OMAP5912 Multimedia Processor Power Management Reference Guide

Literature Number: SPRU753A
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 power management in the OMAP5912 multimedia processor.

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

Power Management

This document describes power management in the OMAP5912 multimedia processor.

1 Ultralow-Power Device

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 chapter describes the ULPD module and outline architecture**.** For further information on clock sources and clock and reset architecture, see Chapter 4 and Chapter 5.

1.1 ULPD Features

The ULPD has the following features:

- ❏ Performs transitions among power modes (awake, big sleep, and deep sleep)
- ❏ Handles idle/wake-up handshake with OMAP3.2
- ❏ Monitors wake-up events
- ❏ Controls system clock input sources for several possible configurations (oscillator/external clocks)
- ❏ Performs calibration of the 32-kHz oscillator
- ❏ Manages the clocks and resets distributed to OMAP3.2 and to some peripherals
- ❏ Handles the power-up sequence
- ❏ Controls an on-chip PLL that generates a 96-MHz clock (for 48-MHz peripheral clocks)
- ❏ Is controlled by the MPU for set up and configuration
- ❏ Manages the sleep signal of an embedded LDO used to regulate the supply voltage for OMAP3.2 digital phase locked loop (DPLL) and the system clock oscillator

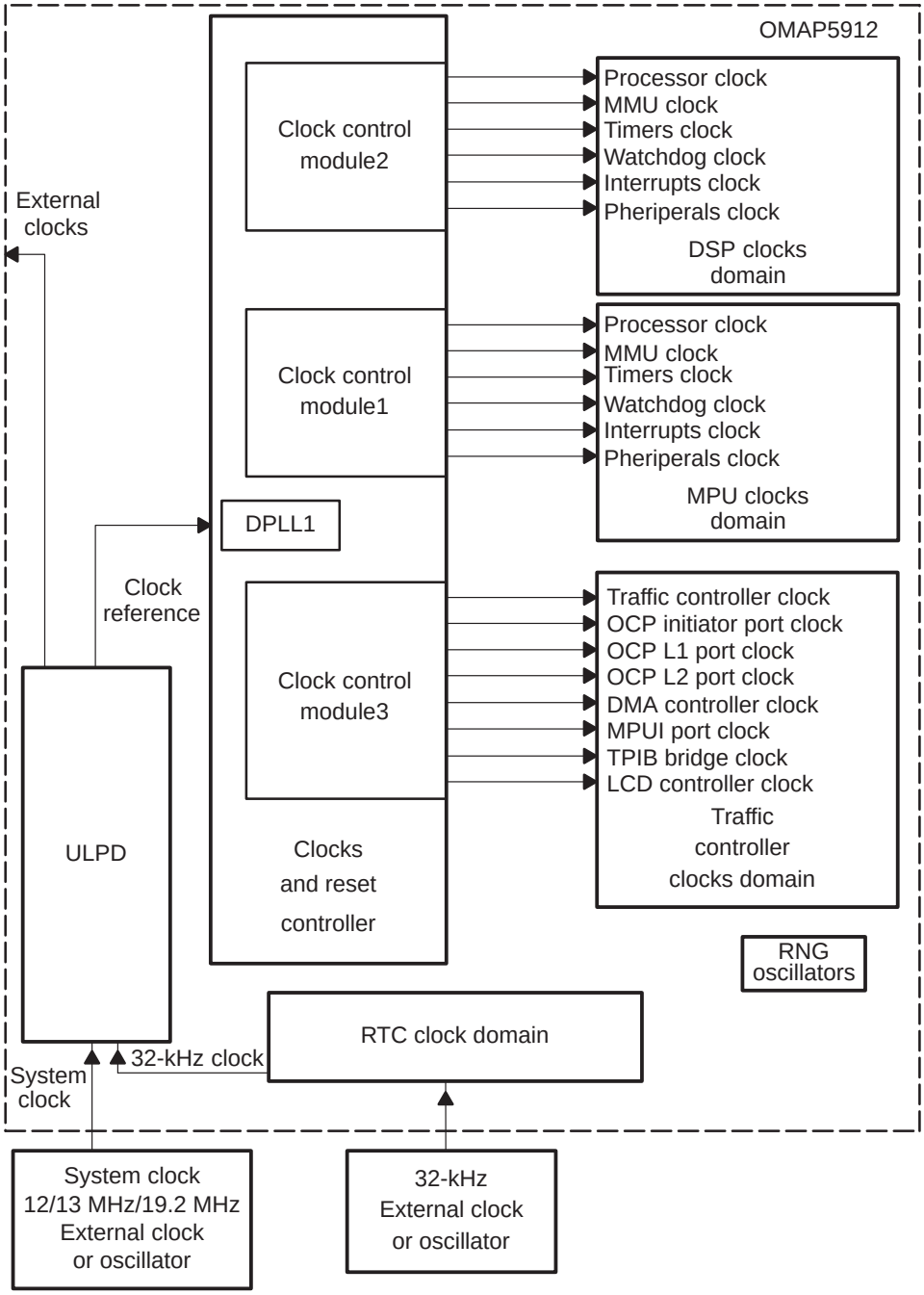
1.2 Overview

The ULPD is a power management module running at 32 kHz (CLK32K).

The ULPD employs three global-system power modes: awake mode, big sleep mode, and deep sleep mode. See Section 1.5, *Power Modes*, for a description of these modes and the transitions between them.

The ULPD controls the various module clocks with several input clock sources provided by the oscillator and by the OMAP3.2 DPLL and the 96-MHz analog phase-locked loop (APLL). It also can sequence the wake-up of the system properly by enabling various analog cells successively. Analog cells are on-chip or external modules that are involved in the generation of the device input clock or supply voltage. Typical analog cells are on-chip oscillators or external supply voltage regulators. ULPD can manage up to six analog cells Figure 1.

Figure 1. ULPD and Clock Domains in OMAP5912



The ULPD is composed of one state machine, a clock management module, and a control register file.

- The state machine (FSM1) manages the global power modes transitions. It handles the idle/wake-up handshake with OMAP3.2 and monitors the wake-up events.

  The state machine controls the input system clock (internal oscillator or external clock source) and generates resets to OMAP3.2 and to some peripherals. It also manages the power-up sequence. The FSM1 uses a set-up timer to manage the sequencing of the wake-up procedure. Each setup timer is associated with a specific analog cell.

- The clock management module is composed of clock-gating logic, multiplexers, and clock dividers. It generates and manages clocks to OMAP3.2 and to some peripherals. It also manages the 48-MHz clocks. Those clocks are generated from an on-chip analog phase locked loop (APLL) operating at 96 MHz. The output of the APLL is divided by two to create a 48-MHz reference clock. From the reference clock are derived the various 48-MHz clocks.

- The control register file is an MPU peripheral connected to the MPU private peripheral bus; it is used to set/configure the features of ULPD.

See Section 1.24 for a detailed description of the registers.

1.3 ULPD Input Clock Sources

The ULPD has two main clock sources: a 32-kHz clock and a system clock of medium frequency (12 MHz, 13 MHz, and 19.2 MHz). These frequencies are also the ULPD_PLL input clock frequency. See Chapter 5 for additional information on the clock source for ULPD.

1.4 ULPD Setup Counters

The ULPD can sequence the wake-up of the system from deep sleep properly by enabling up to six analog cells successively (for example, oscillator and regulator).

The ULPD FSM1 is instantiated in OMAP5912 with two setup counters, each associated with an analog cell. Setup counters are cascaded and must be programmed with the stabilization time of the associated analog cell.

Whenever a counter underflows, it enables the next analog cell and triggers the associated counter.

The cascaded flow of the SETUP_ANALOG_CELL starts with SETUP_ANALOG_CELL2 and ends with SETUP_ANALOG_CELL3.

When SETUP_ANALOG_CELL3 underflows, all the analog cells must be stable and the input system clock is released internally in ULPD.

The ULPD FSM1 can then move in big sleep or awake mode.

1.5 Power Modes

The ULPD handles three global power modes: deep sleep, big sleep, and awake. The ULPD state machine, which is in charge of the wake-up/idle handshake with OMAP3.2, manages the states of the system in each mode and performs transitions between the modes.

1.5.1 Deep Sleep Mode

In deep sleep mode, all internal clocks are inactive except the 32-kHz clock, which is the ULPD state machine clock. In this mode ULPD_PLL is always inactive.

In oscillator mode, the oscillator is disabled; therefore, the system input clock is off, except when POWER_CTRL_REG [9] =0.

In external mode, the input system clock can be on or off; it is not controlled by ULPD.

Note:

OSC12M_STOP output of ULPD is active high every time the state machine is in a deep-sleep state. EXT_CLK_REQ is the same signal but with inverted polarity.

CAUTION

OMAP5912 cannot go into deep−sleep while an emulator (JTAG) is connected

1.5.2 Big Sleep Mode

In big sleep mode, the OMAP input clock is inactive, the 32-kHz clock is active, and the system input clock is active in both oscillator and external modes.

This state is characterized by an external clock request to go active or POWER_CTRL_REG [4] =0.

This mode has a shorter wake-up latency. It also provides clocks (system frequency clocks and/or ULPD_PLL clock) to peripherals whenever requested and while the OMAP3.2 input clock is stopped.

1.5.3 Awake Mode

In awake mode, the OMAP input clock and any requested peripheral clocks are active. In awake mode, the 32-kHz, system clock, OMAP input clock, and any requested peripheral clocks are active.

1.6 External Clock and Voltage Supply Control

The ULPD provides two signals to control the activation or the shut down of the external clock and core voltage supplies.

These two signals, $\overline{\text{LOW\_PWR}}$ and LOW_PWR, behave similarly except that they do not have the same polarity.

LOW_PWR can be controlled by software, whereas $\overline{\text{LOW\_PWR}}$ cannot.

1.6.1 Behavior of $\overline{\text{LOW\_PWR}}$

The $\overline{\text{LOW\_PWR}}$ signal is used in external clock mode.

When low, $\overline{\text{LOW\_PWR}}$ indicates to external devices that the input system clock (SYS_CLK_IN) can be shut down. It can also indicate to an external power management device that the core voltage supply can be lowered to 1.1 V.

The $\overline{\text{LOW\_PWR}}$ signal is asserted low when the ULPD enters the deep sleep state (except at power-up reset) and released upon deep sleep exit (except at power-up reset).

At power-up reset, $\overline{\text{LOW\_PWR}}$ is reset to its inactive value (high).

Figure 2. Assertion of $\overline{\text{LOW\_PWR}}$

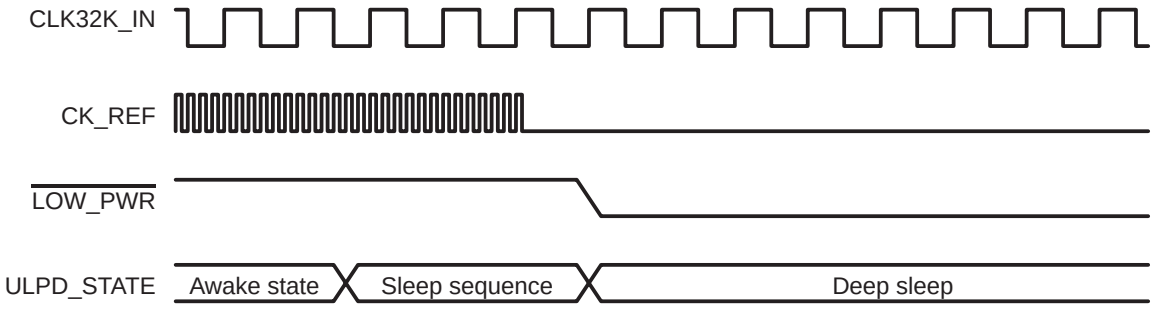
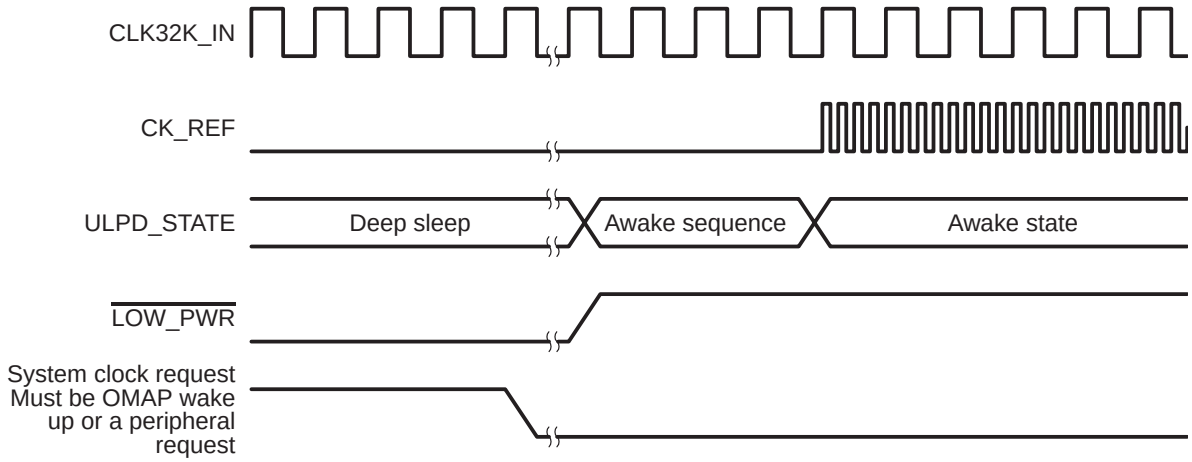


Figure 3. Release of $\overline{LOW\_PWR}$



1.6.2 Behavior of LOW_PWR

The LOW_PWR signal is used in oscillator clock mode to control an external power management device.

When high, the signal LOW_PWR drives the external core voltage supply in low voltage (1.1 V) operations.

LOW_PWR can be set by software so that two types of operations are allowed:

- Reduction of leakage current
- Low-voltage operation at reduced clock frequency, also known as dynamic voltage scaling (DVS)

1.7 Leakage Current Management

The conditions below cause the listed events:

- POWER_CTRL_REG[0] set to 1: Enable LOW_PWR feature
- POWER_CTRL_REG[4] set to 1: Enable transition to deep sleep mode

  or

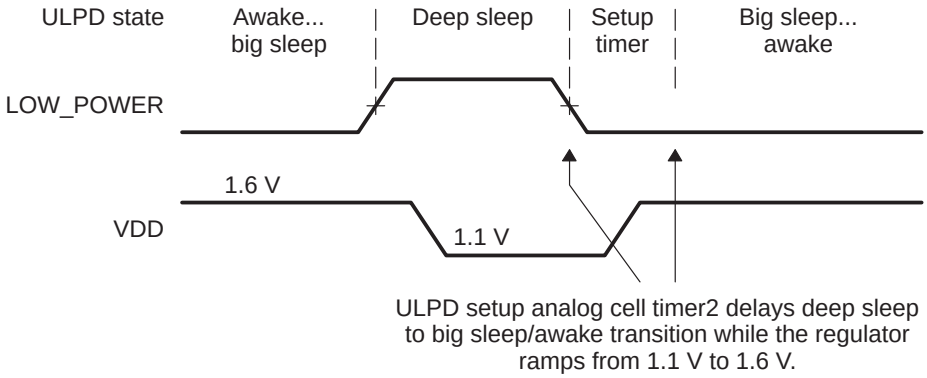
  POWER_CTRL_REG[10] set to 0: DVS disabled

LOW_PWR switches to active high whenever the ULPD enters deep sleep state. In this way, the external core voltage supply can be driven in low-voltage operation by the external power management device.

When the ULPD exits deep sleep mode, LOW_PWR switches back to inactive low and the external core voltage supply ramps up to a nominal 1.5 V.

At reset, the low-power feature is disabled (POWER_CTRL_REG[0] is set to 0). The LOW_PWR signal is inactive low, which indicates a nominal voltage requirement.

Figure 4. Behavior of LOW_PWR in RESET_MODE 0



1.8 Low-Voltage Operation at Reduced Clock Frequency

In this mode of operation, also known as dynamic voltage scaling (DVS), the following conditions must be met:

- ❏ POWER_CTRL_REG[4] set to 0: Disables transition to deep sleep mode
- ❏ POWER_CTRL_REG[10] set to 1: Enables DVS

Whenever OMAP3.2 indicates to ULPD that it is prepared to go into idle, the transition to deep sleep is prevented, and the ULPD moves into big sleep mode. In this mode, the OMAP3.2 input clock is shut down, but the oscillator is still active.

In this case, LOW_PWR follows the value programmed in POWER_CTRL_REG [11].

If POWER_CTRL_REG [11] =1 (min), LOW_PWR switches to active high, driving the external regulator in low-voltage operation.

If POWER_CTRL_REG [11] =0 (max), LOW_PWR switches to inactive low, driving the external regulator in nominal voltage operation.

Whenever OMAP3.2 initiates a wake-up procedure, the ULPD moves back to awake mode but LOW_PWR keeps the value programmed in POWER_CTRL_REG [11].

In this way, when POWER_CTRL_REG [11] =1, OMAP3.2 restarts operations at low voltage. To ramp up the operating voltage back to nominal value, a new

OMAP3.2 idle/wake-up procedure must be performed with POWER_CTRL_REG[11]=0.

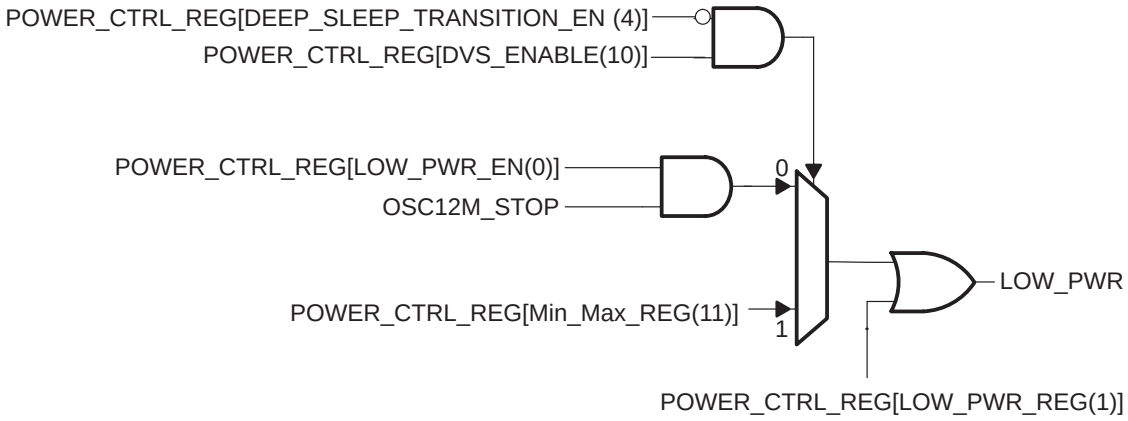
This feature allows dynamic control of the operating voltage of OMAP5912. It provides two operating points (voltage, frequency) to adapt the operating voltage to the performance requirement.

The OMAP3.2 DPLL frequency must be set accordingly before initiating the procedure.

The DVS procedure described here goes through the OMAP3.2 idle and wake-up sequences. To overcome limitations of this procedure, a more direct way to control the OMAP5912 operating voltage is provided.

The user can directly force the LOW_PWR signal to 1 and thereby force the operating voltage to 1.1 V by setting the bit POWER_CTRL_REG[1]. POWER_CTRL_REG [1] cannot force the LOW_PWR signal to 0 (see Figure 5).

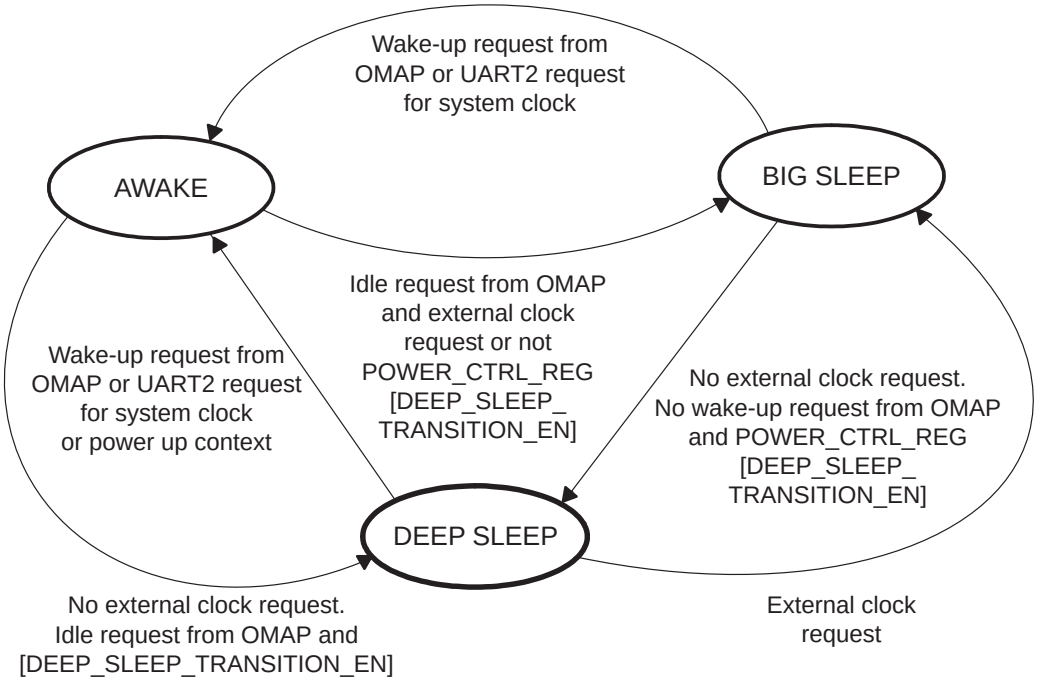
Figure 5. Control of OMAP5912 Low-Power Output by ULPD POWER_CTRL_REG



1.9 Transitions Between Power Modes

Figure 6 shows the basic functional scheme of the FSM1:

Figure 6. Simplified State Diagram of the ULPD FSM1



The ULPD handles the state transitions among deep sleep, big sleep, and awake. The transitions are triggered by the following:

- ❏ External events: Hardware resets or clock requests
- ❏ Internal events: Software resets, watchdog time-out, software clock request, or MPU idle request

Table 1. Sleep Modes versus Active Mode (Summary)

Mode	Power Dissipation	Active Clocks	Comments
Deep sleep	Lowest	32 kHz for wake-up detection	Only the UART2 functional clock using 32 kHz is active.
Big sleep	From modules clocked by active 32-kHz or 48-MHz clocks	System and 32-kHz clock. ULDP output clocks to peripherals if requested	For BCLK and MCLK, the 48-MHz frequency can be divided further by setting respectively SDW_CLK_DIV_CTRL_SEL[7:2] and COM_RATIO_SEL[7:2] in ULPD control registers.
Awake	Nominal	All 32-kHz, 48-MHz, and clocks derived from system clock can be active.	OMAP3.2 input clock is active.

1.10 Power-on Transition to Deep Sleep Mode

At power-up, namely when the power-up input signal $\overline{\text{PWRON\_RESET}}$ is asserted low, the ULPD FSM1 enters deep sleep mode.

In this case, the $\overline{\text{LOW\_PWR}}$ signal is reset to inactive state 1.

When $\overline{\text{PWRON\_RESET}}$ is released, the FSM automatically switches from deep sleep to awake.

