## TIMER ARRAY UNIT D (TAUD)

This section contains a generic description of the Timer Array Unit D (TAUD).

The first part of this section describes all specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of the TAUD.

### Features

#### Number of Units and Channels

This R-car S4 has the following number of TAUD units.

**Table 33.1 Number of Units**

|  |  |
| --- | --- |
| **Product Name** | **R-car S4** |
| Number of Units | 2 (n = 0 to 1) |
| Name | TAUDn |

TAUDn has the following timers for the quantity of channels of timers.

**Table 33.2 TAUDn Unit Configurations and Channels**

|  |  |  |
| --- | --- | --- |
| **Unit Name (Channel Name) TAUDn** | **Channels per Unit** | **R-car S4** |
| TAUD0 | 16 | 16 |
| TAUD1 | 16 | 16 |

**Table 33.3 Index**

|  |  |
| --- | --- |
| **Index** | **Description** |
| n | Throughout this section, the individual TAUD units are identified by the index “n” (n = 0 to 1) |
| m | The TAUD has 16 channels. Throughout this section, the individual channels are identified by the index “m” (m = 0 to 15), thus a certain channel is denoted as CHm.  The even numbered channels (m = 0, 2, 4, 6, 8, 10, 12, 14) are denoted as CHm\_even.  The odd numbered channels (m = 1, 3, 5, 7, 9, 11, 13, 15) are denoted as CHm\_odd |

#### Register Base Addresses

TAUDn base addresses are listed in the following table.

TAUDn register addresses are given as offsets from the base addresses.

**Table 33.4** Register Base Addresses

|  |  |  |
| --- | --- | --- |
| **Base Address Name** | **Base Address** | **Bus Group** |
| <TAUD0\_base> | FFBF 4000H | P-Bus Group 5 |
| <TAUD1\_base> | FFBF 5000H | P-Bus Group 5 |

#### Clock Supply

Clock supply by and to TAUDn is listed in the following table.

**Table 33.5 Clock Supply**

|  |  |  |
| --- | --- | --- |
| **Unit Name** | **Unit Clock Name** | **Supply Clock Name** |
| TAUDn | PCLK | Peripheral high speed clock CLK\_HSB |

#### Interrupt Requests and Error Notifications

TAUDn interrupt requests are listed in the following table.

**Table 33.6 Interrupt and DMA/DTS Requests**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Unit Interrupt Name** | **Description** | **Interrupt Number** | **DMATrigger Number** | **DTSTrigger Number** |
| **TAUD0** | | | | |
| INTTAUD0I0 | Channel 0 interrupt | 10 | Group 0-192 | Group 1-64 |
| INTTAUD0I1 | Channel 1 interrupt | 376 | Group 0-193 | Group 1-65 |
| INTTAUD0I2 | Channel 2 interrupt | 11 | Group 0-194 | Group 1-66 |
| INTTAUD0I3 | Channel 3 interrupt | 377 | Group 0-195 | Group 1-67 |
| INTTAUD0I4 | Channel 4 interrupt | 12 | Group 0-196 | Group 1-68 |
| INTTAUD0I5 | Channel 5 interrupt | 378 | Group 0-197 | Group 1-69 |
| INTTAUD0I6 | Channel 6 interrupt | 13 | Group 0-198 | Group 1-70 |
| INTTAUD0I7 | Channel 7 interrupt | 379 | Group 0-199 | Group 1-71 |
| INTTAUD0I8 | Channel 8 interrupt | 23 | Group 0-200 | Group 1-72 |
| INTTAUD0I9 | Channel 9 interrupt | 380 | Group 0-201 | Group 1-73 |
| INTTAUD0I10 | Channel 10 interrupt | 24 | Group 0-202 | Group 1-74 |
| INTTAUD0I11 | Channel 11 interrupt | 381 | Group 0-203 | Group 1-75 |
| INTTAUD0I12 | Channel 12 interrupt | 25 | Group 0-204 | Group 1-76 |
| INTTAUD0I13 | Channel 13 interrupt | 382 | Group 0-205 | Group 1-77 |
| INTTAUD0I14 | Channel 14 interrupt | 26 | Group 0-206 | Group 1-78 |
| INTTAUD0I15 | Channel 15 interrupt | 383 | Group 0-207 | Group 1-79 |
| **TAUD1** | | | | |
| INTTAUD1I0 | Channel 0 interrupt | 384 | Group 0-208 | Group 1-80 |
| INTTAUD1I1 | Channel 1 interrupt | 385 | Group 0-209 | Group 1-81 |
| INTTAUD1I2 | Channel 2 interrupt | 386 | Group 0-210 | Group 1-82 |
| INTTAUD1I3 | Channel 3 interrupt | 387 | Group 0-211 | Group 1-83 |
| INTTAUD1I4 | Channel 4 interrupt | 388 | Group 0-212 | Group 1-84 |
| INTTAUD1I5 | Channel 5 interrupt | 389 | Group 0-213 | Group 1-85 |
| INTTAUD1I6 | Channel 6 interrupt | 390 | Group 0-214 | Group 1-86 |
| INTTAUD1I7 | Channel 7 interrupt | 391 | Group 0-215 | Group 1-87 |
| INTTAUD1I8 | Channel 8 interrupt | 392 | Group 0-216 | Group 1-88 |
| INTTAUD1I9 | Channel 9 interrupt | 393 | Group 0-217 | Group 1-89 |
| INTTAUD1I10 | Channel 10 interrupt | 394 | Group 0-218 | Group 1-90 |
| INTTAUD1I11 | Channel 11 interrupt | 395 | Group 0-219 | Group 1-91 |
| INTTAUD1I12 | Channel 12 interrupt | 396 | Group 0-220 | Group 1-92 |
| INTTAUD1I13 | Channel 13 interrupt | 397 | Group 0-221 | Group 1-93 |
| INTTAUD1I14 | Channel 14 interrupt | 398 | Group 0-222 | Group 1-94 |
| INTTAUD1I15 | Channel 15 interrupt | 399 | Group 0-223 | Group 1-95 |

This module has no error notifications.

#### Reset Sources

TAUDn reset sources are listed in the following table. TAUDn is initialized by these reset sources.

**Table 33.7 Reset Sources**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Reset Condition** | | | | | | | |
| **Unit Name** | **Register Name** | **Power On Reset** | **System Reset 1** | **SystemReset 2** | **Deep STOP Reset** | **Module Reset** | **JTAG Reset** |
| TAUDn | All registers | √ | √ | √ | √ | √ | — |

#### External Input/Output Signals

External input/output signals of TAUDn are listed below.

**Table 33.8 External Input/Output Signals**

|  |  |  |  |
| --- | --- | --- | --- |
| **Unit Signal Name** | **I/O** | **Description** | **Alternative Port Pin Signal Name** |
| **TAUD0** | | | |
| TAUD0TTIN0 | I | Channel 0 Input\*1\*2 | TAUD0I0 |
| TAUD0TTIN1 | I | Channel 1 Input\*1\*2 | TAUD0I1 |
| TAUD0TTIN2 | I | Channel 2 Input\*1\*2 | TAUD0I2 |
| TAUD0TTIN3 | I | Channel 3 Input\*1\*2 | TAUD0I3 |
| TAUD0TTIN4 | I | Channel 4 Input\*1\*2 | TAUD0I4 |
| TAUD0TTIN5 | I | Channel 5 Input\*1\*2 | TAUD0I5 |
| TAUD0TTIN6 | I | Channel 6 Input\*1\*2 | TAUD0I6 |
| TAUD0TTIN7 | I | Channel 7 Input\*1\*2 | TAUD0I7 |
| TAUD0TTIN8 | I | Channel 8 Input\*1\*2 | TAUD0I8 |
| TAUD0TTIN9 | I | Channel 9 Input\*1\*2 | TAUD0I9 |
| TAUD0TTIN10 | I | Channel 10 Input\*1\*2 | TAUD0I10 |
| TAUD0TTIN11 | I | Channel 11 Input\*1\*2 | TAUD0I11 |
| TAUD0TTIN12 | I | Channel 12 Input\*1\*2 | TAUD0I12 |
| TAUD0TTIN13 | I | Channel 13 Input\*1\*2 | TAUD0I13 |
| TAUD0TTIN14 | I | Channel 14 Input\*1\*2 | TAUD0I14 |
| TAUD0TTIN15 | I | Channel 15 Input\*1\*2 | TAUD0I15 |
| TAUD0TTOUT0 | O | Channel 0 output | TAUD0O0 |
| TAUD0TTOUT1 | O | Channel 1 output | TAUD0O1 |
| TAUD0TTOUT2 | O | Channel 2 output | TAUD0O2 |
| TAUD0TTOUT3 | O | Channel 3 output | TAUD0O3 |
| TAUD0TTOUT4 | O | Channel 4 output | TAUD0O4 |
| TAUD0TTOUT5 | O | Channel 5 output | TAUD0O5 |
| TAUD0TTOUT6 | O | Channel 6 output | TAUD0O6 |
| TAUD0TTOUT7 | O | Channel 7 output | TAUD0O7 |
| TAUD0TTOUT8 | O | Channel 8 output | TAUD0O8 |
| TAUD0TTOUT9 | O | Channel 9 output | TAUD0O9 |
| TAUD0TTOUT10 | O | Channel 10 output | TAUD0O10 |
| TAUD0TTOUT11 | O | Channel 11 output | TAUD0O11 |
| TAUD0TTOUT12 | O | Channel 12 output | TAUD0O12 |
| TAUD0TTOUT13 | O | Channel 13 output | TAUD0O13 |
| TAUD0TTOUT14 | O | Channel 14 output | TAUD0O14 |
| TAUD0TTOUT15 | O | Channel 15 output | TAUD0O15 |
| **TAUD1** | | | |
| TAUD1TTIN0 | I | Channel 0 Input\*1\*2 | TAUD1I0 |
| TAUD1TTIN1 | I | Channel 1 Input\*1\*2 | TAUD1I1 |
| TAUD1TTIN2 | I | Channel 2 Input\*1\*2 | TAUD1I2 |
| TAUD1TTIN3 | I | Channel 3 Input\*1\*2 | TAUD1I3 |
| TAUD1TTIN4 | I | Channel 4 Input\*1\*2 | TAUD1I4 |
| TAUD1TTIN5 | I | Channel 5 Input\*1\*2 | TAUD1I5 |
| TAUD1TTIN6 | I | Channel 6 Input\*1\*2 | TAUD1I6 |
| TAUD1TTIN7 | I | Channel 7 Input\*1\*2 | TAUD1I7 |
| TAUD1TTIN8 | I | Channel 8 Input\*1\*2 | TAUD1I8 |
| TAUD1TTIN9 | I | Channel 9 Input\*1\*2 | TAUD1I9 |
| TAUD1TTIN10 | I | Channel 10 Input\*1\*2 | TAUD1I10 |
| TAUD1TTIN11 | I | Channel 11 Input\*1\*2 | TAUD1I11 |
| TAUD1TTIN12 | I | Channel 12 Input\*1\*2 | TAUD1I12 |
| TAUD1TTIN13 | I | Channel 13 Input\*1\*2 | TAUD1I13 |
| TAUD1TTIN14 | I | Channel 14 Input\*1\*2 | TAUD1I14 |
| TAUD1TTIN15 | I | Channel 15 Input\*1\*2 | TAUD1I15 |
| TAUD1TTOUT0 | O | Channel 0 output | TAUD1O0 |
| TAUD1TTOUT1 | O | Channel 1 output | TAUD1O1 |
| TAUD1TTOUT2 | O | Channel 2 output | TAUD1O2 |
| TAUD1TTOUT3 | O | Channel 3 output | TAUD1O3 |
| TAUD1TTOUT4 | O | Channel 4 output | TAUD1O4 |
| TAUD1TTOUT5 | O | Channel 5 output | TAUD1O5 |
| TAUD1TTOUT6 | O | Channel 6 output | TAUD1O6 |
| TAUD1TTOUT7 | O | Channel 7 output | TAUD1O7 |
| TAUD1TTOUT8 | O | Channel 8 output | TAUD1O8 |
| TAUD1TTOUT9 | O | Channel 9 output | TAUD1O9 |
| TAUD1TTOUT10 | O | Channel 10 output | TAUD1O10 |
| TAUD1TTOUT11 | O | Channel 11 output | TAUD1O11 |
| TAUD1TTOUT12 | O | Channel 12 output | TAUD1O12 |
| TAUD1TTOUT13 | O | Channel 13 output | TAUD1O13 |
| TAUD1TTOUT14 | O | Channel 14 output | TAUD1O14 |
| TAUD1TTOUT15 | O | Channel 15 output | TAUD1O15 |

