Application Processor Lite *ApP Lite*

TZ1000 Series

Reference Manual

MCU Real Time Clock Counter

Revision 1.2

2018-02

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION

Table of Contents

List of Figures

List of Tables

Preface

This document provides the specification for the Real Time Clock Counter designed for the TZ1000 Series.

Intended Audience

This document is intended for the following users.

Driver software developers.

System designers

Conventions in this Document

- The following notational conventions apply to numbers:
  Hexadecimal number: 0xABC
  Decimal number: 123 or 0d123 - Only when it should be explicitly indicated that the number is decimal.
  Binary number: 0b111 - It is possible to omit the "0b" when the number of bit can be distinctly understood from a sentence.
- Low active signals are indicated with a name suffixed with "_N".
- A signal is asserted when it goes to its active level while it is deasserted when it goes to its inactive level.
- A set of multiple signals may be referred to as [m:n].
  Example: S[3:0] indicates four signals, S3, S2, S1 and S0, collectively.
- In the text, register names are enclosed in brackets *[ ]*.
  Example:*[ABCD]*
- A set of multiple registers, fields or bits of the same type may be described collectively using "n".
  Example: *[XYZ1]*, *[XYZ2]*, and *[XYZ3]* to *[XYZn]*
- A range of register bits are referred to as [m:n].
  Example: [3:0] indicates a range from bit 3 to bit 0.
- Values set in registers are indicated using either a hexadecimal or binary number.
- Example: *[ABCD]*.EFG = 0x01 (hexadecimal), *[XYZn]*.VW = 1 (binary)
- Words and bytes are defined as follows:
  Byte: 8 bits
  Halfword: 16 bits
  Word: 32 bits
  Doubleword: 64 bits
- Register bit attributes are defined as follows:
  R: Read-only
  W: Write-only
  W1C: Clear by write of 1 - A write of "1" clears the corresponding bit to 0.
  W1S: Set by write of 1 - A write of "1" sets the corresponding bit to 1.
  R/W: Read/Write
  R/W0C: Read/Clear by write of 0
  R/W1C: Read/Clear by write of 1
  R/W1S: Read/Set by write of 1
  RS/WC: Set by read/Clear by write - Set after a read and cleared after a data write.
- Registers only support word access unless otherwise specified.
- Any registers defined as Reserved in the text must not be rewritten. Also, any values read from such registers should not be used.
- Any bits for which default values are defined as "—" would return undefined values if read.
- When a data is written to a register containing both writable and read-only (R) bit fields, its default values should be written to read-only (R) bit fields. For any bit fields with default values defined as "—," refer to the definitions of the relevant register.
- Default values should be written to any reserved bit fields in a write-only register. For any bit fields with default values defined as "—," refer to the definitions of the relevant register.

Abbreviation

These specifications introduce a part of the abbreviation which they used

AMBA	Advanced Microcontroller Bus Architecture (Arm® AMBA® interconnect)
APB	Advanced Peripheral Bus

1. Overview

The feature of this module, RTC (Real Time Clock), is as follows.

- APB slave which is compliant with AMBA® standard (2.0).
- A timer whose start time is set in a register and which counts up per 1 second.
  The timer can count the time between January 1 in the year 00 and December 31, 59 minutes and 59 seconds, in the year 99.
- Alarm function
  An interrupt signal is issued when the count of the timer matches the set value.
- Periodic interrupt function
  An interrupt signal is continuously issued at the set period (per second, per minute, per hour, per day, per month, or per 1/2 seconds to per 1/256 seconds).
- Interval-timer interrupt function
  A 16-bit timer counter measures the time between 0.977 ms and 32 seconds. An interrupt signal is issued at the set time.
- The ORed interrupt signal of the alarm function, the periodic interrupt, and the interval-timer interrupt can be issued.
- The interrupt status before the mask and after the mask can be checked (the alarm, the periodic, and the interval-timer interrupts).

2. Block Diagram

The internal block diagram is shown in Figure 2.1.

