OMAP5910 Dual-Core Processor Real-Time Clock (RTC) Reference Guide

Literature Number: SPRU687
October 2003

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information 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.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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

Copyright © 2003, Texas Instruments Incorporated

Read This First

About This Manual

The real time clock (RTC) is an embedded module. Its basic features include:

- Time information (seconds/minutes/hours) directly in BCD code
- Calendar information (day/month/year/day of the week) directly in BCD code up to the year 2099
- Interrupt generation, at 1second, 1minute, 1hour, or 1day periods or at a precise, predetermined time of the day (alarm function)
- 30 second time correction
- Oscillator drift compensation

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.

Related Documentation From Texas Instruments

The following documents describe the OMAP5910 device and related peripherals. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

OMAP5910 Dual-Core Processor MPU Subsystem Reference Guide (literature number SPRU671)

OMAP5910 Dual-Core Processor DSP Subsystem Reference Guide (literature number SPRU672)

OMAP5910 Dual-Core Processor Memory Interface Traffic Controller Reference Guide (literature number SPRU673)

OMAP5910 Dual-Core Processor System DMA Controller Reference Guide (literature number SPRU674)

OMAP5910 Dual-Core Processor LCD Controller Reference Guide (literature number SPRU675)

OMAP5910 Dual-Core Processor Universal Asynchronous Receiver/ Transmitter (UART) Devices Reference Guide (literature number SPRU676)

OMAP5910 Dual-Core Processor Universal Serial Bus (USB) and Frame Adjustment Counter (FAC) Reference Guide (literature number SPRU677)

OMAP5910 Dual-Core Processor Clock Generation and System Reset Management Reference Guide (literature number SPRU678)

OMAP5910 Dual-Core Processor General-Purpose Input/Output (GPIO) Reference Guide (literature number SPRU679)

OMAP5910 Dual-Core Processor MMC/SD Reference Guide (literature number SPRU680)

OMAP5910 Dual-Core Processor Inter-Integrated Circuit (I2C) Controller Reference Guide (literature number SPRU681)

OMAP5910 Dual-Core Processor Timer Reference Guide (literature number SPRU682)

OMAP5910 Dual-Core Processor Inter-Processor Communication Reference Guide (literature number SPRU683)

OMAP5910 Dual-Core Processor Camera Interface Reference Guide (literature number SPRU684)

OMAP5905 Dual-Core Processor Multichannel Serial Interface (MCSI) Reference Guide (literature number SPRU685)

OMAP5910 Dual-Core Processor Micro-Wire Interface Reference Guide (literature number SPRU686)

OMAP5910 Dual-Core Processor Real-Time Clock (RTC) Reference Guide (literature number SPRU687)

OMAP5910 Dual-Core Processor HDQ/1-Wire Interface Reference Guide (literature number SPRU688)

OMAP5910 Dual-Core Processor PWL, PWT, and LED Peripheral Reference Guide (literature number SPRU689)

OMAP5910 Dual-Core Processor Multichannel Buffered Serial Port (McBSP) Reference Guide (literature number SPRU708)

Trademarks

OMAP and the OMAP symbol are trademarks of Texas Instruments.

(This page has been left blank intentionally.)

