour/Minute) register.

■ Example of the Timer Interrupt Setting Procedures (with (Hours, Minutes, and Seconds) Elapsed)

Figure 3-7 Setting Procedure of the Timer Interrupt Setting (with (Hours, Minutes, and Seconds) Elapsed)


RTC count block
CPU
Write "1" to the interrupt enable bit and clear the interrupt flag
(1)
Read the TMRUN bit
(2)
TMRUN bit value
Write data to the timer setting register (WTTR)
(3)
Write "0" to the TMEN
(4)
Set TMST to "1"
(5)
(6)
The timer setting register value is loaded to the timer counter and the down count will start with the following condition established: TMRUN= "1"
Timer interrupt
(7)
The timer interrupt will occur and the TMST will be cleared to "0" if the count underflows and the TMRUN will be cleared to "0"

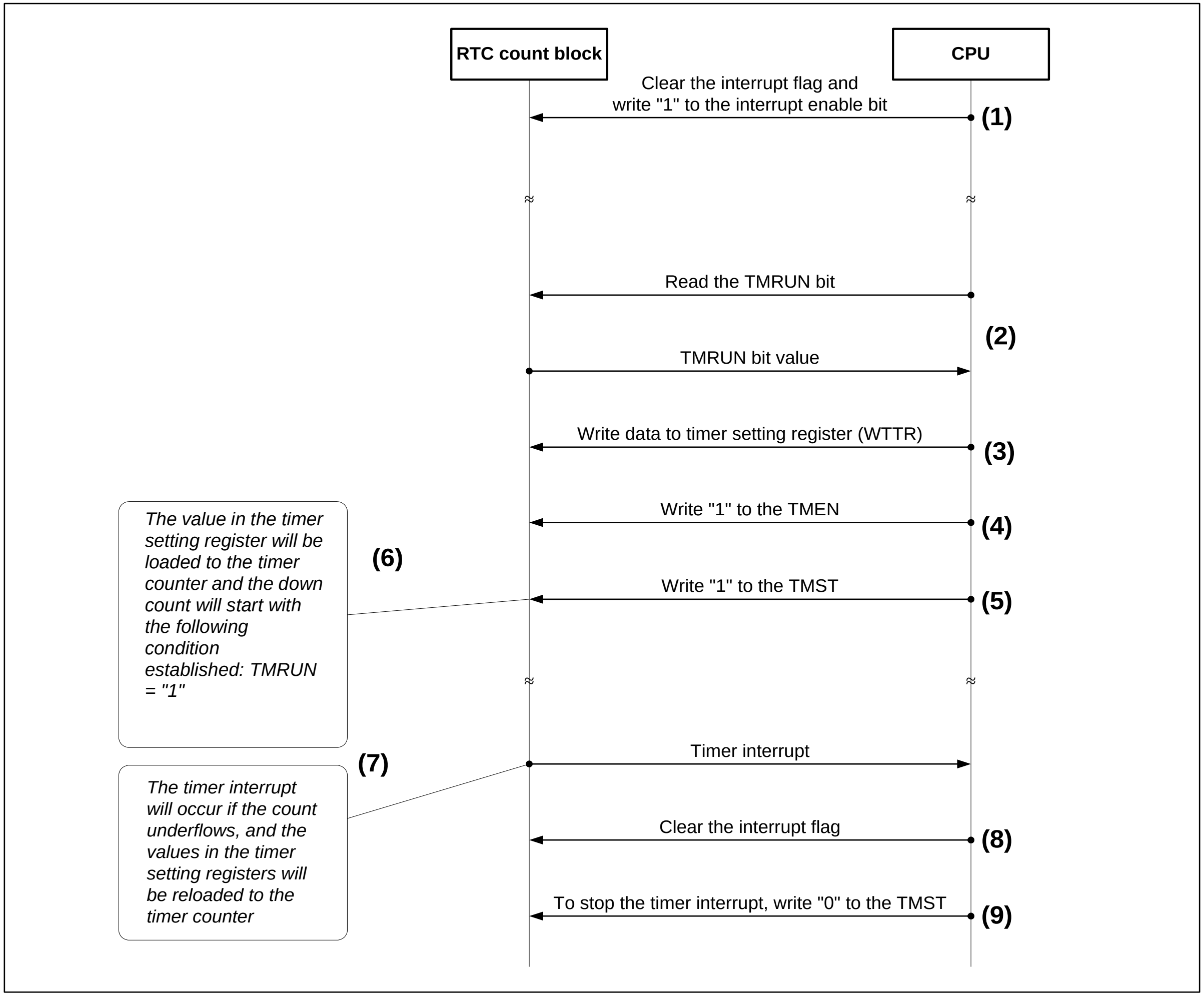
(1) Clear the timer interrupt flag bit with the following condition written: INTTMI = "0".
The timer interrupt is enabled with the following condition written: INTTMIE = "1".
(2) Read the timer counter operation bit (TMRUN) and check whether the value is "0" (not in operation).
(3) Write the desired timer set value to the timer setting register (WTTR).
(4) Write "0" to the timer counter control bit (TMEN).
(5) Write "1" to the timer counter start bit (TMST).
(6) The set value in the timer setting register value will be transferred to the timer counter, and the down count will start.
(7) If the down count underflows, an interrupt request will be generated, and TMST will be cleared to "0". Then the TMRUN will be cleared to "0" after the timer counter is stopped.

<Notes>

- Writing "1" to the TMST is disabled before the TMRUN value becomes "0" after "0" is written to the TMST while the timer counter is in operation (TMRUN = "1").
- TMEN settings can be changed only when the timer counter is not in operation (TMRUN = "0").

■ Example of the Timer Interrupt Setting Procedures (in Intervals of (Hours, Minutes, and Seconds))

Figure 3-8 Setting Procedure of the Timer Interrupt Setting (in Intervals of (Hours, Minutes, and Seconds))


RTC count block
CPU
Clear the interrupt flag and write "1" to the interrupt enable bit
(1)
Read the TMRUN bit
(2)
TMRUN bit value
Write data to timer setting register (WTTR)
(3)
Write "1" to the TMEN
(4)
Write "1" to the TMST
(5)
(6)
The value in the timer setting register will be loaded to the timer counter and the down count will start with the following condition established: TMRUN = "1"
(7)
The timer interrupt will occur if the count underflows, and the values in the timer setting registers will be reloaded to the timer counter
Timer interrupt
Clear the interrupt flag
(8)
To stop the timer interrupt, write "0" to the TMST
(9)

(1) Clear the timer interrupt flag bit with the following condition written: INTTMI = "0".
The timer interrupt is enabled with the following condition written: INTTMIE = "1".
(2) Read the timer counter operation bit (TMRUN) and check whether the value is "0" (not in operation).
(3) Write the desired timer set value to the timer setting register (WTTR).
(4) Write "1" to the timer counter control bit (TMEN).
(5) Write "1" to the timer counter start bit (TMST).
(6) The set value in the timer setting register value will be transferred to the timer counter, and the down count will start.
(7) The RTC count block interrupt request will be generated on completion of the count, and the timer setting register value will be reloaded to the timer counter to continue operation.
(8) Clear the timer interrupt flag bit with the following condition written: INTTMI = "0".
(9) To stop the timer interrupt, write "0" to the TMST.

<Notes>

- Writing "1" to the TMST is disabled before the TMRUN value becomes "0" after "0" is written to the TMST while the timer counter is in operation (TMRUN = "1").
- TMEN settings can be changed only when the timer counter is not in operation (TMRUN = "0").

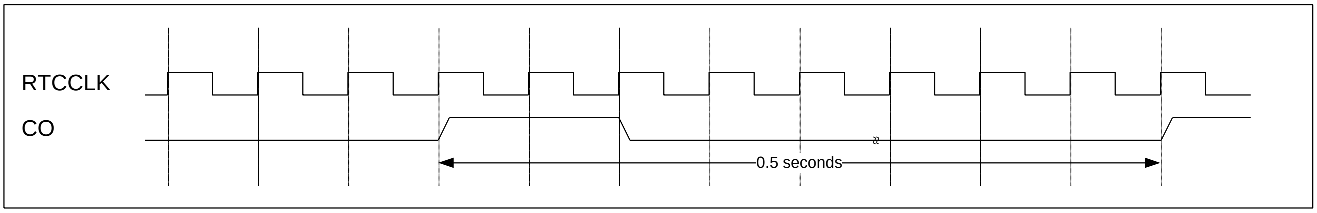
■ Output Operation of the CO External Pin

The RTC count block incorporates an CO external pin for a 0.5-second-clock output.

The CO external pin will output in 0.5-second clock (CO) cycles.

Figure 3-9 shows the waveform of the CO external pin output.

Figure 3-9 Waveform of the CO External Pin Output



4. Resetting of RTC Count Block

This section explains each reset action.

■ Operation of Low-voltage Detection Reset/Power-on Reset

The shaded parts in Table 4-1 are bits that are subject to low-voltage detection resetting/power-on resetting.

The sub-second generation counter, 1-second generation counter, timer counter, and year/month/date/hour/minute/second/day of the week counter are subject to low-voltage detection resetting/power-on resetting as well, though these items are not shown in Table 4-1.

Table 4-1 Low-voltage Detection Reset/Power-on Reset Target bits

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
WTCR1	INTCRIE	INTERIE	INTALIE	INTTMIE	INTHIE	INTMIE	INTSIE	INTSSIE	INTCRI	INTERI	INTALI	INTTMI	INTHI	INTMI	INTSI	INTSSI
	-	-	-	YEN	MOEN	DEN	HEN	MIEN	-	BUSY	SCRST	SCST	SRST	RUN	OE	ST
WTCR2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	TMRUN	TMEN	TMST	-	-	-	-	-	-	-	CREAD
WTBR	-	-	-	-	-	-	-	-	BR23	BR22	BR21	BR20	BR19	BR18	BR17	BR16
	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
WTDR/WTHR/ WTMIR/WTSR	-	-	TD1	TD0	D3	D2	D1	D0	-	-	TH1	TH0	H3	H2	H1	H0
	-	TMI2	TMI1	TMI0	MI3	MI2	MI1	MI0	-	TS2	TS1	TS0	S3	S2	S1	S0
WTYR/WTMOR/ WTDW	-	-	-	-	-	-	-	-	TY3	TY2	TY1	TY0	Y3	Y2	Y1	Y0
	-	-	-	TMO0	MO3	MO2	MO1	MO0	-	-	-	-	-	DW2	DW1	DW0
ALDR/ALHR/ ALMIR	-	-	TAD1	TAD0	AD3	AD2	AD1	AD0	-	-	TAH1	TAH0	AH3	AH2	AH1	AH0
	-	TAMI2	TAMI1	TAMI0	AMI3	AMI2	AMI1	AMI0	-	-	-	-	-	-	-	-
ALYR/ALMOR	-	-	-	-	-	-	-	-	TAY3	TAY2	TAY1	TAY0	AY3	AY2	AY1	AY0
	-	-	-	TAMO0	AMO3	AMO2	AMO1	AMO0	-	-	-	-	-	-	-	-
WTTR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TM17	TM16
	TM15	TM14	TM13	TM12	TM11	TM10	TM9	TM8	TM7	TM6	TM5	TM4	TM3	TM2	TM1	TM0

■ System Reset Operation

The shaded parts in Table 4-2 are bits that are subject to system resetting. The 1-second generation counter and timer counter are subject to system resetting as well, though these items are not shown in Table 4-2.

The sub-second generation counter (for date, time, and timer use) and the year/month/date/hour/minute/second/day of the week counters are not subject to resetting.

Table 4-2 System Reset Target bits

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
WTCR1	INTCRIE	INTERIE	INTALIE	INTTMIE	INTHIE	INTMIE	INTSIE	INTSSIE	INTCRI	INTERI	INTALI	INTTMI	INTHI	INTMI	INTSI	INTSSI
	-	-	-	YEN	MOEN	DEN	HEN	MIEN	-	BUSY	SCRST	SCST	SRST	RUN	OE	ST
WTCR2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	TMRUN	TMEN	TMST	-	-	-	-	-	-	-	CREAD
WTBR	-	-	-	-	-	-	-	-	BR23	BR22	BR21	BR20	BR19	BR18	BR17	BR16
	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
WTDR/WTHR/ WTMIR/WTSR	-	-	TD1	TD0	D3	D2	D1	D0	-	-	TH1	TH0	H3	H2	H1	H0
	-	TMI2	TMI1	TMI0	MI3	MI2	MI1	MI0	-	TS2	TS1	TS0	S3	S2	S1	S0
WTYR/WTMOR/ WTDW	-	-	-	-	-	-	-	-	TY3	TY2	TY1	TY0	Y3	Y2	Y1	Y0
	-	-	-	TMO0	MO3	MO2	MO1	MO0	-	-	-	-	-	DW2	DW1	DW0
ALDR/ALHR/ ALMIR	-	-	TAD1	TAD0	AD3	AD2	AD1	AD0	-	-	TAH1	TAH0	AH3	AH2	AH1	AH0
	-	TAMI2	TAMI1	TAMI0	AMI3	AMI2	AMI1	AMI0	-	-	-	-	-	-	-	-
ALYR/ALMOR	-	-	-	-	-	-	-	-	TAY3	TAY2	TAY1	TAY0	AY3	AY2	AY1	AY0
	-	-	-	TAMO0	AMO3	AMO2	AMO1	AMO0	-	-	-	-	-	-	-	-
WTTR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TM17	TM16
	TM15	TM14	TM13	TM12	TM11	TM10	TM9	TM8	TM7	TM6	TM5	TM4	TM3	TM2	TM1	TM0

■ RTC Reset Operation

The shaded parts in Table 4-3 are bits that are subject to RTC resetting. The 1-second generation counter and timer counter are subject to RTC resetting as well, though these items are not shown in Table 4-3.

The sub-second generation counter (for date, time, and timer use) and the year/month/date/hour/minute/second/day of the week counters are not subject to resetting.

Table 4-3 RTC Reset Target bits

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
WTCR1	INTCRIE	INTERIE	INTALIE	INTTMIE	INTHIE	INTMIE	INTSIE	INTSSIE	INTCRI	INTERI	INTALI	INTTMI	INTHI	INTMI	INTSI	INTSSI
	-	-	-	YEN	MOEN	DEN	HEN	MIEN	-	BUSY	SCRST	SCST	SRST	RUN	OE	ST
WTCR2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	TMRUN	TMEN	TMST	-	-	-	-	-	-	-	CREAD
WTBR	-	-	-	-	-	-	-	-	BR23	BR22	BR21	BR20	BR19	BR18	BR17	BR16
	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
WTDR/WTHR/ WTMIR/WTSR	-	-	TD1	TD0	D3	D2	D1	D0	-	-	TH1	TH0	H3	H2	H1	H0
	-	TMI2	TMI1	TMI0	MI3	MI2	MI1	MI0	-	TS2	TS1	TS0	S3	S2	S1	S0
WTYR/WTMOR/ WTDW	-	-	-	-	-	-	-	-	TY3	TY2	TY1	TY0	Y3	Y2	Y1	Y0
	-	-	-	TMO0	MO3	MO2	MO1	MO0	-	-	-	-	-	DW2	DW1	DW0
ALDR/ALHR/ ALMIR	-	-	TAD1	TAD0	AD3	AD2	AD1	AD0	-	-	TAH1	TAH0	AH3	AH2	AH1	AH0
	-	TAMI2	TAMI1	TAMI0	AMI3	AMI2	AMI1	AMI0	-	-	-	-	-	-	-	-
ALYR/ALMOR	-	-	-	-	-	-	-	-	TAY3	TAY2	TAY1	TAY0	AY3	AY2	AY1	AY0
	-	-	-	TAMO0	AMO3	AMO2	AMO1	AMO0	-	-	-	-	-	-	-	-
WTTR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TM17	TM16
	TM15	TM14	TM13	TM12	TM11	TM10	TM9	TM8	TM7	TM6	TM5	TM4	TM3	TM2	TM1	TM0

5. Leap Year Compliance of RTC Count Block

This section explains the leap year compliance of the RTC count block.

■ Leap Year Compliance

Table 5-1 shows the dates of each month.

Table 5-1 List of Leap Years

Year	Leap year	Month											
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
00	Yes	31	29	31	30	31	30	31	31	30	31	30	31
01 to 03	No	31	28	31	30	31	30	31	31	30	31	30	31
04	Yes	31	29	31	30	31	30	31	31	30	31	30	31
05 to 07	No	31	28	31	30	31	30	31	31	30	31	30	31
08	Yes	31	29	31	30	31	30	31	31	30	31	30	31
09 to 11	No	31	28	31	30	31	30	31	31	30	31	30	31
12	Yes	31	29	31	30	31	30	31	31	30	31	30	31
13 to 15	No	31	28	31	30	31	30	31	31	30	31	30	31
16	Yes	31	29	31	30	31	30	31	31	30	31	30	31
17 to 19	No	31	28	31	30	31	30	31	31	30	31	30	31
20	Yes	31	29	31	30	31	30	31	31	30	31	30	31
21 to 23	No	31	28	31	30	31	30	31	31	30	31	30	31
24	Yes	31	29	31	30	31	30	31	31	30	31	30	31
25 to 27	No	31	28	31	30	31	30	31	31	30	31	30	31
28	Yes	31	29	31	30	31	30	31	31	30	31	30	31
29 to 31	No	31	28	31	30	31	30	31	31	30	31	30	31
32	Yes	31	29	31	30	31	30	31	31	30	31	30	31
33 to 35	No	31	28	31	30	31	30	31	31	30	31	30	31
36	Yes	31	29	31	30	31	30	31	31	30	31	30	31
37 to 39	No	31	28	31	30	31	30	31	31	30	31	30	31
40	Yes	31	29	31	30	31	30	31	31	30	31	30	31
41 to 43	No	31	28	31	30	31	30	31	31	30	31	30	31
44	Yes	31	29	31	30	31	30	31	31	30	31	30	31
45 to 47	No	31	28	31	30	31	30	31	31	30	31	30	31
48	Yes	31	29	31	30	31	30	31	31	30	31	30	31
49 to 51	No	31	28	31	30	31	30	31	31	30	31	30	31

Year	Leap year	Month											
		1	2	3	4	5	6	7	8	9	10	11	12
52	Yes	31	29	31	30	31	30	31	31	30	31	30	31
53 to 55	No	31	28	31	30	31	30	31	31	30	31	30	31
56	Yes	31	29	31	30	31	30	31	31	30	31	30	31
57 to 59	No	31	28	31	30	31	30	31	31	30	31	30	31
60	Yes	31	29	31	30	31	30	31	31	30	31	30	31
61 to 63	No	31	28	31	30	31	30	31	31	30	31	30	31
64	Yes	31	29	31	30	31	30	31	31	30	31	30	31
65 to 67	No	31	28	31	30	31	30	31	31	30	31	30	31
68	Yes	31	29	31	30	31	30	31	31	30	31	30	31
69 to 71	No	31	28	31	30	31	30	31	31	30	31	30	31
72	Yes	31	29	31	30	31	30	31	31	30	31	30	31
73 to 75	No	31	28	31	30	31	30	31	31	30	31	30	31
76	Yes	31	29	31	30	31	30	31	31	30	31	30	31
77 to 79	No	31	28	31	30	31	30	31	31	30	31	30	31
80	Yes	31	29	31	30	31	30	31	31	30	31	30	31
81 to 83	No	31	28	31	30	31	30	31	31	30	31	30	31
84	Yes	31	29	31	30	31	30	31	31	30	31	30	31
85 to 87	No	31	28	31	30	31	30	31	31	30	31	30	31
88	Yes	31	29	31	30	31	30	31	31	30	31	30	31
89 to 91	No	31	28	31	30	31	30	31	31	30	31	30	31
92	Yes	31	29	31	30	31	30	31	31	30	31	30	31
93 to 95	No	31	28	31	30	31	30	31	31	30	31	30	31
96	Yes	31	29	31	30	31	30	31	31	30	31	30	31
97 to 99	No	31	28	31	30	31	30	31	31	30	31	30	31

6. Time Rewrite Error

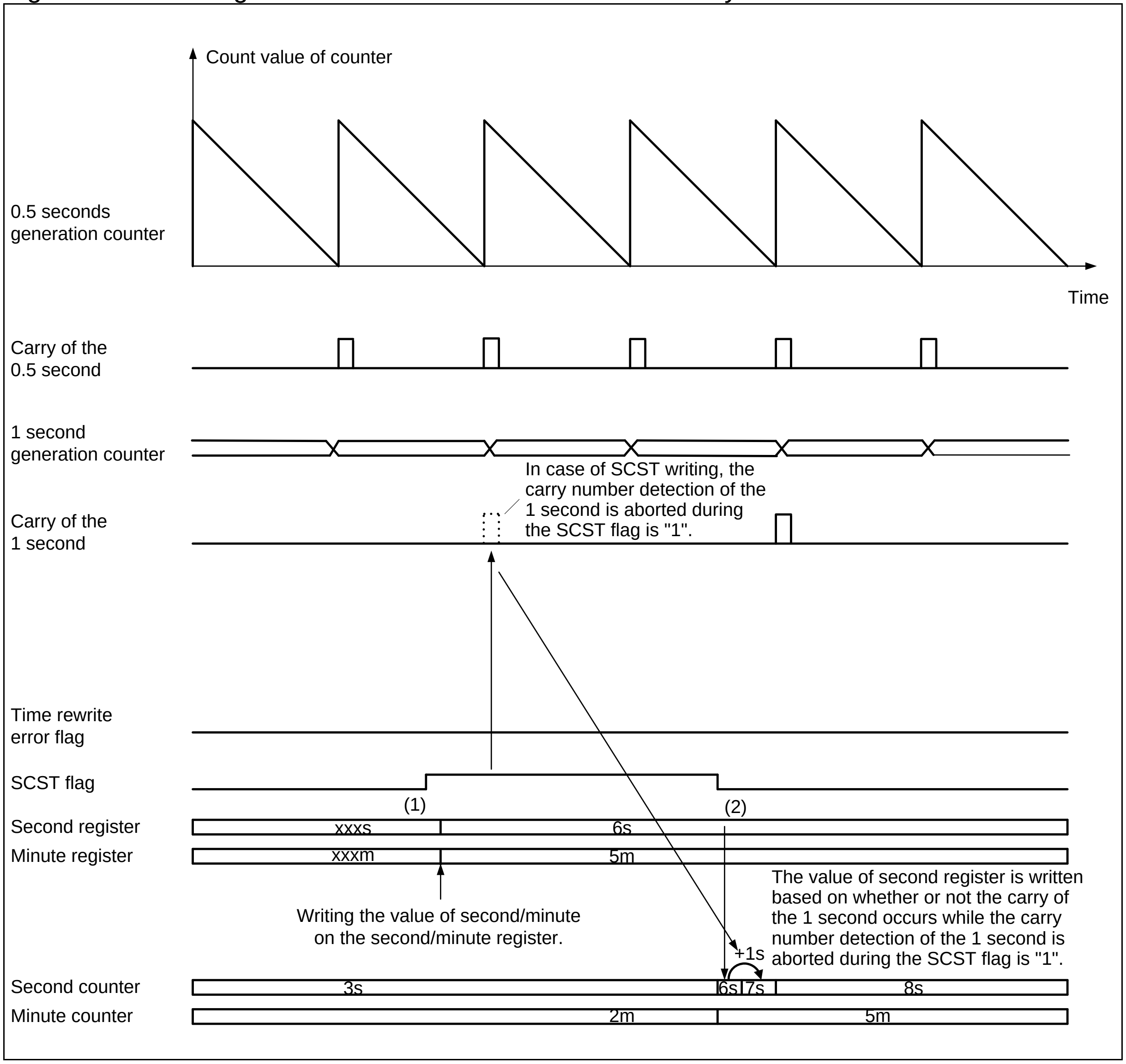
This section explains the time rewrite error during time rewriting (with time count continued).

■ Time Rewrite Error 1

The following provides an example that the carry number of the 0.5 seconds generation counter is two while the SCST flag is set to "1" during time rewriting (with time count continued).

· Rewriting second counter and minute counter only

Figure 6-1 Rewriting second counter and minute counter only


Count value of counter
0.5 seconds generation counter
Time
Carry of the 0.5 second
1 second generation counter
In case of SCST writing, the carry number detection of the 1 second is aborted during the SCST flag is "1".
Carry of the 1 second
Time rewrite error flag
SCST flag
(1)
(2)
Second register
xxxs
6s
Minute register
xxxm
5m
Writing the value of second/minute on the second/minute register.
The value of second register is written based on whether or not the carry of the 1 second occurs while the carry number detection of the 1 second is aborted during the SCST flag is "1".
+1s
Second counter
3s
6s
7s
8s
Minute counter
2m
5m

(1) Set the SCST flag to "1", and write 6 seconds on the second register and 5 minutes on the minute register.

(2) When the carry number of the 0.5 seconds is two while the SCST flag is set to "1" and then the SCST flag is set to "0". 6 seconds will be written on the second counter, then 1 is added and consequently the counter will be 7 seconds.

<Notes>

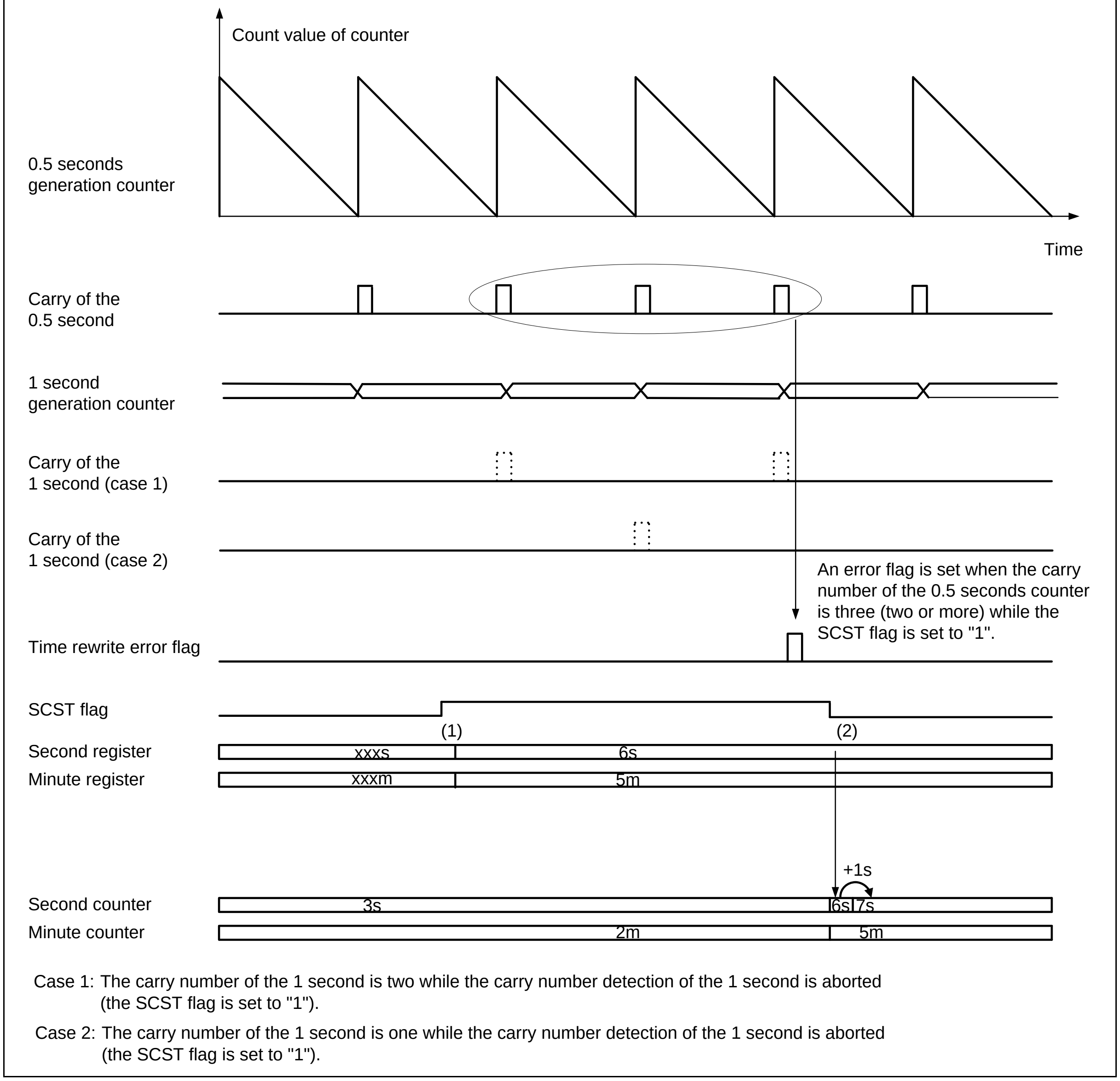
- If the carry number of the 0.5 seconds is two or less while the SCST flag is set to "1", the time rewrite error flag does not become "1".
- Because the carry number detection of the 1 second is aborted while the SCST flag is set to "1", the second counter does not count. When the carry number detection of the 1 second is aborted, the carry number of the 1 second is saved. Then, the value of second register is written on the second counter when the SCST flag is set to "0", and then 1 is added. If there is no carry of the 1 second while the SCST flag is set to "1", the value of second register is written on the second counter, however, 1 is not added.

■ Time Rewrite Error 2

The following provides an example that the carry number of the 0.5 seconds generation counter is three while the SCST flag is set to "1" during time rewriting (with time count continued).

- Rewriting second counter and minute counter only

Figure 6-2 Rewriting second counter and minute counter only


Count value of counter
0.5 seconds generation counter
Time
Carry of the 0.5 second
1 second generation counter
Carry of the 1 second (case 1)
Carry of the 1 second (case 2)
An error flag is set when the carry number of the 0.5 seconds counter is three (two or more) while the SCST flag is set to "1".
Time rewrite error flag
SCST flag
(1)
(2)
Second register
xxxs
6s
Minute register
xxxm
5m
+1s
Second counter
3s
6s
7s
Minute counter
2m
5m

Case 1: The carry number of the 1 second is two while the carry number detection of the 1 second is aborted (the SCST flag is set to "1").

Case 2: The carry number of the 1 second is one while the carry number detection of the 1 second is aborted (the SCST flag is set to "1").

(1) Set the SCST flag to "1", and write 6 seconds on the second register and 5 minutes on the minute register.

(2) When the carry number of the 0.5 seconds is three while the SCST flag is set to "1" and then the SCST flag is set to "0", the time rewrite error flag will be set to "1". 6 seconds will be written on the second counter, then 1 is added and consequently the counter will be 7 seconds.

<Note>

When the time rewrite error flag is set to "1", 1 second shift will occur for case 1 because the carry number of the 1 second is two while the carry number detection of the 1 second is aborted (the SCST flag is set to "1"). 1 second shift will not occur for case 2 because the carry number of the 1 second is one while the carry number detection of the 1 second is aborted (the SCST flag is set to "1"). However, if the time rewrite error occurred, perform time rewriting again because you cannot determine which case happened.

7. Registers of RTC Count Block

The Registers of the RTC count block are listed below.

■ List of Registers of the RTC Count Block

Table 7-1 List of Registers of the RTC Count Block

Abbreviated register name	Register name	Reference
WTCR1	Control register 1	7.1
WTCR2	Control register 2	7.2
WTBR	Counter cycle setting register	7.3
WTDR	Date register	7.4
WTHR	Hour register	7.5
WTMIR	Minute register	7.6
WTSR	Second register	7.7
WTYR	Year register	7.8
WTMOR	Month register	7.9
WTDW	Day of the week register	7.10
ALDR	Alarm date register	7.11
ALHR	Alarm hour register	7.12
ALMIR	Alarm minute register	7.13
ALYR	Alarm year register	7.14
ALMOR	Alarm month register	7.15
WTTR	Timer setting register	7.16

7.1. Control Register 1 (WTCR1)

This register is used to control the operation of the RTC count block.

bit	31	30	29	28	27	26	25	24
Field	INTCRIE	INTERIE	INTALIE	INTTMIE	INTHIE	INTMIE	INTSIE	INTSSIE
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

bit	23	22	21	20	19	18	17	16
Field	INTCRI	INTERI	INTALI	INTTMI	INTHI	INTMI	INTSI	INTSSI
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	Reserved			YEN	MOEN	DEN	HEN	MIEN
Attribute	R			R/W	R/W	R/W	R/W	R/W
Initial value	000			0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	Reserved	BUSY	SCRST	SCST	SRST	RUN	Reserved	ST
Attribute	R	R	R/W	R/W	R/W	R	R	R/W
Initial value	0	0	0	0	0	0	0	0

[bit31] INTCRIE: Year/month/date/hour/minute/second/day of the week counter value read completion interrupt enable bit

This is the year/month/date/hour/minute/second/day of the week counter value read completion interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit30] INTERIE: Time rewrite error interrupt enable bit

This is the time rewrite error interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit29] INTALIE: Alarm interrupt enable bit

This is the alarm interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit28] INTTMIE: Timer interrupt enable bit

This is the timer interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit27] INTHIE: 1-hour interrupt enable bit

This is the 1-hour interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit26] INTMIE: 1-minute interrupt enable bit

This is the 1-minute interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit25] INTSIE: 1-second interrupt enable bit

This is the 1-second interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit24] INTSSIE: 0.5-second interrupt enable bit

This is the 0.5-second interrupt enable bit.

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit23] INTCRI: Year/month/date/hour/minute/second/day of the week counter value read completion interrupt flag bit

This bit indicates the completion of the transfer of the year/month/date/hour/minute/second/day of the week counter values to the year/month/date/hour/minute/second/day of the week registers at the time of the date and time reading with the CREAD bit.

Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	The year/month/date/hour/minute/second/day of the week counter value read has not been completed
	1	The year/month/date/hour/minute/second/day of the week counter value read has been completed
Write	0	This flag is cleared
	1	No effect on operation

[bit22] INTERI: Time rewrite error interrupt flag bit

This bit shows that the second counter has not been counting normally during time rewriting (SCST = 1).

Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	No time rewriting error has been generated
	1	A time rewriting error has been generated
Write	0	This flag is cleared
	1	No effect on operation

[bit21] INTALI: Alarm coinciding flag bit

This bit shows that the year/month/date/hour/minute register value coincides with the year/month/date/hour/minute counter value.

Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	No alarm coinciding has been generated
	1	An alarm coinciding has been generated
Write	0	This flag is cleared
	1	No effect on operation

[bit20] INTTMI: Timer underflow flag bit

This flag will be set to "1" if the timer counter underflows.
Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	No timer underflow has been generated
	1	A timer underflow has been generated
Write	0	This flag is cleared
	1	No effect on operation

[bit19] INTHI: 1-hour flag bit

This flag will be set to "1" if the 1-hour counter is up.
Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	No 1-hour count-up has been generated
	1	A 1-hour count-up has been generated
Write	0	This flag is cleared
	1	No effect on operation

[bit18] INTMI: 1-minute flag bit

This flag will be set to "1" if the 1-minute counter is up.
Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	No 1-minute count-up has been generated
	1	A 1-minute count-up has been generated
Write	0	This flag is cleared
	1	No effect on operation

[bit17] INTSI: 1-second flag bit

This flag will be set to "1" if the 1-second counter is up.
Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	No 1-second count-up has been generated
	1	A 1-second count-up has been generated
Write	0	This flag is cleared
	1	No effect on operation

[bit16] INTSSI: 0.5-second flag bit

This flag will be set to "1" if the 0.5-second counter is up.
Always "1" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	No 0.5-second count-up has been generated
	1	A 0.5-second count-up has been generated
Write	0	This flag is cleared
	1	No effect on operation

[bit15:13] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit12] YEN: Alarm year register enable bit

This bit enables the comparison of the alarm year register and the year counter in the alarm interrupt.
When this bit is set to "1", this can be detected by the alarm interrupt flag (INTALI).

Value	Description
0	The alarm year register is disabled
1	The alarm year register is enabled

[bit11] MOEN: Alarm month register enable bit

This bit enables the comparison of the alarm month register and the month counter.
When this bit is set to "1", this can be detected by the alarm coinciding flag (INTALI).

Value	Description
0	The alarm month register is disabled.
1	The alarm month register is enabled.

[bit10] DEN: Alarm date register enable bit

This bit enables the comparison of the alarm date register and the date counter.
When this bit is set to "1", this can be detected by the alarm coinciding flag (INTALI).

Value	Description
0	The alarm date register is disabled.
1	The alarm date register is enabled.

[bit9] HEN: Alarm hour register enable bit

This bit enables the comparison of the alarm hour register and the hour counter.
When this bit is set to "1", this can be detected by the alarm coinciding flag (INTALI).

Value	Description
0	The alarm hour register is disabled.
1	The alarm hour register is enabled.

[bit8] MIEN: Alarm minute register enable bit

This bit enables the comparison of the alarm minute register and the minute counter.
When this bit is set to "1", this can be detected by the alarm coinciding flag (INTALI).

Value	Description
0	The alarm minute register is disabled.
1	The alarm minute register is enabled.

[bit7] Reserved: Reserved bit

Always "0" is read.
Set "0" when writing data.

[bit6] BUSY: Busy bit

This bit indicates that time rewriting is in process.

Value	Description
0	Time rewriting is not in process
1	One of the following conditions: ・ SCST = "1" ・ SCRST = "1" ・ The set values in the year/month/date/hour/minute/second/day of the week registers are being transferred to the year/month/date/hour/minute/second/day of the week counters

[bit5] SCRST: Sub second generation/1-second generation counter reset bit

This bit performs the reset control of the sub-second generation/1-second generation counter (for date and time use).

Value	Description
0	This bit releases the reset state of the sub-second generation/1-second generation counter (for date and time use)
1	This bit resets the sub-second generation/1-second generation counter (for date and time use)

When this bit is "0" and SCST bit is "0" while RTC is operating (RUN=1), the values in the year/month/date/hour/minute/second/day of the week registers cannot be refreshed. SCST bit and SCRST bit must not be simultaneously set to "1". This bit must not be set to "1" while RTC is stopped (RUN=0).

[bit4] SCST: 1-second clock output stop bit

This bit controls the 1-second clock output of the 1-second generation counter.

Value	Description
0	Output is enabled
1	Output is disabled

When this bit is "0" and SCRST bit is "0" while RTC is operating (RUN=1), the values in the year/month/date/hour/minute/second/day of the week registers cannot be refreshed. SCST bit and SCRST bit must not be simultaneously set to "1". This bit must not be set to "1" while RTC is stopped (RUN=0).

[bit3] SRST: RTC reset bit

See Table 4-3 in “4. Reset Operation of RTC Count Block” for the register bits initialized by the RTC reset function.
Always "0" will be read at the time of read access in the read modify write access mode.

Value	Description
0	When RTC reset is completed
1	The hardware will issue an RTC reset when "1" is written

[bit2] RUN: RTC count block operation bit

This bit indicates the operating state of the RTC count block.
If the ST is set to "0" while the RTC count block is in operation with ST="1", the RTC count block will come to a stop and the following condition will be established: RUN = "0".

Value	Description
0	The RTC count block is not in operation
1	The RTC count block is in operation

[bit1] Reserved: Reserved bit

Always "0" is read.
Set "0" when writing data.

[bit0] ST: Start bit

This bit controls the operation start of the RTC count block.

Value	Description
0	The RTC count block comes to a stop.
1	The set values in the year/month/date/hour/minute/second/day of the week registers are transferred to the year/month/date/hour/minute/second/day of the week counters, and the RTC count block will start operating.

7.2. Control Register 2 (WTCR2)

This register controls the operation of the RTC count block.

bit	31	30	29	28	27	26	25	24
Field	Reserved							
Attribute	R							
Initial value	0x00							

bit	23	22	21	20	19	18	17	16
Field	Reserved							
Attribute	R							
Initial value	0x00							

bit	15	14	13	12	11	10	9	8
Field	Reserved					TMRUN	TMEN	TMST
Attribute	R					R	R/W	R/W
Initial value	00000					0	0	0

bit	7	6	5	4	3	2	1	0
Field	Reserved							CREAD
Attribute	R							R/W
Initial value	0000000							0

[bit31:11] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit10] TMRUN: Timer counter operation bit

This bit indicates the operation of the timer counter.
While the time counter control bit is "0", this bit will be cleared by the hardware if the count underflows. While the timer counter control bit(TMEN) is "1", this bit will be remain "1" until the timer counter start bit (TMST) becomes "0".
If TMST bit is set to "0" while the timer is in operation with the TMST bit set to "1", the timer will stop operating and the following condition will be established: TMRUN = "0".

Value	Description
0	The timer counter is not in operation
1	The timer counter is in operation

[bit9] TMEN: Timer counter control bit

This bit selects the control of the timer counter operating with an elapse of the desired time (with the hours, minutes, and seconds specified) or the timer counter in the desired intervals (with the hours, minutes, and seconds specified).

Value	Description
0	The timer counter will operate with an elapse of the desired time (with the hours, minutes, and seconds specified).
1	The timer counter will operate in intervals of the desired time (with the hours, minutes, and seconds specified).

[bit8] TMST: Timer counter start bit

This bit starts the operation of the timer counter.
When the time counter control bit is "0", this bit will be "0" cleared by the hardware when the count completes.
See the timer counter operation bit (TMRUN) for the operation state of the timer counter. In the case of rewriting the timer setting register, stop this bit once with "0" and rewrite the timer setting register, and reset the timer setting register to "1" to start the operation of the timer setting register.

Value	Description
0	The timer counter stops operating.
1	The timer counter starts operating.

[bit7:1] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit0] CREAD: Year/month/date/hour/minute/second/day of the week counter value read control bit

The values in the year/month/date/hour/minute/second/day of the week counters will be transferred to the year/month/date/hour/minute/second/day of the week registers if this bit is set to "1" and the bit will be "0" cleared on completion of the transfer.
Always "0" will be read at the time of the read access in the read modify write access mode.

Process		Description
Read	0	Data transfer from the year/month/date/hour/minute/second/day of the week counters to the year/month/date/hour/minute/second/day of the week registers has been completed.
	1	Data transfer from the year/month/date/hour/minute/second/day of the week counters to the year/month/date/hour/minute/second/day of the week registers is in process.
Write	0	No effect on operation.
	1	Data copy from the year/month/date/hour/minute/second/day of the week counters to the year/month/date/hour/minute/second/day of the week registers starts.

7.3. Counter Cycle Setting Register (WTBR)

This register stores values to be reloaded to the sub-second generation counter (for date, time, and timer use).

bit	31	30	29	28	27	26	25	24
Field	Reserved							
Attribute	R							
Initial value	0x00							

bit	23	22	21	20	19	18	17	16
Field	BR23	BR22	BR21	BR20	BR19	BR18	BR17	BR16
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

bit	15	14	13	12	11	10	9	8
Field	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit31:24] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit23:0] BR23 to BR0: Counter cycle setting bits

These bits set values to be reloaded to the sub-second generation counter (for date, time, and timer use).
Set this register to a 0.5-second count value for the sub-second generation counter. The value will be reloaded when the sub-second generation counter value becomes "0".
Obtain the Counter Cycle Setting Register (WTBR) set value from the following formula:

WTBR = (0.5 [s] / (2 × RTCCLK cycles [s])) - 1

<Notes>

- In the case of setting the WTBR register, make sure that the ST value is "0" (RTC count block is not in operation) and the TMST value is "0" (timer counter is not in operation).
- Set the counter cycle setting register to value of "7" or larger. If a value of "6" or less is set, the right value will not be read by the year/month/date/hour/minute/second/day of the week counter value read executed.

7.4. Date Register (WTDR)

This register indicates the date information in the RTC count block. The register value is displayed in BCD.

bit	7	6	5	4	3	2	1	0
Field	Reserved		TD1	TD0	D3	D2	D1	D0
Attribute	R		R/W	R/W	R/W	R/W	R/W	R/W
Initial value	00		0	0	0	0	0	0

[bit7:6] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit5:4] TD1, TD0: Date register

This register stores the second digit of the date information in the RTC count block.

[bit3:0] D3 to D0: Date register

This register stores the first digit of the date information in the RTC count block.
0 to 9: Enabled
A to F: Setting disabled

7.5. Hour register (WTHR)

This register indicates the hour information in the RTC count block. The register value is displayed in BCD.

bit	7	6	5	4	3	2	1	0
Field	Reserved		TH1	TH0	H3	H2	H1	H0
Attribute	R		R/W	R/W	R/W	R/W	R/W	R/W
Initial value	00		0	0	0	0	0	0

[bit7:6] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit5:4] TH1, TH0: Hour register

This register stores the second digit of the hour information in the RTC count block.
0 to 2: Enabled
3: Setting disabled

[bit3:0] H3 to H0: Hour register

This register stores the first digit of the hour information in the RTC count block.
0 to 9: Enabled
A to F: Setting disabled

7.6. Minute Register (WTMIR)

This register indicates the minute information in the RTC count block. The register value is displayed in BCD.

bit	7	6	5	4	3	2	1	0
Field	Reserved	TMI2	TMI1	TMI0	MI3	MI2	MI1	MI0
Attribute	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] Reserved: Reserved bit

Always "0" is read.
Set "0" when writing data.

[bit6:4] TMI2 to TMI0: Minute register

This register stores the second digit of the minute information in the RTC count block.
0 to 5: Enabled
6, 7: Setting disabled

[bit3:0] MI3 to MI0: Minute register

This register stores the first digit of the minute information in the RTC count block.
0 to 9: Enabled
A to F: Setting disabled

7.7. Second Register (WTSR)

This register indicates the second information in the RTC count block. The register value is displayed in BCD.

bit	7	6	5	4	3	2	1	0
Field	Reserved	TS2	TS1	TS0	S3	S2	S1	S0
Attribute	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] Reserved: Reserved bit

Always "0" is read.
Set "0" when writing data.

[bit6:4] TS2 to TS0: Second register

This register stores the second digit of the second information in the RTC count block.
0 to 5: Enabled
6, 7: Setting disabled

[bit3:0] S3 to S0: Second register

This register stores the first digit of the second information in the RTC count block.
0 to 9: Enabled
A to F: Setting disabled

7.8. Year Register (WTYR)

This register indicates the year information in the RTC count block. The register value is displayed in BCD.

bit	7	6	5	4	3	2	1	0
Field	TY3	TY2	TY1	TY0	Y3	Y2	Y1	Y0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7:4] TY3 to TY0: Year register

This register stores the second digit of the year information in the RTC count block.
0 to 9: Enabled
A to F: Setting disabled

[bit3:0] Y3 to Y0: Year register

This register stores the first digit of the year information in the RTC count block.
0 to 9: Enabled
A to F: Setting disabled

7.9. Month Register (WTMOR)

This register indicates the month information in the RTC count block. The register value is displayed in BCD.

bit	7	6	5	4	3	2	1	0
Field		Reserved		TMO0	MO3	MO2	MO1	MO0
Attribute		R		R/W	R/W	R/W	R/W	R/W
Initial value		000		0	0	0	0	0

[bit7:5] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit4] TMO0: Month register

This register stores the second digit in the month information in the RTC count block.

[bit3:0] MO3 to MO0: Month register

This register stores the first digit of the month information in the RTC count block.
0 to 9: Enabled
A to F: Setting disabled

7.10. Day of the Week Register (WTDW)

This register indicates the day of the week information in the RTC count block. The register value is displayed in BCD.

bit	7	6	5	4	3	2	1	0
Field	Reserved					DW2	DW1	DW0
Attribute	R					R/W	R/W	R/W
Initial value	00000					0	0	0

[bit7:3] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit2:0] DW2 to DW0: Day of the week register

This register stores the day information in the RTC count block.

"0": Sunday
"1": Monday
"2": Tuesday
"3": Wednesday
"4": Thursday
"5": Friday
"6": Saturday
"7": Setting disabled

7.11. Alarm Date Register (ALDR)

This register indicates the alarm-set date information.

bit	7	6	5	4	3	2	1	0
Field	Reserved		TAD1	TAD0	AD3	AD2	AD1	AD0
Attribute	R		R/W	R/W	R/W	R/W	R/W	R/W
Initial value	00		0	0	0	0	0	0

[bit7:6] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit5:4] TAD1 to TAD0: Alarm date register

This register stores the second digit of the alarm-set date information.

[bit3:0] AD3 to AD0: Alarm date register

This register stores the first digit of the alarm-set date information.
0 to 9: Enabled
A to F: Setting disabled

7.12. Alarm Hour Register (ALHR)

This register indicates the alarm-set hour information.

bit	7	6	5	4	3	2	1	0
Field	Reserved		TAH1	TAH0	AH3	AH2	AH1	AH0
Attribute	R		R/W	R/W	R/W	R/W	R/W	R/W
Initial value	00		0	0	0	0	0	0

[bit7:6] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit5:4] TAH1, TAH0: Alarm hour register

This register stores the second digit of the alarm-set hour information.
0 to 2: Enabled
3 to F: Setting disabled

[bit3:0] AH3 to AH0: Alarm hour register

This register stores the first digit of the alarm-set hour information.
0 to 9: Enabled
A to F: Setting disabled

7.13. Alarm Minute Register (ALMIR)

This register indicates the alarm-set minute information.

bit	7	6	5	4	3	2	1	0
Field	Reserved	TAMI2	TAMI1	TAMI0	AMI3	AMI2	AMI1	AMI0
Attribute	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] Reserved: Reserved bit

Always "0" is read.

[bit6:4] TAMI2 to TAMI0: Alarm minute register

This register stores the second digit of the alarm-set minute information.
0 to 5: Enabled
6, 7: Setting disabled

[bit3:0] AMI3 to AMI0: Alarm minute register

This register stores the first digit of the alarm-set minute information.
0 to 9: Enabled
A to F: Setting disabled

7.14. Alarm Years Register (ALYR)

This register indicates the alarm-set year information.

bit	7	6	5	4	3	2	1	0
Field	TAY3	TAY2	TAY1	TAY0	AY3	AY2	AY1	AY0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7:4] TAY3 to TAY0: Alarm year register

This register stores the second digit of the alarm-set year information.
0 to 9: Enabled
A to F: Setting disabled

[bit3:0] AY3 to AY0: Alarm year register

This register stores the first digit of the alarm-set year information.
0 to 9: Enabled
A to F: Setting disabled

7.15. Alarm Month Register (ALMOR)

This register indicates the alarm-set month information.

bit	7	6	5	4	3	2	1	0
Field	Reserved			TAMO0	AMO3	AMO2	AMO1	AMO0
Attribute	R			R/W	R/W	R/W	R/W	R/W
Initial value	000			0	0	0	0	0

[bit7:5] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit4] TAMO0: Alarm month register

This register stores the second digit of the alarm-set month information.

[bit3:0] AMO3 to AMO0: Alarm month register

This register stores the first digit of the alarm-set month information.
0 to 9: Enabled
A to F: Setting disabled

7.16. Timer Setting Register (WTTR)

This register is used to set a timer set value that will be up with an elapse of the set value (hours, minutes, and seconds) or in intervals of the set value (hours, minutes, and seconds).
A value ranging from 1 second up to 1 day can be set.

bit	31	30	29	28	27	26	25	24
Field	Reserved							
Attribute	R							
Initial value	0x00							

bit	23	22	21	20	19	18	17	16
Field	Reserved						TM17	TM16
Attribute	R						R/W	R/W
Initial value	000000						0	0

bit	15	14	13	12	11	10	9	8
Field	TM15	TM14	TM13	TM12	TM11	TM10	TM9	TM8
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TM7	TM6	TM5	TM4	TM3	TM2	TM1	TM0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit31:18] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit17:0] TM17 to TM0: Timer setting register

This register is a timer setting information bit.
This register is set to store a 1-day timer set value that will be up with an elapse of the set value (hours, minutes, and seconds) or in intervals of the set value (hours, minutes, and seconds).
The available timer value is from 1 second to 1 day at 0.5 seconds intervals.
Obtain the set value for the timer setting register from the following formula:
TM [17:0] = (Set time [s] × 2) - 1

1 to 172799: Enabled
0,172800 to 262143: Setting disabled

8. Usage Precautions

Keep the following points in mind when using the RTC count block.

- Use under the following frequency condition: PCLK2 ≥ RTCCLK / 2
- Change settings for each alarm register when any of the alarm interrupt data control bits (WTCR1:YEN, WTCR1:MOEN, WTCR1:DEN, WTCR1:HEN, WTCR1:MIEN) is "0".
- An interrupt may occur immediately after any of the alarm interrupt data control bits (WTCR1:YEN, WTCR1:MOEN, WTCR1:DEN, WTCR1:HEN, and WTCR1:MIEN) is set to "1". After the interrupt, confirm that the value of Year/Month/Date/Hour/Minute counter by reading them.

CHAPTER 4-3: RTC Clock Control Block (A)

This chapter explains the functions and operation of the RTC clock control block (A).

CODE: 9RTCCLKC_A-E01.2

1. Overview of RTC Clock Control Block

This section shows an overview of the functions of the RTC clock control block.

■ RTC Clock Control Block

The RTC clock control block has the following functions:

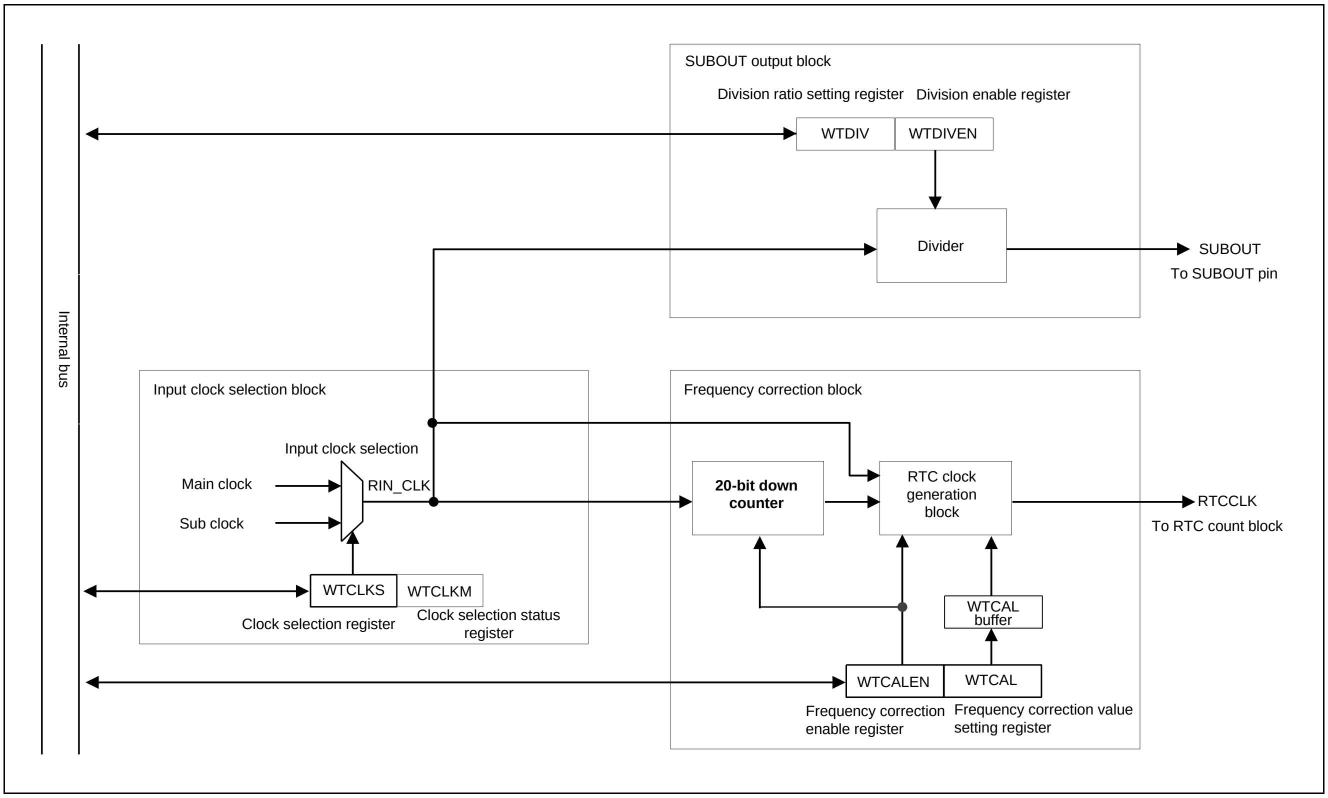
- Generation of a RTC clock (RTCCLK) used for the RTC count block.
- Selection of a main clock and sub clock as an input clock (RIN_CLK).
- Generation of a division clock output for the SUBOUT external pin.
- Frequency correction to compensate for frequency fluctuations resulting from the temperature dependence of the input clock.
  (The above frequency correction function is based on the assumption that a temperature sensor is externally connected.)

2. Configuration of RTC Clock Control Block

The following shows the block diagram.

■ RTC Clock Control Block Diagram

Figure 2-1 RTC Clock Control Block Diagram



● Input Clock Selection Block

Select the input clock (RIN_CLK) as the main clock and sub clock by setting the clock selection register (WTCALS).

● Frequency Correction Block

The frequency correction block masks RIN_CLK and outputs frequency-corrected RTCCLK.
The frequency correction block masks RIN_CLK according the number of the value set in the WTCAL buffer at RIN_CLK × 2^{20} count cycles .

● SUBOUT Output Block

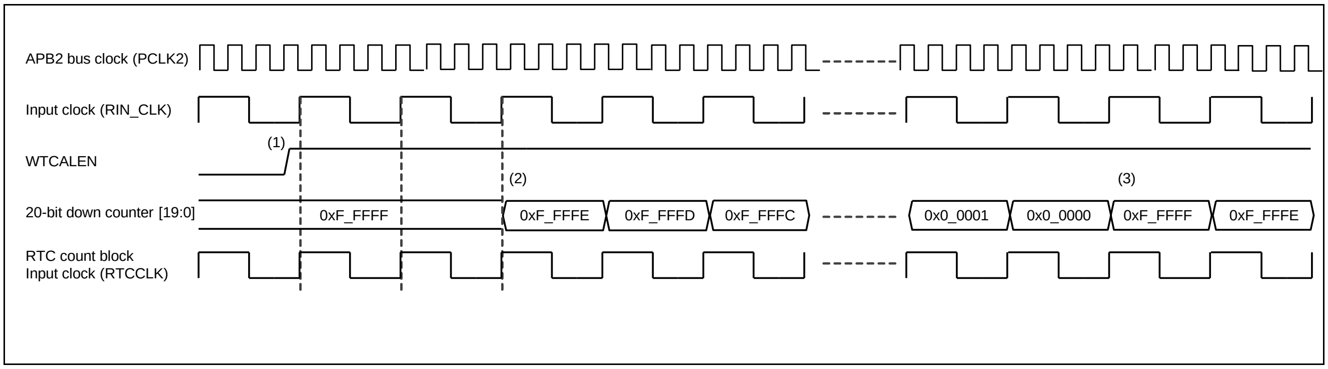
The SUBOUT output block generates a division clock as a SUBOUT external pin output.
No SUBOUT external pin output is possible while in the deep standby RTC mode.

3. Operation of Frequency Correction Block

This section explains the operation of the frequency correction block.

■ Operation of the 20-bit Down Counter

Figure 3-1 Operation Diagram of the 20-bit Down Counter



(1) At the rising edge of the APB2 bus clock (PCLK2), "1" will be written to the frequency correction enable bit (WTCALEN).

(2) The 20-bit down counter will start operating after three input clock (RIN_CLK) cycles from the rising edge of the WTCALEN.

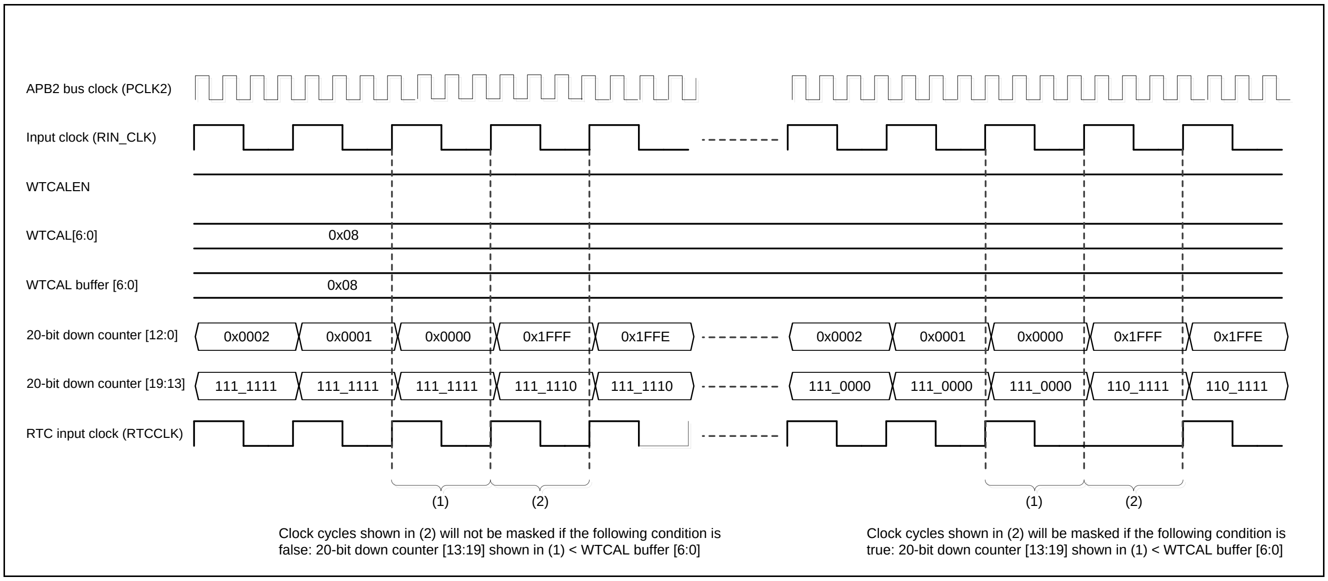
(3) If the 20-bit down counter underflows, it will continue counting down from "0xF_FFFF".

(4) The 20-bit down counter will stop operating when "0" is written to the WTCALEN bit, and the value will be initialized to "0xF_FFFF".

● Operation of the RTC Clock Generation Block

The RTC clock generation block masks clock cycles according to the number of the value set in the WTCAL buffer at 2^{20} count cycles of input clock (RIN_CLK) and performs the frequency correction.
The operation of the RTC clock generation block is shown below:

Figure 3-2 Operation Diagram of the RTC Clock Generation Block (Example: WTCAL = 8)


APB2 bus clock (PCLK2)
Input clock (RIN_CLK)
WTCALEN
WTCAL[6:0]
0x08
WTCAL buffer [6:0]
0x08
20-bit down counter [12:0]
0x0002
0x0001
0x0000
0x1FFF
0x1FFE
0x0002
0x0001
0x0000
0x1FFF
0x1FFE
20-bit down counter [19:13]
111_1111
111_1111
111_1111
111_1110
111_1110
111_0000
111_0000
111_0000
110_1111
110_1111
RTC input clock (RTCCLK)
(1)
(2)
(1)
(2)
Clock cycles shown in (2) will not be masked if the following condition is false: 20-bit down counter [13:19] shown in (1) < WTCAL buffer [6:0]
Clock cycles shown in (2) will be masked if the following condition is true: 20-bit down counter [13:19] shown in (1) < WTCAL buffer [6:0]

(1) The 20-bit down counter [19:13] value and the WTCAL buffer [6:0] value are compared as shown below under the following condition formula: 20-bit down counter [12:0] = "13'h0000"
20-bit down counter [19:13] < WTCAL buffer [6:0]

(2) If the above comparison result is true, the clock next to the RTC clock (RTCCLK) output to the RTC count block will be masked. If false, the next clock will not be masked.

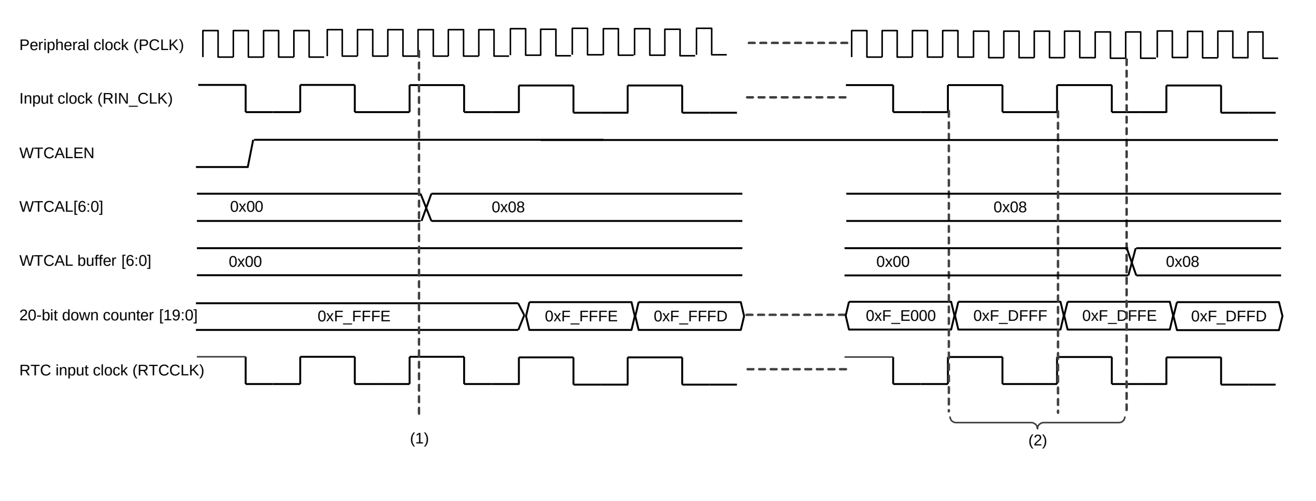
The 20-bit down counter and RTC clock generation block continue operating while the frequency correction enable bit (WTCALEN) is "1". (1) and (2) will be repeated when the 20-bit down counter [12:0] = "13'h0000".
No RTCCLK masking will occur and the condition RTCCLK = RIN_CLK will be accomplished if the WTCALEN bit is "0".

● Operation of the WTCAL Buffer

The frequency correction value setting register (WTCAL) sets the number of clock cycles masked by the frequency correction function.

The frequency correction block incorporates a WTCAL buffer because this block in operation makes the frequency correction value setting register (WTCAL) rewritable.

Figure 3-3 Operation Diagram of the WTCAL Buffer



(1) The WTCAL buffer writes the number of clocks to be masked in the frequency correction value setting register (WTCAL).

(2) The WTCAL value will be transferred to the WTCAL buffer with an elapse of an input clock (RIN_CLK) cycle and three APB2 bus clock (PCLK2) cycles after the following condition is reached: 20-bit down counter [19:0] = 0xFE000

■ Frequency Correction Range

Table 3-1 shows the example of the frequency correction range. It rectifies combining a setup of the WTBR register of a RTC count block, and a WTCAL register.

Table 3-1 Example of Frequency Correction Range (Ideal value)

WTCAL	WTBR=8190		WTBR=8191	
	Correction rate [ppm]	Subclock Frequency [Hz]	Correction rate [ppm]	Subclock Frequency [Hz]
0	122.1	32764.00	0.0	32768.00
1	121.1	32764.03	-1.0	32768.03
2	120.2	32764.06	-1.9	32768.06
:	:	:	:	:
63	62.0	32765.97	-60.1	32769.97
64	61.0	32766.00	-61.0	32770.00
65	60.1	32766.03	-62.0	32770.03
:	:	:	:	:
125	2.9	32767.91	-119.2	32771.91
126	1.9	32767.94	-120.2	32771.94
127	1.0	32767.97	-121.1	32771.97

4. Setting Procedures for RTC Clock Control Block

This section explains the setting procedures for the RTC clock control block.

■ Procedures for the Frequency Correction Settings (with sub clock selected)

(1) Write "0" to the input clock selection bit (WTCLKS) to select the sub clock.

(2) Read the clock selection status bit (WTCLKM) and wait until the sub clock "10" is selected.

(3) Write the correction value to the frequency correction value setting register (WTCAL).
Obtain the set value for the WTCAL from the following formula:
$\text{WTCAL} = \{(\text{Frequency before correction} - (\text{WTBR}+1)\times 4) / \text{Ideal frequency}\} \times 2^{20}$

(4) The frequency correction will be enabled with "1" written to the frequency correction enable bit (WTCALEN).

■ Setting Procedures for the SUBOUT Output Block

(1) Write "0" to the divider output enable bit (WTDIVEN).
The divider will stop operating and the SUBOUT external output will output a low level.

(2) Read the divider status bit (WTDIVRDY) and wait until the value becomes "0" (the divider is not in operation).

(3) Write the divider ratio to the divider ratio setting bit (WTDIV). For divider ratio settings, see "5.5 Divider Ratio Setting Register (WTDIV)".

(4) The divider is set to operation enabled by writing "1" to the divider output enable bit (WTDIVEN).

5. Registers of RTC Clock Control Block

This section shows the list of registers.

■ Registers of the RTC Clock Control Block

Table 5-1 List of Registers of the RTC Clock Control Block

Abbreviated register name	Register name	Reference
WTCLKS	Clock selection register	5.1
WTCLKM	Selection clock status register	5.2
WTCAL	Frequency correction value setting register	5.3
WTCALEN	Frequency correction enable register	5.4
WTDIV	Divider ratio setting register	5.5
WTDIVEN	Divider output enable register	5.6

5.1. Clock Selection Register (WTCLKS)

This register is used for the input clock (RIN_CLK) selection.

bit	7	1	0
Field	Reserved		WTCLKS
Attribute	R		R/W
Initial value	0000000		0

[bit7:1] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit0] WTCLKS: Input clock selection bit

This bit is used for the following input clock (RIN_CLK) selection.

Value	Description
0	Selects the sub clock.
1	Selects the main clock.

<Note>

A software reset, RTC reset, or APB2 bus reset does not initialize this register.

5.2. Selection Clock Status Register (WTCLKM)

This register shows the status of the input clock (RIN_CLK) selection.

bit	7 ... 2	1 ... 0
Field	Reserved	WTCLKM
Attribute	R	R
Initial value	000000	00

[bit7:2] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit1:0] WTCLKM: Clock selection status bits

This bit shows the status of the input clock (RIN_CLK) selection.

Value	Description
0x	The RIN_CLK is not in operation
10	The sub clock is selected.
11	The main clock is selected.

<Note>

A software reset, RTC reset, or APB2 bus reset does not initialize this register.

5.3. Frequency Correction Value Setting Register (WTCAL)

This register sets the frequency correction value for the RTC clock (RTCCLK) output to the RTC count block.

bit	7	6	0
Field	Reserved	WTCAL	
Attribute	R	R/W	
Initial value	0	0000000	

[bit7] Reserved: Reserved bit

Always "0" is read.
Set "0" when writing data.

[bit6:0] WTCAL: Frequency correction value setting bits

The number of cycles to be masked is set from 2^{20} clock cycles (1,048,576 clock cycles).
Eight clocks will be masked from the input clock (RIN_CLK) for each clock cycle of 2^{20} and the RTCCLK output will be generated to the RTC count block if the WTCAL is set to 8.
For the WTCAL set value, see "■Procedures for the Frequency Correction Settings (with sub clock selected)" in "4. Setting Procedures for RTC Clock Control Block".

<Note>

A software reset or APB2 bus reset does not initialize this register.

5.4. Frequency Correction Enable Register (WTCALEN)

This register enables frequency corrections to the RTC clock (RTCCLK) input into the RTC count block.

bit	7	1	0
Field	Reserved		WTCALEN
Attribute	R		R/W
Initial value	0000000		0

[bit7:1] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit0] WTCALEN: Frequency correction enable bit

The frequency correction enable bit enables frequency corrections.

Value	Description
0	Frequency corrections are disabled.
1	Frequency corrections are enabled.

<Note>

A software reset or APB2 bus reset does not initialize this register.

5.5. Divider Ratio Setting Register (WTDIV)

This register sets the divider ratio.

bit	7	4	3	0
Field	Reserved		WTDIV	
Attribute	R		R/W	
Initial value	0000		0000	

[bit7:4] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit3:0] WTDIV: Divider ratio setting bits

The divider ratio setting bit is used to set the divider ratio of the input clock (RIN_CLK) and the divider clock (SUBOUT) that the divider outputs.

Value	Description
0000	No division
0001	Divided by 2
0010	Divided by 4
0011	Divided by 8
0100	Divided by 16
0101	Divided by 32
0110	Divided by 64
0111	Divided by 128
1000	Divided by 256
1001	Divided by 512
Other than the above	Setting disabled

<Note>

Write data to the WTDIV bit when the divider enable bit (WTDIVEN) and the divider output status bit (WTDIVRDY) of the Divider output enable register (WTDIVEN) are set to "0".
A software reset or APB2 bus reset does not initialize this register.

5.6. Divider Output Enable Register (WTDIVEN)

This register enables the divider output.

bit	7	2	1	0
Field	Reserved		WTDIVRDY	WTDIVEN
Attribute	R		R	R/W
Initial value	000000		0	0

[bit7:2] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit1] WTDIVRDY: Divider status bit

This bit shows the operating status of the divider.

Value	Description
0	The divider is not in operation. The SUBOUT external pin output is fixed to Low.
1	The divider is in operation.

[bit0] WTDIVEN: Divider enable bit

This bit is used to enable the operating status of the divider.

Value	Description
0	Stops the divider operation.
1	Enables the operation of the divider.

<Note>

A software reset or APB2 bus reset does not initialize this register.

CHAPTER 4-4: RTC Clock Control Block (B)

This chapter explains the functions and operation of the RTC clock control block (B).

CODE: 9RTCCLKC_B-E01.0

1. Overview of RTC Clock Control Block

This section shows an overview of the functions of the RTC clock control block.

■ RTC Clock Control Block

The RTC clock control block has the following functions:

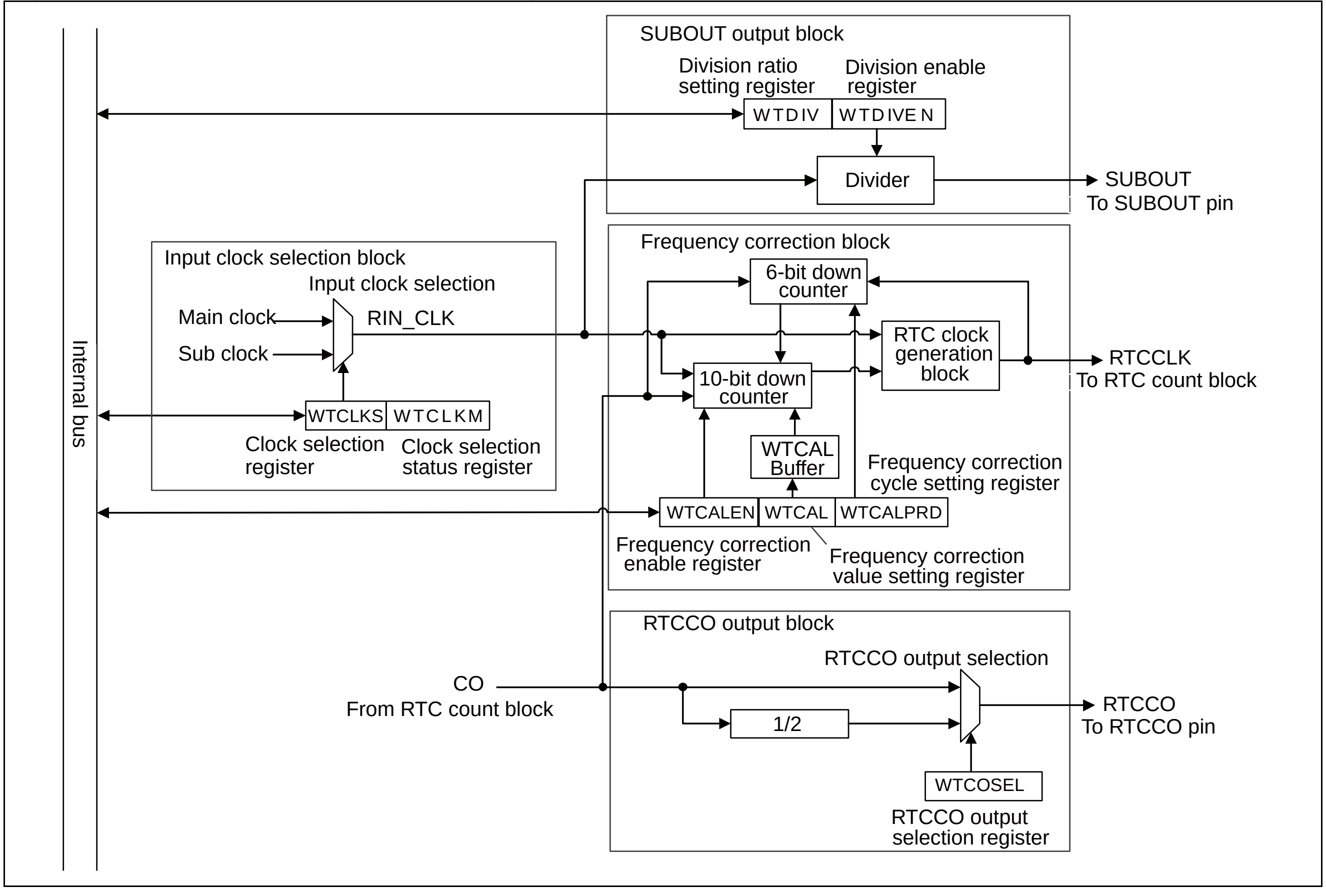
- Generation of a RTC clock (RTCCLK) used for the RTC count block.
- Selection of a main clock and sub clock as an input clock (RIN_CLK).
- Generation of a division clock output for the SUBOUT external pin.
- The pulse generation function for 0.5 second or 1 second outputted to a RTCCO external pin
- Frequency correction to compensate for frequency fluctuations resulting from the temperature dependence of the input clock (Frequency Correction Block).
  (The above frequency correction function is based on the assumption that a temperature sensor is externally connected.)

2. Configuration of RTC Clock Control Block

The following shows the block diagram.

■ RTC Clock Control Block Diagram

Figure 2-1 RTC Clock Control Block Diagram



● Input Clock Selection Block

Select the input clock (RIN_CLK) as the main clock and sub clock by setting the clock selection register (WTCALS).

● Frequency Correction Block

The frequency correction block masks RIN_CLK and outputs frequency-corrected RTCCLK.
The frequency correction block masks RIN_CLK according the number of the value set in the WTCAL buffer at the cycles set as WTCALPRD register.

● SUBOUT Output Block

The SUBOUT output block generates a division clock as a SUBOUT external pin output.
No SUBOUT external pin output is possible while in the deep standby RTC mode.

● RTCCO Output Block

The RTCCO output block generates a signal as a RTCCO external pin output.
The RTCCO external pin output select possible CO signal from a RTC count block or 2 divisions of CO signal.
In deep standby RTC mode, no output from RTCCO is possible.

3. Operation of RTC Clock Control Block

This section explains the operation of RTC clock control block.

■ Frequency Correction Block

A gap of the frequency of RIN_CLK is corrected.
The mask of RIN_CLK is carried out a fixed cycle, and RTCCLK which performed frequency correction is outputted.
A cycle is set as frequency correction cycle setting register (WTCALPRD).
The number of clocks which carry out a mask to frequency correction value setting register (WTCAL) is set.

Figure 3-1 Example of a frequency correction block of operation (WTBR=8190 and WTCALPRD=19)

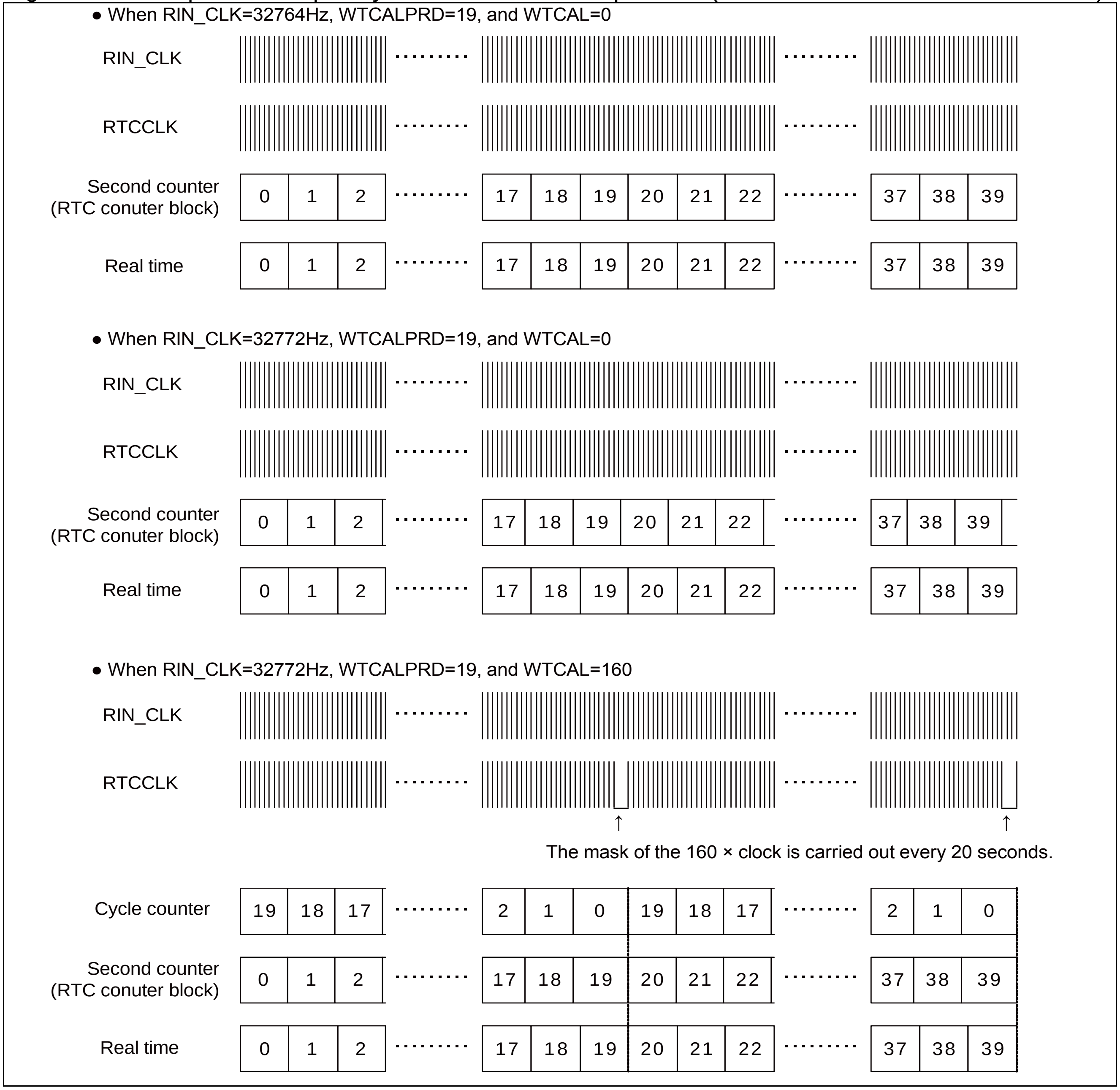
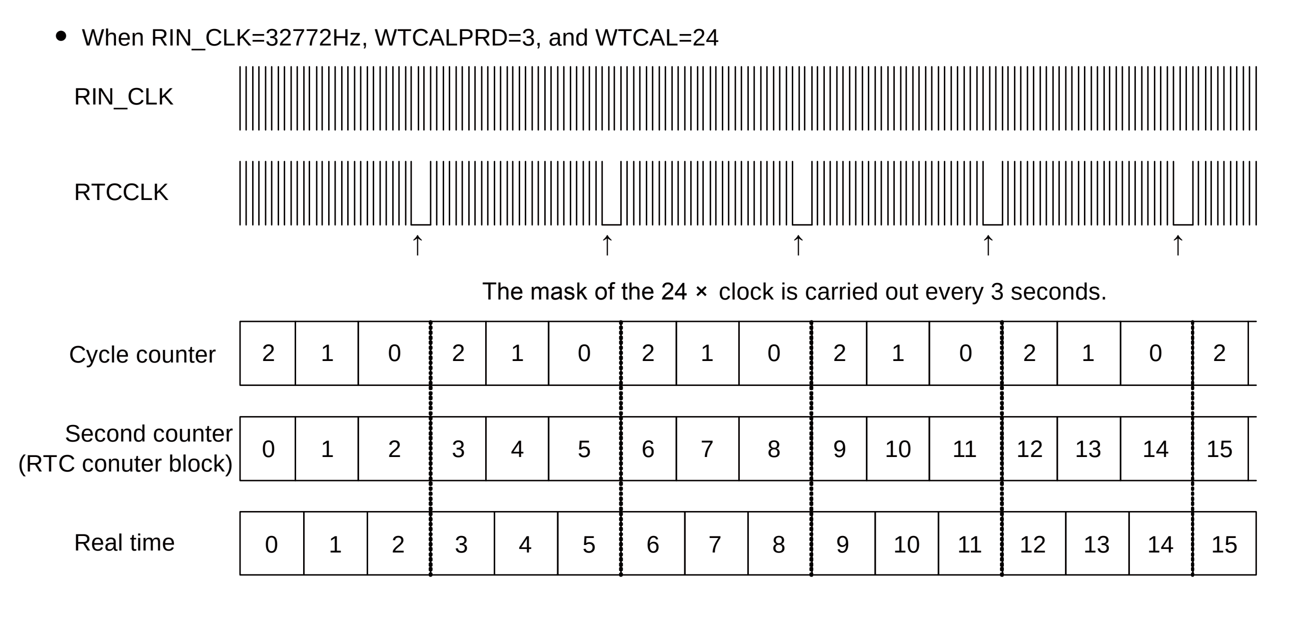

● When RIN_CLK=32764Hz, WTCALPRD=19, and WTCAL=0
RIN_CLK
RTCCLK
Second counter (RTC conuter block)
Real time
0 1 2 17 18 19 20 21 22 37 38 39
● When RIN_CLK=32772Hz, WTCALPRD=19, and WTCAL=0
RIN_CLK
RTCCLK
Second counter (RTC conuter block)
Real time
● When RIN_CLK=32772Hz, WTCALPRD=19, and WTCAL=160
RIN_CLK
RTCCLK
The mask of the 160 × clock is carried out every 20 seconds.
Cycle counter
19 18 17 2 1 0 19 18 17 2 1 0
Second counter (RTC conuter block)
Real time

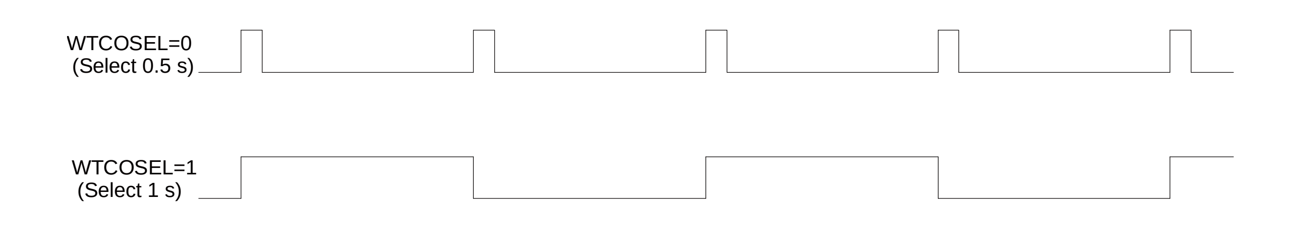
Figure 3-2 Example of a frequency correction block of operation (WTBR=8190 and WTCALPRD=3)



■ RTCCO external pin output clock selection block

By setup of RTCCO clock selection register (WTCOSEL), CO signal (0.5 second) from a RTC count block or the 2 divisions (1 second) of CO signal is selected, and it outputs to a RTCCO external pin.

Figure 3-3 Example of a frequency correction block of operation (WTBR=8190)



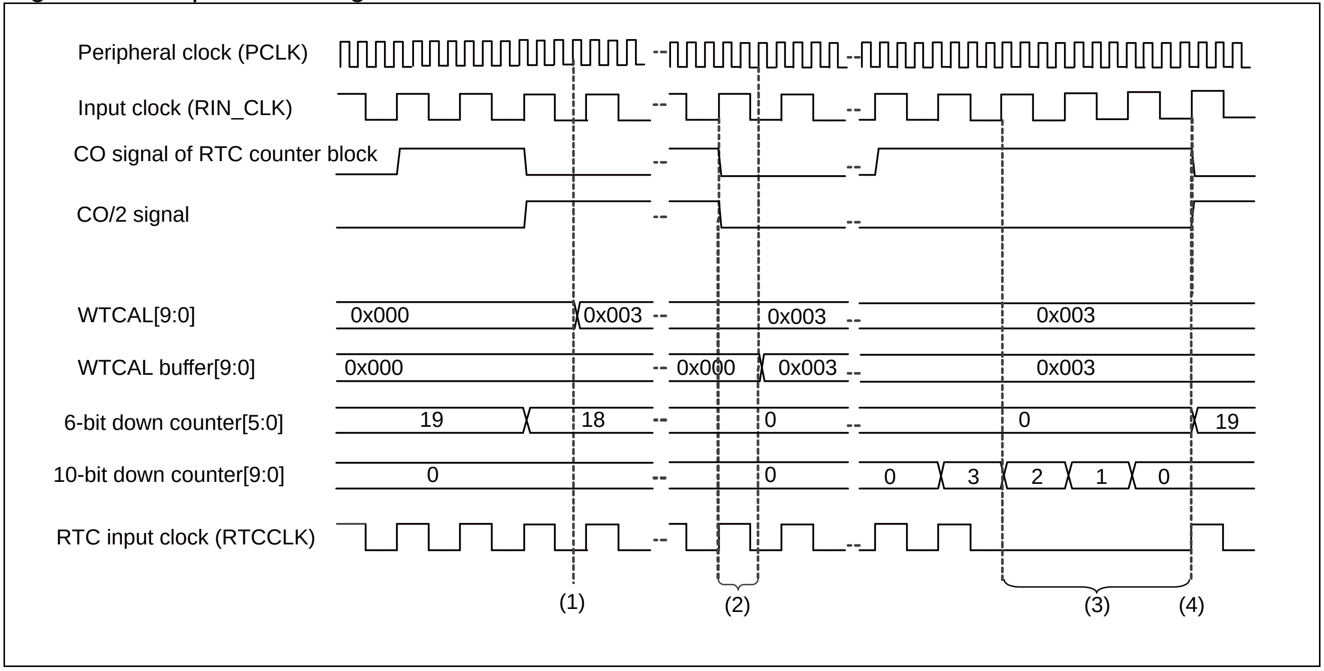
See the chapter of "RTC Count Block" for CO signal (0.5 second) from a RTC count block.

● Operation of the WTCAL Buffer

The frequency correction value setting register (WTCAL) sets the number of clock cycles masked by the frequency correction function.

The frequency correction block incorporates a WTCAL buffer because this block in operation makes the frequency correction value setting register (WTCAL) rewritable.

Figure 3-4 Operation Diagram of the WTCAL Buffer



(1) The WTCAL buffer writes the number of clocks to be masked in the frequency correction value setting register (WTCAL).

(2) 6-bit down counter = 0 it is set to zero and the value of WTCAL is transmitted after 3 clocks of APB2 bus clock (PCLK2) at a WTCAL buffer.

(3) The value of a WTCAL register is loaded to a 10-bit down counter just before the underflow of 6-bit down counter, and carry out the mask of the clock of the set-up value.

(4) 6-bit down counter loads the value of a WTCAL buffer at the time of underflow.

<Note>

The CO signal of a RTC count block is High at the mask of the clock set up by the WTCAL register is carried out.

● Frequency Correction Range

Table 3-1 and Table 3-2 show the example of the frequency correction range. It rectifies combining a setup of the WTBR register of a RTC count block, a WTCAL register, and a WTCALPRD register.

Table 3-1 Example of Frequency Correction Range (WTCALPRD=19) (Ideal value)

WTCAL	WTBR=8190		WTBR=8189	
	Correction rate [ppm]	Subclock Frequency [Hz]	Correction rate [ppm]	Subclock Frequency [Hz]
0	122.1	32764.00	244.1	32760.00
1	120.5	32764.05	242.6	32760.05
2	119.0	32764.10	241.1	32760.10
:	:	:	:	:
79	1.5	32767.95	123.6	32763.95
80	0.0	32768.00	122.1	32764.00
81	-1.5	32768.05	120.5	32764.05
:	:	:	:	:
159	-120.5	32771.95	1.5	32767.95
160	-122.1	32772.00	0.0	32768.00
161	-123.6	32772.05	-1.5	32768.05
:	:	:	:	:
318	-363.2	32779.90	-241.1	32775.90
319	-364.7	32779.95	-242.6	32775.95
320	-366.2	32780.00	-244.1	32776.00

Table 3-2 Example of Frequency Correction Range (WTCALPRD=59) (Ideal value)

WTCAL	WTBR=8190		WTBR=8189	
	Correction rate [ppm]	Subclock Frequency [Hz]	Correction rate [ppm]	Subclock Frequency [Hz]
0	122.1	32764.00	244.1	32760.00
1	121.6	32764.02	243.6	32760.02
2	121.1	32764.03	243.1	32760.03
:	:	:	:	:
239	0.5	32767.98	122.6	32763.98
240	0.0	32768.00	122.1	32764.00
241	-0.5	32768.02	121.6	32764.02
:	:	:	:	:
479	-121.6	32771.98	0.5	32767.98
480	-122.1	32772.00	0.0	32768.00
481	-122.6	32772.02	-0.5	32768.02
:	:	:	:	:
958	-365.2	32779.97	-243.1	32775.97
959	-365.7	32779.98	-243.6	32775.98
960	-366.2	32780.00	-244.1	32776.00

4. Setting Procedures for RTC Clock Control Block

This section explains the setting procedures for the RTC clock control block.

■ Procedures for the Frequency Correction Settings (with sub clock selected)

(1) Write "0" to the input clock selection bit (WTCLKS) to select the sub clock.

(2) Read the clock selection status bit (WTCLKM) and wait until the sub clock "10" is selected.

(3) Write the correction cycle to frequency correction cycle setting register (WTCALPRD) and the correction value to the frequency correction value setting register (WTCAL).
Obtain the set value for the WTCAL from the following formula:
$\text{WTCAL} = \{(\text{Frequency before correction} - (\text{WTBR}+1) \times 4) / (\text{WTCALPRD} + 1)\} \times 2^{20}$

(4) The frequency correction will be enabled with "1" written to the frequency correction enable bit (WTCALEN).

■ Setting Procedures for the SUBOUT Output Block

(1) Write "0" to the divider output enable bit (WTDIVEN).
The divider will stop operating and the SUBOUT external output will output a low level.

(2) Read the divider status bit (WTDIVRDY) and wait until the value becomes "0" (the divider is not in operation).

(3) Write the divider ratio to the divider ratio setting bit (WTDIV). For divider ratio settings, see "5.5 Divider Ratio Setting Register (WTDIV)".

(4) The divider is set to operation enabled by writing "1" to the divider output enable bit (WTDIVEN).

5. Registers of RTC Clock Control Block

This section shows the list of registers.

■ Registers of the RTC Clock Control Block

Table 5-1 List of Registers of the RTC Clock Control Block

Abbreviated register name	Register name	Reference
WTCLKS	Clock selection register	5.1
WTCLKM	Selection clock status register	5.2
WTCAL	Frequency correction value setting register	5.3
WTCALEN	Frequency correction enable register	5.4
WTDIV	Divider ratio setting register	5.5
WTDIVEN	Divider output enable register	5.6
WTCALPRD	Frequency correction cycle setting register	5.7
WTCOSEL	RTCCO output selection register	5.8

5.1. Clock Selection Register (WTCLKS)

This register is used for the input clock (RIN_CLK) selection.

bit	7	1	0
Field	Reserved		WTCLKS
Attribute	R		R/W
Initial value	0000000		0

[bit7:1] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit0] WTCLKS: Input clock selection bit

This bit is used for the following input clock (RIN_CLK) selection.

Value	Description
0	Selects the sub clock.
1	Selects the main clock.

5.2. Selection Clock Status Register (WTCLKM)

This register shows the status of the input clock (RIN_CLK) selection.

bit	7	2	1	0
Field	Reserved		WTCLKM	
Attribute	R		R	
Initial value	000000		00	

[bit7:2] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit1:0] WTCLKM: Clock selection status bits

This bit shows the status of the input clock (RIN_CLK) selection.

Value	Description
0x	The RIN_CLK is not in operation
10	The sub clock is selected.
11	The main clock is selected.

<Note>

A software reset, RTC reset, or APB2 bus reset does not initialize this register.

5.3. Frequency Correction Value Setting Register (WTCAL)

This register sets the frequency correction value for the RTC clock (RTCCLK) output to the RTC count block.

bit	15	10	9	8
Field	Reserved		WTCAL	
Attribute	R		R/W	
Initial value	000000		00	

bit	7	0
Field	WTCAL	
Attribute	R/W	
Initial value	00000000	

[bit15:10] Reserved: Reserved bit

Always "0" is read.
Set "0" when writing data.

[bit9:0] WTCAL: Frequency correction value setting bits

The number of cycles to be masked is set cycle as the WTCALPRD register.
Eight clocks will be masked from the input clock (RIN_CLK) for each 20 s and the RTCCLK output will be generated to the RTC count block if the WTCALPRD is set as 19 and the WTCAL is set to 8.
For the WTCAL set value, see "■ Procedures for the Frequency Correction Settings (with sub clock selected)" in " 4. Setting Procedures for RTC Clock Control Block".

<Note>

A software reset or APB2 bus reset does not initialize this register.

5.4. Frequency Correction Enable Register (WTCALEN)

This register enables frequency corrections to the RTC clock (RTCCLK) input into the RTC count block.

bit	7	1	0
Field	Reserved		WTCALEN
Attribute	R		R/W
Initial value	0000000		0

[bit7:1] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit0] WTCALEN: Frequency correction enable bit

The frequency correction enable bit enables frequency corrections.

Value	Description
0	Frequency corrections are disabled.
1	Frequency corrections are enabled.

<Note>

A software reset or APB2 bus reset does not initialize this register.

5.5. Divider Ratio Setting Register (WTDIV)

This register sets the divider ratio.

bit	7	4	3	0
Field	Reserved		WTDIV	
Attribute	R		R/W	
Initial value	0000		0000	

[bit7:4] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit3:0] WTDIV: Divider ratio setting bits

The divider ratio setting bit is used to set the divider ratio of the input clock (RIN_CLK) and the divider clock (SUBOUT) that the divider outputs.

Value	Description
0000	No division
0001	Divided by 2
0010	Divided by 4
0011	Divided by 8
0100	Divided by 16
0101	Divided by 32
0110	Divided by 64
0111	Divided by 128
1000	Divided by 256
1001	Divided by 512
1010	Divided by 1024
1011	Divided by 2048
1100	Divided by 4096
1101	Divided by 8192
1110	Divided by 16384
1111	Divided by 32768

<Note>

Write data to the WTDIV bit when the divider enable bit (WTDIVEN) and the divider output status bit (WTDIVRDY) of the Divider output enable register (WTDIVEN) are set to "0".
A software reset or APB2 bus reset does not initialize this register.

5.6. Divider Output Enable Register (WTDIVEN)

This register enables the divider output.

bit	7	2	1	0
Field	Reserved		WTDIVRDY	WTDIVEN
Attribute	R		R	R/W
Initial value	000000		0	0

[bit7:2] Reserved: Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit1] WTDIVRDY: Divider status bit

This bit shows the operating status of the divider.

Value	Description
0	The divider is not in operation. The SUBOUT external pin output is fixed to Low.
1	The divider is in operation.

[bit0] WTDIVEN: Divider enable bit

This bit is used to enable the operating status of the divider.

Value	Description
0	Stops the divider operation.
1	Enables the operation of the divider.

<Note>

A software reset or APB2 bus reset does not initialize this register.

5.7. Frequency Correction Cycle Setting Register (WTCALPRD)

This register sets cycle of frequency correction.

bit	7	6	5	0
Field	Reserved		WTCALPRD	
Attribute	R		R/W	
Initial value	00		010011	

[bit7:6] Reserved : Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit5:0] WTCALPRD: frequency correction value setting bits

Set a value which is made by subtracting one from the cycle(second) masking the clock for frequency compensation. For example, if "0" is set up and the cycle of 1 second and 19 will be set up, it will become cycles of 20 s.

<Note>

A software reset or APB2 bus reset does not initialize this register.

5.8. RTCCO Output Selection Register (WTCOSEL)

This register selects RTCCO output.

bit	7 ... 1	0
Field	Reserved	WTCOSEL
Attribute	R	R/W
Initial value	0000000	0

[bit7:1] Reserved : Reserved bits

Always "0" is read.
Set "0" when writing data.

[bit0] WTCOSEL: RTCCO output selection bit

This bit selects RTCCO output.

Value	Description
0	CO signal of a RTC count part is outputted.
1	The 2 divisions of CO signal is outputted.

<Note>

A software reset or APB2 bus reset does not initialize this register.

CHAPTER 5-1: Base Timer Configuration

This chapter explains the configuration of the base timer.

CODE :9xFBTTOP-E01.0

1. Configuration

For the configuration of base timer, see the following relevant chapters.

■ Referred Base Timer Chapter of each product

Table 1-1 Referred Base Timer I/O Select Function Chapter

Product TYPE	Referred Chapter
TYPE0 and TYPE1, TYPE3 to TYPE12	Chapter "Base Timer I/O Select Function(A)"
TYPE2	Chapter "Base Timer I/O Select Function(B)"

Table 1-2 Base Timer Correspondence Table

Product TYPE	Referred Chapter
TYPE0 to TYPE12	Chapter "Base Timer"

CHAPTER 5-2: Base Timer I/O Select Function (A)

This chapter explains about the base timer I/O select function (A).

CODE: 9BFBTSELA-E02.0_FW14-E00.4

1. Overview

The base timer I/O select function sets the I/O mode, and thereby determines the method to input and output signals (external clock, external start trigger, and waveform) to/from the base timer.

By switching timer function, each channel of the base timer can be also used as one of the following timers:

- 16-bit PWM timer
- 16-bit PPG timer
- 16-/32-bit reload timer
- 16-/32-bit PWC timer

■ Overview

One of the following 9 types of I/O modes can be selected for each 2 channels.

Software-based simultaneous startup function is provided for multiple channels, enabling up to 16 channels to be started up via software.

- I/O mode 0: Standard 16-bit timer mode
  This mode operates each channel of the base timer individually.

- I/O mode 1: Timer full mode
  This mode assigns each even channel signal of the base timer with an external pin individually to operate the channel.

- I/O mode 2: Shared external trigger mode
  This mode can input an external startup trigger to two channels of the base timer simultaneously. Using this mode, the base timer of two channels can be started up simultaneously.

- I/O mode 3: Shared channel signal trigger mode
  This mode uses an external signal from another channel as an external startup trigger. This mode cannot be selected for channel 0 or 1.

- I/O mode 4: Timer start/stop mode
  This mode controls the start/stop of the odd channel using the even channel. The odd channel starts on the rising edge of output signal from the even channel, and stops on the falling edge.

- I/O mode 5: Software-based simultaneous startup mode
  This mode starts up multiple channels simultaneously via software.

- I/O mode 6: Software-based startup and timer start/stop mode
  This mode controls the start/stop of the odd channel using the even channel. An even channel is started up via software. The odd channel starts on the rising edge of output signal from the even channel, and stops on the falling edge.

- I/O mode 7: Timer start mode
  This mode controls the start of the odd channel using the even channel. The odd channel starts on the rising edge of output signal from the even channel.

- I/O mode 8: Shared channel signal trigger and timer start/stop mode
  This mode uses an external signal from another channel as an external startup trigger. This mode cannot be selected for channel 0 or 1.

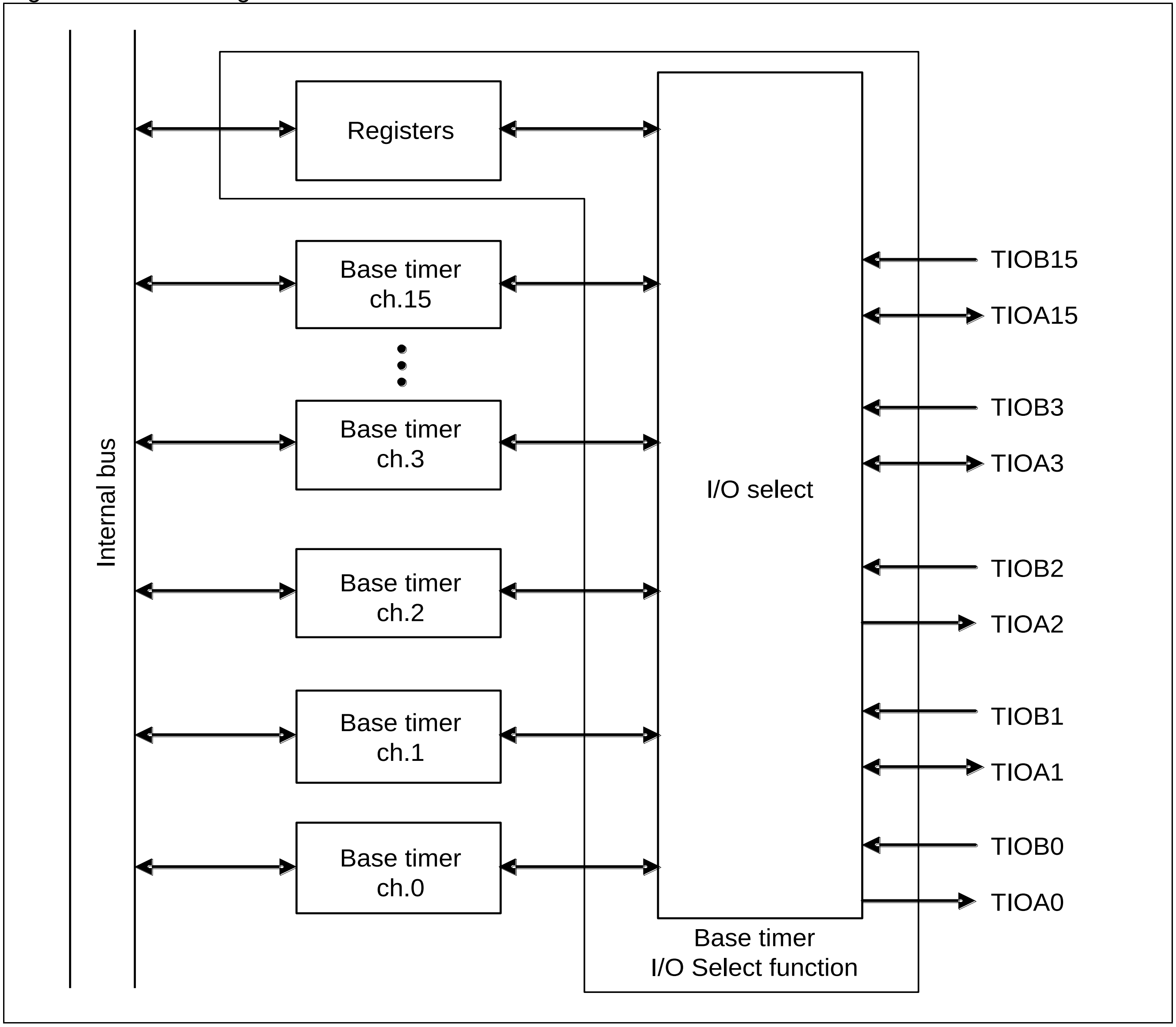
2. Configuration

The base timer I/O select function consists of the following blocks.

■ Block diagram

Figure 2-1 shows the block diagram of the base timer I/O select function.

Figure 2-1 Block diagram of base timer I/O select function



- I/O select
  A circuit that selects the I/O mode of the base timer for each channel.
- Base timer (Channels 0 to 15)
  Base timer channels 0 to 15 (up to 16 channels).
- Registers
  Registers of base timer I/O select function.

3. I/O Mode

This section explains pins used by the base timer I/O select function to set the I/O mode, and also explains each I/O mode.

3.1. Pins

This section explains pins used by the base timer I/O select function to set the I/O mode.

Each channel of the base timer has 2 types of external pins and 5 types of internal signals. Also the base timer I/O select function has 2 types of internal signals. By connecting an internal signal with an external pin, the signal corresponding to the connected (external clock (ECK signal)/external startup trigger (TGIN signal)/waveform (TIN signal)) is input or output to/from the base timer. The external pin and internal signal can be connected by setting the I/O mode of the base timer. The pin used and the signal input or output differ depending on the I/O mode.

■ External pins

- TIOA pin
  This pin is used to output the base timer waveform (TOUT signal), or input an external startup trigger (TGIN signal).

- TIOB pin
  This pin is used to input external startup trigger (TGIN signal)/external clock (ECK signal)/another channel waveform (TIN signal).

■ Internal signals

A signal is input or output to/from the base timer by being connected with an above external pin, or by inputting an output signal from another channel.

- TOUT signal
  This signal is the output waveform of the base timer. (Not used by the 16-/32-bit PWC timer.)

- ECK signal
  This signal is an external clock of the base timer. (Not used by the 16-/32-bit PWC timer.)
  It is input when the external clock is selected as a counting clock.

- TGIN signal
  This signal is the external startup trigger of the base timer. (Not used by the 16-/32-bit PWC timer.)
  When the valid edge of external startup trigger is selected, the base timer detects the edge of this signal to start up.

- TIN signal
  This signal is the input waveform of the base timer. This signal is the waveform to be measured. (Used only by the 16/32-bit PWC timer.)

- DTRG signal
  This signal is the trigger input to the base timer. The base timer stops operating on the falling edge of this signal.

- COUT signal
  This signal is the trigger output of the base timer I/O select function. This signal is output signal to another channel of the base timer.

- CIN signal
  This signal is the trigger input to the base timer I/O select function. This signal is input signal from another channel of the base timer.

■ Connecting the external pin to the internal signal

The external pin and internal signal can be connected by setting the I/O mode of the base timer.

Table 3-1 shows the correspondence between I/O modes and pin connections.

Table 3-1 Correspondence between I/O modes and pin connections

I/O mode	TIOAn (Even channel)		TIOBn (Even channel)		TIOAn+1 (Odd channel)		TIOBn+1 (Odd channel)	
	Connected to	I/O	Connected to	I/O	Connected to	I/O	Connected to	I/O
0	Ch.n TOUT	Outputt	Ch.n ECK/TGIN/ TIN	Input	Ch.n+1 TOUT	Output	Ch.n+1 ECK/TGIN/ TIN	Input
1			Ch.n ECK	Input	Ch.n TGIN	Input	Ch.n TIN	Input
2			Ch.n/Ch.n+1 ECK/TGIN/ TIN *1	Input	Ch.n+1 TOUT	Output	Not used	
3			Not used					
4			Ch.n ECK/TGIN/ TIN	Input				
5			Not used					
6								
7			Ch.n ECK/TGIN/ TIN	Input				
8			Not used					

n: Even (n=0, 2, 4, 6, 8, 10, 12, 14) However, n depends on the number of channels mounted.
Ch.n: Even channel
Ch.n+1: Odd channel
*1: Synchronized by the peripheral clock (PCLK)

3.2. I/O mode

I/O mode selected by the I/O Select Register (BTSEL) determines the functions of external pins and the start/stop timing of the base timer.

■ I/O mode 0 (Standard 16-bit timer mode)

This mode uses each channel of the base timer individually.

Table 3-2 shows the external pins used when this mode is selected.

Table 3-2 External pins used when I/O mode 0 is selected.

	Even channel	Odd channel
Number of input pins	1	1
Number of output pins	1	1

Table 3-3 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-3 External pin connections and input/output signals when I/O mode 0 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOA	Output	TOUT	Outputs the base timer waveform
TIOB	Input	ECK/TGIN/TIN*	Uses the input signal as one of the following signals: · External clock (ECK signal) · External startup trigger (TGIN signal) · Waveform to be measured (TIN signal)

*: The usage of input signals (ECK/TGIN/TIN) differs depending on the Timer Control Register (TMCR) setting of the base timer.

Figure 3-1 provides the block diagram of I/O mode 0 (Standard 16-bit timer mode).

Figure 3-1 I/O mode 0 (Standard 16-bit timer mode) block diagram

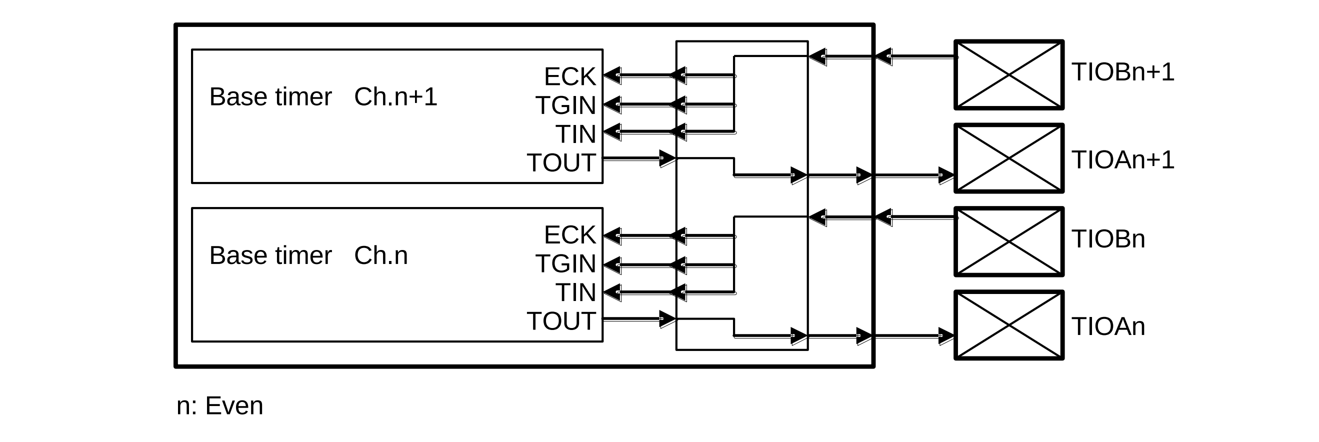


Table 3-4 shows signal connections in I/O mode 0.

Table 3-4 I/O mode 0 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	Output from the TIOAn pin
Input signal from the TIOBn pin	Input to Ch.n as ECK/TGIN/TIN signals
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin
Input signal from the TIOBn+1 pin	Input to Ch.n+1 as ECK/TGIN/TIN signals

n: Even

■ I/O mode 1 (timer full mode)

This mode assigns every even channel signal with an external pin individually.
Table 3-5 shows the external pins used when this mode is selected.

Table 3-5 External pins used when I/O mode 1 is selected.

	Even channel
Number of input pins	3
Number of output pins	1

Table 3-6 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-6 External pin connections and input/output signals when I/O mode 1 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOBn	Input	Even channel ECK	Inputs an external clock (ECK signal) to the even channel.
TIOAn+1	Input	Even channel TGIN	Inputs an external startup trigger (TGIN signal) to the even channel.
TIOBn+1	Input	Even channel TIN	Inputs the waveform to be measured (TIN signal) to the even channel.

n: Even

Figure 3-2 shows the block diagram of I/O mode 1 (timer full mode).

Figure 3-2 I/O mode 1 (timer full mode) block diagram

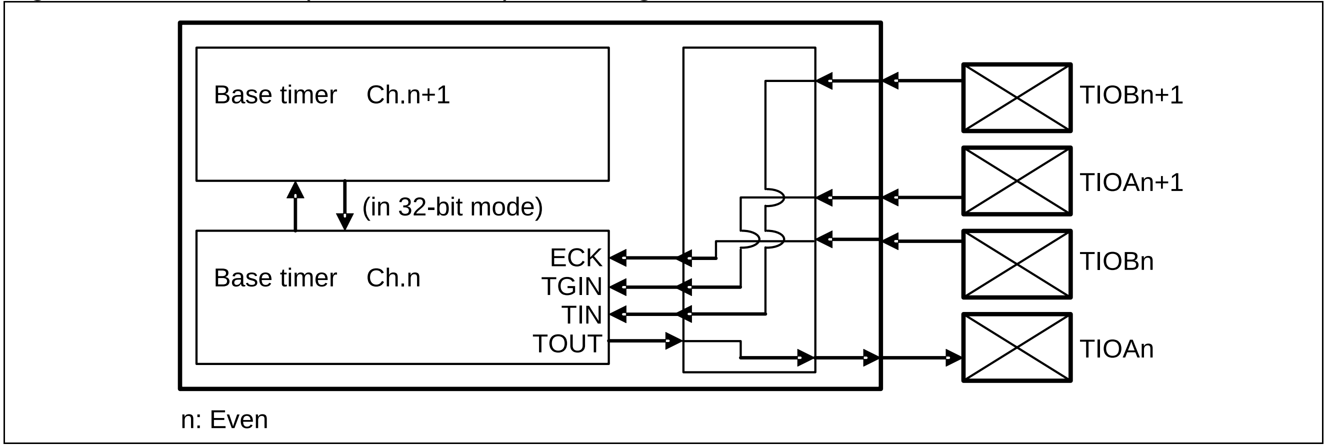


Table 3-7 shows signal connections in I/O mode 1.

Table 3-7 I/O mode 1 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	Output from the TIOAn pin
Input signal from the TIOBn pin	Input to Ch.n as a ECK signal
TIOAn+1 pin	Input to Ch.n as a TGIN signal
TIOBn+1 pin	Input to Ch.n as an TIN signal

n: Even

<Note>

When this mode is selected, the TIOA pins (TIOA1, TIOA3, etc.) corresponding to the odd channel must be set to port input mode with the Port Function Register (PFR) of GPIO.

■ I/O mode 2 (Shared external trigger mode)

This mode shares the input signals (ECK/TGIN/TIN) of the base timer between two channels.
Table 3-8 shows the external pins used when this mode is selected.

Table 3-8 External pins used when I/O mode 2 is selected.

	Even channel	Odd channel
Number of input pins	1 (shared by two channels)	
Number of output pins	1	1

Table 3-9 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-9 External pin connections and input/output signals when I/O mode 2 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn	Input	ECK/TGIN/TIN of even and odd channels *	Input to both the even and odd channels (synchronized by the peripheral clock (PCLK)) and used as one of the following signals: · External clock (ECK signal) · External startup trigger (TGIN signal) · Waveform to be measured (TIN signal)
TIOBn+1	-	-	Not used

n: Even
*: The usage of input signals (ECK/TGIN/TIN) differs depending on the Timer Control Register (TMCR) setting of the base timer.

Figure 3-3 shows the block diagram of I/O mode 2 (Shared external trigger mode).

Figure 3-3 I/O mode 2 (Shared external trigger mode) block diagram

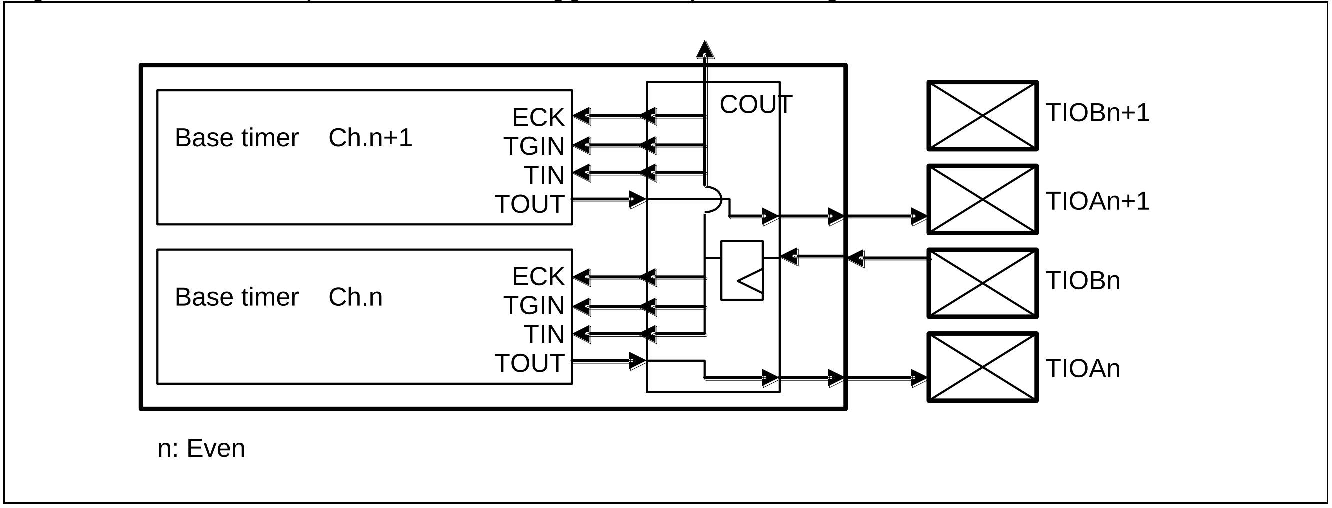


Table 3-10 shows signal connections in I/O mode 2.

Table 3-10 I/O mode 2 signal connections

Connected from (Signal)	Connected to	Remarks
Ch.n TOUT signal	Output from the TIOAn pin	
Input signal from the TIOBn pin	· Input to Ch.n and Ch.n+1 as ECK/TGIN/TIN signals · Output to another channel as a COUT signal	Synchronized by the peripheral clock (PCLK)
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin	

n: Even

<Note>

If the upper two channels of the channels set to this mode (n+2, n+3) are set to I/O mode 3 (Shared channel signal trigger mode), the input signals (ECK/TGIN/TIN) can be input to the 4 channels simultaneously.

(Example: If channels 0 and 1 are set to this mode, and channels 2 and 3 are set to I/O mode 3, input signals (ECK/TGIN/TIN) can be input to four channels of 0 to 3 simultaneously.)

■ I/O mode 3 (Shared channel signal trigger mode)

This mode inputs the COUT signal from channels of the lower two channels as a CIN signal, and uses it as ECK/TGIN/TIN signals.

Table 3-11 shows the external pins used when this mode is selected.

Table 3-11 External pins used when I/O mode 3 is selected.

