Chapter 1
SLAU403F–August 2012–Revised March 2018

Real-Time Clock B (RTC_B)

NOTE: This chapter is an excerpt from the *MSP430x5xx and MSP430x6xx Family User's Guide*. The latest version of the full user's guide is available from http://www.ti.com/lit/pdf/slau208.

The real-time clock RTC_B module provides clock counters with calendar mode, a flexible programmable alarm, and calibration. Note that the RTC_B supports only calendar mode and not counter mode. The RTC_B also support operation in LPM3.5 and device-dependent operation from a backup supply. See the device-specific data sheet for the supported features. This chapter describes the RTC_B module.

Topic **Page**

1.1 Real-Time Clock RTC_B Introduction

The RTC_B module provides configurable clock counters.

RTC_B features include:

- Real-time clock and calendar mode providing seconds, minutes, hours, day of week, day of month, month, and year (including leap year correction)

  Note that only the calendar mode is supported by RTC_B; the counter mode that is available in some other RTC modules is not supported.
- Interrupt capability
- Selectable BCD or binary format
- Programmable alarms
- Calibration logic for time offset correction
- Operation in LPM3.5
- Operation from backup supply with programmable charger for backup supply (device-dependent) (see the Battery Backup chapter)

The RTC_B block diagram for devices supporting LPM3.5 is shown in Figure 1-1.

NOTE: Real-time clock initialization

Most RTC_B module registers have no initial condition. These registers must be configured by user software before use.

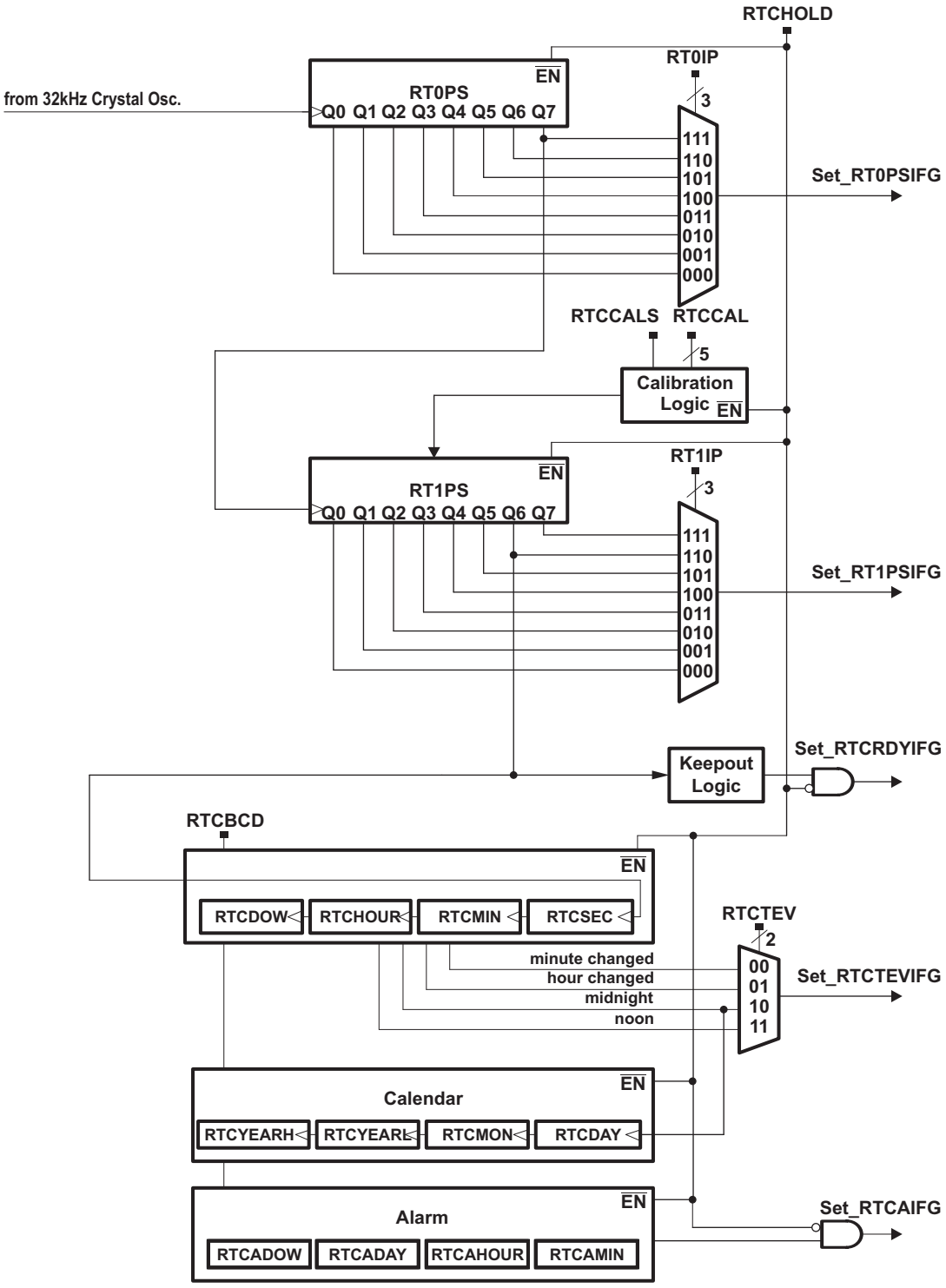

Figure 1-1. RTC_B Block Diagram

1.2 RTC_B Operation

The RTC_B module provides seconds, minutes, hours, day of week, day of month, month, and year in selectable BCD or hexadecimal format. The calendar includes a leap-year algorithm that considers all years evenly divisible by four as leap years. This algorithm is accurate from the year 1901 through 2099.

1.2.1 Real-Time Clock and Prescale Dividers

The prescale dividers, RT0PS and RT1PS, are automatically configured to provide a 1-s clock interval for the RTC_B. The low-frequency oscillator must be operated at 32768 Hz (nominal) for proper RTC_B operation. RT0PS is sourced from the low-frequency oscillator XT1. The output of RT0PS / 256 (Q7) is used to source RT1PS. RT1PS is further divider and the /128 output sources the real-time clock counter registers providing the required 1-second time interval.

When RTCBCD = 1, BCD format is selected for the calendar registers. It is possible to switch between BCD and hexadecimal format while the RTC is counting.

Setting RTCHOLD halts the real-time counters and prescale counters, RT0PS, and RT1PS.

1.2.2 Real-Time Clock Alarm Function

The RTC_B module provides for a flexible alarm system. There is a single user-programmable alarm that can be programmed based on the settings contained in the alarm registers for minutes, hours, day of week, and day of month.