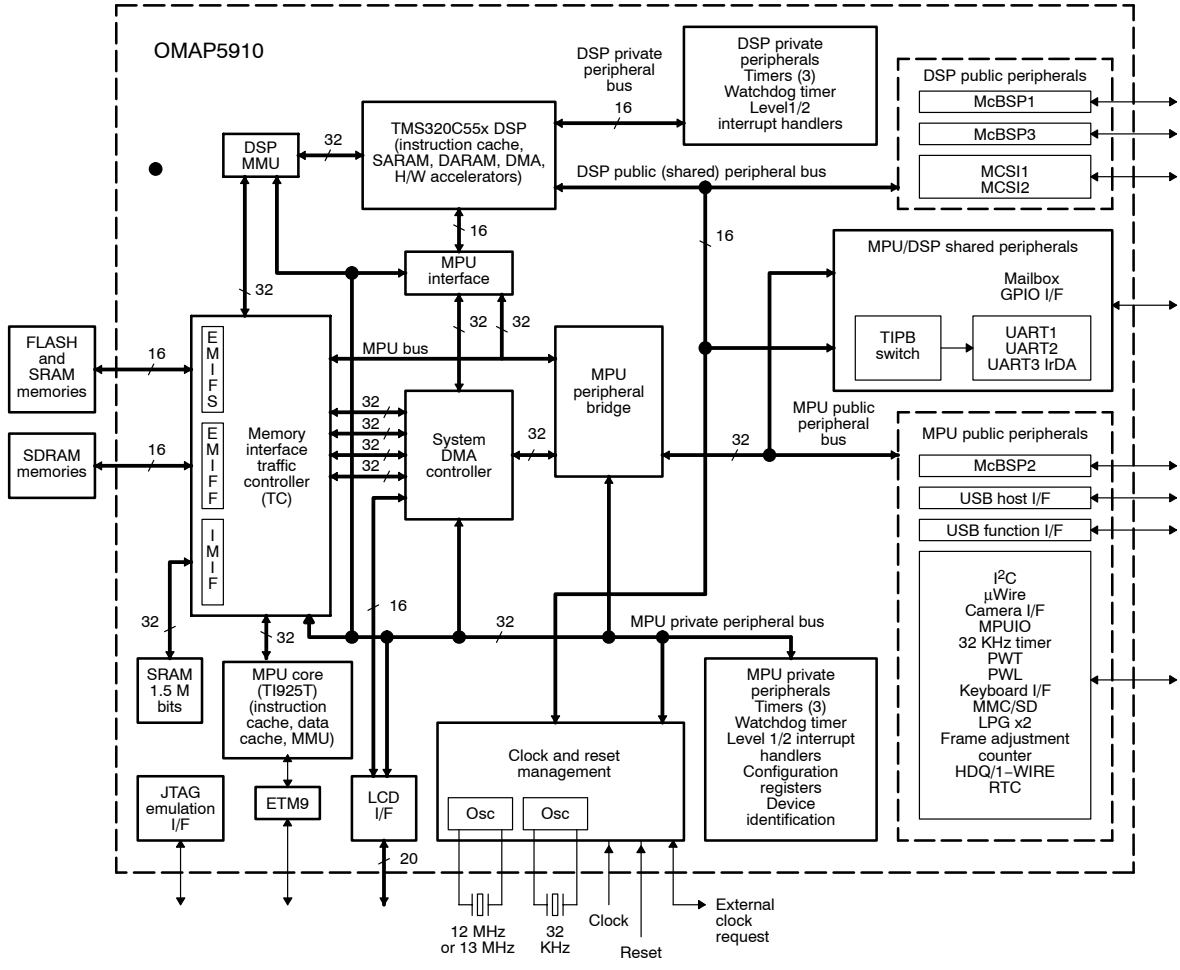
Contents

Figures

Tables

1 Real-Time Clock (RTC)

Figure 1. OMAP5910 Functional Overview

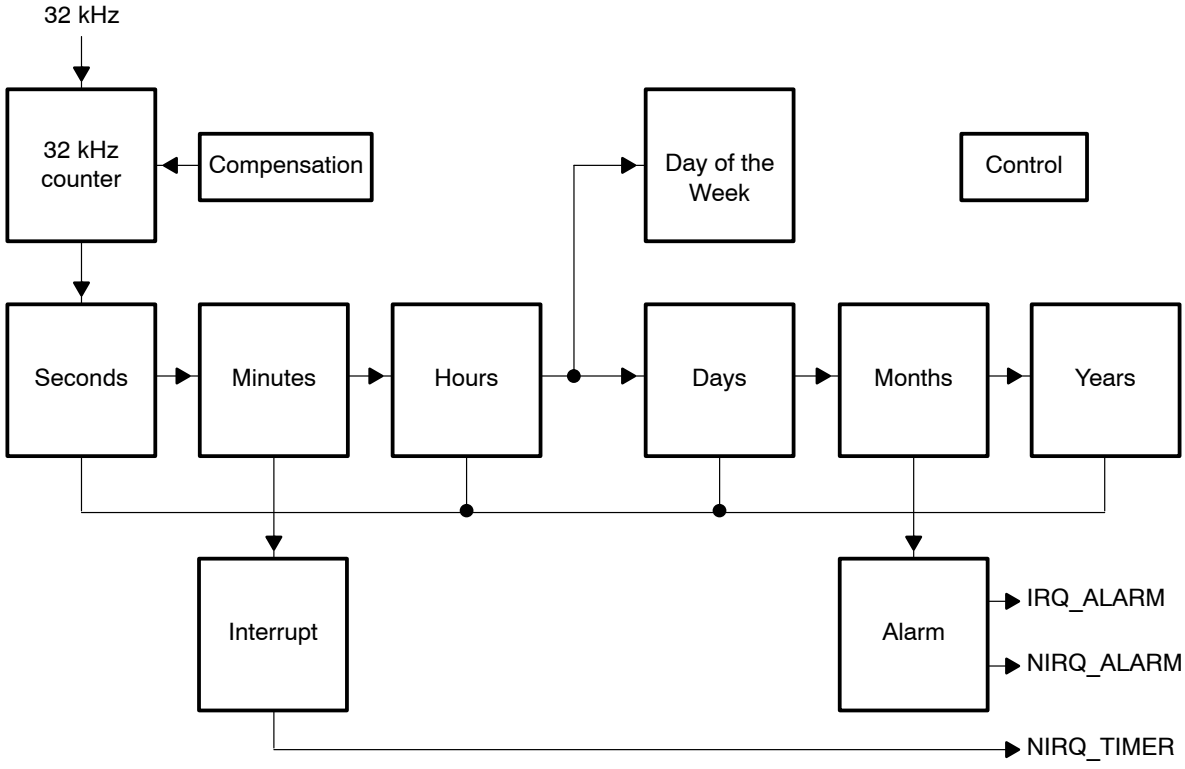


Note: The RTC section is shown at the bottom right in Figure 1.

The real time clock (RTC) shown in Figure 2, is an embedded module. Its basic features include:

- ❏ Time information (seconds/minutes/hours) directly in BCD code
- ❏ Calendar information (day/month/year/day of the week) directly in BCD code up to the year 2099
- ❏ Interrupt generation, at 1second, 1minute, 1hour, or 1day periods or at a precise, predetermined time of the day (alarm function)
- ❏ 30 second time correction
- ❏ Oscillator drift compensation

Figure 2. RTC Clock Diagram



1.1 Register Descriptions

All the time and calendar information is available in dedicated time and calendar registers. Time and calendar register values are written in binary−coded−decimal (BCD) code (see Table 1).