The transition follows this procedure:

1) $\overline{\text{PWRON\_RESET}}$ is asserted low, and $\overline{\text{LOW\_PWR}}$ is reset to 1 (inactive).
2) FSM enters the deep sleep mode and stays in this mode until the $\overline{\text{PWRON\_RESET}}$ signal is released.

1.11 Transitions From Deep Sleep Mode

In deep sleep mode, the FSM monitors the OMAP3.2 wake-up request and external clock requests. OMAP3.2 propagates asynchronously unmasked peripheral interrupts to generate a wake-up request.

External clock requests and the OMAP3.2 wake-up request are respectively initiators of deep sleep to big sleep and deep sleep to awake transitions. The power-up reset initiates a transition from deep sleep to awake.

1.11.1 Transition from Deep Sleep to Big Sleep Mode

Transition to big sleep state occurs when there is at least one specific external clock request (see Table 2).

Table 2. Initiators to Deep Sleep −> Big Sleep Transition

Transition from Deep Sleep to Big Sleep	
External Clock Request	
CONF_CAM_CLKMUX_R (register)	CAMERA I/F
External Clock Request (Continued)	
CONF_MOD_UART1_CLK_MODE_R (register)	UART1
CONF_MOD_UART2_CLK_MODE_R (register)	UART2
CONF_MOD_UART3_CLK_MODE_R (register)	UART3
CONF_MOD_USB_HOST_HHC_UHOST_EN_R (register)	USB OTG
CONF_MOD_MMC_SD_CLK_REQ_R (register)	MMC/SDIO1

Table 2. Initiators to Deep Sleep −> Big Sleep Transition (Continued)

Transition from Deep Sleep to Big Sleep	
CONF_MOD_MMC_SD2_CLK_REQ_R (register)	MMC/SDIO2
Clock request by USB	USB OTG. See Note 4.
MCLKREQ	OMAP5912 input pin
BCLKREQ	OMAP5912 I/O input pin
SOFT_REQ_REG (active)	Software requests. See Note 5.
Other	
DEEP_TRANSITION_ENABLE	ULPD

Notes: 1) Software requests prevent the transition to deep sleep when leaving awake state but are not initiators of deep sleep to big sleep transition.

2) DEEP_SLEEP_TRANSITION_EN is not an initiator of deep sleep to big sleep transition. It keeps FSM1 in big sleep mode and prevents transition to deep sleep mode when leaving the awake state.

3) When in external 48-MHz clock mode (CONF_DPLL_EXT_SEL = 0), clock requests related to 48-MHz clock have no effect on the FSM.

4) The USB can request clock when an external host or device is detected or when the USB enters the resume state.

5) Software requests for clock can be programmed in the ULPD SOFT_REQ_REG (see Table 20).

The transition sequence is as follows:

1) External clock request occurs.

2) $\overline{\text{LOW\_PWR}}$ is asserted high.

3) Depending on the system input clock source mode:

   a) External mode: In this case, the setup counter, SETUP_ANALOG_CELL3, is loaded with the related setup value from the ULPD register that corresponds to the ramp-up time of the external voltage supply. When the counter underflow is generated, it globally enables the system input clock to peripherals.

      At this point it is possible that the external system input clock is not present yet. The clocks for the peripherals are effectively restarted whenever the system input clock arrives, if the corresponding clock request is active.

      This setup stage allows the supply voltage to be stable before enabling the input clocks to peripherals.

b) Oscillator mode: In this case, the setup counter, SETUP_ANALOG_CELL2, is loaded with the related setup value from the ULPD register that corresponds to the maximum time between ramp-up time of the external voltage supply and LDO stabilization time.

When the counter underflow is generated, it enables the oscillator. Then the setup counter, SETUP_ANALOG_CELL3, is loaded with the related setup value from the ULPD register that corresponds to the stabilization delay of the oscillator.

When the counter underflow is generated, it globally enables the system input clock to peripherals.

The clocks for the peripherals are restarted if the corresponding clock request is active. These two setup stages allow the supply voltage and the input system clock to be stable before enabling the input clock to peripherals.

4) FSM1 enters big sleep mode.

1.11.2 Transition from Deep Sleep to Awake Mode

Transition to awake mode occurs whenever a wake-up event occurs. Wake-up events are:

- $\overline{\text{MPU\_RST}}$
- OMAP3.2 asserts the wake-up request. A wake-up request is initiated by peripheral unmasked interrupts.
- UART2 requests system clock.
- $\overline{\text{PWRON\_RESET}}$
- RTC_ON_NOFF
- 32-kHz watchdog reset

Table 3. Initiators of Deep Sleep −> Awake Transition

Transition from Deep Sleep to Awake	
Wake-up Event	
$\overline{\text{PWRON\_RESET}}$	Power-on reset pin
RTC_ON_NOFF	Power-on reset pin
$\overline{\text{MPU\_RST}}$	System reset pin
32-kHz watchdog reset	32-kHz watchdog time-out
Wake-up request	Peripheral unmasked interrupts
PERIPH_$\overline{\text{REQ}}$	
PERIPH_$\overline{\text{REQ}}$	System clock request from UART2

The transition follows one of two sequences:

❑ Sequence 1

1) The following wake-up event occurs:
   a) OMAP3.2 asserts a wake-up request. The wake-up request is initiated by a peripheral unmasked interrupt.
   b) UART2 requests system clock.
   c) Power-up reset, system reset (low on MPU_RST), secure watchdog reset, or 32-kHz watchdog reset
2) $\overline{\text{LOW\_PWR}}$ is asserted high.
3) Depending on the system input clock source mode:
   a) External mode: In this case, the setup down counter, SETUP_ANALOG_CELL3, is loaded with the related setup value from the ULPD register that corresponds to the ramp-up time of the external voltage supply.

      When the counter underflow is generated, it globally enables the system input clock to peripherals.

      At this point, it is possible that the external system input clock is not present yet.

      The peripheral clocks are effectively restarted whenever the system input clock arrives, if the corresponding clock request is active.

      This setup stage is intended to allow the supply voltage to be stable before enabling the input clocks to peripherals.

      At power up, the supply voltage and the input system clock must be stable when the $\overline{\text{PWRON\_RESET}}$ signal is released. In this case, this setup stage is skipped.

b) Oscillator mode: In this case, the setup counter, SETUP_ANALOG_CELL2, is loaded with the related setup value from the ULPD register that corresponds to the maximum time between ramp-up time of the external voltage supply and LDO stabilization time.

When the counter underflow is generated, it enables the oscillator. Then the setup counter, SETUP_ANALOG_CELL3, is loaded with the related setup value from the ULPD register that corresponds to the stabilization delay of the oscillator.

When the counter underflow is generated, it globally enables the system input clock to peripherals.

The clocks for the peripherals are restarted if the corresponding clock request is active. These two setup stages are intended to allow the supply voltage and the input system clock to be stable before enabling the input clock to peripherals.

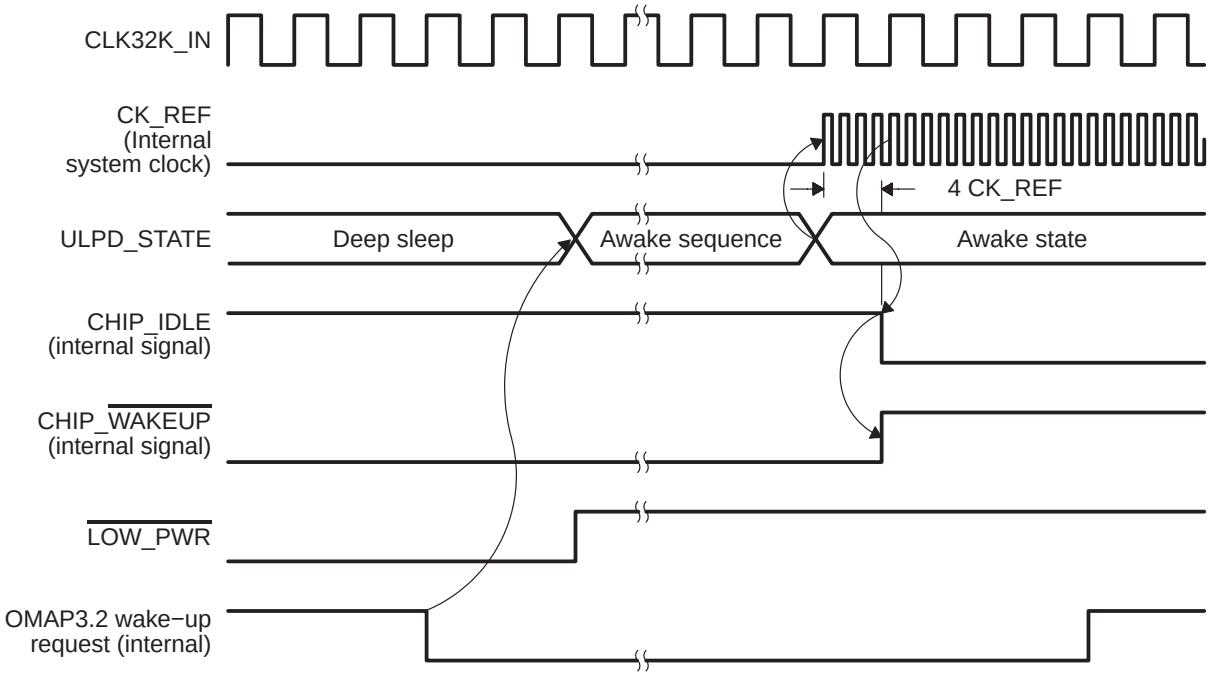
4) FSM1 enters the awake mode.

5) OMAP3.2 input clock is enabled. In external mode, the clock is effectively restarted whenever the system input clock arrives.

6) OMAP3.2 deasserts the CHIP_IDLE signal.

7) ULPD releases the CHIP_$\overline{\text{WAKEUP}}$ signal high.

Figure 7. 2OMAP3.2-Initiated Wake-up Sequence

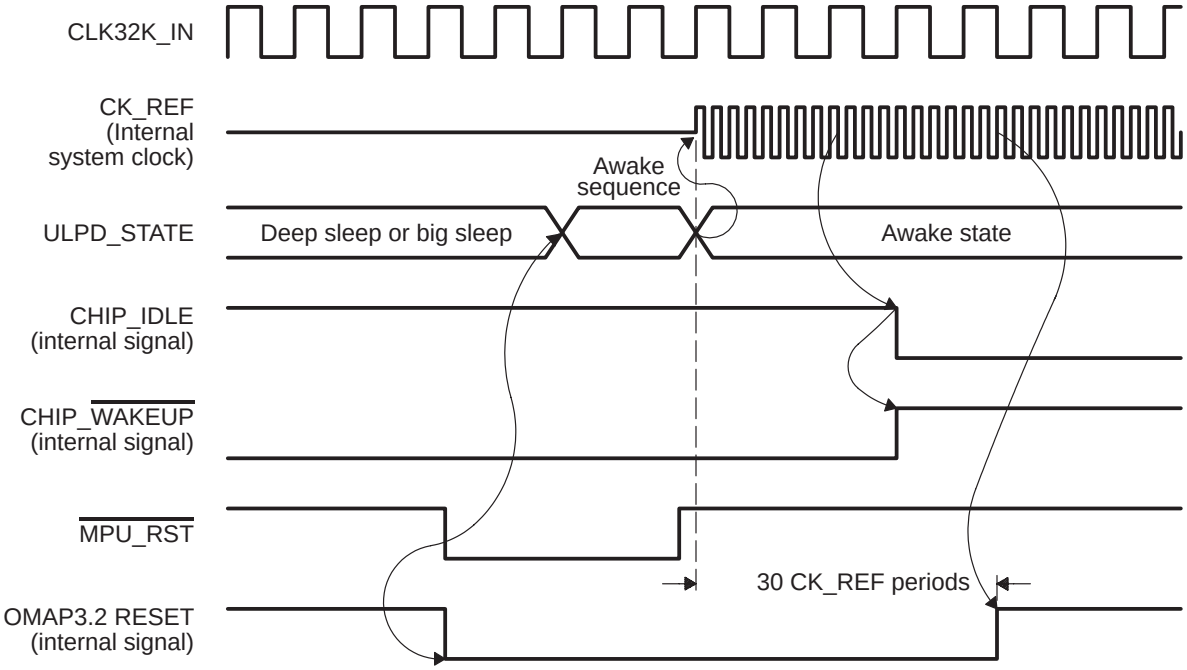


The wake-up sequence requires five CLK32K clock cycles from assertion of OMAP3.2 wake-up request to release of CK_REF.

❑ Sequence 2

1) The following wake-up events occur:
   a) $\overline{\text{MPU\_RST}}$ low event
   b) 32-kHz watchdog time-out
2) ULPD performs the transition to awake as described previously and releases the system clock to OMAP3.2.
3) As soon as the system clock is back on, the ULPD asserts low the OMAP3.2 reset for at least 30 system clock cycles.
4) On detection of OMAP3.2 reset low, OMAP3.2 deasserts the CHIP_IDLE signal.
5) ULPD asserts the CHIP_$\overline{\text{WAKEUP}}$ signal high.

Figure 8. Wake-up Sequence in Case of Warm Reset



1.12 Transitions From Big Sleep Mode

1.12.1 Transition From Big Sleep Mode to Deep Sleep Mode

Three necessary conditions lead to deep sleep mode:

- ❑ Wake-up request from OMAP is not active.
- ❑ Peripherals clock request and software clock request are inactive. Namely, every clock request that is in is not active.
- ❑ POWER_CTRL_REG[4]=1

The transition follows this sequence:

1) All clock requests are inactive; there is no wake-up request from OMAP and POWER_CTRL_REG[4]=1.
2) Input clocks to peripherals are globally disabled.
3) $\overline{\text{LOW\_PWR}}$ is activate low and in oscillator mode the oscillator is also disabled, unless POWER_CTRL_REG[9]=0.
4) FSM1 enters deep sleep mode.

1.12.2 Transition From Big Sleep Mode to Awake Mode

The transition to awake mode occurs when OMAP3.2 requests wake-up or UART2 requests system clock. An OMAP3.2 wake-up request is initiated by the unmasked interrupts of the peripherals.

The transition follows this sequence:

1) Wake-up event occurs (OMAP3.2 wake-up request or UART2 requests system clock)

2) FSM1 enters the awake mode.

3) OMAP3.2 input clock in enabled. In external mode, the clock is effectively restarted whenever the external system input clock arrives.

1.13 Transitions From Awake Mode

1.13.1 Transition From Awake Mode to Deep Sleep Mode

Three necessary conditions lead to deep sleep mode:

- ❑ OMAP3.2 asserts the CHIP_IDLE signal when no clocks are needed in OMAP3.2 or by any peripheral using OMAP3.2 output clocks. For example, DMA_LCD_CTRL.lcd_destination_port = 1 is a condition that prevents the OMAP3.2 chip_idle to be asserted because the clock request corresponding to the external LCD controller clock is kept active.
- ❑ No clock request or software request by peripherals
- ❑ POWER_CTRL_REG [4] =1

The transition follows this sequence:

1) All clock requests are inactive, CHIP_IDLE is asserted high and POWER_CTRL_REG[4]=1. Any incoming interrupt is ignored.

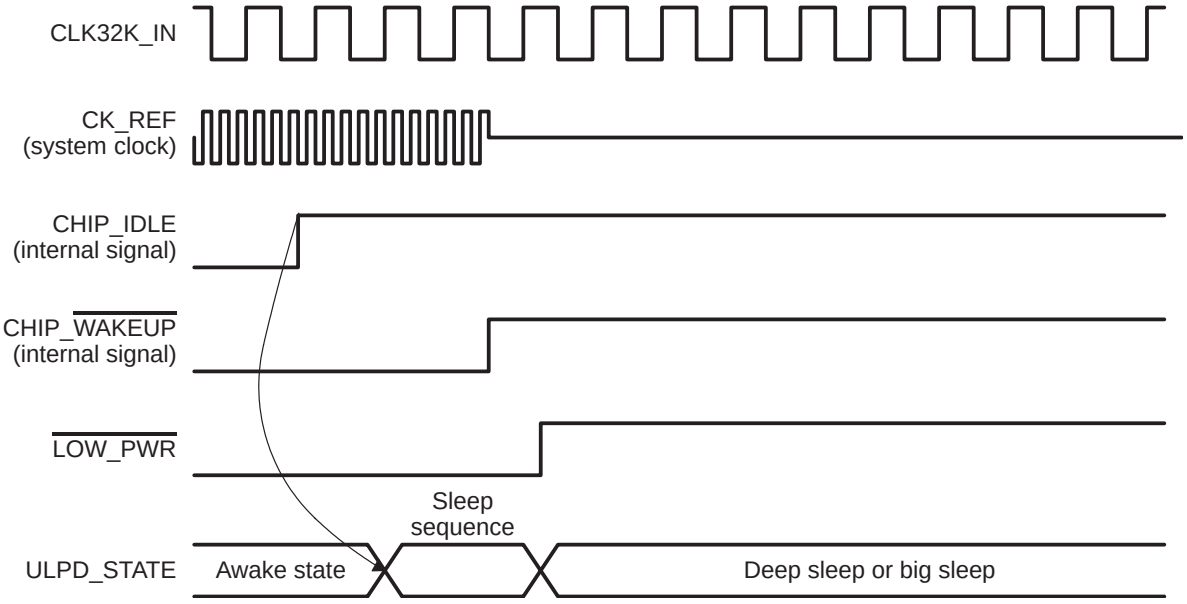
2) The ULPD asserts CHIP_$\overline{\text{WAKEUP}}$ to 0. This is a sleep acknowledge, and OMAP3.2 expects its input clock to be cut off. Interrupts can then be taken again.

3) Input clock to peripherals is globally disabled.

4) $\overline{\text{LOW\_PWR}}$ is asserted low and in oscillator mode the oscillator is also disabled, unless POWER_CTRL_REG[9]=0.

5) FSM1 enters the deep sleep mode.

Figure 9. Sleep Sequence



The sleep sequence requires three cycles of the CLK32K clock from assertion of the CHIP_IDLE signal to assertion of the LOW_PWR signal.

Note:

The name of the signal CHIP_WAKEUP may confuse users. This signal is not a wake-up signal but is an idle acknowledge.

1.13.2 Transition From Awake Mode to Big Sleep Mode

Two necessary conditions lead to big sleep mode:

- ❑ OMAP asserts the CHIP_IDLE (active high) when no clocks are needed in OMAP3.2 or by any peripherals using OMAP3.2 output clocks.
- ❑ At least one specific peripheral clock request is active or POWER_CTRL_REG[4]=0

Sequence:

1) CHIP_IDLE is asserted high. There are active external clock requests or POWER_CTRL_REG[4]=0.
2) OMAP3.2 input clock is disabled.
3) FSM1 enters the big sleep mode.

1.14 ULPD Output Clocks

The ULPD controls the system clock (12 MHz or 19.2 MHz) and the various 48-MHz clocks used by OMAP5912 peripherals.

These clocks supply the OMAP3.2 subsystem and some of the OMAP5912 peripherals.

Figure 10 shows a simplified diagram of the ULPD clock generation scheme. Most of the clocks generated by ULPD can be controlled independently by hardware and by software. In fact, a hardware clock request signal and a software clock request (programmable) bit in the SOFT_REQ_REG register are associated with most of the ULPD-generated clocks.

These clock requests are used in the ULPD to enable/disable the associated clocks.

In addition, the hardware request serves as a wake-up event for the ULPD state machine and the clock generation module it controls (the oscillator and ULPD_PLL).

The software requests are involved in the conditions of the awake to big sleep transition.

Figure 10. Basic Diagram of the ULPD Output and Input Clocks

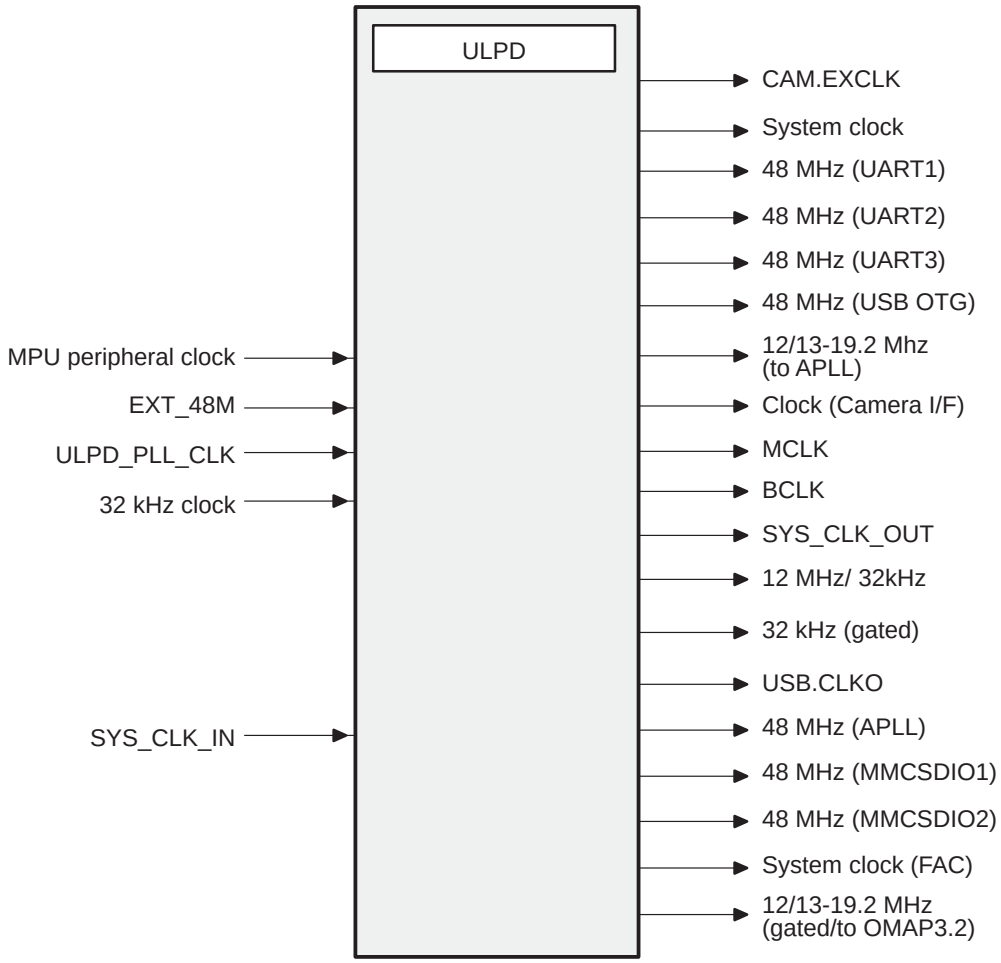


Table 4 describes how clocks are gated by the ULPD. It shows:

- ❏ The source of each clock output with its associated selection signal
- ❏ The specific enable signal and clock request whenever the clock can be gated

See Section 1.24 for more details on registers.

