

32-bit RISC Microcontroller

TXZ, TXZ+ Family

**Reference manual
32-bit Timer Event Counter
(T32A-B)**

Revision 2.3

2020-08

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION

Contents

| | |
|--|----|
| Preface | 5 |
| Related Documents | 5 |
| Conventions | 6 |
| Terms and Abbreviations | 8 |
| 1. Outline | 9 |
| 2. Configuration | 11 |
| 2.1. 16-bit Timer | 11 |
| 2.2. 32-bit Timer | 13 |
| 3. Function and Operation | 15 |
| 3.1. Clock Supply | 15 |
| 3.2. Prescaler | 15 |
| 3.3. Counters (16-bit Counter A/16-bit Counter B/ 32-bit Counter C) | 16 |
| 3.3.1. Count Clock | 16 |
| 3.3.2. Starting Operation | 16 |
| 3.3.3. Stopping Operation | 17 |
| 3.3.4. Count Operation | 19 |
| 3.3.5. Event count operation | 20 |
| 3.3.6. Pulse Counting | 20 |
| 3.3.7. Reloading the Counter | 23 |
| 3.4. Synchronous Operation | 24 |
| 3.5. Comparator | 25 |
| 3.6. Timer Register (<i>[T32AxRGA0/B0/C0]</i> , <i>[T32AxRGA1/B1/C1]</i>) | 25 |
| 3.7. Capture Control | 26 |
| 3.8. Output programmable rectangular wave (PPG) (<i>T32AxOUTA/T32AxOUTB/T32AxOUTC</i>) | 27 |
| 3.9. Interrupts | 28 |
| 3.9.1. Timer Interrupt | 28 |
| 3.9.2. Capture Interrupt | 29 |
| 3.9.3. Interrupt on Every Count | 29 |
| 3.10. DMA Request | 30 |
| 4. Registers | 31 |
| 4.1. List of Registers | 31 |
| 4.2. Details of the Common Register | 34 |
| 4.2.1. <i>[T32AxMOD]</i> (Mode Register) | 34 |
| 4.3. Details of Timer A Register | 35 |
| 4.3.1. <i>[T32AxRUNA]</i> (RUN Register A) | 35 |
| 4.3.2. <i>[T32AxCRA]</i> (Counter Control Register A) | 36 |
| 4.3.3. <i>[T32AxOUTCRA0]</i> (Output Control Register A0) | 37 |
| 4.3.4. <i>[T32AxOUTCRA1]</i> (Output Control Register A1) | 37 |
| 4.3.5. <i>[T32AxRGA0]</i> (Timer Register A0) | 38 |
| 4.3.6. <i>[T32AxRGA1]</i> (Timer Register A1) | 38 |

| | |
|---|----|
| 4.3.7. [T32AxTMRA] (Counter Capture Register A) | 38 |
| 4.3.8. [T32AxRELDA] (Counter Reload Register A) | 39 |
| 4.3.9. [T32AxCAPCRA] (Capture Control Register A) | 39 |
| 4.3.10. [T32AxCAPA0] (Capture Register A0) | 39 |
| 4.3.11. [T32AxCAPA1] (Capture Register A1) | 39 |
| 4.3.12. [T32AxIMA] (Interrupt Mask Register A) | 40 |
| 4.3.13. [T32AxSTA] (Status Register A) | 41 |
| 4.3.14. [T32AxDMAA] (DMA Request Enable Register A) | 41 |
| 4.4. Details of Timer B Register | 42 |
| 4.4.1. [T32AxRUNB] (RUN Register B) | 42 |
| 4.4.2. [T32AxCRB] (Counter Control Register B) | 43 |
| 4.4.3. [T32AxOUTCRB0] (Output Control Register B0) | 44 |
| 4.4.4. [T32AxOUTCRB1] (Output Control Register B1) | 44 |
| 4.4.5. [T32AxRGB0] (Timer Register B0) | 45 |
| 4.4.6. [T32AxRGB1] (Timer Register B1) | 45 |
| 4.4.7. [T32AxTMRB] (Counter Capture Register B) | 45 |
| 4.4.8. [T32AxRELDB] (Counter Reload Register B) | 46 |
| 4.4.9. [T32AxCAPCRB] (Capture Control Register B) | 46 |
| 4.4.10. [T32AxCAPB0] (Capture Register B0) | 46 |
| 4.4.11. [T32AxCAPB1] (Capture Register B1) | 46 |
| 4.4.12. [T32AxIMB] (Interrupt Mask Register B) | 47 |
| 4.4.13. [T32AxSTB] (Status Register B) | 48 |
| 4.4.14. [T32AxDMAB] (DMA Request Enable Register B) | 48 |
| 4.5. Details of Timer C Register | 49 |
| 4.5.1. [T32AxRUNC] (RUN Register C) | 49 |
| 4.5.2. [T32AxCRC] (Counter Control Register C) | 50 |
| 4.5.3. [T32AxOUTCRC0] (Output Control Register C0) | 51 |
| 4.5.4. [T32AxOUTCRC1] (Output Control Register C1) | 51 |
| 4.5.5. [T32AxRGC0] (Timer Register C0) | 52 |
| 4.5.6. [T32AxRGC1] (Timer Register C1) | 52 |
| 4.5.7. [T32AxTMRC] (Counter Capture Register C) | 53 |
| 4.5.8. [T32AxRELDC] (Counter Reload Register C) | 53 |
| 4.5.9. [T32AxCAPCRC] (Capture Control Register C) | 53 |
| 4.5.10. [T32AxCAPC0] (Capture Register C0) | 53 |
| 4.5.11. [T32AxCAPC1] (Capture Register C1) | 53 |
| 4.5.12. [T32AxIMC] (Interrupt Mask Register C) | 54 |
| 4.5.13. [T32AxSTC] (Status Register C) | 55 |
| 4.5.14. [T32AxDMAC] (DMA Request Enable Register C) | 56 |
| 4.5.15. [T32AxPLSCR] (Pulse Counter control Register C) | 56 |
| 5. Use Case | 58 |
| 5.1. Interval Timer | 58 |
| 5.2. Event Counter | 60 |
| 5.3. Programmable Rectangular Wave Output (PPG) | 61 |

| | |
|---|----|
| 5.4. Programmable Rectangular Wave Output (PPG) by External Trigger | 63 |
| 5.5. PPG Outputs by Synchronous Operation | 64 |
| 5.6. Control a stepping motor by PPG Outputs | 66 |
| 5.7. Frequency Measurement Using Capture Function | 67 |
| 5.8. Pulse Width Measurement Using Capture Function | 69 |
| 5.9. Time Difference Measurement Using Capture Function | 70 |
| 6. Revision History | 71 |
| RESTRICTIONS ON PRODUCT USE | 74 |

List of figures

| | |
|--|----|
| Figure 2.1 Block diagram of Timer A and Timer B in 16-bit timer | 11 |
| Figure 2.2 Block diagram of Timer C in 32-bit timer | 13 |
| Figure 3.1 State transition | 21 |
| Figure 3.2 Positive phase (Counter operation when <PDIR>=0) | 22 |
| Figure 3.3 Negative phase (Counter operation when <PDIR>=1) | 22 |
| Figure 3.4 Synchronous Operation | 24 |
| Figure 3.5 Example of Output of programmable rectangular wave using 16-bit Timer A | 27 |
| Figure 5.1 Interval Timer (Count up) | 58 |
| Figure 5.2 Interval Timer (Count down) | 59 |
| Figure 5.3 Event counter | 60 |
| Figure 5.4 Example of PPG output(Low active) | 61 |
| Figure 5.5 PPG output (<Period A, B>) | 62 |
| Figure 5.6 Example of PPG output by External Trigger | 63 |
| Figure 5.7 PPG Outputs by Synchronous Operation(Low active) | 65 |
| Figure 5.8 PPG Outputs by Synchronous Operation (Example of control of stepping motor) | 66 |
| Figure 5.9 Frequency Measurement Using Capture Function | 68 |
| Figure 5.10 Pulse Width Measurement Using Capture Function | 69 |
| Figure 5.11 Time Difference Measurement Using Capture Function | 70 |

List of tables

| | |
|--|----|
| Table 2.1 16-bit timer connection specification | 12 |
| Table 2.2 32-bit timer connection specification | 14 |
| Table 3.1 Note of timer register setting when output PPG | 19 |
| Table 3.2 2-phase pulse count mode | 21 |
| Table 3.3 Transition | 21 |
| Table 3.4 List of Timer Interrupt Sources | 28 |
| Table 3.5 Relationship table between of Capture interrupt and Capture register | 29 |
| Table 3.6 DMA request list | 30 |
| Table 4.1 Noise elimination <NF> | 57 |
| Table 6.1 Revision History | 71 |

Preface

Related Documents

| Document name |
|----------------------------------|
| Exception |
| Clock Control and Operation Mode |
| Product Information |

Conventions

- Numeric formats follow the rules as shown below:
Hexadecimal: 0xABC
Decimal: 123 or 0d123 – Only when it needs to be explicitly shown that they are decimal numbers.
Binary: 0b111 – It is possible to omit the "0b" when the number of bit can be distinctly understood from a sentence.
- "_N" is added to the end of signal names to indicate low active signals.
- It is called "assert" that a signal moves to its active level, "deassert" to its inactive level.
- When two or more signal names are referred, they are described like as [m: n].
Example: S[3: 0] shows four signal names S3, S2, S1 and S0 together.
- The characters surrounded by *[]* defines the register.
Example: *[ABCD]*
- "n" substitutes suffix number of two or more same kind of registers, fields, and bit names.
Example: *[XYZ1], [XYZ2], [XYZ3]-> [XYZn]*
- "x" substitutes suffix number or character of two or more same kind of units and channels in same register name in the Register List.
In case of unit, "x" means A, B, and C ..
Example: *[ADACR0], [ADBCR0], [ADCCR0]->[ADxCR0]*
In case of channel, "x" means 0, 1, and 2..
Example: *[T32A0RUNA], [T32A1RUNA], [T32A2RUNA]->[T32AxRUNA]*
- The bit range of a register is written like as [m: n].
Example: Bit[3: 0] expresses the range of bit 3 to 0.
- The configuration value of a register is expressed by either the hexadecimal number or the binary number.
Example: *[ABCD].EFG = 0x01* (hexadecimal), *[XYZn].VW = 1* (binary)
- Word and Byte represent the following bit length.
Byte: 8 bits
Half word: 16 bits
Word: 32 bits
Double word: 64 bits
- Properties of each bit in a register are expressed as follows:
R: Read only
W: Write only
R/W: Read and Write are possible
- Unless otherwise specified, register access supports only word access.
- The register defined as reserved must not be rewritten. Moreover, do not use the read value.
- The value read from the bit having default value of "-" is unknown.
- When a register containing both of writable bits and read-only bits is written, read-only bits should be written with their default value, In the cases that default is "-", follow the definition of each register.
- Reserved bits of the Write-only register should be written with their default value. In the cases that default is "-", follow the definition of each register.
- Do not use read-modified-write processing to the register of a definition which is different by writing and read out.

All other company names, product names, and service names mentioned herein may be trademarks of their respective companies.

Terms and Abbreviations

Some of abbreviations used in this document are as follows:

| | |
|------|------------------------------|
| DMA | Direct Memory Access |
| PPG | Programmable Pulse Generator |
| T32A | 32-bit Timer Event counter |

1. Outline

T32A can work as a 16-bit timer (timer A, timer B) of 2ch or a 32-bit timer (timer C) of 1ch by 1UNIT circuit unit.

| Function Classification | | Function | | A Functional Description or the range |
|-------------------------|-------------------------|----------------|-----------------------------|---|
| 16-bit timer Timer A | Time control | Interval timer | | This function generates interrupt to CPU at set interval time. |
| | Measurement control | Event counter | | Up counting or down counting or up/down counting can be selected as count operation. And when the count value matches the timer register, you can generate timer interrupt. |
| | | Capture | Frequency measurement | Capture count value on rising edge and falling edge of input pulse. You can calculate frequency from difference of capture data. |
| | | | Pulse width measurement | Capture count value on rising edge and falling edge of input pulse. You can calculate Pulse Width from difference of capture data. |
| | | | Time difference measurement | Capture count value on rising edge or falling edge of input pulse. You can calculate Time difference from difference of capture data. |
| | Rectangular wave output | PPG | | Can output rectangular wave of arbitrary frequency or arbitrary duty. |
| | Synchronous Operation | Counter start | | Timer counter start the count in synch with count start of master timer counter. |
| | | Counter stop | | Timer counter stop the count in synch with count stop of master timer counter. |
| | | Counter reload | | Timer counter is reloaded in synch with reload of master timer counter. |
| 16-bit timer Timer B | Time control | Interval timer | | This function generates interrupt to CPU at set interval time. |
| | Measurement control | Event counter | | Up counting or down counting or up/down counting can be selected as count operation. And when the count value matches the timer register, you can generate timer interrupt. |
| | | Capture | Frequency measurement | Capture count value on rising edge and falling edge of input pulse. You can calculate frequency from difference of capture data. |
| | | | Pulse width measurement | Capture count value on rising edge and falling edge of input pulse. You can calculate Pulse Width from difference of capture data. |
| | | | Time difference measurement | Capture count value on rising edge or falling edge of input pulse. You can calculate Time difference from difference of capture data. |
| | Rectangular wave output | PPG | | Can output rectangular wave of arbitrary frequency or arbitrary duty. |
| | Synchronous Operation | Counter start | | Timer counter start the count in synch with count start of master timer counter. |
| | | Counter stop | | Timer counter stop the count in synch with count stop of master timer counter. |
| | | Counter reload | | Timer counter is reloaded in synch with reload of master timer counter. |

| Function Classification | | Function | | A Functional Description or the range |
|-------------------------|-------------------------|---|-----------------------------|---|
| 32-bit timer Timer C | Time control | Interval timer | | This function generates interrupt to CPU at set interval time. |
| | Measurement control | Event counter | | Up counting or down counting or up/down counting can be selected as count operation. And when the count value matches the timer register, you can generate timer interrupt. |
| | | Pulse count | 1-phase pulse count | Counts the variation in the inputs of T32AxINC0 or T32AxINC1. The counter increment or decrement is selected according to the setting. |
| | | | 2-phase pulse count | Increment or decrement of counter according to the variation in the combination of inputs both T32AxINC0 and T32AxINC1. |
| | | Capture | Frequency measurement | Capture count value on rising edge and falling edge of input pulse. You can calculate frequency from difference of capture data. |
| | | | Pulse width measurement | Capture count value on rising edge and falling edge of input pulse. You can calculate Pulse Width from difference of capture data. |
| | | | Time difference measurement | Capture count value on rising edge or falling edge of input pulse. You can calculate Time difference from difference of capture data. |
| | Rectangular wave output | PPG | | Can output rectangular wave of arbitrary frequency or arbitrary duty. |
| | Synchronous Operation | Counter start | | Timer counter start the count in synch with count start of master timer counter. |
| | | Counter stop | | Timer counter stop the count in synch with count stop of master timer counter. |
| Counter reload | | Timer counter is reloaded in synch with reload of master timer counter. | | |

2. Configuration

T32A consists of two 16-bit timers, which operate as Timer A and Timer B respectively. Timer A and Timer B connect and it operates as 32-bit timer C.

2.1. 16-bit Timer

In 16-bit timer, the T32A is configured with the independent 16-bit Timer A and Timer B.

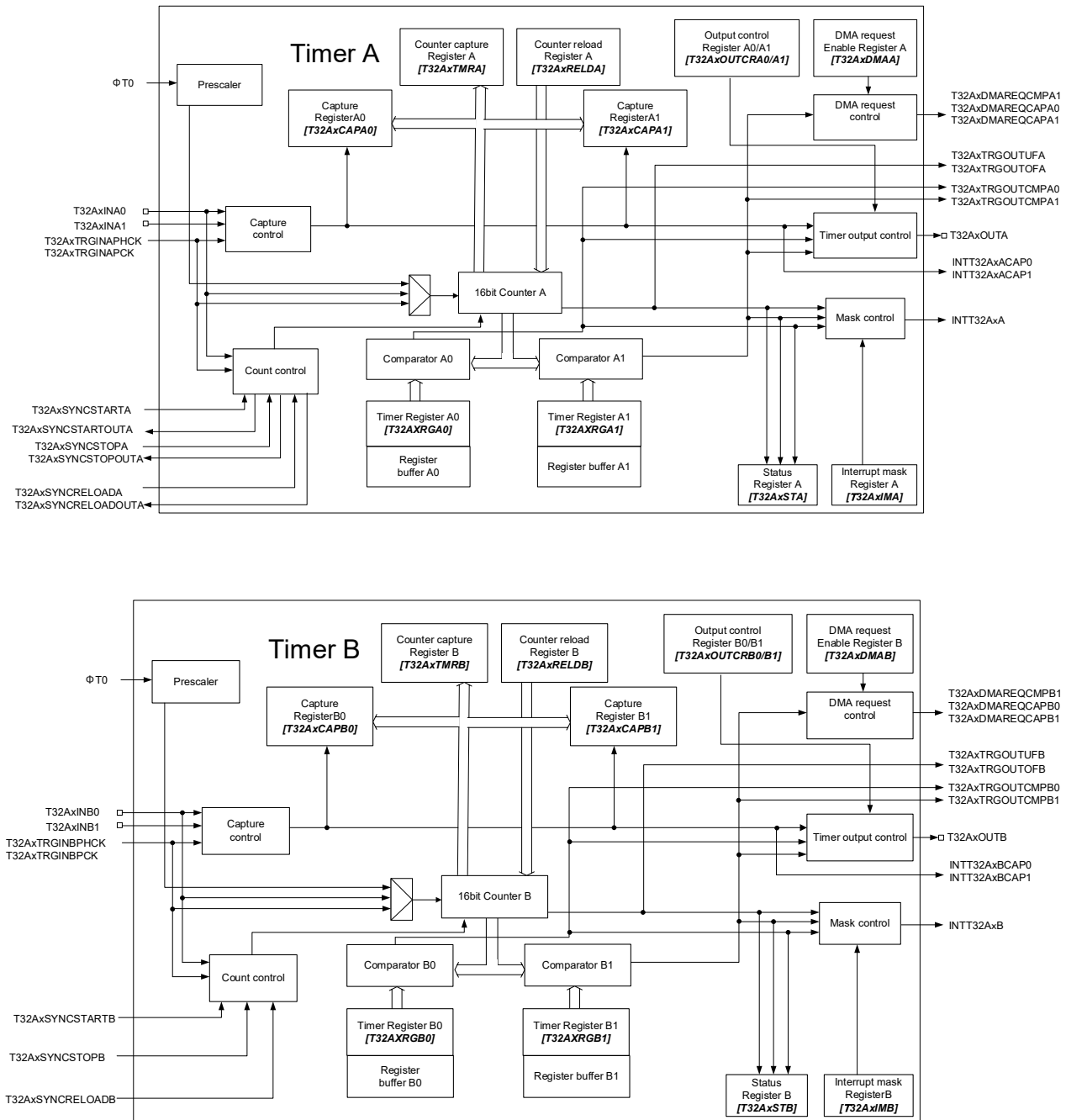


Figure 2.1 Block diagram of Timer A and Timer B in 16-bit timer

Table 2.1 16-bit timer connection specification

| No. | Symbol | Signal name | I/O | Related Reference Manual |
|-----|---------------------|---|--------|----------------------------------|
| 1 | $\Phi T0$ | Prescaler clock | Input | Clock Control and Operation Mode |
| 2 | T32AxINA0 | Timer A external trigger input0 | Input | Product Information |
| 3 | T32AxINA1 | Timer A external trigger input1 | Input | Product Information |
| 4 | T32AxTRGINAPHCK | Timer A trigger from other timer | Input | Product Information |
| 5 | T32AxTRGINAPCK | Timer A internal trigger input | Input | Product Information |
| 6 | T32AxSYNCSTARTA | Timer A synchronous start at trigger input | Input | Product Information |
| 7 | T32AxSYNCSTARTOUTA | Timer A trigger output for synchronous start | Output | Product Information |
| 8 | T32AxSYNCSTOPA | Timer A synchronous stop at trigger input | Input | Product Information |
| 9 | T32AxSYNCSTOPOUTA | Timer A trigger output for synchronous stop | Output | Product Information |
| 10 | T32AxSYNCRELOADA | Timer A synchronous reload at trigger input | Input | Product Information |
| 11 | T32AxSYNCRELOADOUTA | Timer A trigger output for synchronous reload | Output | Product Information |
| 12 | T32AxDMAREQCMPA1 | DMA request at match A1 register | Output | Product Information |
| 13 | T32AxDMAREQCAPA0 | DMA request at capture A0 register | Output | Product Information |
| 14 | T32AxDMAREQCAPA1 | DMA request at capture A1 register | Output | Product Information |
| 15 | T32AxTRGOUTUFA | Timer A underflow trigger | Output | Product Information |
| 16 | T32AxTRGOUTOFA | Timer A overflow trigger | Output | Product Information |
| 17 | T32AxTRGOUTCMPA0 | Timer register A0 match trigger | Output | Product Information |
| 18 | T32AxTRGOUTCMPA1 | Timer register A1 match trigger | Output | Product Information |
| 19 | T32AxOUTA | Timer A output | Output | Product Information |
| 20 | INTT32AxACAPO | Timer A capture register0 interrupt | Output | Exception |
| 21 | INTT32AxACAP1 | Timer A capture register1 interrupt | Output | Exception |
| 22 | INTT32AxA | Timer A match, overflow and underflow interrupt | Output | Exception |
| 23 | T32AxINB0 | Timer B external trigger input0 | Input | Product Information |
| 24 | T32AxINB1 | Timer B external trigger input1 | Input | Product Information |
| 25 | T32AxTRGINBPHCK | Timer B trigger from other timer | Input | Product Information |
| 26 | T32AxTRGINBPCK | Timer B internal trigger input | Input | Product Information |
| 27 | T32AxSYNCSTARTB | Timer B synchronous start at trigger input | Input | Product Information |
| 28 | T32AxSYNCSTOPB | Timer B synchronous stop at trigger input | Input | Product Information |
| 29 | T32AxSYNCRELOADB | Timer B synchronous reload at trigger input | Input | Product Information |
| 30 | T32AxDMAREQCMPB1 | DMA request at match B1 register | Output | Product Information |
| 31 | T32AxDMAREQCAPB0 | DMA request at capture B0 register | Output | Product Information |
| 32 | T32AxDMAREQCAPB1 | DMA request at capture B1 register | Output | Product Information |
| 33 | T32AxTRGOUTUFB | Timer B underflow trigger | Output | Product Information |
| 34 | T32AxTRGOUTOFB | Timer B overflow trigger | Output | Product Information |
| 35 | T32AxTRGOUTCMPB0 | Timer register B0 match trigger | Output | Product Information |
| 36 | T32AxTRGOUTCMPB1 | Timer register B1 match trigger | Output | Product Information |
| 37 | T32AxOUTB | Timer B output | Output | Product Information |
| 38 | INTT32AxBCAPO | Timer B capture register0 interrupt | Output | Exception |
| 39 | INTT32AxBCAP1 | Timer B capture register1 interrupt | Output | Exception |
| 40 | INTT32AxB | Timer B match, overflow and underflow interrupt | Output | Exception |