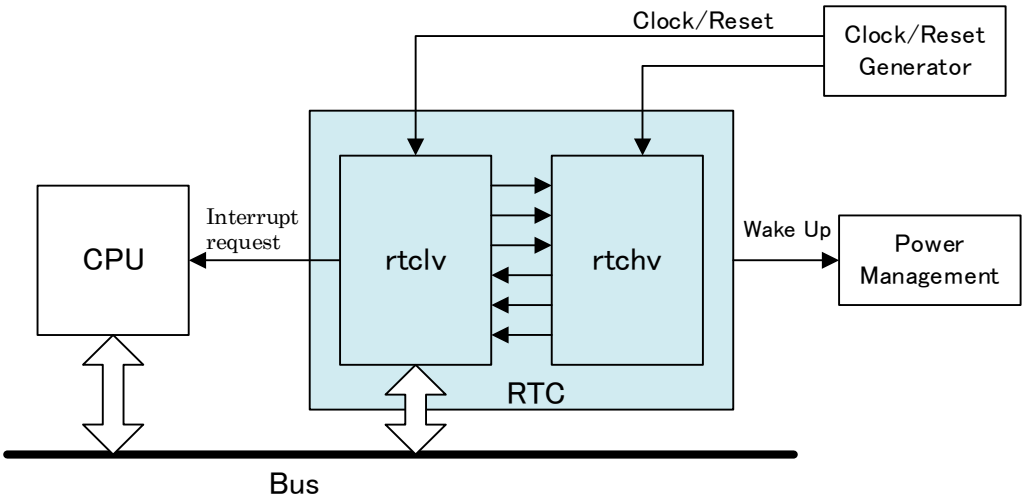

Figure 2.1 RTC and external block connection

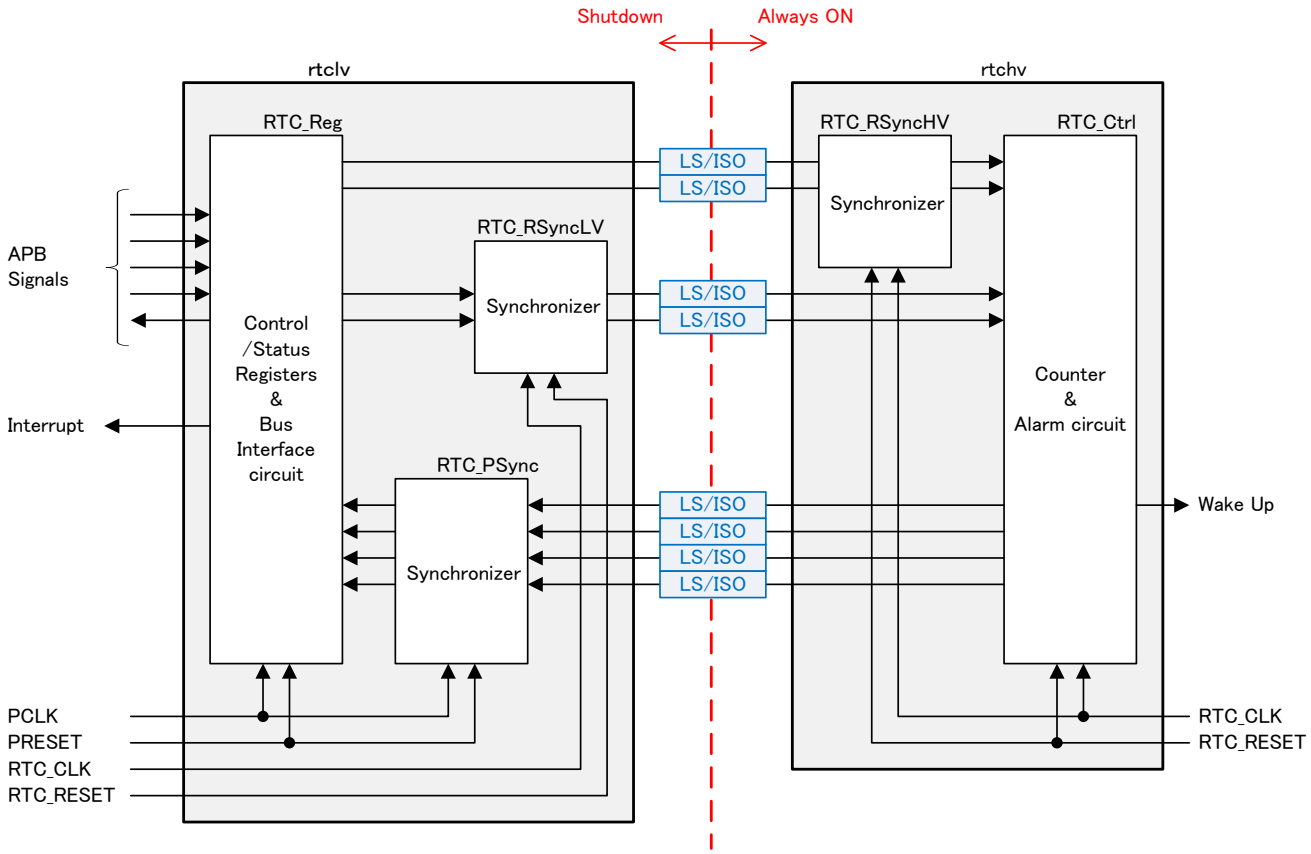

Figure 2.2 RTC internal block diagram

The RTC consists of rtclv and rtchv blocks.

The power of the rtclv block can be shut down in a power mode. This block includes a control register, a status register, a bus interface circuit, and an interrupt signal generator.

The rtchv block is always supplied with power. This block includes an RTC counter, an alarm signal, a periodic and interval-timer interrupt signal, and a interrupt generator.

The internal block diagram of the RTC is shown in Figure 2.2.

The feature of each internal block is as follows.

- **RTC_Reg**
  This block includes a control register, a status register, a bus interface, and others. The set values in the registers are transferred to the counter. And when the alarm signal, the periodic interrupt, or the interval-timer interrupt is detected, an interrupt signal is issued to the CPU. It operates with the APB clock (PCLK).
- **RTC_Ctrl**
  This block is a counter part of the RTC. Counters for the second, minute, o'clock, day, week, month, and year are implemented. It counts up the time per one second. The count value is compared with the values in the registers where the periodic and interval-timer are set and this block generates the periodic interrupt and the interval-timer interrupt signals. It operates with the RTC clock (32.768 kHz).
- **RTC_Psync, RTC_RsyncLV, and RTC_RsyncHV**
  These blocks synchronize signals which are connected between the blocks operating with different clocks. The target signals are the signals between the APB clock domain and the RTC clock domain.

3. Address map

Table 3.1 MCU Real Time Clock Counter Register map

Register Name	Type	Width	Reset Value	Address Offset
RTC_TIMESET1	RW	32	0x0000 0000	0x0000 0000
RTC_TIMESET2	RW	32	0x0000 0000	0x0000 0004
RTC_TIMESET3	RW	32	0x0000 0000	0x0000 0008
RTC_TIMESET4	RW	32	0x0000 0000	0x0000 000C
RTC_CTRL	RW	32	0x0000 0080	0x0000 0010
RTC_STATUS	RO	32	0x0000 0000	0x0000 0014
RTC_CTRL_WIP	RO	32	0x0000 0000	0x0000 0018
RTC_RIS	RO	32	0x0000 0000	0x0000 0020
RTC_MIS	RO	32	0x0000 0000	0x0000 0024
RTC_INTMASK	RW	32	0x0000 0000	0x0000 0028
RTC_PERIODIC	RW	32	0x0000 0000	0x0000 002C
RTC_PERIOD_INTCLR	RW	32	-	0x0000 0030
RTC_ALARM_INTCLR	RW	32	-	0x0000 0034
RTC_INTVAL_INTCLR	RW	32	-	0x0000 0038
RTC_CURRENT1	RO	32	0x0000 0000	0x0000 0040
RTC_CURRENT2	RO	32	0x0000 0000	0x0000 0044
RTC_CURRENT3	RO	32	0x0000 0000	0x0000 0048
RTC_CURRENT4	RO	32	0x0000 0000	0x0000 004C
RTC_ALARM1	RW	32	0x0000 0000	0x0000 0050
RTC_ALARM2	RW	32	0x0000 0000	0x0000 0054
RTC_ALARM3	RW	32	0x0000 0000	0x0000 0058
RTC_INTVAL_SET	RW	32	0x0000 FFFF	0x0000 0060
RTC_INTVAL_TMR	RO	32	0x0000 FFFF	0x0000 0064

4. Functional Description

4.1. Setting Date and Time

The RTC counts the time as a clock. The necessary setting is shown as follows.

(1) Time setting
To set the current time, it is necessary that all of the followings should be set. If a value out of the range or non-existing time like the "0 month 0 day" is set, the RTC operation is not guaranteed.

Table 4.1 Current time setting and setting range

Item	Range	
	10 digit	1 digit
Second	5 to 0	9 to 0
Minute	5 to 0	9 to 0
O'clock (Note 1)	2 to 0	9 to 0
Day	3 to 0	9 to 0
Day of a week (Note 2)	None	6 to 0
Month	1 and 0	9 to 0
Year (Note 3)	9 to 0	9 to 0

If the following setting sets the time of "59 seconds, 59 minutes, 23 o'clock, 31, December, in the year 99, and Saturday."

Table 4.2 Setting example of time

Item	10 digit	1 digit
Second	5	9
Minute	5	9
O'clock	2	3
Day	3	1
Day of a week	None	6
Month	1	2
Year	9	9

Note 1: The 24-hour clock and 12-hour clock are available. Refer to "The other setting items."

Note 2: The setting of the day of a week is as follows.

0: Sunday
1: Monday
2: Tuesday
3: Wednesday
4: Thursday
5: Friday
6: Saturday

Note 3: For a leap year setting, refer to "The other setting items."

(2) Reflection of the setting
The time, the day, the o'clock, the day, and so one are written to the *[RTC_TIMESETn]* (n=1,2,3, and 4) register. Then, by writing 1 to the TMRST field in the *[RTC_CTRL]* register, the RTC is updated to the setting. The TMRST field is automatically cleared to 0 when the load finishes. If the field is written to 1 before it becomes 0, the write is ignored. The TMRST field in the *[RTC_CTRL]* register should be read to check whether the field becomes 0 or not. The values in the *[RTC_TIMESETn]* (n=1,2,3, and 4) register cannot beloaded independently .