Table 4. ULPD Output Clocks Description

OMAP5912 I/O Destination	Clock Selected	Clock Source Selection	Wake-up Request	Soft Request Disable
BCLK	Derived from APLL CLK/2	SDW_SYSCLK_ PLLCLK_SEL CLOCK_CTRL_ REG[2] SDW_CLK_DIV_ CTRL_SEL[7:2] (see Note 1)	BCLKREQ	SOFT_DISABLE_ REQ_REG[1]
MCLK	Derived from APLL CLK/2	COM_SYSCLK_ PLLCLK_SEL CONF_MOD_COM_ MCLK_12_48_ SEL_R CONF_DPLL_EXT_ SEL (see Note 2) CLOCK_CTRL_ REQ[1] COM_RATIO_ SEL[7:2] (see Note 3)	MCLKREQ SOFT_REQ_ REG[6] COM_CLK_DIV_ CTRL_SEL[1] SOFT_REQ_ REG[1]	SOFT_DISABLE_ REQ_REG[0]
UART2	System clock	CLOCK_CTRL_ REG[0]	OMAP3.2 wake-up request UART2 request for system clock SOFT_REQ_ REG[5]	SOFT_DISABLE_ REQ_REG[3]
USB.CLKO	EXT.48M divided by 8 APLL CLK/2	CONF_DPLL_EXT_ SEL CLOCK_CTRL_ REG[4]	Clock request to APLL SOFT_REQ_ REG[0]	
SYS_CLK_OUT	System clock	CLOCK_CTRL_ REG[3]	MCLKREQ SOFT_REQ_ REG[1]	SOFT_DISABLE_ REQ_REG[0]

Table 4. ULPD Output Clocks Description (Continued)

OMAP5912 I/O Destination	Clock Selected	Clock Source Selection	Wake-up Request	Soft Request Disable
CAM.D[7] (See Note 4.)	EXT.48M APLL CLK/2	CONF_DPLL_EXT_SEL	SOFT_REQ_REG[0] Clock request to APLL	
UART1	EXT.48M APLL CLK/2	CONF_DPLL_EXT_SEL	MOD_CONF_CTRL0[29] SOFT_REQ_REG[9]	SOFT_DISABLE_REQ_REG[7]
UART2	EXT.48M APLL CLK/2	CONF_DPLL_EXT_SEL	MOD_CONF_CTRL0[30] SOFT_REQ_REG[10]	SOFT_DISABLE_REQ_REG[8]
UART3	EXT.48M APLL CLK/2	CONF_DPLL_EXT_SEL	MOD_CONF_CTRL0[31] SOFT_REQ_REG[11]	SOFT_DISABLE_REQ_REG[9]
CAMERA I/F	EXT.48M APLL CLK/2	CONF_DPLL_EXT_SEL	CONF_ CAM_CLKMUX_R SOFT_REQ_REG[11]	SOFT_DISABLE_REQ_REG[5]
FAC	MPU peripheral clock	CLOCK_CTRL_REG[5]	Clock request by the USB SOFT_REQ_REG[3]	

Table 4. ULPD Output Clocks Description (Continued)

OMAP5912 I/O Destination	Clock Selected	Clock Source Selection	Wake-up Request	Soft Request Disable
USB OTG	EXT.48M APLL CLK/2	CONF_DPLL_ EXT_SEL	USB request for 48 MHz (see Note 5) CONF_MOD_ USB_HOST_ HHC_UHOST_ EN_R SOFT_REQ_ REG[8]	SOFT_REQ_ REG[4] SOFT_DISABLE_ REQ_REG[6]
MMC/SDIO1	EXT.48M APLL CLK/2	CONF_DPLL_ EXT_SEL	MOD_CONF_ CTRL_0[23] SOFT_REQ_ REG[12]	SOFT_DISABLE_ REQ_REG[10]
MMC/SDIO2	EXT.48M APLL CLK/2	CONF_DPLL_ EXT_SEL	MOD_CONF_ CTRL_0[20] SOFT_REQ_ REG[13]	SOFT_DISABLE_ REQ_REG[11]
CAM.EXCLK	System clock System clock divided by 2	CAM_CLK_CTRL[0] CAM_CLK_CTRL[1]	OMAP3.2 wake-up request	
GPIO	System clock	CAM_CLK_CTRL[2]	OMAP3.2 wake-up request	

Notes: 1) The frequency on the BCLK can be set accordingly: SDW_CLK_DIV_CTRL_SEL[7:2]. The resulting frequency is given in the following table:

SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00000, BCLK = 48 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00001, BCLK = 32 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00002, BCLK = 24 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00003, BCLK = 19.2 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00004, BCLK = 16 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00005, BCLK = 13.7 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00006, BCLK = 12 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00007, BCLK = 9.6 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00008, BCLK = 8 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X00009, BCLK = 6.9 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X000012, BCLK = 3 MHz
SDW_CLK_DIV_CTRL_SEL[7:2] = 0X000032, BCLK = 1 MHz

Other programmed values in SDW_CLK_DIV_CTRL_SEL[7:2] result in BCLK = 48 MHz.

2) If CONF_DPLL_EXT_SEL is set to 1, external 48 MHz clock source is selected instead of the APLL clock source. The external 48 MHz clock is provided through GPIO[14].

3) The frequency on the MCLK can be set accordingly: COM_RATIO_SEL[7:2]. The resulting frequency is given in the following table:

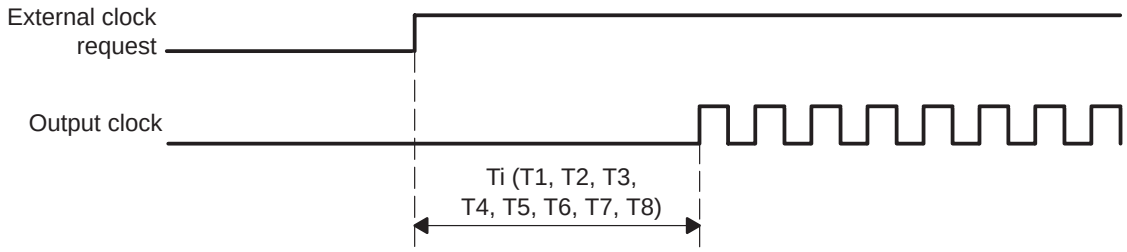
   COM_RATIO_SEL[7:2] = 0X00000, MCLK = 48 MHz
   COM_RATIO_SEL[7:2] = 0X00001, MCLK = 32 MHz
   COM_RATIO_SEL[7:2] = 0X00002, MCLK = 24 MHz
   COM_RATIO_SEL[7:2] = 0X00003, MCLK = 19.2 MHz
   COM_RATIO_SEL[7:2] = 0X00004, MCLK = 16 MHz
   COM_RATIO_SEL[7:2] = 0X00005, MCLK = 13.7 MHz
   COM_RATIO_SEL[7:2] = 0X00006, MCLK = 12 MHz
   COM_RATIO_SEL[7:2] = 0X00007, MCLK = 9.6 MHz
   COM_RATIO_SEL[7:2] = 0X00008, MCLK = 8 MHz
   COM_RATIO_SEL[7:2] = 0X00009, MCLK = 6.9 MHz
   COM_RATIO_SEL[7:2] = 0X000012, MCLK = 3 MHz
   COM_RATIO_SEL[7:2] = 0X000032, MCLK = 1 MHz

   Other programmed values in COM_RATIO_SEL[7:2] result in MCLK = 48 MHz.

4) The 48 MHz from the APLL, which can be observed on CAM.D[7], is the observability mode as configured.
5) The USB will request 48 MHz clock in case of the following event:
   − The USB has detected that either an external host or an external device is attached to one of the configured OMAP5912 USB ports, or
   − The USB exits the suspend mode and enters the resume mode.

Figure 11 shows the latency between the clock request and the clock activation.

Figure 11. Timing Diagram for Clock Request to Clock Available Latency



The parameters (T1 to T8) are the latencies between peripheral clock requests and clock available (see Table 5 and Table 6). These latencies depend on ULPD setup counters and on the FSM1 state when the request is received.

Table 5. Clock Request to Clock Available Latencies

Latency Name	Description
T1	4 x 32-kHz clock cycles + setup analog cell
T2	3 x 32-kHz clock cycles
T3	2 x 32-kHz clock cycles + setup analog cell
T4	1 x 32-kHz clock cycles
T5	1 x 32-kHz clock cycles + APLL lock time
T6	2 x 32-kHz cycles + setup analog cell + APLL lock time
T7	2 x 32-kHz clock cycles
T8	3 x 32-kHz clock cycles + setup analog cell

Table 6. Latencies for Each Peripheral Clock

Name	Wake-Up Request	Time to Get the Clock Active Depending on Initial FSM State		
		Deep Sleep	Big Sleep	Awake
CK_REF	• $\overline{\text{PWRON\_RESET}}$	T1	T2	
	• $\overline{\text{MPU\_RST}}$			
	• RTC_ON_NO$\overline{\text{FF}}$			
	• 32-kHz watchdog time-out			
	• Wake-up request			
	• UART2 requests system clock			
BCLK	• BCLKREQ	System clock		
	• SOFT_REQ_REG[2]	T3	T4	T5
	• SDW_CLK_DIV_CTRL_SEL[1]	PLL clock		
		T6	T5	T5
MCLK	• MCLKREQ	System clock		
	• SOFT_REQ_REG[6]	T3	T4	T4
	• COM_CLK_DIV_CTRL_SEL[1]	PLL clock		
	• CONF_MOD_COM_MCLK_12_48_SEL_R	T6	T5	T5

Table 6. Latencies for Each Peripheral (Continued)Clock (Continued)

Name	Wake-Up Request	Time to Get the Clock Active Depending on Initial FSM State		
		Deep Sleep	Big Sleep	Awake
Functional Clock UART2	• $\overline{\text{PWRON\_RESET}}$ • $\overline{\text{MPU\_RST}}$ • RTC_ON_NOFF • 32-kHz watchdog time-out • Wake-up request • UART2 requests system clock • SOFT_REQ_REG[5]	T8	T7	0
48-MHz USB OTG	• Any hard or soft request related to APLL • SOFT_REQ_REG[0]	T6	T5	T5
SYS_CLK_OUT	• MCLKREQ • SOFT_REQ_REG[1]	T3	T4	T4
48 MHz from APLL	• SOFT_REQ_REG[0] • Any hard or soft request related to APLL	T6	T5	T5
48-MHz UART1	• CONF_MOD_UART1_CLK_MODE_R • SOFT_REQ_REG[9]	T6	T5	T5
48-MHz UART2	• CONF_MOD_UART2_CLK_MODE_R • SOFT_REQ_REG[10]			
48-MHz UART3	• CONF_MOD_UART3_CLK_MODE_R • SOFT_REQ_REG[11]			
Clock for CAMERA IF	• CONF_CAM_CLKMUX_R • SOFT_REQ_REG[7]	T6	T5	T5
System clock for FAC	• Request for 48 MHz from the USB • SOFT_REQ_REG[3]	T6	T2	T2
48 MHz for USB OTG	• Request for 48 MHz from the USB • CONF_MOD_USB_HOST_HHC_UHOST_EN_R • SOFT_REQ_REG[8]	T6	T5	T5

Table 6. Latencies for Each Peripheral (Continued)Clock (Continued)

Name	Wake-Up Request	Time to Get the Clock Active Depending on Initial FSM State		
		Deep Sleep	Big Sleep	Awake
48 MHz for MMCSDIO1	• CONF_MOD_MMC_SD_CLK_REQ_R • SOFT_REQ_REG[12]	T6	T5	T5
48 MHz for MMCSDIO2	• CONF_MOD_MMC_SD2_CLK_REQ_R • SOFT_REQ_REG[13]	T6	T5	T5
CAM.EXCLK	• $\overline{\text{PWRON\_RESET}}$ • $\overline{\text{MPU\_RST}}$ • RTC_ON_NOFF • 32-kHz watchdog time-out • Wake-up request • UART2 requests system clock	T1	T2	
System clock for GPIO	• $\overline{\text{PWRON\_RESET}}$ • $\overline{\text{MPU\_RST}}$ • RTC_ON_NOFF • 32-kHz watchdog time-out • Wake-up request • UART2 requests system clock	T1	T2	

1.15 Power-up and Reset Management

1.15.1 Device Power up

The $\overline{\text{PWRON\_RESET}}$ signal is the power-on reset and is used to reset the ULPD 32-kHz logic and drive the input resets of OMAP3.2.

The $\overline{\text{PWRON\_RESET}}$ is resynchronized on 32 kHz to achieve a clean reset of the ULPD.

It is therefore required to maintain the power-on reset active low for a minimum of two 32-kHz clock cycles.

At power-up reset, RESET_MODE selects between the external mode and the oscillator mode.

❑ If RESET_MODE is at 0, the ULPD starts in oscillator mode. In this case, an on-chip oscillator generates the system input clock.

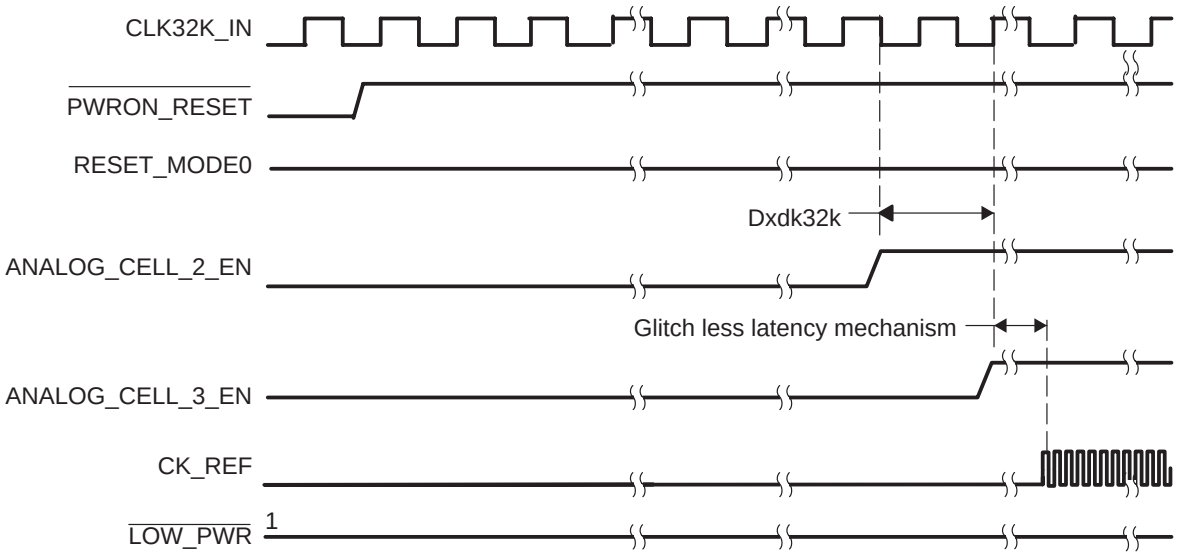
- The ULPD controls the oscillator and inserts proper setup times at power-up to ensure that a stable clock is released to the system.
- In this mode, $\overline{\text{PWRON\_RESET}}$ must be released only when the 32-kHz clock and the supply voltage are stable.
- If RESET_MODE is at 1, the ULPD starts in external clock mode. In this case, no setup time delay is inserted at power up.
- An external chip provides the input clock and ensures that it is stable.
- In external clock mode the $\overline{\text{PWRON\_RESET}}$ must be released only when the system clock, the 32-kHz clock, and the supply voltage are stable.

1.15.2 Generic Power-up Sequence in Oscillator Mode

Power-up in oscillator mode follows this sequence:

1) $\overline{\text{PWRON\_RESET}}$ is released, and the first analog cell is enabled
2) The SETUP_ANALOG_CELL2 counter starts counting in order to account for the power-supply setup time. The reset value of SETUP_ANALOG_CELL2 is 0, because the power supplies are stable before the deassertion of the power-up reset. The SETUP_ANALOG_CELL2 counter is to be used during exit of deep sleep mode while the core power supply is ramping from 1 V.
3) When SETUP_ANALOG_CELL2 underflows, the SETUP_ANALOG_CELL3 counter starts counting. SETUP_ANALOG_CELL3 is used to account for the stabilization of the 12-MHz oscillator.
4) When SETUP_ANALOG_CELL3 underflows, all the analog cells must be stable. The input system clock is released internally in ULPD.
5) The ULPD FSM1 transitions to awake mode.
6) The ULPD enables the input clock to OMAP3.2.

Figure 12. Power-up Sequence in Oscillator Mode



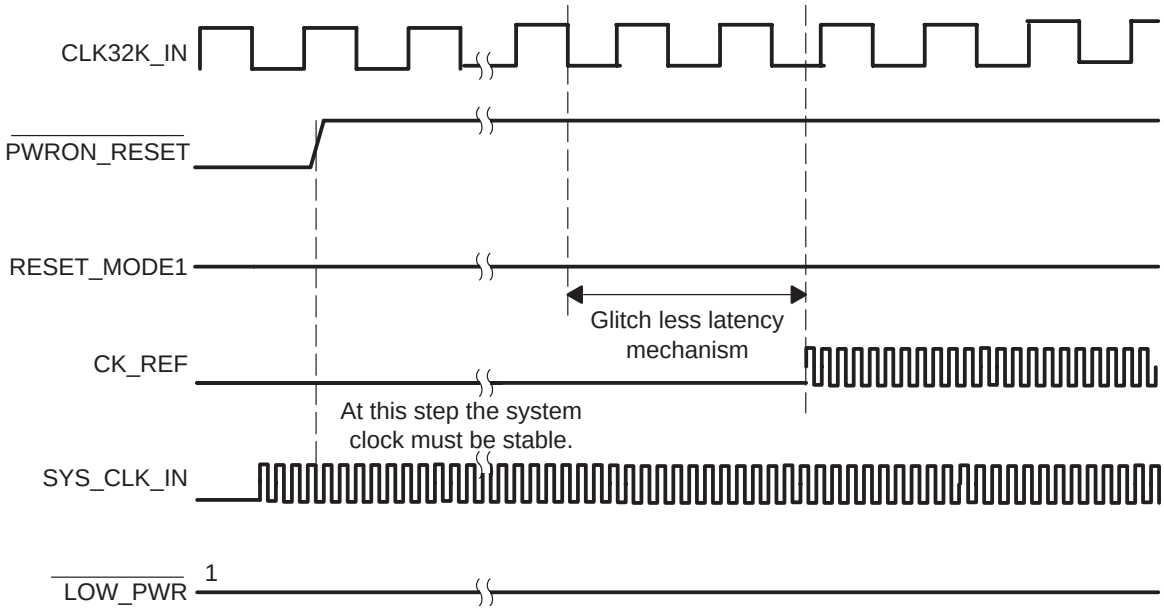
1.15.3 Power-up Sequence in External Clock Mode

In external mode, all the analog cell setup counters are bypassed at power-up reset.

It follows this sequence:

1) $\overline{\text{PWRON\_RESET}}$ is released. $\overline{\text{LOW\_PWR}}$ is reset to inactive state.
2) The input system clock is released internally in ULPD.
3) The ULPD FSM1 moves to awake mode.
4) The ULPD enables the input clock to OMAP3.2.

Figure 13. Power-up Sequence in External Clock Mode



1.16 ULPD Reset Inputs

The ULPD has five distinct reset inputs that act differently on the ULPD generated output reset.

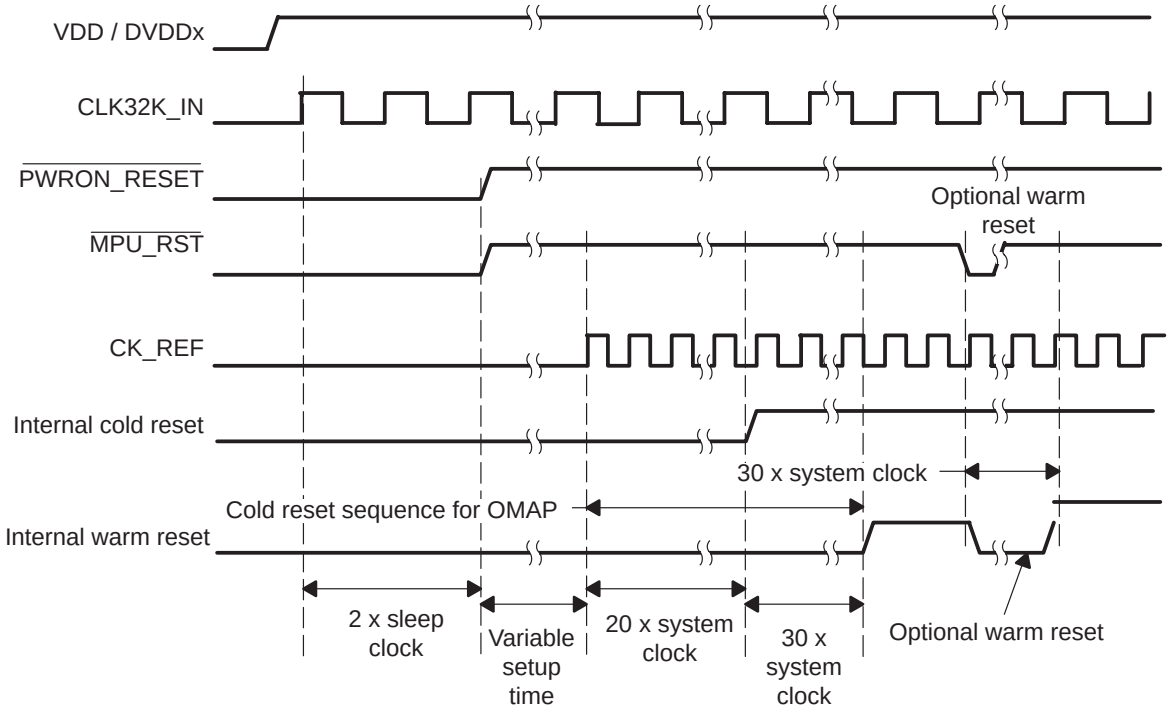
- $\overline{\text{PWRON\_RESET}}$ corresponds to the device power reset and as such must correspond to a device input pin. $\overline{\text{PWRON\_RESET}}$ is a cold reset.
- The $\overline{\text{MPU\_RST}}$ is a device global system reset and is intended to correspond to a device input pin. $\overline{\text{MPU\_RST}}$ is a warm reset.
- Security violation
- 32-kHz watchdog reset
- Secure watchdog reset

1.17 OMAP3.2 Reset Generation

The ULPD propogates the cold and warm resets to OMAP3.2.

OAMP3.2 resets the processors and peripherals. Figure 14 shows the reset procedures.

Figure 14. OMAP3.2 Input Reset Generation



1.18 OMAP3.2 Embedded LDO for DPLL[3] Control

An embedded LDO provides the DPLL of OMAP3.2 and the system clock oscillator with a quiet voltage supply.

The ULPD manages the sleep transition of this e-LDO through the LDO_SLEEP signal.

At reset, the e-LDO is not asleep.

At boot, the software can check that the e-LDO output voltage, POWER_CTRL_REG [6], is stable before switching the DPLL to lock mode.

The e-LDO can be put to sleep permanently by software with POWER_CTRL_REG [8] = 1. Before putting the e-LDO to sleep, the user must first program the DPLL into bypass mode. The device can still run out of the DPLL in bypass mode.

When POWER_CTRL_REG[7] = 1, the ULPD state machine automatically powers down the e-LDO in deep sleep mode and powers it up on exit from deep sleep mode. The ramp-up time of the e-LDO is hidden by the transition time from deep sleep mode to awake mode.

If needed, the e-LDO can be tuned by adjusting SETUP_ANALOG_CELL2 in oscillator mode or SETUP_ANALOG_CELL3 in external mode.