Each alarm register contains an alarm enable (AE) bit that can be used to enable the respective alarm register. By setting AE bits of the various alarm registers, a variety of alarm events can be generated.

- Example 1: A user wishes to set an alarm every hour at 15 minutes past the hour (that is, at 00:15:00, 01:15:00, 02:15:00, etc). This is possible by setting RTCAMIN to 15. By setting the AE bit of the RTCAMIN and clearing all other AE bits of the alarm registers, the alarm is enabled. When enabled, the RTCAIFG is set when the count transitions from 00:14:59 to 00:15:00, 01:14:59 to 01:15:00, 02:14:59 to 02:15:00, and so on.
- Example 2: A user wishes to set an alarm every day at 04:00:00. This is possible by setting RTCAHOUR to 4. By setting the AE bit of the RTCHOUR and clearing all other AE bits of the alarm registers, the alarm is enabled. When enabled, the RTCAIFG is set when the count transitions from 03:59:59 to 04:00:00.
- Example 3: A user wishes to set an alarm for 06:30:00. RTCAHOUR would be set to 6 and RTCAMIN would be set to 30. By setting the AE bits of RTCAHOUR and RTCAMIN, the alarm is enabled. Once enabled, the RTCAIFG is set when the time count transitions from 06:29:59 to 06:30:00. In this case, the alarm event occurs every day at 06:30:00.
- Example 4: A user wishes to set an alarm every Tuesday at 06:30:00. RTCADOW would be set to 2, RTCAHOUR would be set to 6, and RTCAMIN would be set to 30. By setting the AE bits of RTCADOW, RTCAHOUR, and RTCAMIN, the alarm is enabled. Once enabled, the RTCAIFG is set when the time count transitions from 06:29:59 to 06:30:00 and the RTCDOW transitions from 1 to 2.
- Example 5: A user wishes to set an alarm the fifth day of each month at 06:30:00. RTCADAY would be set to 5, RTCAHOUR would be set to 6, and RTCAMIN would be set to 30. By setting the AE bits of RTCADAY, RTCAHOUR, and RTCAMIN, the alarm is enabled. Once enabled, the RTCAIFG is set when the time count transitions from 06:29:59 to 06:30:00 and the RTCDAY equals 5.

NOTE: Setting the alarm

Before setting an initial alarm, all alarm registers including the AE bits should be cleared.

To prevent potential erroneous alarm conditions from occurring, the alarms should be disabled by clearing the RTCAIE, RTCAIFG, and AE bits before writing initial or new time values to the RTC time registers.

NOTE: Invalid alarm settings

Invalid alarm settings are not checked by hardware. It is the user's responsibility that valid alarm settings are entered.

NOTE: Invalid time and date values

Writing of invalid date or time information or data values outside the legal ranges specified in the RTCSEC, RTCMIN, RTCHOUR, RTCDAY, RTCDOW, RTCYEAR, RTCAMIN, RTCAHOUR, RTCADAY, and RTCADOW registers can result in unpredictable behavior.

1.2.3 Reading or Writing Real-Time Clock Registers

Because the system clock may in fact be asynchronous to the RTC_B clock source, special care must be used when accessing the real-time clock registers.

The real-time clock registers are updated once per second. To prevent reading any real-time clock register at the time of an update, which could result in an invalid time being read, a keep-out window is provided. The keep-out window is centered approximately 128/32768 seconds around the update transition. The read-only RTCRDY bit is reset during the keep-out window period and set outside the keep-out the window period. Any read of the clock registers while RTCRDY is reset is considered to be potentially invalid, and the time read should be ignored.

An easy way to safely read the real-time clock registers is to utilize the RTCRDYIFG interrupt flag. Setting RTCRDYIE enables the RTCRDYIFG interrupt. Once enabled, an interrupt is generated based on the rising edge of the RTCRDY bit, causing the RTCRDYIFG to be set. At this point, the application has nearly a complete second to safely read any or all of the real-time clock registers. This synchronization process prevents reading the time value during transition. The RTCRDYIFG flag is reset automatically when the interrupt is serviced, or it can be reset with software.

NOTE: Reading or writing real-time clock registers

When the counter clock is asynchronous to the CPU clock, any read from any RTCSEC, RTCMIN, RTCHOUR, RTCDOW, RTCDAY, RTCMON, or RTCYEAR register while the RTCRDY is reset may result in invalid data being read. To safely read the counting registers, either polling of the RTCRDY bit or the synchronization procedure previously described can be used. Alternatively, the counter register can be read multiple times while operating, and a majority vote taken in software to determine the correct reading. Reading the RT0PS and RT1PS can only be handled by reading the registers multiple times and a majority vote taken in software to determine the correct reading.

Any write to any counting register takes effect immediately. However, the clock is stopped during the write. In addition, RT0PS and RT1PS registers are reset. This could result in losing up to 1 second during a write. Writing of data outside the legal ranges or invalid time stamp combinations results in unpredictable behavior.

1.2.4 Real-Time Clock Interrupts

Six sources for interrupts are available, namely RT0PSIFG, RT1PSIFG, RTCRDYIFG, RTCTEVIFG, RTCAIFG, and RTCOFIFG. These flags are prioritized and combined to source a single interrupt vector. The interrupt vector register (RTCIV) is used to determine which flag requested an interrupt.

The highest-priority enabled interrupt generates a number in the RTCIV register (see register description). This number can be evaluated or added to the program counter (PC) to automatically enter the appropriate software routine. Disabled RTC interrupts do not affect the RTCIV value.

Any access, read or write, of the RTCIV register automatically resets the highest-pending interrupt flag. If another interrupt flag is set, another interrupt is immediately generated after servicing the initial interrupt. In addition, all flags can be cleared by software.

The user-programmable alarm event sources the real-time clock interrupt, RTCAIFG. Setting RTCAIE enables the interrupt. In addition to the user-programmable alarm, the RTC_B module provides for an interval alarm that sources real-time clock interrupt, RTCTEVIFG. The interval alarm can be selected to cause an alarm event when RTCMIN changed or RTCHOUR changed, every day at midnight (00:00:00) or every day at noon (12:00:00). The event is selectable with the RTCTEV bits. Setting the RTCTEVIE bit enables the interrupt.

The RTCRDY bit sources the real-time clock interrupt, RTCRDYIFG, and is useful in synchronizing the read of time registers with the system clock. Setting the RTCRDYIE bit enables the interrupt.