Table 1. Time and Calendar Register Values

Time Unit	Range	Remarks
Year	00 to 99	Leap year: year divisible by 4
		Common year: other year
Month	01 to 12	
Day	01 to 31	01 to 31 for months 1, 3, 5, 7, 8, 10, 12
		01 to 30 for months 4, 6, 9 ,11
		01 to 29 for month 2 (leap year)
		01 to 28 for month 2 (common year)
Week	00 to 06	Day of the week
Hour	00 to 23	00 to 23 in 24-hour mode
		01 to 12 in AM/PM mode
Minutes	00 to 59	
Seconds	00 to 59	

1.2 Register Access

There are three types of registers:

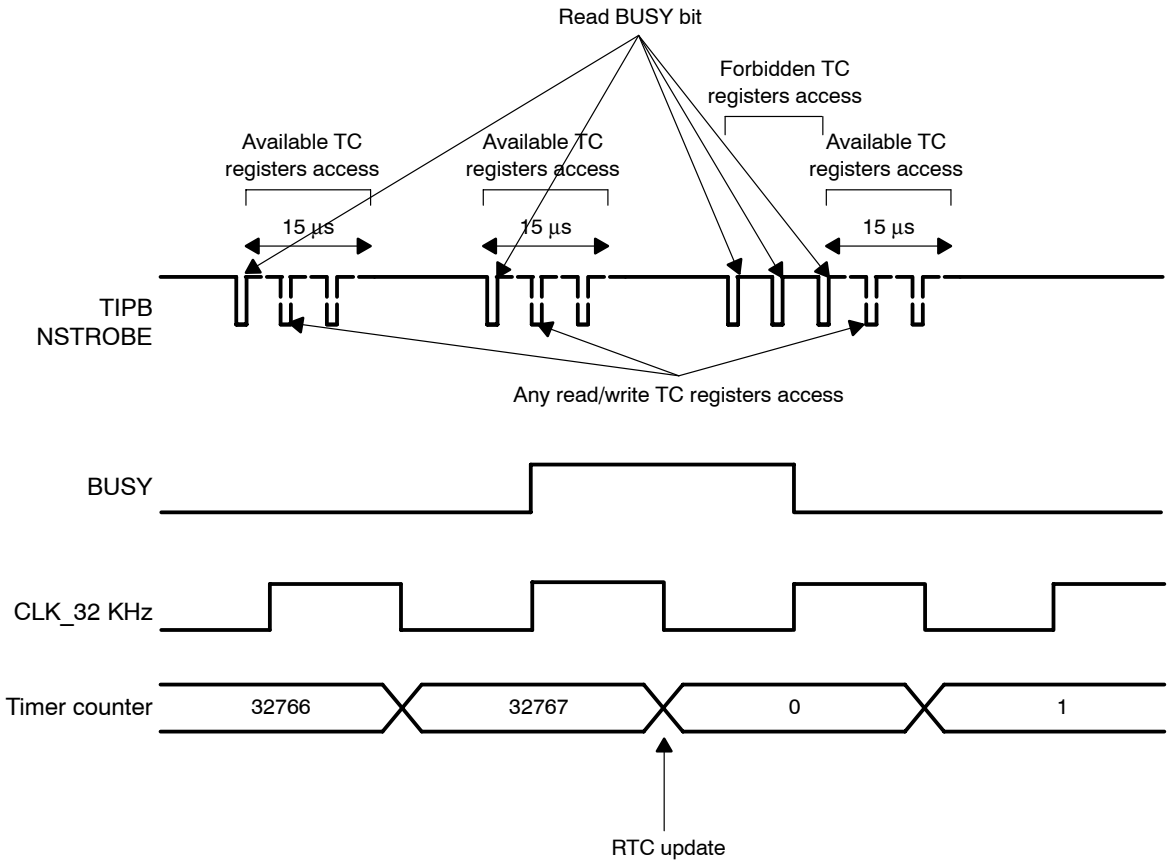
- ❏ Time and calendar registers/alarm
- ❏ General
- ❏ Compensation

These three types have their own access constraints.

1.2.1 Time and Calendar Registers/Alarm Registers

To read or write correct data to and from the time and calendar registers/alarm registers, the MPU must first poll the BUSY bit of the STATUS register until BUSY is equal to zero. From this time, for a period of 15 μs (the available access period), the MPU can safely access the time and calendar registers/alarm registers. At the end of the access period, the MPU must restart the previous sequence. If the MPU accesses the time and calendar registers outside of the access period, the access is not ensured (see Figure 3).

Figure 3. Time and Calendar Registers and Alarm Register Access



Access Period Violation

The programmer should disable all incoming interrupts during the register read process to prevent process interruption and possible violation of the authorized 15 µs access period.

1.2.2 General Registers

The MPU can access the STATUS_REG and the CTRL_REG at any time (except the CTRL_REG[5] bit, which can be changed only when the RTC is stopped).

For the INTERRUPTS_REG, the MPU must respect the available access period to prevent a spurious interrupt.

The RTC_DISABLE bit of the CTRL register can only be used to completely disable the RTC function. When this bit is set, the 32 kHz clock is gated, and the RTC is frozen. From this point, resetting this bit to zero can lead to unexpected behavior. To save power, this bit must only be used if the RTC function is not required in the application.

1.2.3 Compensation Registers

Access to the COMP_MSB_REG and COMP_LSB_REG registers must be only during the available access period. These registers must not be updated during compensation (first second of each hour), but it is acceptable to update them during the second, preceding a compensation event (see Figure 4).

For example, the MPU can load the compensation value into these registers after each hour event during an available access period.