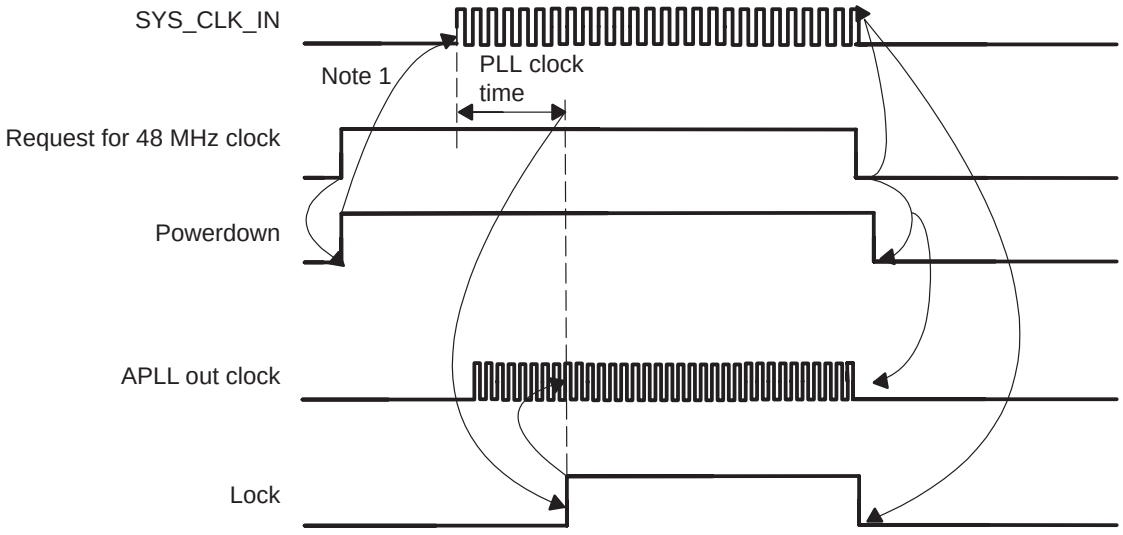
1.19 Analog Phase-Locked Loop Control

To provide a 48-MHz clock to peripherals that request it, the ULPD controls the activation and deactivation of an on-chip analog phase-locked loop (APLL). This APLL delivers a 96-MHz clock that is further divided inside the ULPD.

The ULPD enables APLL and its input clock whenever the system clock is present and a 48-MHz clock request is active.

The APLL signals to the ULPD when it reaches lock state. By reading the LOCK_STATUS bit in the ULPD_PLL_CTRL_STATUS register, the software can determine whether the 48-MHz clocks are stable.

Figure 15. ULPD_PLL Clock Management

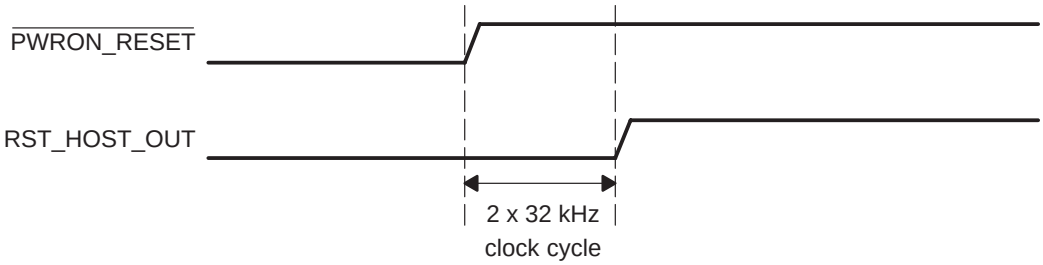


See Section 4−2 for additional information on the APLL.

1.20 Battery Failed Interrupt

After releasing $\overline{\text{PWRON\_RESET}}$, $\overline{\text{RST\_HOST\_OUT}}$ is kept low for two 32-kHz cycles, and then goes inactive high.

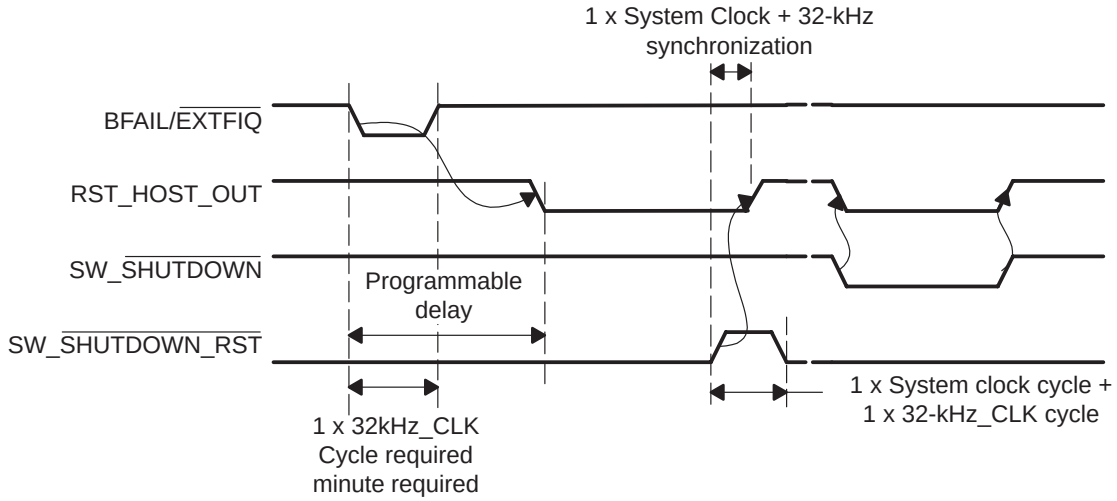
Figure 16. RST_HOST_OUT Activation on PWRON_RESET



Upon a low level on BFAIL/EXTFIQ input, which signals a battery fail event, the ULPD starts a programmable counter. When the counter underflows, ULPD asserts low RST_HOST_OUT and places OMAP3.2 in power-down mode.

The delay from the falling edge of BFAIL/EXTFIQ to the falling edge of RST_HOST_OUT is software programmable (see Table 21). The default value is one CLK32K cycle. The release of the shutdown signal is also controlled by software (see Table 28).

Figure 17. RST_HOST_OUT Activation on BFAIL/EXTFIQ and SW_SHUTDOWN



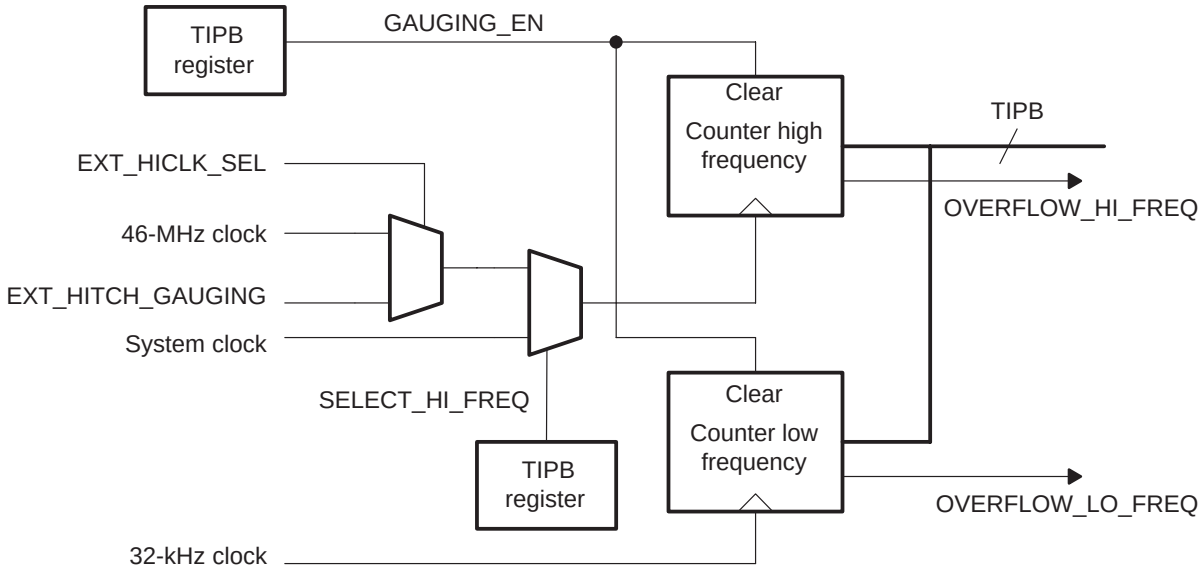
It is also possible to use SW_SHUTDOWN (bit [3] of POWER_CTRL_REG) to force low RST_HOST_OUT.

Note:

What is shown in Figure 17 as programmable delay corresponds to the counter start time (three 32-kHz cycles) plus the counter decrement to 0. The counter initial value corresponds to COUNTER_32_FIQ_REG.

1.21 32-kHz Oscillator Calibration

Figure 18. Functional Block Diagram of Gauging



Because the exact 32-kHz clock frequency is unknown, it is necessary to gauge it by comparing the 32-kHz clock with a higher frequency clock (system clock, APLL clock out, or EXT_HICLK_GAUGING) during any active period.

Note: Software Limitation

The counter is not resynchronized on the TIPB strobe. Therefore, the value is not readable while the counter is running (when gauging is enabled). The procedure is first to disable the gauging (GAUGIG_CTRLREG[0] to 0), and then to read the counter high frequency and the 32-kHz counter value.

1.22 Bad Devices

When the DEVICE_TYPE[1:0] indicates that the device is bad (DEVICE_TYPE = 01), the ULPD keeps both internal cold and warm resets asserted low. In this configuration the CPU cannot boot.

Note:

DEVICE_TYPE corresponds to EFUSE_DEVICE_TYPE.

1.23 ULPD Interrupt Generation

The ULPD generates an interrupt when one of the following events has been detected:

- An overflow occurred on the 32-kHz counter during gauging. This event is the same as the one that triggers the IT_STATUS_REG [2] bit.
- An overflow occurred on the HI_FREQ counter during gauging versus high frequency. This event is the same as the one that triggers the IT_STATUS_REG [1] bit.
- Gauging has stopped. This event is the same as the one that triggers the IT_STATUS_REG [0] bit.
- The following two conditions have been met:
  - USB_MCLK_REQ clock request has been set or SOFT_REQ_REG [3] has been active high for more than two 32-kHz clock cycles.
  - CLOCK_CTRL_REG [5] is cleared to 0.

Table 7 lists the 16-bit ULPD registers. Table 8 through Table 39 describe the register bits.

1.24 ULPD Registers

Unless otherwise specified, all of the registers in this table are reset by any warm reset (for a listing of warm resets, please see Chapter 5).

Table 7. ULPD Registers

Base Address = 0xFFFE 0800			
Name	**Description**	**R/W**	**Offset**
COUNTER_32_LSB_REG	Counter 32 LSB		0x00
COUNTER_32_MSB_REG	Counter 32 MSB		0x04
COUNTER_HIGH_FREQ_LSB_REG	Counter high-frequency LSB		0x08
COUNTER_HIGH_FREQ_MSB_REG	Counter high-frequency MSB		0x0C
GAUGING_CTRL_REG	Gauging control		0x10
IT_STATUS_REG	Interrupt status		0x14
RESERVED	Reserved		0x18
RESERVED	Reserved		0x1C
RESERVED	Reserved		0x20
SETUP_ANALOG_CELL3_ULPD1_REG	Setup analog cell3 ULPD1		0x24

Table 7. ULPD Registers (Continued)

Base Address = 0xFFFE 0800			
Name	**Description**	**R/W**	**Offset**
SETUP_ANALOG_CELL2_ULPD1_REG	Setup analog cell2 ULPD1		0x28
SETUP_ANALOG_CELL1_ULPD1_REG	Setup analog cell1 ULPD1		0x2C
CLOCK_CTRL_REG	Clock control		0x30
SOFT_REQ_REG	Software request		0x34
COUNTER_32_FIQ_REG	Counter 32 FIQ		0x38
RESERVED	Reserved		0x3C
STATUS_REQ_REG	Status request		0x40
PLL_DIV_REG	PLL division		0x44
RESERVED_48	Reserved 48		0x48
ULPD_PLL_CTRL_STATUS	ULPD PLL control status		0x4C
POWER_CTRL_REG	Power control		0x50
STATUS_REQ_REG2	Status request 2		0x54
SLEEP_STATUS	Sleep status		0x58
SETUP_ANALOG_CELL4_ULPD1_REG	Setup analog cell4 ULPD1		0x5C
SETUP_ANALOG_CELL5_ULPD1_REG	Setup analog cell5 ULPD1		0x60
SETUP_ANLOG_CELL6_ULPD1_REG	Setup analog cell6 ULPD1		0x64
SOFT_DISABLE_REQ_REG	Software disable request		0x68
RESET_STATUS	Reset status		0x6C
REVISION_NUMBER	Revision number		0x70
SDW_CLK_DIV_CTRL_SEL	SDW clock divider control select		0x74
COM_CLK_DIV_CTRL_SEL	COM clock divider control select		0x78
CAM_CLK_CTRL	CAM clock control		0x7C
SOFT_REQ_REG2	Software request 2		0x80

Table 8. Counter 32 LSB Register (COUNTER_32_LSB_REG)

Base Address = FFFE 0800, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	COUNTER_SLEEP_ CLK_LSB	Lower value of the number of sleep clock cycles during gauging time	R	0x1

Note: The CPU must check that the IT_STATUS_REG[0] bit has been asserted high before reading this register.

Table 9. Counter 32 MSB Register (COUNTER_32_MSB_REG)

Base Address = 0xFFFE 0800, Offset = 0x04				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:4	RESERVED	Reserved	R	0x0
3:0	COUNTER_32_MSB	Upper value of the number of 32-kHz clock cycles during gauging time	R	0x0

Note: The CPU must check that the IT_STATUS_REG[0] bit has been asserted high before reading this register.

Table 10. Counter High-Frequency LSB Register (COUNTER_HIGH_FREQ_LSB_REG)

Base Address = 0xFFFE 0800, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	COUNTER_HIGH_ FREQ_LSB	Lower value of the number of high-frequency clock during gauging time	R	0x1

Note: The CPU must check that the IT_STATUS_REG[0] bit has been asserted high before reading this register.

Table 11. Counter High-Frequency MSB Register (COUNTER_HIGH_FREQ_MSB_REG)

Base Address = 0xFFFE 0800, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:6	UNUSED	Unused	R	0x0
5:0	COUNTER_HIGH_ FREQ_MSB	Upper value of the number of high-frequency clock during gauging time	R	0x0

Note: The CPU must check that the IT_STATUS_REG[0] bit has been asserted high before reading this register.

Table 12. Gauging Control Register (GAUGING_CTRL_REG)

Base Address = 0xFFFE 0800, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:2	UNUSED	Unused	R/W	0x0
1	SELECT_HI_FREQ_CLOCK	1: High-frequency clock = auxiliary gauging clock 0: High-frequency clock = 12-Mhz clock	R/W	0x0
0	GAUGING_EN	1: Gauging is running. 0: Gauging is stopped.	R/W	0x0

Table 13. Interrupt Status Register (IT_STATUS_REG)

Base Address = 0xFFFE 0800, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:4	UNUSED	Unused	R	0x0
3	IT_WAKEUP_USB	Wake-up interrupt from USB W2FC	R	0x0
2	OVERFLOW_32	An overflow occurred on the 32-kHz counter during gauging.	R	0x0
1	OVERFLOW_HI_FREQ	An overflow occurred on the hi_freq counter during gauging versus high-frequency clock.	R	0x0
0	IT_GAUGING	To inform CPU that gauging is stopped and high-and low-frequency registers can be read.	R	0x0

Table 14. Reserved Register (RESERVED)

Base Address = 0xFFFE 0800, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	RESERVED	Reserved	R/W	0x3FF

Table 15. Reserved Register (RESERVED)

Base Address = 0xFFFE 0800, ?Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	RESERVED	Reserved	R/W	0x3FF

Table 16. Setup Analog Cell3 ULPD1 Register (SETUP_ANALOG_CELL3_REG)

Base Address = 0xFFFE 0800, Offset = 0x24				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	SETUP_ANALOG_CELL3	Setup time of analog cell3 in number of sleep clock cycles	R/W	0x3FF

Note: This setup stage is for the FSM1 of the ULPD.

Table 17. Setup Analog Cell2 ULPD1 Register (SETUP_ANALOG_CELL2_REG)

Base Address = 0xFFFE 0800, Offset = 0x28				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	SETUP_ANALOG_CELL2	Setup time of analog cell2 in number of sleep clock cycles	R/W	0x0

Note: This setup stage is for the FSM1 of the ULPD.

Table 18. Setup Analog Cell1 ULPD1 Register (SETUP_ANALOG_CELL1_REG)

Base Address = 0xFFFE 0800, Offset = 0x2C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	SETUP_ANALOG_CELL1	Setup time of analog cell1 in number of sleep clock cycles	R/W	0x0

Note: This setup stage is for the FSM1 of the ULPD.

Table 19. Clock Control Register (CLOCK_CTRL_REG)

Base Address = 0xFFFE 0800, Offset = 0x30				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:7	UNUSED	Unused	R/W	0x0
6	SLICER_BYPASS	Reserved	R/W	0x0
5	DIS_USB_PVCI_CLK	1: Disable USB W2FC PVCI clock. 0: Enable.	R/W	0x0
4	USB_MCLK_EN	0: Disable USB hub clock output. 1: Enable.	R/W	0x0

Table 19. Clock Control Register (CLOCK_CTRL_REG) (Continued)

Base Address = 0xFFFE 0800, Offset = 0x30				
Bit	**Name**	**Function**	**R/W**	**Reset**
3	TI_RESERVED_EN	0: Disable clock on SYS_CLK_OUT output. 1: Enable.	R/W	0x0
2	SDW_MCLK_INV	0: BLUETOOTH_CLK is low when inactive. 1: CLK is high when inactive	R/W	0x0
1	COM_MCLK_INV	0: Modem clock is low when inactive. 1: CLK is high when inactive.	R/W	0x0
0	MODEM_32K_EN	0: Disable sleep clock on UART (force to 1). 1: Enable sleep clock on UART.	R/W	0x0

Table 20. Software Request Register (SOFT_REQ_REG)

Base Address = 0xFFFE 0800, Offset = 0x34				
Bit	**Name**	**Function**	**R/W**	**Reset**
15	SOFT_CLOCK2_DPLL_REQ	PLL software request reserved for future use 1: Request active 0: Request inactive	R/W	0X0
14	SOFT_CLOCK1_DPLL_REQ	PLL software request reserved for future use. 1: Request active 0: Request inactive	R/W	0x0
13	SOFT_MMC2_DPLL_REQ	ULPD_PLL clock request for MMC2. 1: Active 0: Inactive	R/W	0x0
12	SOFT_MMC_DPLL_REQ	Software ULPD_PLL req for MMC 1: Request active 0: Request inactive	R/W	0x0
11	SOFT_UART3_DPLL_REQ	Software UART3 ULPD_PLL request. 1: Request active 0: Request inactive	R/W	0x0
10	SOFT_UART2_DPLL_REQ	Software UART2 ULPD_PLL request. 1: Request active 0: Request inactive	R/W	0x0
9	SOFT_UART1_DPLL_REQ	Software UART1 ULPD_PLL request. 1: Request active 0: Request inactive	R/W	0x0

Table 20. Software Request Register (SOFT_REQ_REG) (Continued)

Base Address = 0xFFFE 0800, Offset = 0x34				
Bit	**Name**	**Function**	**R/W**	**Reset**
8	SOFT_USB_OTG_DPLL_REQ	Software request for USB OTG for ULPD_PLL clock. 1: Request active 0: Request inactive	R/W	0x0
7	SOFT_CAM_DPLL_MCKO_REQ	Software camera ULPD_PLL request. 1: Request active 0: Request inactive	R/W	0x0
6	SOFT_COM_MCKO_REQ	Software request for COM_MCKO clock. 1: Request active 0: Request inactive	R/W	0x0
5	SOFT_PERIPH_$\overline{\text{REQ}}$	Software system clock request for UART2. 1: Request active 0: Request inactive	R/W	0x0
4	USB_REQ_EN	0: Disable USB client hardware DPLL request 1: Enable	R/W	0x1
3	SOFT_USB_REQ	Software system clock request for USB host. 1: Request active 0: Request inactive	R/W	0x0
2	SOFT_SDW_REQ	Software systm clock request for Bluetooth. 1: Request active 0: Request inactive	R/W	0x0
1	SOFT_COM_REQ	Software system clock request for com processor 1: Request active 0: Request inactive	R/W	0x0
0	SOFT_DPLL_REQ	Software ULPD_PLL clock request 1: Request active 0: Request inactive	R/W	0x0

Table 21. Counter 32 FIQ Register (COUNTER_32_FIQ_REG)

Base Address = 0xFFFE 0800, Offset = 0x38				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	UNUSED	Unused	R/W	0x0
7:0	COUNTER_32_FIQ	Number of 32-kHz clock cycles to delay active modem shutdown signal after receiving FIQ	R/W	0x01

Table 22. Reserved Register (RESERVED)

Base Address = 0xFFFE 0800, Offset = 0x3C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	RESERVED	Reserved	R	0x0

Table 23. Status Request Register (STATUS_REQ_REG)

Base Address = 0xFFFE 0800, Offset = 0x40				
Bit	**Name**	**Function**	**R/W**	**Reset**
15	CLOCK3_DPLL_REQ	ULPD_PLL clock request from request RESERVED3 1: Active 0: Inactive	R	Unknown
14	CLOCK2_DPLL_REQ	ULPD_PLL clock request from request RESERVED2 1: Active 0: Inactive	R	Unknown
13	CLOCK1_DPLL_REQ	ULPD_PLL clock request from request RESERVED1 1: Active 0: Inactive	R	Unknown
12	MMC_DPLL_REQ	ULPD_PLL clock request from MMC 1: Request active 0: Request inactive	R	Unknown
11	UART3_DPLL_REQ	ULPD_PLL clock request from UART3 1: Request active 0: Request inactive	R	Unknown
10	UART2_DPLL_REQ	ULPD_PLL clock request from UART2 1: Request active 0: Request inactive	R	Unknown
9	UART1_DPLL_REQ	ULPD_PLL clock request from UART1 1: Request active 0: Request inactive	R	Unknown
8	USB_HOST_DPLL_REQ	ULPD_PLL clock request from USB host 1: Request active 0: Request inactive	R	Unknown
7	CAM_DPLL_MCLK_REQ	ULPD_PLL clock request from camera interface 1: Request active 0: Request inactive	R	Unknown

Table 23. Status Request Register (STATUS_REQ_REG) (Continued)

Base Address = 0xFFFE 0800, Offset = 0x40				
Bit	**Name**	**Function**	**R/W**	**Reset**
6	USB_DPLL_MCLK_REQ	ULPD_PLL clock request from USB client 1: Request active 0: Request inactive	R	Unknown
5	USB_MCLK_REQ	Hardware system clock request by USB client 1: Request active 0: Request inactive	R	Unknown
4	SDW_MCLK_REQ	Hardware system clock request from Bluetooth 1: Request active 0: Request inactive	R	Unknown
3	COM_MCLK_REQ	System clock request from com processor 1: Request active 0: Request inactive	R	Unknown
2	PERIPH_$\overline{\text{REQ}}$	System clock request from UART2 0: Request active 1: Request inactive	R	Unknown
1	WAKEUP_$\overline{\text{REQ}}$	System clock request from OMAP 0: Request active 1: Request inactive	R	Unknown
0	CHIP_IDLE	Sleep request received from OMAP 1: Request active 0: Request inactive	R	Unknown

Note: Bits 15:3 in this register reflect the state of the clock request regardless of whether they are masked or not (by the corresponding SOFT_DISABLE_REQ_REG bit).