**Note** 1. Setting of the noise filter for the port is required when the channel input pin is used. For details, **Section 2, Noise Filter & Edge/Level Detector.**

**Note** 2. The input signals can be switched by PIC. For details, see Section **41.2.3.3, Timer InputSelect Function.**

#### Internal Input/Output Signals

The internal input/output signals of TAUDn are listed below.

**Table 33.9 Internal Input/Output Signals**

|  |  |  |
| --- | --- | --- |
| **Unit Signal Name** | **Description** | **Connected to** |
| TAUDnTSSTm | Simultaneous channel start trigger input | PIC |

### Overview

#### Functional Overview

The TAUD has the following functions:

* 16 channels.
* 16-bit counter and 16-bit data register per channel.
* Independent channel operation.
* Synchronous channel operation (master and slave operation).
* Generation of different types of output signal.
* Real-time output.
* Counter can be triggered by external signal.
* Interrupt generation.

The Timer Array Unit D is used to perform various count or timer operations and to output a signal which depends on the result of the operation. It contains one prescaler block for count clock generation and 16 channels, each equipped with a 16-bit counter TAUDnCNTm and a 16-bit data register TAUDnCDRm to hold the start or compare value of the counter.

It also contains several control and status registers.

**Independent and synchronous operation**

Every channel can operate in different operation modes, either independently or in combination with other channels (synchronously). When one master channel and one or more slave channels operate in combination, the slave channels depend on the master channel.

When a channel is operated independently, its operation mode and functions are not affected by those of other channels. When a channel is operated synchronously it is either a master or a slave. A master channel can have multiple slaves, and the state of one channel affects that of the other channels. For example, this means that one channel can control when another starts to count, is reset, etc.

#### Terms

In this section, the following terms are used.

**Independent / synchronous channel operation**

Independent or synchronous channel operation describes the dependency of channels on each other:

* If a channel operates independently of all other channels, this is called independent channel operation.
* If a channel operates depending on other channels, this is called synchronouschannel operation.

**Channel group**

In synchronous channel operation, all channels that depend on each other are referred to as a “channel group”.

A channel group has one master channel and one or more slave channels.

**Operation mode**

An operation mode can be selected for every channel m. The operation mode defines the basic operation and features of a channel.

In synchronous channel operation, every channel in the channel group can operate in a different operation mode.

Examples are “Capture Mode”, “Event Count Mode”, and “Interval Timer Mode”.

**Channel output mode**

The channel output mode defines the operation of TAUDnTTOUTm

* of a single channel (independent output operation) or
* of all channels in a channel group (synchronous output operation).

Examples are “Independent Channel Output Mode 1” and “Synchronous Channel Output Mode 2 with Dead Time Output”.

**Channel operation function**

The channel operation function defines the complete function and all features

* of a single channel (independent channel operation) or
* of all channels in a channel group (synchronous channel operation).

**Upper / lower channel**

Depending on the channel number m, a channel with a smaller number or with a larger number is referred to as “upper” or “lower” channel, respectively.

* Upper channel: Channel with a smaller channel number
* Lower channel: Channel with a larger channel number Example:

For channel 5, channel 3 is an upper channel and channel 9 is a lower channel.

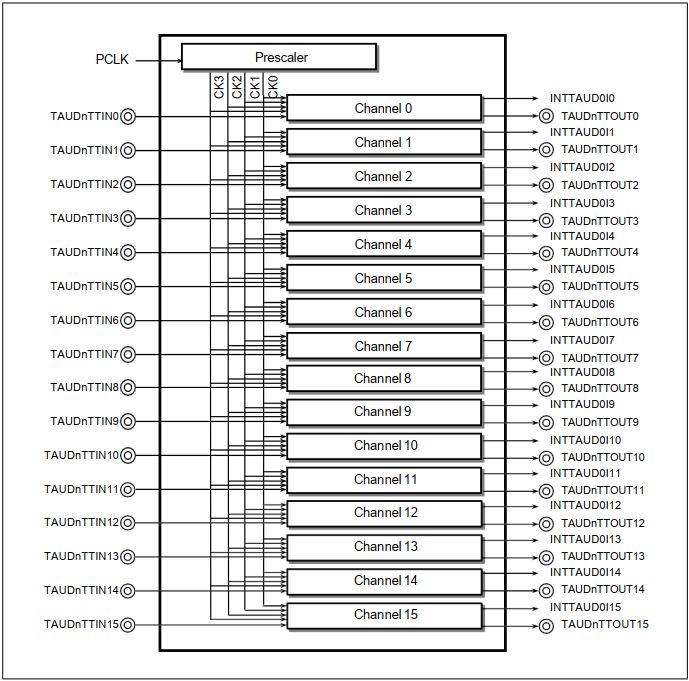
#### Functional List of Timer Operations

This timer provides the following functions by operating each channel independently or by combining multiple channels.

**Table 33.10 Functional List of TAUD Operations**

|  |  |
| --- | --- |
| **Operation Function** | **Example** |
| **Independent Channel Operation Functions** | **Section 33.12** |
| Interval Timer Function | Section 33.12.1 |
| TAUDnTTINm Input Interval Timer Function | Section 33.12.2 |
| External Event Count Function | Section 33.12.4 |
| Delay Count Function | Section 33.12.5 |
| TAUDnTTINm Input Pulse Interval Measurement Function | Section 33.12.7 |
| TAUDnTTINm Input Signal Width Measurement Function  TAUDnTTINm Input Position Detection Function | Section 33.12.8  Section 33.12.9 |
| TAUDnTTINm Input Period Count Detection Function | Section 33.12.10 |
| TAUDnTTINm Input Pulse Interval Judgment Function | Section 33.12.11 |
| TAUDnTTINm Input Signal Width Judgment Function | Section 33.12.12 |
| Overflow Interrupt Output Function (during TAUDnTTINm Width Measurement) | Section 33.12.13 |
| Overflow Interrupt Output Function (during TAUDnTTINm Input Period Count Detection) | Section 33.12.14 |
| **Independent Channel Real-Time Functions** | **Section 33.13** |
| Real-Time Output Function Type 1 | Section 33.13.1 |
| **Independent Channel Simultaneous Rewrite Functions** | **Section 33.14** |
| Simultaneous Rewrite Trigger Generation Function Type 1 | Section 33.14.1 |
| **Synchronous Channel Operation Functions** | **Section 33.15** |
| PWM Output Function | Section 33.15.1 |
| One-Shot Pulse Output Function | Section 33.15.2 |
| Trigger Start PWM Output Function | Section 33.15.3 |
| Delay Pulse Output Function | Section 33.15.4 |
| Offset Trigger Output Function | Section 33.15.5 |
| A/D Conversion Trigger Output Function Type 1 | Section 33.15.6 |
| Triangle PWM Output Function | Section 33.15.7 |
| Triangle PWM Output Function with Dead Time | Section 33.15.8 |
| A/D Conversion Trigger Output Function Type 2 | Section 33.15.9 |
| Interrupt Request Signals Culling Function | Section 33.15.10 |
| **Synchronous Non-Complementary and Complementary Modulation Output Functions** | **Section 33.16** |
| Non-Complementary Modulation Output Function Type 1 | Section 33.16.1 |
| Non-Complementary Modulation Output Function Type 2 | Section 33.16.2 |
| Complementary Modulation Output Function | Section 33.16.3 |

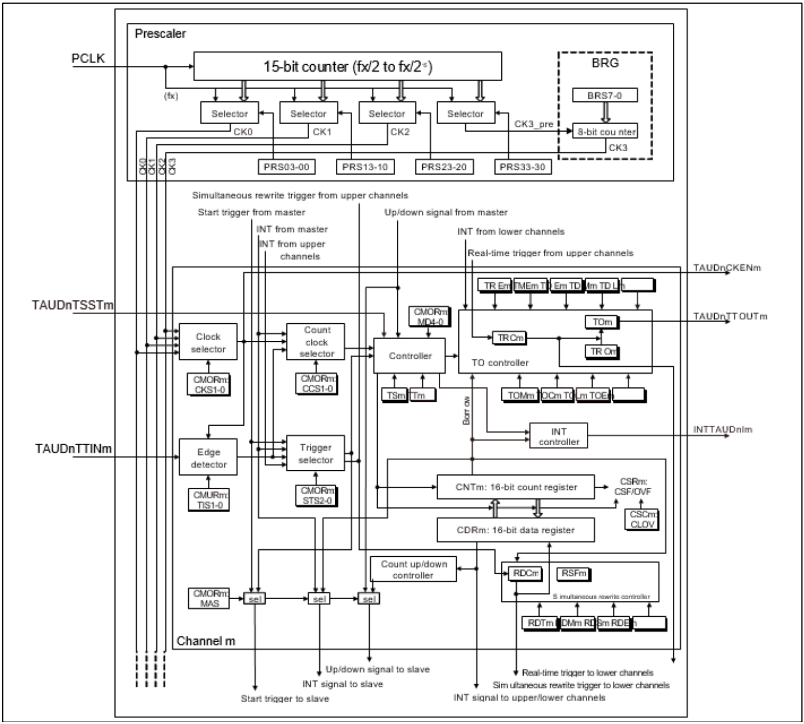
#### TAUD I/O and Interrupt Request Signals



**Figure 33.1** TAUD I/O and Interrupt Request Signals

#### Block Diagram

**Figure** 33.2, Block Diagram of the TAUD shows the main components of the TAUD.



**Figure** 33.2 Block Diagram of the TAUD

The module name “TAUDn” has been omitted from the register names for the sake of clarity in the above figure.