2.2. 32-bit Timer

In 32-bit timer, the T32A is configured with the 32-bit Timer C.

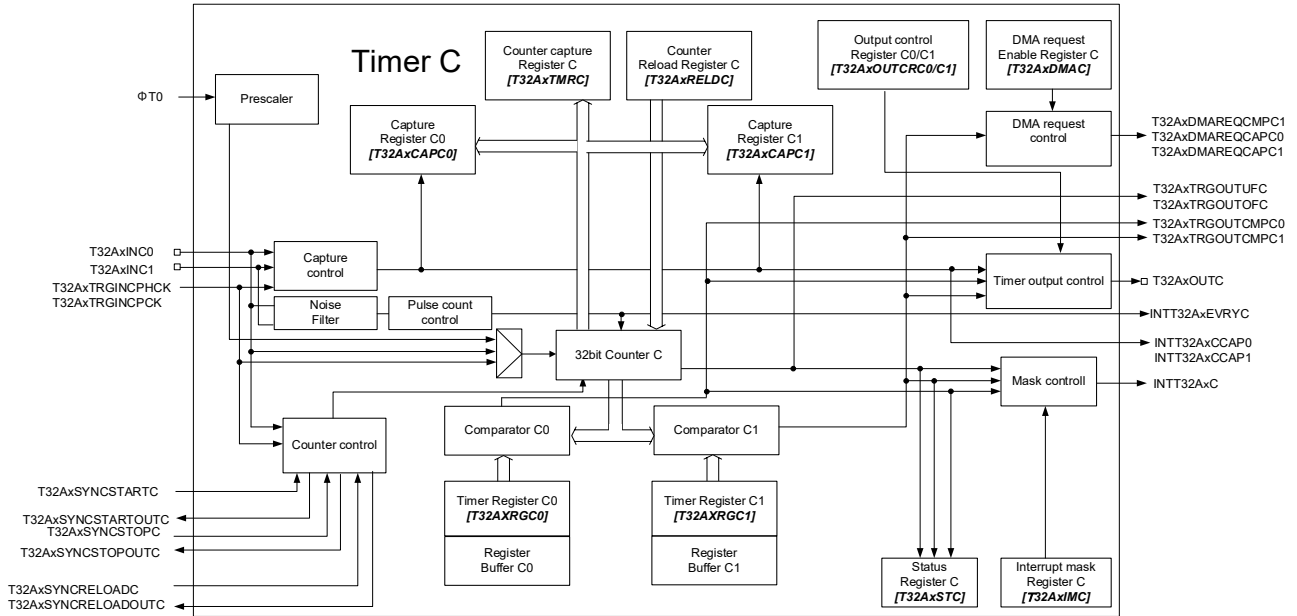


Figure 2.2 Block diagram of Timer C in 32-bit timer

Table 2.2 32-bit timer connection specification

| No. | Symbol | Signal name | I/O | Related Reference Manual |
|-----|---------------------|---|--------|----------------------------------|
| 1 | $\Phi T0$ | Prescaler clock | Input | Clock Control and Operation Mode |
| 2 | T32AxINC0 | Timer C external trigger input0 | Input | Product Information |
| 3 | T32AxINC1 | Timer C external trigger input1 | Input | Product Information |
| 4 | T32AxTRGINCPHCK | Timer C trigger from other timer | Input | Product Information |
| 5 | T32AxTRGINCPCK | Timer C internal trigger input | Input | Product Information |
| 6 | T32AxSYNCSTARTC | Timer C synchronous start at trigger input | Input | Product Information |
| 7 | T32AxSYNCSTARTOUTC | Timer C trigger output for synchronous start | Output | Product Information |
| 8 | T32AxSYNCSTOPC | Timer C synchronous stop at trigger input | Input | Product Information |
| 9 | T32AxSYNCSTOPOUTC | Timer C trigger output for synchronous stop | Output | Product Information |
| 10 | T32AxSYNCRELOADC | Timer C synchronous reload at trigger input | Input | Product Information |
| 11 | T32AxSYNCRELOADOUTC | Timer C trigger output for synchronous reload | Output | Product Information |
| 12 | T32AxDMAREQCMPC1 | DMA request at match C1 register | Output | Product Information |
| 13 | T32AxDMAREQCAPC0 | DMA request at capture C0 register | Output | Product Information |
| 14 | T32AxDMAREQCAPC1 | DMA request at capture C1 register | Output | Product Information |
| 15 | T32AxTRGOUTUFC | Timer C underflow trigger | Output | Product Information |
| 16 | T32AxTRGOUTOFC | Timer C overflow trigger | Output | Product Information |
| 17 | T32AxTRGOUTCMPC0 | Timer register C0 match trigger | Output | Product Information |
| 18 | T32AxTRGOUTCMPC1 | Timer register C1 match trigger | Output | Product Information |
| 19 | T32AxOUTC | Timer C output | Output | Product Information |
| 20 | INTT32AxCCAP0 | Timer C capture register0 interrupt | Output | Exception |
| 21 | INTT32AxCCAP1 | Timer C capture register1 interrupt | Output | Exception |
| 22 | INTT32AxC | Timer C match, overflow and underflow interrupt | Output | Exception |
| 23 | INTT32AxEVRYC | Every count interrupt | Output | Exception |

3. Function and Operation

T32A is composed two 16-bit timer that can be used Timer A and Timer B. Also it can use Timer C that is connected Timer A and Timer B as 32-bit timer. When use Timer C, Timer A and Timer B cannot be used.

The 16-bit timer and the 32-bit timer mode setting are specified with as follows:

| Operation mode |
|----------------------------|
| <i>[T32AxMOD]</i> <MODE32> |

3.1. Clock Supply

When T32A is used, the corresponding clock enable bits should be set to "1" (Clock supply) in fsys supply stop register A (*[CGFSYSENA]* and *[CGFSYSMENA]*), fsys supply stop register B (*[CGFSYSENB]* and *[CGFSYSMENB]*), fsys supply stop register C (*[CGFSYSMENC]*), and fc supply stop register (*[CGFCEN]*).

The corresponding registers and the bit locations depend on a product. Some products do not have all registers. For the details, refer to "Clock Control and Operation Mode" in Reference manual.

3.2. Prescaler

The prescaler divides the prescaler clock ($\Phi T0$) to generate the source clock for the counter.

In 16-bit timer, the division ratio can be selected for Timer A and Timer B respectively. In 32-bit timer, the division ratio can be selected with Timer C. The division ratio specified with as follows:

| Prescaler division ratio selection bits | |
|---|----------------------------|
| Timer A | <i>[T32AxCRA]</i> <PRSCLA> |
| Timer B | <i>[T32AxCRB]</i> <PRSCLB> |
| Timer C | <i>[T32AxCRC]</i> <PRSCLC> |

For the setting of $\Phi T0$, refer to Reference manual "Clock Control and Operation Mode".

3.3. Counters (16-bit Counter A/16-bit Counter B/ 32-bit Counter C)

In 16-bit timer, the Timer A/Timer B operates as 16-bit counters respectively. In 32-bit timer, the Timer C operates as a 32-bit counter.

3.3.1. Count Clock

As for the count clock of Timer A, Timer B, or Timer C, it can be selected from a prescaler output, an output of other timer, an external trigger (T32AxINA0/T32AxINB0/T32AxINC0), or an internal trigger. The count clock can be selected from the bits in the table below.

| The count clock selection bits | |
|--------------------------------|------------------|
| Timer A | [T32AxCRA]<CLKA> |
| Timer B | [T32AxCRB]<CLKB> |
| Timer C (Note) | [T32AxCRC]<CLKC> |

Note: When pulse count operation in 32-bit timer is selected ([T32AxCRC] <UPDNC>=11), the setting above is ignored.

3.3.2. Starting Operation

When "1" is set to the timer operation control bit of each timer, the value in the counter reload register is reloaded (initial reloading). When the start conditions are established, the timer operation starts.

| Counter Reload Register | |
|-------------------------|--------------|
| Timer A | [T32AxRELDA] |
| Timer B | [T32AxRELDB] |
| Timer C | [T32AxRELDC] |

| Timer operation control | |
|-------------------------|-------------------|
| Timer A | [T32AxRUNA]<RUNA> |
| Timer B | [T32AxRUNB]<RUNB> |
| Timer C | [T32AxRUNC]<RUNC> |

When "1" is set to Timer operation control bits, make sure that the timer is stopping. About timer stopping, please refer to "3.3.3 Stopping Operation".

The counter operation is started by a register setting using program, internal triggers, external triggers, output from other timer, or channel synchronous with other channels.

When the pulse counting in 32-bit timer is used, an external trigger cannot be used as the factor to start the counter.

The factor of starting the counter is specified with the counter start condition setting bits.

| The counter start condition setting bits | |
|--|---------------------|
| Timer A | [T32AxCRA] <STARTA> |
| Timer B | [T32AxCRB] <STARTB> |
| Timer C | [T32AxCRC] <STARTC> |

To start the T32A by program, set "000" to the counter start condition setting bits.

Below describes the factor of count operation start.

(1) Timer operation start by program

When "1" is set to "starts the operation by program" in table below, the counter starts operation.

| Starts the operation by program | |
|---------------------------------|----------------------|
| Timer A | [T32AxRUNA]<SFTSTAA> |
| Timer B | [T32AxRUNB]<SFTSTAB> |
| Timer C | [T32AxRUNC]<SFTSTAC> |

(2) Internal trigger

The counter starts operation by an internal trigger from other peripheral function. The internal trigger is selected outside of the T32A. Do not select the same internal trigger to start and stop the counter.

(3) External trigger

The counter starts operation on the rising/falling edge of an external signal (T32AxINA0, T32AxINB0, or T32AxINC0). Do not select the same edge for starting the counter and stopping the counter.

(4) Timer output from other timer

The counter starts operation on the rising/falling edge of the output signal from the other timer. Do not select the same edge for starting the counter and stopping the counter.

(5) Synchronous start

The counter can start synchronously with start of other timer. Synchronous start setting should be specified to the slave timer. For more information, see "3.4 synchronous operation".

3.3.3. Stopping Operation

To check whether the counter is operating or stopping of each timer, use the operation flag of timer.

| Operation flag of Timer | |
|-------------------------|----------------------|
| Timer A | [T32AxRUNA]<RUNFLGA> |
| Timer B | [T32AxRUNB]<RUNFLGB> |
| Timer C | [T32AxRUNC]<RUNFLGC> |

The counter operation is stopped by register setting using program, internal triggers, external triggers, outputs from others timers, channel synchronous with other channels, or compare match between the counter and timer register1.

| Timer register1 | |
|-----------------|-------------|
| Timer A | [T32AxRGA1] |
| Timer B | [T32AxRGB1] |
| Timer C | [T32AxRGC1] |

When the pulse counting in timer C is used ([T32AxCRC] <UPDNC>=11), an external trigger cannot be used as the factor to stop the counter.

The factor of stopping the counter is specified with the counter stop condition setting bits.

| The counter stop condition setting bits | |
|---|---------------------------|
| Timer A | [T32AxCRA] <STOPA> |
| Timer B | [T32AxCRB] <STOPB> |
| Timer C | [T32AxCRC] <STOPC> |

To stop the T32A by program, set "000" to the counter stop condition setting bits.

Below describe the factor of count operation start/stop

(1) Timer operation stop by program

When "1" is set to "Stop the operation by program" in table below, the counter stops operation.

| Stop the operation by program | |
|-------------------------------|------------------------------|
| Timer A | [T32AxRUNA] <SFTSTPA> |
| Timer B | [T32AxRUNB] <SFTSTPB> |
| Timer C | [T32AxRUNC] <SFTSTPC> |

(2) Internal trigger

The counter stops operation by an internal trigger from other peripheral function. The internal trigger is selected outside of the T32A. Do not select the same internal trigger to start and stop the counter.

(3) External trigger

The counter stops operation on the rising/falling edge of an external signal (T32AxINA0, T32AxINB0, or T32AxINC0). Do not select the same edge for starting the counter and stopping the counter.

(4) Signal output from other timer

The counter stops operation on the rising/falling edge of the signal output from the other timer. Do not select the same edge for starting the counter and stopping the counter.

(5) Synchronous start

The counter can stop synchronously with start of other timer. Synchronous stop setting should be specified to the slave timer. For more information, see "3.4 synchronous operation".

(6) Compare match between the counter and timer register1

The counter stops when a compare match between the counter and Timer Register 1 is detected.

| Timer register1 | |
|-----------------|--------------------|
| Timer A | [T32AxRGA1] |
| Timer B | [T32AxRGB1] |
| Timer C | [T32AxRGC1] |

3.3.4. Count Operation

Up counting or down counting or up/down counting can be selected as the counter operation.

| counter operation select | |
|--------------------------|--------------------|
| Timer A | [T32AxCRA] <UPDNA> |
| Timer B | [T32AxCRB] <UPDNB> |
| Timer C | [T32AxCRC] <UPDNC> |

When up counting or down counting is specified, after counter starts, if the reload condition is established, the counter is reloaded. The counter repeats up or down counting until the stop condition is established. For reload condition, refer to "3.3.7 Reloading the Counter"

When up/down counting is specified, the counter starts up counting until the counter matches Timer register1. After the match, the counter operation changes to down counting until the counter value reaches "0". Then the counter operation changes to up counting. The counter repeats this operation until the stop condition is established.

| Timer register1 | |
|-----------------|-------------|
| Timer A | [T32AxRGA1] |
| Timer B | [T32AxRGB1] |
| Timer C | [T32AxRGC1] |

To check whether the counter is operating or stopping, use the operation flag of timer.

| The operation flag of Timer | |
|-----------------------------|-----------------------|
| Timer A | [T32AxRUNA] <RUNFLGA> |
| Timer B | [T32AxRUNB] <RUNFLGB> |
| Timer C | [T32AxRUNC] <RUNFLGC> |

In the counting operation at PPG output, ensure that the relationship in the following table holds for Timer register0 ([T32AxRGA0]/[T32AxRGB0]/[T32AxRGC0]), Timer register1 ([T32AxRGA1]/[T32AxRGB1]/[T32AxRGC1]), and Counter reload register ([T32AxRELDA]/[T32AxRELDB]/[T32AxRELDC])

Table 3.1 Note of timer register setting when output PPG

| Count operation | Setting condition |
|------------------|--|
| Up counting | Setting range of [T32AxRGx1]: $[T32AxRGx1] \geq [T32AxRELDx] + 2$ Setting range of [T32AxRGx0]: $[T32AxRELDx] \leq [T32AxRGx0] \leq [T32AxRGx1]$ Setting of output control register: set or clear |
| Down counting | Setting range of [T32AxRGx1]: $[T32AxRGx1] \leq [T32AxRELDx] - 2$ Setting range of [T32AxRGx0]: $[T32AxRELDx] \geq [T32AxRGx0] \geq [T32AxRGx1]$ Setting of output control register: set or clear |
| Up/down counting | Setting range of [T32AxRGx1]: $[T32AxRGx1] \geq 2$ AND $[T32AxRGx1] = [T32AxRELDx]$ Setting range of [T32AxRGx0]: $0 < [T32AxRGx0] < [T32AxRGx1]$ Setting of output control register: reverse Set to [T32AxRGx1] before set to [T32AxRGx0]. |

3.3.5. Event count operation

The event count operates in Timer A, Timer B and Timer C respectively.

Specified external trigger, internal trigger and signal output from other timer as count clock, it can be event counter.

Select up counting as the counter operation. Count up by external trigger (On the rising edge or on the falling edge), signal output from the other timer (On the rising edge or on the falling edge) or internal trigger. You can read following register as count of events.

| Counter Capture Register | |
|--------------------------|-------------|
| Timer A | [T32AxTMRA] |
| Timer B | [T32AxTMRB] |
| Timer C | [T32AxTMRC] |

3.3.6. Pulse Counting

Pulse counting operates in Timer C.

When "11" is set to the counter operation bit, the T32A enters pulse count mode.

| Counter operation | |
|-------------------|--------------------|
| Timer C | [T32AxCRC] <UPDNC> |

After above setting, select 2-phase pulse count mode or 1-phase pulse count mode with as follows:

| Pulse counter mode | |
|--------------------|----------------------|
| Timer C | [T32AxPLSCR] <PMODE> |

In 1-phase pulse count mode, the counter is incremented or decremented by the change of input of T32AxINC0 or T32AxINC1.

In 2-phase pulse count mode, the counter is incremented or decremented by the combination of the inputs of T32AxINC0 and T32AxINC1.

When the counter overflows, the counter becomes 0x00000000; when the counter underflow, the counter becomes 0xFFFFFFFF. In both cases, the counter continues counting and timer interrupt occurs.

Timer interrupt occur by compare match timer register C. For detail about timer interrupt, refer to "3.9.1 Timer Interrupt".

A noise filter can be inserted to T32AxINC0/T32AxINC1 with the noise elimination time for T32AxINC0 /T32AxINC1 bit as follows:

| Noise elimination time for T32AxINC0/T32AxINC1 | |
|--|-------------------|
| Timer C | [T32AxPLSCR] <NF> |

(1) 1-phase pulse count mode

This mode counts up or counts down according to the level of input pulse to T32AxINC0 or T32AxINC1.

