OMAP5912 Multimedia Processor Real-Time Clock and Split Power Reference Guide

Literature Number: SPRU782A
March 2004

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

Preface

Read This First

About This Manual

This document describes the real-time clock (RTC) block. The RTC is an embedded real-time clock module directly accessible from the TIPB bus interface.

This document also describes split power.

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.

OMAP5912 Multimedia Processor Device Overview and Architecture Reference Guide (literature number SPRU748) introduces the setup, components, and features of the OMAP5912 multimedia processor and provides a high-level view of the device architecture.

OMAP5912 Multimedia Processor OMAP 3.2 Subsystem Reference Guide (literature number SPRU749) introduces and briefly defines the main features of the OMAP3.2 subsystem of the OMAP5912 multimedia processor.

OMAP5912 Multimedia Processor DSP Sybsystem Reference Guide (literature number SPRU750) describes the OMAP5912 multimedia processor DSP subsystem. The digital signal processor (DSP) subsystem is built around a core processor and peripherals that interface with: 1) The

ARM926EJS via the microprocessor unit interface (MPUI); 2) Various standard memories via the external memory interface (EMIF); 3) Various system peripherals via the TI peripheral bus (TIPB) bridge.

OMAP5912 Multimedia Processor Clocks Reference Guide (literature number SPRU751) describes the clocking mechanisms of the OMAP5912 multimedia processor. In OMAP5912, various clocks are created from special components such as the digital phase locked loop (DPLL) and the analog phase-locked loop (APLL).

OMAP5912 Multimedia Processor Initialization Reference Guide (literature number SPRU752) describes the reset architecture, the configuration, the initialization, and the boot ROM of the OMAP5912 multimedia processor.

OMAP5912 Multimedia Processor Power Management Reference Guide (literature number SPRU753) describes power management in the OMAP5912 multimedia processor. The ultralow-power device (ULPD) generates and manages clocks and reset signals to OMAP3.2 and to some peripherals. It controls chip-level power-down modes and handles chip-level wake-up events. In deep sleep mode, this module is still active to monitor wake-up events.This book describes the ULPD module and outline architecture.

OMAP5912 Multimedia Processor Security Features Reference Guide (literature number SPRU754) describes the security features of teh OMAP5912 multimedia processor. The OMAP5912 security scheme relies on the OMAP3.2 secure mode. The distributed security on the OMAP3.2 platform is a Texas Instruments solution to address m-commerce and security issues within a mobile phone environment. The OMAP3.2 secure mode was developed to bring hardware robustness to the overall OMAP5912 security scheme.

OMAP5912 Multimedia Processor Direct Memory Access (DMA) Support Reference Guide (literature number SPRU755) describes the direct memory access support of the OMAP5912 multimedia processor. The OMAP5912 processor has three DMAs:

- The system DMA is embedded in OMAP3.2. It handles DMA transfers associated with MPU and shared peripherals.
- The DSP DMA is embedded in OMAP3.2. It handles DMA transfers associated with DSP peripherals.
- The generic distributed DMA (GDD) is an OMAP5912 resource attached to the SSI peripheral. It handles only DMA transfers associated with the SSI peripheral.

OMAP5912 Multimedia Processor Memory Interfaces Reference Guide (literature number SPRU756) describes the memory interfaces of the OMAP5912 multimedia processor.

- SDRAM (external memory interface fast, or EMIFF)
- Asynchronous and synchronous burst memory (external memory interface slow, or EMIFS)
- NAND flash (hardware controller or software controller)
- CompactFlash on EMIFS interface
- Internal static RAM

OMAP5912 Multimedia Processor Interrupts Reference Guide (literature number SPRU757) describes the interrupts of the OMAP5912 multimedia processor. Three level 2 interrupt controllers are used in OMAP5912:

- One MPU level 2 interrupt handler (also referred to as MPU interrupt level 2) is implemented outside of OMAP3.2 and can handle 128 interrupts.
- One DSP level 2 interrupt handler (also referred to as DSP interrupt level 2.1) is instantiated outside of OMAP3.2 and can handle 64 interrupts.
- One OMAP3.2 DSP level 2 interrupt handler (referenced as DSP interrupt level 2.0) can handle 16 interrupts.

OMAP5912 Multimedia Processor Peripheral Interconnects Reference Guide (literature number SPRU758) describes various periperal interconnects of the OMAP5912 multimedia processor.

OMAP5912 Multimedia Processor Timers Reference Guide (literature number SPRU759) describes various timers of the OMAP5912 multimedia processor.

OMAP5912 Multimedia Processor Serial Interfaces Reference Guide (literature number SPRU760) describes the serial interfaces of the OMAP5912 multimedia processor.

OMAP5912 Multimedia Processor Universal Serial Bus (USB) Reference Guide (literature number SPRU761) describes the universal serial bus (USB) host on the OMAP5912 multimedia processor. The OMAP5912 processor provides several varieties of USB functionality. Flexible multiplexing of signals from the OMAP5912 USB host controller, the OMAP5912 USB function controller, and other OMAP5912 peripherals allow a wide variety of system-level USB capabilities. Many of the OMAP5912 pins can be used for USB-related signals or for signals from other OMAP5912 peripherals. The OMAP5912 top-level pin multiplexing

controls each pin individually to select one of several possible internal pin signal interconnections. When these shared pins are programmed for use as USB signals, the OMAP5912 USB signal multiplexing selects how the signals associated with the three OMAP5912 USB host ports and the OMAP5912 USB function controller can be brought out to OMAP5912 pins.

OMAP5912 Multimedia Processor Multi-channel Buffered Serial Ports (McBSPs) Reference Guide (literature number SPRU762) describes the three multi-channel buffered serial ports (McBSPs) available on the OMAP5912 device. The OMAP5912 device provides multiple high-speed multichannel buffered serial ports (McBSPs) that allow direct interface to codecs and other devices in a system.

OMAP5912 Multimedia Processor Camera Interface Reference Guide (literature number SPRU763) describes two camera inerfaces implemented in the OMAP5912 multimedia processor: compact serial camera port and camera parallel interface.

OMAP5912 Multimedia Processor Display Interface Reference Guide (literature number SPRU764) describes the display interface of the OMAP5912 multimedia processor.

- LCD module
- LCD data conversion module
- LED pulse generator
- Display interface

OMAP5912 Multimedia Processor Multimedia Card (MMC/SD/SDIO) (literature number SPRU765) describes the multimedia card (MMC) interface of the OMAP5912 multimedia processor. The multimedia card/secure data/secure digital IO (MMC/SD/SDIO) host controller provides an interface between a local host, such as a microprocessor unit (MPU) or digital signal processor (DSP), and either an MMC or SD memory card, plus up to four serial flash cards. The host controller handles MMC/SD/SDIO or serial port interface (SPI) transactions with minimal local host intervention.

OMAP5912 Multimedia Processor Keyboard Interface Reference Guide (literature number SPRU766) describes the keyboard interface of the OMAP5912 multimedia processor. The MPUIO module enables direct I/O communication between the MPU (through the public TIPB) and external devices. Two types of I/O can be used: specific I/Os dedicated for 8 x 8 keyboard connection, and general-purpose I/Os.

OMAP5912 Multimedia Processor General-Purpose Interface Reference Guide (literature number SPRU767) describes the general-purpose in-

terface of the OMAP5912 multimedia processor. There are four GPIO modules in the OMAP5912. Each GPIO peripheral controls 16 dedicated pins configurable either as input or output for general purposes. Each pin has an independent control direction set by a programmable register. The two−edge control registers configure events (rising edge, falling edge, or both edges) on an input pin to trigger interrupts or wake−up requests (depending on the system mode). In addition, an interrupt mask register masks out specified pins. Finally, the GPIO peripherals provide the set and clear capabilities on the data output registers and the interrupt mask registers. After detection, all event sources are merged and a single synchronous interrupt (per module) is generated in active mode, whereas a unique wake−up line is issued in idle mode. Eight data output lines of the GPIO3 are ORed together to generate a global output line at the OMAP5912 boundary. This global output line can be used in conjunction with the SSI to provide a CMT−APE interface to the OMAP5912.

OMAP5912 Multimedia Processor VLYNQ Serial Communications Interface Reference Guide (literature number SPRU768) describes the VLYNQ of the OMAP5912 multimedia processor.