Table 24. PLL Division Register (PLL_DIV_REG)

Base Address = 0xFFFE 0800, Offset = 0x44				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	PLL_DIV_FACTOR	PLL clk out division factor. Bypassed if programmed value is 0, else: 0001h: divided by 2 0002h: divided by 4 0004h: divided by 8 .... 8000h: divided by 65536	R/W	0x0

Table 25. Reserved Register (RESERVED_48)

Base Address = 0xFFFE 0800, Offset = 0x48				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	RESERVED	Kept for software compatibility reason. Has no effect on ULPD behavior.	R/W	0x960

Table 26. ULPD PLL Control Status Register (ULPD_PLL_CTRL_STATUS)

Base Address = 0xFFFE 0800, Offset = 0x4C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15	LOCK_STATUS	Gives the lock status of the module that provides the high frequency clock 1: PLL locked 0: PLL unlocked	R	Unknown
14:3	PLL_CTRL_RES	Reserved	R/W	0x0
2:0	PLL_CONTROL	Selects the APLL mode. 000: 19.2 MHz 010: 13 MHz 011: 12 MHz	R/W	0x3

Table 27. Power Control Register (POWER_CTRL_REG)

Base Address = 0xFFFE 0800, Offset = 0x50				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:13	UNUSED	Unused	R	0x0
12	ISOLATION_CONTROL	0: Electrical isolation inactive 1: Electrical isolation active Reset of this bit is done on powerup reset only.@@@@	R/W	0x0
11	MIN_MAX_REG	1: Operation at minimum voltage 0: Operation at nominal voltage	R/W	0x1
10	DVS_ENABLE	0: DVS feature disabled 1: DVS feature enabled	R/W	0x0
9	OSC_STOP_EN	0: The oscillator is not stopped in deep sleep mode. 1: The oscillator is stopped in deep sleep mode.	R/W	0x1

Table 27. Power Control Register (POWER_CTRL_REG) (Continued)

Base Address = 0xFFFE 0800, Offset = 0x50				
Bit	**Name**	**Function**	**R/W**	**Reset**
8	SOFT_LDO_SLEEP	Control the sleep of the e-LDO that supplies the DPLL 1=> LDO sleep is forced to active state. 0=> LDO not in sleep (except in deep sleep mode if LDO_CTRL_EN=1)	R/W	0x0
7	LDO_CTRL_EN	0: The sleep of the LDO is fully controlled by SOFT_LDO_SLEEP bit. 1: The LDO is powered down in deep sleep mode OR if SOFT_LDO_SLEEP=1.	R/W	0x0
6	LDO_STEADY	Stability of LDO output voltage status 0: LDO output voltage not stable 1: LDO output voltage stable	R	Unknown
5	UNUSED	Unused	R	0x0
4	DEEP_SLEEP_TRANSITION_EN	1: Transition to deep sleep mode allowed 0: Transition to deep sleep mode forbidden	R/W	0x1
3	SW_$\overline{\text{SHUTDOWN}}$	Software generation of RST_HOST_OUT 0: RST_HOST_OUT forced to low 1: RST_HOST_OUT not forced to low	R/W	0x1
2	SW_$\overline{\text{SHUTDOWN}}$_RST	Allow to deactivate the output RST_HOST_OUT when at 1	R/W	0x0
1	LOW_PWR_REQ	Low-power software request. 0: Does not force the LOW_PWR signal to active state 1: Forces LOW_PWR signal to active state	R/W	0x0
0	LOW_PWR_EN	0: Low-power feature is disabled. 1: Low-power feature is available.	R/W	0x0

Table 28. Status Request Register 2 (STATUS_REQ_REG2)

Base Address =0xFFFE 0800, Offset = 0x54				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:1	UNUSED	Unused	R	0x0
0	MMC2_DPLL_REQ	Status of the MMC2_PLL_REQ 0: Inactive 1: Active	R	Unknown

Note: Bit 0 in this register reflects the state of the clock request regardless of whether it is masked or not (by the corresponding SOFT_DISABLE_REQ_REG bit).

Table 29. Sleep Status Register (SLEEP_STATUS)

Base Address = 0xFFFE 0800, Offset = 0x58				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:2	UNUSED	Unused	R	0x0
1	BIG_SLEEP	This bit is asserted (1) when waking up from big sleep. This bit is cleared (0) when the FSM goes out of awake.	R	Unknown
0	DEEP_SLEEP	This bit is asserted (1) when waking up from deep sleep. This bit is cleared (0) when the FSM goes out of awake.	R	Unknown

Note: Both bits can be read at 1 in case ULPD goes from deep sleep to big sleep before the transition back to awake.

Table 30. Setup Analog Cell4 ULPD1 Register (SETUP_ANALOG_CELL4_REG)

Base Address = 0xFFFE 0800, Offset = 0x5C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	SETUP_ANALOG_CELL4	Setup time of analog cell4 in number of sleep clock cycles	R/W	0x0

Note: This setup stage is for the FSM1 of the ULPD.

Table 31. Setup Analog Cell5 ULPD1 Register (SETUP_ANALOG_CELL5_REG)

Base Address = 0xFFFE 0800, Offset = 0x60				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	SETUP_ANALOG_CELL5	Setup time of analog cell5 in number of sleep clock cycles	R/W	0x0

Note: This setup stage is for the FSM1 of the ULPD.

Table 32. Setup Analog Cell6 ULPD1 Register (SETUP_ANALOG_CELL6_REG)

Base Address = 0xFFFE 0800, Offset = 0x64				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	SETUP_ANALOG_CELL6	Setup time of analog cell6 in number of sleep clock cycles	R/W	0x0

Note: This setup stage is for the FSM1 of the ULPD.

Table 33. Software Disable Request Register (SOFT_DISABLE_REQ_REG)

Base Address =0xFFFE 0800, Offset = 0x68				
Bit	**Name**	**Function**	**R/W**	**Reset**
15	UNUSED	Unused	R	0x0
14	DIS_CLOCK3_DPLL_REQ	Disable for the PLL hardware request reserved (CLOCK3_DPLL_REQ). 0: Not disabled 1: Disabled	R/W	0x0
13	DIS_CLOCK2_DPLL_REQ	Disable for the PLL hardware request reserved (CLOCK2_DPLL_REQ). 0: Not disabled 1: Disabled	R/W	0x0
12	DIS_CLOCK1_DPLL_REQ	Disable for the PLL hardware request reserved (CLOCK1_DPLL_REQ). 0: Not disabled 1: Disabled	R/W	0x0
11	DIS_MMC2_DPLL_REQ	Disable for hardware request MMC2_DPLL_REQ. 0: Not disabled 1: Disabled	R/W	0x0
10	DIS_MMC_DPLL_REQ	Disable the current hardware request if active 0: Not disabled 1: Disabled	R/W	0x0
9	DIS_UART3_DPLL_REQ	Disable hardware request for UART3. 0: Not disabled 1: Disabled	R/W	0x0
8	DIS_UART2_DPLL_REQ	Disable UART2 PLL hardware request. 0: Not disabled 1: Disabled	R/W	0x0

Table 33. Software Disable Request Register (SOFT_DISABLE_REQ_REG) (Continued)

Base Address =0xFFFE 0800, Offset = 0x68				
Bit	**Name**	**Function**	**R/W**	**Reset**
7	DIS_UART1_DPLL_REQ	Disable UART1 PLL hardware request 0: Not disabled 1: Disabled	R/W	0x0
6	DIS_USB_HOST_DPLL_REQ	Disable the USB host system clock hardware 0: Not disabled 1: Disabled	R/W	0x0
5	DIS_CAM_DPLL_MCLK_REQ	Disable hardware CAM_PLL_MCLK_REQ. 0: Not disabled 1: Disabled	R/W	0x0
4	UNUSED	Unused	R/W	0x0
3	DIS_PERIPH_REQ	Disable hardware PERIPH_REQ request. 0: Not disabled 1: Disabled	R/W	0x0
2	UNUSED	Unused	R/W	0x0
1	DIS_SDW_MCLK_REQ	Disable SDW_MCLK_REQ if active 0: Not disabled 1: Disabled	R/W	0x0
0	DIS_COM_MCLK_REQ	Disable COM_MCLK_REQ and COM_MCKO_SEL if active 0: Not disabled 1: Disabled	R/W	0x0

Table 34. Reset Status Register (RESET_STATUS)

Base Address = 0xFFFE 0800, Offset = 0x6C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:4	UNUSED	Unused	R	0x0
3	32K watchdog time-out	Whenever a 32-kHz watchdog time-out event occurs, this bit is asserted. The user clears this bit by writing a 0. TIPB reset has no effect on this bit. 0: 32-kHz watchdog time-out has not occurred since last PWRON_RESET (or user has cleared this bit). 1: 32-kHz watchdog time-out has occurred since last PWRON_RESET (and since last time user has cleared this bit).	R/W	0x0

Table 34. Reset Status Register (RESET_STATUS) (Continued)

Base Address = 0xFFFE 0800, Offset = 0x6C				
Bit	**Name**	**Function**	**R/W**	**Reset**
2	Security Violation	Whenever a security violation event occurs, this bit is asserted.The user clears this bit by writing a 0. TIPB reset has no effect on this bit. 0: Security violation has not occurred since last PWRON_RESET (or user has cleared this bit). 1: Security violation has occurred since last PWRON_RESET (and since last time user has cleared this bit).	R/W	0x0
1	Secure watchdog time-out	Whenever a secure watchdog time-out event occurs, this bit is asserted.The user clears this bit by writing a 0. TIPB reset has no effect on this bit. 0: Secure watchdog time-out has not occurred since last PWRON_RESET (or user has cleared this bit). 1: Secure watchdog time-out has occurred since last PWRON_RESET (and since last time user has cleared this bit).	R/W	0x0
0	POWER_ON_RESET	Whenever a $\overline{\text{PWRON\_RESET}}$ reset occurs, this bit is asserted.The user clears this bit by writing a 0 TIPB reset has no effect on this bit. 0: $\overline{\text{PWRON\_RESET}}$ not reset (or user has cleared this bit) 1: $\overline{\text{PWRON\_RESET}}$ reset	R/W	0x1

Note: This register is reset by $\overline{\text{PWRON\_RESET}}$ and not by the TIPB reset.

Table 35. Revision Number Register (REVISION_NUMBER)

Base Address = 0xFFFE 0800, Offset = 0x70				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	UNUSED	Unused	R	0x0
7:0	REVISION_NUMBER	Indicates the revision number of the ULPD. The revision number must be read as follows. [7..4].[3..0]. For example, 0x14 must be read like 1.4 revision.	R	Unknown

Note: The 4 LSBs indicate a minor revision, and the 4 MSBs indicate a major revision.

Table 36. SDW Clock Divider Control Select Register (SDW_CLK_DIV_CTRL_SEL)

Base Address = 0xFFFE 0800, Offset = 0x74				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	UNUSED	Unused	R	0x0
7:2	SDW_RATIO_SEL	Select the divider ratio to apply to the APLL output clock to generate BCLK. 000000=>1; 000001=>1.5; 000010=>2; 000011=>2.5 000100=>3; 000101=>3.5; 000110=>4; 000111=>5; 001000=>6; 001001=>7; .... 010010=>16 ... 110010=>48.	R/W	0x0
1	SDW_ULPD_PLL_CLK_REQ	BCLK clock software request. 0: Request inactive 1: Request active	R/W	0x0
0	SDW_SYSCLK_PLLCLK_SEL	0: Select the divided version of APLL output clock for BCLK 1: Select SYSTEM_CLOCK for BCLK	R/W	0x1

Table 37. COM Clock Divider Control Select Register (COM_CLK_DIV_CTRL_SEL)

Base Address = 0xFFFE 0800, Offset = 0x78				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	UNUSED	Unused	R	0x0
7:2	COM_RATIO_SEL	Select the divider ratio to apply to the APLL output clock to generate MCLK. 000000=> 1 000001=>1.5; 000010=>2; 000011=>2.5 000100=> 3; 000101=>3.5 000110=>4; 000111=>5 001000=>6 001001=>7 010010=>16 110010=>48	R/W	0x0

Table 37. COM Clock Divider Control Select Register (COM_CLK_DIV_CTRL_SEL) (Continued)

Base Address = 0xFFFE 0800, Offset = 0x78				
Bit	**Name**	**Function**	**R/W**	**Reset**
1	COM_ULPD_PLL_CLK_REQ	MCLK clock software request 0: Request inactive 1: Request active	R/W	0x0
0	COM_SYSCLK_PLLCLK_SEL	0: Select the divided version of APLL output clock for MCLK 1: Select SYSTEM_CLOCK for MCLK or 48 MHz	R/W	0x1

Table 38. CAM Clock Control Register (CAM_CLK_CTRL)

Base Address = 0xFFFE 0800, Offset = 0x7C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:3	UNUSED	Unused	R	0x0
2	SYSTEM_CLK_EN	Clock enable of the system clock for GPIO modules 0: Clock disabled 1: Clock enabled	R/W	0x0
1	CAM_CLK_DIV	When 0, the CAM.CLKOUT is the system clock. When 1, the CAM.CLKOUT is the system clock.	R/W	0x0
0	CAM_CLOCK_EN	Enable of the CAM.CLKOUT. When 0, the CAM.CLKOUT is off. When 1, the CAM.CLKOUT is on.	R/W	0x0

Table 39. Software Request Register2 (SOFT_REQ_REG2)

Base Address = 0xFFFE 0800, Offset = 0x80				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:1	UNUSED	Unused	R	0x0
0	SOFT_CLOCK3_DPLL_REQ	PLL software request reserved for future use. 1: Request active 0: Request inactive	R/W	0X0

2 Power System Overview

This section provides guidelines for optimizing and reducing the overall device power dissipation.

For a global perspective, see Chapter 4, *Clocks*, and Chapter 5, *Initialization*.

Power consumption has two components:

- Dynamic power consumption
- Static power consumption because of leakage currents

To minimize dynamic power consumption, one of the following must be done:

- Minimize the activity in the circuit by enabling the automatic clock gating or by switching off some unused resources (MGS3/DSP or other module)
- Adjust the clock frequency on each clock domain according to the required MIPS or required bandwidth
- Minimize the external voltage supply (dynamic voltage scaling) if the application does not require the maximum clock frequencies

To minimize the static power consumption or leakage currents when the circuit is in stand-by mode, minimize the external voltage supply once the circuit is in deep sleep mode.

The OMAP5912 architecture implements hardware features so that various power management strategies can be enacted.

2.1 Power Domains

Table 40 lists all OMAP5912 power supplies with their respective nominal value and power domains.

Figure 19 shows how those power domains are connected or can be isolated from one another.

Table 40. Power Domains With Associated Power Supply and Planes

Power Domains	Power Supply	Power Planes
MPU DOMAIN	CVDD (1.05 V−1.65 V)	Core: ASIC gates
	CVDD1 (1.05 V−1.65 V)	Core: Arm926EJS
	CVDD2 (1.05 V−1.65 V)	Core: DPLL
	LDO.FILTER	Core: Analog
	CVDDa (1.6 V)	Core: SDRAM interface
	CVDDDLL (1.6 V)	Display interface
	DVDD1 (1.8 V−2.75 V)	USB interface
	DVDD2 (1.8 V, 3.3 V)[1]	Miscellaneous interfaces
	DVDD3 (1.8 V−3 V)	Peripheral: SDRAM interface
	DVDD4 (1.8 V−2.75 V)[2]	Peripheral: flash interface
	DVDD5 (1.8 V−2.75 V)	MMC interface
	DVDD6 (1.8 V−2.75 V)	Miscellaneous interfaces
	DVDD7 (1.8 V− 2.75 V)	Camera interface
	DVDD (1.8 V−3 V)	Miscellaneous interfaces
	DVDD9 (1.8 V−2.75 V)	Peripheral: RTC I/O
	DVDDRTC (1.8 V−2.75 V)	
DSP DOMAIN	CVDD3 (1.05 V−1.65 V)	Core: DSP
RTC DOMAIN	CVDDRTC (1.6 V)	Core: RTC

Notes: 1) If the integrated USB transceivers are used, the spec for DVDD2 should be nominal 3.3V. If the integrated USB transceivers are not intented to be used, the spec for DVV2 should be either nominal 3.3V or nominal 1.8V

Notes: 2) The SDRAM interface voltage includes 3.3V. You can power DVDD4 in the 3.0~3.6 power range. When using 3.3V power, CONF_VOLTAGE_SDRAM_R bit should be set to 1 (2.75V) for better performance.

Notes: 3) Refer to the *OMAP5912 Data Manual (SPRS231)* for more information on nominal voltage min/max ranges.

OMAP 5912 can be split into power planes and power domains:

- A power plane is composed of components that are supplied by a dedicated power rail. Components from two different power plans can be interconnected (bus interface, peripheral interfaces, and so on.). This interconnection creates leakage current between power planes.
- A power domain is composed of one or several power planes. Components from two different power domains can be interconnected (bus interface, peripheral interfaces, and so on). This interconnection

through isolation layer or level-shifter controls can be electrically cut. In this case, we reduce sensibly leakage current between power domains.

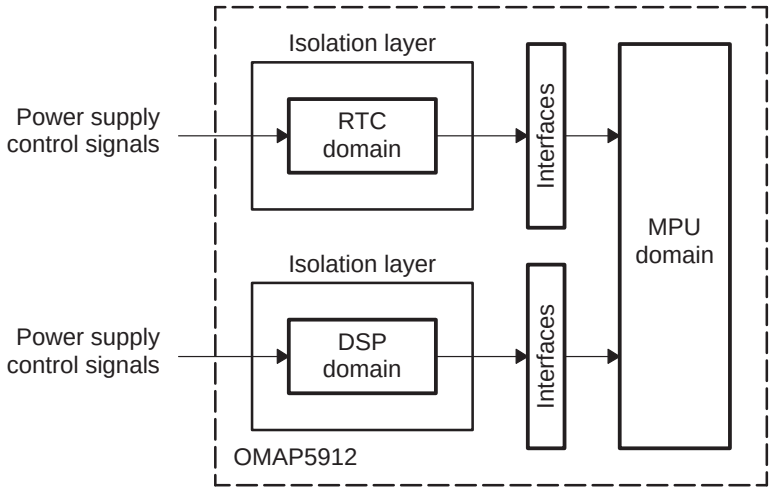
OMAP5912 is divided into three power domains:

- MPU
- DSP
- RTC

Power domains allow users to reduce power during specific system phases (idle phases). This reduction is achieved by switching off one or several domains and allowing others to be powered.

Figure 19 describes the OMAP5912 power domains.

Figure 19. OMAP5912 Power Domains



The different OMAP5912 power domains (RTC, MPU, and DSP) can be powered as shown in Table 41.

Table 41. Powering OMAP5912 Domains

	RTC	MPU	DSP
State 1	V†	V	V
State 2	V	V	0 V
State 3	V	0 V	0 V

† V = 1.5 V or 1.1 V.

In any given state, V is the same voltage for all the domains.

2.2 Clock Domain

OMAP5912 can be split into clock domains and subdomains.

As shown in Figure 20, OMAP5912 is divided into four clock domains.

The RTC clock domain is totally independent and can be isolated by hardware from other clock domains.

The clock and reset management (CLKRST) module manages clocks for the MPU, DSP and traffic controller. For each clock domain, it is possible to adjust the frequency for each clock.

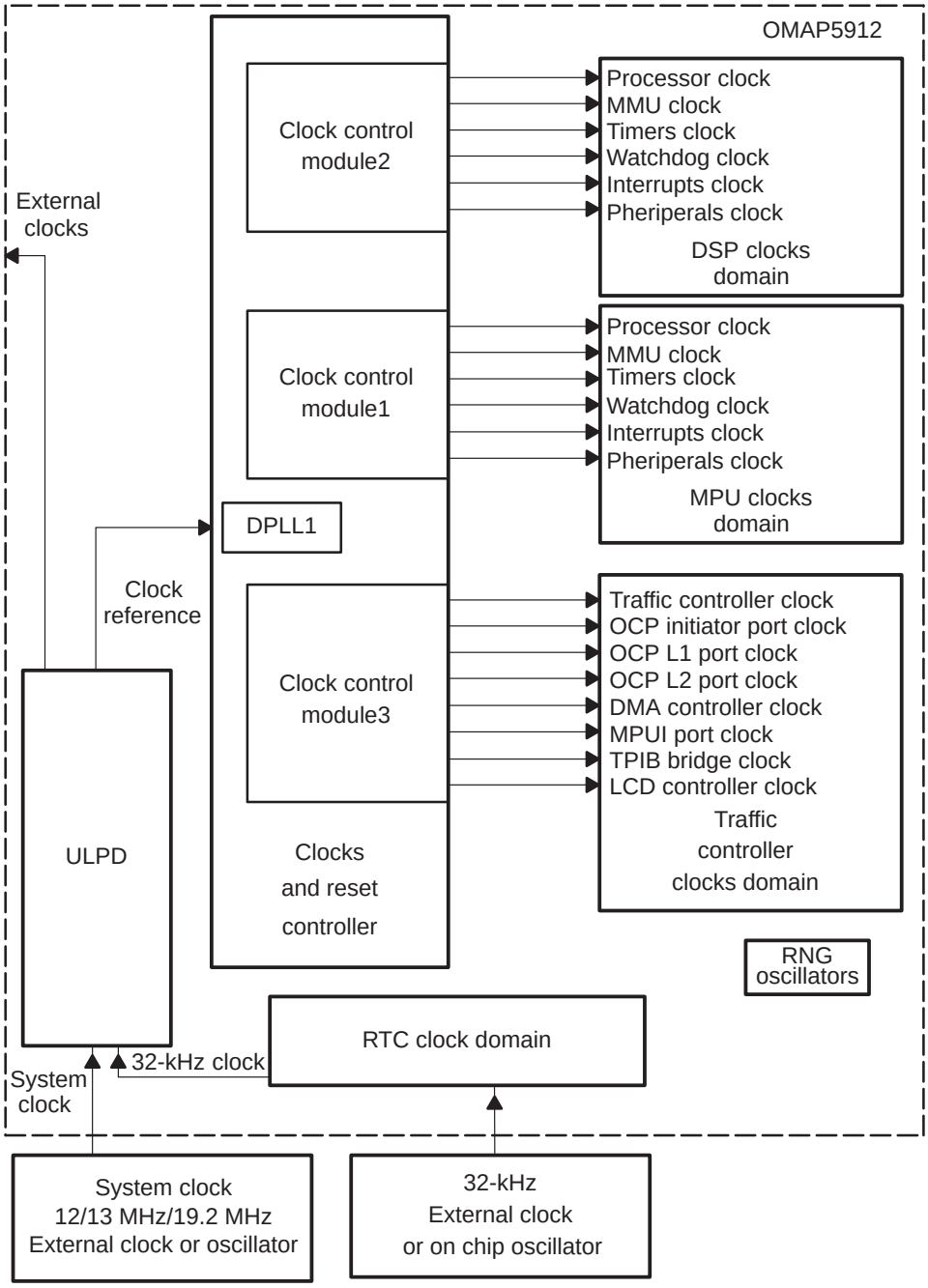
The DPLL1 output frequency is programmable and can be further divided down to provide specific clock values to each domain.