Specified condition of when up counting or when down counting, can select with count condition select bit.

| Count condition specified | |
|---------------------------|--------------------|
| Up counting | [T32AxPLSCR] <PUP> |
| Down counting | [T32AxPLSCR] <PDN> |

Do not set the same value to <PUP> and <PDN>.

(2) 2-phase pulse count mode

In this mode the counter counts up or down according to the change of the signal level input to T32AxINC0/T32AxINC1. By the combination of input signal levels, there are the following four states. These four states show the signals input to T32AxINC0 / T32AxINC1 in hexadecimal notation.

Table 3.2 2-phase pulse count mode

| T32AxINC1 | T32AxINC0 | STATE |
|-----------|-----------|-------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 2 |
| 1 | 1 | 3 |

The figure below shows the state transition from/to up counting to/from down counting.

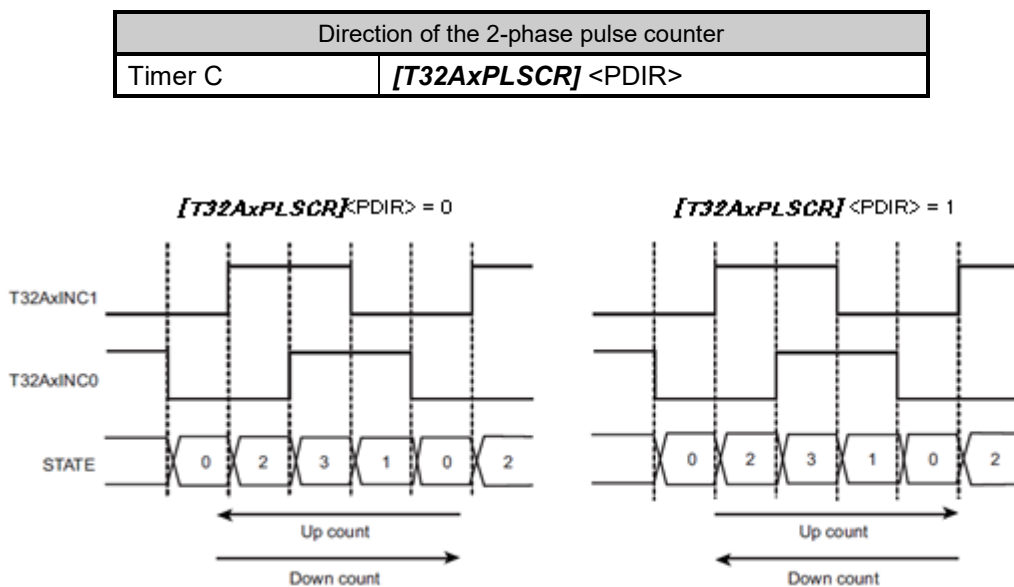


Figure 3.1 State transition

When $[T32AxPLSCR] \langle PDIR \rangle = 0$ and the rising edge of T32AxINC0 changes before T32AxINC1, counter is up counting. When the rising edge of T32AxINC0 change later than T32AxINC1 counter is down counting.

When $[T32AxPLSCR] \langle PDIR \rangle = 1$ and the rising edge of T32AxINC1 changes before T32AxINC0, counter is down counting. When the rising edge of T32AxINC1 change later than T32AxINC0 counter is up counting.

The state transitions of $0 \rightarrow 3$, $3 \rightarrow 0$, $1 \rightarrow 2$, and $2 \rightarrow 1$, in which the counter is not changed, are state errors. These are the interrupt factors (INTT32AxC).

Table 3.3 Transition

| | Up counting | | Down counting | |
|--|-------------------|------------------|-------------------|------------------|
| | Before transition | After transition | Before transition | After transition |
| Positive phase $\langle PDIR \rangle = 0$ | 0 | 1 | 0 | 2 |
| | 1 | 3 | 2 | 3 |
| | 3 | 2 | 3 | 1 |
| | 2 | 0 | 1 | 0 |
| Negative phase $\langle PDIR \rangle = 1$ | 0 | 2 | 0 | 1 |
| | 2 | 3 | 1 | 3 |
| | 3 | 1 | 3 | 2 |
| | 1 | 0 | 2 | 0 |

(a) Positive phase (Counter operation when $\langle \text{PDIR} \rangle = 0$)

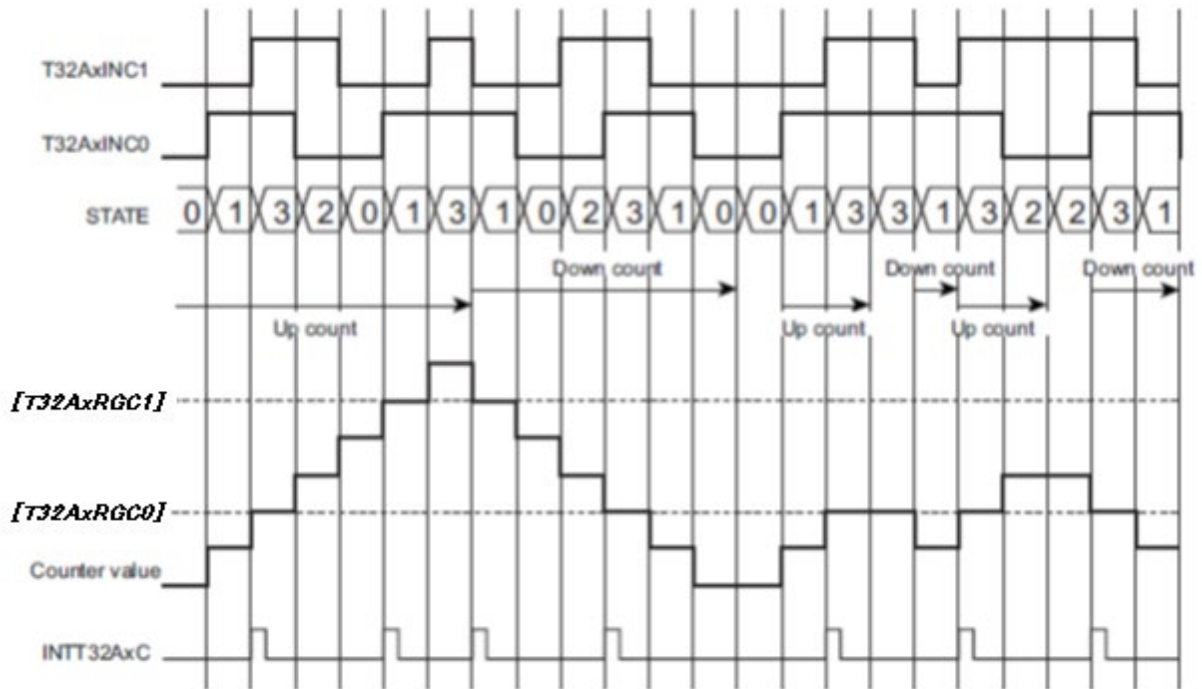


Figure 3.2 Positive phase (Counter operation when $\langle \text{PDIR} \rangle = 0$)

(b) Negative phase (Counter operation when $\langle \text{PDIR} \rangle = 1$)

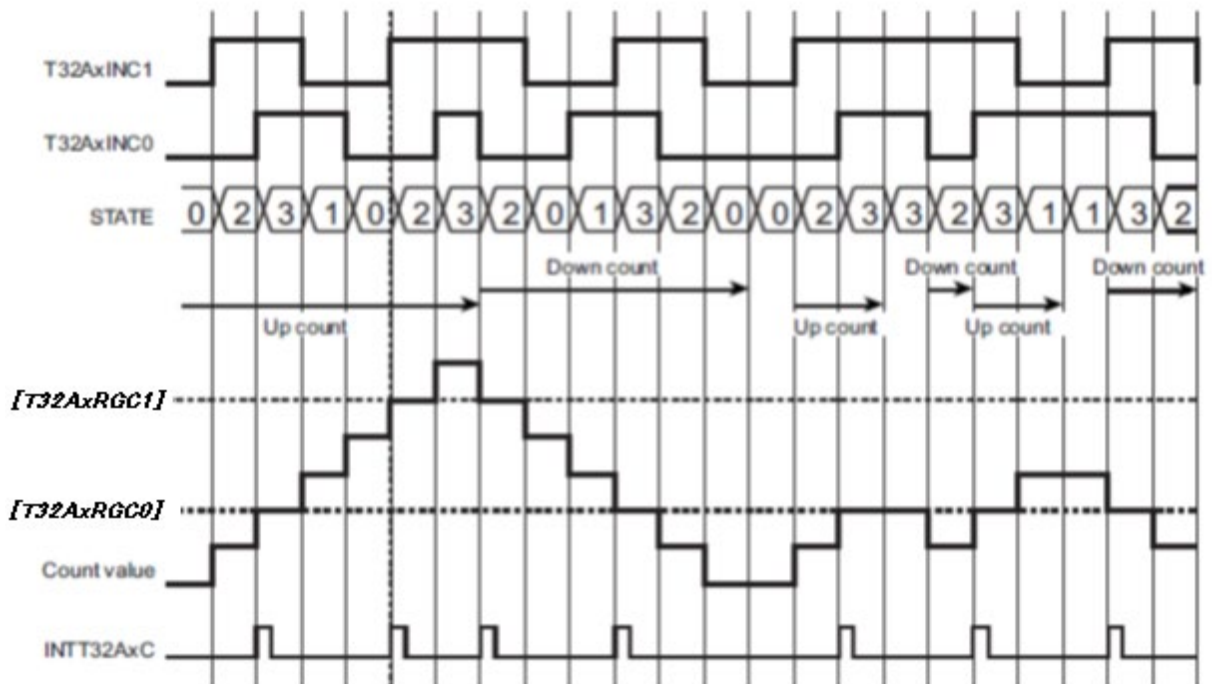


Figure 3.3 Negative phase (Counter operation when $\langle \text{PDIR} \rangle = 1$)

3.3.7. Reloading the Counter

You can re-setup the counter value at an arbitrary timing. The counter is reloaded by an internal trigger, external trigger, signal output from other timer, channel synchronous, compare match between the counter and timer register1.

The reload condition can be set by the counter reload condition bit in the table below.

| Counter Reload condition | |
|--------------------------|--------------------|
| Timer A | [T32AxCRA] <RELDA> |
| Timer B | [T32AxCRB] <RELDB> |
| Timer C | [T32AxCRC] <RELDC> |

The reload value can be set by the counter reload register in the table below.

| Counter reload register | |
|-------------------------|--------------|
| Timer A | [T32AxRELDA] |
| Timer B | [T32AxRELDB] |
| Timer C | [T32AxRELDC] |

When the reload value is set to "0", the counter is cleared. This can be used as counter clearing function. The timer and timer register1 are shown in the table below.

| Timer register1 | |
|-----------------|-------------|
| Timer A | [T32AxRGA1] |
| Timer B | [T32AxRGB1] |
| Timer C (Note) | [T32AxRGC1] |

Note: When the pulse count operation in Timer C is used, an external trigger cannot be used as the factor to reload the counter.

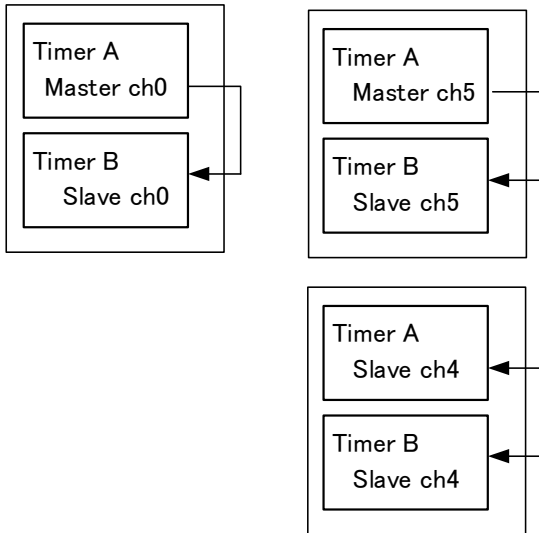
Counter reload condition is as following.

- (1) compare match between the counter and timer register1
The counter is reloaded when a compare match between the counter and timer register 1.
- (2) External trigger
The counter is reloaded or cleared on the rising/falling edge of an external signal (T32AxINA0/T32AxINB0/T32AxINC0).
- (3) Other timer outputs
The counter is reloaded or cleared on the rising/falling edge of a signal output from other timer.
- (4) Internal trigger
The counter is reloaded by an internal trigger from other peripheral functions. An internal trigger is specified outside of the T32A.
- (5) Channel synchronization
The counter can be synchronized with other timer's reloading. Synchronous start setting should be specified to the slave timer.
- (6) No reloading
Set the counter as free-running without reloading.

3.4. Synchronous Operation

Multiple timers can synchronously be started, reloaded, or stopped. Timer A or Timer C as a master timer and multiple slaves can synchronously operate. For the channel connection of Synchronous Operation, refer to "Product Information" of reference manual.

Connection example in 16-bit mode



Connection example in 32-bit mode

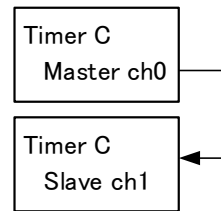


Figure 3.4 Synchronous Operation

3.5. Comparator

The comparator compares the specified value of the timer register and the counter. When a match is detected, a compare match signal is output. This compare match signal is a factor of interrupt requests or the factor to reverse the timer output. The compare match signal is also output as an internal trigger to other channels of the T32A or other peripheral functions.

3.6. Timer Register ([T32AxRGA0/B0/C0], [T32AxRGA1/B1/C1])

The timer registers are used to set the compare value for the counter. Two registers are provided for Timer A, Timer B, and Timer C each.

The timer registers are configured in double-buffering and are paired with the register buffers respectively.

Double buffering is initially disabled.

When double buffering is disabled, the timer register operates as a single buffer. The compare value is written to timer register 0/1 directly for the counter.

When double buffering is enabled, if the counter matches timer register1, the compare value is transferred from register buffer to timer register1. And timer register 0 is updated at same time.

Using double buffering, you can update the compare value regardless of update timing of the timer register.

| Timer register 1 | |
|------------------|-------------|
| Timer A | [T32AxRGA1] |
| Timer B | [T32AxRGB1] |
| Timer C | [T32AxRGC1] |

| Timer register 0 | |
|------------------|-------------|
| Timer A | [T32AxRGA0] |
| Timer B | [T32AxRGB0] |
| Timer C | [T32AxRGC0] |

Double buffering is enabled/disabled with double-buffering control bits in table below.

| Double-buffering control bit | |
|------------------------------|------------------|
| Timer A | [T32AxCRA]<WBFA> |
| Timer B | [T32AxCRB]<WBFB> |
| Timer C | [T32AxCRC]<WBFC> |

Note that, while the counter is stopping, even if double buffering is enabled, data is written to the timer register directly. This operation is single buffering.

3.7. Capture Control

Can capture counter value at any timing. There are 2 ways as below.

(1) Using external triggers, signal output from the other timer, or internal triggers

A counter value is captured to capture register0 and capture register1 at the change of the level of an external trigger or signal output from other timer; or occurrence of an internal trigger.

| Capture register0 | |
|-------------------|--------------|
| Timer A | [T32AxCAPA0] |
| Timer B | [T32AxCAPB0] |
| Timer C | [T32AxCAPC0] |

| Capture register1 | |
|-------------------|--------------|
| Timer A | [T32AxCAPA1] |
| Timer B | [T32AxCAPB1] |
| Timer C | [T32AxCAPC1] |

Can control capture register by selecting with capture timing sets bits.

A capture timing can be selected from the following 7 timings: on the rising/falling edge of T32AxINA0/T32AxINB0/T32AxINC0, on the rising/falling edge of T32AxINA1/T32AxINB1/T32AxINC1, on the rising/falling edge of a signal output from other timer, or the occurrence of internal trigger.

| Capture timing sets bits of capture register0 | |
|---|-----------------------|
| Timer A | [T32AxCAPCRA]<CAPMA0> |
| Timer B | [T32AxCAPCRB]<CAPMB0> |
| Timer C | [T32AxCAPCRC]<CAPMC0> |

| Capture timing sets bits of capture register1 | |
|---|-----------------------|
| Timer A | [T32AxCAPCRA]<CAPMA1> |
| Timer B | [T32AxCAPCRB]<CAPMB1> |
| Timer C | [T32AxCAPCRC]<CAPMC1> |

(2) Reading the counter

When the counter capture register in table below is read during the counter operation, current counter value can be captured and read. When the counter is stopping, the last captured value is maintained.

| Counter capture register | |
|--------------------------|-------------|
| Timer A | [T32AxTMRA] |
| Timer B | [T32AxTMRB] |
| Timer C | [T32AxTMRC] |

3.8. Output programmable rectangular wave (PPG) (T32AxOUTA/T32AxOUTB/T32AxOUTC)

It can output rectangular wave of any frequency and any duty ratio. It can set low active or high active as output pulse. It can output programmable rectangular wave at timer output terminal A/B/C when the counter and timer register match or when the counter is captured in capture register.

When a PPG output is used, specify the corresponding port setting in advance.

A "Low" signal is output at the initial state.

If multiple factors for setting, clearing, and reversing occur at the same time, a PPG output is not changed.

(1) Initial output control

A PPG output can be set, cleared, or reversed with output control bit of Output control register0 in table below. Writing data to this bit is always enabled. Even if the timer is stopping, the data is written to this bit.

| Control of T32AxOUTA/ T32AxOUTB/ T32AxOUTC | |
|--|-----------------------|
| Timer A | [T32AxOUTCRA0] <OCRA> |
| Timer B | [T32AxOUTCRB0] <OCRB> |
| Timer C | [T32AxOUTCRC0] <OCRC> |

(2) PPG output when compare match

A PPG output can be set, cleared, or reversed when the counter matches with timer register.

It is selectable with compare control bit on output control register1 in table below.

| Control of T32AxOUTA / T32AxOUTB / T32AxOUTC by comparator | | | |
|--|----------------|--------------------------------|--------------------------|
| Timer | Output control | Output control factor | Control register/Symbol |
| Timer A | T32AxOUTA | Counter match with [T32AxRGA0] | [T32AxOUTCRA1]<OCRCMPA0> |
| | | Counter match with [T32AxRGA1] | [T32AxOUTCRA1]<OCRCMPA1> |
| Timer B | T32AxOUTB | Counter match with [T32AxRGB0] | [T32AxOUTCRB1]<OCRCMPB0> |
| | | Counter match with [T32AxRGB1] | [T32AxOUTCRB1]<OCRCMPB1> |
| Timer C | T32AxOUTC | Counter match with [T32AxRGC0] | [T32AxOUTCRC1]<OCRCMPC0> |
| | | Counter match with [T32AxRGC1] | [T32AxOUTCRC1]<OCRCMPC1> |

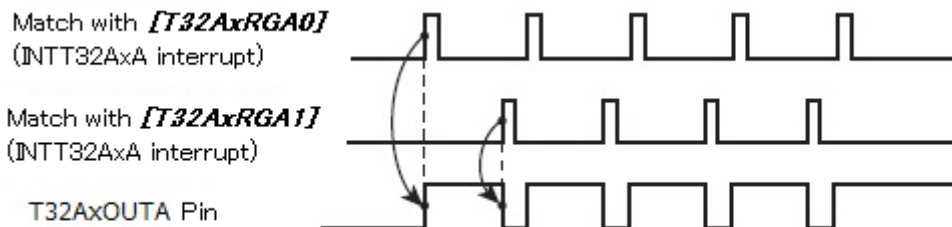


Figure 3.5 Example of Output of programmable rectangular wave using 16-bit Timer A

(3) Capturing

A PPG output can be set, cleared, or reversed with when counter is captured in $[T32AxCAPA0]/[T32AxCAPA1]$, $[T32AxCAPB0]/[T32AxCAPB1]$ and $[T32AxCAPC0]/[T32AxCAPC1]$. It is selectable with capture control bit on output control register1 in table below.

| Control of T32AxOUTA / T32AxOUTB / T32AxOUTC by capture register | | | |
|--|----------------|---------------------------|----------------------------|
| Timer | Output control | Output control factor | Control register/Symbol |
| Timer A | T32AxOUTA | Capture to $[T32AxCAPA0]$ | $[T32AxOUTCRA1]<OCRCAPA0>$ |
| | | Capture to $[T32AxCAPA1]$ | $[T32AxOUTCRA1]<OCRCAPA1>$ |
| Timer B | T32AxOUTB | Capture to $[T32AxCAPB0]$ | $[T32AxOUTCRB1]<OCRCAPB0>$ |
| | | Capture to $[T32AxCAPB1]$ | $[T32AxOUTCRB1]<OCRCAPB1>$ |
| Timer C | T32AxOUTC | Capture to $[T32AxCAPC0]$ | $[T32AxOUTCRC1]<OCRCAPC0>$ |
| | | Capture to $[T32AxCAPC1]$ | $[T32AxOUTCRC1]<OCRCAPC1>$ |

3.9. Interrupts

The following three types of interrupt requests are output.