#### Description of Blocks

The following describes the functional blocks:

**Prescaler block**

The prescaler block provides up to four clock signals (CK0 to CK3) that can be used as count clocks for all channels.

Count clocks CK0 to CK2 are derived from PCLK by a configurable prescaler division factor of to . The fourth count clock CK3 can be adjusted more precisely by using BRG to set an additional division factor that is not a power of 2.

**Clock and count clock selection**

For every channel, the count clock selector selects which of the following is used as the clock source:

* One of the clocks CK0 to CK3 (selected by the clock selector)
* INTTAUDnIm from master channel.
* TAUDnTTINm input signal valid edge.

**Controller**

The controller controls the main operations of the counter:

* Operation mode (selected by bits TAUDnCMORm.TAUDnMD[4:0])
* Counter start enable (TAUDnTS.TAUDnTSm) and counter stop (TAUDnTT.TAUDnTTm)

When counter start is enabled, status flag TAUDnTE.TAUDnTEm is set.

* Count direction (up/down) (can be controlled by master channel)

**Trigger selector**

Depending on the selected operation mode, the counter starts automatically when it is enabled (TAUDnTE.TAUDnTEm = 1), or it waits for an external start trigger signal. Any of the following signals can be used as the start trigger.

* Synchronous channel start trigger input TAUDnTSSTm
* TAUDnTTINm input signal valid edge
* INTTAUDnIm from the master or any upper channel
* Up/down output trigger signal of the master channel
* Dead-time output signal of the TAUDnTTOUTm generation unit.

**Simultaneous rewrite controller**

Simultaneous rewrite control is a function that can be used in synchronous operating modes. The data registers (TAUDnCDRm) of all channels in a channel group can be rewritten at any time. The simultaneous rewrite controller ensures that new data register values of all channels become effective at the same time.

**TAUDnTO controller**

The output control of every channel enables the generation of various output signal forms such as PWM signals or triangular waves.

### Registers

#### List of Registers

TAUD registers are listed in the following table.

For details about <TAUDn\_base>, see Section 33.1.2, Register Base Addresses.

**Table 33.11 List of Registers**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Module Name | Register Name | Symbol | Address | AccessProtection | |
| PBG | Other |
| **TAUDn Prescaler Registers** | | | | | |
| TAUDn | TAUDn Prescaler Clock Select Register | TAUDnTPS | <TAUDn\_base> + 240H | \*1 | — |
| TAUDn Prescaler Baud Rate Setting Register | TAUDnBRS | <TAUDn\_base> + 244H | \*1 | — |
| **TAUDn Control Registers** | | | | | |
| TAUDn | TAUDn Channel Data Register m | TAUDnCDRm | <TAUDn\_base> + m × 4H | \*1 | — |
| TAUDn Channel Counter Register m | TAUDnCNTm | <TAUDn\_base> + 80H + m × 4H | \*1 | — |
| TAUDn Channel Mode OS Register m | TAUDnCMORm | <TAUDn\_base> + 200H + m × 4H | \*1 | — |
| TAUDn Channel Mode User Register m | TAUDnCMURm | <TAUDn\_base> + C0H + m × 4H | \*1 | — |
| TAUDn Channel Status Register m | TAUDnCSRm | <TAUDn\_base> + 140H + m × 4H | \*1 | — |
| TAUDn Channel Status Clear Trigger Register m | TAUDnCSCm | <TAUDn\_base> + 180H + m × 4H | \*1 | — |
| TAUDn Channel Start Trigger Register | TAUDnTS | <TAUDn\_base> + 1C4H | \*1 | — |
| TAUDn Channel Enable Status Register | TAUDnTE | <TAUDn\_base> + 1C0H | \*1 | — |
| TAUDn Channel Stop Trigger Register | TAUDnTT | <TAUDn\_base> + 1C8H | \*1 | — |
| **TAUDn Output Registers** | | | | | |
| TAUDn | TAUDn Channel Output Enable Register | TAUDnTOE | <TAUDn\_base> + 5CH | \*1 | — |
| TAUDn Channel Output Register | TAUDnTO | <TAUDn\_base> + 58H | \*1 | — |
| TAUDn Channel Output Mode Register | TAUDnTOM | <TAUDn\_base> + 248H | \*1 | — |
| TAUDn Channel Output Configuration Register | TAUDnTOC | <TAUDn\_base> + 24CH | \*1 | — |
| TAUDn Channel Output Active Level Register | TAUDnTOL | <TAUDn\_base> + 040H | \*1 | — |
| TAUDn Channel Dead Time Output EnableRegister | TAUDnTDE | <TAUDn\_base> + 250H | \*1 | — |
| TAUDn Channel Dead Time Output Mode Register | TAUDnTDM | <TAUDn\_base> + 254H | \*1 | — |
| TAUDn Channel Dead Time Output LevelRegister | TAUDnTDL | <TAUDn\_base> + 54H | \*1 | — |
| TAUDn Channel Real-time Output Register | TAUDnTRO | <TAUDn\_base> + 4CH | \*1 | — |
| TAUDn Channel Real-time Output EnableRegister | TAUDnTRE | <TAUDn\_base> + 258H | \*1 | — |
| TAUDn Channel Real-time Output ControlRegister | TAUDnTRC | <TAUDn\_base> + 25CH | \*1 | — |
| TAUDn Channel Modulation Output EnableRegister | TAUDnTME | <TAUDn\_base> + 50H | \*1 | — |
| **TAUDn Reload Data Registers** | | | | | |
|  | TAUDn Channel Reload Data Enable Register | TAUDnRDE | <TAUDn\_base> + 260H | \*1 | — |
|  | TAUDn Channel Reload Data Mode Register | TAUDnRDM | <TAUDn\_base> + 264H | \*1 | — |
|  | TAUDn Channel Reload Data Control ChannelSelect Register | TAUDnRDS | <TAUDn\_base> + 268H | \*1 | — |
|  | TAUDn Channel Reload Data Control Register | TAUDnRDC | <TAUDn\_base> + 26CH | \*1 | — |
|  | TAUDn Channel Reload Data Trigger Register | TAUDnRDT | <TAUDn\_base> + 44H | \*1 | — |
|  | TAUDn Channel Reload Status Register | TAUDnRSF | <TAUDn\_base> + 48H | \*1 | — |

Note 1. n = 0: PBG51#1

n = 1: PBG51#2

#### Details of TAUDn Prescaler Registers

##### TAUDnTPS — TAUDn Prescaler Clock Select Register

This register specifies clocks CK0, CK1, CK2, and CK3\_PRE for all channels of the PCLK prescaler. CK3 is generated by dividing CK3\_PRE by the factor specified in TAUDnBRS.

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

#### Details of TAUDn Control Registers

#### Details of TAUDn Simultaneous Rewrite Registers

#### Details of TAUDn Output Registers

#### Details of TAUDn Dead Time Output Registers

#### Details of TAUDn Real-time/Modulation Output Registers

### Operating Procedure

The following lists the general operation procedure for the TAUDn.

After reset release, the operation of each channel is stopped. Clock supply is started and writing to each register is enabled. All circuits and registers of all channels are initialized. The control register of TAUDnTTOUTm is also initialized and outputs a low level.

(1) Set the TAUDnTPS and TAUDnBRS registers to specify the clock frequency of CK0 to CK3.

(2) Configure the desired TAUDn function:

* Set the operation mode
* Set the channel output mode
* Set any other control bits

(3) Enable the counter by setting the TAUDnTS.TAUDnTSm bit to 1.

The counter starts to count immediately, or when an appropriate trigger is detected, depending on the bit settings.

(4) If desired, and if possible for the configured function, stop the counter or perform a forced restart operation during count operation. The counter can be stopped by setting the TAUDnTT.TAUDnTTm bit to 1. The counter can be forcibly restarted by setting the TAUDnTS.TAUDnTSm bit to 1.

(5) Stop the function by setting the TAUDnTT.TAUDnTTm bit to 1.

|  |
| --- |
| **NOTES**  1. A detailed description of the required control bits and the operation of the individual functions are given in Section 33.12, Independent Channel Operation Functions and Section 33.15, Synchronous Channel Operation Functions.  2. The function can be changed while the counter is stopped (TAUDnTE.TAUDnTEm = 0). |

### Concepts of Synchronous Channel Operation

The synchronous channel operation function is implemented using a combination of channel groups (consisted of master and slave channels). Several rules apply to the settings of channels. These rules are detailed in Section 33.5.1, Rules of Synchronous Channel Operation.

Two special features for synchronous channel operation are detailed in the following:

* **Section 33.5.2, Simultaneous Start and Stop of Synchronous Channel Counters**
* **Section 33.6, Simultaneous Rewrite**

#### Rules of Synchronous Channel Operation

**Number of master and slave channels**

* Only even channels (CH0, CH2, CH4, …) can be set as master channels. Any channel apart from CH0 can be set as a slave channel.
* Only channels lower than the master channel can be set as slave channels, and multiple slave channels can be set for one master channel.

Example: If CH2 is a master channel, CH3 and the lower channels (CH3, CH4, CH5, …) can be set as slave channels.

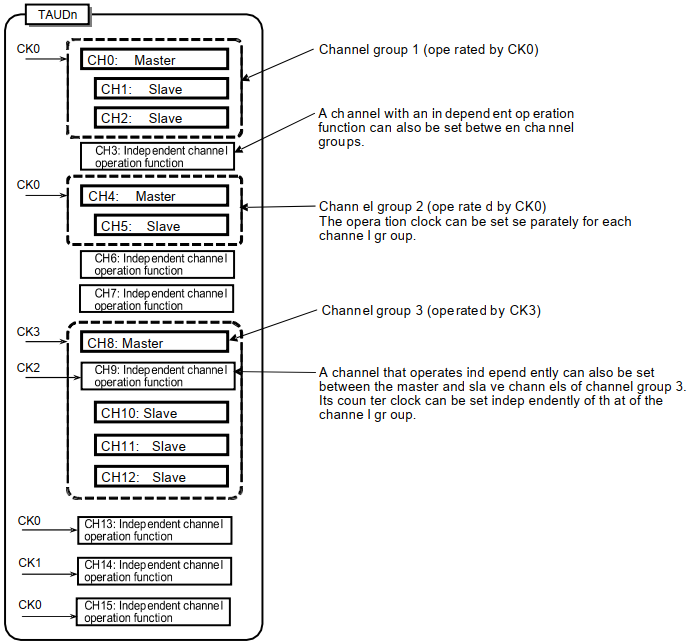
* If multiple master channels are used, slave channels cannot cross the master channels.

Example: If CH0 and CH4 are master channels, CH1 to CH3 can be set as slave channels for CH0, but CH5 to CH15 cannot.

**Operation clock**

* The same operation clock must be set for the master channel and the synchronized slave channel. This is achieved by setting the TAUDnCMORm.TAUDnCKS[1:0] bits of the slave and master channel.