1.2.4 Modify Time and Calendar Registers

To modify the current time, the MPU must write the new time into the time and calendar registers to fix the time and calendar information. The MPU can write into the time and calendar registers without stopping the RTC; but in this case, the MPU must read the status register to ensure that the RTC updating takes greater than 15 μs (bit BUSY should be 0). Then, the MPU must perform all changes in less than 15 μs to prevent partial updating between the start and the end of the writing sequence into the time and calendar registers.

The MPU can stop the RTC by clearing the STOP_RTC bit of the control register (owing to internal resynchronization, the RUN bit of the status must be checked to ensure that the RTC is frozen), update time and calendar values, and restart the RTC by resetting the STOP_RTC bit.

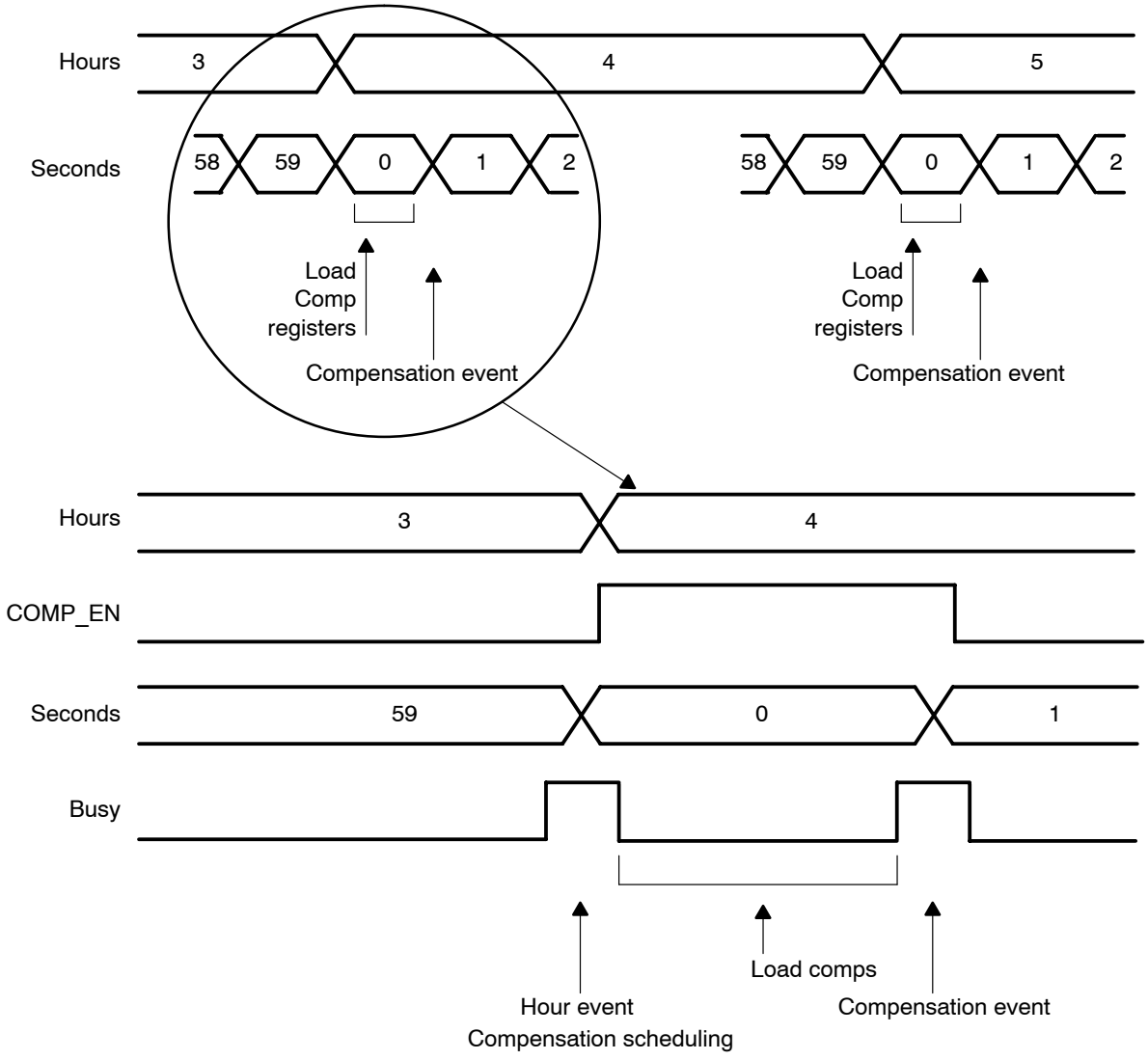
1.2.5 Rounding Seconds

Time can be rounded to the closest minute by setting ROUND_30S bit of the control register. When this bit is set, the time and calendar values are set to the closest minute value at the next second. ROUND_30S bit is automatically cleared when rounding time is performed.

Example:

- If the current time is 10H59M45S, the round operation changes the time to 11H00M00S.
- If the current time is 10H59M29S, the round operation changes the time to 10H59M00S.

Figure 4. Compensation Scheduling



1.2.6 Interrupt Management

RTC can generate two interrupts:

- ❏ Timer interrupt (IRQ_TIMER)
- ❏ Alarm interrupt (IRQ_ALARM_CHIP)

1.2.7 Timer Interrupt

IRQ_TIMER interrupt can be generated periodically every second, minute, hour, or day (RTC_INTERRUPTS_REG[1:0]).

The IT_TIMER bit of the interrupt register enables this interrupt.

The timer interrupt is a negative-edge-sensitive interrupt (low-level pulse duration = 15 μs).