VLYNQ is a serial communications interface that enables the extension of an internal bus segment to one or more external physical devices. The external devices are mapped into local, physical address space and appear as if they are on the internal bus of the OMAP 5912. The external devices must also have a VLYNQ interface. The VLYNQ module serializes bus transactions in one device, transfers the serialized data between devices via a VLYNQ port, and de-serializes the transaction in the external device.

OMAP5912 includes one VLYNQ module connected on OCPT2 target port and OCPI initiator port. These connections are configured via a static switch, which selects either SSI or VLYNQ module. This switch, forbids the simultaneous use of GDD/SSI and VLYNQ. The switch is controlled by the VLYNQ_EN bit in the OMAP5912 configuration control register (CONF_5912_CTRL).

OMAP5912 Multimedia Processor Pinout Reference Guide (literature number SPRU769) provides the pinout of the OMAP5912 multimedia processor. After power-up reset, the user can change the configuration of the default interfaces. If another interface is available on top of the default, it is possible to enable a new interface for each ball by setting the corresponding 3-bit field of the associated FUNC_MUX_CTRL register. It is also possible to configure on-chip pullup/pulldown. This document

also describes the various power domains so that the user can apply the different interfaces seamlessly with external components.

OMAP5912 Multimedia Processor Window Tracer (WT) Reference Guide (literature number SPRU770) describes the window tracer module used to capture the memory transactions from four interfaces: EMIFF, EMIFS, OCP-T1, and OCP-T2. This module is located in the OMAP3.2 traffic controller (TC).

OMAP5912 Multimedia Processor Real-Time Clock Reference Guide (literature number SPRUxxx) describes the real-time clock of the OMAP5912 multimedia processor. The real-time clock (RTC) block is an embedded real-time clock module directly accessible from the TIPB bus interface.

Trademarks

Contents

Figures

Tables

Real-Time Clock (RTC)

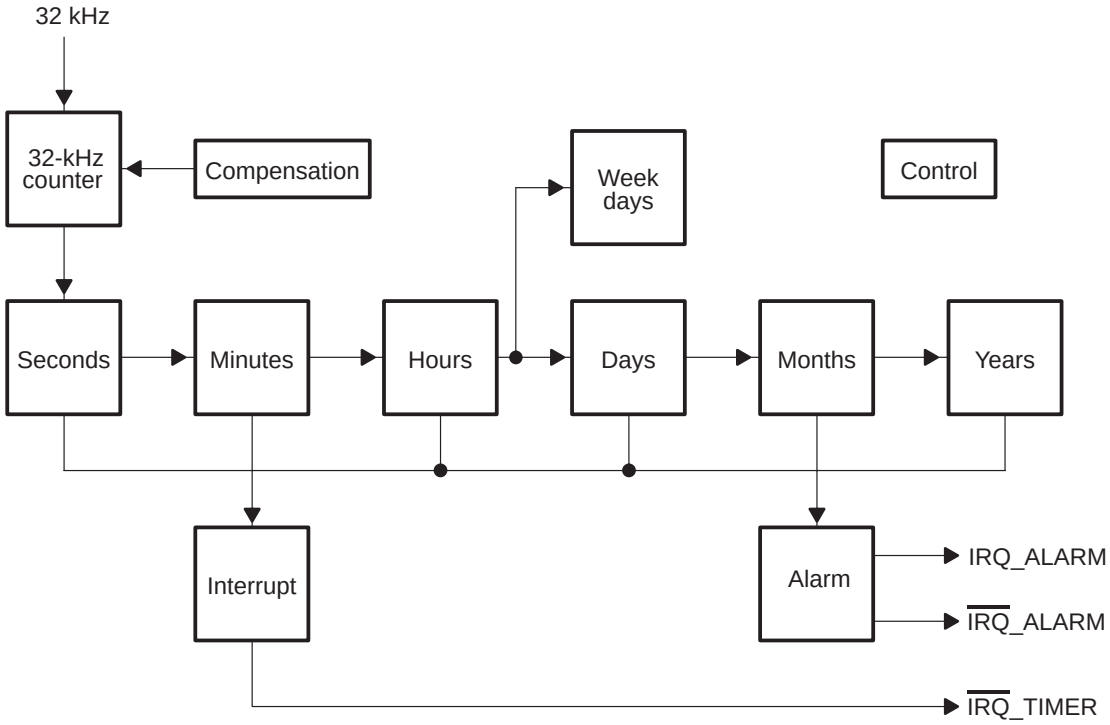
1 Overview

The real-time clock (RTC) block is an embedded real-time clock module directly accessible from the TIPB bus interface. The basic functions of RTC block are:

- Time information (seconds/minutes/hours) directly in binary coded decimal (BCD) code
- Calendar information (day/month/year/day of the week) directly in BCD code up to the year 2099
- Interrupt generation, periodically (1s/1m/1h/1d period) or at a precise time of the day (alarm function)
- 30-s time correction
- Oscillator frequency calibration

Figure 1 shows the real-time clock block.

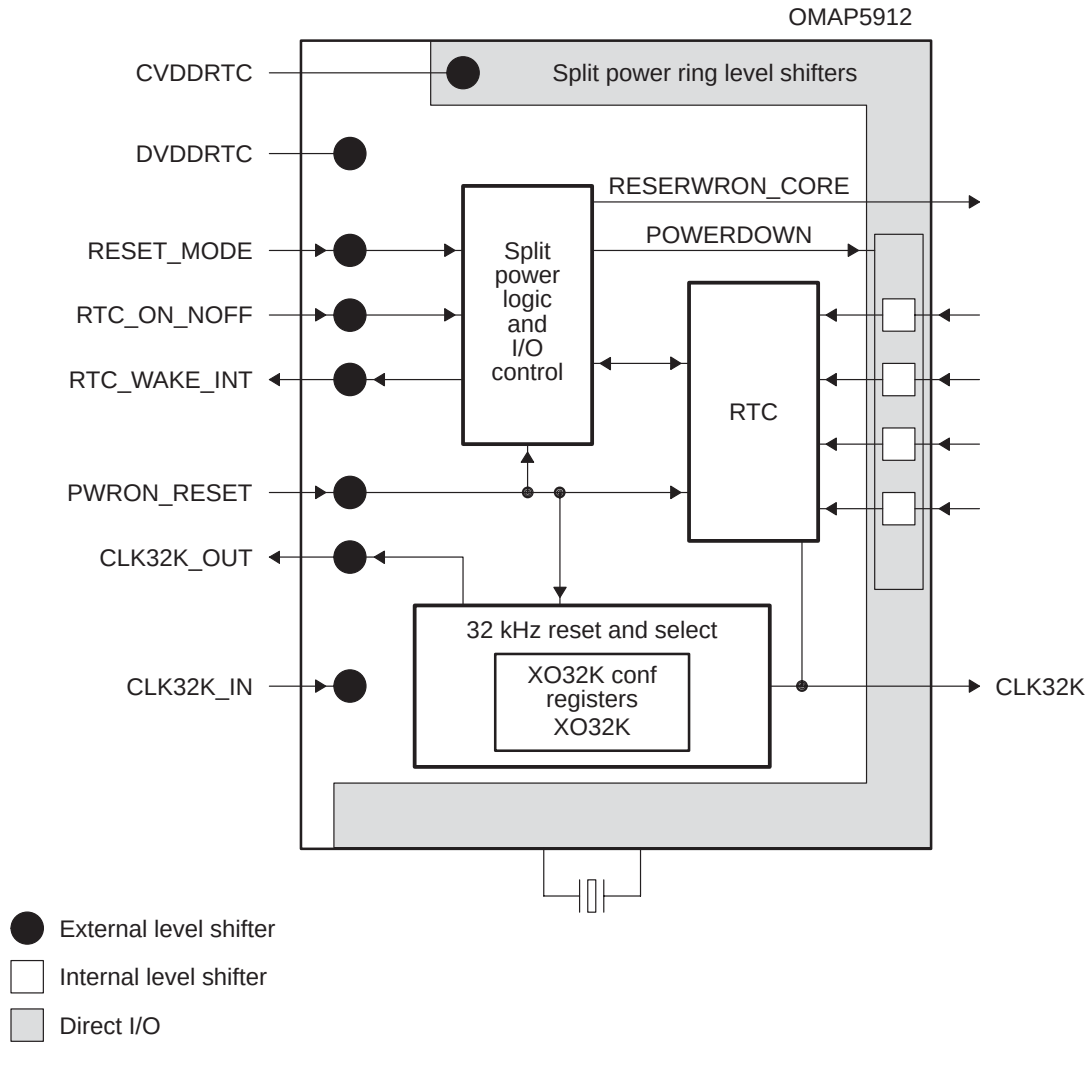
Figure 1. Real-Time Clock



2 Split Power Overview

To achieve minimum consumption in the OFF state in device equipment, some active logic elements are supplied. Those elements are the real-time clock (RTC) and the 32-kHz oscillator (OSC32K) in the digital baseband (DBB), and the power-on reset (POR) and the dedicated regulator in the analog baseband (ABB). This approach is possible when using split power, which splits the core power domain into two subdomains powered with different voltage supplies. Internal level shifters handle the separation between the core and the active domain.