RT0PSIFG can be used to generate interrupt intervals selectable by the RT0IP bits. RT0PS is sourced with low-frequency oscillator clock at 32768 Hz, so intervals of 16384 Hz, 8192 Hz, 4096 Hz, 2048 Hz, 1024 Hz, 512 Hz, 256 Hz, or 128 Hz are possible. Setting the RT0PSIE bit enables the interrupt.

RT1PSIFG can be used to generate interrupt intervals selectable by the RT1IP bits. RT1PS is sourced with the output of RT0PS, which is 128 Hz (32768/256 Hz). Therefore, intervals of 64 Hz, 32 Hz, 16 Hz, 8 Hz, 4 Hz, 2 Hz, 1 Hz, or 0.5 Hz are possible. Setting the RT1PSIE bit enables the interrupt.

NOTE: Changing RT0IP or RT1IP

Changing the settings of the interrupt interval bits RT0IP or RT1IP while the corresponding prescaler is running or is stopped in a non-zero state can result in setting the corresponding interrupt flags.

The RTCOFIFG bit flags a failure of the 32-kHz crystal oscillator. Its main purpose is to wake up the CPU from LPM3.5 if an oscillator failure occurs. On devices with a backup-supply subsystem, it also stores a failure event that occurred while the RTC was operating on the backup supply.

1.2.4.1 RTCIV Software Example

The following software example shows the recommended use of RTCIV and the handling overhead. The RTCIV value is added to the PC to automatically jump to the appropriate routine.

The numbers at the right margin show the necessary CPU cycles for each instruction. The software overhead for different interrupt sources includes interrupt latency and return-from-interrupt cycles, but not the task handling itself.

```
; Interrupt handler for RTC interrupt flags.

RTC_HND                       ; Interrupt latency         6
           ADD &RTCIV,PC      ; Add offset to Jump table  3
           RETI               ; Vector 0: No interrupt    5
           JMP RTCRDYIFG_HND  ; Vector 2: RTCRDYIFG       2
           JMP RTCTEVIFG_HND  ; Vector 4: RTCTEVIFG       2
           JMP RTCAIFG_HND    ; Vector 6: RTCAIFG         5
           JMP RT0PSIFG_HND   ; Vector 8: RT0PSIFG        5
           JMP RT1PSIFG_HND   ; Vector A: RT1PSIFG        5
           JMP RTCOFIFG_HND   ; Vector C: RTCOFIFG        5
           RETI               ; Vector E: Reserved        5

RTCRDYIFG_HND                 ; Vector 2: RTCRDYIFG Flag
           ...                ; Task starts here
           RETI               ; Back to main program      5

RTCTEVIFG_HND                 ; Vector 4: RTCTEVIFG Flag
           ...                ; Task starts here
           RETI               ; Back to main program      5

RTCAIFG_HND                   ; Vector 6: RTCAIFG Flag
           ...                ; Task starts here
           RETI               ; Back to main program      5
```

```
RT0PSIFG_HND                ; Vector 8: RT0PSIFG Flag
            ...             ; Task starts here
            RETI            ; Back to main program      5

RT1PSIFG_HND                ; Vector A: RT1PSIFG Flag
            ...             ; Task starts here
            RETI            ; Back to main program      5

RTCOFIFG_HND                ; Vector C: RTCOFIFG Flag
            ...             ; Task starts here
            RETI            ; Back to main program      5
```

1.2.5 Real-Time Clock Calibration

The RTC_B module has calibration logic that allows for adjusting the crystal frequency in approximately +4-ppm or –2-ppm steps, allowing for higher time keeping accuracy from standard crystals. The RTCCALx bits are used to adjust the frequency. When RTCCALS is set, each RTCCALx LSB causes a ≈ +4-ppm adjustment. When RTCCALS is cleared, each RTCCALx LSB causes a ≈ –2-ppm adjustment.

Calibration is accomplished by periodically adjusting the RT1PS counter based on the RTCCALS and RTCCALx settings. The RT0PS divides the nominal 37268-Hz low-frequency (LF) crystal clock input by 256. A 60-minute period has 32768 cycles/sec × 60 sec/min × 60 min = 117964800 cycles. Therefore, a –2-ppm reduction in frequency (down calibration) approximately equates to adding an additional 256 cycles every 117964800 cycles (256/117964800 = 2.17 ppm). This is accomplished by holding the RT1PS counter for one additional clock of the RT0PS output within a 60-minute period. Similarly, a +4-ppm increase in frequency (up calibration) approximately equates to removing 512 cycles every 117964800 cycle (512/117964800 = 4.34 ppm). This is accomplished by incrementing the RT1PS counter for two additional clocks of the RT0PS output within a 60-minute period. Each RTCCALx calibration bit causes either 256 LF crystal clock cycles to be added every 60 minutes or 512 LF crystal clock cycles to be subtracted every 60 minutes, giving a frequency adjustment of approximately -2 ppm or +4 ppm, respectively.

To calibrate the frequency, the RTCCLK output signal is available at a pin. RTCCALF bits can be used to select the frequency rate of the output signal, either no signal, 512 Hz, 256 Hz, or 1 Hz.

The basic flow to calibrate the frequency is as follows:

1. Configure the RTCCLK pin.
2. Measure the RTCCLK output signal with an appropriate resolution frequency counter ; that is, within the resolution required.
3. Compute the absolute error in ppm: Absolute error (ppm) = $|10^6 (f_{MEASURED} – f_{RTCCLK})/f_{RTCCLK}|$, where f_{RTCCLK} is the expected frequency of 512 Hz, 256 Hz, or 1 Hz.
4. Adjust the frequency by performing the following:
   a. If the frequency is too low, set RTCCALS = 1 and apply the appropriate RTCCALx bits, where RTCCALx = (Absolute Error) / 4.34 rounded to the nearest integer
   b. If the frequency is too high, clear RTCCALS = 0 and apply the appropriate RTCCALx bits, where RTCCALx = (Absolute Error) / 2.17 rounded to the nearest integer

For example, assume that RTCCLK is configured to output at a frequency of 512 Hz. The measured RTCCLK is 511.9658 Hz. This frequency error is approximately 66.8 ppm too low. To increase the frequency by 66.8 ppm, RTCCALS would be set, and RTCCALx would be set to 15 (66.8 / 4.34). Similarly, assume that the measured RTCCLK is 512.0125 Hz. The frequency error is approximately 24.4 ppm too high. To decrease the frequency by 24.4 ppm, RTCCALS would be cleared, and RTCCAL would be set to 11 (24.4 / 2.17).