RTC_STATUS_REG [5:2] are only updated at each new interrupt and show what events have happened, as shown in Table 2.

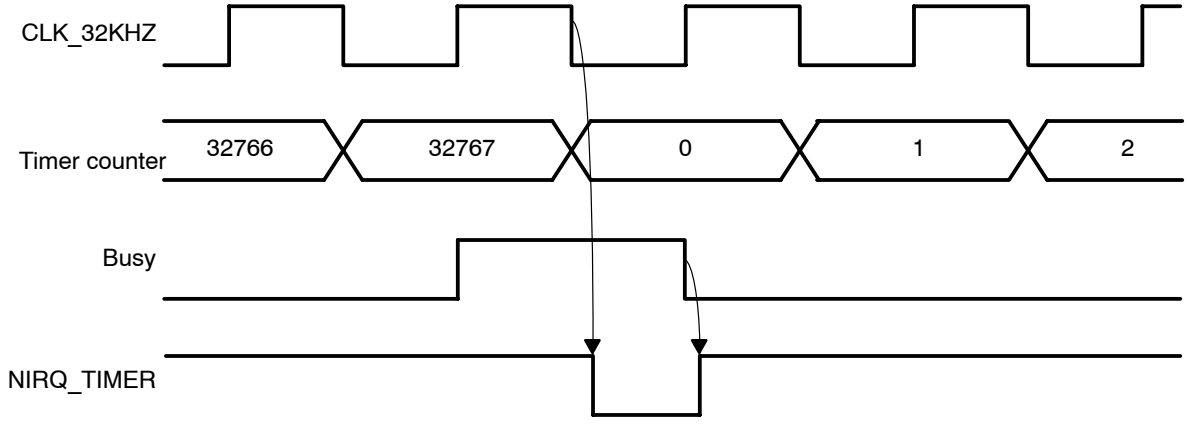
Table 2. Timer Interrupts

RTC_INTERRUPTS_REG[1:0]	11	10	01	00
RTC_STATUS_REG[5] (day)	1	0/1†	0/1†	0/1†
RTC_STATUS_REG[4] (hour)	1	1	0/1†	0/1†
RTC_STATUS_REG[3] (min)	1	1	1	0/1†
RTC_STATUS_REG[2] (sec)	1	1	1	1

† 1 when this event is concurrent with programmed period

Figure 5 shows IRQ generation waveform.

Figure 5. IRQ−Generation Waveform



1.2.8 Alarm Interrupt

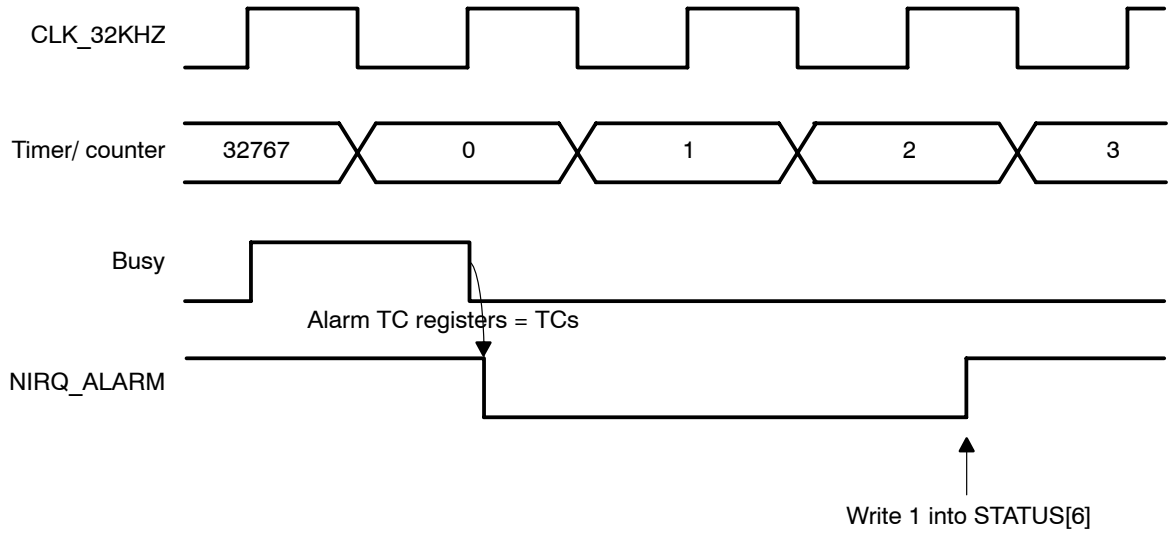
The IRQ_ALARM_CHIP interrupt can be generated when the time set into the time and calendar alarm registers is exactly the same as in the time and calendar registers (see Figure 6).

This interrupt is then generated if the IT_ALARM bit of the interrupts register is set.

This interrupt is low-level sensitive; RTC_STATUS_REG[6] indicates that IRQ_ALARM_CHIP occurred.

This interrupt is disabled by writing 1 into the RTC_STATUS_REG[6].

Figure 6. IRQ Alarm Interrupt Waveform



1.2.9 Oscillator Drift Compensation

To compensate for any inaccuracy of the 32 kHz oscillator, the MPU can perform a calibration of the oscillator frequency, calculate the drift compensation−versus−one-hour period, and load the compensation registers with the drift compensation value (see Figure 7).

Autocompensation is enabled by the AUTO_COMP_EN bit in the RTC_CTRL register.