Figure 2. Split Power System in OMAP5912



3 Internal Level Shifters

Internal level shifters are library-standard macrocells that interface two core domains powered with two different voltage supplies.

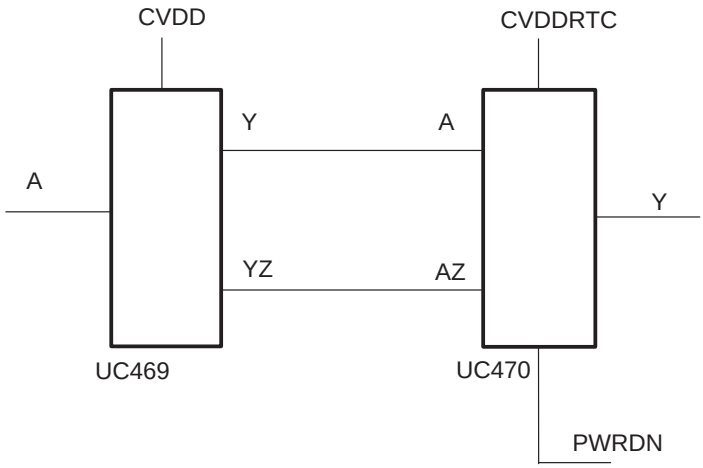
The cell can be seen as a means to isolate one power domain from an other. In the case where one domain is shut down. the level shifters ensure a known state at the boundary of the two domains.

The level shifters are divided into two parts:

- UC469: Powered by the primary power supply. Because of the input signal A, it creates Y and YZ, which are A buffered and A inverted, respectively.

- UC470: Powered by the secondary power supply. Because of the differential signals, A and AZ (given by UC469), it creates the signal Y that is equivalent to the input of UC469 but in the second core supply domain. This cell has a PWRDN signal so that when power supply of UC469 does not exist (input signals A and AZ are ambiguous), this cell does not have any through-current and the output is set to 0.

Figure 3. Internal Level Shifter



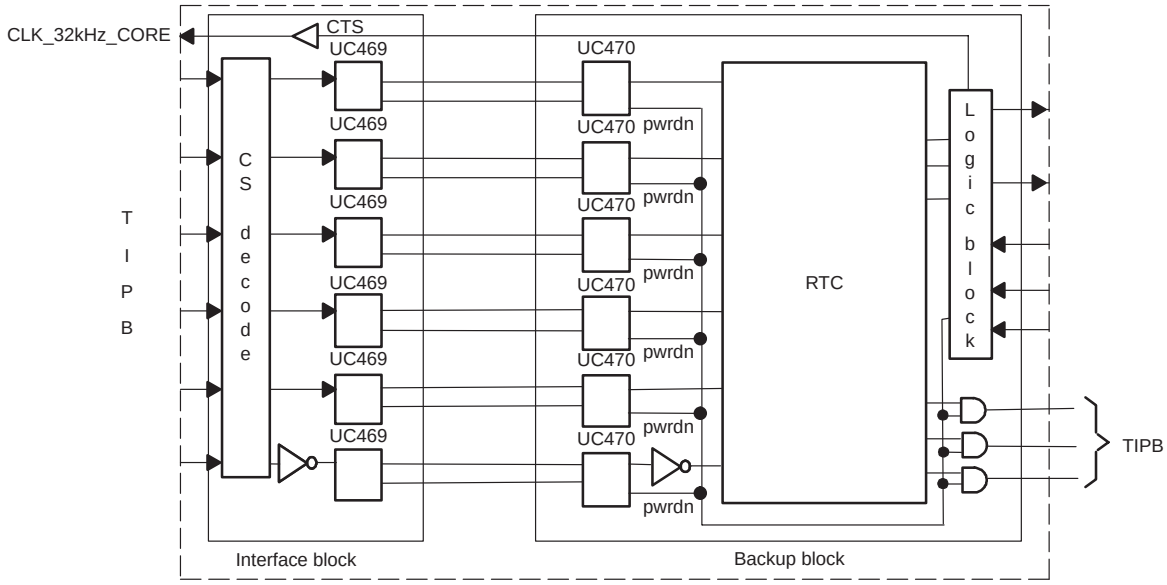
PWRDN = 0 => Y=A

PWRDN =1 => Y= 0

4 Split Power Block

The split power module contains two blocks: the interface block and the backup block. The interface block, between the core and the backup elements, is supplied by the core power supply CVDD. The backup block contains the RTC and some logic.

Figure 4. Split Power Block Diagram



4.1 Interface Block

This block separates the backup elements and the ASIC core. It also contains the UC469 and some logic. The logic allows masking of the TIPB bus signals to avoid consumption of internal level shifters, except in RTC accesses.

4.2 Backup Block

This block contains the elements kept in OFF mode in the DBB, the RTC, the 32-kHz clock, the UC470, and the logic to force the DBB to OFF mode. The module input/outputs are RTC_ON_NOFF, $\overline{\text{PWRON\_RESET}}$, RTC_WAKE_INT, RESET_MODE, and POWERDOWN.

5 Output Control

To keep them from supplying the ASIC core, the split power block outputs are forced to logical zero.

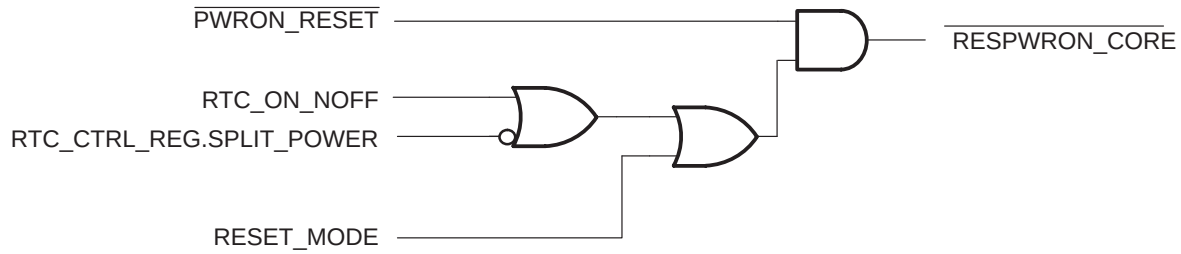
5.1 Backup Signal Management

In OFF mode, only the split power module is supplied. The ON_OFF, RTC_WAKE_INT, and PWRON_RESET signals, which control the activity management of the DBB, are also active.

6 On-Chip Reset Generation

- If RTC_CTRL_REG.SPLIT_POWER = 0 or RESET_MODE = 1 then the OMAP5912 is reset only by PWRON_RESET.:
- If RESET_MODE = 0 and RTC_CTRL_REG.SPLIT_POWER= 1 then PWRON_RESET AND RTC_ON_NOFF will reset OMAP5912.:

Figure 5. ASIC Reset Scheme



6.1 Resets for OMAP5912 Device With Split Power Feature Enabled

The RTC module and its internal real time counter are reset by PWRON_RESET during power up. OMAP5912 ASIC gates are reset by PWRON_RESET_CORE.