The calibration corrects only initial offsets and does not adjust for temperature and aging effects. These effects can be handled by periodically measuring temperature and using the crystal's characteristic curve to adjust the ppm based on temperature, as required.

NOTE: Minimum Possible Calibration

The minimal calibration possible is -4 ppm or +8 ppm. For example, setting RTCCALS = 0 and RTCCAL = 0h would result in a -4 ppm decrease in frequency. Similarly, setting RTCCALS = 1 and RTCCAL = 0h would result in a +8 ppm increase in frequency.

NOTE: Calibration output frequency

The 512-Hz and 256-Hz output frequencies observed at the RTCCLK pin are not affected by changes in the calibration settings, because these output frequencies are generated before the calibration logic. The 1-Hz output frequency is affected by changes in the calibration settings. Because the frequency change is small and infrequent over a very long time interval, it can be difficult to observe.

1.2.6 Real-Time Clock Operation in LPM3.5 Low-Power Mode

The regulator of the Power Management Module (PMM) is disabled upon entering LPM3.5, which causes most of the RTC_B configuration registers to be lost; only the counters and the backup RAM are retained. Table 1-1 lists the retained registers in LPM3.5. Also the configuration of the interrupts is stored so that the configured interrupts can cause a wakeup upon exit from LPM3.5. Interrupt flags that are set before entering LPM3.5 are cleared upon entering LPM3.5 (Note: this can only happen if the corresponding interrupt is not enabled). The interrupt flags RTCTEVIFG, RTCAIFG, RT1PSIFG, and RTCOFIFG can be used as RTC_B wake-up interrupt sources. After restoring the configuration registers (and clearing LOCKLPM5) the interrupts can be serviced as usual. The detailed flow is as follows:

1. Set all I/Os to general purpose I/Os and configure as needed. Optionally configure input interrupt pins for wake-up. Configure RTC_B interrupts for wake-up (set RTCTEVIE, RTCAIE, RT1PSIE, or RTCOFIE. If the alarm interrupt is also used as wake-up event, the alarm registers must be configured as needed).
2. Enter LPMx.5 with LPMx.5 entry sequence.

```
MOV #PMMKEY + PMMREGOFF, &PMMCTL0 ; Open PMM registers for write and set PMMREGOFF
                                  ;
BIS #LPM4,SR                      ; Enter LPMx.5 when PMMREGOFF is set
```

3. LOCKLPM5 is automatically set by hardware upon entering LPMx.5, the core voltage regulator is disabled, and all clocks are disabled except for the 32-kHz crystal oscillator clock if the RTC is enabled with RTCHOLD = 0.
4. An LPMx.5 wake-up event, such as an edge on a wake-up input pin, or an RTC_B interrupt event will start the BOR entry sequence together with the core voltage regulator. All peripheral registers are set to their default conditions. The I/O pin state remains locked as well as the interrupt configuration for the RTC_B.
5. The device can be configured. The I/O configuration and the RTC_B interrupt configuration that was not retained during LPM3.5 should be restored to the values before entering LPM3.5. Then the LOCKLPM5 bit can be cleared, this releases the I/O pin conditions as well as the RTC_B interrupt configuration.
6. After enabling I/O and RTC_B interrupts, the interrupt that caused the wake-up can be serviced.
7. To re-enter LPMx.5, the LOCKLPM5 bit must be cleared before re-entry, otherwise LPMx.5 is not entered.

If the RTC is enabled (RTCHOLD = 0), the 32-kHz oscillator remains active during LPM3.5. The fault detection also remains functional. If a fault occurs during LPM3.5 and the RTCOFIE was set before entering LPM3.5, a wake-up event is issued.

1.3 RTC_B Registers

The RTC_B module registers are listed in Table 1-1. This table also lists the retention during LPMx.5. Registers that are not retained during LPMx.5 must be restored after exit from LPMx.5. The base address for the RTC_B module registers can be found in the device-specific data sheet. The address offsets are given in Table 1-1.

NOTE: Most registers have word or byte register access. For a generic register *ANYREG*, the suffix "_L" (*ANYREG_L*) refers to the lower byte of the register (bits 0 through 7). The suffix "_H" (*ANYREG_H*) refers to the upper byte of the register (bits 8 through 15).

Table 1-1. RTC_B Registers

Offset	Acronym	Register Name	Type	Access	Reset	LPMx.5 Operation
00h	RTCCTL01	Real-Time Clock Control 0, 1	Read/write	Word	7000h	not retained
00h	RTCCTL0 or RTCCTL01_L	Real-Time Clock Control 0	Read/write	Byte	00h	not retained
01h	RTCCTL1 or RTCCTL01_H	Real-Time Clock Control 1	Read/write	Byte	70h	not retained
02h	RTCCTL23	Real-Time Clock Control 2, 3	Read/write	Word	0000h	retained
02h	RTCCTL2 or RTCCTL23_L	Real-Time Clock Control 2	Read/write	Byte	00h	retained
03h	RTCCTL3 or RTCCTL23_H	Real-Time Clock Control 3	Read/write	Byte	00h	retained
08h	RTCPS0CTL	Real-Time Prescale Timer 0 Control	Read/write	Word	0000h	not retained
08h	RTCPS0CTLL or RTCPS0CTL_L		Read/write	Byte	00h	not retained
09h	RTCPS0CTLH or RTCPS0CTL_H		Read/write	Byte	00h	not retained
0Ah	RTCPS1CTL	Real-Time Prescale Timer 1 Control	Read/write	Word	0000h	not retained
0Ah	RTCPS1CTLL or RTCPS1CTL_L		Read/write	Byte	00h	not retained
0Bh	RTCPS0CTLH or RTCPS0CTL_H		Read/write	Byte	00h	not retained
0Ch	RTCPS	Real-Time Prescale Timer 0, 1 Counter	Read/write	Word	none	retained
0Ch	RT0PS or RTCPS_L	Real-Time Prescale Timer 0 Counter	Read/write	Byte	none	retained
0Dh	RT1PS or RTCPS_H	Real-Time Prescale Timer 1 Counter	Read/write	Byte	none	retained
0Eh	RTCIV	Real Time Clock Interrupt Vector	Read	Word	0000h	not retained
10h	RTCTIM0	Real-Time Clock Seconds, Minutes	Read/write	Word	undefined	retained
10h	RTCSEC or RTCTIM0_L	Real-Time Clock Seconds	Read/write	Byte	undefined	retained
11h	RTCMIN or RTCTIM0_H	Real-Time Clock Minutes	Read/write	Byte	undefined	retained
12h	RTCTIM1	Real-Time Clock Hour, Day of Week	Read/write	Word	undefined	retained
12h	RTCHOUR or RTCTIM1_L	Real-Time Clock Hour	Read/write	Byte	undefined	retained
13h	RTCDOW or RTCTIM1_H	Real-Time Clock Day of Week	Read/write	Byte	undefined	retained