The basic concepts of master/slave usage and operation clocks are illustrated in **Figure** 33.3, Grouping of Channels and Assignment of Count Clocks



**Figure** 33.3 Grouping of Channels and Assignment of Count Clocks

**Control trigger signal for master/slave channels**

* Master channels can output control trigger signals to slave channels.
* Slave channels can use control trigger signals from master channels but cannot output control trigger signals for their own to lower channels.
* Master channels cannot use control trigger signals from upper master channels.

#### Simultaneous Start and Stop of Synchronous Channel Counters

Channels that are operated synchronously can be started and stopped simultaneously within the same unit and between the units.

##### Simultaneous Start and Stop within the Same Unit

* To simultaneously start synchronized channels, the TAUDnTS.TAUDnTSm bits of the channels should be set at the same time.
* To simultaneously stop synchronized channels, the TAUDnTT.TAUDnTTm bits of the channels should be set at the same time.

Setting to the TAUDnTS.TAUDnTSm bits to 1 also sets the corresponding TAUDnTE.TAUDnTEm bits to 1, enabling counting. The count start timing depends on operating mode.

##### Simultaneous Start between the Units

Counters in different units can also be started simultaneously if the corresponding counters are enabled before receiving the simultaneous trigger signal.

For details about how to perform simultaneous start between the units, **41.2.3.1, Simultaneous Start Trigger Function.**

### Simultaneous Rewrite

#### Overview of Operations

Simultaneous rewrite describes the ability to change the compare/start value and the output logic of multiple channels at the same time.

The corresponding data and control registers (TAUDnCDRm and TAUDnTOLm) can nevertheless be written at any time. The new value does not affect the counter operation or the output signal until simultaneous rewrite is triggered.

Simultaneous rewrite can be triggered by:

* The counter on the master channel or upper channel (depending on the selected operation mode) reaching a certain value
* INTTAUDnIm being issued on the upper channel specified by TAUDnRDC.TAUDnRDCm

There are four methods for simultaneous rewrite. These are listed in **Table 33.43, Simultaneous Rewrite Methods and when They are Triggered**, along with how to specify them and when they cause simultaneous rewrite to be triggered.

**Table 33.43 Simultaneous Rewrite Methods and when They are Triggered**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method** | **Simultaneous Rewrite Triggered when** | **TAUDnRDE. TAUDnRDEm** | **TAUDnRDS. TAUDnRDSm** | **TAUDnRDM. TAUDnRDMm** |
| — | No simultaneous rewrite | 0 | 0 | 0 |
| A | The master channel (re)starts counting | 1 | 0 | 0 |
| B | Counting is started in the master channel. The master channel starts counting down at the peak of triangular cycle of the corresponding slave channel. | 1 | 0 | 1 |
| C1 | INTTAUDnIm is generated on an upper channel specified by TAUDnRDC.TAUDnRDCm | 1 | 1 | 0/1 |
| C2 | INTTAUDnIm is generated on an upper channel specified by TAUDnRDC.TAUDnRDCm that in turn is triggered by an external signal | 1 | 1 | 0/1 |

**Table** 33.44 lists which of these four methods is available for each channel operation function. For more information about the individual channel operation functions, see the corresponding sections in Section 33.14, Independent Channel Simultaneous Rewrite Functions, Section 33.15, Synchronous Channel Operation Functions, and Section 33.16, Synchronous NonComplementary and Complementary Modulation Output Functions.

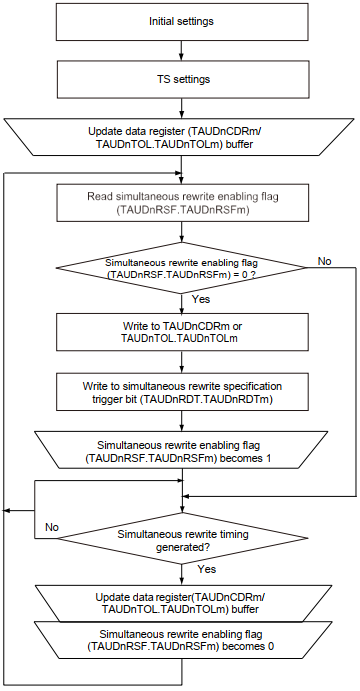
**Table 33.44 Channel Functions and the Methods They Use for Simultaneous Rewrite**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Function** | **A** | **B** | **C1** | **C2** | **TAUDnTOL. TAUDnTOLm** |
| Simultaneous Rewrite Trigger Output Function Type 1 |  |  | √ |  |  |
| PWM Output Function | √ |  | √ |  | √ |
| One-Shot Pulse Output Function | √ |  |  |  |  |
| Trigger Start PWM Output Function | √ |  |  | √ |  |
| Delay Pulse Output Function | √ |  |  |  |  |
| Triangle PWM Output Function |  | √ | √ |  | √ |
| Triangle PWM Output Function with Dead Time |  | √ | √ |  |  |
| Interrupt Request Signals Culling Function | √ | √ | √ |  |  |
| AD Conversion Trigger Output Function Type 1 | √ |  | √ |  |  |
| AD Conversion Trigger Output Function Type 2 |  | √ | √ |  |  |
| Non-Complementary Modulation Output Function Type 1 | √ |  | √ |  |  |
| Non-Complementary Modulation Output Function Type 2 |  | √ | √ |  |  |
| Complementary Modulation Output Function |  | √ | √ |  |  |

Note: √: Available, (Blank): Unavailable

#### How to Control Simultaneous Rewrite

**Figure 33.4, General Procedure for Simultaneous Rewrite** shows the general procedure for simultaneous rewrite. The three main blocks (initial settings, start and counter count operation, and simultaneous rewrite) are explained afterwards.



**Figure** 33.4 General Procedure for Simultaneous Rewrite

##### Initial Settings

* To enable simultaneous rewrite in channel m, set TAUDnRDE.TAUDnRDEm = 1
* To select the type of simultaneous rewrite, set TAUDnRDM.TAUDnRDMm and
* TAUDnRDS.TAUDnRDSm according to the values listed in Table 33.43, Simultaneous Rewrite Methods and when They are Triggered.
* Specify a simultaneous rewrite trigger channel by using TAUDnRDC.TAUDnRDCm.

(Prerequisite: TAUDnRDS.TAUDnRDSm has been set to the upper channel.)

##### Start Counter and Count Operation

* To start all the TAUDnCNTm counters of the channel group, set the corresponding TAUDnTS.TAUDnTSm bits to 1. The values of TAUDnTOL.TAUDnTOLm and the data registers (TAUDnCDRm) are loaded into the corresponding TAUDnTOL.TAUDnTOLm buffer (TAUDnTOL.TAUDnTOLm buf) and data buffer registers (TAUDnCDRm buf) and the counters start.
* Setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1 sets the reload flag (TAUDnRSF.TAUDnRSFm) to 1, enabling simultaneous rewrite. TAUDnRSF.TAUDnRSFm remains set to 1 until simultaneous rewrite is completed.
* When the specified trigger for simultaneous rewrite is detected, the TAUDnRSF.TAUDnRSFm bit is checked to see if simultaneous rewrite is enabled (TAUDnRSF.TAUDnRSFm = 1). If it is, simultaneous rewrite is carried out. Otherwise the simultaneous rewrite is not carried out and waits for the next trigger detection.

##### Simultaneous Rewrite

* When simultaneous rewrite is enabled (TAUDnRSF.TAUDnRSFm = 1) and the simultaneous rewrite trigger is detected, the current values of the data registers are copied to their buffers. These values are then loaded into the corresponding counters and are applied the next time the counter starts or restarts.
* When simultaneous rewrite is complete, the TAUDnRSF.TAUDnRSFm bit is set to 0, and the system awaits the next simultaneous rewrite trigger.

#### Other General Rules of Simultaneous Rewrite

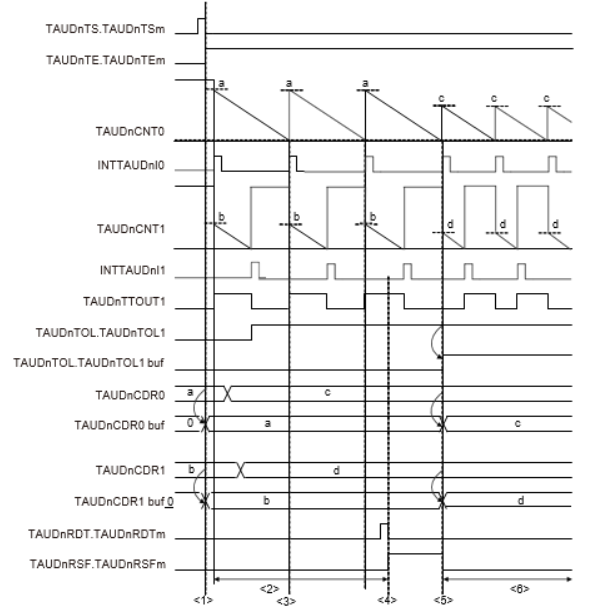
The following rules also apply:

* TAUDnRDE.TAUDnRDEm, TAUDnRDS.TAUDnRDSm, TAUDnRDM.TAUDnRDMm, and TAUDnRDC.TAUDnRDCm cannot be changed while the counter is in operation (TAUDnTE.TAUDnTEm = 1).
* TAUDnTOL.TAUDnTOLm can only be rewritten during operation with PWM output function or triangle PWM output function. For all other output functions, TAUDnTOL.TAUDnTOLmshould be written before the counter starts. If it is rewritten while any other function is used, TAUDnTTOUTm outputs an invalid wave.
* When an upper channel is used as a channel issuing the simultaneous rewrite trigger (TAUDnRDS.TAUDnRDSm = 1), the TAUDnRDC.TAUDnRDCm bit controls all the lower channels. This means that if the TAUDnRDC.TAUDnRDCm bits of CH2 and CH7 are set to 1 and the TAUDnRDC.TAUDnRDCm bits of other channels are set to 0, CH2 and CH7 serve as simultaneous rewrite trigger generation channels. CH2 controls the lower channels CH3 to CH6, and CH7 controls the lower channels CH8 to CH15.
* If simultaneous rewrite is enabled and an upper channel is selected for the simultaneous rewrite trigger (TAUDnRDE.TAUDnRDEm and TAUDnRDS.TAUDnRDSm = 1) but no upper channel is set (TAUDnRDC.TAUDnRDC[15:0] = 0), simultaneous rewrite cannot take place.

#### Types of Simultaneous Rewrite

In the following section, the four simultaneous rewrite methods are explained using timing diagrams.

##### Simultaneous Rewrite when the Master Channel (Re)starts Counting (Method A)



**Figure 33.5 Simultaneous Rewrite when the Master Channel (Re)starts Counting**

**Setting:**

CH0 is the master channel, which starts counting down, and CH1 represents an arbitrary slave channel. The simultaneous rewrite method A is applied.

**Description:**

(1) When TAUDnTS.TAUDnTSm is set to 1, TAUDnCDRm value is copied to the TAUDnCDRm buffer and TAUDnTOL.TAUDnTOLm value is copied to the TAUDnTOL.TAUDnTOLm buffer.