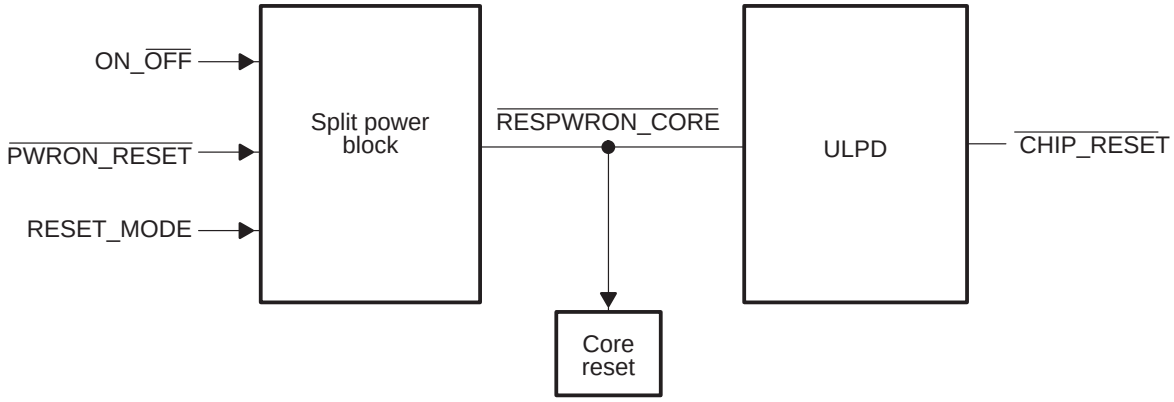
Subsequent resets are asserted with RTC_ON_NOFF. The RTC module is not reset by the assertion of RTC_ON_NOFF. While RTC_ON_NOFF is asserted low, the system powers down OMAP5912 ASIC gates.

6.2 Resets for OMAP5912 Device With Split Power Feature Not Used

For OMAP5912 devices that are forced during reset in reset mode 1, PWRON_RESET_CORE is logically equal to PWRON_RESET and only to PWRON_RESET.

For OMAP5912 devices that are forced during reset in reset mode 0 (OMAP1510 legacy) and no split power, PWRON_RESET_CORE is logically equal to PWRON_RESET and only to PWRON_RESET.

Figure 6. $\overline{\text{PWRON\_RESET}}$ Connection



RTC_WAKE_INT

The RTC_WAKE_INT pin now collects the interrupt sources used to awaken the ULPD from deep sleep. It is gated with RTC_CTRL_REG.SPLIT_POWER and the RTC alarm. In OFF mode the IRQ_SET is set to 0, and only the IRQ_ALARM_EXT can generate an interrupt on the pin RTC_WAKE_INT.

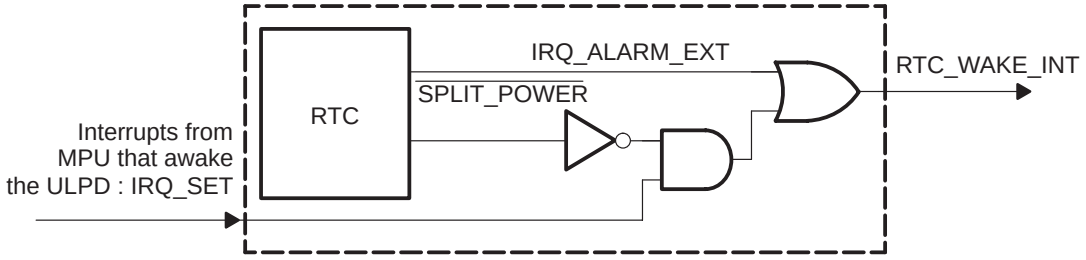
With a device in ON state, both IRQ_ALARM_EXT and IRQ_SET can generate an RTC_WAKE_INT.

The RTC_CTRL_REG.SPLIT_POWER signal generates RTC_WAKE_INT to avoid an RTC_WAKE_INT in ON mode for the ABB.

Figure 7 shows the generation of the RTC_WAKE_INT.

In the OMAP5912 core, IRQ_SET is not used and is set to 0. RTC_WAKE_INT is used only for the ABB.

Figure 7. RTC_WAKE_INT Generation

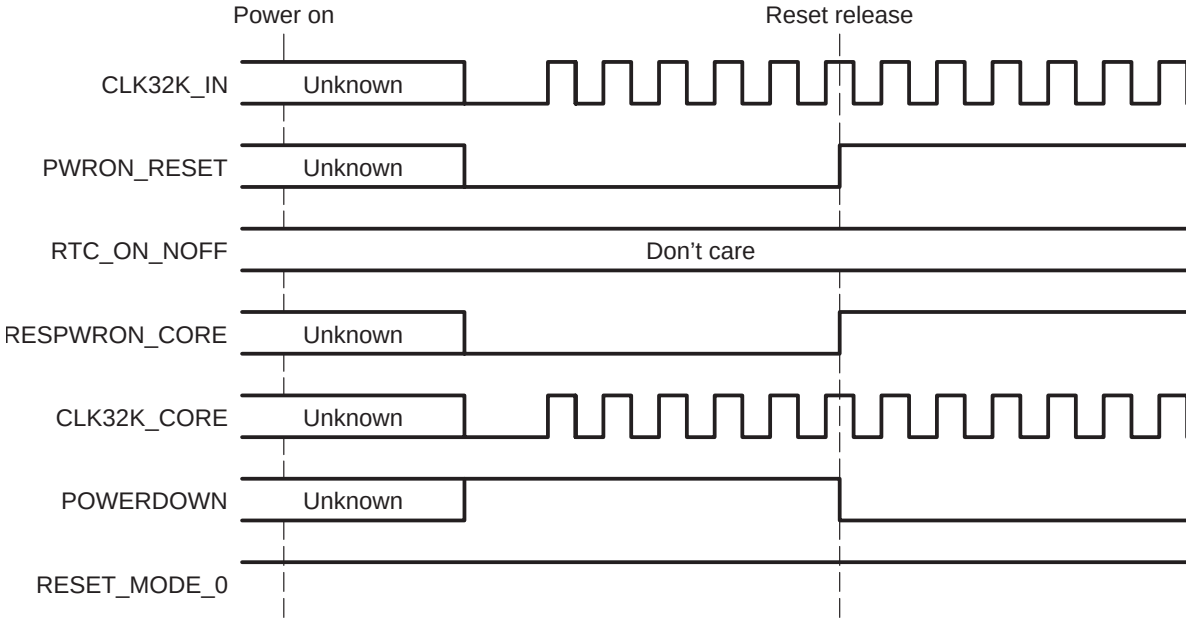


7 Using Split Power

7.1 ABB Functions Incompatible With Split Power

Do not use split power. The power cannot be cut in the core.