(3) Adjustment of the setting time (ADJUST function)
The RTC has a correction function for the setting of the second. The function is as follows.

- The second counter value is set to 0 when the count value is amond 0 to 29.
- The minute counter value is incremented and the second counter is set to 0 when the second couter value is among 30 to 59.

The increment is done only in the minute counter. The o'clock counter and the successive counters are not incremented. Some examples are shown as follows.

- At January 1, 11 o'clock, 09 minutes, and 29 seconds, the ADJUST is done. => January 1, 11 o'clock, 09 minutes, and 00 seconds.
- At January 1, 11 o'clock, 09 minutes, and 30 seconds, the ADJUST is done. => January 1, 11 o'clock 10 minutes, 00 seconds.
- At January 1, 23 o'clock, 59 minutes, and 30 seconds, the ADJUST is done. => January 1, 23 o'clock, 00 minutes, and 00 seconds.

When the ADJUST function is used, the [RTC_CTRL] register should be set as follows.

- TMEN field (bit 7) => Set to 1.
- TMRST field (bit 6) => Set to 0.
- ADJUST field (bit 4) => Set to 1.
- PRRST field (bit 2) => Set to 0.
- ALRST field (bit 0) => Set to 0.

The TMRST, PRRST, and ALRST fields are given priority to the ADJUST field. When the ADJUST function is used, the TMRST, ALRST, and PRRST fields should be set to 1. Then, they become 0. And the ADJUST field should be set to 1. Before the TMRST, ALRST, and PRRST fields become 0, to write 1 to the ADJUST field is ignored.

4.2. Alarm Setting

The setting of the alarm time is the same as that of the current time.

Table 4.3 Alarm time setting and setting range

Item	Range	
	10 digit	1 digit
Minute	5 to 0	9 to 0
O'clock	2 to 0	9 to 0
Day	3 to 0	9 to 0
Day of a week	None	6 to 0

The alarm time is set to the *[RTC_ALARMn]* (n=1,2,3) registers. And, the ALRST field in the *[RTC_CTRL]* register should be written to 1, then the value becomes valid in the RTC. The alarm time is not loaded when the TMRST field in the *[RTC_CTRL]* register is active, because the setting path of the ALARM is different from the setting path of the RTC current time. The ALEN field in the *[RTC_CTRL]* register also enables or disables the ALARM function.

The ADJUST function is not supported by the aram time setting.

The alarm time is available in both 12-hour and 24-hour modes. In the 12-hour mode, the setting of AM or PM is done by the bit 5 in the ***[RTC_ALARM2]*** register. For the setting of the PM, the bit 5, AL_HOURSET_OVER10[1], is set to 1.

If the ALARM_INTMASK bit in the ***[RTC_INTMASK]*** register is set to 1, the intrrupt signal (RTC_INT) becomes High when the RTC counter value matches the alarm time.

4.3. Periodic Interrupt Setting

The periodic interrupt can be set to the setting per second, per minute, per hour, per day, and per month.

Table 4.4 Periodic interrupt setting

[RTC_PERIODIC] Register Setting	Interrupt period
0b0000	No interrupt
0b0001	1 second (at the count up per second)
0b0010	1 mimute and 00 seconds
0b0011	1 hour, 00 minutes, and 00 seconds
0b0100	1 day, 00 hours, 00 minutes, and 00 seconds
0b0101	Every month at the first day, 00 o'clock AM, 00 minutes, and 00 seconds
0b0110	Setting is disabled.
0b0111	Setting is disabled.
0b1000	1/2 seconds
0b1001	1/4 seconds
0b1010	1/8 seconds
0b1011	1/16 seconds
0b1100	1/32 seconds
0b1101	1/64 seconds
0b1110	1/128 seconds
0b1111	1/256 seconds

The setting value (in Table 4.4) is set to the ***[RTC_PERIODIC]*** register. And, the PRRST bit in the ***[RTC_CTRL]*** register should be written to 1, then the value becomes valid in the RTC. The PREN field in the ***[RTC_CTRL]*** register enables or disables the periodic interrupt function.

If the PERIOD_INTMASK bit in the ***[RTC_INTMASK]*** register is set to 1, the intrrupt signal (RTC_INT) becomes High periodically.

4.4. Interval-timer Interrupt Setting

A 16-bit timer counter is implemented in this module in addition to the clock/calendar counter. It is used to generate an interval-timer interrupt. The source clock frequency is 2.048 kHz.

The interval time is set to the ***[RTC_INTVAL_SET]*** register. The time should be set as the source clock counts. The time interval of (the setting value + 1) cycles is the time period the interrupt is generated. The value 0x0001 to 0xFFFF can be set to the ***[RTC_INTVAL_SET]*** register. 0x0000 is inhibited. That is, the setting time available is in the range of 0.977 ms to 32 seconds and the resolution is 0.488 ms.

When the INTVALRST bit in the ***[RTC_CTRL]*** register is set to 1, the setting time is valid. Then, the INTVALEN bit in the ***[RTC_CTRL]*** register should be set to 1 to start the timer counter.

When the INTVALEN bit changes from 0 to 1, the value of the ***[RTC_INTVAL_SET]*** register is loaded to the 16-bit timer counter, and the counter starts to do down-counting. When the timer counter becomes 0, the interrupt is generated.

After the timer counter becomes 0, the counter loads the value of the ***[RTC_INTVAL_SET]*** register, again. If the continuous interrupt is not necessary, the INTVALEN bit should be set to 0 to stop the counter.

Current count value of timer counter can be read from ***[RTC_INTVAL_TMR]*** register. The interrupt is asserted at the next source clock (2.048 kHz) count when the timer counter becomes 0. Therefore, if you read ***[RTC_INTVAL_TMR]*** register just after the interrupt, re-loaded counter value (i.e. the same value as ***[RTC_INTVAL_SET]*** register) will be read.

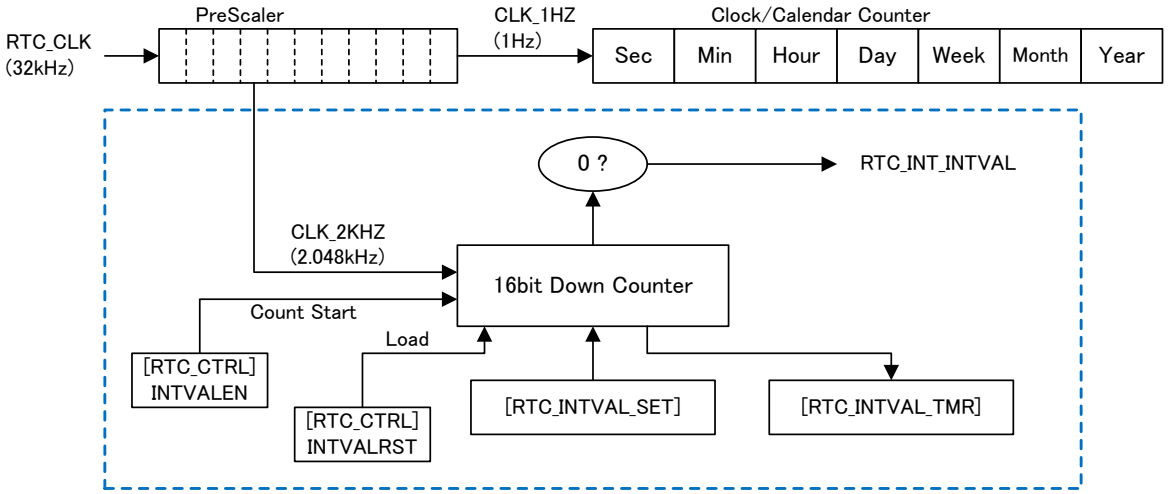

Figure 4.1 Block diagram of interval timer interrupt control

4.5. Interrupt Causes

Three interrupt causes are supported.