- Timer interrupt
INTT32AxA, INTT32AxB, INTT32AxC
- Capture interrupt
INTT32AxACAP0/1, INTT32AxBCAP0/1, INTT32AxCCAP0/1
- Interrupt on every count
INTT32AxEVRYC

3.9.1. Timer Interrupt

Timer interrupts are INTT32AxA, INTT32AxB, and INTT32AxC. Timer interrupt is generated by some factors in table below.

Table 3.4 List of Timer Interrupt Sources

| Timer | Interrupt name | Interrupt Source |
|---------|----------------|--|
| Timer A | INTT32AxA | Counter match with $[T32AxRGA0]$, $[T32AxRGA1]$ |
| | | Counter overflow |
| | | Counter underflow |
| Timer B | INTT32AxB | Counter match with $[T32AxRGB0]$, $[T32AxRGB1]$ |
| | | Counter overflow |
| | | Counter underflow |
| Timer C | INTT32AxC | Counter match with $[T32AxRGC0]$, $[T32AxRGC1]$ |
| | | Counter overflow |
| | | Counter underflow |

When Timer C is used in 2-phase pulse count if the counter does not count up or count down, a state transition error (INTT32AxC) is output.

Each factor can be masked with interrupt mask register in table below.

| Interrupt Mask Register | |
|-------------------------|------------|
| Timer A | [T32AxIMA] |
| Timer B | [T32AxIMB] |
| Timer C | [T32AxIMC] |

Even if the mask register is enabled, the occurrence of the factor is set in status register in table below. To clear the state, write "1" to the corresponding bit.

| Status register | |
|-----------------|------------|
| Timer A | [T32AxSTA] |
| Timer B | [T32AxSTB] |
| Timer C | [T32AxSTC] |

3.9.2. Capture Interrupt

Capture interrupts are output when the counter value is captured to the capture register. Capture interrupt and capture register are table in below.

Table 3.5 Relationship table between of Capture interrupt and Capture register

| Relationship between of Capture interrupt and Capture register | | |
|--|----------------|------------------|
| Timer | Interrupt name | Capture register |
| Timer A | INTT32AxACAP0 | [T32AxCAPA0] |
| | INTT32AxACAP1 | [T32AxCAPA1] |
| Timer B | INTT32AxBCAP0 | [T32AxCAPB0] |
| | INTT32AxBCAP1 | [T32AxCAPB1] |
| Timer C | INTT32AxCCAP0 | [T32AxCAPC0] |
| | INTT32AxCCAP1 | [T32AxCAPC1] |

3.9.3. Interrupt on Every Count

An interrupt on every count (INTT32AxEVRYC) only occur in pulse count mode (*[T32AxCRC]*<UPDNC>=11) with 32-bit mode. This interrupt is output on every up counting or down counting.

3.10. DMA Request

A DMA request is issued to the DMA controller when the counter matches the timer register1 or the capture. When a DMA transfer is performed, enable the corresponding bit of the *[T32AxDMAA]*/*[T32AxDMAB]*/*[T32AxDMAC]* register.

| DMA Request Enable Register | |
|-----------------------------|--------------------|
| Timer A | <i>[T32AxDMAA]</i> |
| Timer B | <i>[T32AxDMAB]</i> |
| Timer C | <i>[T32AxDMAC]</i> |

Table 3.6 DMA request list

| Timer | DMA request | DMA request factor | Register |
|---------|------------------|---|------------------------------|
| Timer A | T32AxDMAREQCPA1 | Detect match between timer register A1(<i>[T32AxRGA1]</i>) and counter | <i>[T32AxDMAA]</i> <DMAENA2> |
| | T32AxDMAREQCAPA0 | Capture to capture register A0 (<i>[T32AxCAPA0]</i>) | <i>[T32AxDMAA]</i> <DMAENA0> |
| | T32AxDMAREQCAPA1 | Capture to capture register A1 (<i>[T32AxCAPA1]</i>) | <i>[T32AxDMAA]</i> <DMAENA1> |
| Timer B | T32AxDMAREQCMPB1 | Detect match between timer register B1(<i>[T32AxRGB1]</i>) and counter | <i>[T32AxDMAB]</i> <DMAENB2> |
| | T32AxDMAREQCAPB0 | Capture to capture register B0 (<i>[T32AxCAPB0]</i>) | <i>[T32AxDMAB]</i> <DMAENB0> |
| | T32AxDMAREQCAPB1 | Capture to capture register B1 (<i>[T32AxCAPB1]</i>) | <i>[T32AxDMAB]</i> <DMAENB1> |
| Timer C | T32AxDMAREQCMPC1 | Detect match between timer register C1 (<i>[T32AxRGC1]</i>) and counter | <i>[T32AxDMAC]</i> <DMAENC2> |
| | T32AxDMAREQCAPC0 | Capture to capture register C0 (<i>[T32AxCAPC0]</i>) | <i>[T32AxDMAC]</i> <DMAENC0> |
| | T32AxDMAREQCAPC1 | Capture to capture register C1 (<i>[T32AxCAPC1]</i>) | <i>[T32AxDMAC]</i> <DMAENC1> |

4. Registers

4.1. List of Registers

This section describes the control registers and their addresses.

There is common register, Timer A registers, Timer B registers, and Timer C registers. Common registers are used in regardless of the mode. Timer A registers and Timer B registers are used in 16-bit timer. Timer C registers are used in 32-bit timer.

(1) Base Address

| Function | | Channel/Unit | Base Address | | |
|----------------------------|------|--------------|--------------|------------|------------|
| | | | TYPE 1 | TYPE 2 | TYPE3 |
| 32-bit Timer Event counter | T32A | ch0 | 0x400BA000 | 0x400C1000 | 0x40061000 |
| | | ch1 | 0x400BA100 | 0x400C1400 | 0x40061400 |
| | | ch2 | 0x400BA200 | 0x400C1800 | 0x40061800 |
| | | ch3 | 0x400BA300 | 0x400C1C00 | 0x40061C00 |
| | | ch4 | 0x400BA400 | 0x400C2000 | 0x40062000 |
| | | ch5 | 0x400BA500 | 0x400C2400 | 0x40062400 |
| | | ch6 | 0x400BA600 | 0x400C2800 | 0x40062800 |
| | | ch7 | 0x400BA700 | 0x400C2C00 | 0x40062C00 |
| | | ch8 | 0x400BA800 | 0x400C3000 | 0x40063000 |
| | | ch9 | 0x400BA900 | 0x400C3400 | 0x40063400 |
| | | ch10 | 0x400BAA00 | 0x400C3800 | 0x40063800 |
| | | ch11 | 0x400BAB00 | 0x400C3C00 | 0x40063C00 |
| | | ch12 | 0x400BAC00 | 0x400C4000 | 0x40064000 |
| | | ch13 | - | 0x400C4400 | 0x40064400 |
| | | ch14 | - | 0x400C4800 | - |
| ch15 | - | 0x400C4C00 | - | | |

Note: The channel/unit and base address type are different by products. Please refer to "Product Information" of the reference manual for the details.

(2) Common register

| Register Name | Address(Base+) |
|---------------|-------------------|
| Mode register | [T32AxMOD] 0x0000 |

(3) Timer A register (16-bit timer)

| Register Name | | Address(Base+) |
|-------------------------------|----------------|----------------|
| RUN Register A | [T32AxRUNA] | 0x0040 |
| Counter Control Register A | [T32AxCRA] | 0x0044 |
| Capture Control Register A | [T32AxCAPCRA] | 0x0048 |
| Output Control Register A0 | [T32AxOUTCRA0] | 0x004C |
| Output Control Register A1 | [T32AxOUTCRA1] | 0x0050 |
| Status Register A | [T32AxSTA] | 0x0054 |
| Interrupt Mask Register A | [T32AxIMA] | 0x0058 |
| Counter Capture Register A | [T32AxTMRA] | 0x005C |
| Counter Reload Register A | [T32AxRELDA] | 0x0060 |
| Timer Register A0 | [T32AxRGA0] | 0x0064 |
| Timer Register A1 | [T32AxRGA1] | 0x0068 |
| Capture Register A0 | [T32AxCAPA0] | 0x006C |
| Capture Register A1 | [T32AxCAPA1] | 0x0070 |
| DMA Request Enable Register A | [T32AxDMAA] | 0x0074 |

(4) Timer B register (16-bit timer)

| Register Name | | Address(Base+) |
|-------------------------------|----------------|----------------|
| RUN Register B | [T32AxRUNB] | 0x0080 |
| Counter Control Register B | [T32AxCRB] | 0x0084 |
| Capture Control Register B | [T32AxCAPCRB] | 0x0088 |
| Output Control Register B0 | [T32AxOUTCRB0] | 0x008C |
| Output Control Register B1 | [T32AxOUTCRB1] | 0x0090 |
| Status Register B | [T32AxSTB] | 0x0094 |
| Interrupt Mask Register B | [T32AxIMB] | 0x0098 |
| Counter Capture Register B | [T32AxTMRB] | 0x009C |
| Counter Reload Register B | [T32AxRELDB] | 0x00A0 |
| Timer Register B0 | [T32AxRGB0] | 0x00A4 |
| Timer Register B1 | [T32AxRGB1] | 0x00A8 |
| Capture Register B0 | [T32AxCAPB0] | 0x00AC |
| Capture Register B1 | [T32AxCAPB1] | 0x00B0 |
| DMA Request Enable Register B | [T32AxDMAB] | 0x00B4 |

(5) Timer C register (32-bit timer)

| Register Name | | Address(Base+) |
|-------------------------------|-----------------------|----------------|
| RUN Register C | <i>[T32AxRUNC]</i> | 0x00C0 |
| Counter Control Register C | <i>[T32AxCRC]</i> | 0x00C4 |
| Capture Control Register C | <i>[T32AxCAPCRC]</i> | 0x00C8 |
| Output Control Register C0 | <i>[T32AxOUTCRC0]</i> | 0x00CC |
| Output Control Register C1 | <i>[T32AxOUTCRC1]</i> | 0x00D0 |
| Status Register C | <i>[T32AxSTC]</i> | 0x00D4 |
| Interrupt Mask Register C | <i>[T32AxIMC]</i> | 0x00D8 |
| Counter Capture Register C | <i>[T32AxTMRC]</i> | 0x00DC |
| Counter Reload Register C | <i>[T32AxRELDC]</i> | 0x00E0 |
| Timer Register C0 | <i>[T32AxRGC0]</i> | 0x00E4 |
| Timer Register C1 | <i>[T32AxRGC1]</i> | 0x00E8 |
| Capture Register C0 | <i>[T32AxCAPC0]</i> | 0x00EC |
| Capture Register C1 | <i>[T32AxCAPC1]</i> | 0x00F0 |
| DMA Request Enable Register C | <i>[T32AxDMAC]</i> | 0x00F4 |
| Pulse Count Control Register | <i>[T32AxPLSCR]</i> | 0x00F8 |

4.2. Details of the Common Register

4.2.1. $[T32AxMOD]$ (Mode Register)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:2 | - | 0 | R | Read as "0" |
| 1 | HALT | 0 | R/W | <p>Sets the operation in HALT mode when debugging. 0: Operation 1: Stop</p> <p>Sets the counter operation when the T32A enters HALT mode when debugging.</p> |
| 0 | MODE32 | 0 | R/W | <p>Sets the operation mode. 0: 16-bit mode 1: 32-bit mode</p> <p>When "0" is set to this bit, T32A operate as 16-bit mode that is enable timer A/ timer B and disable Timer C. When "1" is set to this bit, T32A operate as 32-bit mode that is enable timer C and disable Timer A/Timer B. When the operation mode is changed, all registers of Timer A, timer B, and timer C are initialized except $[T32AxMOD]$. When attempting to change the operation mode, check whether the timer is stopping using $[T32AxRUNA]$ <RUNFLGA> and $[T32AxRUNB]$ <RUNFLGB> (16-bit mode), or $[T32AxRUNC]$ <RUNFLGC> (32-bit mode).</p> |

Note: When $[T32AxRUNA]$ <RUNA>=0, $[T32AxRUNB]$ <RUNB>=0 and $[T32AxRUNC]$ <RUNC>=0, set the $[T32AxMOD]$ register.

4.3. Details of Timer A Register

4.3.1. [T32AxRUNA](RUN Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:5 | - | 0 | R | Read as "0" |
| 4 | RUNFLGA | 0 | R | Indicates the operation flag of Timer A. 0: Stop 1: Operation |
| 3 | - | 0 | R | Read as "0" |
| 2 | SFTSTPA | 0 | W | Stops the operation by program. 1: Stops the counter operation. When "1" is set to this bit during the counter operation, the counter stops. Writing "0" has no meaning. Read as "0". |
| 1 | SFTSTAA | 0 | W | Starts the operation by program. 1: Starts the counter operation. When <RUNA> is "1", if "1" is set to <SFTSTAA>, the counter starts operation. Writing "0" has no meaning. Read as "0". |
| 0 | RUNA | 0 | R/W | Controls the operation of Timer A. 0: Disabled 1: Enabled When "1" is set to <RUNA>, Timer A enters the state where it waits for a startup factor. If a startup factor occurs, counting starts. Before "1" is set to <RUNA>, check whether <RUNFLGA> is "0". |

4.3.2. [T32AxCRA] (Counter Control Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|---|
| 31 | - | 0 | R | Read as "0" |
| 30:28 | PRSCLA[2:0] | 000 | R/W | Selects the prescaler division ratio. 000: 1/1 001: 1/2 010: 1/8 011: 1/32 100: 1/128 101: 1/256 110: 1/512 111: 1/1024 |
| 27 | - | 0 | R | Read as "0" |
| 26:24 | CLKA[2:0] | 000 | R/W | Selects the count clock. 000: Prescaler output 001: Internal trigger (Note1) 010: On the rising edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 011: On the falling edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 100: On the rising edge of the external trigger (T32AxINA0). 101: On the falling edge of the external trigger (T32AxINA0). 110-111: Reserved |
| 23:21 | - | 0 | R | Read as "0" |
| 20 | WBFA | 0 | R/W | Controls double-buffering. 0: Disabled 1: Enabled |
| 19:18 | - | 0 | R | Read as "0" |
| 17:16 | UPDNA[1:0] | 00 | R/W | Selects the counter operation. 00: Up counting 01: Down counting 10: Up/down counting 11: Reserved In case of Up/down counting, the counter starts counting up after the operation starts. Then, when the counter value matches Timer Register A1, the counter changes its operation to the down count operation. Subsequently, when the counter value becomes "0x0000", the counter changes its operation to the up count operation. This operation is repeated. |
| 15:11 | - | 0 | R | Read as "0" |
| 10:8 | RELDA[2:0] | 000 | R/W | Counter reload condition 000: None (free running) 001: Internal trigger (Note1) 010: On the rising edge of the external trigger (T32AxINA0). 011: On the falling edge of the external trigger (T32AxINA0). 100: On the rising edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 101: On the falling edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 110: Synchronous operation (slave channel) 111: A match between the counter and Timer Register A1 |
| 7 | - | 0 | R | Read as "0" |
| 6:4 | STOPA[2:0] | 000 | R/W | Sets the counter stop condition. 000: No trigger is used. 001: Internal trigger (Note1) 010: On the rising edge of the external trigger (T32AxINA0). 011: On the falling edge of the external trigger (T32AxINA0). 100: On the rising edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 101: On the falling edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 110: Synchronous operation (slave channel) 111: A match between the counter and Timer Register A1 |

| | | | | |
|-----|-------------|-----|-----|---|
| 3 | - | 0 | R | Read as "0" |
| 2:0 | STARTA[2:0] | 000 | R/W | Sets the counter start condition. 000: No trigger is used. 001: Internal trigger (Note1) 010: On the rising edge of the external trigger (T32AxINA0). 011: On the falling edge of the external trigger (T32AxINA0). 100: On the rising edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 101: On the falling edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 110: Synchronous operation (slave channel) 111: Reserved |

Note1: For details, refer to "Product Information" of reference manual.

Note2: Do not make any changes of [T32AxCRA] register while the [T32AxRUNA]<RUNA>=1.

4.3.3. [T32AxOUTCRA0] (Output Control Register A0)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:2 | - | 0 | R | Read as "0" |
| 1:0 | OCRA[1:0] | 00 | W | Controls T32AxOUTA. 00: No change. 01: Set ("High") 10: Clear ("Low") 11: Reverse Read as "00". |

4.3.4. [T32AxOUTCRA1] (Output Control Register A1)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|---------------|-------------|------|--|
| 31:8 | - | 0 | R | Read as "0" |
| 7:6 | OCRCAPA1[1:0] | 00 | R/W | Controls T32AxOUTA by capture register A1 Controls T32AxOUTA when the counter value is captured in [T32AxCAPA1]. 00: Invalid 01: Set ("High") 10: Clear ("Low") 11: Reverse |
| 5:4 | OCRCAPA0[1:0] | 00 | R/W | Controls T32AxOUTA by capture register A0 Controls T32AxOUTA when the counter value is captured in [T32AxCAPA0]. 00: Invalid 01: Set ("High") 10: Clear ("Low") 11: Reverse |
| 3:2 | OCRCMPA1[1:0] | 00 | R/W | Controls T32AxOUTA by comparator A1 Controls T32AxOUTA when the counter value matches [T32AxRGA1]. 00: Invalid 01: Set ("High") 10: Clear ("Low") 11: Reverse |
| 1:0 | OCRCMPA0[1:0] | 00 | R/W | Controls T32AxOUTA by comparator A0 Controls T32AxOUTA when the counter value matches [T32AxRGA0]. 00: Invalid 01: Set ("High") 10: Clear ("Low") 11: Reverse |

Note: Do not make any changes of [T32AxOUTCRA1] register while the [T32AxRUNA]<RUNA>=1.

4.3.5. [T32AxRGA0] (Timer Register A0)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|------------|-------------|------|--|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | RGA0[15:0] | 0x0000 | R/W | Sets the value for comparing with the counter. |

Note1: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG".

Note2: When writing to this register, note the following.

- Case of the single buffer:
 - When continuously writing to the same timer register, wait for at least 2 clocks (ΦT_0) between the first and second writes. It is the same for the third time and later.
- Case of the double buffer:
 - Write to the timer register only once in the interrupt service routine (ISR) of the cyclic interrupt. It is prohibited to write to the same timer register more than once within the same cycle (*).
 - (*: The period from the occurrence of a timer interrupt to the occurrence of the next timer interrupt.)

4.3.6. [T32AxRGA1] (Timer Register A1)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|------------|-------------|------|--|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | RGA1[15:0] | 0x0000 | R/W | Sets the value for comparing with the counter. |

Note1: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG".

Note2: When writing to this register, note the following.

- Case of the single buffer:
 - When continuously writing to the same timer register, wait for at least 2 clocks (ΦT_0) between the first and second writes. It is the same for the third time and later.
- Case of the double buffer:
 - Write to the timer register only once in the interrupt service routine (ISR) of the cyclic interrupt. It is prohibited to write to the same timer register more than once within the same cycle (*).
 - (*: The period from the occurrence of a timer interrupt to the occurrence of the next timer interrupt.)

4.3.7. [T32AxTMRA] (Counter Capture Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|------------|-------------|------|---|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | TMRA[15:0] | 0x0000 | R | When this register is read during the counter operation, the current value of counter will be captured. And this counter value can be read. When the counter is stopping, the last captured value can be read. |

4.3.8. [T32AxRELDA] (Counter Reload Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|---|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | RELDA[15:0] | 0x0000 | R/W | Sets the value to be reloaded to the counter. |

Note: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG".