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-12 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-12 External pin connections and input/output signals when I/O mode 3 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-4 shows the block diagram of I/O mode 3 (Shared channel signal trigger mode).

Figure 3-4 I/O mode 3 (Shared channel signal trigger mode) block diagram

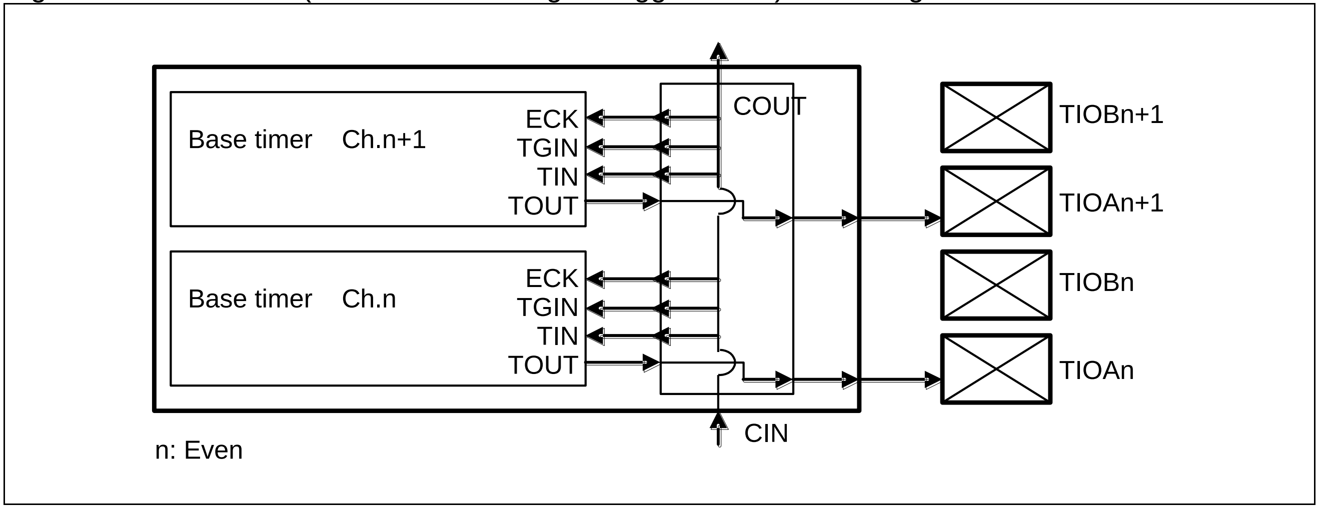


Table 3-13 shows signal connections in I/O mode 3.

Table 3-13 I/O mode 3 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	Output from the TIOAn pin
CIN signal *	· Input to Ch.n and Ch.n+1 as ECK/TGIN/TIN signals · Output to another channel as a COUT signal
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

*: The COUT signal from another channel is input as a CIN signal.

The following shows Ch.n-2/n-1 signals that can be input to ECK/TGIN/TIN of Ch.n/n+1.

- Signal that the peripheral clock generates by synchronizing TIOBn-2 input in I/O mode 2.
- Trigger signal input from Ch.n-4/n-3 in I/O mode 3.
- TIOAn-2 output in I/O mode 4.
- TIOAn-2 output in I/O mode 6.
- TIOAn-2 output in I/O mode 7.
- Trigger signal input from Ch.n-4/n-3 in I/O mode 8.

<Notes>

- Select the rising edge as a trigger input edge using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
- The channels set to this mode use the COUT signal from lower two channels (n-2 and n-1) as a CIN signal. (Example: If channels 2 and 3 are set to this mode, they use the COUT signal from channels 0 and 1.) Therefore, channels 0 and 1 cannot be set to this mode.

■ I/O mode 4 (Timer start/stop mode)

This mode can control the start/stop of the odd channel using the even channel.
The odd channel starts on the rising edge of output waveform (TOUT signal) of the even channel, and stops on the falling edge.

Table 3-14 shows the external pins used when this mode is selected.

Table 3-14 External pins used when I/O mode 4 is selected.

	Even channel	Odd channel
Number of input pins	1	Not used
Number of output pins	1	1

Table 3-15 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-15 External pin connections and input/output signals when I/O mode 4 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn	Input	ECK/TGIN/TIN of even channel *	Input to the even channel and used as one of the following signals: · External clock (ECK signal) · External startup trigger (TGIN signal) · Waveform to be measured (TIN signal)
TIOBn+1	-	-	Not used

n: Even
*: The usage of input signals (ECK/TGIN/TIN) differs depending on the Timer Control Register (TMCR) setting of the base timer.

Figure 3-5 shows the block diagram of I/O mode 4 (Timer start/stop mode).

Figure 3-5 I/O mode 4 (Timer start/stop mode) block diagram

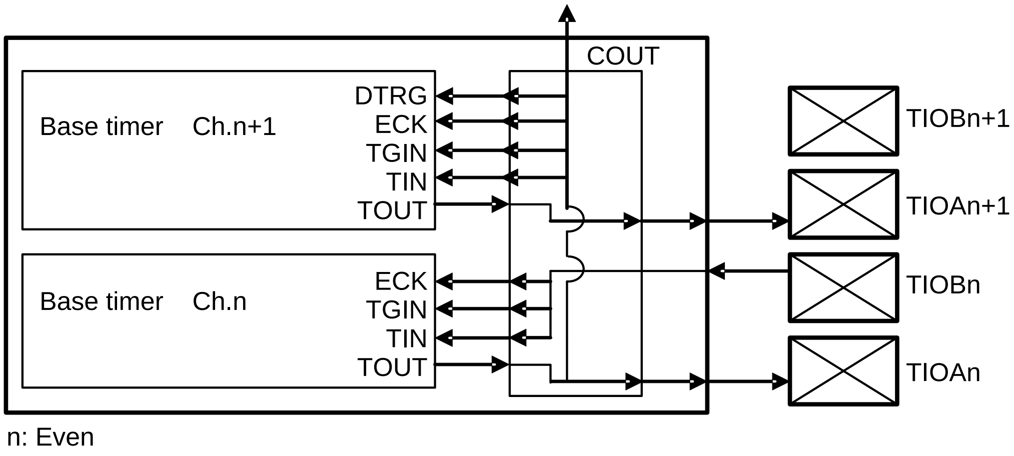


Table 3-16 shows signal connections in I/O mode 4.

Table 3-16 I/O mode 4 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	· Output from the TIOAn pin · Input to Ch.n+1 as ECK/TGIN/TIN and DTRG signals · Output to another channel as a COUT signal
Input signal from the TIOBn pin	Input to Ch.n as ECK/TGIN/TIN signals
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

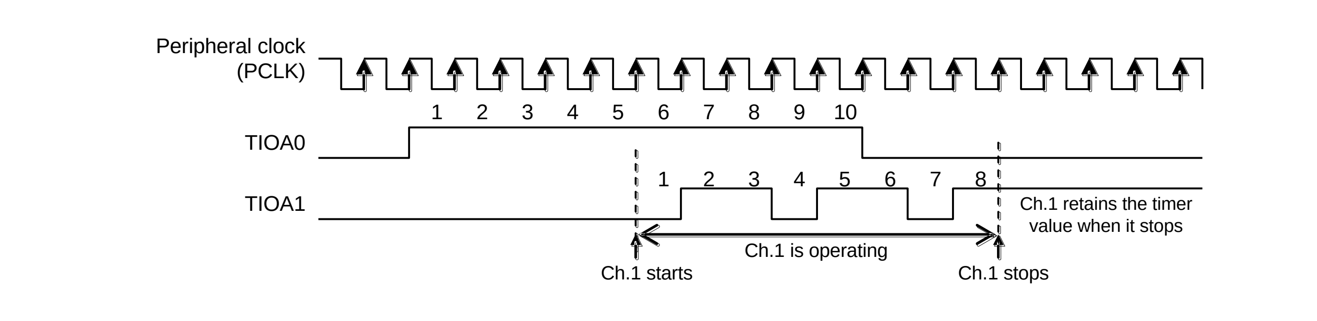
<Notes>

- Select the rising edge as a trigger input edge of the odd channel using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
- The odd channel stops operating when a falling edge is detected in the DTRG signal.

Figure 3-6 shows example operation when I/O mode 4 (Timer start/stop mode) is selected, and when channels 0 and 1 are used as PWM timer.

Base timer Ch.0	Set value	Base timer Ch.1	Set value
Cycle Setup Register (PCSR)	0x0010	Cycle Setup Register (PCSR)	0x0002
Duty Setup Register (PDUT)	0x0009	Duty Setup Register (PDUT)	0x0001
Timer Control Register (TMCR)	0x0013	Timer Control Register (TMCR)	0x0112

Figure 3-6 I/O mode 4 (Timer start/stop mode) operation example



■ I/O mode 5 (Software-based simultaneous startup mode)

This mode starts up multiple channels simultaneously using the Software-based Simultaneous Startup Register (BTSSSR).

All the channels corresponding to the Software-based Simultaneous Startup Register (BTSSSR) bits that have been set to "1" start up simultaneously.

Table 3-17 shows the external pins used when this mode is selected.

Table 3-17 External pins used when I/O mode 5 is selected.

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-18 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-18 External pin connections and input/output signals when I/O mode 5 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-7 shows the block diagram of I/O mode 5 (Software-based simultaneous startup mode).

Figure 3-7 I/O mode 5 (Software-based simultaneous startup mode) block diagram

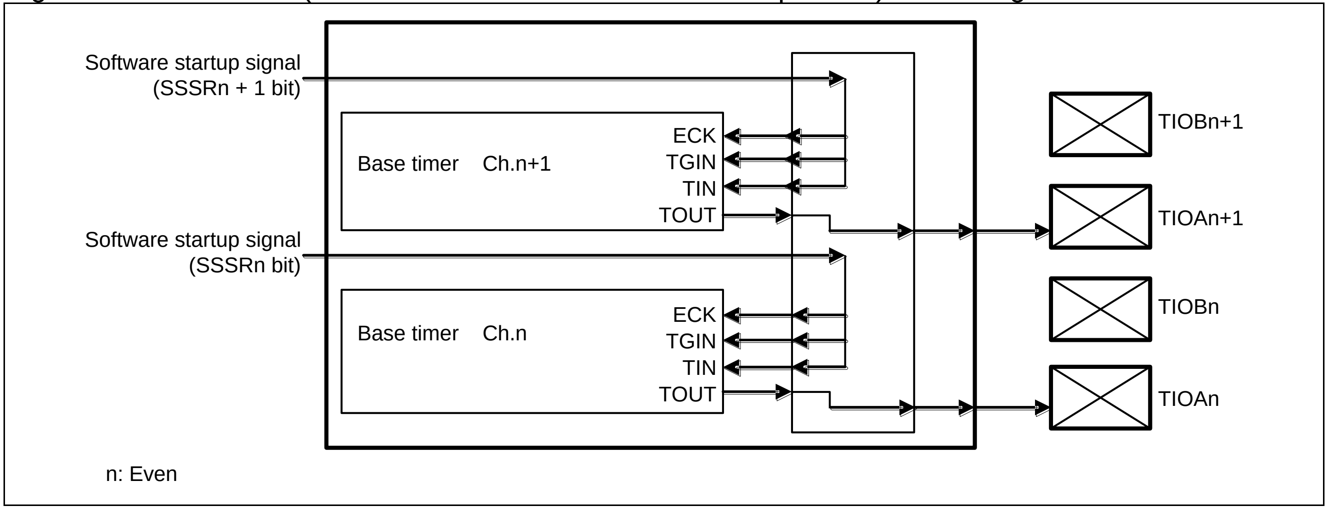


Table 3-19 shows signal connections in I/O mode 5.

Table 3-19 I/O mode 5 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	Output from the TIOAn pin
Software startup signal (Write "1" to the SSSRn bit in the BTSSSR)	Input to Ch.n as ECK/TGIN/TIN signals
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin
Software startup signal (Write "1" to the SSSRn+1 bit in the BTSSSR)	Input to Ch.n+1 as ECK/TGIN/TIN signals

n: Even

BTSSSR: Software-based Simultaneous Startup Register

When "1" is written to a Software-based Simultaneous Startup Register (BTSSSR), a rising edge is input (ECK/TGIN/TIN signals) to the channel corresponding to the bit.

<Note>

Select the rising edge as a trigger input edge using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)

■ I/O mode 6 (Software-based startup and timer start/stop mode)

This mode can control the start/stop of the odd channel using the even channel.

The even channel can be started by writing "1" to the Software-based Simultaneous Startup Register (BTSSSR).

The odd channel starts when the rising edge is detected in output waveform (TOUT signal) of the even channel, and stops when the falling edge is detected.

Table 3-20 shows the external pins used when this mode is selected.

Table 3-20 External pins used when I/O mode 6 is selected.

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-21 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-21 External pin connections and input/output signals when I/O mode 6 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-8 shows the block diagram of I/O mode 6 (Software-based startup and timer start/stop mode).

Figure 3-8 I/O mode 6 (Software-based startup and timer start/stop mode) block diagram

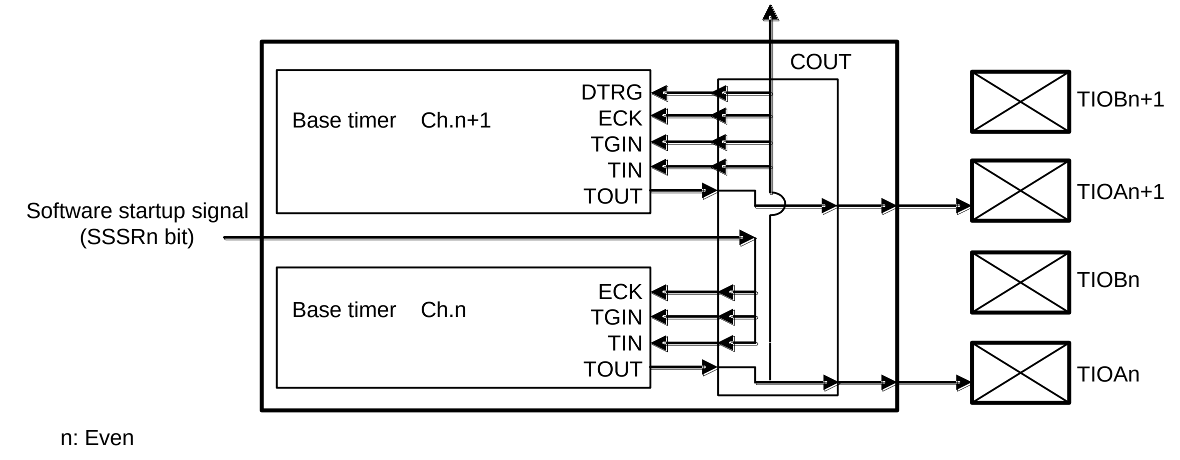


Table 3-22 shows signal connections in I/O mode 6.

Table 3-22 I/O mode 6 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	· Output from the TIOAn pin · Input to Ch.n+1 as ECK/TGIN/TIN/DTRG signals · Output to another channel as a COUT signal
Software startup signal (Write "1" to the SSSRn bit in the BTSSSR)	Input to Ch.n as ECK/TGIN/TIN signals
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

BTSSSR: Software-based Simultaneous Startup Register

When "1" is written to the Software-based Simultaneous Startup Register (BTSSSR) bit corresponding to the even channel you want to start up, a rising edge is input (ECK/TGIN/TIN signals) to the channel.

The start/stop timing of Ch.n is the same as that for I/O mode 4.

<Notes>

- Select the rising edge as a trigger input edge using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
- The odd channel stops operating when a falling edge is detected in the DTRG signal.

■ I/O mode 7 (Timer start mode)

This mode uses the output waveform (TOUT signal) from the even channel as input signals (ECK/TGIN/TIN signals) of the odd channel.

Table 3-23 shows the external pins used when this mode is selected.

Table 3-23 External pins used when I/O mode 7 is selected.

	Even channel	Odd channel
Number of input pins	1	Not used
Number of output pins	1	1

Table 3-24 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-24 External pin connections and input/output signals when I/O mode 7 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn	Input	Even channel ECK/ TGIN/TIN *	Input to the even channel and used as one of the following signals: · External clock (ECK signal) · External startup trigger (TGIN signal) · Waveform to be measured (TIN signal)
TIOBn+1	-	-	Not used

n: Even

*: The usage of input waveforms (ECK/TGIN/TIN signals) differs depending on the Timer Control Register (TMCR) setting.

Figure 3-9 shows the block diagram of I/O mode 7 (Timer start mode).

Figure 3-9 I/O mode 7 (Timer start mode) block diagram

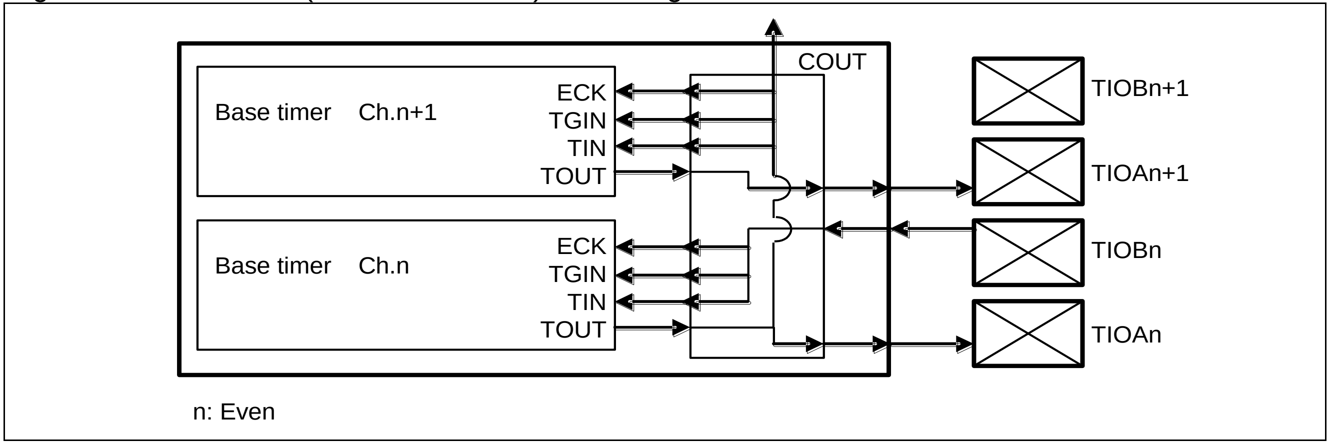


Table 3-25 shows signal connections in I/O mode 7.

Table 3-25 I/O mode 7 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	· Output from the TIOAn pin · Input to Ch.n+1 as ECK/TGIN/TIN signals · Output to another channel as a COUT signal
Input signal from the TIOBn pin	Input to Ch.n as ECK/TGIN/TIN signals
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

The start timing of Ch.n is the same as that for I/O mode 4.

■ I/O mode 8 (Shared channel signal trigger and timer start/stop mode)

This mode inputs the COUT signal from channels of the lower two channels as a CIN signal, and uses it as an external startup trigger (TGIN signal).

Table 3-26 shows the external pins used when this mode is selected.

Table 3-26 External pins used when I/O mode 8 is selected.

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-27 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-27 External pin connections and input/output signals when I/O mode 8 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-10 shows the block diagram of I/O mode 8 (Shared channel signal trigger and timer start/stop mode).

Figure 3-10 I/O mode 8 (Shared channel signal trigger and timer start/stop mode) block diagram

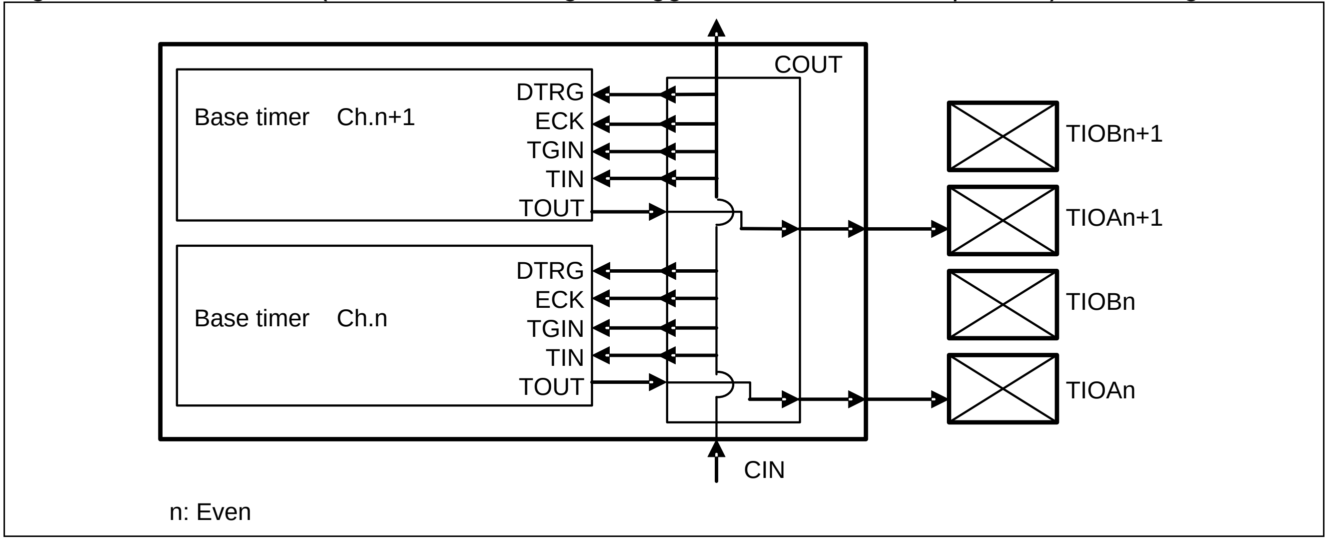


Table 3-28 shows signal connections in I/O mode 8.

Table 3-28 I/O mode 8 signal connections

Connected from (Signal)	Connected to
Ch.n TOUT signal	Output from the TIOAn pin
Ch.n+1 TOUT signal	Output from the TIOAn+1 pin
CIN signal *	· Input to Ch.n and Ch.n+1 as ECK/TGIN/TIN and DTRG signals · Output to another channel as a COUT signal

n: Even

*: The COUT signal from another channel is input as a CIN signal.

The following shows Ch.n-2/n-1 signals that can be input to ECK/TGIN/TIN of Ch.n/n+1.

- Signal that the peripheral clock generates by synchronizing TIOBn-2 input in I/O mode 2.
- Trigger signal input from Ch.n-4/n-3 in I/O mode 3.
- TIOAn-2 output in I/O mode 4.
- TIOAn-2 output in I/O mode 6.
- TIOAn-2 output in I/O mode 7.
- Trigger signal input from Ch.n-4/n-3 in I/O mode 8.

<Notes>

- The channels set to this mode use the COUT signal from lower 2 channels (n-2 and n-1) as a CIN signal. (Example: If channels 2 and 3 are set to this mode, they use the COUT signal from channels 0 and 1.) Therefore, channels 0 and 1 cannot be set to this mode.
- Select the rising edge as a trigger input edge, for the channel set in this mode, using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
  However, do not enable this setting if the timer function is set to the 16/32-bit PWC timer using FMD2 to FMD0 bits in the Timer Control Register (TMCR) of the base timer (FMD2 to FMD0 are set to 0b100).
- Base timer stops operating when a falling edge is detected in the DTRG signal.

4. Registers

This section provides the register list of the base timer I/O select function.

■ Base Timer I/O Select Function Registers

Table 4-1 Register list of Base timer I/O select function

Abbreviation	Register name	Reference
BTSEL0123	I/O Select Register	4.1
BTSEL4567	I/O Select Register	4.2
BTSEL89AB	I/O Select Register	4.3
BTSELCDEF	I/O Select Register	4.4
BTSSSR	Software-based Simultaneous Startup Register	4.5

4.1. I/O Select Register (BTSEL0123)

This register selects the I/O mode for channels 0 to 3 of the base timer.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SEL23_3	SEL23_2	SEL23_1	SEL23_0	SEL01_3	SEL01_2	SEL01_1	SEL01_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SEL23_3 to SEL23_0: I/O select bits for Ch.2/Ch.3

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

[bit11:8] SEL01_3 to SEL01_0: I/O select bits for Ch.0/Ch.1

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Notes>

- Channels 0 and 1 are the lowest channels of the base timer, and cannot use the modes that use signal from lower channels. Therefore, the following modes cannot be selected for the channels:
  - I/O mode 3 (Shared channel signal trigger mode)
  - I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
- Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

4.2. I/O Select Register (BTSEL4567)

This register selects the I/O mode for channels 4 to 7 of the base timer.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SEL67_3	SEL67_2	SEL67_1	SEL67_0	SEL45_3	SEL45_2	SEL45_1	SEL45_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SEL67_3 to SEL67_0: I/O select bits for Ch.6/Ch.7

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

[bit11:8] SEL45_3 to SEL45_0: I/O select bits for Ch.4/Ch.5

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Note>

Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

4.3. I/O Select Register (BTSEL89AB)

This register selects the I/O mode for channels 8 to 11 of the base timer.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SELAB_3	SELAB_2	SELAB_1	SELAB_0	SEL89_3	SEL89_2	SEL89_1	SEL89_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SELAB_3 to SELAB_0: I/O select bits for Ch.10/Ch.11

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

[bit11:8] SEL89_3 to SEL89_0: I/O select bits for Ch.8/Ch.9

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Note>

Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

4.4. I/O Select Register (BTSELCDEF)

This register selects the I/O mode for channels 12 to 15 of the base timer.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SELEF_3	SELEF_2	SELEF_1	SELEF_0	SELCD_3	SELCD_2	SELCD_1	SELCD_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SELEF_3 to SELEF_0: I/O select bits for Ch.14/Ch.15

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

[bit11:8] SELCD_3 to SELCD_0: I/O select bits for Ch.12/Ch.13

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Note>

Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

4.5. Software-based Simultaneous Startup Register (BTSSSR)

This register starts up the base timer using software simultaneously.

Up to 16 channels can be started simultaneously if the bits corresponding to the channel are set to "1".

■ Register configuration

bit	15 … 0
Field	SSSR15 to SSSR0
Attribute	W
Initial value	0xXXXX

■ Register functions

[bit15:0] SSSR15 to SSSR0: Software-based simultaneous startup bits

Value	Software-based simultaneous startup bits
0	Writing "0" to these bits is invalid
1	Starts Ch.x of the base timer

x: 15 to 0

<Notes>

- Do not write to this register unless set to either of the following modes:
  - I/O mode 5 (Software-based simultaneous startup mode)
  - I/O mode 6 (Software-based startup and timer start/stop mode)(Even channels only)
- For the channel started up by using this register, select the rising edge as a trigger input edge using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)

CHAPTER 5-3: Base Timer I/O Select Function (B)

This chapter explains about the base timer I/O select function (B).

CODE: 9BFBTSEL_B-E01.0_FW14-E00.3

1. Overview

The base timer I/O select function sets the I/O mode, and thereby determines the method to input and output signals (external clock, external start trigger, and waveform) to/from the base timer.

By switching timer function, each channel of the base timer can be also used as one of the following timers:

- 16-bit PWM timer
- 16-bit PPG timer
- 16-/32-bit reload timer
- 16-/32-bit PWC timer

■ Overview

One of the following 9 types of I/O modes can be selected for each 2 channels.
Software-based simultaneous startup function is provided for multiple channels, enabling up to 16 channels to be started up via software.

- I/O mode 0: Standard 16-bit timer mode
  This mode operates each channel of the base timer individually.

- I/O mode 1: Timer full mode
  This mode assigns each even channel signal of the base timer with an external pin individually to operate the channel.
  In this mode, TYPE2 product cannot use TIOA9 input as TGIN input of ch.8. For details refer to “I/O mode 1(timer full mode) in “3.2 I/O Mode”.

- I/O mode 2: Shared external trigger mode
  This mode can input an external startup trigger to two channels of the base timer simultaneously. Using this mode, the base timer of two channels can be started up simultaneously.

- I/O mode 3: Shared channel signal trigger mode
  This mode uses an external signal from another channel as an external startup trigger. This mode cannot be selected for channel 0 or 1.

- I/O mode 4: Timer start/stop mode
  This mode controls the start/stop of the odd channel using the even channel. The odd channel starts on the rising edge of output signal from the even channel, and stops on the falling edge.

- I/O mode 5: Software-based simultaneous startup mode
  This mode starts up multiple channels simultaneously via software.

- I/O mode 6: Software-based startup and timer start/stop mode
  This mode controls the start/stop of the odd channel using the even channel. An even channel is started up via software. The odd channel starts on the rising edge of output signal from the even channel, and stops on the falling edge.

- I/O mode 7: Timer start mode
  This mode controls the start of the odd channel using the even channel. The odd channel starts on the rising edge of output signal from the even channel.

- I/O mode 8: Shared channel signal trigger and timer start/stop mode
  This mode uses an external signal from another channel as an external startup trigger. This mode cannot be selected for channel 0 or 1.

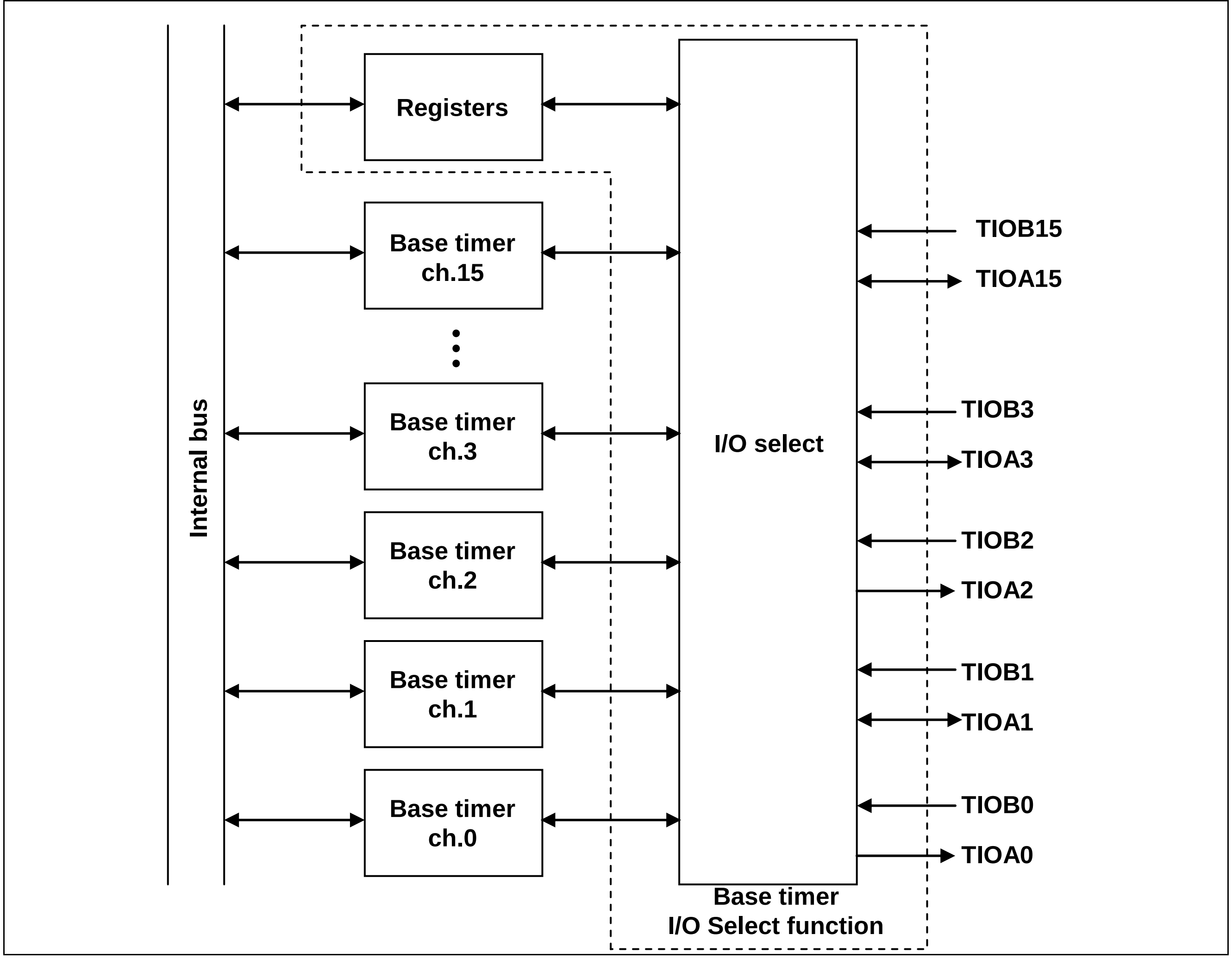
2. Configuration

The base timer I/O select function consists of the following blocks.

■ Block diagram

Figure 2-1 shows the block diagram of the base timer I/O select function.

Figure 2-1 Block diagram of base timer I/O select function



- I/O select
  A circuit that selects the I/O mode of the base timer for each channel.

- Base timer (Channels 0 to 15)
  Base timer channels 0 to 15 (up to 16 channels).

- Registers
  Registers of base timer I/O select function.

3. I/O Mode

This section explains pins used by the base timer I/O select function to set the I/O mode, and also explains each I/O mode.

3.1. Pins

This section explains pins used by the base timer I/O select function to set the I/O mode.

Each channel of the base timer has 2 types of external pins and 5 types of internal signals. Also the base timer I/O select function has 2 types of internal signals. By connecting an internal signal with an external pin, the signal corresponding to the connected (external clock (ECK signal)/external startup trigger (TGIN signal)/waveform (TIN signal)) is input or output to/from the base timer. The external pin and internal signal can be connected by setting the I/O mode of the base timer. The pin used and the signal input or output differ depending on the I/O mode.

■ External pins

- TIOA pin
  This pin is used to output the base timer waveform (TOUT signal), or input an external startup trigger (TGIN signal).

- TIOB pin
  This pin is used to input external startup trigger (TGIN signal)/external clock (ECK signal)/another channel waveform (TIN signal).

■ Internal signals

A signal is input or output to/from the base timer by being connected with an above external pin, or by inputting an output signal from another channel.

- TOUT signal
  This signal is the output waveform of the base timer. (Not used by the 16/32-bit PWC timer.)

- ECK signal
  This signal is an external clock of the base timer. (Not used by the 16/32-bit PWC timer.)
  It is input when the external clock is selected as a counting clock.

- TGIN signal
  This signal is the external startup trigger of the base timer. (Not used by the 16/32-bit PWC timer.)
  When the valid edge of external startup trigger is selected, the base timer detects the edge of this signal to start up.

- TIN signal
  This signal is the input waveform of the base timer. This signal is the waveform to be measured. (Used only by the 16/32-bit PWC timer.)

- DTRG signal
  This signal is the trigger input to the base timer. The base timer stops operating on the falling edge of this signal.

- COUT signal
  This signal is the trigger output of the base timer I/O select function. This signal is output signal to another channel of the base timer.

- CIN signal
  This signal is the trigger input to the base timer I/O select function. This signal is input signal from another channel of the base timer.

■ Connecting the external pin to the internal signal

The external pin and internal signal can be connected by setting the I/O mode of the base timer.

Table 3-1 shows the correspondence between I/O modes and pin connections.

Table 3-1 Correspondence between I/O modes and pin connections

I/O mode	TIOAn (Even channel)		TIOBn (Even channel)		TIOAn+1 (Odd channel)		TIOBn+1 (Odd channel)	
	Connected to	I/O	Connected to	I/O	Connected to	I/O	Connected to	I/O
0	ch.n TOUT	Output	ch.n ECK/TGIN/ TIN	Input	ch.n+1 TOUT	Output	ch.n+1 ECK/TGIN/ TIN	Input
1			ch.n ECK	Input	ch.n TGIN[*2]	Input	ch.n TIN	Input
2			ch.n/ch.n+1 ECK/TGIN/ TIN [*1]	Input	ch.n+1 TOUT	Output	Not used	
3			Not used					
4			ch.n ECK/TGIN/ TIN	Input				
5			Not used					
6								
7			ch.n ECK/TGIN/ TIN	Input				
8			Not used					

n: Even (n=0, 2, 4, 6, 8, 10, 12, 14) However, n depends on the number of channels mounted.
ch.n: Even channel
ch.n+1: Odd channel
*1: Synchronized by the peripheral clock (PCLK)
*2: In I/O mode 1, TIOA Input cannot be used as TGIN input of ch.8. For details, refer “I/O mode 1(timer full mode) in “3.2 I/O Mode”.

3.2. I/O Mode

I/O mode selected by the I/O Select Register (BTSEL) determines the functions of external pins and the start/stop timing of the base timer.

■ I/O mode 0 (Standard 16-bit timer mode)

This mode uses each channel of the base timer individually.

Table 3-2 shows the external pins used when this mode is selected.

Table 3-2 External pins used when I/O mode 0 is selected

	Even channel	Odd channel
Number of input pins	1	1
Number of output pins	1	1

Table 3-3 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-3 External pin connections and input/output signals when I/O mode 0 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOA	Output	TOUT	Outputs the base timer waveform
TIOB	Input	ECK/TGIN/TIN*	Uses the input signal as one of the following signals: ・ External clock (ECK signal) ・ External startup trigger (TGIN signal) ・ Waveform to be measured (TIN signal)

*: The usage of input signals (ECK/TGIN/TIN) differs depending on the Timer Control Register (TMCR) setting of the base timer.

Figure 3-1 provides the block diagram of I/O mode 0 (Standard 16-bit timer mode).

Figure 3-1 I/O mode 0 (Standard 16-bit timer mode) block diagram

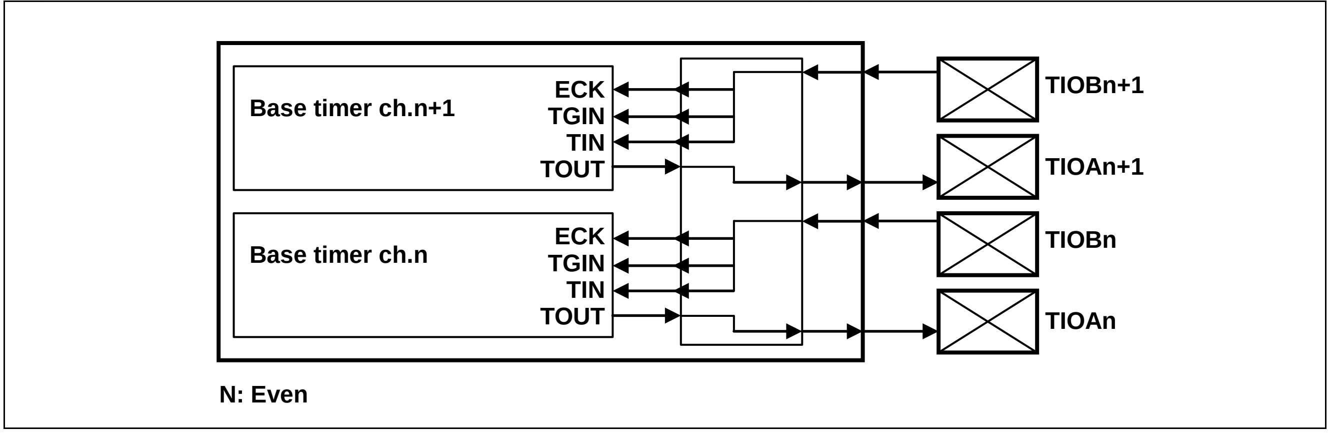


Table 3-4 shows signal connections in I/O mode 0.

Table 3-4 I/O mode 0 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	Output from the TIOAn pin
Input signal from the TIOBn pin	Input to ch.n as ECK/TGIN/TIN signals
ch.n+1 TOUT signal	Output from the TIOAn+1 pin
Input signal from the TIOBn+1 pin	Input to ch.n+1 as ECK/TGIN/TIN signals.

n: Even

■ I/O mode 1 (timer full mode)

This mode assigns every even channel signal with an external pin individually.

Table 3-5 shows the external pins used when this mode is selected.

Table 3-5 External pins used when I/O mode 1 is selected.

	Even channel
Number of input pins	3
Number of output pins	1

Table 3-6 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-6 External pin connections and input/output signals when I/O mode 1 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOBn	Input	Even channel ECK	Inputs an external clock (ECK signal) to the even channel.
TIOAm+1	Input	Even channel TGIN	Inputs an external startup trigger (TGIN signal) to the even channel.
TIOBn+1	Input	Even channel TIN	Inputs the waveform to be measured (TIN signal) to the even channel.

n : 0, 2, 4, 6, 8, 10, 12, 14
m: 0, 2, 4, 6, 10, 12, 14

<Note>

When ch.8 and ch.9 of the base timer are used in I/O mode 1(timer full mode), do not use the external start trigger input, TGIN (TIOA9). To make the setting of not using the external start trigger input, TGIN(TION9), set EGS1 and EGS0 bits of Timer Control Register(ch.9-TMCR) in the Base Timer to trigger input enable(EGS1 and EGS0 are “0b00”).

Figure 3-2 shows the block diagram of I/O mode 1 (timer full mode).

Figure 3-2 I/O mode 1 (timer full mode) block diagram

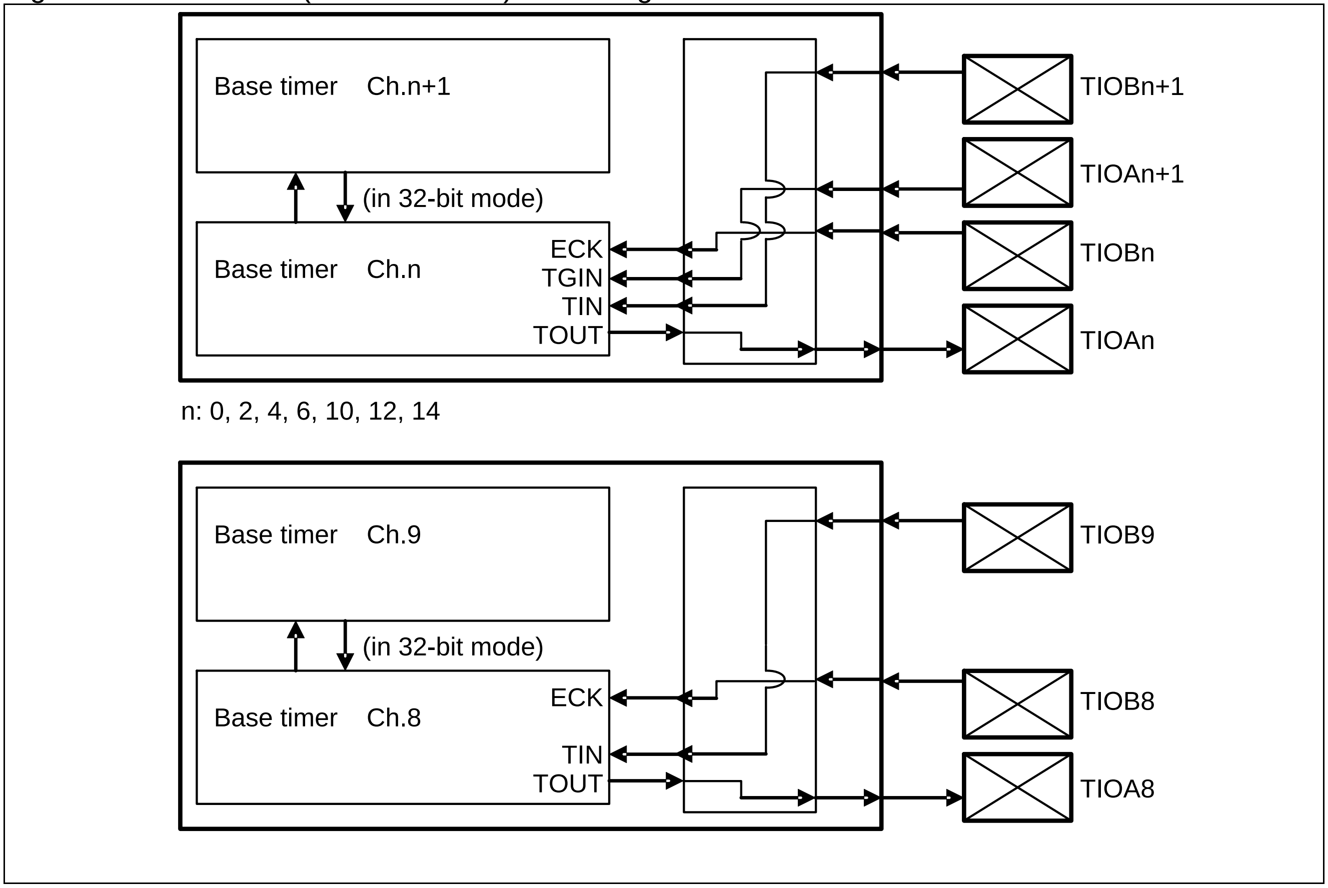


Table 3-7 shows signal connections in I/O mode 1.

Table 3-7 I/O mode 1 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	Output from the TIOAn pin
Input signal from the TIOBn	Input to ch.n as a ECK signal
TIOAm+1 pin	Input to ch.m as a TGIN signal
TIOBn+1 pin	Input to ch.n as an TIN signal

n: 0, 2, 4, 6, 8, 10, 12, 14
m: 0, 2, 4, 6, 10, 12, 14

<Notes>

When this mode is selected, the TIOA pins (TIOA1, TIOA3, etc.) corresponding to the odd channel must be set to port input mode with the Port Function Register (PFR) of GPIO.

When ch.8 and ch.9 of the base timer are used in I/O mode 1(timer full mode), do not use the external start trigger input, TGIN (TIOA9). To make the setting of not using the external start trigger input, TGIN(TION9), set EGS1 and EGS0 bits of Timer Control Register(ch.9-TMCR) in the Base Timer to trigger input enable(EGS1 and EGS0 are “0b00”).

■ I/O mode 2 (Shared external trigger mode)

This mode shares the input signals (ECK/TGIN/TIN) of the base timer between two channels.
Table 3-8 shows the external pins used when this mode is selected.

Table 3-8 External pins used when I/O mode 2 is selected

	Even channel	Odd channel
Number of input pins	1(shared by two channels)	
Number of output pins	1	1

Table 3-9 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-9 External pin connections and input/output signals when I/O mode 2 is selected

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	I/O	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn	Input	ECK/TGIN/TIN of even and odd channels*	Input to both the even and odd channels (synchronized by the peripheral clock (PCLK)) and used as one of the following signals: - External clock (ECK signal) - External startup trigger (TGIN signal) - Waveform to be measured (TIN signal)
TIOBn+1	-	-	Not used

n: Even

*: The usage of input signals (ECK/TGIN/TIN) differs depending on the Timer Control Register (TMCR) setting of the base timer.

Figure 3-3 shows the block diagram of I/O mode 2 (Shared external trigger mode).

Figure 3-3 I/O mode 2 (Shared external trigger mode) block diagram

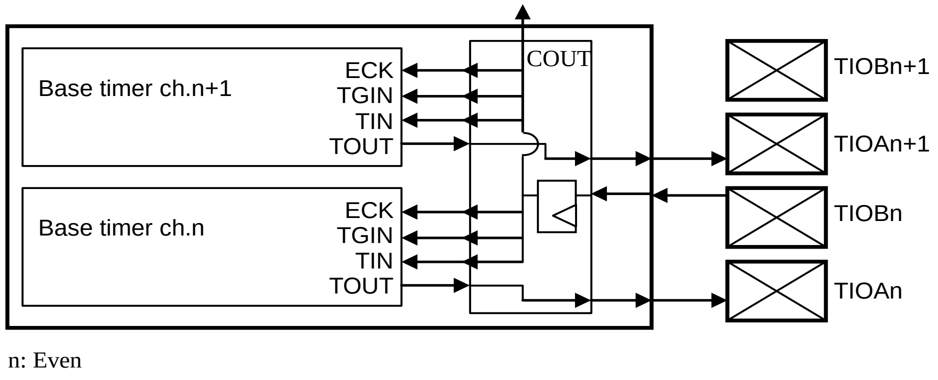


Table 3-10 shows signal connections in I/O mode 2.

Table 3-10 I/O mode 2 signal connections

Connected from (Signal)	Connected to	Remarks
ch.n TOUT signal	Output from the TIOAn pin	
Input signal from the TIOBn pin	・Input to ch.n and ch.n+1 as ECK/TGIN/TIN signals ・Output to another channel as a COUT signal	Synchronized by the peripheral clock (PCLK)
ch.n+1 TOUT signal	Output from the TIOAn+1 pin	

n: Even

<Note>

If the upper two channels of the channels set to this mode (n+2, n+3) are set to I/O mode 3 (Shared channel signal trigger mode), the input signals (ECK/TGIN/TIN) can be input to the 4 channels simultaneously.

(Example: If channels 0 and 1 are set to this mode, and channels 2 and 3 are set to I/O mode 3, input signals (ECK/TGIN/TIN) can be input to four channels of 0 to 3 simultaneously.)

■ I/O mode 3 (Shared channel signal trigger mode)

This mode inputs the COUT signal from channels of the lower two channels as a CIN signal, and uses it as ECK/TGIN/TIN signals.

Table 3-11 shows the external pins used when this mode is selected.

Table 3-11 External pins used when I/O mode 3 is selected.

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-12 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-12 External pin connections and input/output signals when I/O mode 3 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-4 shows the block diagram of I/O mode 3 (Shared channel signal trigger mode).

Figure 3-4 I/O mode 3 (Shared channel signal trigger mode) block diagram

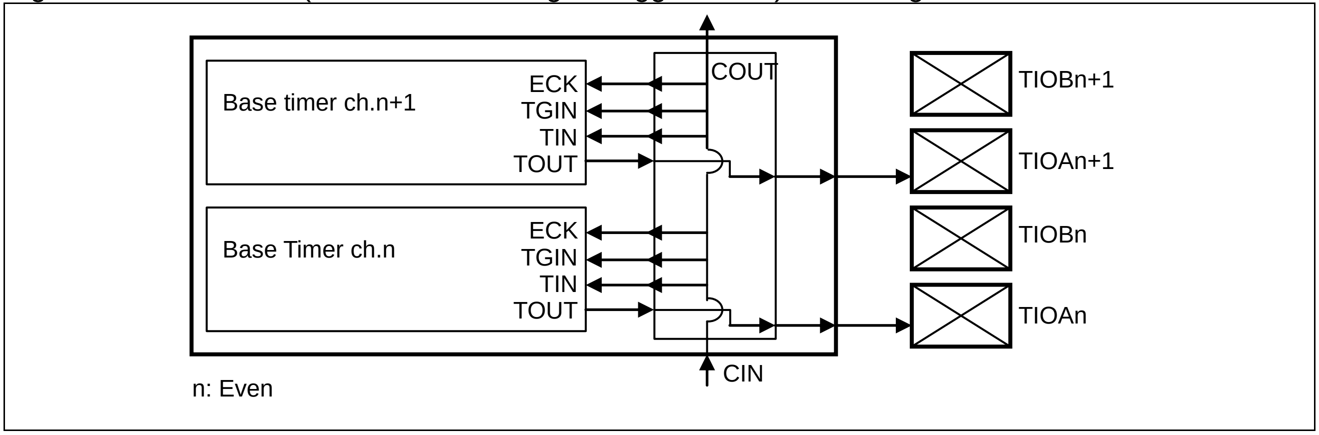


Table 3-13 shows signal connections in I/O mode 3.

Table 3-13 I/O mode 3 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	Output from the TIOAn pin
CIN signal*	・ Input to ch.n and ch.n+1 as ECK/TGIN/TIN signals ・ Output to another channel as a COUT signal
ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

*: The COUT signal from another channel is input as a CIN signal.

The following shows ch.n-2/n-1 signals that can be input to ECK/TGIN/TIN of ch.n/n+1.

- Signal that the peripheral clock generates by synchronizing TIOBn-2 input in I/O mode 2.
- Trigger signal input from ch.n-4/n-3 in I/O mode 3.
- TIOAn-2 output in I/O mode 4.
- TIOAn-2 output in I/O mode 6.
- TIOAn-2 output in I/O mode 7
- Trigger signal input from ch.n-4/n-3 in I/O mode 8.

<Notes>

- Select the rising edge as a trigger input edge using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
- The channels set to this mode use the COUT signal from lower two channels (n-2 and n-1) as a CIN signal. (Example: If channels 2 and 3 are set to this mode, they use the COUT signal from channels 0 and 1.) Therefore, channels 0 and 1 cannot be set to this mode.

■ I/O mode 4 (Timer start/stop mode)

This mode can control the start/stop of the odd channel using the even channel.

The odd channel starts on the rising edge of output waveform (TOUT signal) of the even channel, and stops on the falling edge.

Table 3-14 shows the external pins used when this mode is selected.

Table 3-14 External pins used when I/O mode 4 is selected.

	Even channel	Odd channel
Number of input pins	1	Not used
Number of output pins	1	1

Table 3-15 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-15 External pin connections and input/output signals when I/O mode 4 is selected

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn	Input	ECK/TGIN/TIN of even channel*	Input to the even channel and used as one of the following signals: - External clock (ECK signal) - External startup trigger (TGIN signal) - Waveform to be measured (TIN signal)
TIOBn+1	-	-	Not used

n: Even

*: The usage of input signals (ECK/TGIN/TIN) differs depending on the Timer Control Register (TMCR) setting of the base timer.

Figure 3-5 shows the block diagram of I/O mode 4 (Timer start/stop mode).

Figure 3-5 I/O mode 4 (Timer start/stop mode) block diagram

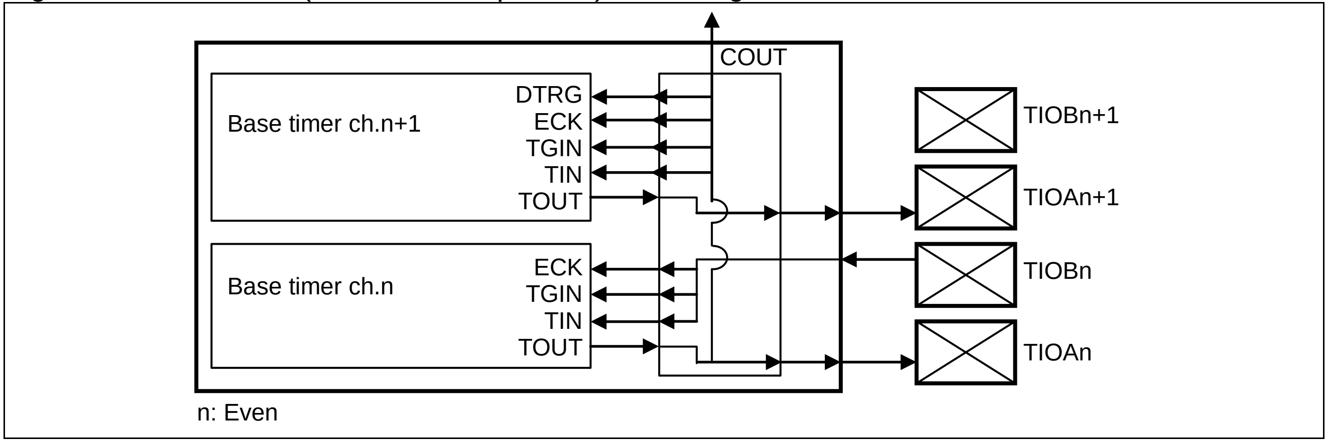


Table 3-16 shows signal connections in I/O mode 4.

Table 3-16 I/O mode 4 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	・Output from the TIOAn signal ・Input to ch.n+1 as ECK/TGIN/TIN and DTRG signals ・Output to another channel as a COUT signal
Input signal from the TIOBn pin	Input to ch.n as ECK/TGIN/TIN signal
ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

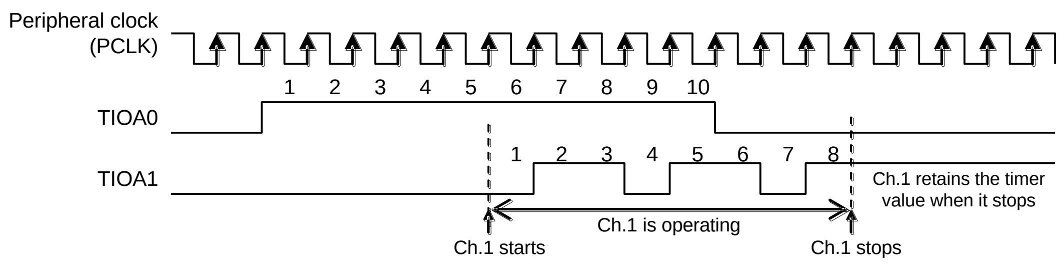
<Notes>

- Select the rising edge as a trigger input edge of the odd channel using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
- The odd channel stops operating when a falling edge is detected in the DTRG signal.

Figure 3-6 shows example operation when I/O mode 4 (Timer start/stop mode) is selected, and when channels 0 and 1 are used as PWM timer

Base timer ch.0	Set value	Base timer ch.1	Set value
Cycle Setup Register (PCSR)	0x0010	Cycle Setup Register (PCSR)	0x0002
Duty Setup Register (PDUT)	0x0009	Duty Setup Register (PDUT)	0x0001
Timer Control Register (TMCR)	0x0013	Timer Control Register (TMCR)	0x0112

Figure 3-6 I/O mode 4 (Timer start/stop mode) operation example



■ I/O mode 5 (Software-based simultaneous startup mode)

This mode starts up multiple channels simultaneously using the Software-based Simultaneous Startup Register (BTSSSR).

All the channels corresponding to the Software-based Simultaneous Startup Register (BTSSSR) bits that have been set to "1" start up simultaneously.

Table 3-17 shows the external pins used when this mode is selected.

Table 3-17 External pins used when I/O mode 5 is selected

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-18 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-18 External pin connections and input/output signals when I/O mode 5 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-7 shows the block diagram of I/O mode 5 (Software-based simultaneous startup mode).

Figure 3-7 I/O mode 5 (Software-based simultaneous startup mode) block diagram

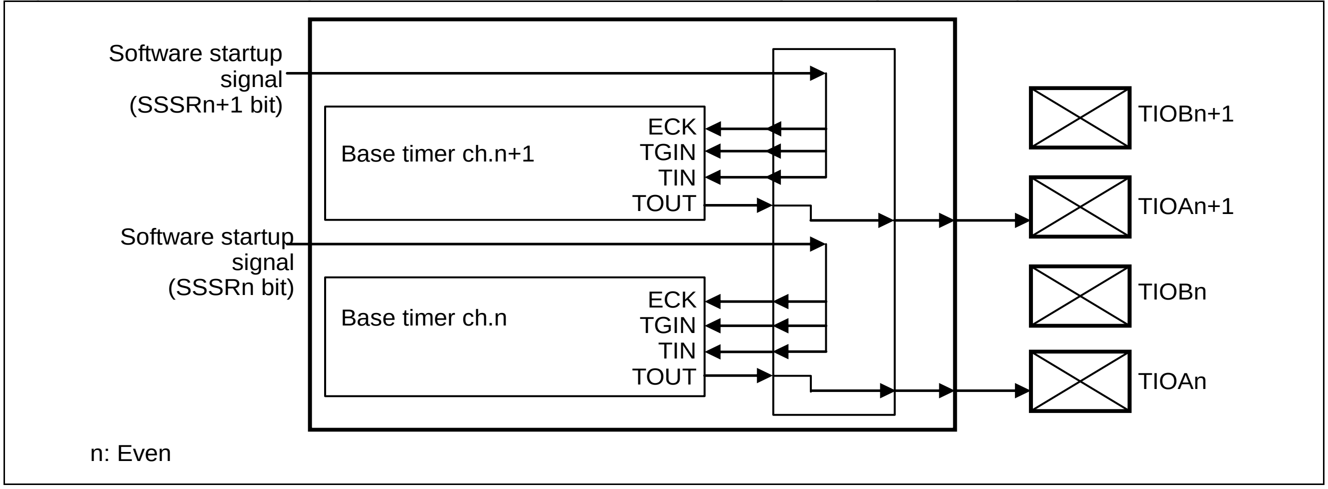


Table 3-19 shows signal connections in I/O mode 5.

Table 3-19 I/O mode 5 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	Output from the TIOAn pin
Software startup signal (Write "1" to the SSSRn bit in the BTSSSR)	Input to ch.n as ECK/TGIN/TIN signal
ch.n+1 TOUT signal	Output from the TIOAn+1 pin
Software startup signal (Write "1" to the SSSRn+1 bit in the BTSSSR)	Input to ch.n+1 as ECK/TGIN/TIN signals

n: Even

BTSSSR: Software-based Simultaneous Startup Register

When "1" is written to a Software-based Simultaneous Startup Register (BTSSSR), a rising edge is input (ECK/TGIN/TIN signals) to the channel corresponding to the bit.

<Note>

Select the rising edge as a trigger input edge using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)

■ I/O mode 6 (Software-based startup and timer start/stop mode)

This mode can control the start/stop of the odd channel using the even channel.
The even channel can be started by writing "1" to the Software-based Simultaneous Startup Register (BTSSSR).
The odd channel starts when the rising edge is detected in output waveform (TOUT signal) of the even channel, and stops when the falling edge is detected.
Table 3-20 shows the external pins used when this mode is selected.

Table 3-20 External pins used when I/O mode 6 is selected.

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-21 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-21 External pin connections and input/output signals when I/O mode 6 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-8 shows the block diagram of I/O mode 6 (Software-based startup and timer start/stop mode).

Figure 3-8 I/O mode 6 (Software-based startup and timer start/stop mode) block diagram

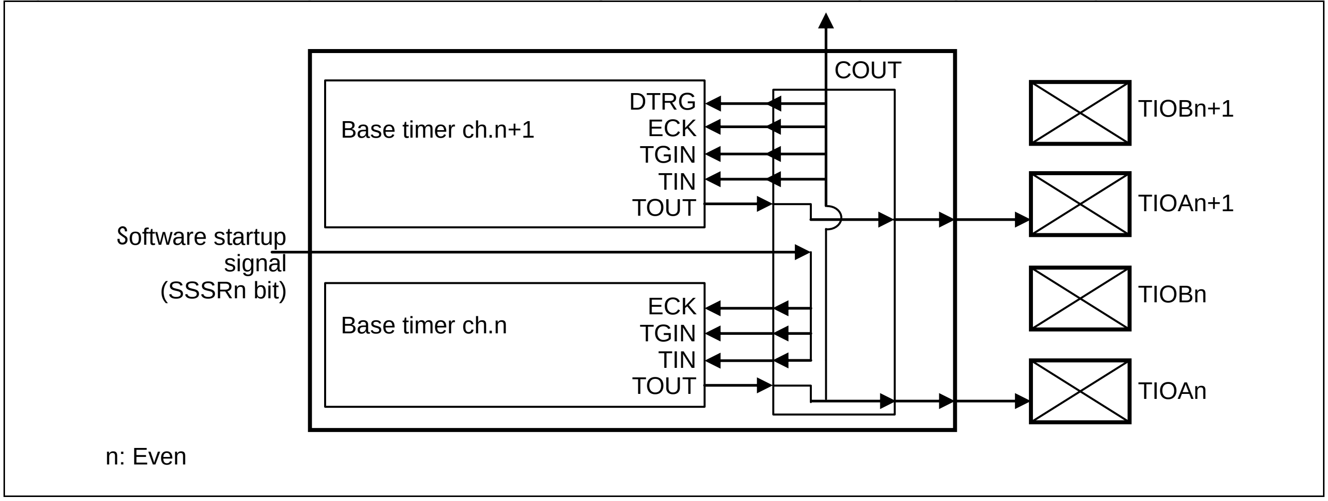


Table 3-22 shows signal connections in I/O mode 6.

Table 3-22 I/O mode 6 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	・Output from the TIOAn pin ・Input to ch.n+1 as ECK/TGIN/TIN/DTRG signals ・Output to another channel as a COUT signal
Software startup signal (Write "1" to the SSSRn bit in the BTSSSR)	Input to ch.n as ECK/TGIN/TIN signals
ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

BTSSSR: Software-based Simultaneous Startup Register

When "1" is written to the Software-based Simultaneous Startup Register (BTSSSR) bit corresponding to the even channel you want to start up, a rising edge is input (ECK/TGIN/TIN signals) to the channel..
The start/stop timing of ch.n is the same as that for I/O mode 4.

<Notes>

- Select the rising edge as a trigger input edge using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
- The odd channel stops operating when a falling edge is detected in the DTRG signal.

■ I/O mode 7 (Timer start mode)

This mode uses the output waveform (TOUT signal) from the even channel as input signals (ECK/TGIN/TIN signals) of the odd channel.

Table 3-23 shows the external pins used when this mode is selected.

Table 3-23 External pins used when I/O mode 7 is selected.

	Even channel	Even channel
Number of input pins	1	Not used
Number of output pins	1	1

Table 3-24 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-24 External pin connections and input/output signals when I/O mode 7 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn	Input	Even channel ECK/ TGIN/TIN*	Input to the even channel and used as one of the following signals: = External clock (ECK signal) ・ External startup trigger (TGIN signal) ・ Waveform to be measured (TIN signal)
TIOBn+1	-	-	Not used

n: Even

*: The usage of input waveforms (ECK/TGIN/TIN signals) differs depending on the Timer Control Register (TMCR) setting.

Figure 3-9 shows the block diagram of I/O mode 7 (Timer start mode).

Figure 3-9 I/O mode 7 (Timer start mode) block diagram

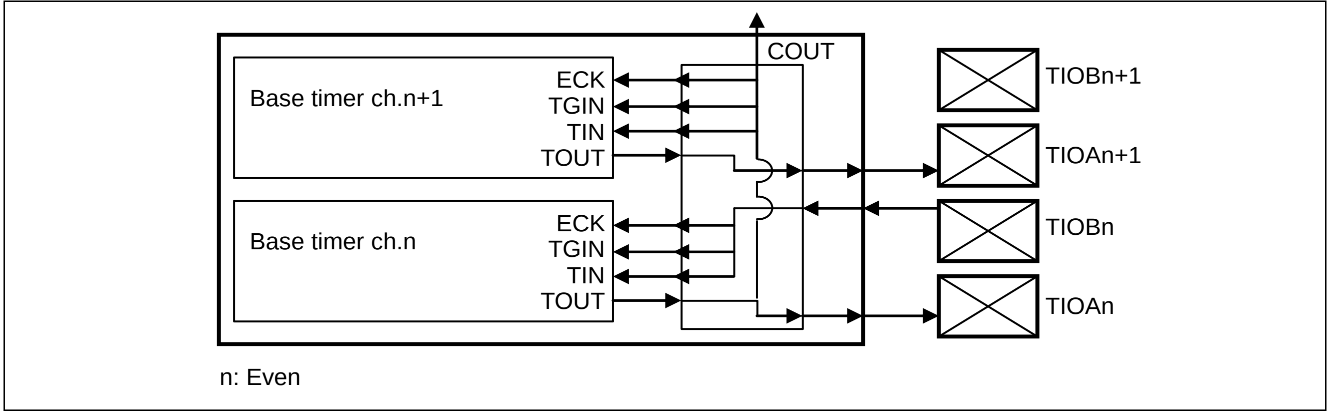


Table 3-25 shows signal connections in I/O mode 7.

Table 3-25 I/O mode 7 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	・ Output from the TIOAn pin ・ Input to ch.n+1 as ECK/TGIN/TIN signals ・ Output to another channel as a COUT signal
Input signal from the TIOBn pin	Input to ch.n as ECK/TGIN/TIN signals
ch.n+1 TOUT signal	Output from the TIOAn+1 pin

n: Even

The start timing of ch.n is the same as that for I/O mode 4.

■ I/O mode 8 (Shared channel signal trigger and timer start/stop mode)

This mode inputs the COUT signal from channels of the lower two channels as a CIN signal, and uses it as an external startup trigger (TGIN signal).

Table 3-26 shows the external pins used when this mode is selected.

Table 3-26 External pins used when I/O mode 8 is selected.

	Even channel	Odd channel
Number of input pins	Not used	
Number of output pins	1	1

Table 3-27 shows the internal signals to which the external pins connect, and signals input or output.

Table 3-27 External pin connections and input/output signals when I/O mode 8 is selected.

External pin	I/O	Connected to (internal signal)	Signal input/output
TIOAn	Output	Even channel TOUT	Outputs the even channel waveform
TIOAn+1	Output	Odd channel TOUT	Outputs the odd channel waveform
TIOBn TIOBn+1	-	-	Not used

n: Even

Figure 3-10 shows the block diagram of I/O mode 8 (Shared channel signal trigger and timer start/stop mode).

Figure 3-10 I/O mode 8 (Shared channel signal trigger and timer start/stop mode) block diagram

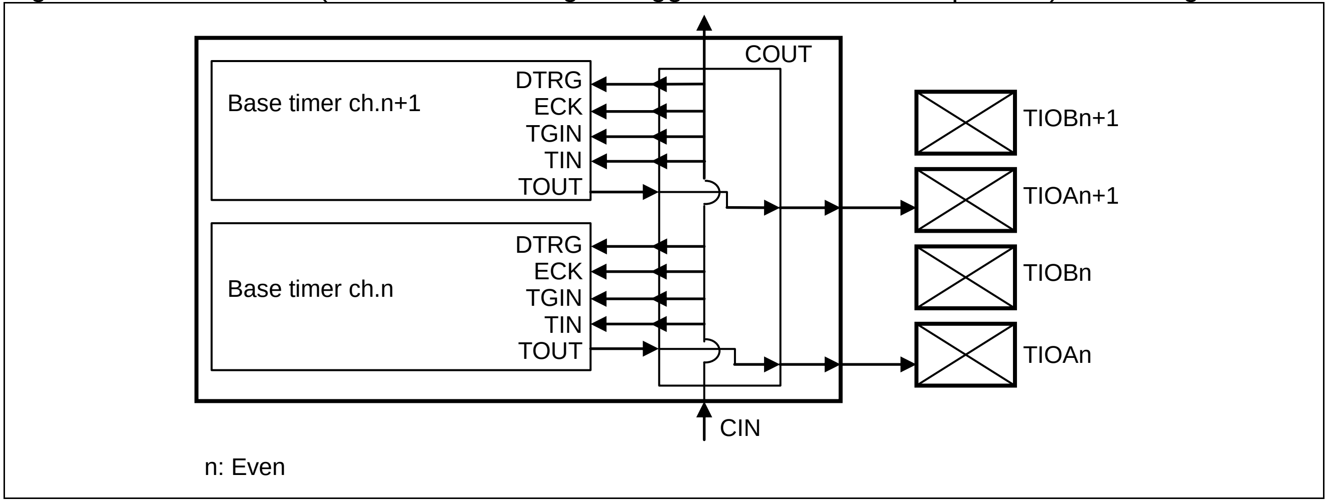


Table 3-28 shows signal connections in I/O mode 8.

Table 3-28 I/O mode 8 signal connections

Connected from (Signal)	Connected to
ch.n TOUT signal	Output from the TIOAn pin
ch.n+1 TOUT signal	Output from the TIOAn+1 pin
CIN signal*	・ Input to ch.n and ch.n+1 as ECK/TGIN/TIN and DTRG signals ・ Output to another channel as a COUT signal

n: Even

* : The COUT signal from another channel is input as a CIN signal.

The following shows ch.n-2/n-1 signals that can be input to ECK/TGIN/TIN of ch.n/n+1.

- Signal that the peripheral clock generates by synchronizing TIOBn-2 input in I/O mode 2.
- Trigger signal input from Ch.n-4/n-3 in I/O mode 3.
- TIOAn-2 output in I/O mode 4.
- TIOAn-2 output in I/O mode 6.
- TIOAn-2 output in I/O mode 7.
- Trigger signal input from Ch.n-4/n-3 in I/O mode 8.

<Notes>

- The channels set to this mode use the COUT signal from lower 2 channels (n-2 and n-1) as a CIN signal. (Example: If channels 2 and 3 are set to this mode, they use the COUT signal from channels 0 and 1.) Therefore, channels 0 and 1 cannot be set to this mode.
- Select the rising edge as a trigger input edge, for the channel set in this mode, using the EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)
  However, do not enable this setting if the timer function is set to the 16/32-bit PWC timer using FMD2 to FMD0 bits in the Timer Control Register (TMCR) of the base timer (FMD2 to FMD0 are set to 0b100).
- Base timer stops operating when a falling edge is detected in the DTRG signal.

4. Registers

This section provides the register list of the base timer I/O select function.

■ Base Timer I/O Select Function Registers

Table 4-1 Register list of Base timer I/O select function

Abbreviation	Register name	Reference
BTSEL0123	I/O Select Register	4.1
BTSEL4567	I/O Select Register	4.2
BTSEL89AB	I/O Select Register	4.3
BTSELCDEF	I/O Select Register	4.4
BTSSSR	Software-based Simultaneous Startup Register	4.5

4.1. I/O Select Register (BTSEL0123)

This register selects the I/O mode for channels 0 to 3 of the base timer.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SEL23_3	SEL23_2	SEL23_1	SEL23_0	SEL01_3	SEL01_2	SEL01_1	SEL01_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SEL23_3 to SEL23_0: I/O select bits for ch.2/ch.3

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

[bit11:8] SEL01_3 to SEL01_0: I/O select bits for ch.0/ch.1

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Notes>

- Channels 0 and 1 are the lowest channels of the base timer, and cannot use the modes that use signal from lower channels. Therefore, the following modes cannot be selected for the channels:
  - I/O mode 3 (Shared channel signal trigger mode)
  - I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
- Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

4.2. I/O Select Register (BTSEL4567)

This register selects the I/O mode for channels 4 to 7 of the base timer.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SEL67_3	SEL67_2	SEL67_1	SEL67_0	SEL45_3	SEL45_2	SEL45_1	SEL45_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SEL67_3 to SEL67_0: I/O select bits for ch.6/ch.7

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

[bit11:8] SEL45_3 to SEL45_0: I/O select bits for ch.4/ch.5

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Note>

Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

4.3. I/O Select Register (BTSEL89AB)

This register selects the I/O mode for channels 8 to 11 of the base timer

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SELAB_3	SELAB_2	SELAB_1	SELAB_0	SEL89_3	SEL89_2	SEL89_1	SEL89_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SELAB_3 to SELAB_0: I/O select bits for Ch.10/Ch.11

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

[bit11:8] SEL89_3 to SEL89_0: I/O select bits for Ch.8/Ch.9

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Note>

Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

When ch.8 and ch.9 of the base timer are used in I/O mode 1(timer full mode), do not use the external start trigger input, TGIN (TIOA9). To make the setting of not using the external start trigger input, TGIN(TION9), set EGS1 and EGS0 bits of Timer Control Register(ch.9-TMCR) in the Base Timer to trigger input enable(EGS1 and EGS0 are "0b00").

4.4. I/O Select Register (BTSELCDEF)

This register selects the I/O mode for channels 12 to 15 of the base timer.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	SELEF_3	SELEF_2	SELEF_1	SELEF_0	SELCD_3	SELCD_2	SELCD_1	SELCD_0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:12] SELEF_3 to SELEF_0: I/O select bits for ch.14/ch.15

bit15	bit14	bit13	bit12	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited

[bit11:8] SELCD_3 to SELCD_0: I/O select bits for ch.12/ch.13

bit11	bit10	bit9	bit8	I/O select bits
0	0	0	0	I/O mode 0 (Standard 16-bit timer mode)
0	0	0	1	I/O mode 1 (Timer full mode)
0	0	1	0	I/O mode 2 (Shared external trigger mode)
0	0	1	1	I/O mode 3 (Shared channel signal trigger mode)
0	1	0	0	I/O mode 4 (Timer start/stop mode)
0	1	0	1	I/O mode 5 (Software-based simultaneous startup mode)
0	1	1	0	I/O mode 6 (Software-based startup and timer start/stop mode)
0	1	1	1	I/O mode 7 (Timer start mode)
1	0	0	0	I/O mode 8 (Shared channel signal trigger and timer start/stop mode)
Others				Setting is prohibited.

<Note>

Before rewriting this register, set the base timer to reset mode using the FMD[2:0] bits in the Timer Control Register (TMCR) of the base timer. (Set FMD[2:0] to 0b000.)

4.5. Software-based Simultaneous Startup Register (BTSSSR)

This register starts up the base timer using software simultaneously
Up to 16 channels can be started simultaneously if the bits corresponding to the channel are set to "1".

■ Register configuration

bit	15	0
Field	SSSR15 to SSSR0	
Attribute	W	
Initial value	0xXXXX	

■ Register functions

[bit15:0] SSSR15 to SSSR0: Software-based simultaneous startup bits

Value	Software-based simultaneous startup bits
0	Writing "0" to these bits is invalid
1	Starts ch.x of the base timer

x : 15 to 0

<Notes>

- Do not write to this register unless set to either of the following modes:
  - I/O mode 5 (Software-based simultaneous startup mode)
  - I/O mode 6 (Software-based startup and timer start/stop mode)(Even channels only)
- For the channel started up by using this register, select the rising edge as a trigger input edge using the - - - EGS1 and EGS0 bits in the Timer Control Register (TMCR) of the base timer. (Set EGS1 and EGS0 to 0b01.)

CHAPTER 5-4: Base Timer

This chapter explains the functions and operations of the base timer.

CODE: FM10-E03.2

1. Overview of Base Timer

The function of the base timer can be set to either of the reset mode, 16-bit PWM timer, 16-bit PPG timer, 16-/32-bit reload timer, or 16/32-bit PWC timer using the FMD[2:0] bits in the Timer Control Register (TMCR). The following provides an overview of each selectable timer function.

■ Relationship between mode settings and timer functions

Settings of FMD[2:0] bits of Timer Control Register (TMCR)	Function
000	Reset mode
001	16-bit PWM timer
010	16-bit PPG timer
011	16/32-bit reload timer
100	16/32-bit PWC timer

■ Reset mode

The reset mode is a status where the base timer macros are reset (with each register set to the initial value). Be sure to set this mode before switching to a different timer function or T32 bit setting. However, it is not necessary to set this mode before setting the timer function or T32 bit immediately after the macros are reset.

■ 16-bit PWM timer

This timer consists of a 16-bit down counter, a 16-bit data register with a cycle set buffer, a 16-bit compare register with a duty set buffer, and a pin controller.

The cycle and duty data is stored in a buffered register and thus can be rewritten while the timer is in operation.

The counter clock of the 16-bit down counter can be selected from eight internal clocks (1, 4, 16, 128, 256, 512, 1024, and 2048 frequency divisions of the machine clock) and three external events (detection of a rising edge, a falling edge, or both).

The one-shot mode where the counting stops at an underflow or the continuous mode where the counting is repeated after reloading can be selected.

The start event of the 16-bit PWM timer can be selected from a software trigger and three external events (detection of a rising edge, a falling edge, or both).

■ 16-bit PPG timer

This timer consists of a 16-bit down counter, a 16-bit data register for setting the HIGH width, a 16-bit data register for setting the LOW width, and a pin controller.

The count clock of the 16-bit down counter can be selected from eight internal clocks (1, 4, 16, 128, 256, 512, 1024, and 2048 frequency divisions of the machine clock) and three external events (detection of a rising edge, a falling edge, or both).

The one-shot mode where the counting stops at an underflow or the continuous mode where the counting is repeated after reloading can be selected.

The start event of the 16-bit PPG timer can be selected from a software trigger and three external events (detection of a rising edge, a falling edge, or both).

■ 16-/32-bit reload timer

This timer consists of a 16-bit down counter, a 16-bit reload register, and a pin controller.

The count clock of the 16-bit down counter can be selected from eight internal clocks (1, 4, 16, 128, 256, 512, 1024, and 2048 frequency divisions of the machine clock) and three external events (detection of a rising edge, a falling edge, or both).

The one-shot mode where the counting stops at an underflow or the continuous mode where the counting is repeated after reloading can be selected.

The start event of the 16/32-bit reload timer can be selected from a software trigger and three external events (detection of a rising edge, a falling edge, or both).

■ 16-/32-bit PWC timer

This timer consists of a 16-bit up counter, a measurement input pin, and a control register.

This timer measures the time between any events using an external pulse input.

The reference count clock can be selected from eight internal clocks (1, 4, 16, 128, 256, 512, 1024, and 2048 frequency divisions).

Measurement modes	HIGH pulse width (↑ to ↓) / LOW pulse width (↓ to ↑) Rising cycle (↑ to ↑) / Falling cycle (↓ to ↓) Edge interval measurement (↑ or ↓ to ↓ or ↑)

An interrupt request can be generated when the measurement is completed.

One-time or continuous measurement can be selected.

2. Block Diagram of Base Timer

Figure 2-1 to Figure 2-4 show block diagrams of the base timer in each mode.

Figure 2-1 Block diagram of 16-bit PWM timer

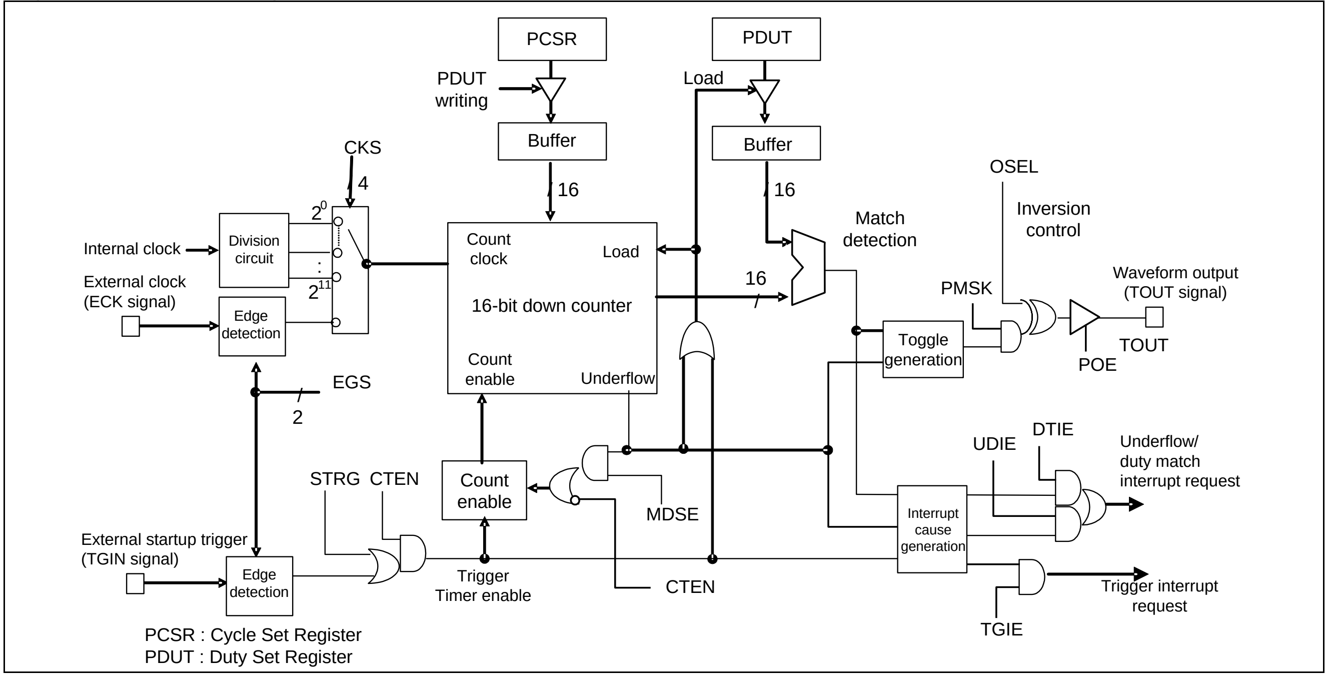


Figure 2-2 Block diagram of 16-bit PPG timer

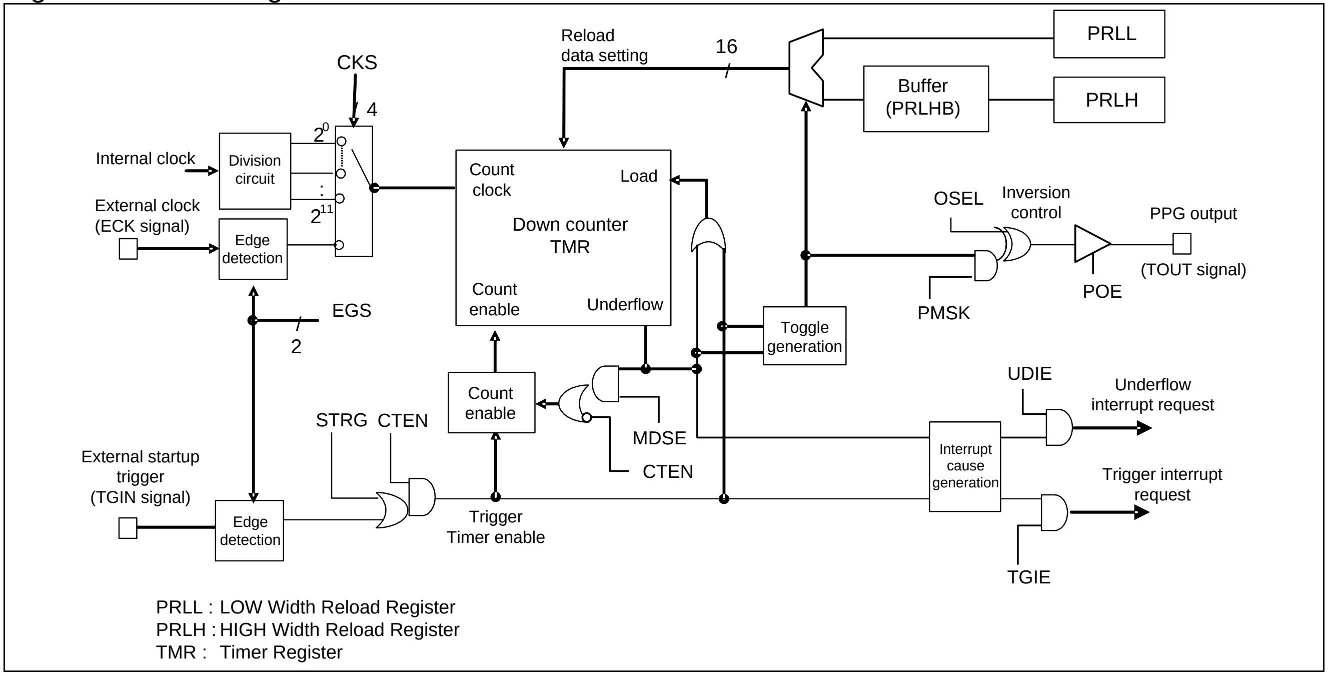
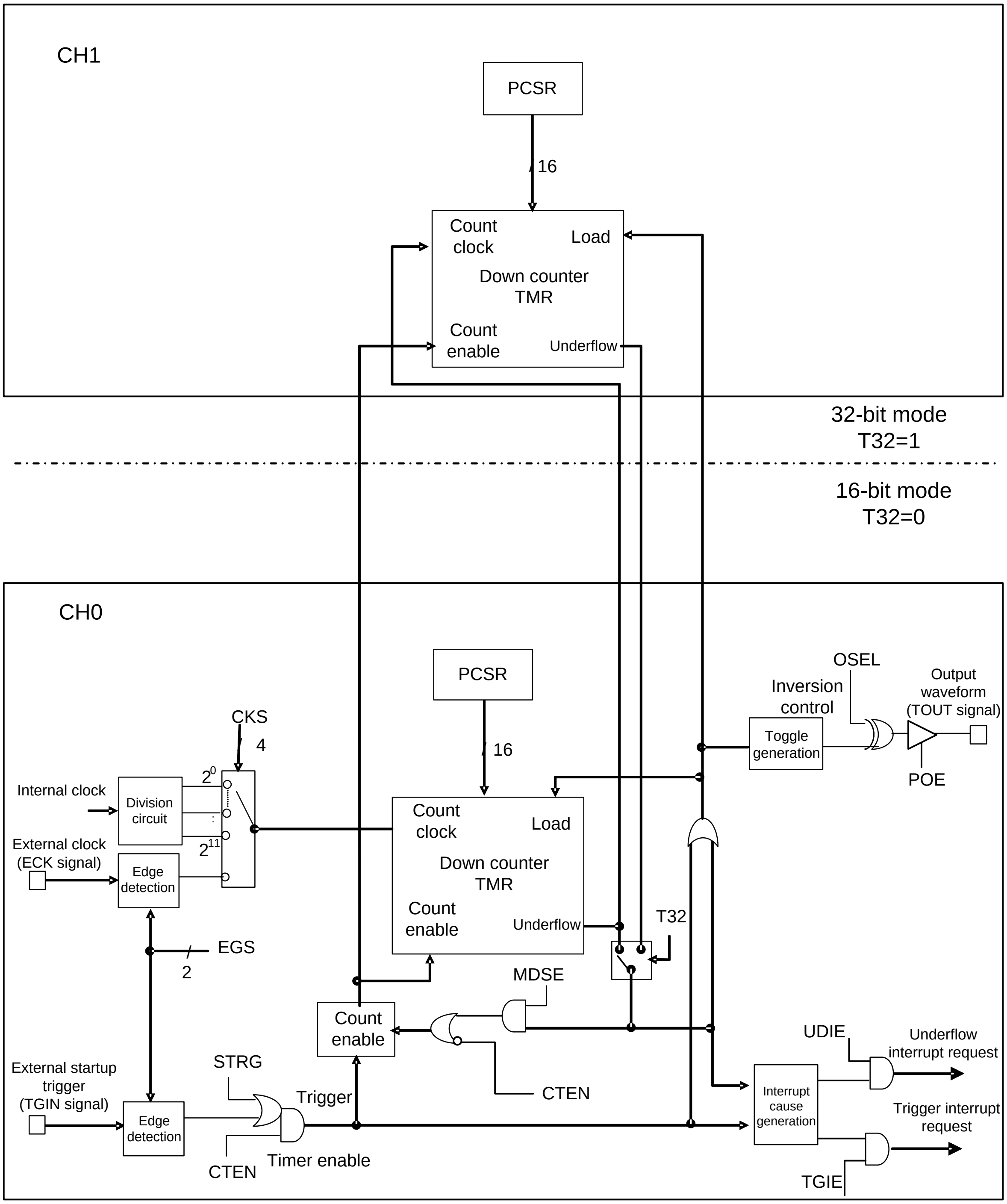
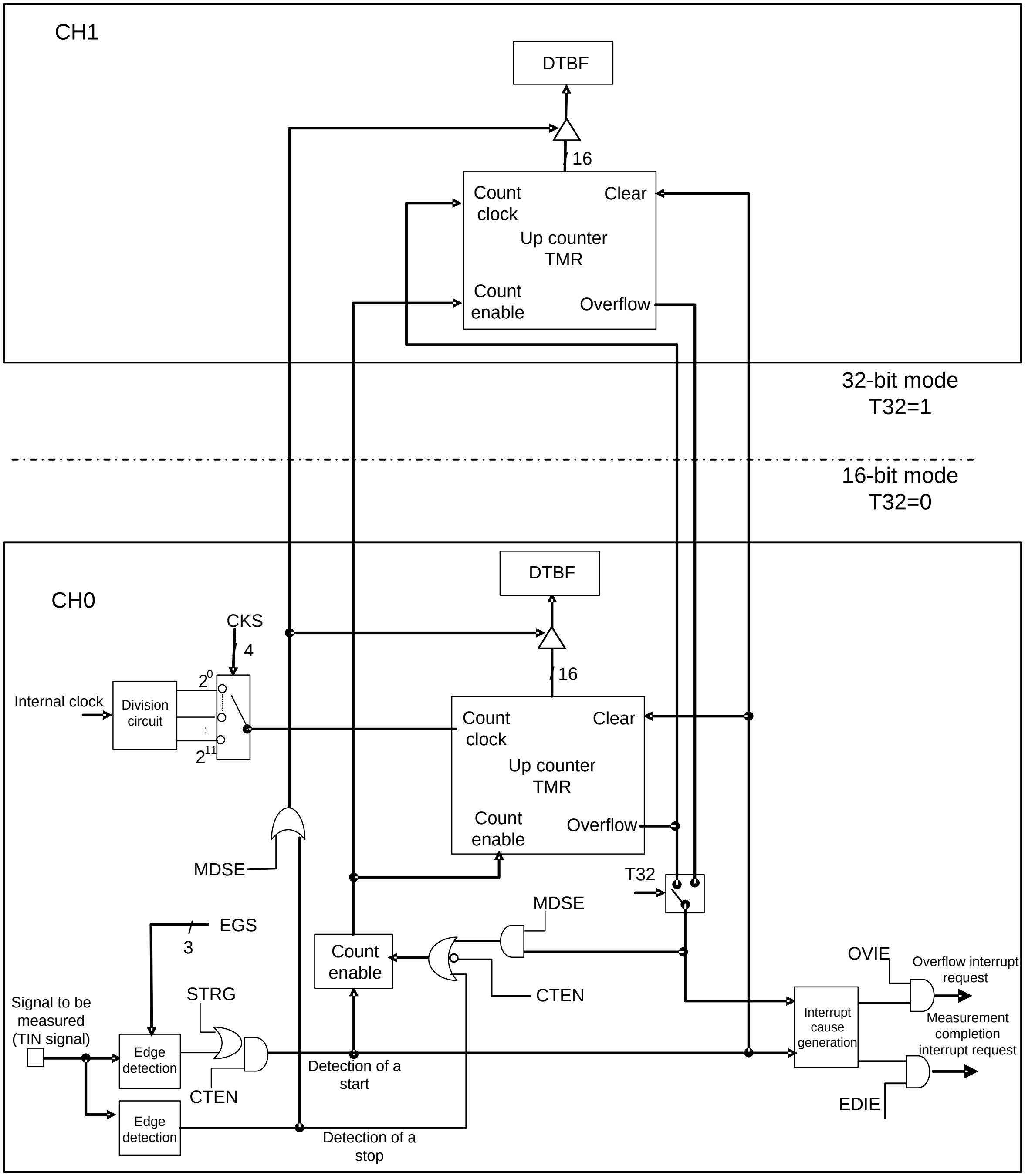


Figure 2-3 Block diagram of 16/32-bit reload timer (ch.1 and ch.0)



PCSR : Base Timer 1 Cycle Set Register
TMR : Base Timer 1 Timer Register
PCSR : Base Timer 0 Cycle Set Register
TMR : Base Timer 0 Timer Register

Figure 2-4 Block diagram of 16/32-bit PWC timer (ch.1 and ch.0)



DTBF : Base Timer 1 Data Buffer Register
TMR : Base Timer 1 Timer Register
DTBF : Base Timer 0 Data Buffer Register
TMR : Base Timer 0 Timer Register

3. Operations of the Base Timer

This section explains operations of the base timer.

■ Operations of the base timer

● Reset mode

The reset mode is a status where the base timer macros are reset (with each register set to the initial value). Be sure to set this mode before switching to a different timer function or T32 bit setting. However, it is not necessary to set this mode before setting the timer function or T32 bit immediately after the macros are reset. In a 32-bit mode, setting this mode for the even channel also resets the odd channel. It is not necessary to set the reset mode for the odd channel.

● 16-bit PWM timer

When triggered, the 16-bit PWM timer starts decrementing from the cycle set value. First, it outputs a LOW level pulse. When the 16-bit down counter matches the value set in the PWM Duty Set Register, the output inverts to the HIGH level. Then, the output inverts again to the LOW level when a counter underflow occurs. This can generate waveforms with any cycle and duty.

● 16-bit PPG timer

When triggered, the 16-bit PPG timer starts decrementing from the value set in the LOW Width Reload Register. First, it outputs a LOW level pulse. The output inverts to the HIGH level upon an underflow. Then, it starts decrementing from the value set in the HIGH Width Reload Register. The output inverts to the LOW level when an underflow occurs. This can generate waveforms having any LOW and HIGH widths.

● 16-bit reload timer

When triggered, the 16-bit reload timer starts decrementing from the cycle set value. When an underflow occurs on the 16-bit down counter, an interrupt flag is set. The output is either the toggle output where the level inverts according to the MDSE bit setting as an underflow occurs or the pulse output where the level is HIGH at the start of counting and LOW at the occurrence of an underflow.

● 32-bit reload timer

This timer has the same basic operations as the 16-bit reload timer. However, it uses two channels, even and odd, to operate as a 32-bit reload timer. The even channel operates as a lower 16-bit timer and the odd channel as an upper 16-bit timer. The interrupt control and output waveform control are defined by the settings for the even channel only. When setting the cycle, first write it in the upper register (odd channel) and then in the lower register (even channel). When reading the timer value, read it from the lower register (even channel) and then from the upper channel (odd channel).

● 16-bit PWC timer

The PWC timer starts the 16-bit up counter with input of the specified measurement start edge and stops the counter with detection of a measurement end edge. The value counted in between is stored as a pulse width in the data buffer register.

● 32-bit PWC timer

This timer has the same basic operations as the 16-bit PWC timer. However, it uses two channels, even and odd, to operate as a 32-bit PWC timer. The even channel operates as a lower 16-bit counter and the odd channel as an upper 16-bit counter. The interrupt control is defined by the settings for the even channel only. When reading the measured or count value, read it from the lower register (even channel) and then from the upper channel (odd channel).

4. 32-bit mode operations

Using two channels, the reload timer and PWC provide 32-bit mode operations. This section explains the basic functions and operations of the 32-bit mode functions.

■ 32-bit mode functions

This function enables the operations of a 32-bit data reload timer or 32-bit data PWC timer by combining two channels of base timers. Since the upper 16-bit timer counter value in the odd channel is read together with the lower 16-bit timer counter value in the even channel, the timer counter value can be read during operation.

■ 32-bit mode settings

First, set “0b000” to set the reset mode to reset the status of the FMD[2:0] bits in the TMCR register of the even channel. Then, as carried out for in 16-bit mode, select the reload or PWC timer and set the operation. By writing “1” also to the T32 bit in the TMCR register, the 32-bit operation mode is set. Do not change “0” for the T32 bit in the odd channel. It is also not necessary to set it to reset mode. For the reload timer, set the reload value of the upper 16 of 32 bits in the PWM Cycle Set Register of the odd channel. Then, set the reload value of the lower 16 bits in the PWM Cycle Set Register of the even channel.

Because transition to 32-bit operation mode is reflected immediately after the T32 bit is written, stop the counting before changing the settings in each channel.

To change from 32-bit mode to 16-bit mode, set it to reset mode by setting “0b000” for the FMD[2:0] bits in the TMCR register of the even channel. This resets the status of both even and odd channels, enabling settings in 16-bit mode in each channel.

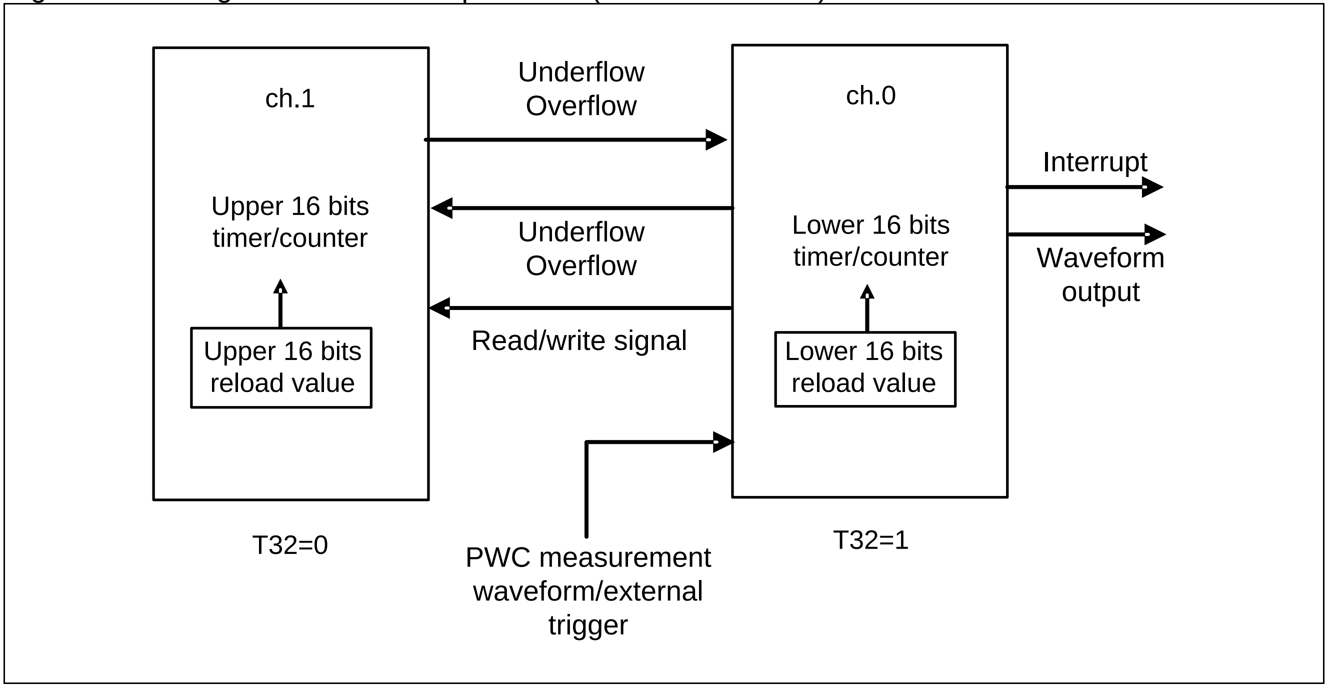
■ 32-bit mode operations

After transition to 32-bit mode, if the reload or PWC timer is started by control of the even channel, the timer/counter in the even channel operates with the lower 16 bits. Also, the time/counter in the odd channel operates with the upper 16 bits.

The operations in 32-bit mode are defined by the settings for the even channel. For this reason, the settings for the odd channel (except the Cycle Set Register for the reload timer) are ignored. For the timer start, waveform output, and interrupt signal functions, settings for the even channel are also applied (the even channel is masked and fixed to “LOW”).

Figure 4-1 shows the configuration of ch.0 and ch.1.

Figure 4-1 Configuration of 32-bit operations (for ch.0 and ch.1)


ch.1
Upper 16 bits timer/counter
Upper 16 bits reload value
T32=0
Underflow Overflow
Underflow Overflow
Read/write signal
PWC measurement waveform/external trigger
ch.0
Lower 16 bits timer/counter
Lower 16 bits reload value
T32=1
Interrupt
Waveform output

5. Base Timer Interrupt

This section provides a list of interrupt request flags, interrupt enable bits, and interrupt causes for each function of the base timer.

■ Interrupt control bits and interrupt causes for each function

Table 5-1 shows the interrupt control bits and interrupt causes for each function.

Table 5-1 Interrupt control bits and interrupt causes in each mode

	Status Control Register (STC)			
	Interrupt request flag bit	Interrupt request enable bit	Interrupt factors	Interrupt factor output signal
PWM timer function (16-bit PWM timer)	UDIR: bit0	UDIE: bit4	Detection of an underflow	IRQ0
	DTIR: bit1	DTIE: bit5	Detection of a match in duty	
	TGIR: bit2	TGIE: bit6	Detection of a timer start trigger	IRQ1
PPG timer function (16-bit PPG timer)	UDIR: bit0	UDIE: bit4	Detection of an underflow	IRQ0
	TGIR: bit2	TGIE: bit6	Detection of a timer start trigger	IRQ1
Reload timer function (16-/32-bit Reload timer)	UDIR: bit0	UDIE: bit4	Detection of an underflow	IRQ0
	TGIR: bit2	TGIE: bit6	Detection of a timer start trigger	IRQ1
PWC timer function (16-/32-bit PWC timer)	OVIR: bit0	OVIE: bit4	Detection of an overflow	IRQ0
	EDIR: bit2	EDIE: bit6	Detection of the completion of measurement	IRQ1

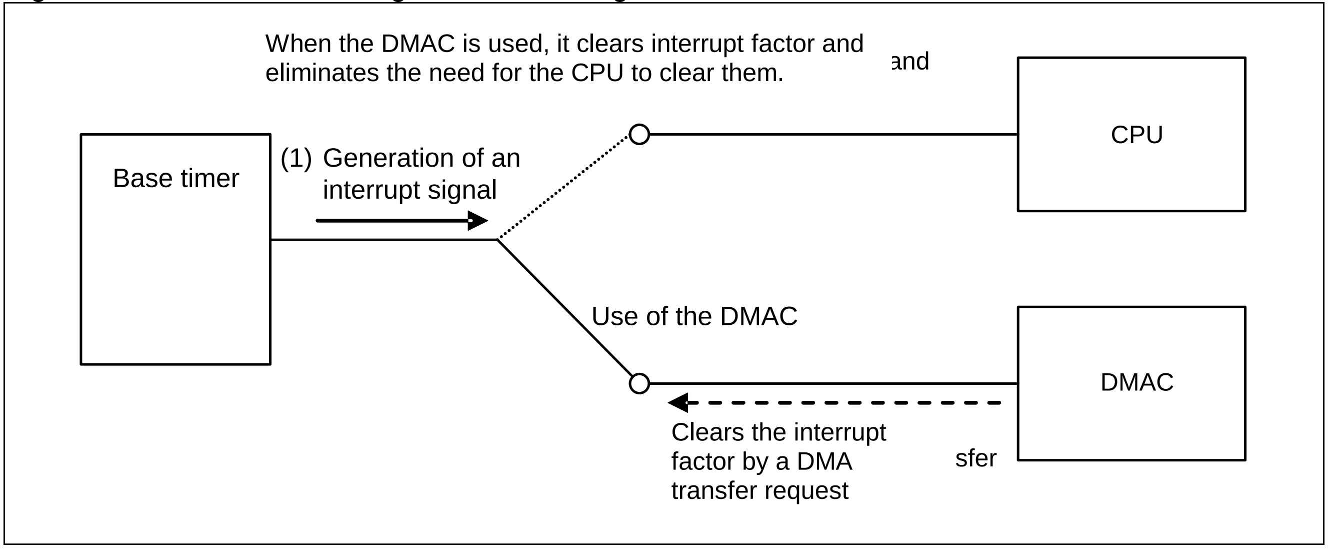
6. Starting the DMA Controller (DMAC)

The DMAC can be started using the generation of an interrupt factors by the base timer.

■ DMA transfer operation using interrupt factors of the base timer

The DMAC can be started using the generation of an interrupt factor by the base timer. Figure 6-1 gives an overview of starting the DMAC using the base timer.

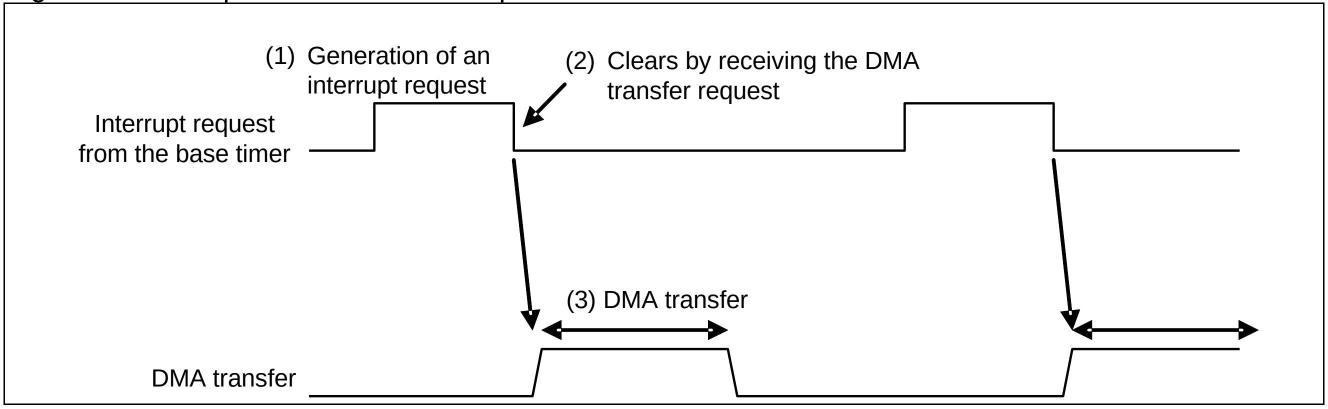
Figure 6-1 Overview of starting the DMAC using the base timer


When the DMAC is used, it clears interrupt factor and eliminates the need for the CPU to clear them.
and
Base timer
(1) Generation of an interrupt signal
CPU
Use of the DMAC
DMAC
Clears the interrupt factor by a DMA transfer request
sfer

Before starting the DMAC using the base timer, configure the DMAC. For settings details on the DMAC, see Chapters "DMAC" and "Interrupt" in "Peripheral Manual".
For “Interrupt” Chapter, select the appropriate chapter from “Interrupt (A)”, “Interrupt (B)”, and “Interrupt (C)” depending on the product used.

Figure 6-2 gives an example of a DMA transfer operation using an interrupt request by the base timer.

Figure 6-2 Example of DMA transfer operation


(1) Generation of an interrupt request
(2) Clears by receiving the DMA transfer request
Interrupt request from the base timer
(3) DMA transfer
DMA transfer

7. Registers of Base timer

This section provides register lists of the base timer in each mode.

■ List of registers used when the 16-bit PWM timer is selected

Table 7-1 List of registers used when the 16-bit PWM timer is selected

Abbreviation	Register name	Reference
TMCR	Timer Control Register	9.1.6
TMCR2	Timer Control Register 2	9.1.6
STC	Status Control Register	9.1.6
PCSR	PWM Cycle Set Register	9.1.7
PDUT	PWM Duty Set Register	9.1.8
TMR	Timer Register	9.1.9

■ List of registers used when the 16-bit PPG timer is selected

Table 7-2 List of registers used when the 16-bit PPG timer is selected

Abbreviation	Register name	Reference
TMCR	Timer Control Register	9.2.6
TMCR2	Timer Control Register 2	9.2.6
STC	Status Control Register	9.2.6
PRLL	LOW Width Reload Register	9.2.7
PRLH	HIGH Width Reload Register	9.2.8
TMR	Timer Register	9.2.9

■ List of registers used when the reload timer is selected

Table 7-3 List of registers used when the reload timer is selected

Abbreviation	Register name	Reference
TMCR	Timer Control Register	9.3.3
TMCR2	Timer Control Register 2	9.3.3
STC	Status Control Register	9.3.3
PCSR	PWM Cycle Set Register	9.3.4
TMR	Timer Register	9.3.5

■ List of registers used when the PWC timer is selected

Table 7-4 List of registers used when the PWC timer is selected

Abbreviation	Register name	Reference
TMCR	Timer Control Register	9.4.2
TMCR2	Timer Control Register 2	9.4.2
STC	Status Control Register	9.4.2
DTBF	Data Buffer Register	9.4.3

8. Notes on using the base timer

This section provides notes on using the base timer.

■ Notes on setting the program common to each timer

- It is prohibited to rewrite the following bits in the TMCR2 and TMCR registers during operation. Rewriting of the bits must be performed before starting or after stopping the operation.

[TMCR2 bit8], [TMCR bit14:12]	CKS3 to CKS0 : Clock selection bits
[bit10:8]	EGS2, EGS1, EGS0 : Measurement edge selection bits
[bit7]	T32 : 32-bit timer selection bit (When the reload timer PWC function is selected)
[bit6:4]	FMD[2:0] : Timer function selection bits
[bit2]	MDSE : Measurement mode (one-shot/continuous) selection bit

- When the FMD[2:0] bits in the TMCR register are set to reset mode with "0b000", all registers of the base timer are initialized. Therefore, all registers must be set again.
- When the FMD[2:0] bits in the TMCR register are set to reset mode with "0b000", settings for the bits other than the FMD[2:0] bits in the TMCR register are ignored and initialized.

■ Notes on using the 16-bit PWM/PPG/reload timer

- If the interrupt request flag set timing coincides the clear timing, the flag set operation takes precedence and the clear operation is not performed.
- If the load timing and count timing of the down counter coincide, the load operation takes precedence.
- Set the timer function with the FMD[2:0] bits in the TMCR register, and then set the cycle, duty, HIGH width, and LOW width.
- In one-shot mode, if a restart is detected at the end of counting, the count value is reloaded and the restart operation is started.

■ Notes on using the PWC timer

- If the count start enable bit (CTEN) is set to "1", the counter is cleared. As the result, the data existed in the counter before the start is enabled becomes invalid.
- If the setting for PWC mode (FMD[2:0] = 0b100) and the setting for starting measurement (CTEN = 1) are performed simultaneously in system reset/reset mode, the resultant operation may depend on the status of the last measurement signal.
- In continuous measurement mode, if a measurement start edge is detected at the same time a restart is set, the counting is started immediately from "0x0001".
- If a restart is performed after the count operation has been started, the following operations may occur depending on the timing.
  - If it coincides with a measurement end edge in pulse width one-shot measurement mode:
    The timer is restarted and waits for detection of a measurement start edge. However, a measurement end flag (EDIR) is set.

  - If it coincides with a measurement end edge in pulse width continuous measurement mode:
    The timer is restarted and waits for detection of a measurement start edge. However, a measurement end flag (EDIR) is set and the measurement result at the time is transferred to the DTBF.

When restarting the timer during operation, pay attention to flag operations as described above and use the interrupt control.

9. Descriptions of base timer functions

This section explains each function of the base timer.

■ Base timer functions

1. PWM timer function
2. PPG timer function
3. Reload timer function
4. PWC timer function

9.1. PWM timer function

The function of the base timer can be set to either the 16-bit PWM timer, 16-bit PPG timer, 16/32-bit reload timer, or 16/32-bit PWC timer using the FMD[2:0] bits in the Timer Control Register. This section explains the timer functions available when PWM is set.

1. 16-bit PWM timer operations
2. One-shot operation
3. Interrupt factors and timing chart
4. Output waveforms
5. PWM timer operation flowchart
6. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PWM timer is selected
7. PWM Cycle Set Register (PCSR)
8. PWM Duty Set Register (PDUT)
9. Timer Register (TMR)

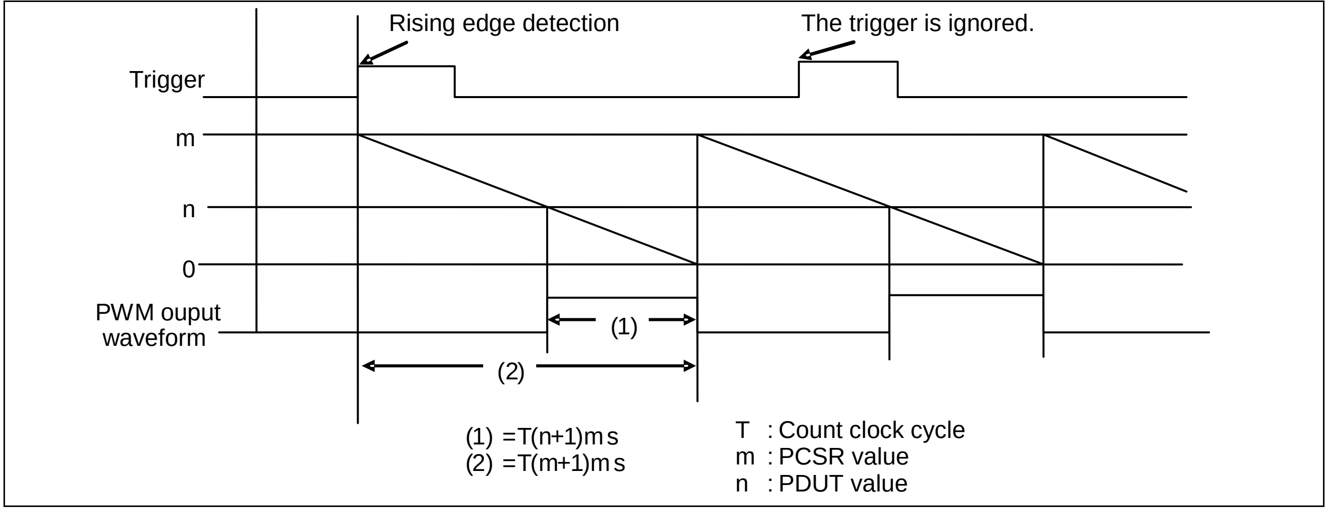
9.1.1. 16-bit PWM timer operations

In PWM timer operations, waveforms in the specified cycle from the detection of a trigger can be output in one-shot or continuously. The cycle of the output pulse can be controlled by changing the PCSR value. The duty ratio can be controlled by changing the PDUT value. After writing data to the PCSR, be sure to write it to the PDUT.

■ Continuous operation

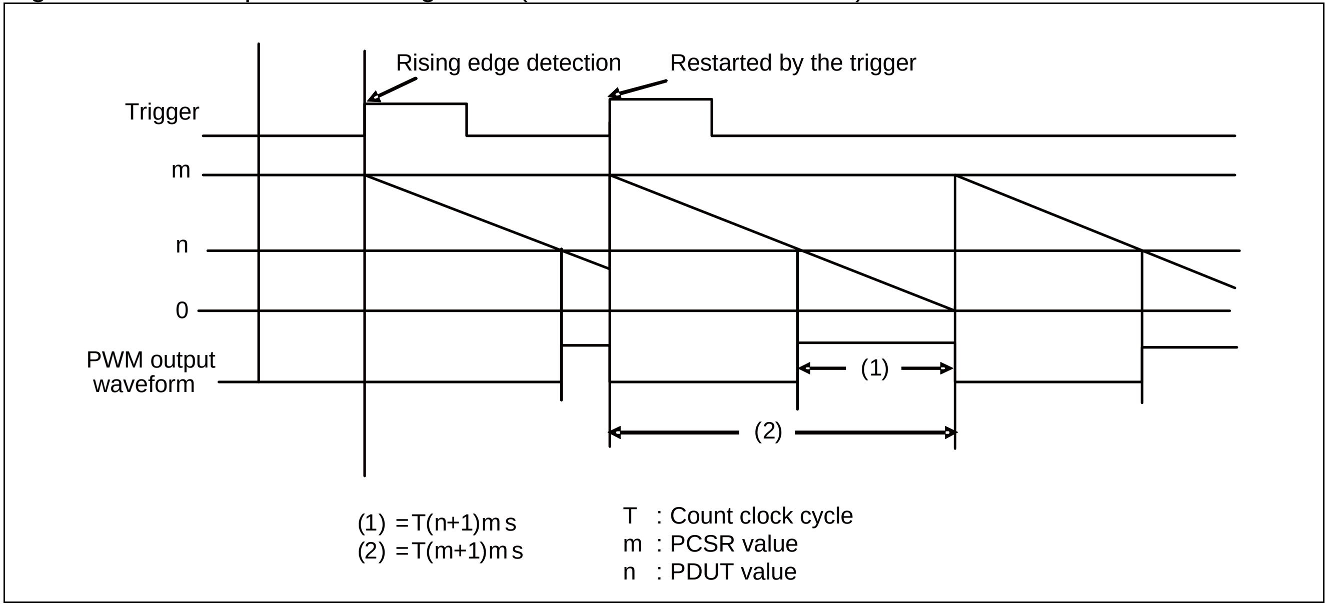
● When a restart is disabled (RTGEN = 0)

Figure 9-1 PWM operation timing chart (when a restart is disabled)



● When a restart is enabled (RTGEN = 1)

Figure 9-2 PWM operation timing chart (when a restart is enabled)



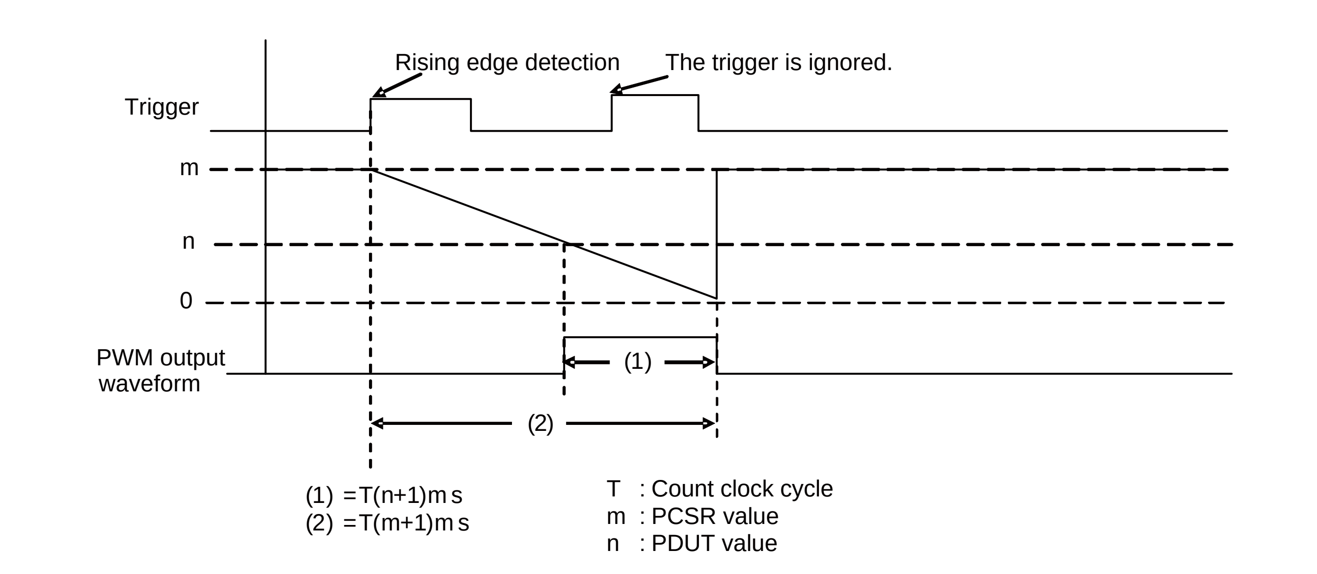
9.1.2. One-shot operation

In one-shot operation, a single pulse of any width can be output using a trigger. When a restart is enabled, the counter is reloaded when an edge is detected during operation.