Figure 8. OMAP5912 in RESET_MODE = 1 or RTC_CTRL_REG.SPLIT_POWER = 0



7.2 ABB Functions Compatible With Split Power

Figure 9. Startup With RTC_CTRL_REG.SPLIT_POWER = 1 for OMAP5912 RESET_MODE = 0

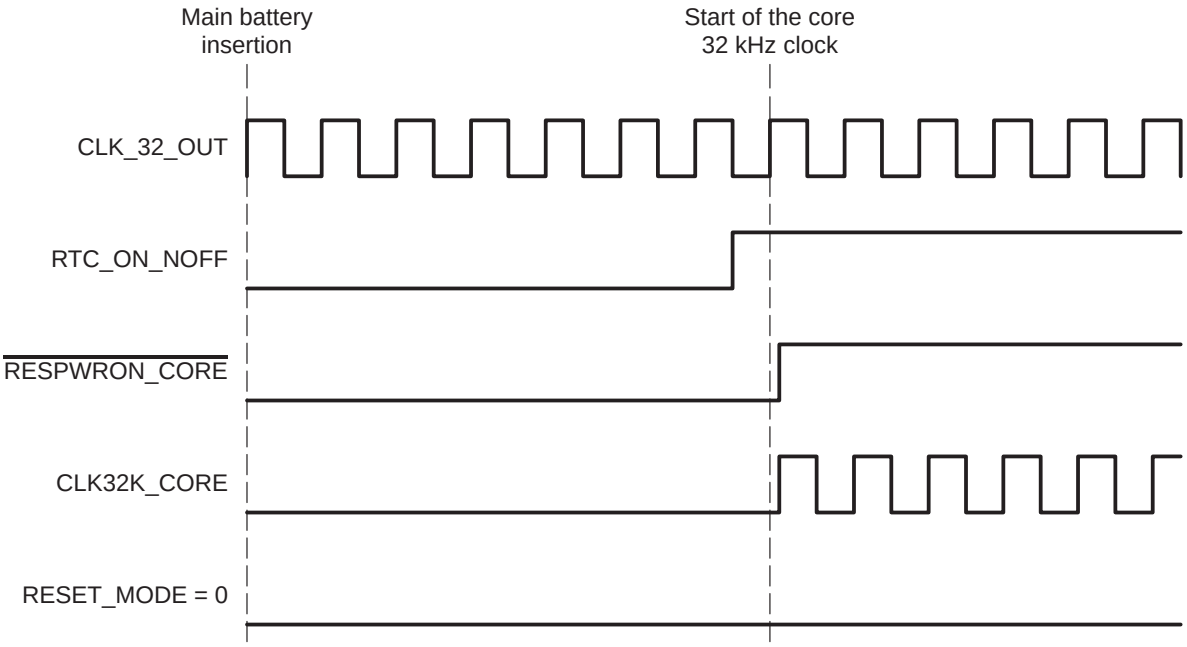
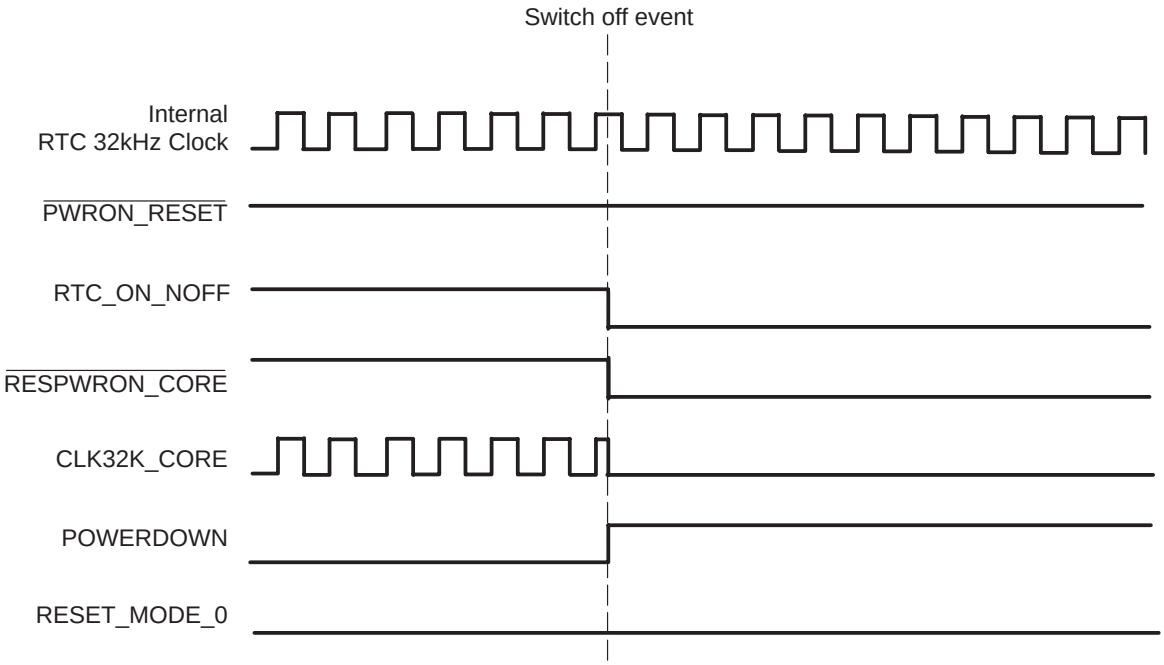


Figure 9 assumes that backup battery is already inserted, only the RTC power domain is powered and that the RTC is isolated from the core (RTC_ON_NOFF ball is held low, and SPLIT_POWER bit in RTC is set to 1). Once ‘the main battery is plugged in, the core domain is reset by $\overline{\text{PWRON_RESET}}$_CORE until RTC_ON_NOFF becomes high; the isolation mode also becomes inactive. The ULPD state machine can start.

7.3 ON to OFF Description

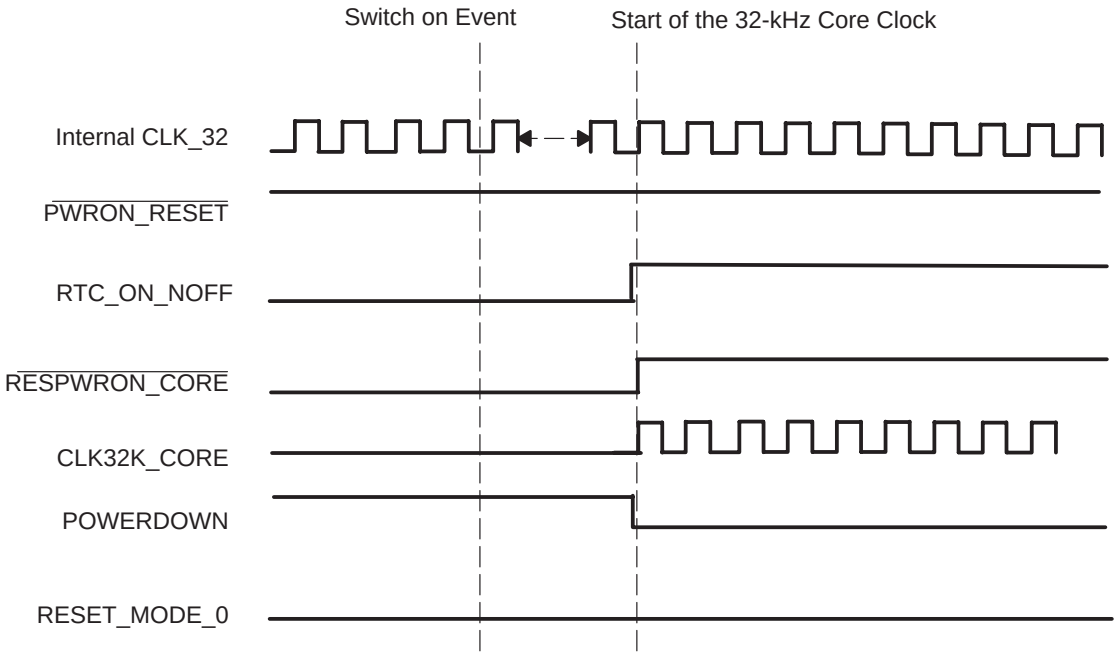
Figure 10. ON to OFF With SPLIT POWER = 1 for OMAP5912 RESET_MODE = 0



On a switch-off event, RTC_ON_NOFF is set to 0. The 32-kHz clock is not fed to OMAP5912 core anymore, and PWRON_RESET_CORE is set to 0 to indicate that the device goes into OFF state. PWRON_RESET_CORE becomes active, and the DBBcore is reset. POWERDOWN becomes active and the backup module is isolated. The ABB circuit disables the CORE LDO regulators, and thus the DBB core is not powered.

7.4 OFF to ON Description

Figure 11. OFF to ON With RTC_CTRL_REG.SPLIT_POWER = 1 for OMAP5912 RESET_MODE = 0



On a switch-on event, the ASIC LDO regulators are enabled. The DBB core is also supplied and reset by PWRON_RESET_CORE until RTC_ON_NOFF is set to 1. Then, the isolation mode is inactive, the ULPD state machine receives PWRON_RESET_CORE, and the clock starts its state machine.

8 Interrupt Management

RTC can generate three interrupts:

- A timer interrupt (IRQ_TIMER),
- An alarm interrupt (IRQ_ALARM_CHIP)
- An alarm interrupt external (IRQ_ALARM_EXT or IRQ_ALARM_CHIP inverted)

8.1 Timer Interrupt

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

The IT_TIMER bit of the interrupt register enables this interrupt.

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

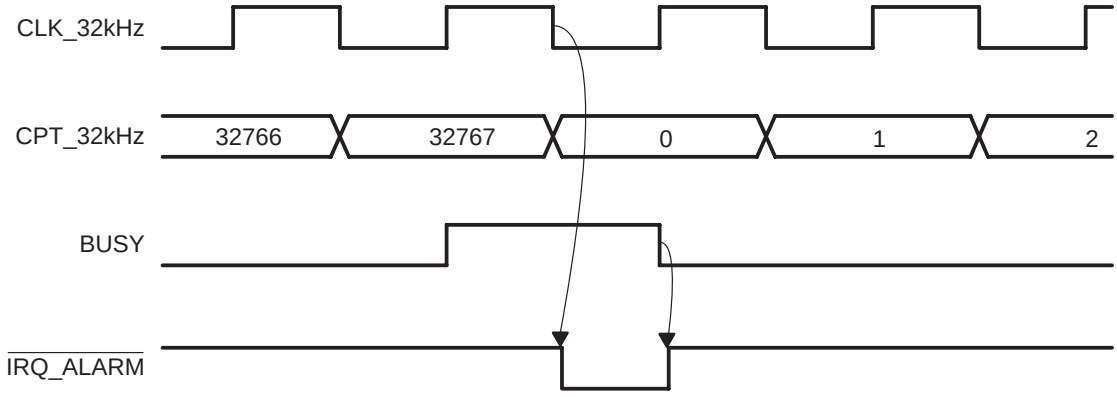
The RTC_STATUS_REG[5:2] are only updated at each new interrupt and show the events that have occurred, according to Table 1.