4.3.9. [T32AxCAPCRA] (Capture Control Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|-------------|-------------|------|---|
| 31:7 | - | 0 | R | Read as "0" |
| 6:4 | CAPMA1[2:0] | 000 | R/W | Sets the capture timing of [T32AxCAPA1]. 000: Disabled 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the T32AxINA0 input pin 011: On the falling edge of the T32AxINA0 input pin. 100: On the rising edge of the T32AxINA1 input pin. 101: On the falling edge of the T32AxINA1 input pin. 110: On the rising edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 111: On the falling edge of an output from other timer (T32AxTRGINAPHCK) (Note1) |
| 3 | - | 0 | R | Read as "0" |
| 2:0 | CAPMA0[2:0] | 000 | R/W | Sets the capture timing of [T32AxCAPA0]. 000: Disabled 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the T32AxINA0 input pin 011: On the falling edge of the T32AxINA0 input pin. 100: On the rising edge of the T32AxINA1 input pin. 101: On the falling edge of the T32AxINA1 input pin. 110: On the rising edge of an output from other timer (T32AxTRGINAPHCK) (Note1) 111: On the falling edge of an output from other timer (T32AxTRGINAPHCK) (Note1) |

Note1: For details, refer to "Product Information" of reference manual.

Note2: Do not make any changes of [T32AxCAPCRA] register while the [T32AxRUNA]<RUNA>=1.

4.3.10. [T32AxCAPA0] (Capture Register A0)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|------------------------------------|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | CAPA0[15:0] | 0x0000 | R | Captures the value in the counter. |

4.3.11. [T32AxCAPA1] (Capture Register A1)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|------------------------------------|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | CAPA1[15:0] | 0x0000 | R | Captures the value in the counter. |

4.3.12. [T32AxIMA] (Interrupt Mask Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:4 | - | 0 | R | Read as "0" |
| 3 | IMUFA | 0 | R/W | Controls to mask the underflow interrupt request. 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the underflow interrupt request. |
| 2 | IMOFA | 0 | R/W | Controls to mask the overflow interrupt request. 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the overflow interrupt request. |
| 1 | IMA1 | 0 | R/W | Control to mask the match detection interrupt request ([T32AxRGA1]). 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the match detection request when the counter matches [T32AxRGA1]. |
| 0 | IMA0 | 0 | R/W | Control to mask the match detection interrupt request ([T32AxRGA0]). 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the match detection request when the counter matches [T32AxRGA0]. |

Note1: Even when the interrupt mask register ([T32AxIMA]) is enabled, a status flag of the counter is set to the [T32AxSTA].

Note2: Do not make any changes of [T32AxIMA] register while the [T32AxRUNA]<RUNA>=1.

4.3.13. [T32AxSTA] (Status Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:4 | - | 0 | R | Read as "0" |
| 3 | INTUFA | 0 | R | Indicates an underflow flag. 0: No underflow occurred. 1: An underflow occurred. When the counter underflows, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 2 | INTOFA | 0 | R | Indicates an overflow flag. 0: No overflow occurred. 1: An overflow occurred. When the counter overflows, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 1 | INTA1 | 0 | R | Indicates a match flag. 0: No match is detected. 1: A match between the counter and [T32AxRGA1]. When a match between the counter value and the Timer Register A1 ([T32AxRGA1]) is detected, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 0 | INTA0 | 0 | R | Indicates a match flag. 0: No match is detected. 1: A match between the counter and [T32AxRGA0]. When a match between the counter value and the Timer Register A0 ([T32AxRGA0]) is detected, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |

Note1: Even when the interrupt mask register ([T32AxIMA]) is enabled, a status flag is set to the [T32AxSTA] register.

Note2: Only the interrupt request that is not masked with [T32AxIMA] is output to the CPU.

Note3: When setting flag by internal signal and clearing flag by program occur at the same time, clearing flag has priority.

4.3.14. [T32AxDMAA] (DMA Request Enable Register A)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:3 | - | 0 | R | Read as "0" |
| 2 | DMAENA2 | 0 | R/W | Selects a DMA request occurrence condition: A match between the counter and the Timer Register A1 ([T32AxRGA1]) 0: Disabled 1: Enabled |
| 1 | DMAENA1 | 0 | R/W | Selects a DMA request occurrence condition: Input capture 1 ([T32AxCAPA1]) 0: Disabled 1: Enabled |
| 0 | DMAENA0 | 0 | R/W | Selects a DMA request occurrence condition: Input capture 0 ([T32AxCAPA0]) 0: Disabled 1: Enabled |

Note: Do not make any changes of [T32AxDMAA] register while the [T32AxRUNA]<RUNA>=1.

4.4. Details of Timer B Register

4.4.1. [T32AxRUNB] (RUN Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:5 | - | 0 | R | Read as "0" |
| 4 | RUNFLGB | 0 | R | Indicates an operation flag of Timer B. 0: Stop 1: Operation |
| 3 | - | 0 | R | Read as "0" |
| 2 | SFTSTPB | 0 | W | Stops the counter operation by program. 1: Stops the counter operation. When "1" is set to this bit during the counter operation, the counter operation stops. Writing "0" has no meaning. Read as "0". |
| 1 | SFTSTAB | 0 | W | Starts the counter operation by program. 1: Starts the counter operation. When <RUNB> is "1", if "1" is set to <SFTSTAB>, the counter starts operation. Writing "0" has no meaning. Read as "0". |
| 0 | RUNB | 0 | R/W | Controls the operation of Timer B. 0: Disabled 1: Enabled When "1" is set to <RUNB>, Timer B enters the state where it waits for a startup factor. If a startup factor occurs, counting starts. Before "1" is set to <RUNB>, check whether <RUNFLGB> is "0". |

4.4.2. [T32AxCRB] (Counter Control Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|--|
| 31 | - | 0 | R | Read as "0" |
| 30:28 | PRSCLB[2:0] | 000 | R/W | Selects the prescaler division ratio. 000: 1/1 001: 1/2 010: 1/8 011: 1/32 100: 1/128 101: 1/256 110: 1/512 111: 1/1024 |
| 27 | - | 0 | R | Read as "0" |
| 26:24 | CLKB[2:0] | 000 | R/W | Selects the count clock. 000: Prescaler output 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 011: On the falling edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 100: On the rising edge of the external trigger (T32AxINB0). 101: On the falling edge of the external trigger (T32AxINB0). 110-111: Reserved |
| 23:21 | - | 0 | R | Read as "0" |
| 20 | WBFB | 0 | R/W | Controls double-buffering. 0: Disabled 1: Enabled |
| 19:18 | - | 0 | R | Read as "0" |
| 17:16 | UPDNB[1:0] | 00 | R/W | Selects the counter operation. 00: Up counting 01: Down counting 10: Up/down counting 11: Reserved In case of Up/down counting, the counter starts counting up after the operation starts. Then, when the counter value matches Timer Register B1, the counter changes its operation to the down count operation. Subsequently, when the counter value becomes "0x0000", the counter changes its operation to the up count operation. This operation is repeated. |
| 15:11 | - | 0 | R | Read as "0" |
| 10:8 | RELDDB[2:0] | 000 | R/W | Counter reload condition 000: None (free running) 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the external trigger (T32AxINB0). 011: On the falling edge of the external trigger (T32AxINB0). 100: On the rising edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 101: On the falling edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 110: Synchronous operation (slave channel) 111: A match between the counter and Timer Register B1 |
| 7 | - | 0 | R | Read as "0" |
| 6:4 | STOPB[2:0] | 000 | R/W | Sets the counter stop condition. 000: No trigger is used. 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the external trigger (T32AxINB0). 011: On the falling edge of the external trigger (T32AxINB0). 100: On the rising edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 101: On the falling edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 110: Synchronous operation (slave channel) 111: A match between the counter and Timer Register B1 |
| 3 | - | 0 | R | Read as "0" |

| | | | | |
|-----|-------------|-----|-----|---|
| 2:0 | STARTB[2:0] | 000 | R/W | <p>Sets the counter start condition.</p> <p>000: No trigger is used.</p> <p>001: Internal trigger (T32AxTRGINAPCK) (Note1)</p> <p>010: On the rising edge of the external trigger (T32AxINB0).</p> <p>011: On the falling edge of the external trigger (T32AxINB0).</p> <p>100: On the rising edge of an output from other timer (T32AxTRGINBPHCK) (Note1)</p> <p>101: On the falling edge of an output from other timer (T32AxTRGINBPHCK) (Note1)</p> <p>110: Synchronous operation (slave channel)</p> <p>111: Reserved</p> |
|-----|-------------|-----|-----|---|

Note1: For details, refer to "Product Information" of reference manual.

Note2: Do not make any changes of *[T32AxCRB]* register while the *[T32AxRUNB]<RUNB>=1*.

4.4.3. *[T32AxOUTCRB0]* (Output Control Register B0)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:2 | - | 0 | R | Read as "0" |
| 1:0 | OCRB[1:0] | 00 | W | <p>Controls T32AxOUTB.</p> <p>00: No change.</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> <p>Read as "00".</p> |

4.4.4. *[T32AxOUTCRB1]* (Output Control Register B1)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|---------------|-------------|------|--|
| 31:8 | - | 0 | R | Read as "0" |
| 7:6 | OCRCAPB1[1:0] | 00 | R/W | <p>Controls T32AxOUTB by capture register B1</p> <p>Controls T32AxOUTB when the counter value is captured in <i>[T32AxCAPB1]</i>.</p> <p>00: Invalid</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> |
| 5:4 | OCRCAPB0[1:0] | 00 | R/W | <p>Controls T32AxOUTB by capture register B0</p> <p>Controls T32AxOUTB when the counter value is captured in <i>[T32AxCAPB0]</i>.</p> <p>00: Invalid</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> |
| 3:2 | OCRCMPB1[1:0] | 00 | R/W | <p>Controls T32AxOUTB by comparator B1</p> <p>Controls T32AxOUTB when the counter value matches <i>[T32AxRGB1]</i>.</p> <p>00: Invalid</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> |
| 1:0 | OCRCMPB0[1:0] | 00 | R/W | <p>Controls T32AxOUTB by comparator B0</p> <p>Controls T32AxOUTB when the counter value matches <i>[T32AxRGB0]</i>.</p> <p>00: Invalid</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> |

Note: Do not make any changes of *[T32AxOUTCRB1]* register while the *[T32AxRUNB]<RUNB>=1*.

4.4.5. [T32AxRGB0] (Timer Register B0)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|------------|-------------|------|--|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | RGB0[15:0] | 0x0000 | R/W | Sets the value for comparing with the counter. |

Note1: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG".

Note2: When writing to this register, note the following.

- Case of the single buffer:
 - When continuously writing to the same timer register, wait for at least 2 clocks (ΦT_0) between the first and second writes. It is the same for the third time and later.
- Case of the double buffer:
 - Write to the timer register only once in the interrupt service routine (ISR) of the cyclic interrupt. It is prohibited to write to the same timer register more than once within the same cycle (*).
 - (*: The period from the occurrence of a timer interrupt to the occurrence of the next timer interrupt.)

4.4.6. [T32AxRGB1] (Timer Register B1)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|------------|-------------|------|--|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | RGB1[15:0] | 0x0000 | R/W | Sets the value for comparing with the counter. |

Note1: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG".

Note2: When writing to this register, note the following.

- Case of the single buffer:
 - When continuously writing to the same timer register, wait for at least 2 clocks (ΦT_0) between the first and second writes. It is the same for the third time and later.
- Case of the double buffer:
 - Write to the timer register only once in the interrupt service routine (ISR) of the cyclic interrupt. It is prohibited to write to the same timer register more than once within the same cycle (*).
 - (*: The period from the occurrence of a timer interrupt to the occurrence of the next timer interrupt.)

4.4.7. [T32AxTMRB] (Counter Capture Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|------------|-------------|------|---|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | TMRB[15:0] | 0x0000 | R | When this register is read during the counter operation, the current value of counter will be captured. And this counter value can be read. When the counter is stopping, the last captured value can be read. |

4.4.8. [T32AxRELD B] (Counter Reload Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|--------------|-------------|------|---|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | RELD B[15:0] | 0x0000 | R/W | Sets the value to be reloaded to the counter. |

Note: About value of counter when output PPG, refer to " Table 3.1 Note of timer register setting when output PPG".

4.4.9. [T32AxCAPCR B] (Capture Control Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|-------------|-------------|------|---|
| 31:7 | - | 0 | R | Read as "0" |
| 6:4 | CAPMB1[2:0] | 000 | R/W | Sets the capture timing of [T32AxCAPB1]. 000: Disabled 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the T32AxINB0 input pin 011: On the falling edge of the T32AxINB0 input pin. 100: On the rising edge of the T32AxINB1 input pin. 101: On the falling edge of the T32AxINB1 input pin. 110: On the rising edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 111: On the falling edge of an output from other timer (T32AxTRGINBPHCK) (Note1) |
| 3 | - | 0 | R | Read as "0" |
| 2:0 | CAPMB0[2:0] | 000 | R/W | Sets the capture timing of [T32AxCAPB0]. 000: Disabled 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the T32AxINB0 input pin 011: On the falling edge of the T32AxINB0 input pin. 100: On the rising edge of the T32AxINB1 input pin. 101: On the falling edge of the T32AxINB1 input pin. 110: On the rising edge of an output from other timer (T32AxTRGINBPHCK) (Note1) 111: On the falling edge of an output from other timer (T32AxTRGINBPHCK) (Note1) |

Note1: For details, refer to "Product Information" of reference manual.

Note2: Do not make any changes of [T32AxCAPCR B] register while the [T32AxRUNB]<RUNB>=1.

4.4.10. [T32AxCAPB0] (Capture Register B0)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|------------------------------------|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | CAPB0[15:0] | 0x0000 | R | Captures the value in the counter. |

4.4.11. [T32AxCAPB1] (Capture Register B1)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|------------------------------------|
| 31:16 | - | 0 | R | Read as "0" |
| 15:0 | CAPB1[15:0] | 0x0000 | R | Captures the value in the counter. |

4.4.12. [T32AxIMB] (Interrupt Mask Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:4 | - | 0 | R | Read as "0" |
| 3 | IMUFB | 0 | R/W | Controls to mask the underflow interrupt request. 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the underflow interrupt request. |
| 2 | IMOFB | 0 | R/W | Controls to mask the overflow interrupt request. 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the overflow interrupt request. |
| 1 | IMB1 | 0 | R/W | Control to mask the match detection interrupt request ([T32AxRGB1]). 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the match detection request when the counter matches [T32AxRGB1]. |
| 0 | IMB0 | 0 | R/W | Control to mask the match detection interrupt request ([T32AxRGB0]). 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the match detection request when the counter matches [T32AxRGB0]. |

Note1: Even when the interrupt mask register ([T32AxIMB]) is enabled, a status flag of the counter is set to the [T32AxSTB].

Note2: Do not make any changes of [T32AxIMB] register while the [T32AxRUNB]<RUNB>=1.

4.4.13. [T32AxSTB] (Status Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:4 | - | 0 | R | Read as "0" |
| 3 | INTUFB | 0 | R | Indicates an underflow flag. 0: No underflow occurred. 1: An underflow occurred. When the counter underflows, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 2 | INTOFB | 0 | R | Indicates an overflow flag. 0: No overflow occurred. 1: An overflow occurred. When the counter overflows, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 1 | INTB1 | 0 | R | Indicates a match flag. 0: No match is detected. 1: A match between the counter and [T32AxRGB1] is detected. When a match between the counter value and the Timer Register B1 ([T32AxRGB1]) is detected, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 0 | INTB0 | 0 | R | Indicates a match flag. 0: No match is detected. 1: A match between the counter and [T32AxRGB0] is detected. When a match between the counter value and the Timer Register B0 ([T32AxRGB0]) is detected, "1" is set to this bit.. |
| | | | W | 0: Don't care. 1: Cleared to "0". |

Note1: Even when the interrupt mask register [T32AxIMB] is enabled, a status flag is set to the [T32AxSTB] register.

Note2: Only the interrupt request that is not masked with [T32AxIMB] is output to the CPU.

Note3: When setting flag by internal signal and clearing flag by program occur at the same time, clearing flag has priority.

4.4.14. [T32AxDMAB] (DMA Request Enable Register B)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:3 | - | 0 | R | Read as "0" |
| 2 | DMAENB2 | 0 | R/W | Selects a DMA request occurrence condition: A match between the counter and the Timer Register B1 ([T32AxRGB1]) 0: Disabled 1: Enabled |
| 1 | DMAENB1 | 0 | R/W | Selects a DMA request occurrence condition: Input capture 1 ([T32AxCAPB1]) 0: Disabled 1: Enabled |
| 0 | DMAENB0 | 0 | R/W | Selects a DMA request occurrence condition: Input capture 0 ([T32AxCAPB0]) 0: Disabled 1: Enabled |

Note: Do not make any changes of [T32AxDMAB] register while the [T32AxRUNB]<RUNB>=1.

4.5. Details of Timer C Register

4.5.1. [T32AxRUNC] (RUN Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:5 | - | 0 | R | Read as "0" |
| 4 | RUNFLGC | 0 | R | Indicates an operation flag of Timer C. 0: Stop 1: Operation |
| 3 | - | 0 | R | Read as "0" |
| 2 | SFTSTPC | 0 | W | Stops the operation by program. 1: Stops the counter operation. When "1" is set to this bit during the counter operation, the counter stops. Writing "0" has no meaning. Read as "0". |
| 1 | SFTSTAC | 0 | W | Starts the operation by program. 1: Starts the counter operation. When <RUNC> is "1", if "1" is set to <SFTSTAC>, the counter starts operation. Writing "0" has no meaning. Read as "0". |
| 0 | RUNC | 0 | R/W | Controls the operation of Timer C. 0: Disabled 1: Enabled When "1" is set to <RUNC>, Timer A enters the state where it waits for a startup factor. If a startup factor occurs, counting starts. Before "1" is set to <RUNC>, check whether <RUNFLGC> is "0". |

4.5.2. [T32AxCRC] (Counter Control Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|-------------|-------------|------|---|
| 31 | - | 0 | R | Read as "0" |
| 30:28 | PRSCLC[2:0] | 000 | R/W | Selects the prescaler division ratio. 000: 1/1 001: 1/2 010: 1/8 011: 1/32 100: 1/128 101: 1/256 110: 1/512 111: 1/1024 |
| 27 | - | 0 | R | Read as "0" |
| 26:24 | CLKC[2:0] | 000 | R/W | Selects the count clock. 000: Prescaler output 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of an output from other timer (T32AxTRGINCPHCK) (Note1) 011: On the falling edge of an output from other timer (T32AxTRGINCPHCK) (Note1) 100: On the rising edge of the external trigger (T32AxINC0). 101: On the falling edge of the external trigger (T32AxINC0). 110-111: Reserved |
| 23:21 | - | 0 | R | Read as "0" |
| 20 | WBFC | 0 | R/W | Controls double-buffering. 0: Disabled 1: Enabled |
| 19:18 | - | 0 | R | Read as "0" |
| 17:16 | UPDNC[1:0] | 00 | R/W | Selects the counter operation. 00: Up counting 01: Down counting 10: Up/down counting 11: Pulse counting For details, refer to the pulse count control register ([T32AxPLSCR]). In case of Up/down counting, the counter starts counting up after the operation starts. Then, when the counter value matches Timer Register C1, the counter changes its operation to the down count operation. Subsequently, when the counter value becomes "0x0000", the counter changes its operation to the up count operation. This operation is repeated. |
| 15:11 | - | 0 | R | Read as "0" |
| 10:8 | RELDC[2:0] | 000 | R/W | Counter reload condition 000: None (free running) 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the external trigger (T32AxINC0). 011: On the falling edge of the external trigger (T32AxINC0). 100: On the rising edge of an output from other timer (T32AxTRGINCPHCK) (Note1) 101: On the falling edge of an output from other timer (T32AxTRGINCPHCK) (Note1) 110: Synchronous operation (slave channel) 111: A match between the counter and Timer Register C1 |
| 7 | - | 0 | R | Read as "0" |

| | | | | |
|-----|-------------|-----|-----|---|
| 6:4 | STOPC[2:0] | 000 | R/W | <p>Sets the counter stop condition.</p> <p>000: No trigger is used.</p> <p>001: Internal trigger (T32AxTRGINAPCK) (Note1)</p> <p>010: On the rising edge of the external trigger (T32AxINC0).</p> <p>011: On the falling edge of the external trigger (T32AxINC0).</p> <p>100: On the rising edge of an output from other timer (T32AxTRGINCPHCK) (Note1)</p> <p>101: On the falling edge of an output from other timer (T32AxTRGINCPHCK) (Note1)</p> <p>110: Synchronous operation (slave channel)</p> <p>111: A match between the counter and Timer Register C1</p> |
| 3 | - | 0 | R | Read as "0" |
| 2:0 | STARTC[2:0] | 000 | R/W | <p>Sets the counter start condition.</p> <p>000: No trigger is used.</p> <p>001: Internal trigger (T32AxTRGINAPCK) (Note1)</p> <p>010: On the rising edge of the external trigger (T32AxINC0).</p> <p>011: On the falling edge of the external trigger (T32AxINC0).</p> <p>100: On the rising edge of an output from other timer (T32AxTRGINCPHCK) (Note1)</p> <p>101: On the falling edge of an output from other timer (T32AxTRGINCPHCK) (Note1)</p> <p>110: Synchronous operation (slave channel)</p> <p>111: Reserved</p> |

Note1: For details, refer to "Product Information" of reference manual.