Table 1-1. RTC_B Registers (continued)

Offset	Acronym	Register Name	Type	Access	Reset	LPMx.5 Operation
14h	RTCDATE	Real-Time Clock Date	Read/write	Word	undefined	retained
14h	RTCDAY	Real-Time Clock Day of Month	Read/write	Byte	undefined	retained
	or RTCDATE_L					
15h	RTCMON	Real-Time Clock Month	Read/write	Byte	undefined	retained
	or RTCDATE_H					
16h	RTCYEAR	Real-Time Clock Year(1)	Read/write	Word	undefined	retained
18h	RTCAMINHR	Real-Time Clock Minutes, Hour Alarm	Read/write	Word	undefined	retained
18h	RTCAMIN	Real-Time Clock Minutes Alarm	Read/write	Byte	undefined	retained
	or RTCAMINHR_L					
19h	RTCAHOUR	Real-Time Clock Hours Alarm	Read/write	Byte	undefined	retained
	or RTCAMINHR_H					
1Ah	RTCADOWDAY	Real-Time Clock Day of Week, Day of Month Alarm	Read/write	Word	undefined	retained
1Ah	RTCADOW	Real-Time Clock Day of Week Alarm	Read/write	Byte	undefined	retained
	or RTCADOWDAY_L					
1Bh	RTCADAY	Real-Time Clock Day of Month Alarm	Read/write	Byte	undefined	retained
	or RTCADOWDAY_H					
1Ch	BIN2BCD	Binary-to-BCD Conversion Register	Read/write	Word	00h	not retained
1Eh	BCD2BIN	BCD-to-Binary Conversion Register	Read/write	Word	00h	not retained

(1) Do not access the RTCYEAR register in byte mode.

1.3.1 RTCCTL0 Register

Real-Time Clock Control 0 Register

Figure 1-2. RTCCTL0 Register

7	6	5	4	3	2	1	0
RTCOFIE(1)	RTCTEVIE(1)	RTCAIE(1)	RTCRDYIE	RTCOFIFG	RTCTEVIFG	RTCAIFG	RTCRDYIFG
rw-0	rw-0	rw-0	rw-0	rw-(0)	rw-(0)	rw-(0)	rw-(0)

(1) The configuration of these bits is retained during LPMx.5 until LOCKLPM5 is cleared, but not the register bits itself; therefore, reconfiguration after wake-up from LPMx.5 before clearing LOCKLPM5 is required.

Table 1-2. RTCCTL0 Register Description

Bit	Field	Type	Reset	Description
7	RTCOFIE	RW	0h	32-kHz crystal oscillator fault interrupt enable. This interrupt can be used as LPMx.5 wake-up event. 0b = Interrupt not enabled 1b = Interrupt enabled (LPMx.5 wake-up enabled)
6	RTCTEVIE	RW	0h	Real-time clock time event interrupt enable. In modules supporting LPMx.5 this interrupt can be used as LPMx.5 wake-up event. 0b = Interrupt not enabled 1b = Interrupt enabled (LPMx.5 wake-up enabled)
5	RTCAIE	RW	0h	Real-time clock alarm interrupt enable. In modules supporting LPMx.5 this interrupt can be used as LPMx.5 wake-up event. 0b = Interrupt not enabled 1b = Interrupt enabled (LPMx.5 wake-up enabled)
4	RTCRDYIE	RW	0h	Real-time clock ready interrupt enable. 0b = Interrupt not enabled 1b = Interrupt enabled
3	RTCOFIFG	RW	0h	32-kHz crystal oscillator fault interrupt flag. This interrupt can be used as LPMx.5 wake-up event. 0b = No interrupt pending 1b = Interrupt pending. A 32-kHz crystal oscillator fault occurred after last reset.
2	RTCTEVIFG	RW	0h	Real-time clock time event interrupt flag. In modules supporting LPMx.5 this interrupt can be used as LPMx.5 wake-up event. 0b = No time event occurred 1b = Time event occurred
1	RTCAIFG	RW	0h	Real-time clock alarm interrupt flag. In modules supporting LPMx.5 this interrupt can be used as LPMx.5 wake-up event. 0b = No time event occurred 1b = Time event occurred
0	RTCRDYIFG	RW	0h	Real-time clock ready interrupt flag 0b = RTC cannot be read safely 1b = RTC can be read safely

1.3.2 RTCCTL1 Register

Real-Time Clock Control Register 1

Figure 1-3. RTCCTL1 Register

7	6	5	4	3	2	1	0
RTCBCD	RTCHOLD[1]	Reserved	RTCRDY	Reserved		RTCTEVx[1]	
rw-(0)	rw-(1)	r1	r-(1)	r0	r0	rw-(0)	rw-(0)

(1) The configuration of these bits is retained during LPMx.5 until LOCKLPM5 is cleared, but not the register bits itself; therefore, reconfiguration after wake-up from LPMx.5 before clearing LOCKLPM5 is required.

Table 1-3. RTCCTL1 Register Description

Bit	Field	Type	Reset	Description
7	RTCBCD	RW	0h	Real-time clock BCD select. Selects BCD counting for real-time clock. 0b = Binary-hexadecimal code selected 1b = BCD Binary coded decimal (BCD) code selected
6	RTCHOLD	RW	1h	Real-time clock hold 0b = Real-time clock is operational. 1b = The calendar is stopped as well as the prescale counters, RT0PS, and RT1PS.
5	Reserved	R	1h	Reserved. Always read as 1.
4	RTCRDY	RW	1h	Real-time clock ready 0b = RTC time values in transition 1b = RTC time values safe for reading. This bit indicates when the real-time clock time values are safe for reading.
3-2	Reserved	R	0h	Reserved. Always read as 0.
1-0	RTCTEVx	RW	0h	Real-time clock time interrupt event 00b = Minute changed 01b = Hour changed 10b = Every day at midnight (00:00) 11b = Every day at noon (12:00)

1.3.3 RTCCTL2 Register

Real-Time Clock Control 2 Register

Figure 1-4. RTCCTL2 Register

7	6	5	4	3	2	1	0
RTCCALS	Reserved	RTCCALx					
rw-(0)	r0	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

Table 1-4. RTCCTL2 Register Description

Bit	Field	Type	Reset	Description
7	RTCCALS	RW	0h	Real-time clock calibration sign 0b = Frequency adjusted down 1b = Frequency adjusted up
6	Reserved	R	0h	Reserved. Always read as 0.
5-0	RTCCALx	RW	0h	Real-time clock calibration. Each LSB represents approximately +4-ppm (RTCCALS = 1) or a –2-ppm (RTCCALS = 0) adjustment in frequency.