■ One-shot operation

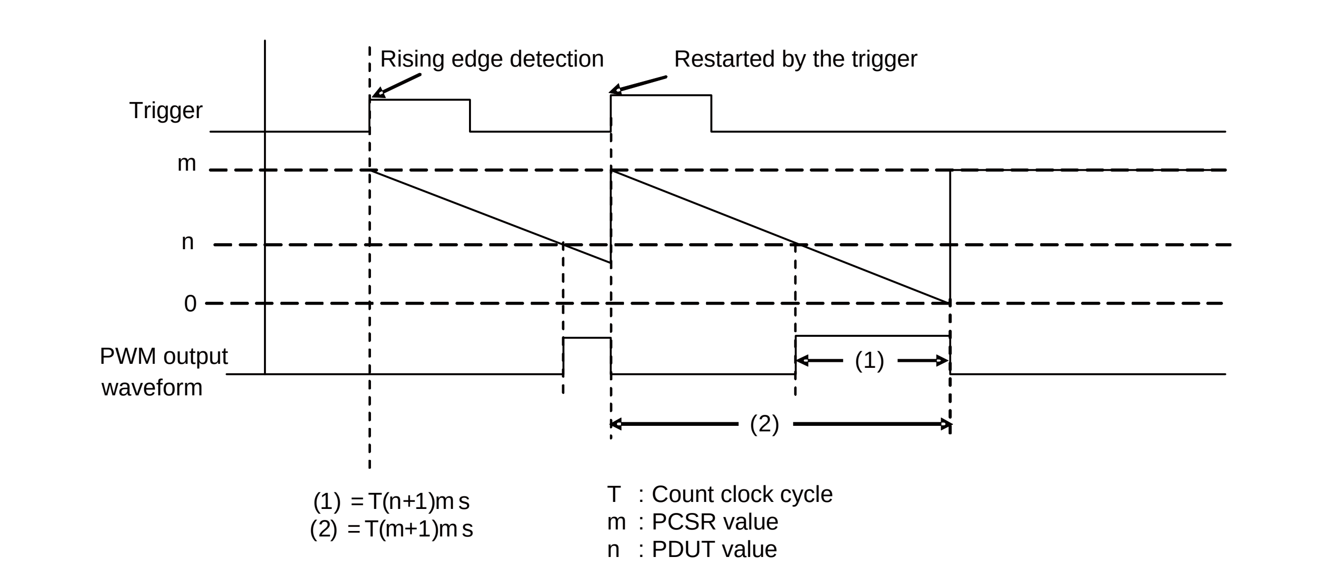
● When a restart is disabled (RTGEN = 0)

Figure 9-3 One-shot operation timing chart (trigger restart is disabled)



● When a restart is enabled (RTGEN = 1)

Figure 9-4 One-shot operation timing chart (trigger restart is enabled)



9.1.3. Interrupt factors and timing chart

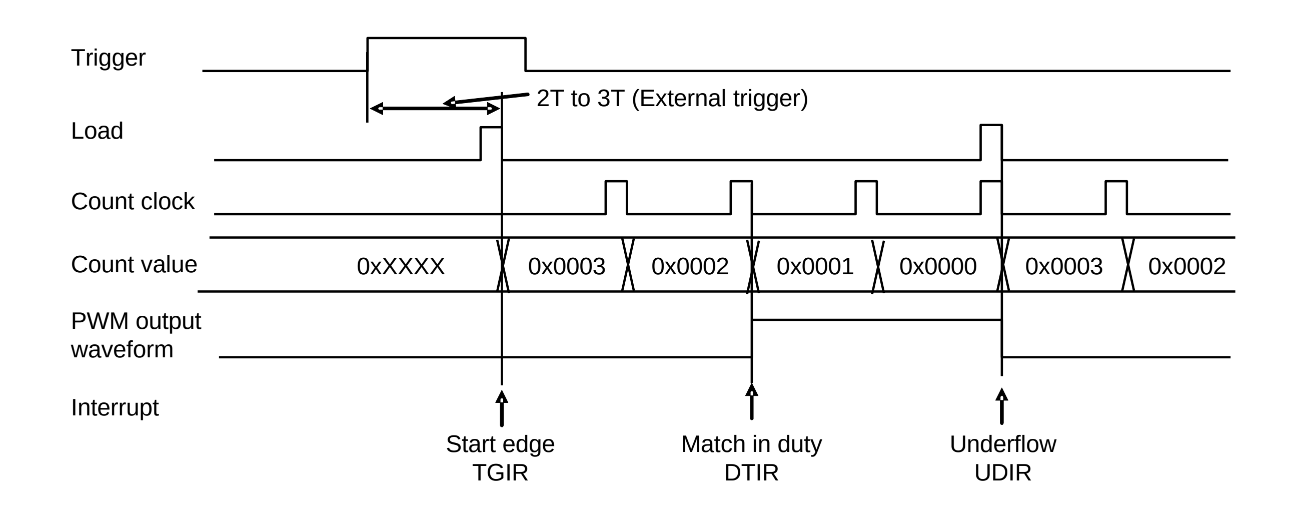
This section explains interrupt factors and a timing chart.

■ Interrupt factors and timing chart (PWM output: Normal polarity)

As a time from trigger input to loading of the counter value, T is required for software triggering or 2T to 3T (T: machine cycle) for external triggering.

Figure 9-5 shows the interrupt factors and a timing chart where the cycle set value = 3 and duty value = 1.

Figure 9-5 Interrupt factors and timing chart of the PWM timer



9.1.4. Output waveforms

This section explains the PWM output.

■ How to make an all-LOW or all-HIGH PWM output

Figure 9-6 shows how to make all-LOW PWM output and Figure 9-7 shows how to make all-HIGH output.

Figure 9-6 Example of outputting all-LOW level waveforms as PWM output

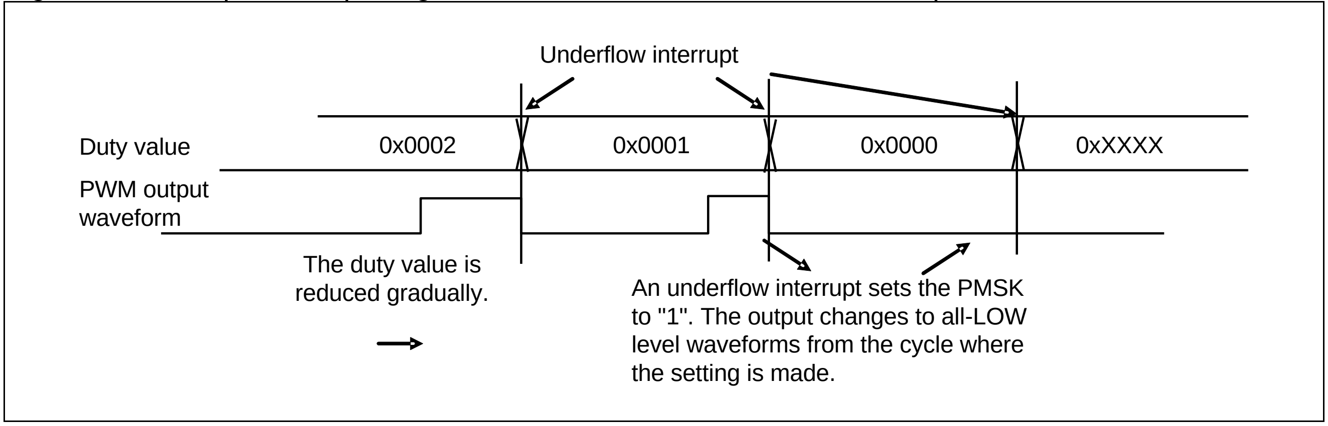
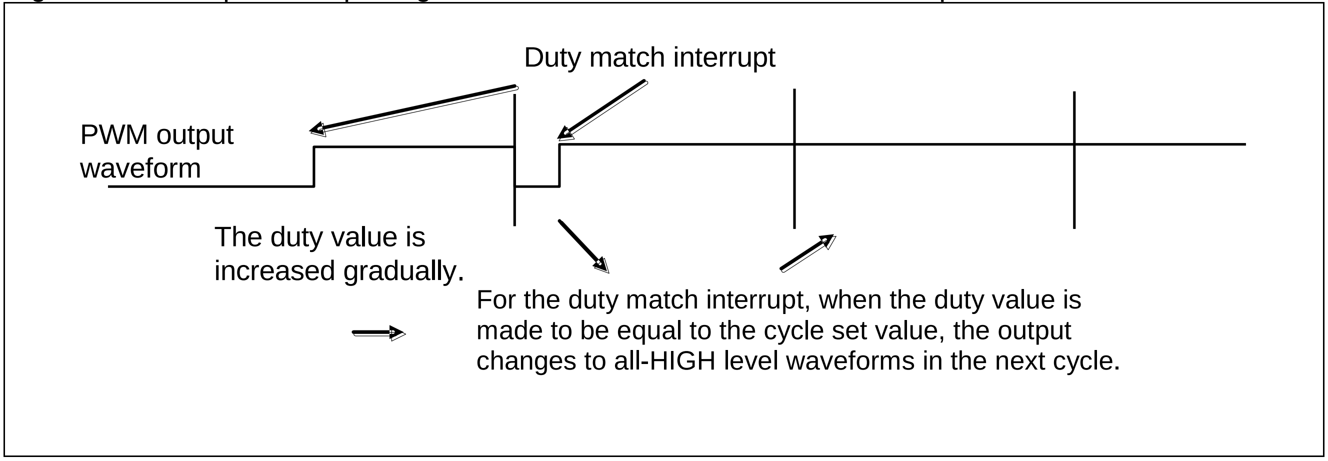


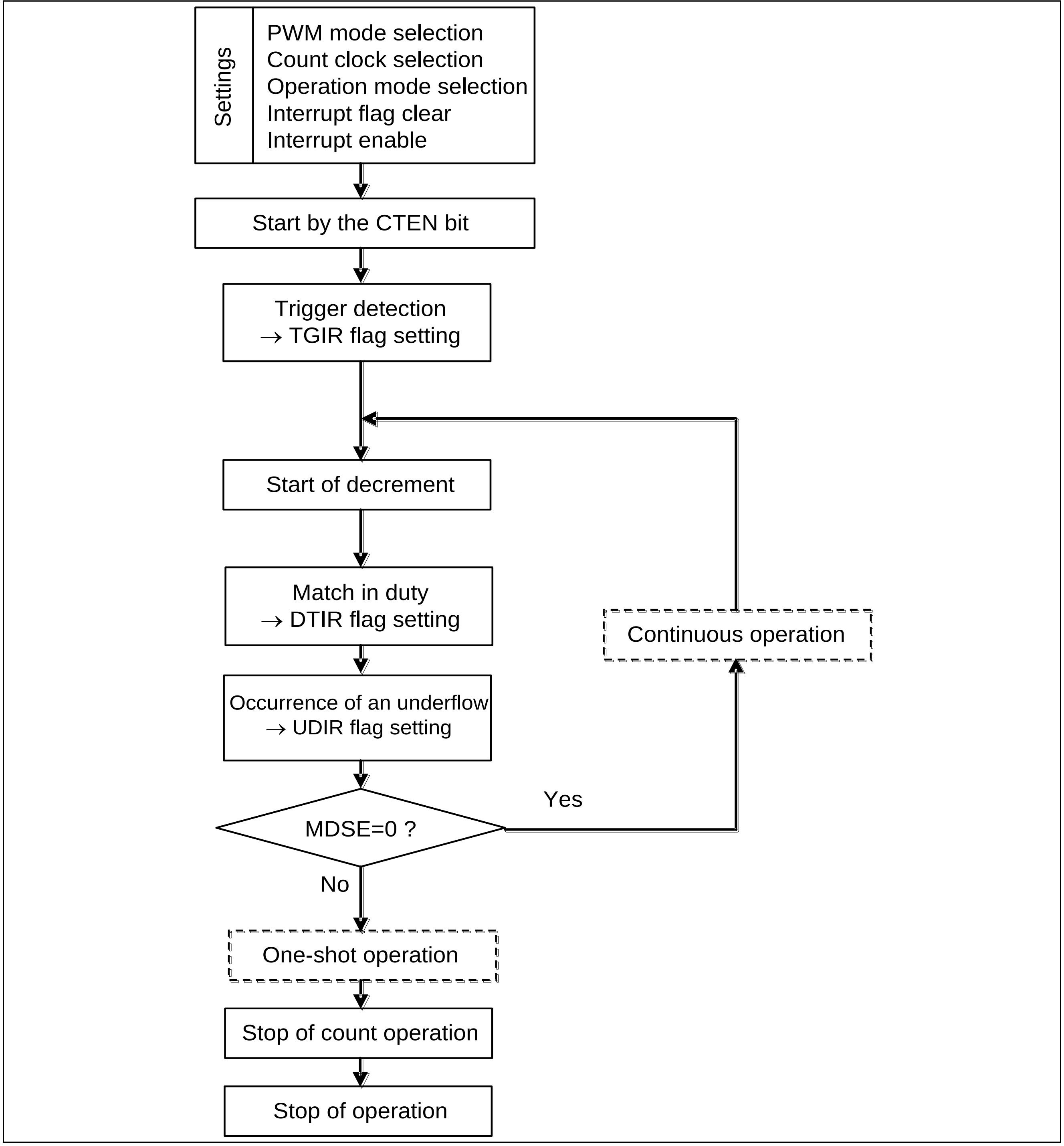
Figure 9-7 Example of outputting all-HIGH level waveforms as PWM output


Duty match interrupt
PWM output
waveform
The duty value is
increased gradually.
For the duty match interrupt, when the duty value is
made to be equal to the cycle set value, the output
changes to all-HIGH level waveforms in the next cycle.

9.1.5. PWM timer operation flowchart

This section provides an operation flowchart of the PWM timer.

■ PWM timer operation flowchart



9.1.6. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PWM timer is selected

The Timer Control Register (TMCR) controls the PWM timer. Note that some bits cannot be rewritten while the PWM timer is in operation.

■ Timer Control Register (Upper bytes of TMCR)

bit	15	14	13	12	11	10	9	8
Field	Reserved	CKS2	CKS1	CKS0	RTGEN	PMSK	EGS1	EGS0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit15] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit14:12, TMCR2:bit8] CKS3 to CKS0: Count clock selection bits

- Select the count clock for the 16-bit down counter.
- Changes to the count clock setting are applied immediately. For this reason, changes to CKS3 through CKS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the Count operation enable bit (CTEN).

CKS3	CKS2	CKS1	CKS0	Description
0	0	0	0	ϕ
0	0	0	1	ϕ /4
0	0	1	0	ϕ /16
0	0	1	1	ϕ /128
0	1	0	0	ϕ /256
0	1	0	1	External clock (rising edge event)
0	1	1	0	External clock (falling edge event)
0	1	1	1	External clock (both edge event)
1	0	0	0	ϕ /512
1	0	0	1	ϕ /1024
1	0	1	0	ϕ /2048
Values other than the above				Setting is prohibited.

[bit11] RTGEN: Restart enable bit

This bit enables restart by a software trigger or trigger input.

Value	Description
0	Restart disabled
1	Restart enabled

[bit10] PMSK: Pulse output mask bit

- This bit controls the output level of PWM output waveforms.
- When this bit is set to "0", PWM waveforms are output as they are.
- When this bit is set to "1", the PWM output is masked with LOW output regardless of the cycle and duty set values.
-

Value	Description
0	Normal output
1	Fixed to LOW output

<Note>

When Output polarity specification bit (OSEL) of Timer Control Register (Lower bytes of TMCR) is set to inverted output, setting PMSK bit to "1" masks the output with HIGH.

[bit9:8] EGS1, EGS0: Trigger input edge selection bits

- These bits select a valid edge for input waveforms as an external start cause and set the trigger condition.
- When the initial value or "0b00" is set, the timer is not started by external waveforms because the setting means that no valid edge is selected for input waveforms.
- Changes to EGS1 or EGS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit9	bit8	Description
0	0	Trigger input disabled
0	1	Rising edge
1	0	Falling edge
1	1	Both edges

<Note>

If the STRG bit is set to "1", software triggering is enabled regardless of the EGS1 and EGS0 settings.

■ Timer Control Register (Lower bytes of TMCR)

bit	7	6	5	4	3	2	1	0
Field	Reserved	FMD2	FMD1	FMD0	OSEL	MDSE	CTEN	STRG
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit6:4] FMD2 to FMD0: Timer function selection bits

- These bits select the timer function.
- When the FMD[2:0] bits are set to "0b001", the PWM function is selected.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit6	bit5	bit4	Description
0	0	0	Reset mode
0	0	1	16-bit PWM timer
0	1	0	16-bit PPG timer
0	1	1	16/32-bit reload timer
1	0	0	16/32-bit PWC timer
1	0	1	Setting is prohibited.
1	1	0	
1	1	1	

[bit3] OSEL: Output polarity specification bit

This bit sets the polarity of the PWM output.

Polarity	After reset	Match in duty	Underflow
Normal	LOW output	↑	↓
Inverted	HIGH output	↓	↑

Value	Description
0	Normal polarity
1	Inverted polarity

[bit2] MDSE: Mode selection bit

- This bit selects continuous pulse output or one-shot pulse output.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

Value	Description
0	Continuous operation
1	One-shot operation

[bit1] CTEN: Count operation enable bit

- This bit enables the operation of the down counter.
- When the counter is in operation enabled status (the CTEN bit is "1"), writing "0" to this bit stops the counter.

Value	Description
0	Stop
1	Operation enabled

<Note>

By writing "1" to CNTEN bit, the output waveform becomes Low.

[bit0] STRG: Software trigger bit

- When the CTEN bit is "1", writing "1" to the STRG bit enables software triggering.
- The read value of the STRG bit is always "0".

Value	Description
0	Invalid
1	Start triggered by software

<Notes>

- Software triggering is also enabled when "1" is written to the CTEN and STRG bits simultaneously.
- If the STRG bit is set to "1", software triggering is enabled regardless of the EGS1 and EGS0 settings.

■ Timer Control Register 2 (TMCR2)

bit	15	14	13	12	11	10	9	8
Field	Reserved							CKS3
Attribute	R/W							R/W
Initial value	0000000							0

Note: This register is placed above the STC register.

[bit15:9] Reserved: Reserved bits

The read value is "0".

Set "0" to these bits.

[bit8] CKS3: Count clock selection bit

See "Count clock selection bit" in "9.1.6 Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PWM timer is selected".

■ Status Control Register (STC)

bit	7	6	5	4	3	2	1	0
Field	Reserved	TGIE	DTIE	UDIE	Reserved	TGIR	DTIR	UDIR
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

Note: The TMCR2 register is placed in the upper bytes of this register.

[bit7] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit6] TGIE: Trigger interrupt request enable bit

- This bit controls interrupt requests of bit2 TGIR.
- When the TGIE bit is enabled, setting bit2 TGIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit5] DTIE: Duty match interrupt request enable bit

- This bit controls interrupt requests of bit1 DTIR.
- When the DTIE bit is enabled, setting bit1 DTIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit4] UDIE: Underflow interrupt request enable bit

- This bit controls interrupt requests of bit0 UDIR.
- When the UDIE bit is enabled, setting bit0 UDIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit3] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit2] TGIR: Trigger interrupt request bit

- When a software trigger or trigger input is detected, the TGIR bit is set to "1".
- The TGIR bit is cleared by writing "0".
- Even if "1" is written to the TGIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt cause.
1	Detects an interrupt cause.

[bit1] DTIR: Duty match interrupt request bit

- When the count value matches the duty set value, the DTIR bit is set to "1".
- The DTIR bit is cleared by writing "0".
- Even if "1" is written to the DTIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt cause.
1	Detects an interrupt cause.

[bit0] UDIR: Underflow interrupt request bit

- When a count value underflow from 0x0000 to 0xFFFF occurs, the UDIR bit is set to "1".
- The UDIR bit is cleared by writing "0".
- Even if "1" is written to the UDIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt cause.
1	Detects an interrupt cause.

9.1.7. PWM Cycle Set Register (PCSR)

The PWM Cycle Set Register (PCSR) is a buffered register for setting the cycle. Transfer to the Timer Register is performed at startup and underflow.

bit	15	0
Field	PCSR [15:0]	
Attribute	R/W	
Initial value	0xXXXX	

This is a buffered register for setting the cycle. Transfer to the Timer Register is performed at startup and underflow.

When initializing or rewriting the PWM Cycle Set Register, be sure to perform writing to the PWM Duty Set Register after performing writing to the PWM Cycle Set Register.

- Do not access the PCSR register with 8-bit data.
- Set the cycle for the PCSR register after setting the PWM function using the FMD[2:0] bits in the TMCR register.

9.1.8. PWM Duty Set Register (PDUT)

The PWM Duty Set Register (PDUT) is a buffered register for setting the duty. Transfer from the buffer is performed at an underflow.

bit	15	0
Field	PDUT[15:0]	
Attribute	R/W	
Initial value	0xXXXX	

This is a buffered register for setting the duty. Transfer from the buffer is performed at an underflow.

When the cycle set register value is set equal to the duty set register value, an all-HIGH pulse is output under normal polarity and an all-LOW pulse is output under inverted polarity.

Do not set a value that makes PCSR < PDUT. The PWM output becomes undefined.

- Do not access the PDUT register with 8-bit data.
- Set the duty for the PDUT register after setting the PWM function using the FMD[2:0] bits in the TMCR register.

9.1.9. Timer Register (TMR)

The Timer Register (TMR) reads the value of the 16-bit down counter.

bit	15	0
Field	TMR[15:0]	
Attribute	R	
Initial value	0x0000	

The value of the 16-bit down counter is read.

- Do not access the TMR register with 8-bit data.

9.2. PPG timer function

The function of the base timer can be set to either the 16-bit PWM timer, 16-bit PPG timer, 16-/32-bit reload timer, or 16-/32-bit PWC timer using the FMD[2:0] bits in the Timer Control Register. This section explains the timer functions available when PPG is set.

1. 16-bit PPG timer operations
2. Continuous operation
3. One-shot operation
4. Interrupt factors and timing chart
5. PPG timer operation flowchart
6. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PPG timer is selected
7. LOW Width Reload Register (PRLL)
8. HIGH Width Reload Register (PRLH)
9. Timer Register (TMR)

9.2.1. 16-bit PPG timer operations

In PPG timer operations, any output pulse can be controlled by setting the LOW and HIGH widths of the pulse in respective reload registers.

■ Overview of operations

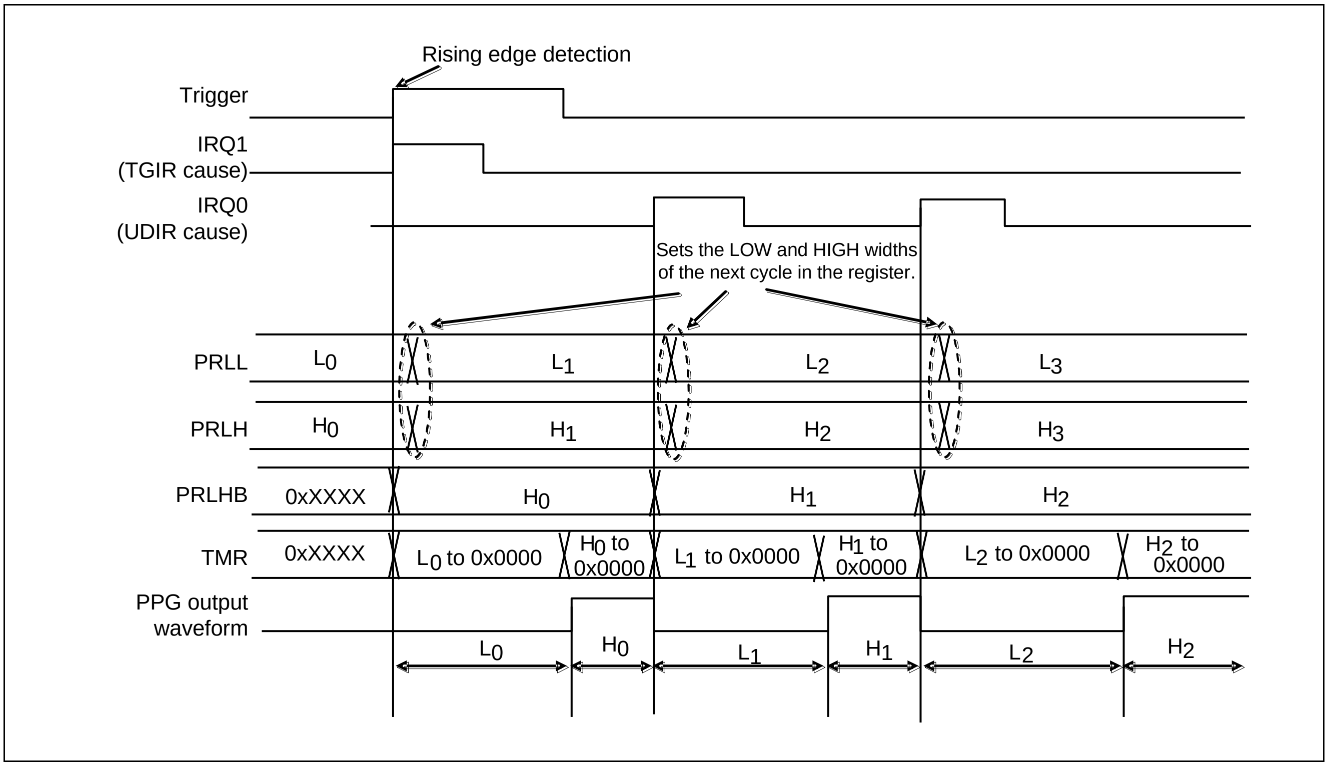
Two 16-bit reload registers for setting the LOW and HIGH widths, respectively, and one buffer for setting the HIGH width are used (PRLL, PRLH, and PRLHB).

A start trigger initially causes the PRLL set value to be loaded to the 16-bit down counter and, at the same time, the PRLH set value to be transferred to the PRLHB. The PPG timer changes the output level to LOW and counts down for every count clock. Upon detection of an underflow, the PPG timer reloads the PRLHB value to the counter, inverts the PPG output waveforms, and continues to count down. At the next detection of an underflow, it inverts the PPG output waveforms, reloads the PRLL set value to the counter, and transfers the PRLH set value to the PRLHB.

This operation causes the output waveform to be pulse output having LOW and HIGH widths corresponding to the values in the respective reload registers.

■ Timing of writing to the reload registers

Data writing to the PRLL and PRLH reload registers occurs upon detection of a start trigger and during the period from when an underflow interrupt cause (UDIR) is set to when the next cycle starts. The data set here is used as the setting for the next cycle. The items of data set in the PRLL and PRLH are automatically transferred to the TMR and PRLHB, respectively, when a start trigger is detected and when an underflow occurs at the completion of HIGH width counting. The data transferred to the PRLHB is automatically reloaded to the TMR when an underflow occurs at the completion of LOW width counting.



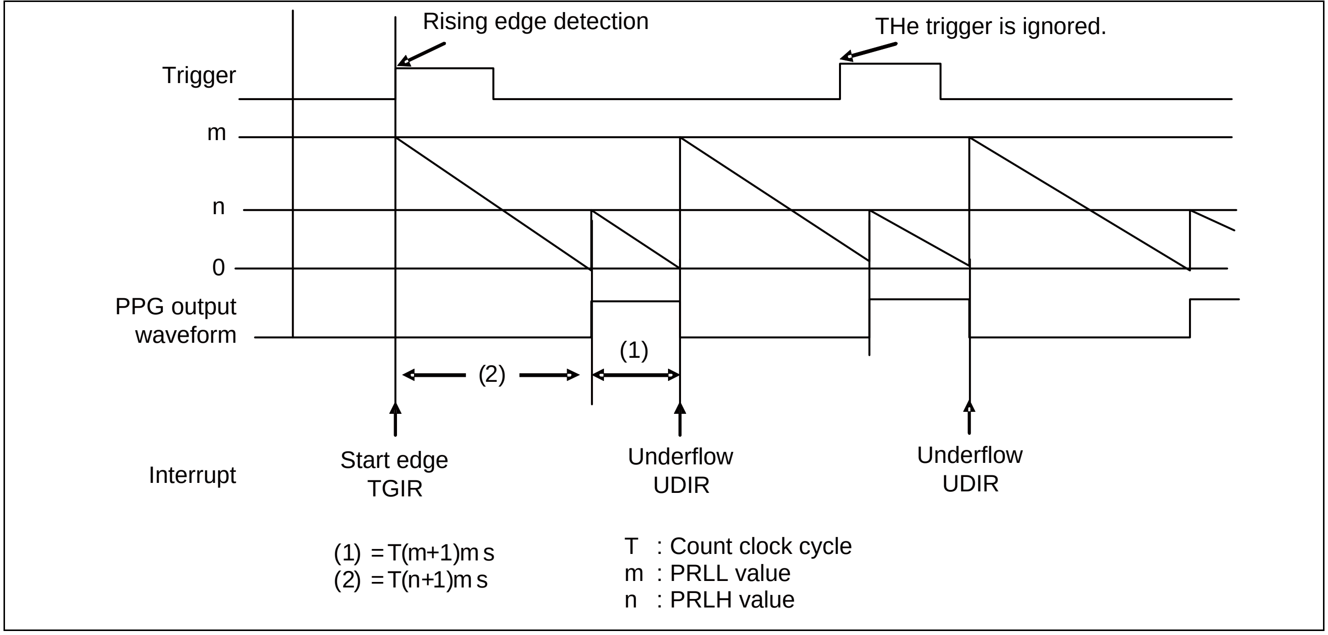
9.2.2. Continuous operation

In continuous operations, any pulse can be output continuously by updating the LOW and HIGH widths at the set timing of each interrupt cause. When a restart is enabled, the counter is reloaded when an edge is detected during operation.

■ Continuous operation

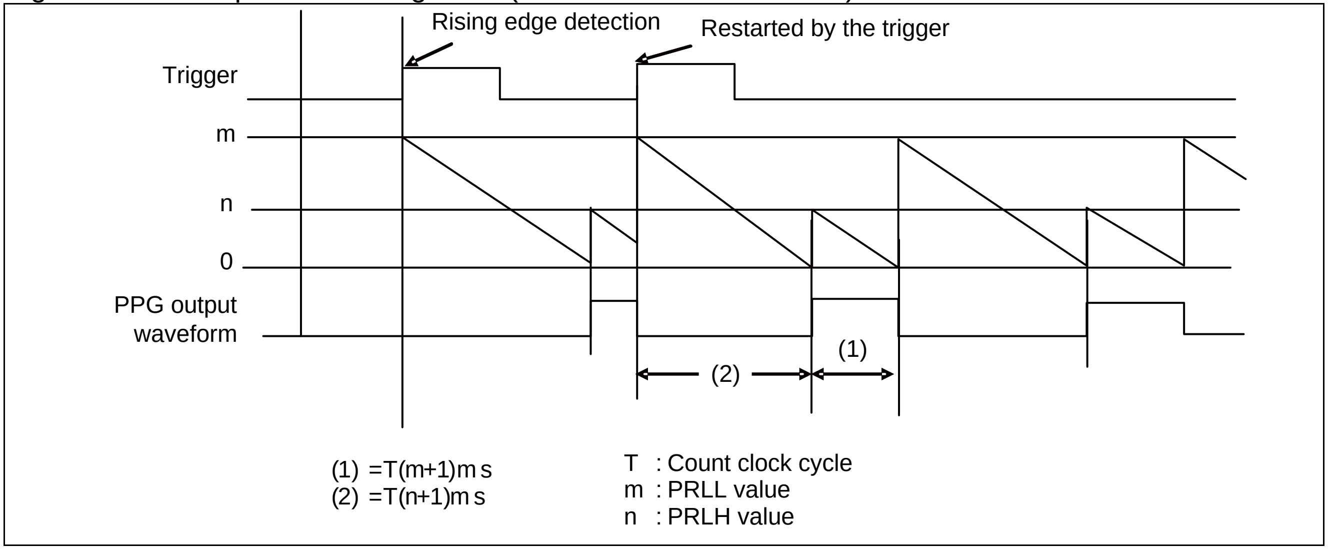
● When a restart is disabled (RTGEN = 0)

Figure 9-8 PPG operation timing chart (when a restart is disabled)



● When a restart is enabled (RTGEN = 1)

Figure 9-9 PPG operation timing chart (when a restart is enabled)



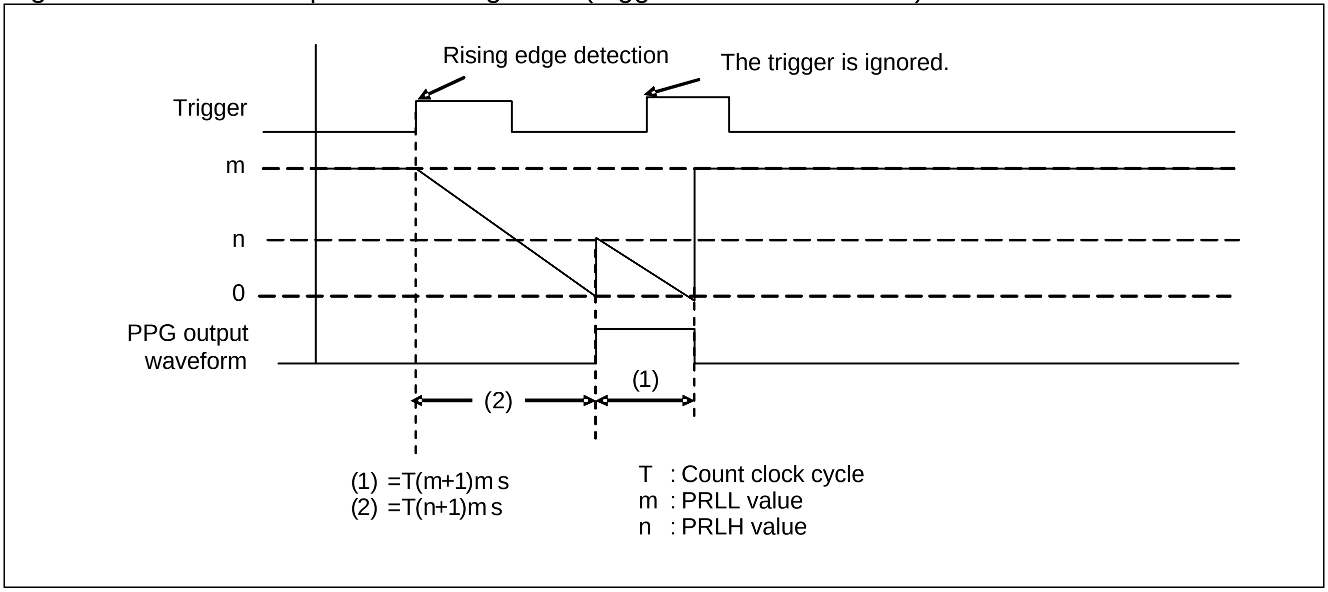
9.2.3. One-shot operation

In one-shot operation, a single pulse of any width can be output using a trigger. When a restart is enabled, the counter is reloaded when an edge is detected during operation.

■ One-shot operation

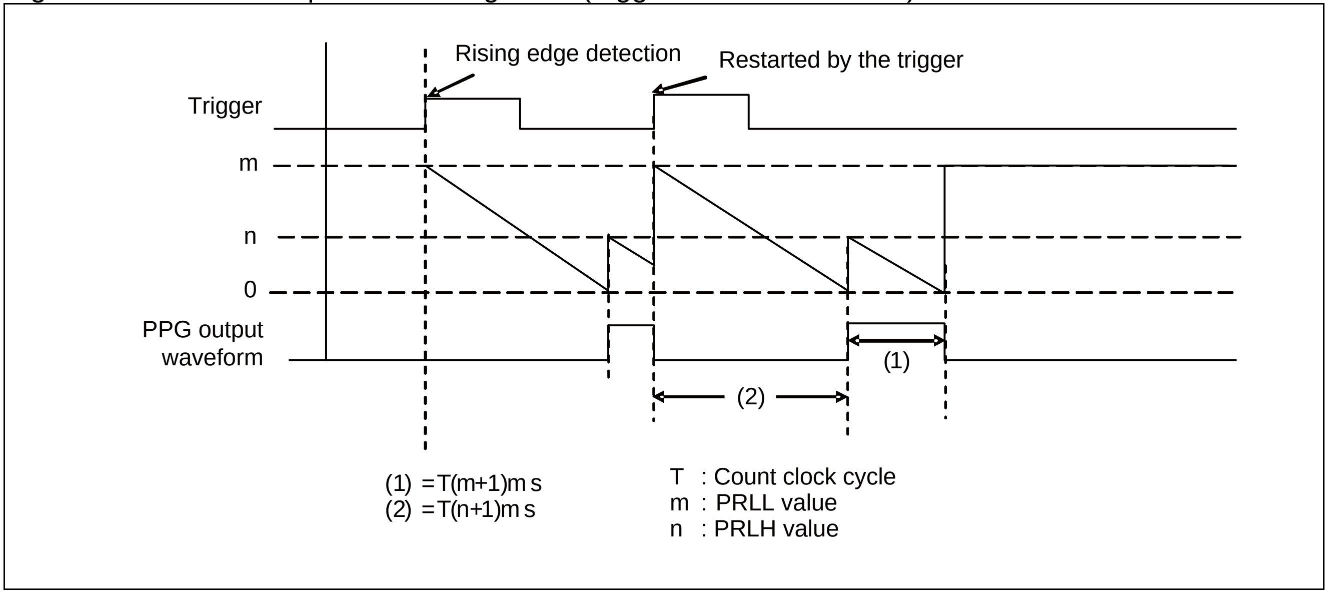
● When a restart is disabled (RTGEN = 0)

Figure 9-10 One-shot operation timing chart (trigger restart is disabled)



● When a restart is enabled (RTGEN = 1)

Figure 9-11 One-shot operation timing chart (trigger restart is enabled)



■ Relation between reload value and pulse width

The output pulse width is equal to the 16-bit reload register value added by 1, and which is multiplied by the count clock cycle. Therefore, when the reload register value is "0x0000", the pulse width is equal to one count clock cycle. When the reload register value is "0xFFFF", the pulse width is equal to 65536 count clock cycle. The pulse width calculation formulas are as follows:

$$PL = T \times (L + 1)$$
$$PH = T \times (H + 1)$$

PL	:	Width of LOW pulse
PH	:	Width of HIGH pulse
T	:	Count clock cycle
L	:	PRLL value
H	:	PRLH value

9.2.4. Interrupt factors and timing chart

This section explains interrupt factors and a timing chart.

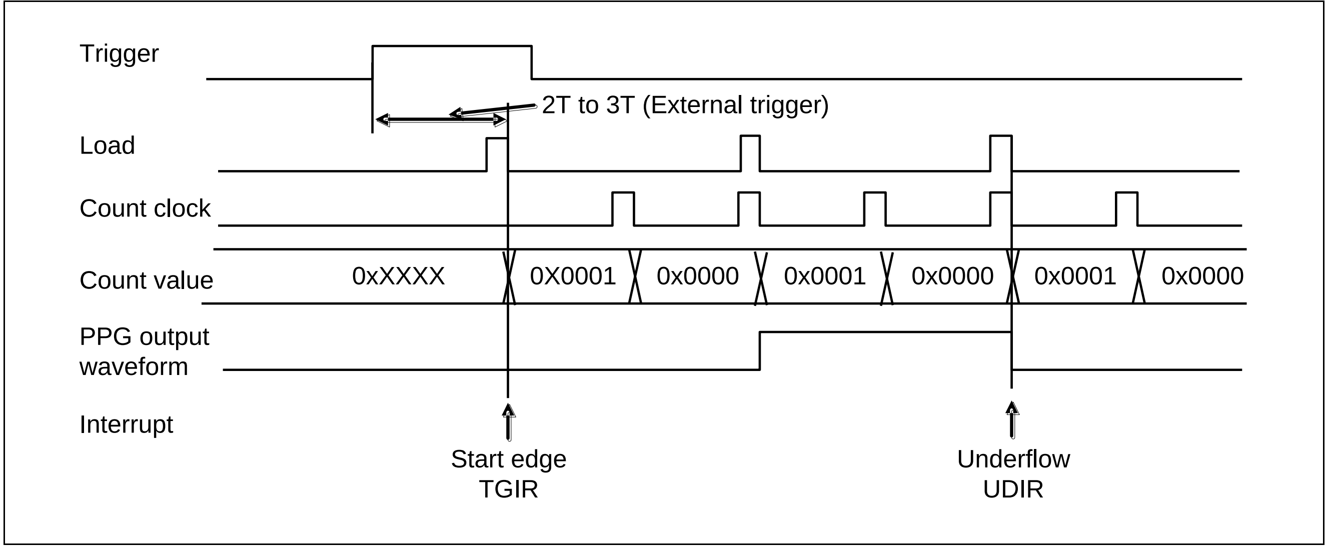
■ Interrupt factors and timing chart (PPG output: Normal polarity)

As a time from trigger input to loading of the counter value, T is required for software triggering or 2T to 3T (T: machine cycle) for external triggering.

Interrupt factors are set to detection of a PPG start trigger and an underflow in HIGH level output.

Figure 9-12 shows the interrupt factors and a timing chart where LOW width set value = 1 and HIGH width set value = 1.

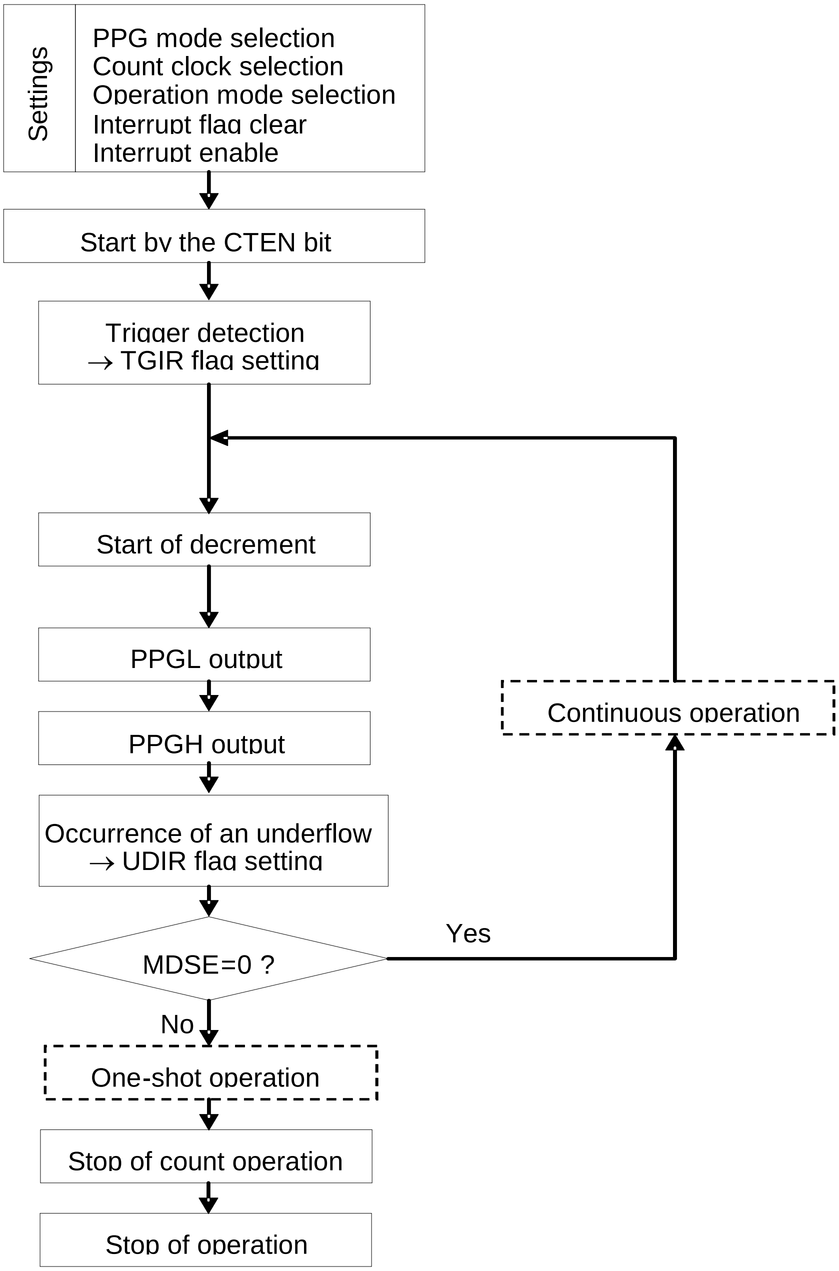
Figure 9-12 Interrupt factors and timing chart of the PPG timer



9.2.5. PPG timer operation flowchart

This section provides an operation flowchart of the PPG timer.

■ PPG timer operation flowchart



9.2.6. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PPG timer is selected

The Timer Control Register (TMCR) controls the PPG timer. Note that some bits cannot be rewritten while the PPG timer is in operation.

■ Timer Control Register (Upper bytes of TMCR)

bit	15	14	13	12	11	10	9	8
Field	Reserved	CKS2	CKS1	CKS0	RTGEN	PMSK	EGS1	EGS0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit15] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit14:12, TMCR2: bit 8] CKS3 to CKS0: Count clock selection bits

- Select the count clock for the 16-bit down counter.
- Changes to the count clock setting are applied immediately. For this reason, changes to CKS3 through CKS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

CKS3	CKS2	CKS1	CKS0	Description
0	0	0	0	ϕ
0	0	0	1	ϕ /4
0	0	1	0	ϕ /16
0	0	1	1	ϕ /128
0	1	0	0	ϕ /256
0	1	0	1	External clock (rising edge event)
0	1	1	0	External clock (falling edge event)
0	1	1	1	External clock (both edge event)
1	0	0	0	ϕ /512
1	0	0	1	ϕ /1024
1	0	1	0	ϕ /2048
Others				Setting is prohibited.

[bit11] RTGEN: Restart enable bit

This bit enables restart by a software trigger or trigger input.

Value	Description
0	Restart disabled
1	Restart enabled

[bit10] PMSK: Pulse output mask bit

- This bit controls the output level of PPG output waveforms.
- When this bit is set to "0", PPG waveforms are output as they are.
- When this bit is set to "1", the PPG output is masked with LOW output regardless of the cycle and duty set values.

Value	Description
0	Normal output
1	Fixed to LOW output

<Note>

When OSEL in bit3 is set to inverted output, setting PMSK to "1" masks the output with HIGH.

[bit9:8] EGS1, EGS0: Trigger input edge selection bits

- These bits select a valid edge for input waveforms as an external start cause and set the trigger condition.
- When the initial value or "0b00" is set, the timer is not started by external waveforms because the setting means that no valid edge is selected for input waveforms.
- Changes to EGS1 or EGS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit9	bit8	Description
0	0	Trigger input disabled
0	1	Rising edge
1	0	Falling edge
1	1	Both edges

<Note>

If the STRG bit is set to "1", software triggering is enabled regardless of the EGS1 and EGS0 settings.

■ Timer Control Register (Lower bytes of TMCR)

bit	7	6	5	4	3	2	1	0
Field	Reserved	FMD2	FMD1	FMD0	OSEL	MDSE	CTEN	STRG
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

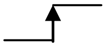
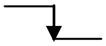
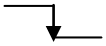
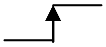
[bit6:4] FMD2 to FMD0: Timer function selection bits

- These bits select the timer function.
- When the FMD[2:0] bits are set to "0b010", the PPG function is selected.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit6	bit5	bit4	Description
0	0	0	Reset mode
0	0	1	16-bit PWM timer
0	1	0	16-bit PPG timer
0	1	1	16-/32-bit reload timer
1	0	0	16-/32-bit PWC timer
1	0	1	Setting is prohibited.
1	1	0	
1	1	1	

[bit3] OSEL: Output polarity specification bit

This bit sets the polarity of the PPG output.

Polarity	After reset	Completion of LOW width counting	Completion of HIGH width counting
Normal	LOW output		
Inverted	HIGH output		

Value	Description
0	Normal polarity
1	Inverted polarity

[bit2] MDSE: Mode selection bit

- This bit selects continuous pulse output or one-shot pulse output.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

Value	Description
0	Continuous operation
1	One-shot operation

[bit1] CTEN: Count operation enable bit

- This bit enables the operation of the down counter.
- When the counter is in operation enabled status (the CTEN bit is "1"), writing "0" to this bit stops the counter.

Value	Description
0	Stop
1	Operation enabled

<Note>

By writing "0" to CTEN, PPG output is set to Low.

[bit0] STRG: Software trigger bit

- When the CTEN bit is "1", writing "1" to the STRG bit enables software triggering.
- The read value of the STRG bit is always "0".

Value	Description
0	Invalid
1	Start triggered by software

<Notes>

- Software triggering is also enabled when "1" is written to the CTEN and STRG bits simultaneously.
- If the STRG bit is set to "1", software triggering is enabled regardless of the EGS1 and EGS0 settings.

■ Timer Control Register 2 (Upper bytes of TMCR2)

bit	15	14	13	12	11	10	9	8
Field	Reserved							CKS3
Attribute	R/W							R/W
Initial value	0000000							0

Note: This register is placed above the STC register.

[bit15:9] Reserved: Reserved bits

The read value is "0".

Set "0" to these bits.

[bit8] CKS3: Count clock selection bit

See "Count clock selection bit" in "9.2.6 Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PPG timer is selected".

■ Status Control Register (STC)

bit	7	6	5	4	3	2	1	0
Field	Reserved	TGIE	Reserved	UDIE	Reserved	TGIR	Reserved	UDIR
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

Note: The TMCR2 register is placed in the upper bytes of this register.

[bit7] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit6] TGIE: Trigger interrupt request enable bit

- This bit controls interrupt requests of bit2:TGIR.
- When the TGIE bit is enabled, setting bit2:TGIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit5] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit4] UDIE: Underflow interrupt request enable bit

- This bit controls interrupt requests of bit0:UDIR.
- When the UDIE bit is enabled, setting bit0:UDIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit3] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit2] TGIR: Trigger interrupt request bit

- When a software trigger or trigger input is detected, the TGIR bit is set to "1".
- The TGIR bit is cleared by writing "0".
- Even if "1" is written to the TGIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt factor.
1	Detects an interrupt factor.

[bit1] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit0] UDIR: Underflow interrupt request bit

- When a count value underflow from 0x0000 to 0xFFFF occurs during counting from the value for which the HIGH width is set, the UDIR bit is set to "1".
- The UDIR bit is cleared by writing "0".
- Even if "1" is written to the UDIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt factor.
1	Detects an interrupt factor.

9.2.7. LOW Width Reload Register (PRLL)

The LOW Width Reload Register (PRLL) is a register used to set the LOW width of PPG output waveforms. Transfer to the Timer Register is performed at detection of a start trigger or at an underflow after the completion of HIGH width counting.

bit	15	0
Field	PRLL[15:0]	
Attribute	R/W	
Initial value	0xXXXX	

This register is used to set the LOW width of PPG output waveforms. Transfer to the Timer Register is performed at detection of a start trigger and at an underflow at the completion of HIGH width counting.

- Do not access the PRLL register with 8-bit data.
- Set the LOW width for the PRLL register after setting the PPG function using the FMD[2:0] bits in the TMCR register.

9.2.8. HIGH Width Reload Register (PRLH)

The HIGH Width Reload Register (PRLH) is a buffered register used to set the HIGH width of PPG output waveforms. Transfer from the PRLH to the buffer register is performed at detection of a start trigger and at an underflow after the completion of HIGH width counting. Transfer from the buffer register to the Timer Register is performed at an underflow at the completion of LOW width counting.

bit	15 … 0
Field	PRLH[15:0]
Attribute	R/W
Initial value	0xXXXX

This register is used to set the HIGH width of PPG output waveforms. Transfer from the PRLH to the buffer register is performed at detection of a start trigger and at an underflow at the completion of HIGH width counting. Transfer from the buffer register to the Timer Register is performed at an underflow at the completion of LOW width counting.

- Do not access the PRLH register with 8-bit data.
- Set the HIGH width for the PRLH register after setting the PPG function using the FMD[2:0] bits in the TMCR register.

9.2.9. Timer Register (TMR)

The Timer Register (TMR) reads the value of the 16-bit down counter.

bit	15	0
Field	TMR[15:0]	
Attribute	R	
Initial value	0x0000	

The value of the 16-bit down counter is read.

- Do not access the TMR register with 8-bit data.

9.3. Reload timer function

The function of the base timer can be set to either the 16-bit PWM timer, 16-bit PPG timer, 16-/32-bit reload timer, or 16-/32-bit PWC timer using the FMD[2:0] bits in the Timer Control Register. This section explains the timer functions available when the reload timer is set.

1. Operations of the 16-bit reload timer
2. Reload timer operation flowchart
3. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the reload timer is selected
4. PWM Cycle Set Register (PCSR)
5. Timer Register (TMR)

9.3.1. Operations of the 16-bit reload timer

In reload timer operations, countdown is performed from the value set in the PWM Cycle Set Register in synchronization with the count clock. This operation continues until the count value reaches 0 or the cycle setting is loaded automatically to stop the countdown.

■ Count operation performed when the internal clock is selected

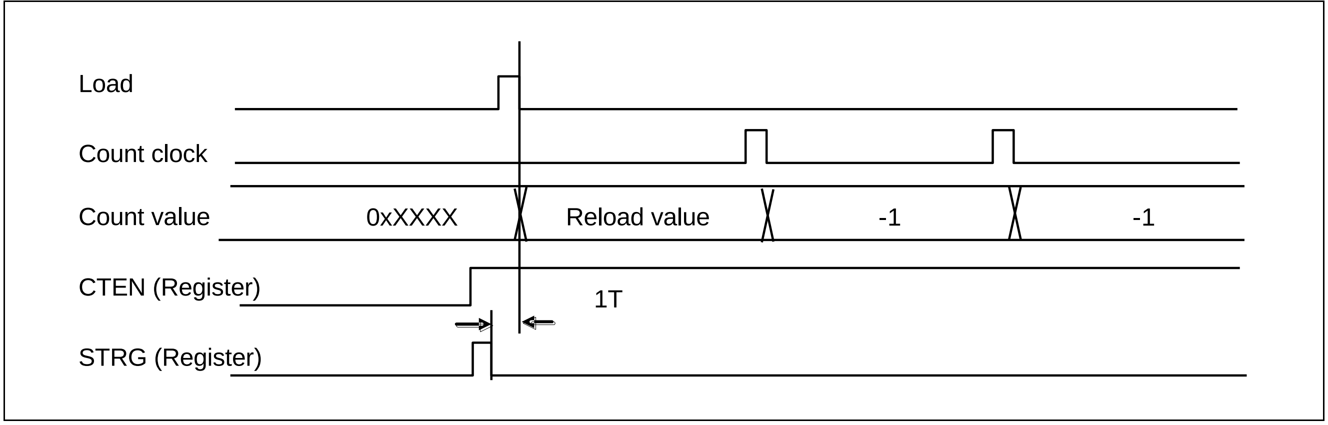
To start the count operation at the same time counting is enabled, write "1" to both CTEN and STRG bits in the Timer Control Register. When the timer is started (CTEN = 1), trigger input with the STRG bit is valid regardless of the operation mode.

When the count operation is enabled and the timer is started with a software trigger or an external trigger, the value in the PWM Cycle Set Register is loaded to the counter and countdown is started.

It takes a time of 1T (T: machine cycle) from setting of a counter start trigger to loading of the PWM Cycle Set Register data to the counter.

Figure 9-13 shows the start of the counter by a software trigger and counter operation.

Figure 9-13 Count operation performed when the internal clock is selected



■ Underflow operation

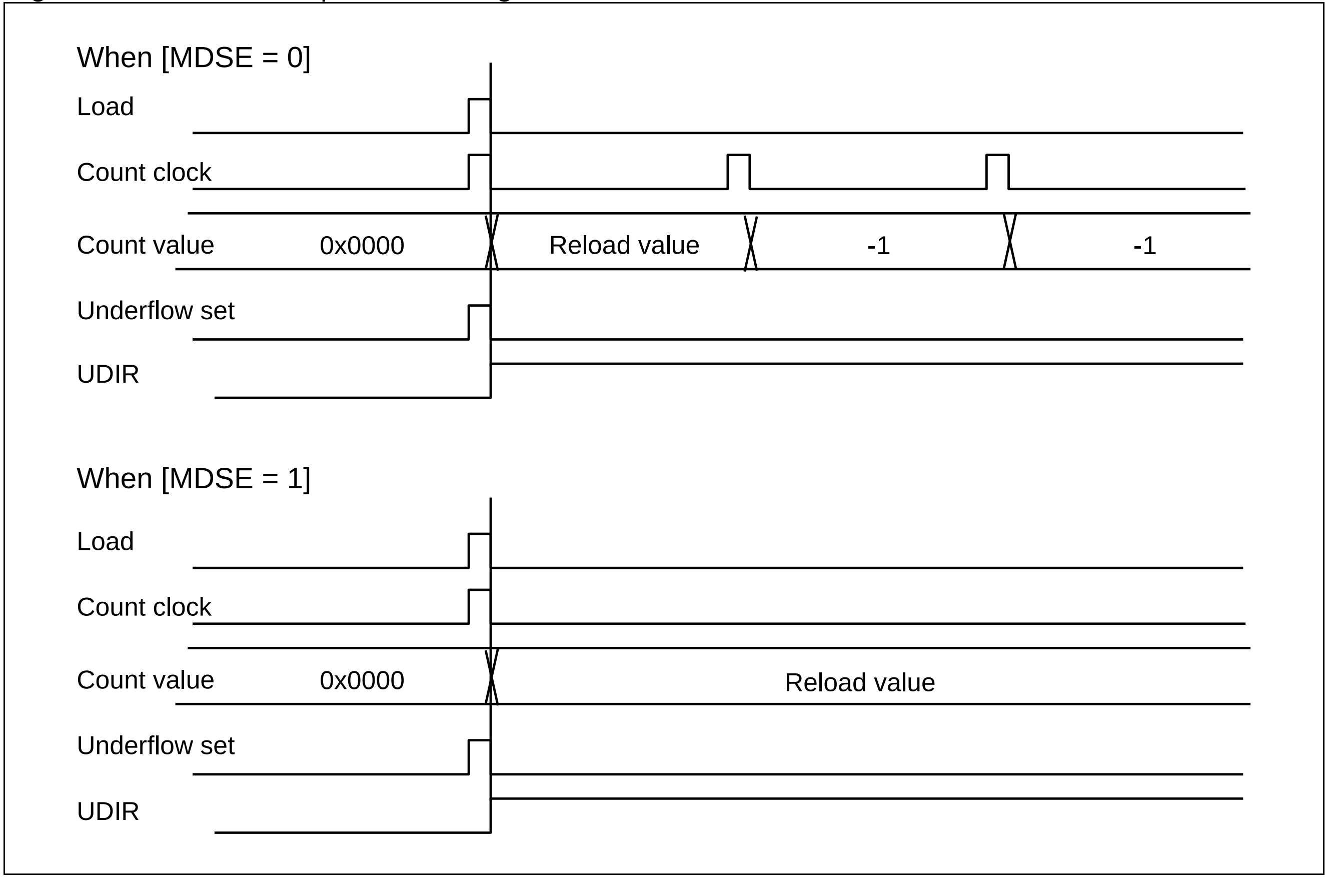
An underflow occurs when the counter value changes from 0x0000 to 0xFFFF. Therefore, an underflow occurs at a count of [Set value in the PWM Cycle Set Register + 1].

When an underflow occurs, the contents of the PWM Cycle Set Register (PCSR) are loaded to the counter. When the MDSE bit in the Timer Control Register (TMCR) is "0", the count operation continues. When the MDSE bit is "1", the counter stops while keeping the loaded counter value.

An underflow sets the UDIR bit in the Status Control Register (STC). In this case, an interrupt request occurs when the UDIE bit is "1".

Figure 9-14 shows a timing chart of underflow operations.

Figure 9-14 Underflow operation timing chart

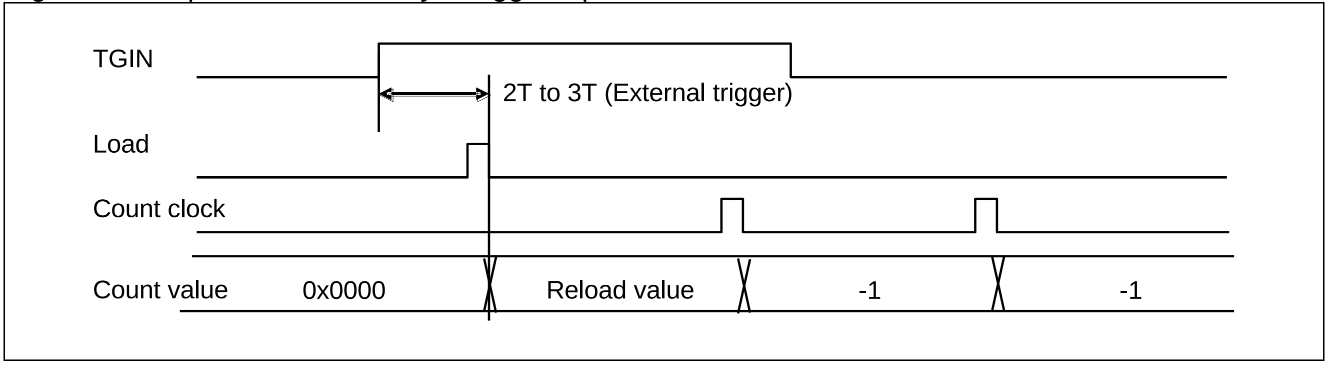


■ Operation of the input pin function

The TGIN pin can be used for trigger input. When a valid edge is input to the TGIN pin, the contents of the PWM Cycle Set Register are loaded to the counter and the count operation is started. As a time from trigger input to loading of the counter value, 2T to 3T (T: machine cycle) is required.

Figure 9-15 shows a trigger input operation performed when a rising edge is specified as a valid edge.

Figure 9-15 Operation caused by a trigger input

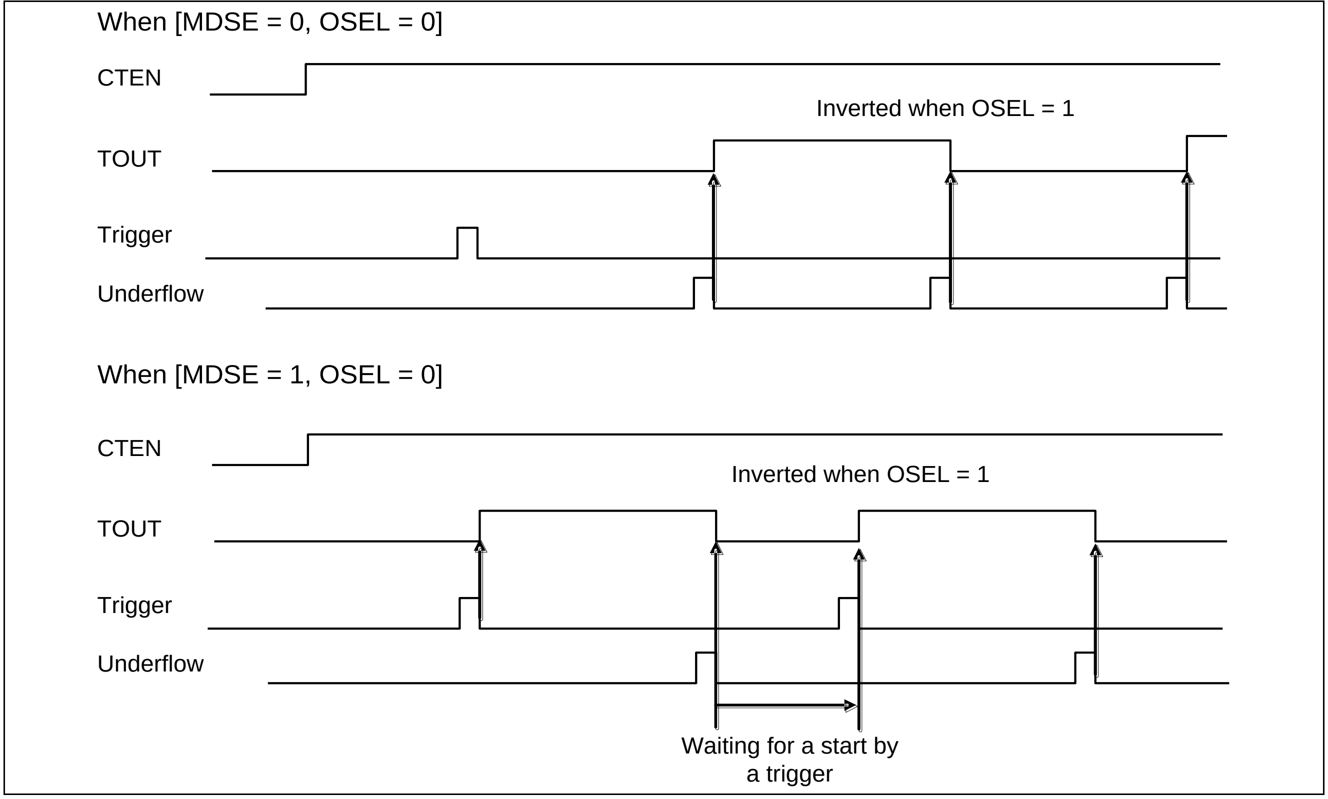


■ Operation of the output pin function

The TOUT output pin functions as, in reload mode, toggle output inverted by an underflow and, in one-shot mode, pulse output indicating that counting is in progress. The output polarity can be set with the OSEL bit in the Timer Control Register (TMCR). If OSEL = 0, toggle output has an initial value of "0", and one-shot pulse output is "1" during counting. When OSEL is set to 1, the output waveform is inverted.

Figure 9-16 shows a timing chart of output pin function operations.

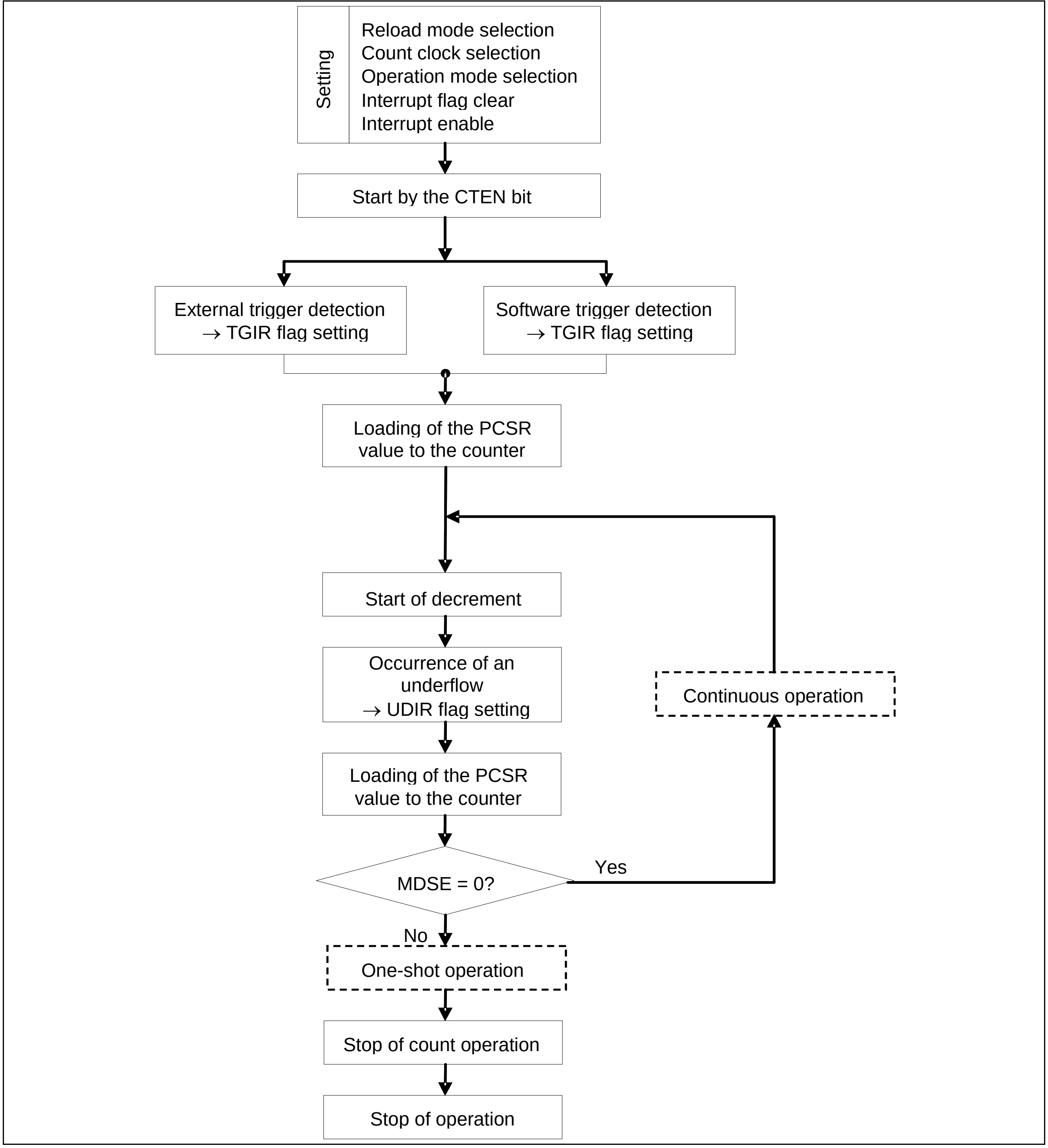
Figure 9-16 Output pin function operation timing chart



9.3.2. Reload timer operation flowchart

This section provides an operation flowchart of the reload timer.

■ Reload timer operation flowchart



9.3.3. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the reload timer is selected

The Timer Control Register (TMCR) controls timer operations.

■ Timer Control Register (Upper bytes of TMCR)

bit	15	14	13	12	11	10	9	8
Field	Reserved	CKS2	CKS1	CKS0	Reserved		EGS1	EGS0
Attribute	R/W	R/W	R/W	R/W	R/W		R/W	R/W
Initial value	0	0	0	0	0b00		0	0

[bit15] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit14:12, TMCR2:bit8] CKS3 to CKS0: Count clock selection bits

- Select the count clock for the 16-bit down counter.
- Changes to the count clock setting are applied immediately. For this reason, changes to CKS3 through CKS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

CKS3	CKS2	CKS1	CKS0	Description
0	0	0	0	ϕ
0	0	0	1	ϕ /4
0	0	1	0	ϕ /16
0	0	1	1	ϕ /128
0	1	0	0	ϕ /256
0	1	0	1	External clock (rising edge event)
0	1	1	0	External clock (falling edge event)
0	1	1	1	External clock (both edge event)
1	0	0	0	ϕ /512
1	0	0	1	ϕ /1024
1	0	1	0	ϕ /2048
Others				Setting is prohibited.

[bit11:10] Reserved : Reserved bits

The read value is "0".
Set "0" to these bits.

[bit9:8] EGS1, EGS0: Trigger input edge selection bits

- These bits select a valid edge for input waveforms as an external start cause and set the trigger condition.
- When the initial value or "0b00" is set, the timer is not started by external waveforms because the setting means that no valid edge is selected for input waveforms.
- Changes to EGS1 or EGS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit9	bit8	Description
0	0	Trigger input disabled
0	1	External trigger (rising edge)
1	0	External trigger (falling edge)

<Note>

If the STRG bit is set to "1", software triggering is enabled regardless of the EGS1 and EGS0 settings.

■ Timer Control Register 2 (Lower bytes of TMCR)

bit	7	6	5	4	3	2	1	0
Field	T32	FMD2	FMD1	FMD0	OSEL	MDSE	CTEN	STRG
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] T32: 32-bit timer selection bit

- This bit selects the 32-bit timer function.
- When the FMD[2:0] bits are set to "0b011" to select the reload timer function, setting the T32 bit to "1" selects 32-bit timer mode.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit (see 32-bit mode operations of "4 32-bit mode operations").

Value	Description
0	16-bit timer mode
1	32-bit timer mode

[bit6:4] FMD2 to FMD0: Timer function selection bits

- These bits select the timer function.
- When the FMD[2:0] bits are set to "0b011", the reload timer function is selected.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit6	bit5	bit4	Description
0	0	0	Reset mode
0	0	1	Selection of the PWM function
0	1	0	Selection of the PPG function
0	1	1	Selection of the reload timer function
1	0	0	Selection of the PWC function
Others			Setting is prohibited.

[bit3] OSEL: Output polarity specification bit

- This bit selects whether to invert the timer output level.
- Used in combination with bit 2 MDSE, this bit generates the following output waveforms.

MDSE	OSEL	Output waveforms
0	0	Toggle output at the LOW level at the start of counting
0	1	Toggle output at the HIGH level at the start of counting
1	0	Rectangular waves at the HIGH level during counting
1	1	Rectangular waves at the LOW level during counting

Value	Description
0	Normal polarity
1	Inverted polarity

[bit2] MDSE: Mode selection bit

- When the MDSE bit is set to "0", reload mode is selected. When a count value underflow from 0x0000 to 0xFFFF occurs, the reload register value is loaded to the counter at the same time, and the count operation is continued.
- When the MDSE bit is set to "1", one-shot mode is selected. A count value underflow from 0x0000 to 0xFFFF stops the operation.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

Value	Description
0	Reload mode
1	One-shot mode

[bit1] CTEN: Timer enable bit

- This bit enables the operation of the down counter.
- When the counter is in operation enabled status (the CTEN bit is "1"), writing "0" to this bit stops the counter.

Value	Description
0	Stop
1	Operation enabled

<Note>

By writing "0" to CTEN, TOUT is set to Low.

[bit0] STRG: Software trigger bit

- When the CTEN bit is "1", writing "1" to the STRG bit enables software triggering.
- The read value of the STRG bit is always "0".

Value	Description
0	Invalid
1	Start triggered by software

<Notes>

- Software triggering is also enabled when "1" is written to the CTEN and STRG bits simultaneously.
- If the STRG bit is set to "1", software triggering is enabled regardless of the EGS1 and EGS0 settings.

■ Timer Control Register 2 (Upper bytes of TMCR2)

bit	15	14	13	12	11	10	9	8
Field	Reserved							CKS3
Attribute	R/W							R/W
Initial value	0000000							0

Note: This register is placed above the STC register.

[bit15:9] Reserved: Reserved bits

The read value is "0".

Set "0" to these bits.

[bit8] CKS3: Count clock selection bit

See "Count clock selection bit" in "9.3.3 Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the reload timer is selected".

■ Status Control Register (STC)

bit	7	6	5	4	3	2	1	0
Field	Reserved	TGIE	Reserved	UDIE	Reserved	TGIR	Reserved	UDIR
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

Note: The TMCR2 register is placed in the upper bytes of this register.

[bit7] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit6] TGIE: Trigger interrupt request enable bit

- This bit controls interrupt requests of bit2 TGIR.
- When the TGIE bit is enabled, setting bit2 TGIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit5] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit4] UDIE: Underflow interrupt request enable bit

- This bit controls interrupt requests of bit0 UDIR.
- When the UDIE bit is enabled, setting bit0 UDIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit3] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit2] TGIR: Trigger interrupt request bit

- When a software trigger or trigger input is detected, the TGIR bit is set to "1".
- The TGIR bit is cleared by writing "0".
- Even if "1" is written to the TGIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt factor.
1	Detects an interrupt factor.

[bit1] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit0] UDIR: Underflow interrupt request bit

- When a count value underflow from 0x0000 to 0xFFFF occurs during counting from the value for which the HIGH width is set, the UDIR bit is set to "1".
- The UDIR bit is cleared by writing "0".
- Even if "1" is written to the UDIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt factor.
1	Detects an interrupt factor.

9.3.4. PWM Cycle Set Register (PCSR)

The PWM Cycle Set Register (PCSR) is a register for storing the initial counter value. In 32-bit mode and for the even channel, the initial count value of the lower 16 bits is stored. For the odd channel, the initial count value of the upper 16 bits is stored. The initial value after a reset is undefined. Do not access this register with 8-bit data.

bit	15	0
Field	PCSR [15:0]	
Attribute	R/W	
Initial value	0xXXXX	

This is a register for setting the cycle. Transfer to the Timer Register is performed at an underflow.

- Do not access the PCSR register with 8-bit data.
- Set the cycle for the PCSR register after setting the reload timer function using the FMD[2:0] bits in the TMCR register.
- When writing data in the PCSR register in 32-bit mode, access the upper 16-bit data (odd channel data) first, and then access the lower 16-bit data (even channel data).

9.3.5. Timer Register (TMR)

The Timer Register (TMR) is a register that reads the count value of a timer. In 32-bit mode and for the even channel, the count value of the lower 16 bits is read. For the odd channel, the count value of the upper 16 bits is read. The initial value is undefined.

Do not access this register with 8-bit data.

bit	15	0
Field	TMR [15:0]	
Attribute	R	
Initial value	0xXXXX	

The value of the 16-bit down counter is read.

- Do not access the TMR register with 8-bit data.
- When reading the TMR register in 32-bit mode, read the lower 16-bit data (even channel data) first, and then read the upper 16-bit data (odd channel data).

9.4. PWC timer function

The function of the base timer can be set to either the 16-bit PWM timer, 16-bit PPG timer, 16-/32-bit reload timer, or 16-/32-bit PWC timer using the FMD[2:0] bits in the Timer Control Register. This section explains the timer functions available when PWC is set.

1. Operations of the PWC timer
2. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PWC timer is selected
3. Data Buffer Register (DTBF)

9.4.1. Operations of the PWC timer

The PWC timer has the pulse width measurement function. Five types of count clock are available for measuring the time and cycle between any input pulse events by the counter. This section explains the basic functions and operations of the pulse width measurement function.

■ Pulse width measurement function

Count operation is not performed until the counter is started and cleared to "0x0000" and the specified measurement start edge is input. Upon detecting a measurement start edge, the counter starts count-up from "0x0001" and stops counting upon detecting a measurement end edge. The value counted in between is stored as a pulse width in the register.

An interrupt request can be generated when the measurement is completed or an overflow occurs.

After the completion of measurement, it operates as follows depending on the measurement mode:

· In one-shot measurement mode:	Stops the operation.
· In continuous measurement mode:	Transfers the counter value to the buffer register and stops counting until the measurement start edge is input again.

Figure 9-17 Pulse width measurement operation (one-shot measurement mode/HIGH width measurement)

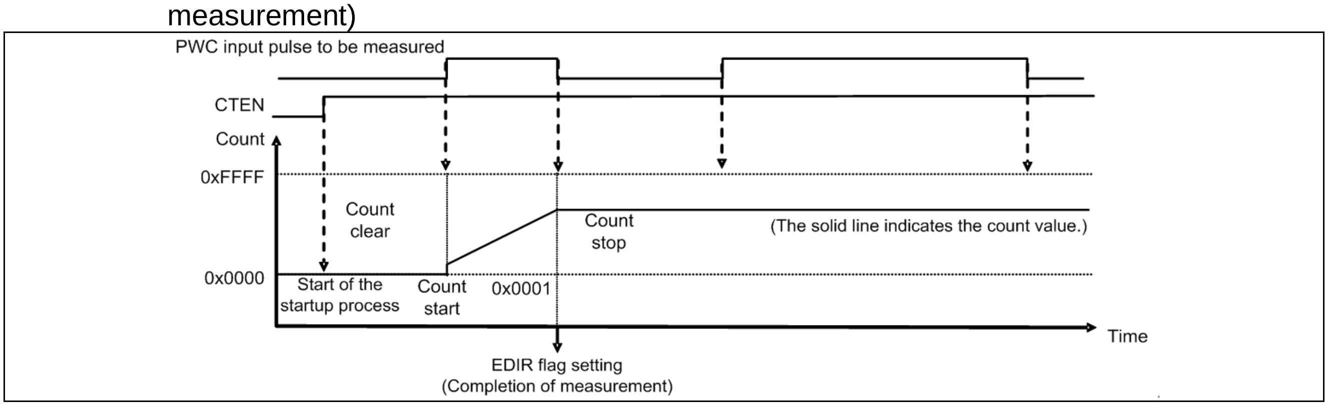
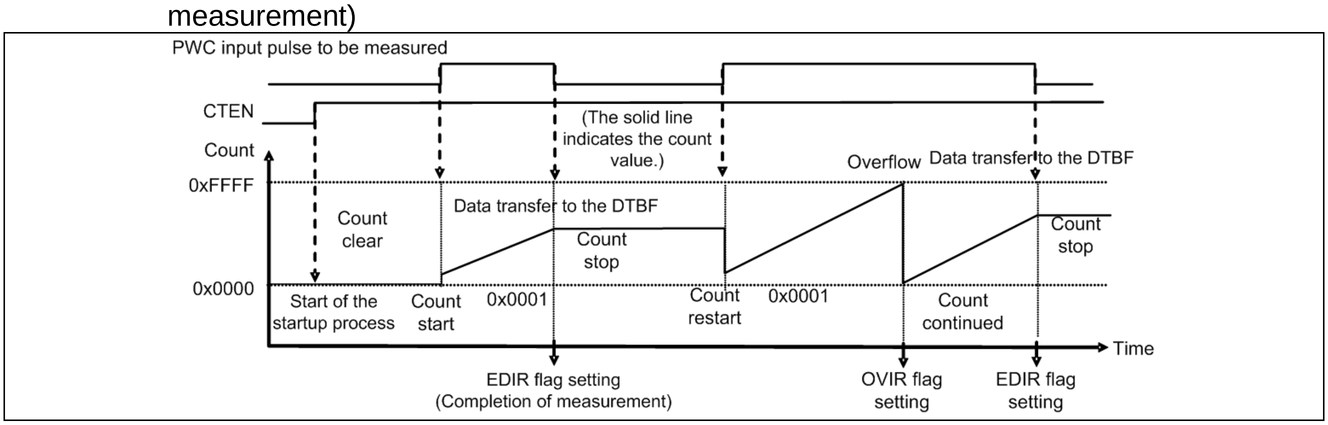


Figure 9-18 Pulse width measurement operation (continuous measurement mode/HIGH width measurement)



■ Selection of count clock

The count clock of the counter can be selected from eight types by setting bit8: CKS3 in the TMCR2 register and bit14:12: CKS2, CKS1, and CKS0 in the TMCR register.

The selectable count clocks are as follows:

TMCR2 and TMCR registers	Internal count clock to be selected
CKS3, CKS2, CKS1, and CKS0 bits	
0000	Machine clock [Initial value]
0001	1/4 frequency of the machine clock
0010	1/16 frequency of the machine clock
0011	1/128 frequency of the machine clock
0100	1/256 frequency of the machine clock
0101	Setting is prohibited.
0110	
0111	
1000	1/512 frequency of the machine clock
1001	1/1024 frequency of the machine clock
1010	1/2048 frequency of the machine clock
Others	Setting is prohibited.

The machine clock is selected as the initial value after a reset.
Be sure to select the count clock before starting the counter.

■ Selection of operation mode

Set the TMCR to select the operation/measurement mode.

Operation mode setting ... TMCR bit10:8: EGS2, EGS1, and EGS0 (Selection of measurement edge)
Measurement mode setting ... TMCR bit2: MDSE (Selection of one-shot/continuous measurement)

The following provides a list of operation mode settings.

Operation mode		MDSE	EGS2	EGS1	EGS0
↑ to ↓ HIGH pulse width measurement	Continuous measurement: Buffer enabled	0	0	0	0
	One-shot measurement: Buffer disabled	1	0	0	0
↑ to ↑ Cycle measurement between rising edges	Continuous measurement: Buffer enabled	0	0	0	1
	One-shot measurement: Buffer disabled	1	0	0	1
↓ to ↓ Cycle measurement between falling edges	Continuous measurement: Buffer enabled	0	0	1	0
	One-shot measurement: Buffer disabled	1	0	1	0
↑ or ↓ to ↑ or ↓ Interval measurement between all edges	Continuous measurement: Buffer enabled	0	0	1	1
	One-shot measurement: Buffer disabled	1	0	1	1
↓ to ↑ LOW pulse width measurement	Continuous measurement: Buffer enabled	0	1	0	0
	One-shot measurement: Buffer disabled	1	1	0	0
Setting is prohibited.		0	1	0	1
		1	1	0	1
		0	1	1	0
		1	1	1	0
		0	1	1	1
		1	1	1	1

HIGH pulse width measurement in one-shot measurement mode is selected as the initial value after a reset.
Be sure to select an operation mode before starting the counter.

■ Starting and stopping pulse width measurement

Set bit1: CTEN bit in the TMCR to start, restart, or stop forcibly each operation.
The pulse width measurement is started or restarted by writing "1" to the CTEN bit, and it is stopped forcibly by writing "0" to the CTEN bit.

CTEN	Function
1	Starts or restarts the pulse width measurement.
0	Forcibly stops the pulse width measurement.

■ Operation after a restart

After the counter is restarted in pulse measurement mode, counting is not performed until a measurement start edge is input. After a measurement start edge is detected, the 16-bit up counter starts counting from "0x0001".

■ Restart

An operation to start the counter again after it has been started and while it is in operation (writing "1" again while the CTEN bit is "1") is referred to as a restart. When restarted, the counter performs the following operation:

- When waiting for a measurement start edge:
  Has no effect on operation.

- When performing measurement:
  Clears the count to "0x0000" and waits for a measurement start edge again. When detection of a measurement end edge and the restart operation occur simultaneously, a measurement end flag (EDIR) is set and, when in continuous measurement mode, the measurement result is transferred to the DTBF.

■ Stop

In one-shot measurement mode, since the count operation is stopped automatically by a counter overflow or completion of measurement, you do not have to be aware of the stop. In continuous measurement mode or when you want to stop the operation before it stops automatically, you have to stop it forcibly.

■ Counter clear and initial value

The 16-bit up counter is cleared to "0x0000" in the following cases:

- When a reset is performed
- When "1" is written to bit 1: CTEN bit in the TMCR (including the cases for restarting)

The 16-bit up counter is initialized to "0x0001" in the following case:

- When a measurement start edge is detected

■ Details on pulse width measurement operations

● One-shot measurement and continuous measurement

Pulse width measurement can be performed in two modes: one for performing measurement only one time and the other for performing it continuously. Each mode is selected with the MDSE bit in the TMCR (see "■ Selection of operation mode"). Differences between these modes are as follows:

One-shot measurement mode:

When the first measurement end edge is input, the counter stops counting, a measurement end flag (EDIR) in the STC is set, and no further measurement is performed.
However, when restarted at the same time, it waits to start measurement.

Continuous measurement mode:

When a measurement end edge is input, the counter stops counting, a measurement end flag (EDIR) in the STC is set, and the counter stops until the measurement start edge is input again. When the measurement start edge is input again, the counter is initialized to "0x0001" and measurement is started. After the measurement is completed, the result in the counter is transferred to the DTBF.

Be sure to select/change measurement modes while the counter is stopped.

● Measurement result data

There are differences between one-shot and continuous modes in handling of measurement results and counter values and in DTBF functions. Differences between these modes in handling of measurement results are as follows:

One-shot measurement mode:

Reading the DTBF during operation obtains the count value being measured.
Reading the DTBF after the completion of measurement obtains the measurement result data.

Continuous measurement mode:

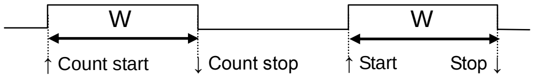
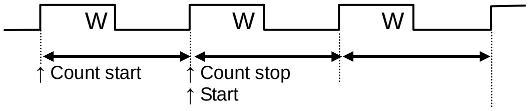
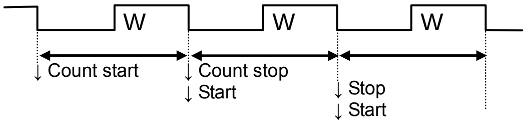
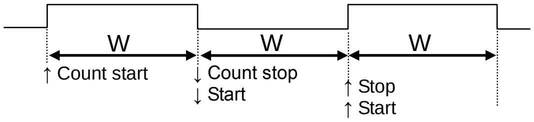
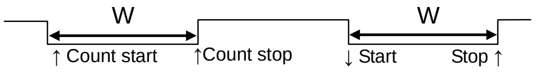
After the measurement is completed, the result in the counter is transferred to the DTBF.

Reading the DTBF obtains the last measurement result. The previous measurement result is retained during the measurement operation. It is not possible to read the count value being measured.

In continuous measurement mode, if the next measurement is completed before the measurement result is read, the previous result is overwritten by the new result. In this case, an error flag (ERR) in the STC is set. The error flag is cleared automatically when the DTBF is read.

● Measurement modes and count operations

The measurement mode can be selected from five types, differing in which part of the input pulse is measured. The following are explanations:

Measurement mode	EGS[2:0]	Item to be measured (W: Pulse width to be measured)
HIGH pulse width measurement	000	 W W ↑ Count start ↓ Count stop ↑ Start Stop ↓ The width of the HIGH period is measured. Count (measurement) start: At detection of a rising edge Count (measurement) end: At detection of a falling edge
Cycle measurement between rising edges	001	 The cycle between rising edges is measured. Count (measurement) start: At detection of a rising edge Count (measurement) end: At detection of a rising edge
Cycle measurement between falling edges	010	 W W W ↓ Count start ↓ Count stop ↓ Start ↓ Stop ↓ Start The cycle between falling edges is measured. Count (measurement) start: At detection of a falling edge Count (measurement) end: At detection of a falling edge
Pulse width measurement between all edges	011	 W W W ↑ Count start ↓ Count stop ↓ Start ↑ Stop ↑ Start The width between continuously input edges is measured. Count (measurement) start: At detection of an edge Count (measurement) end: At detection of an edge
LOW pulse width measurement	100	 The width of the LOW period is measured. Count (measurement) start: At detection of a falling edge Count (measurement) end: At detection of a rising edge

In any measurement mode, the counter is cleared to "0x0000" when started, and it does not perform the count operation until a measurement start edge is input. Once a measurement start edge is input, the counter continues incrementing for every count clock until a measurement end edge is input.

In pulse width measurement between all edges or cycle measurement in continuous measurement mode, an end edge is also a start edge for the next measurement.

Pulse width/cycle calculation method

After completion of measurement, the measured pulse width/cycle can be calculated as follows from the measurement result data stored in the DTBF:

$T_W = n \times t$	T_W	:	Measured pulse width/cycle
	n	:	Measurement result data stored in the DTBF
	t	:	Count clock cycle

Generation of interrupt requests

Two interrupt requests can be generated.

- Interrupt request due to a counter overflow

  When count-up causes an overflow during measurement, an overflow flag (OVIR) is set and an interrupt request is generated if overflow interrupt requests are enabled.

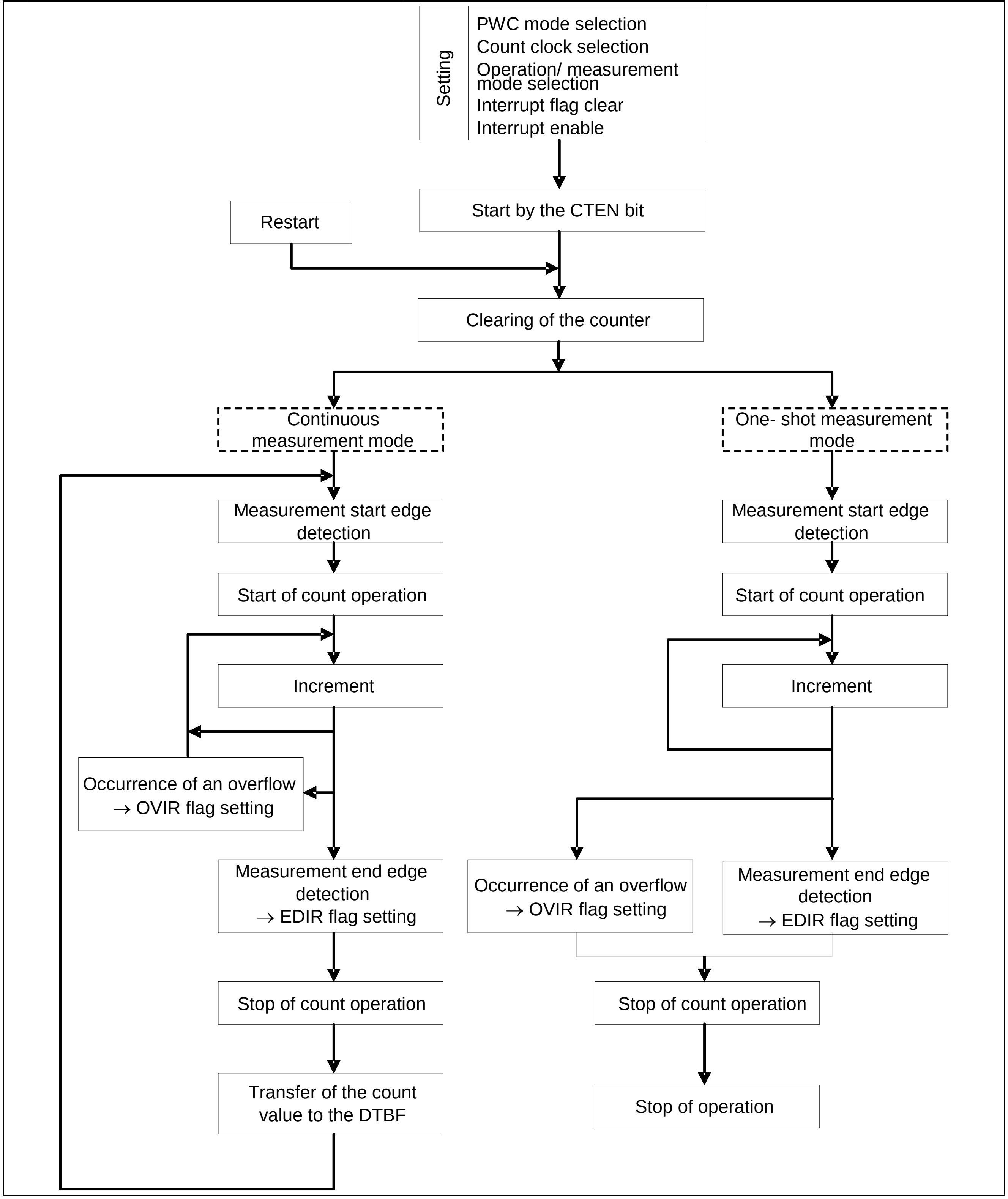
- Interrupt request due to completion of measurement

  When a measurement end edge is detected, a measurement end flag (EDIR) in the STC is set and an interrupt request is generated if measurement end interrupt requests are enabled.

  The measurement end flag (EDIR) is cleared automatically when the measurement result DTBF is read.

● Pulse width measurement operation flowchart

Figure 9-19 Pulse width measurement operation flowchart


Setting
PWC mode selection
Count clock selection
Operation/ measurement mode selection
Interrupt flag clear
Interrupt enable
Start by the CTEN bit
Restart
Clearing of the counter
Continuous measurement mode
One- shot measurement mode
Measurement start edge detection
Start of count operation
Increment
Occurrence of an overflow → OVIR flag setting
Measurement end edge detection → EDIR flag setting
Stop of count operation
Transfer of the count value to the DTBF
Measurement start edge detection
Start of count operation
Increment
Occurrence of an overflow → OVIR flag setting
Measurement end edge detection → EDIR flag setting
Stop of count operation
Stop of operation

9.4.2. Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PWC timer is selected

The Timer Control Register (TMCR) controls timer operations.

■ Timer Control Register (Upper bytes of TMCR)

bit	15	14	13	12	11	10	9	8
Field	Reserved	CKS2	CKS1	CKS0	Reserved	EGS2	EGS1	EGS0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit15] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit14:12, TMCR2:bit8] CKS3 to CKS0: Count clock selection bits

· Select the count clock for the 16-bit down counter.

· Changes to the count clock setting are applied immediately. For this reason, changes to CKS3 through CKS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

CKS3	CKS2	CKS1	CKS0	Description
0	0	0	0	ϕ
0	0	0	1	ϕ /4
0	0	1	0	ϕ /16
0	0	1	1	ϕ /128
0	1	0	0	ϕ /256
0	1	0	1	Setting is prohibited.
0	1	1	0	
0	1	1	1	
1	0	0	0	ϕ /512
1	0	0	1	ϕ /1024
1	0	1	0	ϕ /2048
Others				Setting is prohibited.

[bit11] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit10:8] EGS2 to EGS0: Measurement edge selection bits

- These bits set measurement edge conditions.
- Changes to EGS2, EGS1, or EGS0 must be made when the counting is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit10	bit9	bit8	Description
0	0	0	HIGH pulse width measurement (↑ to ↓)
0	0	1	Cycle measurement between rising edges (↑ to ↑)
0	1	0	Cycle measurement between falling edges (↓ to ↓)
0	1	1	Pulse width measurement between all edges (↑ or ↓ to ↓ or ↑)
1	0	0	LOW pulse width measurement (↓ to ↑)
1	0	1	Setting is prohibited.
1	1	0	
1	1	1	

■ Timer Control Register (Lower bytes of TMCR)

bit	7	6	5	4	3	2	1	0
Field	T32	FMD2	FMD1	FMD0	Reserved	MDSE	CTEN	Reserved
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] T32: 32-bit timer selection bit

- This bit selects the 32-bit timer function.
- When the FMD[2:0] bits are set to "0b100" to select the PWC function, setting the T32 bit to "1" selects 32-bit PWC mode.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit (see 32-bit mode operations of "4 32-bit mode operations").

Value	Description
0	16-bit timer mode
1	32-bit timer mode

[bit6:4] FMD2 to FMD0: Timer function selection bits

- These bits select the timer function.
- When the FMD[2:0] bits are set to "0b100", the PWC timer function is selected.
- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

bit6	bit5	bit4	Description
0	0	0	Reset mode
0	0	1	Selection of the PWM function
0	1	0	Selection of the PPG function
0	1	1	Selection of the reload timer function
1	0	0	Selection of the PWC function
1	0	1	Setting is prohibited.
1	1	0	
1	1	1	

[bit3] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit2] MDSE: Mode selection bit

- Changes must be made while the timer is stopped (CTEN = "0"). However, it is possible to make changes at the same time you set "1" to the CTEN bit.

Value	Description
0	Continuous measurement mode (Buffer register enabled)
1	One-shot measurement mode (Stops after one measurement)

[bit1] CTEN: Timer enable bit

- This bit enables the start or restart of the up counter.
- When the counter is in operation enabled status (the CTEN bit is "1"), writing "1" restarts the counter. The counter is cleared and waits for a measurement start edge.
- When the counter is in operation enabled status (the CTEN bit is "1"), writing "0" to this bit stops the counter.

Value	Description
0	Stop
1	Operation enabled

<Note>

By writing "0" to CTEN, the output waveform is set to Low.

[bit0] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

■ Timer Control Register 2 (Upper bytes of TMCR2)

bit	15	14	13	12	11	10	9	8
Field	Reserved							CKS3
Attribute	R/W							R/W
Initial value	0000000							0

Note: This register is placed above the STC register.

[bit15:9] Reserved: Reserved bits

The read value is "0".

Set "0" to these bits.

[bit8] CKS3: Count clock selection bit

See "Count clock selection bit" in "9.4.2 Timer Control Registers (TMCR and TMCR2) and Status Control Register (STC) used when the PWC timer is selected".

■ Status Control Register (STC)

bit	7	6	5	4	3	2	1	0
Field	ERR	EDIE	Reserved	OVIE	Reserved	EDIR	Reserved	OVIR
Attribute	R	R/W	R/W	R/W	R/W	R	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

Note: The TMCR2 register is placed in the upper bytes of this register.

[bit7] ERR: Error flag bit

- This flag indicates that the next measurement has been completed in continuous measurement mode before the measurement result is read from the DTBF register. In this case, the result of the previous measurement in the DTBF register is replaced by that of the next measurement.
- The measurement is continued regardless of the ERR bit value.
- The ERR bit is read-only. Writing a value does not affect the bit value.
- The ERR bit is cleared by reading the measurement result (DTBF).

Value	Description
0	Normal status
1	A measurement result not yet read was overwritten by the next measurement result.

[bit6] EDIE: Measurement completion interrupt request enable bit

- This bit controls interrupt requests of bit 2 EDIR.
- When the EDIE bit is enabled, setting bit 2 EDIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit5] Reserved : Reserved bit

The read value is "0".
Set "0" to this bit.

[bit4] OVIE: Overflow interrupt request enable bit

- This bit controls interrupt requests of bit 0 OVIR.
- When the OVIE bit is enabled, setting bit 0 OVIR generates an interrupt request to the CPU.

Value	Description
0	Disables interrupt requests.
1	Enables interrupt requests.

[bit3] Reserved: Reserved bit

The read value is "0".
Set "0" to this bit.

[bit2] EDIR: Measurement completion interrupt request bit

- This bit indicates that the completion of measurement. The flag is set to "1" when the measurement is completed.
- The EDIR bit is cleared by reading the measurement result (DTBF).
- The EDIR bit is read-only. Writing a value does not affect the bit value.

Value	Description
0	Reads the measurement result (DTBF).
1	Detects an interrupt cause.

[bit1] Reserved: Reserved bit

The read value is "0".

Set "0" to this bit.

[bit0] OVIR: Overflow interrupt request bit

- When a count value overflow from 0xFFFF to 0x0000 occurs, the flag is set to "1".
- The OVIR bit is cleared by writing "0".
- Even if "1" is written to the OVIR bit, the bit value is not affected.
- The read value of read-modify-write instructions is "1" regardless of the bit value.

Value	Description
0	Clears an interrupt factor.
1	Detects an interrupt factor.

9.4.3. Data Buffer Register (DTBF)

The Data Buffer Register (DTBF) is a register that reads the measured or count value of the PWC timer. In 32-bit mode, the value of the lower 16 bits is read for the even channel and that of the upper 16 bits for the odd channel.

Do not access this register with 8-bit data.

bit	15	0
Field	DTBF [15:0]	
Attribute	R	
Initial value	0x0000	

- The DTBF register is read-only in both continuous and one-shot measurement modes. Writing a value does not change the register value.
- In continuous measurement mode (TMCR bit3 MDSE = 1), this register works as a buffer register that stores the previous measurement result.
- In one-shot measurement mode (TMCR bit3 MDSE = 0), the DTBF register accesses the up counter directly. The count value can be read during counting. The measurement value is retained after the completion of measurement.
- Do not access the DTBF register with 8-bit data.

CHAPTER 6: Multifunction Timer

This chapter explains the multifunction timer unit.

CODE: 9BFMFT-E01.5

1. Overview of Multifunction Timer

The multifunction timer is a function block that enables three-phase motor control. In conjunction with PPG and A/D converter (called "ADC" hereafter), it can provide a variety of motor controls. An overview of the multifunction timer is provided below.

■ Functions

The multifunction timer has the following functions.

- It can output PWM signals with any cycle/pulse length (PWM signal output function).
- It can start PPG in synchronization with PWM signal output. It can superimpose PPG's output signal on the PWM signal and output it (DC chopper waveform output function).
- It can generate a non-overlap signal that maintains the response time of the power transistor (dead time) from PWM signal output (dead timer function).
- It can capture the timing of changing the input signal and its pulse width in synchronization with PWM signal output (Input capture function).
- It can start ADC at any timing, in synchronization with PWM signal output (ADC start function).
- It performs noise canceling to the emergency motor shutdown interrupt signal (DTTIX input signal). It can set freely the pin state at the time of motor shutdown, when a valid signal input is detected (DTIF interrupt function).

■ Block Configuration

The multifunction timer (1 unit) consists of the following function blocks.

· Free-run Timer Unit	:	3channels
· Output Compare Unit	:	6channels (2channels ×3units)
· Waveform Generator Unit	:	3channels
· Noise Canceller Unit	:	1channel
· Input Capture Unit	:	4channels (2channels ×2units)
· ADC Start Compare Unit	:	3channels
· ADC Start Trigger Selector Unit	:	3channels

The multifunction timer is configured to enable one three-phase motor control, when 1 unit is used. Some models in this series contain multiple units of the multifunction timer, which are configured to support multiple three-phase motor controls.

■ Abbreviations

In this chapter, the following abbreviations are used in explanations.

MFT	Multifunction Timer Unit
PPG	Programmable Pulse Generator Unit
FRT	Free-run Timer Unit
FRTS	Free-run Timer Selector
OCU	Output Compare Unit
WFG	Waveform Generator Unit
NZCL	Noise Canceller Unit
ICU	Input Capture Unit
ADCMP	ADC Start Compare Unit
ATSA	ADC Start Trigger Selector Unit

2. Configuration of Multifunction Timer

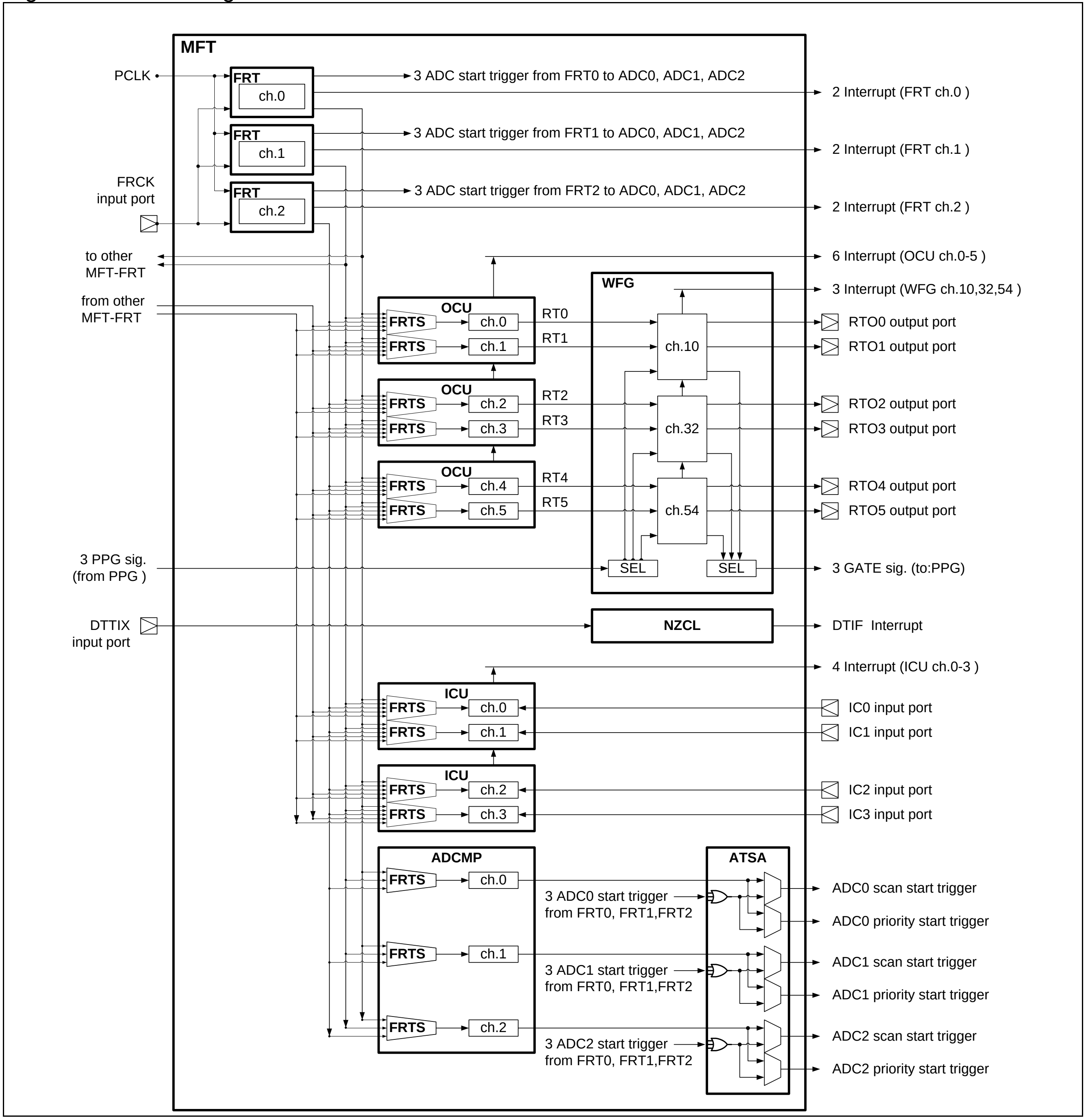
This section explains the configuration of the multifunction timer and the functions of each function block and I/O pin.

2.1. Block Diagram of Multifunction Timer

■ Block Diagram

Figure 2-1 shows the block diagram of the entire function timer.

Figure 2-1 Block Diagram of Multifunction Timer

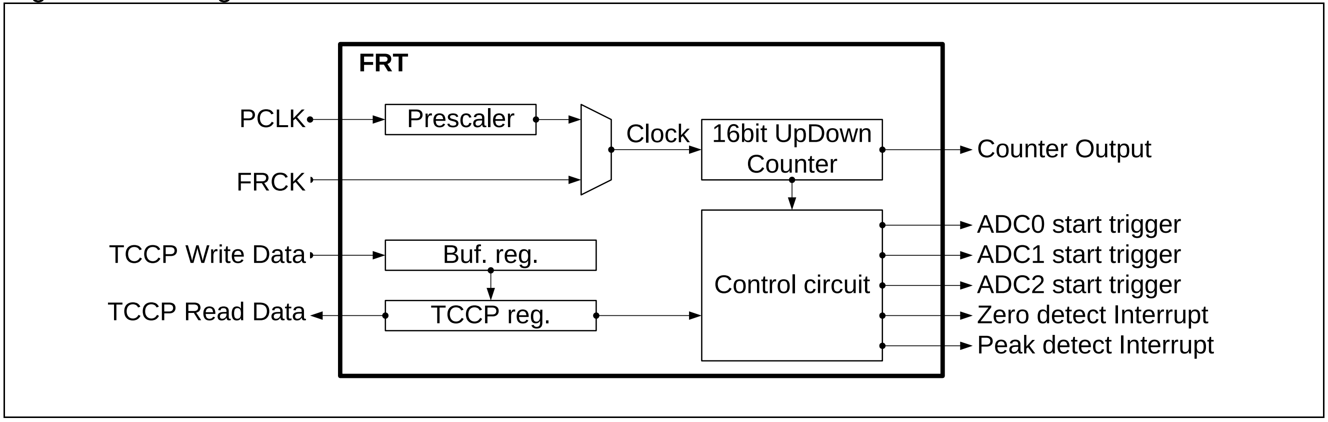


2.2. Description of Each Function Block

■ FRT: 3channels

- FRT is a timer function block that outputs the reference counter value for the operation of each function block in MFT.
- FRT consists of a clock prescaler, 16-bit Up/Down counter, cycle setting register (TCCP register) and controller. Figure 2-2 shows the configuration of FRT.
  - The clock prescaler divides the peripheral clock (PCLK) signal in LSI to generate the operating clock for the 16-bit Up/Down counter.
  - The TCCP register sets the count cycle for the 16-bit Up/Down counter. It has a buffer register to change a cycle during count operation.
  - The 16-bit Up/Down counter performs Up-count or Down-count operation in the count cycle specified by the TCCP register in order to output a counter value.
- The following processing can be achieved by instructing the controller via CPU.
  - The division ratio of the clock prescaler can be selected.
  - The use of PCLK (internal clock) and FRCK (external clock) can be selected.
  - The count mode for the 16-bit Up/Down-counter can be selected to specify whether to start or stop the count operation.
  - The buffer register function of the TCCP register can be enabled or disabled.
  - Interrupt can be generated to CPU by detecting a case where the count value is set to "0x0000" or the peak value (= TCCP value) (two interrupt signals output per FRT1 channel).
  - AD conversion start signals can be generated to each ADC by detecting a case where the count value is set to "0x0000" (three AD conversion start signals output per FRT1 channel).
- Each MFT unit is in a 3-channel configuration with three FRT's, which can operate independently from one another.
- Inside MFT, the output of the FRT counter value is connected to OCU, ICU and ADCMP. These units have a circuit (FRTS) that selects FRT to be connected. Interlocked operation can be performed based on the output of the counter value of the selected FRT. All of the units can be interlocked by a single FRT, or 2 or 3 groups can be formed as interlocked operational groups.

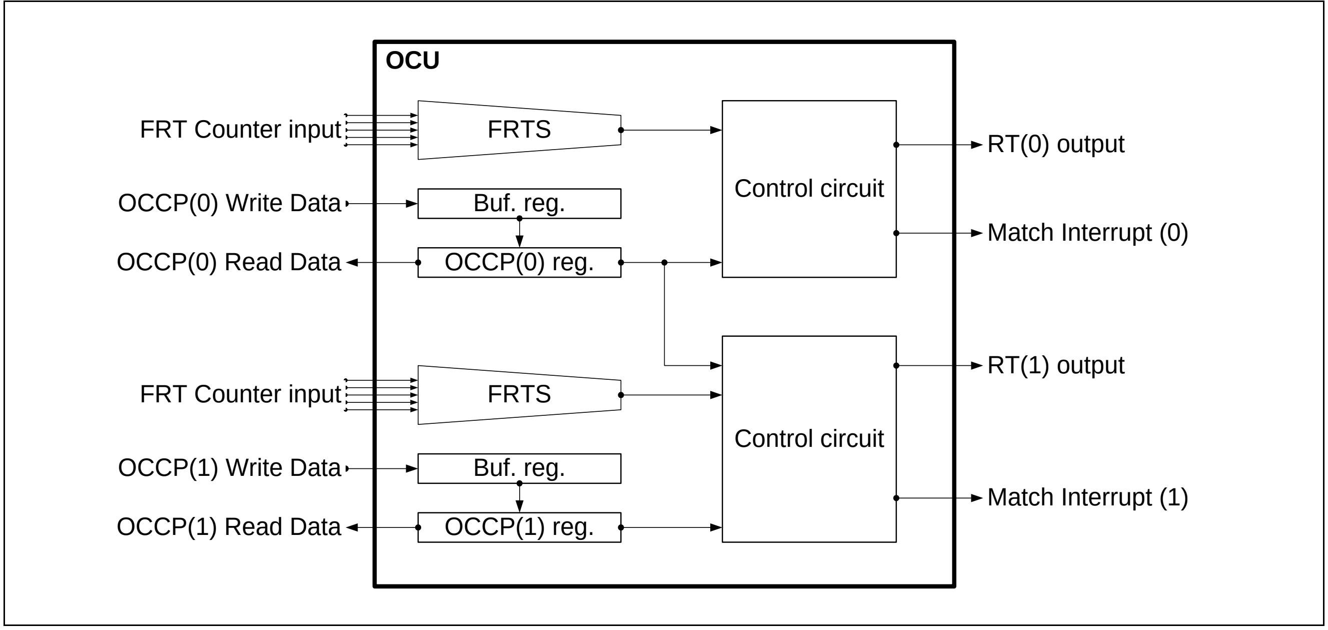
Figure 2-2 Configuration of FRT



■ OCU: 6channels (2ch. ×3)

- OCU is a function block that generates and outputs PWM signals based on the counter value of FRT. The signal names of PWM signals output by OCU are RT0 to RT5. These signals are output to LSI's external output pins via WFG.
- OCU consists of FRTS, compare value store register (OCCP register) and controller. The basic unit is in a 2-channel configuration with two sets of each circuit. Figure 2-3 shows the configuration of OCU.
  - FRTS is a circuit that selects the counter value of FRT to be connected to OCU for use.
  - The OCCP register specifies the timing of changing the PWM signal as the compare value for the FRT counter value. It has a buffer register to enable data to be written to the OCCP register asynchronously from FRT's count operation.
- The following processing can be achieved by instructing the controller from CPU.
  - FRT to be connected to OCU can be selected.
  - Whether to enable or disable OCU's operation can be specified.
  - The output level of the RT0 to RT5 signal can be specified directly when OCU's operation is disabled.
  - The output level of the RT0 to RT5 signal changes, if OCU's operation is enabled, the FRT counter value is compared with the value of the compare value store register and then it is detected that these values match. Signals with any cycle or pulse length can be output by setting the value of the OCCP register beforehand.
  - The following modes are available for the change conditions of the RT0 to RT5 signal and can be selected:
    - Up-count, 1-change mode
    - Up-count, 2-change mode
    - Up/Down-count, Active High mode
    - Up/Down-count, Active Low mode
  - Interrupts can be generated to CPU, when it is detected that the value of the OCCP register matches the FRT counter value.
  - Whether or not to use the buffer register of the OOCP register can be specified and the timing of transfer from the buffer register can be selected.
- Each MFT contains three of these OCU's and consists of a total of 6 compare registers, 6 output signal pins and 6 interrupt outputs (2-channel × 3-unit configuration).

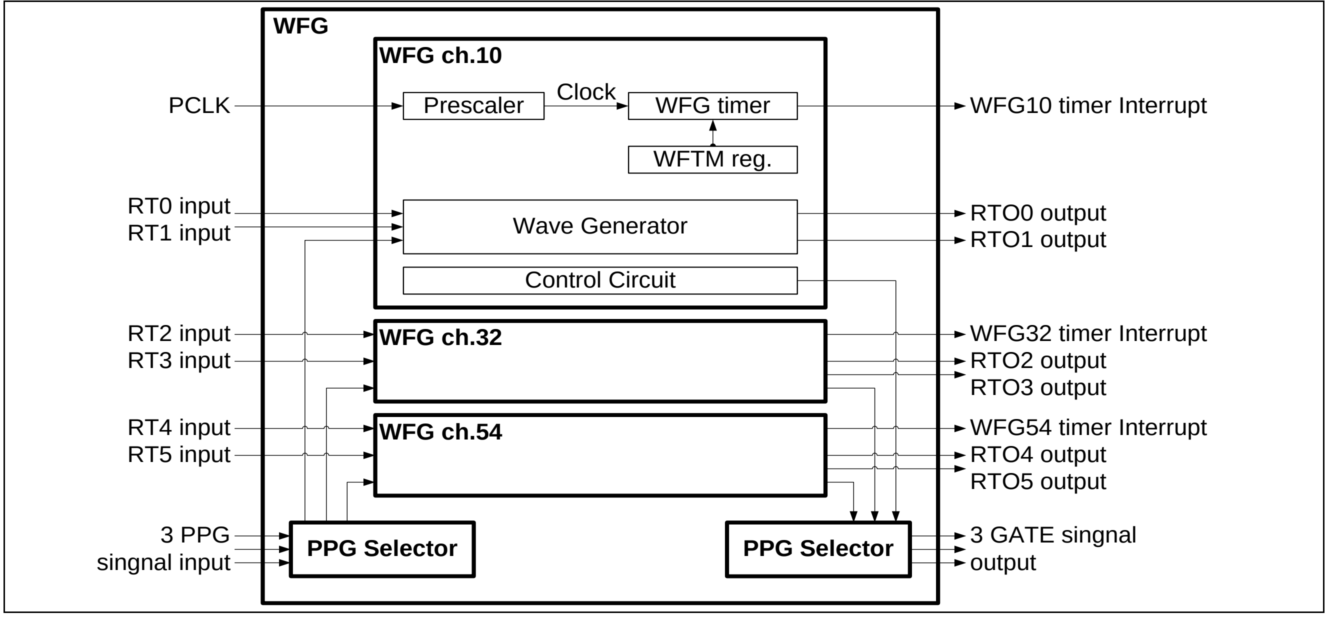
Figure 2-3 Configuration of OCU



■ WFG: 3channels

- WFG is a function block that is located at the back of OCU and generates signal waveforms for motor control from the RT0 to RT5 and PPG signals (PPG is located outside the multifunction timer).
- The signal outputs to LSI external pins from WFG are called "RTO0" to "RTO5". They are divided into blocks: the block that outputs RTO0 and RTO1 from RT0 and RT1; the block that outputs RTO2 and RTO3 from RT2 and RT3; and the block that outputs RTO4 and RTO5 from RT4 and RT5. They are called "WFG ch.10", "WFG ch.32" and "WFG ch.54", respectively.
- WFG consists of a clock prescaler, 16-bit timer (WFG timer), WFG timer initial value register (WFTM register), waveform generator, PPG timer unit selectors and controller. Figure 2-4 shows the configuration of WFG.
  - The clock prescaler divides the peripheral clock signal (PCLK) in LSI to generate the operating clock for the WFG timer.
  - The WFG timer is a timer circuit that counts the time set by the WFTM register and generates signal waveforms. In an operation mode that does not use a timer to generate a waveform, it can be used as a single reload timer, allowing interrupts to be generated regularly to CPU. Each WFG timer has one WFG timer interrupt output.
  - The WFTM register can be used to set any count time to the WFG timer.
  - The waveform generator is a block that generates LSI external output signals through waveform generation processing based on the RT0 to RT5 signals from OCU, signals from PPG and the count state of the WFG timer.
  - The PPG timer unit selector is a circuit that selects the PPG timer unit to be used by WFG. It selects the output destination of the instruction signal (GATE signal) for PPG activation and the PPG output signal.
- The following processing can be achieved by instructing the controller from CPU.
  - The division ratio of the clock prescaler can be selected.
  - The following modes are available and can be selected for wave generation.
    - Through mode
    - RT-PPG mode
    - Timer-PPG mode
    - RT-dead timer mode
    - PPG-dead timer mode
  - GATE signal can be output to instruct PPG to start up.
  - Output polarity can be reversed in RT-dead timer mode and PPG timer mode.

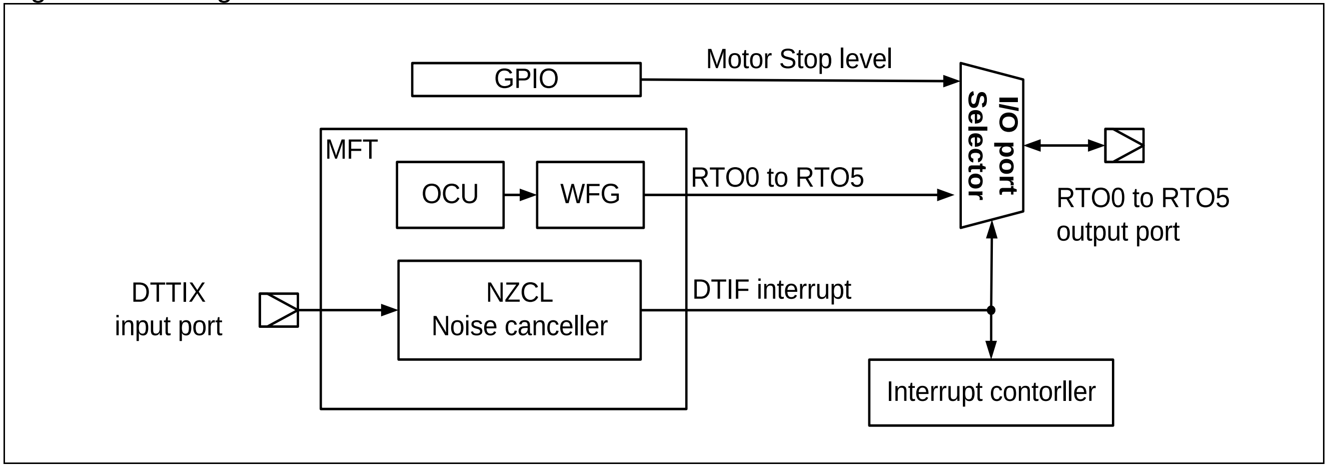
Figure 2-4 Configuration of WFG



■ NZCL

- NZCL is a function block that performs noise cancellation to the external interrupt input signal (DTTIX signal) for emergency shutdown of the motor and generates DTIF interrupts to CPU.
- NZCL consists of a noise canceller and controller.
- It can be switched to the state of the GPIO port which is also used for WFG's external output signals (RTO0 to RTO5) using the selection function of the I/O port block while DTIF interrupt is being generated. Emergency shutdown of the motor can be performed by setting the I/O state of the GPIO port to the Motor Stop level.
- Figure 2-5 shows the configuration of NZCL and I/O port selector.