(1) Alarm interrupt (RTC_INT_ALARM)
The ALEN bit in the ***[RTC_CTRL]*** register is enabled, and when the timer count mathces the alarm setting value, the interrupt is generated. The interrupt signal is cleared by writing any value to the ***[RTC_ALARM_INTCLR]*** register.

(2) Periodic interrupt (RTC_INT_PERIOD)
The PREN bit in the ***[RTC_CTRL]*** register is enabled, and the periodic interrupt is generated depending on the setting value in the***[RTC_PERIODIC]*** register. The interrupt signal is cleared by writing any value to the ***[RTC_PERIOD_INTCLR]*** register.

If the interrupt is cleared within one second after its generation, the interrupt is generated again. To prevent this, the PREN bit in the ***[RTC_CTRL]*** register should be disabled after the interrupt is generated.

(3) Interval-timer interrupt (RTC_INT_INTVAL)
The INTVALEN bit in the ***[RTC_CTRL]*** register is enabled, and when the internal 16-bit timer counter becomes 0, the interval-timer interrupt is generated.

The interval time between the 0 to 1 change of the INTVALEN bit and the interrupt generation depends on the value set to the ***[RTC_INTVAL_SET]*** register. While the INTVALEN bit in the ***[RTC_CTRL]*** register is enabled, the 16-bit timer counter continues to operate. So, the interrupt can be generated periodically. To stop the interrupt, the INTVALEN bit in the ***[RTC_CTRL]*** register should be disabled and, then, the ***[RTC_INTVAL_INTCLR]*** register should be written to any data.
The intrerrupt signal, RTC_INT, is a common output pin for the alarm interrupt, the periodic interrupt, and the interval-timer interrupt. This signal becomes High when one of the interrupts is generated.

When an interrupt is detected, the ***[RTC_RIS]*** or ***[RTC_MIS]*** register should be read to identify the interrupt.

4.6. Other Setting Items (Leap year, 12-hour or 24-hour, Timer Stop)

- Leap year setting
  The RTC caluculates to find a leap year count using the register setting value. The setting value increments in the leap year counter which synchronizes the year counter. The value changes from 00 to 01, 10, and 11. The "00" is defined as a leap year count.
  The leap year counter can be monitored with ***[RTC_CURRENT4]*** register.

- 12-hour or 24-hour mode
  The HR24_12 field in the ***[RTC_CTRL]*** register sets the 12-hour mode or 24-hour mode of the RTC.
  - 1: 24-hour mode
  - 0: 12-hour mode

  In the 12-hour mode, the upper bit of HOURSET_OVER10 field (bit 5) in the ***[RTC_TIMESET2]*** register is used as the flag of AM or PM.
  - AM: 0
  - PM: 1

  Table 4.5 shows the setting values in the HOURSET_OVER10 field and HOURSET_UNDER10 field in ***[RTC_TIMESET2]*** in either 24-hour or 12-hour mode.

Table 4.5 *[RTC_TIMESET2]*.HOURSET_OVER10 and HOURSET_UNDER10 settings

24-hour mode	HOURSET_ OVER10 setting value	HOURSET_ UNDER10 setting value	12-hour mode	HOURSET_ OVER10 setting value	HOURSET_ UNDER10 setting value
0 o'clock	0	0	AM 0 o'clock	0	0
1 o'clock	0	1	AM 1 o'clock	0	1
2 o'clock	0	2	AM 2 o'clock	0	2
3 o'clock	0	3	AM 3 o'clock	0	3
4 o'clock	0	4	AM 4 o'clock	0	4
5 o'clock	0	5	AM 5 o'clock	0	5
6 o'clock	0	6	AM 6 o'clock	0	6
7 o'clock	0	7	AM 7 o'clock	0	7
8 o'clock	0	8	AM 8 o'clock	0	8
9 o'clock	0	9	AM 9 o'clock	0	9
10 o'clock	1	0	AM 10 o'clock	1	0
11 o'clock	1	1	AM 11 o'clock	1	1
12 o'clock	1	2	PM 0 o'clock	2	0
13 o'clock	1	3	PM 1 o'clock	2	1
14 o'clock	1	4	PM 2 o'clock	2	2
15 o'clock	1	5	PM 3 o'clock	2	3
16 o'clock	1	6	PM 4 o'clock	2	4
17 o'clock	1	7	PM 5 o'clock	2	5
18 o'clock	1	8	PM 6 o'clock	2	6
19 o'clock	1	9	PM 7 o'clock	2	7
20 o'clock	2	0	PM 8 o'clock	2	8
21 o'clock	2	1	PM 9 o'clock	2	9
22 o'clock	2	2	PM 10 o'clock	3	0
23 o'clock	2	3	PM 11 o'clock	3	1

The setting values in the table are the same as in the HOURVAL_OVER10 field and the HOURVAL_UNDER10 field in the ***[RTC_CURRENT2]*** register, and in the AL_HOURSET_OVER10 field and the AL_HOURSET_UNDER10 field in the ***[RTC_ALARM2]*** register.

If the alarm function is enabled and the setting of the HR24_12 bit is changed, the AL_HOURSET_OVER10 field and the AL_HOURSET_UNDER10 field in the ***[RTC_ALARM2].***

register should be changed according to the hour mode change. It is recommended that when the HR24_12 bit is changed, the alarm function should be temporary disabled to prevent an alarm -function.

- Timer stop
  The TMEN field in the ***[RTC_CTRL]*** register stops the RTC counter.
  - 1: RTC counter operating
  - 0: RTC counter stop

The counter stops and the RTC is updated if both the TMRST field and the TMEN field are written to 1.

4.7. Supplementary Note

Timer data read
When the timer data is read, the counter value and the read data may be different from the counter value by 1 second maximum. This occurs by the following reason. The bus clock and the RTC clock are different. And it takes some time to read all the timer data registers. So, while reading, the RTC counter may be updated. It is recommended that the read should be done twice and the two data should be checked they are identical.