(2) The TAUDnCDRm and TAUDnTOL.TAUDnTOLm registers can be written at any time.

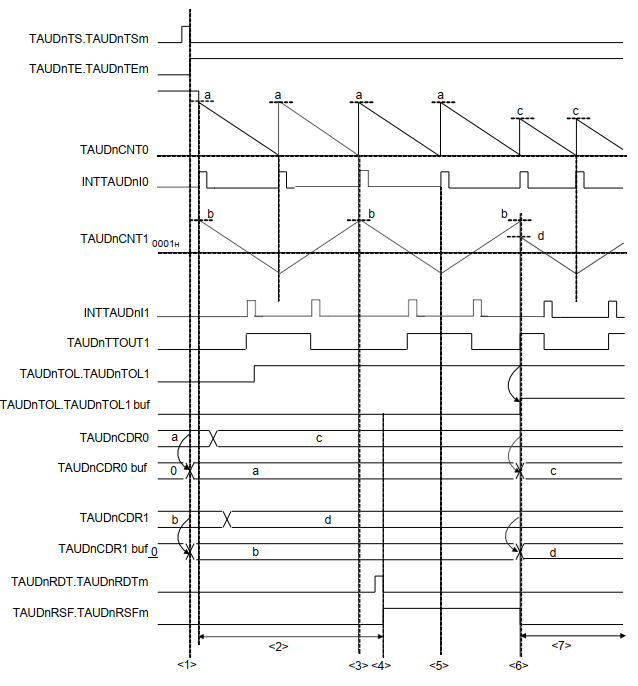
(3) CH0 restarts counting, but simultaneous rewrite does not occur because it is disabled (TAUDnRSF.TAUDnRSFm = 0)

(4) The reload data trigger bit (TAUDnRDT.TAUDnRDTm) is set to 1 which sets the status flag (TAUDnRSF.TAUDnRSFm = 1), enabling simultaneous rewrite.

(5) Because simultaneous rewrite is enabled, it is triggered when CH0 restarts counting. The TAUDnCDRm value is loaded into the TAUDnCDRm buffer and the TAUDnTOL.TAUDnTOLm value is loaded into the TAUDnTOL.TAUDnTOLm buffer.

(6) The counters count down and await the next simultaneous rewrite trigger. The valuesof TAUDnCDRm and TAUDnTOL.TAUDnTOLm can be changed again.

##### Simultaneous Rewrite at the Peak of a Triangular Wave of Slave Channel (Method B)



**Figure 33.6 Simultaneous Rewrite at the Peak of a Triangular Wave of Slave Channel**

**Setting:**

CH0 is the master channel which performs counting down, and CH1 represents an arbitrary slave channel. The simultaneous rewrite method B is applied.

**Description:**

(7) When TAUDnTS.TAUDnTSm is set to 1, TAUDnCDRm value is copied to the TAUDnCDRm buffer.

(8) The TAUDnCDRm and TAUDnTOL registers can be written at any time.

(9) Simultaneous rewrite does not occur because it is disabled (TAUDnRSF.TAUDnRSFm = 0).

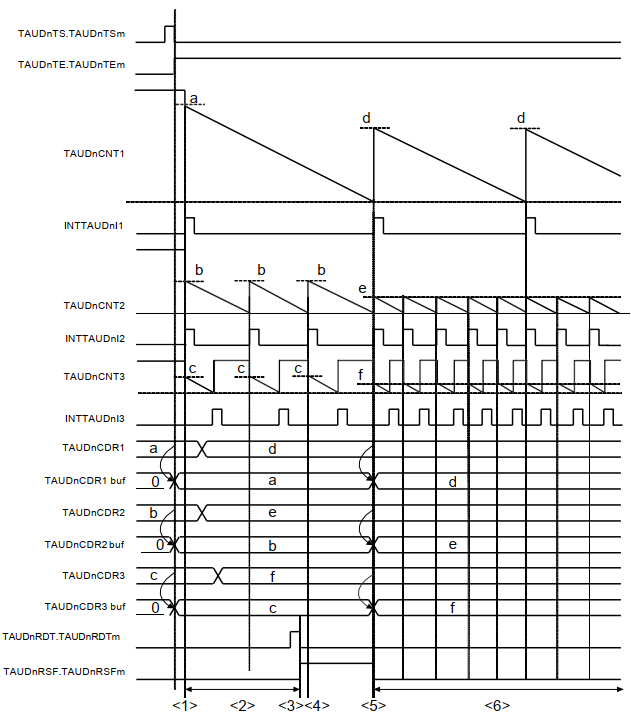
(10) The reload data trigger bit (TAUDnRDT.TAUDnRDTm) is set to 1 which sets the status flag (TAUDnRSF.TAUDnRSFm = 1), enabling simultaneous rewrite.

(11) Simultaneous rewrite does not take place at the bottom of the triangular cycle.

(12) Simultaneous rewrite takes place at the top of the triangular cycle. The TAUDnCDRm value is loaded into the TAUDnCDRm buffer, the TAUDnTOL.TAUDnTOLm value is loaded into the TAUDnTOL.TAUDnTOLm buffer.

(13) The counters count down and await the next simultaneous rewrite trigger. The valuesof TAUDnCDRm and TAUDnTOL.TAUDnTOLm can be changed again.

##### Simultaneous Rewrite when INTTAUDnIm is Generated on an UpperChannel Specified by TAUDnRDC.TAUDnRDCm (Method C1)



**Figure 33.7 Simultaneous Rewrite When INTTAUDnIm Is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm**

**Setting:**

CH1 is an upper channel which performs counting down, CH2 is a master channel, and CH3 is the slave channel. The simultaneous rewrite method C1 is applied. The TAUDnRDC register specifies a channel which generates simultaneous rewrite triggers.

Description:

(14) When TAUDnTS.TAUDnTSm is set to 1, TAUDnCDRm value is copied to the TAUDnCDRm buffer.

(15) The TAUDnCDRm register is always ready to write.

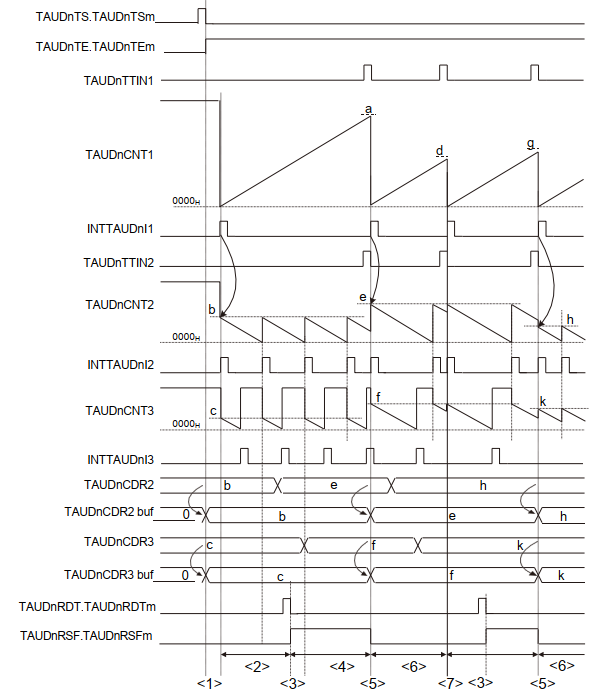
(16) By setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1, the status flag is set (TAUDnRSF.TAUDnRSFm = 1) to enable simultaneous rewrite.

(17) Simultaneous rewrite is triggered only by a CH1 interrupt. Therefore, simultaneous rewrite is not conducted even if enabled.

(18) Simultaneous rewrite is triggered by INT1 which is generated when counter 1 reaches 0000H. The TAUDnCDRm values are loaded into the corresponding TAUDnCDRmbuffers.

(19) The counter counts down and awaits the next simultaneous rewrite trigger. The values of the TAUDnCDRm registers can be rechanged.

##### Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal (Method C2)



**Figure 33.8 Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal**

**Setting:**

CH1 is an upper channel which performs counting up, CH2 is a master channel, and CH3 is the slave channel. The synchronous channel operation method C2 is applied. The TAUDnRDC register specifies which upper channel is monitored for an INTTAUDnIm trigger.

**Description:**

(20) When TAUDnTS.TAUDnTSm is set to 1, TAUDnCDRm value is copied to the TAUDnCDRm buffer. However, as TAUDnCDR1 operates in capture mode, TAUDnCDR1 value is not copied to the TAUDnCDR1 buffer.

(21) The TAUDnCDRm register is always ready to write.

(22) By setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1, the status flag is set (TAUDnRSF.TAUDnRSFm = 1) to enable simultaneous rewrite.

(23) Simultaneous rewrite is triggered only by a CH1 interrupt. Therefore, simultaneous rewrite is not conducted even if enabled.

(24) Simultaneous rewrite is triggered by INT1 which is caused by external signal TIN1. The TAUDnCDRm values are written to the corresponding TAUDnCDRm buffers.

(25) The counters count down and await the next simultaneous rewrite trigger. The values of the TAUDnCDRm registers can be changed again.

(26) An external signal occurs at TIN2 but simultaneous rewrite does not take place because it is disabled (TAUDnRSF.TAUDnRSFm = 0).

### Channel Output Modes

The output of the TAUDnTTOUTm pin can be controlled in two ways, the latter of which can be further split into individual modes.

* By software (TAUDnTOE.TAUDnTOEm = 0)

When controlled by software, the value written in the output register bit (TAUDnTO.TAUDnTOm) is sent to the output pin (TAUDnTTOUTm).

* By TAUD signals (TAUDnTOE.TAUDnTOEm = 1)

When controlled by TAUD signals, the output level of TAUDnTTOUTm is set or reset or toggled by internal signals. The value of TAUDnTO.TAUDnTOm is updated accordingly to reflect the value of TAUDnTTOUTm.

* Independently (TAUDnTOM.TAUDnTOMm = 0)

In case of independent operation, the output of the TAUDnTTOUTm pin is only affected by settings of channel m. Therefore, independent channel operation should be selected (TAUDnTOM.TAUDnTOMm = 0).

* Synchronously (TAUDnTOM.TAUDnTOMm = 1)

In case of synchronous operation, the output of the TAUDnTTOUTm pin is affected by settings of channel m and those of other channels. Therefore, synchronous channel operation should be selected for all synchronized channels (TAUDnTOM.TAUDnTOMm = 1).

The TAUDnTO.TAUDnTOm bit can always be read to determine the current value of TAUDnTTOUTm, regardless of whether the pin is controlled by software, operated independently, or operated synchronously.

**Control bits**

The settings of the control bits required to select a specific channel output mode are listed in **Table 33.45, Channel Output Modes.**

The channel output modes are described in details below.

* + Section 33.7.2, Channel Output Modes Controlled Independently by TAUDnSignals
  + Section 33.7.3, Channel Output Modes Controlled Synchronously byTAUDn Signals

**Batch operation of TAUDnTOm bit**

Whether a set value is reflected to the TAUDnTOm bit or not is controlled by the TAUDnTOE.TAUDnTOEm bit.