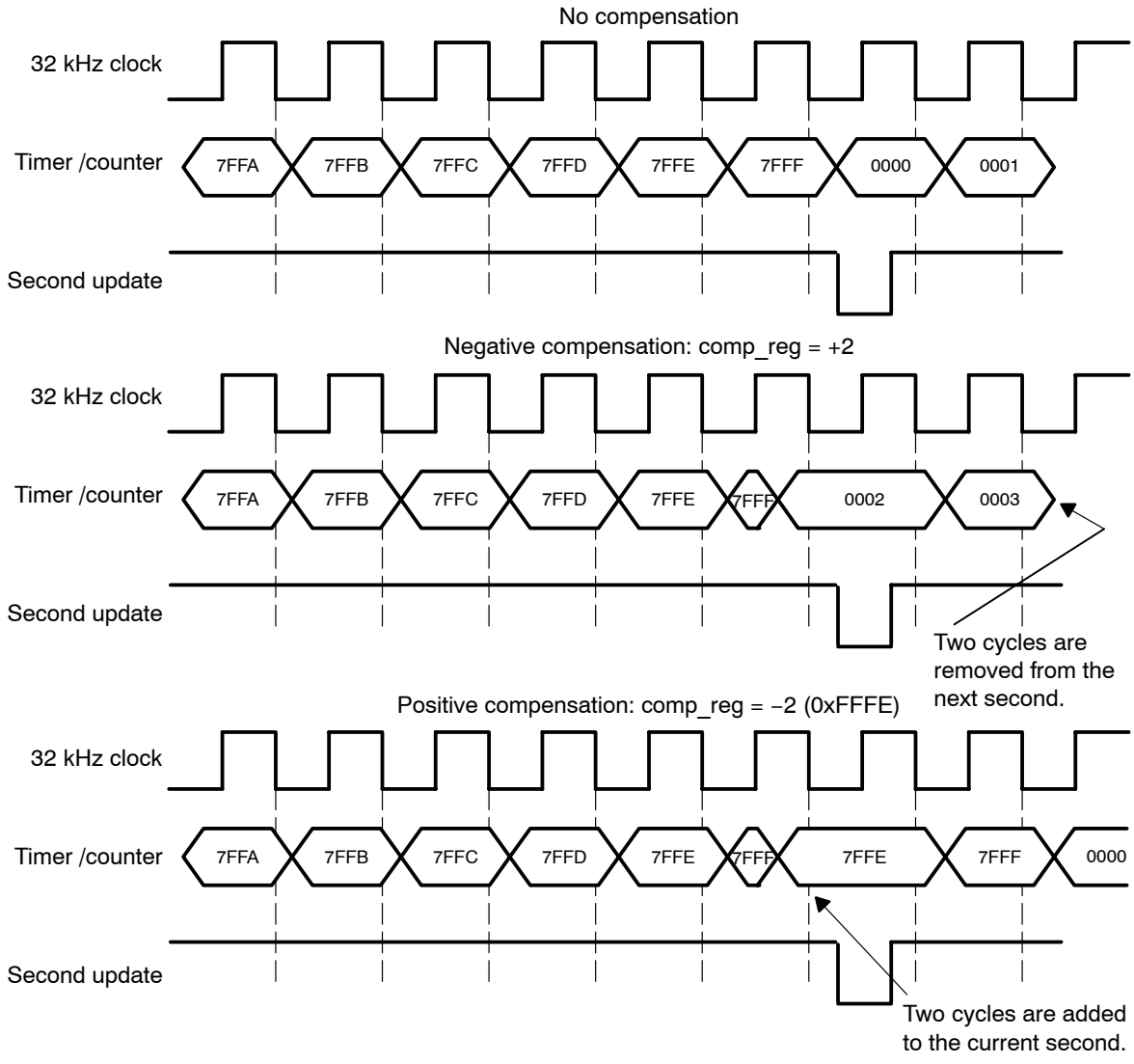
If the COMP_REG value is positive, compensation occurs after the second change event. COMP_REG cycles are removed from the next second.

If the COMP_REG value is negative, compensation occurs before the second change event. COMP_REG cycles are added to the current second.

This enables compensation with one 32 kHz period accuracy each hour.

Figure 7 summarizes the effects of positive and negative compensation.

Figure 7. Positive and Negative Compensation Effect



1.3 Register Descriptions and Mapping

Table 3 lists the RTC registers. Tables 4 through 21 describe the register bits.

Table 3. RTC Registers

Register	Description	Size	Access	Base Address	Offset Address
SECONDS_REG	Seconds	8 bits	R/W	FFFB:4800	0x00
MINUTES_REG	Minutes	8 bits	R/W	FFFB:4800	0x04
HOURS_REG	Hours	8 bits	R/W	FFFB:4800	0x08
DAYS_REG	Days	8 bits	R/W	FFFB:4800	0x0C
MONTHS_REG	Months	8 bits	R/W	FFFB:4800	0x10
YEARS_REG	Years	8 bits	R/W	FFFB:4800	0x14
WEEK_REG	Day of the week	8 bits	R/W	FFFB:4800	0x18
Reserved		8 bits		FFFB:4800	0x1C
ALARM_SECOND_REG	Alarm seconds	8 bits	R/W	FFFB:4800	0x20
ALARM_MINUTES_REG	Alarm minutes	8 bits	R/W	FFFB:4800	0x24
ALARM_HOURS_REG	Alarm hours	8 bits	R/W	FFFB:4800	0x28
ALARM_DAYS_REG	Alarm days	8 bits	R/W	FFFB:4800	0x2C
ALARM_MONTHS_REG	Alarm months	8 bits	R/W	FFFB:4800	0x30
ALARM_YEARS_REG	Alarm years	8 bits	R/W	FFFB:4800	0x34
Reserved		8 bits	R/W	FFFB:4800	0x38
Reserved		8 bits	R/W	FFFB:4800	0x3C
RTC_CTRL_REG	RTC control	8 bits	R/W	FFFB:4800	0x40
RTC_STATUS_REG	RTC status	8 bits	R/W	FFFB:4800	0x44
RTC_INTERRUPTS_REG	RTC interrupts	8 bits	R/W	FFFB:4800	0x48
RTC_COMP_LSB_REG	RTC compensation LSB	8 bits	R/W	FFFB:4800	0x4C
RTC_COMP_MSB_REG	RTC compensation MSB	8 bits	R/W	FFFB:4800	0x50