Current counter value (Read data from ***[RTC_CURRENT]*** register)
The RTC block operates with the PCLK (1 MHz to 12 MHz variable in this product) and the RTC_CLK (32 kHz). The read and write for the RTC register synchronize the former clock, and the RTC internal counter synchronizes the latter clock.
When the internal counter loads the values of the ***[RTC_TIMSETn]*** (n=1,2,3,and 4) registers, a time lag of several RTC_CLK cycles may be generated because the data is transferred across the different clock domains.

After the ***[RTC_TIMESETn]*** (n=1,2,3,and 4) registers are written, the TMRST bit in the ***[RTC_CTRL]*** register should be set to 1to load the setting value. Otherwise, the setting data are not transferred to the counter. To confirm the load, the TMRST field in the ***[RTC_CTRL]*** register should be checked.

4.8. Power Management

4.8.1. Power Mode

- Active/Sleep0/Sleep1/Sleep2/ Wait/Wait-Retention/Retention: Operates.
- RTC: The counter always operates. Some registers lose their data in some power modes. Those registers should be set after returning to the appropriate power mode.
- STOP mode: 32 kHz clock stops. So, the counter also stops. Some registers lose their data in this mode. Those registers should be set after returning to the appropriate power mode. The following describes about the registers which should be set after returning from the RTC/STOP mode.

4.8.2. Operation in each power mode and reset

The operation of the register part (rtclv) in a power mode is different from that of the counter circuits (rtchv) in this product.

The counter circuits (rtchv) are always supplied with power and the timer operation continues with the RTC clock. In this product, 32 kHz oscillator can operates in all power modes except the STOP mode. The 32 kHz clock is not supplied with the RTC after the power-on. Software should set to the clock enable and the clock is supplied to the RTC. For detail, refer to the PMU (Power Management Unit) specification.

The power of the register part (rtclv) can be shut-downed in specific power modes. In the RTC mode and the STOP mode, the power is shut-downed in this product. Even in the RTC mode or the STOP mode, the counter (rtchv) operates normally because the necessary information of the registers are stored in the counter circuits.

The cause of the reset of the register part (rtclv) is different from that of the counter circuits (rtchv). The reset to the counter circuits (rtchv) is asserted by the BrownOut reset which is generated when the power supply is dropped. (At the power-on, the same reset is asserted.) On the other hand, the reset to the register part (rtclv) is asserted by the BrownOut, the input from the SYS_RESET pin, software reset by the CPU, and others.

The reset state of the registers is different according to the reset cause. The reset cause is listed in Table 4.6. The register which is indicated in "initialized" column in the table should be set after the reset deassertion. The register indicated in "uninitialized" column stores the data before the reset assertion, and the setting is unnecessary.

Table 4.6 Register initial state

Register/Bit		BrownOut Reset	SYS_RESET Pin Assertion WDT Reset CPU LOCKUP Reset CPU SYSRESETREQ
[RTC_TIMESET1]		Initialized	Initialized
[RTC_TIMESET2]		Initialized	Initialized
[RTC_TIMESET3]		Initialized	Initialized
[RTC_TIMESET4]		Initialized	Initialized
[RTC_CTRL]	INTVALEN	Initialized	Uninitialized
	INTVALRST	Initialized	Initialized
	TMEN	Initialized	Uninitialized
	TMRST	Initialized	Initialized
	HR24_12	Initialized	Uninitialized
	ADJUST	Initialized	Initialized
	PREN	Initialized	Uninitialized
	PRRST	Initialized	Initialized
	ALEN	Initialized	Uninitialized
	ALRST	Initialized	Initialized
[RTC_RIS]		Initialized	Uninitialized
[RTC_MIS]		Initialized	Initialized
[RTC_INTMASK]		Initialized	Initialized
[RTC_PERIODIC]		Initialized	Uninitialized
[RTC_PERIOD_INTCLR]		Initialized	Initialized
[RTC_ALARM_INTCLR]		Initialized	Initialized
[RTC_INTVALINTCLR]		Initialized	Initialized
[RTC_CURRENT1]		Initialized	Uninitialized
[RTC_CURRENT2]		Initialized	Uninitialized
[RTC_CURRENT3]		Initialized	Uninitialized
[RTC_CURRENT4]		Initialized	Uninitialized
[RTC_ALARM1]		Initialized	Uninitialized
[RTC_ALARM2]		Initialized	Uninitialized
[RTC_ALARM3]		Initialized	Uninitialized

4.9. Start-up and Stop Procedure

4.9.1. Start-up sequence

The start-up procedure is as follows.

For the PMU (Power Management Unit) register specification, refer to the PMU specification.

(1) Set *[CONFIG_OSC32K]*.OSC32K_EN to 1. This enables the OSC32K oscillation. Then, wait until the 32 kHz clock becomes stable (1 s maximum).
(2) Set *[CG_OFF_POWERDOMAIN]*.CG_PM to 1. This starts the clock supply to the PM domain.
(3) Set *[SRST_OFF_POWERDOMAIN]*.SRST_PM to 1. This deasserts the reset to the PM domain.
(4) Set *[CG_OFF_PA]*. G_rtcclk_rtchv_rtcclk to 1. This starts the clock supply to the rtchv.
(5) Set *[SRST_OFF_PA]*.SRST_asyncrst_rtchv_rtcrstn to 1. This deasserts the reset to the rtchv.
(6) Set *[CG_OFF_PM_1]*.CG_ppier0clk_rtclv_pclk to 1. This starts the clock supply to the rtclv.
(7) Set *[SRST_OFF_PM_1]*.SRST_asyncrst_rtclv_prstn and *[SRST_OFF_PM_1]*.SRST_asyncrst_rtclv_busrstn to 1. This deasserts the reset to the rtclv.
(8) After procedure of (5), wait 125 µs. (While waitng, you can proceed (6),(7).)
After procedure of (7), wait times below according to the setting of ***[RTCLV_RSYNC_SETTING]*** register.
0b00: 5 µs
0b01: 70 µs
0b10: 100 µs

Then, the RTC counter starts to operate. The initial value in the counter is 00 year, 0 month, 0 days, 00 o'clock, 00 minutes, and 00 seconds. For the time or calendar setting, refer to Section 4.1.

The resets other than the power-on and BrownOut are not effective to the rtchv. So, (4) and (5) in the start-up sequence are unnecessary.

4.9.2. Stop sequence

The stop sequence is as follows.

In the case that only the counter stops temporarily:

(1) clear *[RTC_CTRL]*.TMEN to 0. This disables the counter.

If the TMEN bit in the *[RTC_CTRL]* register is cleared to 0, the counter stops. The time value is saved. The time count can continue to operate if the TMEN bit is set to 1.

In the case that the controller is not used (the whole system stops):

(1) set *[SRST_ON_PM_1]*.SRST_asyncrst_rtclv_prstn and *[SRST_ON_PM_1]*.SRST_asyncrst_rtclv_busrstn to 1. This asserts the reset to the rtclv.
(2) set *[CG_ON_PM_1]*.CG_ppier0clk_rtclv_pclk and *[CG_ON_PM_1]*.CG_rtcclk_rtclv_busclk to 1. This stops the clock supply to the rtclv.

4.9.3. Clock Setting

Clock frequency can be selected through PMU registers.