Table 1. Timer Interrupt Events

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 periodical period.

Figure 12. Periodic Interrupt



8.2 Alarm Interrupt

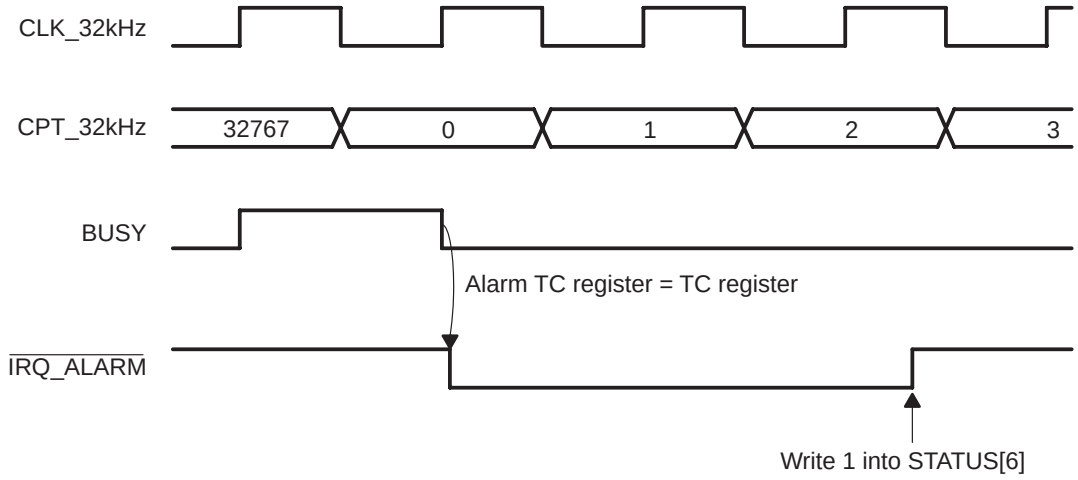
IRQ_ALARM_CHIP interrupt can be generated when the time set into the time and calendar ALARM registers is identical in the time and calendar registers.

This interrupt is then generated if the IT_ALARM bit of the interrupt 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 13. Alarm Interrupt



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.

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

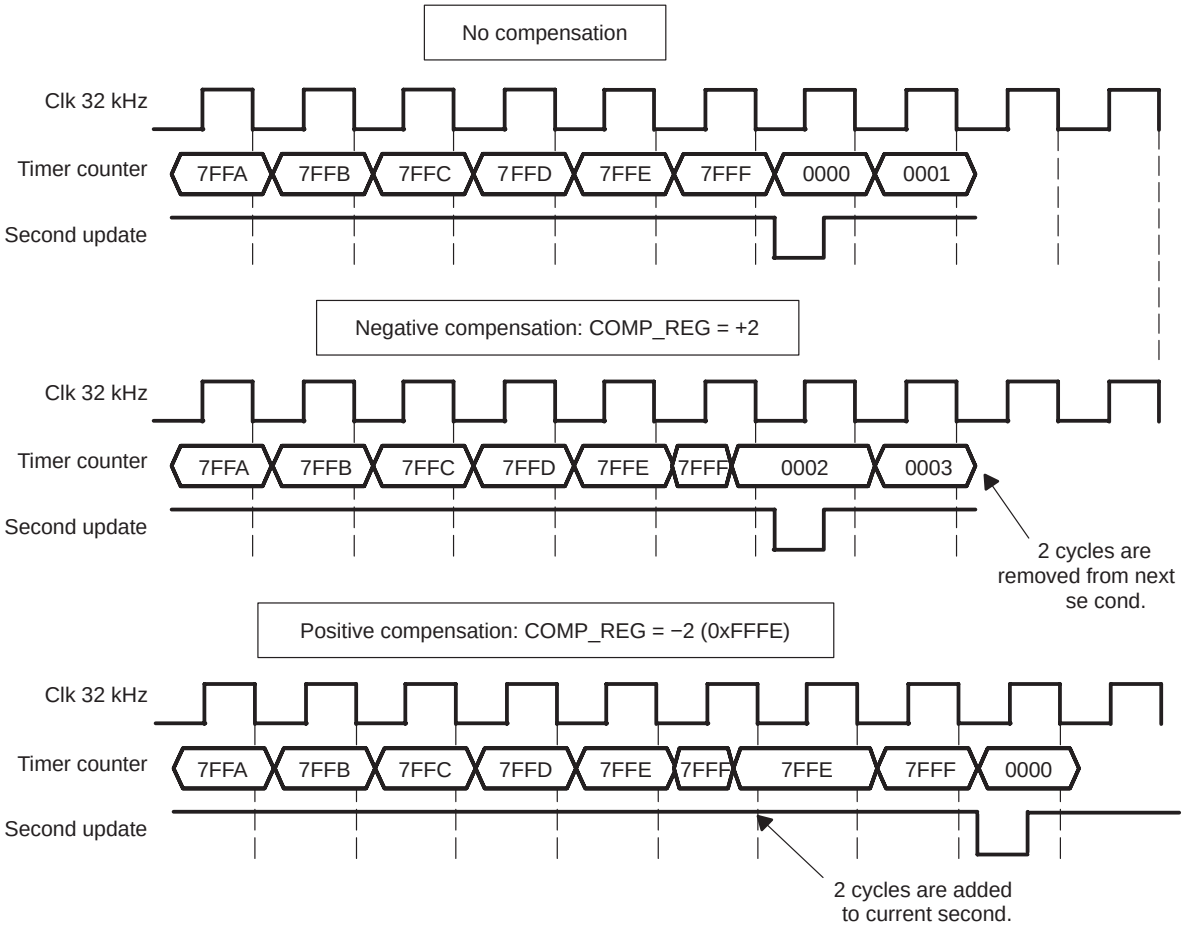
If the COMP_REG value is positive, compensation occurs *after* the second change event. COMP_REG cycles are removed from the next second.

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

This compensation enables a 32-kHz period accuracy each hour.

The waveform in Figure 14 summarizes the positive and negative compensation effect.

Figure 14. Oscillator Drift Compensation



10 Split Power Compatibility

The RTC and the 32-kHz oscillator are the only elements that must be active in the device in OFF state. Therefore, the RTC has been modified to use split power. One register has been modified (RTC_CTRL_REG), and one register has been added (RTC_RES_PROG_REG).

In the RTC_CTRL_REG, the $\overline{\text{RTC\_CTRL\_REG.SPLIT\_POWER}}$ bit has been added so the user can choose whether to use the power split mode.

The RTC_RES_PROG_REG has been added to back up the resistance value of the oscillator in OFF state.

A power-down signal has been added to set the outputs to 0 in OFF.