Table 4. Seconds Register (SECONDS_REG)

Bits	Field	Description	R/W	Reset Value
7	Reserved		R	0
6–4	SEC1	2nd digit of seconds Range is 0 to 5	R/W	000
3–0	SEC0	1st digit of seconds Range is 0 to 9	R/W	0000

Table 5. Minutes Register (MINUTES_REG)

Bits	Field	Description	R/W	Reset Value
7	Reserved		R	0
6–4	MIN1	2nd digit of minutes Range is 0 to 5	R/W	000
3–0	MIN0	1st digit of minutes Range is 0 to 9	R/W	0000

Table 6. Hours Register (HOURS_REG)

Bits	Field	Value	Description	R/W	Reset Value
7	PM_nAM		Only used in PM_AM mode (otherwise 0)	R	0
		0	AM		
		1	PM		
6	Reserved			R	0
5–4	HOUR1		2nd digit of hours Range is 0 to 2	R/W	00
3–0	HOUR0		1st digit of hours Range is 0 to 9	R/W	0000

Table 7. Days Register (DAYS_REG)

Bits	Field	Description	R/W	Reset Value
7–6	Reserved		R	0
5–4	DAY1	2nd digit of days Range from 0 to 3	R/W	00
3–0	DAY0	1st digit of days Range from 0 to 9	R/W	0001

Table 8. Months Register (MONTHS_REG)

Bits	Field	Description	R/W	Reset Value
7–5	Reserved		R	000
4	MONTH1	2nd digit of months Range from 0 to 1	R/W	0
3–0	MONTH0	1st digit of months Range from 0 to 9	R/W	0001

Note: Usual notation for month value: 01: January, 02: February, ... 12: December

Table 9. Years Register (YEARS_REG)

Bits	Field	Description	R/W	Reset Value
7–4	YEAR1	2nd digit of years Range from 0 to 9	R/W	0000
3–0	YEAR0	1st digit of years Range from 0 to 9	R/W	0000

Table 10. Weeks Register (WEEKS_REG)

Bits	Field	Description	R/W	Reset Value
7–3	Reserved		R	00000
2–0	WEEK	1st digit of day of the week Range from 0 to 6	R/W	000

Note: Sunday = 0; . . . ; Saturday = 6

Table 11. Alarm Seconds Register (ALARM_SECONDS_REG)

Bits	Field	Description	R/W	Reset Value
7	Reserved		R	0
6–4	ALARM_SEC1	2nd digit of seconds Range from 0 to 5	R/W	000
3–0	ALARM_SEC0	1st digit of seconds Range from 0 to 9	R/W	0000

Table 12. Alarm Minutes Register (ALARM_MINUTES_REG)

Bits	Field	Description	R/W	Reset Value
7	Reserved		R	0
6–4	ALARM_MIN1	2nd digit of minutes Range from 0 to 5	R/W	000
3–0	ALARM_MIN0	1st digit of minutes Range from 0 to 9	R/W	0000

Table 13. Alarm Hours Register (ALARM_HOURS_REG)

Bits	Field	Value	Description	R/W	Reset Value
7	ALARM_PM_nAM		Only used in PM_AM mode (otherwise 0)	R	0
		0	AM		
		1	PM		
6	Reserved			R	0
5–4	ALARM_HOUR1		2nd digit of hours Range from 0 to 2	R/W	00
3–0	ALARM_HOUR0		1st digit of hours Range from 0 to 9	R/W	0000

Table 14. Alarm Days Register (ALARM_DAYS_REG)

Bits	Field	Description	R/W	Reset Value
7–6	Reserved		R	00
5–4	ALARM_DAY1	2nd digit for days Range from 0 to 3	R/W	00
3–0	ALARM_DAY0	1st digit for days Range from 0 to 9	R/W	0001

Table 15. Alarm Months Register (ALARM_MONTHS_REG)

Bits	Field	Description	R/W	Reset Value
7–5	Reserved		R	000
4	ALARM_MONTH1	2nd digit of months Range from 0 to 1	R/W	0
3–0	ALARM_MONTH0	1st digit of months Range from 0 to 9	R/W	0001

Table 16. Alarm Years Register (ALARM_YEARS_REG)

Bits	Field	Description	R/W	Reset Value
7–4	ALARM_YEAR1	2nd digit of years Range from 0 to 9	R/W	0000
3–0	ALARM_YEAR0	1st digit of years Range from 0 to 9	R/W	0000

Table 17. RTC Control Register (RTC_CTRL_REG)

Bits	Field	Reset Value	Description	R/W
7	Reserved			R
6	RTC_DISABLE†	0	RTC enabled	R/W
		1	RTC disabled (no 32 kHz clock)	
5	SET_32_COUNTER‡	0	No action	R/W
		1	Sets the 32 kHz counter with COMP_REG (14:0) value	
4	TEST_MODE	0	Functional mode	R/W
		1	Test mode (autocompensation is enabled when 32 kHz counter reaches its end)	
3	MODE_12_24§	0	24-hour mode	R/W
		1	12-hour mode (PM/AM mode)	
2	AUTO_COMP	0	Autocompensation disabled	R/W
		1	Autocompensation enabled	
1	ROUND_30S¶	0	No update	R/W
		1	Rounding enabled	

† RTC_DISABLE can be written to 1 to disable RTC clock. Behavior is unpredictable if this bit is reset to 0 after having been set to 1.