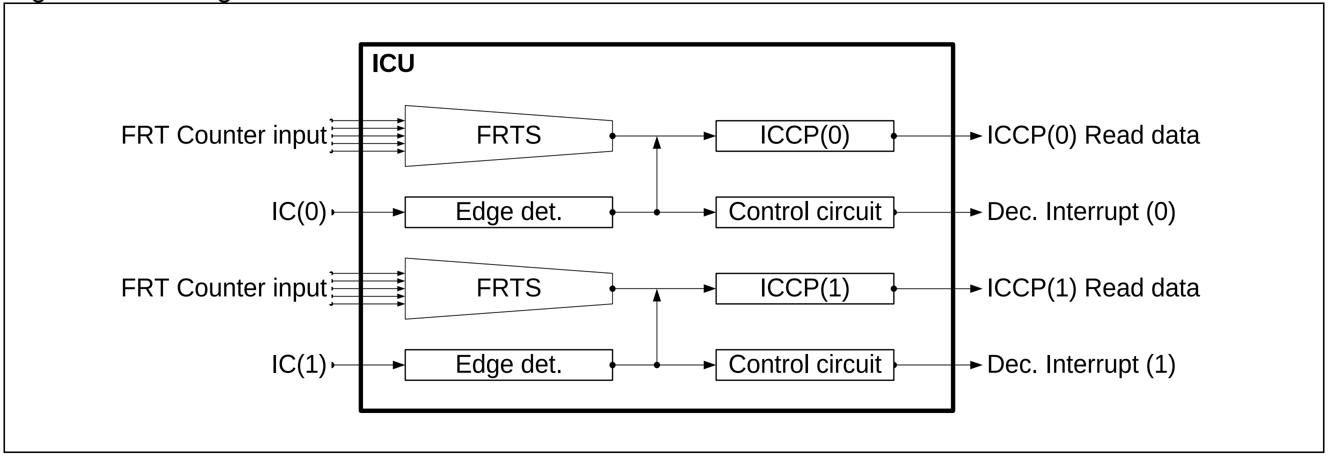
Figure 2-5 Configuration of NZCL and I/O Port Selector



■ ICU: 4channels (2ch. ×2)

- ICU is a function block that captures the FRT count value and generates an interrupt to CPU when a valid edge is detected at an external input pin signal.
- ICU consists of FRTS, edge detector, 16-bit capture register and control register. Its basic unit is in a 2-channel configuration with two sets of each circuit. Figure 2-6 shows the configuration of ICU.
  - FRTS is a circuit that selects the counter value of the FRT to be connected to ICU for use.
  - The edge detector is a circuit that detects the valid edge of the input signal.
  - The ICCP register captures the timing of changing the input signal as FRT's count value.
- The following processing can be achieved by instructing the controller from CPU.
  - FRT to be connected to ICU can be selected.
  - The valid edge of the input signal can be selected from rising edge, falling edge and both edges.
  - Whether to enable or disable ICU's operation can be specified.
  - An interrupt can be generated to CPU when the valid edge is detected and the capture operation is performed.
- Each MFT contains 2 ICU's and consists of a total of 4 external input pins and 4 capture registers (2-channel × 2-unit configuration). LSI external input signals to ICU are called "IC0" to "IC3".
- Some ICU input signals can be switched to LSI's internal signals for use, other than LSI external pins, using the selection function of the I/O port block (see "2.3 I/O Pins of Multifunction Timer Unit" for more details).

Figure 2-6 Configuration of ICU



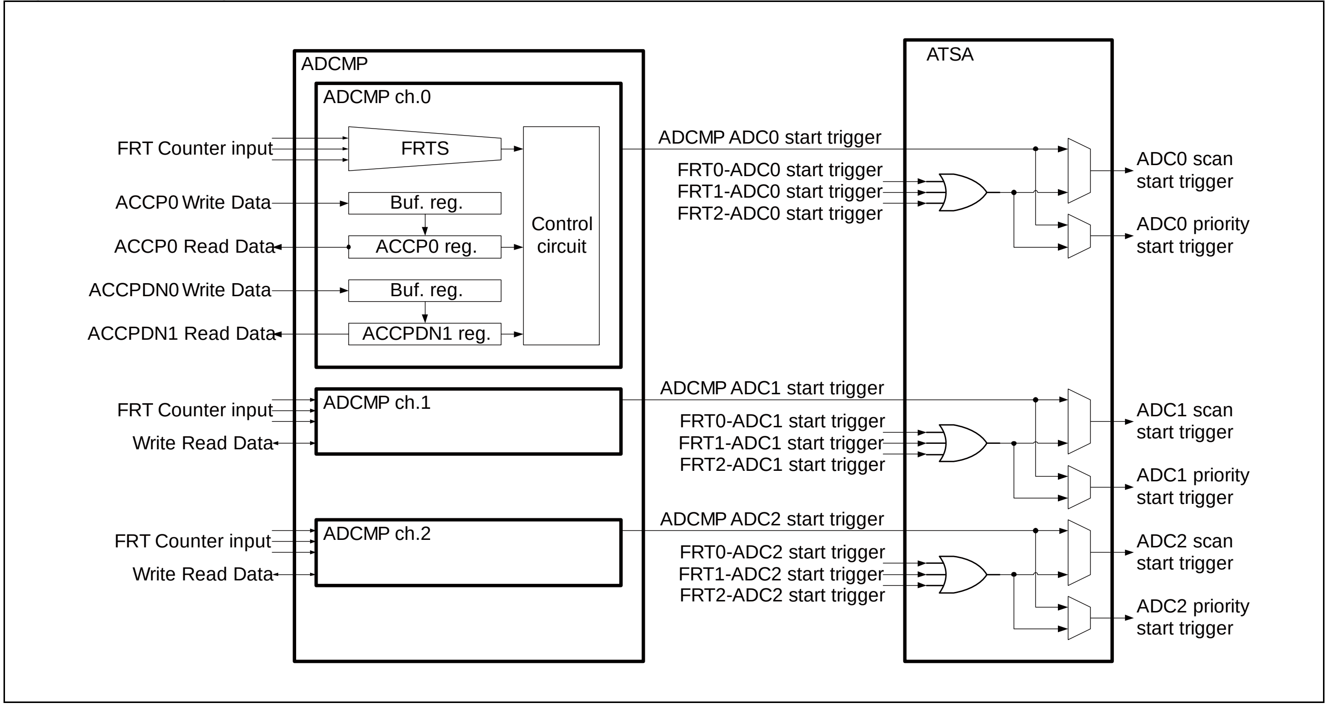
■ ADCMP: 3channels

- ADCMP is a function block that generates AD conversion start signals at any timing of FRT cycle.
- ADCMP is in a 3-channel configuration with each corresponding to one of the 3 units of ADC mounted.
- ADCMP consists of FRTS, two 16-bit compare registers (ACCP register and ACCPDN register) and control register. Figure 2-7 shows the configuration of ADCMP and ATSA.
  - FRTS is a circuit that selects the counter value of the FRT to be connected to ADCMP for use.
  - The ACCP register and ACCPDN register specify the timing of starting AD conversion as the compare value of FRT's counter value. Each has a buffer register so that writing to the ACCP register and ACCPDN register can be done asynchronously from FRT's count operation.
- The following processing can be achieved by instructing the controller from CPU.
  - FRT to be connected to ADCMP can be selected.
  - Whether to enable or disable ADCMP operation can be specified.
  - The timing of starting AD conversion can be set with a specified FRT count direction.
  - Whether or not to use the ACCP register and ACCPDN's buffer register can be specified and the timing of transferring from the buffer register can be selected.

■ ATSA: 3channels

- ATSA is a function block that selects and outputs ADC's start signals based on the value of the control register.
- ATSA consists of a logic OR circuit for AD start signals and a circuit that selects the AD start signals from ADCMP.
- The following is processed.
  - Start signals from FRT to each ADC undergo logic OR operation for each corresponding ADC unit.
  - The above ADC start signal and other ADC start signal from ADCMP are selected by register setting.
  - There are two types of ADC start triggers: scan start triggers and priority start triggers. The configuration allows each to be selected and output.
- ATSA is configured to support each start trigger of three mounted ADC units. It outputs 6 ADC start signals.

Figure 2-7 Configuration of ADCMP and ATSA



2.3. I/O Pins of Multifunction Timer Unit

■ Correspondence with LSI External I/O Pins

Of all the I/O signals illustrated in the block diagrams, Table 2-1 shows a list of correspondence between I/O pins of MFT unit and LSI external I/O pins. In this series, some models have more than one MFT unit. Therefore, LSI pin names are composed of I/O pin names shown in the block diagrams plus MFT's unit number (0, 1, 2). It should be noted that descriptions in this chapter are based on the pin names shown in the block diagrams.

Table 2-1 Correspondence Table for I/O Pins of MFT Unit and LSI External I/O Pins

Name of MFT Unit Pin (Pin Name in Block Diagram)	Function	Name of LSI Pin		
		MFT-unit0	MFT-unit1	MFT-unit2
FRCK	FRT external input clock	FRCK0	FRCK1	FRCK2
DTTIX	Motor emergency shutdown interrupt input	DTTI0X	DTTI1X	DTTI2X
RTO0	WFG→PWM output ch.0	RTO00	RTO10	RTO20
RTO1	WFG→PWM output ch.1	RTO01	RTO11	RTO21
RTO2	WFG→PWM output ch.2	RTO02	RTO12	RTO22
RTO3	WFG→PWM output ch.3	RTO03	RTO13	RTO23
RTO4	WFG→PWM output ch.4	RTO04	RTO14	RTO24
RTO5	WFG→PWM output ch.5	RTO05	RTO15	RTO25
IC0	ICU input ch.0	IC00	IC10	IC20
IC1	ICU input ch.1	IC01	IC11	IC21
IC2	ICU input ch.2	IC02	IC12	IC22
IC3	ICU input ch.3	IC03	IC13	IC23

ICU's input pins can be switched with the following LSI internal signals, in addition to the external pin inputs, using the selector function of the I/O port block.

- SYNC signal when the LYN function of the multifunction serial block is used
- Internal CR oscillator/oscillation frequency trimming input signal

For details, see the chapter "I/O PORT" in "Peripheral Manual".

■ Interrupt Signal Outputs

Of all the I/O signals illustrated in the block diagrams, Table 2-2 shows a list of interrupt signals generated from the MFT unit. Any model that contains more than one MFT unit has interrupt outputs equivalent to the number of mounted MFT units.

Table 2-2 List of Interrupt Signals Generated from MFT Unit

Generation Block	Interrupt Type
FRT ch.0	Zero value detection interrupt
FRT ch.1	Zero value detection interrupt
FRT ch.2	Zero value detection interrupt
FRT ch.0	Peak value detection interrupt
FRT ch.1	Peak value detection interrupt
FRT ch.2	Peak value detection interrupt
OCU ch.0	Match detection interrupt
OCU ch.1	Match detection interrupt
OCU ch.2	Match detection interrupt
OCU ch.3	Match detection interrupt
OCU ch.4	Match detection interrupt
OCU ch.5	Match detection interrupt
ICU ch.0	Input signal edge detection interrupt
ICU ch.1	Input signal edge detection interrupt
ICU ch.2	Input signal edge detection interrupt
ICU ch.3	Input signal edge detection interrupt
NZCL	DTIF interrupt (emergency motor shutdown interrupt)
WFG ch.10	WFG timer 10 interrupt
WFG ch.32	WFG timer 32 interrupt
WFG ch.54	WFG timer 54 interrupt

■ Other I/O Signals

Of all the I/O signals illustrated in the block diagrams, the following section explains the other signals.

● PCLK

This is an LSI internal peripheral clock signal used in the MFT unit. It uses the clock signal of the APB bus to be connected. FRT (when the LSI internal peripheral clock is selected) and the WFG timer operate based on the count clock divided from PCLK.

● FRT input and FRT output of external MFT

A model containing more than one MFT unit can use FRT count output for the other MFT. This connection configuration allows OCU and ICU mounted separately on multiple MFT units to be interlocked by a single FRT.
(A model containing 2 MFT units can output 12 channels of PWM simultaneously. A model containing 3 MFT units can output 18 channels of PWM simultaneously.
For details, see "6.1 Connection of Model Containing Multiple MFT's".

● GATE signal / PPG signal

GATE signal is PPG's start signal that is output from MFT and input to PPG. PPG signal is output from PPG and input to MFT. PPG units to be connected for these signals vary depending on the mounted MFT unit. For details of their connection, see "6.1 Connection of Model Containing Multiple MFT's".

● AD conversion start signal

A total of 6 AD conversion start signals are output: scan start signals and priority start signals for each of the three units of ADC.

In models containing more than one MFT unit, start signals undergo logic OR for each ADC unit and are used in each ADC unit. For details, see the chapter "A/D Converter" in "Analog Macro Part".

3. Operations of Multifunction Timer

This section provides examples of operations of the multifunction timer and explains its setting procedures.

3.1. Example of Operation of Multifunction Timer - 1

"Example of Operation of Multifunction Timer - 1" explains the cases where each function block is operated in the following modes:

FRT	:	Up-count mode, without interrupt
OCU	:	Up-count mode (1-change), with interrupt
WFG	:	RT-PPG mode, generation of GATE signal, superimposition of PPG signal
ICU	:	Rising edge detection mode, with interrupt

■ Time Chart

Figure 3-1 Time Chart of Main Registers and I/O Signals of Each Block

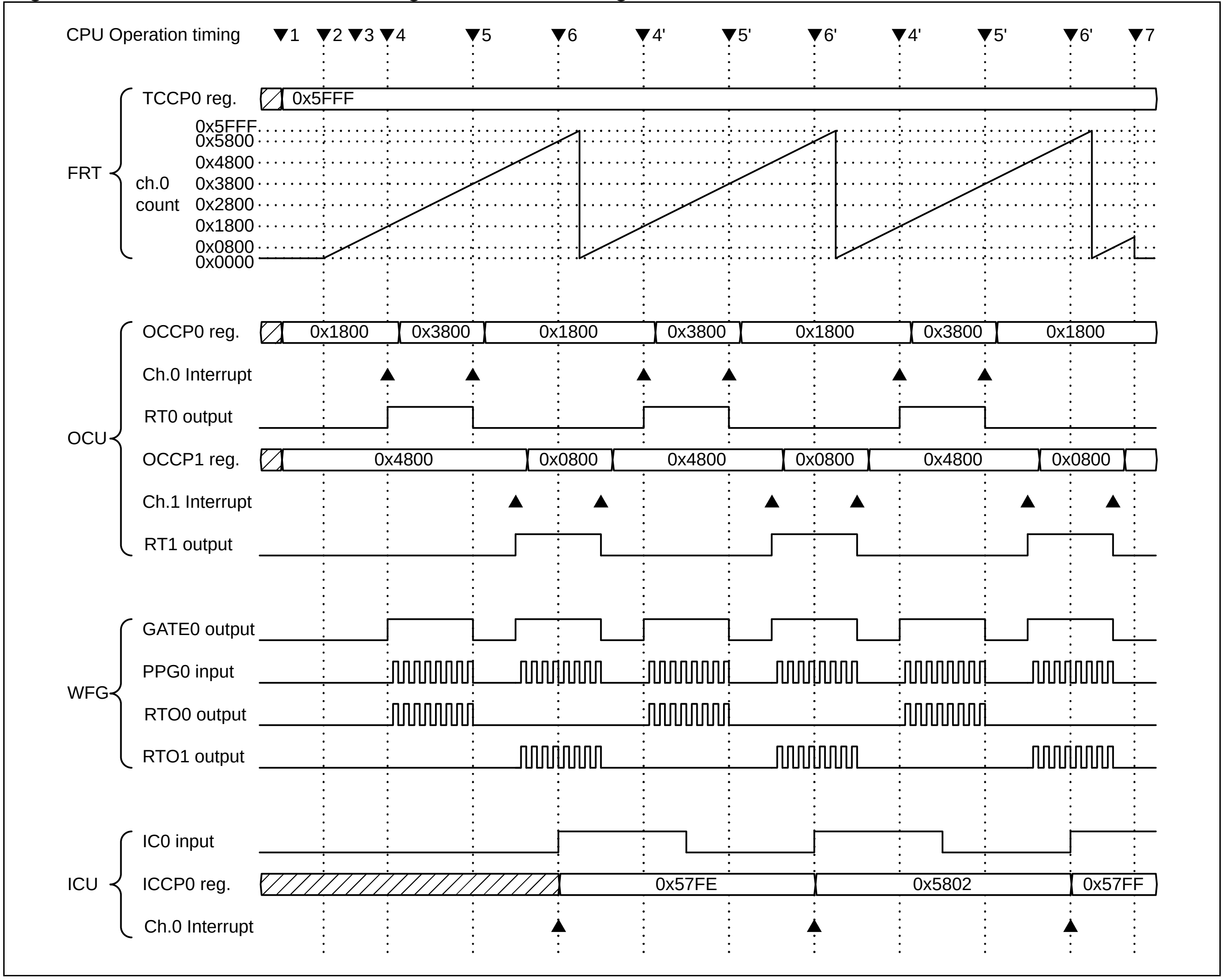


Figure 3-1 shows the time chart of main registers and I/O signals for each block. From top to bottom, the figure indicates CPU operation, FRT operation, OCU operation, WFG operation and ICU operation.

The following section explains the operation of each function block and what is controlled from CPU at the timing 1 to 7. It also shows specific examples of CPU register settings at each of the timings. For details of register settings, see "4 Registers of Multifunction Timer". It should be noted that in addition to the above, LSI I/O port block, interrupt control block and PPG must be set separately.

■ Operation of FRT, OCU

Timing of CPU Operation Timing 1 in Figure 3-1

- Set Up-count mode operation to FRT-ch.0 (TCSA0 register write).
- Set an operating cycle to FRT-ch.0 (TCCP0 register write). In this example, "0x5FFF" is set. When the prescaler is set to 1/128 and PCLK to 40MHz, the count cycle of FRT is 78.6432ms.
- Connect and set FRT-ch.0 to OCU-ch.0/ch.1 (OCFS10 register write).
- Set OCU-ch.0/ch.1 to Up-count mode (1-change) operation. Also specify the initial output level of output signals (RT0, RT1) (OCSA10, OCSB10 and OCSC register write).
- Set the timing of changing the output signal (RT0) for OCU-ch.0 (OCCP0 register write). In this example, "0x1800" is set. The value is written to the buffer register, and then transferred to the OCCP0 register.
- Set the timing of changing the output signal (RT1) for OCU-ch.1 (OCCP1 register write). In this example, "0x4800" is set. The value is written to the buffer register, and then transferred to the OCCP1 register.

Timing of CPU OPeratoion Timing 2 in Figure 3-1

- Instruct FRT-ch.0 to start count operation (TCSA0 register write).
- In Up-count mode, FRT-ch.0 starts counting from "0x0000" and continues the Up-count operation until the TCCP value is reached (=0x5FFF), as shown in the figure. Then, it returns to "0x0000" and continues counting.

Timing of CPU Operation Timing 3 in Figure 3-1

- Instruct OCU-ch.0/ch.1 to enable the operation (OCSA10 register write).

Timing of CPU Operation Timing 4 in Figure 3-1

- When OCU-ch.0 detects that the value of the FRT counter has reached "0x1800" and matched the setting value of OCCP0, it changes the output signal (RT0) from the Low level to High level. It also generates an interrupt to CPU.
- CPU determines that an interrupt has been generated from OCU-ch.0, because "1" is set to the match detection flag of OCU-ch.0 (OCSA10 register read).
- Update the timing of changing the output signal (RT0) for OCU-ch.0 to "0x3800" (OCCP0 register write).
- CPU clears the match detection flag and returns from the interrupt (OCSA10 register write).

Timing of CPU Operation Timing 5 in Figure 3-1

- When OCU-ch.0 detects that the value of the FRT counter matches the value of OCCP0, it changes the output signal (RT0) from the High level to Low level. It also generates an interrupt to CPU.
- CPU identifies the interrupt from OCU-ch.0 (OCSA10 register read).
- Update the OCCP0 register of OCU-ch.0 to "0x1800" (OCCP0 register write).
- CPU clears the match detection flag and returns from the interrupt (OCSA10 register write).

After that, repeat Operations 4 and 5 so that PWM waveform in the FRT cycle can be achieved for the RT0 output signal, as shown in the figure. Similarly, PWM output waveform can also be achieved for the RT1 by updating the value of the OCCP1 register every time an interrupt occurs.

■ Operation of WFG

Timing of CPU Operation Timing 1 in Figure 3-1

- Set RT-PPG mode operation to WFG-ch.10 (WFSA10 register write).

Timing of CPU Operation Timing 4 in Figure 3-1

- When the RT0 signal from OCU-ch.0 is changed to the High level, WFG asserts the GATE signal (GATE0) and instructs PPG-ch.0 to start.
- When the GATE signal is asserted, PPG-ch.0 starts the output of the PPG signal (PPG0).
- WFG superimposes and outputs the PPG signal to RTO0 while the RT0 signal remains at the High level.

Timing of CPU Operation Timing 5 in Figure 3-1

- When the RT0 signal is changed to the Low level, WFG deasserts the GATE signal and gives a stop instruction.
- PPG-ch.0 changes the PPG signal to the Low level and stops the output.
- WFG changes the RTO0 signal to the Low level and stops the output.

WFG performs the same operation to the RT1 signal from OCU-ch.1, and superimposes and outputs the PPG signal to RTO1. DC chopper control waveform, as shown in the figure, can be output to RTO0/RTO1 by using the WFG function.

■ Operation of ICU

Timing of CPU Operation Timing 1 in Figure 3-1

- Connect and set FRT-ch.0 to ICU-ch.0/ch.1 (ICFS10 register write).
- Set rising edge detection operation of the input signal to ICU-ch.0 (ICSA10 register write).

Timing of CPU Operation Timing 6 in Figure 3-1

- When a rising edge is detected at the input signal (IC0), ICU-ch.0 stores FRT's count value to the ICCP0 register. It also generates an interrupt to CPU.
- CPU determines that an interrupt has been generated from ICU-ch.0, because "1" is set to the valid edge detection flag of ICU-ch.0 (ICSA10 register read).
- CPU captures the position of the rising edge of the signal (ICCP0 register read).
- CPU clears the valid edge detection flag and returns from the interrupt (ICSA10 register write).

■ Completion of Processing

Timing of CPU Operation Timing 7 in Figure 3-1

The processing at Timing 7 indicates the procedure for completing the output of the PWM signal.

- Disable the operation of OCU-ch.0 and ch.1 (OCSA10 register write).
- Set the level of the output signals (RT0, RT1) for OCU-ch.0 and ch.1 (OCSB10 register write).
- Disable the operation of ICU-ch.0 (ICSA10 register write).
- When the output of OCU stops, WFG does not perform its operation.
- Instruct FRT-ch.0 to stop the count operation (TCSA0 register write).
- Set "0x0000" to FRT's count value (TCDT0 register write).

■ Processing of Other Channels

The above example explained the operation with 2 channels of OCU, one channel of WFG and one channel of ICU. If OCU-6ch, WFG-3ch, and ICU-3ch are connected to the same FRT to perform interlocking control, three-phase motor control can be achieved.

■ Details of Register Settings

Register settings in Example of Operation of Multifunction Timer - 1 are detailed below.

The meanings of the symbols in the table are as follows:

Operation	HW	Half-word write access
	BW	Byte write access
	HR	Half-word read access
	BR	Byte read access
Value	NM	Indicates either writing the same value as the register value that has already been set or reading from the register to write the original value (No Modify).
	1(RMW)	Indicates writing "1", if register clear is not intended. In the case of update by RMW access (see "6.2 Treatment of Event Detect Register and Interrupt"), it indicates that the read value can be written back.
	Other	Indicates setting bits of other channels and no relation to this explanatory example.
	DC	Indicates no relation to the read value (Don't Care).

Table 3-1 Example of Operation 1 – Register Settings 1

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
1	FRT	TCSA0	HW	CLK[3:0]	0111	Clock division prescaler setting: 1/128
				SCLR	0	Soft clear: Do nothing
				MODE	0	Count mode setting: Up-count mode
				STOP	1	FRT count operation: Stop counting
				BFE	1	TCCP buffer function: Enable
				ICRE	0	Peak value detection interrupt: Disable
				ICLR	0	Peak value detection: Clear
				Reserved	000	-
				IRQZE	0	Zero value detection interrupt: Disable
				IRQZF	0	Zero value detection: Clear
				ECKE	0	Selection of clock used: Internal clock
		TCCP0	HW	TCCP	0x5FFF	Set FRT cycle
	OCU	OCFS10	BW	FSO0[3:0]	0000	FRT connected to ch.0: FRT ch.0
				FSO1[3:0]	0000	FRT connected to ch.1: FRT ch.0
		OCSA10	BW	CST0	0	ch.0 operation state: Operation disable
				CST1	0	ch.1 operation state: Operation disable
				BDIS0	1	ch.0 OCCP buffer function: Disable
				BDIS1	1	ch.1 OCCP buffer function: Disable
				IOE0	1	ch.0 interrupt: Enable
				IOE1	1	ch.1 interrupt: Enable
				IOP0	0	ch.0 match detection: Clear
				IOP1	0	ch.1 match detection: Clear
		OCSB10	BW	OTD0	0	RT0 output level initial setting: Low
				OTD1	0	RT1 output level initial setting: Low
				Reserved	00	-
				CMOD	0	ch.0/ch.1 operation mode: Up-count (1-change)
				BTS0	0	ch.0 buffer transfer: don't care
				BTS1	0	ch.1 buffer transfer: don't care
				Reserved	0	-
		OCSC	BW	MOD0	0	ch.0 operation mode: Up-count (1-change)
				MOD1	0	ch.1 operation mode: Up-count (1-change)
				MOD2	Other	ch.2 operation mode:
				MOD3	Other	ch.3 operation mode:
				MOD4	Other	ch.4 operation mode:
				MOD5	Other	ch.5 operation mode:
				Reserved	00	-
		OCCP0	HW	OCCP	0x1800	Specify ch.0 change timing
		OCCP1	HW	OCCP	0x4800	Specify ch.1 change timing

Table 3-2 Example of Operation 1 – Register Settings 2

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
1	WFG	WFSA10	HW	DCK[2:0]	000	Clock division prescaler setting: 1/1 (don't care)
				TMD[2:0]	001	Operation mode: Select RT-PPG mode
				GTEN[1:0]	11	Gate signal generation: RT0, RT1 signal logic OR
				PSEL[1:0]	00	Connecting PPG : PPG0
				PGEN[1:0]	11	PPG reflection: Logic AND of PPG signal to RTO0/RTO1 signals
				DMOD	0	Output polarity: don't care
				Reserved	000	-
	ICU	ICFS10	BW	FSI0[3:0]	0000	FRT connected to ch.0: FRT ch.0
				FSI1[3:0]	Other	FRT connected to ch.1:
		ICSA10	BW	EG0[1:0]	01	ch.0 operation state: Operation enabled, rising edge
				EG1[1:0]	Other	ch.1 operation state:
				ICE0	1	ch.0 interrupt: Enable
				ICE1	Other	ch.1 interrupt:
				ICP0	0	ch.0 edge detection: Clear
				ICP1	Other	ch.1 edge detection:
2	FRT	TCSA0	HW	CLK[3:0]	NM	Clock division prescaler setting:
				SCLR	NM	Soft clear:
				MODE	NM	Count mode setting:
				STOP	0	FRT count operation: Start counting
				BFE	NM	TCCP buffer function:
				ICRE	NM	Peak value detection interrupt:
				ICLR	1(RMW)	Peak value detection: Do nothing
				Reserved	NM	-
				IRQZE	NM	Zero value detection interrupt:
				IRQZF	1(RMW)	Zero value detection: Do nothing
				ECKE	NM	Selection of clock used: Internal clock
3	OCU	OCSA10	BW	CST0	1	ch.0 operation state: Operation enabled
				CST1	1	ch.1 operation state: Operation enabled
				BDIS0	NM	ch.0 OCCP buffer function:
				BDIS1	NM	ch.1 OCCP buffer function:
				IOE0	NM	ch.0 interrupt:
				IOE1	NM	ch.1 interrupt:
				IOP0	1	ch.0 match detection: Do nothing
				IOP1	1	ch.1 match detection: Do nothing

Table 3-3 Example of Operation 1 – Register Settings 3

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
4	OCU	OCSA10	BR	CST0	DC	ch.0 operation state:
				CST1	DC	ch.1 operation state:
				BDIS0	DC	ch.0 OCCP buffer function:
				BDIS1	DC	ch.1 OCCP buffer function:
				IOE0	DC	ch.0 interrupt:
				IOE1	DC	ch.1 interrupt:
				IOP0	1	ch.0 match detection: Match detected
				IOP1	0	ch.1 match detection: Match not detected
		OCCP0	HW	OCCP0	0x3800	Specify ch.0 change timing
		OCSA10	BW	CST0	NM	ch.0 operation state:
				CST1	NM	ch.1 operation state:
				BDIS0	NM	ch.0 OCCP buffer function:
				BDIS1	NM	ch.1 OCCP buffer function:
				IOE0	NM	ch.0 interrupt:
				IOE1	NM	ch.1 interrupt:
				IOP0	0	ch.0 match detection: Flag cleared
				IOP1	1(RMW)	ch.1 match detection: Do nothing
5	OCU	OCSA10	BR	CST0	DC	ch.0 operation state:
				CST1	DC	ch.1 operation state:
				BDIS0	DC	ch.0 OCCP buffer function:
				BDIS1	DC	ch.1 OCCP buffer function:
				IOE0	DC	ch.0 interrupt:
				IOE1	DC	ch.1 interrupt:
				IOP0	1	ch.0 match detection: Match detected
				IOP1	0	ch.1 match detection: Match not detected
		OCCP0	HW	OCCP0	0x1800	Specify ch.0 change timing
		OCSA10	BW	CST0	NM	ch.0 operation state:
				CST1	NM	ch.1 operation state:
				BDIS0	NM	ch.0 OCCP buffer function:
				BDIS1	NM	ch.1 OCCP buffer function:
				IOE0	NM	ch.0 interrupt:
				IOE1	NM	ch.1 interrupt:
				IOP0	0	ch.0 match detection: Flag cleared
				IOP1	1(RMW)	ch.1 match detection: Do nothing

Table 3-4 Example of Operation 1 – Register Settings 4

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
6	ICU	ICSA10	BR	EG0[1:0]	DC	ch.0 operation state:
				EG1[1:0]	DC	ch.1 operation state:
				ICE0	DC	ch.0 interrupt:
				ICE1	DC	ch.1 interrupt:
				ICP0	1	ch.0 edge detection: Edge detected
				ICP1	0	ch.1 edge detection: Edge not detected
		ICCP0	HW	ICCP0	0x57FE	Capture ch.0 capture value
		ICSA10	BW	EG0[1:0]	NM	ch.0 operation state:
				EG1[1:0]	NM	ch.1 operation state:
				ICE0	NM	ch.0 interrupt:
				ICE1	NM	ch.1 interrupt:
				ICP0	0	ch.0 edge detection: Clear
				ICP1	1(RMW)	ch.1 edge detection: Do nothing
7	OCU	OCSA10	BW	CST0	0	ch.0 operation state: Disabled
				CST1	0	ch.1 operation state: Disabled
				BDIS0	NM	ch.0 OCCP buffer function:
				BDIS1	NM	ch.1 OCCP buffer function:
				IOE0	NM	ch.0 interrupt:
				IOE1	NM	ch.1 interrupt:
				IOP0	1	ch.0 match detection: Do nothing
				IOP1	1	ch.1 match detection: Do nothing
		OCSB10	BW	OTD0	0	RT0 output level: Low
				OTD1	0	RT1 output level: Low
				Reserved	NM	-
				CMOD	NM	ch.0/ch.1 operation mode:
				BTS0	NM	ch.0 buffer transfer:
				BTS1	NM	ch.1 buffer transfer:
				Reserved	NM	-
	ICU	ICSA10	BW	EG0[1:0]	00	ch.0 operation state: Operation disabled
				EG1[1:0]	00	ch.1 operation state: Operation disabled
				ICE0	NM	ch.0 interrupt:
				ICE1	NM	ch.1 interrupt:
				ICP0	1	ch.0 edge detection: Do nothing
				ICP1	1	ch.1 edge detection: Do nothing
	FRT	TCSA0	HW	CLK[3:0]	NM	Clock division prescaler setting:
				SCLR	1	Soft clear: Initialize FRT
				MODE	NM	Count mode setting:
				STOP	1	FRT count operation: Stop counting
				BFE	NM	TCCP buffer function:
				ICRE	NM	Peak value detection interrupt:
				ICLR	1	Peak value detection: Do nothing
				Reserved	NM	-
				IRQZE	NM	Zero value detection interrupt:
				IRQZF	1	Zero value detection: Do nothing
				ECKE	NM	Selection of clock used:
		TCDT0	HW	TCDT	0x0000	Initialize FRT count value

3.2. Example of Operation of Multifunction Timer - 2

"Example of Operation of Multifunction Timer - 2" explains the cases where each function block is operated in the following modes:

FRT	: Up/Down-count mode, with interrupt
OCU	: Up/Down-count mode (Active High), without interrupt
WFG	: RT-dead timer mode (Active High)
ADCMP/ATSA	: Instruct ADCunit0 to start scan conversion under the match condition for Up-counting

■ Time Chart

Figure 3-2 Time Chart of Main Registers and I/O Signals of Each Block

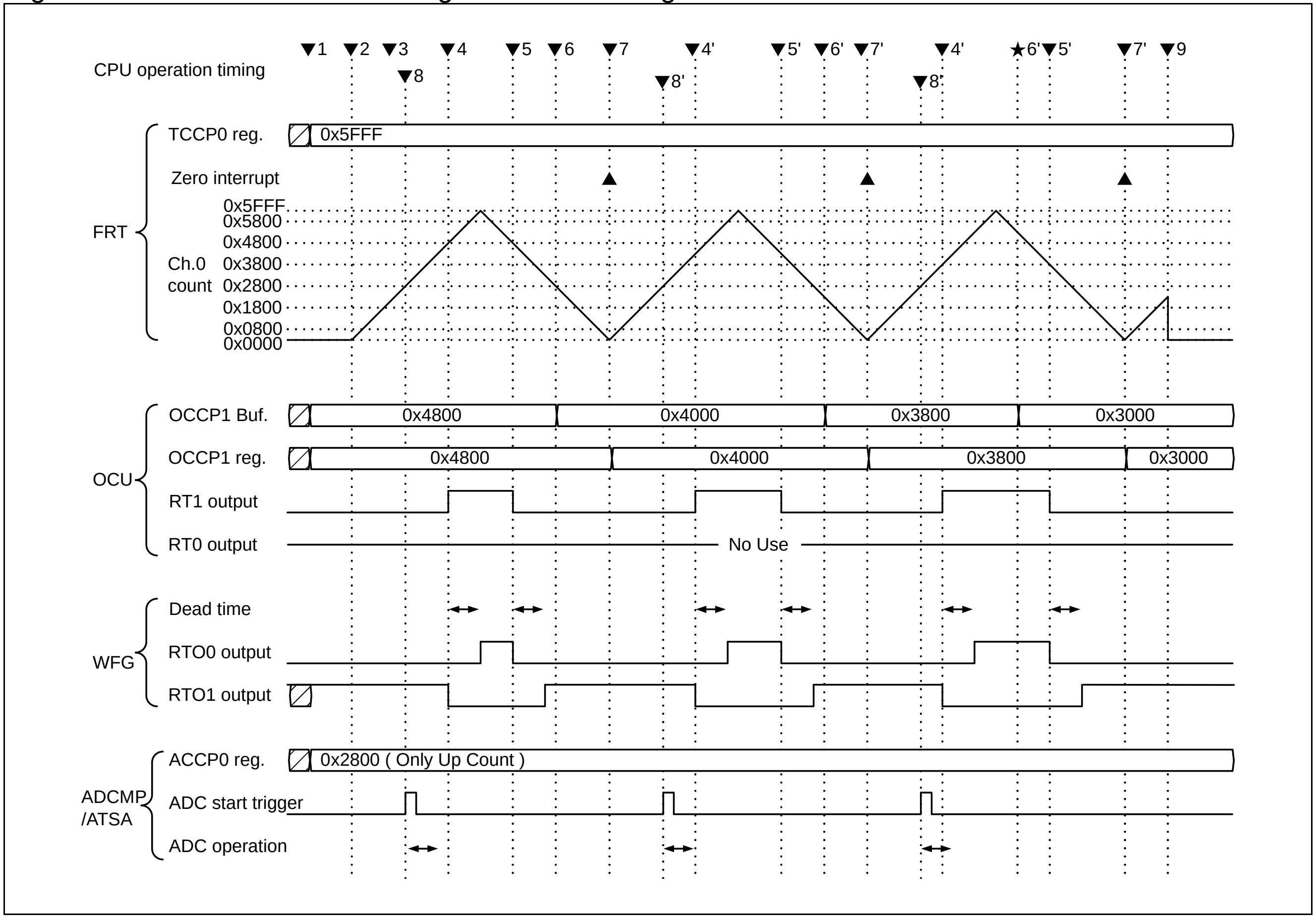


Figure 3-2 shows the time chart of main registers and I/O signals for each block. From top to bottom, the figure indicates CPU operation, FRT operation, OCU operation, WFG operation and ADCMP operation.
The following section explains the operation of each function block and what is controlled from CPU at the timing 1 to 9. It also shows specific examples of CPU register settings at each of the timings. For details of register settings, see "4 Registers of Multifunction Timer". It should be noted that in addition to the above, LSI I/O port block, interrupt control block and ADC must be set separately.

■ Operation of FRT, OCU

Timing of CPU Operation Timing 1 in Figure 3-2

- Set Up/Down-count mode operation to FRT-ch.0 (TCSA0 register write).
- Set an operating cycle to FRT-ch.0 (TCCP0 register write). In this example, "0x5FFF" is set. When the FRT prescaler is set to 1/4 and PCLK to 40MHz, the count cycle of FRT is 4.915ms.
- Connect and set FRT-ch.0 to OCU-ch.1 (OCFS10 register write).
- Set OCU-ch.1 to Up/Down-count mode (Active High) operation. Also specify the initial output level of the output signal (RT1) (OCSA10 register, OCSB10 register and OCSC register write).
- Set the timing of changing the output signal (RT1) for OCU-ch.1 (OCCP1 register write). In this example, "0x4800" is set. The value is written to the buffer register, and then transferred to the OCCP1 register.

Timing of CPU Operation Timing 2 in Figure 3-2

- Instruct FRT-ch.0 to start count operation (TCSA0 register write).
- In Up/Down-count mode, FRT-ch.0 starts counting from "0x0000" and continues the Up-count operation until the TCCP value is reached (=0x5FFF), as shown in the figure. After that, switch over the count direction and perform the Down-count operation until the TCCP value (=0x5FFF) is reached. Then, continue counting.

Timing of CPU Operation Timing 3 in Figure 3-2

- Instruct OCU-ch.0/ch.1 to enable the operation (OCSA10 register write).

Timing of CPU Operation Timing 4 in Figure 3-2

- When OCU-ch.1 detects that the value of the FRT counter has reached "0x4800" and matched the setting value of OCCP1 during the Up-count operation, it changes the output signal (RT1) from the Low to High level.

Timing of CPU Operation Timing 5 in Figure 3-2

- When OCU-ch.1 detects that the value of the FRT counter has reached "0x4800" and matched the setting value of OCCP1 during the Down-count operation, it changes the output signal (RT1) from the High to Low level.

Timing of CPU Operation Timing 6 in Figure 3-2

- CPU sets the timing of changing the output signal (RT1) for OCU-ch.1 in the next FRT cycle (OCCP1 register write). As OCCP buffer function is valid and Zero value detection transfer mode is selected, the written value is first stored in the buffer register. Then, when the value of the FRT counter reaches the Zero value (Timing 7), the written value is transferred to the OCCP1 register and reflected upon OCU output. For this reason, even if writing takes place before Timing 5, as indicated by ★ in the figure, it does not affect the timing of changing the output signal (RT1).

Timing of CPU Operation Timing 7 in Figure 3-2

- When the count value reaches "0x0000", FRT-ch.0 generates Zero value detection interrupt to CPU (Zero value detection interrupt is not generated at Timing 3).
- CPU determines that an interrupt has been generated from FRT-ch.0, because "1" is set to the Zero value detection flag of FRT-ch.0 (TCSA0 register read).
- CPU clears the Zero value detection flag and returns from the interrupt (TCSA0 register write).

After that, repeat Operations 4 to 7 so that PWM waveform with the peak value of the FRT counter symmetrical about the RT1 output can be achieved.

■ WFG's Operation

Timing of CPU Operation Timing 1 in Figure 3-2

- Initialize WFG-ch.10 to RT-dead timer mode (Active High) (WFSA10 register write).
- Set the dead time to WFG-ch10 (WFTM register write). In this example, "0x0010" is set. When the WFG prescaler is set to 1/2 and PCLK to 40MHz, the dead time to be inserted is 0.8µs.
- In this mode of WFG, signals at the same level as RT1 and the opposite level are output to the output signals of WFG (RTO0 and RTO1) respectively.

Timing of CPU Operation Timings 4 and 5 in Figure 3-2

- When the RT1 signal changes from the Low to High level (or from the High to Low level), a specified dead time (transistor response time at the destination of the output) is inserted into the RTO0 and RTO1 signals, as shown in the figure, and the output level is changed.
- Non-overlap signal containing a dead time shown in the figure can be output to RTO0 and RTO1 by using the WFG function.

■ Operation of ADCMP/ATSA

Timing of CPU Operation Timing 1 in Figure 3-2

- Set ADCMP-ch.0 to instruct ADC-unit0 to start AD conversion with the match condition for FRT's Up-count operation (ACSA and ACSB register write).
- Initialize ATSA to select ADC-unit0's conversion start signal from ADCMP as the scan conversion start signal (ATSA register write).
- Set the timing of starting AD conversion (ACCP0 register write). In this example, "0x2800" is set.

Timing of CPU Operation Timing 3 in Figure 3-2

- Instruct ADCMP-ch.0 to enable its operation (ACSA register write).

Timing of CPU Operation Timing 8 in Figure 3-2

- When ADCMP-ch.0 and ATSA detect that the value of the FRT counter has reached "0x2800" during the Up-count operation, they output ADC-unit0's scan conversion start signal.

■ Completion of Processing

Timing of CPU Operation Timing 9 in Figure 3-2

The processing at Timing 9 indicates the procedure for completing the output of the PWM signal.

- Disable the operation of OCU- ch.1 (OCSA10 register write).
- Set the level of the output signals (RT0, RT1) for OCU-ch.1 (OCSB10 register write).
- Disable the operation of ADCMP-ch.0 (ACSA register write).
- When the output of OCU stops, WFG does not perform its operation.
- Instruct FRT-ch.0 to stop the count operation (TCSA0 register write).
- Set "0x0000" to FRT's count value (TCDT0 register write).

The above example explained the operation with 1 channel of OCU, 1 channel of WFG and 1 channel of ADCMP. If OCU-3ch, WFG-3ch and ADCMP-3ch are connected to the same FRT to perform interlocking control, three-phase motor control can be achieved.

■ Details of Register Settings

The procedure for register settings in Example of Operation – 2 is as follows.

The meanings of the symbols in the table are the same as those in Example of Operation – 1.

Table 3-5 Example of Operation 2 - Register Settings 1

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
1	FRT	TCSA0	HW	CLK[3:0]	0010	Clock division prescaler setting: 1/4
				SCLR	0	Soft clear: Do nothing
				MODE	1	Count mode setting: Up/Down-count mode
				STOP	1	FRT count operation: Stop counting
				BFE	1	TCCP buffer function: Enable
				ICRE	0	Peak value detection interrupt: Disable
				ICLR	0	Peak value detection: Clear
				Reserved	000	-
				IRQZE	1	Zero value detection interrupt: Enable
				IRQZF	0	Zero value detection: Clear
				ECKE	0	Selection of clock to be used: Internal clock
		TCCP0	HW	TCCP	0x5FFF	Set FRT cycle
	OCU	OCFS10	BW	FSO0[3:0]	Other	FRT connected to ch.0:
				FSO1[3:0]	0000	FRT connected to ch.1: FRT ch.0
		OCSA10	BW	CST0	Other	ch.0 operation state:
				CST1	0	ch.1 operation state: Operation disabled
				BDIS0	Other	ch.0 OCCP buffer function:
				BDIS1	0	ch.1 OCCP buffer function: Enable
				IOE0	Other	ch.0 interrupt:
				IOE1	0	ch.1 interrupt: Disable
				IOP0	Other	ch.0 match detection:
				IOP1	0	ch.1 match detection: Clear
		OCSB10	BW	OTD0	Other	RT0 output level:
				OTD1	0	RT1 output level: Low
				Reserved	00	-
				CMOD	0	ch.0/ch.1 operation mode: Up/Down (Active High)
				BTS0	Other	ch.0 buffer transfer:
				BTS1	0	ch.1 buffer transfer: Zero value detection transfer
				Reserved	0	-
		OCSC	BW	MOD0	Other	ch.0 operation mode:
				MOD1	1	ch.1 operation mode: Up/Down (Active High)
				MOD2	Other	ch.2 operation mode:
				MOD3	Other	ch.3 operation mode:
				MOD4	Other	ch.4 operation mode:
				MOD5	Other	ch.5 operation mode:
				Reserved	00	-
		OCCP1	HW	OCCP	0x4800	Specify ch.1 change timing

Table 3-6 Example of Operation 2 - Register Settings 2

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
1	WFG	WFSA10	HW	DCK[2:0]	001	Clock division prescaler setting: 1/2
				TMD[2:0]	100	Operation mode: Select RT-dead timer mode
				GTEN[1:0]	00	Gate signal generation: don't care
				PSEL[1:0]	00	Connecting PPG: don't care
				PGEN[1:0]	00	PPG reflection: don't care
				DMOD	0	Output polarity: Active High
				Reserved	000	-
		WFTM10	HW	WFTM	0x0010	Set dead time value
	ADCMP	ACSA	HW	CE0[1:0]	00	ch.0 operation state: Operation disabled
				CE1[1:0]	Other	ch.1 operation state:
				CE2[1:0]	Other	ch.2 operation state:
				Reserved	00	-
				SEL0[1:0]	01	Selection of ch.0 start timing: At Up-count only
				SEL1[1:0]	Other	Selection of ch.1 start timing:
				SEL2[1:0]	Other	Selection of ch.2 start timing:
				Reserved	00	-
		ACSB	BW	BDIS0	1	ch.0 buffer function: Disable
				BDIS1	Other	ch.1 buffer function:
				BDIS2	Other	ch.2 buffer function:
				Reserved	0	-
				BTS0	0	ch.0 buffer transfer: don't care
				BTS1	Other	ch.1 buffer transfer:
				BTS2	Other	ch.2 buffer transfer:
				Reserved	0	
		ACCP0	HW	ACCP	0x2800	Specify start timing for ADC0
	ATSA	ATSA	HW	AD0S[1:0]	00	ADC0 scan conversion start: ADCMP ch.0
				AD1S[1:0]	Other	ADC1 scan conversion function:
				AD2S[1:0]	Other	ADC2 scan conversion function:
				Reserved	00	-
				AD0P[1:0]	00	ADC0 priority conversion start: ADCMP ch.0
				AD1P[1:0]	Other	ADC1 priority conversion start:
				AD2P[1:0]	Other	ADC2 priority conversion start:
				Reserved	00	-

Table 3-7 Example of Operation 2 – Register Settings 3

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
2	FRT	TCSA0	HW	CLK[3:0]	NM	Clock division prescaler setting:
				SCLR	NM	Soft clear:
				MODE	NM	Count mode setting:
				STOP	0	FRT count operation: Start counting
				BFE	NM	TCCP buffer function:
				ICRE	NM	Peak value detection interrupt:
				ICLR	1(RMW)	Peak value detection: Do nothing
				Reserved	NM	-
				IRQZE	NM	Zero value detection interrupt:
				IRQZF	1(RMW)	Zero value detection: Do nothing
				ECKE	NM	Selection of clock to be used: Internal clock
3	OCU	OCSA10	BW	CST0	NM	ch.0 operation state:
				CST1	1	ch.1 operation state: Operation enabled
				BDIS0	NM	ch.0 OCCP buffer function:
				BDIS1	NM	ch.1 OCCP buffer function:
				IOE0	NM	ch.0 interrupt:
				IOE1	NM	ch.1 interrupt:
				IOP0	1	ch.0 match detection: Do nothing
				IOP1	1	ch.1 match detection: Do nothing
	ADCMP	ACSA	HW	CE0[1:0]	01	ch.0 operation state: Operation enabled Connecting FRT: FRT ch.0
				CE1[1:0]	NM	ch.1 operation state:
				CE2[1:0]	NM	ch.2 operation state:
				Reserved	NM	-
				SEL0[1:0]	NM	Selection of ch.0 start timing:
				SEL1[1:0]	NM	Selection of ch.1 start timing:
				SEL2[1:0]	NM	Selection of ch.2 start timing:
				Reserved	NM	-
6	OCU	OCCP1	HW	OCCP1	0x4000	Specify ch.1 change timing

Table 3-8 Example of Operation 2 – Register Settings 4

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
7	FRT	TCSA0	HR	CLK[3:0]	DC	Clock division prescaler setting:
				SCLR	DC	Soft clear:
				MODE	DC	Count mode setting:
				STOP	DC	FRT count operation:
				BFE	DC	TCCP buffer function:
				ICRE	DC	Peak value detection interrupt:
				ICLR	DC	Peak value detection:
				Reserved	DC	-
				IRQZE	DC	Zero value detection interrupt:
				IRQZF	1	Zero value detection: Zero value detected
				ECKE	DC	Selection of clock used:
		TCSA0	HW	CLK[3:0]	NM	Clock division prescaler setting:
				SCLR	NM	Soft clear:
				MODE	NM	Count mode setting:
				STOP	NM	FRT count operation:
				BFE	NM	TCCP buffer function:
				ICRE	NM	Peak value detection interrupt:
				ICLR	1(RMW)	Peak value detection: Do nothing
				Reserved	NM	-
				IRQZE	NM	Zero value detection interrupt:
				IRQZF	0	Zero value detection: Flag cleared
				ECKE	NM	Selection of clock used:

Table 3-9 Example of Operation 2 – Register Settings 5

Setting Timing	Name of Target Block	Name of Register	Operation	bit Field	Value	Description of Setting
9	OCU	OCSA10	BW	CST0	NM	ch.0 operation state:
				CST1	0	ch.1 operation state: Disable
				BDIS0	NM	ch.0 OCCP buffer function:
				BDIS1	NM	ch.1 OCCP buffer function:
				IOE0	NM	ch.0 interrupt:
				IOE1	NM	ch.1 interrupt:
				IOP0	1	ch.0 match detection: Do nothing
				IOP1	1	ch.1 match detection: Do nothing
		OCSB10	BW	OTD0	NM	RT0 output level:
				OTD1	0	RT1 output level: Low
				Reserved	NM	-
				CMOD	NM	ch.0/ch.1 operation mode:
				BTS0	NM	ch.0 buffer transfer:
				BTS1	NM	ch.1 buffer transfer:
				Reserved	NM	-
	ADCMP	ACSA	HW	CE0[1:0]	00	ch.0 operation state: Operation disabled
				CE1[1:0]	NM	ch.1 operation state:
				CE2[1:0]	NM	ch.2 operation state:
				Reserved	NM	-
				SEL0[1:0]	NM	Selection of ch.0 start timing:
				SEL1[1:0]	NM	Selection of ch.1 start timing:
				SEL2[1:0]	NM	Selection of ch.2 start timing:
				Reserved	NM	-
	FRT	TCSA0	HW	CLK[3:0]	NM	Clock division prescaler setting:
				SCLR	1	Soft clear: Initialize FRT
				MODE	NM	Count mode setting:
				STOP	1	FRT count operation: Stop counting
				BFE	NM	TCCP buffer function:
				ICRE	NM	Peak value detection interrupt:
				ICLR	1	Peak value detection: Do nothing
				Reserved	NM	-
				IRQZE	NM	Zero value detection interrupt:
				IRQZF	1	Zero value detection: Do nothing
				ECKE	NM	Selection of clock used:
		TCDT0	HW	TCDT	0x0000	Initialize FRT count value

4. Registers of Multifunction Timer

This section explains the registers of the multifunction timer.

4.1. Individual Notation and Common Notation of Channel Numbers in Descriptions of Functions

This section explains the individual notation and common notation of channel numbers in descriptions of the functions in this chapter.

As the multifunction timer unit contains multiple blocks of the same function and consists of multiple channel circuits, there are some common matters across the channels.
Where there is no need to distinguish among the channels, and functions that are common to all the channels are to be explained, a notation without channel numbers and a notation with parentheses (common notation) are used to avoid repeated explanations and simplify their explanation.
Where there is a need to make distinctions in explaining operation among channels, I/O signals or control registers, a notation clearly stating channel numbers (individual notation) is used in such explanation.
The notation rules and examples are provided below.

- Where channel numbers are notated directly, that indicates individual notation.
  This notation indicates that the operation, I/O signal or control register of the corresponding channel is explained.
- Some control registers control 2 channels at the same time. In such cases, the two corresponding channel numbers are stated in individual notation to distinguish between them.
- Where channel numbers are omitted from a notation, that indicates the common notation.
  This notation indicates that the operation, I/O signal or control register which is common to all the channels is explained to omit the repetition of such explanation.
- Where channel numbers are stated with a figure in parentheses, that indicates the common notation for some channels.
  Where there is a need for distinguishing between even-numbered channels and odd-numbered channels among the channels mounted, (0) and (1) are stated respectively.
  In this case, (0) indicates that a function that is common to the even-numbered channels is explained, while (1) indicates that a function that is common to the odd-numbered channels is explained.

Example 1: ICU-ch.3 of MFT unit 0 can select the calibration input of the internal CR oscillator.

Example 1 is an example of the individual notation, which indicates that the calibration input of the internal CR oscillator can be selected by only ICU-ch.3 of MFT unit 0. This notation indicates that the calibration input of the internal CR oscillator cannot be selected by ICU-ch.0 to ch.2 of MFT unit 0 or ICU ch.0 to ch.3 of other MFT units.

Example 2: The ICFS10 register is a register that selects FRT to be connected to ICU-ch.1 and ICU-ch.0.

Example 3: The ICFS32 register is a register that selects FRT to be connected to ICU-ch.3 and ICU-ch.2.

Examples 2 and 3 are examples of the individual notation that states a control register (ICFS) with two channel numbers (10 and 32).

Example 4: The ICFS register is a register that selects FRT to be connected to ICU.

Example 4 is an example of the common notation that omits the channel numbers of the control register (ICFS). What the description explains means that similar to Examples 2 and 3, repeated explanations are omitted by the common notation.

Example 5: ICFS10.FSI0[3:0] is a register that selects FRT to be connected to ICU-ch.0.

Example 6: ICFS10.FSI1[3:0] is a register that selects FRT to be connected to ICU-ch.1.

Example 7: ICFS32.FSI0[3:0] is a register that selects FRT to be connected to ICU-ch.2.

Example 8: ICFS32.FSI1[3:0] is a register that selects FRT to be connected to ICU-ch.3.

Examples 5 to 8 are examples of the individual notation that clearly identifies the correspondence between the control bit and the channel in the control registers by stating two channel numbers in the control register (ICFS).

Example 9: ICFS.FSI0[3:0] is a register that selects FRT to be connected to ICU-ch.(0).

Example 10: ICFS.FSI1[3:0] is a register that selects FRT to be connected to ICU-ch.(1).

Examples 9 and 10 are examples of the common notation with parentheses that omits the channel numbers of the control register. What the description explains means that similar to Examples 5 to 8, repeated explanations are omitted by the common notation.
It should be noted that where the common notation is used in explanation of each function block, as shown above, it must be converted into the individual notation for the relevant channel when it is read.
Table 4-1 to Table 4-3 show the correspondence table between the individual notation and common notation. For the correspondence between the individual notation and common notation regarding register names, see the list of registers.

Table 4-1 Individual Notation and Common Notation of OCU

Channel Number		5	4	3	2	1	0
Notation for explaining OCU operation	Individual	ch.5	ch.4	ch.3	ch.2	ch.1	ch.0
	Common	ch.(1)	ch.(0)	ch.(1)	ch.(0)	ch.(1)	ch.(0)
Notation of names of signals output from OCU	Individual	RT5	RT4	RT3	RT2	RT1	RT0
	Common	RT(1)	RT(0)	RT(1)	RT(0)	RT(1)	RT(0)

Table 4-2 Individual Notation and Common Notation of WFG

Channel Number		54		32		10	
Notation for explaining WFG operation	Individual	ch.54		ch.32		ch.10	
	Common	No notation					
Names of signals input from OCU	Individual	RT5	RT4	RT3	RT2	RT1	RT0
	Common	RT(1)	RT(0)	RT(1)	RT(0)	RT(1)	RT(0)
Names of signals output from WFG	Individual	RTO5	RTO4	RTO3	RTO2	RTO1	RTO0
	Common	RTO(1)	RTO(0)	RTO(1)	RTO(0)	RTO(1)	RTO(0)
Names of PPG input signals after selected to be input from PPG	Individual	CH10_PPG		CH32_PPG		CH54_PPG	
	Common	CH_PPG					
Names of GATE signals before selected to be output to PPG	Signal Name	CH10_GATE		CH32_GATE		CH54_GATE	
	Common	CH_GATE					

Table 4-3 Individual Notation and Common Notation of ICU

Channel Number		3	2	1	0
Notation for explaining ICU operation	Individual	ch.3	ch.2	ch.1	ch.0
	Common	ch.(1)	ch.(0)	ch.(1)	ch.(0)
Notation of names of ICU input signals	Individual	IC3	IC2	IC1	IC0
	Common	IC(1)	IC(0)	IC(1)	IC(0)

4.2. List of Registers of Multifunction Timer

This section provides a list of the registers that exist in the multifunction timer unit.

Table 4-4 shows a list of the registers that exist in the multifunction timer unit.
The control registers of the multifunction timer unit are in the same configuration across the mounted channels. In this section, the operation of registers of the same function is explained using the common notation. The List of Registers states names in the individual notation and the common notation for each register. Replace the name in the common notation that appears in descriptions with the name in the individual notation when reading the descriptions.
Registers shown in the List of Registers refer to the registers that exist in the Multifunction Timer 1 unit. A model containing more than one multifunction timer unit has sets of the same registers for the number of the multifunction timer units.

Table 4-4 List of Registers of Multifunction Timer Unit

Block Name	Register Name (Individual Notation)	Register Function	bit Width	Access	See	Register Name (Common Notation)	
FRT	TCSA0	FRT ch.0 control register A	16	B, H	4.3.1	TCSA	
	TCSA1	FRT ch.1 control register A					
	TCSA2	FRT ch.2 control register A					
	TCSB0	FRT ch.0 control register B	16	B, H	4.3.2	TCSB	
	TCSB1	FRT ch.1 control register B					
	TCSB2	FRT ch.2 control register B					
	TCCP0	FRT ch.0 cycle setting register	16	H	4.3.3	TCCP	
	TCCP1	FRT ch.1 cycle setting register					
	TCCP2	FRT ch.2 cycle setting register					
	TCDT0	FRT ch.0 count value register	16	H	4.3.4	TCDT	
	TCDT1	FRT ch.1 count value register					
	TCDT2	FRT ch.2 count value register					
OCU	OCFS10	OCU ch.1, ch.0 connecting FRT select register	8	B, H	4.3.5	OCFS	
	OCFS32	OCU ch.3, ch.2 connecting FRT select register					
	OCFS54	OCU ch.5, ch.4 connecting FRT select register		B			
	OCSA10	OCU ch.1, ch.0 control register A	8	B, H	4.3.6	OCSA	
	OCSA32	OCU ch.3, ch.2 control register A					
	OCSA54	OCU ch.5, ch.4 control register A					
	OCSB10	OCU ch.1, ch.0 control register B	8	B, H	4.3.7	OCSB	
	OCSB32	OCU ch.3, ch.2 control register B					
	OCSB54	OCU ch.5, ch.4 control register B					
	OCSC	OCU ch.5 to ch.0 control register C	8	B	4.3.8	OCSC	
	OCCP0	OCU ch.0 compare value store register	16	H	4.3.9	OCCP	OCCP(0)
	OCCP1	OCU ch.1 compare value store register					OCCP(1)
	OCCP2	OCU ch.2 compare value store register					OCCP(0)
	OCCP3	OCU ch.3 compare value store register					OCCP(1)
	OCCP4	OCU ch.4 compare value store register					OCCP(0)
	OCCP5	OCU ch.5 compare value store register					OCCP(1)

Block Name	Register Name (Individual Notation)	Register Function	bit Width	Access	See	Register Name (Common Notation)
WFG	WFSA10	WFG ch.10 control register A	16	H	4.3.10	WFSA
	WFSA32	WFG ch.32 control register A				
	WFSA54	WFG ch.54 control register A				
	WFTM10	WFG ch.10 timer value register	16	H	4.3.11	WFTM
	WFTM32	WFG ch.32 timer value register				
	WFTM54	WFG ch.54 timer value register				
NZCL	NZCL	NZCL control register	16	H	4.3.12	NZCL
	WFIR	WFG Interrupt control register	16	H	4.3.13	WFIR
ICU	ICFS10	ICU ch.1, ch.0 connecting FRT select register	8	B, H	4.3.14	ICFS
	ICFS32	ICU ch.3, ch.2 connecting FRT select register				
	ICSA10	ICU ch.1, ch.0 control register A	8	B, H	4.3.15	ICSA
	ICSA32	ICU ch.3, ch.2 control register A				
	ICSB10	ICU ch.1, ch.0 control register B	8	B, H	4.3.16	ICSB
	ICSB32	ICU ch.3, ch.2 control register B				
	ICCP0	ICU ch.0 capture value store register	16	H	4.3.17	ICCP
	ICCP1	ICU ch.1 capture value store register				
	ICCP2	ICU ch.2 capture value store register				
	ICCP3	ICU ch.3 capture value store register				
ADCMP	ACSA	ADCMP ch.2 to ch.0 control register A	16	B, H	4.3.18	ACSA
	ACSB	ADCMP ch.2 to ch.0 control register B	8	B	4.3.19	ACSB
	ACCP0	ADCMP ch.0 compare value store register	16	H	4.3.20	ACCP
	ACCP1	ADCMP ch.1 compare value store register				
	ACCP2	ADCMP ch.2 compare value store register				
	ACCPDN0	ADCMP ch.0 compare value store register (For the down-count direction only)	16	H	4.3.21	ACCPDN
	ACCPDN1	ADCMP ch.1 compare value store register (For the down-count direction only)				
	ACCPDN2	ADCMP ch.2 compare value store register (For the down-count direction only)				
ATSA	ATSA	ADC start trigger select register	16	H	4.3.22	ATSA

4.3. Details of Register Functions

This section explains details of the registers that exist in the multifunction timer unit.

4.3.1. FRT Control Register A (TCSA)

TCSA is a 16-bit register that controls FRT.
Each mounted channel has three registers: TCSA0, TCSA1 and TCSA2.
TCSA0 controls FRT-ch.0.
TCSA1 controls FRT-ch.1.
TCSA2 controls FRT-ch.2.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	ECKE	IRQZF	IRQZE	Reserved			ICLR	ICRE
Attribute	R/W	R/W	R/W	-			R/W	R/W
Initial value	0	0	0	000			0	0

bit	7	6	5	4	3	2	1	0
Field	BFE	STOP	MODE	SCLR	CLK[3:0]			
Attribute	R/W	R/W	R/W	W	R/W			
Initial value	0	1	0	0	0000			

■ Functions of Register

[bit3:0] CLK[3:0]

Process	Value	Function
Write	0000	Sets FRT's count clock cycle to the same value as PCLK.
	0001	Sets FRT's count clock cycle to PCLK multiplied by 2.
	0010	Sets FRT's count clock cycle to PCLK multiplied by 4.
	0011	Sets FRT's count clock cycle to PCLK multiplied by 8.
	0100	Sets FRT's count clock cycle to PCLK multiplied by 16.
	0101	Sets FRT's count clock cycle to PCLK multiplied by 32.
	0110	Sets FRT's count clock cycle to PCLK multiplied by 64.
	0111	Sets FRT's count clock cycle to PCLK multiplied by 128.
	1000	Sets FRT's count clock cycle to PCLK multiplied by 256.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

The CLK[3:0] bits are bit that sets the count clock cycle of FRT counter (16-bit Up/Down counter).
Change the setting of this register while FRT is stopping.
As for FRT count clock, either the PCLK in LSI which is divided by the prescaler or an external clock input can be selected for use. As this register setting is the setting for the prescaler, its value has no meaning if an external clock input is selected.
FRT's count clock cycle is determined based on the PCLK cycle and the clock division ratio set by this register.

The following table shows examples of CLK[3:0] settings and FRT count clock cycles.

CLK[3:0]	Cycle Ratio	FRT Count Clock Cycle		
		PCLK=25 ns (40 MHz)	PCLK=33.3 ns (33 MHz)	PCLK=50 ns (20 MHz)
0000	1	25 ns	30 ns	50 ns
0001	2	50 ns	61 ns	100 ns
0010	4	100 ns	121 ns	200 ns
0011	8	200 ns	242 ns	400 ns
0100	16	400 ns	485 ns	800 ns
0101	32	800 ns	970 ns	1.6 µs
0110	64	1.6 µs	1.9 µs	3.2 µs
0111	128	3.2 µs	3.9 µs	6.4 µs
1000	256	6.4 µs	7.8 µs	12.8 µs

[bit4] SCLR

Process	Value	Function
Write	0	Cancels FRT operation state initialization request.
	1	Issues FRT operation state initialization request.
Read	-	"0" is always read.

The SCLR bit is a bit that requests FRT operation state initialization.

There are two ways to use this bit, as described below.

1. When stopping FRT counter
   Write "1" to issue an initialization request for FRT's operation state when stopping FRT counter.
2. When clearing FRT counter through clock synchronization
   Write "1" to issue a request to clear FRT's count value to "0x0000" through synchronization when operating FRT in Up-count mode.

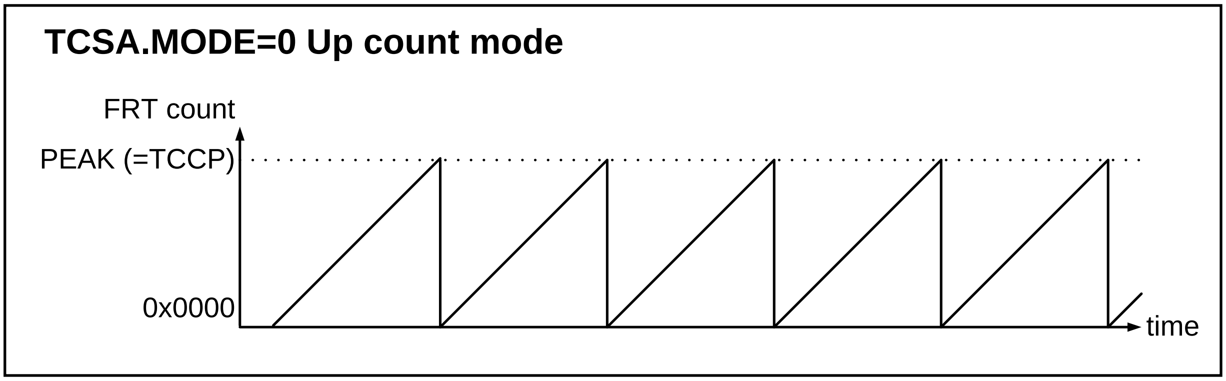
For information about how to use this register, see [bit6] STOP.

[bit5] MODE

Process	Value	Function
Write	0	Sets FRT's count mode to Up-count mode.
	1	Sets FRT's count mode to Up/Down-count mode.
Read	-	Read the register setting.

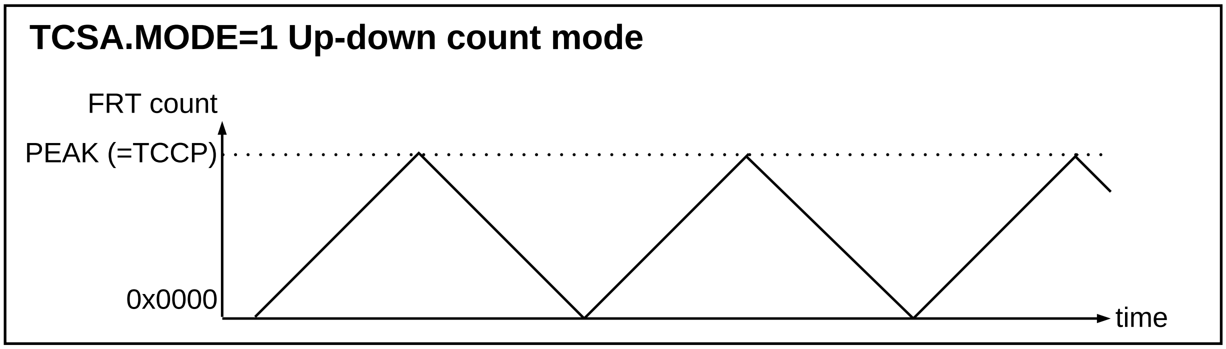
The MODE bit is a bit that selects FRT's count mode.
Change the setting of this bit while FRT is stopping.
In Up-count mode, FRT performs the following operation.
FRT's counter starts Up-count operation from "0x0000". After up-counting to the value set by the TCCP register, the value of the counter becomes "0x0000". Then, the Up-count operation is repeated. FRT's count cycle is "(TCCP+1) × Count clock cycle".
Change in the value of FRT's counter is shown below.

0x0000	(Zero value)	
0x0001		
0x0002		
.		
.	(Up-count)	FRT's count cycle
.		
TCCP-2		
TCCP-1		
TCCP	(Peak value)	
0x0000	(Zero value)	
0x0001		
0x0002		
.		
.	(Up-count)	FRT's count cycle
.		
TCCP-2		
TCCP-1		
TCCP	(Peak value)	
0x0000	(Zero value)	
0x0001		


TCSA.MODE=0 Up count mode

In Up/Down-count mode, FRT performs the following operation.
FRT's counter starts Up-count operation from "0x0000". After up-counting to the value set by the TCCP register, it starts Down-count operation. When it returns to "0x0000", it starts up-counting again and repeats the count operation.
FRT's count cycle is "(TCCP) × 2 × Count clock cycle".
Change in the value of FRT's counter is shown below.

0x0000	(Zero value)	
0x0001		
0x0002		
.		
.	(Up-count)	
.		
TCCP-2		
TCCP-1		
TCCP	(Peak value)	FRT's count cycle
TCCP-1		
TCCP-2		
.		
.	(Down-count)	
.		
0x0002		
0x0001		
0x0000	(Zero value)	
0x0001		
0x0002		
.		
.	(Up-count)	FRT's count cycle
.		
TCCP-2		
TCCP-1		
TCCP	(Peak value)	
TCCP-1		
TCCP-2		
.		
.	(Down-count)	
.		
0x0002		
0x0001		
0x0000	(Zero value)	
0x0001		


TCSA.MODE=1 Up-down count mode

[bit6] STOP

Process	Value	Function
Write	0	Puts FRT in operating state.
	1	Puts FRT in stopping state.
Read	-	Reads the register setting

The STOP bit is a bit that controls the start and stop of FRT's operation.
This register is used in the combination with SCLR, as shown below.

1. When starting FRT's counter operation:
   When "0" is written to STOP and SCLR while FRT's count operation is stopped, FRT starts counting.

2. When clearing the count value of FRT's counter to "0x0000" through synchronization in Up-count mode:
   If "0" is written to STOP and "1" is written to SCLR during FRT's count operation in Up-count mode, FRT's count value is cleared to "0x0000" in FRT's next count clock. If "0" is written to SCLR before the counter is cleared, the counter clear request is cancelled and the counter value is not cleared. Do not write SCLR=0 until it can be checked that the counter value is cleared. This operation cannot be performed in Up/Down-count mode.

3. When stopping the operation of FRT's counter:
   If "1" is written to STOP and SCLR during FRT's count operation, FRT stops the count operation. In some cases, FRT's counter value is not initialized to "0x0000" even after FRT stops, depending on the state of FRT's count clock. Always write "0x0000" to TCDT afterwards to clear FRT's counter value to "0x0000".

To rewrite to another register in the same address area during FRT's count operation, write "0" to STOP and SCLR.

To rewrite to another register in the same address area while FRT's count operation is stopped, write "1" to STOP and "0" to SCLR.

Figure 4-1 FRT Count Start, Clear and Stop (Up-count Mode)

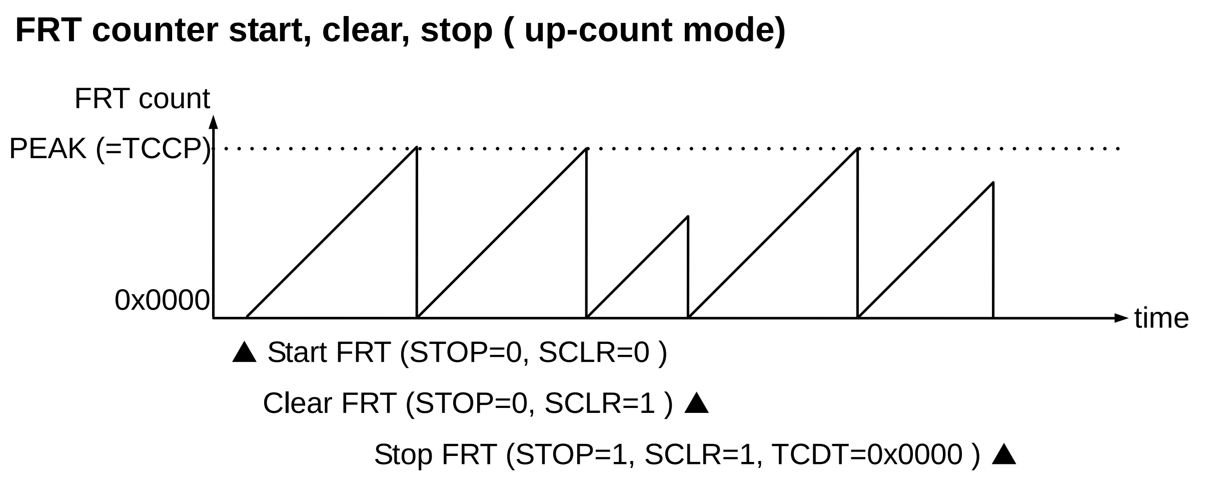
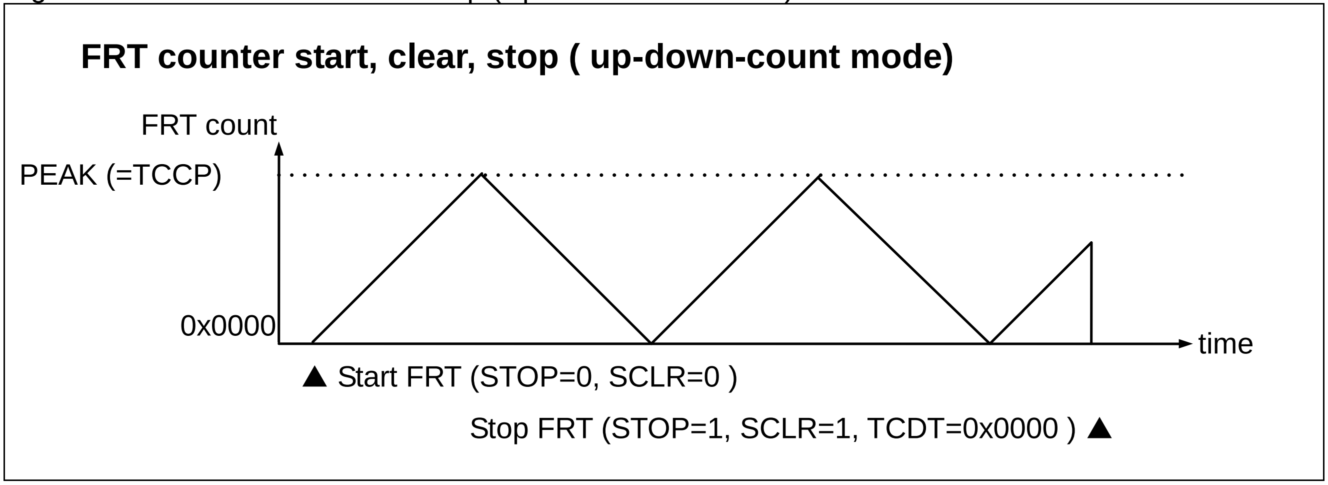

FRT counter start, clear, stop (up-count mode)
FRT count
PEAK (=TCCP)
0x0000
time
▲ Start FRT (STOP=0, SCLR=0)
Clear FRT (STOP=0, SCLR=1) ▲
Stop FRT (STOP=1, SCLR=1, TCDT=0x0000) ▲

Figure 4-2 FRT Count Start and Stop (Up/Down-count Mode)


FRT counter start, clear, stop (up-down-count mode)
FRT count
PEAK (=TCCP)
0x0000
time
▲ Start FRT (STOP=0, SCLR=0)
Stop FRT (STOP=1, SCLR=1, TCDT=0x0000) ▲

[bit7] BFE

Process	Value	Function
Write	0	Disables TCCP's buffer function.
	1	Enables TCCP's buffer function.
Read	-	Reads the register setting.

The BFE bit is a bit that specifies whether to enable or disable the buffer function of the TCCP register.
See "4.3.3 FRT Cycle Setting Register (TCCP)".

[bit8] ICRE

Process	Value	Function
Write	0	Does not generate interrupt when "1" is set to ICLR.
	1	Generates interrupt when "1" is set to ICLR.
Read	-	Read he register setting.

The ICRE bit is a bit that specifies whether to notify CPU of the event that ICLR is set as an interrupt (enabling interrupt) or not to notify it (disabling interrupt).
See "6.2 Treatment of Event Detect Register and Interrupt".

[bit9] ICLR

Process	Value	Function
Write	0	Clears this register to "0".
	1	Does nothing.
Read	0	Indicates that no match has been detected between FRT's count value and TCCP value.
	1	Indicates that a match has already been detected between FRT's count value and TCCP value.
Read at RMW access		"1" is always read.

The ICLR bit is a bit that is set to "1" when a match is detected between FRT's count value and TCCP value during FRT operation (hereinafter referred to as "Peak value detection".
By reading this bit, whether FRT's count value has reached the TCCP value or not can be determined.
This bit can be cleared by writing "0".
This bit does nothing, if "1" is written. Always write "1" to the register when rewriting to another register in the same address area.
"1" is always read from this bit at RMW access.
See "6.2 Treatment of Event Detect Register and Interrupt".

[bit12:10] Reserved: Reserved bits
"0" must be written at write access. Read value is "0".

[bit13] IRQZE

Process	Value	Function
Write	0	Does not generate interrupt when "1" is set to IRQZF.
	1	Generates interrupt, when "1" is set to IRQZF.
Read	-	Reads the register setting.

The IRQZE bit is a bit that specifies whether to notify CPU of the event that IRQZF is set as an interrupt (enabling interrupt) or not to notify it (disabling interrupt).
See "6.2 Treatment of Event Detect Register and Interrupt".

[bit14] IRQZF

Process	Value	Function
Write	0	Clears this register to "0".
	1	Does nothing.
Read	0	Indicates that a match between FRT's count value and "0x0000" has not been detected.
	1	Indicates that a match between FRT's count value and "0x0000" has already been detected.
Read at RMW access		"1" is always read.

The IRQZF bit is a bit that is set to "1" when a match between FRT's count value and "0x0000" has been detected during FRT operation (hereinafter referred to as "Zero value detection").

By reading this bit, whether FRT's count value has reached "0x0000" or not can be determined.
This bit is not set at "0x0000" from which FRT starts counting or at "0x0000" to which the counter value has been cleared by TCSA.SCLR.
This bit can be cleared by writing "0".
This bit does nothing, if "1" is written. Always write "1" to the register when rewriting to another register in the same address area.
"1" is always read from this bit at RMW access.
See "6.2 Treatment of Event Detect Register and Interrupt".

[bit15] ECKE

Process	Value	Function
Write	0	Uses the internal clock (PCLK) as FRT's count clock.
	1	Uses an external input clock (FRCK) as FRT's count clock.
Read	-	Reads the register setting.

The ECKE bit is a bit that selects the clock signal to be used as FRT's count clock.

Change the setting of this bit while FRT is stopping.

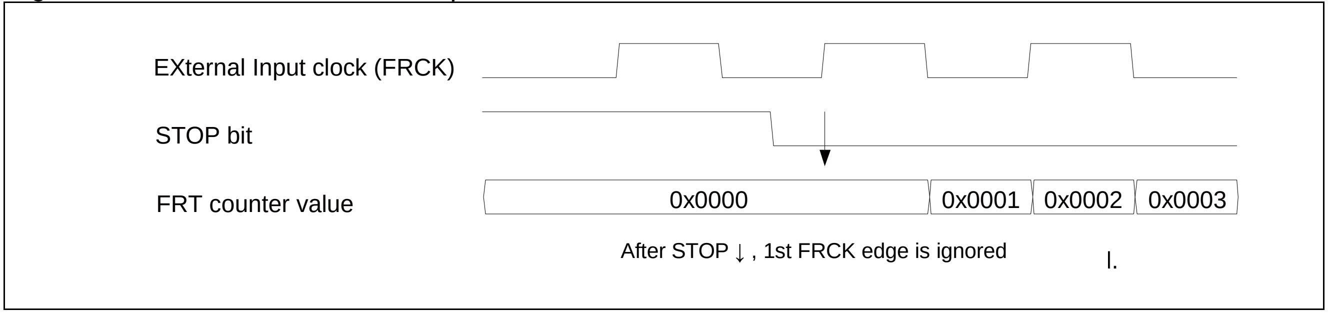
To select the internal clock, the clock division ratio must be set by CLK[3:0].

To select an external input clock, the FRCK pin to be used in the I/O port block must be predetermined.

To operate it with an external input clock, the count operation is performed both at the rising edge and falling edge of an external input clock signal.

To operate it with an external input clock, the first edge from the external input clock after FRT operation starts (writing "0" to STOP) is ignored, irrespective of the rising or falling edge, and the count operation starts from the next edge.

Figure 4-3 Selection of External Input Clock


EXternal Input clock (FRCK)
STOP bit
FRT counter value
0x0000
0x0001
0x0002
0x0003
After STOP ↓ , 1st FRCK edge is ignored
l.

4.3.2. FRT Control Register B (TCSB)

TCSB is a 16-bit register that controls FRT.
Each mounted channel has three registers: TCSB0, TCSB1 and TCSB2.
TCSB0 controls FRT ch.0.
TCSB1 controls FRT ch.1.
TCSB2 controls FRT ch.2.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved							
Attribute	-							
Initial Value	0x00							

bit	7	6	5	4	3	2	1	0
Field	Reserved					AD2E	AD1E	AD0E
Attribute	-					R/W	R/W	R/W
Initial Value	00000					0	0	0

■ Functions of Register

[bit0] AD0E

Process	Value	Function
Write	0	Does not output AD conversion start signal to ADC unit0 upon Zero value detection by FRT.
	1	Outputs AD conversion start signal to ADC unit0 upon Zero value detection by FRT.
Read	-	Reads the register setting.

[bit1] AD1E

Process	Value	Function
Write	0	Does not output AD conversion start signal to ADC unit1 upon Zero value detection by FRT.
	1	Outputs AD conversion start signal to ADC unit1 upon Zero value detection by FRT.
Read	-	Reads the register setting.

[bit2] AD2E

Process	Value	Function
Write	0	Does not output AD conversion start signal to ADC unit2 upon Zero value detection by FRT.
	1	Outputs AD conversion start signal to ADC unit2 upon Zero value detection by FRT.
Read	-	Reads the register setting.

AD0E, AD1E and AD2E are registers that select AD conversion start signal output upon Zero value detection by FRT. These registers are used to start ADC conversion upon Zero value detection by FRT. Each of the AD conversion start signals for the 3 channels of FRT undergoes logic OR by ADC unit to which they are to be output. See the entire block diagram.

The conversion start signal from FRT ch.0, FRT ch.1 and FRT ch.2 to ADC unit0 has undergone logic OR.
The conversion start signal from FRT ch.0, FRT ch.1 and FRT ch.2 to ADC unit1 has undergone logic OR.
The conversion start signal from FRT ch.0, FRT ch.1 and FRT ch.2 to ADC unit2 has undergone logic OR.

Due to the above configuration, attention must be paid when starting AD conversion from multiple FRT's to the same ADC. To start AD conversion at FRT's count value other than Zero value detection, ADCMP can be used to output the AD conversion start signal.

It can be selected in the ATSA block whether to use the ADC conversion start signal achieved upon Zero value detection by FRT and the ADC conversion start signal achieved by ADCMP for starting ADC's scan conversion or priority conversion.

[bit15:3] Reserved : Reserved bits

"0" must be written at write access. Read value is "0".

4.3.3. FRT Cycle Setting Register (TCCP)

TCCP is a 16-bit register that sets FRT's count cycle.
Each mounted channel has three registers: TCCP0, TCCP1 and TCCP2.
TCCP0 sets the cycle for FRT ch.0.
TCCP1 sets the cycle for FRT ch.1.
TCCP2 sets the cycle for FRT ch.2.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	TCCP[15:0]															
Attribute	R/W															
Initial Value	0xFFFF															

■ Functions of Register

[bit15:0] TCCP[15:0]

Process	Function
Write	Sets FRT's cycle. Stores the written value to the TCCP buffer register.
Read	Reads the value in the TCCP register (not the value in the TCCP buffer register).

TCCP is a 16-bit register that sets FRT's count cycle.

Depending on the TCCP value and FRT's count mode, FRT's count cycle varies, as shown below.

In Up-count mode:
FRT's count cycle = (TCCP+1) × FRT's count clock cycle

In Up/Down-count mode:
FRT's count cycle = TCCP × 2 × FRT's count clock cycle

When data is written to this address area, the data is first stored in the buffer register. And then, the data is transferred from the buffer register to the TCCP register under the following conditions.

When the buffer function is disabled:
Data is transferred immediately after it is written to the buffer register.

When the buffer function is enabled:
Data is transferred, when FRT is stopped or when FRT's count value has reached "0x0000".

Whether the buffer function is enabled or disabled is determined by the value in the TCSA.BFE register.

FRT's count cycle can be changed by rewriting this register during FRT's count operation. If data is read from this address area, the value in the TCCP register is read, rather than the value in the buffer register. Therefore, it should be noted that no bit can be rewritten by RMW access to this address area when the buffer function is enabled.

It is prohibited to write "0x0000" to this register.

Figure 4-4 shows an example of changing FRT's cycle when the buffer function is enabled.

When TCCP's buffer function is enabled, a value written to the buffer register is transferred to the TCCP register upon the next Zero value detection. FRT's count cycle is changed in the next FRT cycle after the writing.

Figure 4-4 Example of Changing FRT's Cycle (When Buffer Function is Enabled)

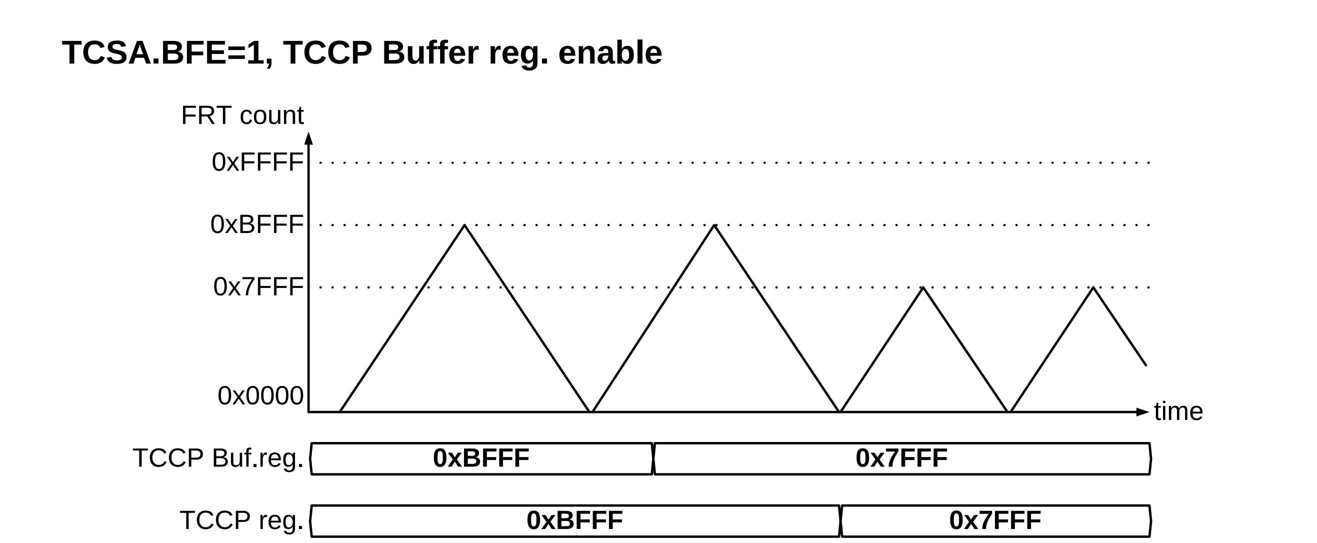
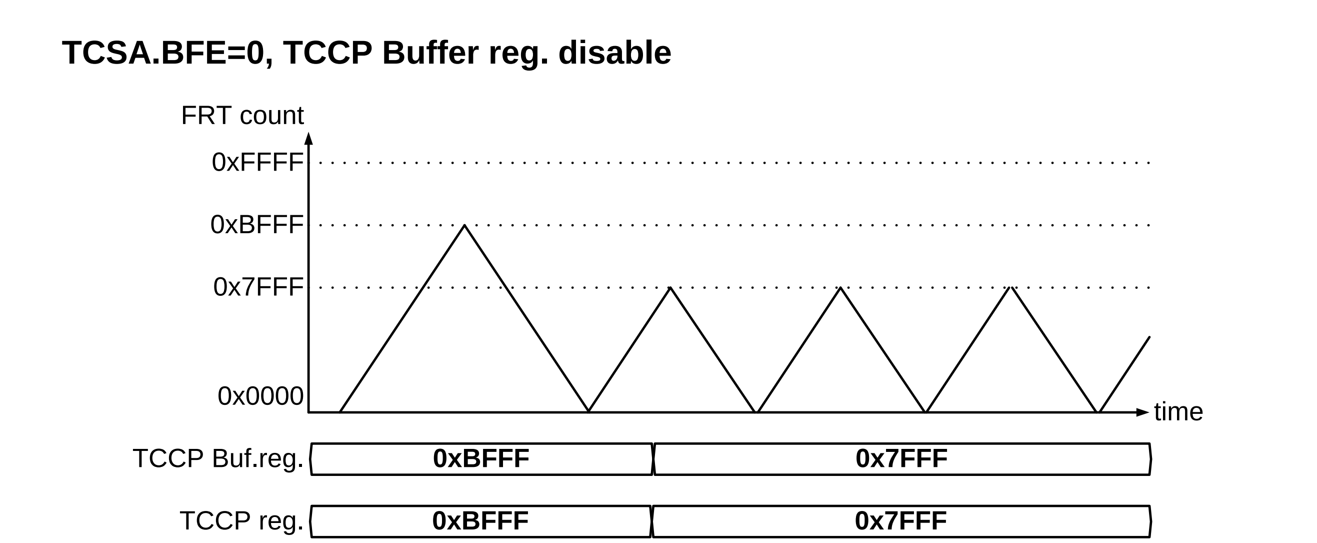


Figure 4-5 shows an example of changing FRT's cycle when the buffer function is disabled.

When TCCP's buffer function is disabled, the value in the buffer register is immediately reflected on the TCCP register; therefore, FRT's cycle can be changed in the same cycle in which the value was written. In this case, it should be noted that if a value smaller than FRT's count value is written as the TCCP value at this point, FRT's count value counts up to "0xFFFF".

Figure 4-5 Example of Changing FRT's Cycle (When Buffer Function is Disabled)



4.3.4. FRT Count Value Register (TCDT)

TCDT is a 16-bit register that reads and writes FRT's count value.
Each mounted channel has three registers: TCDT0, TCDT1 and TCDT2.
TCDT0 is the timer count value of FRT-ch.0.
TCDT1 is the timer count value of FRT-ch.1.
TCDT2 is the timer count value of FRT-ch.2.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	TCDT[15:0]															
Attribute	R/W															
Initial Value	0x0000															

■ Functions of Register

[bit15:0] TCDT[15:0]

Process	Value	Function
Write	0x0000	Sets FRT's count value to "0x0000" (possible only when FRT is stopped).
	Other than above	Setting is prohibited.
Read	-	Reads FRT's current count value.

The TCDT register is a 16-bit register that reads and writes FRT's count value.

The value read from TCDT register is FRT's count value of that point.

Do not write any data during FRT's operation or a value other than "0x0000".

If FRT is operated, and then stopped, make sure to write "0x0000" to TCDT register and initialize FRT's count value in order to prepare for its restart.

4.3.5. OCU Connecting FRT Select Register (OCFS)

OCFS is an 8-bit register that selects and sets FRT to be connected to OCU.
Each mounted channel has three registers: OCFS10, OCFS32 and OCFS54.
OCFS10 controls OCU ch.1 and OCU ch.0.
OCFS32 controls OCU ch.3 and OCU ch.2.
OCFS54 controls OCU ch.5 and OCU ch.4.
OCFS10 and OCFS54 are located at even-numbered addresses, while OCFS32 is located at an odd-numbered address. Therefore, their bit positions are [7:0] and [15:8].

■ Configuration of Register

bit	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0
Field	FSO1[3:0]				FSO0[3:0]			
Attribute	R/W				R/W			
Initial Value	0000				0000			

■ Functions of Register

[bit3:0/11:8] FSO0[3:0]

Process	Value	Function
Write	0000	Connects FRT ch.0 to OCU ch.(0).
	0001	Connects FRT ch.1 to OCU ch.(0).
	0010	Connects FRT ch.2 to OCU ch.(0).
	0011 0100	For models with multiple MFT units: Connects FRT of an external MFT. For models with one MFT unit: Setting is prohibited.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

[bit7:4/15:12] OCFS.FSO1[3:0]

Process	Value	Function
Write	0000	Connects FRT ch.0 to OCU ch.(1).
	0001	Connects FRT ch.1 to OCU ch.(1).
	0010	Connects FRT ch.2 to OCU ch.(1).
	0011 0100	For models with multiple MFT units: Connects FRT of an external MFT. For models with one MFT unit: Setting is prohibited.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

FSO0[3:0] are bit that selects FRT to be connected to ch.(0) of OCU and uses it.
FSO1[3:0] are bit that selects FRT to be connected to ch.(1) of OCU and uses it.
Change the setting of this register, while the operation of the OCU to be connected is prohibited.
For models with multiple MFT units, the connection to FRT that exists in another MFT unit can be selected. For related settings, see "6.1 Connection of Model Containing Multiple MFT's".

4.3.6. OCU Control Register A (OCSA)

OCSA is an 8-bit register that controls OCU's operation.
Each mounted channel has three registers: OCSA10, OCSA32 and OCSA54.
OCSA10 controls OCU ch.1 and OCU ch.0.
OCSA32 controls OCU ch.3 and OCU ch.2.
OCSA54 controls OCU ch.5 and OCU ch.4.

■ Configuration of Register

bit	7	6	5	4	3	2	1	0
Field	IOP1	IOP0	IOE1	IOE0	BDIS1	BDIS0	CST1	CST0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	1	1	0	0

■ Functions of Register

[bit0] CST0

Process	Value	Function
Write	0	Disables the operation of OCU ch.(0). Reflects the value written to OCSB.OTD0 on the RT(0) output pin.
	1	Enables the operation of OCU ch.(0). Ignores the value written to OCSB.OTD0.
Read	-	Reads the register setting.

[bit1] CST1

Process	Value	Function
Write	0	Disables the operation of OCU ch.(1). Reflects the value written to OCSB.OTD1 on the RT(1) output pin.
	1	Enables the operation of OCU ch.(1). Ignores the value written to OCSB.OTD1.
Read	-	Reads the register setting.

CST0 is a register that selects the operation state of OCU ch(0).
CST1 is a register that selects the operation state of OCU ch(1).

Each channel of OCU, when the operation is enabled, performs the following operation according to the operation mode setting, at the timing when the value specified in the OCCP register matches FRT's count value.

- It changes the output level of the RT0 to RT5 output pins and outputs the PWM signal.
- It sets "1" to the IOP0 and IOP1 registers and notifies CPU of the state change.

If the values do not match and the operation is disabled, the output level of the output pins maintains the last state.

For OCU's operation modes, see "4.4 Details of OCU Output Waveform".

If OCU's operation is enabled, the writing to the OCSB.OTD0 and OCSB.OTD1 registers is ignored and not reflected on the level of the output pins.

<Notes>

Always follow the procedure below and perform control when starting PWM signal output by OCU.

1. Initial setting
   Set FRT operation mode (FRT control register other than TCSA.STOP).
   Set OCU operation mode and initialize the output level (OCU control register other than OCSA.CST0 and OCSA.CST1).
   Set the OCCP compare value (writing the OCCP value).
2. Start FRT count operation (writing "0" to TCSA.STOP).
3. Enable OCU's operation (writing "1" to OCSA.CST0 and OCSA.CST1).

Always follow the procedure below and perform control when finishing PWM signal output by OCU.

1. Disable OCU's operation (writing "0" to CST0 and CST1).
2. Reset the output level of the OCU output pins (writing to OCSB.OTD0 and OCSB.OTD1, if necessary).
3. Stop FRT's count operation (writing "1" to TCSA.STOP and TCSA.SCLR, writing 0x0000 to TCDT).

[bit2] BDIS0

Process	Value	Function
Write	0	Enables the buffer function of the OCCP(0) register.
	1	Disables the buffer function of the OCCP(0) register.
Read	-	Reads the register setting.

[bit3] BDIS1

Process	Value	Function
Write	0	Enables the buffer function of the OCCP(1) register.
	1	Disables the buffer function of the OCCP(1) register.
Read	-	Reads the register setting.

The BDIS0 bit is a bit that specifies whether to enable or disable the buffer register function of OCCP(0).
The BDIS1 bit is a bit that specifies whether to enable or disable the buffer register function of OCCP(1).

Change the setting of these bits, while OCU's operation is disabled.

See "4.3.9 OCU Compare Value Store Register (OCCP)".

<Note>

When using FRT in Up/Down-count mode, make sure to enable the buffer function of OCCP and use it as the transfer mode at Zero value detection.

[bit4] IOE0

Process	Value	Function
Write	0	Does not generate interrupt, when "1" is set to IOP0.
	1	Generates interrupt, when "1" is set to IOP0.
Read	-	Reads the register setting.

[bit5] IOE1

Process	Value	Function
Write	0	Does not generate interrupt, when "1" is set to IOP1.
	1	Generates interrupt, when "1" is set to IOP1.
Read	-	Reads the register setting.

The IOE0 bit is a bit that specifies whether to notify CPU of the event that "1" is set to IOP0 as an interrupt (enabling interrupt) or not to notify it (disabling interrupt).

The IOE1 bit is a bit that specifies whether to notify CPU of the event that "1" is set to IOP1 as an interrupt (enabling interrupt) or not to notify it (disabling interrupt).

See "6.2 Treatment of Event Detect Register and Interrupt".

[bit6] IOP0

Process	Value	Function
Write	0	Clears this register to "0".
	1	Does nothing.
Read	0	Indicates that no match has been detected between FRT's count value and OCCP(0) value at OCU ch.(0).
	1	Indicates that a match has already been detected between FRT's count value and OCCP(0) value at OCU ch.(0).
Read at RMW access		"1" is always read.

[bit7] IOP1

Process	Value	Function
Write	0	Clears this register to "0".
	1	Does nothing.
Read	0	Indicates that no match has been detected between FRT's count value and OCCP(1) value at OCU ch.(1).
	1	Indicates that a match has already been detected between FRT's count value and OCCP(1) value at OCU ch.(1).
Read at RMW access		"1" is always read.

The IOP0 bit is a bit that is set to "1" when a match is detected between FRT's count value and OCCP(0) value when the operation of OCU ch.(0) is enabled.

The IOP1 bit is a bit that is set to "1" when a match is detected between FRT's count value and OCCP(1) value when the operation of OCU ch.(1) is enabled.

By reading from this bit, whether FRT's count value has reached the OCCP value or not can be determined.

This register can be cleared by writing "0".

This bit does nothing, if "1" is written. Always write "1" to the register when rewriting to another register in the same address area.

"1" is always read from this bit at RMW access.

See "6.2 Treatment of Event Detect Register and Interrupt".

<Note>

When FRT is in Up/Down-count mode, these registers are not set, even if FRT's count value has matched the OCCP value at its peak.

4.3.7. OCU Control Register B (OCSB)

OCSB is an 8-bit register that controls OCU's operation.
Each mounted channel has three registers: OCSB10, OCSB32 and OCSB54.
OCSB10 controls OCU ch.1 and OCU ch.0.
OCSB32 controls OCU ch.3 and OCU ch.2.
OCSB54 controls OCU ch.5 and OCU ch.4.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved	BTS1	BTS0	CMOD	Reserved	Reserved	OTD1	OTD0
Attribute	-	R/W	R/W	R/W	-	-	R/W	R/W
Initial Value	-	1	1	0	-	-	0	0

■ Functions of Register

[bit8] OTD0

Process	Value	Function
Write	0	Sets the output level of the RT(0) pin to the Low level, when OCSA.CST0=0. Does nothing, when OCSA.CST0=1.
	1	Sets the output level of the RT(0) pin to the High level, when OCSA.CST0=0. Does nothing, when OCSA.CST0=1.
Read	0	Indicates that the RT(0) output pin is in the Low-level output state.
	1	Indicates that the RT(0) output pin is in the High-level output state.

[bit9] OTD1

Process	Value	Function
Write	0	Sets the output level of the RT(1) pin to the Low level, when OCSA.CST1=0. Does nothing, when OCSA.CST1=1.
	1	Sets the output level of the RT(1) pin to the High level, when OCSA.CST1=0. Does nothing, when OCSA.CST1=1.
Read	0	Indicates that the RT(1) output pin is in the Low-level output state.
	1	Indicates that the RT(1) output pin is in the High-level output state.

The OTD0 bit is a bit that reads the state of the RT(0) output pin of OCU ch.(0) and sets its output level.

The OTD1 bit is a bit that reads the state of the RT(1) output pin of OCU ch.(1) and sets its output level.

The output level of the OCU output pins (RT0 to RT5) can be set by writing to these bits when OCU's operation is disabled. When OCU's operation is enabled, the writing to these bits is ignored. The read value of these bits indicates the output level of the OCU output pins, irrespective of OCU's operation state.

<Notes>

- After being processed by WFG, OCU's output pins (RT0 to RT5) become LSI's external output pins (RTO0 to RTO5). For this reason, the level of OCU's output pins does not match the level of LSI's external output pins in some of WFG's operation modes; therefore care must be taken. The state of LSI's external output pins can be read from the PDIR register of the I/O port block.
- Follow the procedure below to set the output level to Low by stopping OCU's operation when CST0=1 (OCU operation enabled) and OTD0=1 (High-level output).
  - No value can be written to OTD0 while OCU's operation is enabled; therefore, first write "0" to CST0 to stop OCU's operation.
  - Then, write "0" to OTD0 to set the output level to Low.

  It should be noted that if the above steps were reversed, the value written to OTD0 would be ignored. It should also be noted that if CST0=0 and OTD0=0 were written to the OCSA and OCSB registers by half-word access, the value written to OTD0 would be ignored because OCU's operation is enabled. Similarly, care must be taken to writing to OTD1.

[bit11:10] Reserved :Reserved bits

Process	Function
Write	"0" must be written at write access.
Read	"0" is read.

[bit12] CMOD

Process	Value	Function
Write	0	Writes "0" to this register.
	1	Writes "1" to this register.
Read	-	Reads the register setting.

The CMOD bit is a bit that selects OCU's operation mode in combination with OCSC:MOD0 to MOD5.

Change the setting of this bit, while OCU's operation is disabled.

For details of operation modes by this bit setting, see "4.4 Details of OCU Output Waveform".

When setting OCSB10:CMOD, the common setting applies to ch.1 and ch.0.

When setting OCSB32:CMOD, the common setting applies to ch.3 and ch.2.

When setting OCSB54:CMOD, the common setting applies to ch.5 and ch.4.

[bit13] BTS0

Process	Value	Function
Write	0	Performs buffer transfer of the OCCP(0) register upon Zero value detection by FRT.
	1	Performs buffer transfer of the OCCP(0) register upon Peak value detection by FRT.
Read	-	Reads the register setting.

[bit14] BTS1

Process	Value	Function
Write	0	Performs buffer transfer of the OCCP(1) register upon Zero value detection by FRT.
	1	Performs buffer transfer of the OCCP(1) register upon Peak value detection by FRT.
Read	-	Reads the register setting.

The BTS0 bit is a bit that specifies the timing of transfer from the buffer register to the OCCP(0) register when the buffer function of the OCCP(0) register is enabled.

The BTS1 bit is a bit that specifies the timing of transfer from the buffer register to the OCCP(1) register when the buffer function of the OCCP(1) register is enabled.

Change the setting of these bits while OCU's operation is disabled.

The setting of these bits has no meaning, when the buffer function is disabled (OCSA:BDIS1=1, OCSA:BDIS0=1).

See "4.3.9 OCU Compare Value Store Register (OCCP)".

<Note>

When using FRT in Up/Down-count mode, make sure to enable OCCP's buffer function and select the transfer mode for Zero value detection.

[bit15] Reserved:Reserved bit

Process	Function
Write	The written value is ignored.
Read	An undefined value is read.

4.3.8. OCU Control Register C (OCSC)

OCSC is an 8-bit register that controls OCU's operation.
This register controls all of OCU ch.0 to ch.5.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved	Reserved	MOD5	MOD4	MOD3	MOD2	MOD1	MOD0
Attribute	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	-	-	0	0	0	0	0	0

■ Functions of Register

[bit13:8] MOD5 to MOD0

Process	Value	Function
Write	0	Writes "0" to this register.
	1	Writes "1" to this register.
Read	-	Reads the register setting.

MOD0 and MOD1 determine the operation mode of OCU ch.0/ch.1 in combination with OCSB10:CMOD.

MOD2 and MOD3 determine the operation mode of OCU ch.2/ch.3 in combination with OCSB32:CMOD.

MOD4 and MOD5 determine the operation mode of OCU ch.4/ch.5 in combination with OCSB54:CMOD.

Change the setting of these bits while OCU's operation is disabled.

For the operation modes by these bits setting, see "4.4 Details of OCU Output Waveform".

[bit15:14] Reserved: Reserved bits

Process	Function
Write	The written value is ignored.
Read	An undefined value is read.

4.3.9. OCU Compare Value Store Register (OCCP)

OCCP is a 16-bit register that specifies the timing of changing OCU's output signal as the compare value of FRT's count value.
Each mounted channel has six registers: OCCP0 to OCCP5.
OCCP0 stores the compare value of OCU ch.0 (2-change mode, ch.1 compare value).
OCCP1 stores the compare value of OCU ch.1.
OCCP2 stores the compare value of OCU ch.2 (2-change mode, ch.3 compare value).
OCCP3 stores the compare value of OCU ch.3.
OCCP4 stores the compare value of OCU ch.4 (2-change mode, ch.5 compare value).
OCCP5 stores the compare value of OCU ch.5.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	OCCP[15:0]															
Attribute	R/W															
Initial Value	0x0000															

■ Functions of Register

[bit15:0] OCCP[15:0]

Process	Function
Write	Specifies the timing of changing OCU's output signal. Stores the written value to the buffer register.
Read	Reads the value in the OCCP register (not the value in the OCCP buffer register).

OCCP is a 16-bit register that specifies the timing of changing OCU's output signal as the compare value of FRT's count value.
When data is written to this address area, the data is first stored in the buffer register. And then, the data is transferred from the buffer register to the OCCP register under the following conditions.

When the buffer function is disabled:
Data is transferred immediately after it is written to the buffer register.
When the buffer function is enabled and the transfer upon Zero value detection is enabled:
Data is transferred, when FRT's Count is stopped or when FRT's count value has reached "0x0000".
When the buffer function is enabled and the transfer upon Peak value detection is enabled:
Data is transferred, when FRT's Count is stopped or when FRT's count value has matched the TCCP value.

The enabling/disabling of the buffer function and the timing of data transfer are determined by the value of the corresponding register OCSA:BDIS1/BDSI0 or OCSB:BTS1/BTS0.
When OCU's operation is enabled, the pulse width of OCU's output signal can be changed by rewriting to this register. When the buffer function is disabled, the written value can be immediately reflected on the OCCP register. When the buffer function is enabled, the settings in the OCCP register for multiple channels can be synchronized.
If data is read from this address area, the value in the OCCP register is read, rather than the value in the buffer register. Therefore, it should be noted that no bit can be rewritten by RMW access to this address area when the buffer function is enabled.
If "0x0000" or "0xFFFF" is written to this register when FRT is in Up/Down-count mode, a fixed value can be output. For details, see "4.4 Details of OCU Output Waveform".

4.3.10. WFG Control Register A (WFSA)

WFSA is a 16-bit register that controls WFG's operation.
Each mounted channel has three registers: WFSA10, WFSA32 and WFSA54.
WFSA10 controls WFG ch.10 (the output processing block of OCU ch.1 and OCU ch.0).
WFSA32 controls WFG ch.32 (the output processing block of OCU ch.3 and OCU ch.2).
WFSA54 controls WFG ch.54 (the output processing block of OCU ch.5 and OCU ch.4).
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved	Reserved	DMOD[1:0]		PGEN[1:0]		PSEL[1:0]	
Attribute	-	-	R/W		R/W		R/W	
Initial Value	-	-	00		00		00	

bit	7	6	5	4	3	2	1	0
Field	GTEN[1:0]		TMD[2:0]			DCK[2:0]		
Attribute	R/W		R/W			R/W		
Initial Value	00		000			000		

■ Functions of Register

[bit2:0] DCK[2:0]

Process	Value	Function
Write	000	Sets the count clock cycle of the WFG timer to the same value as PCLK.
	001	Sets the count clock cycle of the WFG timer to PCLK multiplied by 2.
	010	Sets the count clock cycle of the WFG timer to PCLK multiplied by 4.
	011	Sets the count clock cycle of the WFG timer to PCLK multiplied by 8.
	100	Sets the count clock cycle of the WFG timer to PCLK multiplied by 16.
	101	Sets the count clock cycle of the WFG timer to PCLK multiplied by 32.
	110	Sets the count clock cycle of the WFG timer to PCLK multiplied by 64
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

The DCK[2:0] bits are bit that sets the count clock cycle of the WFG timer.

Change the setting of these bits, while the WFG timer is stopping.

The count clock of the WFG timer is generated by dividing the PCLK in LSI by the prescaler. These bits set the division ratio of the prescaler.

The count clock cycle of the WFG timer is determined according to PCLK cycle and the clock division ratio set by this register.

The table below shows examples of DCK[2:0] settings and the count clock cycle of the WFG timer.

DCK[2:0]	Cycle Ratio	Count Clock Cycle of WFG Timer		
		PCLK=25 ns (40 MHz)	PCLK=33.3 ns (33 MHz)	PCLK=50 ns (20 MHz)
000	1	25 ns	30 ns	50 ns
001	2	50 ns	61 ns	100 ns
010	4	100 ns	121 ns	200 ns
011	8	200 ns	242 ns	400 ns
100	16	400 ns	485 ns	800 ns
101	32	800 ns	970 ns	1.6 µs
110	64	1.6 µs	1.9 µs	3.2 µs

[bit5:3] TMD[2:0]

Process	Value	Function
Write	000	Sets WFG's operation mode to Through mode.
	001	Sets WFG's operation mode to RT-PPG mode.
	010	Sets WFG's operation mode to Timer- PPG mode.
	100	Sets WFG's operation mode to RT-dead timer mode.
	111	Sets WFG's operation mode to PPG-dead timer mode.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

The TMD[2:0] bits are bit that selects WFG's operation mode.
For the operation modes by these bits setting, see "4.5 Details of WFG Output Waveform".
When WFG's operation mode is set to Through mode (TMD[2:0]=000) or RT-PPG mode (TMD[2:0]=001), the WFG timer can be used as an independent reload timer.
Change the setting of these bits while OCU and PPG timer unit to be connected are stopping. If the value set in this register is rewritten to a different value, the count state of the WFG timer is reset.

[bit7:6] GTEN[1:0]

Process	Value	Function
Write	00	Does not generate the CH_GATE signal.
	Other than above	Generates the CH_GATE signal. For details, see "4.5 Details of WFG Output Waveform".
Read	-	Reads the register setting.

GTEN[1:0] is a register that selects the output condition of the CH_GATE signal for each channel of WFG, in combination with WFSA:TMD[2:0].

This register has no meaning, when TMD[2:0] is set to 000,100. Change the setting, while OCU and PPG timer units to be connected are stopping.

The CH_GATE signal is generated based on the RT input signal and WFG timer operation at each channel of WFG (for details, see "4.5 Details of WFG Output Waveform").

The start trigger signal (GATE signal) for the PPG timer unit is generated from the CH_GATE signal (for details, see the section regarding PSEL[1:0]).

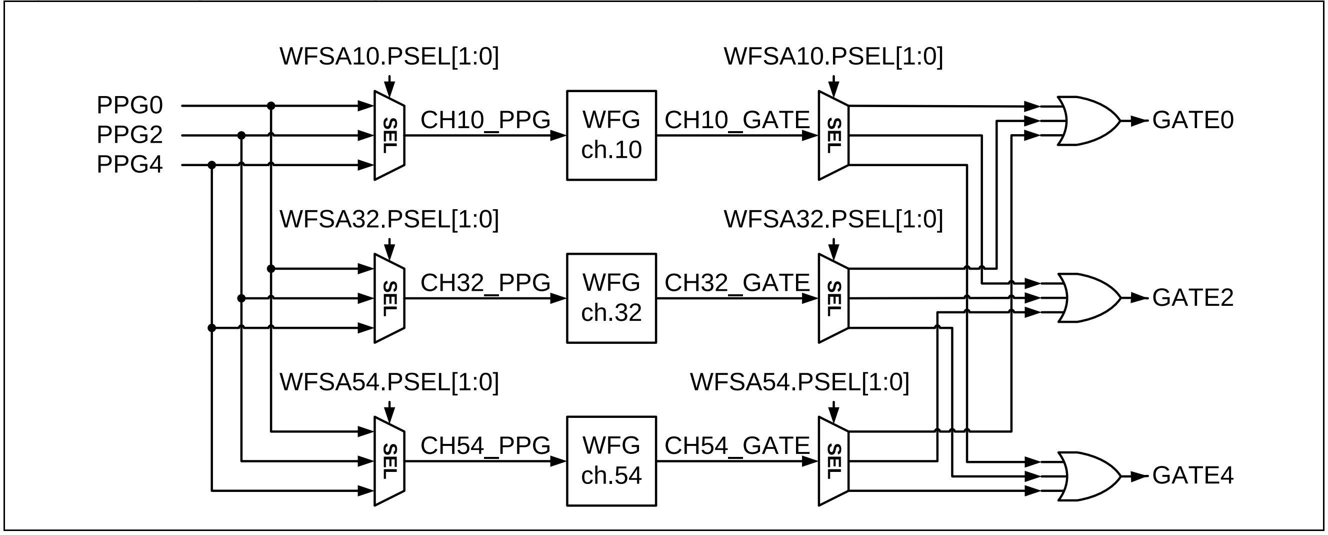
The PPG timer unit to be connected to WFG can start the output of the PPG signal, using the GATE signal, and each channel of WFG can superimpose the PPG signal from the PPG timer unit on the RTO signal output (for details, see the section regarding PSEL[1:0] and "4.5 Details of WFG Output Waveform").

[bit9:8] PSEL[1:0]

Process	Value	Function
Write	00	Sets the output destination of the GATE signal to ch.0 of the PPG timer unit. Sets the input source of the PPG signal to ch.0 of the PPG timer unit.
	01	Sets the output destination of the GATE signal to ch.2 of the PPG timer unit. Sets the input source of the PPG signal to ch.2 of the PPG timer unit.
	10	Sets the output destination of the GATE signal to ch.4 of the PPG timer unit. Sets the input source of the PPG signal to ch.4 of the PPG timer unit.
	11	Setting is prohibited.
Read	-	Reads the register setting.

The PSEL[1:0] bits are bit that selects the PPG timer unit to be used at each channel of WFG.
These bits select the PPG timer unit to be used as the output destination of the GATE signal and the input source of the PPG signal at once. Change the setting of this register, while OCU and PPG timer unit to be connected are stopping. Figure 4-6 shows a diagram of configuration of the PPG selector for MFTunit0.

Figure 4-6 Diagram of Configuration of PPG Selector for MFTunit0



The following section explains the configuration and operation of the PPG selector.

Each channel of WFG can output a trigger signal (CH_GATE signal) to start the PPG timer unit.

The CH10_GATE signal, CH32_GATE signal and CH54_GATE signal refer to the GATE signal for each channel of WFG, which has been generated at WFG ch.10, WFG ch.32 and WFG ch.54, respectively.

After its output is selected by WFSA:PSEL[1:0] for each PPG timer unit to be connected, each CH_GATE signal undergoes logic OR by PPG timer unit and is output to each PPG unit.

The GATE0 signal, GATE2 signal and GATE4 signal refer to the GATE signal that is output to ch.0, ch.2 and ch.4 of the PPG timer unit, respectively.

Each PPG timer unit can be started by the GATE signal and output the PPG signal.

The PPG0 signal, PPG2 signal and PPG4 signal refer to the PPG signal that is output from ch.0, ch.2 and ch.4 of the PPG timer unit, respectively, and input to WFG.

The CH10_PPG signal, CH32_PPG signal and CH54_PPG signal refer to the PPG signal that is used at WFG ch.10, WFG ch.32 and WFG ch.54 respectively, whose input has been selected by WFSA.PSEL[1:0].

- Setting example 1)
  WFSA10:PSEL[1:0]=00, WFSA32:PSEL[1:0]=00, and WFSA54:PSEL[1:0]=00 selects the common use of ch.0 of the PPG timer unit at all the channels of WFG.

  GATE0 becomes the logic OR signal of CH10_GATE, CH32_GATE and CH54_GATE. Both GATE2 and GATE4 are set to fixed Low output. Each channel of WFG instructs ch.0 of the PPG timer unit to start up.

  All of CH10_PPG, CH32_PPG and CH54_PPG become the PPG0 signal. Each channel of WFG uses the output signal of ch.0 of the PPG timer unit for waveform generation.

- Setting example 2)
  WFSA10:PSEL[1:0]=00, WFSA32:PSEL[1:0]=01, and WFSA54:PSEL[1:0]=10 selects the individual use of ch.0, ch.2 and ch.4 of the PPG timer unit for each channel of WFG.

  GATE0=CH10_GATE, GATE2=CH32_GATE, and GATE4=CH54_GATE are output, separately. Each channel instructs ch.0, ch.2 or ch.4 of the PPG timer unit to start up, individually.

  CH10_PPG = PPG0, CH32_PPG=PPG2, and CH54_PPG=PPG4 are set. Each channel of WFG uses the output signal of the corresponding PPG timer unit for waveform generation.

<Notes>

- The PSEL[1:0] bits are set differently between MFT unit0 and MFT unit1 and the channel number of the PPG timer unit to be connected is also different. The descriptions above are intended for the PSEL[1:0] bits in MFT unit0. For information about MFTunit1, see "6.1 Connection of Model Containing Multiple MFT's".
- To use the GATE signal, the PPG timer unit must be set beforehand. For details, see the chapter "PPG".
- Even without the use of the GATE signal, the PPG timer unit can start outputting upon instruction by CPU.

[bit11:10] PGEN[1:0]

Process	Value	Function
Write	00	Does not reflect the CH_PPG signal on WFG output (RTO output).
	Other than above	Specifies the condition to be used to reflect the CH_PPG signal on WFG output. For details of the reflection conditions, see "4.5 Details of WFG Output Waveform".
Read	-	Reads the register setting.

PGEN[1:0] are bit that specifies how to reflect the CH_PPG signal that is input to each channel of WFG on WFG output, in combination with TMD[2:0].

When WFG's operation mode is set to Through mode, the CH_PPG signal can be output to the RTO pin without any change, according to the setting of PGEN[1:0]. This register setting has no meaning, if TMD[2:0] is set to 100, 111. Change the setting, while OCU and PPG timer unit to be connected are stopping.