Bus clock (APB clock) is the clock divided by the value set in *[PRESCAL_MAIN]*.PSSEL_CD_PPIER0, and the source clock selected by *[CSM_MAIN]*.CSMSEL_MAIN. Do not set the frequency less than that of RTC clock (32 kHz) for the bus clock. Change the frequency of the bus clock when *[RTC_CTRL_WIP]* register is "0."

Source of RTC clock can be OSC32K or SIOSC32K, both of them are 32 kHz clock. This selection is done by *[CSM_RTC]*.CSMSEL_RTC.

When you change the RTC clock source, RTC operation should be disabled (see the procedure of "In the case that the controller is not used" in 5.9.2).

5. Precaution for Usage

5.1. Access Restriction Associated with Register Access

The registers in this module are mapped to a 4 KB space. The registers are allocated to 256-byte space starting at 0x0, with 32-bit interval in the little endian format. For the access to unallocated space, a write is ignored and a read returns data 0.

The 32-bit access to a register is recommended. If the 8-bit access is done, the bits other than the access byte are written to unexpected data. A register read has no problems in the 32-bit access. The target 8 bits can be read correctly. The 16-bit access can be done successfully because the higher 16 bits in all registers are reserved in this product.

When the following bits of ***[RTC_CTRL]*** register is set to "1,"

INTVALRST (bit8)
TMRST (bit6)
ADJUST (bit4)
PRRST (bit2)
ALRST (bit0)

if you access these bits successively, the interval time should be more than 12 cycles in RTC clock cycle (32 kHz). Alternatively you can read these bits and confirm they are not "1," then you can set "1" to these bits.

When the following bits of ***[RTC_CTRL]*** register written successively,

INTVALEN (bit9)
TMEN (bit7)
HR24_12 (bit5)
PREN (bit3)
ALEN (bit1)

if you write different value (i.e. write "1" after write "0," or write "0" after write "1"), the interval time should be more than 4 cycles in RTC clock cycle (32 kHz). Alternatively you can read ***[RTC_CTRL_WIP]*** register and confirm corresponding bits are not "1," then you can write to these bits.

Before the ***[RTC_TIMESETn]*** (n=1,2,3,4), ***[RTC_ALARMn]*** (n=1,2,3), ***[RTC_PERIODIC]***, and ***[RTC_INTVAL_SET]*** registers are written, the TMRST, ALRST, PRRST, and INTVALRST bits in the ***[RTC_CTRL]*** register should be checked to be 0, respectively.

5.2. Precaution for Changing Power Mode

When transiting to Wait, Wait-Retention, Retention, RTC, or STOP mode, change the power mode when ***[RTC_CTRL_WIP]*** register is "0."

When you are going to do one of the following;

- change the power-mode to Sleep2, Wait, Wait-Retention, Retention, RTC, or STOP
- change the power-mode to Sleep0 or Sleep1 with ***[POWERMODE_SLEEP_CG_ON]***.SLEEP_CG_ON_rtclv set to "1"

you should proceed the sequence below before changing the power-mode.

(1) Write to ***[RTC_CTRL]*** register.
You can write to Bit-band alias region of ***[RTC_CTRL]*** register's reserved bit (ex. 0x4280_027C) instead if the value of ***[RTC_CTRL]*** register need not to be changed.

(2) Wait until ***[RTC_CTRL_WIP]*** becomes to 0x0.

5.3. Precaution for ADJUST operation

When ADJUST function (described in 5.1.(3)) is done, the pre-scaler (which counts 1 second using 32 kHz clock) is reset to zero value.

This can cause the second counter to become count of 00 according to the counting value of the pre-scaler at the time ADJUST has been made.

- When the counting value < 0.5 s: the second counter becomes 00
- When the counting value >= 0.5 s: the second counter becomes 01

In order to set the second counter 00 definitely, take the following procedure.

(1) Set ***[RTC_CTRL]***.ADJUST=1
(2) Wait ***[RTC_CTRL]***.ADJUST=0
(3) Set ***[RTC_CTRL]***.ADJUST=1
(4) Wait ***[RTC_CTRL]***.ADJUST=0

Similarly, this pre-scaler reset may induce occurrence of the following interrupt (if the condition is met).

- Periodic Interrupt (when every second or every $1/2^n$ second is selected)
- Interval-timer Interrupt

In addition, ADJUST make the minute counter to be incremented if the second counter is less than 30, which may induce occurrence of the following interrupt (if the condition is met).

- Periodic Interrupt (when every minute is selected)
- Alarm Interrupt

If the occurrence of these interrupts are undesirable, disable all the interrupt when ADJUST operation is done.

6. Details of Registers

6.1. RTC_TIMESET1

RTC_TIMESET1			
Description	Time Set 1 Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0000		
Physical address View0	0x4004 0000		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:15	Reserved	-	-	-
14:12	MINSET_OVER10	Sets the minute counter (ten's place). The settable values are 0 to 5 (0b000 to 0b101). No other values can be set (writing disable).	RW modify	0x0
11:8	MINSET_UNDER10	Sets the minute counter (one's place). The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0
7	Reserved	-	-	-
6:4	SECSET_OVER10	Sets the second counter (ten's place). The settable values are 0 to 5 (0b000 to 0b101). No other values can be set (writing disable).	RW modify	0x0
3:0	SECSET_UNDER10	Sets the second counter (one's place). The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0

6.2. RTC_TIMESET2

RTC_TIMESET2			
Description	Time Set 2 Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0004		
Physical address View0	0x4004 0004		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:14	Reserved	-	-	-
13:12	DAYSET_OVER10	Sets the day counter (ten's place). The settable values are 0 to 3 (0b00 to 0b11).	RW modify	0x0
11:8	DAYSET_UNDER10	Sets the day counter (one's place). The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0
7:6	Reserved	-	-	-

5:4	HOURSET_OVER10	Sets the hour counter (ten's place). The settable values are 0 to 3 (0b00 to 0b11). (Note) In 12-hour mode, bit5 is used to differentiate AM/PM. So, to set the time in the afternoon, set bit5 to "1".	RW modify	0x0
3:0	HOURSET_UNDER10	Sets the hour counter (one's place). The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0

6.3. RTC_TIMESET3

RTC_TIMESET3			
Description	Time Set 3 Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0008		
Physical address View0	0x4004 0008		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:16	Reserved	-	-	-
15:12	YEARSET_OVER10	Sets the year counter (ten's place). The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0
11:8	YEARSET_UNDER10	Sets the year counter (one's place). The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0
7:5	WEEKSET	Sets the day-of-week counter. 000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (unsettable (writing disable))	RW modify	0x0
4	MONTHSET_OVER10	Sets the month counter (ten's place). The settable values are 0 and 1 (0b0 to 0b1).	RW modify	0
3:0	MONTHSET_UNDER10	Sets the month counter (one's place). The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0

6.4. RTC_TIMESET4

RTC_TIMESET4			
Description	Time Set 4 Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 000C		
Physical address View0	0x4004 000C		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:2	Reserved	-	-	-
1:0	LEAP	Sets the leap year ("0": leap year) The settable values are 0 to 3 (0b00 to 0b11).	RW modify	0x0