Note2: Do not make any changes of *[T32AxCRC]* register while the *[T32AxRUNC]<RUNC>=1*.

4.5.3. *[T32AxOUTCRC0]* (Output Control Register C0)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:2 | - | 0 | R | Read as "0" |
| 1:0 | OCRC[1:0] | 00 | W | <p>Controls T32AxOUTC.</p> <p>00: No change.</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> <p>Read as "00".</p> |

4.5.4. *[T32AxOUTCRC1]* (Output Control Register C1)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|---------------|-------------|------|--|
| 31:8 | - | 0 | R | Read as "0" |
| 7:6 | OCRCAPC1[1:0] | 00 | R/W | <p>Controls T32AxOUTC by capture register C1</p> <p>Controls T32AxOUTC when the counter value is captured in <i>[T32AxCAPC1]</i>.</p> <p>00: Invalid</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> |
| 5:4 | OCRCAPC0[1:0] | 00 | R/W | <p>Controls T32AxOUTC by capture register C0</p> <p>Controls T32AxOUTC when the counter value is captured in <i>[T32AxCAPC0]</i>.</p> <p>00: Invalid</p> <p>01: Set ("High")</p> <p>10: Clear ("Low")</p> <p>11: Reverse</p> |

| | | | | |
|-----|---------------|----|-----|--|
| 3:2 | OCRCMPC1[1:0] | 00 | R/W | Controls T32AxOUTC by comparator C1 Controls T32AxOUTC when the counter value matches [T32AxRGC1] . 00: Invalid 01: Set ("High") 10: Clear ("Low") 11: Reverse |
| 1:0 | OCRCMPC0[1:0] | 00 | R/W | Controls T32AxOUTC by comparator C0 Controls T32AxOUTC when the counter value matches [T32AxRGC0] . 00: Invalid 01: Set ("High") 10: Clear ("Low") 11: Reverse |

Note: Do not make any changes of **[T32AxOUTCRC1]** register while the **[T32AxRUNC]<RUNC>=1**.

4.5.5. [T32AxRGC0] (Timer Register C0)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:0 | RGC0[31:0] | 0x00000000 | R/W | Sets the value for comparing with the counter. |

Note1: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG" about setting of count value.

Note2: When writing to this register, note the following.

- Case of the single buffer:

When continuously writing to the same timer register, wait for at least 2 clocks (ΦT_0) between the first and second writes. It is the same for the third time and later.
- Case of the double buffer:

Write to the timer register only once in the interrupt service routine (ISR) of the cyclic interrupt. It is prohibited to write to the same timer register more than once within the same cycle (*).

(*: The period from the occurrence of a timer interrupt to the occurrence of the next timer interrupt.)

4.5.6. [T32AxRGC1] (Timer Register C1)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:0 | RGC1[31:0] | 0x00000000 | R/W | Sets the value for comparing with the counter. |

Note1: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG".

Note2: When writing to this register, note the following.

- Case of the single buffer:

When continuously writing to the same timer register, wait for at least 2 clocks (ΦT_0) between the first and second writes. It is the same for the third time and later.
- Case of the double buffer:

Write to the timer register only once in the interrupt service routine (ISR) of the cyclic interrupt. It is prohibited to write to the same timer register more than once within the same cycle (*).

(*: The period from the occurrence of a timer interrupt to the occurrence of the next timer interrupt.)

4.5.7. [T32AxTMRC] (Counter Capture Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:0 | TMRC[31:0] | 0x00000000 | R | When this register is read during the counter operation, the current value of counter will be captured. And this counter value can be read. When the counter is stopping, the last captured value can be read. |

4.5.8. [T32AxRELDC] (Counter Reload Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|-------------|-------------|------|---|
| 31:0 | RELDC[31:0] | 0x00000000 | R/W | Sets the value to be reloaded to the counter. |

Note: About value of counter when output PPG, refer to "Table 3.1 Note of timer register setting when output PPG".

4.5.9. [T32AxCAPCRC] (Capture Control Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|-------------|-------------|------|---|
| 31:7 | - | 0 | R | Read as "0" |
| 6:4 | CAPMC1[2:0] | 000 | R/W | Sets the capture timing of [T32AxCAPC1]. 000: Disabled 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the T32AxINC0 input pin 011: On the falling edge of the T32AxINC0 input pin. 100: On the rising edge of the T32AxINC1 input pin. 101: On the falling edge of the T32AxINC1 input pin. 110: On the rising edge of an output from other timer (T32AxTRGINCPHCK) (Note1) 111: On the falling edge of an output from other timer (T32AxTRGINCPHCK) (Note1) |
| 3 | - | 0 | R | Read as "0" |
| 2:0 | CAPMC0[2:0] | 000 | R/W | Sets the capture timing of [T32AxCAPC0]. 000: Disabled 001: Internal trigger (T32AxTRGINAPCK) (Note1) 010: On the rising edge of the T32AxINC0 input pin 011: On the falling edge of the T32AxINC0 input pin. 100: On the rising edge of the T32AxINC1 input pin. 101: On the falling edge of the T32AxINC1 input pin. 110: On the rising edge of an output from other timer (T32AxTRGINCPHCK) (Note1) 111: On the falling edge of an output from other timer (T32AxTRGINCPHCK) (Note1) |

Note1: For details, refer to "Product Information" of reference manual.

Note2: Do not make any changes of [T32AxCAPCRC] register while the [T32AxRUNC]<RUNC>=1.

4.5.10. [T32AxCAPC0] (Capture Register C0)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|-------------|-------------|------|------------------------------------|
| 31:0 | CAPC0[31:0] | 0x00000000 | R | Captures the value in the counter. |

4.5.11. [T32AxCAPC1] (Capture Register C1)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|-------------|-------------|------|------------------------------------|
| 31:0 | CAPC1[31:0] | 0x00000000 | R | Captures the value in the counter. |

4.5.12. [T32AxIMC] (Interrupt Mask Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:5 | - | 0 | R | Read as "0" |
| 4 | IMSTERR | 0 | R/W | Controls to mask the status transition error interrupt request (only in 2-phase pulse count mode) 0: An interrupt request is not masked. 1: Masks the interrupt request. Control whether mask the status transition error interrupt request in 2-phase pulse count mode. |
| 3 | IMUFC | 0 | R/W | Controls to mask the underflow interrupt request. 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the underflow interrupt request. |
| 2 | IMOFC | 0 | R/W | Controls to mask the overflow interrupt request. 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the overflow interrupt request. |
| 1 | IMC1 | 0 | R/W | Control to mask the match detection interrupt request ([T32AxRGC1]). 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the match detection request when the counter matches [T32AxRGC1]. |
| 0 | IMC0 | 0 | R/W | Control to mask the match detection interrupt request ([T32AxRGC0]). 0: An interrupt request is not masked. 1: Masks the interrupt request. Controls whether mask the match detection request when the counter matches [T32AxRGC0]. |

Note1: Even when the interrupt mask register ([T32AxIMC]) is enabled, a status flag of the counter is set to the [T32AxSTC].

Note2: Do not make any changes of [T32AxIMC] register while the [T32AxRUNC]<RUNC>=1.

4.5.13. [T32AxSTC] (Status Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|---|
| 31:5 | - | 0 | R | Read as "0" |
| 4 | INTSTERR | 0 | R | Indicates the status transition error interrupt request (only in 2-phase pulse count mode) 0: No status transition error occurred. 1: A status transition error occurred. When a status transition error occurs in 2-phase pulse count mode, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 3 | INTUFC | 0 | R | Indicates an underflow flag. 0: No underflow occurred. 1: An underflow occurred. When the counter underflows, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 2 | INTOFC | 0 | R | Indicates an overflow flag. 0: No overflow occurred. 1: An overflow occurred. When the counter overflows, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 1 | INTC1 | 0 | R | Indicates a match flag. 0: No match is detected. 1: A match between the counter and [T32AxRGC1] is detected. When a match between the counter value and the Timer Register C1 ([T32AxRGC1]) is detected, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |
| 0 | INTC0 | 0 | R | Indicates a match flag. 0: No match is detected. 1: A match between the counter and [T32AxRGC0] is detected. When a match between the counter value and the Timer Register C0 ([T32AxRGC0]) is detected, "1" is set to this bit. |
| | | | W | 0: Don't care. 1: Cleared to "0". |

Note1: Even when the interrupt mask register [T32AxIMC] is enabled, a status flag is set to the [T32AxSTC] register.

Note2: Only the interrupt request that is not masked with [T32AxIMC] is output to the CPU.

Note3: When setting flag by internal signal and clearing flag by program occur at the same time, clearing flag has priority.

4.5.14. [T32AxDMAC] (DMA Request Enable Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|------|------------|-------------|------|--|
| 31:3 | - | 0 | R | Read as "0" |
| 2 | DMAENC2 | 0 | R/W | Selects a DMA request occurrence condition: A match between the counter and the Timer Register C1 ([T32AxRGC1]) 0: Disabled 1: Enabled |
| 1 | DMAENC1 | 0 | R/W | Selects a DMA request occurrence condition: Input capture 1 ([T32AxCAPC1]) 0: Disabled 1: Enabled |
| 0 | DMAENC0 | 0 | R/W | Selects a DMA request occurrence condition: Input capture 0 ([T32AxCAPC0]) 0: Disabled 1: Enabled |

Note: Do not make any changes of [T32AxDMAC] register while the [T32AxRUNC]<RUNC>=1.

4.5.15. [T32AxPLSCR] (Pulse Counter control Register C)

| Bit | Bit Symbol | After Reset | Type | Function |
|-------|------------|-------------|------|---|
| 31:15 | - | 0 | R | Read as "0" |
| 14:12 | PDN[2:0] | 000 | R/W | Count down condition in 1-phase pulse counter mode. 000: No count down. 001: No count down. 010: On the rising edge of T32AxINC0. 011: On the falling edge of T32AxINC0. 100: On the rising edge of T32AxINC1. 101: On the falling edge of T32AxINC1. 110: On the rising/falling edge of T32AxINC0. 111: On the rising/falling edge of T32AxINC1. |
| 11 | - | 0 | R | Read as "0" |
| 10:8 | PUP[2:0] | 000 | R/W | Count up condition in 1-phase pulse count mode. 000: No count up. 001: No count up. 010: On the rising edge of T32AxINC0 011: On the falling edge of T32AxINC0 100: On the rising edge of T32AxINC1 101: On the falling edge of T32AxINC1 110: On the rising edge/falling edge of T32AxINC0 111: On the rising edge/falling edge of T32AxINC1 |
| 7:6 | - | 0 | R | Read as "0" |
| 5:4 | NF[1:0] | 00 | R/W | Selects the noise elimination time for T32AxINC0/T32AxINC1 For detail, refer to "Table 4.1 Noise elimination <NF>". 00: None 01: Eliminates a signal less than 2/ Φ T0 as noise. 10: Eliminates a signal less than 4/ Φ T0 as noise. 11: Eliminates a signal less than 8/ Φ T0 as noise. |
| 3:2 | - | 0 | R | Read as "0" |
| 1 | PDIR | 0 | R/W | Selects the direction of the 2-phase pulse counter. 0: Positive direction 1: Negative direction |
| 0 | PMODE | 0 | R/W | Selects the pulse counter mode. 0: 2-phase pulse counter mode 1: 1-phase pulse counter mode |

For details of noise elimination <NF>, see below:

Table 4.1 Noise elimination <NF>

| <NF> | The level at which noise is surely eliminated. | The level at which a signal is surely recognized | The level at which noise is surely eliminated (Time) [CGSYSCR]<PRCKST[3:0]>=0000 fc=40MHz |
|------|--|--|---|
| 01 | Less than $2/\Phi T_0$ | $3/\Phi T_0$ or more | 50ns |
| 10 | Less than $4/\Phi T_0$ | $5/\Phi T_0$ or more | 100ns |
| 11 | Less than $8/\Phi T_0$ | $9/\Phi T_0$ or more | 200ns |

5. Use Case

5.1. Interval Timer

T32A can generate interrupt periodically.

The following is the definition of the period.

| | Period |
|---------------|------------------------------|
| Up counting | $[T32AxRGA1] - [T32AxRELDA]$ |
| Down counting | $[T32AxRELDA] - [T32AxRGA1]$ |

The following case is an example when periodical interrupts are generated by counting up Timer A. Set "0x0000" to the reload register $[T32AxRELDA]$ and set the interval time to the timer register $[T32AxRGA1]$, and set the counter reloaded condition. The reload condition is a compare match between the counter and $[T32AxRGA1]$. After the operation is started, an INTT32Ax interrupt occurs after the interval time has elapsed. The counter is reloaded and continues count up.

Counter Control Register A ($[T32AxCRA]$)

Clock selection <CLKA[2:0]> Any clock
 Count Operation <UPDNA[1:0]> Up counting
 Start operation <STARTA[2:0]> Any timing
 Stop operation <STOPA[2:0]> Any timing
 Reload condition <RELDA[2:0]> A match with $[T32AxRGA1]$

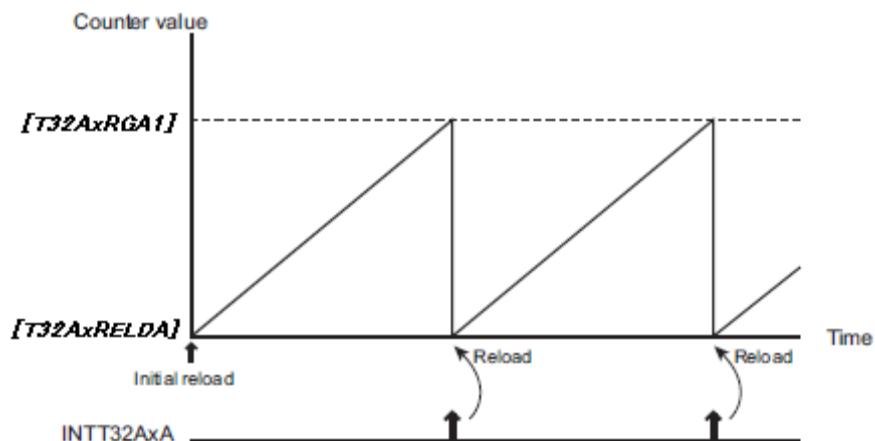


Figure 5.1 Interval Timer (Count up)

The following case is an example when periodical interrupts are generated by counting down Timer A. Set the interval time to the reload register $[T32AxRELDA]$, and set "0x0000" to the timer register $[T32AxRGA1]$, and set the reload condition. The reload condition is a compare match between the counter and $[T32AxRGA1]$. After the operation is started, the counter counts down from the reloaded value to "0x0000". When the counter reaches "0x0000", an INTT32AxA interrupt occurs. The counter is reloaded and continues counting down.

Counter Control Register A ($[T32AxCRA]$)

Clock selection <CLKA[2:0]> Any clock
 Count operation <UPDNA[1:0]> Down counting
 Start operation <STARTA[2:0]> Any timing
 Stop operation <STOPA[2:0]> Any timing
 Reload Condition <RELDA[2:0]> A match with $[T32AxRGA1]$

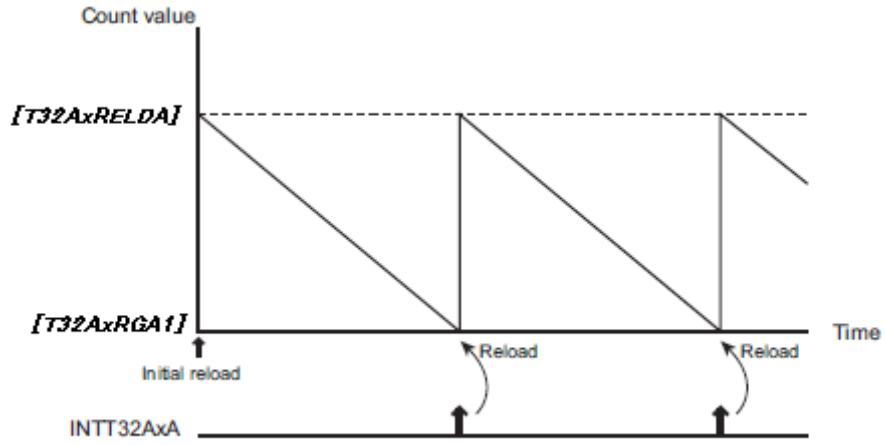


Figure 5.2 Interval Timer (Count down)

5.2. Event Counter

An external trigger, internal trigger, or timer output from other timer can be selected as the event counter. As the count operation, up counting should be selected. The counter counts up on the rising edge/falling edge of the external trigger, on the rising edge/falling edge of the output from other timer, or at the occurrence of the internal trigger. The number of occurrences of the events can be read from $[T32AxTMRA]$ as the count value.

Counter Control Register A ($[T32AxCRA]$)

| | |
|-------------------------------|--|
| Clock selection <CLKA[2:0]> | External triggers / Internal triggers/ outputs from other timer. |
| Count operation <UPDNA[1:0]> | Up counting |
| Start operation <STARTA[2:0]> | Any timing |
| Stop operation <STOPA[2:0]> | Any timing |
| Reload condition <RELDA[2:0]> | Any condition |

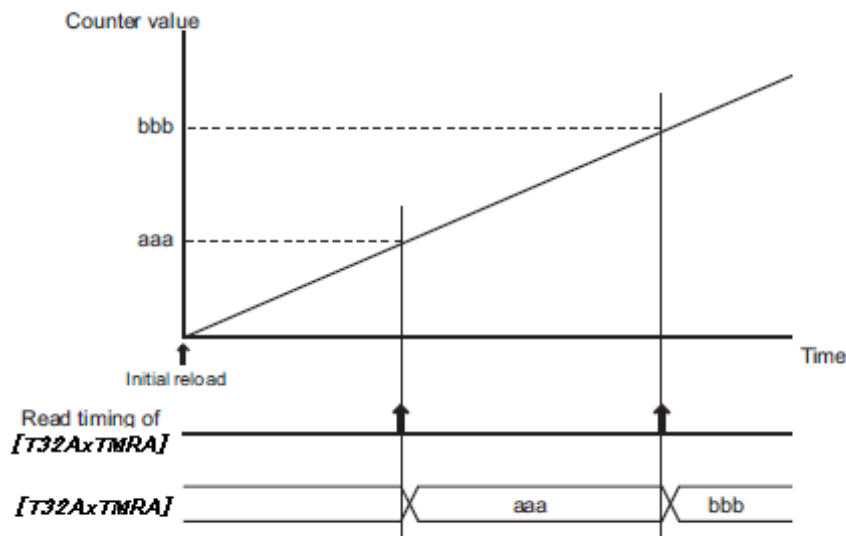


Figure 5.3 Event counter

5.3. Programmable Rectangular Wave Output(PPG)

The T32A can output rectangular wave at any frequency and any duty ratio.
The following is the definition of the period and duty ratio.

| | Period | Duty ratio |
|-------------|------------------------------|------------------------------|
| Up counting | $[T32AxRGA1] - [T32AxRELDA]$ | $[T32AxRGA0] - [T32AxRELDA]$ |

An output from Timer A (T32AxOUTA) is initially a "Low" signal. The level of the signal can be changed with $[T32AxOUTCRA0] <OCRA>$. Writing data to $<OCRA>$ is always enabled.

A timer output can be set, cleared, or reversed when Timer A counter matches the timer register ($[T32AxRGA0]$, $[T32AxRGA1]$). Any rectangular wave can be output from the T32AxOUTA pin.