1.3.4 RTCCTL3 Register

Real-Time Clock Control 3 Register

Figure 1-5. RTCCTL3 Register

7	6	5	4	3	2	1	0
Reserved						RTCCALFx	
r0	r0	r0	r0	r0	r0	rw-(0)	rw-(0)

Table 1-5. RTCCTL3 Register Description

Bit	Field	Type	Reset	Description
7-2	Reserved	R	0h	Reserved. Always read as 0.
1-0	RTCCALFx	RW	0h	Real-time clock calibration frequency. Selects frequency output to RTCCLK pin for calibration measurement. The corresponding port must be configured for the peripheral module function. 00b = No frequency output to RTCCLK pin 01b = 512 Hz 10b = 256 Hz 11b = 1 Hz

1.3.5 RTCSEC Register – Hexadecimal Format

Real-Time Clock Seconds Register – Hexadecimal Format

Figure 1-6. RTCSEC Register

7	6	5	4	3	2	1	0
0	0	Seconds					
r-0	r-0	rw	rw	rw	rw	rw	rw

Table 1-6. RTCSEC Register Description

Bit	Field	Type	Reset	Description
7-6	0	R	0h	Always reads as 0.
5-0	Seconds	RW	undefined	Seconds. Valid values are 0 to 59.

1.3.6 RTCSEC Register – BCD Format

Real-Time Clock Seconds Register – BCD Format

Figure 1-7. RTCSEC Register

7	6	5	4	3	2	1	0
0	Seconds – high digit			Seconds – low digit			
r-0	rw	rw	rw	rw	rw	rw	rw

Table 1-7. RTCSEC Register Description

Bit	Field	Type	Reset	Description
7	0	R	0h	Always reads as 0.
6-4	Seconds – high digit	RW	undefined	Seconds – high digit. Valid values are 0 to 5.
3-0	Seconds – low digit	RW	undefined	Seconds – low digit. Valid values are 0 to 9.

1.3.7 RTCMIN Register – Hexadecimal Format

Real-Time Clock Minutes Register – Hexadecimal Format

Figure 1-8. RTCMIN Register

7	6	5	4	3	2	1	0
0	0	Minutes					
r-0	r-0	rw	rw	rw	rw	rw	rw

Table 1-8. RTCMIN Register Description

Bit	Field	Type	Reset	Description
7-6	0	R	0h	Always reads as 0.
5-0	Minutes	RW	undefined	Minutes. Valid values are 0 to 59.

1.3.8 RTCMIN Register – BCD Format

Real-Time Clock Minutes Register – BCD Format

Figure 1-9. RTCMIN Register

7	6	5	4	3	2	1	0
0	Minutes – high digit			Minutes – low digit			
r-0	rw	rw	rw	rw	rw	rw	rw

Table 1-9. RTCMIN Register Description

Bit	Field	Type	Reset	Description
7	0	R	0h	Always reads as 0.
6-4	Minutes – high digit	RW	undefined	Minutes – high digit. Valid values are 0 to 5.
3-0	Minutes – low digit	RW	undefined	Minutes – low digit. Valid values are 0 to 9.

1.3.9 RTCHOUR Register – Hexadecimal Format

Real-Time Clock Hours Register – Hexadecimal Format

Figure 1-10. RTCHOUR Register

7	6	5	4	3	2	1	0
0	0	0	Hours				
r-0	r-0	r-0	rw	rw	rw	rw	rw

Table 1-10. RTCHOUR Register Description

Bit	Field	Type	Reset	Description
7-5	0	R	0h	Always reads as 0.
4-0	Hours	RW	undefined	Hours. Valid values are 0 to 23.

1.3.10 RTCHOUR Register – BCD Format

Real-Time Clock Hours Register – BCD Format

Figure 1-11. RTCHOUR Register

7	6	5	4	3	2	1	0
0	0	Hours – high digit		Hours – low digit			
r-0	r-0	rw	rw	rw	rw	rw	rw

Table 1-11. RTCHOUR Register Description

Bit	Field	Type	Reset	Description
7-6	0	R	0h	Always reads as 0.
5-4	Hours – high digit	RW	undefined	Hours – high digit. Valid values are 0 to 2.
3-0	Hours – low digit	RW	undefined	Hours – low digit. Valid values are 0 to 9.

1.3.11 RTCDOW Register

Real-Time Clock Day of Week Register

Figure 1-12. RTCDOW Register

7	6	5	4	3	2	1	0
0	0	0	0	0	Day of week		
r-0	r-0	r-0	r-0	r-0	rw	rw	rw

Table 1-12. RTCDOW Register Description

Bit	Field	Type	Reset	Description
7-3	0	R	0h	Always reads as 0.
2-0	Day of week	RW	undefined	Day of week. Valid values are 0 to 6.

1.3.12 RTCDAY Register – Hexadecimal Format

Real-Time Clock Day of Month Register – Hexadecimal Format

Figure 1-13. RTCDAY Register

7	6	5	4	3	2	1	0
0	0	0	Day of month				
r-0	r-0	r-0	rw	rw	rw	rw	rw

Table 1-13. RTCDAY Register Description

Bit	Field	Type	Reset	Description
7-5	0	R	0h	Always reads as 0.
4-0	Day of month	RW	undefined	Day of month. Valid values are 1 to 31.

1.3.13 RTCDAY Register – BCD Format

Real-Time Clock Day of Month Register – BCD Format

Figure 1-14. RTCDAY Register

7	6	5	4	3	2	1	0
0	0	Day of month – high digit		Day of month – low digit			
r-0	r-0	rw	rw	rw	rw	rw	rw

Table 1-14. RTCDAY Register Description

Bit	Field	Type	Reset	Description
7-6	0	R	0h	Always reads as 0.
5-4	Day of month – high digit	RW	undefined	Day of month – high digit. Valid values are 0 to 3.
3-0	Day of month – low digit	RW	undefined	Day of month – low digit. Valid values are 0 to 9.