6.5. RTC_CTRL

RTC_CTRL			
Description	Control Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0010		
Physical address View0	0x4004 0010		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:10	Reserved	-	-	-
9	INTVALEN	Interval timer enable 1: Enable 0: Disable	RW modify	0
8	INTVALRST	Loads the interval timer setting. After setting data loading is complete, 0 is automatically set. At writing: 1: Starts loading of the setting data. 0: No operation At reading: 1: Loading of the setting data is in progress. 0: Loading of the setting data is complete.	RW modify	0
7	TMEN	RTC operation enable 1: Enable 0: Disable	RW modify	1
6	TMRST	Loading the time counter with TIMESET1 to 4 register settings After loading of setting data to the time counter is complete, 0 is automatically set. At writing: 1: Starts loading of setting data. 0: No operation At reading: 1: Loading of the setting data is in progress. 0: Loading of setting data is complete.	RW modify	0

5	HR24_12	24/12Hr selection 1: 24-hour notation 0: 12-hour notation	RW modify	0
4	ADJUST	Time count adjustment After adjustment is complete, 0 is automatically set. At writing: 1: Starts adjustment. 0: No operation At reading: 1: Adjustment in progress 0: Adjustment completed	RW modify	0
3	PREN	Periodic interrupt function enable 1: Enable 0: Disable	RW modify	0
2	PRRST	Loads the periodic interrupt setting. After loading of setting data is complete, 0 is automatically set. At writing: 1: Starts loading of the setting data. 0: No operation At reading: 1: Loading of the setting data is in progress. 0: Loading of the setting data is complete.	RW modify	0
1	ALEN	Alarm function enable 1: Enable 0: Disable	RW modify	0
0	ALRST	Loads the alarm setting. After loading of setting data is complete, 0 is automatically set. At writing: 1: Starts loading of the setting data. 0: No operation At reading: 1: Loading of the setting data is in progress. 0: Loading of the setting data is complete.	RW modify	0

6.6. RTC_STATUS

RTC_STATUS			
Description			
Address Region	rtclv	**Type:**	RO
Offset	0x0000 0014		
Physical address View0	0x4004 0014		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:1	Reserved	-	-	-
0	RTC_READY	If the RTC counter value becomes ready to be read, 1 is automatically set.	RO	0

6.7. RTC_CTRL_WIP

RTC_CTRL_WIP			
Description	Control Register write access status		
Address Region	rtclv	**Type:**	RO
Offset	0x0000 0018		
Physical address View0	0x4004 0018		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:10	Reserved	-	-	-
9	INTVALEN	[RTC_CTRL].INTVALEN bit status of write access 0: writable 1: not writable	RO	0
8	INTVALRST	[RTC_CTRL].INTVALRST bit status of write access 0: writable 1: not writable	RO	0
7	TMEN	[RTC_CTRL].TMEN bit status of write access 0: writable 1: not writable	RO	0
6	TMRST	[RTC_CTRL].TMRST bit status of write access 0: writable 1: not writable	RO	0
5	HR24_12	[RTC_CTRL].HR24_12 bit status of write access 0: writable 1: not writable	RO	0
4	ADJUST	[RTC_CTRL].ADJUST bit status of write access 0: writable 1: not writable	RO	0
3	PREN	[RTC_CTRL].PREN bit status of write access 0: writable 1: not writable	RO	0
2	PRRST	[RTC_CTRL].PRRST bit status of write access 0: writable 1: not writable	RO	0
1	ALEN	[RTC_CTRL].ALEN bit status of write access 0: writable 1: not writable	RO	0
0	ALRST	[RTC_CTRL].ALRST bit status of write access 0: writable 1: not writable	RO	0

6.8. RTC_RIS

RTC_RIS			
Description	Raw Interrupt Status Register		
Address Region	rtclv	**Type:**	RO
Offset	0x0000 0020		
Physical address View0	0x4004 0020		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:3	Reserved	-	-	-
2	INTVAL_INTRAWST	Interval-timer interrupt status 1: Interrupt request available 0: No interrupt request	RO	0
1	PERIOD_INTRAWST	Periodic interrupt status 1: Interrupt request available 0: No interrupt request	RO	0
0	ALARM_INTRAWST	Alarm interrupt status 1: Interrupt request available 0: No interrupt request	RO	0

6.9. RTC_MIS

RTC_MIS			
Description	Masked Interrupt Status Register		
Address Region	rtclv	**Type:**	RO
Offset	0x0000 0024		
Physical address View0	0x4004 0024		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:3	Reserved	-	-	-
2	INTVAL_INTMASKST	Interval-timer interrupt status 1: Interrupt request available 0: No interrupt request Not set when INTVAL_INTMASK=0.	RO	0
1	PERIOD_INTMASKST	Periodic interrupt status 1: Interrupt request available 0: No interrupt request Not set when PERIOD_INTMASK=0.	RO	0
0	ALARM_INTMASKST	Alarm interrupt status 1: Interrupt request available 0: No interrupt request Not set when ALARM_INTMASK=0.	RO	0

6.10. RTC_INTMASK

RTC_INTMASK			
Description	Interrupt Mask Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0028		
Physical address View0	0x4004 0028		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:3	Reserved	-	-	-
2	INTVAL_INTMASK	Interval-timer interrupt 1: No interrupt mask (interrupt enable) 0: Interrupt mask (interrupt disable)	RW modify	0
1	PERIOD_INTMASK	Periodic interrupt 1: No interrupt mask (interrupt enable) 0: Interrupt mask (interrupt disable)	RW modify	0
0	ALARM_INTMASK	Alarm interrupt 1: No interrupt mask (interrupt enable) 0: Interrupt mask (interrupt disable)	RW modify	0

6.11. RTC_PERIODIC

RTC_PERIODIC			
Description	Periodic Interrupt Set Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 002C		
Physical address View0	0x4004 002C		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:4	Reserved	-	-	-
3:0	PERIODIC_INTSET	Sets the periodic interrupt (every second, every minute, every hour, every day, every month) 0000: No periodic interrupt occurrence 0001: Every second (at the same time with second counting-up) 0010: 00 second every minute 0011: 00 minute 00 second every hour 0100: 00 hour 00 minute 00 second every day 0101: 00 hour 00 minute 00 second every month 0110: Reserved (unsettable (writing disable)) 0111: Reserved (unsettable (writing disable)) 1000: Every 1/2 second 1001: Every 1/4 second 1010: Every 1/8 second 1011: Every 1/16 second 1100: Every 1/32 second	RW modify	0x0

		1101: Every 1/64 second 1110: Every 1/128 second 1111: Every 1/256 second		

6.12. RTC_PERIOD_INTCLR

RTC_PERIOD_INTCLR			
Description	Periodic Interrupt Clear		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0030		
Physical address View0	0x4004 0030		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:0	PERIOD_INTCLR	Clears the periodic interrupt. If a value is written to this register, the periodic interrupt is cleared, regardless of the value to be written.	RW clear	-