The TAUDnTOm setting is written only to the bit (channel) set with TAUDnTOE.TAUDnTOEm bit = 0 when a write to the TAUDnTO register is attempted. No TAUDnTOm setting is reflected to the bit (channel) set with TAUDnTOE.TAUDnTOEm bit = 1.

|  |
| --- |
| **NOTE**  TAUDnTO.TAUDnTOm bit is placed so that its bit number corresponds to a channel number. |

**Output logic**

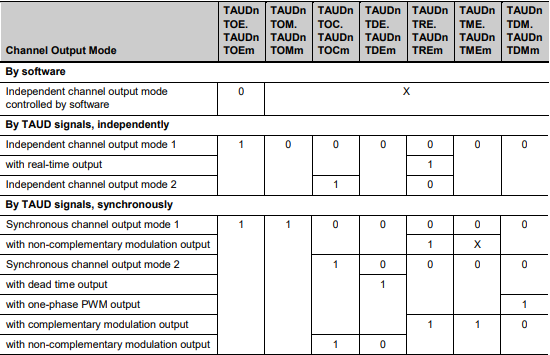
Positive logic or negative logic of the output is specified by control bit TAUDnTOL.TAUDnTOLm.

The value of TAUDnTOL.TAUDnTOLm bit should be set before the counter is started. It can only be changed during operation with PWM output function or triangle PWM output function. Otherwise, changes to TAUDnTOL.TAUDnTOLm result in an undefined TAUDnTTOUTm signal output.

**See Section 33.6, Simultaneous Rewrite.**

The various channel output modes and the channel output control bits are listed in Table 33.45.

**Table 33.45 Channel Output Modes**



* All combinations not listed in this table are forbidden.
* Bits marked with an x can be set to any value.

|  |
| --- |
| **NOTES**  1. The following bits cannot be changed during count operation (TAUDnTE.TAUDnTEm = 1):   * TAUDnTOM.TAUDnTOMm * TAUDnTOC.TAUDnTOCm * TAUDnTDE.TAUDnTDEm * TAUDnTRE.TAUDnTREm * TAUDnTDM.TAUDnTDMm   2. The following bits cannot be changed during count operation (TAUDnTE.TAUDnTEm = 1) except in channel output modes with modulation output:   * TAUDnTME.TAUDnTMEm * TAUDnTDL.TAUDnTDLm |

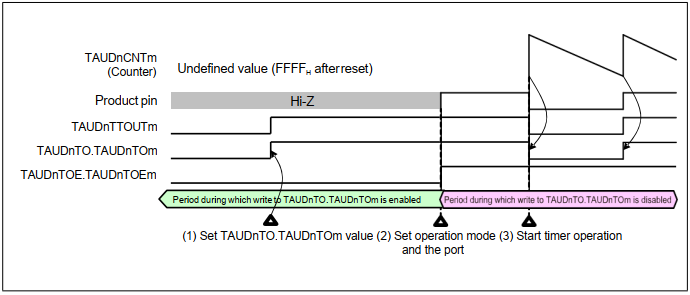
#### General Procedures for Specifying a Channel Output Mode

This section describes the general procedures for specifying a TAUDnTTOUTm channel output mode. The prerequisite is that timer output operation is disabled (TAUDnTOE.TAUDnTOEm = 0).

(1) Set TAUDnTO.TAUDnTOm to specify the initial level of the TAUDnTTOUTm output.

(2) Set channel output mode according to Table 33.45, Channel Output Modes, and the output logic using the TAUDnTOL.TAUDnTOLm bit.

(3) Start the counter (TAUDnTS.TAUDnTSm = 1).

****

**Figure 33.9 General Procedure for Specifying a TAUDnTTOUTm Channel Output Mode**

#### Channel Output Modes Controlled Independently by TAUDn Signals

This section lists the channel output modes that are controlled independently by TAUDn signals. The control bits used to specify a mode are listed in Table 33.45, Channel Output Modes.

##### 33.7.2.1 Independent Channel Output Mode 1

**Set/reset conditions**

In this output mode, TAUDnTTOUTm toggles when INTTAUDnIm is detected. The value of TAUDnTOL.TAUDnTOLm is ignored.

**Prerequisites**

There are no prerequisites other than those shown in Table 33.45, Channel Output Modes.

##### Independent Channel Output Mode 1 with Real-Time Output

In this output mode, the value of TAUDnTRO.TAUDnTROm bit of the trigger channel is output to TAUDnTTOUTm. The trigger channel is specified by setting the corresponding TAUDnTRC.TAUDnTRCm bit to 1. It controls all lower channels for which TAUDnTRC.TAUDnTRCm = 0.

**Set/reset conditions**

The value of TAUDnTRO.TAUDnTROm bit is sent to TAUDnTTOUTm only when an INTTAUDnIm interrupt occurs on the trigger channel. The interrupt is generated either:

* at certain specified intervals or
* on detection of a valid TAUDnTTINm input edge/counter start

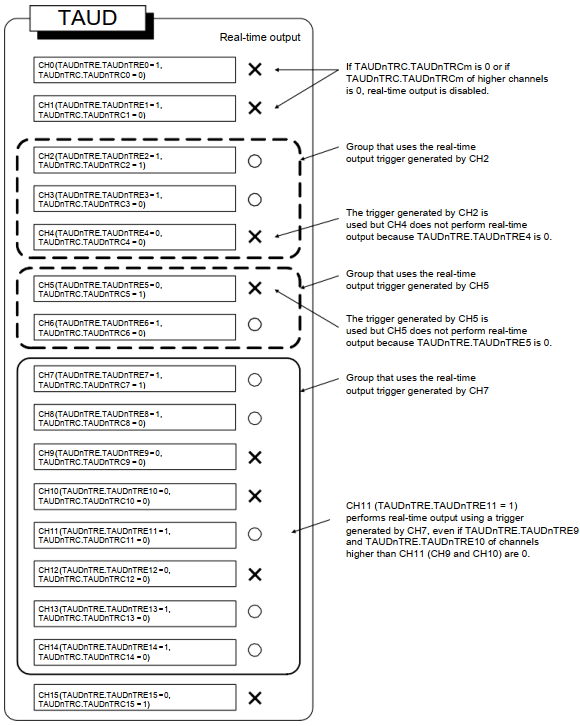
The type of trigger is set using the TAUDnCMORm.TAUDnMD[4:1] bits.

**Prerequisites**

Both the master and slave channels can be set as a trigger generation channel. A channel for which TAUDnTRC.TAUDnTRCm is set to 1 serves as a trigger generation channel regardless of the value of TAUDnTRE.TAUDnTREm.

If there is no channel for which TAUDnTRC.TAUDnTRCm is set to 1 or if TAUDnTRC.TAUDnTRC0 = 0, real-time output cannot take place.

This can be seen in **Figure 33.10.**

****

**Figure 33.10 Real-Time Output**

##### Independent Channel Output Mode 2

**Set/reset conditions**

In this output mode, TAUDnTTOUTm is set when INTTAUDnIm occurs at the time of count start, and reset when INTTAUDnIm occurs due to a match betweenTAUDnCNTm and TAUDnCDRm.

**Prerequisites**

There are no prerequisites other than those shown in **Table 33.45, Channel Output Modes.**

#### Channel Output Modes Controlled Synchronously by TAUDn Signals

This section lists the channel output modes that are controlled synchronously by TAUDn signals. The control bits used to specify a mode are listed in **Table 33.45, Channel Output Modes.**

##### 33.7.3.1 Synchronous Channel Output Mode 1

**Set/reset conditions**

In this output mode, INTTAUDnIm of master channel serves as a set signal and INTTAUDnIm of the slave channel as a reset signal. If INTTAUDnIm of master channel and INTTAUDnIm of the slave channel are generated at the same time, INTTAUDnIm of the slave channel (reset signal) has priority over INTTAUDnIm (set signal) of master channel, i.e., the master channel is ignored.

**Prerequisites**

There are no prerequisites other than those shown in **Table 33.45, Channel Output Modes.**

##### Synchronous Channel Output Mode 1 with Non-Complementary Modulation Output

**Set/reset conditions**

In this output mode, TAUDnTTOUTm outputs the result of an AND operation between the PWM output and the real-time output bit (TAUDnTRO.TAUDnTROm) of a channel.

The phase period to which the dead time is added is specified using the TAUDnTDL.TAUDnTDLm bit; for positive phase set TAUDnTDL.TAUDnTDLm = 0 and for negative phase set TAUDnTDL.TAUDnTDLm = 1.

**Prerequisites**

A set of at least three channels is required to generate the PWM output. The master channel and slave channel 1 generate a period, and slave channel 2 generates the duty cycle. In typical applications, five more slave channels are also used that operate in the same manner as slave channel 2.

Only the PWM output and the real-time output bit of the same channel can be combined.

TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can only be changed during count operation.

* If TAUDnTME.TAUDnTMEm is changed, its new value is applied upon detection of INTTAUDnIm on the specified channel.
* If TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm are changed, their new values are applied upon detection of INTTAUDnIm on the master channel.

##### Synchronous Channel Output Mode 2

In this output mode, the operating mode should be set to count-up/-down mode. The result is a triangle PWM output at TAUDnTTOUTm. For details, see **Section 33.15.7, Triangle PWM Output Function.**

**Set/reset conditions**

TAUDnCNTm of the slave channel counts down and up alternatively. When it passes 0001H it generates an interrupt, causing TAUDnTTOUTm to toggle.

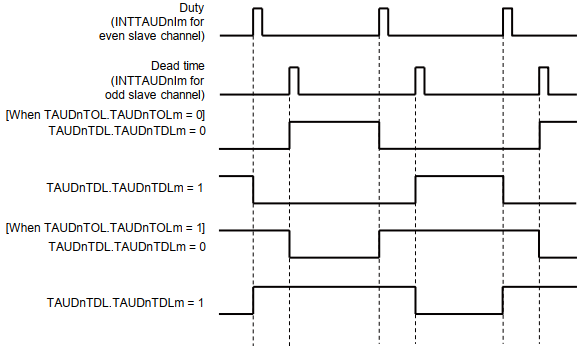
**Prerequisites**

A set of two channels is required to generate the triangle PWM output. TAUDnTTOUTm should be set to 0 before the function starts.

##### Synchronous Channel Output Mode 2 with Dead Time Output

In this output mode, a dead time delay is added to TAUDnTTOUTm. The set/reset conditions are shown in **Figure 33.11.**

**Set/reset conditions**



**Figure 33.11 Set/Reset Conditions for Synchronous Channel Output Mode 2 with Dead Time Output**

With regard to the edge to which dead time is added, set TAUDnTDL.TAUDnTDLm = 0 for rising edges and TAUDnTDL.TAUDnTDLm = 1 for falling edges

**Prerequisites**

Dead time control requires a set of three channels, each operating in the following modes:

* One master channel

The master channel should be set to interval timer mode.

* One even slave channel

The even slave channel should be set count-up/-down mode.