[bit13:12] DMOD[1:0]

Process	Value	Function
Write	00	Outputs RTO(1) and RTO(0) signals without changing the level.
	01	Outputs both RTO(1) and RTO(0) signals reversed.
	10	Outputs the RTO(0) signal reversed. Outputs the RTO(1) signal without changing the level.
	11	Outputs the RTO(0) signal without changing the level. Outputs the RTO(1) signal reversed.
Read	-	Reads the register setting.

This bit setting allows the polarity for RTO(0) and RTO(1) output signals to be selected. This bit is valid for any value of TMD[2:0]. Note that the scope of validity of the function of this bit is different from FM3 family products.

In RT dead timer mode, RT dead timer filter mode, PPG dead timer mode, and PPG dead timer filter mode (WFSA.TMD=100, 101, 110, 111), DMOD=00,01 is used when using IGBT, N-Ch driver×2, or other same-polarity driver. DMOD=10,11 is used when using MOSFET(N-Ch＋P-Ch) or other drivers with different polarity. Check and set the specifications for the driver that is connected.

<Notes>

· As shown in Figure 4-25 of 4.5 Details of WFG Output Waveform, if RT dead timer mode (WFSA.TMD=100), DMOD=10, or other incorrect settings are made, a short-circuit will occur between the power supply and GND.

[bit15:14] Reserved : Reserved bits

Has no effect on operation. Read value is undefined.

4.3.11. WFG Timer Value Register (WFTM)

WFTM is a 16-bit register that sets the initial value of the WFG timer.
Each mounted channel has three registers: WFTM10, WFTM32 and WFTM54.
WFTM10 sets the initial value of the WFG timer for WFG ch.10 (the output processing block of OCU ch.1 and ch.0).
WFTM32 sets the initial value of the WFG timer for WFG ch.32 (the output processing block of OCU ch.3 and ch.2).
WFTM54 sets the initial value of the WFG timer for WFG ch.54 (the output processing block of OCU ch.5 and ch.4).
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	WFTM[15:0]															
Attribute	R/W															
Initial Value	0x0000															

■ Functions of Register

[bit15:0] WFTM[15:0]

Process	Function
Write	Sets the initial value of the WFG timer. Setting "0x0000" means 65536.
Read	Reads the register setting.

WFTM register is a 16-bit register that sets the initial value of the WFG timer.
The operating time of the WFG timer can be set as shown below, according to the setting of WFTM.

Operating time of WFG timer = WFTM value × Operation clock cycle of WFG timer

When WFG's operation mode is Timer PPG mode, RT-dead timer mode or PPG-dead timer mode (WFSA:TMD[2:0]=010, 100, 111), the WFG timer loads the initial value from the WFTM register, starts Down-count operation, when instructed to start up, and then stops once the counting is completed.

- In Timer PPG mode:
  WFG timer counts the time set in the WFG timer operation flag.

- In RT-dead timer mode and PPG-dead timer mode:
  WFG timer counts the dead time of the non-overlap signal.

When WFG's operation mode is Through mode or RT-PPG mode (WFSA:TMD[2:0]=000, 001), the WFG timer is not used for generation of output waveforms. In these modes, therefore, it can be used as a reload timer that generates interrupts to CPU in the intervals set by WFTM. For information about how to use it as a reload timer, see "4.3.13 WFG Interrupt Control Register (WFIR)".

This register can be rewritten, regardless of whether the WFG timer is currently operating or stopping.
A new value rewritten to this register becomes valid from the next startup of the timer.

4.3.12. NZCL Control Register (NZCL)

NZCL is a 16-bit register that controls DTIF interrupt (interrupt for emergency motor shutdown by signal input from the DTTIX pin).
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved							
Attribute	-							
Initial Value	0x00							

bit	7	6	5	4	3	2	1	0
Field	Reserved			SDTI	NWS[2:0]			DTIE
Attribute	-			W	R/W			R/W
Initial Value	000			0	000			0

■ Functions of Register

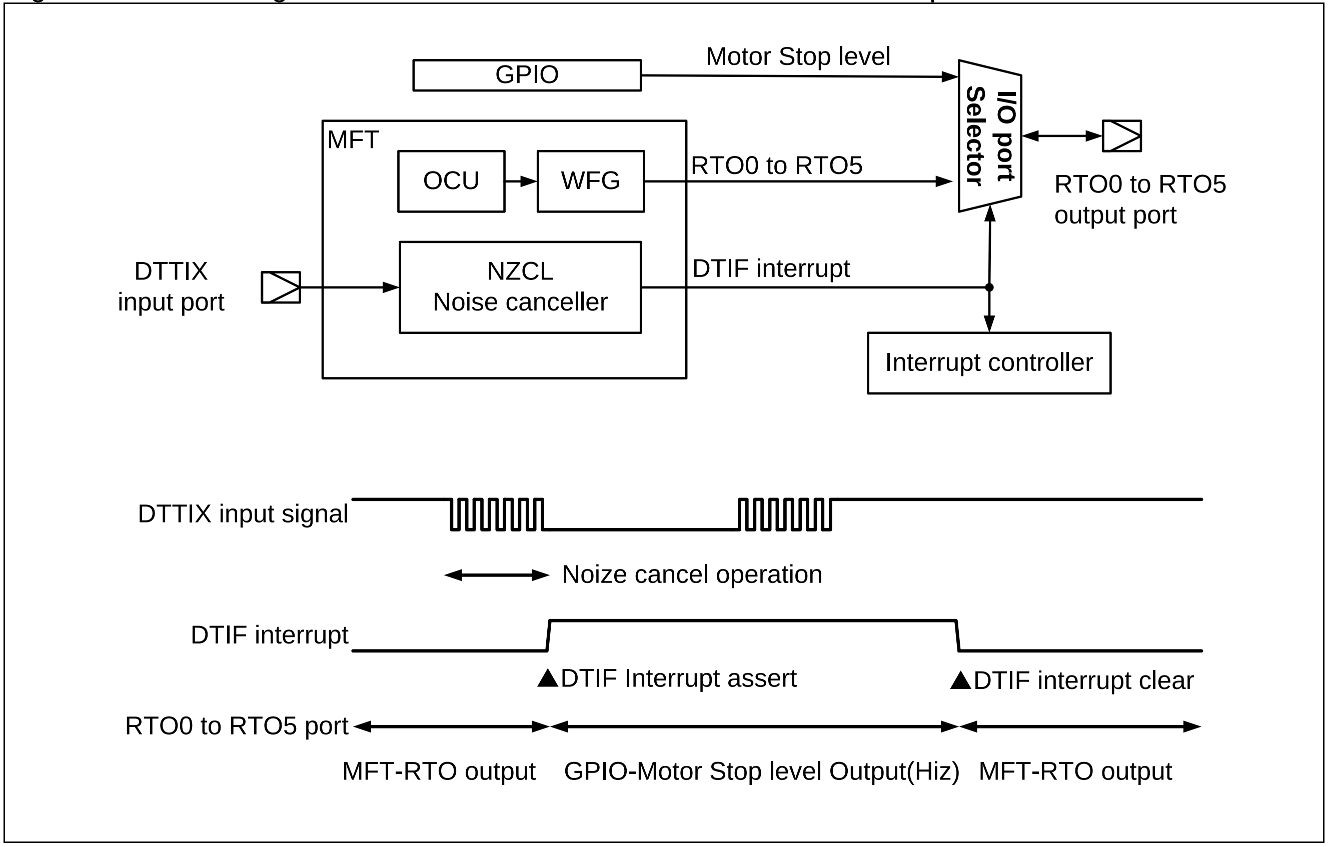
[bit0] DTIE

Process	Value	Function
Write	0	Ignores the signal input from the DTTIX pin.
	1	Generates DTIF interrupt by signal input from the DTTIX pin.
Read	-	Reads the register setting.

DTIE is a register that specifies whether or not to generate DTIF interrupt by signal input from the DTTIX pin.

Figure 4-7 shows a block diagram and time chart of the DTIX pin and DTIF interrupt.

Figure 4-7 Block Diagram and Time Chart of DTIX Pin and DTIF Interrupt



The DTTIX pin is a special pin dedicated to inputting an external interrupt signal for emergency motor shutdown. When the Low level is input, it recognizes the signal as a request for emergency motor shutdown.
The input signal of this pin is input to the noise canceller. If a Low-level pulse no less than the value set by the noise canceller is input, the WFIR.DTIF register is set, the DTIF interrupt signal is asserted, and an interrupt is generated to CPU.
The DTIF interrupt signal is connected to the interrupt controller and the I/O port selector.
The I/O port selector can switch the state of the RTO0 to RTO5 output pins to the setting state of the sharing GPIO port, while DTIF interrupt is being generated.
The signal required for emergency motor shutdown can be output to the RTO0 to RTO5 pins by setting the GPIO pin shared with the RTO0 to RTO5 pins to the motor non-operating level beforehand.
The generated interrupt signal is deasserted by clearing the WFIR.DTIF register (writing "1" to the WFIR.DTIC).

Table 4-5 shows a list of function settings of the GPIO pin.
PFR, DDR and PDOR in the table refer to the corresponding registers of the GPIO port that are shared with the RTO0 to RTO5 pins.

Table 4-5 Setting List of Motor Non-operating Level by DTTIX Pin Interrupt

	Setting of GPIO Register					DTIF Signal Level	State of RTO Pin
	PFR	EPFR1 [11:0]	EPFR1 [12]	DDR	PDOR		
When switching the output state of the pin by DTIF interrupt	1	101010101010 Or 010101010101	1	1	1	0	Output RTO0 to RTO5
						1	Output High level
				1	0	0	Output RTO0 to RTO5
						1	Output Low level
				0	don't care	0	Output RTO0 to RTO5
						1	Hi-Z state
When not switching the output state of the pin by DTIF interrupt			0	don't care	don't care	0	Output RTO0 to RTO5
						1	

- PFR, EPFR1[11:0] is the basic setting for using the LSI pin as RTO output of MFT.
- EPFR1[12] is a bit that specifies whether or not to switch the pin function by interrupt.
- The EPFR1 register controls the pin used in MFT unit0. In the case of MFT unit1, the EPFR2 is used.
- Setting the DDR, PDOR register specifies the motor non-operating level when the pin function is switched.

If the output state is not to be switched by DTIF interrupt (EPFR1[12]=0), the state of the output pin is not switched, but DTIF interrupt is generated; therefore, CPU can receive interrupt notification.

[bit3:1] NWS[2:0]

Process	Value	Function
Write	000	DTIF interrupt is generated immediately after Low-level input from the DTTIX pin. (No noise-canceling)
	001	Sets the noise-canceling width to 4 PCLK cycles.
	010	Sets the noise-canceling width to 8 PCLK cycles.
	011	Sets the noise-canceling width to 16 PCLK cycles.
	100	Sets the noise-canceling width to 32 PCLK cycles.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

The NWS[2:0] bits are bit that sets the noise-canceling width of the noise-canceller for the DTTIX pin input signal.

[bit4] SDTI

Process	Value	Function
Write	0	Does nothing.
	1	Forcibly generates DTIF interrupt, rather than by DTIE setting.
Read	-	"0" is always read.

The SDTI bit is a bit that generates DTIF interrupt by writing to the register via software.

Writing "1" to this register sets the WFIR:DTIF register and generates interrupt, irrespective of NZCL:DTIE setting and the state of the DTTIX pin. Writing to this register allows for the use of the output switch function for the RTO pin in the I/O port controller. The generated interrupt signal is deasserted by clearing the WFIR:DTIF register (i.e. writing "1" to the WFIR.DTIC register).

[bit15:5] Reserved: Reserved bits

"0" must be written at write access. Read value is "0".

4.3.13. WFG Interrupt Control Register (WFIR)

WFIR is a register that controls DTIF interrupt and the interrupt from the WFG timer.
This register is a special register dedicated to interrupt control, and each register bit is configured so that its state is not affected by writing "0".
For this reason, reading before writing to the register is not required. Also, each register bit is configured so that its state is not affected by writing the read value back.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	TMIS54	TMIE54	TMIC54	TMIF54	TMIS32	TMIE32	TMIC32	TMIF32
Attribute	W	R/W	W	R	W	R/W	W	R
Initial Value	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	TMIS10	TMIE10	TMIC10	TMIF10	Reserved		DTIC	DTIF
Attribute	W	R/W	W	R	-		W	R
Initial Value	0	0	0	0	00		0	0

■ Functions of Register

[bit0] DTIF

Process	Value	Function
Write	-	Writing is ignored.
Read	0	Indicates that DTIF interrupt has not been generated.
	1	Indicates that DTIF interrupt has been generated.

[bit1] DTIC

Process	Value	Function
Write	0	Does nothing.
	1	Clears WFIR.DTIF and deasserts the DTIF interrupt signal.
Read	-	"0" is always read.

The DTIF bit is a bit that checks the state of DTIF interrupt.

The DTIC bit is a bit that clears DTIF and deasserts the DTIF interrupt signal.

The DTIF bit is set by inputting the emergency motor shutdown signal from the DTTIX pin or writing "1" to the NZCL:SDTI register. When DTIF is set, the DTIF interrupt signal is asserted and an interrupt is generated to CPU.

Writing "1" to DTIC clears DTIF and deasserts the DTIF interrupt signal.

If interrupt processing has been performed by DTIF interrupt, make sure to clear DTIF when returning from the interrupt.

[bit3:2] Reserved : Reserved bits

"0" must be written at write access. Read value is "0".

[bit4] TMIF10

Process	Value	Function
Write	-	Writing is ignored.
Read	0	Indicates that WFG10 timer interrupt has not been generated.
	1	Indicates that WFG10 timer interrupt has been generated.

[bit5] TMIC10

Process	Value	Function
Write	0	Does nothing.
	1	Clears TMIF10 and deasserts the WFG10 timer interrupt signal.
Read	-	"0" is always read.

[bit6] TMIE10

Process	Value	Function
Write	0	Does nothing.
	1	Starts the WFG10 timer (or does nothing, if it has already been started).
Read	0	Indicates that the WFG10 timer is currently stopped.
	1	Indicates that the WFG10 timer is currently in operation.

[bit7] TMIS10

Process	Value	Function
Write	0	Does nothing.
	1	Stops the WFG10 timer (and also clears an interrupt at the same time, if it occurs, and deasserts the interrupt signal).
Read	-	"0" is always read.

The TMIF10 bit is a bit that checks the state of WFG10 timer interrupt.

The TMIC10 bit is a bit that clears WFG10 timer interrupt and deasserts the interrupt signal.

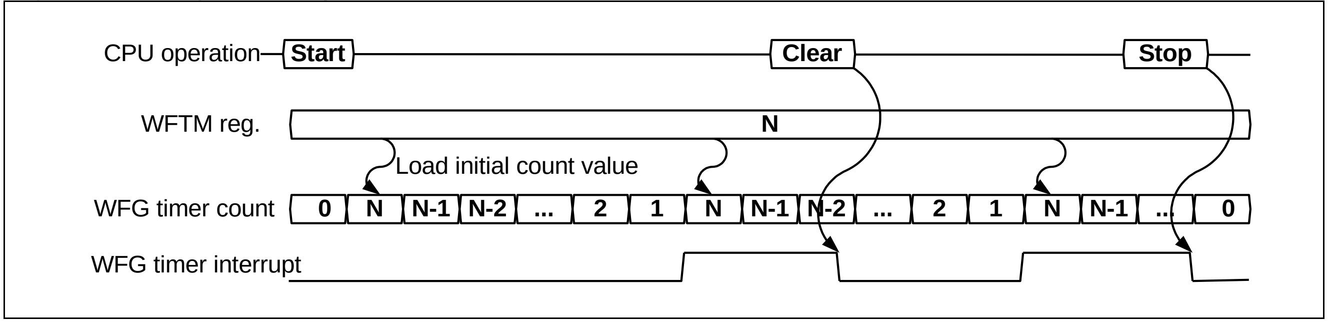
The TMIE10 bit is a bit that starts the WFG10 timer.

The TMIS10 bit is a bit that stops the WFG10 timer, clears the interrupt and deasserts the interrupt signal.

If the WFG timer for ch.10 of WFG is not used for waveform generation (WFSA10:TMD[2:0]=000, 001), the WFG10 timer can be used as an independent reload timer which generates interrupts regularly to CPU.

Figure 4-8 shows a diagram of the operation when the WFG timer is used as a reload timer.

Figure 4-8 Diagram of Operation when WFG Timer is Used as Reload Timer



Below is the procedure for using the WFG timer as a reload timer.

- First, set the initial value of the timer to the WFTM register and the clock division ratio to WFSA:DCK.
- Interval time of the interrupt generated from the timer = (WFTM value << WFSA:DCK) × PCLK cycle
- Writing "1" to WFIR:TMIE starts the timer.
- The WFG timer loads the initial value for the WFTM register, performs Down-count operation, and generates an interrupt when the count value is set to "1".
- At the same time, it reloads the initial value from the WFTM register and continues the Down-count operation.
- If "1" is read from TMIEn, it indicates that the WFG timer is operating as a reload timer.
- If "1" is read from TMIFn, it indicates that an interrupt has occurred.
- Writing "1" to TMICn allows TMIFn to be cleared and the interrupt signal to be deasserted.
- Writing "1" to TMISn allows the count operation of the WFG timer to be stopped and no further interrupts to occur.
- If an interrupt has already occurred when "1" is written to TMISn, the timer is stopped, TMIFn is cleared and the interrupt signal is deasserted at the same time.
- The value in the WFTM register can be rewritten during the timer operation. The changed value is reflected from the next reload time.
- If interrupt processing has been performed by WFG timer interrupt, make sure to clear TMIFn when returning from the interrupt.
- The following priority order applies to the processing, if "1" is written to TMISn, TMICn and TMIEn at the same time:
  (Highest priority) Stopping the timer > Clearing the timer interrupt > Starting the timer (Lowest priority)

[bit8] TMIF32

Process	Value	Function
Write	-	Writing is ignored.
Read	0	Indicates that WFG32 timer interrupt has not been generated.
	1	Indicates that WFG32 timer interrupt has been generated.

[bit9] TMIC32

Process	Value	Function
Write	0	Does nothing.
	1	Clears TMIF32 and deasserts the interrupt signal of the WFG32 timer.
Read	-	"0" is always read.

[bit10] TMIE32

Process	Value	Function
Write	0	Does nothing.
	1	Starts the WFG32 timer (or does nothing, if it has already been started).
Read	0	Indicates that the WFG32 timer is currently stopped.
	1	Indicates that the WFG32 timer is currently in operation.

[bit11] TMIS32

Process	Value	Function
Write	0	Does nothing.
	1	Stops the WFG32 timer (and also clears an interrupt at the same time, if it occurs, and deasserts the interrupt signal).
Read	-	"0" is always read.

The TMIF32 bit is a bit that checks the state of WFG32 timer interrupt.

The TMIC32 bit is a bit that clears WFG32 timer interrupt and deasserts the interrupt signal.

The TMIE32 bit is a bit that starts the WFG32 timer.

The TMIS32 bit is a bit that stops the WFG32 timer, clears the interrupt and deasserts the interrupt signal.

If the WFG timer for ch.32 of WFG is not used for waveform generation (WFSA32:TMD[2:0]=000, 001), the WFG32 timer can be used as an independent reload timer which generates interrupts regularly to CPU.

These bits are used in the same way as for WFIR:TMIF10, TMIC10, TMIE10 and TMIS10.

[bit12] TMIF54

Process	Value	Function
Write	-	Writing is ignored.
Read	0	Indicates that the WFG54 timer interrupt has not been generated.
	1	Indicates that the WFG54 timer interrupt has been generated.

[bit13] TMIC54

Process	Value	Function
Write	0	Does nothing.
	1	Clears TMIF54 and deasserts the interrupt signal of the WFG54 timer.
Read	-	"0" is always read.

[bit14] TMIE54

Process	Value	Function
Write	0	Does nothing.
	1	Starts the WFG54 timer (or does nothing, if it has already been started).
Read	0	Indicates that the WFG54 timer is currently stopped.
	1	Indicates that the WFG54 timer is currently in operation.

[bit15] TMIS54

Process	Value	Function
Write	0	Does nothing.
	1	Stops the WFG54 timer (and also clears an interrupt at the same time, if it occurs, and deasserts the interrupt signal).
Read	-	"0" is always read.

The TMIF54 bit is a bit that checks the state of WFG54 timer interrupt.

The TMIC54 bit is a bit that clears WFG54 timer interrupt and deasserts the interrupt signal.

The TMIE54 bit is a bit that starts the WFG54 timer.

The TMIS54 bit is a bit that stops the WFG54 timer, clears the interrupt and deasserts the interrupt signal.

If the WFG timer for ch.54 of WFG is not used for waveform generation (WFSA54:TMD[2:0]=000, 001), the WFG54 timer can be used as an independent reload timer which generates interrupts regularly to CPU.

These bits are used in the same way as for WFIR:TMIF10, WFIR:TMIC10, WFIR:TMIE10 and WFIR:TMIS10.

4.3.14. ICU Connecting FRT Select Register (ICFS)

ICFS is an 8-bit register that selects and sets FRT to be connected to ICU.
Each mounted channel has two registers: ICFS10 and ICFS32.
ICFS10 controls ICU ch.1 and ICU ch.0.
ICFS32 controls ICU ch.3 and ICU ch.2.
ICFS10 is located at an even-numbered address, while ICFS32 is located at an odd-numbered address; therefore, their bit positions are [7:0] and [15:8].

■ Configuration of Register

bit	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0
Field	FSI1[3:0]				FSI0[3:0]			
Attribute	R/W				R/W			
Initial Value	0000				0000			

■ Functions of Register

[bit3:0/11:8] FSI0[3:0]

Process	Value	Function
Write	0000	Connects FRT ch.0 to ICU ch.(0).
	0001	Connects FRT ch.1 to ICU ch.(0).
	0010	Connects FRT ch.2 to ICU ch.(0).
	0011 0100	For models with multiple MFT units: Connects FRT of an external MFT. For models with one MFT unit: Setting is prohibited.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

[bit7:4/15:12] FSI1[3:0]

Process	Value	Function
Write	0000	Connects FRT ch.0 to ICU ch.(1).
	0001	Connects FRT ch.1 to ICU ch.(1).
	0010	Connects FRT ch.2 to ICU ch.(1).
	0011 0100	For models with multiple MFT units: Connects FRT of an external MFT. For models with one MFT unit: Setting is prohibited.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

The FSI0[3:0] bits are bit that selects FRT to be connected to ICU-ch.(0) for use.
The FSI1[3:0] bits are bit that selects FRT to be connected to ICU-ch.(1) for use.
For models with multiple MFT units, the connection to FRT that exists in another MFT unit can be selected. For related settings, see "6.1 Connection of Model Containing Multiple MFT's".
Change the setting of these bits, while the operation of the ICU to be connected is disabled.

4.3.15. ICU Control Register A (ICSA)

ICSA is an 8-bit register that controls ICU's operation.
Each mounted channel has two registers: ICSA10 and ICSA32.
ICSA10 controls ICU ch.1 and ICU ch.0.
ICSA32 controls ICU ch.3 and ICU ch.2.

■ Configuration of Register

bit	7	6	5	4	3	2	1	0
Field	ICP1	ICP0	ICE1	ICE0	EG1[1:0]		EG0[1:0]	
Attribute	R/W	R/W	R/W	R/W	R/W		R/W	
Initial Value	0	0	0	0	00		00	

■ Functions of Register

[bit1:0] EG0[1:0]

Process	Value	Function
Write	00	Disables the operation of ICU ch.(0). Ignores IC(0) signal input.
	01	Enables the operation of ICU ch.(0). Treats only the rising edge of IC(0) signal input as a valid edge.
	10	Enables the operation of ICU ch.(0). Treats only the falling edge of IC(0) signal input as a valid edge.
	11	Enables the operation of ICU ch.(0). Treats both the rising and falling edges of IC(0) signal input as valid edges.
Read	-	Reads the register setting.

[bit3:2] EG1[1:0]

Process	Value	Function
Write	00	Disables the operation of ICU ch.(1). Ignores IC(1) signal input.
	01	Enables the operation of ICU ch.(1). Treats only the rising edge of IC(1) signal input as a valid edge.
	10	Enables the operation of ICU ch.(1). Treats only the falling edge of IC(1) signal input as a valid edge.
	11	Enables the operation of ICU ch.(1). Treats both the rising and falling edges of IC(1) signal input as valid edges.
Read	-	Reads the register setting.

The EG0[1:0] bits are bit that enables/disables the operation of ICU ch.(0) and selects a valid edge(s).
The EG1[1:0] bits are bit that enables/disables the operation of ICU ch.(1) and selects a valid edge(s).

If a valid edge is detected at the input signal when ICU's operation is enabled, it performs the capture operation that captures FRT's count output to the ICCP register. At the same time, it notifies CPU that the valid edge has been detected. The valid edge of the input signal can be selected from the rising edge only, the falling edge only, or both rising and falling edges.

When the operation is disabled, it does nothing and ignores the input signal.

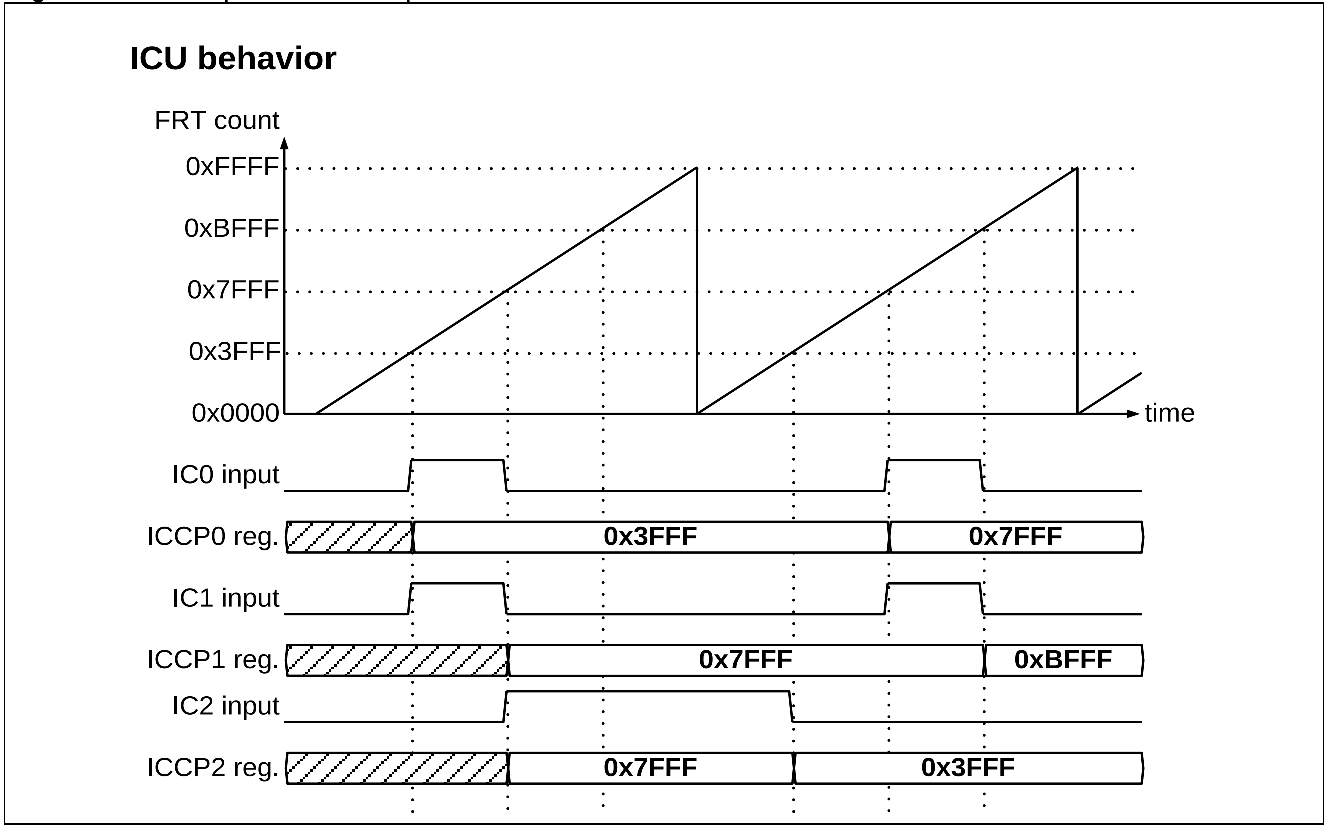
Figure 4-9 shows an example of ICU's operation.

ICU ch.0 indicates the operation to be performed upon detection of the rising edge of IC0 signal input.

ICU ch.1 indicates the operation to be performed upon detection of the falling edge of IC1 signal input.

ICU ch.2 indicates the operation to be performed upon detection of both the rising and falling edges of IC signal input.

Figure 4-9 Example of ICU's Operation



[bit4] ICE0

Process	Value	Function
Write	0	Does not generate interrupt, when "1" is set to ICP0.
	1	Generates interrupt, when "1" is set to ICP0.
Read	-	Reads the register setting.

[bit5] ICE1

Process	Value	Function
Write	0	Does not generate interrupt, when "1" is set to ICP1.
	1	Generates interrupt, when "1" is set to ICP1.
Read	-	Reads the register setting

The ICE0 bit is a bit that specifies whether to notify CPU of the event that "1" is set to ICP0 as an interrupt (enabling interrupt) or not to notify it (disabling interrupt).
The ICE1 bit is a bit that specifies whether to notify CPU of the event that "1" is set to ICP1 as an interrupt (enabling interrupt) or not to notify it (disabling interrupt).
See "6.2 Treatment of Event Detect Register and Interrupt".

[bit6] ICP0

Process	Value	Function
Write	0	Clears this register to "0".
	1	Does nothing.
Read	0	Indicates that no valid edge has been detected at ICU ch.(0) and no capture operation has been performed.
	1	Indicates that a valid edge has been detected at ICU ch.(0) and the capture operation has been performed.
Read at RMW access		"1" is always read.

[bit7] ICP1

Process	Value	Function
Write	0	Clears this register to "0".
	1	Does nothing.
Read	0	Indicates that no valid edge has been detected at ICU ch.(1) and no capture operation has been performed.
	1	Indicates that a valid edge has been detected at ICU ch.(1) and the capture operation has been performed.
Read at RMW access		"1" is always read.

The ICP0 bit is a bit to which "1" is set upon detection of a valid edge/capture operation, when the operation of ICU-ch.(0) is enabled.
The ICP1 bit is a bit to which "1" is set upon detection of a valid edge/capture operation, when the operation of ICU-ch.(1) is enabled.
By reading from this bit, it can be determined whether or not a valid edge has been detected and the capture operation has been performed.
This bit can be cleared by writing "0".
This bit does nothing, if "1" is written. Always write "1" to the register when rewriting to another register in the same address area.
"1" is always read from this bit at RMW access.
See "6.2 Treatment of Event Detect Register and Interrupt".

4.3.16. ICU Control Register B (ICSB)

ICSB is an 8-bit register that reads the operation state of ICU.
Each mounted channel has two registers: ICSB10 and ICSB32.
ICSB10 reads the operation state of ICU ch.1 and ICU ch.0.
ICSB32 reads the operation state of ICU ch.3 and ICU ch.2.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	IEI1	IE10
Attribute	-	-	-	-	-	-	R	R
Initial Value	-	-	-	-	-	-	0	0

■ Functions of Register

[bit8] IEI0

Process	Value	Function
Write	-	Writing is ignored.
Read	0	Indicates that the latest capture operation of ICU ch.(0) was performed at a falling edge.
	1	Indicates that the latest capture operation of ICU ch.(0) was performed at a rising edge.

[bit9] IEI1

Process	Value	Function
Write	-	Writing is ignored.
Read	0	Indicates that the latest capture operation of ICU ch.(1) was performed at a falling edge.
	1	Indicates that the latest capture operation of ICU ch.(1) was performed at a rising edge.

The IEI0 bit is a bit that indicates the latest valid edge of ICU ch.(0).

The IEI1 bit is a bit that indicates the latest valid edge of ICU ch.(1).

By reading from this bit, at which edge the latest capture operation was performed can be determined.

As the initial value of this bit is "0", "0" can be read if the capture operation has never been performed. It is also updated every time the valid edge of an input signal is detected. After the capture operation is performed, it is necessary to read from this bit before the next valid edge.

[bit15:10] Reserved: Reserved bits

The written value is ignored. Read value is undefined.

4.3.17. ICU Capture value store register (ICCP)

ICCP is a 16-bit register that reads the value captured to ICU.
Each mounted channel has four registers: ICCP0, ICCP1, ICCP2 and ICCP3.
ICCP0 stores the capture value of ICU ch.0.
ICCP1 stores the capture value of ICU ch.1.
ICCP2 stores the capture value of ICU ch.2.
ICCP3 stores the capture value of ICU ch.3.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	ICCP[15:0]															
Attribute	R															
Initial Value	0xXXXX															

■ Functions of Register

[bit15:0] ICCP[15:0]

Process	Function
Write	Writing is ignored.
Read	Reads the data captured to ICU.

ICCP is a 16-bit register that reads the value captured at each channel of ICU.

As the initial value of these bits are undefined, a meaningless value is read if the capture operation has never been performed.

This register is updated every time the valid edge of an input signal is detected. After the capture operation is performed, it is necessary to read from this register before the next valid edge.

4.3.18. ADCMP Control Register A (ACSA)

ACSA is a 16-bit register that controls ADCMP's operation.
This register controls all of ch.0, ch.1 and ch.2 of ADCMP.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved	Reserved	SEL2[1:0]		SEL1[1:0]		SEL0[1:0]	
Attribute	-	-	R/W		R/W		R/W	
Initial Value	0	0	00		00		00	

bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	CE2[1:0]		CE1[1:0]		CE0[1:0]	
Attribute	-	-	R/W		R/W		R/W	
Initial Value	0	0	00		00		00	

■ Functions of Register

[bit1:0] CE0[1:0]

Process	Value	Function
Write	00	Disables the operation of ADCMP ch.0.
	01	Enables the operation of ADCMP ch.0. Connects FRT ch.0 to ADCMP ch.0.
	10	Enables the operation of ADCMP ch.0. Connects FRT ch.1 to ADCMP ch.0.
	11	Enables the operation of ADCMP ch.0. Connects FRT ch.2 to ADCMP ch.0.
Read	-	Reads the register setting.

[bit3:2] CE1[1:0]

Process	Value	Function
Write	00	Disables the operation of ADCMP ch.1.
	01	Enables the operation of ADCMP ch.1. Connects FRT ch.0 to ADCMP ch.1.
	10	Enables the operation of ADCMP ch.1. Connects FRT ch.1 to ADCMP ch.1.
	11	Enables the operation of ADCMP ch.1. Connects FRT ch.2 to ADCMP ch.1.
Read	-	Reads the register setting.

[bit5:4] CE2[1:0]

Process	Value	Function
Write	00	Disables the operation of ADCMP ch.2.
	01	Enables the operation of ADCMP ch.2. Connects FRT ch.0 to ADCMP ch.2.
	10	Enables the operation of ADCMP ch.2. Connects FRT ch.1 to ADCMP ch.2.
	11	Enables the operation of ADCMP ch.2. Connects FRT ch.2 to ADCMP ch.2.
Read	-	Reads the register setting.

CE0[1:0], CE1[1:0], and CE2[1:0] are registers that specify whether to enable or disable the operation of ADCMP ch.0, ADCMP ch.1, and ADCMP ch.2 respectively and also select the FRT to be connected.

When the operation is enabled, ADCMP outputs the start instruction signal of AD conversion to ADC at the timing when there is a match between the compare value specified by the ADCMPDN register and the count value of the FRT connected. To enable the operation of ADCMP, make sure to set the values of the ACCP and ACCPDN registers beforehand.

ADCMP ch.0 instructs ADC unit0 to start AD conversion.

ADCMP ch.1 instructs ADC unit1 to start AD conversion.

ADCMP ch.2 instructs ADC unit2 to start AD conversion.

The AD conversion start signal output from ADCMP is connected to each ADC unit, after its output is selected by ATSA.

When the operation is disabled, ADCMP does nothing. ADCMP can only select the connection to the FRT that exists within its own MFT.

If the buffer function of ACCP and ACCPDN registers is to be used, use FRT-ch.0 for FRT to be connected.

[bit7:6] Reserved : Reserved bits

"0" must be written at write access. Read value is "0".

[bit9:8] SEL0[1:0]

Process	Value	Function
Write	00	Instructs AD to be started, when FRT is in Up-count/Peak/Down-count state and it matches the setting value of ACCP0. Ignores the setting value of ACCPDN0.
	01	Instructs AD to be started, when FRT is in Up-count state and it matches the setting value of ACCP0. Ignores the setting value of ACCPDN0.
	10	Instructs AD to be started, when FRT is in Peak/Down-count state and it matches the setting value of ACCP0. Ignores the setting value of ACCPDN0.
	11	Instructs AD to be started, when FRT is in Up-count state and it matches the setting value of ACCP0. Or instructs AD to be started, when FRT is in Peak/Down-count state and it matches the setting value of ACCPDN0.
Read	-	Reads the register setting.

[bit11:10] SEL1[1:0]

Process	Value	Function
Write	00	Instructs AD to be started, when FRT is in Up-count/Peak/Down-count state and it matches the setting value of ACCP1. Ignores the setting value of ACCPDN1.
	01	Instructs AD to be started, when FRT is in Up-count state and it matches the setting value of ACCP1. Ignores the setting value of ACCPDN1.
	10	Instructs AD to be started, when FRT is in Peak/Down-count state and it matches the setting value of ACCP1. Ignores the setting value of ACCPDN1.
	11	Instructs AD to be started, when FRT is in Up-count state and it matches the setting value of ACCP1. Or instructs AD to be started, when FRT is in Peak/Down-count state and it matches the setting value of ACCPDN1.
Read	-	Reads the register setting.

[bit13:12] SEL2[1:0]

Process	Value	Function
Write	00	Instructs AD to be started, when FRT is in Up-count/Peak/Down-count state and it matches the setting value of ACCP2. Ignores the setting value of ACCPDN2.
	01	Instructs AD to be started, when FRT is in Up-count state and it matches the setting value of ACCP2. Ignores the setting value of ACCPDN2.
	10	Instructs AD to be started, when FRT is in Peak/Down-count state and it matches the setting value of ACCP2. Ignores the setting value of ACCPDN2.
	11	Instructs AD to be started, when FRT is in Up-count state and it matches the setting value of ACCP2. Or instructs AD to be started, when FRT is in Peak/Down-count state and it matches the setting value of ACCPDN2.
Read	-	Reads the register setting.

ACSA:SEL0[1:0], ACSA:SEL1[1:0], and ACSA:SEL2[1:0] are registers that specify which count state FRT should be in to instruct AD conversion to be started at each channel of ADCMP.

Change the setting of these register, when the operation of ADCMP to be connected is disabled.

When using FRT in Up-count mode, set "00" to these registers for use.

Figure 4-10 shows an example of the operation when SEL is set to 00.

Figure 4-10 Example of Operation when SEL=00

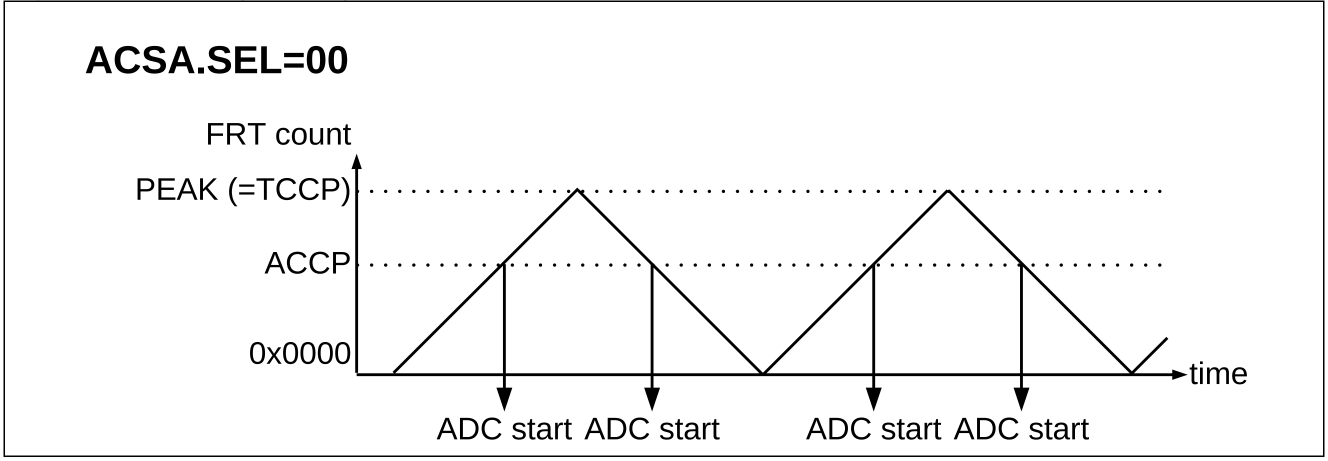


Figure 4-11 shows an example of the operation when SEL is set to 01.

Figure 4-11 Example of Operation when SEL=01

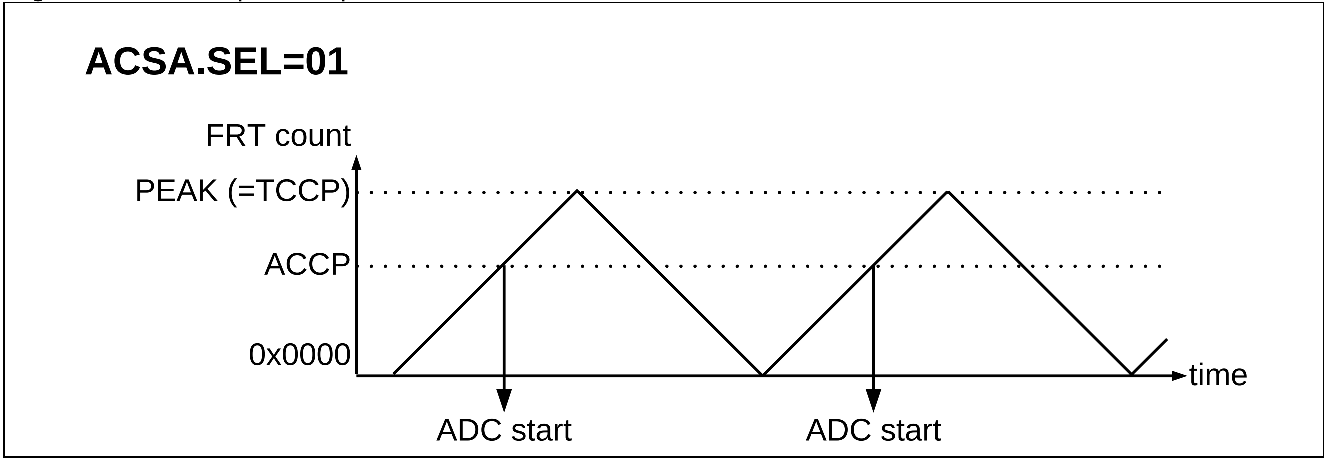


Figure 4-12 shows an example of the operation when SEL is set to 10.

Figure 4-12 Example of Operation when SEL=10

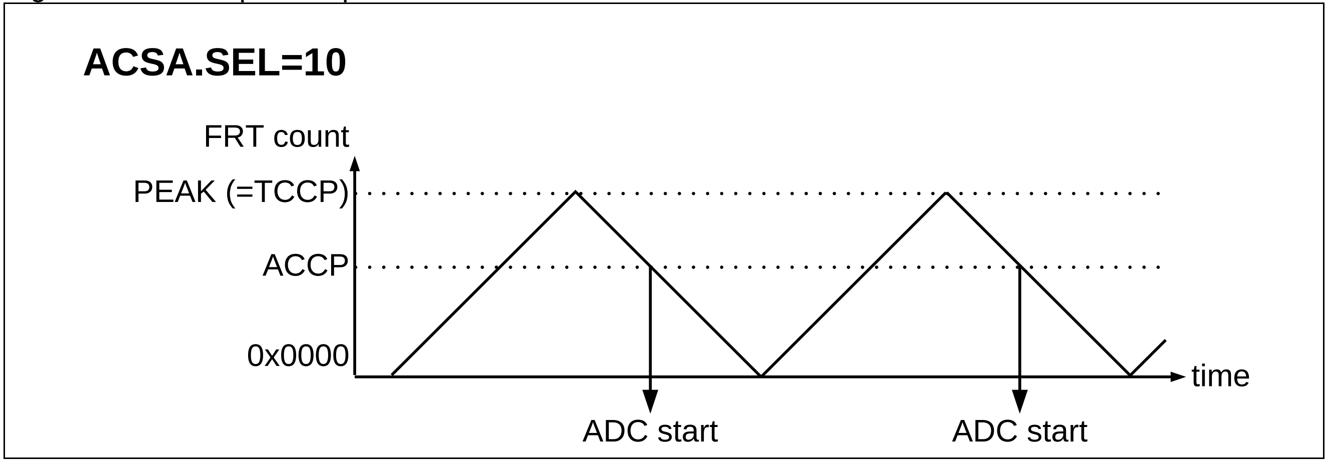
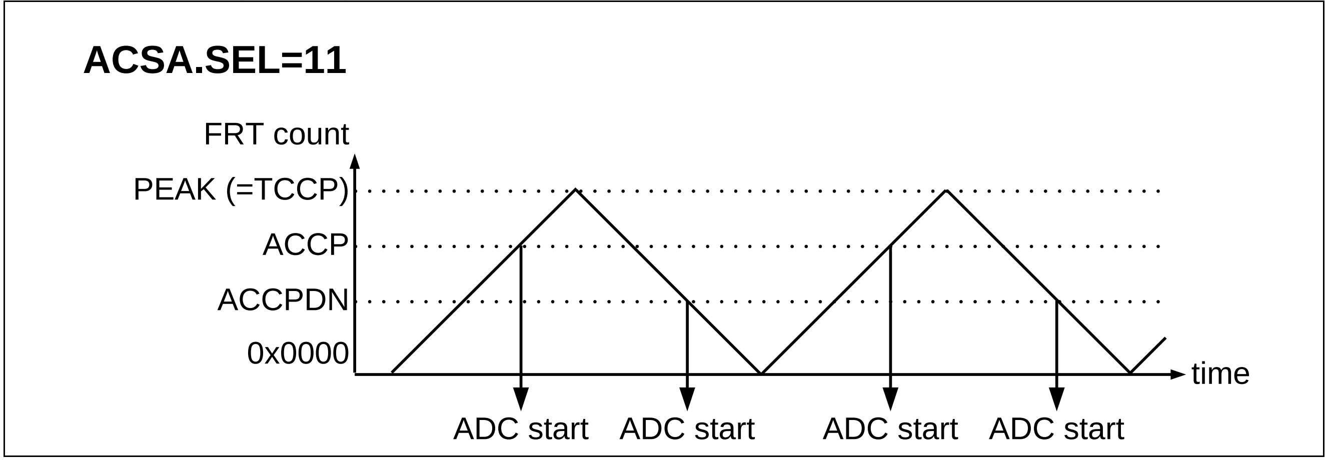


Figure 4-13 shows an example of the operation when SEL is set to 11.

Figure 4-13 Example of Operation when SEL=11



[bit15:14] Reserved : Reserved bits

Process	Function
Write	"0" must be written at write access.
Read	"0" is read.

4.3.19. ADCMP Control Register B (ACSB)

ACSB is an 8-bit register that controls ADCMP's operation.
This register controls all of ch.0, ch.1 and ch.2 of ADCMP.

■ Configuration of Register

bit	7	6	5	4	3	2	1	0
Field	Reserved	BTS2	BTS1	BTS0	Reserved	BDIS2	BDIS1	BDIS0
Attribute	-	R/W	R/W	R/W	-	R/W	R/W	R/W
Initial Value	0	0	0	0	0	1	1	1

■ Functions of Register

[bit0] BDIS0

Process	Value	Function
Write	0	Enables the buffer function of the ACCP0 and ACCPDN0 registers.
	1	Disables the buffer function of the ACCP0 and ACCPDN0 registers.
Read	-	Reads the register setting.

[bit1] BDIS1

Process	Value	Function
Write	0	Enables the buffer function of the ACCP1 and ACCPDN1 registers.
	1	Disables the buffer function of the ACCP1 and ACCPDN1 registers.
Read	-	Reads the register setting.

[bit2] BDIS2

Process	Value	Function
Write	0	Enables the buffer function of the ACCP2 and ACCPDN2 registers.
	1	Disables the buffer function of the ACCP2 and ACCPDN2 registers.
Read	-	Reads the register setting.

The BDIS0 bit is a bit that specifies whether to enable or disable the buffer function of the ACCP0 and ACCPDN0 registers.
The BDIS1 bit is a bit that specifies whether to enable or disable the buffer function of the ACCP1 and ACCPDN1 registers.
The BDIS2 bit is a bit that specifies whether to enable or disable the buffer function of the ACCP2 and ACCPDN2 registers.
Change the setting of these bits, when the operation of ADCMP to be connected is disabled.
If the buffer function of ACCP and ACCPDN registers is to be used, use FRT ch.0 for FRT to be connected.
See "4.3.20 ADCMP Compare Value Store Register (ACCP)" and "4.3.21 ADCMP Compare Value Store Register, Down-count Direction Only (ACCPDN)".

[bit3] Reserved: Reserved bit

"0" must be written at write access. Read value is "0".

[bit4] BTS0

Process	Value	Function
Write	0	Performs buffer transfer of the ACCP0 and ACCPDN0 registers upon Zero value detection by FRT.
	1	Performs buffer transfer of the ACCP0 and ACCPDN0 registers upon Peak value detection by FRT.
Read	-	Reads the register setting.

[bit5] BTS1

Process	Value	Function
Write	0	Performs buffer transfer of the ACCP1 and ACCPDN1 registers upon Zero value detection by FRT.
	1	Performs buffer transfer of the ACCP1 and ACCPDN1 registers upon Peak value detection by FRT.
Read	-	Reads the register setting.

[bit6] BTS2

Process	Value	Function
Write	0	Performs buffer transfer of the ACCP2 and ACCPDN2 registers upon Zero value detection by FRT.
	1	Performs buffer transfer of the ACCP2 and ACCPDN2 registers upon Peak value detection by FRT.
Read	-	Reads the register setting.

The ACSB:BTS0 bit is a bit that specifies the timing of transferring data from the buffer register to the ACCP0 and ACCPDN0 registers when the buffer function is enabled.

The ACSB:BTS1 bit is a bit that specifies the timing of transferring data from the buffer register to the ACCP1 and ACCPDN1 registers when the buffer function is enabled.

The ACSB:BTS2 bit is a bit that specifies the timing of transferring data from the buffer register to the ACCP2 and ACCPDN2 registers when the buffer function is enabled.

Change the setting of these bits, when the operation of ADCMP to be connected is disabled.

See "4.3.20 ADCMP Compare Value Store Register (ACCP)" and "4.3.21 ADCMP Compare Value Store Register, Down-count Direction Only (ACCPDN)".

[bit7] Reserved : Reserved bit

"0" must be written at write access. Read value is "0".

4.3.20. ADCMP Compare Value Store Register (ACCP)

ACCP is a 16-bit register that specifies the timing of starting AD conversion at ADCMP as the compare value of the FRT count value.
Each mounted channel has three registers: ACCP0, ACCP1 and ACCP2.
ACCP0 stores the compare value of ADCMP ch.0.
ACCP1 stores the compare value of ADCMP ch.1.
ACCP2 stores the compare value of ADCMP ch.2.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	ACCP[15:0]															
Attribute	R/W															
Initial Value	0x0000															

■ Functions of Register

[bit15:0] ACCP[15:0]

Process	Function
Write	Specifies the timing of starting AD conversion. The value is written to the ACCP buffer register.
Read	Reads the value in the ACCP register (not the value in the ACCP buffer register).

ACCP is a register that specifies the timing of starting AD conversion.
Each specifies the timing of starting AD conversion in combination with the settings of ACSA:SEL0[1:0], ACSA:SEL1[1:0], and ACSA:SEL2[1:0].

When data is written to this address area, the data is first stored in the buffer register. And then, the data is transferred from the buffer register to the ACCP register under the following conditions.

When the buffer function is disabled:
Data is transferred immediately after it is written to the buffer register.

When the buffer function is enabled and the transfer upon Zero value detection is enabled:
Data is transferred, when FRT's Count is stopped or when FRT's count value has reached "0x0000".

When the buffer function is enabled and the transfer upon Peak value detection is enabled:
Data is transferred, when FRT's Count is stopped or when FRT's count value has matched the TCCP value.

The enabling/disabling of the buffer function and the timing of data transfer are determined by the value of the corresponding register ACSB:BDIS0, BDIS1, BDIS2, BTS0, BDIS1, or BDIS2.

During FRT's count operation, the timing of starting AD conversion can be changed by rewriting to this register. When the buffer function is disabled, the written value can be immediately reflected on the ACCP register. When the buffer function is enabled, the settings in the ACCP register for multiple channels can be synchronized.

If data is read from this address area, the value in the ACCP register is read, rather than the value in the buffer register. Therefore, it should be noted that no bit can be rewritten by RMW access to this address area when the buffer function is enabled.

<Notes>

- AD conversion cannot be started by writing "0x0000" to this register.
- As the initial value of this register is "0x0000", make sure to rewrite it to another value before use.
- To start AD conversion upon Zero value detection, use the starting method by the TCSB:AD0E, TCSB:AD1E and TCSB:AD2E registers.
- If the buffer function of ACCP register is to be used, use FRT ch.0 for FRT to be connected.
- It should be noted that when "01", "10", "11" are set to ACSA:SEL0[1:0], SEL1[1:0], SEL[1:0], FRT's Peak value (=TCCP) will be ignored, even if it is set to ACCP.

4.3.21. ADCMP Compare Value Store Register, Down-count Direction Only (ACCPDN)

ACCPDN is a 16-bit register that specifies the timing of starting AD conversion at ADCMP as the compare value of the FRT count value.
Each mounted channel has three registers: ACCPDN0, ACCPDN1 and ACCPDN2.
ACCPDN0 stores the compare value of ADCMP ch.0.
ACCPDN1 stores the compare value of ADCMP ch.1.
ACCPDN2 stores the compare value of ADCMP ch.2.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	ACCPDN[15:0]															
Attribute	R/W															
Initial Value	0x0000															

■ Functions of Register

[bit15:0] ACCPDN[15:0]

Process	Function
Write	Specifies the timing of starting AD conversion. The value is written to the ACCPDN buffer register.
Read	Reads the value in the ACCPDN register (not the value in the ACCPDN buffer register).

The ACCPDN register is a register that specifies the timing of starting AD conversion.

Each specifies the timing of starting AD conversion, only when the setting values of ACSA:SEL0[1:0], ACSA:SEL1[1:0], ACSA:SEL2[1:0] is 11.

When data is written to this address area, the data is first stored in the buffer register. And then, the data is transferred from the buffer register to the ACCP register under the following conditions.

When the buffer function is disabled:
Data is transferred immediately after it is written to the buffer register.

When the buffer function is enabled and the transfer upon Zero value detection is enabled:
Data is transferred, when FRT's Count is stopped or when FRT's count value has reached "0x0000".

When the buffer function is enabled and the transfer upon Peak value detection is enabled:
Data is transferred, when FRT's Count is stopped or when FRT's count value has matched the TCCP value.

The enabling/disabling of the buffer function and the timing of data transfer are determined by the value of the corresponding register ACSB:BDIS0, BDIS1, BDIS2, BTS0, BDIS1, or BDIS2.

During FRT's count operation, the timing of starting AD conversion can be changed by rewriting to this register. When the buffer function is disabled, the written value can be immediately reflected on the ACCPDN register. When the buffer function is enabled, the settings in the ACCPDN register for multiple channels can be synchronized.

If data is read from this address area, the value in the ACCPDN register is read, rather than the value in the buffer register. Therefore, it should be noted that no bit can be rewritten by RMW access to this address area when the buffer function is enabled.

<Notes>

- AD conversion cannot be started by writing "0x0000" to this register.
- If the buffer function of ACCPDN register is to be used, use FRT ch.0 for FRT to be connected.

4.3.22. ADC Start Trigger Select Register (ATSA)

ATSA is a 16-bit register that selects ADC's start signal which is output from MFT.
This register is used to select a start trigger for ADC unit0, unit1 and unit2.
It should be noted that this register does not allow for byte access.

■ Configuration of Register

bit	15	14	13	12	11	10	9	8
Field	Reserved		AD2P		AD1P		AD0P	
Attribute	-		R/W		R/W		R/W	
Initial Value	-		0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	Reserved		AD2S		AD1S		AD0S	
Attribute	-		R/W		R/W		R/W	
Initial Value	-		0	0	0	0	0	0

■ Functions of Register

[bit1:0] AD0S[1:0]

Process	Value	Function
Write	00	Selects the start signal of ADCMP ch.0 as ADC unit0 scan conversion start signal.
	01	Selects the logic OR signal of FRT ch.0 to ch.2 start signal as ADC unit0 scan conversion start signal.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

[bit3:2] AD1S[1:0]

Process	Value	Function
Write	00	Selects the start signal of ADCMP ch.1 as ADC unit1 scan conversion start signal.
	01	Selects the logic OR signal of FRT ch.0 to ch.2 start signal as ADC unit1 scan conversion start signal.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

[bit5:4] AD2S[1:0]

Process	Value	Function
Write	00	Selects the start signal of ADCMP ch.2 as ADC unit2 scan conversion start signal.
	01	Selects the logic OR signal of FRT ch.0 to ch.2 start signal as ADC unit2 scan conversion start signal.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

The AD0S[1:0] bits are bit that selects the start signal to be used to start the scan conversion of ADC unit0.

The AD1S[1:0] bits are bit that selects the start signal to be used to start the scan conversion of ADC unit1.

The AD2S[1:0] bits are bit that selects the start signal to be used to start the scan conversion of ADC unit2.

The starting method used for ADC's scan conversion start signal that is output from MFT can be selected from starting by ADCMP or starting by FRT ch.0 to ch.2. The above is selected by the setting in these bits. Change the setting of these bits, when the operation of ADCMP to be connected is disabled.

For models containing multiple MFT's, the ADC scan conversion start signal from MFT undergoes logic OR for each MFT unit, and then it is connected to ADC. For details, see the chapter "A/D Converter" in "Analog Macro Part".

[bit7:6] Reserved : Reserved bits

The written value is ignored. Read value is undefined.

[bit9:8] AD0P[1:0]

Process	Value	Function
Write	00	Selects the start signal of ADCMP ch.0 as ADC unit0 priority conversion start signal.
	01	Selects the logic OR signal of FRT-ch.0 to ch.2 as ADC unit0 priority conversion start signal.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

[bit11:10] AD1P[1:0]

Process	Value	Function
Write	00	Selects the start signal of ADCMP ch.1 as ADC unit1 priority conversion start signal.
	01	Selects the logic OR signal of FRT-ch.0 to ch.2 as ADC unit1 priority conversion start signal.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

[bit13:12] AD2P[1:0]

Process	Value	Function
Write	00	Selects the start signal of ADCMP ch.2 as ADC unit2 priority conversion start signal.
	01	Selects the logic OR signal of FRT-ch.0 to ch.2 as ADC unit2 priority conversion start signal.
	Other than above	Setting is prohibited.
Read	-	Reads the register setting.

The AD0P[1:0] bits are bit that selects the start signal to be used to start priority conversion of ADC unit0.

The AD1P[1:0] bits are bit that selects the start signal to be used to start priority conversion of ADC unit1.

The AD2P[1:0] bits are bit that selects the start signal to be used to start priority conversion of ADC unit2.

The starting method used for ADC's priority conversion start signal that is output from MFT can be selected from starting by ADCMP or starting by FRT ch.0 to ch.2. The above is selected by the setting in these bits. Change the setting of these bits, when the operation of ADCMP to be connected is disabled.

For models containing multiple MFT's, the ADC priority conversion start signal from MFT undergoes logic OR for each MFT unit, and then it is connected to ADC. For details, see the chapter "A/D Converter" in "Analog Macro Part".

[bit15:14] Reserved: Reserved bits

The written value is ignored. Read value is undefined.

4.4. Details of OCU Output Waveform

This section provides details of the output waveform of the RT output signal in each mode of OCU.

■ List of OCU Operation Modes

The operation modes of the OCU are selected by the following register settings. Table 4-6 shows a list of register setting values and the operation modes of OCU-ch.(0) and OCU-ch.(1).

Table 4-6 Register Setting Values and the Operation Modes of OCU-ch.(0) and OCU-ch.(1)

Register Setting				Operation Mode Selected	
TCSA: MODE -Ch.(1) (*1)	TCSA: MODE -Ch.(0) (*2)	OCSB: CMOD (*3)	OCSC: MOD (*4)	CH(1) Operation Mode	CH(0) Operation Mode
0	0	0	00	Up-count mode (1-change)	Up-count mode (1-change)
0	0	1	00	Up-count mode (2-change)	Up-count mode (1-change)
0	1	0	01	Up-count mode (1-change)	Up/Down-count mode (Active High)
1	0	0	10	Up/Down-count mode (Active High)	Up-count mode (1-change)
1	0	1	10	Up/Down-count mode (Active Low)	Up-count mode (1-change)
1	1	0	11	Up/Down-count mode (Active High)	Up/Down-count mode (Active High)
1	1	1	11	Up/Down-count mode (Active Low)	Up/Down-count mode (Active Low)

*1 TCSA:MODE-ch.(1) indicates the TCSA:MODE value of FRT to be connected to OCU ch.(1) selected by the OCFS register.

*2 TCSA:MODE-ch.(0) indicates the TCSA:MODE value of FRT to be connected to OCU ch.(0) selected by the OCFS register.

*3 OCSB:CMOD indicates the OCSB10:CMOD value for ch.1-ch.0. It indicates the OCSB32:CMOD value for ch.3-ch.2. It indicates the OCSB54:CMOD value for ch.5-ch.4.

*4 OCSC:MOD indicates the OCSC:MOD[1:0] value for ch.1-ch.0. It indicates the OCSC:MOD[3:2] value for ch.3-ch.2. It indicates the OCSC:MOD[5:4] value for ch.5-ch.4.

*5 OCSB:CMOD and OCSC:MOD[5:0] cannot be used in combinations other than listed above.

*6 OCU ch.(0) cannot use Up-count mode (2-change).

■ List of Changes of the RT(0) and RT(1) Signals in OCU Operation Modes

When each channel of OCU is in the state of Operation enabled, if the FRT counter value matches the OCCP register value, the output signal level changes. In addition, the changes of the output signal level are determined by the operation mode of OCU, the value of OCCP, and the count state of FRT. Table 4-7 shows a list of OCU ch.(0) operation modes, register settings and RT(0) signal outputs. Table 4-8 shows a list of OCU ch.(1) operation modes, register settings and RT(1) signal outputs.

Table 4-7 Details of OCU-ch.(0) Operation and RT(0) Signal Outputs

Name of Operation Mode	OCCP(0) value				
	0x0000	0xFFFF	Other than 0x0000 & 0xFFFF		
			Up	Peak	Down
Up-count mode (1-change)	M:Rev U:No	M:Rev U:No	M:Rev U:No	M:Rev U:No	-
Up/Down-count mode (Active High)	All-Act	All-Ina	M:Act U:No	M:No U:No (*7)	M:Ina U:No
Up/Down-count mode (Active Low)	All-Act	All-Ina	M:Act U:No		M:Ina U:No

Table 4-8 Details of OCU-ch.(1) Operation and RT(1) Signal Outputs

Name of Operation Mode	OCCP(1) value					OCCP(0) value
	0x0000	0xFFFF	Other than 0x0000, 0xFFFF			
			Up	Peak	Down	
Up-count mode (1-change)	M:Rev U:No	M:Rev U:No	M:Rev U:No	M:Rev U:No	-	-
Up-count mode (2-change)	M:Rev U:No	M:Rev U:No	M:Rev U:No	M:Rev U:No		M:Rev U:No
Up/Down-count mode (Active High)	All-Act	All-Ina	M:Act U:No	M:No U:No (*7)	M:Ina U:No	-
Up/Down-count mode (Active Low)	All-Act	All-Ina	M:Act U:No		M:Ina U:No	-

* Meanings of symbols in Table 4-7 and Table 4-8

Up:	Operation when FRT is up-counting
Peak:	Operation when FRT's count value is the Peak value (=TCCP value)
Down:	Operation when FRT is down-counting
M:	Operation when FRT's count value matches the OCCP value
U:	Operation when FRT's count value does not match the OCCP value
Rev:	Change of the output signal level to the Reversed level.
Act:	Change of the output signal level to the Active level. No change, if the previous output level was already Active.
Ina:	Change of the output signal level to the Inactive level. No change, if the previous output level was already Inactive.
No:	No change of the output signal level.
All-Act:	Change of the output signal level to the Active level, while the OCCP value is that value.
All-Ina:	Change of the output signal level to the Inactive level, while the OCCP value is that value.

*7 In Up/Down count mode, if the peak FRT count value matches OCCP value, the RT(0) and RT(1) output signals do not change, and the OCSA.IOP0 and IOP1 flags are not set.

■ Up-count Mode (1-change)

When Up-count mode (1-change) is selected, the following operation applies.

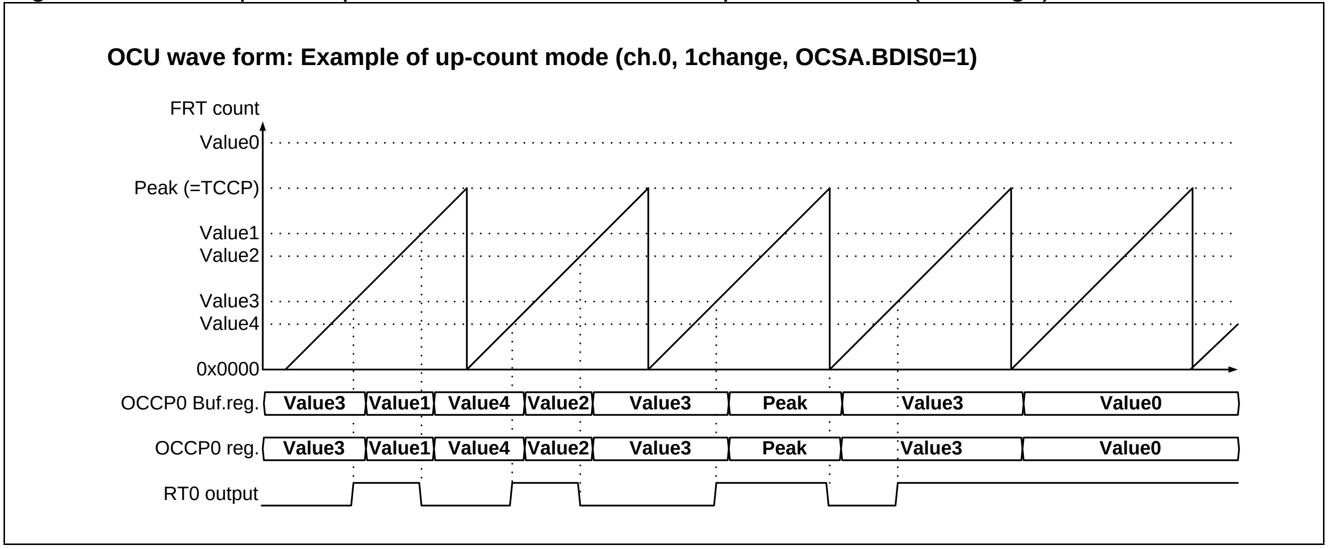
Regardless of FRT's count state, the output level of the RT(0) signal is reversed when FRT's count value matches OCCP(0).

Regardless of FRT's count state, the output level of the RT(1) signal is reversed when FRT's count value matches OCCP(1).

In this mode, OCU-ch.(0) and ch.(1) can operate independently from each other.

Figure 4-14 shows an example of operation waveform when OCU-ch.0 is in Up-count mode (1-change). This figure illustrates the state in which the buffer function of the OCCP0 register is disabled.

Figure 4-14 Example of Operation Waveform in OCU Up-count Mode (1-change)



<Note>

A note on Up-count mode (1-change) is as follows:

If a value larger than the Peak value of FRT's counter (e.g. Value0 in Figure 4-14) is set to OCCP, the output does not change.

■ Up-count Mode (2-change)

When Up-count mode (2-change) is selected, the following operation applies.

Regardless of FRT's count state, the output level of the RT(1) signal is reversed when FRT's count value matches OCCP(0) or OCCP(1).

This mode can be used only by OCU ch.(1), not by ch.(0). Also, as OCU ch.(0) and ch.(1) perform interlocked operation, they cannot operate independently from each other. If 2-change mode is selected for OCU ch(1), OCU ch.(0) operates in 1-change mode to perform the operation that changes according to the OCCP(0) value.

Figure 4-15 Example 1 of Operation Waveform in OCU Up-count Mode (2-change)

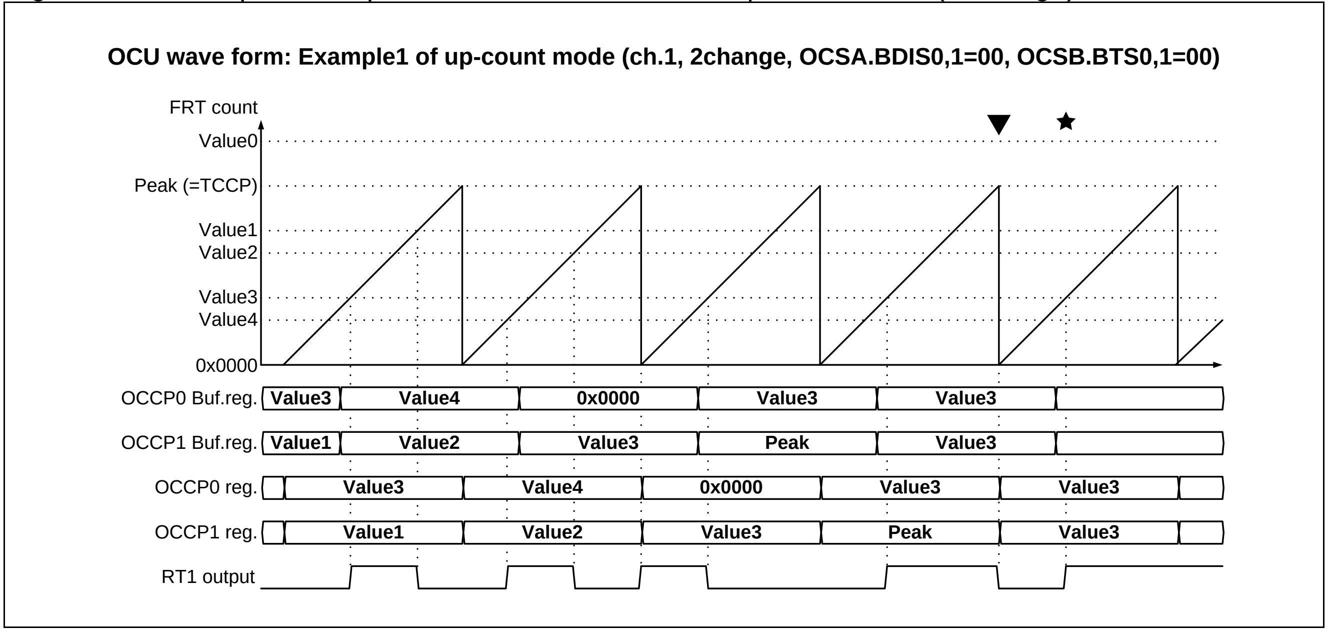


Figure 4-16 Example 2 of Operation Waveform in OCU Up-count Mode (2-change)

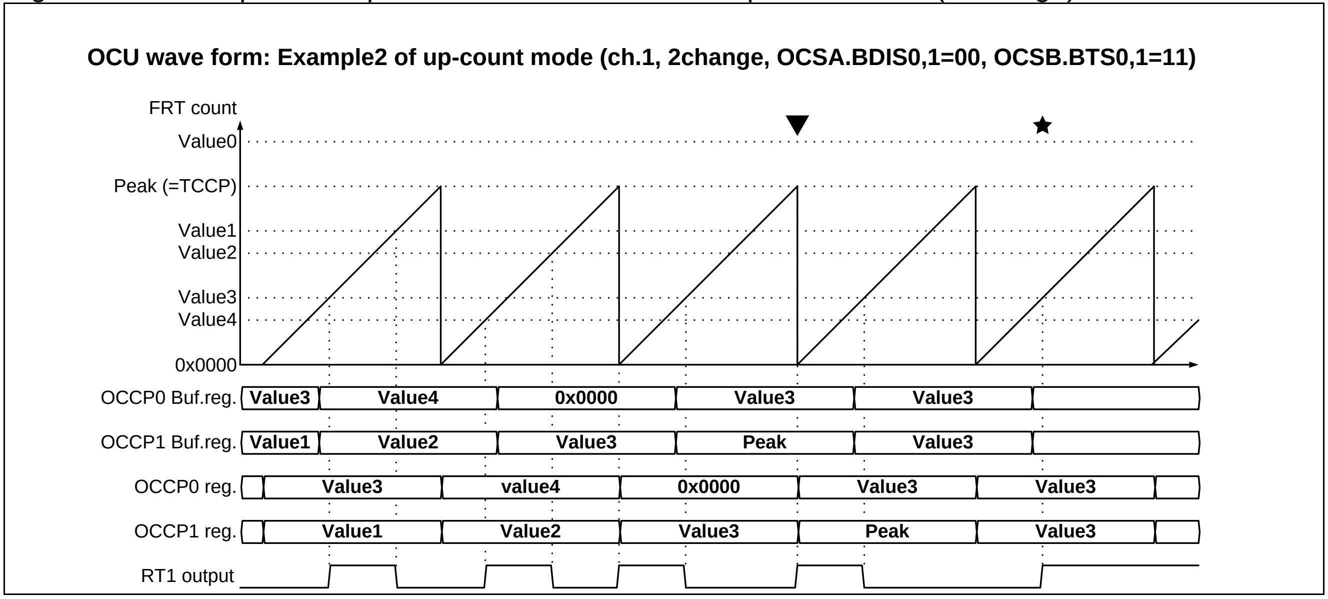


Figure 4-15 shows Example 1 of the operation waveform when OCU ch.1 is in Up-count mode (2-change). This figure is based on the conditions that the buffer function of the OCCP1 register is enabled and Zero value transfer is set.

Figure 4-16 shows Example 2 of the operation waveform when OCU ch.1 is in Up-count mode (2-change). This figure is based on the conditions that the buffer function of the OCCP1 register is enabled and Peak value transfer is set.

Due to the difference in the timing of OCCP data transfer between Figure 4-15 and Figure 4-16, they operate differently when the Peak value is set to OCCP1.

In the case of Figure 4-15, data is transferred from the OCCP1 buffer register to the OCCP1 register, when FRT's counter value is "0x0000". The output level is reversed at the timing indicated by ▼, under the condition: OCCP1=Peak value.

In the case of Figure 4-16, data is transferred from the OCCP1 buffer register to the OCCP1 register, when FRT's counter value reaches the Peak value. As the register values are compared immediately after the transfer, the output level is reversed at the timing indicated by ▼, under the condition: OCCP1=Peak value.

<Notes>

Notes on Up-count mode (2-change) are as follows:

- If a value larger than the Peak value of FRT's counter (e.g. Value0 in Figure 4-15) is set to OCCP, the output does not change.
- If the same value is set to OCCP(0) and OCCP(1), the output level is reversed at the timing indicated by ★, as shown in Figure 4-15 and Figure 4-16.
- It is necessary to set the operation enable flags of both OCSA:CST0 and OCSA:CST1.
- OCSA:IOP0 is set when FRT's count value matches OCCP(0).
- OCSA:IOP1 is set when FRT's count value matches OCCP(1).
- It is necessary to apply the same settings for which FRT is to be connected; whether to enable or disable the buffer function; and the transfer timing, for both OCU ch.(0) and ch.(1).

■ Up/Down-count Mode (Active-High)

When Up/Down-count mode (Active-High) is selected, the following operation applies.

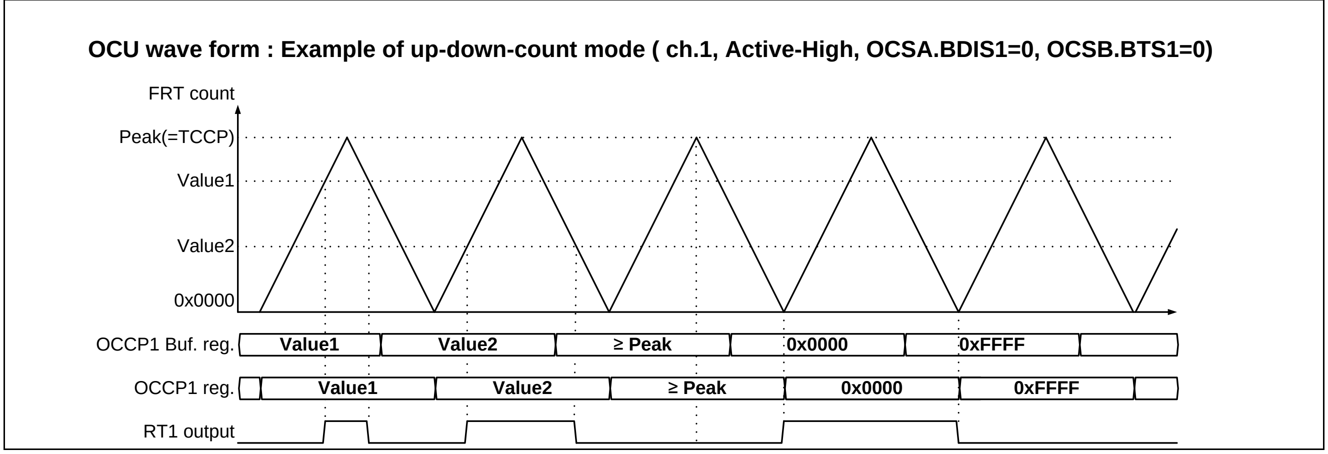
If FRT is up-counting, the output level of RT(0) signal is changed to Active level (High), when FRT's count value matches OCCP(0). If FRT is down-counting, the output level is changed to Inactive level (Low), when FRT's count value matches OCCP(0). When "0x0000" is set to OCCP(0), all-active (High) is output, and then, it returns to Inactive level (Low), when a value other than "0x0000" is set. As long as "0xFFFF" is set to OCCP(0), the Inactive level (Low) is output.

The operation of the RT(1) signal output is the same as for RT(0) according to the value in OCCP(1).

In this mode, OCU ch.(0) and ch.(1) can operate independently from each other.

Figure 4-17 shows an example of the operation waveform when OCU ch.1 is in Up/Down-count mode (Active-High). This figure is based on the conditions that the buffer function of the OCCP register is enabled and the Zero value transfer is selected.

Figure 4-17 Example of Operation Waveform in OCU Up/Down-count Mode (Active-High)



<Notes>

Notes on Up/Down-count mode (Active-High) are as follows:

- In this mode, enable the buffer function of OCCP and select Zero value transfer mode for use.
- If "0x0000" is set to OCCP register when OCU's operation is enabled, the output level is changed to Active level immediately, regardless of FRT's count value.
- If a value no less than the Peak value of FRT's counter is set to OCCP, the output level does not change, even when FRT's counter value reaches its peak value. Also, the IOP0 and IOP1 registers are not set.

■ Up/Down-count Mode (Active-Low)

When Up/Down-count mode (Active- Low) is selected, the following operation applies.

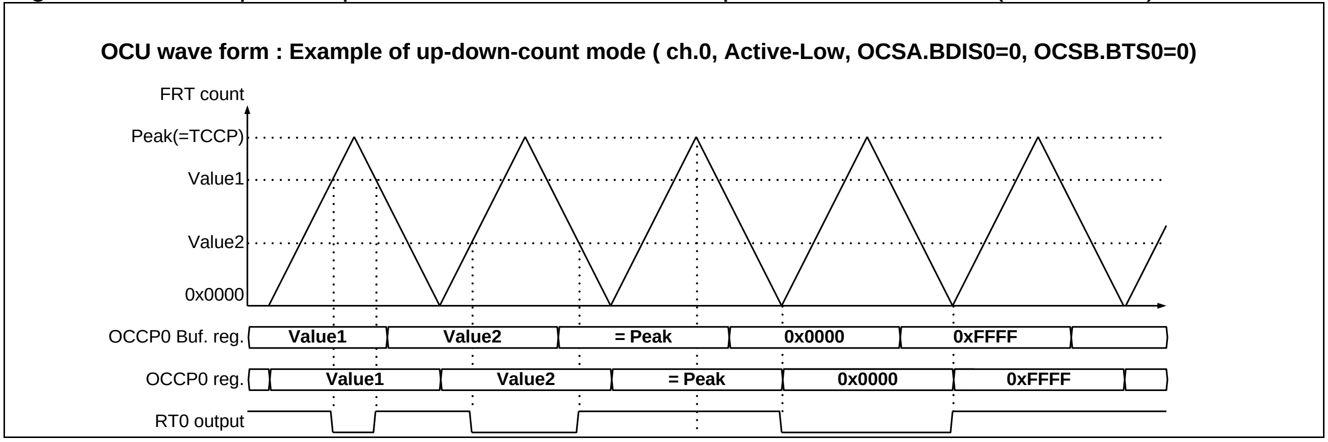
If FRT is up-counting, the output level of RT(0) signal is changed to Active level (Low), when FRT's count value matches OCCP(0). If FRT is down-counting, the output level is changed to Inactive level (High), when FRT's count value matches OCCP(0). When "0x0000" is set to OCCP(0), all-active (Low) is output, and then, it returns to Inactive level (High), when a value other than "0x0000" is set. As long as "0xFFFF" is set to OCCP(0), the Inactive level (High) is output.

The operation of the RT(1) signal output is the same as for RT(0) according to the value in OCCP(1).

In this mode, OCU ch.(0) and ch.(1) can operate independently from each other.

Figure 4-18 shows an example of the operation waveform when OCU ch.0 is in Up/Down-count mode (Active-Low). This figure is based on the conditions that the buffer function of the OCCP register is enabled and the Zero value transfer is selected.

Figure 4-18 Example of Operation Waveform in OCU Up/Down-count Mode (Active-Low)



<Notes>

Notes on Up/Down-count mode (Active-Low) are as follows:

- In this mode, enable the buffer function of OCCP and select Zero value transfer mode for use.
- If "0x0000" is set to OCCP register when OCU's operation is enabled, the output level is changed to Active level immediately, regardless of FRT's count value.
- If a value no less than the Peak value of FRT's counter is set to OCCP, the output level does not change, even when FRT's counter value reaches its peak value. Also, the IOP0 and IOP1 registers are not set.

4.5. Details of WFG Output Waveform

This section provides details of the output waveform in each mode of WFG.

■ List of WFG Operation Modes

Table 4-9 shows a list of WFG operation modes, register settings and CH_GATE signal outputs.

Table 4-10 shows a list of WFG operation modes, register settings, RTO(1) signal and RTO(0) signal outputs.

Table 4-9 List of Details of CH_GATE Signal Outputs

Operation Mode	WFSA: TMD[2:0]	WFSA: GTEN[1:0]	CH_GATE Signal Output
Through mode	000	don't care	Always outputs Low-level signals
RT-PPG mode	001	00	Always outputs Low-level signals
		01	Outputs RT(0) signal without change
		10	Outputs RT(1) signal without change
		11	Outputs High-level signals when either RT(1) signal or RT(0) signal is High-level Outputs Low-level signals when both RT(1) and RT(0) signals are Low-level
Timer-PPG mode	010	00	Always outputs Low-level signals
		01	Outputs WFG timer active flag0
		10	Outputs WFG timer active flag1
		11	Outputs High-level signals when either of WFG timer active flags is "1" Outputs Low-level signals when both of WFG timer active flags are "0"
RT-dead timer mode	100	don't care	Always outputs Low-level signals
PPG-dead timer mode	111	00	Always outputs Low-level signals
		01	Outputs RT(0) signal without change
		10	Outputs RT(1) signal without change
		11	Outputs High-level signals when either RT(1) signal or RT(0) signal is High-level Outputs Low-level signals when both RT(1) and RT(0) signals are Low-level

* The CH_GATE signals in the table refer to CH10_GATE, CH32_GATE and CH54_GATE before being selected by WFSA:PSEL[1:0], as shown in the diagram of WFG-PPG connection.

Table 4-10 List of Output Details of RTO Pin

Operation Mode	WFSA: TMD[2:0]	WFSA: PGEN[1:0]	WFSA: DMOD	Output of RTO(1) Signal	Output of RTO(0) Signal
Through mode	000	00	don't care	Outputs RT(1) signal through	Outputs RT(0) signal through
		01		Outputs RT(1) signal through	Outputs CH_PPG signal through
		10		Outputs CH_PPG signal through	Outputs RT(0) signal through
		11		Outputs CH_PPG signal through	Outputs CH_PPG signal through
RT-PPG mode	001	00	don't care	Outputs RT(1) signal through	Outputs RT(0) signal through
		01		Outputs RT(1) signal through	(*A) Outputs Low-level signals when RT(0) is Low-level Outputs CH_PPG signal when RT(0) signal is High-level
		10		(*B) Outputs Low-level signals when RT(1) signal is Low-level Outputs CH_PPG signal when RT(1) signal is High-level	Outputs RT(0) signal through
		11		Same as *B	Same as *A
Timer-PPG mode	010	00	don't care	(*D) Outputs Low-level signals when WFG timer active flag1 is "0" Outputs High-level signals when WFG timer active flag1 is "1"	(*C) Outputs Low-level signals when WFG timer active flag0 is "0" Outputs High-level signals when WFG timer active flag0 is "1"
		01		Same as *D	(*E) Outputs Low-level signals when WFG timer active flag0 is "0" Outputs CH_PPG signal when WFG timer active flag0 is "1"
		10		(*F) Outputs Low-level signals when WFG timer active flag1 is "0" Outputs CH_PPG signal when WFG timer active flag1 is "1"	Same as *C
		11		Same as *F	Same as *E
RT RT-dead timer mode	100	don't care	setting enabled	Starts WFG timer at the rising and falling edges of the RT(1) signal and generates the non-overlap signal. The generated non-overlap signal polarity is configurable by WFSA.DMOD[1:0] bit.	
PPG-dead timer mode	111	don't care	setting enabled	Starts WFG timer at the rising and falling edges of the CH_PPG signal and generates the non-overlap signal. The generated non-overlap signal polarity is configurable by WFSA.DMOD[1:0] bit.	

* The CH_PPG signals in the table refer to CH10_PPG, CH32_PPG and CH54_PPG selected by WFSA:PSEL[1:0], as shown in the diagram of WFG-PPG connection.

* The WFSA.DMOD[1:0] setting can be used to change the output polarity of the RTO(0) and RTO(1) signals as shown below.

In the case of WFSA.DMOD[1:0] =00, RTO(0) and RTO(1) signals are output with normal polarity.

In the case of WFSA.DMOD[1:0] =01, RTO(0) and RTO(1) signals are output with reversed polarity.

In the case of WFSA.DMOD[1:0] =10, RTO(0) signal is output with reversed polarity, and RTO(1) signal is output with normal polarity.

In the case of WFSA.DMOD[1:0] =11, RTO(1) signal is output with reversed polarity, and RTO(0) signal is output with normal polarity.

■ Through Mode

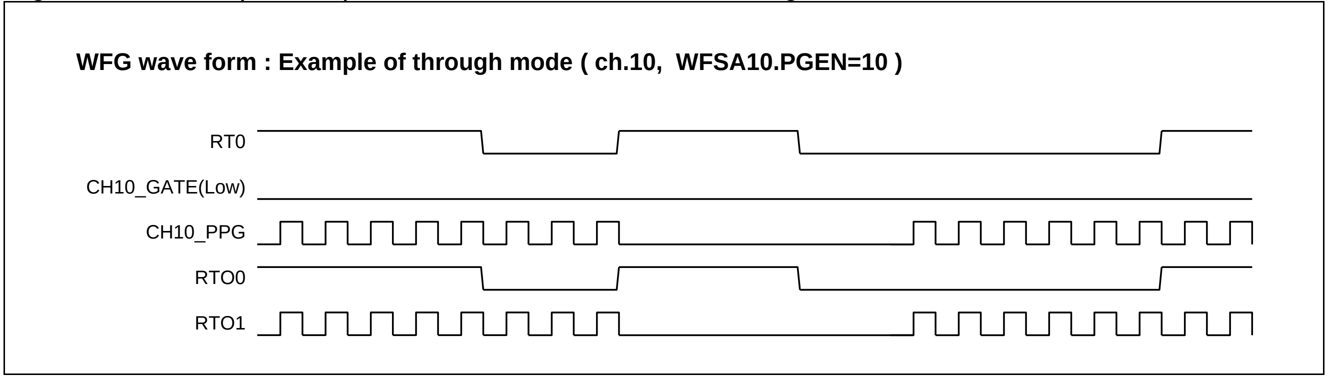
The operation in Through mode is as follows (see the List of Output Details).

The output of the CH_GATE signal is always fixed to the Low level.

The RTO(1) and RTO(0) signals output the RT(1), RT(0), and CH_PPG signals through without change by PGEN[1:0] setting.

Figure 4-19 shows an example of the operation waveform in Through mode of WFG ch.10. In this example, the RT0 signal and the CH10_PPG signal are output through to RTO0 and RTO1, respectively (PPG timer unit can start outputting without the use of the GATE signal).

Figure 4-19 Example of Operation Waveform in WFG-Through Mode


WFG wave form : Example of through mode (ch.10, WFSA10.PGEN=10)
RT0
CH10_GATE(Low)
CH10_PPG
RTO0
RTO1

■ RT-PPG Mode

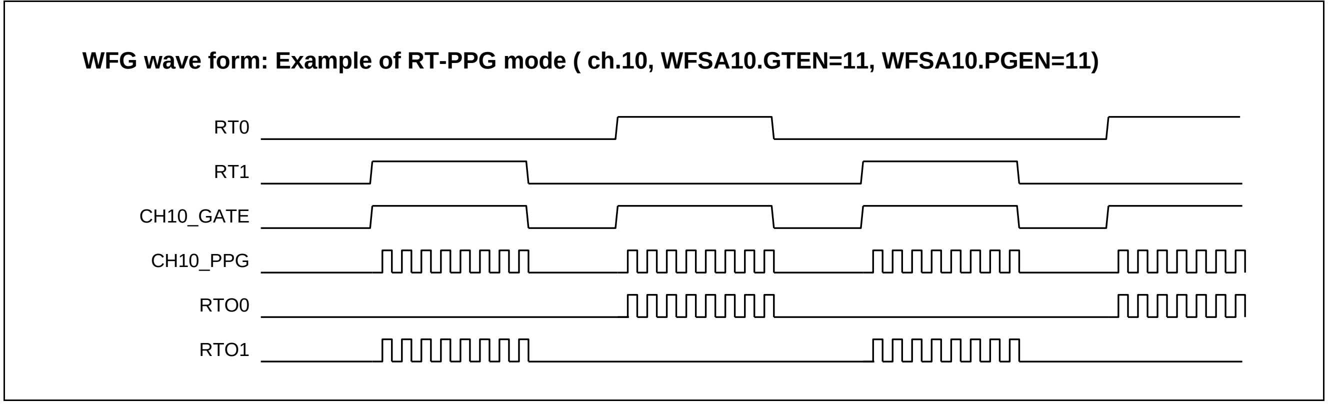
The operation in RT-PPG mode is as follows (see the List of Output Details).

The CH_GATE signal outputs the RT(1) signal, RT(0) signal or the logic OR signal of each signal by GTEN[1:0] setting.

The RTO(1) and RTO(0) signals output the RT(1) signal, RT(0) signal, CH_PPG signal, or the logic AND signal of each signal by PGEN[1:0] setting.

Figure 4-20 shows an example of the operation waveform in RT-PPG mode of WFG ch.10. In this example, the CH0_GATE signal is generated from both RT1 and RT0 to start PPG-ch.0. The CH0_PPG signal is superimposed on RTO0 and RTO1 to output.

Figure 4-20 Example of Operation Waveform in WFG-RT-PPG Mode


WFG wave form: Example of RT-PPG mode (ch.10, WFSA10.GTEN=11, WFSA10.PGEN=11)
RT0
RT1
CH10_GATE
CH10_PPG
RTO0
RTO1

■ Timer PPG Mode

The operation in Timer-PPG mode is as follows (see the List of Output Details).

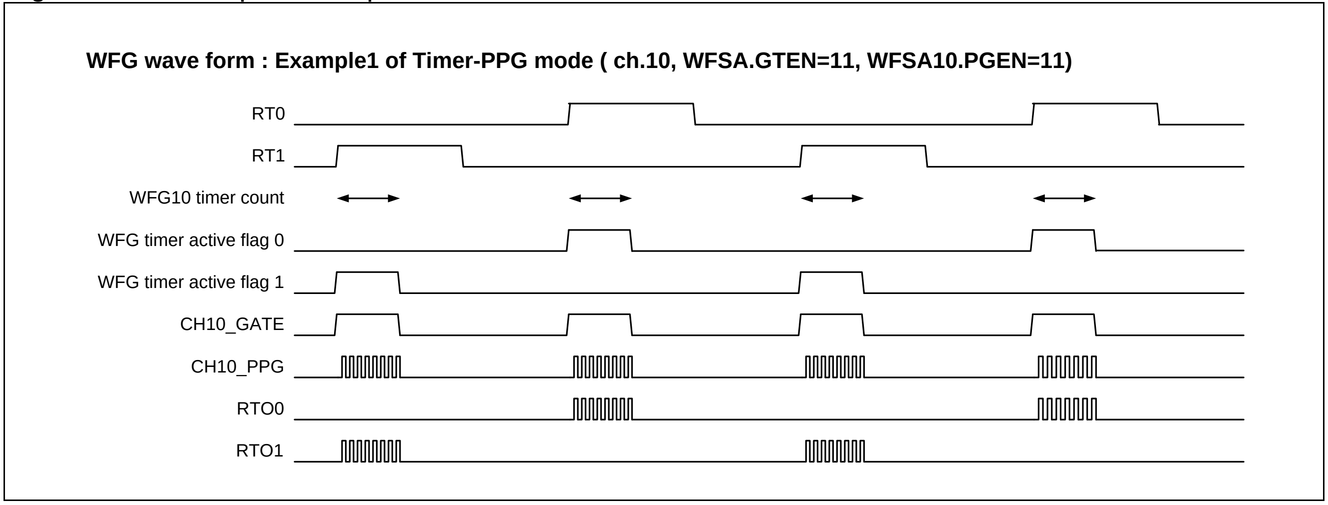
Each channel of WFG has two flags: WFG timer active flag0 and WFG timer active flag1. This mode outputs a waveform using these flags.

The CH_GATE signal outputs WFG timer active flag1, WFG timer active flag0, or the logic OR signal of each signal by GTEN[1:0] setting.

The RTO(1) and RTO(0) signals output these active flags, CH_PPG signal, or the logic AND signal of each signal by PGEN[1:0] setting.

Figure 4-21 shows Example 1 of the operation waveform in Timer PPG mode of WFG ch.10.

Figure 4-21 Example 1 of Operation Waveform in WFG-Timer PPG Mode



The WFG timer active flags operate as follows:

WFG timer active flag0 is set to "1", when the rising edge of the RT(0) signal is detected.

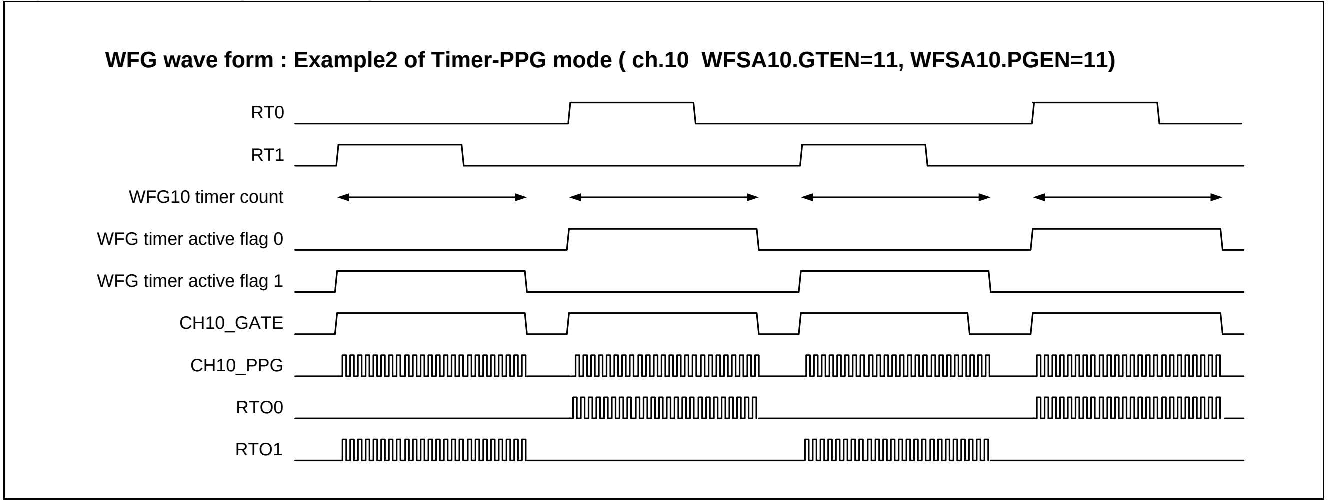
WFG timer active flag1 is set to "1", when the rising edge of the RT(1) signal is detected.

When either of the WFG timer active flags is set, the WFG timer loads the initial value from the WFTM register and starts Down-count operation. After counting, it resets both of the WFG timer active flags to "0". Therefore, irrespective of the pulse width of the RT(0) and RT(1) signals, the WFG timer active flags are set for the cycle setting time of the WFG timer from the rising edge of each signal. During this period, the CH_PPG output can be superimposed on RTO.

By the time this mode is selected by writing to the WFSA register, each WFG timer active flag is already reset. When the mode is selected, the output of the RTO(0) and RTO(1) signals is set to the Low level, regardless of the output level of the RT(0) signal, RT(1) signal and CH_PPG signal.

Figure 4-22 shows Example 2 of the operation waveform in Timer PPG mode of WFG ch.10.

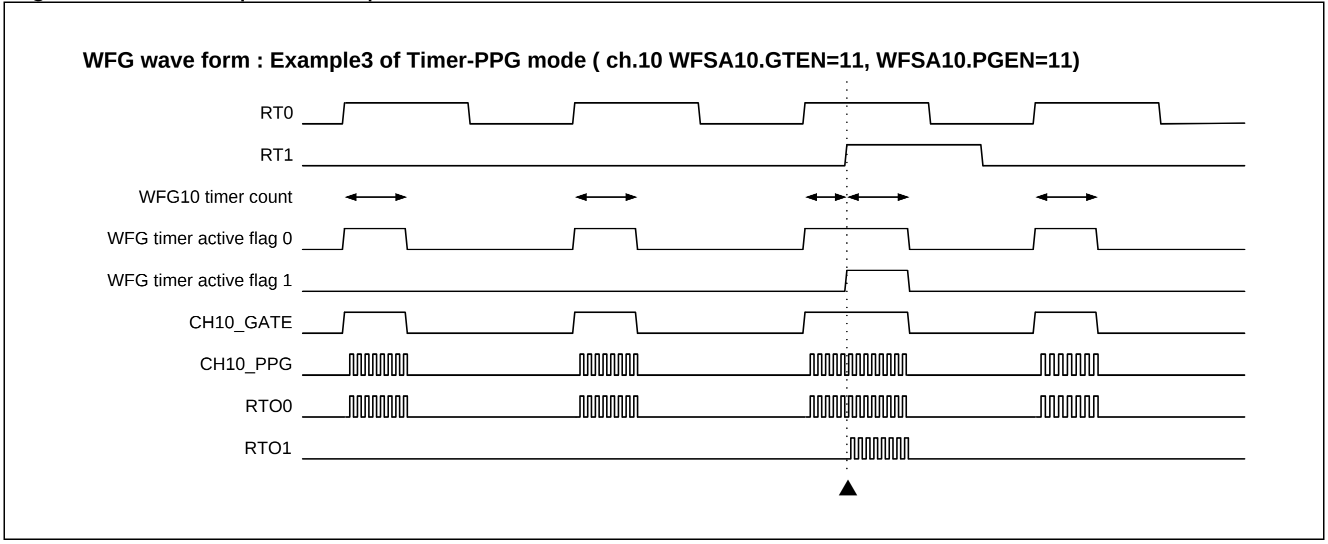
Figure 4-22 Example 2 of Operation Waveform in WFG-Timer PPG Mode



This figure shows an example of making the time setting of the WFG timer (WFTM) longer than the pulse length of RT0 and RT1. It indicates that although the same signals as in Figure 4-21 are input for the RT0 and RT1 signals, the outputs that are different from the ones shown in Figure 4-21 can be achieved because of the timer setting time.

Figure 4-23 shows Example 3 of the operation waveform in Timer PPG mode of WFG ch.10.

Figure 4-23 Example 3 of Operation Waveform in WFG-Timer PPG Mode



This figure shows an exceptional case. The following operation is performed at the point indicated by ▲ in the figure. WFG timer active flag0 is set at the rising edge of the RT0 signal and WFG10 timer is in operation. In the meantime, the rising edge of the RT1 signal is detected and WFG timer active flag1 is set.

In this case, WFG10 timer reloads the initial value and performs the operation that will restart the timer count. Each WFG timer active flag is reset, when the counting by WFG10 timer is completed. For this reason, the period in which WFG timer active flag0 is set becomes longer than the timer setting, as shown in the figure. Therefore, the output of the waveform shown in the figure can be achieved for RTO0 and RTO1.

■ RT-dead Timer Mode

The operation in RT-dead timer mode is as follows (see Tables 4-9 and 4-10).

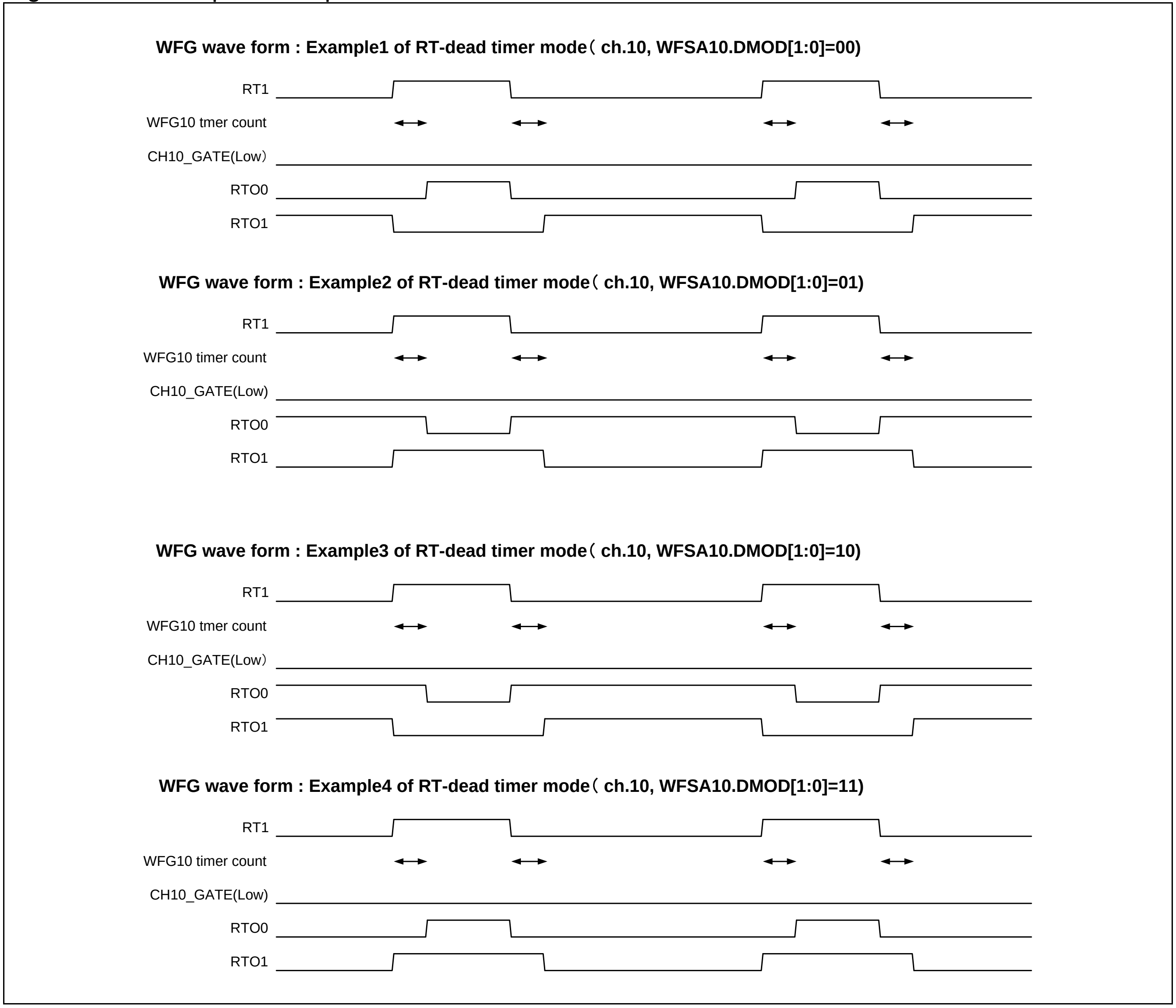
The output of the CH_GATE signal is always fixed to the Low level.

As for the RTO(1) and RTO(0) signals, the non-overlap signal that has the dead time set by WFG timer based on the RT(1) signal is output.

In this mode, the RT(0) signal and the CH_PPG signal are not used. This mode assumes that the output polarity of OCU's RT(1) output is Active High. The output polarity of RTO(0) and RTO(1) can be selected by WFSA.DMOD.

Figure 4-24 shows Example 1 of the operation waveform in RT-dead timer mode of WFG ch.10.

Figure 4-24 Example 1 of Operation Waveform in WFG-RT-dead Timer Mode


WFG wave form : Example1 of RT-dead timer mode（ch.10, WFSA10.DMOD[1:0]=00)
RT1
WFG10 tmer count
CH10_GATE(Low）
RTO0
RTO1
WFG wave form : Example2 of RT-dead timer mode（ch.10, WFSA10.DMOD[1:0]=01)
RT1
WFG10 timer count
CH10_GATE(Low)
RTO0
RTO1
WFG wave form : Example3 of RT-dead timer mode（ch.10, WFSA10.DMOD[1:0]=10)
RT1
WFG10 tmer count
CH10_GATE(Low）
RTO0
RTO1
WFG wave form : Example4 of RT-dead timer mode（ch.10, WFSA10.DMOD[1:0]=11)
RT1
WFG10 timer count
CH10_GATE(Low)
RTO0
RTO1

When normal polarity (Active High) of RTO(0) signal and RTO(1) signal is selected by WFSA.DMOD=00, operation is as follows.

When the rising edge of the RT(1) signal is detected, the output of the RTO(1) signal is set to the Low level and WFG timer starts. Then, when the delay time by the WFG timer setting elapses, the RTO(0) signal is set to the High level.

When the falling edge of the RT(1) signal is detected, the output of the RTO(0) signal is set to the Low level and

WFG timer starts. Then, when the delay time by the WFG timer setting elapses, the RTO(1) signal is set to the High level.
When this mode is selected by writing to the WFSA register, the RTO(0) signal is set to the same output level as for the RT(1) signal, while the RTO(1) signal is set to the output level that is opposite from that of the RT(1) signal.

<Notes>

When an external circuit is connected like that shown in Figure 4-25, and if RT dead timer mode (WFSA.TMD=100), DMOD=10, or other incorrect settings are made, a short-circuit will occur between the power supply and GND.

Figure 4-25 External Circuit Example

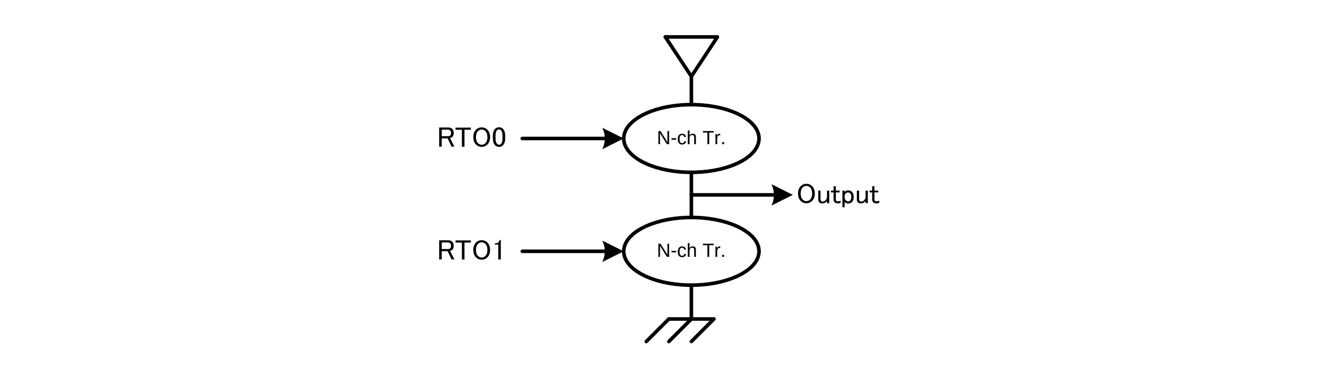
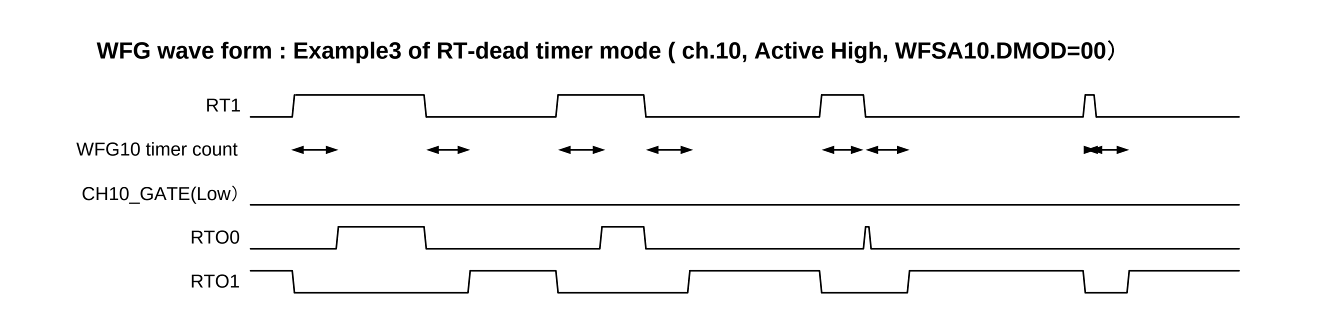


Figure 4-26 shows Example 3 of the operation waveform in RT-dead timer mode of WFG ch.10.

Figure 4-26 Example 3 of Operation Waveform in WFG-RT-dead Timer Mode


WFG wave form : Example3 of RT-dead timer mode (ch.10, Active High, WFSA10.DMOD=00）
RT1
WFG10 timer count
CH10_GATE(Low）
RTO0
RTO1

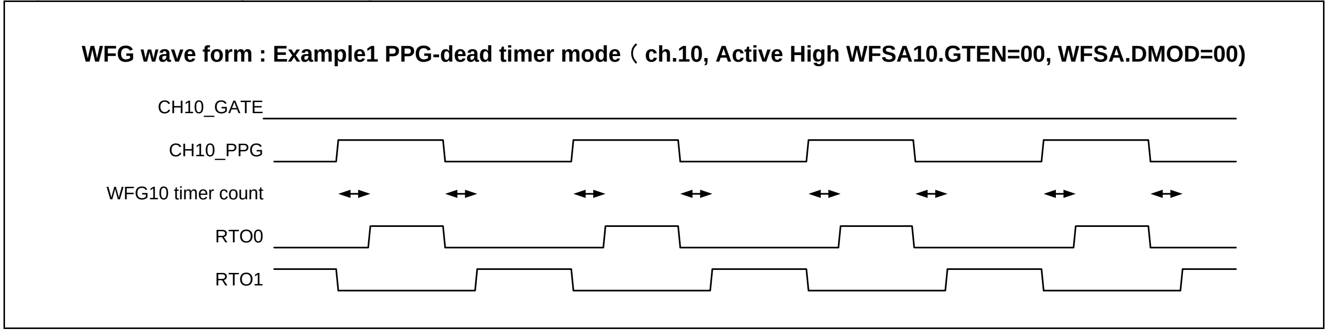
This figure shows a case of DMOD=00. A pulse shorter than the dead time set by the WFTM register is input to the last RT1 signal in the figure. In this case, WFG10 timer starts counting at the rising edge of the RT1 signal and loads the initial value at the next falling edge to restart the operation. Therefore, no pulse will be output to RTO0. After the falling edge of RT1, RTO1 is set to the High level when the timer setting time has elapsed.

In the case of DMOD=01, 10, and 11, no pulse is output to RTO(0), just like the above.

■ PPG Dead Timer Mode

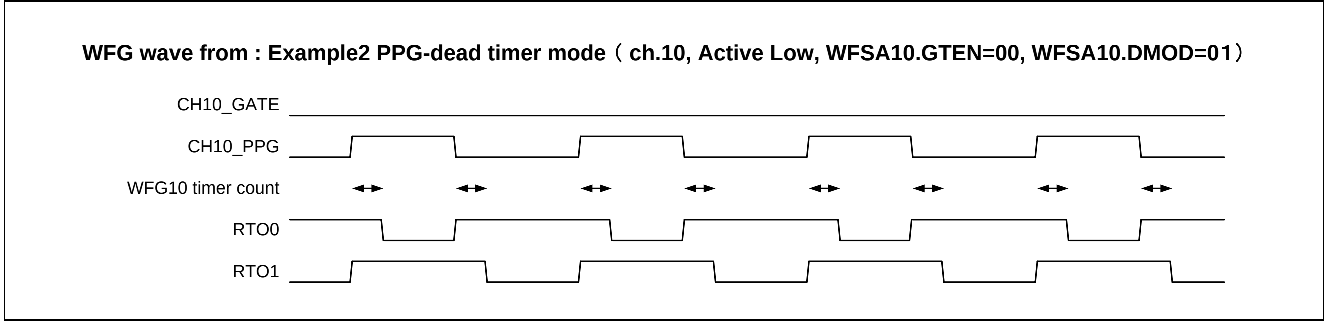
The operation in PPG-dead timer mode is as follows (see Tables 4-9 and 4-10).
The CH_GATE signal outputs the RT(1) signal, RT(0) signal or logic OR signal by GTEN[1:0] setting.
As for the RTO(1) and RTO(0) signals, the non-overlap signal that has the dead time set by WFG timer based on the CH_PPG signal is output.
In this mode, the RT(0) and RT(1) signals are used only for the output of the CH_GATE signal.
Figure 4-27 shows Example 1 of the operation waveform in PPG-dead timer mode of WFG ch.10.

Figure 4-27 Example 1 of Operation Waveform in WFG-PPG Dead Timer Mode


WFG wave form : Example1 PPG-dead timer mode（ch.10, Active High WFSA10.GTEN=00, WFSA.DMOD=00)
CH10_GATE
CH10_PPG
WFG10 timer count
RTO0
RTO1

This figure shows an example of the case where WFSA.DMOD=00 and normal polarity (Active High) are selected.
As shown in the figure, the WFG timer starts at the rising and falling edges of the CH_PPG signal and the non-overlap signal with the delay time set by the WFTM applied to the RTO(0) and RTO(1) signals is output.
When this mode is selected by writing to the WFSA register, the RTO(0) signal is set to the same output level as for the CH_PPG signal and the RTO(1) signal is output at the output level that is opposite from that of the CH_PPG signal.
Figure 4-28 shows Example 2 of the operation waveform in PPG-dead timer mode of WFG ch.10.

Figure 4-28 Example 2 of Operation Waveform in WFG-PPG-dead Timer Mode


WFG wave from : Example2 PPG-dead timer mode（ch.10, Active Low, WFSA10.GTEN=00, WFSA10.DMOD=01）
CH10_GATE
CH10_PPG
WFG10 timer count
RTO0
RTO1

This figure shows an example of the case where WFSA.DMOD=01 and reversed polarity (Activator) are selected by WFSA.DMOD=1. As shown in the figure, the non-overlap signal with the output level reversed from that of the RTO(0) and RTO(1) signals is output.
When this mode is selected by writing to the WFSA register, the RTO(0) signal is set to the output level that is opposite from that of the CH_PPG signal and the RTO(1) signal is output at the same output level as for the CH_PPG signal.
If the pulse width of the CH_PPG signal is shorter than the WFG timer, no pulse will be output to RTO(0), just like the case in Figure 4-26.

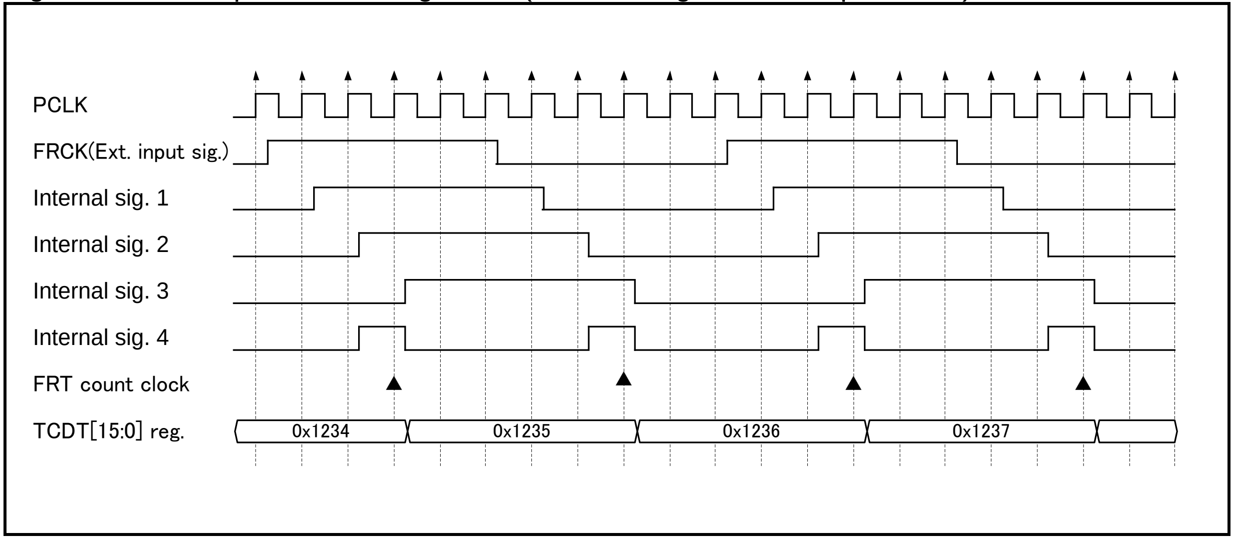
5. Detailed Timing of Multifunction Timer Input/Output Signals

The detailed timing charts for the multifunction timer input/output signals are shown below.

5.1. FRT Operation Detailed Timing When Using External Input Clock

The timing chart for the FRT count operation when using an external input clock (FRCK) is shown in Figure 5-1.

Figure 5-1 FRT Operation Timing Chart (When Using External Input Clock)



5.2. OCU and WFG Operation Detailed Timing

The OCU and WFG operation timing charts are shown in Figure 5-2 and Figure 5-3. The IOP register, RTx output signal, and WFG RTOx output signal changes when a match is detected in OCU are shown.
Figure 5-2 is an example when through mode (WFSA.TMD=000) is selected by the WFG. The FRT performs count operation using the PCLK frequency-dividing clock. The figure is an example of four frequency divisions (TCSA.CLK[3:0]=0010).
The WFG RTOx signal is changed from the OCU RTx signal after one cycle by PCLK. The RTOx signal is an external output terminal for the microcontroller. Delays are generated based on the load capacity of the external output terminal.