1.3.14 RTCMON Register – Hexadecimal Format

Real-Time Clock Month Register – Hexadecimal Format

Figure 1-15. RTCMON Register

7	6	5	4	3	2	1	0
0	0	0	0	Month			
r-0	r-0	r-0	r-0	rw	rw	rw	rw

Table 1-15. RTCMON Register Description

Bit	Field	Type	Reset	Description
7-4	0	R	0h	Always reads as 0.
3-0	Month	RW	undefined	Month. Valid values are 1 to 12.

1.3.15 RTCMON Register – BCD Format

Real-Time Clock Month Register

Figure 1-16. RTCMON Register

7	6	5	4	3	2	1	0
0	0	0	Month – high digit	Month – low digit			
r-0	r-0	r-0	rw	rw	rw	rw	rw

Table 1-16. RTCMON Register Description

Bit	Field	Type	Reset	Description
7-5	0	R	0h	Always reads as 0.
4	Month – high digit	RW	undefined	Month – high digit. Valid values are 0 or 1.
3-0	Month – low digit	RW	undefined	Month – low digit. Valid values are 0 to 9.

1.3.16 RTCYEAR Register – Hexadecimal Format

Real-Time Clock Year Register – Hexadecimal Format

Figure 1-17. RTCYEAR Register

15	14	13	12	11	10	9	8
0	0	0	0	Year – high byte			
r-0	r-0	r-0	r-0	rw	rw	rw	rw
7	6	5	4	3	2	1	0
rw	rw	rw	rw	rw	rw	rw	rw

Table 1-17. RTCYEAR Register Description

Bit	Field	Type	Reset	Description
15-12	0	R	0h	Always reads as 0.
11-8	Year – high byte	RW	undefined	Year – high byte. Valid values of Year are 0 to 4095.
7-0	Year – low byte	RW	undefined	Year – low byte. Valid values of Year are 0 to 4095.

1.3.17 RTCYEAR Register – BCD Format

Real-Time Clock Year Register – BCD Format

Figure 1-18. RTCYEAR Register

15	14	13	12	11	10	9	8
0	Century – high digit			Century – low digit			
r-0	rw	rw	rw	rw	rw	rw	rw
7	6	5	4	3	2	1	0
Decade				Year – lowest digit			
rw	rw	rw	rw	rw	rw	rw	rw

Table 1-18. RTCYEAR Register Description

Bit	Field	Type	Reset	Description
15	0	R	0h	Always reads as 0.
14-12	Century – high digit	RW	undefined	Century – high digit . Valid values are 0 to 4.
11-8	Century – low digit	RW	undefined	Century – low digit. Valid values are 0 to 9.
7-4	Decade	RW	undefined	Decade. Valid values are 0 to 9.
3-0	Year – lowest digit	RW	undefined	Year – lowest digit. Valid values are 0 to 9.

1.3.18 RTCAMIN Register – Hexadecimal Format

Real-Time Clock Minutes Alarm Register – Hexadecimal Format

Figure 1-19. RTCAMIN Register

7	6	5	4	3	2	1	0
AE	0	Minutes					
rw	r-0	rw	rw	rw	rw	rw	rw

Table 1-19. RTCAMIN Register Description

Bit	Field	Type	Reset	Description
7	AE	RW	undefined	Alarm enable 0b = This alarm register is disabled 1b = This alarm register is enabled
6	0	R	0h	Always reads as 0.
5-0	Minutes	RW	undefined	Minutes. Valid values are 0 to 59.

1.3.19 RTCAMIN Register – BCD Format

Real-Time Clock Minutes Alarm Register – BCD Format

Figure 1-20. RTCAMIN Register

7	6	5	4	3	2	1	0
AE	Minutes – high digit			Minutes – low digit			
rw	rw	rw	rw	rw	rw	rw	rw

Table 1-20. RTCAMIN Register Description

Bit	Field	Type	Reset	Description
7	AE	RW	undefined	Alarm enable 0b = This alarm register is disabled 1b = This alarm register is enabled
6-4	Minutes – high digit	RW	undefined	Minutes – high digit. Valid values are 0 to 5.
3-0	Minutes – low digit	RW	undefined	Minutes – low digit. Valid values are 0 to 9.

1.3.20 RTCAHOUR Register – Hexadecimal Format

Real-Time Clock Hours Alarm Register – Hexadecimal Format

Figure 1-21. RTCAHOUR Register

7	6	5	4	3	2	1	0
AE	0	0	Hours				
rw	r-0	r-0	rw	rw	rw	rw	rw

Table 1-21. RTCAHOUR Register Description

Bit	Field	Type	Reset	Description
7	AE	RW	undefined	Alarm enable 0b = This alarm register is disabled 1b = This alarm register is enabled
6-5	0	R	0h	Always reads as 0.
4-0	Hours	RW	undefined	Hours. Valid values are 0 to 23.

1.3.21 RTCAHOUR Register – BCD Format

Real-Time Clock Hours Alarm Register – BCD Format

Figure 1-22. RTCAHOUR Register

7	6	5	4	3	2	1	0
AE	0	Hours – high digit		Hours – low digit			
rw	r-0	rw	rw	rw	rw	rw	rw

Table 1-22. RTCAHOUR Register Description

Bit	Field	Type	Reset	Description
7	AE	RW	undefined	Alarm enable 0b = This alarm register is disabled 1b = This alarm register is enabled
6	0	R	0h	Always reads as 0.
5-4	Hours – high digit	RW	undefined	Hours – high digit. Valid values are 0 to 2.
3-0	Hours – low digit	RW	undefined	Hours – low digit. Valid values are 0 to 9.

1.3.22 RTCADOW Register

Real-Time Clock Day of Week Alarm Register

Figure 1-23. RTCADOW Register

7	6	5	4	3	2	1	0
AE	0	0	0	0	Day of week		
rw	r-0	r-0	r-0	r-0	rw	rw	rw

Table 1-23. RTCADOW Register Description

Bit	Field	Type	Reset	Description
7	AE	RW	undefined	Alarm enable 0b = This alarm register is disabled 1b = This alarm register is enabled
6-3	0	R	0h	Always reads as 0.
2-0	Day of week	RW	undefined	Day of week. Valid values are 0 to 6.

1.3.23 RTCADAY Register – Hexadecimal Format

Real-Time Clock Day of Month Alarm Register – Hexadecimal Format

Figure 1-24. RTCADAY Register

7	6	5	4	3	2	1	0
AE	0	0	Day of month				
rw	r-0	r-0	rw	rw	rw	rw	rw

Table 1-24. RTCADAY Register Description

Bit	Field	Type	Reset	Description
7	AE	RW	undefined	Alarm enable 0b = This alarm register is disabled 1b = This alarm register is enabled
6-5	0	R	0h	Always reads as 0.
4-0	Day of month	RW	undefined	Day of month. Valid values are 1 to 31.