When double buffering is enabled, a value in the register buffer A0/A1 is transferred to $[T32AxRGA0/A1]$ when the counter matches $[T32AxRGA1]$. This can change the frequency and duty ratio without regard to the update timing of the timer register.

Counter Control Register A ($[T32AxCRA]$)

Clock selection $<CLKA[2:0]>$ Any clock
 Count operation $<UPDNA[1:0]>$ Up counting
 Start operation $<STARTA[2:0]>$ Any timing
 Stop operation $<STOPA[2:0]>$ Any timing
 Reload condition $<RELDA[2:0]>$ A match with the Timer Register A1 ($[TX32ARGA1]$) (Reloaded to 0x0000)
 Controls double-buffering $<WBFA>$ Enable

This figure shows a waveform when the output control register is specified as below:

Output Control Register A0 ($[T32AxOUTCRA0]$)

Operation of T32AxOUTA $<OCRA[1:0]>$ No change

Output Control Register A1 ($[T32AxOUTCRA1]$)

Operation when the counter matches $[T32AREGA0]$ Set ("High")

$<OCRCMPA0[1:0]>$.

Operation when the counter matches $[T32AREGA1]$ Clear ("Low")

$<OCRCMPA1[1:0]>$.

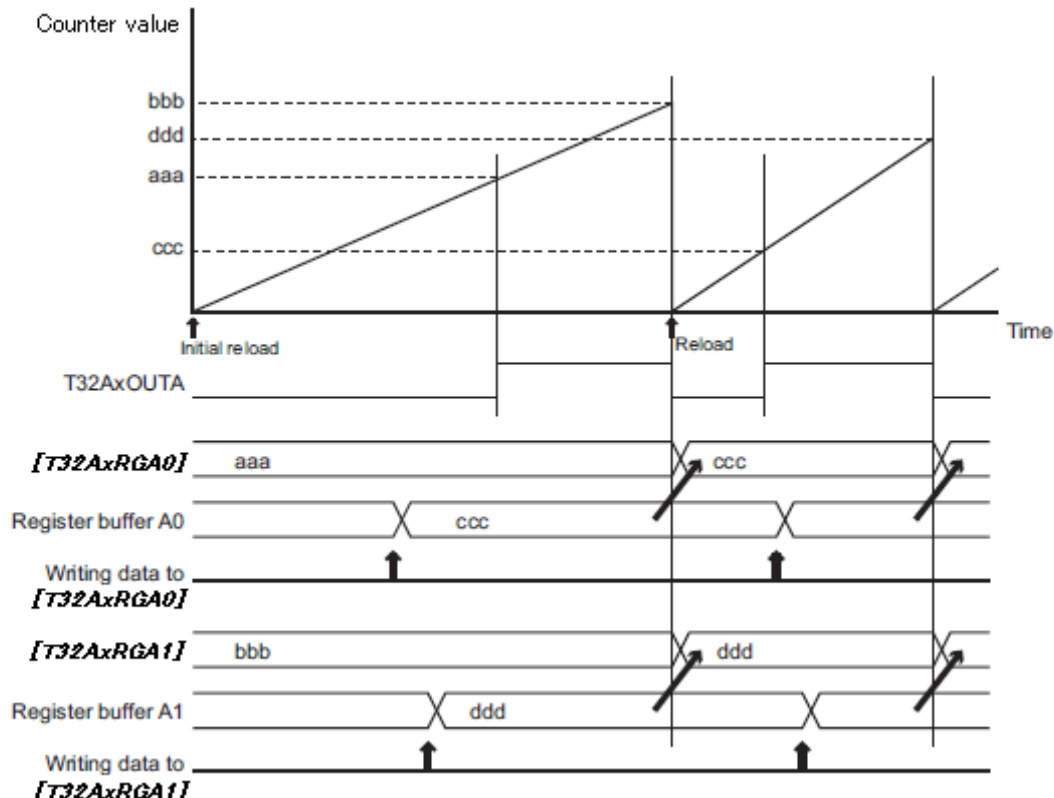


Figure 5.4 Example of PPG output(Low active)

While a duty ratio is changed periodically, when T32AxOUTA is controlled, if a duty ratio unexpectedly exceeds the setting value or "0", the following operation will be led.

When a duty ratio is equal to a period (Period A), the conditions of set and clear of the timer output are established at the same time. Since multiple factors are established, T32AxOUTA is not changed.

When a duty ratio is equal to "0" (Period B), the set and clear conditions are established at the same time by period match, but the output does not change when multiple factors are simultaneously satisfied. In the example of the waveform below, "High" continues to be output

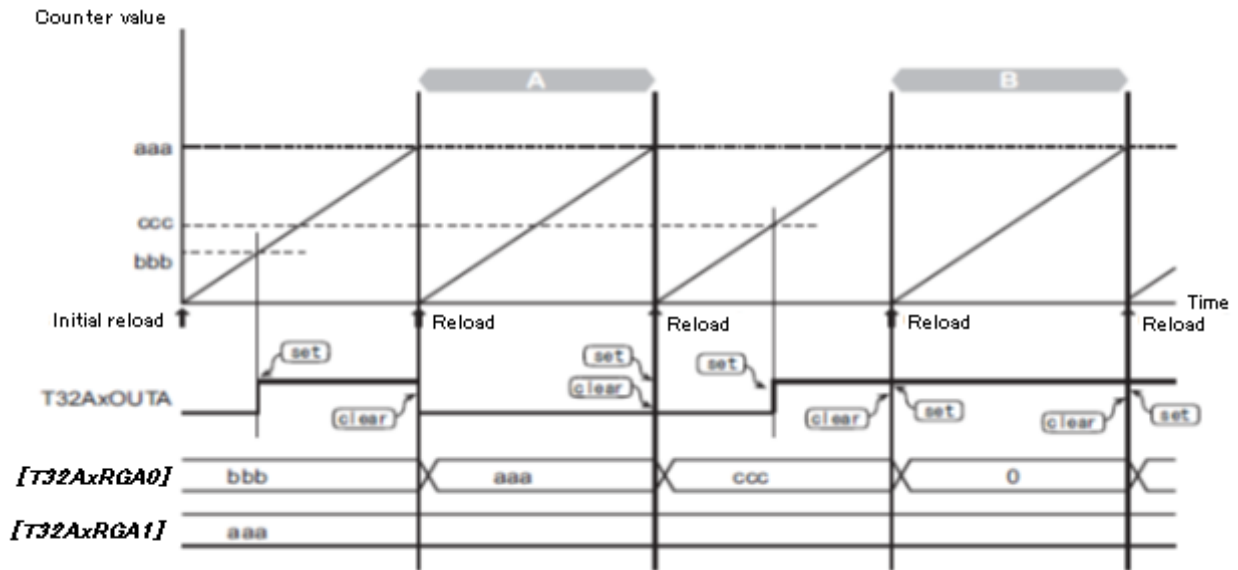


Figure 5.5 PPG output (<Period A, B>)

5.4. Programmable Rectangular Wave Output (PPG) by External Trigger

By starting count using an external trigger, a PPG with short delay time, which is too short to be handled by program can be output.

Counter Control Register A (*[T32AxCRA]*)

| | |
|-------------------------------|--|
| Clock selection <CLKA[2:0]> | Any clock |
| Count operation <UPDNA[1:0]> | Up counting |
| Start operation <STARTA[2:0]> | On the rising edge of the external trigger |
| Stop operation <STOPA[2:0]> | Any timing |
| Reload Condition <RELDA[2:0]> | A match with the Timer Register A1 (<i>[T32AxRGA1]</i>) (Reloaded to 0x0000) |

The figure shows a waveform when the output control register is specified as below:

Output Control Register A0 (*[T32AxOUTCRA0]*)

Operation of T32AxOUTA <OCRA[1:0]> No change

Output Control Register A0 (*[T32AxOUTCRA1]*)

Operation when the counter matches *[T32AxRGA0]*.
<OCRCMPA0[1:0]> Set ("High")

Operation when the counter matches *[T32AxRGA1]*.
<OCRCMPA1[1:0]> Clear ("Low")

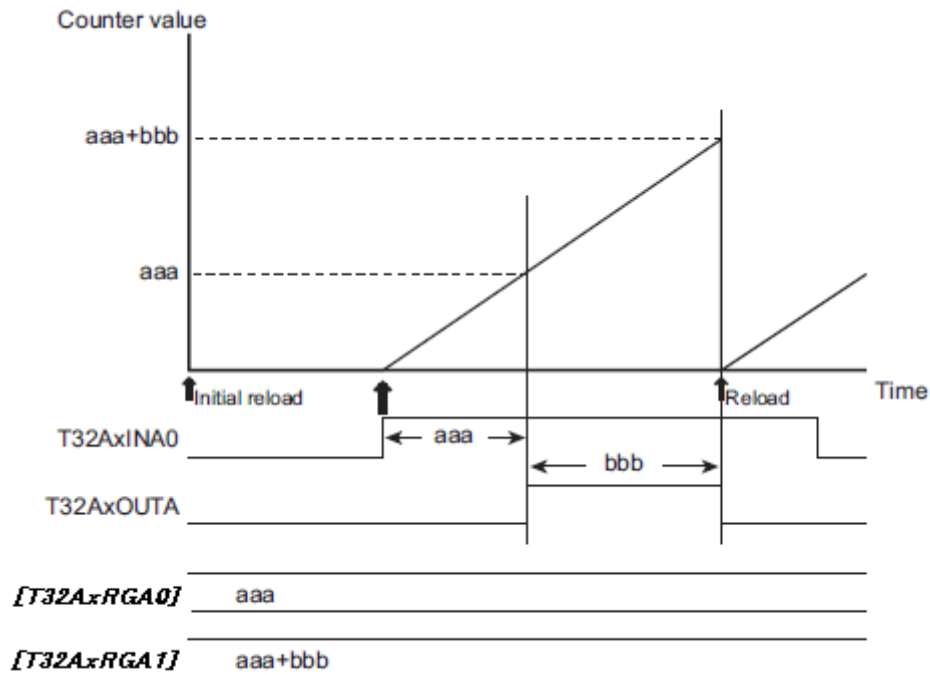


Figure 5.6 Example of PPG output by External Trigger

A delay time "aaa" from the external trigger is specified to *[T32AxRGA0]*. The value "aaa" + "bbb" of which a delay time "aaa" is added to one-shot pulse width "bbb" is specified to *[T32AxRGA1]*.

At this time, if an external trigger pulse is input to T32AxINA0, the counter starts on the rising edge of the external trigger. When the counter value is incremented to "aaa", the counter value matches *[T32AxRGA0]*. This allows T32AxOUTA to be "High" level. When the counter counts up until the value is "aaa" + "bbb", the counter value matches *[T32AxRGA1]*. This clears T32AxOUTA and allows T32AxOUTA to be "Low" level.

5.5. PPG Outputs by Synchronous Operation

PPG outputs are signals having different duty ratio programmable rectangular waves at the same frequency using the combination of multiple timers. In this case, Timer A is used as the master, and Timer B is used as the slave.

The following is the definition of the period and duty ratio.

| | Period | Duty ratio (Timer A output) | Duty ratio (Timer B output) |
|-------------|------------------------------|------------------------------|--|
| Up counting | $[T32AxRGA1] - [T32AxRELDA]$ | $[T32AxRGA0] - [T32AxRELDA]$ | $[T32AxRGB0] - [T32AxRELDB] + [T32AxRGA1] - [T32AxRGB1]$ |

Set Timer A to output a programmable rectangular wave. When the counter of timer A reaches "aaa", it matches $[T32AxRGA0]$ and then the level of T32AxOUTA reverse. When the counter reaches "bbb", it matches $[T32AxRGA1]$ and then counter value is reloaded and counter is continue count up.

Set the conditions of start, stop, and reload of Timer B to synchronous operation. Set the value to output a desired rectangular wave to the timer register B0/B1 $[T32AxRGB0/ T32AxRGB1]$. (set "ccc" to $[T32AxRGB0]$, and set "ddd" to $[T32AxRGB1]$) Set the reload value to the same value as the reload value of Timer A.

Counter Control Register A ($[T32AxCRA]$)

| | |
|-------------------------------|---|
| Clock selection <CLKA[2:0]> | Any clock |
| Count operation <UPDNA[1:0]> | Up counting |
| Start operation <STARTA[2:0]> | Any timing |
| Stop operation <STOPA[2:0]> | Any timing |
| Reload condition <RELDA[2:0]> | A match with $[T32AxRGA1]$ (0x0000 is reloaded) |

Timer Register A0/A1

| | |
|-------------------------------------|-------|
| Timer Register A0 ($[T32AxRGA0]$) | "aaa" |
| Timer Register A1 ($[T32AxRGA1]$) | "bbb" |

Output Control Register A1 ($[T32AxOUTCRA1]$)

| | |
|---|---------------|
| Operation when the counter matches $[T32AxRGA0]$. <OCRCMPA0[1:0]> | Set ("High") |
| Operation when the counter matches $[T32AxRGA1]$. <OCRCMPA1[1:0]> | Clear ("Low") |

Counter Control Register B ($[T32AxCRB]$)

| | |
|-------------------------------|-----------------------|
| Clock selection <CLKB[2:0]> | Same as Timer A |
| Count operation <UPDNB[1:0]> | Up counting |
| Start operation <STARTB[2:0]> | Synchronous operation |
| Stop operation <STOPB[2:0]> | Synchronous operation |
| Reload Condition <RELDB[2:0]> | Synchronous operation |

Timer Register B0/B1

| | |
|-------------------------------------|-------|
| Timer Register B0 ($[T32AxRGB0]$) | "ccc" |
| Timer Register B1 ($[T32AxRGB1]$) | "ddd" |

Output Control Register B1 ($[T32AxOUTCRB1]$)

| | |
|---|---------------|
| Operation when the counter matches $[T32AxRGB0]$. <OCRCMPB0[1:0]> | Set ("High") |
| Operation when the counter matches $[T32AxRGB1]$. <OCRCMPB1[1:0]> | Clear ("Low") |

5.6. Control a stepping motor by PPG Outputs

PPG outputs control a stepping motor using the combination of two timers. In this example, PPG outputs are used in 32-bit timer. It uses internal trigger for timer stop condition. The internal trigger cannot be used depending on the product. Refer to "Product Information" of reference manual.

One timer (channel 1) is used to output a rectangular wave; another timer (channel 0) is used to control the timer (channel 1). The counter value of channel 0 indicates the specific position information.

Channel 0 uses the rectangular output from channel 1 as the count clock. It counts on the falling edge of the signal from channel 1. After the reload value (bbb) and the compare match value (aaa) are set, perform the soft start.

Start the operation of channel 1 at the desired conditions. Connect an internal trigger of channel 1 to the match detection circuit of channel 0 to stop the channel 1 at this condition.

When a rectangular wave of channel 1 is output, channel 0 starts down counting from the reloaded value until the counter is aaa to output an interrupt. Channel 1 detects a compare match as an internal trigger and stops operation. Since a rectangular wave is stopped to be output, channel 0 stops operation as well.

Change the setting of ch0 to count up and set the value (ccc) used for a compare match.

When channel 1 restarts to output a rectangular wave, channel 0 starts up counting until the counter is ccc to output an interrupt.

Counter Control Register C (*T32AxCRC*) (channel 0)

| | |
|-------------------------------|--|
| Clock selection <CLKC[2:0]> | On the falling edge of an output from other timer (rectangular wave output of channel 1) |
| Count operation <UPDNC[1:0]> | Down counting (switching to up counting during the count operation) |
| Start operation <STARTC[2:0]> | No trigger is used |
| Stop operation <STOPC[2:0]> | Any timing |
| Reload Condition <RELDC[2:0]> | None (free-running) |

RUN Register C (*T32AxRUNC*) (channel 0)

| | |
|---------------------------|---|
| Start operation <SFTSTAC> | Start the counter operation by the program. |
|---------------------------|---|

Counter Control Register C (*T32AxCRC*) (channel 1)

| | |
|-------------------------------|---|
| Clock selection <CLKC[2:0]> | Any clock |
| Count Operation <UPDNC[1:0]> | Any count operation |
| Start operation <STARTC[2:0]> | Any timing |
| Stop operation <STOPC[2:0]> | Internal trigger (a match detection of channel 0) |
| Reload Condition <RELDC[2:0]> | Any condition |

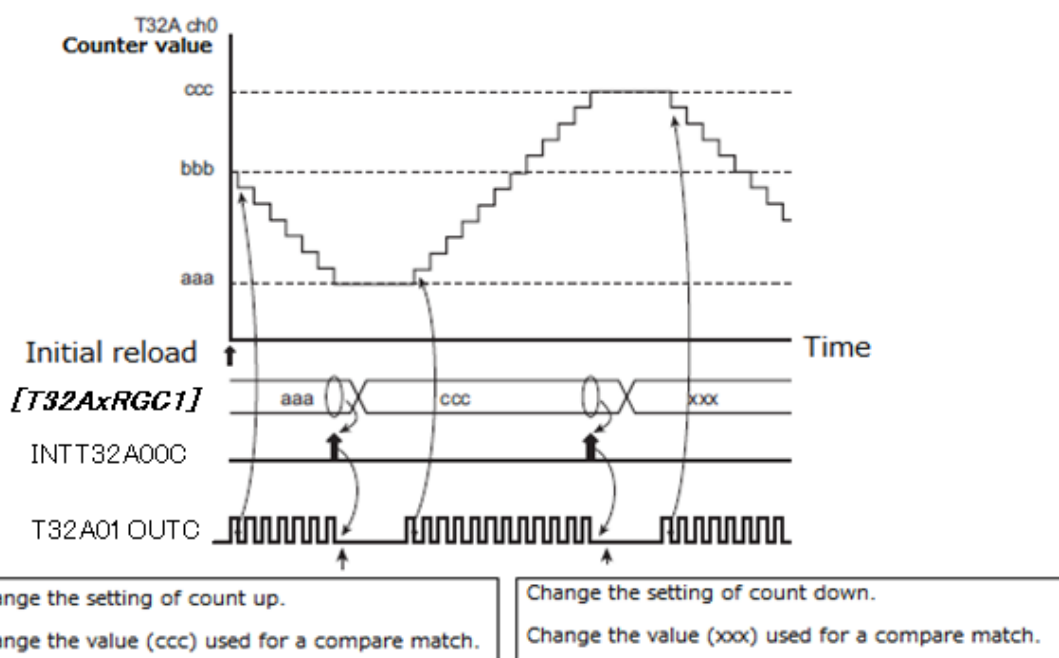


Figure 5.8 PPG Outputs by Synchronous Operation (Example of control of stepping motor)

5.7. Frequency Measurement Using Capture Function

The following case is an example when a clock frequency input from an external source is measured.

In this case, Timer B is used as an interval timer; Timer A is used to output a programmable rectangular wave.

Use the external trigger (T32AxINB0) as the count clock for Timer B. Timer B counts up as a free-running counter.

Specify Timer A to generate the pulses (T32AxOUTA) for measurement time. Connect Timer A to an output from other timer at Timer B (T32AxTRGINBPHCK).

Specify the capture control register [T32AxCAPCRB] of Timer B to capture the counter value to [T32AxCAPB0] on the rising edge of an output from other timer and to capture the counter value to [T32AxCAPB1] on the falling edge of an output from other timer.

When the counter of Timer A matches [T32AxRGA0], T32AxOUTA rises. The counter value of Timer B is captured to [T32AxCAPB0]. When the counter of Timer A matches [T32AxRGA1], T32AxOUTA falls. The counter value of Timer B is captured to [T32AxCAPB1].

A frequency is determined by dividing ([T32AxCAPB1]-[T32AxCAPB0]) by the pulse width of T32AxOUTA.