Each domain is further subdivided into subdomains that can be independently activated or deactivated.

This flexible architecture allows different power-saving mechanisms for different users.

For a complete overview of the clock and reset module, see chapter 2 *OMAP3.2 Subsystem.*

Figure 20. OMAP5912 Clock Domains



3 Power Management User Services

3.1 Power Services

Power services include software and hardware mechanisms that allow the user to reduce OMAP5912 chip power-consumption during all system phases.

3.2 Static Clock Management

During major system phases, the clocks of each domain can be immediately validated or reconfigured by software register modification. Each new static configuration is held until the next control register modifications. This management is defined as static because no external hardware events are allowed to modify the last clock tree configuration .

3.2.1 DPLL1 Clock

DPLL1 synthesizes a frequency clock from an input clock reference. Each clock domain can use the DPLL clock or directly clock reference to build each clock subdomain.

Static management of the DPLL clock allows users to reduce global chip consumption through one or several clock domain controls.

Action to reduce consumption:

- ❏ Reduce DPLL frequency value (divider register configuration)

Clock domain control through DPLL configuration is done by software writes to CLKRST register *DPPL1_CTL*.

See Section 6 for software configuration details.

3.2.2 DSP/MPU/TRAFFIC Clocks

Static management allows users to reduce global chip consumption through one or several specific clock subdomain controls.

Actions to reduce consumption:

- ❏ Reduce clock subdomain frequency value (divider register configuration).
- ❏ For each domain that is idled, deactivate the respective clocks (bit register configuration).

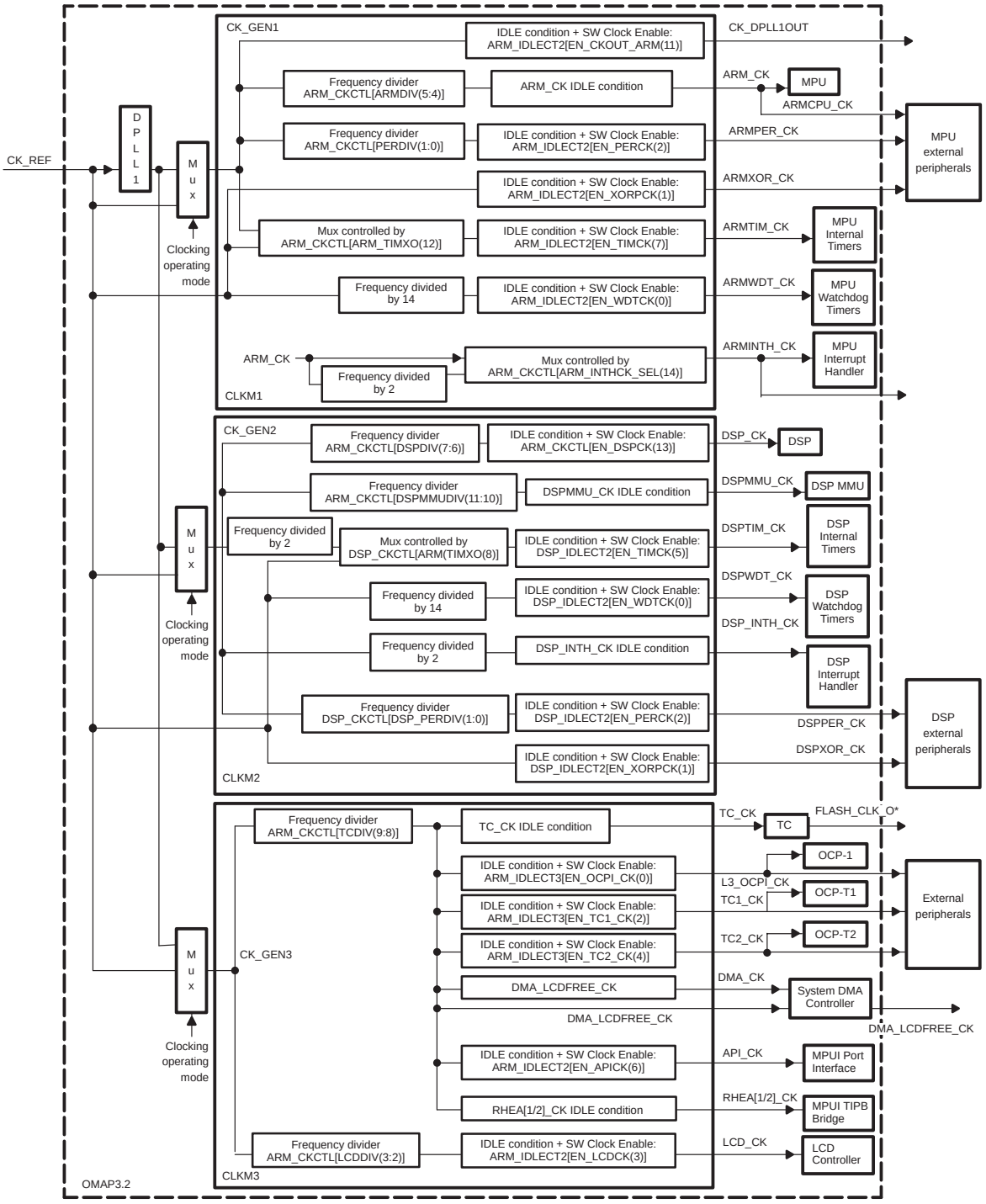
In all cases, software is responsible for determining and managing the effects of clock modifications on the system (error transitions, performance decrease, transition delays, and so on).

Table 42 describes all OMAP3.2 subsystem clocks, and Figure 21 provides an overview of their generation.

Table 42. OMAP3.2 Subsystem Clocks

Clock Name	Description
CK_DPLL1OUT	Clock from DPPL1, same as MPU clock
ARMCPU_CK	Clock with same frequency as MPU clock, same as ARM_CK
ARMXOR_CK	MPU peripheral clock, fixed, generated from CK_GEN1, can be gated
ARMPER_CK	MPU peripheral clock, divided from CK_GEN1, can be gated
ARM_INTH_CK	MPU clock interrupt handler
DSPXOR_CK	DSP peripheral clock, fixed, generated from CK_GEN2, can be gated
DSPPER_CK	DSP peripheral clock, divided from CK_GEN2, can be gated
TC_CK	Clock from EMIFS
TC1_CK	Clocks from OCP-T1
TC2_CK	Clocks from OCP-T2
L3_OCPI_CK	Same frequency as TC clock
DMA_LCDFREE_CK	Interface clock for LCD

Figure 21. OMAP3.2 Clock Generation



Clock subdomain control is done by software write to the CLKRST registers:

- ARM_CKCTL
- ARM_IDLECT1
- ARM_IDLECT2
- ARM_SYSST
- DSP_CKCTL
- DSP_IDLECT1
- DSP_IDLECT2
- DSP_SYSST

As shown in Figure 21, the clock reference to build all other clocks is provided and controlled by the ULPD (ultralow-power device) module.

The ULPD module performs several functions, which can be divided into three groups:

- Power-mode control
- Clock reference management and calibration
- External clocks management

The ULPD can manage three principal OMAP 5912 power states: awake, big sleep, and deep sleep.

ULPD management directly affects activation or deactivation of the CLKREF clock.

Note:

CLKRST static configuration affects ULPD dynamic-power mode mechanisms.

3.2.3 DSP (MGS3) Clocks Management

Global Power Management

Power can be globally managed by the idle configuration register (ICR) located in the TIPB bridge.

Table 43. Idle Configuration Register (ICR)

MPU Base Address (byte) = 0xE100 0000, DSP Base Address (word) = 0x00 0000, Offset = 0x01 (word)				
Bit	**Name**	**Function**	**R/W ***	**Reset**
15:8	Reserved	Reserved	R	0
7	RSV	Reserved idle domain.	R/W	0

Table 43. Idle Configuration Register (ICR) (Continued)

MPU Base Address (byte) = 0xE100 0000, DSP Base Address (word) = 0x00 0000, Offset = 0x01 (word)				
Bit	**Name**	**Function**	**R/W ***	**Reset**
6	RSV	Reserved idle domain.	R/W	0
5	EMIF	EMIF idle domain.	R/W	0
4	DPLL	DPLL idle domain.	R/W	0
3	PER	PER idle domain.	R/W	0
2	CACHE	CACHE idle domain.	R/W	0
1	DMA	DMA idle domain.	R/W	0
0	CPU	CPU idle domain.	R/W	0

Note: The R/W values indicate DSP access only. MPU access is read only.

This register defines six domains:

- CPU
- DMA
- ICache
- Peripherals
- Clock generator (DPLL)
- EMIF

To put one or more domains in idle mode, the user must set the corresponding bits in this register to 1 and then execute the DSP idle instruction.

A second register, the idle status register (ISTR), also located in the TIP bridge, reflects the state of the DSP when the idle instruction is executed. Table 44 defines ISTR.

Table 44. Idle Status Register (ISTR)

MPU Base Address (byte) = 0xE100 0000, DSP Base Address (word) = 0x00 0000, Offset = 0x02 (word)				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	Reserved	Reserved	R	0
7	RHEA_IDLE7_TR	Reserved idle status	R	0
6	RHEA_IDLE6_TR	Reserved idle status	R	0
5	RHEA_IDLEEMIF_TR	EMIF idle status	R	0
4	RHEA_IDLEDPLL_TR	DPLL idle status.	R	0
3	RHEA_IDLEPERH_TR	PER idle status.	R	0

Table 44. Idle Status Register (ISTR) (Continued)

MPU Base Address (byte) = 0xE100 0000, DSP Base Address (word) = 0x00 0000, Offset = 0x02 (word)				
Bit	**Name**	**Function**	**R/W**	**Reset**
2	RHEA_IDLECACHE_TR	CACHE idle status.	R	0
1	RHEA_IDLEDMA_TR	DMA idle status.	R	0
0	RHEA_IDLECPU_TR	CPU idle status.	R	0

The TIPB bridge performs some checks when the DSP idle instruction is executed:

- If the DPLL bit is set to 1 in the ICR and one or more of the CPU, DMA, EMIF, or CACHE bits are set to 0, a bus error is generated and the idle request is canceled.
- Before setting the DPLL into idle, the TIPB bridge checks whether DMA, EMIF, and CACHE are inactive. If they are not, it waits for these blocks to become inactive before entering idle mode.

All peripherals are part of the same domain. Therefore, when the peripherals domain is put in idle mode, the clocks of all peripherals are stopped.

TIPB and MPU blocks are not part of any domain. The DPLL must be disabledto put them in idle mode.

If some blocks are in idle mode but both the CPU and the DPLL are not, the program flow is maintained. It is thus possible to exit from idle mode by writing in the ICR register and by executing an idle instruction. This is the only wake procedure under software control; the other wake procedures are under hardware control.

Local Power Management

The EMIF global control register (see Table 45) allows control of the SDRAM clock (it can be enabled or stopped) and the SBSRAM clock (it can be enabled or stopped and the frequency can also be divided by 2). The EMIF external bus can also be placed in low-power state with this register.

Table 45. EMIF Global Control Register (GCR)

MPU Base Address (byte) = 0xE100 0800, DSP Base Address (word) = 0x00 0800, Offset = 0x00 (word)				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:11	RESERVED	Reserved	R	0
10:9	MEMFREQ	MEMory clock FREQuency MEMFREQ = 00: SBSRAM and/or the SDRAM interface is configured for 1x mode and the CLKMEM clock frequency is equal to the DSP clock frequency MEMFREQ = 01: SBSRAM and/or the SDRAM interface is configured for 1/2x mode and the CLKMEM clock frequency is equal to the DSP clock frequency divided by 2. MEMFREQ = 10 or 11: Reserved for future clock rates.	R	0
8	RESERVED	Reserved	R/W	0
7	WPE	Write posting enable WPE = 0, write posting is disabled (for debug). WPE = 1, write posting is enabled.	R/W	0
6	RESERVED	Reserved	R/W	0
5	MEMCEN	MEM Clock ENable MEMCEN = 0, CLKMEM held high MEMCEN = 1, CLKMEM enabled to clock	R/W	1
4	ARDYOFF	Async. ready off.	R/W	0
3	ARDY	Value of the ARDY input.	R/W	0
2	HOLD	Value of HOLD input	R/W	0
1	HOLDA	Value of HOLDA output	R/W	0
0	NOHOLD	External HOLD disable NOHOLD = 0, hold enabled NOHOLD = 1, hold disabled	R/W	0

The TIPB control mode register (CMR) includes wait state bits that can be used to set the strobe frequency of the peripherals. Table 46 defines CMR.

Table 46. TIPB Control Mode Register (CMR)

MPU Base Address (byte) = 0xE100 0000, DSP Base Address (word) = 0x00 0000, Offset = 0x00 (word)					
Bit	**Name**	**Description**	**Reset**	**CPU Access**	**MPU Access**
[15−9]	Timeout(6:0)	Strobe cycles	0x7F	Read/Write	Read
[8]	Wait State 3 (strobe 1)	Strobe 1 length (high bit)	0	Read/Write	Read
[7]	Wait State 2 (strobe 1)	Strobe 1 length (medium bit)	0	Read/Write	Read
[6]	Wait state1 (strobe 1)	Strobe 1 length (low bit)	1	Read/Write	Read
[5]	Wait state 3 (strobe 0)	Strobe 0 length (high bit)	0	Read/Write	Read
[4]	Wait state 2 (strobe 0)	Strobe 0 length (medium bit)	0	Read/Write	Read
[3]	Wait state 1 (strobe 0)	Strobe 0 length (low bit)	1	Read/Write	Read
[2]	First priority	Priority modes	1	Read/Write	Read
[1]	Bus error	Application flag error	0	Read/Clear	Read(0 in HOM)
[0]	Mode	SAM or HOM	1 (HOM)	Read	Read

The strobe frequency can go from 1/2x to 1/9x of the DSP clock frequency.

Table 47. Wait State Strobe Frequency NIL

Number of Wait States	Strobe Period	DSP Max Frequency (MHz)	Strobe Frequency (MHz)
0	DSP clk/2	200	100
1	DSP clk/3	200	66
2	DSP clk/4	200	50
3	DSP clk/5	200	40
4	DSP clk/6	200	33
5	DSP clk/7	200	28
6	DSP clk/8	200	25
7	DSP clk/9	200	22

3.2.4 RNG CLOCKS

The RNG module is divided into four principal blocks:

- RING oscillators
- RNG generator
- State machine
- Output and input registers

The random numbers are accessible to the application in a 32-bit read-only register (RNG_OUT). Once the register is read, the RNG module immediately generates a new value. If no RNG read is performed, the RNG module goes idle (IDLERNG) after a maximum of 2^{24} cycles of the input system clock.

In this state the ring oscillators and RNG generator are completely stopped, while only the internal state machine stays active, clocked by the input system clock.

There are three ways to reduce RNG module consumption:

- Total RNG shutdown (automatic)
- Partial RNG shutdown
- Total RNG shutdown (reset RNG)

Total RNG Shutdown

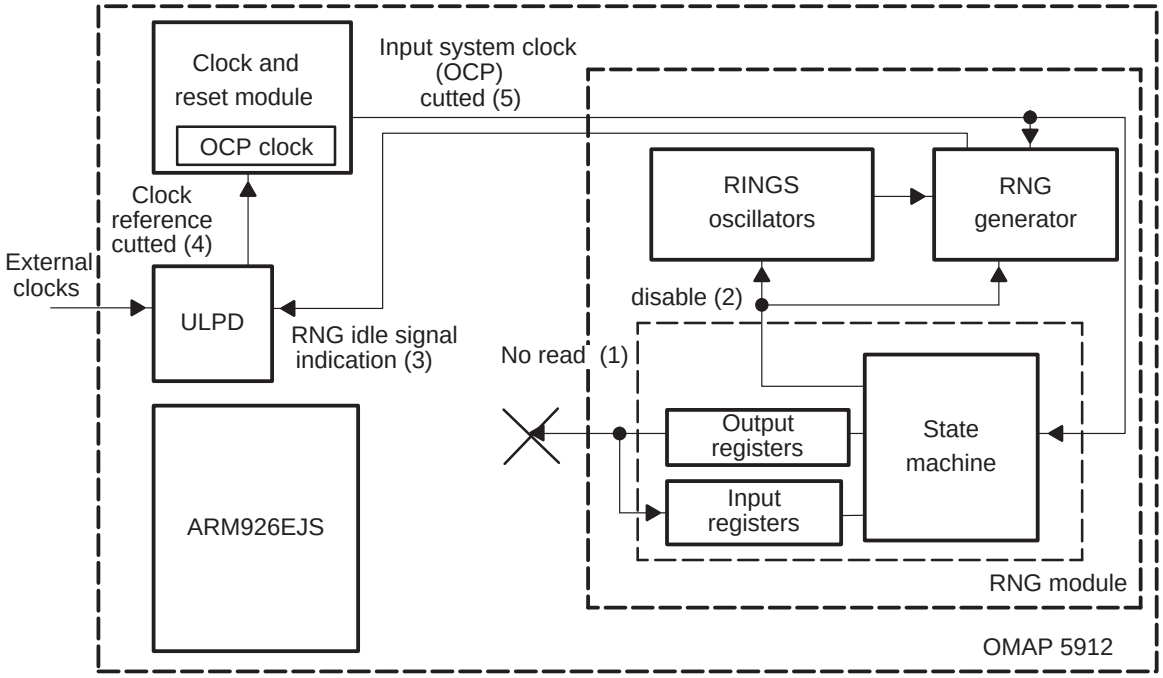
The best and safest way to go into sleep mode is to stop reading the last generated random number. After a maximum of 2^{24} cycles of the input clock, the state machine is in its IDLERNG state. This requires that the input clock be present so that the state machine can jump to its different state. At this state, the rings and the 24-bit counter are off (if bit 0 of RNG_MASK is 1). The RNG can be kept in this state indefinitely and as soon as a random number is read, the state machine wakes up. When the state machine is in its IDLERNG state, a signal at the boundary goes low, indicating that the input clock can be cut off. There is no need to perform a soft reset.

The counter, being synchronous with a clock derived from the input system clock, is still incrementing if the autoidle bit of the module is not set by writing 1 in the bit 0 of the RNG_MASK register (offset address 0x40).

Because all RNG activities are shut off, this is the best method for RNG power consumption reduction. It takes a maximum of 2^{24} cycles to reach the IDLERNG state where the RNG oscillator is shut off.

The input clock can be restarted anytime and no reset is required to restart RNG module.

Figure 22. Total Automatic RNG Shutdown

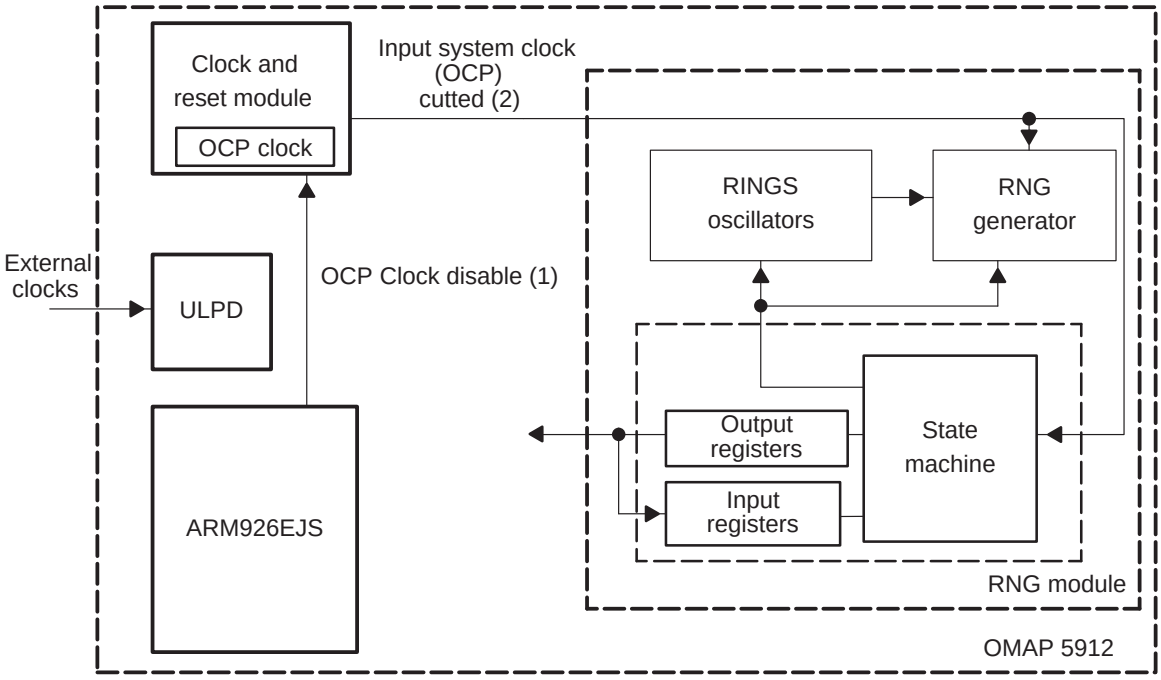


Partial RNG Shutdown: Input Clock Cut Off

The second safest wayto reduce RNG module consumption is to cut the input clock (OCP). The state machine does not jump any more to the different states, the entropy is preserved in the system, and cutting the clock does not affect the randomness. The input clock can be restarted anytime and no reset is needed to correctly restart the RNG module.

Because the state machine can be in any state when the input clock is cut off, one of the rings can still be on. The state machine does not reach the IDLERNG anymore, so the ring is still on. Hence, there is a penalty on power consumption reduction.

Figure 23. Partial RNG Shut Down



Total RNG Shutdown: Reset RNG Module

If an application no longer needs the RNG functions and needs to go into deep sleep mode without waiting for a transition to the IDLERNG state, the application can write 0 in the field RESET_COUNT of the RNG_CONFIG and the input system clock can be switched off directly.

If this scenario is completed, and if a random number is needed afterwards, it is then best to perform a soft reset, because randomness cannot be ensured. The penalty is that again 2^{25} cycles are needed before the first random number is ready. This method must be done as a last resort when RNG is not used anymore.

Enter secure mode through normal entry sequence, and then activate the RNG test mode by writing 1 into the RNG_CTRL (0x08) bit number 1.

This unlocks access to the RESET_COUNT field (bits 11−6) of RNG_CONFIG register (0x28).

At this point you have broken the RNG functionality and the RNG cannot be used again for security functions unless a proper soft reset is performed with bit 11 set to 1.

Fill the field RESET_COUNT (bits 11−6) with zeros. The ring oscillators are OFF.

Stop the OCP clock in OMAP and the RNG is OFF.

3.2.5 Externals Clocks

Clocks to external interfaces can be requested on demand. See Chapter 5, *Initialization*, for additional information.

3.3 Dynamic Management

3.3.1 Autogating Mechanisms

Some modules provide internal mechanisms to allow automatic cutting of their internal clocks when they are not required; in this way, the power is decreased. With this automatic clock gating, the internal clock can be halted when no external request is active or when the last task is ended.