* One odd slave channel (even channel + 1)

The odd slave channel should be set to one-count mode.

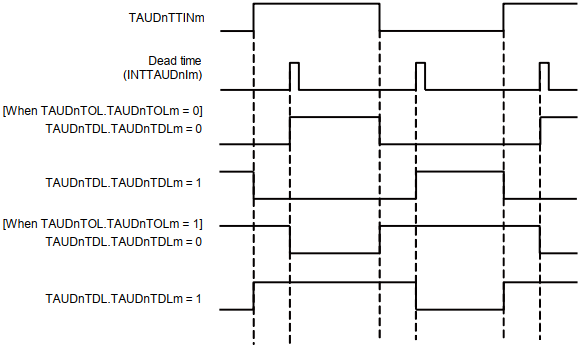
The values of the following bits should be the same for the odd channel and the even channel:

* TAUDnTOE.TAUDnTOEm
* TAUDnTME.TAUDnTMEm
* TAUDnTRE.TAUDnTREm
* TAUDnTOM.TAUDnTOMm
* TAUDnTOC.TAUDnTOCm
* TAUDnTDE.TAUDnTDEm
* TAUDnTDM.TAUDnTDMm

##### Synchronous Channel Output Mode 2 with One-Phase PWM Output

In this output mode, a dead time delay is added to TAUDnTTOUTm. The set/reset conditions are shown in Figure 33.12.

**Set/reset conditions**

****

**Figure 33.12 Set/Reset Conditions for Synchronous Channel Output Mode 2 with One-Phase PWM Output**

With regard to the edge to which dead time is added, set TAUDnTDL.TAUDnTDLm = 0 for rising edges and TAUDnTDL.TAUDnTDLm = 1 for falling edges.

**Prerequisites**

One-phase PWM output control requires a set of two channels:

* One even slave channel
* One odd slave channel (even channel + 1)

The odd slave channel should be set to one-count mode.

The values of the following bits should be the same for the odd channel and the even channel:

* TAUDnTOE.TAUDnTOEm
* TAUDnTME.TAUDnTMEm
* TAUDnTRE.TAUDnTREm
* TAUDnTOM.TAUDnTOMm
* TAUDnTOC.TAUDnTOCm
* TAUDnTDE.TAUDnTDEm
* TAUDnTDM.TAUDnTDMm

##### Synchronous Channel Output Mode 2 with Complementary Modulation Output

**Set/reset conditions**

In this output mode, TAUDnTTOUTm outputs a PWM signal, a high signal, or a low signal depending on the value of real-time output bit (TAUDnTRO.TAUDnTROm), the modulation output bit (TAUDnTME.TAUDnTMEm), and the output level bit (TAUDnTOL.TAUDnTOLm) of a pair of slave channels.

For details, see Section 33.16.3, Complementary Modulation Output Function.

**Prerequisites**

A set of at least four channels is required for this mode. The master channel and slave channel 1 generate a period, slave channel 2 generates a duty cycle, and slave channel 3 generates dead time. Slave channels 2 and 3 are a pair. In typical applications, four more channels are also used, which operates in the same manner as slave channels 2 and 3 respectively.

TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can only be changed during count operation.

* If TAUDnTME.TAUDnTMEm is changed during operation, its new value is applied upon detection of INTTAUDnIm at the specified channel.
* If TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm are changed, their new values are applied upon detection of INTTAUDnIm on an even slave channel.

##### Synchronous Channel Output Mode 2 with Non-Complementary Modulation Output

The difference from synchronous channel output mode 1 with non-complementary modulation output is the PWM wave shape.

Mode 1 has a square wave while mode 2 has a triangular wave.

### Start Timing in Each Operating Modes

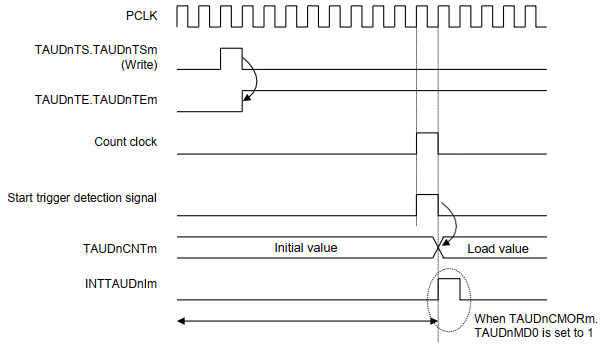
This section describes the timing at which the counter starts after TAUDnTS.TAUDnTSm is set to 1 in each operating mode.

In all modes, the value of data register and whether or not an interrupt occurs depends on mode and register settings.

|  |
| --- |
| **CAUTION**  The count start timing described in this section is for your reference. Actually, the count start timing depends on the count clock timing. |

#### Interval Timer Mode, Judge Mode, Capture Mode, Count-up/-downMode, and Count Capture Mode

The counter starts operating with the next count clock cycle after TAUDnTS.TAUDnTSm is set to 1. The value of data register is also loaded when the counter starts.

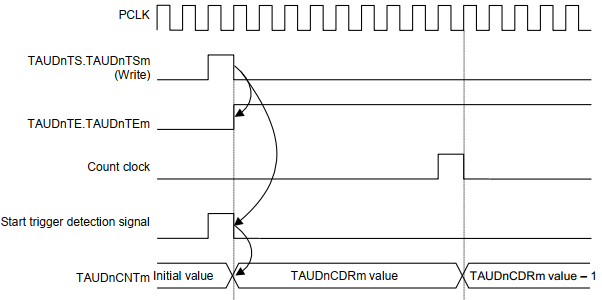
****

**Figure 33.13 Start Timing in Interval Timer Mode, Judge Mode, Capture Mode, Count- up/-down Mode, and Count Capture Mode**

|  |
| --- |
| **NOTE**  Make sure to set TAUDnCMORm.TAUDnMD0 to 0 when using the count-up/-down mode. |

#### Event Count Mode

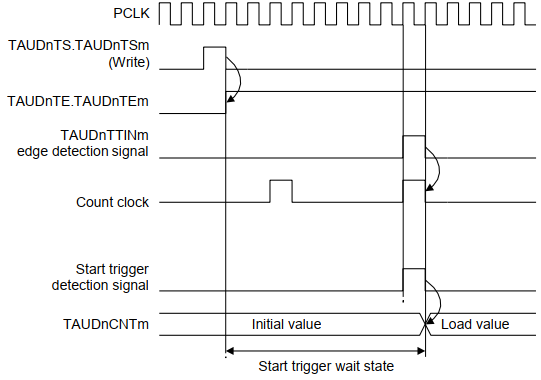
The value of data register is loaded as soon as TAUDnTS.TAUDnTSm is set to 1. The counter also starts immediately. The value of data register decrements when the subsequent count clock cycle starts.

****

**Figure 33.14 Start Timing in Event Count Mode**

#### Other Operating Modes

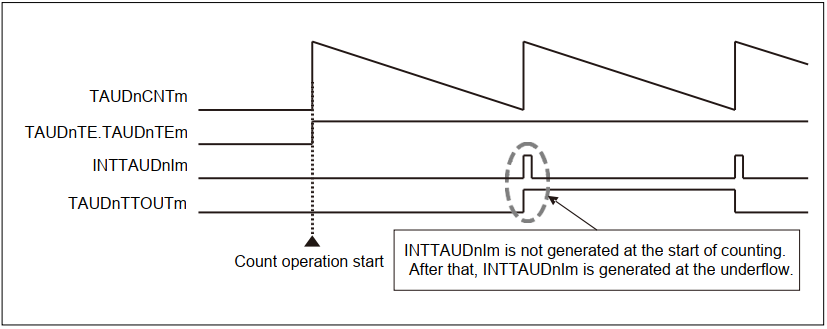
In other operating modes, the counter operation start timing is triggered only upon detection of a valid edge of TAUDnTTINm. Once the counter starts, the value of data register is also loaded. The count clock cycles, which are irrelevant to start of counter operation, determine the frequency with which all operations take place.

****

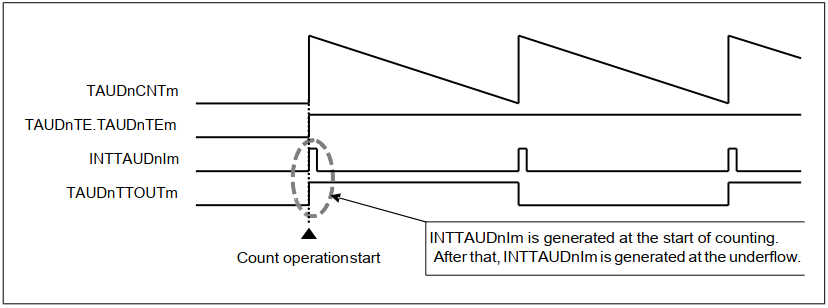
**Figure 33.15 Count Start Timing in Other Operating Modes**

### TAUDnTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts

When the counter starts, it is possible to specify whether an INTTAUDnIm is generated using the TAUDnCMORm.TAUDnMD0 bit. The generation of INTTAUDnIm when the TAUDnCMORm.TAUDnMD0 bit starts counting and the effect to TAUDnTTOUTm depend on the selected function. For details, refer to the description of TAUDnCMORm.TAUDnMD0 of each function.



**Figure 33.16 INTTAUDnIm Generation Timing (when TAUDnCMORm.TAUDnMD0 = 0)**

****

**Figure 33.17 INTTAUDnIm Generation Timing (when TAUDnCMORm.TAUDnMD0 = 1)**

### Interrupt Generation upon Overflow

In certain independent functions, an interrupt is not generated when the counter value reaches FFFFH and an overflow occurs during count-up. This section describes how to generate an interrupt by combining channel operation in a mode that counts up and in a mode that counts down.

The appropriate operation mode for the second channel depends on the operation mode of the first channel. Nevertheless, the principle is the same for all combinations:

* Find an operation mode for the second channel that counts down in such a manner, that it reaches 0000H at the same time as the first channel overflows (TAUDnCNTm = FFFFH).
* Set TAUDnCDRm of the second channel to FFFFH.
* The two channels must count at the same speed (i.e. they must have the same count clock).
* Both channels are triggered by the same TAUDnTTINm input.
* The trigger detection settings (TAUDnCMORm.TAUDnSTS[2:0] and TAUDnCMURm.TAUDnTIS[1:0]) must be identical for both channels.

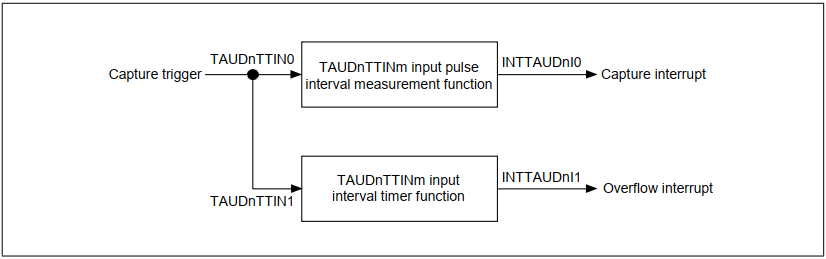
**Result**:

The down-counter of the second channel reaches 0000H at exactly the same time as the up-counter of the first channel overflows (TAUDnCNTm = FFFFH). Thus the second channel generates the desired interrupt.