Figure 5-2 OCU-WFG Operation Timing Chart (WFG Through Mode)

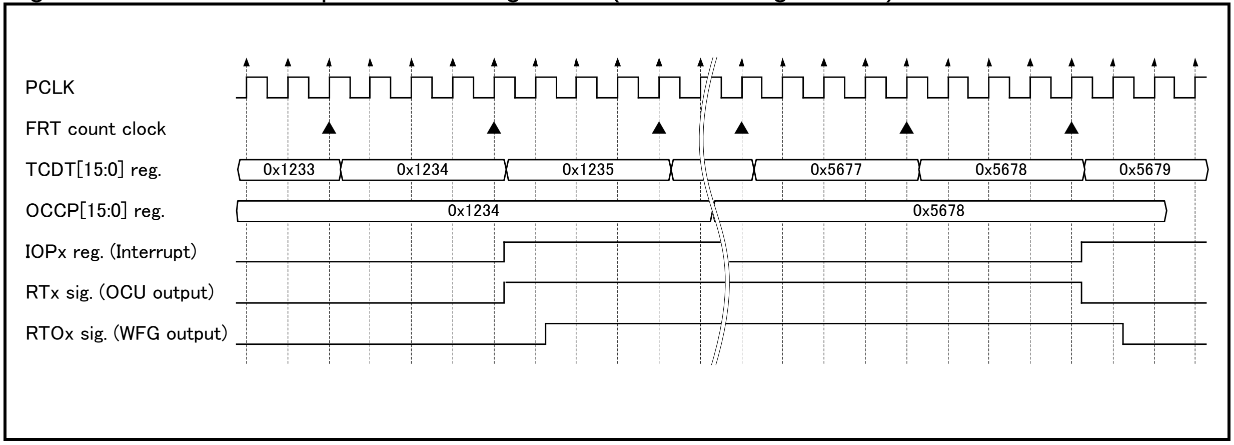
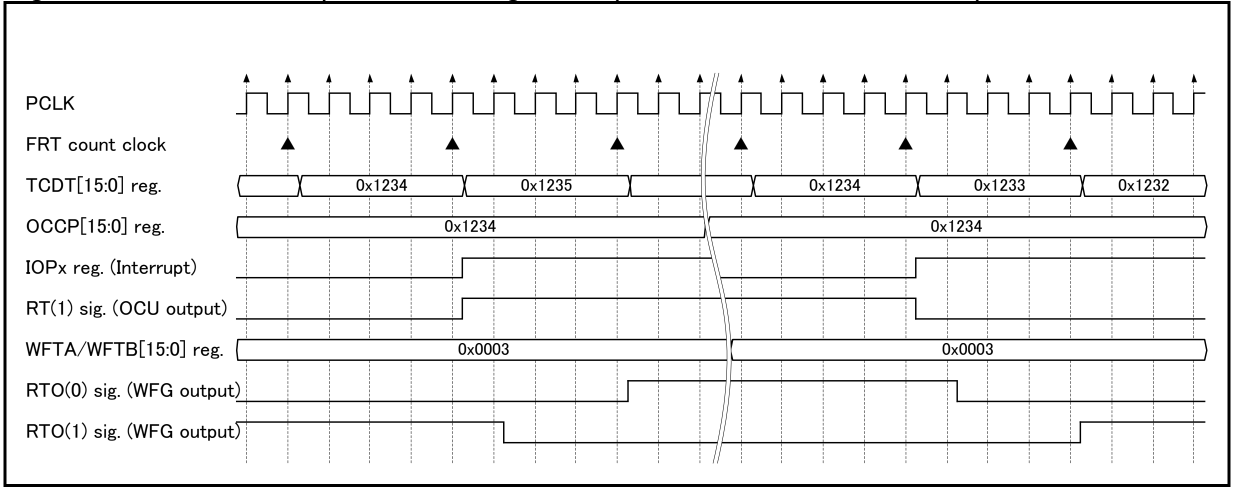


Figure 5-3 shows an example when RT dead timer mode (WFSA.TMD=100) is selected by the WFG. This shown an example where WFTA/WFTB=0x0003 and WFSA.DCK=000 are specified, and a dead time of 3*PCLK is inserted. If the dead time is not inserted, the RTO(0) and RTO(1) of the WFG are changed from the OCU RT(1) signal after one cycle by PCLK. If a dead time is inserted, after one cycle by PCLK, the output is changed after the specified dead time.

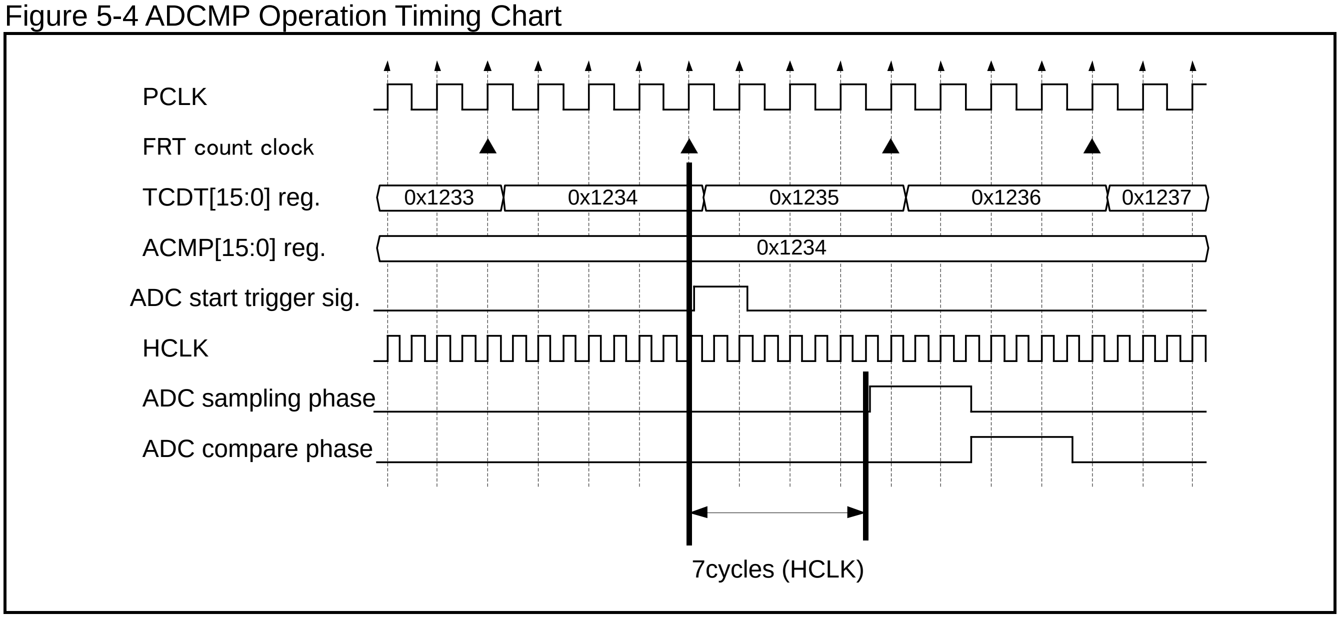
Figure 5-3 OCU-WFG Operation Timing Chart (WFG RT Dead Timer Mode)



5.3. ADCMP Operation Detailed Timing

The ADCMP operation timing chart is shown in Figure 5-4. The operation timing from FRT counter match detection to the beginning of ADC startup in ADCMP is shown. Time for a total of seven cycles in HCLK is required from FRT count clock match detection to ADC conversion start. The ADC sampling time and compare time are specified by making settings at the ADC side.

Figure 5-4 ADCMP Operation Timing Chart



5.4. ICU Operation Detailed Timing

The ICU operation timing charts are shown in Figure 5-5 and Figure 5-6.

These charts contain the operation where the FRT count value is captured by the ICCP register and the operation where the ICP register (interrupt flag) is set from the signal change of the external input terminal (ICx). The capture timing is determined by the change timing of the input signal only and is not affected by the FRT count clock.

Figure 5-5 ICU Operation Timing Chart (Input Signal Rising)

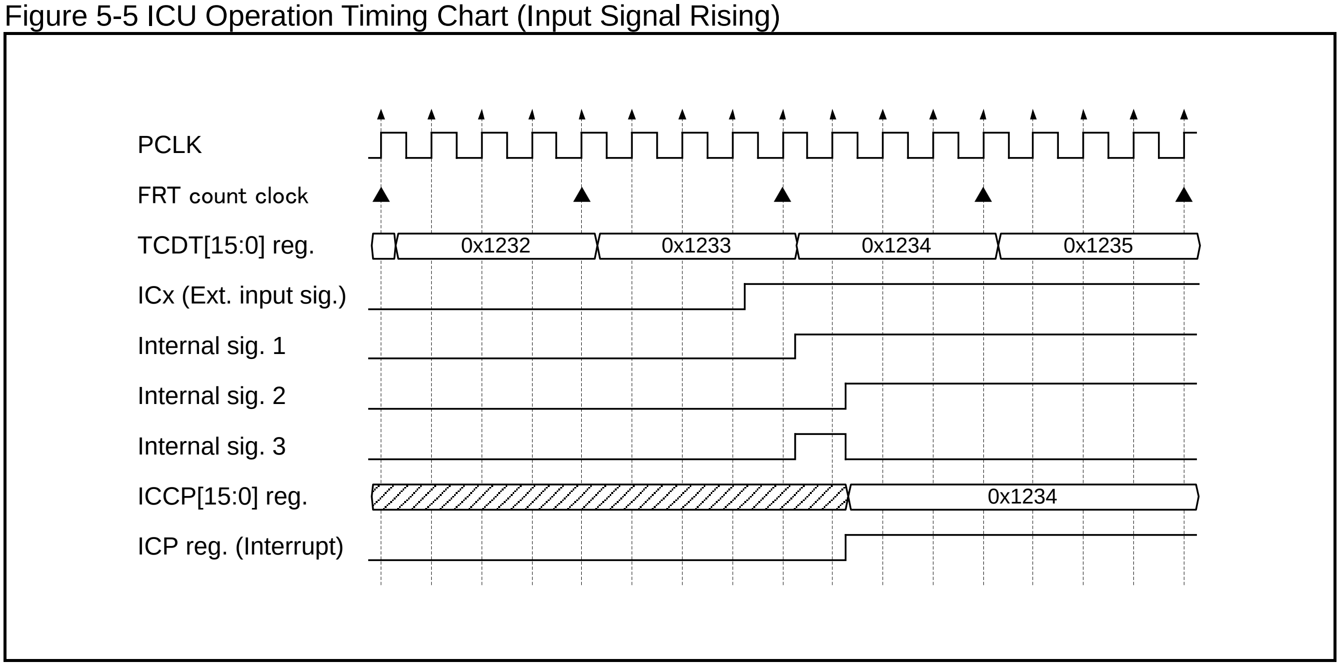
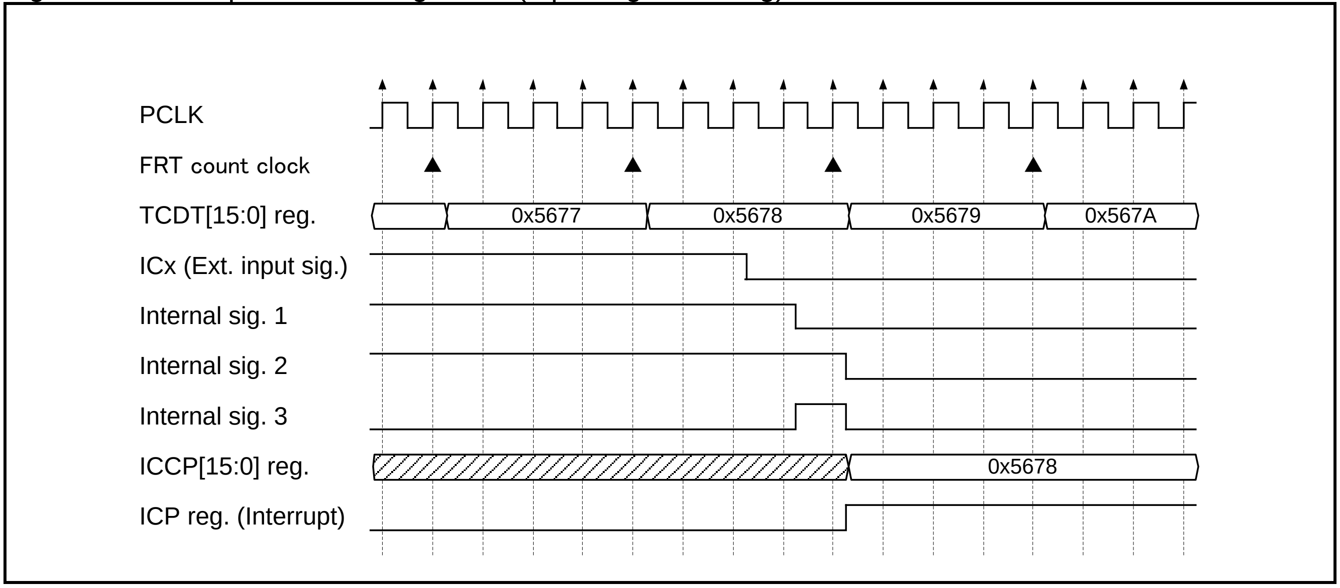


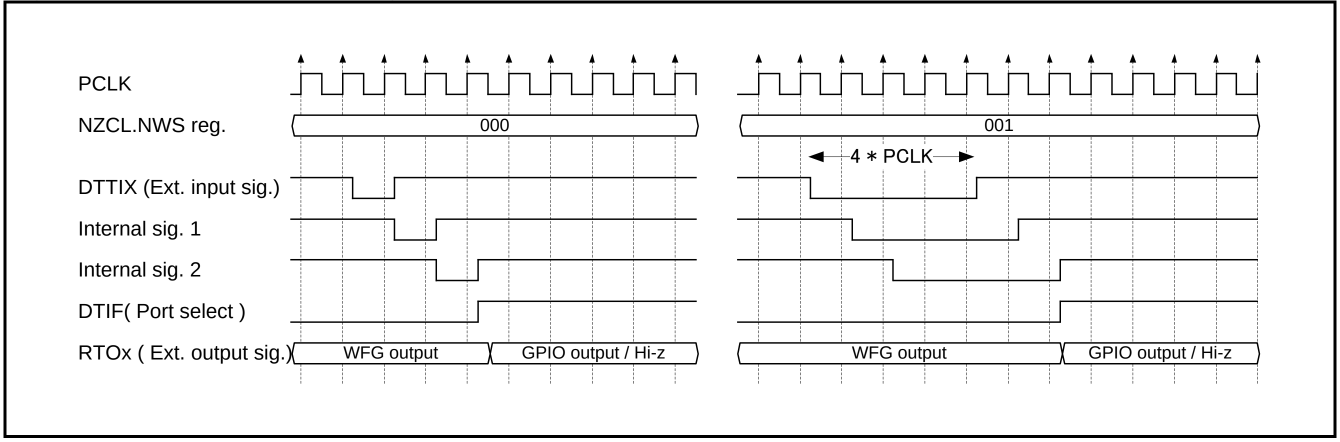
Figure 5-6 ICU Operation Timing Chart (Input Signal Falling)



5.5. DTTIX Input Detailed Timing

The timing chart from the DTTIX signal until the I/O port output is changed by passing through the digital noise filter is shown in Figure 5-7.

Figure 5-7 DTTIX-DTIF Timing Chart



6. Usage Precautions

The section explains the precautions when using multifunction timer.

6.1. Connection of Model Containing Multiple MFT's

This section explains the connection of models that contain multiple MFT's.

For models containing more than one multifunction timer unit, the connection of the I/O signals of the multifunction timer varies depending on the unit.
This section explains such connection differences for each multifunction timer unit.

6.1.1. Selection of FRT Connected to OCU and ICU

OCU and ICU are configured to be able to select FRT for other multifunction timer units. This section explains FRT connection between multifunction timer units and the selection method.

■ Model Containing Two MFT's

Figure 6-1 shows a diagram of FRT connected between multifunction timer units for a model containing 2 multifunction timer units.

Figure 6-1 Diagram of FRT Connected between Multifunction Timer Units
(For Model Containing 2 Multifunction Timer Units)

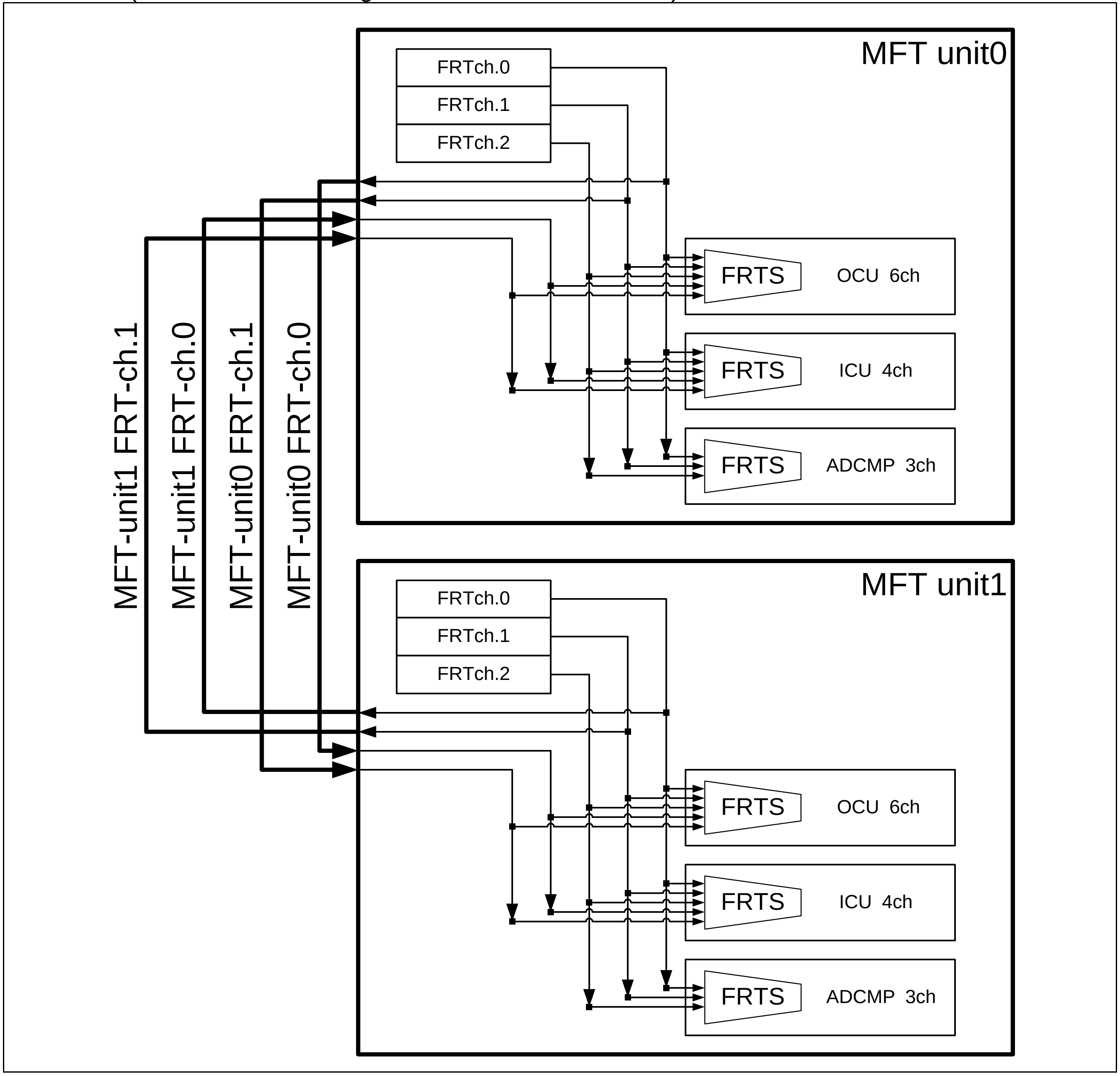


Table 6-1 shows the register settings of OCFS and ICFS of MFT unit0 and where they are connected.
Table 6-2 shows the register settings of OCFS and ICFS of MFT unit1 and where they are connected.

Table 6-1 OCFS and ICFS Register Settings for MFT unit0
(For Model Containing 2 Multifunction Timer Units)

Register Name		Setting	Function
OCFS	FSO0[3:0]	0011	Connects FRT ch.0 of MFT unti1 to OCU ch.(0).
		0100	Connects FRT ch.1 of MFT unti1 to OCU ch.(0).
	FSO1[3:0]	0011	Connects FRT ch.0 of MFT unti1 to OCU ch.(1).
		0100	Connects FRT ch.1 of MFT unti1 to OCU ch.(1).
ICFS	FSI0[3:0]	0011	Connects FRT ch.0 of MFT unti1 to ICU ch.(0).
		0100	Connects FRT ch.1 of MFT unti1 to ICU ch.(0).
	FSI1[3:0]	0011	Connects FRT ch.0 of MFT unti1 to ICU ch.(1).
		0100	Connects FRT ch.1 of MFT unti1 to ICU ch.(1).

Table 6-2 OCFS and ICFS Register Settings for MFT unit1
(For Model Containing 2 Multifunction Timer Units)

Register Name		Setting	Function
OCFS	FSO0[3:0]	0011	Connects FRT ch.0 of MFT unti0 to OCU ch.(0).
		0100	Connects FRT ch.1 of MFT unti0 to OCU ch.(0).
	FSO1[3:0]	0011	Connects FRT ch.0 of MFT unti0 to OCU ch.(1).
		0100	Connects FRT ch.1 of MFT unti0 to OCU ch.(1).
ICFS	FSI0[3:0]	0011	Connects FRT ch.0 of MFT unti0 to ICU ch.(0).
		0100	Connects FRT ch.1 of MFT unti0 to ICU ch.(0).
	FSI1[3:0]	0011	Connects FRT ch.0 of MFT unti0 to ICU ch.(1).
		0100	Connects FRT ch.1 of MFT unti0 to ICU ch.(1).

■ Model Containing Three MFT's

Figure 6-2 shows a diagram of FRT connected between multifunction timer units for a model containing 3 multifunction timer units.

Figure 6-2 Diagram of FRT Connected between Multifunction Timer Units
(For Model Containing 3 Multifunction Timer Units)

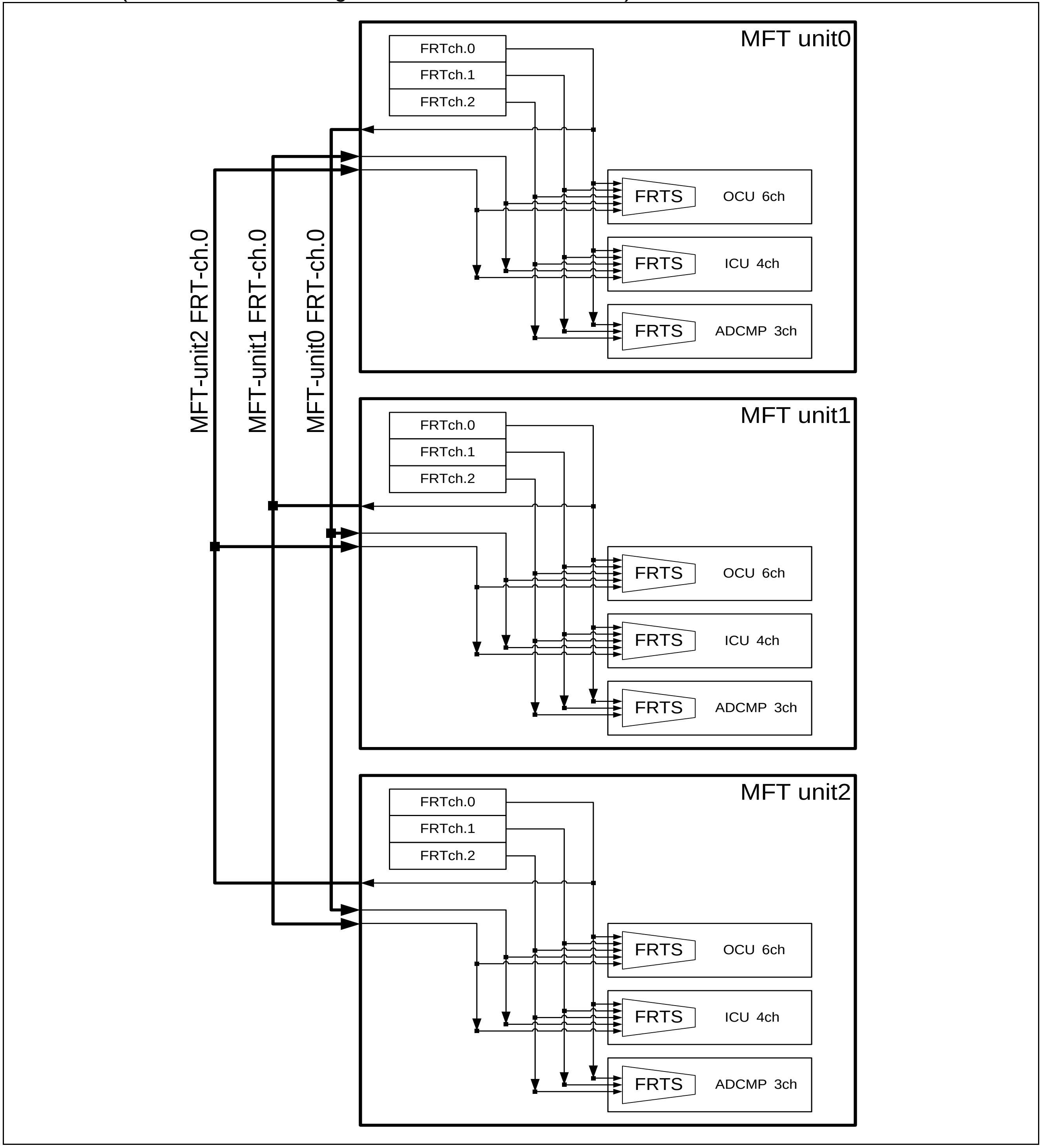


Table 6-3 shows the register settings of OCFS and ICFS of MFT-unit0 and where they are connected.
Table 6-4 shows the register settings of OCFS and ICFS of MFT-unit1 and where they are connected.
Table 6-5 shows the register settings of OCFS and ICFS of MFT-unit2 and where they are connected.

Table 6-3 OCFS and ICFS Register Settings for MFT unit0
(For Model Containing 3 Multifunction Timer Units)

Register Name		Setting	Function
OCFS	FSO0[3:0]	0011	Connects FRT ch.0 of MFT unti1 to OCU ch.(0).
		0100	Connects FRT ch.0 of MFT unti2 to OCU ch.(0).
	FSO1[3:0]	0011	Connects FRT ch.0 of MFT unti1 to OCU ch.(1).
		0100	Connects FRT ch.0 of MFT unti2 to OCU ch.(1).
ICFS	FSI0[3:0]	0011	Connects FRT ch.0 of MFT unti1 to ICU ch.(0).
		0100	Connects FRT ch.0 of MFT unti2 to ICU ch.(0).
	FSI1[3:0]	0011	Connects FRT ch.0 of MFT unti1 to ICU ch.(1).
		0100	Connects FRT ch.0 of MFT unti2 to ICU ch.(1).

Table 6-4 OCFS and ICFS Register Settings for MFT unit1
(For Model Containing 3 Multifunction Timer Units)

Register Name		Setting	Function
OCFS	FSO0[3:0]	0011	Connects FRT ch.0 of MFT unti0 to OCU ch.(0).
		0100	Connects FRT ch.0 of MFT unti2 to OCU ch.(0).
	FSO1[3:0]	0011	Connects FRT ch.0 of MFT unti0 to OCU ch.(1).
		0100	Connects FRT ch.0 of MFT unti2 to OCU ch.(1).
ICFS	FSI0[3:0]	0011	Connects FRT ch.0 of MFT unti0 to ICU ch.(0).
		0100	Connects FRT ch.0 of MFT unti2 to ICU ch.(0).
	FSI1[3:0]	0011	Connects FRT ch.0 of MFT unti0 to ICU ch.(1).
		0100	Connects FRT ch.0 of MFT unti2 to ICU ch.(1).

Table 6-5 OCFS and ICFS Register Settings for MFT unit2
(For Model Containing 3 Multifunction Timer Units)

Register Name		Setting	Function
OCFS	FSO0[3:0]	0011	Connects FRT ch.0 of MFT unti0 to OCU ch.(0).
		0100	Connects FRT ch.0 of MFT unti1 to OCU ch.(0).
	FSO1[3:0]	0011	Connects FRT ch.0 of MFT unti0 to OCU ch.(1).
		0100	Connects FRT ch.0 of MFT unti1 to OCU ch.(1).
ICFS	FSI0[3:0]	0011	Connects FRT ch.0 of MFT unti0 to ICU ch.(0).
		0100	Connects FRT ch.0 of MFT unti1 to ICU ch.(0).
	FSI1[3:0]	0011	Connects FRT ch.0 of MFT unti0 to ICU ch.(1).
		0100	Connects FRT ch.0 of MFT unti1 to ICU ch.(1).

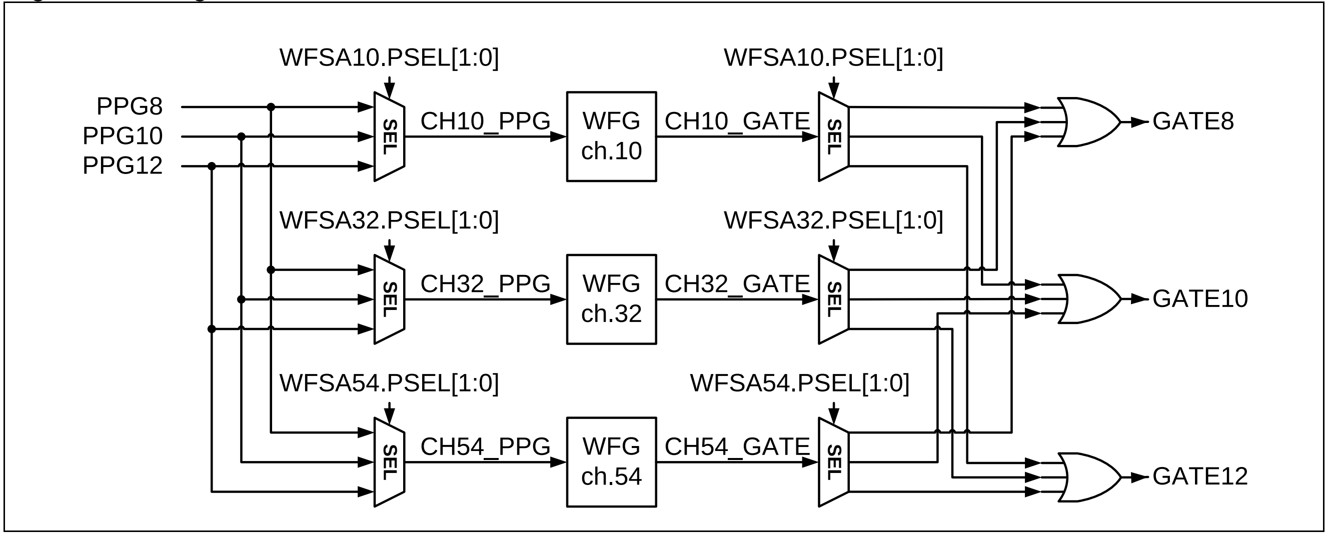
6.1.2. PPG Timer Unit Connected to WFG

The PPG timer unit to be connected to WFG varies depending on the multifunction timer unit used. This section explains the connection of the PPG timer unit and the selection method.

■ MFT unit1

PPG timer unit ch.8, ch.10 and ch.12 are connected to WFG of MFTunit1, as shown in Figure 6-3.

Figure 6-3 Diagram of WFG-PPG Connection at MFTunit1



In case of WFG in MFT unit1, the following is selected by the setting of the PSEL[1:0] bits.

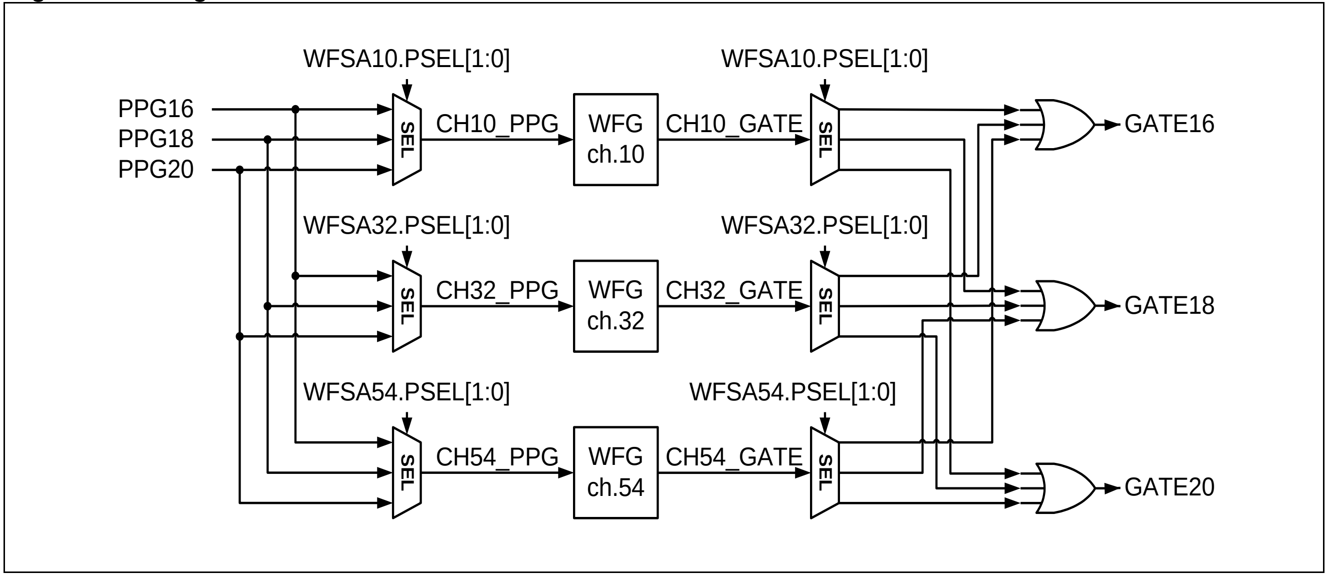
[bit9:8] PSEL[1:0]

Process	Value	Function
Write	00	Sets the output destination of the GATE signal to PPG timer unit ch.8. Sets the input source of the PPG signal to PPG timer unit ch.8.
	01	Sets the output destination of the GATE signal to PPG timer unit ch.10. Sets the input source of the PPG signal to PPG timer unit ch.10.
	10	Sets the output destination of the GATE signal to PPG timer unit ch.12. Sets the input source of the PPG signal to PPG timer unit ch.12.
	11	Setting is prohibited.
Read	-	Reads the register setting.

■ MFT-unit2

PPG timer unit ch.16, ch.18 and ch.20 are connected to WFG of MFT unit2, as shown in Figure 6-4.

Figure 6-4 Diagram of WFG-PPG Connection at MFT unit2



In case of WFG in MFT unit2, the following is selected by the setting of the PSEL[1:0] bits.

[bit9:8] PSEL[1:0]

Process	Value	Function
Write	00	Sets the output destination of the GATE signal to PPG timer unit ch.16. Sets the input source of the PPG signal to PPG timer unit ch.16.
	01	Sets the output destination of the GATE signal to PPG timer unit ch.18. Sets the input source of the PPG signal to PPG timer unit ch.18.
	10	Sets the output destination of the GATE signal to PPG timer unit ch.20. Sets the input source of the PPG signal to PPG timer unit ch.20.
	11	Setting is prohibited.
Read	-	Reads the register setting.

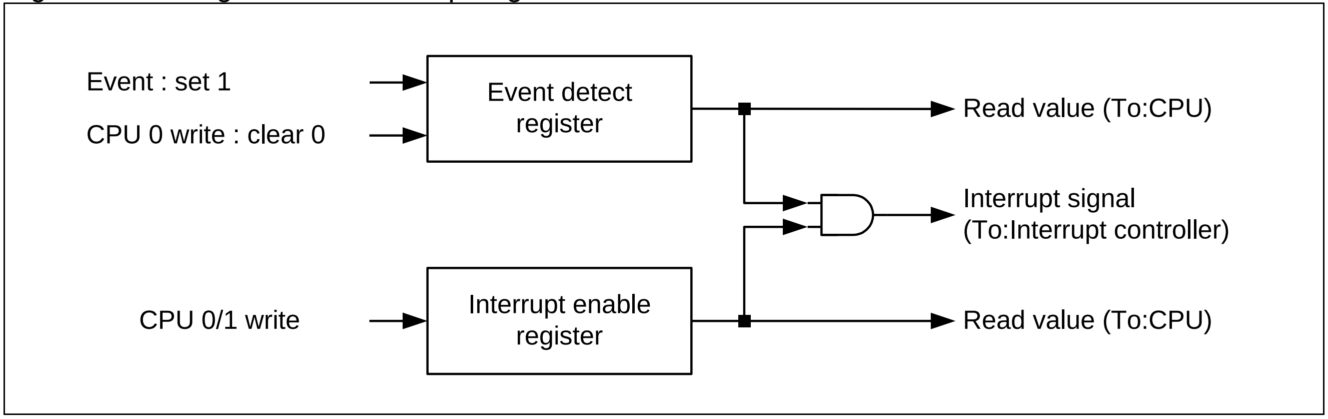
6.2. Treatment of Event Detect Register and Interrupt

This section provides notes on the event detect register in the multifunction timer unit, the operation and control of interrupt-related circuits.

■ Configuration of Circuit

Figure 6-5 shows the configuration of the interrupt signal generator.

Figure 6-5 Configuration of Interrupt Signal Generator



- Event detect register
  Each function block has an event detect register to notify CPU that a specific event (e.g. detection of the rising edge of the input signal at ICU) has occurred. This register indicates "0" when the relevant event has not occurred. It is set to "1", when the event occurs.

- Interrupt enable register
  There is an interrupt enable register to specify whether or not to notify CPU of the above event as an interrupt. As shown in the figure, the logic AND of the values in the event detect register and the interrupt enable register is connected to the interrupt controller (NVIC) as an interrupt signal.

- Writing to and reading from each register
  The event detect register can be read from CPU at any time, regardless of the value in the interrupt enable register. It can be cleared by writing "0", but cannot be set by writing "1". The interrupt enable register allows any value to be set from CPU and can be read.

■ Circuit Operation

- Operation when the interrupt enable register is set to "0" (interrupt disabled)
  Even when an event occurs and "1" is set to the event detect register, no interrupt occurs. In this case, the occurrence of the event can be recognized by reading from the event detect register regularly via CPU.

- Operation when the interrupt enable register is set to "1" (interrupt enabled)
  When an event occurs and "1" is set to the event detect register, the interrupt signal is asserted and an interrupt occurs. CPU can recognize the occurrence of the event by the interrupt.

■ Clearing Event Detect Register

Generally, the event detect register cannot be cleared automatically. In order to recognize the occurrence of the next event after "1" is set to the event detect register, the event detect register must be cleared via CPU beforehand. If it is not cleared via CPU, CPU cannot recognize the occurrence of the succeeding events.

■ Returning from Interrupt Processing

When an interrupt is processed using an interrupt signal, it is necessary to clear the event detect register when returning from the interrupt processing, deassert the interrupt signal, and then return from the interrupt. Returning from an interrupt without deasserting the interrupt signal will result in the same interrupt process taking place again with no way out of that process.

■ Value Written to Event Detect Register

The write value and read value of the event detect register have the following meanings:

- Writing "0" : Clears the register.
- Writing "1" : Does nothing.
- Reading "0" : No event occurred.
- Reading "1" : Event occurred.

Because the event detect register is in the configuration described above, when a value is read from the event detect register via CPU, the value cannot be normally written back. This is due to the following reason.

When "0" is successfully read from the event detect register at a certain point, it indicates that the event has yet to occur at that point. Next, writing the value back to the event detect register without change (i.e. writing "0") means instructing the event detect register to be cleared. If an event occurs during the period from the reading via CPU to the writing the value back, the register will be cleared, preventing that event from being recognized.

For the above reason, when writing to the event detect register, "1" must be always written (i.e. doing nothing), unless the register is intended to be cleared. An example is provided below.

The ICSA10 register is in the following configuration based on the 8-bit register.

bit	7	6	5	4	3	2	1	0
Field	ICP1	ICP0	ICE1	ICE0	EG1[1:0]		EG0[1:0]	

The ICP1 and ICP0 registers are event detect registers that notify CPU of an event upon edge detection at ICU-ch.1 and ICU-ch.0, respectively.

If "01111111" is read from these registers at a certain point, for example, it indicates that a valid edge is detected (ICP0=1) at ch.0 and no valid edge is detected (ICP1=0) at ch.1.

Then, write "0" back to bit6 in order to clear the ICP0 register. At that point, it is not possible to set the value in the ICP0 register to "0" and write "00111111" back due to the reason explained above. It is because information about any possible detection of an event at ch.1 will be cleared during the period from reading from the register to writing the value back.

Therefore, in order to clear the ICP0 register, it is necessary to write "10111111" back with bit6=0 and bit7=1.

■ Read Value Mask Function at RMW (Read Modify Write) Access

Since the above procedure is complicated, a masking function is provided to mask the read value of the event detect register to "1" at RMW access for the value to be written back.

In this model, RMW access occurs, when write access is made to the bit-banding alias area.

Write access to the bit-banding alias area is the RMW access used to read all of the register bits in the address area where the target bit exists, rewrite only the target bit and write all the register bits back.

In the example of the ICSA10 register provided earlier, assume that the value "01111111" is read at a certain point. To write "0" to bit6 so that the ICP0 register will be cleared, write access to the normal address area requires bit7=1 and bit6=0 to be written. However, if "0" is written to bit6 by write access to the bit-banding alias area, the hardware performs the following operation:

- It read the value in the ICSA10 register.
- At this point, the ICP1 and ICP0 registers return a read value masked to "1" because of the RMW access. In other words, the value to be read is "11111111".
- Write the value "10111111" to the ICSA10 register, where only the value of bit6(ICP0) has been replaced with "0".

bit7 cannot be cleared because the device operates as described above. How to write back the value of bit6 is described in this example. In case of writing back the values of bit7 and bit5 to bit0, the read values of bit7 and bit6 are masked to "1" also; therefore, it is unnecessary to consider the writing back value. For this reason, this configuration allows rewriting the register without considering the writing back value to the event detection register in case of writing access to the bit-banding alias area.

* Read access to the bit-banding alias area is not RMW access; therefore the value of the register is unmasked when reading.

■ List of Event Detect Registers and Interrupt Enable Registers

Table 6-6 shows a list of the event detect registers and interrupt enable registers that exist in the multifunction timer unit as well as their interrupt signals.

Table 6-6 List of Event Detect Registers and Interrupt Enable Registers

Block Name	Target Event	Event Detect Register	Interrupt Enable Register	Name of Interrupt Signal
FRT ch.0	Detection of FRT0 == 0x0000	TCSA0:IRQZF	TCSA0:IRQZE	Zero value detection interrupt
FRT ch.1	Detection of FRT1 == 0x0000	TCSA1:IRQZF	TCSA1:IRQZE	Zero value detection interrupt
FRT ch.2	Detection of FRT2 == 0x0000	TCSA2:IRQZF	TCSA2:IRQZE	Zero value detection interrupt
FRT ch.0	Detection of FRT0 == TCCP0	TCSA0:ICLR	TCSA0:ICRE	Peak value detection interrupt
FRT ch.1	Detection of FRT1 == TCCP1	TCSA1:ICLR	TCSA1:ICRE	Peak value detection interrupt
FRT ch.2	Detection of FRT2 == TCCP2	TCSA2:ICLR	TCSA2:ICRE	Peak value detection interrupt
OCU ch.0	Detection of FRT == OCCP0	OCSA10:IOP0	OCSA10:IOE0	Match detection interrupt
OCU ch.1	Detection of FRT == OCCP1	OCSA10:IOP1	OCSA10:IOE1	Match detection interrupt
OCU ch.2	Detection of FRT == OCCP2	OCSA32:IOP0	OCSA32:IOE0	Match detection interrupt
OCU ch.3	Detection of FRT == OCCP3	OCSA32:IOP1	OCSA32:IOE1	Match detection interrupt
OCU ch.4	Detection of FRT == OCCP4	OCSA54:IOP0	OCSA54:IOE0	Match detection interrupt
OCU ch.5	Detection of FRT == OCCP5	OCSA54:IOP1	OCSA54:IOE1	Match detection interrupt
ICU ch.0	Detection of valid edge	ICSA10:ICP0	ICSA10:ICE0	Input signal edge detection interrupt
ICU ch.1	Detection of valid edge	ICSA10:ICP1	ICSA10:ICE1	Input signal edge detection interrupt
ICU ch.2	Detection of valid edge	ICSA32:ICP0	ICSA32:ICE0	Input signal edge detection interrupt
ICU ch.3	Detection of valid edge	ICSA32:ICP1	ICSA32:ICE1	Input signal edge detection interrupt

The interrupts shown in Table 6-7 below do not have an interrupt enable register, as they are dedicated to interrupts (i.e. polling not assumed). If "1" is set to the interrupt flag when the target event occurs, an interrupt occurs.

Table 6-7 List of Interrupt Flag Registers and Interrupt Enable Registers

Block Name	Target Event	Interrupt Flag Register	Interrupt Enable Register	Name of Interrupt Signal
NZCL	Input of emergency motor shutdown signal	WFIR:DTIF	None	DTIF interrupt
WFG ch.10	Completion of WFG10 timer count	WFIR:TMIF10	None	WFG10 timer interrupt
WFG ch.32	Completion of WFG32 timer count	WFIR:TMIF32	None	WFG32 timer interrupt
WFG ch.54	Completion of WFG54 timer count	WFIR:TMIF54	None	WFG54 timer interrupt

CHAPTER 7-1: PPG Configuration

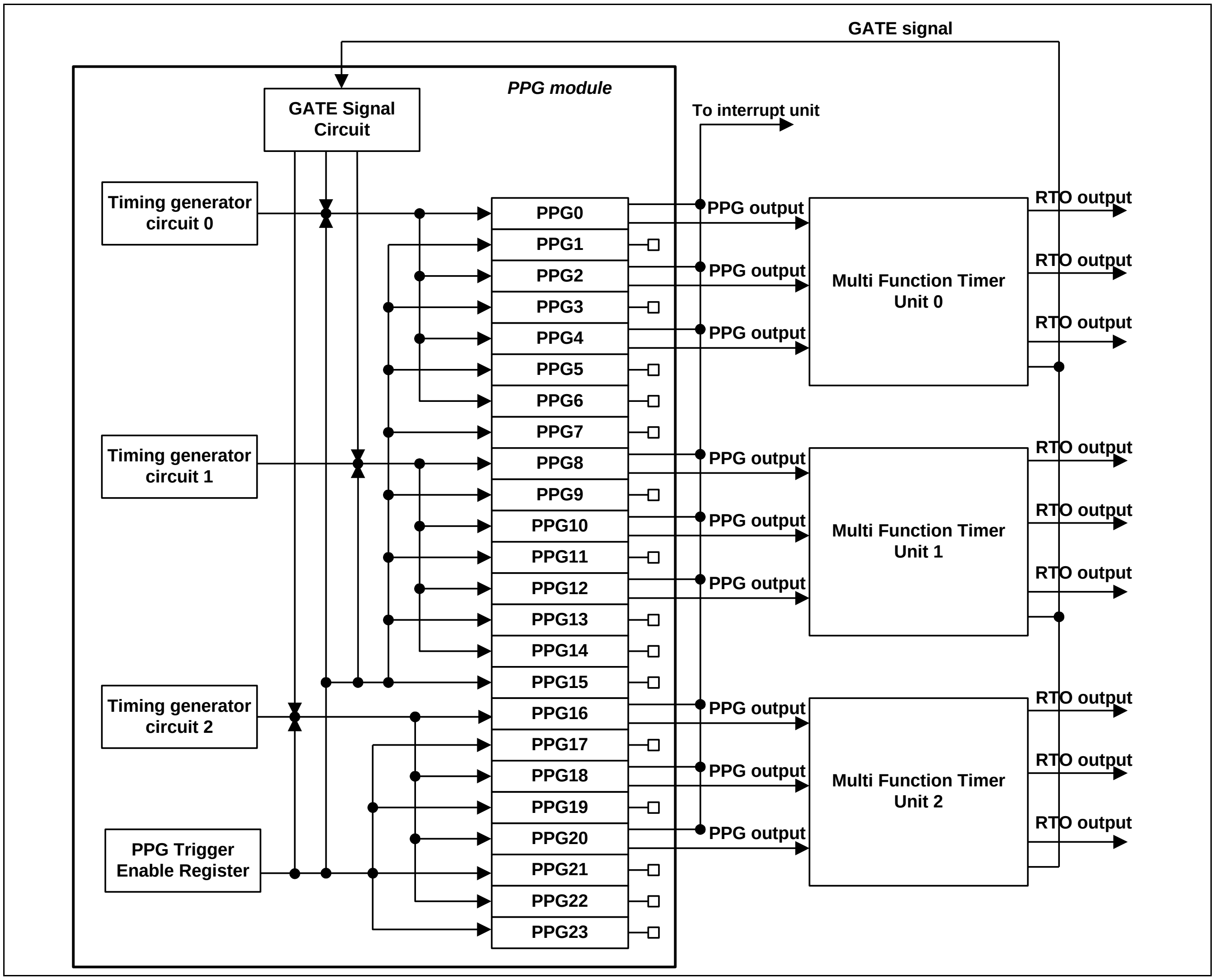
This chapter explains the PPG configuration.

CODE: 9BFPPGTOP-E01.3

1. Configuration

This section explains the PPG configuration.

The following shows the PPG configuration.



(Note) The number of MFT, Timing generator and PPG varies with the products.
For details, see "Data Sheet" of a product used.

● PPG connection

- PPG output is transferred from the output RTO pin of the multifunction timer via the multifunction timer module.
- PPG output and PPG interrupt are connected only to the PPG0, PPG2, PPG4, PPG8, PPG10, PPG12, PPG16, PPG18 and PPG20. Therefore, no output is obtained from other PPG channels.
- A PPG start factor can be set to a PPG channel with no output connected, but no output is obtained from such a PPG channel.
- Furthermore, any PPG operation mode (8-bit, 8+8-bit, 16-bit, or 16+16-bit mode) can be selected, but no output is obtained from a PPG channel with no output connected.

● Differences between timing generators 0, 1 and 2

- Timing generator 0
  - Compare Register : COMP0/COMP2/COMP4/COMP6
    PPG channel to be triggered : ch.0/ch.2/ch.4/ch.6
- Timing generator 1
  - Compare Register : COMP1/COMP3/COMP5/COMP7
  - PPG channel to be triggered : ch.8/ch.10/ch.12/ch.14
- Timing generator 2
  - Compare Register : COMP8/COMP10/COMP12/COMP14
  - PPG channel to be triggered : ch.16/ch.18/ch.20/ch.22

● Setting the EDGE bit in the PPG GATE Function Control Register

The EDGE bit in the PPG GATE Function Control Register (GATEC) can be set only to "EDGE=0".
*: Started only at the rising edge of the GATE signal.

● Combinations of operation modes and PPG channels with output enabled

PPG channel	8-bit mode	8+8-bit mode	16-bit mode	16+16-bit mode
PPG ch.0	PPG0 output	PPG0 output	PPG0 output	PPG0 output
PPG ch.1	Not available	PPG0 prescaler		
PPG ch.2	PPG2 output	PPG2 output	PPG2 output	PPG0 prescaler
PPG ch.3	Not available	PPG2 prescaler		
PPG ch.4	PPG4 output	PPG4 output	PPG4 output	PPG4 output
PPG ch.5	Not available	PPG4 prescaler		
PPG ch.6	Not available	Not available	Not available	PPG4 prescaler
PPG ch.7	Not available	Not available		
PPG ch.8	PPG8 output	PPG8 output	PPG8 output	PPG8 output
PPG ch.9	Not available	PPG8 prescaler		
PPG ch.10	PPG10 output	PPG10 output	PPG10 output	PPG8 prescaler
PPG ch.11	Not available	PPG10 prescaler		
PPG ch.12	PPG12 output	PPG12 output	PPG12 output	PPG12 output
PPG ch.13	Not available	PPG12 prescaler		
PPG ch.14	Not available	Not available	Not available	PPG12 prescaler
PPG ch.15	Not available	Not available		
PPG ch.16	PPG16 output	PPG16 output	PPG16 output	PPG16 output
PPG ch.17	Not available	PPG16 prescaler		
PPG ch.18	PPG18 output	PPG18 output	PPG18 output	PPG16 prescaler
PPG ch.19	Not available	PPG18 prescaler		
PPG ch.20	PPG20 output	PPG20 output	PPG20 output	PPG20 output
PPG ch.21	Not available	PPG20 prescaler		
PPG ch.22	Not available	Not available	Not available	PPG20 prescaler
PPG ch.23	Not available	Not available		

CHAPTER 7-2: PPG

This chapter explains the PPG function.

CODE: 9BFPPG-E02.5

1. Overview

This section describes the overview of PPG function.

The Programmable Pulse Generator (PPG) module can perform pulse output of arbitrary cycle and duty ratio controlled by timer operation.

● Features of PPG module

- 8-bit PPG operation mode is supported.
- 16-bit PPG operation mode is supported.
- 8+8-bit PPG operation mode is supported.
- 16+16-bit PPG operation mode is supported.
- The output level for the PPG can be inverted, including the initial output level during PPG stop.
- An arbitrary PPG cycle can be selected by selecting the PPG count clock.
- The PPG can output a pulse wave with an arbitrary duty ratio by making the register setting.
  This module can also be used in conjunction with an external circuit to form a D/A converter.
- If interrupt enable is set, an interrupt can be generated when the PPG output is changed (when the count for reload value ends, and an underflow occurs).

● PPG Start Trigger Method

PPG start trigger can be selected from following three methods.

- Start triggered by PPG start trigger register writing
- Start triggered by the Timing Generator Circuit
- Start triggered by GATE signal from the multifunction timer

Besides the above start methods, start by IGBT mode can also be selected. For details, see the chapter on PPG IGBT mode.

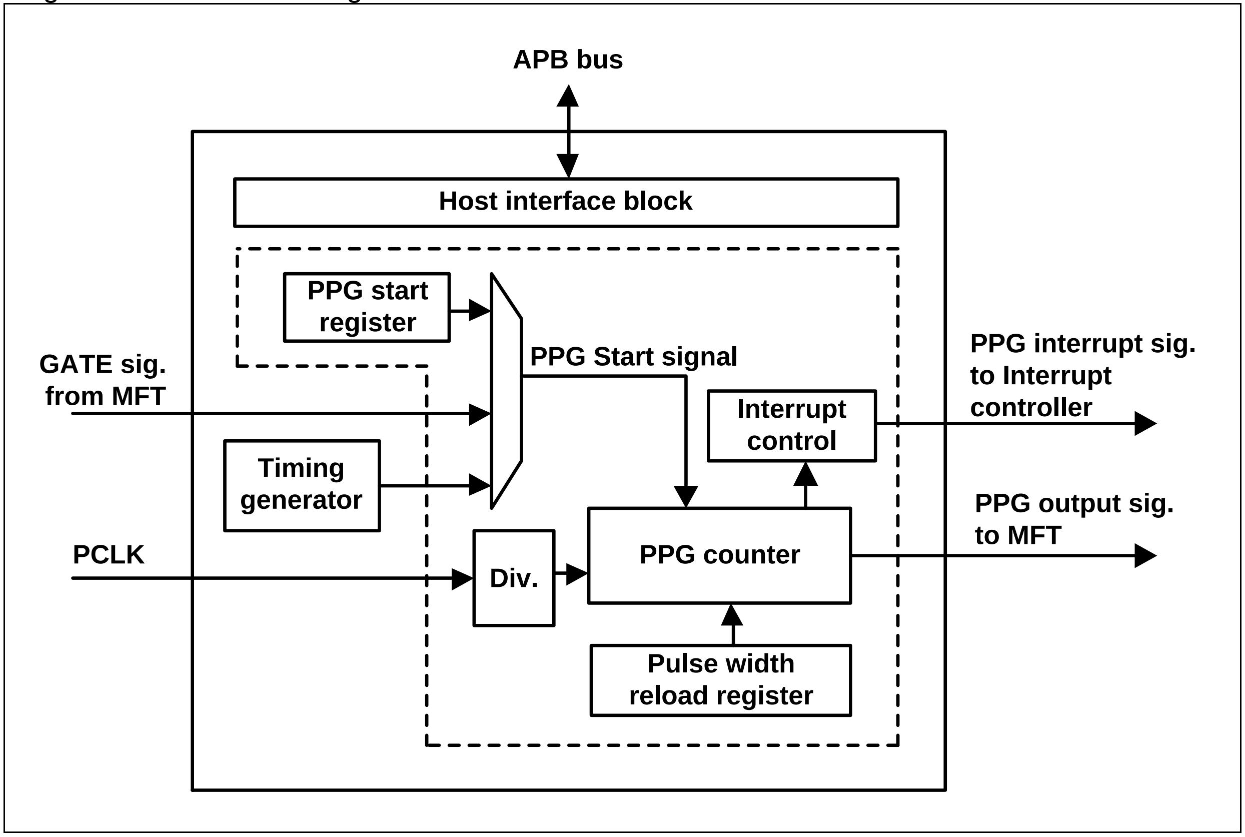
2. Configuration and Block Diagrams of PPG

This section shows the block diagrams of PPG.

■ PPG circuit block diagram

Figure 2-1 shows the block diagram of the PPG.

Figure 2-1 PPG block diagram



The PPG module consists of the following functional blocks.

- Host Interface Block
  This performs function control of each PPG block based on instructions from the CPU.

- PPG Start Register
  This register directly generates PPG start signals by register writing from the CPU.

- Timing Generator
  This is a circuit for starting multiple PPGs individually at the specified timing. PPG start signals are generated separately for multiple PPGs at the timing that was set by the internal compare registers. Internal configuration block diagrams are shown in Figure 3-14, Figure 3-15, and Figure 3-16.

- PPG Start Signal Selector
  The PPG start signal is selected by specifying the control register value. PPG start from PPG start register, PPG start from timing generator, or PPG start by GATE signal from multifunction timer (MFT) can be selected.

- PCLK Divider
  The PPG operates using the peripheral clock signal (PCLK) as a reference clock. The count clock used by the PPG counter is generated by the PCLK frequency divider. The pulse width of the output signal from PPG is specified based on the PCLK cycle.

- PPG Counter and Pulse Width Reload Register
  The Low pulse width and High pulse width of the PPG output signal is specified in the reload register from the CPU. The PPG counter performs a count operation of the specified pulse width and changes the PPG output signal.

- Interrupt Output Circuit
  The PPG interrupt signal is generated and output when the PPG output signal is changed.

In Figure 2-1, the section enclosed by the dashed lines indicates that multiple PPG channels are installed. Also, the connection topology of the PPG channel and the number of available channels change based on the selected PPG operation mode. The channel connection diagrams for each operation mode are shown in Figure 3-1, Figure 3-3, Figure 3-4, and Figure 3-5.

Among the PPG output signals obtained by PPG timer operation, some channel outputs can be output to external terminals by passing through a multifunction timer. Also, some PPG interrupt outputs are connected to interrupt controllers for enabling execution of interrupt processes.

For details on the PPG output terminal that is output to external terminals by passing through a multifunction timer and on PPG interrupts connected to an interrupt controller, see the chapter PPG configuration.

3. Operations of PPG

This section shows the operation of PPG.

3.1. PPG circuit operations

The PPG module can output pulse signals having arbitrary cycle and duty ratio.
The pulse output can be controlled based on the timer operation.

■ PPG Operations

This explains operation of the PPG timer circuit. The configuration in 8-bit PPG operation mode is shown in Figure 3-1. The input/output signal waveform is shown in Figure 3-2.

Figure 3-1 Configuration in 8-bit PPG Mode

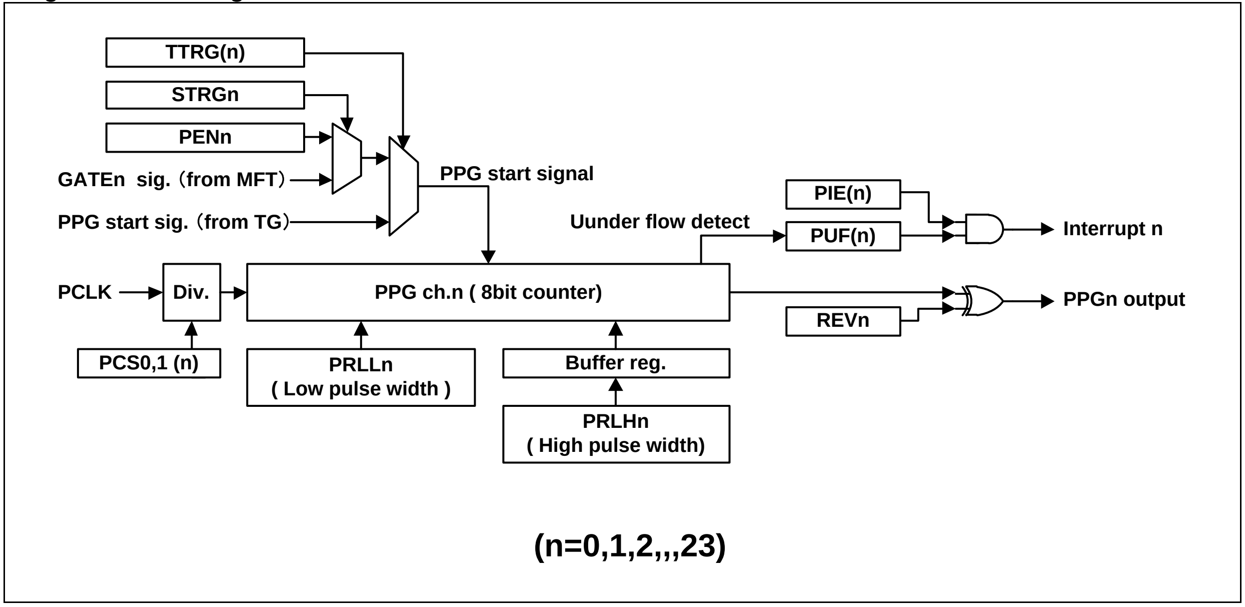
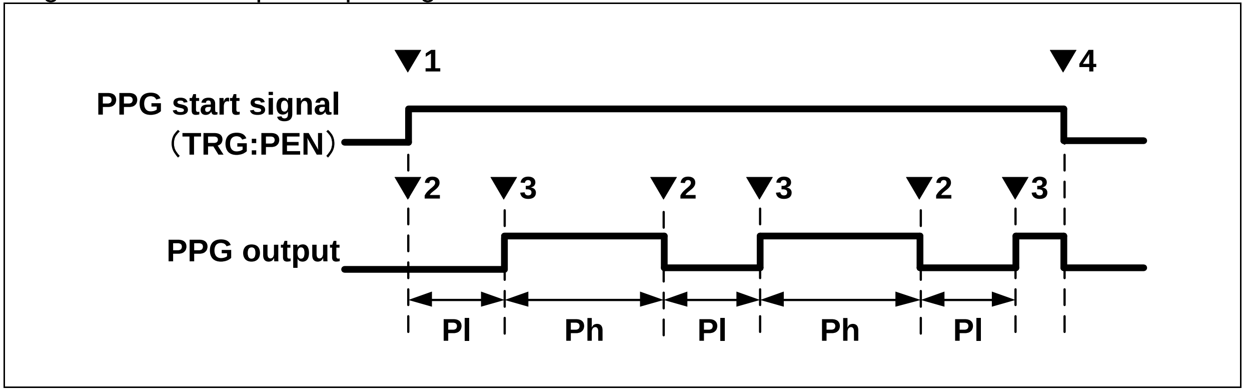


Figure 3-2 PPG Input/output Signal Waveform



The ▼1 to ▼4 in the text below indicate the timing shown in Figure 3-2. This section explains operation when the start method by writing the PPG start register (TRG:PEN) is specified and the positive polarity (REVC:REV=0) is specified.

The initial level of PPG output is Low. The PPG starts operation by asserting the start signal. At the timing in ▼1, "1" is written to the PPG start register (TRG:PEN) from the CPU, and the PPG start signal is asserted. During the period that the PPG start signal is asserted, the PPG timer circuit continues to repeat the operations of ▼2 and ▼3 below.

At the timing of ▼2, the PPG output is set to the Low level. Also, the Low width value is loaded to the PPG counter from the Low width setting reload register (PRLL), and the countdown is started. A wait is performed until the time specified in the Low width has elapsed.

At the timing of ▼3, the PPG output is set to the High level. Also, the High width value is loaded to the PPG counter from the High width setting reload register (PRLH), and the countdown is started. A wait is performed until the time specified in the High width has elapsed.

This operation generates output waveforms having the specified Low width and High width. In Figure 3-2, the Low width (Pl) and High width (Ph) of the output pulse can be specified as shown below.

T (count clock cycle)	= Count clock cycle selected by the PPGC: PCS1 and PCS0 registers
Pl (PPG output Low width)	= T x (PRLL register value + 1)
Ph (PPG output High width)	= T x (PRLH register value + 1)

At the timing in ▼4, "0" is written to the PPG start register (TRG:PEN) from the CPU, and the start signal is negated. The PPG timer circuit stops the count operation. Even if the count operation is in progress, output is set to the Low level.

■ PPG Operation Modes

The PPG has an 8-bit length counter for each channel. Multiple PPG channels can be connected to enable generation of output pulses with longer count lengths. The PPG operation modes below can be used.

- 8-bit PPG Operation Mode
  Figure 3-1 shows the channel connection diagram for this operation mode. In this mode, operation is performed as an 8-bit length counter PPG independently for each channel. The output pulse width can be specified as an 8-bit length value.

- 16-bit PPG Operation Mode
  Figure 3-3 shows the channel connection diagram for this operation mode. In this mode, two channels are connected to perform operation as a 16-bit length counter PPG. The output pulse width can be specified as a 16-bit length value.

- 8+8-bit PPG Operation Mode
  Figure 3-4 shows the channel connection diagram for this operation mode. Two channels are used. One of the PPGs operates as a prescaler. (This is referred to as the "prescaler side" below) The other PPG operates using the prescaler side output as an operation clock. (This is referred to as the "PPG output side" below.) In this mode, operation is performed by specifying the output pulse width as an 8-bit length value separately for the prescaler side and PPG output side. The counter on the PPG output side performs the count operation at both the rising and falling edges of the prescaler side output signal.

- 16+16-bit PPG Operation Mode
  Figure 3-5 shows the channel connection diagram for this operation mode. Four channels are used. Two channels are connected respectively to perform operation as a 16-bit length counter. One PPG operates as the prescaler side. The other PPG operates as the PPG output side using the prescaler side output as an operation clock. In this mode, operation is performed by specifying the output pulse width as a 16-bit length value separately for the prescaler side and PPG output side. The counter on the PPG output side performs the count operation at both the rising and falling edges of the prescaler side output signal.

Figure 3-3 Configuration in 16-bit PPG Mode

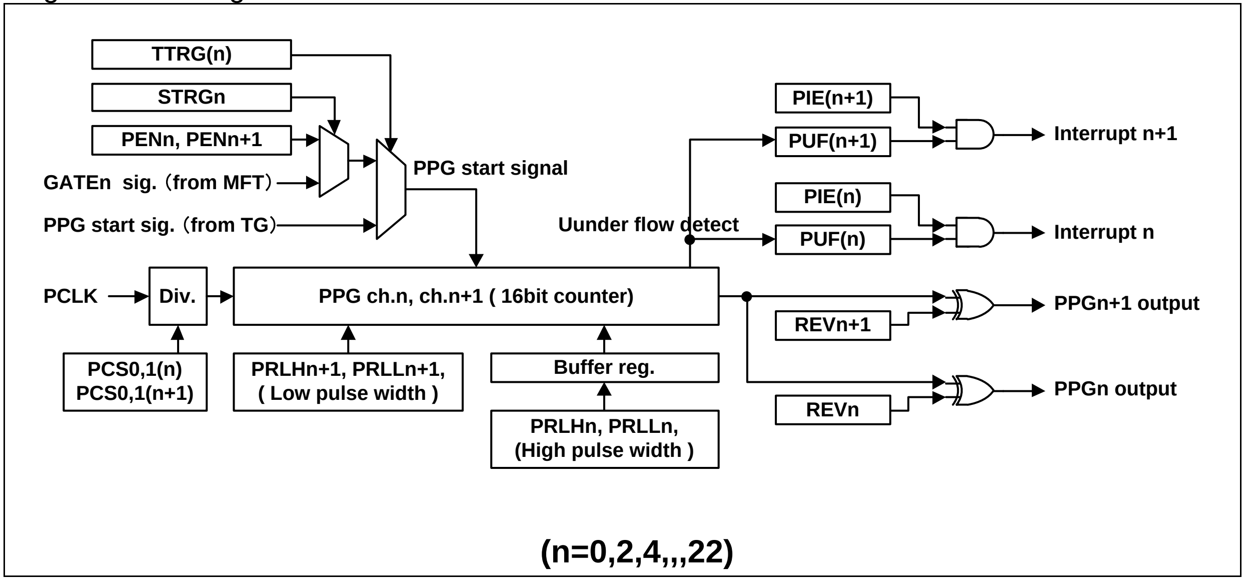


Figure 3-4 Configuration in 8+8-bit PPG Mode

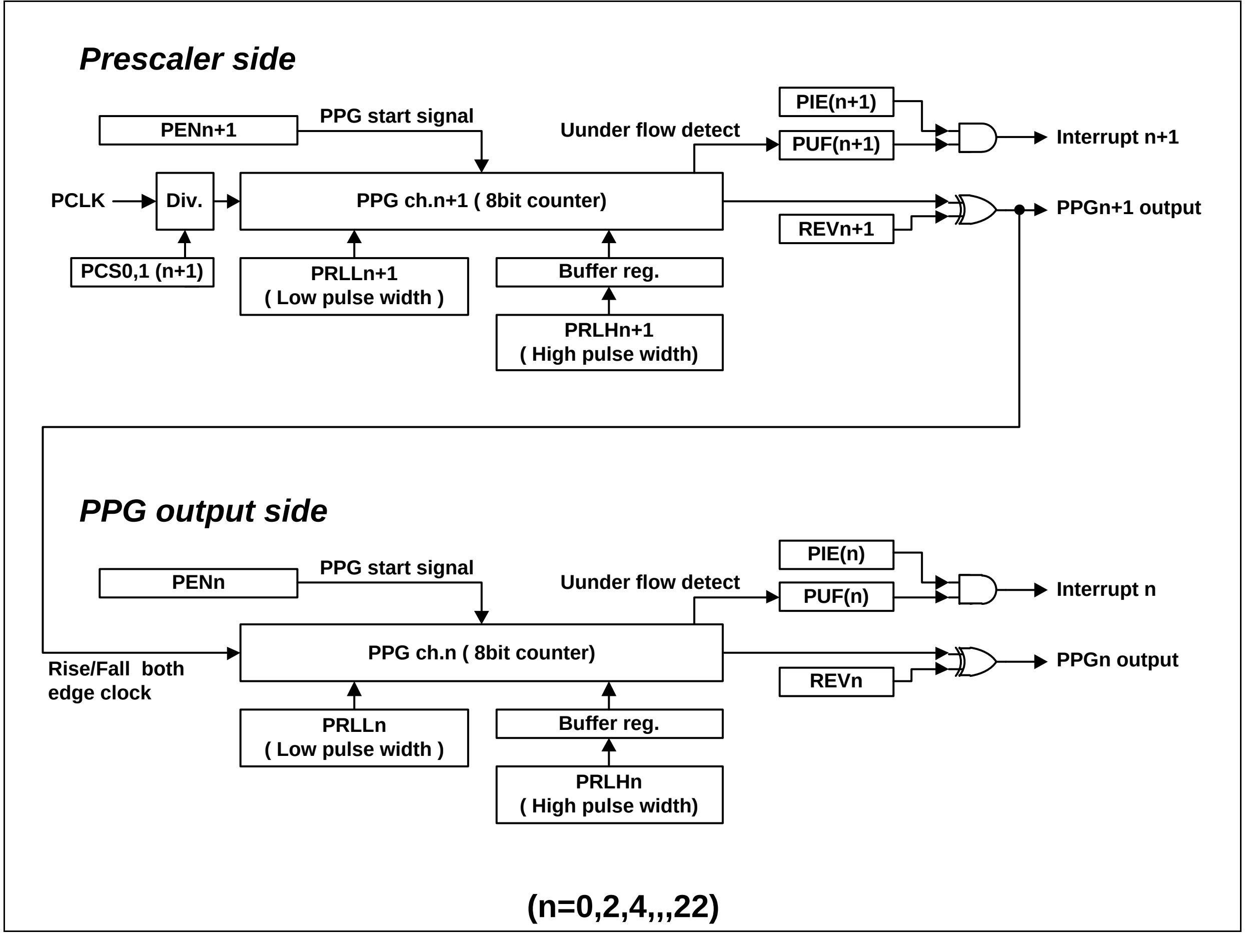
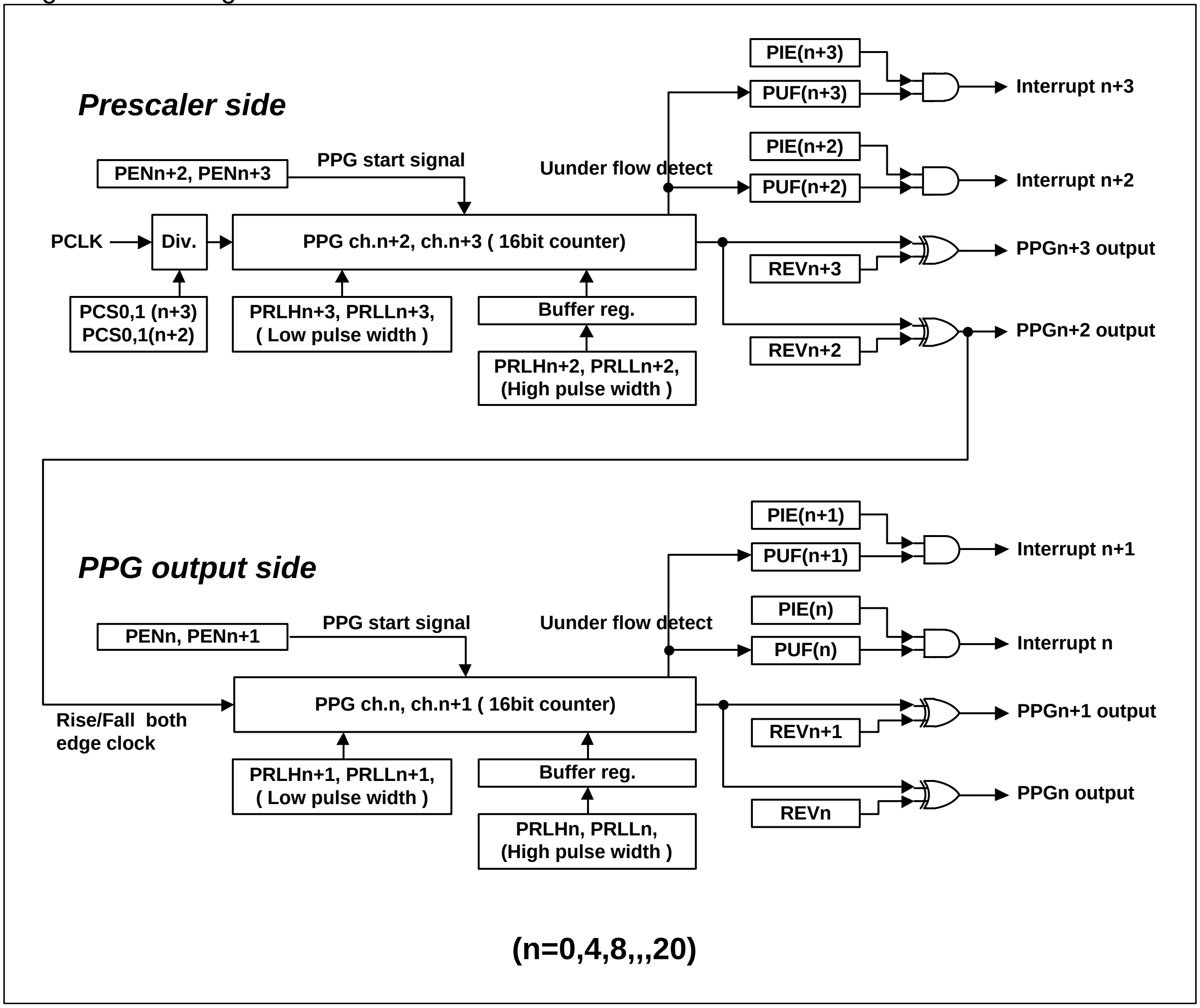


Figure 3-5 Configuration in 16+16-bit PPG Mode



■ Selecting the PPG Operation Mode

The PPGC:MD[1:0] register setting can be used to select the operation mode for each PPG channel. The operation mode for each channel (channel connection, used as prescaler side, used as PPG output side) is uniquely determined by this register setting.

The PPGCn:MD[1:0] register exists only for registers where n is even. The operation mode for both channel n (even numbers) and channel n+1 (odd numbers) is determined by specifying the PPGCn:MD[1:0] register. For 16+16-bit PPG mode, both the PPGCn:MD[1:0] register and PPGCn+2:MD[1:0] register are set to "11".

Table 3-1 shows a list of PPG operation modes which are selected based on the PPGCn:MD[1:0] register value and PPGCn+2:MD[1:0] register value.

The index n indicating the register number and channel number in Table 3-1 stands for n=0, 4, 8, 12, 16, and 20. The same combination can be specified for ch.0 to ch.3, ch.4 to ch.7, ch.8 to ch.11, ch.12 to ch.15, ch.16 to ch.19, and ch.20 to ch.23.

The operation mode is selected before starting the PPG. During PPG operation, the operation mode cannot be changed.

Table 3-1 Selecting PPG Operation Mode

Register setting				Selected operation mode			
PPGCn		PPGCn+2		Ch.n+0	Ch.n+1	Ch.n+2	Ch.n+3
MD1	MD0	MD1	MD0				
0	0	0	0	8 PPG : ☆ ◎	8 PPG : ★ ◎	8 PPG : ☆ ◎	8 PPG : ★ ◎
0	0	0	1	8 PPG : ☆ ◎	8 PPG : ★ ◎	8+8 out : ★ ● ⬅	8+8 pre: ★ ◎
0	0	1	0	8 PPG : ☆ ◎	8 PPG : ★ ◎	16 PPG : ☆○	
0	0	1	1	Setting is prohibited.			
0	1	0	0	8+8 out: ★ ● ⬅	8+8 pre: ★◎	8 PPG : ☆◎	8 PPG : ★ ◎
0	1	0	1	8+8 out: ★ ● ⬅	8+8 pre: ★◎	8+8 out : ★ ●	8+8 pre: ★ ◎
0	1	1	0	8+8 out: ★ ● ⬅	8+8 pre: ★◎	16 PPG : ☆○	
0	1	1	1	Setting is prohibited.			
1	0	0	0	16 PPG : ☆ ○		8 PPG : ☆ ◎	8 PPG : ★ ◎
1	0	0	1	16 PPG : ☆ ○		8+8 out : ★ ● ⬅	8+8 pre: ★ ◎
1	0	1	0	16 PPG : ☆ ○		16 PPG : ☆ ○	
1	0	1	1	Setting is prohibited.			
1	1	0	0				
1	1	0	1				
1	1	1	0				
1	1	1	1	16+16 out: ★ ● ⬅		16+16 pre: ★ ○	

Details of Table 3-1 are described below.

8 PPG: Indicates that 8-bit PPG operation mode is selected.

16 PPG: Indicates that 16-bit PPG operation mode is selected.
Two channels are connected, and 16-bit length counter operation is performed.

8+8 pre: Indicates that 8+8-bit PPG operation mode is selected.
Indicates that prescaler side operation is performed.
Arrows in the table indicate the output direction of the prescaler clock.

8+8 out: Indicates that 8+8-bit PPG operation mode is selected.
Indicates that PPG output side operation is performed.

16+16 pre: Indicates that 16+16-bit PPG operation mode is selected.
Two channels are connected, and 16-bit length counter operation is performed.
Indicates that prescaler side operation is performed.
Arrows in the table indicate the output direction of the prescaler clock.

16 + 16 out: Indicates that 16+16-bit PPG operation mode is selected.
Two channels are connected, and 16-bit length counter operation is performed.
Indicates that PPG output side operation is performed.

☆ mark, ★ mark: Indicates the available start methods. This is described in the section on selecting the PPG start method.

◎ mark, ○ mark, ● mark: Indicates the available count clocks. This is described in the section on selecting the count clock.

■ Selecting the PPG Start Method

The PPG can be started by one of the three methods below.

- Direct writing from the CPU to the PPG start register (TRG:PEN) is used to start PPG
- GATE signal from a multifunction timer is used to start the PPG
- PPG start signal of a timing generator is used to start the PPG

Besides the above start methods, start method by IGBT mode can also be selected. For details, see the chapter on PPG IGBT mode.

For the operation modes indicated by the ☆mark in Table 3-1, the PPG start method can be selected from the above three. For the operation modes indicated by the ★mark, only the start method of direct writing to the PPG start register (TRG:PEN) can be used. The PPG start method is selected by the GATEC:STRG and PPGC:TTRG registers. This setting is used to connect the corresponding start signal to the PPG. The register settings, PPG operation modes, and selected PPG start signals are shown in Table 3-2. The GATEC:STRGn and PPGCn:TTRG registers exist only when n is an even number. The start methods for both channel n (even numbers) and channel n+1 (odd numbers) are determined by specifying the registers.

Table 3-2 Selecting the PPG Start Method

Register setting		PPG operation mode	PPG start signal	Remarks
GATEC: STRGn	PPGCn: TTRG			
0	0	8-bit	PPG start register (TRG:PEN)	
		16-bit		*1
		8+8-bit		*1
		16+16-bit		*1, *2
1	0	8-bit	GATE signal from multifunction timer	*3
		16-bit		*4
		8+8-bit	Setting prohibited	*7
		16+16-bit		*7
X	1	8-bit	Start signal from timing generator	*5
		16-bit		*6
		8+8-bit	Setting prohibited	*7
		16+16-bit		*7

*1: For operation modes that use multiple PPG channels (8+8-bit, 16-bit, and 16+16-bit) and the start method of writing the PPG start register (TRG:PEN), "1" is written simultaneously to the TRG:PEN registers of all channels being used to start the PPG. Also, "0" is written simultaneously to the TRG:PEN registers to stop the PPG. The count cycle may be shifted if values are not written simultaneously.

*2: In 16+16-bit PPG operation mode, set both the n channel and n+2 channel registers to "0" for the GATEC:STRGn and PPGCn:TTRG registers.

*3: In this case, the GATE signal from the multifunction timer is connected to even channels only. Odd channels connect TRG:PEN.

*4: In this case, GATE signals from the multifunction timer are connected to both even and odd channels.

*5: In this case, the start signal from the timing generator is connected to even channels only. Odd channels connect TRG:PEN.

*6: In this case, the start signals from the timing generator are connected to both the even and odd channels.

*7: In the case of 8+8-bit mode and 16+16-bit mode, the start by writing the PPG start register (TRG:PEN) only can be selected.

■ Selecting the Count Clock

The count clock for each PPG channel can be selected from four frequency dividing ratios by using the PCLK frequency divider. Table 3-3 shows the register settings and selected count clock.

Table 3-3 Count Clock Selection Table

PPGC:PCS1	PPGC:PCS0	Count clock operation
0	0	Count clock performs 1 count for every PCLK
0	1	Count clock performs 1 count for every 4 PCLK cycles.
1	0	Count clock performs 1 count for every 16 PCLK cycles
1	1	Count clock performs 1 count for every 64 PCLK cycles

In Table 3-1, the count clock can be selected using the PPGC:PCS1 and PCS0 registers for operation modes indicated by the ◎ mark and ○ mark. For the operation modes indicated by the ○ mark, the count clock selection settings for the even channel side and odd channel side (PSC1, PSC0) must always be set to the same value. For the operation modes indicated by the ● mark, both edges of the prescaler side output are used for the count clock. The PSC1 and PSC0 register settings for these channels are ignored.

■ Specifying the Reload Register and Pulse Width

The pulse width of the PPG output signal is specified by the reload registers (PRLH and PRLL). The pulse width can be changed during PPG operation. Both the High width and Low width are specified. The pulse width that is output is a value found by multiplying the count clock cycle by a value that is +1 added to the value written to the reload register. If PPG channels are connected to form a 16-bit length, the 8-bit reload register that specifies the pulse width is also connected for specifying a 16-bit length value. For details, see 5-10 PPG reload registers n (PRLHn and PRLLn n=0 to 23). The relationship between the setting examples and output pulse widths are shown in Table 3-4.

Table 3-4 Examples of Reload Register Setting Values

PPG bit width	Reload register value	Output pulse width
8-bit	0x00	1 x Count clock cycle
	0x01	2 x Count clock cycle
	0xFF	256 x Count clock cycle
16-bit	0x0000	1 x Count clock cycle
	0x0001	2 x Count clock cycle
	0x00FF	256 x Count clock cycle
	0x0100	257 x Count clock cycle
	0xFFFF	65536 x Count clock cycle

■ Buffer Function of High Width Setting Reload Register

The High width setting reload register includes a buffer function. Operation of the buffer function is shown in Figure 3-6.

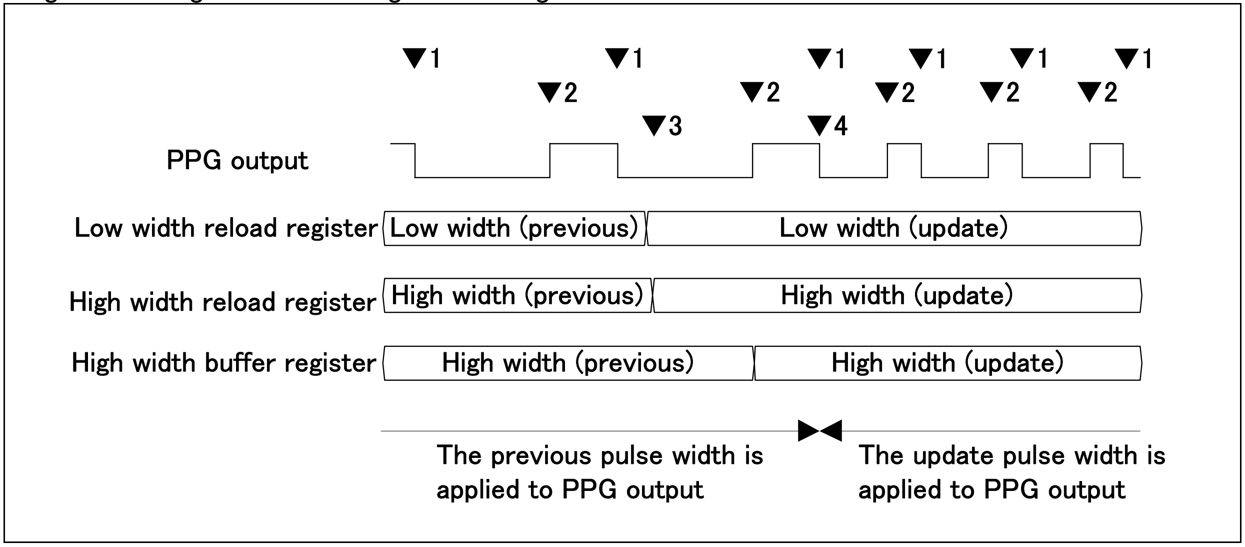
When the PPG output signal changes from High to Low (▼1 in the figure), the PPG counter imports the setting value of the Low width from the reload register. When the PPG output signal changes from Low to High (▼2 in figure), the PPG counter imports the setting value of the High width from the buffer register.

At the timing of ▼3, if the Low width setting value and High width setting value are changed from the CPU, the High width setting before the update is stored to the buffer register. At the next ▼2 timing, the PPG counter imports the High width setting value before the update from the buffer register and applies it to the output pulse width.

For this reason, until the timing at ▼4, the Low width and High width setting values before the update are applied in the PPG output. At the timing of ▼4, the Low width and High width setting values after the update are applied in the PPG output.

When the output pulse width settings are updated, a combination of the Low width and High width settings can be maintained.

Figure 3-6 High Width Setting Buffer Register Function


▼1
▼2
▼3
▼4
PPG output
Low width reload register
Low width (previous)
Low width (update)
High width reload register
High width (previous)
High width (update)
High width buffer register
High width (previous)
High width (update)
The previous pulse width is applied to PPG output
The update pulse width is applied to PPG output

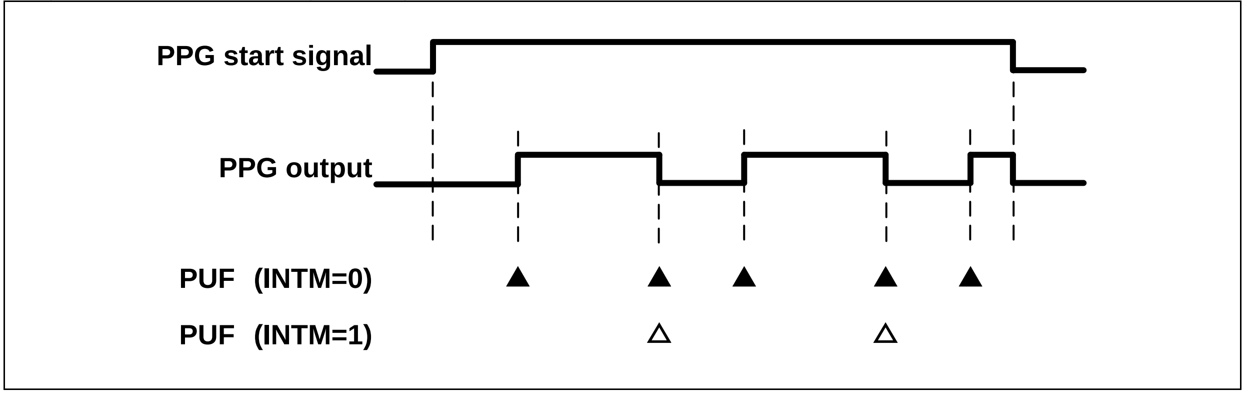
■ Interrupts

When the PPG changes the output signal (when the reload value count is ended and an underflow has occurred), 1 is set to the PUF of the PPG operation mode control register (PPGC). The selections below are performed using PPGCn: INTM.

- When PPGC:INTM=0, the setting is made when both the Low pulse and the High pulse end.
- When PPGC:INTM=1, the setting is made when the High pulse ends.

Figure 3-7 shows the PUF setting timing selection based on the INTM value. The PUF setting timing when INTM=0 (indicated by ▲) and when INTM=1 (indicated by △) is shown.

Figure 3-7 PUF Setting Timing Selection Based on INTM

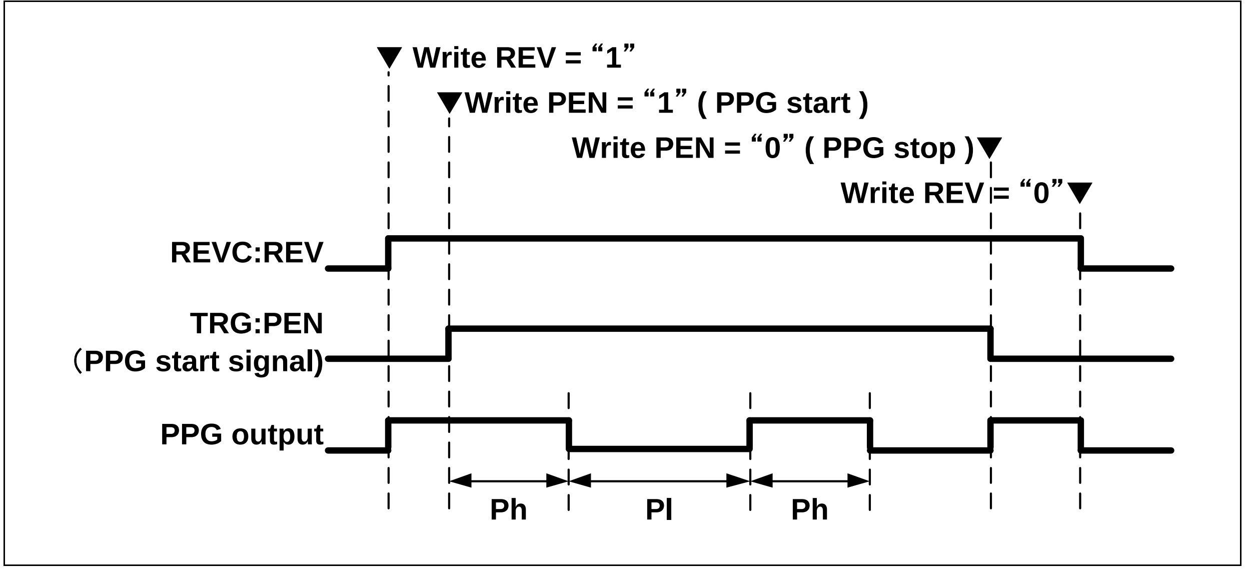


If the PPG channels are connected to perform operation using a 16-bit length, the PUF for both the even and odd channels are set simultaneously.
The PUFs that are set from PPG can be cleared by writing "0" to PUF from the CPU.
If interrupt enable is set by the PPG operation mode control register (PIE=1), PUF can be used to assert the interrupt signal.

■ Polarity Reversal by REVn Register

The polarity of the PPG output signal can be reversed using the REVC:REV register setting. Figure 3-8 shows the output waveform for REV=1 and the 8-bit PPG operation mode.

Figure 3-8 Output Waveform When REVn=1



The connection diagrams of Figure 3-1, Figure 3-3, Figure 3-4, and Figure 3-5 show configurations where the PPG output is reversed by the REV register value directly. For this reason, when REV=1, the followings are performed.

- The output level before operation start of the PPG output and output level after operation stop are reversed to the High level.
- The Low-High of the output pulse is reversed, and the relationship of the Low width setting and High width setting of the reload register is reversed.
- PUF is set when PPGC:INTM=1 and when the Low pulse ends.
- In 8+8-bit PPG operation mode and 16+16-bit PPG operation mode, the operation clock supplied to the PPG output side from the prescaler side is reversed.

In Figure 3-8, the Low width (Pl) and High width (Ph) for the output pulse can be specified as shown below.

T (Count clock cycle）	= Count clock cycle selected by PPGC:PCS1 and PCS0 registers
Pl (PPG output Low width)	= T x (PRLH register value + 1)
Ph (PPG output High width)	= T x (PRLL register value + 1)

■ 8-bit PPG Operation Mode Example

This section shows an operation example of 8-bit PPG operation mode using PPG-ch.0. Table 3-5 shows the register initial settings used in this operation example. Figure 3-9 shows the output waveforms of this operation example.

Table 3-5 8-bit PPG Operation Mode Register Setting Example

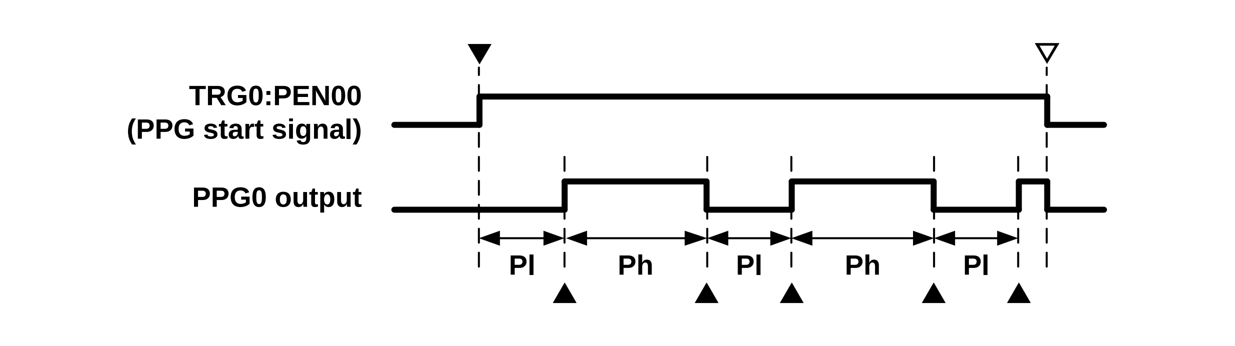
Register name	Bit write value	Setting description	Remarks

PPGC0	TTRG=0
	MD1,MD0=00
	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		Start by TRG0:PEN00
		8-bit PPG operation mode
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*1

GATEC0	STRG0=0
	STRG2=X
		Start by TRG0:PEN00
		This is no relation setting for other PPG channel.	

REVC0	REV00=0 REV01 to 15=X	PPG0 is output at positive polarity This is no relation setting for other PPG channel.	

Figure 3-9 Example of 8-bit PPG Operation Mode Output Waveform



The symbols in Figure 3-9 have the meanings shown below.

▼: PPG operation start (See *2)
▽: PPG operation stop (See *3)
▲: PPGC0:PUF setting timing (See *4)
Pl: PPG0 output Low width
Ph: PPG0 output High width

The Low width (Pl) and High width (Ph) of the PPG0 output can be specified as shown below.

T (Count clock cycle) = PCLK cycle x 4 (See *1)
Pl (PPG0 output Low width) = T x (PRLL0 register value + 1)
Ph (PPG0 output High width) = T x (PRLH0 register value + 1)

*1: T (Count clock cycle) is determined by the clock cycle selected by the count clock selection registers (PPGC0:PCS1,PCS0) of PPG0.
*2: When 1 is written to TRG0:PEN00, the PPG start signal is asserted, and the PPG starts operation.
*3: When 0 is written to TRG0:PEN00, the PPG start signal is negated, and the PPG stops operation.
*4: Based on the specified PPGC0:INTM=0, PPGC0:PUF is set at the timing indicated by ▲.

■ 8+8-bit PPG Operation Mode Example

This section shows an operation example of 8+8-bit PPG operation mode using PPG-ch.0 and ch.1. Table 3-6 shows the register settings used in this operation example. Figure 3-10 shows the output waveform of this operation example.

Table 3-6 8+8-bit PPG Operation Mode Register Setting Example

Register name	Bit write value	Setting description	Remarks

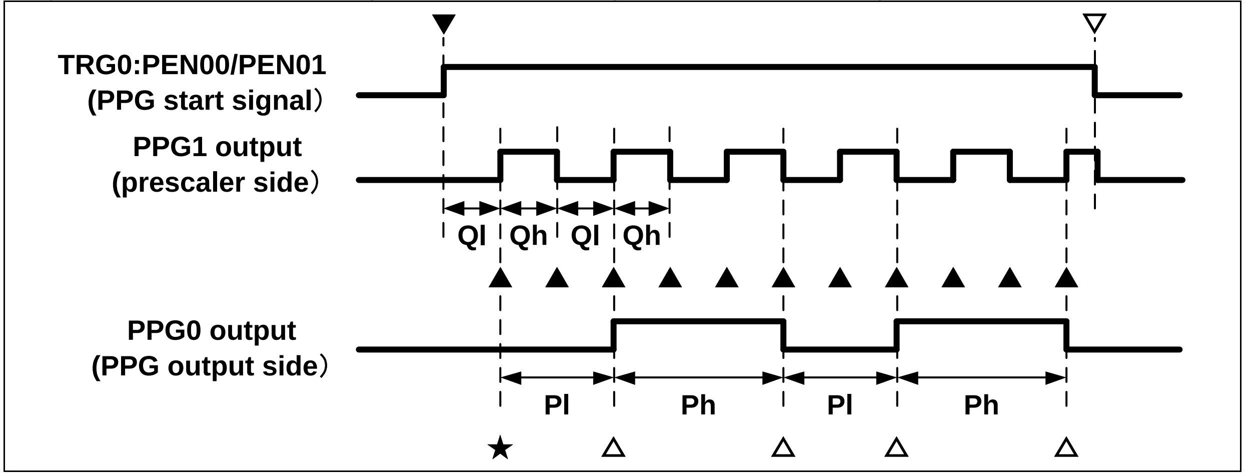
PPGC0	TTRG=0
	MD1,MD0=01
	PCS1,PCS0=00
	INTM=0
	PUF=0
	PIE=0
		Start by TRG0:PEN00/PEN01
		8+8-bit PPG operation mode
		Setting is ignored
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*2

PPGC1	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*1

GATEC0	STRG0=0
	STRG2=X
		Start by TRG0:PEN00/PEN01
		This is no relation setting for other PPG channel.	

REVC0	REV00=0
	REV01=0
	REV02 to 15=X
		PPG0 (PPG output side) is output at positive polarity
		PPG1 (prescaler side) is output at positive polarity
		This is no relation setting for other PPG channel.
			*9

Figure 3-10 8+8-bit PPG Operation Mode Output Waveform Example



The symbols in Figure 3-10 have the meanings shown below.

▼:	PPG operation start (See *3)
▽:	PPG operation stop (See *4)
▲:	PPGC1:PUF setting timing (prescaler side, see *5)
△:	PPGC0:PUF setting timing (PPG output side, see *6)
★:	PPG output side count operation start timing (See *7)
Ql:	PPG1 output (prescaler side) Low pulse width
Qh:	PPG1 output (prescaler side) High pulse width
Pl:	PPG0 output (PPG output side) Low pulse width
Ph:	PPG0 output (PPG output side) High pulse width

The Low width (Ql) and High width (Qh) of the PPG1 output (prescaler side) and the Low width (Pl) and High width (Ph) of the PPG0 output (PPG output side) can be specified as shown below.

T (Count clock cycle）	= PCLK cycle x 4 (See *1)
Ql (PPG1 output pulse Low width)	= T x (PRLL1 register value + 1)
Qh (PPG1 output pulse High width)	= T x (PRLH1 register value + 1)
Qa (PPG output side clock cycle)	= (Ql + Qh)/2 (Ql and Qh average value: See *8)
Pl (PPG0 output Low width)	= T x (PRLL0 register value + 1)
Ph (PPG0 output High width)	= T x (PRLH0 register value + 1)

*1: T (Prescaler side clock cycle) is determined by the clock cycle selected by the count clock selection registers (PPGC1:PCS1 and PCS0) of PPG1 (Prescaler side).

*2: The values of the count clock selection registers (PPGC0:PCS1 and PCS0) of PPG0 (PPG output side) are ignored.

*3: When 11 is written simultaneously to TRG0:PEN00 and PEN01, the PPG start signal is asserted, and the PPG starts operation. If 11 is not written simultaneously, the count cycle may be shifted.

*4: When "00" is written simultaneously to TRG0:PEN00 and PEN01, the PPG start signal is negated, and the PPG stops operation. If "00" is not written simultaneously, the count cycle may be shifted.

*5: Based on the specified PPGC1:INTM=0, PPGC1:PUF is set at the timing indicated by ▲. (Prescaler side)

*6: Based on the specified PPGC0:INTM=0, PPGC0:PUF is set at the timing indicated by △. (PPG output side)

*7: PPG0 (PPG output side) starts counting from the position of ★ after Ql has elapsed after the start instruction in ▼. The PPG output side imports the output pulse width from the reload register at the timing in ★. After the ▼ start instruction, if the output pulse width setting on the PPG output side was overwritten before the timing at ★, the setting for the output pulse width before that time is not applied.

*8: The PPG0 (PPG output side) count operation is performed at both the PPG1 (prescaler side) output rising and falling edges. For this reason, in the above equation, Qa is the average value of Ql and Qh. It is recommended that the same value be used for the Low pulse width and High pulse width at the prescaler side. Be careful because, if the values for the Low pulse width and High pulse width at the prescaler side are different, when the pulse count on the PPG output side is odd, the output pulse width on the PPG output side will not match the above equation.

*9: The PPG0 (PPG output side) output signal and PPG1 (Prescaler side) output signal can be reversed using the REVC0:REV00 and REV01 registers.

■ 16-bit PPG Operation Mode Example

This section shows an operation example of 16-bit PPG operation mode using PPG-ch.0 and ch.1. Table 3-7 shows the register settings used in this operation example. Figure 3-11 shows the output waveform of this operation example.

Table 3-7 16-bit PPG Operation Mode Register Setting Example

Register name	Bit write value	Setting description	Remarks

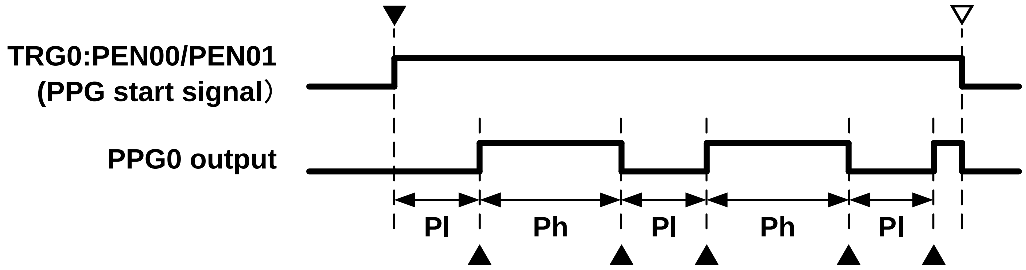
PPGC0	TTRG=0
	MD1,MD0=10
	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		Start by TRG0:PEN00/PEN01
		16-bit PPG operation mode
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*1

PPGC1	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*1

GATEC0	STRG0=0
	STRG2=X
		Start by TRG0:PEN00/PEN01
		This is no relation setting for other PPG channel.	

REVC0	REV00=0
	REV01=0
	REV02 to 15=X
		PPG0 is output at positive polarity
		PPG1 is output at positive polarity
		This is no relation setting for other PPG channel.
			*5

Figure 3-11 16-bit PPG Operation Mode Output Waveform Example



The symbols in Figure 3-11 have the meanings shown below.

▼:	PPG operation start (See *2)
▽:	PPG operation stop (See *3)
▲:	PPGC0:PUF and PPGC1:PUF setting timing (See *4)
Pl:	PPG0 output Low pulse width
Ph:	PPG0 output High pulse width

The Low width (Pl) and High width (Ph) of the PPG0 output can be specified as shown below.

T (Count clock cycle)	= PCLK cycle x 4 (See Note 4.)
Pl (PPG0 output pulse Low width)	=T x (PRLH1 register value x 256 + PRLL1 register value + 1)
Ph (PPG0 output pulse High width)	=T x (PRLH0 register value x 256 + PRLL0 register value + 1)

*1: The values for the PPG0 and PPG1 count clock selection registers (PPGC0:PCS1 and PCS0) and (PPGC1:PCS1 and PCS0) must be the same. T (Count clock cycle) is determined by this value.

*2: When 11 is written simultaneously to TRG0:PEN00 and PEN01, the PPG start signal is asserted, and the PPG starts operation. If 11 is not written simultaneously, the count cycle may be shifted.

*3: When 00 is written simultaneously to TRG0:PEN00 and PEN01, the PPG start signal is negated, and the PPG stops operation. If 00 is not written simultaneously, the count cycle may be shifted.

*4: Based on the specified PPGC0:INTM=0, PPGC0:PUF is set at the timing indicated by ▲. Based on the specified PPGC1:INTM=0, PPGC1:PUF is set at the timing indicated by ▲. In this case, both flags are set simultaneously.

*4: Although it is omitted in the figure, when REVC0:REV00=0 and REVC0:REV01=0 are specified, the same output waveform as the PPG0 output is obtained for PPG1 output.

■ 16+16-bit PPG Operation Mode Example

This section shows an operation example of 16+16-bit PPG operation mode using PPG-ch.0, ch.1, ch.2, and ch.3. Table 3-8 shows the register settings used in this operation example. Figure 3-12 shows the output waveform of this operation example.

Table 3-8 16+16-bit PPG Operation Mode Register Setting Example

Register name	Bit write value	Setting description	Remarks

PPGC0	TTRG=0
	MD1,MD0=11
	PCS1,PCS0=00
	INTM=0
	PUF=0
	PIE=0
		Start by TRG0:PEN00/PEN01/PEN02/PEN03
		16+16-bit PPG operation mode
		Setting is ignored
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*2

PPGC1	PCS1,PCS0=00
	INTM=0
	PUF=0
	PIE=0
		Setting is ignored
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*2

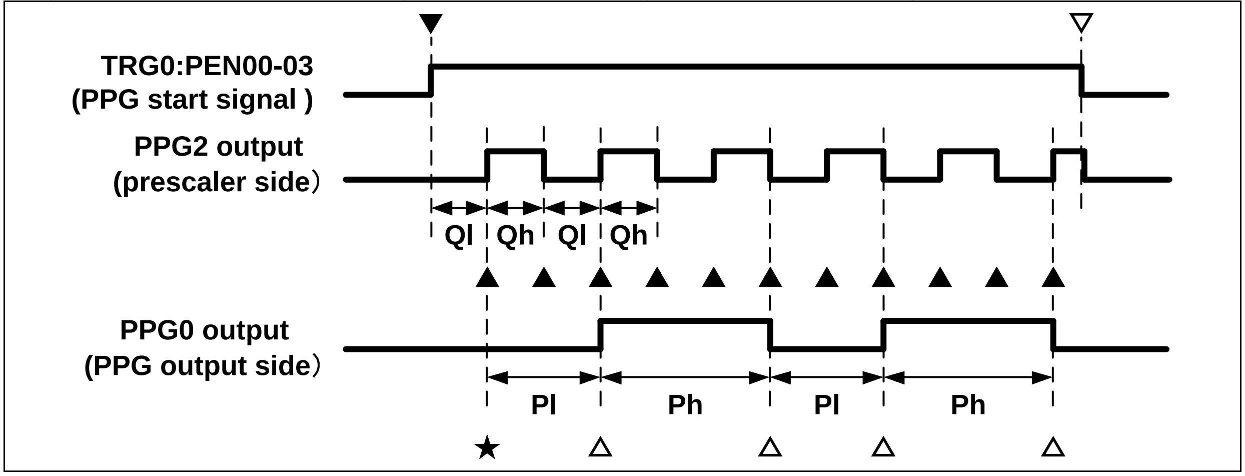
PPGC2	TTRG=0
	MD1,MD0=11
	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		Start by TRG0:PEN00/PEN01/PEN02/PEN03
		16+16-bit PPG operation mode
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*1

PPGC3	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited
			*1

GATEC0	STRG0=0
	STRG2=0
		Start by TRG0:PEN00/PEN01/PEN02/PEN03
		Start by TRG0:PEN00/PEN01/PEN02/PEN03	

REVC0	REV00=0
	REV01=0
	REV02=0
	REV03=0
	REV04 to 15=X
		PPG0 and PPG1 (PPG output side) are output at positive polarity
		PPG2 and PPG3 (prescaler side) are output at positive polarity
		This is no relation setting for other PPG channel.
			*9

Figure 3-12 16+16-bit PPG Operation Mode Output Waveform Example



The symbols in Figure 3-12 have the meanings shown below.

▼:	PPG operation start (See *3)
▽:	PPG operation stop (See *4)
▲:	PPGC2:PUF and PPGC3:PUF setting timing (prescaler side, See *5)
△:	PPGC0:PUF and PPGC1:PUF setting timing (PPG output side, see *6)
★:	PPG output side count operation start timing (See *7)
Ql:	PPG2 output (prescaler side) Low pulse width
Qh:	PPG2 output (prescaler side) High pulse width
Pl:	PPG0 output (PPG output side) Low pulse width
Ph:	PPG0 output (PPG output side) High pulse width

The Low width (Ql) and High width (Qh) of the PPG2 output (prescaler side) and the Low width (Pl) and High width (Ph) of the PPG0 output (PPG output side) can be specified as shown below.

T (Prescaler side clock cycle)	= PCLK cycle x 4 (See Note 1.)
Ql (PPG2 output pulse Low width)	= T x (PRLH3 register value x 256 + PRLL3 register value + 1)
Qh (PPG2 output pulse High width)	= T x (PRLH2 register value x 256 + PRLL2 register value + 1)
Qa (PPG output side clock cycle)	= (Ql + Qh)/2 (Ql and Qh average value: See Note 8.)
Pl (PPG0 output pulse Low width)	= Qa x (PRLH1 register value x 256 + PRLL1 register value + 1)
Ph (PPG0 output pulse High width)	= Qa x (PRLH0 register value x 256 + PRLL0 register value + 1)

*1: T (Prescaler side clock cycle) is determined by the count clock cycle selected by the count clock selection registers (PPGC2:PCS1, PCS0 and PPGC3:PCS1, PCS0) of PPG2 and PPG3 (16-bit prescaler side). PCS1 and PCS0 of PPGC2 and PPGC3 must always be set to the same value.

*2: The values of the count clock selection registers (PPGC0:PCS1, PCS0 and PPGC1:PCS1, PCS0) of PPG0 and PPG1 (16-bit PPG output side) are ignored.

*3: When 1111 is written simultaneously to TRG0:PEN00 to PEN03, the PPG start signal is asserted, and the PPG starts operation. If 1111 is not written simultaneously, the count cycle may be shifted.

*4: When "0000" is written simultaneously to TRG0:PEN00 to PEN03, the PPG start signal is negated, and the PPG stops operation. If 0000 is not written simultaneously, the count cycle may be shifted.

*5: Based on the specified PPGC2:INTM=0, PPGC2:PUF is set at the timing indicated by ▲. Based on the specified PPGC3:INTM=0, PPGC3:PUF is set at the timing indicated by ▲. In this case, both flags are set simultaneously.

*6: Based on the specified PPGC0:INTM=0, PPGC0:PUF is set at the timing indicated by △. Based on the specified PPGC1:INTM=0, PPGC1:PUF is set at the timing indicated by △. In this case, both flags are set simultaneously.

*7: PPG0 (PPG output side) starts counting from the position of ★ after Ql has elapsed after the start instruction in ▼. The PPG output side imports the output pulse width from the reload register at the timing in ★. After the ▼ start instruction, if the output pulse width setting on the PPG output side was overwritten before the timing at ★, the setting for the output pulse width before that time is not applied.

*8: The PPG0 (PPG output side) count operation is performed at both the PPG2 (prescaler side) output rising and falling edges. For this reason, in the above equation, Qa is the average value of Ql and Qh. It is recommended that the same value be used for the Low pulse width and High pulse width at the prescaler side. Be careful because, if the values for the Low pulse width and High pulse width at the prescaler side are different, when the pulse count on the PPG output side is odd, the output pulse width on the PPG output side will not match the above equation.

*9: Although it is omitted from the figure, when REVC0:REV00 to REV03=0000, the same output waveform as PPG0 output is obtained for PPG1 output, and the same output waveform as PPG2 output is obtained for PPG3 output.

■ Example of PPG Operation by GATE Signals from Multifunction Timer

This section shows an operation example of PPG start by GATE signals from a multifunctional timer (in the case of 16-bit PPG operation mode using PPG-ch.0 and ch.1). Table 3-9 shows the register settings used in this operation example. Figure 3-13 shows the output waveform of this operation example.

Table 3-9 Register Setting Example of Start from Multifunction Timer (16-bit PPG Operation Mode)

Register name	Bit write value	Setting description	Remarks

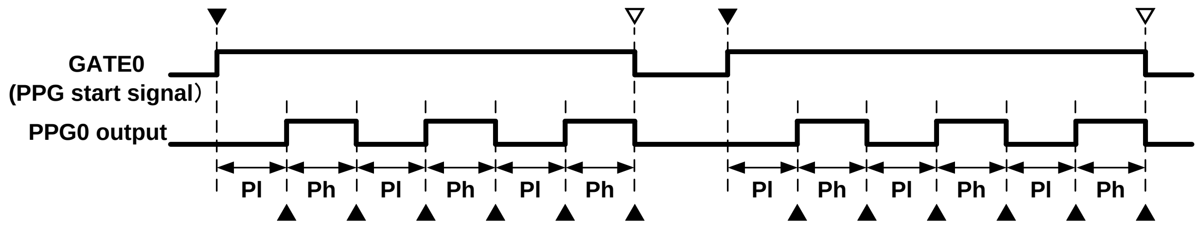
PPGC0	TTRG=0
	MD1,MD0=10
	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		Start by GATE signal from multifunction timer
		16-bit PPG operation mode
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited	

PPGC1	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited	

GATEC0	STRG0=1
	STRG2=X
		Start by GATE signal from multifunction timer
		This is no relation setting for other PPG channel.	

REVC0	REV00=0
	REV01=0
	REV02 to 15=X
		PPG0 is output at positive polarity
		PPG1 is output at positive polarity
		This is no relation setting for other PPG channel.	

Figure 3-13 Example of PPG Operation by GATE Signal from Multifunction Timer



The symbols in Figure 3-13 have the meanings shown below.

▼:	PPG operation start (Asserting GATE0 signal from multifunction timer)
▽:	PPG operation stop (Negating GATE0 signal from multifunction timer)
▲:	PPGC0:PUF and PPGC1:PUF setting timing
Pl:	PPG0 output Low pulse width
Ph:	PPG0 output High pulse width

The Low width (Pl) and High width (Ph) of the PPG0 output can be specified as shown below.

T (Count clock cycle)	= PCLK cycle x 4 (based on selected clock)
Pl (PPG0 output pulse Low width)	= T x (PRLH1 register value x 256 + PRLL1 register value + 1)
Ph (PPG0 output pulse High width)	= T x (PRLH0 register value x 256 + PRLL0 register value + 1)

3.2. Timing Generator Circuit Operations

The timing generator circuit is used for individually starting multiple PPGs at a specified timing. The PPG timer start signal is generated and output when the 8-bit upcounter and compare registers are compared and match.

■ Timing Generator Configuration

The timing generator consists of a prescaler, 8-bit upcounter, and four compare registers. Four PPG start signals are output per timing generator unit.

The block diagrams for timing generators 0, 1, and 2 are shown in Figure 3-14, Figure 3-15, and Figure 3-16, respectively.

Figure 3-14 Timing Generator Circuit 0 Block Diagram

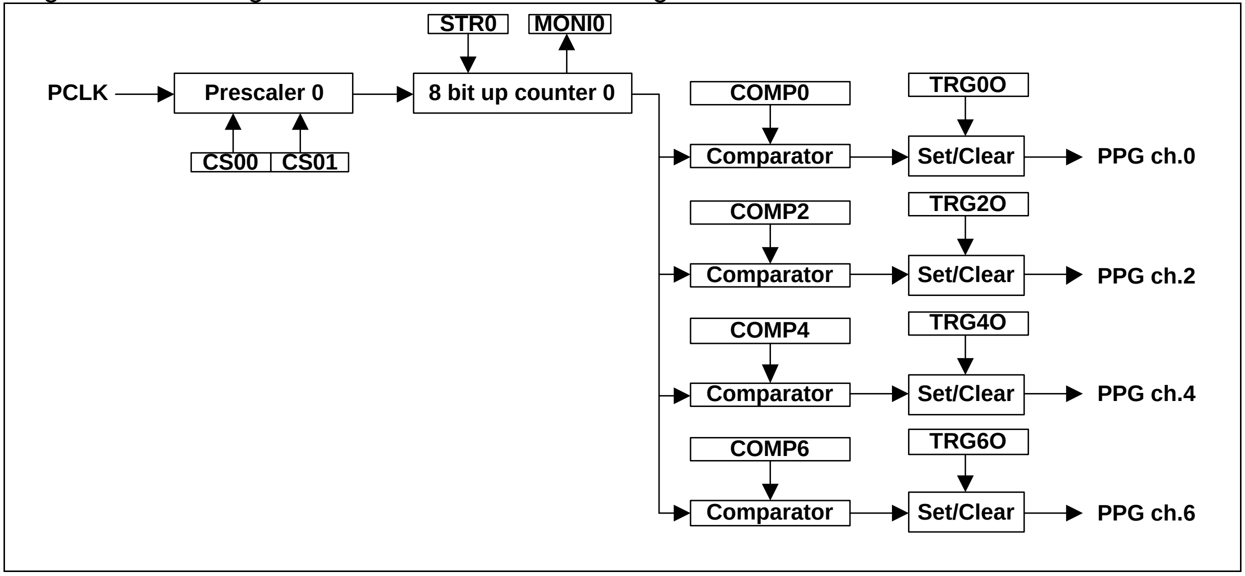


Figure 3-15 Timing Generator Circuit 1 Block Diagram

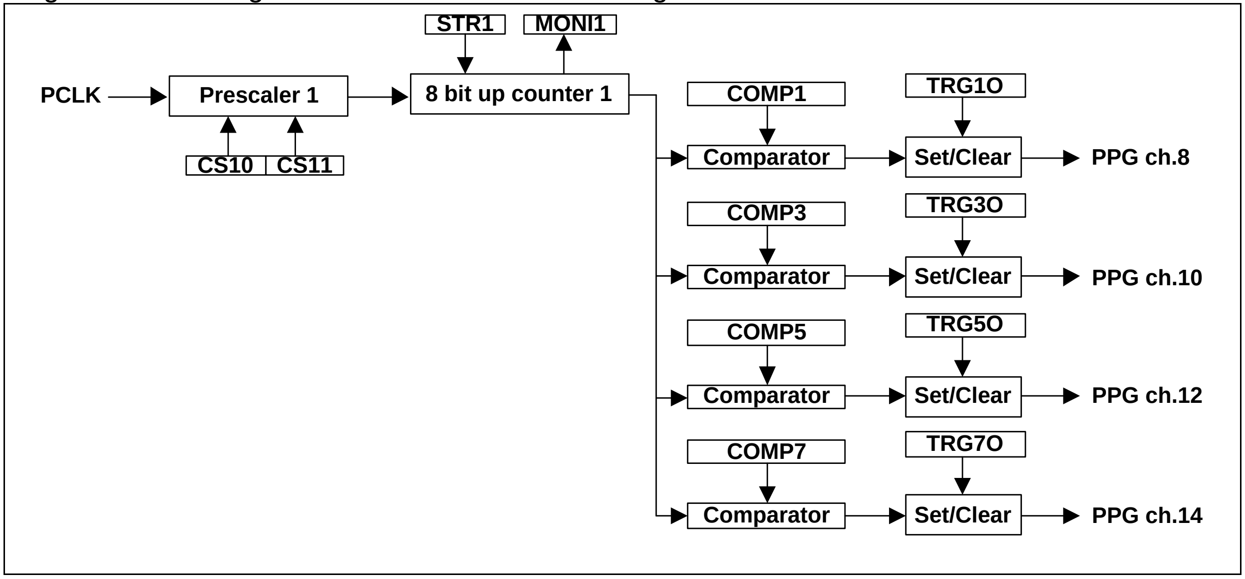
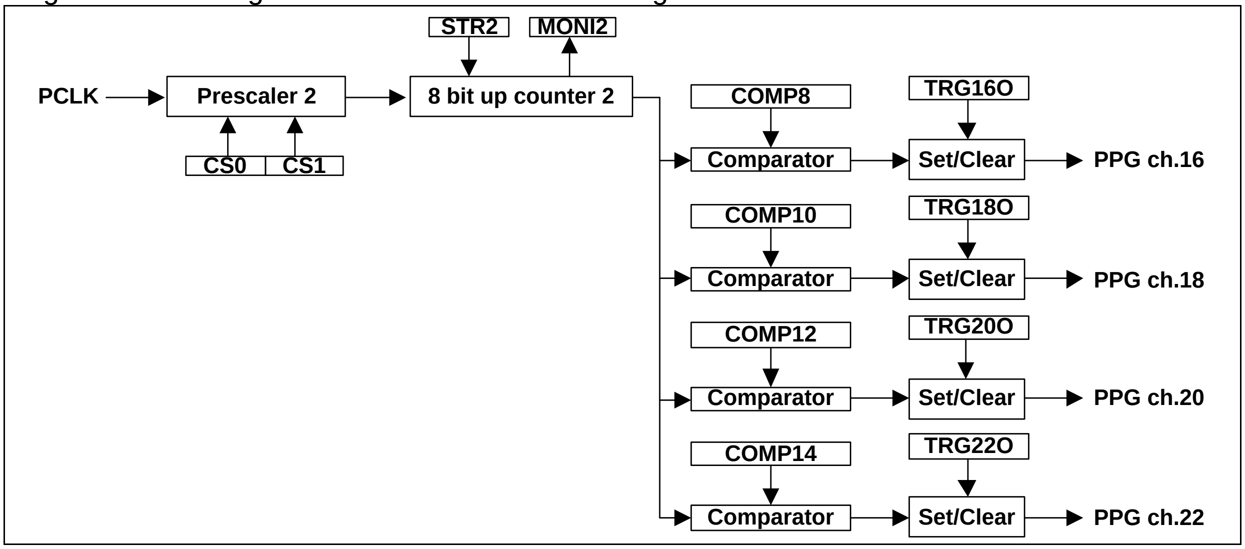


Figure 3-16 Timing Generator Circuit 2 Block Diagram



The blocks making up the timing generator circuit are described below.