11 RTC Registers

There are three types of registers:

- Time and calendar, and time and calendar alarm
- General
- Compensation

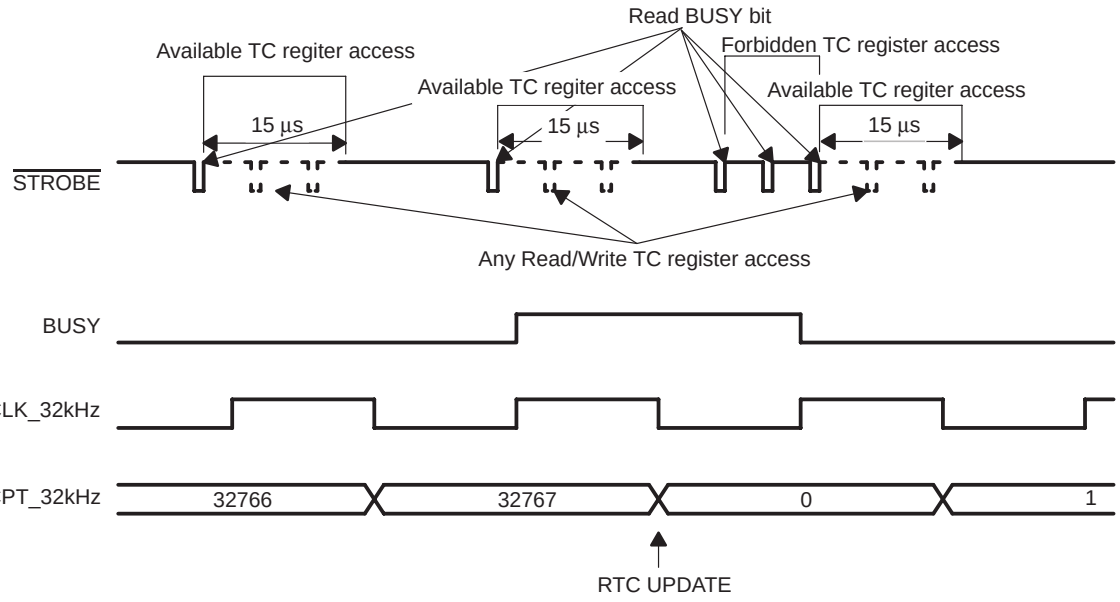
These three types have their own access constraints.

11.1 Time and Calendar Registers and Time and Calendar Alarm Registers

To read or write correct data from/to the time and calendar registers and time and calendar alarm registers, the MPU must first read the BUSY bit of the STATUS register until BUSY is equal to zero. From this time, and for a time of 15 μs (the available access period), the MPU can perform several accesses into the time and calendar registers and time and calendar alarm registers with guaranteed read/write data. At the end of one available access period, the MPU must restart the previous sequence. If the MPU accesses the time and calendar registers during an unavailable access period, access is not ensured.

To remove any possibility of interrupting the registers read process, thus introducing a potential risk of violating the authorized 15−μs access period, it is strongly recommended that the user disable all incoming interrupts during the register read process.

Figure 15. Time and Calendar Register and Time and Calendar Alarm Register Access



11.2 General Registers

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

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

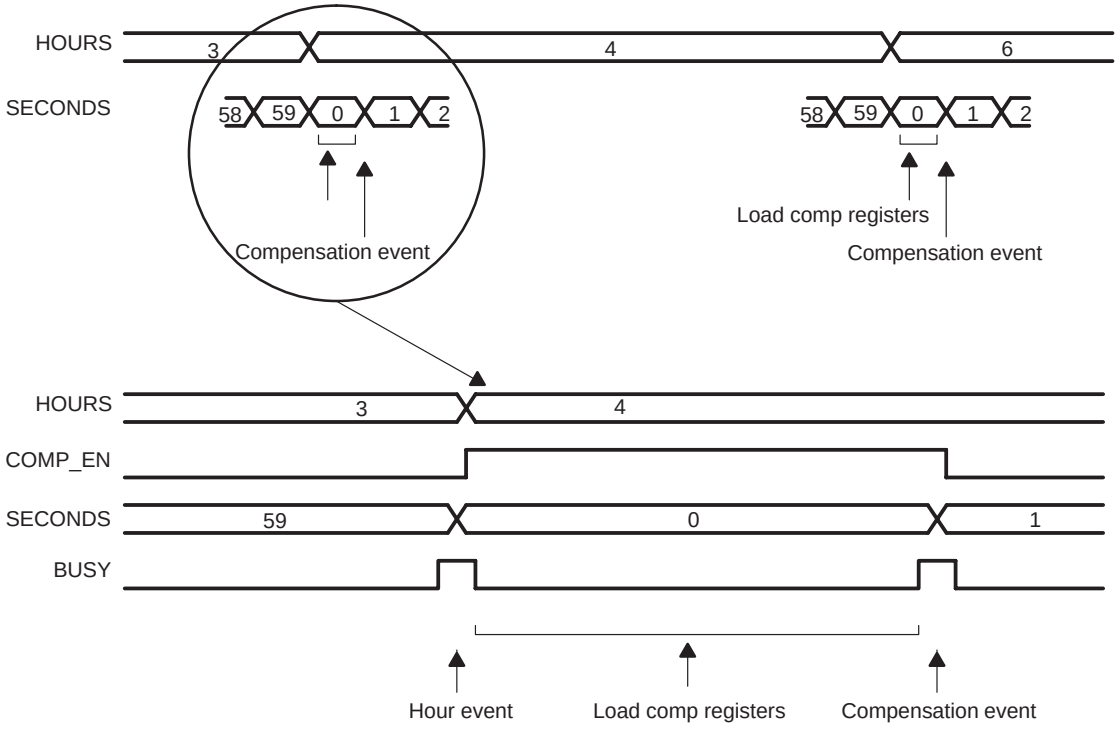
The RTC_DISABLE bit of the CTRL register must be used only to completely disable the RTC function. When this bit is set, the 32-kHz clock is gated, and the RTC is frozen. From this point, resetting this bit to zero can lead to unexpected behavior. This bit must only be used if the RTC function is unwanted in the application, to save power.

11.3 Compensation Registers

Access to the COMP_MSB_REG and COMP_LSB_REG registers must respect the available access period. These registers must not be updated during compensation (first second of each hour), but they can be updated during the second access period preceding a compensation event.

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

Figure 16. Compensation Scheduling



12 Setting Time and Calendar Information

12.1 Modify Time and Calendar Registers

To modify the current time, the MPU writes the new time into time and calendar registers to fix the time/calendar information. The MPU can write into time and calendar registers without stopping the RTC, but in this case the MPU must read the status register to be sure that the RTC updating takes place in more than 15 µs (bit BUSY must be at 0). The MPU must perform all changes in less than 15 µs, to prevent partial updating between the beginning and the end of the writingsequence into time and calendar registers.

Also, 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), updating the time and calendar values, and restarting the RTC by resetting STOP_RTC bit.

12.2 Rounding Seconds

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

Example:

If current time is 10H59M45S, round operation changes time to 11H00M00S.

If current time is 10H59M29S, round operation changes time to 10H59M00S.

Table 2 lists the RTC registers. The tables below describe the register bits. The RTC register types are grouped as follows:

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

Table 2. RTC Registers

Base Address = 0xFFFB 4800		
Name	**Description**	**Address Offset**
SECONDS_REG	Seconds	0x00
MINUTES_REG	Minutes	0x04
HOURS_REG	Hours	0x08
DAYS_REG	Days	0x0C
MONTHS_REG	Months	0x10
YEARS_REG	Years	0x14
WEEKS_REG	Weeks	0x18
RESERVED	Reserved	0x1C
ALARM_SECONDS_REG	Alarm seconds	0x20
ALARM_MINUTES_REG	Alarm minutes	0x24
ALARM_HOURS_REG	Alarm hours	0x28
ALARM_DAYS_REG	Alarm days	0x2C
ALARM_MONTHS_REG	Alarm months	0x30
ALARM_YEARS_REG	Alarm years	0x34
RESERVED	Reserved	0x38

Table 2. RTC Registers (Continued)

Base Address = 0xFFFB 4800		
Name	**Description**	**Address Offset**
RESERVED	Reserved	0x3C
RTC_CTRL_REG	RTC control	0x40
RTC_STATUS_REG	RTC status	0x44
RTC_INTERRUPTS_REG	RTC interrupt	0x48
RTC_COMP_LSB_REG	RTC compensation LSB	0x4C
RTC_COMP_MSB_REG	RTC compensation MSB	0x50
RTC_OSC_REG	RTC oscillator	0x54

12.2.1 Time and Calendar Registers

The time and calendar information is available in dedicated registers, called time and calendar registers. These register values are written in binary coded decimal (BCD) code.