Counter Control Register A ([T32AxCRB])

| | |
|-------------------------------|------------------|
| Clock selection <CLKB[2:0]> | External trigger |
| Count operation <UPDNB[1:0]> | Up counting |
| Start operation <STARTB[2:0]> | Any timing |
| Stop operation <STOPB[2:0]> | Any timing |
| Reload condition <RELDB[2:0]> | Any condition |

Capture Control Register B ([T32AxCAPCRB])

| | |
|---|--|
| Capture timing of [T32AxCAPB0] <CAPMB0[2:0]> | On the rising edge of an output from other timer<T32AxTRGINBPHCK> |
| Capture timing of [T32AxCAPB1] <CAPMB1[2:0]> | On the falling edge of an output from other timer<T32AxTRGINBPHCK> |

Counter Control Register A ([T32AxCRA])

| | |
|-------------------------------|---|
| Clock selection <CLKA[2:0]> | Any clock |
| Count operation <UPDNA[1:0]> | Up counting |
| Start operation <STARTA[2:0]> | Any timing |
| Stop operation <STOPA[2:0]> | Any timing |
| Reload condition <RELDA[2:0]> | A match with Timer Register A1 (Reloaded to 0x0000) |

Timer Register A0([T32AxRGA0]) Set the value to generate the pulse for the measurement time.
Timer Register A1([T32AxRGA1])

Output Control Register A([T32AxOUTCRA0])

Operation of T32AxOUTA <OCRA[1:0]> No change

Output Control Register A1([T32AxOUTCRA1])

Operation when the counter matches [T32AxRGA0].
<OCRCMPA0[1:0]> Set ("High")
Operation when the counter matches [T32AxRGA1].
<OCRCMPA1[1:0]> Clear ("Low")

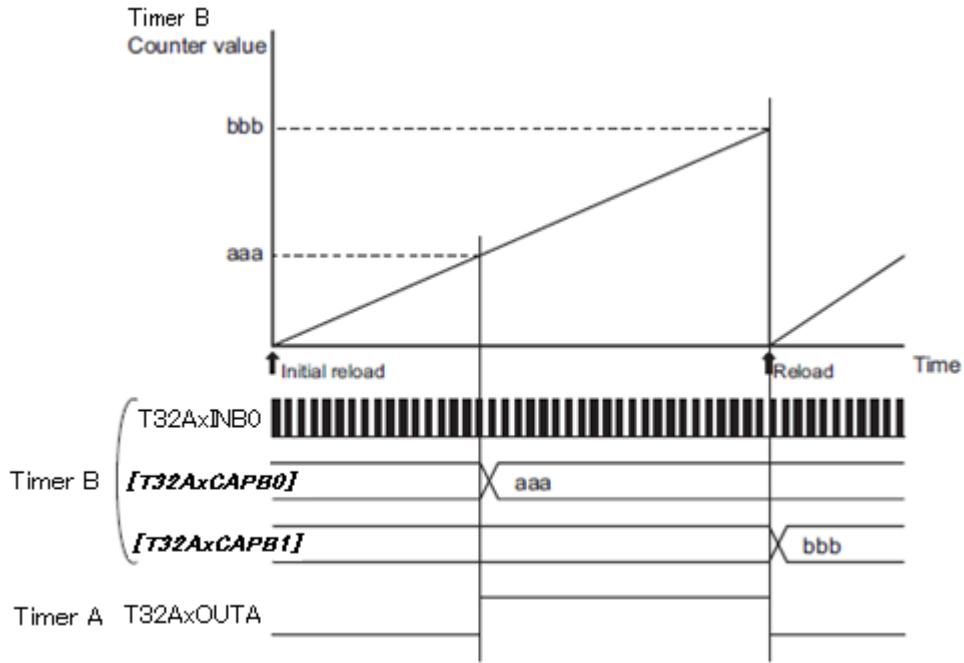


Figure 5.9 Frequency Measurement Using Capture Function

5.8. Pulse Width Measurement Using Capture Function

The following case is an example to measure the "High" level pulse width input from an external source.

With *[T32AxCAPCRA]*, the counter value is captured to *[T32AxCAPA0]* on the rising edge of the T32AxINA0 pin; to *[T32AxCAPA1]* on the falling edge of the T32AxINA0.

When a rising signal of the external pulse is input to the T32AxINA0 pin, the counter value is captured to *[T32AxCAPA0]*. When a falling signal of external pulse is input to T32AxINA0, the counter value is captured to *[T32AxCAPA1]*.

The width of "High" level of the external pulse can be determined the difference between *[T32AxCAPA1]* and *[T32AxCAPA0]*; then, multiplying the difference by the clock cycle of the prescaler output.

Counter Control Register A (*[T32AxCRA]*)

| | |
|-------------------------------|------------------|
| Clock selection <CLKA[2:0]> | Prescaler output |
| Count Operation <UPDNA[1:0]> | Up counting |
| Start operation <STARTA[2:0]> | Any timing |
| Stop operation <STOPA[2:0]> | Any timing |
| Reload condition <RELDA[2:0]> | Any condition |

Capture Control Register A (*[T32AxCAPCRA]*)

| | |
|--|----------------------------------|
| Capture timing of <i>[T32AxCAPA0]</i> <CAPMA0[2:0]> | On the rising edge of T32AxINA0 |
| Capture timing of <i>[T32AxCAPA1]</i> <CAPMA1[2:0]> | On the falling edge of T32AxINA0 |

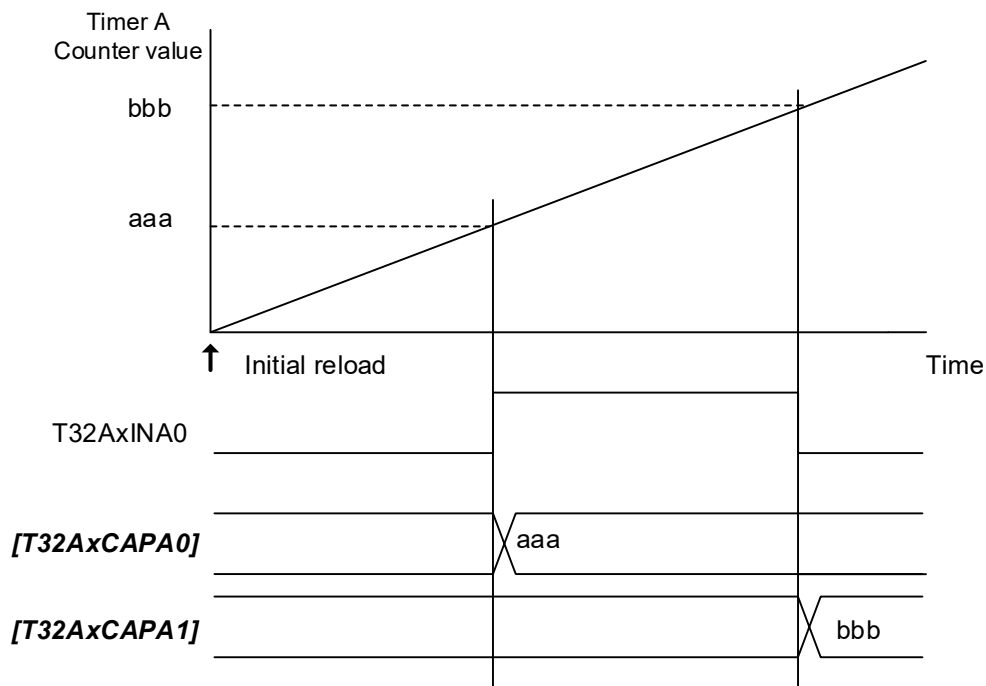


Figure 5.10 Pulse Width Measurement Using Capture Function

5.9. Time Difference Measurement Using Capture Function

The following case is an example to measure the time difference between the pulses input from an external source.

With $[T32AxCAPCRA]$, capture the counter value to $[T32AxCAPA0]$ on the rising edge of the T32AxINA0 pin; capture the counter value to $[T32AxCAPA1]$ on the rising edge of the T32AxINA1 pin.

A time difference is determined by subtracting $[T32AxCAPA0]$ from $[T32AxCAPA1]$; then multiplying the calculated value by the clock cycle of the prescaler output.

Counter Control Register A ($[T32AxCRA]$)

| | |
|-------------------------------|------------------|
| Clock selection <CLKA[2:0]> | Prescaler output |
| Count operation <UPDNA[1:0]> | Up counting |
| Start operation <STARTA[2:0]> | Any timing |
| Stop operation <STOPA[2:0]> | Any timing |
| Reload condition <RELDA[2:0]> | Any condition |

Capture Control Register A ($[T32AxCAPCRA]$)

| | |
|---|---------------------------------|
| Capture timing of $[T32AxCAPA0]$ <CAPMA0[2:0]> | On the rising edge of T32AxINA0 |
| Capture timing of $[T32AxCAPA1]$ <CAPMA1[2:0]> | On the rising edge of T32AxINA1 |

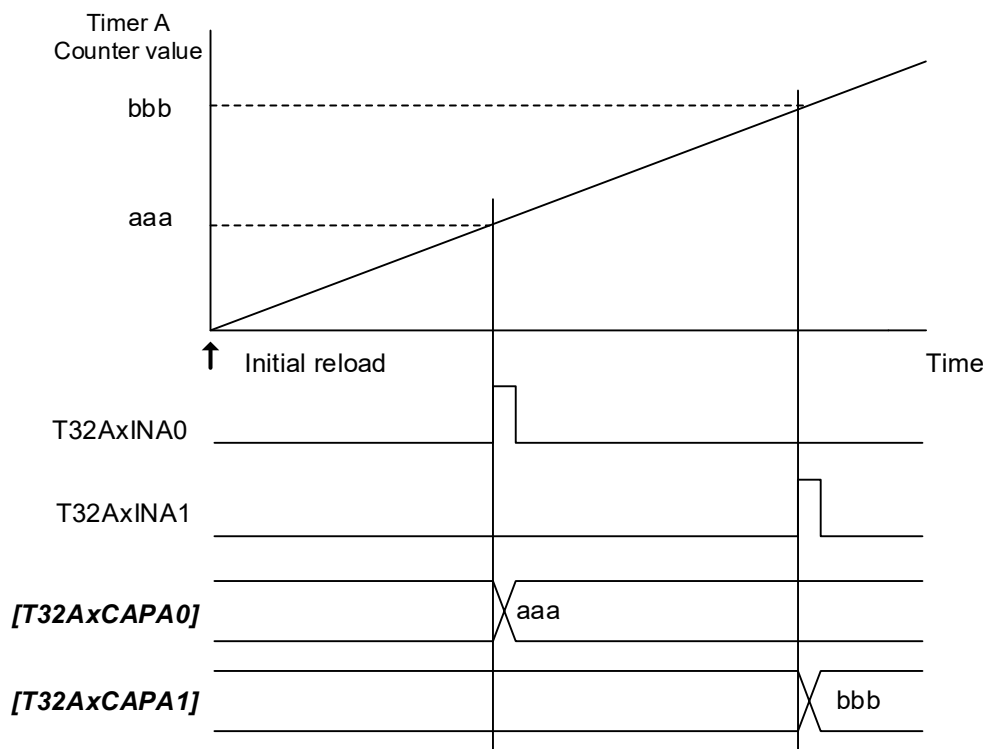


Figure 5.11 Time Difference Measurement Using Capture Function

6. Revision History

Table 6.1 Revision History

| Revision | Date | Description |
|----------|------------|---|
| 1.0 | 2017-09-12 | First release |
| 2.0 | 2018-04-23 | <ul style="list-style-type: none"> Terms and Abbreviations revised T32A Table 2.2 revised signal name of T32AxDMAREQCAP0 Figure 3.4 revised 32-bit timer connection 3.8 revised the title of table in 1 4.1 add base address TYPE3 4.3.8 revised name of Bit symbol 4.4.8 revised name of Bit symbol 5.3 revised register name([T32AxOUTCRA1] →[T32AxOUTCRA0]) Figure 5.10 revised name of register Figure 5.11 revised name of register |
| 2.1 | 2019-07-23 | <ul style="list-style-type: none"> ·Figure 2.1 revised signal name of INTT32AxACAP0, INTT32AxACAP1, INTT32AxBCAP0 and INTT32AxBCAP1. ·Table 2.1 revised signal name of INTT32AxACAP0, INTT32AxACAP1, INTT32AxBCAP0 and INTT32AxBCAP1. ·Figure 2.2 revised signal name of INTT32AxCCAP0 and INTT32AxCCAP1. ·Table 2.2 revised signal name of INTT32AxCCAP0 and INTT32AxCCAP1. ·3.9 revised signal name of INTT32AxACAP0/1, INTT32AxBCAP0/1 and INTT32AxCCAP0/1. ·3.9.2 revised the signal name of INTT32AxACAP0, INTT32AxACAP1, INTT32AxBCAP0, INTT32AxBCAP1, INTT32AxCCAP0 and INTT32AxCCAP1. ·4.3.9 add the signal name "T32AxTRGINAPCK". ·4.4.2 add the signal name "T32AxTRGINAPCK". ·4.4.9 add the signal name "T32AxTRGINAPCK". ·4.5.2 add the signal name "T32AxTRGINAPCK". ·4.5.9 add the signal name "T32AxTRGINAPCK". |
| 2.2 | 2020-05-07 | <ul style="list-style-type: none"> ·Added "TXZ +" to the family. ·Deleted the information of trademark. ·1. 32-bit timer Timer C: Corrected the description of Pulse count. ·3.1. Added "fsys supply stop register C". ·4.1. (1) Base address: ch14 and ch15 were added to TYPE 2 ·4.3.5. / 4.3.6 / 4.4.5 / 4.4.6 / 4.5.5 / 4.5.6 Added Note 2:. |
| 2.3 | 2020-08-20 | <ul style="list-style-type: none"> ·1. 32-bit timer Timer C: Revised the description of 2-pulse count. ·3.1. Clock Supply Revised "TSP1" → "T32A" ·3.3.2. Starting Operation: Revised "-- set to the timer operation control bit --" Revised " Below describes ---" ·3.3.4. Count Operation Revised " ---, the counter starts up counting ---" ·3.3.5. Event count operation Revised " --- count of events." ·3.3.6. Pulse Counting Revised " --- T32AxINC0 and T32AxINC1." Revised " --- timer interrupt occurs." (2)2-phase pulse count mode Revised " --- changes before T32AxINC1, counter is up counting." |

- Revised " --- changes before T32AxINC0, counter is down counting."
Revised " --- are state errors."
- 3.3.7. Reloading the Counter
Revised " The timer and timer register1 are shown --- "
 - 3.4. Synchronous Operation
Revised " ---, refer to "Product Information" of reference manual."
 - 3.6. Timer Register ([T32AxRGA0/B0/C0], [T32AxRGA1/B1/C1])
Revised " When double buffering is disabled, --- "
Revised " The compare value is written to timer register 0/1 --- "
Revised " When double buffering is enabled, --- "
Revised " ---, the compare value is transferred from --- "
Revised " And timer register 0 is updated at same time."
Revised " ---, you can update the compare value regardless of update timing of the timer register."
 - 3.7. Capture Control
Revised " --- to capture register0 and capture register1 ---"
 - 3.8. Output programmable rectangular wave
Revised " --- when the counter and timer register match ---"
Revised " --- when the counter is captured in capture register."
Revised " (1) Initial output control "
Revised " A PPG output can be set, --- "
 - 3.9. Interrupts
Revised " --- three types of interrupt requests ---"
 - 4.2.1. [T32AxMOD](Mode Register)
Note: Revised " When [T32AxRUNA]<RUNA>=0, [T32AxRUNB]<RUNB>=0 and [T32AxRUNC]<RUNC>=0, set the [T32AxMOD] register. "
 - 4.3.5. / 4.3.6 / 4.4.5 / 4.4.6 / 4.5.5 / 4.5.6
Note 1: Revised " --- PPG, refer to "Table 3.1 ---"
Note 2: Revised " When continuously writing ---"
" Note" → " *"
 - 4.3.8. [T32AxRELDA] (Counter Reload Register A)
Note 1: Revised " --- PPG, refer to "Table 3.1 ---"
 - 4.3.12. [T32AxIMA] (Interrupt Mask Register A)
Note 1: Revised " Even when the interrupt mask register ([T32AxIMA]) is enabled, --- "
 - 4.3.13. [T32AxSTA] (Status Register A)
Bit Symbol: INTA1
Revised " 1: A match between the counter and [T32AxRGA1]. "
Revised " --- the counter value and the Timer Register A1 --- "
Bit Symbol: INTA0
Revised " 1: A match between the counter and [T32AxRGA0] "
Revised " --- the counter value and the Timer Register A0 ---"
Note1:
Revised " Even when the interrupt mask register ([T32AxIMA]) is enabled, --- ".
Note3:
Revised " When setting flag by internal signal and clearing flag by program occur at the same time, clearing flag has priority. "
 - 4.3.14. [T32AxDMAA] (DMA Request Enable Register A)
Bit Symbol: DMAENA2
Revised " --- and the Timer Register A1 ([T32AxRGA1])"
 - 4.4.8. [T32AxRELDB] (Counter Reload Register B)
Note 1: Revised " --- PPG, refer to "Table 3.1 ---"
 - 4.4.12. [T32AxIMB] (Interrupt Mask Register B)
Note 1: Revised " Even when the interrupt mask register ([T32AxIMB]) is enabled, --- "
 - 4.4.13. [T32AxSTB] (Status Register B)

| | | |
|--|--|--|
| | | <p>Bit Symbol: INTB1 Revised " 1: A match between the counter and [T32AxRGB1]. " Revised " --- the counter value and the Timer Register B1 --- "</p> <p>Bit Symbol: INTB0 Revised " 1: A match between the counter and [T32AxRGB0] " Revised " --- the counter value and the Timer Register B0 --- "</p> <p>Note1: Revised " Even when the interrupt mask register ([T32AxIMB]) is enabled, --- "</p> <p>Note3: Revised " When setting flag by internal signal and clearing flag by program occur at the same time, clearing flag has priority. "</p> <p>·4.4.14. [T32AxDMAB] (DMA Request Enable Register B) Bit Symbol: DMAENB2 Revised " --- and the Timer Register B1 ([T32AxRGB1])"</p> <p>·4.5.8. [T32AxRELDC] (Counter Reload Register C) Note 1: Revised " --- PPG, refer to "Table 3.1 ---"</p> <p>·4.5.12. [T32AxIMC] (Interrupt Mask Register C) Note 1: Revised " Even when the interrupt mask register ([T32AxIMC]) is enabled, --- "</p> <p>·4.5.13. [T32AxSTC] (Status Register C) Bit Symbol: INTC1 Revised " 1: A match between the counter and [T32AxRGC1]. " Revised " --- the counter value and the Timer Register C1 --- "</p> <p>Bit Symbol: INTC0 Revised " 1: A match between the counter and [T32AxRGC0] " Revised " --- the counter value and the Timer Register C0 --- "</p> <p>Note1: Revised " Even when the interrupt mask register ([T32AxIMC]) is enabled, --- "</p> <p>Note3: Revised " When setting flag by internal signal and clearing flag by program occur at the same time, clearing flag has priority. "</p> <p>·4.5.14. [T32AxDMAC] (DMA Request Enable Register C) Bit Symbol: DMAENC2 Revised " --- and the Timer Register C1 ([T32AxRGC1])"</p> <p>·5.1. Interval Timer Revised " T32A can generate interrupt periodically. " Revised " The following is the definition of the period. "</p> <p>·5.2. Event Counter Revised " The number of occurrences --- "</p> <p>·5.3. Programmable Rectangular Wave Output(PPG) Revised " The following is the definition of the period and duty ratio. " Revised " --- will be led." Revised " When a duty ratio is equal to a period (Period A), the --- "</p> <p>·5.4. Programmable Rectangular Wave Output (PPG) by External Trigger Revised " A delay time "aaa" from the external trigger --- "</p> <p>·5.5. PPG Outputs by Synchronous Operation Revised " The following is the definition of the period and duty ratio. " Revised " --- counter value is reloaded --- " Revised " --- timer register B0/B1 ---" Added " Set the reload value to the same value as the reload value of Timer A. "</p> <p>·5.6. Control a stepping motor by PPG Outputs Revised " Refer to "Product Information" of reference manual. "</p> |
|--|--|--|

RESTRICTIONS ON PRODUCT USE

Toshiba Corporation and its subsidiaries and affiliates are collectively referred to as "TOSHIBA". Hardware, software and systems described in this document are collectively referred to as "Product".

- TOSHIBA reserves the right to make changes to the information in this document and related Product without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE").** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, lifesaving and/or life supporting medical equipment, equipment used for automobiles, trains, ships and other transportation, traffic signaling equipment, equipment used to control combustions or explosions, safety devices, elevators and escalators, and devices related to power plant. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative or contact us via our website.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**