1.3.24 RTCADAY Register – BCD Format

Real-Time Clock Day of Month Alarm Register – BCD Format

Figure 1-25. RTCADAY Register

7	6	5	4	3	2	1	0
AE	0	Day of month – high digit		Day of month – low digit			
rw	r-0	rw	rw	rw	rw	rw	rw

Table 1-25. RTCADAY Register Description

Bit	Field	Type	Reset	Description
7	AE	RW	undefined	Alarm enable 0b = This alarm register is disabled 1b = This alarm register is enabled
6	0	R	0h	Always reads as 0.
5-4	Day of month – high digit	RW	undefined	Day of month – high digit. Valid values are 0 to 3.
3-0	Day of month – low digit	RW	undefined	Day of month – low digit. Valid values are 0 to 9.

1.3.25 RTCPS0CTL Register

Real-Time Clock Prescale Timer 0 Control Register

Figure 1-26. RTCPS0CTL Register

15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved			RT0IPx(1)			RT0PSIE	RT0PSIFG
r0	r0	r0	rw-(0)	rw-(0)	rw-(0)	rw-0	rw-(0)

(1) The configuration of these bits is retained during LPMx.5 until LOCKLPM5 is cleared, but not the register bits itself; therefore, reconfiguration after wake-up from LPMx.5 before clearing LOCKLPM5 is required.

Table 1-26. RTCPS0CTL Register Description

Bit	Field	Type	Reset	Description
15-5	Reserved	R	0h	Reserved. Always reads as 0.
4-2	RT0IPx	RW	0h	Prescale timer 0 interrupt interval 000b = Divide by 2 001b = Divide by 4 010b = Divide by 8 011b = Divide by 16 100b = Divide by 32 101b = Divide by 64 110b = Divide by 128 111b = Divide by 256
1	RT0PSIE	RW	0h	Prescale timer 0 interrupt enable 0b = Interrupt not enabled 1b = Interrupt enabled
0	RT0PSIFG	RW	0h	Prescale timer 0 interrupt flag 0b = No time event occurred 1b = Time event occurred

1.3.26 RTCPS1CTL Register

Real-Time Clock Prescale Timer 1 Control Register

Figure 1-27. RTCPS1CTL Register

15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved			RT1IPx(1)			RT1PSIE(1)	RT1PSIFG
r0	r0	r0	rw-(0)	rw-(0)	rw-(0)	rw-0	rw-(0)

(1) The configuration of these bits is retained during LPMx.5 until LOCKLPM5 is cleared, but not the register bits themselves; therefore, reconfiguration after wake-up from LPMx.5 before clearing LOCKLPM5 is required.

Table 1-27. RTCPS1CTL Register Description

Bit	Field	Type	Reset	Description
15-5	Reserved	R	0h	Reserved. Always reads as 0.
4-2	RT1IPx	RW	0h	Prescale timer 1 interrupt interval 000b = Divide by 2 001b = Divide by 4 010b = Divide by 8 011b = Divide by 16 100b = Divide by 32 101b = Divide by 64 110b = Divide by 128 111b = Divide by 256
1	RT1PSIE	RW	0h	Prescale timer 1 interrupt enable 0b = Interrupt not enabled 1b = Interrupt enabled (LPMx.5 wake-up enabled)
0	RT1PSIFG	RW	0h	Prescale timer 1 interrupt flag. In modules supporting LPMx.5 this interrupt can be used as LPMx.5 wake-up event. 0b = No time event occurred 1b = Time event occurred

1.3.27 RTCPS0 Register

Real-Time Clock Prescale Timer 0 Counter Register

Figure 1-28. RTCPS0 Register

7	6	5	4	3	2	1	0
RT0PS							
rw	rw	rw	rw	rw	rw	rw	rw

Table 1-28. RTCPS0 Register Description

Bit	Field	Type	Reset	Description
7-0	RT0PS	RW	undefined	Prescale timer 0 counter value

1.3.28 RTCPS1 Register

Real-Time Clock Prescale Timer 1 Counter Register

Figure 1-29. RTCPS1 Register

7	6	5	4	3	2	1	0
RT1PS							
rw	rw	rw	rw	rw	rw	rw	rw

Table 1-29. RTCPS1 Register Description

Bit	Field	Type	Reset	Description
7-0	RT1PS	RW	undefined	Prescale timer 1 counter value

1.3.29 RTCIV Register

Real-Time Clock Interrupt Vector Register

Figure 1-30. RTCIV Register

15	14	13	12	11	10	9	8
RTCIVx							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
RTCIVx							
r0	r0	r0	r0	r-(0)	r-(0)	r-(0)	r0

Table 1-30. RTCIV Register Description

Bit	Field	Type	Reset	Description
15-0	RTCIVx	R	0h	Real-time clock interrupt vector value 00h = No interrupt pending 02h = Interrupt Source: RTC ready; Interrupt Flag: RTCRDYIFG; Interrupt Priority: Highest 04h = Interrupt Source: RTC interval timer; Interrupt Flag: RTCTEVIFG 06h = Interrupt Source: RTC user alarm; Interrupt Flag: RTCAIFG 08h = Interrupt Source: RTC prescaler 0; Interrupt Flag: RT0PSIFG 0Ah = Interrupt Source: RTC prescaler 1; Interrupt Flag: RT1PSIFG 0Ch = Interrupt Source: RTC oscillator failure; Interrupt Flag: RTCOFIFG 0Eh = Reserved; Interrupt Priority: Lowest

1.3.30 BIN2BCD Register

Binary-to-BCD Conversion Register

Figure 1-31. BIN2BCD Register

15	14	13	12	11	10	9	8
BIN2BCDx							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
7	6	5	4	3	2	1	0
BIN2BCDx							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-31. BIN2BCD Register Description

Bit	Field	Type	Reset	Description
15-0	BIN2BCDx	RW	0h	Read: 16-bit BCD conversion of previously written 12-bit binary number Write: 12-bit binary number to be converted

1.3.31 BCD2BIN Register

BCD-to-Binary Conversion Register

Figure 1-32. BCD2BIN Register

15	14	13	12	11	10	9	8
BCD2BINx							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
7	6	5	4	3	2	1	0
BCD2BINx							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-32. BCD2BIN Register Description

Bit	Field	Type	Reset	Description
15-0	BCD2BINx	RW	0h	Read: 12-bit binary conversion of previously written 16-bit BCD number Write: 16-bit BCD number to be converted

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ('TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265