6.13. RTC_ALARM_INTCLR

RTC_ALARM_INTCLR			
Description	Alarm Interrupt Clear		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0034		
Physical address View0	0x4004 0034		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:0	ALARM_INTCLR	Clears the alarm interrupt. If a value is written to this register, the alarm interrupt is cleared, regardless of the value to be written.	RW clear	-

6.14. RTC_INTVAL_INTCLR

RTC_INTVAL_INTCLR			
Description	Interval Interrupt Clear		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0038		
Physical address View0	0x4004 0038		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:0	INTVAL_INTCLR	Clears the interval-timer interrupt. If a value is written to this register, the interval-timer interrupt is cleared, regardless of the value to be written.	RW clear	-

6.15. RTC_CURRENT1

RTC_CURRENT1			
Description	Current Time 1 Register		
Address Region	rtclv	**Type:**	RO
Offset	0x0000 0040		
Physical address View0	0x4004 0040		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:15	Reserved	-	-	-
14:12	MINVAL_OVER10	Current value of the minute counter (ten's place)	RO	0x0
11:8	MINVAL_UNDER10	Current value of the minute counter (one's place)	RO	0x0
7	Reserved	-	-	-
6:4	SECVAL_OVER10	Current value of the second counter (ten's place)	RO	0x0
3:0	SECVAL_UNDER10	Current value of the second counter (one's place)	RO	0x0

6.16. RTC_CURRENT2

RTC_CURRENT2			
Description	Current Time 2 Register		
Address Region	rtclv	**Type:**	RO
Offset	0x0000 0044		
Physical address View0	0x4004 0044		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:14	Reserved	-	-	-
13:12	DAYVAL_OVER10	Current value of the day counter (ten's place)	RO	0x0
11:8	DAYVAL_UNDER10	Current value of the day counter (one's place)	RO	0x0
7:6	Reserved	-	-	-
5:4	HOURVAL_OVER10	Current value of the hour counter (ten's place)	RO	0x0
3:0	HOURVAL_UNDER10	Current value of the hour counter (one's place)	RO	0x0

6.17. RTC_CURRENT3

RTC_CURRENT3				
Description	Current Time 3 Register			
Address Region	rtclv	**Type:**	RO	
Offset	0x0000 0048			
Physical address View0	0x4004 0048			
Physical address View1	-			
Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:16	Reserved	-	-	-
15:12	YEARVAL_OVER10	Current value of the year counter (ten's place)	RO	0x0
11:8	YEARVAL_UNDER10	Current value of the year counter (one's place)	RO	0x0
7:5	WEEKVAL	Current value of the day-of-week counter	RO	0x0
4	MONTHVAL_OVER10	Current value of the month counter (ten's place)	RO	0
3:0	MONTHVAL_UNDER10	Current value of the month counter (one's place)	RO	0x0

6.18. RTC_CURRENT4

RTC_CURRENT4				
Description	Current Time 4 Register			
Address Region	rtclv	**Type:**	RO	
Offset	0x0000 004C			
Physical address View0	0x4004 004C			
Physical address View1	-			
Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:2	Reserved	-	-	-
1:0	LEAP	Current value of the leap year counter	RO	0x0

6.19. RTC_ALARM1

RTC_ALARM1				
Description	Alarm Set 1 Register			
Address Region	rtclv	**Type:**	RW	
Offset	0x0000 0050			
Physical address View0	0x4004 0050			
Physical address View1	-			
Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:15	Reserved	-	-	-
14:12	AL_MINSET_OVER10	Sets the minute counter (ten's place) for alarm. The settable values are 0 to 5 ((0b000 to 0b101).	RW modify	0x0

		No other values can be set (writing disable).		
11:8	AL_MINSET_UNDER10	Sets the minute counter (one's place) for alarm. The settable values are 0 to 9 ((0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0
7:0	Reserved	-	-	-

6.20. RTC_ALARM2

RTC_ALARM2			
Description	Alarm Set 2 Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0054		
Physical address View0	0x4004 0054		
Physical address View1	-		

Bitfield Details

Bits	Name	Description	Access	Reset
31:14	Reserved	-	-	-
13:12	AL_DAYSET_OVER10	Sets the day counter (ten's place) for alarm. The settable values are 0 to 3 (0b00 to 0b11).	RW modify	0x0
11:8	AL_DAYSET_UNDER10	Sets the minute counter (one's place) for alarm. The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0
7:6	Reserved	-	-	-
5:4	AL_HOURSET_OVER10	Sets the hour counter (ten's place) for alarm. The settable values are 0 to 3 (0b00 to 0b11). (Note) In 12-hour mode, bit5 is used to differentiate AM/PM. So, to set the time in the afternoon, set bit5 to "1".	RW modify	0x0
3:0	AL_HOURSET_UNDER10	Sets the hour counter (one's place) for alarm. The settable values are 0 to 9 (0b0000 to 0b1001). No other values can be set (writing disable).	RW modify	0x0

6.21. RTC_ALARM3

RTC_ALARM3			
Description	Alarm Set 3 Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0058		
Physical address View0	0x4004 0058		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:8	Reserved	-	-	-
7:5	AL_WEEKSET	Sets the day-of-week counter for alarm. 000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved	RW modify	0x0
4:0	Reserved	-	-	-

6.22. RTC_INTVAL_SET

RTC_INTVAL_SET			
Description	Interval Timer Set Register		
Address Region	rtclv	**Type:**	RW
Offset	0x0000 0060		
Physical address View0	0x4004 0060		
Physical address View1	-		

Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:16	Reserved	-	-	-
15:0	INTVAL_SET	Sets the interval-timer interrupt generation time. This value is set with the number of cycles of the source clock (2.048 kHz) of the timer. The time of (setting value + 1) cycles is the period that continues until interrupt generation. Setting of 0x0000 is prohibited.	RW modify	0xFFFF

6.23. RTC_INTVAL_TMR

RTC_INTVAL_TMR				
Description				
Address Region	rtclv	**Type:**	RO	
Offset	0x0000 0064			
Physical address View0	0x4004 0064			
Physical address View1	-			
Bitfield Details				
Bits	**Name**	**Description**	**Access**	**Reset**
31:16	Reserved	-	-	-
15:0	COUNT_VAL	Current value of interval-timer counter can be read.	RO	0xFFFF

7. Revision History

Table 7.1 Revision History

Revision	Date	Description
0.1	2014-03-14	Newly released
0.2	2014-03-18	Changed expression of periodic interrupt and interval-timer interrupt.
0.3	2014-06-17	Changed some description as ES2 specification change.
0.4	2014-10-07	Deleted description about ***[CG_ON_PM_1]***.register. Added description about wait times in start-up procedure. Added restriction for changing bus clock frequency. Added precaution when entering PowerMode.
0.5	2014-11-25	Added description about RTC clock change. Added note about the procedure after RTC interrupt clear.
1.0	2015-01-22	Official version.
1.1	2015-12-28	Modified ***[RTC_CTRL]*** setting value in ADJUST operation Added section 5.3
1.2	2018-02-06	Changed header, footer and the last page. Changed corporate name and descriptions. Modified Arm logo and descriptions.

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, 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, devices related to electric power, and equipment used in finance-related fields. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative.
- 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.**