- The prescaler can be used to select from four types of 8-bit upcounter operation clocks (PCLK/2, PCLK/8, PCLK/32, and PCLK/64).
- The 8-bit upcounter starts the count operation by writing TTCRx:STR="1".
- The 8-bit upcounter operation status can be read from the TTCRx:MONI bit.
- The four compare registers (COMPx) support the four respective PPG channels and set the start timing for each.
- If the count value of the 8-bit upcounter and compare register value both match, the respective PPG start signal is asserted.
- Writing TTCRx:TRGx0="0" negates the PPG start signal.
- Writing TTCRx:TRGx0="1" does not negate the PPG start signal.
- Once the 8-bit upcounter has counted up to 0xFF, the count operation is stopped.

The PPG start signal that is output from the timing generator is connected to the PPG even channels. The correspondence of the COMP register numbers, TTCRx:TRGxO register numbers, and the PPG channel numbers is different. Pay careful attention to the numbers when setting the registers. Table 3-10 shows the correspondence of the timing generator numbers, COMP register numbers, TTCRx:TRGxO register numbers, and PPG channel numbers.

Table 3-10 Correspondence between Timing Generator Register Numbers and PPG Channels

Timing generator	COMP register	TRGxO register	Connected PPG channel number
Timing generator 0	COMP0	TTCR0:TRG0O	PPG ch.0
	COMP2	TTCR0:TRG2O	PPG ch.2
	COMP4	TTCR0:TRG4O	PPG ch.4
	COMP6	TTCR0:TRG6O	PPG ch.6
Timing generator 1	COMP1	TTCR1:TRG1O	PPG ch.8
	COMP3	TTCR1:TRG3O	PPG ch.10
	COMP5	TTCR1:TRG5O	PPG ch.12
	COMP7	TTCR1:TRG7O	PPG ch.14
Timing generator 2	COMP8	TTCR2:TRG16O	PPG ch.16
	COMP10	TTCR2:TRG18O	PPG ch.18
	COMP12	TTCR2:TRG20O	PPG ch.20
	COMP14	TTCR2:TRG22O	PPG ch.22

■ Timing Generator Configuration

This section shows an operation example for PPG ch.2, ch.4, ch.6, and ch.8 start by timing generator 0. Table 3-11 shows an example of the initial register settings, and Figure 3-17 shows an example of the input/output signal waveform.

Table 3-11 Register Settings when Startinged by Timing Generator (16-bit PPG Operation Mode)

Register name	Bit write value	Setting description	Remarks

PPGC0
PPGC2
PPGC4
PPGC6
	TTRG=1
	MD1,MD0=10
	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		Start by timing generator
		16-bit PPG operation mode
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited	

PPGC1
PPGC3
PPGC5
PPGC7
	PCS1,PCS0=01
	INTM=0
	PUF=0
	PIE=0
		PCLK/4 is selected for count clock
		PUF interrupt flag is set for both Low and High
		PUF flag is initialized
		Generation of interrupts is prohibited	

TTCR0	STR0=0
	MONI0=0
	CS01,CS00=00
	TRG0O=0
	TRG2O=0
	TRG4O=0
	TRG6O=0
		Counter operation does not start (initial setting)
		Write value is ignored
		PCLK/2 is selected for 8-bit upcounter clock
		Start signal initialization for PPG ch.0
		Start signal initialization for PPG ch.2
		Start signal initialization for PPG ch.4
		Start signal initialization for PPG ch.6	

REVC0	REV00-PPG07 REV08~15=X	PPG0 to PPG7 are output at positive polarity This is no relation setting for other PPG channel.	
COMP0	COMP0=0x40	Specifies output start timing of PPG0/PPG1	
COMP2	COMP2=0x80	Specifies output start timing of PPG2/PPG3	
COMP4	COMP4=0xC0	Specifies output start timing of PPG4/PPG5	
COMP6	COMP6=0xF0	Specifies output start timing of PPG6/PPG7	

The controlled content from the CPU and PPG operation at the timing of ▼ 1 to ▼6 shown in Figure 3-17 are described in detail below.

▼1 timing:
As shown in Table 3-11, the PPG and timing generator initial settings are performed. Because the initial settings cannot be performed during operation of the 8-bit upcounter, the initial settings are performed after reading the TTCR0:MONI0 register to confirm that the counter has stopped operation. When PPGC0, 2, 4, 6:TTRG=1 is specified, start from the timing generator is selected. The clock used by the 8-bit upcounter is selected by the TTCR0:CS01 and CS00 registers. The start timing of each PPG is specified by the COMP0, 2, 4, and 6 registers. If the PPG start signal is not asserted, COMPx=0x00 is specified.

▼2 timing:
Writing of TTCR0:STR0="1" is performed. The 8-bit upcounter starts the count operation. The TTCR0:MONI0 register is used to read the count operation status of the 8-bit upcounter. During the count operation, 1 is read. While the count operation is stopped, 0 is read.

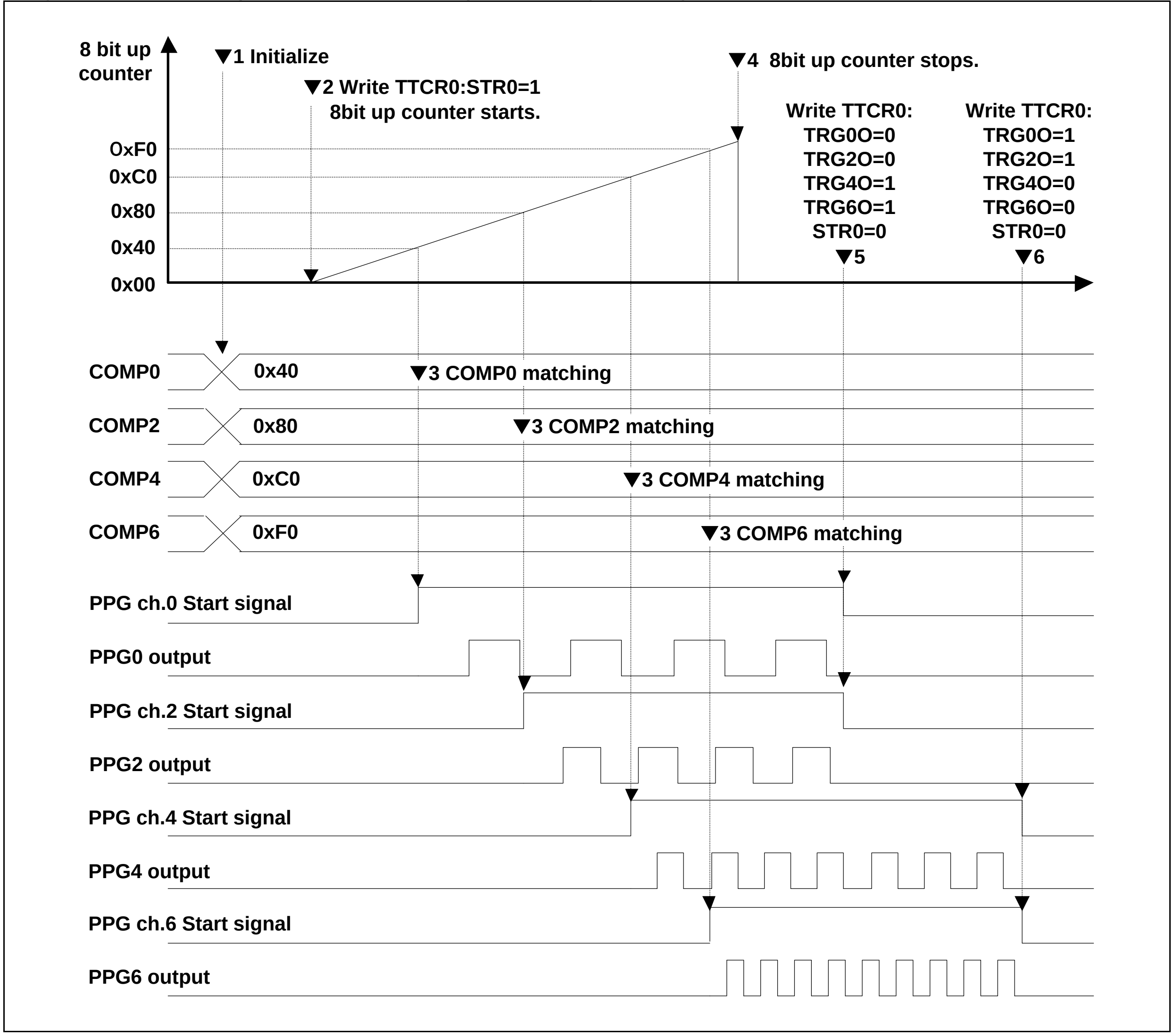
▼3 timing:
When the counter value of the 8-bit upcounter matches the value for COMP0, COMP2, COMP4, or COMP6, the PPG start signal from the timing generator for the respective channel is asserted. Each PPG starts output when the respective start signal is asserted.

▼4 timing:
The 8-bit upcounter stops once it has counted up to 0xFF.

▼5 timing:
Writing is performed for TRG0O=TRG2O=0, TRG4O=TRG6O=1, and STR0=0. Instructions are issued to PPG0 and PPG2 to stop operation and issued to PPG4 and PPG6 to continue operation. STR0=0 is written so that the 8-bit upcounter does not restart.

▼6 timing:
Writing is performed for TRG0O=TRG2O=1, TRG4O=TRG6O=0, and STR0=0, and a stop instruction is issued to PPG4 and PPG6.

Figure 3-17 Example of PPG Start Operation by Timing Generator 0



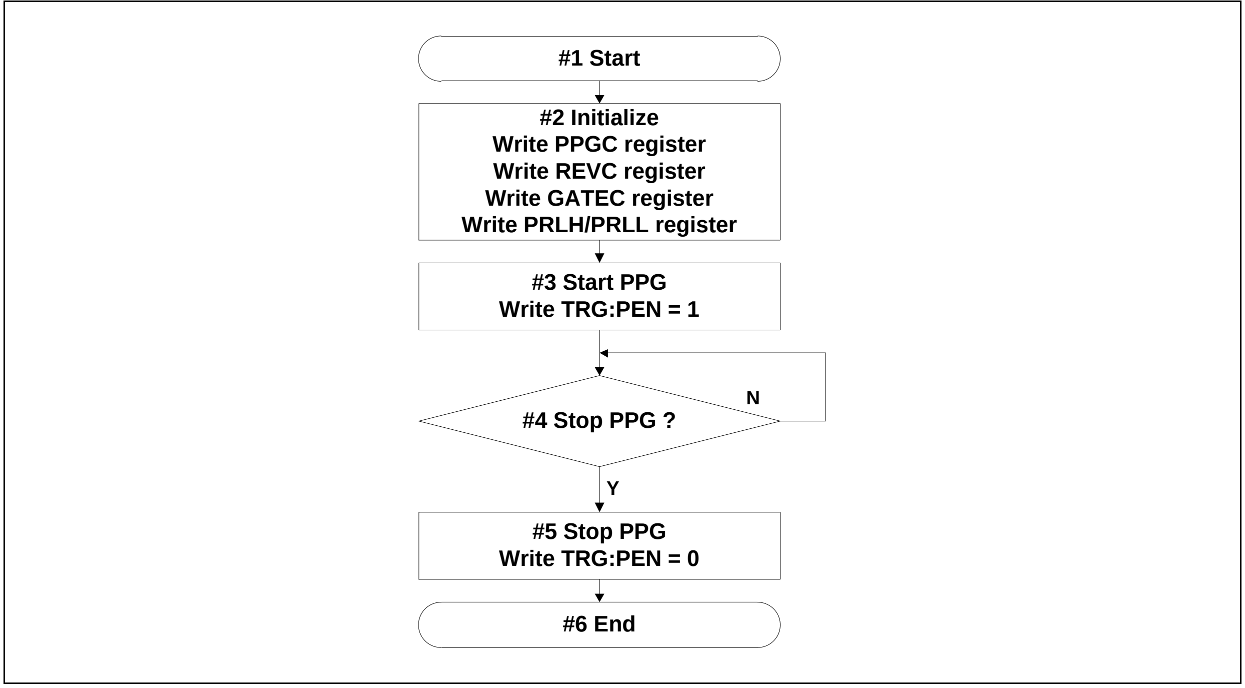
4. PPG Setup Procedure Example

This section explains a setting procedure example of PPG.

■ Example of PPG Start by Writing to PPG Start Register

Figure 4-1 shows an example of the setting procedure when PPG start by writing directly to the PPG start register (TRG:PEN) is selected. The numbers in the figure correspond to the numbers in the explanation below.

Figure 4-1 PPG Start by Direct Writing to PPG Start Register



#1: The setting procedure of this example is started.

#2: Each register is initialized. The settings PPGC:TTRG=0 and GATEC:STRG=0 are made to select PPG start by direct writing to the PPG start register (TRG:PEN). For the initial setting values of each register, see Table 3-5, Table 3-6, Table 3-7, and Table 3-8.

#3: 1 is written to the PPG start register (TRG:PEN), and an instruction is issued to start PPG output.

#4: After PPG output is started, output continues until a command is issued to stop the start operation. When the start operation is stopped, proceed to #5.

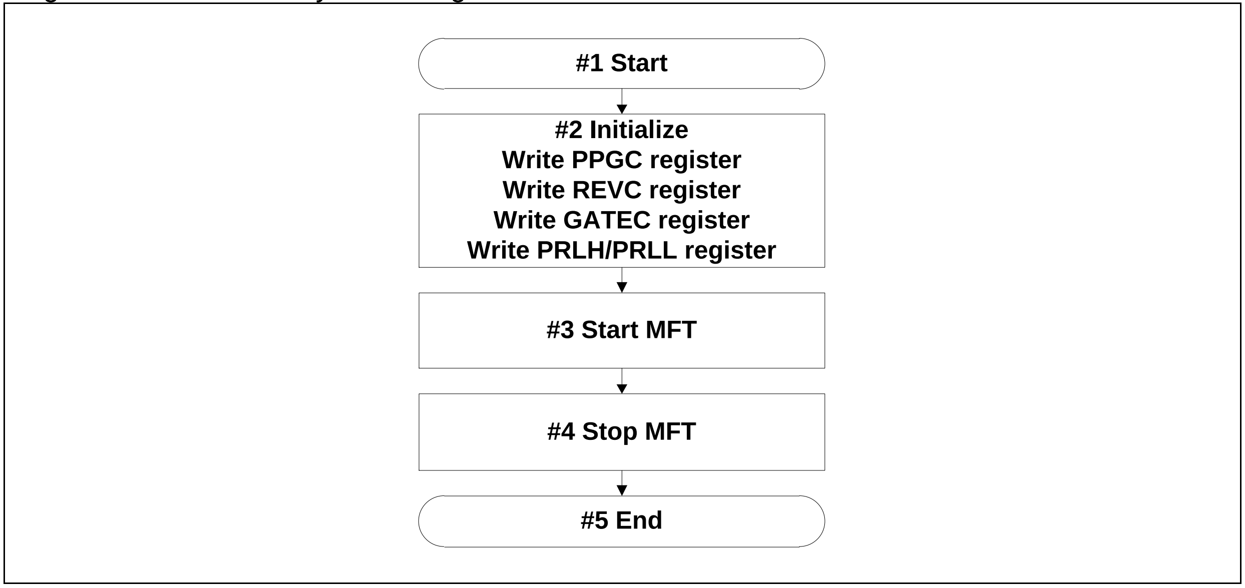
#5: 0 is written to the PPG start register (TRG:PEN), and an instruction is issued to stop PPG output.

#6: The setting procedure of this example is ended.

■ Example of PPG Start by GATE Signal from Multifunction Timer

Figure 4-2 shows an example of the setting procedure when PPG start by GATE signal from multifunction timer (MFT) is selected. The numbers in the figure correspond to the numbers in the explanation below.

Figure 4-2 PPG Start by GATE Signal from Multifunction Timer



#1: The setting procedure of this example is started.

#2: Each register is initialized. The settings PPGC:TTRG=0 and GATEC:STRG=1 are made to select start by GATE signal from multifunction timer. For the initial settings of each register, see Table 3-9.

#3: Operation of the multifunction timer is started. Control is performed so that PPG output is started and stopped based on asserting and negating of the GATE signals supplied from the multifunction timer.

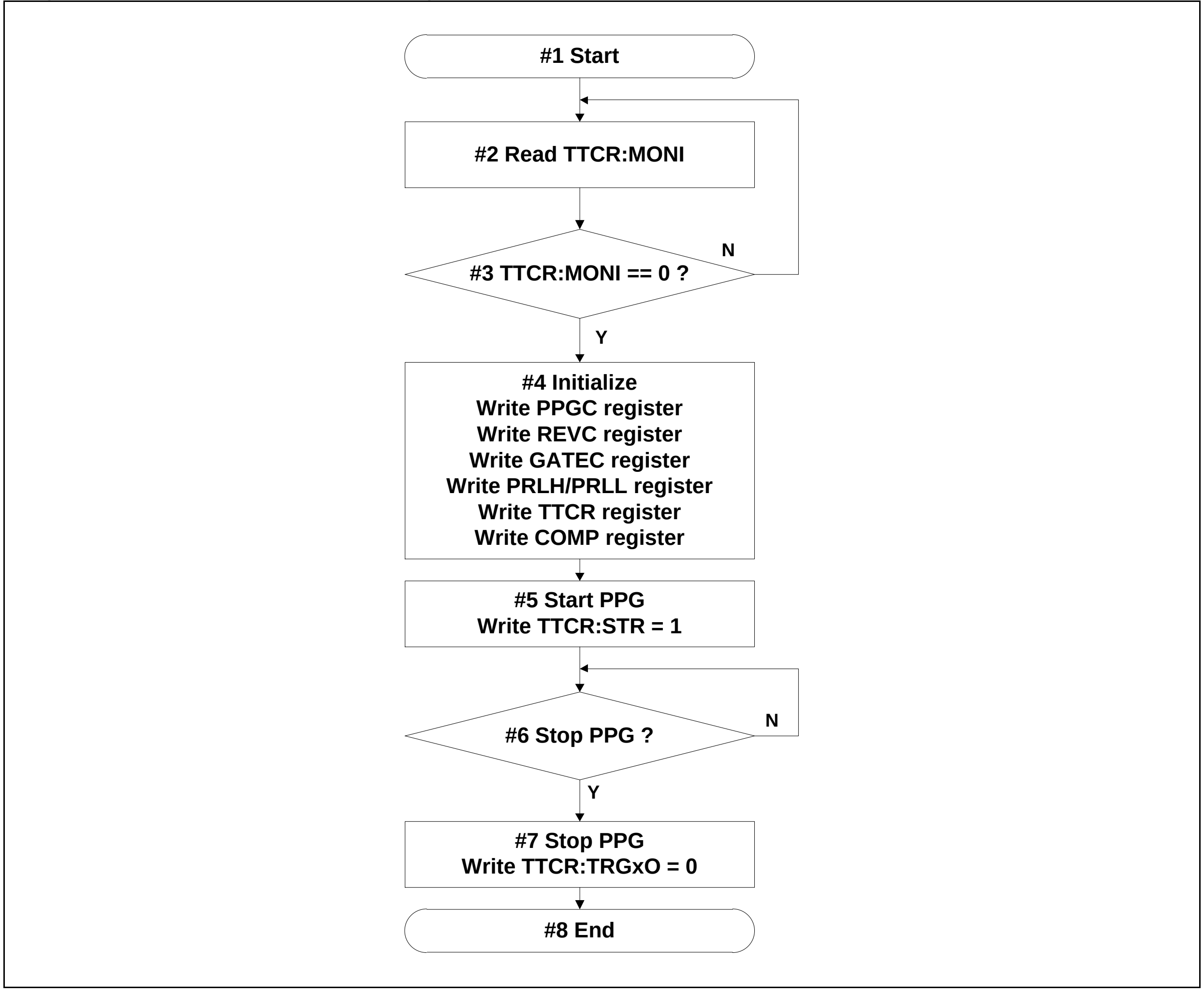
#4: Operation of the multifunction timer is stopped. When operation is stopped, perform control at the multifunction timer side so that the GATE signal ends in the negated state. When the GATE signal is in the negated state, the PPG stops any further output.

#5: The setting procedure of this example is ended.

■ Example of PPG Start by Timing Generator

Figure 4-3 shows an example of the setting procedure when PPG start by timing generator is selected. The numbers in the figure correspond to the numbers in the explanation below.

Figure 4-3 PPG Start from Timing Generator



#1: The setting procedure of this example is started.

#2, #3: If performing PPG start by a timing generator, initialization cannot be performed during the count operation of the 8-bit upcounter, and so reading of TTCR:MONI is performed to confirm that the count operation has stopped before proceeding to #4.

#4: Each register is initialized. The setting PPGC:TTRG=1 is made to select start by timing generator. For the initial settings of each register, see Table 3-11.

#5: 1 is written to TTCR:STR, and an instruction is issued to start the count of the 8-bit upcounter. When the setting value of the COMP register matches the value of the 8-bit upcounter, the PPG start signal is asserted, and PPG output is started.

#6: After PPG is started, output continues until a command is issued to stop the start operation. When the start operation is stopped, proceed to #7.

#7: "0" is written to the TTCR:TRGxO register, and an instruction is issued to stop PPG output. The corresponding PPG start signal is negated, and PPG output is stopped.

#8: The setting procedure of this example is ended.

5. PPG Registers

This section explains the registers of PPG.

Table 5-1 lists the PPG Registers.

Table 5-1 PPG Register list

Abbreviation	Register name	Reference
TTCR0	Timing Generator PPG Start Trigger Control Register 0	5.1
TTCR1	Timing Generator PPG Start Trigger Control Register 1	5.2
TTCR2	Timing Generator PPG Start Trigger Control Register 2	5.3
COMP0	Timing Generator PPG Compare Register 0	5.4
COMP1	Timing Generator PPG Compare Register 1	
COMP2	Timing Generator PPG Compare Register 2	
COMP3	Timing Generator PPG Compare Register 3	
COMP4	Timing Generator PPG Compare Register 4	
COMP5	Timing Generator PPG Compare Register 5	
COMP6	Timing Generator PPG Compare Register 6	
COMP7	Timing Generator PPG Compare Register 7	
COMP8	Timing Generator PPG Compare Register 8	
COMP10	Timing Generator PPG Compare Register 10	
COMP12	Timing Generator PPG Compare Register 12	
COMP14	Timing Generator PPG Compare Register 14	
TRG0	PPG Start Register 0	5.5
TRG1	PPG Start Register 1	5.6
REVC0	Output Reverse Register 0	5.7
REVC1	Output Reverse Register 1	5.8
PPGC0	PPG Operation Mode Control Register 0	5.9
PPGC1	PPG Operation Mode Control Register 1	
PPGC2	PPG Operation Mode Control Register 2	
PPGC3	PPG Operation Mode Control Register 3	
PPGC4	PPG Operation Mode Control Register 4	
PPGC5	PPG Operation Mode Control Register 5	
PPGC6	PPG Operation Mode Control Register 6	
PPGC7	PPG Operation Mode Control Register 7	
PPGC8	PPG Operation Mode Control Register 8	
PPGC9	PPG Operation Mode Control Register 9	
PPGC10	PPG Operation Mode Control Register 10	
PPGC11	PPG Operation Mode Control Register 11	

Abbreviation	Register name	Reference
PPGC12	PPG Operation Mode Control Register 12	5.9
PPGC13	PPG Operation Mode Control Register 13	
PPGC14	PPG Operation Mode Control Register 14	
PPGC15	PPG Operation Mode Control Register 15	
PPGC16	PPG Operation Mode Control Register 16	
PPGC17	PPG Operation Mode Control Register 17	
PPGC18	PPG Operation Mode Control Register 18	
PPGC19	PPG Operation Mode Control Register 19	
PPGC20	PPG Operation Mode Control Register 20	
PPGC21	PPG Operation Mode Control Register 21	
PPGC22	PPG Operation Mode Control Register 22	
PPGC23	PPG Operation Mode Control Register 23	
PRLH0	PPG Reload Register H0	5.10
PRLL0	PPG Reload Register L0	
PRLH1	PPG Reload Register H1	
PRLL1	PPG Reload Register L1	
PRLH2	PPG Reload Register H2	
PRLL2	PPG Reload Register L2	
PRLH3	PPG Reload Register H3	
PRLL3	PPG Reload Register L3	
PRLH4	PPG Reload Register H4	
PRLL4	PPG Reload Register L4	
PRLH5	PPG Reload Register H5	
PRLL5	PPG Reload Register L5	
PRLH6	PPG Reload Register H6	
PRLL6	PPG Reload Register L6	
PRLH7	PPG Reload Register H7	
PRLL7	PPG Reload Register L7	
PRLH8	PPG Reload Register H8	
PRLL8	PPG Reload Register L8	
PRLH9	PPG Reload Register H9	
PRLL9	PPG Reload Register L9	
PRLH10	PPG Reload Register H10	
PRLL10	PPG Reload Register L10	
PRLH11	PPG Reload Register H11	
PRLL11	PPG Reload Register L11	
PRLH12	PPG Reload Register H12	

Abbreviation	Register name	Reference
PRLL12	PPG Reload Register L12	5.10
PRLH13	PPG Reload Register H13	
PRLL13	PPG Reload Register L13	
PRLH14	PPG Reload Register H14	
PRLL14	PPG Reload Register L14	
PRLH15	PPG Reload Register H15	
PRLL15	PPG Reload Register L15	
PRLH16	PPG Reload Register H16	
PRLL16	PPG Reload Register L16	
PRLH17	PPG Reload Register H17	
PRLL17	PPG Reload Register L17	
PRLH18	PPG Reload Register H18	
PRLL18	PPG Reload Register L18	
PRLH19	PPG Reload Register H19	
PRLL19	PPG Reload Register L19	
PRLH20	PPG Reload Register H20	
PRLL20	PPG Reload Register L20	
PRLH21	PPG Reload Register H21	
PRLL21	PPG Reload Register L21	
PRLH22	PPG Reload Register H22	
PRLL22	PPG Reload Register L22	
PRLH23	PPG Reload Register H23	
PRLL23	PPG Reload Register L23	
GATEC0	Gate Function Control Register 0	5.11
GATEC4	Gate Function Control Register 4	
GATEC8	Gate Function Control Register 8	
GATEC12	Gate Function Control Register 12	
GATEC16	Gate Function Control Register 16	
GATEC20	Gate Function Control Register 20	

5.1. Timing Generator PPG Start Trigger Control Register 0 (TTCR0)

The TTCR0 Register controls Timing Generator 0.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	TRG6O	TRG4O	TRG2O	TRG0O	CS01	CS00	MONI0	STR0
Attribute	W	W	W	W	R/W	R/W	R	W
Initial value	-	-	-	-	0	0	0	-

■ Register functions

[bit15:12] TRG6O, TRG4O, TRG2O, TRG0O: PPG trigger stop bits

These bits can be used to negate the PPG start signal generated by the timing generator 0.

Process	Function
Reading	"1" is always read.
Writing 0	Disables the PPG start trigger. (LOW output)
Writing 1	No effect on the operation

[bit11:10] CS01, CS00: 8-bit UP counter clock select bits

These bits set an operation clock of the 8-bit UP counter.

bit11	bit10	Function
0	0	PCLK/2 [Initial value]
0	1	PCLK/8
1	0	PCLK/32
1	1	PCLK/64

[bit9] MONI0: 8-bit UP counter operation state monitor bit

This bit indicates the operation state of the 8-bit UP counter.

Process	Function
Reading 0	The 8-bit UP counter is stopped. [Initial value]
Reading 1	The 8-bit UP counter is operating.
Writing	No effect on the opearation

[bit8] STR0: 8-bit UP counter operation enable bit

This bit issues an instruction to start operation of the 8-bit UP counter.

Process	Function
Reading	"0" is always read.
Writing 0	No effect on the operation
Writing 1	Starts the 8-bit UP counter.

<Notes>

- In certain cases, the number of the TRGxO bit and the number of the PPG channel being controlled may be different. See Table 3-10.
- If the PPG start signal is asserted based on a match with the compare register and writing of TRGxO=0 occurs at the same time, the PPG start signal negate is given priority.
- If writing of TRGxO=0 is performed before the PPG start signal is asserted based on a match with the compare register, there is no effect on operation.
- After counting is started, the 8-bit upcounter stops once it has counted up to 0xFF. Once the count has started, to start the count again from 0x00, issue a count start instruction after first confirming that the count operation has stopped using the MONI0 bit.
- Change the value of the CS01 and CS00 bits is prohibited during operation of the 8-bit upcounter.

5.2. Timing Generator PPG Start Trigger Control Register 1 (TTCR1)

The TTCR1 Register controls a start of Timing Generator1.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	TRG7O	TRG5O	TRG3O	TRG1O	CS11	CS10	MONI1	STR1
Attribute	W	W	W	W	R/W	R/W	R	W
Initial value	-	-	-	-	0	0	0	-

■ Register functions

[bit15:12] TRG7O, TRG5O, TRG3O, TRG1O: PPG trigger stop bits

These bits are used to negate the PPG start signal generated by the timing generator.

Process	Function
Reading	"1" is always read.
Writing 0	PPG start signal from timing generator is negated, and PPG output is stopped.
Writing 1	No effect on the operation

[bit11:10] CS11, CS10: 8-bit UP counter clock select bits

These bits set an operation clock of 8-bit UP counter.

bit11	bit10	Function
0	0	PCLK/2 [Initial value]
0	1	PCLK/8
1	0	PCLK/32
1	1	PCLK/64

[bit9] MONI1: 8-bit UP counter operation state monitor bit

This bit indicates the operation state of 8-bit UP counter.

Process	Function
Reading 0	The 8-bit UP counter is stopped. [Initial value]
Reading 1	The 8-bit UP counter is operating.
Writing	No effect on the operation

[bit8] STR1: 8-bit UP counter operation enable bit

This bit enables the operation of 8-bit UP counter.

Process	Function
Reading	"0" is always read.
Writing 0	No effect on the operation
Writing 1	Starts the 8-bit UP counter.

<Notes>

- See notes in Timing generator PPG start trigger control register 0 (TTCR0). These notes also apply to the TTCR1 register in the same way.

5.3. Timing Generator PPG Start Trigger Control Register 2 (TTCR2)

The TTCR2 Register controls a start of Timing generator2.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	TRG22O	TRG20O	TRG18O	TRG16O	CS21	CS20	MONI2	STR2
Attribute	W	W	W	W	R/W	R/W	R	W
Initial value	-	-	-	-	0	0	0	-

■ Register functions

[bit15:12] TRG22O, TRG20O, TRG18O, TRG16O: PPG trigger stop bits

These bits are used to negate the PPG start signal generated by the timing generator.

Process	Function
Reading	"1" is always read.
Writing	PPG start signal from timing generator is negated, and PPG output is stopped.
Writing	No effect on the operation

[bit11:10] CS21, CS20: 8-bit UP counter clock select bits

These bits set an operation clock of 8-bit UP counter.

bit11	bit10	Function
0	0	PCLK/2 [Initial value]
0	1	PCLK/8
1	0	PCLK/32
1	1	PCLK/64

[bit9] MONI2: 8-bit UP counter operation state monitor bit

This bit indicates the operation state of 8-bit UP counter.

Process	Function
Reading 0	The 8-bit UP counter is stopped. [Initial value]
Reading 1	The 8-bit UP counter is operating.
Writing	No effect on the operation

[bit8] STR2: 8-bit Counter Operation Enable bit

This bit issues an instruction to start operation of 8-bit UP counter.

Process	Function
Reading	"0" is always read.
Writing 0	No effect on the operation
Writing 1	Starts the 8-bit UP counter.

<Notes>

- See notes in Timing generator PPG start trigger control register 0 (TTCR0). These notes also apply to the TTCR2 register in the same way.

5.4. Timing Generator PPG Compare Register (COMPn n=0,2,4,6,1,3,5,7,8,10,12,14)

The COMP Register sets a Compare value of the Timing Generator.

■ Register configuration

bit	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0
Field	COMP							
Attribute	R/W							
Initial value	0x00							

■ Register functions

[bit15:8/bit7:0] COMP: Compare bits

These bits can be used to set the PPG compare register value when started by a timing generator.

<Notes>

- This register is an 8-bit compare register, and one register is provided for each PPG start signal. In certain cases, the number of this register and the number of the PPG channel being controlled may be different. See Table 3-10.
- When this register value matches the value of the 8-bit upcounter, a start signal is asserted for the corresponding PPG.
- When this register value is 0x00, no comparison or matching is made with the 8-bit upcounter value, and the PPG start signal is not asserted.
- Writing of this register is prohibited during operation of the 8-bit upcounter.

5.5. PPG Start Register 0 (TRG0)

The TRG0 Register is a PPG start register that directly starts PPG0 to PPG15.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	PEN15	PEN14	PEN13	PEN12	PEN11	PEN10	PEN09	PEN08
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	PEN07	PEN06	PEN05	PEN04	PEN03	PEN02	PEN01	PEN00
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:0] PEN15 to PEN00: PPG Start Trigger bits

This is the PPG start register for directly starting each PPG channel. When 1 is written, the PPG start signal is asserted, and the PPG is started. When "0" is written, the PPG start signal is negated, and the PPG is stopped. Simultaneous PPG start and simultaneous PPG stop are possible by simultaneous writing to multiple channels.

Value	Function
0	PPG start signal is negated, and PPG operation is stopped. [Initial value]
1	PPG start signal is asserted, and PPG operation is started.

<Notes>

- The PEN bit number n (n=0, 1, 2, ..., 15) corresponds to the channel number of each PPG.
- If PPG start by PEN register is selected by the specified values for the PPGC:TTRG register and GATEC:STRG register, PPG can be started and stopped from the PEN register.
- If PPG start by PEN register is not selected, the PEN register value is ignored. See Table 3-2.
- For operation modes that use multiple PPG channels (8+8-bit, 16-bit, and 16+16-bit) and the start method of writing the PPG start register (TRG:PEN), 1 is written simultaneously to the TRG:PEN registers of all channels being used to start the PPG. Also, 0 is written simultaneously to the TRG:PEN registers to stop the PPG. The count cycle may be shifted if values are not written simultaneously.

5.6. PPG Start Register 1 (TRG1)

The TRG1 Register is a PPG start regster that directly starts PPG16 to PPG23.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	Reserved							
Attribute	-							
Initial value	-							

bit	7	6	5	4	3	2	1	0
Field	PEN23	PEN22	PEN21	PEN20	PEN19	PEN18	PEN17	PEN16
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit7:0] PEN23 to PEN16: PPG Start Trigger bits

This is the PPG start register for directly starting each PPG channel. When 1 is written, the PPG start signal is asserted, and the PPG is started. When 0 is written, the PPG start signal is negated, and the PPG is stopped. Simultaneous PPG start and simultaneous PPG stop are possible by simultaneous writing to multiple channels.

Value	Function
0	PPG start signal is negated, and PPG operation is stopped. [Initial value]
1	PPG start signal is asserted, and PPG operation is started.

<Notes>

- The PEN bit number n (n=16, 17, ..., 23) corresponds to the channel number of each PPG.
  See the notes in "PPG start register 0 (TRG0)". These notes also apply to the TRG1 register in the same way.

5.7. Output Reverse Register 0 (REVC0)

The REVC0 Register sets an output polarity of PPG0 to PPG15 output value.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	REV15	REV14	REV13	REV12	REV11	REV10	REV09	REV08
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

bit	7	6	5	4	3	2	1	0
Field	REV07	REV06	REV05	REV04	REV03	REV02	REV01	REV00
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit15:0] REV15 to REV00: PPG Output Reverse Enable bits

These bits set the polarity of each PPG channel output signal.

Value	Function
0	Normal output (LOW output when PPG is not operating) [Initial value]
1	Reverse the output. (HIGH output when PPG is stopped)

<Notes>

- The REV bit number n (n=0, 1, 2, ..., 15) corresponds to the channel number of each PPG.
- The connection diagrams of Figure 3-1, Figure 3-3, Figure 3-4, and Figure 3-5 show configurations where the PPG output is reversed by the REV register value directly. For this reason, when REV=1, the followings are performed.
  - The output level before operation start of the PPG output and the output level after operation stop are reversed to the High level.
  - The Low-High of the output pulse is reversed, and the relationship of the Low width setting and High width setting of the reload resistor is reversed.
  - PUF is set when PPGC:INTM=1 and when the Low pulse ends.
  - In 8+8-bit PPG operation mode and 16+16-bit PPG operation mode, the operation clock supplied to the PPG output side from the prescaler side is reversed.

5.8. Output Reverse Register 1 (REVC1)

The REVC1 Register sets an output polarity of PPG16 to PPG23 output value.

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	Reserved							
Attribute	-							
Initial value	-							

bit	7	6	5	4	3	2	1	0
Field	REV23	REV22	REV21	REV20	REV19	REV18	REV17	REV16
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

■ Register functions

[bit7:0] REV23 to REV16: PPG Output Reverse Enable bits

These bits set the polarity of each PPG channel output signal.

Value	Function
0	Normal output (LOW output when PPG is not operating) [Initial value]
1	Invert the output. (HIGH output when PPG is stopped)

<Notes>

- The REV bit number n (n=16, 17, ..., 23) corresponds to the channel number of each PPG. The (n=16, 17, 18, ..., 23) of the REVn register corresponds to the channel number of each PPG.
- See the notes in the "Output reversal register 0 (REVC0)". These notes also apply to the REV1 register in the same way.

5.9. PPG Operation Mode Control Register (PPGC0 to 23)

The PPGC Register sets an interrupt, an operation mode, and the prescaler data.

■ PPGC Register configuration list

bit 15 ... 8	7 ... 0	Initial value	Attribute	Corresponding PPG
PPGC0	PPGC1	0x0000	R/W	PPG0, PPG1
PPGC2	PPGC3	0x0000	R/W	PPG2, PPG3
PPGC4	PPGC5	0x0000	R/W	PPG4, PPG5
PPGC6	PPGC7	0x0000	R/W	PPG6, PPG7
PPGC8	PPGC9	0x0000	R/W	PPG8, PPG9
PPGC10	PPGC11	0x0000	R/W	PPG10, PPG11
PPGC12	PPGC13	0x0000	R/W	PPG12, PPG13
PPGC14	PPGC15	0x0000	R/W	PPG14, PPG15
PPGC16	PPGC17	0x0000	R/W	PPG17, PPG16
PPGC18	PPGC19	0x0000	R/W	PPG19, PPG18
PPGC20	PPGC21	0x0000	R/W	PPG21, PPG20
PPGC22	PPGC23	0x0000	R/W	PPG23, PPG22

<Notes>

- The PPGC register number n (n=0, 1, 2, ..., 23) corresponds to the channel number of the PPG being controlled.
- This register is located on the upper side (bit[15:8]) when n is even.
- This register is located on the lower side (bit[7:0]) when n is odd.
- The register configuration is different for the upper side and lower side. There is a control bit that exists only on the even channel side.

■ PPGCn register configuration (when n is even)

bit	15	14	13	12	11	10	9	8
Field	PIE	PUF	INTM	PCS1	PCS0	MD1	MD0	TTRG

■ PPGCn register configuration (when n is odd)

bit	7	6	5	4	3	2	1	0
Field	PIE	PUF	INTM	PCS1	PCS0	Reserved		

■ Register functions

[bit15/bit7] PIE: PPG Interrupt Enable bit

This bit is used to select enable/disable for PPG interrupts. When interrupt enable is set, PUF can be used to assert the interrupt signal. As shown in the connection diagrams in Figure 3-1, Figure 3-3, Figure 3-4, and Figure 3-5, the interrupt output signal is the logical AND signal of the PUF value and PIE value.

Value	Function
0	Disables an interrupt. [Initial value]
1	Enables an interrupt.

[bit14/bit6] PUF: PPG Counter Underflow bit

This bit is used to notify the CPU when a PPG output pulse change event occurs. It is set to 1 by the underflow of the PPG counter. It can be cleared by setting to 0 from the CPU.

Process	Function
Reading 0	No underflow of PPG Counter has been detected. [Initial value]
Reading 1	An underflow of PPG Counter has been detected.
Writing 0	Clears the PUF flag.
Writing 1	No effect on the operation

The PPG counter changes the output pulse when the down-count for the specified pulse width value ends and an underflow occurs. PUF is set to 1 by this counter underflow. PUF is an event register that notifies the CPU when an output pulse change event occurs. The selection below is performed by setting PPGCn:INTM.

- When PPGCn:INTM=0, the settings are made by the underflow when the Low pulse width count is ended and by the underflow when the High pulse width count is ended.
- When PPGCn:INTM=1, the setting is made by the underflow when the High pulse width count is ended.

If the PPG channels are connected and operated at a length of 16 bits, the PUFs for both the even and odd channels are set simultaneously.

The PUF that is set from the PPG can be cleared by writing 0 from the CPU to the PUF. Once a PUF is set, it is not cleared from the PPG. To enable the CPU to recognize an output pulse change event, the PUF must be cleared from the CPU whenever the PUF is set.

The PUF is cleared by writing 0. As a result, if writing access to the PPGCn register is performed without clearing the PUF, write 1 to the PUF. When reading during read modify write access, 1 is read regardless of the PUF value.

[bit13/bit5] INTM: Interrupt Mode Select bit

This bit sets the interrupt mode.

Value	Function
0	The PUF bit is set to "1" at the underflow when the Low pulse width count is ended and at underflow when the High pulse width count is ended. [Initial value]
1	The PUF bit is set to 1 at the underflow when the High pulse width count is ended.

[bit12:11/bit4:3] PCS1, PCS0: PPG DOWN Counter Operation Clock Select bits

These bits set an operation clock of PPG's DOWN counter.

See the Count clock selection.

bit12	bit11	Function
0	0	PCLK [Initial value]
0	1	PCLK/4
1	0	PCLK/16
1	1	PCLK/64

[bit10:9] MD1, MD0: PPG Operation Mode Set bits

These bits set the PPG operation mode.

These bits are found in even channels (n=0, 2, 4, ..., 22) only. By setting these bits, the operation modes for both the PPG even channels (n) and odd channels (n+1) are specified.

When 16+16-bit PPG operation mode is set, 4 channels are used. The PPGCm:MD1,MD0 = PPGCm+2:MD1,MD = 11 (m=0, 4, 8, 12, 16, 20) setting is performed.

See Selecting the PPG operation mode.

bit10	bit9	Function
0	0	Both even channels (n) and odd channels (n+1) are set to 8-bit PPG operation mode. [Initial value]
0	1	This sets to 8+8-bit PPG operation mode. Even channels (n) are set to the 8-bit PPG output side. Odd channels (n+1) are set to the 8-bit prescaler side.
1	0	The even (n) and odd (n+1) channels are connected and set to 16-bit PPG operation mode.
1	1	This sets to 16+16-bit PPG operation mode. This connects PPGm and PPGm+1 and sets to the 16-bit PPG output side. This connects PPGm+2 and PPGm+3 and sets to the 16-bit prescaler side.

[bit8] TTRG: PPG start trigger select bit

This bit is used to select the PPG start signal. This bit is found in even channels only. The PPG start signals for both even and odd channels are selected by a combination of this bit setting and the GATECx:STRGn register setting. The available start signals vary depending on the selected PPG operation mode.

See Selecting the PPG start method.

Value	Function
0	Selects start from PPG start register (TRG:PEN) or start by GATE signal from multifunction timer. [Initial value]
1	Selects start signal from timing generator.

[bit2:0] Reserved : Reserved bits

000 is read from these bits.

Set these bits to 000 when writing.

5.10. PPG Reload Register (PRLHn, PRLLn n=0 to 23)

The PRLHn and PRLLn registers set the PPG output pulse width.

■ PRLHn/PRLLn Register configuration list

bit 15 8	7 0	Initial value	Attribute	Corresponding PPG
PRLH0	PRLL0	0xXXXX	R/W	PPG0
PRLH1	PRLL1	0xXXXX	R/W	PPG1
PRLH2	PRLL2	0xXXXX	R/W	PPG2
PRLH3	PRLL3	0xXXXX	R/W	PPG3
PRLH4	PRLL4	0xXXXX	R/W	PPG4
PRLH5	PRLL5	0xXXXX	R/W	PPG5
PRLH6	PRLL6	0xXXXX	R/W	PPG6
PRLH7	PRLL7	0xXXXX	R/W	PPG7
PRLH8	PRLL8	0xXXXX	R/W	PPG8
PRLH9	PRLL9	0xXXXX	R/W	PPG9
PRLH10	PRLL10	0xXXXX	R/W	PPG10
PRLH11	PRLL11	0xXXXX	R/W	PPG11
PRLH12	PRLL12	0xXXXX	R/W	PPG12
PRLH13	PRLL13	0xXXXX	R/W	PPG13
PRLH14	PRLL14	0xXXXX	R/W	PPG14
PRLH15	PRLL15	0xXXXX	R/W	PPG15
PRLH16	PRLL16	0xXXXX	R/W	PPG16
PRLH17	PRLL17	0xXXXX	R/W	PPG17
PRLH18	PRLL18	0xXXXX	R/W	PPG18
PRLH19	PRLL19	0xXXXX	R/W	PPG19
PRLH20	PRLL20	0xXXXX	R/W	PPG20
PRLH21	PRLL21	0xXXXX	R/W	PPG21
PRLH22	PRLL22	0xXXXX	R/W	PPG22
PRLH23	PRLL23	0xXXXX	R/W	PPG23

■ Register configuration

bit	15	14	13	12	11	10	9	8
Field	PRLH							
Attribute	R/W							
Initial value	0xXX							

bit	7	6	5	4	3	2	1	0
Field	PRLL							
Attribute	R/W							
Initial value	0xXX							

■ Register functions

[bit15:8] PRLH: PPG Reload Register HIGH Set bits

These bits specify the PPG pulse width.

Process	Function
During writing	Any value can be written.
During reading	The register value is read. The initial value is undefined.

[bit7:0] PRLL: PPG Reload Register LOW Set bits

These bits specify the PPG pulse width.

Process	Function
During writing	Any value can be written.
During reading	The register value is read. The initial value is undefined.

This register specifies the width of the PPG output pulse. The pulse width can be changed during PPG operation. Both the High width and Low width are specified. The pulse width that is output is found by multiplying the count clock cycle by the written value with +1 added. If the PPG channels are connected to make a 16-bit length, the reload registers are also connected to specify a value that is 16 bits in length. When High width is set, the buffer register function is enabled. See Specifying the reload register and pulse width and Buffer function of high width setting reload register.

The setting content is determined uniquely by the PPG operation mode. The channel number 0 to 3 settings are shown below. The settings for channel numbers 4 and higher use the same combinations.

● 8-bit operation mode combination

PRLH0	PRLL0
The high width(8bit) of PPG0	The low width(8bit) of PPG0

PRLH1	PRLL1
The high width(8bit) of PPG1.	The low width(8bit) of PPG1

PRLH2	PRLL2
The high width(8bit) of PPG2	The low width(8bit) of PPG2

PRLH3	PRLL3
The high width(8bit) of PPG3	The low width(8bit) of PPG3

● 8+8-bit operation mode combination

PRLH0	PRLL0
The high width(8bit) of PPG0(PPG output side)	The low width(8bit) of PPG0(PPG output side)

PRLH1	PRLL1
The high width(8bit) of PPG1(prescaler side)	The low width(8bit) of PPG1(prescaler side)

PRLH2	PRLL2
The high width(8bit) of PPG2(PPG output side)	The low width(8bit) of PPG2(PPG output side)

PRLH3	PRLL3
The high width(8bit) of PPG3(prescaler side)	The low width(8bit) of PPG2(prescaler side)

- 16-bit operation mode combination

PRLH0	PRLL0

The high width(16bit) of PPG0/PPG1

PRLH1	PRLL1

The low width(16bit) of PPG0/PPG1

PRLH2	PRLL2

The high width(16bit) of PPG2/PPG3

PRLH3	PRLL3

The low width(16bit) of PPG2/PPG3

- 16+16-bit operation mode combination

PRLH0	PRLL0

The high width(16bit) of PPG0/PPG1(PPG output side)

PRLH1	PRLL1

The low width(16bit) of PPG0/PPG1(PPG output side)

PRLH2	PRLL2

The high width(16bit) of PPG2/PPG3(prescaler side)

PRLH3	PRLL3

The low width(16bit) of PPG2/PPG3(prescaler side)

5.11. Gate Function Control Register (GATECn n=0,4,8,12,16,20)

The GATEC Registers specify the start of the PPG using a GATE signal sent from the multifunction timer.

■ GATEC Register configuration list

bit 15 8	7 0	Initial value	Attribute	Corresponding PPG
Reserved	GATEC0	0x00	R/W	PPG2, PPG0
Reserved	GATEC4	0x00	R/W	PPG6, PPG4
Reserved	GATEC8	0x00	R/W	PPG10, PPG8
Reserved	GATEC12	0x00	R/W	PPG14, PPG12
Reserved	GATEC16	0x00	R/W	PPG18, PPG16
Reserved	GATEC20	0x00	R/W	PPG22, PPG20

■ Register configuration

bit	7	6	5	4	3	2	1	0
Field	Reserved		STRGn+2	EDGEn+2	Reserved		STRGn	EDGEn
Attribute	-		R/W	R/W	-		R/W	R/W
Initial value	-		0	0	-		0	0

■ Register functions

[bit7:6] Reserved: Reserved bits

0b00 is read from these bits.
Set these bits to 0b00 when writing.

[bit5] STRGn+2: Select trigger bit n+2 (where, n=0, 4, 8, 12, 16 or 20)

This bit is used to select the PPG start signal. The start signals for both PPGn+3 and PPGn+2 are selected by a combination of this bit and the PPGCn+2:TTRG: register. See Selecting the PPG start method.

Value	Function
0	Selects start from PPG start register (TRGx:PEN). [Initial value]
1	Selects start from GATE signal from multifunction timer.

[bit4] EDGEn+2: Start Effective Level Select bit n+2 (where, n=0, 4, 8, 12, 16 or 20)

This register is used by writing 0. A value of 0 is read.

[bit3:2] Reserved: Reserved bits

0b00 is read from these bits.
Set these bits to 0b00 when writing.

[bit1] STRGn: Select trigger bit n (where, n=0, 4, 8, 12, 16 or 20)

This bit is used to select the PPG start signal. The start signals for both PPGn+1 and PPGn are selected by a combination of this bit and the PPGCn:TTRG: register. See Selecting the PPG start method.

Value	Function
0	Selects start from PPG start register (TRGx:PEN). [Initial value]
1	Selects start from GATE signal from multifunction timer.

[bit0] EDGEn: Start Effective Level Select bit n (where, n=0, 4, 8, 12, 16 or 20)

This register is used by writing 0. A value of 0 is read.

6. Notes on using PPG

This section explains the notes when using the PPG.

● PPG Output Operations

When the PPG is operating, the pulse output waveform of LOW level period and HIGH level period are continuously output.
Once the pulse output has started, the PPG does not stop the output until PPG operation is stopped.

A reset signal must be entered or the PPG stop setting must be set to stop the operation.
The following explains PPG stop conditions.

- Start triggered by the Timing Generator Circuit
  Start signal is negated by writing PPGC:TRGxO=0

- Start triggered by GATE signal from the multifunction timer
  GATE signal from multifunction timer is negated

- Start triggered by PPG start register (TRG) writing
  Start signal is negated by writing TRG:PEN=0

● PPG Operation Mode Setting

The PPG operation mode is determined by setting the MD[1:0] bit of each PPGC register.
Be sure to always select the PPG operation mode before starting PPG.

● Other Module Settings

PPG pulses are output via the I/O port of the multifunction timer. The multifunction timer settings are explained in Chapter Multifunction Timer. For details on waveform output to I/O ports, see Chapter I/O Ports in Peripheral Manual. Also, for details on interrupts, see Chapter Interrupts in Peripheral Manual.

● PPG Output Signal and Interrupt Signal

Among the PPG output signals obtained by PPG timer operation, some channel outputs can be output to external terminals by passing through a multifunction timer. Also, some PPG interrupt outputs can be connected to interrupt controllers for performing interrupt processes.

See the chapter "PPG configuration" for details on the PPG output terminals that are output to external terminals by passing through a multifunction timer and PPG interrupts connected to interrupt controllers.

CHAPTER 7-3: PPG IGBT Mode

Functions and operations of PPG IGBT mode are explained as follows. (Only TYPE7 to TYPE 9, TYPE11, and TYPE12 products)

CODE: 9xFPPGIGBT-J01.0

1. Overview

Outline of IGBT mode is explained as follows.

PPG IGBT mode is a mode which outputs wave forms adequate for IGBT control. The wave forms are created with a combination of PPG outputs.

● IGBT mode features

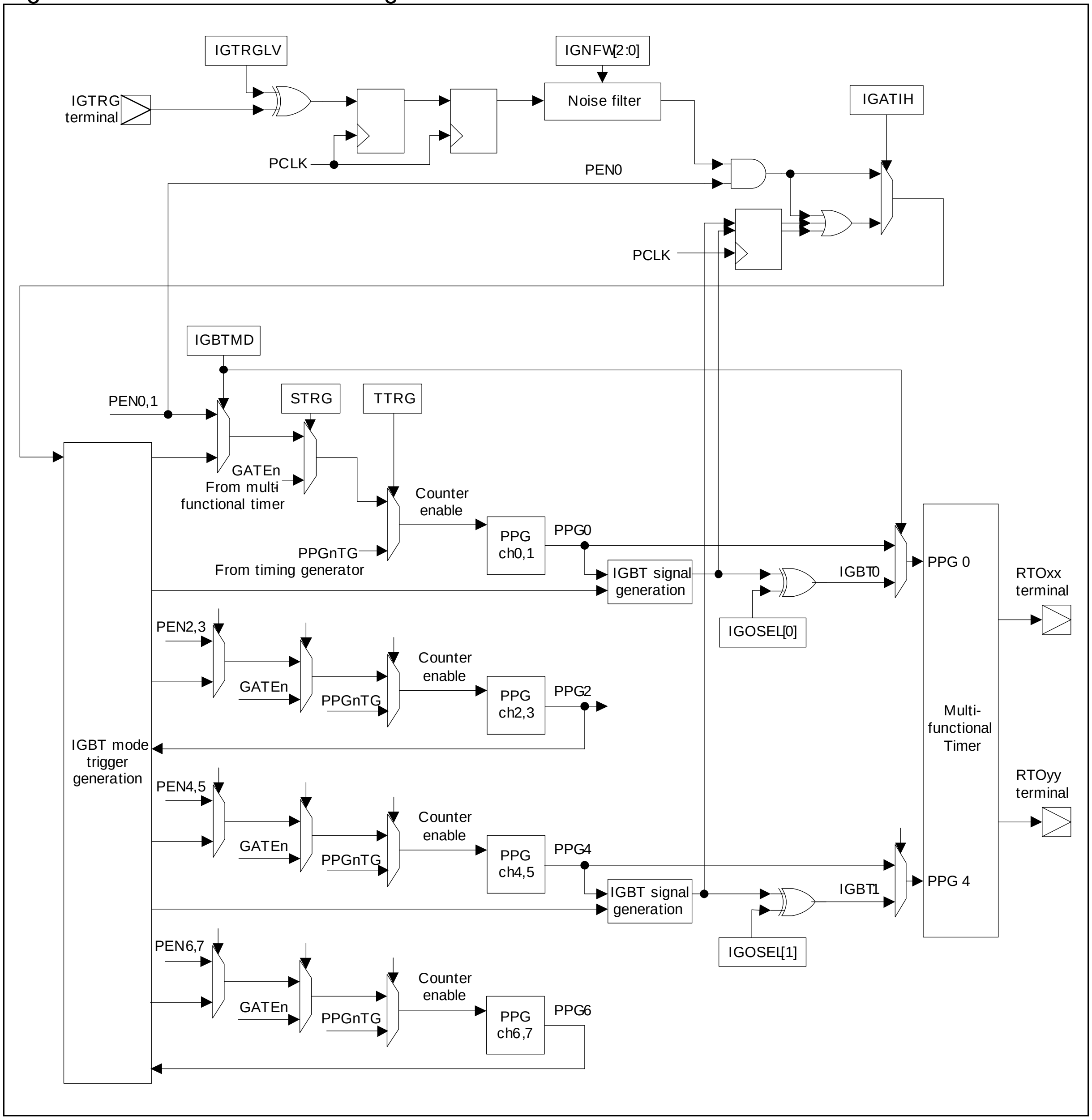
- IGBT output: 2ch
- Inverted IGBT output
- Triggering with external input IGTRG pin
- Inverted IGTRG input
- Trigger disable is selectable for active outputs
- Delay from trigger inputs through active outputs
- Noise filtering for IGTRG input pin

2. Configuration

Configuration of IGBT mode is as follows.

■ IGBT mode block diagram

Figure 2-1 IGBT Mode Block Diagram



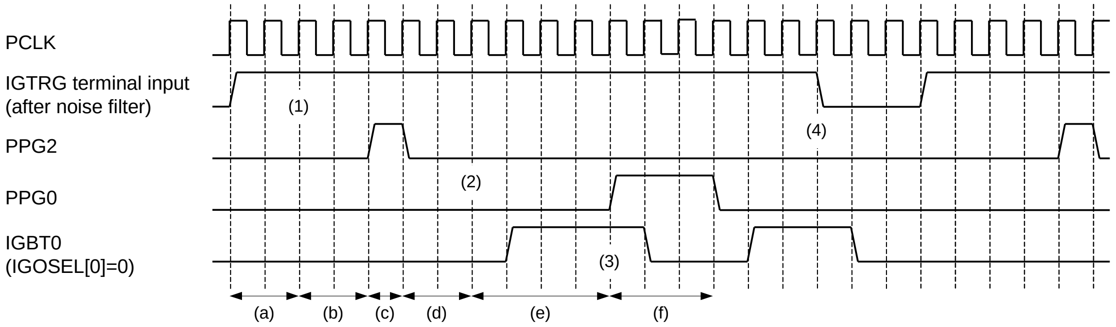
3. Operations

Operations in IGBT mode are explained as follows.

■ IGBT mode wave form output

Combinations of 16-bit PPG 2ch outputs create various wave forms.

Figure 3-1 Example of Operation when IGATIH=0



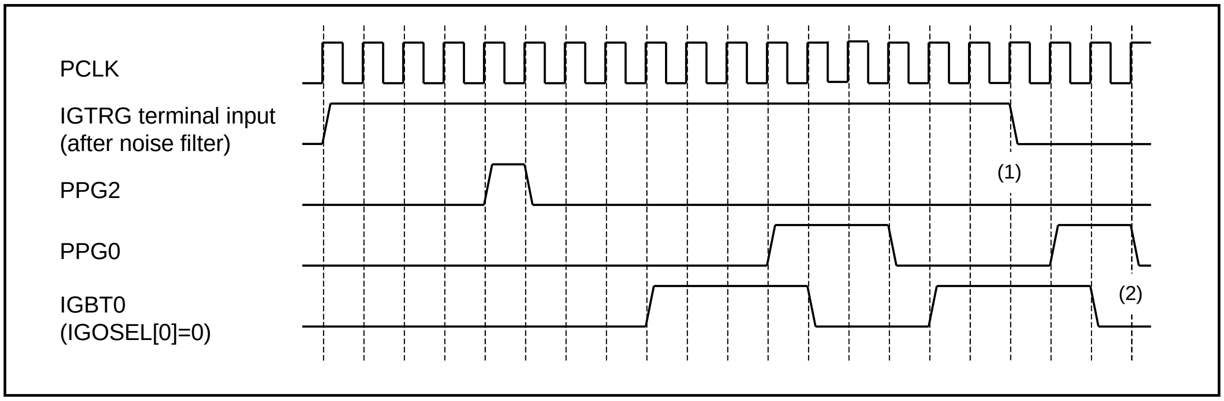
(a) 2 cycles fixed
(b) PPG2 L width setting = 2 cycles (minimum)
(c) PPG2 H width setting = 1 cycle (fixed)
(d) 2 cycles fixed
(e) PPG0 L width setting = 4 cycles
(f) PPG0 H width setting = 3 cycles

Figure 3-1 shows the example of IGBT0 operation when IGATIH=0. IGTRG input pin at “High” level after noise filter enables operations.

1. When the IGTRG input pin after noise filter becomes “High” level, PPG2 starts operating 2 cycles later.
2. After 2 cycles of PPG2 "H" pulses are output, PPG0 starts operating and the PPG2 stops. At that time, IGBT0 output becomes active.
3. After 1 cycle of PPG0 "H" output, IGBT0 becomes inactive.
4. When the IGTRG input pin after noise filter becomes "Low" level, the PPG0 stops and the IGBT0 becomes inactive.

IGBT0 signal is delayed by one clock in the multifunction timer and is output from RTOxx pin. From which RTOxx pin the signal is output can be selected with registers of the multifunction timer.

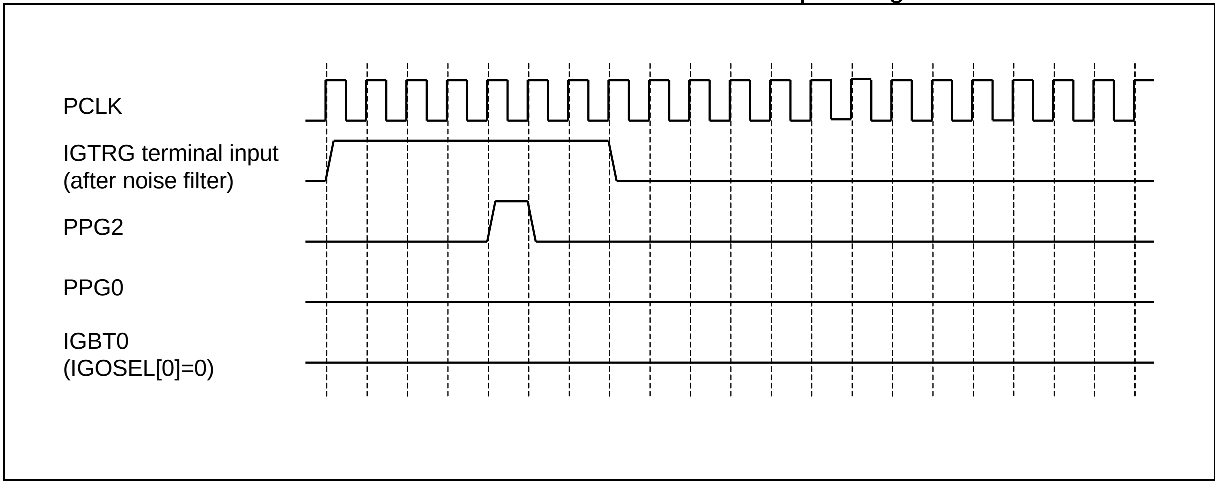
Figure 3-2 Example of Operation when IGATIH=1



IGBT0 signal is delayed by one clock in the multifunction timer and is output from RTOxx pin. From which RTOxx pin the signal is output can be selected with registers of the multifunction timer.
Figure 3-2 shows the example of IGBT0 operation when IGATIH=1.

1. Until the IGTRG input pin after noise filter becomes "Low" level, the same operation when IGATIH=0 applies. While IGBT0 is active, the operation will continue even if the IGTRG input pin after noise filter becomes "Low" level.
2. When the IGTRG input pin after noise filter becomes "Low" level while IGBT0 is inactive, the operation stops.

Figure 3-3 IGTRG after noise filter becomes Low while PPG2 is operating



When the IGTRG input pin after noise filter becomes "Low" level while PPG2 is operating, the operation stops before IGBT0 becomes inactive as shown in Figure 3-3.

Figure 3-4 Example of 2ch Operation (IGATIH=1)

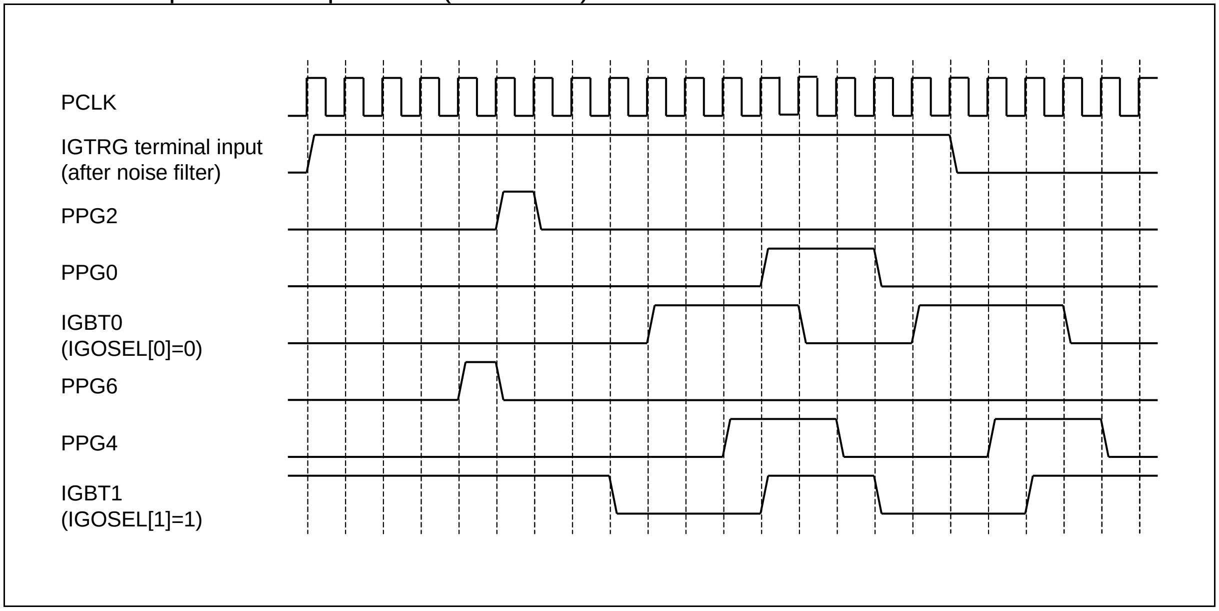


Figure 3-4 shows the example of 2ch when IGATIH=1. When any of IGBT0 or IGBT1 is active, the IGTRG input pin after noise filter will be ignored and the operation continues.

IGTRGLV bit and the trigger input level are shown in Table 3-1.

Table 3-1

IGTRGLV	Trigger input level
0	Operates when High
1	Operates when Low

IGOSEL bit and the output active level are shown in Table 3-2.

Table 3-2

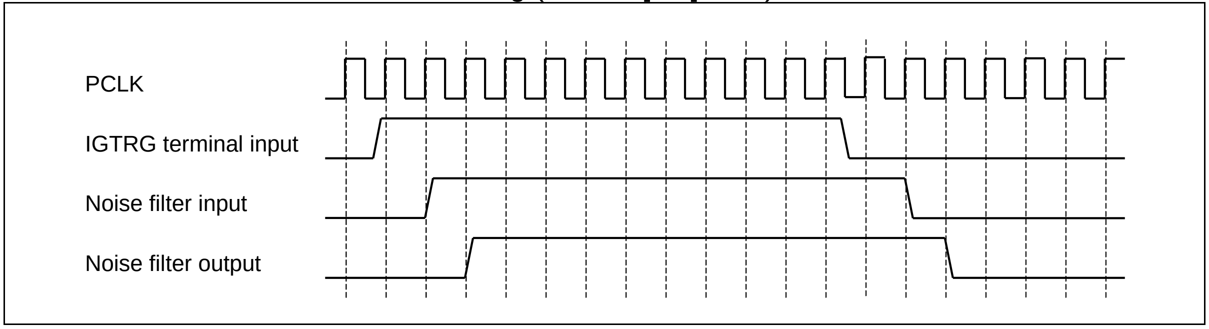
IGOSEL	Output active level
0	High
1	Low

■ Noise filter operations

The input signal at IGTRG pin is synchronized with PCLK×2 clock and then input to noise filter.

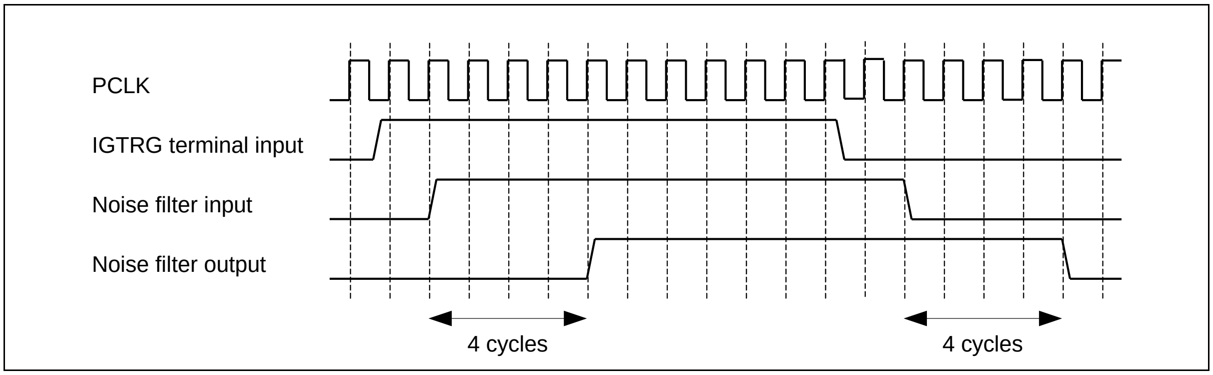
Wave forms without noise filtering (IGNFW[2:0]=000) are shown in Figure 3-5. 2 to 3 cycles of PCLK are delayed from the IGTRG input pin to noise filter output.

Figure 3-5 Wave forms without noise filtering (IGNFW[2:0]=000)



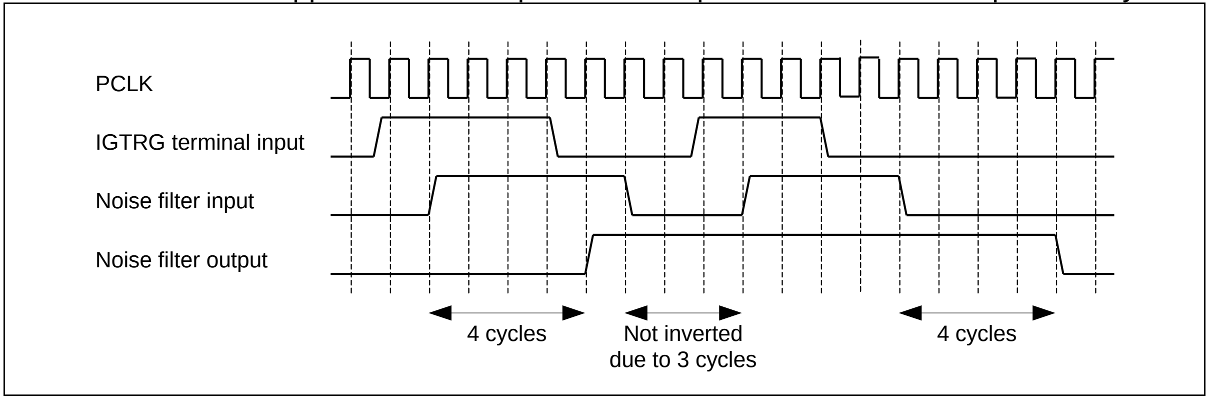
As an example while noise filter is enabled, a case where the noise filtering width is set to 4 PCLK cycles (IGNFW[2:0]=001) is explained. 5 to 6 cycles of PCLK are delayed from the IGTRG input pin to noise filter output. If a value opposite to the output value is input to the noise filter input for 4 cycles or more, the noise filter output will be inverted. The example is show in Figure 3-6.

Figure 3-6 When a value opposite to the output value is input to the noise filter input for 4 cycles or more



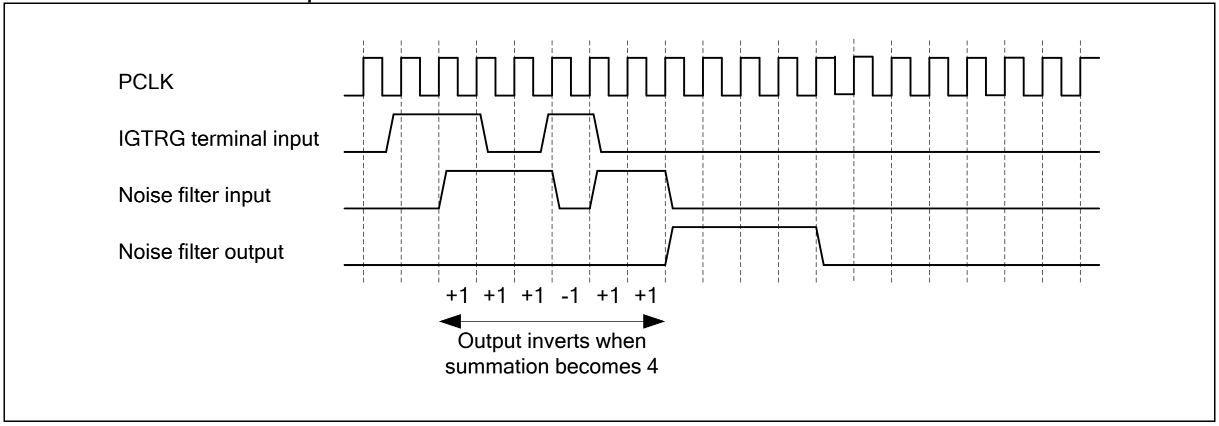
Wave forms in the case where a value opposite to the output value is input to the noise filter input for 3 cycles are shown in Figure 3-7.

Figure 3-7 When a value opposite to the output value is input to the noise filter input for 3 cycles



The calculation for a value opposite to the output value becomes “+1” and a value same with the output value becomes “-1”, and the output will be inverted when the summation becomes “4”. The example of wave forms is show in Figure 3-8.

Figure 3-8 Waveform example



Setting values of IGNFW[2:0] and maximum number of delay cycles from IGTRG pin to the noise filter are shown in Table 3-3.

Table 3-3 Delay from IGTRG Pin Input to Noise Filter Output

IGNFW[2:0]	Number of delay PCLK cycles
000	2 to 3
001	5 to 6
010	9 to 10
011	17 to 18
100	33 to 34

4. Example of Setting Procedure

Example of setting procedure for IGBT mode is explained as follows.

■ Example of Setting Procedure

1. Set TTRG bit of PPGC register to "0", STRG bit of GATEC register to "0" and PEN bit of TRG register to "0".
2. Set WFG operation mode for multifunctional timer to through mode.
3. Select PPG output for RTO pin of multifunctional timer WFG.
4. Set relevant bits of IGBTC register and write "1" to IGBTMD bit to change to IGBT mode.
5. Set I/O ports to RTO pin output.
6. Set PPG cycle.
7. Set PEN bit of TRG register for the channel to be used to "1".

5. Register

Configuration and functions of register used in IGBT mode are explained as follows.

■ IGBT mode register list

Abbreviated Register Name	Register Name	Reference
IGBTC	IGBT Mode Control Register	5.1

5.1. IGBT Mode Control Register (IGBTC)

IGBT Mode Control Register (IGBTC) controls operations in IGBT mode.

bit	7	6	5	4	3	2	1	0
Field	IGATIH	IGNFW[2:0]			IGOSEL[1:0]		IGTRGLV	IGBTMD
Attribute	R/W	R/W			R/W		R/W	R/W
Initial Value	0	000			00		0	0

[bit7] IGATIH: Stop prohibition mode selection in output active bit

Value	Description
0	Normal mode
1	Stop prohibition mode in output active

[bit6:4] IGNFW[2:0]: Noise filter width selection bit

Value	Description
000	No noise filter operation
001	Noise filter width is set to 4 PLCK cycles width.
010	Noise filter width is set to 8 PLCK cycles width.
011	Noise filter width is set to 16 PLCK cycles width.
100	Noise filter width is set to 32 PLCK cycles width.
Values other than the above	Setting is prohibited.

[bit3:2] IGOSEL[1:0]: Output level selection bit

IGBT0 corresponds to IGOSEL[0] and IGBT1 to IGOSEL[1].

Value	Description
0	Normal output
1	Inverted output

[bit1] IGTRGLV: Trigger input level selection bit

Value	Description
0	Normal input
1	Inverted input

[bit0] IGBTMD: IGBT mode selection bit

Value	Description
0	Normal mode
1	IGBT mode

6. Usage Precautions

Precautions for IGBT are as follows.

■ Precautions of IGBT mode

- Set PPG to 16bit mode.
- Set “L” width for PPG2 and PPG6 to 2 cycles or more.
- Set “H” width for PPG2 and PPG6 to 1 cycle.
- Do not change the IGBTC register while PPG is running.

CHAPTER 8-1: Quadrature Position/Revolution Counter

This chapter explains the functions and operations of the Quadrature Position/Revolution Counter (QPRC).

CODE: FX13-E02.3

1. Overview

The Quadrature Position/Revolution Counter is used to measure the position of Position Encoder. Also, it can be used as an up/down counter depending on the setting. The Quadrature Position/Revolution Counter contains a 16-bit position counter, a 16-bit revolution counter, two 16-bit compare registers, a control register, and its control circuit.

■ Features of Quadrature Position/Revolution Counter

● The position counter can be operated in one of the following 3 counting modes:

- PC_Mode1 : Up/down count mode
- PC_Mode2 : Phase difference count mode (supporting the 2-time and 4-time frequency multiplication)
- PC_Mode3 : Count mode with direction

● The revolution counter can be operated in one of the following 3 counting modes:

- RC_Mode1 : The revolution counter can count up or down at a ZIN active edge only.
- RC_Mode2 : The revolution counter can count up or down with an output value of position counter only.
- RC_Mode3 : The revolution counter can count up or down both with an output value of position counter and a signal at ZIN active edge.

● A signal edge detection can be set for detecting an input event from three AIN, BIN and ZIN external pins

- Detection of falling edge
- Detection of rising edge
- Detection of both rising and falling edges

● The following two functions can be selected for input in ZIN pin

- Counter clear function
- Gate function

● An interrupt request can be generated if:

- The position counter value matches the Position Compare Register,
- The position counter value matches the Position and Revolution Compare Register value, or the revolution counter value matches the Position and Revolution Compare Register value,
- The position counter underflows,
- The position counter overflows (that is, the position counter value matches the value of the QPRC Maximum Position Register),
- The position counter is reset at a ZIN active edge,
- The counting of position counter is inverted,
- The position counter matches the Position Compare Register value, and the revolution counter matches the Position and Revolution Compare Register value, or
- An outrange revolution counter value is detected.

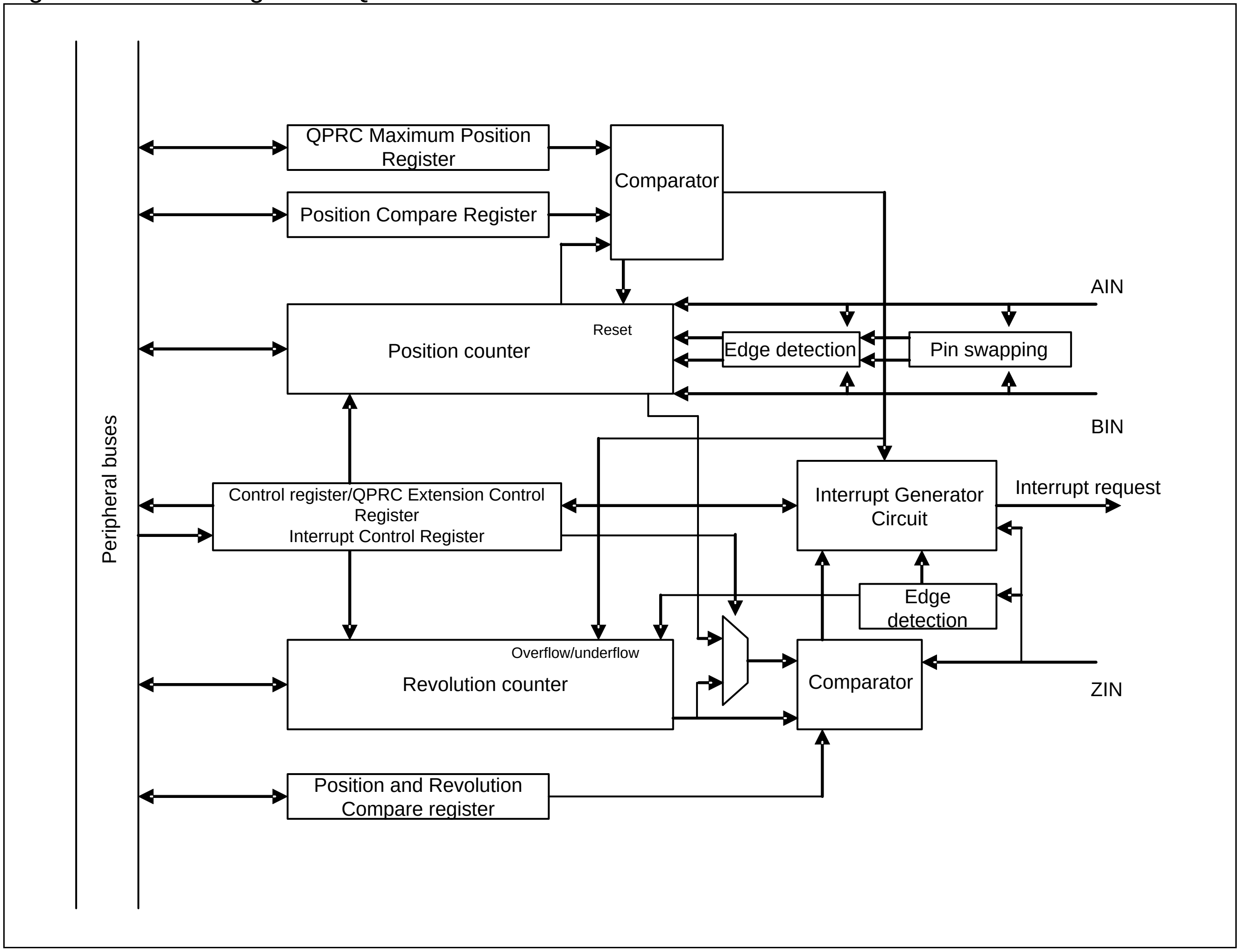
● The following useful functions are provided for counting

- Swap function of AIN and BIN external pins
- Mask reset function of the position counter
- Count direction check function during position counter operation or during overflow/underflow occurrence

2. Configuration

The following shows the configuration of Quadrature Position/Revolution Counter.

Figure 2-1 Block diagram of Quadrature Position/Revolution Counter


Peripheral buses
QPRC Maximum Position Register
Position Compare Register
Comparator
AIN
Reset
Position counter
Edge detection
Pin swapping
BIN
Control register/QPRC Extension Control Register
Interrupt Control Register
Interrupt Generator Circuit
Interrupt request
Edge detection
Overflow/underflow
Revolution counter
Comparator
ZIN
Position and Revolution Compare register

3. Operations

This section explains the operation of Quadrature Position/Revolution Counter.

■ Operation of position counter

The position counter receives an input signal from AIN or BIN external pin as an event of count clock, and increments or decrements the counter. As listed in Table 3-1, the position counter can select a counting mode by setting of the position counter mode bits (QCR:PCM[1:0]) of a control register. The counting conditions depend on the selected count mode.

The position counter is counted up or down in the following ZIN conditions only.

- If the ZIN function is set to the count clear function (QCR:CGSC="0")
- If the ZIN function is set to the Gate function (QCR:CGSC="1"), the ZIN low-level detection (QCR:CGE[1:0]="01") is set, and the ZIN is low level
- If the ZIN function is set to the Gate function (QCR:CGSC="1"), the ZIN high-level detection (QCR:CGE[1:0]="10") is set, and the ZIN is high level

If the ZIN function is set to the Gate function (QCR:CGSC="1") and if a level other than ZIN high- or low-level detection (QCR:CGE[1:0]="00" or "11") is set, the position counter is not counted up or down.

Also, if the AIN and BIN configurations are swapped by SWAP bits of a control register, the AIN and BIN pins are swapped and the position counter is counted up or down.
For example, if PC_Mode1 (QCR:PCM[1:0]="01") and AES[1:0]="10" (rising edge) and BES[1:0]="01" (falling edge) are set, the following occurs.

- If QCR:SWAP="0" and when a rising edge of AIN signal is detected, the position counter is counted up. When a falling edge of BIN signal is detected, the position counter is counted down.
- If QCR.SWAP="1", the position counter is counted down at a falling edge of AIN signal but it is counted up at a rising edge of BIN signal.

Table 3-1 Counting conditions of AIN and BIN pin position counter

Position count mode (PC_MODE)	AIN counting conditions	BIN counting conditions
Count disable PC_Mode0:QCR:PCM[1:0]="00"	Position counter disable	Position counter disable
Up/down counting PC_Mode1: QCR:PCM[1:0]="01"	AIN Active edge	BIN Active edge
Phase difference count PC_Mode2:QCR:PCM[1:0]="10"	AIN Active edge or high/low level	High/low level or BIN active edge
Counting with direction PC_Mode3:QCR:PCM[1:0]="11"	High/low level	BIN Active edge

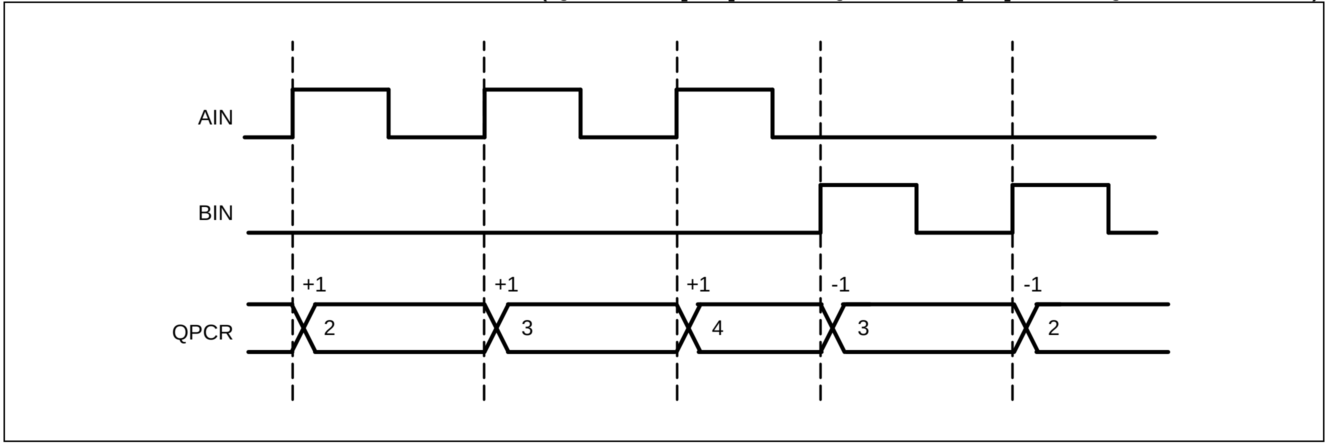
<Note>

The active edge of AIN signal and the active edge of BIN signal mean a rising edge, a falling edge, or both of edges set by the AIN Detection Edge Select bits (QCR:AES[1:0]="01" or "10" or "11") or by the BIN Detection Edge Select bits (QCR:BES[1:0]="01" or "10" or "11").

● PC_Mode1: Up/down count mode

- An external signal entered from AIN or BIN external pin is received as the counting clock, and the position counter is counted up or down.
- In this mode, the position counter is counted up when an active edge of AIN signal is detected. When an active edge of BIN signal is detected, the position counter is counted down.

Figure 3-1 Operations in up/down count mode
(QCR:AES[1:0]="10", QCR:BES[1:0]="10", QCR:SWAP="0")



● PC_Mode2: Phase difference count mode (supporting the 1-time, the 2-time and 4-time frequency multiplication)

- This mode is useful for counting the difference between phases A and B of "encoder output signal." If the phase-A and phase-B outputs are respectively connected to the AIN and BIN pins and if phase A is leading phase B, the counter is counted up. If delayed, the counter is counted down.
- In this mode, when an active edge of AIN signal is detected, the BIN signal level is checked and the position counter counts it. In the opposite case, the position counter also counts it.
- Counting in the 4-time or 2-time frequency multiplication can be made by setting the AES and BES bits of QPRC Control Register (QCR). The counting in these frequency multiplication modes allows more accurate position measurement as its counting resolution is very high.

Table 3-2 AES and BES bit settings in frequency multiplication mode

Frequency multiplication mode	AES[1:0] setting	BES[1:0] setting
1-time frequency multiplication mode	01	00
	10	00
	00	01
	00	10
2-time frequency multiplication mode	11	00
	00	11
4-time frequency multiplication mode	11	11

Table 3-3 Counting in 1-time frequency multiplication mode
(QCR:AES[1:0]="10", QCR:BES[1:0]="00", QCR:SWAP="0")

Edge detection pin	Detection edge	Level Check pin	Input level	Counting direction	Figure 3-2 Timing
AIN	Rising edge	BIN	Low	Up	(1)
	Rising edge		High	Down	(2)
	Falling edge		Low	Keep	(3)
	Falling edge		High	Keep	(4)

Figure 3-2 Operation in 1-time frequency multiplication mode
(QCR:AES[1:0]="10", QCR:BES[1:0]="00", QCR:SWAP="0")

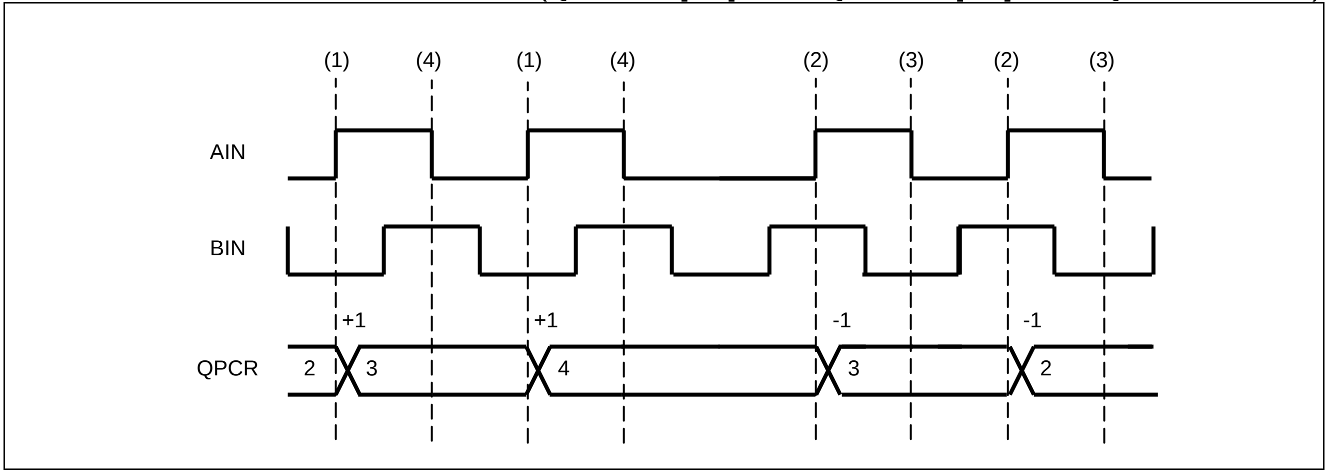


Table 3-4 Counting in 2-time frequency multiplication mode
(QCR:AES[1:0]="00", QCR:BES[1:0]="11", QCR:SWAP="0")

Edge detection pin	Detection edge	Level Check pin	Input level	Counting direction	Figure 3-3 Timing
BIN	Rising edge	AIN	High	Up	(1)
	Rising edge		Low	Down	(2)
	Falling edge		High	Down	(3)
	Falling edge		Low	Up	(4)

Figure 3-3 Operation in 2-time frequency multiplication mode
(QCR:AES[1:0]="00", QCR:BES[1:0]="11", QCR:SWAP="0")

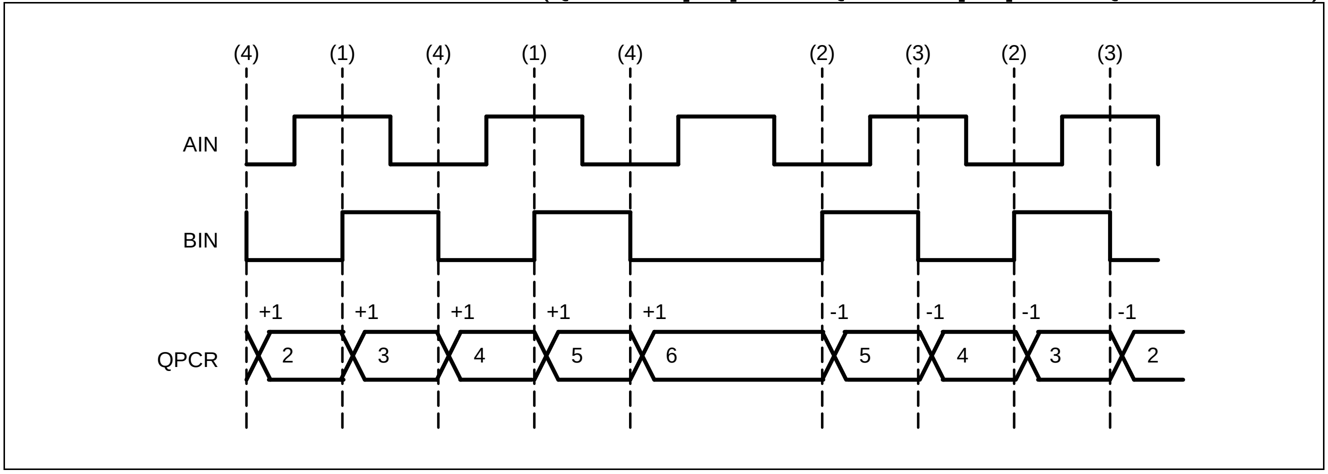
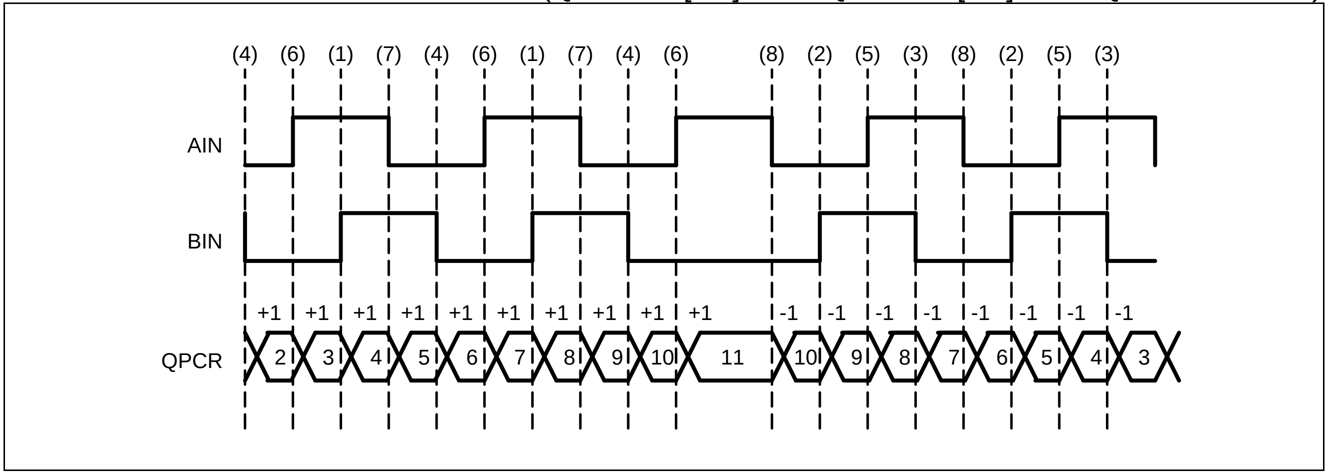


Table 3-5 Counting in 4-time frequency multiplication mode
(QCR:AES[1:0]="11", QCR:BES[1:0]="11")

Edge detection pin	Detection edge	Level Check pin	Input level	Counting direction	Figure 3-4 Timing
BIN	Rising edge	AIN	High	Up	(1)
	Rising edge		Low	Down	(2)
	Falling edge		High	Down	(3)
	Falling edge		Low	Up	(4)
AIN	Rising edge	BIN	High	Down	(5)
	Rising edge		Low	Up	(6)
	Falling edge		High	Up	(7)
	Falling edge		Low	Down	(8)

Figure 3-4 Operation in 4-time frequency multiplication mode
(QCR:AES[1:0]="11", QCR:BES[1:0]="11", QCR:SWAP="0")



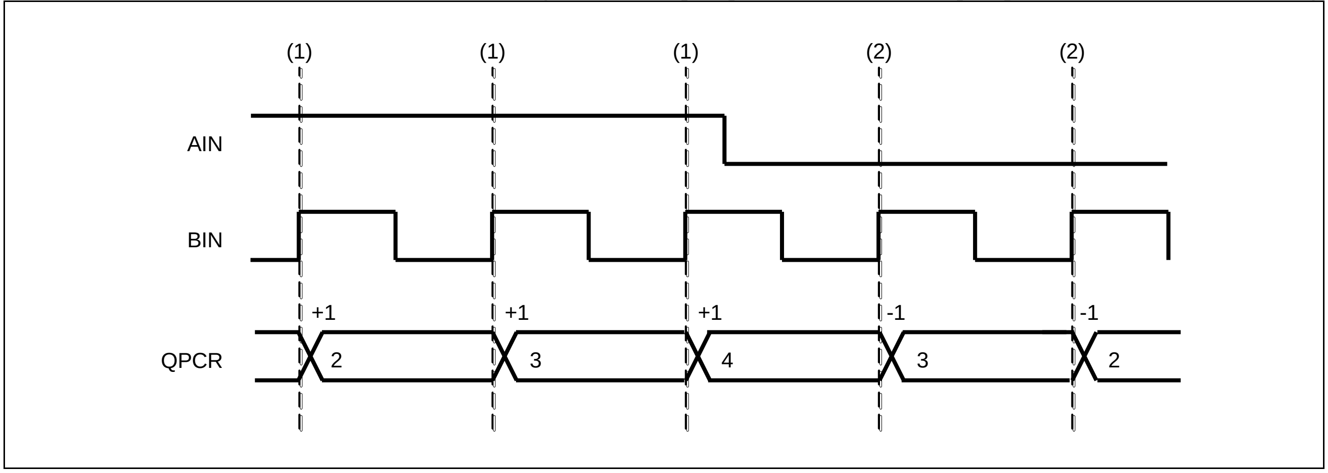
● PC_Mode3: Count mode with direction

- A signal entered from the BIN external pin is received as the counting clock, and an input level of the signal entered form the AIN external pin is used for count direction control for counter up/down counting.
- In this mode, when an active edge of BIN signal is detected, the AIN signal level is checked and the position counter counted up or down. A rising edge, a falling edge, or both can be set as the active edge.

Table 3-6 Counting in the direction control counting mode

Edge detection pin	Detection edge	Level Check pin	Input level	Counting direction	Figure 3-5 Timing
BIN	Active edge	AIN	High	Up	(1)
	Active edge		Low	Down	(2)

Figure 3-5 Operation in the direction control counting mode (QCR:AES[1:0]="00", QCR:BES[1:0]="10", QCR:SWAP="0")



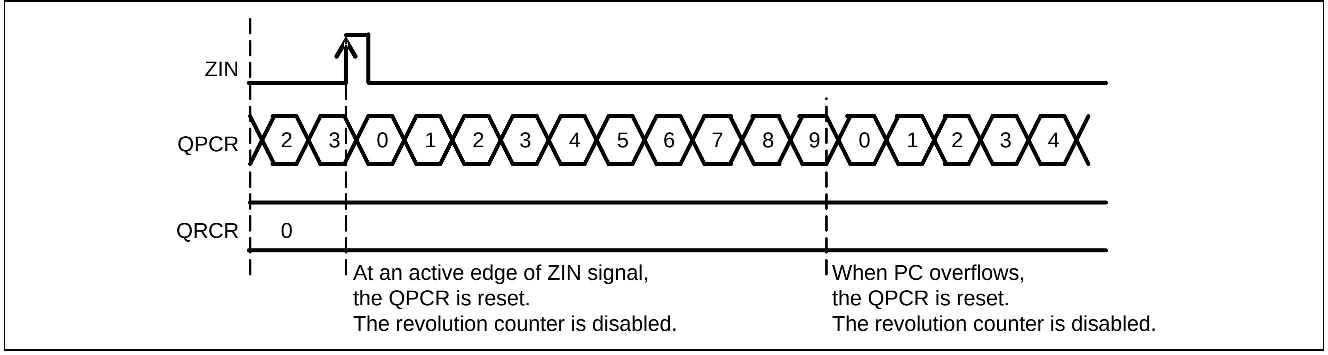
■ Operation of revolution counter

When the revolution counter receives an input from the ZIN pin (having the counter clear function) or an output of position counter (underflow or overflow), it is counted up or down. A rising edge, a falling edge, or both can be set as the active edge of ZIN signal.
The counting conditions of revolution counter depend on the selected mode as follows.

● RC_Mode0 (QCR:RCM[1:0]="00")

- The revolution counter is disabled.
- When the ZIN signal is used for counter clear function (QCR:CGSC="0"), the active edge of ZIN signal is reset. Also, the position counter is reset when this counter overflows.

Figure 3-6 RC_Mode0 operation (QPRC Maximum Position Register QMPR=9, QCR:CGSC="0")

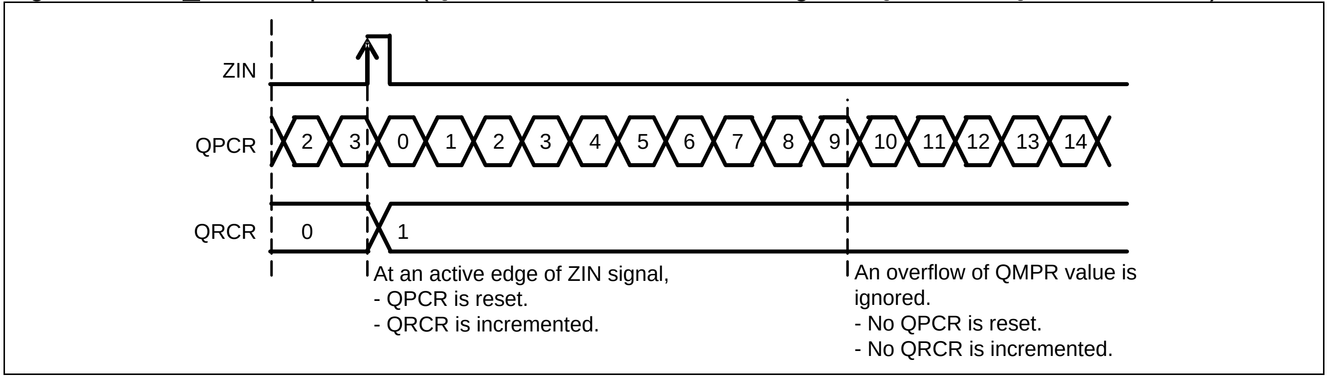

ZIN
QPCR
2
3
0
1
2
3
4
5
6
7
8
9
0
1
2
3
4
QRCR
0
At an active edge of ZIN signal,
the QPCR is reset.
The revolution counter is disabled.
When PC overflows,
the QPCR is reset.
The revolution counter is disabled.

QPCR: QPRC Position Count Register
QRCR: QPRC Revolution Count Register

● RC_Mode1 (QCR:RCM[1:0]="01")

- When ZIN signal is used for the counter clear function (QCR:CGSC="0"), the revolution counter is operated only at an active edge of ZIN signal (but an input from the position counter is ignored).
- When an active edge of ZIN signal is detected during incrementing of position counter (QICR:DIRPC="0"), the revolution counter is counted up. When an active edge of ZIN is detected during decrementing of position counter (QICR:DIRPC="1"), it is counted down.
- When the ZIN signal is used for counter clear function (QCR:CGSC="0"), the position counter is reset only at an active edge of ZIN signal.
- The position counter is not reset even when an overflow of position counter is detected. When an overflow of position counter is detected, the position counter is counted up and the overflow flag (QICR:OFDF) is set to "1".

Figure 3-7 RC_Mode1 operation (QPRC Maximum Position Register QMPR=9, QCR:CGSC="0")



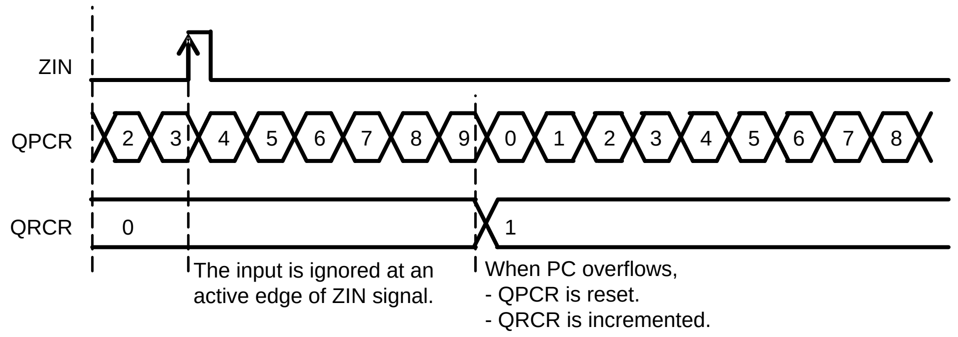
<Notes>

- When an active edge of ZIN signal and an active edge which counts down position counter are detected at the same time during incrementing of position counter (QICR:DIRPC="0"), the revolution counter is counted down.
- When an active edge of ZIN signal and an active edge which counts up position counter are detected at the same time during decrementing of position counter (QICR:DIRPC="1"), the revolution counter is counted up.
- When an active edge of ZIN signal, an active edge of AIN signal, and an active edge of BIN signal are detected at the same time, the revolution counter is counted up or down in accordance with the last position counter direction bit (QICR:DIRPC).

● RC_Mode2 (QCR:RCM[1:0]="10")

- The revolution counter is counted up or down only by the output value of position counter.
- The position counter is reset only when an overflow of position counter is detected (but an event of ZIN signal is ignored).
- When an overflow of position counter is detected in any of 3 position counter modes (PC_Mode1, PC_Mode2 and PC_Mode3), the position counter is counted up. When an underflow of it is detected, the position counter is counted down.

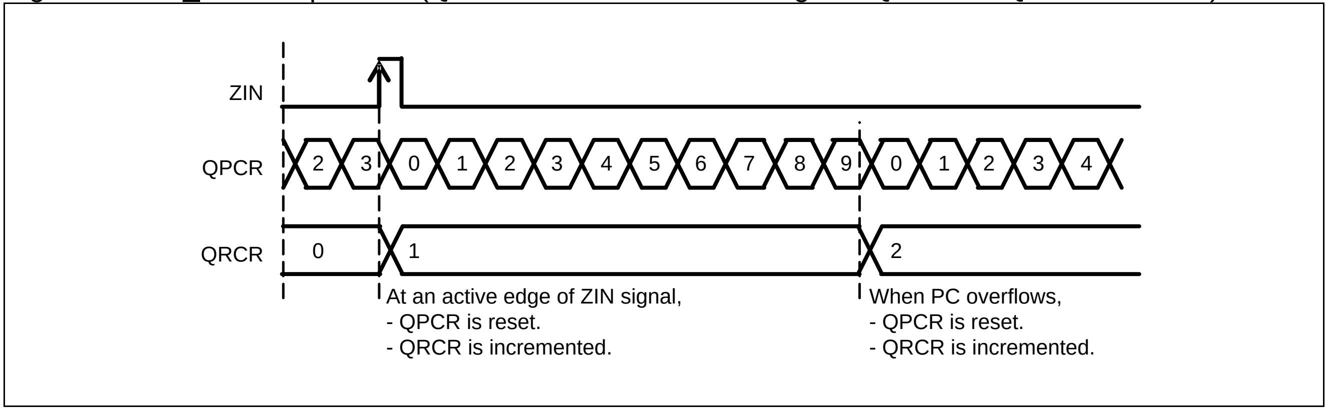
Figure 3-8 RC_Mode2 operation (QPRC Maximum Position Register QMPR=9)



● RC_Mode3 (QCR:RCM[1:0]="11")

- In this mode, the revolution counter is counted up or down with an output value from the position counter or at an active edge of ZIN when the ZIN is used as counter clear function (QCR:CGSC="0").
- When an active edge of ZIN signal is detected during incrementing of position counter (QICR:DIRPC="0") or when an overflow of position counter is detected, the revolution counter is counted up.
- When an active edge of ZIN signal is detected during decrementing of position counter (QICR:DIRPC="1") or when an underflow of position counter is detected, the revolution counter is counted down.
- When the ZIN signal is used for the counter clear function (QCR:CGSC="0"), the position counter is reset at an active edge of ZIN signal or at detection of position counter overflow.

Figure 3-9 RC_Mode3 operation (QPRC Maximum Position Register QMPR=9, QCR:CGSC="0")



At an active edge of ZIN signal,
- QPCR is reset.
- QRCR is incremented.

When PC overflows,
- QPCR is reset.
- QRCR is incremented.

<Notes>

- When an active edge of ZIN signal and an active edge which counts down position counter are detected at the same time during incrementing of position counter (QICR:DIRPC="0"), the revolution counter is counted down.
- When an active edge of ZIN signal and an active edge which counts up position counter are detected at the same time during decrementing of position counter (QICR:DIRPC="1"), the revolution counter is counted up.
- When an active edge of ZIN signal, an active edge of AIN signal, and an active edge of BIN signal are detected at the same time, the revolution counter is counted up or down in accordance with the last position counter direction bit (QICR:DIRPC).

■ Absolute value of positions

In RC_Mode2 and 3 modes (when the revolution counter operates with an output of position counter), each position has the following absolute value.

QPRC Position Count Register (QPCR) + QPRC Revolution Count Register (QRCR) × (QPRC Maximum Position Register (QMPR) +1)

Example: Time measurement

The revolution counter counts the "hours", and the position counter counts the "minutes".

If QMPR="59", QPCR="20", and QRCR="5"

Time = 20 + 5 × (59 + 1)

= 320 minutes. This is the absolute value in position counter units (minutes).

■ Quadrature Position/Revolution Counter interrupts

The following table defines the conditions where an interrupt request of Quadrature Position/Revolution Counter can generate.

Table 3-7 Generation conditions of Quadrature Position/Revolution Counter interrupt requests

Interrupt request	Interrupt request flag	Interrupt request is enabled if	Interrupt request is cleared if
Count inversion interrupt request	QICR:CDCF="1"	QICR:CDCIE="1"	QICR:CDCF is set to "0".
Zero index interrupt request	QICR:ZIIF="1"	QICR:OUZIE="1"	QICR:ZIIF is set to "0".
Overflow interrupt request	QICR:OFDF="1"		QICR:OFDF is set to "0".
Underflow interrupt request	QICR:UFDF="1"		QICR:UFDF is set to "0".
PC and RC match interrupt request	QICR:QPRCMF="1"	QICR:QPRCMIE="1"	QICR:QPRCMF is set to "0".
PC match interrupt request	QICR:QPCMF="1"	QICR:QPCMIE="1"	QICR:QPCMF is set to "0".
PC match and RC match interrupt request	QICR:QPCNRCMF="1"	QICR:QPCNRCMIE="1"	QICR:QPCNRCMF is set to "0".
Outrange interrupt request	QECR:ORNGF="1"	QICR:ORNGIE="1"	QECR:QRNGF is set to "0".

QICR: QPRC Interrupt Control Register
QECR: QPRC Extension Control Register

■ Operation example of QPRC Maximum Position Register (QMPR) interrupt

The QPRC Maximum Position Register (QMPR) value is used as the reload data to the position counter when an overflow or underflow of position counter is detected.

When the position counter value matches the QPRC Maximum Position Register (QMPR) value, the operation of the revolution counter depends on the selected mode as follows:

- When the position counter is counted up in RC_Mode0 (QCR:RCM[1:0]="00"), RC_Mode2 (QCR:RCM[1:0]="10") or RC_Mode3 (QCR:RCM[1:0]="11"), the overflow flag (QICR:OFDF) is set to "1" and the position counter is reset.
- When the position counter is counted up in RC_Mode1 (QCR:RCM[1:0]="01"), the overflow flag (QICR:OFDF) is set to "1". During this time, the position counter is not reset but is counted up.

The following gives an operation example where the QPRC Maximum Position Register (QMPR) is used in RC_Mode2 (QCR:RCM[1:0]="10").

During counting up

When the position counter maximum value overflows to "0x0000", the revolution counter is counted up. During this time, the overflow flag (QICRL:OFDF) is set to "1".

Example: If the QPRC Maximum Position Register (QMPR) is set to "18"

Position counter	15	16	17	18	0	1	2
Revolution counter	1	1	1	1	2	2	2

During counting down

When an underflow is detected (a value lower than "0x0000" is detected) and when the value of Quad Counter Maximum Position Counter Register (QMPR) is reloaded to the position counter, the revolution counter is counted down. During this time, the underflow flag (QICRL:UFDF) is set to "1".

Example: If the QPRC Maximum Position Register (QMPR) is set to "5"

Position counter	4	3	2	1	0	5	4	3	2	1	0	5
Revolution counter	1	1	1	1	1	0	0	0	0	0	0	0xFFFF

<Note>

The counting direction of position counter depends on the AIN and BIN external input signals only.

■ Position counter reset mask function

The position counter reset mask function can be used only when RC_Mode0 (QCR:RCM[1:0]="00") or RC_Mode3 (QCR:RCM[1:0]="11") is selected. This function operates regardless of setting of the position counter mode (PC_Mode1, PC_Mode2 or PC_Mode3).

The position counter reset mask function is executed in the following sequence.

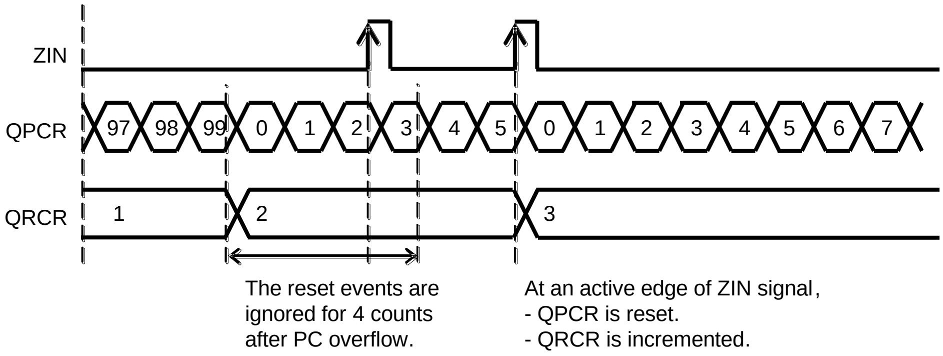
1. When an active event of ZIN signal, an overflow of position counter, or an underflow of position counter is detected, a value being set by the position counter reset mask bits (QCR:PCRM[1:0]) is set to the mask counter (*1).
2. When the position counter is counted up or down in the same counting direction, the mask counter (*1) is counted down.
   The position counter is reset only when the mask counter (*1) is set to "0x0". Also, the revolution counter is not counted up or down.
   When a count inversion of the position counter is detected, the mask counter (*1) is set to "0x0".
3. If the mask counter (*1) is set to "0x0", the position counter is set to "0x0000" when an active edge of ZIN signal or an overflow of position counter is detected.

*1 : The number of times to mask both the reset of position counter and the counting up/down of revolution counter is counted. The masking continues until this counter value reaches "0x0".

The following gives an operation example where the position counter reset mask function is used in RC_Mode3 (QCR:RCM[1:0]="11").

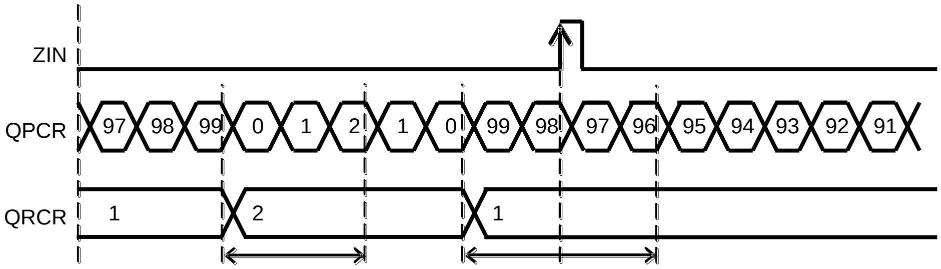
Example 1:
Active edge(s) of ZIN signal are ignored for four (4) counts(QPCR=0 to 3) of position counter after occurrence of position counter overflow.

Figure 3-10 Position counter reset mask operation example 1
(QMPR=99, QCR:PCRM[1:0]="10", QCR:CGSC="0")


ZIN
QPCR
97
98
99
0
1
2
3
4
5
0
1
2
3
4
5
6
7
QRCR
1
2
3
The reset events are ignored for 4 counts after PC overflow.
At an active edge of ZIN signal,
- QPCR is reset.
- QRCR is incremented.

Example 2:
Active edge(s) of ZIN signal are ignored for four (4) counts(QPCR=99 to 96) of position counter after count inversion of position counter.

Figure 3-11 Position counter reset mask operation example 2
(QMPR=99, QCR:PCRM[1:0]="10", QCR:CGSC="0")



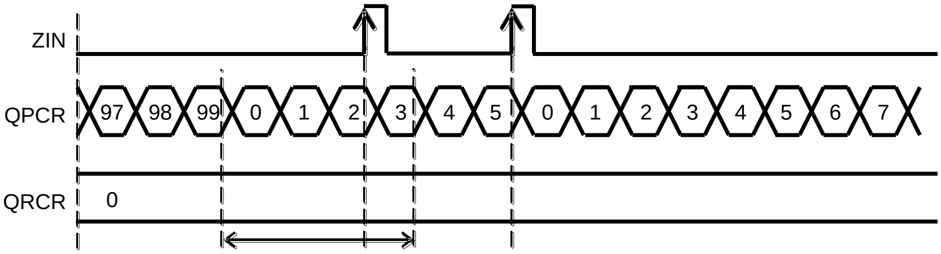
The reset events are ignored after PC overflow. The Mask function is canceled as PC counting was inverted.

The reset events are ignored for 4 counts after PC underflow.

The following gives an operation example where the position counter reset mask function is used in RC_Mode0 (QCR:RCM[1:0]="00").

Example 3:
Active edge(s) of ZIN signal are ignored for four (4) counts(QPCR=0 to 3) of position counter after occurrence of position counter overflow if the revolution counter is disabled.

Figure 3-12 Position counter reset mask operation example 3
(QMPR=99, QCR:PCRM[1:0]="10", QCR:CGSC="0")



The reset events are ignored for 4 counts after PC overflow.

At an active edge of ZIN signal, the QPCR is reset.

<Notes>

- While the position counter reset mask function is operating, the mask function is released and the position counter can be reset in the following conditions.
  - When the position counter mode bit (QCR:PCM[1:0]) is changed
  - When the revolution counter mode bit (QCR:RCM[1:0]) is changed
  - When the direction of the position counter is changed
- Even if an overflow or underflow of the position counter occurs without inversion of the position counter while the position counter reset mask function is operating in RC_Mode0 (QCR:RCM[1:0]="00") or RC_Mode3 (QCR:RCM[1:0]="11"), the revolution counter is not counted up or down. However, if an overflow occurs, the position counter becomes "0". If an underflow occurs, the QMPR is reloaded to the position counter. The overflow interrupt request flag bit (QICR:OFDF) or the underflow interrupt request flag bit (QICR:UFDF) is set to "1".

4. Registers

This section explains the configuration and functions of the registers used for the Quadrature Position/Revolution Counter (QPRC).

■ List of Quadrature Position/Revolution Counter registers

Abbreviation	Register name	Reference
QPCR	QPRC Position Count Register	4.1
QRCR	QPRC Revolution Count Register	4.2
QPCCR	QPRC Position Counter Compare Register	4.3
QPRCR	QPRC Position and Revolution Counter Compare Register	4.4
QCR	QPRC Control Register	4.5
QECR	QPRC Extension Control Register	4.6
QICRL	Low-Order Bytes of QPRC Interrupt Control Register	4.7
QICRH	High-Order Bytes of QPRC Interrupt Control Register	4.8
QMPR	QPRC Maximum Position Register	4.9

4.1. QPRC Position Count Register (QPCR)

The QPRC Position Count Register (QPCR) indicates the position counter.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	QPCR[15:0]															
Attribute	R/W															
Initial value	0x0000															

[bit15:0] QPCR:

Reading this register reads out the current value of the position counter. While the position counter stops counting (QCR:PSTP="1"), the count value can be written to this register. Also, write access is ignored while the counter is counting (QCR:PSTP=0),

This register is set to 0x0000 in one of the following conditions.