‡ SET_32_COUNTER must only be used when the RTC is frozen. The set operation is asynchronous, which means the RTC counter is frozen to the compensation value as long as this bit is set. The correct sequence: reset STOP_RTC (freezes RTC), set SET_32_COUNTER bit, set STOP_RTC (launches RTC), then reset SET_32_COUNTER bit (3 register writes).

§ You can switch between the two modes at any time without disturbing the RTC. Read or write is always performed using the current mode.

¶ ROUND_30S is a toggle bit: MPU can only write 1, RTC clears it. If the MPU sets this bit and then reads it, the MPU reads 1 until the rounding to the closet minute is performed at the next second.

Table 17. RTC Control Register (RTC_CTRL_REG) (Continued)

Bits	Field	Reset Value	Description	R/W
0	STOP_RTC	0	RTC is frozen	R/W
		1	RTC is running	

† RTC_DISABLE can be written to 1 to disable RTC clock. Behavior is unpredictable if this bit is reset to 0 after having been set to 1.
‡ SET_32_COUNTER must only be used when the RTC is frozen. The set operation is asynchronous, which means the RTC counter is frozen to the compensation value as long as this bit is set. The correct sequence: reset STOP_RTC (freezes RTC), set SET_32_COUNTER bit, set STOP_RTC (launches RTC), then reset SET_32_COUNTER bit (3 register writes).
§ You can switch between the two modes at any time without disturbing the RTC. Read or write is always performed using the current mode.
¶ ROUND_30S is a toggle bit: MPU can only write 1, RTC clears it. If the MPU sets this bit and then reads it, the MPU reads 1 until the rounding to the closet minute is performed at the next second.

Table 18. RTC Status Register (RTC_STATUS_REG)

Bits	Field	Value	Description	R/W	Reset Value
7	POWER_UP†		Indicates that a reset occurred	R/W	1
6	ALARM‡		Indicates that an alarm interrupt has been generated	R/W	0
5	1D_EVENT		One day has occurred	R	0
4	1H_EVENT		One hour has occurred	R	0
3	1M_EVENT		One minute has occurred	R	0
2	1S_EVENT		One second has occurred	R	0
1	RUN§	0	RTC is frozen	R	0
		1	RTC is running		
0	BUSY	0	Updating event in more than 15 μs	R	0
		1	Updating event		

† POWER_UP is set by a reset and cleared by writing 1 to this bit.
‡ The alarm interrupt keeps its low level until the MPU writes 1 in the ALARM bit of this register. The timer interrupt is a low-level pulse (15 μs duration).
§ The STOP_RTC signal is synchronized on the 32 kHz clock, so only 1 clock period can elapse between the write to STOP_RTC and the RTC actually being stopped. The RUN bit shows the actual state of the RTC.

Table 19. RTC Interrupts Register (RTC_INTERRUPTS_REG)

Bits	Field	Value	Description	R/W	Reset Value
7–4	Reserved			R	0000
3	IT_ALARM		Enable one interrupt when the alarm value is reached (time and calendar alarms) by the time and calendars.	R/W	0
2	IT_TIMER		Enable periodic interrupt	R/W	0
		0	Interrupt disabled		
		1	Interrupt enabled		
1–0	EVERY		Interrupt period	R/W	00
		0	Every second		
		1	Every minute		
		2	Every hour		
		3	Every day		

Note: The MPU must respect the busy period to prevent a spurious interrupt.

Table 20. RTC Compensation LSB Register (RTC_COMP_LSB_REG)

Bits	Field	Description	R/W	Reset Value
7–0	RTC_COMP_LSB	Indicates number of 32 kHz periods to be added into the 32 kHz counter every hour.	R/W	0x00

Note: This register must be written in twos complement. That means that to add one 32 kHz oscillator period every hour, MPU must write FFFF into RTC_COMP_MSB_REG and RTC_COMP_LSB_REG. To remove one 32 kHz oscillator period every hour, MPU must write 0001 into RTC_COMP_MSB_REG and RTC_COMP_LSB_REG. The 7FFF value is forbidden.

Table 21. RTC Compensation MSB Register (RTC_COMP_MSB_REG)

Bits	Field	Description	R/W	Reset Value
7–0	RTC_COMP_MSB	Indicates number of 32 kHz periods to be added into the 32 kHz counter every hour.	R/W	0x00

Note: This register must be written in twos complement. That means that to add one 32 kHz oscillator period every hour, MPU must write FFFF into RTC_COMP_MSB_REG and RTC_COMP_LSB_REG. To remove one 32 kHz oscillator period every hour, MPU must write 0001 into RTC_COMP_MSB_REG and RTC_COMP_LSB_REG. The 7FFF value is forbidden.

Index