Table 3. Time and Calendar Register Time Units

Time Unit	Range	Remarks
Year	00 to 99	Leap year: year divisible by four
		Common year: other years
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	Weekday
Hour	00 to 23	00 to 23 in 24 hours mode
		01 to 12 in AM/PM mode

Table 3. Time and Calendar Register Time Units (Continued)

Minutes	00 to 59
Seconds	00 to 59

Table 4. Seconds Register (SECONDS_REG)

Base Address = 0xFFFB 4800, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	SEC1	2nd digit of seconds Range is 0 to 5.	R/W	0000
3:0	SEC0	1st digit of seconds Range is 0 to 9.	R/W	0000

Table 5. Minutes Register (MINUTES_REG)

Base Address = 0xFFFB 4800, Offset = 0x04				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	MIN1	2nd digit of minutes Range is 0 to 5.	R/W	0000
3:0	MIN0	1st digit of minutes Range is 0 to 9.	R/W	0000

Table 6. Hours Register (HOURS_REG)

Base Address = 0xFFFB 4800, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
7	PM_$\overline{\text{AM}}$	Only used in PM_AM mode (otherwise 0) 0: AM 1: PM	R/W	0
6:4	HOUR1	2nd digit of hours Range is 0 to 2.	R/W	000
3:0	HOUR0	1st digit of hours Range is 0 to 9.	R/W	0000

Table 7. Days Register (DAYS_REG)

Table 7. Days Register (DAYS_REG) (Continued)

Base Address = 0xFFFB 4800, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	DAY1	2nd digit of days Range is 0 to 3.	R/W	0000
3:0	DAY0	1st digit of days Range is 0 to 9.	R/W	0001

Table 8. Months Register (MONTHS_REG)

Base Address = 0xFFFB 4800, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	MONTH1	2nd digit of months Range is 0 to 1.	R/W	0000
3:0	MONTH0	1st digit of months Range is 0 to 9.	R/W	0001

Usual notation taken for the month value:

01: January
02: February
....
12: December

Table 9. Years Register (YEARS_REG)

Base Address = 0xFFFB 4800, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	YEAR1	2nd digit of years Range is 0 to 9.	R/W	0000
3:0	YEAR0	1st digit of years Range is 0 to 9.	R/W	0000

Table 10. Weeks Register (WEEKS_REG)

Base Address = 0xFFFB 4800, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
3:0	WEEK	1st digit of days in a week Range is 0 to 6.	R/W	0000

Table 11. Reserved

Base Address = 0xFFFB 4800, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
???:0	RESERVED	Reserved		

Table 12. Alarm Seconds Register (ALARM_SECONDS_REG)

Base Address = 0xFFFB 4800, Offset = 0x20				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	ALARM_SEC1	2nd digit of seconds Range is 0 to 5.	R/W	0000
3:0	ALARM_SEC0	1st digit of seconds Range is 0 to 9.	R/W	0000

Table 13. Alarm Minutes Register (ALARM_MINUTES_REG)

Base Address = 0xFFFB 4800, Offset = 0x24				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	ALARM_MIN1	2nd digit of minutes Range is 0 to 5.	R/W	0000
3:0	ALARM_MIN0	1st digit of minutes Range is 0 to 9.	R/W	0000

Table 14. Alarm Hours Register (ALARM_HOURS_REG)

Base Address = 0xFFFB 4800, Offset = 0x28				
Bit	**Name**	**Function**	**R/W**	**Reset**
7	ALARM_PM_$\overline{AM}$	Only used in PM_AM mode (otherwise 0) 0: AM 1: PM	R/W	0
6:4	ALARM_HOUR1	2nd digit of hours Range is 0 to 2.	R/W	000
3:0	ALARM_HOUR0	1st digit of hours Range is 0 to 9.	R/W	0000

Table 15. Alarm Days Register (ALARM_DAYS_REG)

Base Address = 0xFFFB 4800, Offset = 0x2C				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	ALARM_DAY1	2nd digit for days Range is 0 to 3.	R/W	0000
3:0	ALARM_DAY0	1st digit for days Range is 0 to 9.	R/W	0001

Table 16. Alarm Months Register (ALARM_MONTHS_REG)

Base Address = 0xFFFB 4800, Offset = 0x30				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	ALARM_MONTH1	2nd digit of months Range is 0 to 1.	R/W	0000
3:0	ALARM_MONTH0	1st digit of months Range is 0 to 9.	R/W	0001

Table 17. Alarm Years Register (ALARM_YEARS_REG)

Base Address = 0xFFFB 4800, Offset = 0x34				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	ALARM_YEAR1	2nd digit of years Range is 0 to 9.	R/W	0000

Table 17. Alarm Years Register (ALARM_YEARS_REG) (Continued)

Base Address = 0xFFFB 4800, Offset = 0x34				
Bit	**Name**	**Function**	**R/W**	**Reset**
3:0	ALARM_YEAR0	1st digit of years Range is 0 to 9.	R/W	0000

Table 18. RTC Control Register (RTC_CTRL_REG)

Base Address = 0xFFFB 4800, Offset = 0x40				
Bit	**Name**	**Function**	**R/W**	**Reset**
7	$\overline{\text{SPLIT\_POWER}}$	0: Cannot use split power 1: Can use split power	R/W	0
6	RTC_disable	0: RTC enabled 1: RTC disabled (no 32-kHz clock)	R/W	0
5	SET_32_COUNTER	0: No action 1: Set the 32-kHz counter with COMP_REG value.	R/W	0
4	TEST_MODE	0: Functional mode 1: Test mode (autocompensation is enabled when the 32-kHz counter reaches its end)	R/W	0
3	MODE_12_24	0: 24-hour mode 1: 12-hour mode (PM/AM mode)	R/W	0
2	AUTO_COMP	0: No autocompensation 1: Autocompensation enabled	R/W	0
1	ROUND_30S	0: No update 1: When a 1 is written, the time is rounded to the closest minute.	R/W	0
0	STOP_RTC	0: RTC is frozen. 1: RTC is running.	R/W	0

The SET_32_COUNTER must only be used when the RTC is frozen.

ROUND_30S is a toggle bit. MPU can only write 1, and RTC clears it. If the MPU sets the ROUND_30S bit and then reads it, the MPU read 1 until the round-to-the-closest-minute is performed at the next second.

MODE_12_24: It is possible to switch between the two modes at any time without disturbing the RTC. Read or write is always performed with the current mode.

Table 19. RTC Status Register (RTC_STATUS_REG)

Base Address = 0xFFFB 4800, Offset = 0x44				
Bit	**Name**	**Function**	**R/W**	**Reset**
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. 1: RTC is running.	R	0
0	BUSY	0: Updating event in more than 15 µs 1: Updating event	R	0

The alarm interrupt keeps its low level until the MPU writes 1 in the ALARM bit of the RTC_STATUS_REG register.

The timer interrupt is a low-level pulse (15-µs duration).

The RUN bit shows the real state of the RTC. Because the STOP_RTC signal is resynchronized on the 32-kHz clock, the action of this bit is delayed.

POWER_UP is set by a reset and is cleared by writing 1 in this bit.

Table 20. RTC Interrupts Register (RTC_INTERRUPTS_REG)

Base Address = 0xFFFB 4800, Offset = 0x48				
Bit	**Name**	**Function**	**R/W**	**Reset**
3	IT_ALARM	Enables one interrupt when the alarm value is reached (time and calendar alarm registers) by the time and calendar registers	R/W	0
2	IT_TIMER	Enable periodic interrupt 0: Interrupt disabled 1: Interrupt enabled	R/W	0
1:0	EVERY	Interrupt period 0: Every second 1: Every minute 2: Every hour 3: Every day	R/W	00

Note:

The MPU must respect the BUSY period to prevent spurious interrupt.

Table 21. RTC Compensation LSB Register (RTC_COMP_LSB_REG)

Base Address = 0xFFFB 4800, Offset = 0x4C				
Bit	**Name**	**Function**	**R/W**	**Reset**
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. This means that to add one 32-kHz oscillator period every hour, the MPU must write FFFF into RTC_COMP_MSB_REG and RTC_COMP_LSB_REG. To remove one 32-kHz oscillator period every hour, the MPU must write 0001 into RTC_COMP_MSB_REG and RTC_COMP_LSB_REG. The 7FFF value is not allowed.

Table 22. RTC Compensation MSB Register (RTC_COMP_MSB_REG)

Base Address = 0xFFFB 4800, Offset = 0x50				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:0	RTC_COMP_MSB	Indicates number of 32-kHz periods to be added into the 32-kHz counter every hour	R/W	0x00

Table 23. RTC Oscillator Register (RTC_OSC_REG)

Base Address = 0xFFFB 4800, Offset = 0x54				
Bit	**Name**	**Function**	**R/W**	**Reset**
4	OSC32K_PWRDN_R	Control of 32-kHz oscillator power down (function mode)	R/W	0
3:0	SW_RES_PROG	Value of the oscillator resistance	R/W	

The oscillator receives the register value when TST_OSC32K_MUX_CTRL = 0 (functional mode); otherwise, the value resistance and the power down are from the JTAG register.

Index