These mechanisms can be enabled or disabled by software configuration; when enabled, the clock gating is completely transparent to the user.

OMAP3.2 Autogating

Table 48 lists the OMAP3.2 modules that provide the autogating clock feature along with the relevant bit register to enable or disable the feature.

Table 48. OMAP3.2 Modules With Clock Autogating Enable Feature

Module	Register Name	Autogating Enable Bit (Register Field)
System DMA	DMA_GCR	Bit 3, CLK_AUTOGATING_ON
Traffic controller (TC) OCP−T1, OCP−T2	CONFIG_REG	Bit 0, AUTO_GATED_CLK
EMIFS	CONFIG_REG	Bit 2, PWD_EN
EMIFF	SDRAM_CONFIG_2	Bit 2, SD_AUTO_CLK

OMAP5912 Peripherals Autogating

Table 49 lists the OMAP5912 peripherals that provide the autogating clock feature along with the relevant bit register to enable or disable the feature.

Table 49. OMAP5912 Peripherals With Clock Autogating Enable Feature

Module	Register Name	Autogating Enable Bit (Register Field)
32-kHz watchdog	WD_SYSCONFIG	Bit 0, AutoIdle
Secure watchdog	WD_SYSCONFIG	Bit 0, AutoIdle
RNG	RNG_MASK	Bit 0, AutoIdle
DES3DES	DES_MASK	Bit 0, AutoIdle
SHA-1/MD5	SHA_MASK	Bit 0, AutoIdle
OMAP5912 configuration	MOD_CONF_CTRL_1	Bit 26, Enable for the SSI interconnect Bit 25, Enable for the OCP interconnect
MPU level 2 interrupt handler	OCP_CFG	Bit 0, AutoIdle
DSP level 2.1 interrupt handler	OCP_CFG	Bit 0, AutoIdle
NAND flash controller	NND_SYSCFG	Bit 0, AutoIdle
SPI	SPI_SCR	Bit 0, AutoIdle
Dual-mode timer [7:0]	TIOCP_CFG	Bit 0, AutoIdle
UART [3:1]	SYSC	Bit 0, AutoIdle
GPIO [4:1]	GPIO_SYSCONFIG	Bit 0, AutoIdle
GDD	GDD_GCR	Bit 3, CLOCK_AUTOGATING_ON

Notes: 1) At reset, the autogating enable bit is disabled to minimize the power consumption. It is the software responsibility to enable the clock autogating feature for the above modules.

2) When not used, the STI module can be totally shut down by software resetting the bit 15 (STIEN field) of the STI enable register (STI_ER).

3) When not used, the GDD module can be totally shut down by software setting the bit 0 (SWITCH_OFF field) of the GDD global control register (GDD_GCR).

MGS3/DSP Autogating

The EMIF is able to disable some internal and external clocks depending on its activity. This autogating feature can be managed by software via the power management register.

The power management register controls whether or not the EMIF tries to disable internal and external clocks when not needed, to reduce power consumption. Some of these options may not work at the maximum operating frequency or with all memory devices. Table 50 describes the fields in the power management control register.

Table 50. Power Management Control Register

Field	Description
MEMIPM	Memory contoller internal clock power management. MEMIPM = 1 Each internal memory controller clock is disabled when no access is pending or active on that memory interface. MEMIPM = 0. All memory controller clocks are disabled only when the EMIF is in the IDLE or HOLD state.
CPUIPM	CPU interface internal clock power management. CPUIPM = 1 The internal clock driving the CPU/DMA pipeline registers and stall logic is disabled when no request input is active, no request is pending, and no access is active in the EMIF. CPUIPM = 0 The internal clock driving the CPU/DMA pipeline registers and stall logic is disabled only when the EMIF is in the IDLE or HOLD state.
MAINIPM	Main internal clock power management. MAINIPM = 1: The internal clock driving the arbitration, scheduler, external address, data logic, and so on, is disabled when no request is pending and no access is active in the EMIF. MAINIPM = 0: The internal clock driving the arbitration, scheduler, external address, data logic, and so on, is only disabled when the EMIF is in the IDLE or HOLD state.
SBSXPM	SBSRAM external clock power management. SBSXPM = 1: The SBSCLK_R clock is disabled by the SBSCEN bit whenever no SBSRAM accesses are pending. The EMIF ensures that the memory device is clocked at least twice before the access starts. SBSRPM = 0: The SBSCLK_R clock is disabled only by the SBSCEN TIPB bit.

3.4 ULPD Power Modes Management

3.4.1 Introduction

The ULPD is a power management module running at 32 kHz.

The ULPD has two main clock sources: a 32-kHz clock and a system clock of medium frequency (12 MHz, 13 MHz, and 19.2 MHz).

The ULPD provides system reference used by all clock domains.

The ULPD internal state machine provides three system power modes: awake, big sleep, and deep sleep. These three modes determine global clock activity and overall consumption in OMAP5912.

Idle states are defined for each clock domain or subdomain.

3.4.2 ULPD Mode Descriptions

The following describes the three system power modes:

- In deep sleep (lowest-consumption mode):
  - Only Clk32k clock, used by the ULPD and RTC modules, is on.
  - The system clock is off.
- In big sleep (low-consumption mode):
  - Clk32k and the system clocks, used by ULPD, RTC modules, and external peripherals, are on.
  - The OMAP5912 clock reference is off.
- In awake mode:
  - Clk32k and system clocks are on.
  - OMAP5912 clock reference provide by ULPD is on.

From a consumption point of view, it is best to go into idle mode (standby MPU signal activated) whenever possible.

3.4.3 ULPD Mechanisms Description

To handle power mode transition, the ULPD module controls principally:

- Standby wait for interruption signal received from the ARM926EJS processor.This signal is generated from specific MPU instruction decoding (software running).
- Idle request signals sent to all OMAP5912 clock domains
- Idle acknowledge signals from all OMAP5912 clock domains
- Wake-up signal (from external or internal requests)

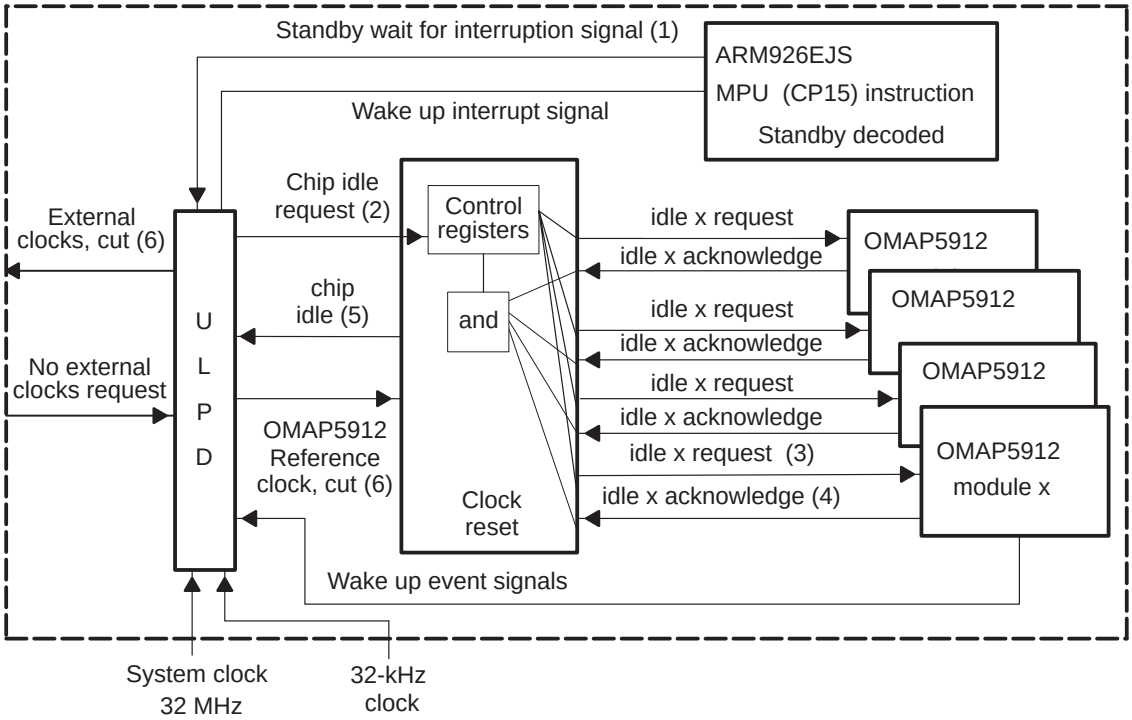
The different clock requests to the ULPD are described in Table 4.

Software is responsible for requesting the transition from the awake mode to the deep sleep or big sleep modes.

All OMAP clock domains determine whether to validate transitions (with acknowledge signals) to the next state according to their activity.

The main steps in the transition from the awake mode to other modes are shown in Figure 24.

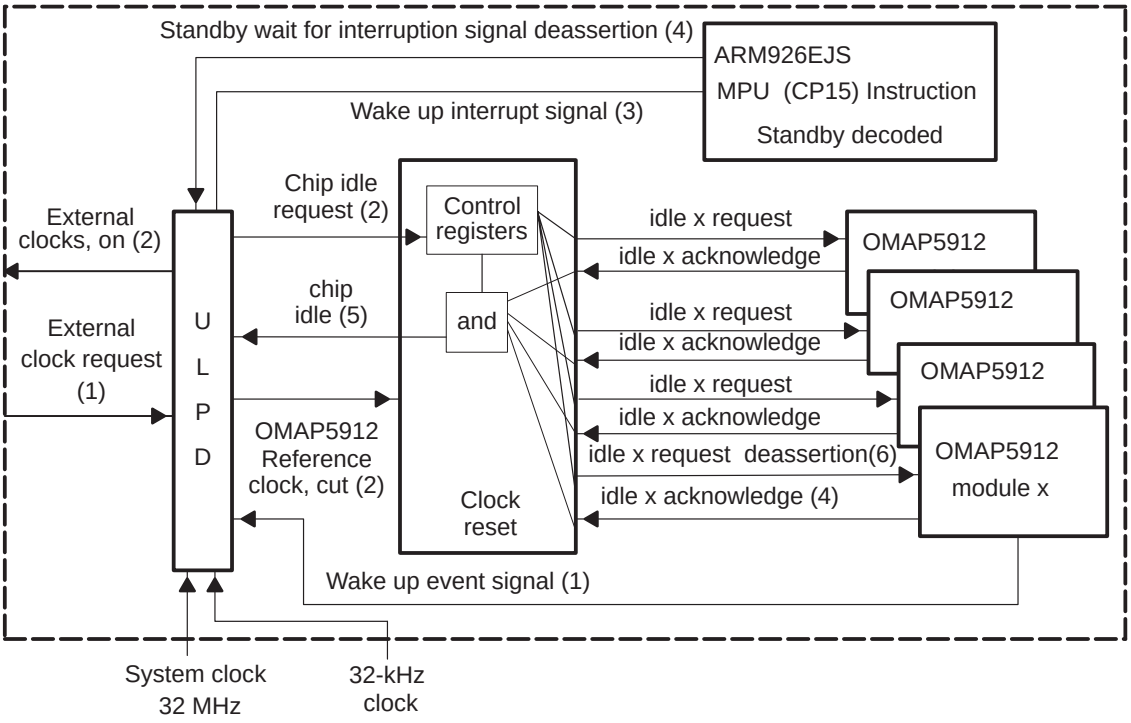
Figure 24. OMAP5912 Shutdown Request Management



During the low-consumption states of deep sleep and big sleep, the ULPD scans hardware events, interrupts, and external clock requests to restore the clock reference system (or/and external clocks) and wake up the ARM926EJS processor.

The main steps in the transition from the deep sleep or big sleep modes to the awake mode are shown in Figure 25.

Figure 25. OMAP5912 Wake-Up Management



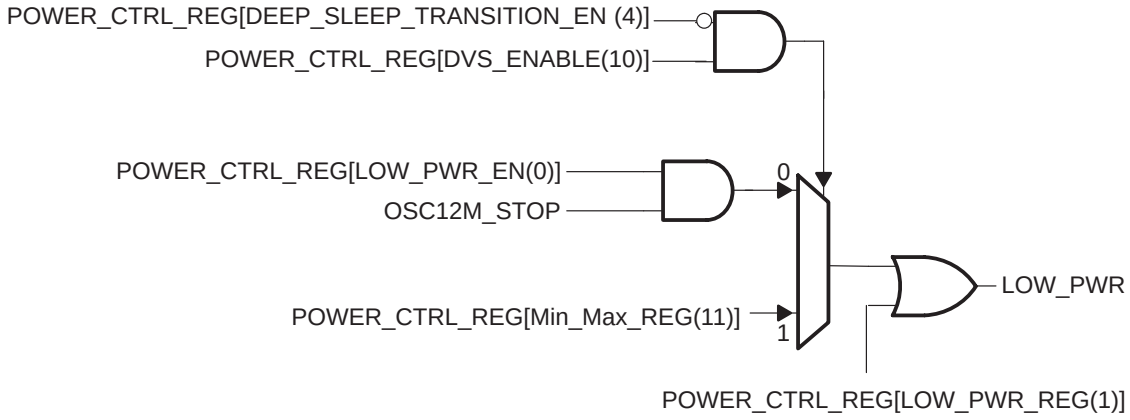
3.4.4 Control of External Clock and Voltage Supplies

The ULPD provides the $\overline{\text{LOW\_PWR}}$ and LOW_PWR signals to control the activation or the shutdown of the external clock and the core voltage supplies.

- $\overline{\text{LOW\_PWR}}$ indicates to external devices that the system clock can be shut down (in external clock mode).
- LOW_PWR drives the external core voltage supply in low voltage (1.1 V) operations. The behavior of the LOW_PWR signal is set by software and can be configured to allow two types of operation:
  - Reduction of leakage current in deep sleep mode
  - Low-voltage operation at reduced clock frequency (dynamic voltage scaling)

To enable the operation at reduced clock frequency under external low-voltage supplies, the LOW_PWR signal is set by software through the ULPD power control register (POWER_CTRL_REG bits 1, 4, 10, and 11). Figure 26 summarizes the LOW_PWR signal behavior.

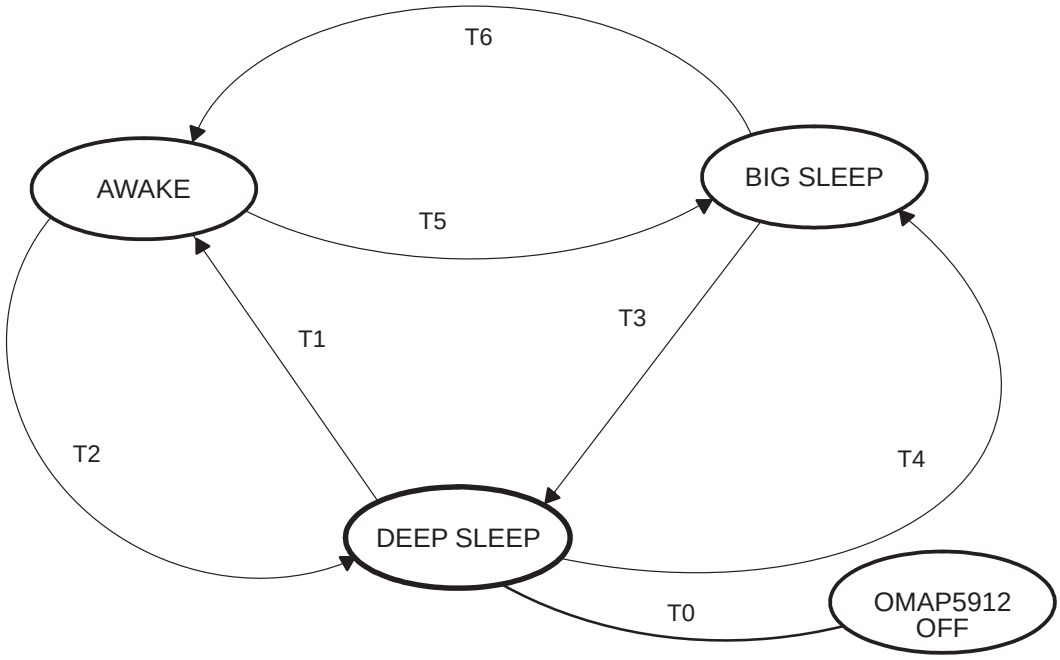
Figure 26. Software Control of the LOW_PWR Signal



3.4.5 Transitions Between ULPD Modes

Figure 27 describes the transition flow between the ULPD modes.

Figure 27. Transition Flow



T0: Power up

T1: Wake-up request from OMAP3.2 or from peripherals or after power up context

T2: Idle request from OMAP3.2, externals clocks not requested, and deep sleep authorized

T3: No wake-up request from OMAP3.2 nor from external peripherals, and deep sleep authorized

T4: External clock request

T5: Idle request from OMAP3.2 but either at least one active external clock request or deep sleep transition not authorized

T6: Either wake-up request from OMAP3.2 or from external peripherals

3.5 Power Domain Management

3.5.1 RTC Domain Management

The RTC power domain allows the user to achieve minimum consumption during the OFF state of OMAP5912.

In the OFF state only the RTC power domain OMAP 5912 is supplied. The DSP and MPU domains are switched off (all power supplies associated equal 0 volts).

The RTC power domain supplies only real-time clock functions and associated reset controls.

The user must control the ON_OFF signal during MPU and DSP domain switch-off and switch-on.

Figure 28 describes the OFF state of OMAP5912, and Figure 29 describes the ON state.

Figure 28. OMAP 5912 State OFF

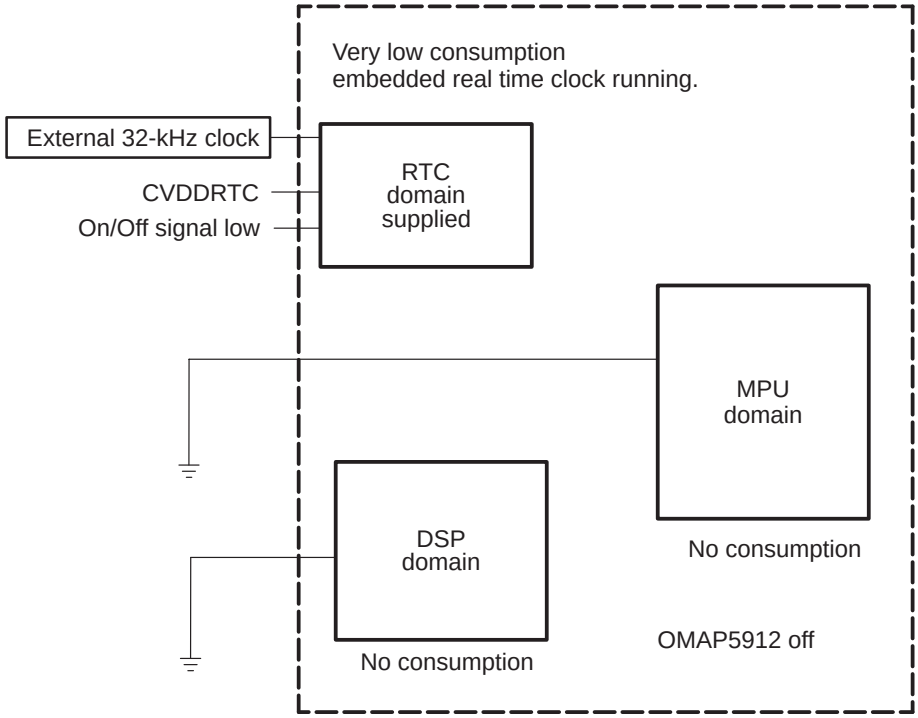
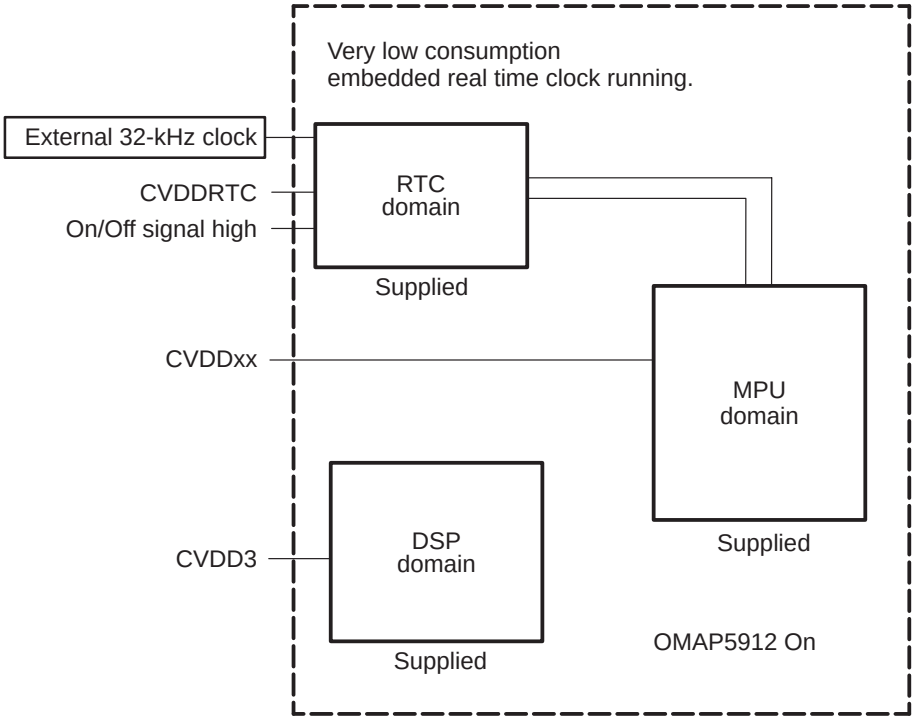


Figure 29. OMAP 5912 State ON



3.5.2 DSP Domain Management

The DSP power domain is surrounded by an isolation wrapper to provide proper electrical isolation and the appropriate interface state at the boundary of the MPU and DSP power domains.

The assumption for OMAP5912 is that the power domains are connected exclusively to external power supplies (not necessarily dedicated). No e-LDO and no power switch are implemented in OMAP5912 to supply the MPU and DSP domains.

When the DSP is in isolation mode, it is possible to cut its dedicated power supplies. The isolation wrapper is controlled by bit[12] of POWER_CTRL_REG in the ULPD register file. Table 51 describes the isolation control bit.

Table 51. DSP Isolation Control

12	ISOLATION_CONTROL	0: Electrical isolation inactive 1: Electrical isolation active Reset of this bit is done upon power up reset only (PWRON_RESET).	R/W	0x0

4 Dynamic Voltage Scaling

To minimize the leakage current when OMAP5912 is in deep sleep mode, decrease the external supply voltages once the deep sleep state is validated. The dynamic voltage scaling (DVS) feature enables the operation at reduced clock frequency when the external supply voltages are low.

The OMAP5912 provides two signals that control the core voltage supplies and activation or shutdown of the external clock, depending on the clock mode (oscillator or external). These two signals, LOW_PWR and LOW_PWR, behave similarly except that they do not have the same polarity, and LOW_PWR can be controlled by software, whereas LOW_PWR cannot.