The following sections list the operating modes that count down that are required to match specific operating modes that count up, as well as example timing diagrams.

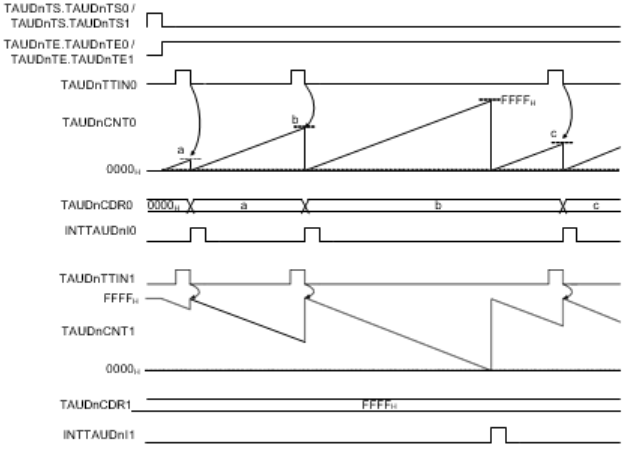
#### Combination of the TAUDnTTINm Input Pulse Interval Measurement Function and the TAUDnTTINm Input Interval Timer Function

When the capture trigger is input simultaneously to TAUDnTTINm of both channels, INTTAUDnIm of the TAUDnTTINm input interval timer function can detect the overflow when TAUDnCNTm of the TAUDnTTINm input pulse interval measurement function exceeds FFFFH.



**Figure 33.18 Combination of the TAUDnTTINm Input Pulse Interval Measurement Function and the TAUDnTTINm Input Interval Timer Function**

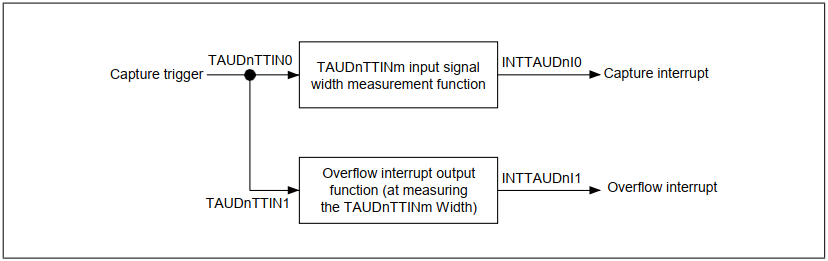
**Timing diagram**



**Figure 33.19 Interrupt Generation via Combination of the TAUDnTTINm Input Pulse Interval Measurement Function and the TAUDnTTINm Input Interval Timer Function**

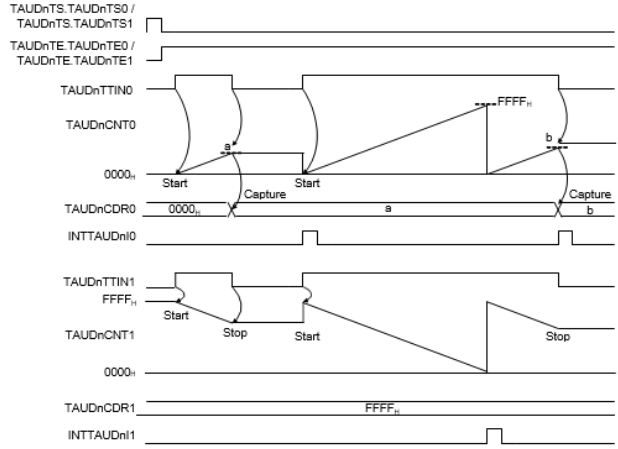
#### Combination of the TAUDnTTINm Input Signal Width Measurement Function and the Overflow Interrupt Output Function (at Measuring the TAUDnTTINm Width)

When the capture trigger is input simultaneously to TAUDnTTINm of both channels, INTTAUDnIm of the overflow interrupt output function (at measuring the TAUDnTTINm width) can detect the overflow when TAUDnCNTm of the TAUDnTTINm input signal width measurement function exceeds FFFFH.



**Figure 33.20 Combination of the TAUDnTTINm Input Signal Width Measurement Function and the Overflow Interrupt Output Function (at Measuring the TAUDnTTINm Width)**

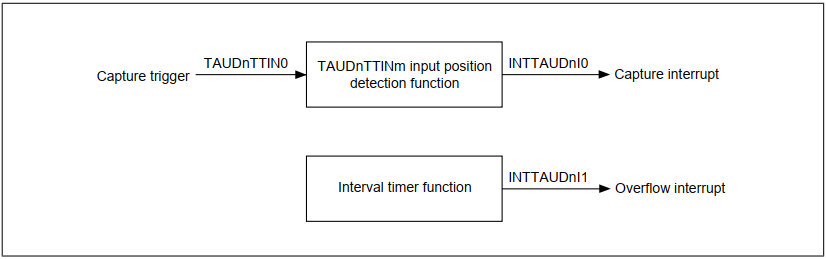
**Timing diagram**

****

**Figure 33.21 Interrupt Generation via Combination of the TAUDnTTINm Input Signal Width Measurement Function and the Overflow Interrupt Output Function (at Measuring the TAUDnTTINm Width)**

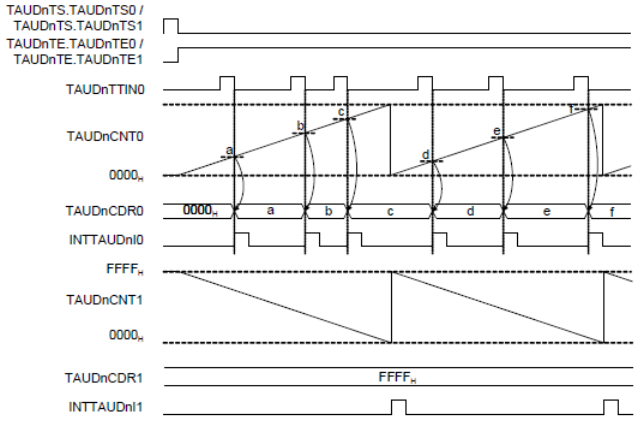
#### Combination of the TAUDnTTINm Input Position Detection Function and the Interval Timer Function

When the counters of both channels are enabled simultaneously, INTTAUDnIm of the interval timer function can detect the overflow when TAUDnCNTm of the TAUDnTTINm input position detection function exceeds FFFFH.

****

**Figure 33.22 Combination of the TAUDnTTINm Input Position Detection Function and the Interval Timer Function**

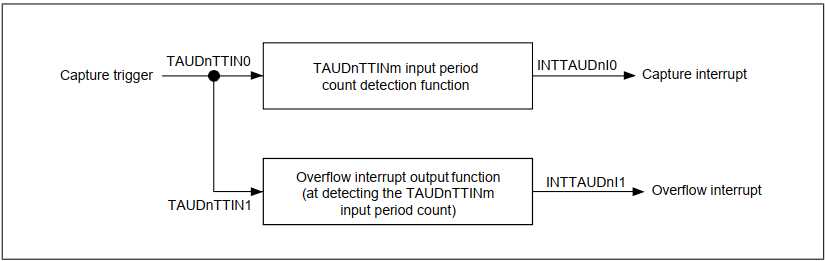
**Timing diagram**



**Figure 33.23 Interrupt Generation via Combination of the TAUDnTTINm Input Position Detection Function and the Interval Timer Function**

#### Combination of the TAUDnTTINm Input Period Count Detection Function and the Overflow Interrupt Output Function (at Detecting the TAUDnTTINm Input Period Count)

When the capture trigger is input simultaneously to TAUDnTTINm of both channels, INTTAUDnIm of the overflow interrupt output function (at detecting the TAUDnTTINm input period count) can detect the overflow when TAUDnCNTm of the TAUDnTTINm input period count detection function exceeds FFFFH.



**Figure 33.24 Combination of the TAUDnTTINm Input Period Count Detection Function and the Overflow Interrupt Output Function (at Detecting the TAUDnTTINm Input Period Count)**

**Timing diagram**

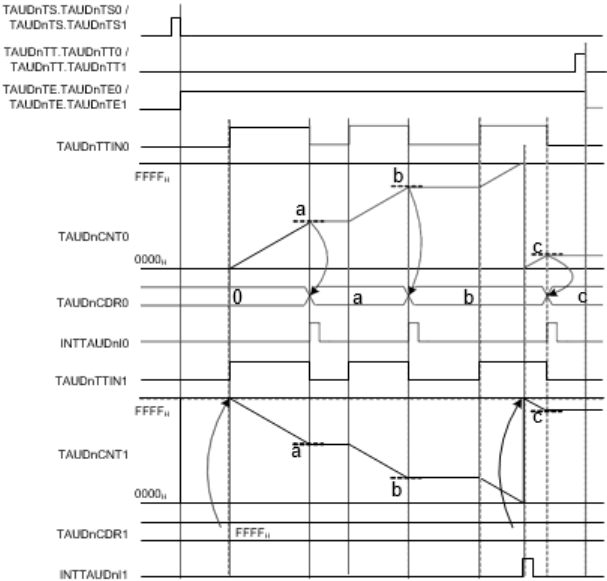
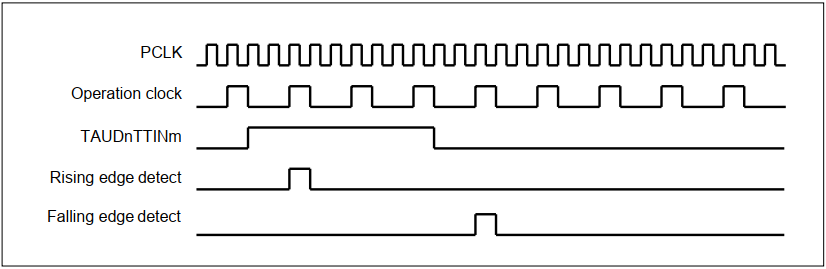


Figure 33.25 Interrupt Generation via Combination of the TAUDnTTINm Input Period Count Detection Function and the Overflow Interrupt Output Function (at Detecting the TAUDnTTINm Input Period Count)

### TAUDnTTINm Edge Detection

Edge detection is based on the operation clock. This means that an edge can only be detected at the next rising edge of the operation clock. This can lead to a maximum delay of one operation clock cycle.

**Figure 33.26 shows when edge detection takes place.**

****

**Figure 33.26 Basic Edge Detection Timing**

**Figure 33.26** shows an operation timing image. Actually, a noise filter or synchronization circuit

which is located between the TAUDnIm pin and TAUDn causes a delay time.

### Independent Channel Operation Functions

The following sections list the independent channel operation functions provided by the Timer Array

Unit D. For a general overview of independent channel operation, see Section 33.2, Overview.

This section describes functions that generate interrupts at regular intervals or with a specified delay.