- Reset
- A ZIN active edge is detected in the following conditions.
  - The ZIN function is set to the counter clear function (QCR:CGSC="0") in RC_Mode1 (QCR:RCM[1:0]="01").
  - After the position counter has been incremented or decremented by the mask set value when the count inversion of the position counter is not detected where the ZIN function is set to the counter clear function (QCR:CGSC="0") and the reset mask function of the position counter is valid (QCR:PCRM[1:0]="01" or "10" or "11") in RC_Mode0 (QCR:RCM[1:0]="00") or RC_Mode3 (QCR:RCM[1:0]="11")
  - The ZIN function is set to the counter clear function (QCR:CGSC="0") and the reset mask function of the position counter is invalid (QCR:PCRM[1:0]="00") in RC_Mode0(QCR:RCM[1:0]="00") or RC_Mode3(QCR:RCM[1:0]="11").
- A position counter overflow is detected in the following conditions.
  - RC_Mode2(QCR:RCM[1:0]="10")
  - After the position counter has been incremented or decremented by the mask set value when the count inversion of the position counter is not detected where the ZIN function is set to the counter clear function (QCR:CGSC="0") and the reset mask function of the position counter is valid (QCR:PCRM[1:0]="01" or "10" or "11") in RC_Mode0 (QCR:RCM[1:0]="00") or RC_Mode3 (QCR:RCM[1:0]="11")
  - The ZIN function is set to the counter clear function (QCR:CGSC="0") and the reset mask function of the position counter is invalid (QCR:PCRM[1:0]="00") in RC_Mode0(QCR:RCM[1:0]="00") or RC_Mode3(QCR:RCM[1:0]="11").
- 0x0000 is written to this QPCR while the position counter is under suspension (QCR:PSTP="1").

The value of the QPRC Maximum Position Register (QMPR) is set to this register in the following condition.

- A position counter underflow is detected.

<Notes>

- Do not access the QPRC Position Count Register (QPCR) with a byte access instruction.
- After the count value was written to the QPRC Position Count Register (QPCR) while the position counter was under suspension (QCR:PSTP="1") in RC_Mode0 (QCR:RCM[1:0]="00"), RC_Mode1 (QCR:RCM[1:0]="01"), or RC_Mode3(QCR:RCM[1:0]="11"), if a ZIN active edge is detected with the count function (QCR:CGSC="0"), the QPRC Position Count Register (QPCR) will be set to 0x0000.
  To write the count value to the QPRC Position Count Register (QPCR), disable the ZIN detection edge (QCR:CGE[1:0]="00") before writing it to the QPCR.

4.2. QPRC Revolution Count Register (QRCR)

The QPRC Revolution Count Register (QRCR) indicates the revolution counter.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	QRCR[15:0]															
Attribute	R/W															
Initial value	0x0000															

[bit15:0] QRCR:

Reading this register reads out the current value of the revolution counter. While the revolution counter stops counting (QCR:RCM[1:0]="00"), the count value can be written to this register. Also, write access is ignored while the counter is counting (QCR:RCM[1:0]≠00),

This register is set to 0x0000 in one of the following conditions.

- Reset
- 0x0000 is written to this register while the revolution counter is under suspension (QCR:RCM[1:0]="00").

<Notes>

- Do not access the QPRC Revolution Count Register (QRCR) with a byte access instruction.
- As the direction of the position counter is not detected in PC_Mode0 (QCR:PCM[1:0]="00"), the last position counter direction bit (QICR:DIRPC) becomes indefinite. Therefore, if the mode is changed from PC_Mode0 (QCR:PCM[1:0]="00") to another mode, when a ZIN active edge is detected before an AIN/BIN active edge is detected, the following operations apply.
  - The position counter is reset if the mode is RC_Mode0 (QCR:RCM[1:0]="00"), RC_Mode1 (QCR:RCM[1:0]="01"), or RC_Mode3 (QCR:RCM[1:0]="11")
  - The revolution counter is not counted up or down

4.3. QPRC Position Counter Compare Register (QPCCR)

The QPRC Position Counter Compare Register (QPCCR) is used to compare with the count value of the position counter.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	QPCCR[15:0]															
Attribute	R/W															
Initial value	0x0000															

[bit15:0] QPCCR[15:0]:

If the value of this register matches that of the position counter, the QPRC position counter comparison match flag (QICR:QPCMF) is set to "1". This Compare Register can be used only to compare with the count value of the position counter.

<Note>

Do not access the QPRC Position Counter Compare Register (QPCCR) with a byte access instruction.

4.4. QPRC Position and Revolution Counter Compare Register (QPRCR)

The QPRC Position and Revolution Counter Compare Register (QPRCR) is used to compare with the selected count value of the position or revolution counter.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	QPRCR[15:0]															
Attribute	R/W															
Initial value	0x0000															

[bit15:0] QPRCR[15:0]:

Use the RSEL bit of the QPRC Control Register (QCR) to select the position counter or revolution counter to be compared with. If the value of this register matches that of the position or revolution counter, the QPRC position and revolution counter comparison match flag (QICR:QPRCMF) is set to "1".

<Note>

Do not access the QPRC Position and Revolution Counter Compare Register (QPRCR) with a byte access instruction.

4.5. QPRC Control Register (QCR)

The QPRC Control Register (QCR) is used to specify the operation mode of the position counter or 16-bit revolution counter. It is also used to start or stop each counter.

■ Low-Order Bytes of QPRC Control Register (QCRL)

bit	7	6	5	4	3	2	1	0
Field	SWAP	RSEL	CGSC	PSTP	RCM1	RCM0	PCM1	PCM0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] SWAP: Swap bit

This bit is used to swap the connections of the AIN input and BIN input to the position counter.
When this bit is set to "0", the AIN pin is used for the AIN input of the position counter, and the BIN pin is used for the BIN input of the position counter. When this bit is set to "1", the AIN pin is used for the BIN input of the position counter, and the BIN pin is used for the AIN input of the position counter.

Value	Description
0	No swap
1	Swaps AIN and BIN inputs.

<Note>

Change the swap bit (SWAP) when the position counter is disabled (PCM[1:0]="00").

[bit6] RSEL: Register function selection bit

This bit is used to select the position counter or revolution counter to be compared with the QPRC position and revolution counter compare register.

Value	Description
0	Compares the value of the QPRC Position and Revolution Counter Compare Register (QPRCR) with that of the position counter.
1	Compares the value of the QPRC Position and Revolution Counter Compare Register (QPRCR) with that of the revolution counter.

<Note>

When the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) and also the value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR), the PC match and RC match interrupt request flag bit (QICR: QPCNRCMF) is set to "1" regardless of the setting of this bit.

[bit5] CGSC: Count clear or gate selection bit

This bit is used to select the function of the ZIN external pin.

When the counter clear function is enabled (QGSC="0"), the ZIN pin clears the position counter if the revolution count mode is set to RC_Mode0 (RCM[1:0]="00"), RC_Mode1 (RCM[1:0]="01"), or RC_Mode3 (RCM[1:0]="11"). The CGE1 and CGE0 bits of the QCR register clear the position counter by selecting a valid edge of the ZIN pin and detecting the selected edge.

When the gate function is enabled (QGSC="1"), the ZIN pin controls the count operation of the position counter. The CGE1 and CGE0 bits of the QCR register count the position counter at the valid level of the ZIN pin.

Value	Description
0	Counter clear function
1	Gate function

[bit4] PSTP: Position counter stop bit

This bit is used to stop the position counter.

Value	Description
0	Enables count operation.
1	Stops count operation.

[bit3:2] RCM1, RCM0: Revolution counter mode bits

These bits are used to select the count mode of the revolution counter and the reset mode of the position counter. For the effect on the position counter, see "n Operation of revolution counter".

bit3	bit2	Description
0	0	Disables the revolution counter (RC_Mode0).
0	1	The revolution counter is counted up or down only with a ZIN active edge (RC_Mode1).
1	0	The revolution counter is counted up or down only when overflow or underflow is detected in the position counter that matches QMPR (RC_Mode2).
1	1	The revolution counter is counted up or down in two cases: a position counter overflow or underflow is detected and a ZIN active edge is detected (RC_Mode3).

[bit1:0] PCM1, PCM0: Position counter mode bits

These bits are used to select the count mode of the position counter.

bit1	bit0	Description
0	0	Disables the position counter (PC_Mode0) to stop it.
0	1	Up-down count mode (PC_Mode1) Increments the value with an AIN active edge and decrements it with a BIN active edge.
1	0	Phase difference count mode (PC_Mode2) Counts up if AIN is leading BIN and down if BIN is leading AIN.
1	1	Directional count mode (PC_Mode3) Counts up or down with the BIN active edge and AIN level.

<Note>

As the direction of the position counter is not detected in PC_Mode0 (PCM[1:0]="00"), the last position counter direction bit (QICR:DIRPC) becomes indefinite. Therefore, if the mode is changed from PC_Mode0 (PCM[1:0]="00") to another mode, when a ZIN active edge is detected before an AIN/BIN active edge is detected, the following operations apply.

- The position counter is reset if the mode is RC_Mode0 (RCM[1:0]="00"), RC_Mode1 (RCM[1:0]="01"), or RC_Mode3 (RCM[1:0]="11")
- The revolution counter is not counted up or down

■ High-Order Bytes of QPRC Control Register (QCRH)

bit	15	14	13	12	11	10	9	8
Field	CGE1	CGE0	BES1	BES0	AES1	AES0	PCRM1	PCRM0
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit15:14] CGE1, CGE0: Detection edge selection bits

These bits are used to select the detection edge when the ZIN external pin is used for the counter clear function (CGSC="0"). They are also used to select the detection level when the ZIN external pin is used for the gate function (CGSC="1").

bit15	bit14	ZIN used for counter clear function (CGSC="0")	ZIN used for gate function (CGSC="1")
0	0	Disables edge detection.	Disables level detection.
0	1	Detects a falling edge.	Detects level "L".
1	0	Detects a rising edge.	Detects level "H".
1	1	Detects a rising or falling edge.	Disables level detection.

[bit13:12] BES1, BES0: BIN detection edge selection bits

These bits are used to select the detection edge of the BIN external pin.

bit13	bit12	Description
0	0	Disables edge detection.
0	1	Detects a falling edge.
1	0	Detects a rising edge.
1	1	Detects rising and falling edges.

[bit11:10] AES1, AES0: AIN detection edge selection bits

These bits are used to select the detection edge of the AIN external pin.

bit11	bit10	Description
0	0	Disables edge detection.
0	1	Detects a falling edge.
1	0	Detects a rising edge.
1	1	Detects rising and falling edges.

[bit9:8] PCRM1, PCRM0: Position counter reset mask bits

These bits are used to specify the period (mask time) to ignore the events shown below after detecting a position counter overflow or underflow or detecting a ZIN active edge.

- Position counter resetting
- Revolution counter increment or decrement

This mask function is released when the count direction of the position counter is changed, and restarts when a position counter overflow or underflow is detected or a ZIN active edge is detected.

bit9	bit8	Description
0	0	No reset mask
0	1	The position counter reset or a revolution counter count-up or -down events are ignored until the position counter changes twice.
1	0	The position counter reset or a revolution counter count-up or -down events are ignored until the position counter changes four times.
1	1	The position counter reset or a revolution counter count-up or -down events are ignored until the position counter changes eight times.

<Notes>

- The position counter reset mask function is available only in RC_Mode0 (RCM[1:0]="00") and RC_Mode3 (RCM[1:0]="11"). This function operates regardless of the setting of the position counter mode (PC_Mode1, PC_Mode2, or PC_Mode3).
- While the position counter reset mask function is operating, the mask function is released and the position counter can be reset in the following conditions.
  - When the position counter mode bit (PCM[1:0]) is changed
  - When the revolution counter mode bit (RCM[1:0]) is changed
  - When the direction of the position counter is changed

4.6. QPRC Extension Control Register (QECR)

The QPRC Extension Control Register (QECR) is used to select that the revolution counter is inside the count range, indicate that the revolution counter is outside the count range, or control whether or not to generate an interrupt when the revolution counter gets out of the range.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	Reserved													ORNGIE	ORNGF	ORNGMD
Attribute	-													R/W	R/W	R/W
Initial value	0000000000000													0	0	0

[bit15:3] Reserved bits

Always write "0" to these bits. The read value is "0".

[bit2] ORNGIE: Outrange interrupt enable bit

This bit is used to control whether or not to issue an interrupt notification to the CPU when the outrange interrupt request flag (ORNGF) is set to "1". When this bit is set to "1", an interrupt is generated if the value of the revolution counter gets out of the range (ORNGF="1").

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit1] ORNGF: Outrange interrupt request flag bit

This flag indicates that the revolution counter is outside the count range.

If a positive number is selected as the outrange mode of the revolution counter (ORNGMD="0"), this flag is set to "1" when the revolution counter changes from 0x0001 to 0x0000 after counting down or when it changes from 0xFFFE to 0xFFFF after counting up.

If the 8K value is selected as the outrange mode of the revolution counter (ORNGMD="1"), this flag is set to "1" when the revolution counter changes from 0x8001 to 0x8000 after counting down or when it changes from 0x7FFE to 0x7FFF after counting up.

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Out of range is not detected.	Clears this bit.
1	Out of range is detected.	No effect.

[bit0] ORNGMD: Outrange mode selection bit

This bit defines the outrange mode of the revolution counter.

Value	Description
0	Selects a positive number (in the range from 0x0000 to 0xFFFF).
1	Selects the 8K value (in the range from 0x0000 to 0x7FFF).

4.7. Low-Order Bytes of QPRC Interrupt Control Register (QICRL)

The Low-Order Bytes of QPRC Interrupt Control Register (QICRL) are used to control a position counter overflow or underflow interrupt, zero index interrupt, QPRC position counter comparison match interrupt, or QPRC position and revolution counter comparison match interrupt.

bit	7	6	5	4	3	2	1	0
Field	ZIIF	OFDF	UFDF	OUZIE	QPRCMF	QPRCMIE	QPCMF	QPCMIE
Attribute	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0

[bit7] ZIIF: Zero index interrupt request flag bit

This flag is set to "1" when the position counter is reset by the ZIN input.

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Does not detect zero index.	Clears this bit.
1	Detects zero index.	No effect.

<Note>

The zero index interrupt request flag bit (ZIIF) is not set to "1" even if ZIN is used as the gate function (QCR:CGSC="1") or the position counter is reset in RC_Mode2 (QCR:RCM[1:0]="10").

[bit6] OFDF: Overflow interrupt request flag bit

This flag indicates that a position counter overflow occurs. When the set value of the QPRC Maximum Position Register (QMPR) matches the value of the position counter and the position counter is counted up, this bit is set to 1.

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Does not detect overflow.	Clears this bit.
1	Detects overflow.	No effect.

[bit5] UFDF: Underflow interrupt request flag bit

This flag indicates that a position counter underflow occurs. When the position counter is 0x0000 and the position counter is counted down, this bit is set to "1".

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Does not detect underflow.	Clears this bit.
1	Detects underflow.	No effect.

[bit4] OUZIE: Overflow, underflow, or zero index interrupt enable bit

This bit is used to control whether or not to issue an interrupt notification to the CPU when the overflow interrupt request flag bit (OFDF), underflow interrupt request flag bit (UFDF), or zero index interrupt request flag bit (ZIIF) is set to "1". When this bit is set to "1", an interrupt is generated if overflow is detected (OFDF="1"), underflow is detected (UFDF="1"), or zero index is detected (ZIIF="1").

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit3] QPRCMF: PC and RC match interrupt request flag bit

This flag indicates whether the value of the position counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) or the value of the revolution counter (QRCR) matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR).

When the comparison between the position counter and QPRC Position and Revolution Counter Compare Register (QPRCR) is selected (QCR:RSEL="0"), this flag is set to "1" if the value of the position counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR).

When the comparison between the revolution counter and QPRC Position and Revolution Counter Compare Register (QPRCR) is selected (QCR:RSEL="1"), this flag is set to "1" if the value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR).

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Does not detect comparison match with the QPRCR value.	Clears this bit.
1	Detects a comparison match with the QPRCR value.	No effect.

<Notes>

- If the register function selection bit (QCR:RSEL) is set to "0", the PC and RC match interrupt request flag bit (QPRCMF) is set to "1" immediately when one of the following conditions is satisfied.
  - The mode is changed to PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11") when the position counter is disabled (QCR:PCM[1:0]="00") and the value of the position counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR).
  - The value of the position counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) when data is written to the QPRC Position Count Register (QPCR) or QPRC Position and Revolution Counter Compare Register (QPRCR) in PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11").

- If the register function selection bit (QCR:RSEL) is set to "1", the PC and RC match interrupt request flag bit (QPRCMF) is set to "1" immediately when one of the following conditions is satisfied.
  - The value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) by writing data to the QPRC Position and Revolution Counter Compare Register (QPRCR) when the mode is RC_Mode1 (QCR:RCM[1:0]="01"), RC_Mode2 (QCR:RCM[1:0]="10"), or RC_Mode3 (QCR:RCM[1:0]="11").
  - The value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) by changing the mode from RC_Mode0 (QCR:RCM[1:0]="00") to another mode.

- When the register function selection bit (QCR:RSEL) is changed, the PC and RC match interrupt request flag bit (QPRCMF) is set to "1" immediately if one of the following conditions is satisfied.
  - The value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) when the register function selection bit (QCR:RSEL) is changed from "0" to "1" in the mode other than RC_Mode0 (QCR.RCM[1:0]="00").
  - The value of the position counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) when the register function selection bit (QCR:RSEL) is changed from "1" to "0" in the mode other than RC_Mode0 (QCR.RCM[1:0]="00").

[bit2] QPRCMIE: PC and RC match interrupt enable bit

This bit is used to control whether or not to issue an interrupt notification to the CPU when the PC and RC match interrupt request flag (QPRCMF) is set to "1". When this bit is set to "1", an interrupt is generated if the value of the position or revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) (QPRCMF="1").

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit1] QPCMF: PC match interrupt request flag bit

This flag indicates whether or not the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR).

This flag is set to "1" if the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR).

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Does not detect comparison match with the QPCCR value.	Clears this bit.
1	Detects a comparison match with the QPCCR value.	No effect.

<Notes>

The PC match interrupt request flag bit (QPCMF) is set to "1" immediately when one of the following conditions is satisfied.

- The mode is changed to PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11") when the position counter is disabled (QCR:PCM[1:0]="00") and the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR).
- The value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) by writing to the QPRC Position Count Register (QPCR) when the position counter stop bit (QCR:PSTP) is "1" and when the mode is PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11").
- The value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) by writing to the QPRC Position Counter Compare Register (QPCCR) when the mode is PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11").

[bit0] QPCMIE: PC match interrupt enable bit

This bit is used to control whether or not to issue an interrupt notification to the CPU when the PC match interrupt request flag (QPCMF) is set to "1".

When this bit is set to "1", an interrupt is generated if the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) (QPCMF="1").

Value	Description
0	Interrupt disabled
1	Interrupt enabled

4.8. High-Order Bytes of QPRC Interrupt Control Register (QICRH)

The High-Order Bytes of QPRC Interrupt Control Register (QICRH) are used to control a match between the position counter and QPCCR, a match between the revolution counter and QPRCR, and a count inversion interrupt. They are also used to indicate the direction of the position counter when the last underflow or overflow interrupt was detected or the last value of the position counter was changed.

bit	15	14	13	12	11	10	9	8
Field	Reserved		QPCNRCMF	QPCNRCMIE	DIROU	DIRPC	CDCF	CDCIE
Attribute	-		R/W	R/W	R	R	R/W	R/W
Initial value	00		0	0	0	0	0	0

[bit15:14] Reserved: Reserved bits

Always write "0" to these bits. The read value is "0".

[bit13] QPCNRCMF: PC match and RC match interrupt request flag bit

This flag indicates whether or not the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) and the value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR).

This flag is set to "1" when the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) (QPCMF="1") and the value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR).

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Does not detect match.	Clears this bit.
1	Detects a match.	No effect.

<Notes>

The PC match and RC match interrupt request flag bit (QPCNRCMF) is set to "1" immediately when one of the following conditions is satisfied.

- The mode is changed to PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11") when the position counter is disabled (QCR:PCM[1:0]="00") and the revolution counter is in the mode other than RC_Mode0(QCR:RCM[1:0]="00") while the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) and the value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR).
- The value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) when data is written to the QPRC Position Count Register (QPCR) or QPRC Position Counter Compare Register (QPCCR) where the value of the revolution counter matches that of the QPRC Position & Revolution Counter Compare Register (QPRCR) when the mode is PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11") and the revolution counter is in the mode other than RC_Mode0 (QCR:RCM[1:0]="00").
- The value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) when the data is written to the QPRC Position and Revolution Counter Compare Register (QPRCR) in the mode other than RC_Mode0 (QCR:RCM[1:0]="00") where the specified value matches that of the QPRC Position Count Register (QPCR) or QPRC Position Counter Compare Register (QPCCR) in PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11").
- The value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) when the mode is changed from RC_Mode0 (QCR:RCM[1:0]="00") to another mode where the specified value matches that of the QPRC Position Count Register (QPCR) or QPRC Position Counter Compare Register (QPCCR) in PC_Mode1 (QCR:PCM[1:0]="01"), PC_Mode2 (QCR:PCM[1:0]="10"), or PC_Mode3 (QCR:PCM[1:0]="11").
- This bit is set to "1" when the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) and the value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) regardless of the setting of the register function selection bit (QCR:RSEL).

[bit12] QPCNRCMIE: PC match and RC match interrupt enable bit

This bit is used to control whether or not to issue an interrupt notification to the CPU when the PC match and RC match interrupt request flag (QPCNRCMF) is set to "1".
When this bit is set to "1", an interrupt is generated if the value of the position counter matches that of the QPRC Position Counter Compare Register (QPCCR) and the value of the revolution counter matches that of the QPRC Position and Revolution Counter Compare Register (QPRCR) (QPCNRCMF="1").

Value	Description
0	Interrupt disabled
1	Interrupt enabled

[bit11] DIROU: Last position counter flow direction bit

This bit indicates the direction of the position counter when the last position counter overflow or underflow was detected.

Value	Description
0	The position counter was incremented.
1	The position counter was decremented.

[bit10] DIRPC: Last position counter direction bit

This bit indicates the count direction when the position counter was last changed.

Value	Description
0	The position counter was incremented.
1	The position counter was decremented.

<Note>

As the direction of the position counter is not detected in PC_Mode0 (QCR:PCM[1:0]="00"), the last position counter direction bit (QICR:DIRPC) becomes indefinite. Therefore, if the mode is changed from PC_Mode0 (QCR:PCM[1:0]="00") to another mode, when a ZIN active edge is detected before an AIN/BIN active edge is detected, the following operations apply.

- The position counter is reset if the mode is RC_Mode0 (QCR:RCM[1:0]="00"), RC_Mode1 (QCR:RCM[1:0]="01"), or RC_Mode3 (QCR:RCM[1:0]="11")
- The revolution counter is not counted up or down

[bit9] CDCF: Count inversion interrupt request flag bit

This bit indicates whether or not the position counter inverted the count direction.

This bit is set to "1" when the position counter inverts the count direction. Inverting the count direction means that the counter counts down at the next counting after counting up, or the counter counts up at the next counting after counting down.

This flag can only be cleared to "0" in write mode. Setting "1" has no effect.

"1" is read by the read-modify-write access operation.

Value	Description	
	Read	Write
0	Does not invert the count direction of the position counter.	Clears this bit.
1	Inverts the count direction of the position counter at least once.	No effect.

<Note>

As the direction of the position counter is not detected in PC_Mode0 (QCR:PCM[1:0]="00"), the last position counter direction bit (QICR:DIRPC) becomes indefinite. Therefore, after the mode is changed from PC_Mode0 (QCR:PCM[1:0]="00") to another mode, even if an AIN/BIN active edge is detected and the direction of the position counter is inverted, the count inversion interrupt request flag bit (QICR:CDCF) is not set to "1".

[bit8] CDCIE: Count inversion interrupt enable bit

This bit is used to control whether or not to issue an interrupt notification to the CPU when the count inversion interrupt request flag (CDCF) is set to "1".

When this bit is set to "1", an interrupt is generated if the count direction of the position counter is inverted (CDCF="1").

Value	Description
0	Interrupt disabled
1	Interrupt enabled

4.9. QPRC Maximum Position Register (QMPR)

The QPRC Maximum Position Register (QMPR) is used to specify the maximum value of the position counter.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	QMPR[15:0]															
Attribute	R/W															
Initial value	0xFFFF															

[bit15:0] QMPR:

When the set value of the QPRC Maximum Position Register (QMPR) matches the value of the position counter and the position counter is counted up, a position counter overflow is detected (QICR:OFDF=1). The set value of the QPRC Maximum Position Register (QMPR) is reloaded to the position counter if a position counter underflow is detected (QICR:UFDF=1).

<Note>

Do not access the QPRC Maximum Position Register (QMPR) with a byte access instruction.

CHAPTER 8-2: Quad Counter Position Rotation Count Display Function

This chapter describes the quad counter position rotation count dispay function (integrated in the products after TYPE2).

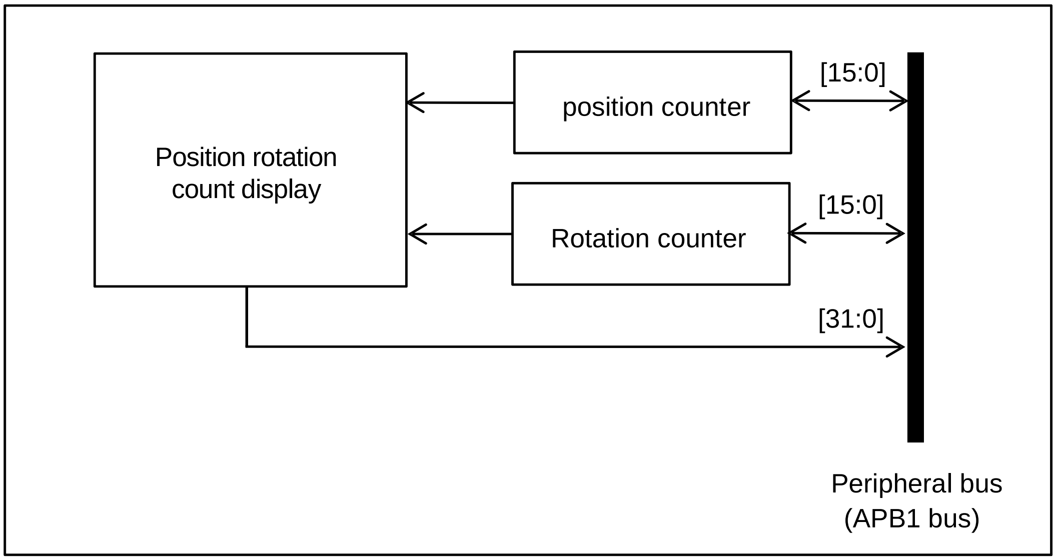
CODE: 9BFQPRCRR-E01.1

1. Overview and Configuration

This section describes the overview of the quad counter position rotation count display function.

The products after TYPE2 have functions that can read the value of the quad counter position count register (QPCR) and the value of the quad counter rotation count register (QRCR) simultaneously.

Figure 1-1 Block diagram of quad counter position rotation count register



2. Register

This section describes the register of the quad counter position rotation count display function.

Register abbreviation	Register name	Reference
QPRCRR	Quad Counter Position Rotation Count Register	2.1

2.1. Quad Counter Position Rotation Count Register (QPRCRR)

The values of the quad counter position count register (QPCR) and quad counter rotation count register (QRCR) are displayed.

■ Register configuration

bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Field	QRCRR [15:0]															
Attribute	R															
Initial value	0x0000															

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	QPCRR [15:0]															
Attribute	R															
Initial value	0x0000															

■ Register function

[bit31:16] QRCRR: Quad counter rotation count display bit

This is a mirror register that can read the same value as the quad counter rotation count register (QRCR). Writing is disabled.

[bit15:0] QPCRR: Quad counter position count display bit

This is a mirror register that can read the same value as the quad counter position count register (QPCR). Writing is disabled.

By using this register, the values of the quad counter position count register (QPCR) and quad counter rotation count register (QRCR) can be read simultaneously.

Appendixes

This chapter shows the register map, list of notes, limitations and product type list.

CODE: 9BFAPPENDIXES-E03.0

A. Register Map

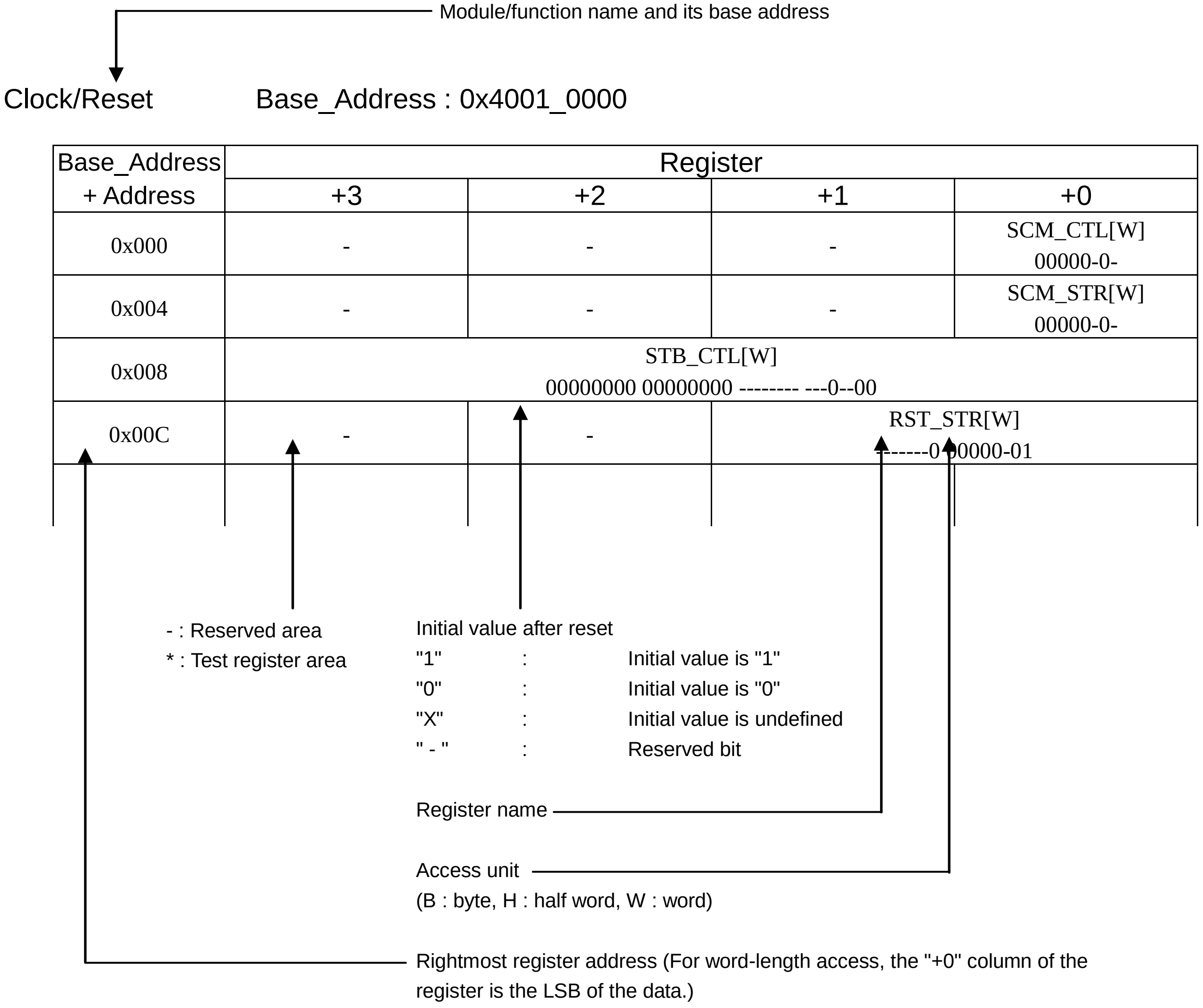
This chapter shows the register map.

CODE: 9BFREGMAP-E06.0

1. Register Map

Register map is shown on the table every module/function.

[How to read the each table]



Module/function name and its base address

Clock/Reset Base_Address : 0x4001_0000

Base_Address + Address	Register			
	+3	+2	+1	+0
0x000	-	-	-	SCM_CTL[W] 00000-0-
0x004	-	-	-	SCM_STR[W] 00000-0-
0x008	STB_CTL[W] 00000000 00000000 -------- ---0--00			
0x00C	-	-	RST_STR[W] -------0 00000-01	

- : Reserved area
* : Test register area

Initial value after reset

"1"	:	Initial value is "1"
"0"	:	Initial value is "0"
"X"	:	Initial value is undefined
" - "	:	Reserved bit

Register name

Access unit
(B : byte, H : half word, W : word)

Rightmost register address (For word-length access, the "+0" column of the register is the LSB of the data.)

<Notes>

- The register table is represented in the little-endian.
- When performing a data access, the addresses should be as below according to the access size.
  - Word access: Address should be multiples of 4 (least significant 2 bits should be "0x00")
  - Half word access: Address should be multiples of 2 (least significant bit should be "0x0")
  - Byte access: -
- Do not access the test register area.
- Do not access the area that is not written in the register table.
- When the register is accessed by larger unit than register size, for the reserved area to access at the same time, the read value is undefined, and writing is invalid.
- The respective meanings of *1 to *8 in the register map are as follows:

- *1: Initial value for TYPE0.
- *2: Initial value for TYPE1 to TYPE7.
- *3: Initial value for TYPE0, TYPE3, and TYPE7.
- *4: Initial value for TYPE1, TYPE2, TYPE4, and TYPE5.
- *5: Initial value for TYPE6, TYPE8, and TYPE9.
- *6: Initial value for TYPE3 and TYPE7.
- *7: Initial value for TYPE6 and TYPE8.
- *8: Initial value for TYPE9 to TYPE12.

1.1. Flash I/F

Base_Address : 0x4000_0000

■ Products other than TYPE6, and TYPE8 to TYPE12

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	FASZR[B,H,W]			
0x004	FRWTR[B,H,W]			
0x008	FSTR[B,H,W]			
0x00C	*			
0x010	FSYNDN[B,H,W]			
0x014	FBFCR[B,H,W]			
0x018 - 0x0FC	-	-	-	-
0x100	CRTRMM[B,H,W]			
0x104 - 0xFFC	-	-	-	-

■ TYPE6, and TYPE8 to TYPE11 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	-
0x004	FRWTR[B,H,W]			
0x008	FSTR[B,H,W]			
0x00C - 0x01C	-	-	-	-
0x020	FICR[B,H,W]			
0x024	FISR[B,H,W]			
0x028	FICLR[B,H,W]			
0x02C - 0x0FC	-	-	-	-
0x100	CRTRMM[B,H,W]			
0x104 - 0xFFC	-	-	-	-

■ TYPE12 products

Base_Address + Address	Register			
	+3	+2	+1	+0
0x000	-	-	-	-
0x004	FRWTR[B,H,W]			
0x008	FSTR[B,H,W]			
0x00C - 0x01C	-	-	-	-
0x020	FICR[B,H,W]			
0x024	FISR[B,H,W]			
0x028	FICLR[B,H,W]			
0x02C - 0x084	-	-	-	-
0x088	FSTR1[B,H,W]			
0x08C - 0x0FC	-	-	-	-
0x100	CRTRMM[B,H,W]			
0x104 - 0xFFC	-	-	-	-

Note:

For details of Flash I/F registers, see "FLASH PROGRAMMING MANUAL" of the product used.

1.2. Unique ID

Base_Address : 0x4000_0200

Base_Address + Address	Register			
	+3	+2	+1	+0
0x000	UIDR0 [W] XXXXXXXX XXXXXXXX XXXXXXXX XXXX----			
0x004	UIDR1 [W] -------- -------- ---XXXXX XXXXXXXX			
0x008 - 0xDFC	-	-	-	-

1.3. Clock/Reset

Base_Address : 0x4001_0000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	SCM_CTL[W] 00000-0-
0x004	-	-	-	SCM_STR[W] 00000-0-
0x008	STB_CTL[W] 00000000 00000000 -------- ---0-000			
0x00C	-	-	RST_STR[W] -------0 00000-01	
0x010	-	-	-	BSC_PSR[W] -----000
0x014	-	-	-	APBC0_PSR[W] ------00
0x018	-	-	-	APBC1_PSR[W] 1--0--00
0x01C	-	-	-	APBC2_PSR[W] 1--0--00
0x020	-	-	-	SWC_PSR[W] X-----00
0x024 - 0x027	-	-	-	-
0x028	-	-	-	TTC_PSR[W] ------00
0x02C - 0x02F	-	-	-	-
0x030	-	-	-	CSW_TMR[W] -0000000
0x034	-	-	-	PSW_TMR[W] ---0-000
0x038	-	-	-	PLL_CTL1[W] 00000000
0x03C	-	-	-	PLL_CTL2[W] ---00000
0x040	-	-	CSV_CTL[W] -111--00 ------11	

A. Register Map

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x044	-	-	-	CSV_STR[W] ------00
0x048	-	-	FCSWH_CTL[W] 11111111 11111111	
0x04C	-	-	FCSWL_CTL[W] 00000000 00000000	
0x050	-	-	FCSWD_CTL[W] 00000000 00000000	
0x054	-	-	-	DBWDT_CTL[W] 0-0-----
0x058	-	-	-	*
0x05C - 0x05F	-	-	-	-
0x060	-	-	-	INT_ENR[W] --0--000
0x064	-	-	-	INT_STR[W] --0--000
0x068	-	-	-	INT_CLR[W] --0--000
0x06C - 0xFFC	-	-	-	-

1.4. HW WDT

Base_Address : 0x4001_1000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	WDG_LDR[W]			
	00000000 00000000 11111111 11111111			
0x004	WDG_VLR[W]			
	XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX			
0x008	WDG_CTL[W]			
	-	-	-	------11
0x00C	WDG_ICL[W]			
	-	-	-	XXXXXXXX
0x010	WDG_RIS[W]			
	-	-	-	-------0
0x014 - 0xBFC	-	-	-	-
0xC00	WDG_LCK[W]			
	00000000 00000000 00000000 00000001			
0xC04 - 0xFFC	-	-	-	-

1.5. SW WDT

Base_Address : 0x4001_2000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	WdogLoad[W] 11111111 11111111 11111111 11111111			
0x004	WdogValue[W] 11111111 11111111 11111111 11111111			
0x008	WdogControl[W]			
	-	-	-	------00
0x00C	WdogIntClr[W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX			
0x010	WdogRIS[W]			
	-	-	-	-------0
0x014 - 0xBFC	-	-	-	-
0xC00	WdogLock[W] 00000000 00000000 00000000 00000000			
0xC04 - 0xFFC	-	-	-	-

1.6. Dual_Timer

Base_Address : 0x4001_5000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	Timer1Load[W] 00000000 00000000 00000000 00000000			
0x004	Timer1Value[W] 11111111 11111111 11111111 11111111			
0x008	Timer1Control[W] -------- -------- -------- 00100000			
0x00C	Timer1IntClr[W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX			
0x010	Timer1RIS[W] -------- -------- -------- -------0			
0x014	Timer1MIS[W] -------- -------- -------- -------0			
0x018	Timer1BGLoad[W] 00000000 00000000 00000000 00000000			
0x020	Timer2Load[W] 00000000 00000000 00000000 00000000			
0x024	Timer2Value[W] 11111111 11111111 11111111 11111111			
0x028	Timer2Control[W] -------- -------- -------- 00100000			
0x02C	Timer2IntClr[W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX			
0x030	Timer2RIS[W] -------- -------- -------- -------0			
0x034	Timer2MIS[W] -------- -------- -------- -------0			
0x038	Timer2BGLoad[W] 00000000 00000000 00000000 00000000			
0x040 - 0xFFC	-	-	-	-

1.7. MFT

unit0	Base_Address : 0x4002_0000
unit1	Base_Address : 0x4002_1000
unit2	Base_Address : 0x4002_2000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	OCCP0[H,W] 00000000 00000000	
0x004	-	-	OCCP1[H,W] 00000000 00000000	
0x008	-	-	OCCP2[H,W] 00000000 00000000	
0x00C	-	-	OCCP3[H,W] 00000000 00000000	
0x010	-	-	OCCP4[H,W] 00000000 00000000	
0x014	-	-	OCCP5[H,W] 00000000 00000000	
0x018	-	-	OCSB10[B,H,W] -110--00	OCSA10[B,H,W] 00001100
0x01C	-	-	OCSB32[B,H,W] -110--00	OCSA32[B,H,W] 00001100
0x020	-	-	OCSB54[B,H,W] -110--00	OCSA54[B,H,W] 00001100
0x024	-	-	OCSC[B,H,W] --000000	-
0x028	-	-	TCCP0[H,W] 11111111 11111111	
0x02C	-	-	TCDT0[H,W] 00000000 00000000	
0x030	-	-	TCSA0[B,H,W] 000---00 01000000	
0x034	-	-	TCSB0[B,H,W] -------- -----000	
0x038	-	-	TCCP1[H,W] 11111111 11111111	
0x03C	-	-	TCDT1[H,W] 00000000 00000000	

Base_Address + Address	Register			
	+3	+2	+1	+0
0x040	-	-	TCSA1[B,H,W] 000---00 01000000	
0x044	-	-	TCSB1[B,H,W] -------- -----000	
0x048	-	-	TCCP2[H,W] 11111111 11111111	
0x04C	-	-	TCDT2[H,W] 00000000 00000000	
0x050	-	-	TCSA2[B,H,W] 000---00 01000000	
0x054	-	-	TCSB2[B,H,W] -------- -----000	
0x058	-	-	OCFS32[B,H,W] 00000000	OCFS10[B,H,W] 00000000
0x05C	-	-	-	OCFS54[B,H,W] 00000000
0x060	-	-	ICFS32[B,H,W] 00000000	ICFS10[B,H,W] 00000000
0x064	-	-	-	-
0x068	-	-	ICCP0[H,W] XXXXXXXX XXXXXXXX	
0x06C	-	-	ICCP1[H,W] XXXXXXXX XXXXXXXX	
0x070	-	-	ICCP2[H,W] XXXXXXXX XXXXXXXX	
0x074	-	-	ICCP3[H,W] XXXXXXXX XXXXXXXX	
0x078	-	-	ICSB10[B,H,W] ------00	ICSA10[B,H,W] 00000000
0x07C	-	-	ICSB32[B,H,W] ------00	ICSA32[B,H,W] 00000000
0x080	-	-	WFTM10[H,W] 00000000 00000000	
0x084	-	-	WFTM32[H,W] 00000000 00000000	

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x088	-	-	WFTM54[H,W] 00000000 00000000	
0x08C	-	-	WFSA10[H,W] ---00000 000000	
0x090	-	-	WFSA32[H,W] ---00000 000000	
0x094	-	-	WFSA54[H,W] ---00000 000000	
0x098	-	-	WFIR[H,W] 00000000 0000--00	
0x09C	-	-	NZCL[H,W] -------- ---00000	
0x0A0	-	-	ACCP0[H,W] 00000000 00000000	
0x0A4	-	-	ACCPDN0[H,W] 00000000 00000000	
0x0A8	-	-	ACCP1[H,W] 00000000 00000000	
0x0AC	-	-	ACCPDN1[H,W] 00000000 00000000	
0x0B0	-	-	ACCP2[H,W] 00000000 00000000	
0x0B4	-	-	ACCPDN2[H,W] 00000000 00000000	
0x0B8	-	-	-	ACSB[B,H,W] -000-111
0x0BC	-	-	ACSA[B,H,W] --000000 --000000	
0x0C0	-	-	ATSA[H,W] --000000 --000000	
0x0C4 - 0x0FC	-	-	-	-

1.8. PPG

Base_Address : 0x4002_4000

Base_Address + Address	Register			
	+3	+2	+1	+0
0x000	-	-	TTCR0 [B,H,W] 11110000	-
0x004	-	-	-	*
0x008	-	-	COMP0 [B,H,W] 00000000	-
0x00C	-	-	-	COMP2 [B,H,W] 00000000
0x010	-	-	COMP4 [B,H,W] 00000000	-
0x014	-	-	-	COMP6 [B,H,W] 00000000
0x018 - 0x01C	-	-	-	-
0x020	-	-	TTCR1 [B,H,W] 11110000	-
0x024	-	-	-	*
0x028	-	-	COMP1 [B,H,W] 00000000	-
0x02C	-	-	-	COMP3 [B,H,W] 00000000
0x030	-	-	COMP5 [B,H,W] 00000000	-
0x034	-	-	-	COMP7 [B,H,W] 00000000
0x038 - 0x03C	-	-	-	-
0x040	-	-	TTCR2 [B,H,W] 11110000	-
0x044	-	-	-	*
0x048	-	-	COMP8 [B,H,W] 00000000	-
0x04C	-	-	-	COMP10 [B,H,W] 00000000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x050	-	-	COMP12 [B,H,W] 00000000	-
0x054	-	-	-	COMP14 [B,H,W] 00000000
0x58 - 0x0FC	-	-	-	-
0x100	-	-	TRG0 [B,H,W] 00000000 00000000	
0x104	-	-	REVC0 [B,H,W] 00000000 00000000	
0x108 - 0x13C	-	-	-	-
0x140	-	-	TRG1 [B,H,W] -------- 00000000	
0x144	-	-	REVC1 [B,H,W] -------- 00000000	
0x148 - 0x1FC	-	-	-	-
0x200	-	-	PPGC0 [B,H,W] 00000000	PPGC1 [B,H,W] 00000000
0x204	-	-	PPGC2 [B,H,W] 00000000	PPGC3 [B,H,W] 00000000
0x208	-	-	PRLH0 [B,H,W] XXXXXXXX	PRLL0 [B,H,W] XXXXXXXX
0x20C	-	-	PRLH1 [B,H,W] XXXXXXXX	PRLL1 [B,H,W] XXXXXXXX
0x210	-	-	PRLH2 [B,H,W] XXXXXXXX	PRLL2 [B,H,W] XXXXXXXX
0x214	-	-	PRLH3 [B,H,W] XXXXXXXX	PRLL3 [B,H,W] XXXXXXXX
0x218	-	-	-	GATEC0 [B,H,W] --00---00
0x21C - 0x23C	-	-	-	-
0x240	-	-	PPGC4 [B,H,W] 00000000	PPGC5 [B,H,W] 00000000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x244	-	-	PPGC6 [B,H,W] 00000000	PPGC7 [B,H,W] 00000000
0x248	-	-	PRLH4 [B,H,W] XXXXXXXX	PRLL4 [B.H.W] XXXXXXXX
0x24C	-	-	PRLH5 [B,H,W] XXXXXXXX	PRLL5 [B,H,W] XXXXXXXX
0x250	-	-	PRLH6 [B,H,W] XXXXXXXX	PRLL6 [B,H,W] XXXXXXXX
0x254	-	-	PRLH7 [B,H,W] XXXXXXXX	PRLL7 [B,H,W] XXXXXXXX
0x258	-	-	-	GATEC4 [B,H,W] --00--00
0x25C - 0x27C	-	-	-	-
0x280	-	-	PPGC8 [B,H,W] 00000000	PPGC9 [B,H,W] 00000000
0x284	-	-	PPGC10 [B,H,W] 00000000	PPGC11 [B,H,W] 00000000
0x288	-	-	PRLH8 [B,H,W] XXXXXXXX	PRLL8 [B,H,W] XXXXXXXX
0x28C	-	-	PRLH9 [B,H,W] XXXXXXXX	PRLL9 [B,H,W] XXXXXXXX
0x290	-	-	PRLH10 [B,H,W] XXXXXXXX	PRLL10 [B,H,W] XXXXXXXX
0x294	-	-	PRLH11 [B,H,W] XXXXXXXX	PRLL11 [B,H,W] XXXXXXXX
0x298	-	-	-	GATEC8 [B,H,W] --00--00
0x29C - 0x2BC	-	-	-	-
0x2C0	-	-	PPGC12 [B,H,W] 00000000	PPGC13 [B,H,W] 00000000
0x2C4	-	-	PPGC14 [B,H,W] 00000000	PPGC15 [B,H,W] 00000000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x2C8	-	-	PRLH12 [B,H,W] XXXXXXXX	PRLL12 [B,H,W] XXXXXXXX
0x2CC	-	-	PRLH13 [B,H,W] XXXXXXXX	PRLL13 [B,H,W] XXXXXXXX
0x2D0	-	-	PRLH14 [B,H,W] XXXXXXXX	PRLL14 [B,H,W] XXXXXXXX
0x2D4	-	-	PRLH15 [B,H,W] XXXXXXXX	PRLL15 [B,H,W] XXXXXXXX
0x2D8	-	-	-	GATEC12 [B,H,W] --00--00
0x2DC - 0x2FC	-	-	-	-
0x300	-	-	PPGC16 [B,H,W] 00000000	PPGC17 [B,H,W] 00000000
0x304	-	-	PPGC18 [B,H,W] 00000000	PPGC19 [B,H,W] 00000000
0x308	-	-	PRLH16 [B,H,W] XXXXXXXX	PRLL16 [B,H,W] XXXXXXXX
0x30C	-	-	PRLH17 [B,H,W] XXXXXXXX	PRLL17 [B,H,W] XXXXXXXX
0x310	-	-	PRLH18 [B,H,W] XXXXXXXX	PRLL18 [B,H,W] XXXXXXXX
0x314	-	-	PRLH19 [B,H,W] XXXXXXXX	PRLL19 [B,H,W] XXXXXXXX
0x318	-	-	-	GATEC16[B,H,W] --00--00
0x31C - 0x33C	-	-	-	-
0x340	-	-	PPGC20 [B,H,W] 00000000	PPGC21 [B,H,W] 00000000
0x344	-	-	PPGC22 [B,H,W] 00000000	PPGC23 [B,H,W] 00000000
0x348	-	-	PRLH20 [B,H,W] XXXXXXXX	PRLL20 [B,H,W] XXXXXXXX

Base_Address + Address	Register			
	+3	+2	+1	+0
0x34C	-	-	PRLH21 [B,H,W] XXXXXXXX	PRLL21 [B,H,W] XXXXXXXX
0x350	-	-	PRLH22 [B,H,W] XXXXXXXX	PRLL22 [B,H,W] XXXXXXXX
0x354	-	-	PRLH23 [B,H,W] XXXXXXXX	PRLL23 [B,H,W] XXXXXXXX
0x358	-	-	-	GATEC20 [B,H,W] --00--00
0x35C - 0x37C	-	-	-	-
0x380	-	-	-	IGBTC [B,H,W] 00000000
0x384 - 0xFFC	-	-	-	-

1.9. Base Timer

ch.0	Base Address : 0x4002_5000
ch.1	Base Address : 0x4002_5040
ch.2	Base Address : 0x4002_5080
ch.3	Base Address : 0x4002_50C0
ch.4	Base Address : 0x4002_5200
ch.5	Base Address : 0x4002_5240
ch.6	Base Address : 0x4002_5280
ch.7	Base Address : 0x4002_52C0
ch.8	Base Address : 0x4002_5400
ch.9	Base Address : 0x4002_5440
ch.10	Base Address : 0x4002_5480
ch.11	Base Address : 0x4002_54C0
ch.12	Base Address : 0x4002_5600
ch.13	Base Address : 0x4002_5640
ch.14	Base Address : 0x4002_5680
ch.15	Base Address : 0x4002_56C0

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	PCSR/PRLL [H,W] XXXXXXXX XXXXXXXX	
0x004	-	-	PDUT/PRLH/DTBF [H,W] XXXXXXXX XXXXXXXX	
0x008	-	-	TMR [H,W] 00000000 00000000	
0x00C	-	-	TMCR [B,H,W] -0000000 00000000	
0x010	-	-	TMCR2 [B,H,W] -------0	STC [B,H,W] 0000-000
0x014 - 0x03C	-	-	-	-

1.10. IO Selector for ch.0-ch.3 (Base Timer)

Base Address : 0x4002_5100

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	BTSEL0123 [B,H,W] 00000000	-
0x004 - 0x0FC	-	-	-	-

1.11. IO Selector for ch.4-ch.7(Base Timer)

Base Address : 0x4002_5300

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	BTSEL4567 [B,H,W] 00000000	-
0x004 - 0x0FC	-	-	-	-

1.12. IO Selector for ch.8-ch.11(Base Timer)

Base Address : 0x4002_5500

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	BTSEL89AB [B,H,W] 00000000	-
0x004 - 0x0FC	-	-	-	-

1.13. IO Selector for ch.12-ch.15 (Base Timer)

Base Address : 0x4002_5700

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	BTSELCDEF [B,H,W] 00000000	-
0x004 - 0x0FC	-	-	-	-

1.14. Software-based Simultaneous Startup (Base Timer)

Base Address : 0x4002_5F00

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000 - 0x0FB	-	-	-	-
0x0FC	-	-	BTSSSR [B,H,W] XXXXXXXX XXXXXXXX	

1.15. QPRC

ch.0	Base Address : 0x4002_6000
ch.1	Base Address : 0x4002_6040
ch.2	Base Address : 0x4002_6080

<table>
<tr><th rowspan="2">Base_Address
+ Address</th><th colspan="4">Register</th></tr>
<tr><th>+3</th><th>+2</th><th>+1</th><th>+0</th></tr>
<tr><td>0x000</td><td>-</td><td>-</td><td colspan="2">QPCR [H,W]
00000000 00000000</td></tr>
<tr><td>0x004</td><td>-</td><td>-</td><td colspan="2">QRCR [H,W]
00000000 00000000</td></tr>
<tr><td>0x008</td><td>-</td><td>-</td><td colspan="2">QPCCR [H,W]
00000000 00000000</td></tr>
<tr><td>0x00C</td><td>-</td><td>-</td><td colspan="2">QPRCR [H,W]
00000000 00000000</td></tr>
<tr><td>0x010</td><td>-</td><td>-</td><td colspan="2">QMPR [H,W]
11111111 11111111</td></tr>
<tr><td>0x014</td><td>-</td><td>-</td><td>QICRH [B,H,W]
--000000</td><td>QICRL [B,H,W]
00000000</td></tr>
<tr><td>0x018</td><td>-</td><td>-</td><td>QCRH [B,H,W]
00000000</td><td>QCRL [B,H,W]
00000000</td></tr>
<tr><td>0x01C</td><td>-</td><td>-</td><td colspan="2">QECR [B,H,W]
-------- -----000</td></tr>
<tr><td>0x020 - 0x038</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>0x03C</td><td colspan="2">QPCRR [B,H,W]
00000000 00000000</td><td colspan="2">QRCRR [B,H,W]
00000000 00000000</td></tr>
</table>

1.16. 12-bit A/DC

unit0 Base_Address : 0x4002_7000
unit1 Base_Address : 0x4002_7100
unit2 Base_Address : 0x4002_7200

■ TYPE0 to TYPE2, TYPE4, and TYPE5 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	ADCR[B,H,W] 000-0000	ADSR[B,H,W] 00---000
0x004	-	-	-	*
0x008	-	-	SCCR[B,H,W] 1000-000	SFNS[B,H,W] ----0000
0x00C	SCFD[B,H,W] XXXXXXXX XXXX---- ---1--XX ---XXXXX			
0x010	-	-	SCIS3[B,H,W] 00000000	SCIS2[B,H,W] 00000000
0x014	-	-	SCIS1[B,H,W] 00000000	SCIS0[B,H,W] 00000000
0x018	-	-	PCCR[B,H,W] 1000-000	PFNS[B,H,W] --XX--00
0x01C	PCFD[B,H,W] XXXXXXXX XXXX---- ---1-XXX ---XXXXX			
0x020	-	-	-	PCIS[B,H,W] 00000000
0x024	CMPD[B,H,W] 00000000 00------		-	CMPCR[B,H,W] 00000000
0x028	-	-	ADSS3[B,H,W] 00000000	ADSS2[B,H,W] 00000000
0x02C	-	-	ADSS1[B,H,W] 00000000	ADSS0[B,H,W] 00000000
0x030	-	-	ADST0[B,H,W] 00010000	ADST1[B,H,W] 00010000
0x034	-	-	-	ADCT[B,H,W] 00000111
0x038	-	-	SCTSL[B,H,W] ----0000	PRTSL[B,H,W] ----0000
0x03C	-	-	-	ADCEN[B,H,W] --00--00
0x040 - 0x0FC	-	-	-	-

■ TYPE3, and TYPE6 to TYPE12 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	ADCR[B,H,W] 000-0000	ADSR[B,H,W] 00---000
0x004	-	-	-	*
0x008	-	-	SCCR[B,H,W] 1000-000	SFNS[B,H,W] ----0000
0x00C	SCFD[B,H,W] XXXXXXXX XXXX---- ---1--XX ---XXXXX			
0x010	-	-	SCIS3[B,H,W] 00000000	SCIS2[B,H,W] 00000000
0x014	-	-	SCIS1[B,H,W] 00000000	SCIS0[B,H,W] 00000000
0x018	-	-	PCCR[B,H,W] 10000000	PFNS[B,H,W] --XX--00
0x01C	PCFD[B,H,W] XXXXXXXX XXXX---- ---1-XXX ---XXXXX			
0x020	-	-	-	PCIS[B,H,W] 00000000
0x024	CMPD[B,H,W] 00000000 00------		-	CMPCR[B,H,W] 00000000
0x028	-	-	ADSS3[B,H,W] 00000000	ADSS2[B,H,W] 00000000
0x02C	-	-	ADSS1[B,H,W] 00000000	ADSS0[B,H,W] 00000000
0x030	-	-	ADST0[B,H,W] 00010000	ADST1[B,H,W] 00010000
0x034	-	-	-	ADCT[B,H,W] 00000111
0x038	-	-	SCTSL[B,H,W] ----0000	PRTSL[B,H,W] ----0000
0x03C	-	-	ADCEN[B,H,W] 11111111 ------00	
0x040 - 0x0FC	-	-	-	-

1.17. 10-bit D/AC

Base_Address : 0x4002_8000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	DACR0[B,H,W] -------0	DADR0[B,H,W] ------XX XXXXXXXX	
0x004	-	DACR1[B,H,W] -------0	DADR1[B,H,W] ------XX XXXXXXXX	
0x008 - 0x0FC	-	-	-	-

1.18. CR Trim

Base_Address : 0x4002_E000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	MCR_PSR [B,H,W] ------01
0x004	-	-	MCR_FTRM[B,H,W] ------01 10000000 *1 ------01 10001110 *6 -------- 01111111 *4 ------10 00000000 *5	
0x008	-	-	-	MCR_TTRM [B,H,W] --011111
0x00C	MCR_RLR[W] 00000000 00000000 00000000 00000001			
0x010 - 0x0FC	-	-	-	-

1.19. EXTI

Base_Address : 0x4003_0000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	ENIR[B,H,W] 00000000 00000000 00000000 00000000			
0x004	EIRR[B,H,W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX			
0x008	EICL[B,H,W] 11111111 11111111 11111111 11111111			
0x00C	ELVR[B,H,W] 00000000 00000000 00000000 00000000			
0x010	ELVR1[B,H,W] 00000000 00000000 00000000 00000000			
0x014	-	-	NMIRR[B,H,W] -------- -------0	
0x018	-	-	NMICL[B,H,W] -------- -------1	
0x01C	-	-	-	-
0x020 - 0x0FC	-	-	-	-

1.20. INT-Req. READ

Base_Address : 0x4003_1000

■ Products other than TYPE3/TYPE7

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	DRQSEL[B,H,W] 00000000 00000000 00000000 00000000			
0x004	*			
0x008	ODDPKS[B] ---00000	-	-	*
0x00C	-	-	-	IRQCMODE[B,H,W] -------0
0x010	EXC02MON[B,H,W] -------- -------- -------- ------00			
0x014	IRQ00MON[B,H,W] -------- -------- -------- -------0			
0x018	IRQ01MON[B,H,W] -------- -------- -------- -------0			
0x01C	IRQ02MON[B,H,W] -------- -------- -------- -------0			
0x020	IRQ03MON[B,H,W] -------- -------- ----0000 00000000			
0x024	IRQ04MON[B,H,W] -------- -------- -------- 00000000			
0x028	IRQ05MON[B,H,W] -------- 00000000 00000000 00000000			
0x02C	IRQ06MON[B,H,W] -------- ----0000 00000000 00000000			
0x030	IRQ07MON[B,H,W] -------- -------- -------- ------00			
0x034	IRQ08MON[B,H,W] -------- -------- -------- ----0000			
0x038	IRQ09MON[B,H,W] -------- -------- -------- ------00			
0x03C	IRQ10MON[B,H,W] -------- -------- -------- ----0000			

<table>
<tr><th rowspan="2">Base_Address
+ Address</th><th colspan="4">Register</th></tr>
<tr><th>+3</th><th>+2</th><th>+1</th><th>+0</th></tr>
<tr><td>0x040</td><td colspan="4">IRQ11MON[B,H,W]
-------- -------- -------- ------00</td></tr>
<tr><td>0x044</td><td colspan="4">IRQ12MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
<tr><td>0x048</td><td colspan="4">IRQ13MON[B,H,W]
-------- -------- -------- ------00</td></tr>
<tr><td>0x04C</td><td colspan="4">IRQ14MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
<tr><td>0x050</td><td colspan="4">IRQ15MON[B,H,W]
-------- -------- -------- ------00</td></tr>
<tr><td>0x054</td><td colspan="4">IRQ16MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
<tr><td>0x058</td><td colspan="4">IRQ17MON[B,H,W]
-------- -------- -------- ------00</td></tr>
<tr><td>0x05C</td><td colspan="4">IRQ18MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
<tr><td>0x060</td><td colspan="4">IRQ19MON[B,H,W]
-------- -------- -------- ------00</td></tr>
<tr><td>0x064</td><td colspan="4">IRQ20MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
<tr><td>0x068</td><td colspan="4">IRQ21MON[B,H,W]
-------- -------- -------- ------00</td></tr>
<tr><td>0x06C</td><td colspan="4">IRQ22MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
<tr><td>0x070</td><td colspan="4">IRQ23MON[B,H,W]
-------- -------- -------0 00000000</td></tr>
<tr><td>0x074</td><td colspan="4">IRQ24MON[B,H,W]
-------- -------- -------- --000000</td></tr>
<tr><td>0x078</td><td colspan="4">IRQ25MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
<tr><td>0x07C</td><td colspan="4">IRQ26MON[B,H,W]
-------- -------- -------- ----0000</td></tr>
</table>

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x080	IRQ27MON[B,H,W] -------- -------- -------- ---00000			
0x084	IRQ28MON[B,H,W] -------- ------00 00000000 00000000			
0x088	IRQ29MON[B,H,W]			
	-------- -------- ----0000 00000000			
0x08C	IRQ30MON[B,H,W] -------- ------00 00000000 00000000			
0x090	IRQ31MON[B,H,W] -------- -------- 00000000 00000000			
0x094	IRQ32MON[B,H,W] -------- -------- -------- ----0000			
0x098	IRQ33MON[B,H,W] -------- -------- -------- -----000			
0x09C	IRQ34MON[B,H,W] -------- -------- -------- ---00000			
0x0A0	IRQ35MON[B,H,W] -------- -------- -------- --000000			
0x0A4	IRQ36MON[B,H,W] -------- -------- -------- --000000			
0x0A8	IRQ37MON[B,H,W] -------- -------- -------- -0000000			
0x0AC	IRQ38MON[B,H,W] -------- -------- -------- -------0			
0x0B0	IRQ39MON[B,H,W] -------- -------- -------- -------0			
0x0B4	IRQ40MON[B,H,W] -------- -------- -------- -------0			
0x0B8	IRQ41MON[B,H,W] -------- -------- -------- -------0			
0x0BC	IRQ42MON[B,H,W] -------- -------- -------- -------0			

Base_Address + Address	Register			
	+3	+2	+1	+0
0x0C0	IRQ43MON[B,H,W] -------- -------- -------- -------0			
0x0C4	IRQ44MON[B,H,W] -------- -------- -------- -------0			
0x0C8	IRQ45MON[B,H,W] -------- -------- -------- -------0			
0x0CC	IRQ46MON[B,H,W]			
	-------- -------- 00000000 00000000			
0x0D0	IRQ47MON[B,H,W] -------- -------- ----0--- --------			
0x0D4 - 0x1FC	-	-	-	-
0x200	DRQSEL1[B,H,W] -------- -------- -------- ---00000			
0x204	DQESEL[B,H,W] 00000000 00000000 00000000 00000000			
0x208	*			
0x20C	ODDPKS[B] ---00000	-	-	*
0x210	RCINTSEL3[B,H,W] ---00000	RCINTSEL2[B,H,W] ---00000	RCINTSEL1[B,H,W] ---00000	RCINTSEL0[B,H,W] ---00000
0x214	RCINTSEL7[B,H,W] ---00000	RCINTSEL6[B,H,W] ---00000	RCINTSEL5[B,H,W] ---00000	RCINTSEL4[B,H,W] ---00000
0x218 - 0xFFC	-	-	-	-

■ TYPE3/TYPE7 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	*			
0x004	*			
0x008	*			
0x00C	-	-	-	-
0x010	EXC02MON[B,H,W] -------- -------- -------- ------00			
0x014	IRQ00MON[B,H,W] -------- -------- -------- -------0			
0x018	IRQ01MON[B,H,W] -------- -------- -------- -------0			
0x01C	IRQ02MON[B,H,W] -------- -------- -------- -------0			
0x020	IRQ03MON[B,H,W] -------- -------- -------- ----0000			
0x024	IRQ04MON[B,H,W]			
	-------- -------- -------- -0000000			
0x028	IRQ05MON[B,H,W] -------- -------- -------- 0-------			
0x02C	IRQ06MON[B,H,W] -------- -------- -------- -------0			
0x030	IRQ07MON[B,H,W] -------- -------- -------- ------00			
0x034	IRQ08MON[B,H,W] -------- -------- -------- -------0			
0x038	IRQ09MON[B,H,W] -------- -------- -------- ------00			
0x03C	IRQ10MON[B,H,W] -------- -------- -------- -------0			
0x040	IRQ11MON[B,H,W] -------- -------- -------- ------00			
0x044	IRQ12MON[B,H,W] -------- -------- -------- -------0			

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x048	IRQ13MON[B,H,W] -------- -------- -------- ------00			
0x04C	IRQ14MON[B,H,W] -------- -------- -------- -------0			
0x050	IRQ15MON[B,H,W] -------- -------- -------- ------00			
0x054	IRQ16MON[B,H,W] -------- -------- -------- -------0			
0x058	IRQ17MON[B,H,W] -------- -------- -------- ------00			
0x05C	IRQ18MON[B,H,W] -------- -------- -------- -------0			
0x060	IRQ19MON[B,H,W] -------- -------- -------- ------00			
0x064	IRQ20MON[B,H,W] -------- -------- -------- ------0			
0x068	IRQ21MON[B,H,W] -------- -------- -------- ------00			
0x06C	IRQ22MON[B,H,W]			
	-------- -------- -------- -----000			
0x070	IRQ23MON[B,H,W] -------- -------- -------- --0--000			
0x074	IRQ24MON[B,H,W] -------- -------- -------- ----0000			
0x078	IRQ25MON[B,H,W] -------- -------- -------- --000000			
0x07C	IRQ26MON[B,H,W] -------- -------- -------- ----0000			
0x080	IRQ27MON[B,H,W] -------- -------- -------- --000000			
0x084	IRQ28MON[B,H,W] -------- -------- 00000000 00000000			

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x088	IRQ29MON[B,H,W] -------- -------- -------- ---0----			
0x08C	IRQ30MON[B,H,W] -------- -------- -------- --0-----			
0x090	IRQ31MON[B,H,W] -------- -------- -------- -0------			

1.21. LCDC

Base_Address : 0x4003_2000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	LCDCC3[B,H,W] 0011111-	LCDCC2[B,H,W] --010100	LCDCC1[B,H,W] -00000--
0x004	LCDC_PSR[B,H,W] -------- -0000000 00000000 00000000			
0x008	LCDC_COMEN[B,H,W] -------- -------- -------- 00000000			
0x00C	LCDC_SEGEN1[B,H,W] 00000000 00000000 00000000 00000000			
0x010	LCDC_SEGEN2[B,H,W] -------- -------- -------- 00000000			
0x014	-	-	LCDC_BLINK[B,H,W] 00000000 00000000	
0x018	-	-	-	-
0x01C	LCDRAM03[B,H,W] 00000000	LCDRAM02[B,H,W] 00000000	LCDRAM01[B,H,W] 00000000	LCDRAM00[B,H,W] 00000000
0x020	LCDRAM07[B,H,W] 00000000	LCDRAM06[B,H,W] 00000000	LCDRAM05[B,H,W] 00000000	LCDRAM04[B,H,W] 00000000
0x024	LCDRAM11[B,H,W] 00000000	LCDRAM10[B,H,W] 00000000	LCDRAM09[B,H,W] 00000000	LCDRAM08[B,H,W] 00000000
0x028	LCDRAM15[B,H,W] 00000000	LCDRAM14[B,H,W] 00000000	LCDRAM13[B,H,W] 00000000	LCDRAM12[B,H,W] 00000000
0x02C	LCDRAM19[B,H,W] 00000000	LCDRAM18[B,H,W] 00000000	LCDRAM17[B,H,W] 00000000	LCDRAM16[B,H,W] 00000000
0x030	LCDRAM23[B,H,W] 00000000	LCDRAM22[B,H,W] 00000000	LCDRAM21[B,H,W] 00000000	LCDRAM20[B,H,W] 00000000
0x034	LCDRAM27[B,H,W] 00000000	LCDRAM26[B,H,W] 00000000	LCDRAM25[B,H,W] 00000000	LCDRAM24[B,H,W] 00000000
0x038	LCDRAM31[B,H,W] 00000000	LCDRAM30[B,H,W] 00000000	LCDRAM29[B,H,W] 00000000	LCDRAM28[B,H,W] 00000000
0x03C	LCDRAM35[B,H,W] 00000000	LCDRAM34[B,H,W] 00000000	LCDRAM33[B,H,W] 00000000	LCDRAM32[B,H,W] 00000000
0x040	LCDRAM39[B,H,W] 00000000	LCDRAM38[B,H,W] 00000000	LCDRAM37[B,H,W] 00000000	LCDRAM36[B,H,W] 00000000
0x044 - 0x0FC	-	-	-	-

1.22. GPIO

Base_Address : 0x4003_3000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	PFR0[B,H,W] ---- ---- ---- ---- 0000 0000 0001 1111			
0x004	PFR1[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x008	PFR2[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x00C	PFR3[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x010	PFR4[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x014	PFR5[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x018	PFR6[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x01C	PFR7[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x020	PFR8[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x024	PFR9[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x028	PFRA[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x02C	PFRB[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x030	PFRC[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x034	PFRD[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x038	PFRE[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x03C	PFRF[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x040 - 0x0FC	-	-	-	-

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x100	PCR0[B,H,W] ---- ---- ---- ---- 0000 0000 0001 1111			
0x104	PCR1[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x108	PCR2[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x10C	PCR3[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x110	PCR4[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x114	PCR5[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x118	PCR6[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x11C	PCR7[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x120	PCRB[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x124	PCR9[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x128	PCRA[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x12C	PCRB[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x130	PCRC[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x134	PCRD[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x138	PCRE[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x13C	PCRF[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x140 - 0x1FC	-			
0x200	DDR0[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x204	DDR1[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x208	DDR2[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x20C	DDR3[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x210	DDR4[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x214	DDR5[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x218	DDR6[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x21C	DDR7[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x220	DDR8[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x224	DDR9[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x228	DDRA[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x22C	DDRB[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x230	DDRC[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x234	DDRD[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x238	DDRE[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x23C	DDRF[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x240 - 0x2FC	-	-	-	-
0x300	PDIR0[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x304	PDIR1[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x308	PDIR2[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x30C	PDIR3[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x310	PDIR4[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x314	PDIR5[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x318	PDIR6[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x31C	PDIR7[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x320	PDIR8[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x324	PDIR9[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x328	PDIRA[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x32C	PDIRB[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x330	PDIRC[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x334	PDIRD[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x338	PDIRE[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x33C	PDIRF[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x340 - 0x3FC	-	-	-	-
0x400	PDOR0[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x404	PDOR1[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x408	PDOR2[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x40C	PDOR3[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x410	PDOR4[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x414	PDOR5[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x418	PDOR6[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x41C	PDOR7[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x420	PDOR8[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x424	PDOR9[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x428	PDORA[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x42C	PDORB[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x430	PDORC[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x434	PDORD[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x438	PDORE[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x43C	PDORF[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x440 - 0x4FC	-	-	-	-
0x500	ADE[B,H,W] 1111 1111 1111 1111 1111 1111 1111 1111			
0x504 - 0x57C	-	-	-	-
0x580	SPSR[B,H,W] ---- ---- ---- ---- ---- ---- ---0 ---1 *1 ---- ---- ---- ---- ---- ---- ---0 0101 *2			
0x584 - 0x5FC	-	-	-	-

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x600	EPFR00[B,H,W] ---- --00 ---- --11 --0- --0- 0000 --00			
0x604	EPFR01[B,H,W] 0000 0000 0000 0000 ---0 0000 0000 0000			
0x608	EPFR02[B,H,W] 0000 0000 0000 0000 ---0 0000 0000 0000			
0x60C	EPFR03[B,H,W] 0000 0000 0000 0000 ---0 0000 0000 0000			
0x610	EPFR04[B,H,W] --00 0000 --00 00-- --00 0000 -000 00--			
0x614	EPFR05[B,H,W] --00 0000 --00 00-- --00 0000 --00 00--			
0x618	EPFR06[B,H,W] 0000 0000 0000 0000 0000 0000 0000 0000			
0x61C	EPFR07[B,H,W] ---- 0000 0000 0000 0000 0000 0000 ----			
0x620	EPFR08[B,H,W] ---- 0000 0000 0000 0000 0000 0000 0000			
0x624	EPFR09[B,H,W] 0000 0000 0000 0000 0000 0000 0000 0000			
0x628	EPFR10[B,H,W] 0000 0000 0000 0000 0000 0000 0000 0000			
0x62C	EPFR11[B,H,W] ---- --00 0000 0000 0000 0000 0000 0000			
0x630	EPFR12[B,H,W] --00 0000 --00 00-- --00 0000 --00 00--			
0x634	EPFR13[B,H,W] --00 0000 --00 00-- --00 0000 --00 00--			
0x638	EPFR14[B,H,W] 0000 0000 0000 0000 0000 0000 0000 0000			
0x63C	EPFR15[B,H,W] 0000 0000 0000 0000 0000 0000 0000 0000			
0x640	EPFR16[B,H,W] ---- 0000 0000 0000 0000 0000 0000 ----			

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x644	EPFR17[B,H,W] ---- 0000 0000 0000 0000 0000 0000 ----			
0x648	EPFR18[B,H,W] ---- ---- ---- ---- ---- ---- ---- 0000			
0x64C - 0x6FC	-	-	-	-
0x700	PZR0[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x704	PZR1[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x708	PZR2[B,H,W]			
	---- ---- ---- ---- 0000 0000 0000 0000			
0x70C	PZR3[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x710	PZR4[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x714	PZR5[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x718	PZR6[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x71C	PZR7[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x720	PZR8[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x724	PZR9[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x728	PZRA[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x72C	PZRB[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x730	PZRC[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x734	PZRD[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x738	PZRE[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x73C	PZRF[B,H,W] ---- ---- ---- ---- 0000 0000 0000 0000			
0x740 - 0x7FC	-	-	-	-
0x800	*			
0x804	*			
0x808 - 0xFFC	-	-	-	-

1.23. HDMI-CEC/Remote Control Receiver

ch.0	Base_Address : 0x4003_4000
ch.1	Base_Address : 0x4003_4100

Base_Address + Address	Register			
	+3	+2	+1	+0
0x000	-	-	-	TXCTRL[B,H,W] --0000-0
0x004	-	-	-	TXDATA[B,H,W] 00000000
0x008	-	-	-	TXSTS[B,H,W] --00---0
0x00C	-	-	-	SFREE[B,H,W] ----0000
0x010 - 0x03F	-	-	-	-
0x040	-	-	RCCR[B,H,W] 0---0000	RCST[B,H,W] 00000000
0x044	-	-	RCSHW[B,H,W] 00000000	RCDAHW[B,H,W] 00000000
0x048	-	-	RCDBHW[B,H,W] 00000000	-
0x04C	-	-	RCADR1[B,H,W] ---00000	RCADR2[B,H,W] ---00000
0x050	-	-	RCDTHH[B,H,W] 00000000	RCDTHL[B,H,W] 00000000
0x054	-	-	RCDTLH[B,H,W] 00000000	RCDTLL[B,H,W] 00000000
0x058	-	-	RCCKD[H,W] ---00000 00000000	
0x05C	-	-	RCRC[B,H,W] ---0---0	RCRHW[B,H,W] 00000000
0x060	-	-	RCLE[B,H,W] 00000-00	-
0x064	-	-	RCLELW[B,H,W] 00000000	RCLESW[B,H,W] 00000000
0x068 - 0x0FC	-	-	-	-

1.24. LVD

Base_Address : 0x4003_5000

■ TYPE0/TYPE1/TYPE2/TYPE4/TYPE5 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	LVD_CTL [B,H,W] 010000--
0x004	-	-	-	LVD_STR [B,H,W] 0-------
0x008	-	-	-	LVD_CLR [B,H,W] 1-------
0x00C	LVD_RLR[W] 00000000 00000000 00000000 00000001			
0x010	-	-	-	LVD_STR2 0-------
0x014 - 0xFFC	-	-	-	-

■ TYPE3, and TYPE6 to TYPE12 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	LVD_CTL[B, H, W] 1-0001-- 0-00000- *6 100001-- 000100-- *7 100001-- 000011-- *8	
0x004	-	-	-	LVD_STR[B,H,W] 0-------
0x008	-	-	-	LVD_CLR[B,H,W] 1-------
0x00C	LVD_RLR[W] 00000000 00000000 00000000 00000001			
0x010	-	-	-	LVD_STR2 01------
0x014 - 0x7FC	-	-	-	-

1.25. DS_Mode

Base_Address : 0x4003_5100

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	REG_CTL[B,H,W] -------0
0x004	-	-	-	RCK_CTL[B,H,W] ------01
0x008 - 0x6FC	-	-	-	-
0x700	-	-	-	PMD_CTL[B,H,W] -------0
0x704	-	-	-	WRFSR[B,H,W] ------00
0x708	-	-	WIFSR[B,H,W] ------00 00000000	
0x70C	-	-	WIER[B,H,W] ------00 00000-00	
0x710	-	-	-	WILVR[B,H,W] -----000
0x714	-	-	-	DSRAMR[B,H,W] ------00
0x718 - 0x7FC	-	-	-	-
0x800	BUR04[B,H,W] 00000000	BUR03[B,H,W] 00000000	BUR02[B,H,W] 00000000	BUR01[B,H,W] 00000000
0x804	BUR08[B,H,W] 00000000	BUR07[B,H,W] 00000000	BUR06[B,H,W] 00000000	BUR05[B,H,W] 00000000
0x808	BUR012[B,H,W] 00000000	BUR11[B,H,W] 00000000	BUR10[B,H,W] 00000000	BUR09[B,H,W] 00000000
0x80C	BUR16[B,H,W] 00000000	BUR15[B,H,W] 00000000	BUR14[B,H,W] 00000000	BUR13[B,H,W] 00000000
0x810 - 0xEFC	-	-	-	-

1.26. USB Clock

Base_Address : 0x4003_6000

■ Products other than TYPE2

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	UCCR[B,H,W] ------00
0x004	-	-	-	UPCR1[B,H,W] ------00
0x008	-	-	-	UPCR2[B,H,W] -----000
0x00C	-	-	-	UPCR3[B,H,W] ---00000
0x010	-	-	-	UPCR4[B,H,W] ---10111 *1 -0111011 *2
0x014	-	-	-	UP_STR[B,H,W] -------0
0x018	-	-	-	UPINT_ENR[B,H,W] -------0
0x01C	-	-	-	UPINT_CLR[B,H,W] -------0
0x020	-	-	-	UPINT_STR[B,H,W] -------0
0x024	-	-	-	UPCR5[B,H,W] ----0100
0x028 - 0x02C	-	-	-	-
0x030	-	-	-	USBEN[B,H,W] -------0
0x034 - 0x0FC	-	-	-	-

■ TYPE2 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	UCCR[B,H,W] -0000000
0x004	-	-	-	UPCR1[B,H,W] ------00
0x008	-	-	-	UPCR2[B,H,W] -----000
0x00C	-	-	-	UPCR3[B,H,W] ---00000
0x010	-	-	-	UPCR4[B,H,W] -0111011
0x014	-	-	-	UP_STR[B,H,W] -------0
0x018	-	-	-	UPINT_ENR[B,H,W] -------0
0x01C	-	-	-	UPINT_CLR[B,H,W] -------0
0x020	-	-	-	UPINT_STR[B,H,W] -------0
0x024	-	-	-	UPCR5[B,H,W] ----0100
0x028	-	-	-	UPCR6[B,H,W] ----0010
0x02C	-	-	-	UPCR7[B,H,W] -------0
0x030	-	-	-	USBEN[B,H,W] -------0
0x034	-	-	-	USBEN1[B,H,W] -------0
0x038 - 0x0FC	-	-	-	-

1.27. CAN_Prescaler

Base_Address : 0x4003_7000

Base_Address + Address	Register			
	+3	+2	+1	+0
0x000	-	-	-	CANPRE[B,H,W] ----1011
0x004 - 0xFFC	-	-	-	-

1.28. MFS

■ Products other than TYPE8/TYPE12

ch.0	Base_Address : 0x4003_8000
ch.1	Base_Address : 0x4003_8100
ch.2	Base_Address : 0x4003_8200
ch.3	Base_Address : 0x4003_8300
ch.4	Base_Address : 0x4003_8400
ch.5	Base_Address : 0x4003_8500
ch.6	Base_Address : 0x4003_8600
ch.7	Base_Address : 0x4003_8700

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	SCR/ IBCR[B,H,W] 0--00000	SMR[B,H,W] 000-00-0
0x004	-	-	SSR[B,H,W] 0-000011	ESCR/ IBSR[B,H,W] 00000000
0x008	-	-	RDR/TDR[H,W] -------0 00000000	
0x00C	-	-	BGR1[B,H,W] 00000000	BGR0[B,H,W] 00000000
0x010	-	-	ISMK[B,H,W] --------	ISBA[B,H,W] --------
0x014	-	-	FCR1[B,H,W] ---00100	FCR0[B,H,W] -0000000
0x018	-	-	FBYTE2[B,H,W] 00000000	FBYTE1[B,H,W] 00000000
0x1C			EIBCR[B, H, W] --001100	-
0x020 - 0x0FC	-	-	-	-

MFS Noise Filter Control Base_Address : 0x4003_8800

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	I2CDNF[B,H,W] 00000000	
0x004 - 0x0FC	-	-	-	-

■ TYPE8/TYPE12 products

ch.0	Base_Address : 0x4003_8000
ch.1	Base_Address : 0x4003_8100
ch.2	Base_Address : 0x4003_8200
ch.3	Base_Address : 0x4003_8300
ch.4	Base_Address : 0x4003_8400
ch.5	Base_Address : 0x4003_8500
ch.6	Base_Address : 0x4003_8600
ch.7	Base_Address : 0x4003_8700
ch.8	Base_Address : 0x4003_8800
ch.9	Base_Address : 0x4003_8900
ch.10	Base_Address : 0x4003_8A00
ch.11	Base_Address : 0x4003_8B00
ch.12	Base_Address : 0x4003_8C00
ch.13	Base_Address : 0x4003_8D00
ch.14	Base_Address : 0x4003_8E00
ch.15	Base_Address : 0x4003_8F00

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	SCR/ IBCR[B,H,W] 0--00000	SMR[B,H,W] 00-000-0
0x004	-	-	SSR[B,H,W] 0-000011	ESCR/ IBSR[B,H,W] 00000000
0x008	-	-	RDR/TDR[H,W] -------0 00000000	
0x00C	-	-	BGR1[B,H,W] 00000000	BGR0[B,H,W] 00000000
0x010	-	-	ISMK[B,H,W] --------	ISBA[B,H,W] --------
0x014	-	-	FCR1[B,H,W] ---00100	FCR0[B,H,W] -0000000
0x018	-	-	FBYTE2[B,H,W] 00000000	FBYTE1[B,H,W] 00000000
0x1C			EIBCR[B, H, W --001000	-
0x020 - 0x0FC	-	-	-	-

1.29. CRC

Base_Address : 0x4003_9000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	CRCCR[B,H,W] -0000000
0x004	CRCINIT[B,H,W] 11111111 11111111 11111111 11111111			
0x008	CRCIN[B,H,W] 00000000 00000000 00000000 00000000			
0x00C	CRCR[B,H,W] 11111111 11111111 11111111 11111111			

1.30. Watch Counter

Base_Address : 0x4003_A000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	WCCR[B,H,W] 00--0000	WCRL[B,H,W] --000000	WCRD[B,H,W] --000000
0x004 - 0x00C	-	-	-	-
0x010	-	-	CLK_SEL[B,H,W] -----000 -------0	
0x014	-	-	-	CLK_EN[B,H,W] ------00
0x018 - 0xFFC	-	-	-	-

1.31. RTC

Base_Address : 0x4003_B000

■ TYPE3/TYPE4/TYPE5 products

Base_Address + Address	Register			
	+3	+2	+1	+0
0x000	WTCR1[B,H,W] 00000000 00000000 ---00000 -00000-0			
0x004	WTCR2[B,H,W] -------- -------- -----000 -------0			
0x008	WTBR[B,H,W] -------- 00000000 00000000 00000000			
0x00C	WTDR[B,H,W] --000000	WTHR[B,H,W] --000000	WTMIR[B,H,W] -0000000	WTSR[B,H,W] -0000000
0x010	-	WTYR[B,H,W] 00000000	WTMOR[B,H,W] ---00000	WTDW[B,H,W] -----000
0x014	ALDR[B,H,W] --000000	ALHR[B,H,W] --000000	ALMIR[B,H,W] -0000000	-
0x018	-	ALYR[B,H,W] 00000000	ALMOR[B,H,W] ---00000	-
0x01C	WTTR[B,H,W] -------- ------00 00000000 00000000			
0x020	-	-	WTCLKM[B,H,W] ------00	WTCLKS [B,H,W] -------0
0x024	-	-	WTCALEN[B,H,W] -------0	WTCAL [B,H,W] -0000000
0x028	-	-	WTDIVEN[B,H,W] ------00	WTDIV [B,H,W] ----0000
0x02C - 0xFFC	-	-	-	-

■ TYPE6 to TYPE12 products

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	WTCR1[B,H,W] 00000000 00000000 ---00000 -00000-0			
0x004	WTCR2[B,H,W] -------- -------- -----000 -------0			
0x008	WTBR[B,H,W] -------- 00000000 00000000 00000000			
0x00C	WTDR[B,H,W] --000000	WTHR[B,H,W] --000000	WTMIR[B,H,W] -0000000	WTSR[B,H,W] -0000000
0x010	-	WTYR[B,H,W] 00000000	WTMOR[B,H,W] ---00000	WTDW[B,H,W] -----000
0x014	ALDR[B,H,W] --000000	ALHR[B,H,W] --000000	ALMIR[B,H,W] -0000000	-
0x018	-	ALYR[B,H,W] 00000000	ALMOR[B,H,W] ---00000	-
0x01C	WTTR[B,H,W] -------- ------00 00000000 00000000			
0x020	-	-	WTCLKM[B,H,W] ------00	WTCLKS [B,H,W] -------0
0x024	-	WTCALEN[B,H,W] -------0	WTCAL [B,H,W] ------00 00000000	
0x028	-	-	WTDIVEN[B,H,W] ------00	WTDIV [B,H,W] ----0000
0x02C	-	-	-	WTCALPRD [B,H,W] --010011
0x030	-	-	-	WTCOSEL [B,H,W] -------0
0x034 - 0xFFC	-	-	-	-

1.32. Low-speed CR Prescaler

Base_Address : 0x4003_B000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	-	LCR_PRSLD[B,H,W] --010011
0x004 - 0xFFC	-	-	-	-

1.33. EXT-Bus I/F

Base_Address : 0x4003_F000

<table>
<tr><th rowspan="2">Base_Address
+ Address</th><th colspan="4">Register</th></tr>
<tr><th>+3</th><th>+2</th><th>+1</th><th>+0</th></tr>
<tr><td>0x000</td><td colspan="4">MODE0[W]
-------- -------- --000-00 00000000</td></tr>
<tr><td>0x004</td><td colspan="4">MODE1[W]
-------- -------- --000-00 00000000</td></tr>
<tr><td>0x008</td><td colspan="4">MODE2[W]
-------- -------- --000-00 00000000</td></tr>
<tr><td>0x00C</td><td colspan="4">MODE3[W]
-------- -------- --000-00 00000000</td></tr>
<tr><td>0x010</td><td colspan="4">MODE4[W]
-------- -------- --000-00 00000001</td></tr>
<tr><td>0x014</td><td colspan="4">MODE5[W]
-------- -------- --000-00 00000000</td></tr>
<tr><td>0x018</td><td colspan="4">MODE6[W]
-------- -------- --000-00 00000000</td></tr>
<tr><td>0x01C</td><td colspan="4">MODE7[W]
-------- -------- --000-00 00000000</td></tr>
<tr><td>0x020</td><td colspan="4">TIM0[W]
00000101 01011111 11110000 00001111</td></tr>
<tr><td>0x024</td><td colspan="4">TIM1[W]
00000101 01011111 11110000 00001111</td></tr>
<tr><td>0x028</td><td colspan="4">TIM2[W]
00000101 01011111 11110000 00001111</td></tr>
<tr><td>0x02C</td><td colspan="4">TIM3[W]
00000101 01011111 11110000 00001111</td></tr>
<tr><td>0x030</td><td colspan="4">TIM4[W]
00000101 01011111 11110000 00001111</td></tr>
<tr><td>0x034</td><td colspan="4">TIM5[W]
00000101 01011111 11110000 00001111</td></tr>
<tr><td>0x038</td><td colspan="4">TIM6[W]
00000101 01011111 11110000 00001111</td></tr>
<tr><td>0x03C</td><td colspan="4">TIM7[W]
00000101 01011111 11110000 00001111</td></tr>
</table>

A. Register Map

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x040	AREA0[W] -------- -0001111 -------- 00000000			
0x044	AREA1[W] -------- -0001111 -------- 00010000			
0x048	AREA2[W] -------- -0001111 -------- 00100000			
0x04C	AREA3[W] -------- -0001111 -------- 00110000			
0x050	AREA4[W] -------- -0001111 -------- 01000000			
0x054	AREA5[W] -------- -0001111 -------- 01010000			
0x058	AREA6[W] -------- -0001111 -------- 01100000			
0x05C	AREA7[W] -------- -0001111 -------- 01110000			
0x060	ATIM0[W] -------- -------- ----0100 01011111			
0x064	ATIM1[W] -------- -------- ----0100 01011111			
0x068	ATIM2[W] -------- -------- ----0100 01011111			
0x06C	ATIM3[W] -------- -------- ----0100 01011111			
0x070	ATIM4[W] -------- -------- ----0100 01011111			
0x074	ATIM5[W] -------- -------- ----0100 01011111			
0x078	ATIM6[W] -------- -------- ----0100 01011111			
0x07C	ATIM7[W] -------- -------- ----0100 01011111			
0x080 - 0x2FC	-	-	-	-
0x300	DCLKR[W] -------- -------- -------- ---00001			
0x304 - 0x3FC	-	-	-	-

1.34. USB

ch.0 Base_Address : 0x4004_2100
ch.1 Base_Address : 0x4005_2100

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	-	-	HCNT1[B,H,W] -----001	HCNT0[B,H,W] 00000000
0x004	-	-	HERR[B,H,W] 00000011	HIRQ[B,H,W] 0-000000
0x008	-	-	HFCOMP[B,H,W] 00000000	HSTATE[B,H,W] --010010
0x00C	-	-	HRTIMER(1/0)[B,H,W] 00000000 00000000	
0x010	-	-	HADR[B,H,W] -0000000	HRTIMER(2)[B,H,W] -----00
0x014	-	-	HEOF(1/0)[B,H,W] --000000 00000000	
0x018	-	-	HFRAME(1/0)[B,H,W] -----000 00000000	
0x01C	-	-	-	HTOKEN [B,H,W] 00000000
0x020	-	-	UDCC[B,H,W] -------- 10100-00	
0x024	-	-	EP0C[H,W] ------0- -1000000	
0x028	-	-	EP1C[H,W] 01100001 00000000	
0x02C	-	-	EP2C[H,W] 0110000- -1000000	
0x030	-	-	EP3C[H,W] 0110000- -1000000	
0x034	-	-	EP4C[H,W] 0110000- -1000000	
0x038	-	-	EP5C[H,W] 0110000- -1000000	
0x03C	-	-	TMSP[H,W] -----000 00000000	

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x040	-	-	UDCIE[B,H,W] --000000	UDCS[B,H,W] --000000
0x044	-	-	EP0IS[H,W] 10---1-- --------	
0x048	-	-	EP0OS[H,W] 100--00- -XXXXXXX	
0x04C	-	-	EP1S[H,W] 100-000X XXXXXXXX	
0x050	-	-	EP2S[H,W] 100-000- -XXXXXXX	
0x054	-	-	EP3S[H,W] 100-000- -XXXXXXX	
0x058	-	-	EP4S[H,W] 100-000- -XXXXXXX	
0x05C	-	-	EP5S[H,W] 100-000- -XXXXXXX	
0x060	-	-	EP0DTH [B,H,W] XXXXXXXX	EP0DTL [B,H,W] XXXXXXXX
0x064	-	-	EP1DTH [B,H,W] XXXXXXXX	EP1DTL [B,H,W] XXXXXXXX
0x068	-	-	EP2DTH [B,H,W] XXXXXXXX	EP2DTL [B,H,W] XXXXXXXX
0x06C	-	-	EP3DTH [B,H,W] XXXXXXXX	EP3DTL [B,H,W] XXXXXXXX
0x070	-	-	EP4DTH [B,H,W] XXXXXXXX	EP4DTL [B,H,W] XXXXXXXX
0x074	-	-	EP5DTH [B,H,W] XXXXXXXX	EP5DTL [B,H,W] XXXXXXXX
0x078 - 0x07C	-	-	-	-

1.35. DMAC

Base_Address : 0x4006_0000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	DMACR[B,H,W] 00-00000 -------- -------- --------			
0x010	DMACA0[B,H,W] 00000000 0---0000 00000000 00000000			
0x014	DMACB0[B,H,W] --000000 00000000 00000000 -------0			
0x018	DMACSA0[B,H,W] 00000000 00000000 00000000 00000000			
0x01C	DMACDA0[B,H,W] 00000000 00000000 00000000 00000000			
0x020	DMACA1[B,H,W] 00000000 0---0000 00000000 00000000			
0x024	DMACB1[B,H,W] --000000 00000000 00000000 -------0			
0x028	DMACSA1[B,H,W] 00000000 00000000 00000000 00000000			
0x02C	DMACDA1[B,H,W] 00000000 00000000 00000000 00000000			
0x030	DMACA2[B,H,W] 00000000 0---0000 00000000 00000000			
0x034	DMACB2[B,H,W] --000000 00000000 00000000 -------0			
0x038	DMACSA2[B,H,W] 00000000 00000000 00000000 00000000			
0x03C	DMACDA2[B,H,W] 00000000 00000000 00000000 00000000			
0x040	DMACA3[B,H,W] 00000000 0---0000 00000000 00000000			
0x044	DMACB3[B,H,W] --000000 00000000 00000000 -------0			
0x048	DMACSA3[B,H,W] 00000000 00000000 00000000 00000000			
0x04C	DMACDA3[B,H,W] 00000000 00000000 00000000 00000000			

A. Register Map

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x050	DMACA4[B,H,W] 00000000 0---0000 00000000 00000000			
0x054	DMACB4[B,H,W] --000000 00000000 00000000 -------0			
0x058	DMACSA4[B,H,W] 00000000 00000000 00000000 00000000			
0x05C	DMACDA4[B,H,W] 00000000 00000000 00000000 00000000			
0x060	DMACA5[B,H,W] 00000000 0---0000 00000000 00000000			
0x064	DMACB5[B,H,W] --000000 00000000 00000000 -------0			
0x068	DMACSA5[B,H,W] 00000000 00000000 00000000 00000000			
0x06C	DMACDA5[B,H,W] 00000000 00000000 00000000 00000000			
0x070	DMACA6[B,H,W] 00000000 0---0000 00000000 00000000			
0x074	DMACB6[B,H,W] --000000 00000000 00000000 -------0			
0x078	DMACSA6[B,H,W] 00000000 00000000 00000000 00000000			
0x07C	DMACDA6[B,H,W] 00000000 00000000 00000000 00000000			
0x080	DMACA7[B,H,W] 00000000 0---0000 00000000 00000000			
0x084	DMACB7[B,H,W] --000000 00000000 00000000 -------0			
0x088	DMACSA7[B,H,W] 00000000 00000000 00000000 00000000			
0x08C	DMACDA7[B,H,W] 00000000 00000000 00000000 00000000			
0x090 - 0x0FC	-	-	-	-

1.36. CAN

ch.0 Base_Address : 0x4006_2000
ch.1 Base_Address : 0x4006_3000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	STATR[B,H,W] -------- 00000000		CTRLR[B,H,W] -------- 000-0001	
0x004	BTR[B,H,W] -0100011 00000001		ERRCNT[B,H,W] 00000000 00000000	
0x008	TESTR[B,H,W] -------- X00000--		INTR[B,H,W] 00000000 00000000	
0x00C	-	-	BRPER[B,H,W] -------- ----0000	
0x010	IF1CMSK[B,H,W] -------- 00000000		IF1CREQ[B,H,W] 0------- 00000001	
0x014	IF1MSK2[B,H,W] 11-11111 11111111		IF1MSK1[B,H,W] 11111111 11111111	
0x018	IF1ARB2[B,H,W] 00000000 00000000		IF1ARB1[B,H,W] 00000000 00000000	
0x01C	-	-	IF1MCTR[B,H,W] 00000000 0---0000	
0x020	IF1DTA2[B,H,W] 00000000 00000000		IF1DTA1[B,H,W] 00000000 00000000	
0x024	IF1DTB2[B,H,W] 00000000 00000000		IF1DTB1[B,H,W] 00000000 00000000	
0x028 - 0x02F	-	-	-	-
0x030	IF1DTA1[B,H,W] 00000000 00000000		IF1DTA2[B,H,W] 00000000 00000000	
0x034	IF1DTB1[B,H,W] 00000000 00000000		IF1DTB2[B,H,W] 00000000 00000000	
0x038 - 0x03C	-	-	-	-
0x040	IF2CMSK[B,H,W] -------- 00000000		IF2CREQ[B,H,W] 0------- 00000001	
0x044	IF2MSK2[B,H,W] 11-11111 11111111		IF2MSK1[B,H,W] 11111111 11111111	

A. Register Map

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x048	IF2ARB2[B,H,W] 00000000 00000000		IF2ARB1[B,H,W] 00000000 00000000	
0x04C	-	-	IF2MCTR[B,H,W] 00000000 0---0000	
0x050	IF2DTA2[B,H,W] 00000000 00000000		IF2DTA1[B,H,W] 00000000 00000000	
0x054	IF2DTB2[B,H,W] 00000000 00000000		IF2DTB1[B,H,W] 00000000 00000000	
0x058 - 0x05C	-	-	-	-
0x060	IF2DTA1[B,H,W] 00000000 00000000		IF2DTA2[B,H,W] 00000000 00000000	
0x064	IF2DTB1[B,H,W] 00000000 00000000		IF2DTB2[B,H,W] 00000000 00000000	
0x068 - 0x07C	-	-	-	-
0x080	TREQR2[B,H,W] 00000000 00000000		TREQR1[B,H,W] 00000000 00000000	
0x084 – 0x08F	-	-	-	-
0x090	NEWDT2[B,H,W] 00000000 00000000		NEWDT1[B,H,W] 00000000 00000000	
0x094 - 0x09F	-	-	-	-
0x0A0	INTPND2[B,H,W] 00000000 00000000		INTPND1[B,H,W] 00000000 00000000	
0x0A4 - 0x0AF	-	-	-	-
0x0B0	MSGVAL2[B,H,W] 00000000 00000000		MSGVAL1[B,H,W] 00000000 00000000	
0x0B4 - 0xFFC	-	-	-	-

1.37. Ether-MAC

ch.0 Base_Address : 0x4006_4000

ch.1 Base_Address : 0x4006_7000

<Note>

For the register details of Ether-MAC block, refer to the "Ethernet Part".

1.38. Ether-Control

Base_Address : 0x4006_6000

<Note>

For the register details of Ether-Control block, refer to the "Ethernet Part".

1.39. WorkFlash_IF

Base_Address : 0x200E_0000

Base_Address	Register			
+ Address	+3	+2	+1	+0
0x000	WFASZR[B,H,W]			
0x004	WFRWTR[B,H,W]			
0x008	WFSTR[B,H,W]			
0x00C - 0xFFF	-	-	-	-

<Note>

For the register details of Workflash IF block, refer to the "Flash Programming Manual" of the product used.

B. List of Notes

This section explains notes for each function.

1. Notes when high-speed CR is used for the master clock

CODE: 9BPRECAUTION-E01.3

1. Notes when high-speed CR is used for the master clock

This section explains notes when the high-speed CR is used for the master clock.

The frequency of the high-speed CR varies depending on the temperature and/or the power supply voltage.
The following table shows notes on each function macro when the high-speed CR is used for the master clock.
Furthermore, pay attention to notes when the high-speed CR is used as an input clock of the PLL and the master clock is selected for PLL.

● Notes on Each Macro

Macro	Function/mode	Notes
Base Clock	HCLK/FCLK	The maximum frequency of the high-speed CR shall not exceed the upper limit of the internal operation clock frequency specified in the "Data Sheet" of the product used.
Timer	Multi-function Timer Base Timer Watch Timer Dual Timer Watch Dog Timer Quadrature	The frequency variation of the high-speed CR should be considered for the timer count value of each macro.
A/D Converter	Sampling Time Compare Tim	Considering the frequency variation of the high-speed CR, the sampling time and the compare time of the A/D converter shall satisfy the specification specified in the "Data Sheet" of the product used.
USB	-	As the frequency accuracy does not meet the required specification, these macros cannot be used when the high-speed CR is used for the master clock.
Ethernet-MAC		
CAN		
Multi Function Serial Interface	UART	Even if the frequency of the high-speed CR is the minimum or the maximum value, the baud rate error should be considered. The baud rate error shall not exceed the limit.
	CSIO	The frequency variation of the high-speed CR should be considered for the communication of each macro.
	I2C	
	LIN	As the required frequency accuracy cannot be met, this function cannot be used as master. As slave, this function can be used. As a slave, the specified baud rate has more error at the maximum/minimum frequency of high-speed clock. So, if the error limit of the baud rate is exceeded, this function cannot be used.
Debug Interface	Serial Wire	As the frequency variation of the high-speed CR, the SWV(Serial Wire View) may not be used.
Flash Memory	Serial Write	The serial write cannot be supported for TYPE0, TYPE1, TYPE2, and TYPE4 products When the serial write is required, the clock should be supplied to the X0/X1pins.
External Bus Interface	Clock Output	When the external bus clock output is used, the frequency variation of the high-speed CR should be considered for devices to be connected.

C. List of Limitations

This section shows the differences between series.

1. List of Limitations for TYPE0 Products
2. List of Limitations for TYPE1 Products

CODE: 9BLIMITIONS-E02.0

1. List of Limitations for TYPE0 Products

This section shows the differences in the MB9A100A Series, MB9B500A/400A/300A/100A Series, MB9A100 Series and MB9B500/400/300/100 Series in a table.

The "Items" in the table are as written in this manual.

Item	Details
Timer Part 1.6.7 Hardware Watchdog Timer Load Register (WDG_LDR)	Following restrictions should be added to the <Notes> of "6.7. Hardware Watchdog Timer Load Register". • If a value is written to WDG_LDR again during the reloading period of the Hardware watchdog timer * (low-speed CR 4 cycle period after reloading the counter), the writing operation is ignored. Read the software of the appropriate register to check whether the writing value have been reflected to WDG_LDR properly. * The condition of counter reloading 1. Clearing watchdog timer (Writing a value to WDG_ICL register) 2. Writing a value to WDG_LDR register
Timer Part 1.6.9 Hardware Watchdog Timer Control Register (WDG_CTL)	Following restrictions should be added to the <Notes> of "6.9. Hardware Watchdog Timer Control Register". After writing "0" to the INTEN (watchdog counter enable) bit of the WDG_CTL register, if "1" is written again within 2 cycles of the low-speed CR (50KHz to 150KHz), operation may resume without reloading the count value from WDG_LDR. When setting the INTEN bit to "1" again after setting it to "0", always ensure a period of 2 clock cycles of the low-speed CR before setting. Alternatively, clear the timer using the WDG_ICL register immediately after writing "1" to INTEN to execute a reload.
Timer Part 3-2 Watch Counter	Following restrictions should be added to "CHAPTER 3-2: Watch Counter". *These restrictions are only for MB9A100 Series and MB9B500/400/300/100 Series. In Sub timer mode or Low speed CR timer mode, when the watch counter with sub crystal oscillator is used, the count value would be delayed from the actual time at the returning from an interrupt, by lengthening the interval of the low speed CR×35 cycles (Typ 350μs) watch counter. In Sub sleep mode or Low speed CR sleep mode, the counter value is not delayed.

Item	Details
Analog Macro Part 1-3.5.13 Sampling Time Selection Register (ADSS)	Following restrictions should be added to "5.13. Sampling Time Selection Register". In this series, the sampling time set in the Sampling Time Setup Register (ADST1) cannot be used. Enable the sampling time set in the Sampling Time Setup Register (ADST0) only. Always write "0" to each bit of the Sampling Time Selection Register (ADSS0 to ADSS3).
Communication Macro Part 1-2.7.9 1-3.5.9 1-4.6.9 1-5.5.12 FIFO Byte Register (FBYTE)	Following notes should be added to "7.9. FIFO Byte Register (FBYTE)" in chapter 1-2, "5.9. FIFO Byte Register (FBYTE)" in chapter 1-3, "6.9. FIFO Byte Register (FBYTE)" in chapter 1-4, "5.12. FIFO Byte Register (FBYTE)" in chapter 1-5. · If all the following conditions are met, the receive data full flag (SSR:RDRF) is not set to "1" despite the valid data of the number of FBYTE settings in the receive FIFO. If the setting value of FBYTE is "2" or more, this operation is not applied. · The setting value of FBYTE is "1". · Both the number of valid data of receive FIFO and the number of FBYTE settings are "1". · The data in receive FIFO is read at the same time when the multi-function serial interface macro receives the data and the received data is written to receive FIFO. However, in case that one of the followings occurs later, the receive data full flag (SSR:RDRF) is set to "1". · Next data is received. · The receive time idle of 8-bit time or more is detected when the receive FIFO idle is enabled (FCR:FRIIE=1).
Communication Macro Part 3-1.2 ■ End-point configuration of the USB device	Following notes should be added to "■End-point configuration of USB device". USB device does not support ISO (isochronous transfer). Only Comb1 of setting combinations is valid.
Communication Macro Part 3-1.3.6 DMA transfer function	Following restrictions should be added to "■ Automatic data size transfer mode". In this series, if the IN direction Automatic data size transfer mode is used in the Short packet transfer, packet transfer may not start even after DMA transfer is finished. In addition, it is prohibited to set USB as both the transfer source and transfer destination. [Workaround] Transfer data using CPU.

<table>
<tr><th>Item</th><th>Details</th></tr>
<tr><td>Communication Macro
Part 3-1.3.7
NULL transfer function

Communication Macro
Part 3-1.5.3
EP1 to 5 Status Registers
(EP1C to EP5C)</td><td>The following description should be added as the NULL transfer mode restriction.

In this series, NULL transfer may not start after DMA transfer, even in the NULL transfer mode. Use this mode under the setting of EP1C to EP5C:NULE = "0".

[Workaround]
To perform the NULL transfer, firstly set DMAE = "0" and clear the DRQ bit without writing the buffer data.
See Notes of [bit10] DRQ bit in "23-1.5.9 EP1 to 5 Status Registers (EP1S to EP5S)".</td></tr>
<tr><td>Communication Macro
Part 3-1.5.3
EP1 to EP5 Control Register
(EP1C to EP5C)</td><td>[bit 14:13] TYPE: The following end-point transfer types are supported.<table><tr><th>TYPE</th><th>Operation mode</th></tr><tr><td>00</td><td>Setting is prohibited</td></tr><tr><td>01</td><td>Setting is prohibited</td></tr><tr><td>10</td><td>Bulk transfer</td></tr><tr><td>11</td><td>Interrupt transfer</td></tr></table></td></tr>
<tr><td>Communication Macro
Part 3-1.5.10
EP0 to EP5 Data Registers
(EP0DTH to EP5DTH/
EP0DTL to EP5DTL)</td><td>Following restrictions should be added to "5.10. EP0 to EP5 Data Registers".

In this series, an indefinite data is read if serial read access to the above register is performed on the AHB bus.

[Workaround]
Please make the software to prevent the serial read. In the programming using C language, unintended serial read access on AHB bus may occur because of the optimization by the compiler option etc. Please refer to "■ Reference 1" for the workaround.</td></tr>
</table>

2. List of Limitations for TYPE1 Products

This section shows the differences in the MB9A002 Series, MB9A310 Series, MB9A110 Series, in a table.

The "Items" in the table are as written in this manual.

Item	Details
Communication Macro Part 1-2.7.9 1-3.5.9 1-4.6.9 1-5.5.12 FIFO Byte Register (FBYTE)	Following notes should be added to "7.9. FIFO Byte Register (FBYTE)" in chapter 1-2, "5.9. FIFO Byte Register (FBYTE)" in chapter 1-3, "6.9. FIFO Byte Register (FBYTE)" in chapter 1-4, "5.12. FIFO Byte Register (FBYTE)" in chapter 1-5. · If all the following conditions are met, the receive data full flag (SSR:RDRF) is not set to "1" despite the valid data of number of FBYTE settings in the receive FIFO. If the setting value of FBYTE is "2" or more, this operation is not applied. · The setting value of FBYTE is "1". · Both the number of valid data of receive FIFO and the number of FBYTE settings are "1" · The data in receive FIFO is read at the same time when the multi-function serial interface macro receives the data and the received data is written to receive FIFO. However, in case that one of the followings occurs later, the receive data full flag (SSR:RDRF) is set to "1". · Next data is received. · The receive time idle of 8-bit time or more is detected when the receive FIFO idle is enabled (FCR:FRIIE=1).

■ Reference 1

Example: If the following C source codes are compiled, serial read access may occur because of the optimization by the compiler option etc.

```
void do_ep0o(void)
{
        int i;
        int length;
        unsigned int b0,b1,b2,b3;

        b0 = (unsigned int)IO_EP0DT;
        b1 = (unsigned int)IO_EP0DT;
        b2 = (unsigned int)IO_EP0DT;
        b3 = (unsigned int)IO_EP0DT;
        buffer[0] = (unsigned short)b0;
        buffer[1] = (unsigned short)b1;
        buffer[2] = (unsigned short)b2;
        buffer[3] = (unsigned short)b3;
}
```

The following is a workaround. (Execute processing in the following order)

```
void do_ep0o(void)
{
        int i;
        int length;
        volatile int b0;

        b0 = (unsigned int)IO_EP0DT;
        buffer[0] = (unsigned short)b0;
        b0 = (unsigned int)IO_EP0DT;
        buffer[1] = (unsigned short)b0;
        b0 = (unsigned int)IO_EP0DT;
        buffer[2] = (unsigned short)b0;
        b0 = (unsigned int)IO_EP0DT;
        buffer[3] = (unsigned short)b0;
}
```

D. Product TYPE List

This section describes the product TYPE.

CODE: 9xTYPE_LIST-E04.0

1. Product TYPE List

In this manual, the products are classified into the following groups and are described as follows.
For the descriptions such as "TYPE0", see the relevant items of the target product in the list below.

Table 1 TYPE0 product list

Description in this manual	Flash memory size			
	512 Kbytes	384 Kbytes	256 Kbytes	128 Kbytes
TYPE0	MB9BF506N MB9BF506R MB9BF506NA MB9BF506RA MB9BF506NB MB9BF506RB	MB9BF505N MB9BF505R MB9BF505NA MB9BF505RA MB9BF505NB MB9BF505RB	MB9BF504N MB9BF504R MB9BF504NA MB9BF504RA MB9BF504NB MB9BF504RB	-
	MB9BF406N MB9BF406R MB9BF406NA MB9BF406RA	MB9BF405N MB9BF405R MB9BF405NA MB9BF405RA	MB9BF404N MB9BF404R MB9BF404NA MB9BF404RA	-
	MB9BF306N MB9BF306R MB9BF306NA MB9BF306RA MB9BF306NB MB9BF306RB	MB9BF305N MB9BF305R MB9BF305NA MB9BF305RA MB9BF305NB MB9BF305RB	MB9BF304N MB9BF304R MB9BF304NA MB9BF304RA MB9BF304NB MB9BF304RB	-
	MB9BF106N MB9BF106R MB9BF106NA MB9BF106RA	MB9BF105N MB9BF105R MB9BF105NA MB9BF105RA	MB9BF104N MB9BF104R MB9BF104NA MB9BF104RA	MB9BF102N MB9BF102R MB9BF102NA MB9BF102RA
	-	MB9AF105N MB9AF105R MB9AF105NA MB9AF105RA	MB9AF104N MB9AF104R MB9AF104NA MB9AF104RA	MB9AF102N MB9AF102R MB9AF102NA MB9AF102RA

Table 2 TYPE1 product list

Description in this manual	Flash memory size				
	512 Kbytes	384 Kbytes	256 Kbytes	128 Kbytes	64 Kbytes
TYPE1	MB9AF316M MB9AF316N MB9AF316MA MB9AF316NA	MB9AF315M MB9AF315N MB9AF315MA MB9AF315NA	MB9AF314L MB9AF314M MB9AF314N MB9AF314LA MB9AF314MA MB9AF314NA	MB9AF312L MB9AF312M MB9AF312N MB9AF312LA MB9AF312MA MB9AF312NA	MB9AF311L MB9AF311M MB9AF311N MB9AF311LA MB9AF311MA MB9AF311NA
	MB9AF116M MB9AF116N MB9AF116MA MB9AF116NA	MB9AF115M MB9AF115N MB9AF115MA MB9AF115NA	MB9AF114L MB9AF114M MB9AF114N MB9AF114LA MB9AF114MA MB9AF114NA	MB9AF112L MB9AF112M MB9AF112N MB9AF112LA MB9AF112MA MB9AF112NA	MB9AF111L MB9AF111M MB9AF111N MB9AF111LA MB9AF111MA MB9AF111NA

Table 3 TYPE2 product list

Description in this manual	Flash memory size		
	1 Mbyte	768 Kbytes	512 Kbytes
TYPE2	MB9BFD18S MB9BFD18T	MB9BFD17S MB9BFD17T	MB9BFD16S MB9BFD16T
	MB9BF618S MB9BF618T	MB9BF617S MB9BF617T	MB9BF616S MB9BF616T
	MB9BF518S MB9BF518T	MB9BF517S MB9BF517T	MB9BF516S MB9BF516T
	MB9BF418S MB9BF418T	MB9BF417S MB9BF417T	MB9BF416S MB9BF416T
	MB9BF318S MB9BF318T	MB9BF317S MB9BF317T	MB9BF316S MB9BF316T
	MB9BF218S MB9BF218T	MB9BF217S MB9BF217T	MB9BF216S MB9BF216T
	MB9BF118S MB9BF118T	MB9BF117S MB9BF117T	MB9BF116S MB9BF116T

Table 4 TYPE3 product list

Description in this manual	Flash memory size	
	128 Kbytes	64 Kbytes
TYPE3	MB9AF132K MB9AF132L	MB9AF131K MB9AF131L
	MB9AF132KA MB9AF132LA	MB9AF131KA MB9AF131LA
	MB9AF132KB MB9AF132LB	MB9AF132KB MB9AF132LB

Table 5 TYPE4 product list

Description in this manual	Flash memory size			
	512 Kbytes	384 Kbytes	256 Kbytes	128 Kbytes
TYPE4	MB9BF516N MB9BF516R	MB9BF515N MB9BF515R	MB9BF514N MB9BF514R	MB9BF512N MB9BF512R
	MB9BF416N MB9BF416R	MB9BF415N MB9BF415R	MB9BF414N MB9BF414R	MB9BF412N MB9BF412R
	MB9BF316N MB9BF316R	MB9BF315N MB9BF315R	MB9BF314N MB9BF314R	MB9BF312N MB9BF312R
	MB9BF116N MB9BF116R	MB9BF115N MB9BF115R	MB9BF114N MB9BF114R	MB9BF112N MB9BF112R

Table 6 TYPE5 product list

Description in this manual	Flash memory size	
	128 Kbytes	64 Kbytes
TYPE5	MB9AF312K	MB9AF311K
	MB9AF112K	MB9AF111K

Table 7 TYPE6 product list

Description in this manual	Flash memory size		
	256 Kbytes	128 Kbytes	64 Kbytes
TYPE6	MB9AFB44L MB9AFB44M MB9AFB44N MB9AFB44LA MB9AFB44MA MB9AFB44NA MB9AFB44LB MB9AFB44MB MB9AFB44NB	MB9AFB42L MB9AFB42M MB9AFB42N MB9AFB42LA MB9AFB42MA MB9AFB42NA MB9AFB42LB MB9AFB42MB MB9AFB42NB	MB9AFB41L MB9AFB41M MB9AFB41N MB9AFB41LA MB9AFB41MA MB9AFB41NA MB9AFB41LB MB9AFB41MB MB9AFB41NB
	MB9AFA44L MB9AFA44M MB9AFA44N MB9AFA44LA MB9AFA44MA MB9AFA44NA MB9AFA44LB MB9AFA44MB MB9AFA44NB	MB9AFA42L MB9AFA42M MB9AFA42N MB9AFA42LA MB9AFA42MA MB9AFA42NA MB9AFA42LB MB9AFA42MB MB9AFA42NB	MB9AFA41L MB9AFA41M MB9AFA41N MB9AFA41LA MB9AFA41MA MB9AFA41NA MB9AFA41LB MB9AFA41MB MB9AFA41NB
	MB9AF344L MB9AF344M MB9AF344N MB9AF344LA MB9AF344MA MB9AF344NA MB9AF344LB MB9AF344MB MB9AF344NB	MB9AF342L MB9AF342M MB9AF342N MB9AF342LA MB9AF342MA MB9AF342NA MB9AF342LB MB9AF342MB MB9AF342NB	MB9AF341L MB9AF341M MB9AF341N MB9AF341LA MB9AF341MA MB9AF341NA MB9AF341LB MB9AF341MB MB9AF341NB
	MB9AF144L MB9AF144M MB9AF144N MB9AF144LA MB9AF144MA MB9AF144NA MB9AF144LB MB9AF144MB MB9AF144NB	MB9AF142L MB9AF142M MB9AF142N MB9AF142LA MB9AF142MA MB9AF142NA MB9AF142LB MB9AF142MB MB9AF142NB	MB9AF141L MB9AF141M MB9AF141N MB9AF141LA MB9AF141MA MB9AF141NA MB9AF141LB MB9AF141MB MB9AF141NB

Table 8 TYPE7 product list

Description in this manual	Flash memory size	
	128 Kbytes	64 Kbytes
TYPE7	MB9AFA32L MB9AFA32M MB9AFA32N	MB9AFA31L MB9AFA31M MB9AFA31N
	MB9AF132M MB9AF132N	MB9AF131M MB9AF131N
	MB9AFAA2L MB9AFAA2M MB9AFAA2N	MB9AFAA1L MB9AFAA1M MB9AFAA1N
	MB9AF1A2L MB9AF1A2M MB9AF1A2N	MB9AF1A1L MB9AF1A1M MB9AF1A1N

Table 9 TYPE8 product list

Description in this manual	Flash memory size		
	512 Kbytes	384 Kbytes	256 Kbytes
TYPE8	MB9AF156M MB9AF156N MB9AF156R MB9AF156MA MB9AF156NA MB9AF156RA MB9AF156MB MB9AF156NB MB9AF156RB	MB9AF155M MB9AF155N MB9AF155R MB9AF155MA MB9AF155NA MB9AF155RA MB9AF155MB MB9AF155NB MB9AF155RB	MB9AF154M MB9AF154N MB9AF154R MB9AF154MA MB9AF154NA MB9AF154RA MB9AF154MB MB9AF154NB MB9AF154RB

Table 10 TYPE9 product list

Description in this manual	Flash memory size		
	256 Kbytes	128 Kbytes	64 Kbytes
TYPE9	MB9BF524K MB9BF524L MB9BF524M	MB9BF522K MB9BF522L MB9BF522M	MB9BF521K MB9BF521L MB9BF521M
	MB9BF324K MB9BF324L MB9BF324M	MB9BF322K MB9BF322L MB9BF322M	MB9BF321K MB9BF321L MB9BF321M
	MB9BF124K MB9BF124L MB9BF124M	MB9BF122K MB9BF122L MB9BF122M	MB9BF121K MB9BF121L MB9BF121M

Table 11 TYPE10 product list

Description in this manual	Flash memory size
	64 Kbytes
TYPE10	MB9BF121J

Table 12 TYPE11 product list

Description in this manual	Flash memory size
	64 Kbytes
TYPE11	MB9AF421K MB9AF421L
	MB9AF121K MB9AF121L

Table 13 TYPE12 product list

Description in this manual	Flash memory size	
	1.5 Mbytes	1 Mbytes
TYPE12	MB9BF529S MB9BF529T MB9BF529SA MB9BF529TA	MB9BF528S MB9BF528T MB9BF528SA MB9BF528TA
	MB9BF429S MB9BF429T MB9BF429SA MB9BF429TA	MB9BF428S MB9BF428T MB9BF428SA MB9BF428TA
	MB9BF329S MB9BF329T MB9BF329SA MB9BF329TA	MB9BF328S MB9BF328T MB9BF328SA MB9BF328TA
	MB9BF129S MB9BF129T MB9BF129SA MB9BF129TA	MB9BF128S MB9BF128T MB9BF128SA MB9BF128TA