4.1 Low Voltage With Chip Totally Shut Down

4.1.1 Oscillator Clock Mode

In deep sleep mode, to minimize the power consumption caused by leakage currents, decrease the external supply voltages. To enable this feature, the following conditions must be met:

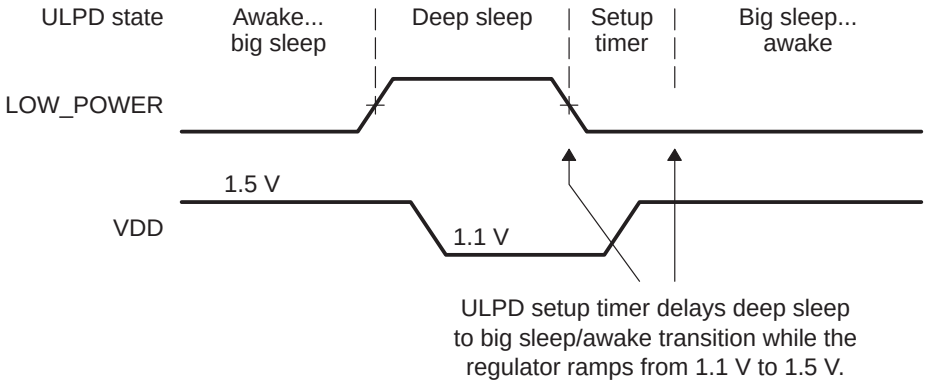
- Set the ULPD POWER_CTRL_REG[0] bit (LOW_PWR_EN field) to 1 to enable the low-power feature.
- Set the ULPD POWER_CTRL_REG[4] bit (DEEP_SLEEP_TRANSITION_EN field) to 1 to enable transition to the deep sleep state or to reset the POWER_CTRL_REG[10] bit to disable the DVS feature.

If the above register settings are made, the external signal LOW_PWR switches to active high whenever the ULPD enters the deep sleep state. In this way, the external core voltage supply can be driven at low voltage, reducing the leakage current.

When the ULPD exits the deep sleep state, the signal LOW_PWR switches back to inactive low and the external core voltage supply ramps up to nominal 1.6 V.

At reset, the LOW_PWR feature is disabled and POWER_CTRL_REG[0] is set to 0. The LOW_PWR signal is inactive low, which indicates nominal voltage requirement. Figure 30 describes the behavior of the signal.

Figure 30. Behavior of LOW_PWR Signal



4.1.2 External Clock Mode

The $\overline{\text{LOW\_PWR}}$ signal is used in external clock mode. When active low, $\overline{\text{LOW\_PWR}}$ indicates to external devices that the input system clock (system clock) can be shut down. It also indicates that the external core voltage supply can be lowered to 1.1 V to reduce the chip leakage-current consumption.

The $\overline{\text{LOW\_PWR}}$ signal is asserted low when the ULPD enters the deep sleep state (except at power-up reset) and is released upon deep sleep exit (except at power-up reset). At power-up reset, $\overline{\text{LOW\_PWR}}$ is reset to its inactive value (high). Figure 31 describes the assertion of the signal, and Figure 32 describes the release of the signal.

Figure 31. Assertion of the $\overline{\text{LOW\_PWR}}$ Signal

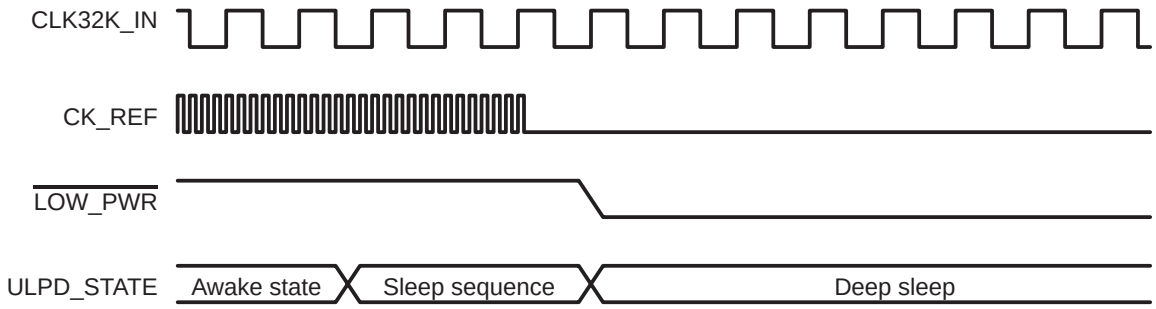
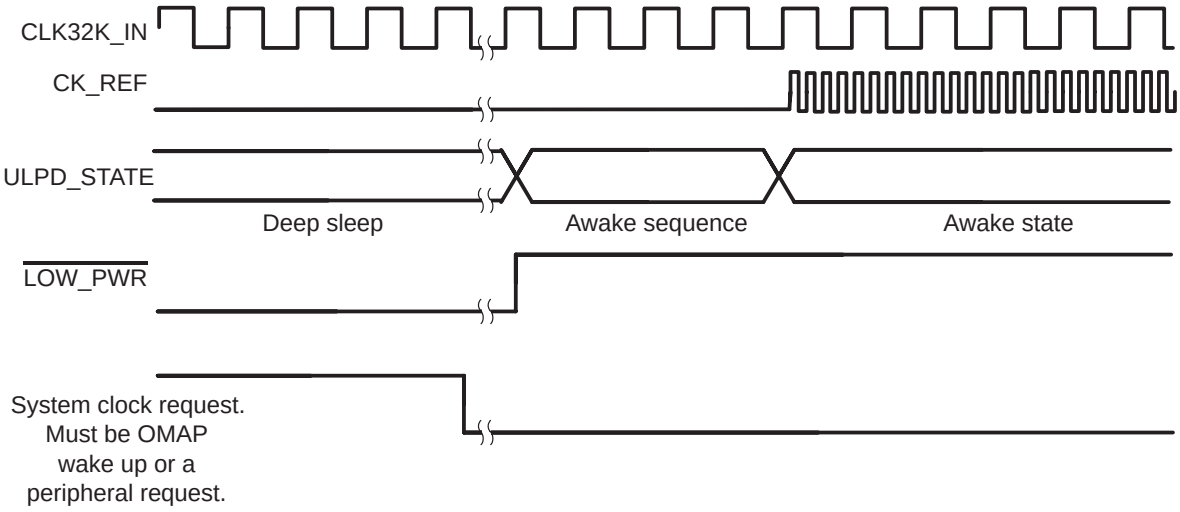


Figure 32. Release of the $\overline{\text{LOW\_PWR}}$ Signal



4.2 Low Voltage With Chip Running at Reduced Clock Frequencies

If the application does not require many MIPS, operate at low voltage and reduced frequency. For instance, at a reduced core voltage of 1.05 V, the MPU frequency can be scaled down to 80 MHz and the traffic controller frequency to 40 MHz.

5 OMAP5912 Power Modes

The global OMAP5912 system power mode depends on the authorized combinations of MPU and DSP domain states.

On OMAP5912, the power management policy (software) is run exclusively on the MPU. The DSP domain behaves as a slave, so the DSP domain power state transitions are performed under the control of the MPU software. However, the transition from active to inactive for the DSP domain state is an exception to this rule. This transition is controlled by DSP software (idle instruction).

The power state transition of the MPU domain is controlled by software and hardware. The hardware part is composed mainly of the ULPD module FSM, which performs automatic transition between some of the MPU domain states.

Table 52 lists the eight possible system power modes for OMAP5912.

Table 52. Global OMAP5912 System Power Modes

Global OMAP5912 Power Mode	Chip State	MPU Domain State	DSP Domain State	Input System Clock	32-kHz Clock
ACTIVE MODE	1.1	Active state	Active state	On	On
	1.2	Active state	Inactive state	On	On
	1.3	Active state	Sleep state	On	On
BIG SLEEP MODE	2.1	Inactive state	Inactive state	On	On
	2.2	Inactive state	Sleep state	On	On
DEEP SLEEP MODE	3.1	Pending state	Pending state	On or Off	On
	3.2	Pending state	Sleep state	On or Off	On
OFF MODE	0	Sleep state	Sleep state	Off	On or Off

The possible MPU system (ARM926EJS, TC, DMA, and memory interfaces—all OMAP5912 peripherals) states are described in Table 53.

Table 53. MPU Domain States

MPU Domain State	CLOCKS						ULPD State	Power Supply Voltage
	OMAP3.2 Input Clock	OMAP3.2 PLL	MPU and TC Clocks	32 kHZ Clock	ULPD PLL	Peripheral Clocks		
Active	On	On or Off	On or Off	On	On or Off	On or Off	Awake	1.6 or 1.1 V
Inactive	Off	Off	Off	On	Off	Off	Big Sleep	1.6 or 1.1 V
Pending	Off	Off	Off	On	Off	Off	Deep Sleep	1.1 V
Sleep	Off	Off	Off	Off	Off	Off	Off	0 V

State retention is ensured when the MPU subsystem is in active, inactive, and pending states.

In sleep state, there is no state retention.

The possible DSP system (MGS3) states are described in Table 54.

Table 54. DSP Domain States

DSP Domain State	CLOCKS		MGS3 Reset	DSP Isolation Wrapper	Power Supply Voltage
	MGS3 Input Clock	MGS3 Sub-domain Clocks			
Active	On	On or Off	Released	Deactivated	1.6 or 1.1 V
Inactive	Off	Off	Released	Deactivated	1.6 or 1.1 V
Pending (see Note)	Off	Off	Released	Deactivated	1.1 V
Sleep	Off	Off	Asserted	Activated	0 V

Note: The DSP subsystem is in pending state when it was set to inactive state before the MPU subsystem goes into pending state (with the ULPD in deep sleep state).

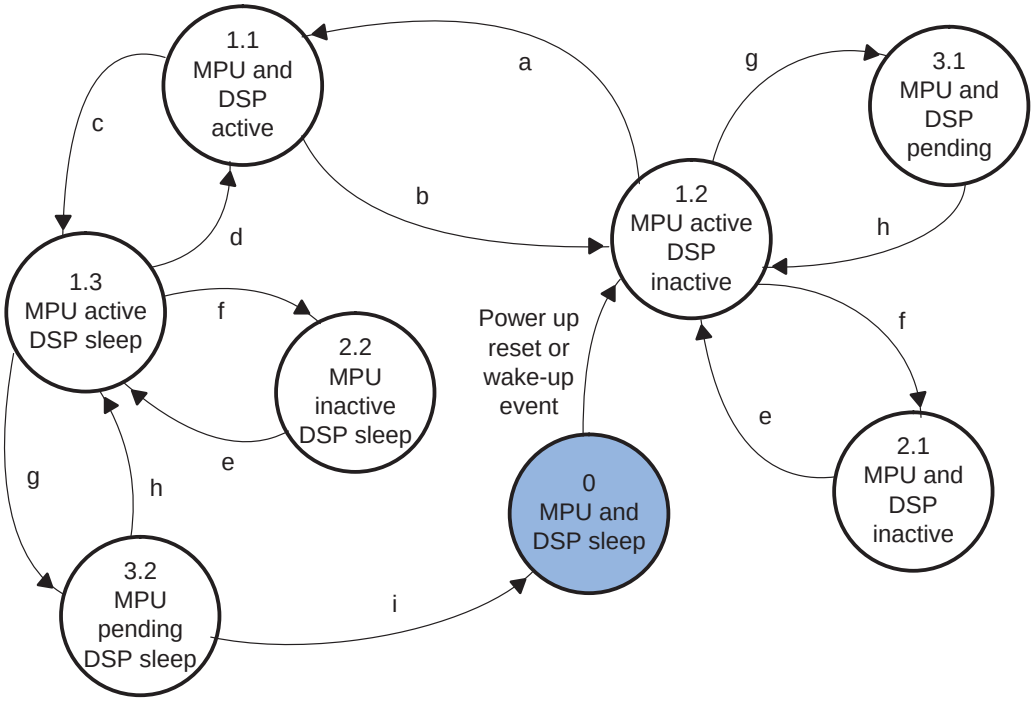
State retention is ensured when the DSP subsystem is in active, inactive, and pending states.

In sleep state, there is no state retention.

5.1 OMAP5912 Power Mode Transitions

Figure 33 describes the OMAP5912 power mode transitions. Note that the initial state is state 0 (the only shaded state).

Figure 33. Power Mode Transitions



1) DSP wake-up from inactive state

   Wake-up is automatically handled by the OMAP3.2 CLKRST module upon any unmasked interrupts or reset.

   a) Any unmasked interrupt that fires up is asynchronously propagated to the OMAP3.2 CLKRST module as a wake-up event.

   b) The CLKRST module detects the wake-up event.

   The OMAP3.2 CLKRST module automatically restarts the DSP input clock DSP_CK and the DSP subdomain clock.

2) DSP active to inactive state transition

   a) DSP power domain is in active state.

   b) Either the MPU requests the DSP to enter idle mode for inactive state via mailbox or shared memory, or the DSP OS enters the IDLE process.

   c) The DSP programs the ICR register of the DSP to shut down all megacell subdomain clocks.

   d) The DSP programs registers DSP_IDLECT1 and/or DSP_IDLECT2 of the OMAP3.2 CLKRST module to set the state of the OMAP3.2

DSP clock subdomains (CLKM2) (these belong to the MPU power domain and must be requested by the MPU).

e) The DSP masks/unmasks interrupts in the DSP interrupt handler to ensure a wake-up path.

f) The DSP executes the IDLE instruction.

g) The DSP input clock (DSP_CK) is shut down automatically by the OMAP CLKRST module.

h) The DSP power domain is in inactive state. The MPU can detect when the transition is completed by monitoring the IDLE_DSP bit in the ARM_SYSST register of the OMAP3.2 CLKRST module.

3) DSP active to sleep state transition

a) The MPU requests the DSP to enter into DSP idle mode for sleep state via mailbox or shared memory.

b) DSP programs the ICR register of DSP to shut down *all* megacell subdomain clocks.

c) DSP programs registers DSP_IDLECT1 and/or DSP_IDLECT2 of the OMAP3.2 CLKRST module to set the state of the OMAP3.2 DSP clock subdomains (CLKM2) (these belong to the MPU domain and are requested by MPU).

d) The DSP masks all interrupts in the DSP interrupt handler. There is no direct wake-up path through DSP interrupts.

e) DSP executes an IDLE instruction.

f) The DSP input clock (DSP_CK) is shut down automatically by the OMAP3.2 CLKRST module.

g) The DSP power domain is in inactive state. The MPU can detect when the transition is completed by monitoring the IDLE_DSP bit in the ARM_SYSST register of the OMAP3.2 CLKRST module.

h) The MPU puts the DSP into reset state by asserting the DSP_EN and DSP_RST bits into the ARM_RSTCT1 register of the CLKRST module.

i) The MPU disables the DSP clock by deasserting the EN_DSPCK bit in the ARM_CKCTL register of CLKRST.

j) The MPU programs the ULPD POWER_CTRL_REG[12] to assert the ISOLATION_CONTROL bit to activate the DSP domain isolation wrapper.

k) The MPU sets the GPIO that controls the external analog switch to power down the DSP domain. (It may also be done through an SPI/I^2C interface). The DSP domain is in sleep state.

4) DSP wake-up from sleep state

   In this case, the wake-up is handled entirely by the MPU software. There is no wake-up path through DSP interruptions.

   a) The MPU sets the GPIOs that control the external analog switch to power on the DSP domain. (It can also be done through an SPI/I^2C interface.)

   b) The MPU waits during the ramp-up time of the DSP domain VDD.

   c) The MPU programs the ULPD POWER_CTRL_REG[12] to release the ISOLATION_CONTROL bit to deactivate the DSP domain isolation wrapper.

   d) The MPU enables the DSP clock by asserting the EN_DSPCK bit in the ARM_CKCTL register of CLKRST.

   e) The MPU releases the DSP reset state by deasserting the DSP_EN and DSP_RST bits into the ARM_RSTCT1 register of the CLKRST module.

   f) The DSP boots.

5) MPU wake-up from inactive state

   a) Any unmasked interrupt in the MPU interrupt handler is asynchronously forwarded to ULPD: theWAKEUP_$\overline{REQ}$ signal is asserted.

   b) If DSP is also in inactive state, any unmasked interrupt in the DSP interrupt handler is asynchronously forwarded to the ULPD: the WAKEUP_$\overline{REQ}$ signal is asserted.

   c) ULPD detects wake-up request: WAKEUP_$\overline{REQ}$ or clock request at ULPD

   d) ULPD FSM automatically moves into its awake mode and enables the OMAP3.2 input clock.

   e) The DPLL clock restarts automatically.

   f) The MPU and TC clocks restart automatically.

6) MPU active to inactive state transition

   a) The MPU places the external SDRAM in self-refresh mode (program EMIFF register).

   b) The MPU programs the ARM_IDLECT1, ARM_IDLECT2, and ARM_IDLECT3 registers to ensure that all OMAP MPU and TC subdomain clocks are shut down.

c) The MPU programs the IDLDPLL_ARM and IDLIF _ARM bits of ARM_IDLECT1 to allow TC and DPLL to go into idle mode.

d) The MPU programs the MPU interrupt handlers to mask/unmask interrupts and ensure the wake-up path.

e) The MPU programs the SOFT_DISABLE_REQ_REQ register of ULPD to ensure a wake-up path.

f) The MPU clears POWER_CTRL_REG[4] of ULPD (ULPD big sleep mode)

g) MPU executes the STANDBYWFI instruction.

h) All OMAP MPU and TC clock domains are automatically shutdown by the CLKRST module.

i) DPLL is automatically put in idle by the CLKRST module.

j) CLKRST asserts the CHIP_IDLE signal to the ULPD.

k) The ULPD disables the OMAP3.2 input clock and asserts the CHIP_$\overline{\text{WAKEUP}}$ signal.

l) The ULPD FSM automatically moves into sleep mode. The MPU domain is in inactive state.

7) MPU active to pending state transition

a) The MPU places the external SDRAM in self-refresh mode (program EMIFF register).

b) The MPU programs the ARM_IDLECT1, ARM_IDLECT2, and ARM_IDLECT3 registers to ensure that all OMAP MPU and TC subdomain clocks are shut down.

c) The MPU programs the IDLDPLL_ARM and IDLIF_ARM bits of ARM_IDLECT1 to allow TC and DPLL to go into idle mode.

d) The MPU programs the MPU interrupt handlers to mask/unmask interrupts and ensure the wake-up path.

e) The MPU programs the SOFT_DISABLE_REQ_REQ register of the ULPD to ensure the wake-up path.

f) The MPU sets bit POWER_CTRL_REG[4] of the ULPD (ULPD deep sleep mode).

g) The MPU sets bit POWER_CTRL_REG[0] of the ULPD (LOW_PWR feature enabled).

h) The MPU sets/clears bit POWER_CTRL_REG[9] of the ULPD (oscillator control).

i) The MPU programs the ULPD registers SETUP_ANALOG_CELL2 and SETUP_ANALOG_CELL3 with appropriate stabilization time for the external regulator and the oscillator.

j) The MPU executes the STANDBYWFI instruction.

k) All OMAP MPU and TC clock domains are automatically shutdown by the CLKRST module.

l) The DPLL is automatically put in idle by the CLKRST module.

m) CLKRST asserts the CHIP_IDLE signal to the ULPD.

n) The ULPD disables the OMAP3.2 input clock and asserts the CHIP_$\overline{\text{WAKEUP}}$ signal.

o) The ULPD FSM automatically moves into deep sleep mode.

p) If POWER_CTRL_REG[9] of the ULPD was set, the oscillator is stopped.

q) The ULPD asserts the signal LOW_PWR. This drives the external MPU domain regulator into low-voltage operations at 1.1 V. The MPU domain is in pending state.

r) If the DSP domain was set to inactive state before initiating the MPU domain transition, the final state of the DSP domain is also pending.

8) MPU wake-up from pending state

a) Any unmasked interrupt in the MPU interrupt handler is asynchronously forwarded to ULPD: WAKEUP_$\overline{\text{REQ}}$ signal is asserted.

b) If DSP is also in pending state, any unmasked interrupt in DSP interrupt handler is asynchronously forwarded to ULPD: WAKEUP_$\overline{\text{REQ}}$
signal is asserted.

c) The ULPD detects wake-up requests: WAKEUP_$\overline{\text{REQ}}$ or clock request at ULPD.

d) The ULPD exits its deep sleep mode and releases the LOW_PWR signal.

e) The MPU domain external regulator ramps up.

f) The ULPD FSM automatically moves into its awake mode. Delays programmed in SETUP_ANALOG_CELL2 and SETUP_ANALOG_CELL3 are inserted during the deep sleep to awake transition to allow the oscillator and voltage supplies to stabilize.

g) The ULPD enables the OMAP3.2 input clock.

h) The DPLL clock restarts automatically.

i) The MPU and TC clocks restart automatically.

9) The OMAP5912 chip to sleep state transition

For OMAP5912 the MPU domain sleep state is not a true domain state in the sense that it cannot be entered or exited without the intervention of another device. An external device must shut down the MPU domain regulator.

The wake-up can be performed only through a power-up sequence by asserting the OMAP5912 power up reset.

6 OMAP5912 Power Management Software User Guide

10) Set RNG_CONF_IDLE_MODE, bit 6 of the RESET_CONTROL register, to 0 so that there is no waiting for RNG to enter the idle state to enter CHIP_IDLE:

- Command WR32 0xFFFE 1140 0x0000 003F

Note: By clearing bit 6 in RESET_CONTROL, the RNG idle acknowledge is ignored. This implies that the RNG will not prevent the chip from entering idle but the internal ring oscillators may still consume power. If bit 6 in RESET_CONTROL is not cleared, then the RNG must be configured to shut down its internal ring oscillators before allowing the system to idle. This implies that the latency between awake and idle states is longer.

11) Disable the MPU watchdog timer by writing the sequence F5, A0 in the TIMER_MODE register:

- Command WR32 0xFFFE C808 0x0000 00F5
- Command WR32 0xFFFE C808 0x0000 00A0

12) Disable the DSP watchdog timer by writing the sequence F5, A0 in the TIMER_MODE register:

- Command WR16 0xE100 6808 0x00F5
- Command WR16 0xE100 6808 0x00A0

13) Configure the ARM_IDLECT1 register to turn off DPLL1, LCD, DMA, ARMPER, internal timer, MPUI, ARMXOR, and MPU watchdog clocks when the MPU enters idle state:

- Command WR32 0xFFFE CE04 0x0000 17FF

14) Configure the ARM_IDLECT2 register to turn off the external GPIO clock:

- Command WR32 0xFFFE CE08 0x0000 FDFF

15) Configure the ARM_IDLECT3 register to turn off the TC2, TC1, L3OCP clocks when the MPU enters idle state:

    - Command WR32 0xFFFE CE24 0x0000 FFFF

16) Configure the DSP_IDLECT1 register to turn off GPIO, internal timer, MPUI, DSPPER , DSPXOR, and DSP watchdog clocks when the MPU enters idle state:

    - Command WR32 0xFFFE CE84 0x0000 01CF

17) Configure the DSP_IDLECT2 register to turn off the external UART clock:

    - Command WR32 0xFFFE CE88 0x0000 FFF7

18) Configure the TC_EMIF_SLOW_IF_CONFIG_REG to set global power-down enable and IMIF power-down enable bits:

    - Command WR32 0xFFFE CC0C 0x0000 000C

19) Configure the TC_EMIF_FAST_SDRAM_CONFIG_REG to disable the SDRAM clock and put SDRAM FSM into power-down mode:

    - Command WR32 0xFFFE CC20 0x0C00 B07F

Index

T

U